

OTRS 3.0 - Administrator-Handbuch

OTRS 3.0 - Administrator-Handbuch

Copyright © 2003-2011 OTRS AG

René Bakker, Stefan Bedorf, Michiel Beijen, Shawn Beasley, Hauke Böttcher, Jens Bothe, Udo Bretz, Martin Edenhofer, Carlos Javier García, Martin Gruner, Manuel Hecht, Christopher Kuhn, André Mindermann, Marc Nilius, Elva María Novoa, Henning Oswald, Martha Elia Pascual, Thomas Raith, Carlos Fernando Rodríguez, Stefan Rother, Burchard Steinbild, Daniel Zamorano.

Dieses Werk ist geistiges Eigentum der OTRS AG.

Es darf als Ganzes oder in Auszügen kopiert werden, vorausgesetzt, dieser Copyright-Vermerk befindet sich auf jeder Kopie.

UNIX ist ein eingetragenes Warenzeichen von X/Open Company Limited. Linux ist ein eingetragenes Warenzeichen von Linus Torvalds.

MS-DOS, Windows, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows 2003 und Windows Vista sind eingetragene Warenzeichen der Microsoft Corporation. Andere Warenzeichen oder registrierte Warenzeichen: SUSE und YaST von SUSE Linux GmbH, Red Hat und Fedora von Red Hat Inc., Debian von Software in the Public Interest, Inc., Mandrake von MandrakeSoft, SA. MySQL und das MySQL Logo sind eingetragene Warenzeichen Oracle Corporation und/oder verbundener Unternehmen.

Alle Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt und sind möglicherweise eingetragene Warenzeichen.

Die Firma OTRS AG richtet sich im Wesentlichen nach den Schreibweisen der Hersteller. Andere hier genannte Produkte können Warenzeichen des jeweiligen Herstellers sein.

Inhaltsverzeichnis

Vorwort	xi
1. Trouble-Ticket-Systeme - die Grundlagen	1
Was ist ein Trouble-Ticket-System, und warum benötigen Sie eins?	1
Was ist ein Trouble-Ticket?	2
2. OTRS - Open Ticket Request System	3
Grundlagen	3
Features	3
Die wichtigsten neuen Features von OTRS 3.0	5
Neue Features von OTRS 2.4	6
Neue Features von OTRS 2.3	9
Neue Features von OTRS 2.2	10
Neue Features von OTRS 2.1	11
Hardware und Software-Anforderungen	12
Community	13
Kommerzieller Support und Dienstleistungen für OTRS	13
3. Installation	15
Der einfache Weg - Installation fertiger Pakete	15
Installation des RPM auf einem SuSE-Linux-Server	15
Installation von OTRS auf einem CentOS-System	17
Installation von OTRS auf einem Debian-System	17
Installation von OTRS auf einem Ubuntu-System	17
Installation von OTRS auf Microsoft Windows-Systemen	17
Manuelle Installation (Linux, Unix)	17
Vorbereiten der manuellen Installation	17
Installation der für OTRS benötigten Perl-Module	18
Konfiguration des Apache-Webservers	21
Einrichten der Datenbank	22
Einrichten der von OTRS benötigten cron-Jobs	29
Upgrade des OTRS Frameworks	33
Upgrade mit dem Windows-Installer	38
4. Erste Schritte	39
Agenten-Weboberfläche	39
Kunden-Weboberfläche	39
Öffentliche Weboberfläche	40
Die erste Anmeldung	41
Agenten-Weboberfläche - ein Überblick	41
Was verbirgt sich hinter dem Begriff Queue?	43
Benutzereinstellungen	44
5. Der Administrationsbereich von OTRS	46
Grundlagen	46
Benutzer, Gruppen und Rollen	46
Benutzer	46
Gruppen	47
Rollen	51
Kundenbenutzer und Kundengruppen	55
Kunden	55
Kundengruppen	56
Queues	58
Anreden, Signaturen, Anlagen und Antwortvorlagen	60
Anreden	60
Signaturen	61

Anlagen	62
Antwortvorlagen	64
Automatische Antworten	67
E-Mail-Adressen	69
Benachrichtigungen	71
S/MIME	73
PGP	73
Status	74
Die SysConfig (Verwaltung der Systemkonfiguration)	75
Einrichten von Mail-Konten	76
Einkommende Nachrichten filtern	76
Ausführen von automatisierten Jobs mit Hilfe des GenericAgents	79
Administrator-E-Mail	80
Sitzungsverwaltung	81
System Log	82
SQL-Abfragen mit Hilfe der SQL Box	83
Paketverwaltung	83
6. Systemkonfiguration	85
Die Konfigurationsdateien von OTRS	85
Konfiguration des Systems mit Hilfe des grafischen Konfigurations-Frontends	85
7. E-Mails versenden/empfangen	87
E-Mails versenden	87
Via Sendmail (Standard)	87
Via SMTP server/smarthost	87
E-Mails empfangen	87
E-Mail-Konten, die über die OTRS-Oberfläche konfiguriert werden	87
Via Kommandozeilen-Programm und z. B. procmail (otrs.PostMaster.pl)	89
E-Mails via POP3 oder IMAP und fetchmail für otrs.PostMaster.pl empfangen	89
Filterung/Verteilung über PostMaster-Module (für komplexere Verteilungsszenarien)	90
8. Zeitabhängige Funktionen in OTRS	92
Arbeitszeiten, Feiertage und Zeitzonen für das System festlegen	92
Arbeitszeiten	92
Feiertage mit festem Datum	92
TimeVacationDaysOneTime	92
Automatische Ticket-"Freigabe" (unlock)	93
9. Ticket-Verantwortlicher und Beobachten von Tickets	94
Ticket-Verantwortlicher	94
Tickets beobachten	95
10. Einrichtung der PDF-Ausgabe	97
11. Einbinden externer Backends	98
Kundenbenutzerdaten	98
Kundenbenutzer Backend	98
Datenbank (Standard)	98
LDAP	102
Verwenden mehrerer Kunden Backends	105
Backends für die Authentifizierung von Agenten und Kunden	108
Authentifizierungs-Backends für Agenten	108
Authentifizierungs-Backends für Kunden	111
Kunden-Selbstregistrierung anpassen	113
Anpassen der Weboberfläche	113
Kunden-Mapping	113
Anpassen der Kunden-Tabelle in der OTRS-Datenbank	115
12. Status	117

Vordefinierte Ticket-Status	117
Neu	117
Öffnen	117
Warten zur Erinnerung	117
Warten auf erfolglos schließen	117
Warten auf erfolgreich schließen	117
Zusammengefügt	117
Erfolgreich geschlossen	117
Erfolglos geschlossen	117
Anpassen von Ticket-Status	118
13. Anpassen der Ticket-Prioritäten	121
14. Erstellung eigener Themes	122
15. Übersetzung des OTRS-Frontends	123
16. PGP	124
17. S/MIME	128
18. Access Control Lists (ACLs)	132
19. Statistikmodul	136
Bedienung des Moduls durch den Agent	136
Übersicht	136
Ansehen und Generieren von Statistiken	137
Bearbeiten / Neu	140
Importieren	145
Verwaltung des Statistik-Moduls durch den OTRS-Administrator	146
Rechte, Gruppen und Queues	146
Die SysConfig (Verwaltung der Systemkonfiguration)	146
Verwaltung des Statistik-Moduls durch den System-Administrator	146
Datenbanktabelle	147
Auflistung aller Dateien	147
Caching	147
otrs.GenerateStats.pl	147
Automatisiertes Generieren von Statistiken - Cron	148
Statische Statistiken	148
Wiederverwerten alter statischer Statistiken	148
Standardstatistiken	149
20. Zusätzliche Applikationen	150
FAQ	150
21. Leistungsverbesserung	151
OTRS	151
TicketIndexModule	151
TicketStorageModule	151
Tickets archivieren	152
Datenbank	153
MySQL	153
PostgreSQL	153
Webserver	153
Persistente Datenbankverbindungen	154
Vorgeladene Module - startup.pl	154
Perl Module bei Änderung neu laden	154
Die richtige Strategie wählen	154
mod_gzip/mod_deflate	154
22. Datensicherung	155
Datensicherung	155
Wiederherstellung der Daten	155
A. Weitere Quellen	157

Homepage OTRS.org	157
Mailinglisten	157
Fehler berichten	158
Kommerzieller Support	159
B. Referenz der Konfigurationsoptionen	160
Framework	160
Core	160
Core::LinkObject	169
Core::Log	171
Core::MIME-Viewer	173
Core::MirrorDB	174
Core::PDF	175
Core::Package	178
Core::PerformanceLog	180
Core::SOAP	181
Core::Sendmail	182
Core::Session	184
Core::SpellChecker	189
Core::Stats	190
Core::Stats::Graph	191
Core::Time	196
Core::Time::Calendar1	201
Core::Time::Calendar2	204
Core::Time::Calendar3	207
Core::Time::Calendar4	210
Core::Time::Calendar5	214
Core::Time::Calendar6	217
Core::Time::Calendar7	220
Core::Time::Calendar8	223
Core::Time::Calendar9	227
Core::Web	230
Core::WebUserAgent	245
Crypt::PGP	245
Crypt::SMIME	248
Frontend::Admin::AdminCustomerUser	249
Frontend::Admin::ModuleRegistration	249
Frontend::Agent	267
Frontend::Agent::Dashboard	273
Frontend::Agent::LinkObject	277
Frontend::Agent::ModuleMetaHead	277
Frontend::Agent::ModuleNotify	278
Frontend::Agent::ModuleRegistration	280
Frontend::Agent::NavBarModule	286
Frontend::Agent::Preferences	287
Frontend::Agent::SearchRouter	292
Frontend::Agent::Stats	292
Frontend::Customer	296
Frontend::Customer::Auth	305
Frontend::Customer::ModuleMetaHead	313
Frontend::Customer::ModuleNotify	313
Frontend::Customer::ModuleRegistration	314
Frontend::Customer::Preferences	315
Frontend::Public	320
Frontend::Public::ModuleRegistration	320

Ticket	321
Core	321
Core::FulltextSearch	321
Core::LinkObject	322
Core::PostMaster	323
Core::Stats	335
Core::Ticket	337
Core::TicketACL	355
Core::TicketBulkAction	356
Core::TicketDynamicFieldDefault	357
Core::TicketWatcher	364
Frontend::Admin::ModuleRegistration	364
Frontend::Agent	375
Frontend::Agent::CustomerSearch	383
Frontend::Agent::Dashboard	385
Frontend::Agent::ModuleMetaHead	388
Frontend::Agent::ModuleNotify	389
Frontend::Agent::ModuleRegistration	389
Frontend::Agent::Preferences	406
Frontend::Agent::SearchRouter	413
Frontend::Agent::Ticket::ArticleAttachmentModule	413
Frontend::Agent::Ticket::ArticleComposeModule	414
Frontend::Agent::Ticket::ArticleViewModule	414
Frontend::Agent::Ticket::ArticleViewModulePre	415
Frontend::Agent::Ticket::MenuModule	416
Frontend::Agent::Ticket::MenuModulePre	424
Frontend::Agent::Ticket::ViewBounce	429
Frontend::Agent::Ticket::ViewBulk	430
Frontend::Agent::Ticket::ViewClose	434
Frontend::Agent::Ticket::ViewCompose	441
Frontend::Agent::Ticket::ViewCustomer	444
Frontend::Agent::Ticket::ViewEmailNew	445
Frontend::Agent::Ticket::ViewEscalation	448
Frontend::Agent::Ticket::ViewForward	449
Frontend::Agent::Ticket::ViewFreeText	451
Frontend::Agent::Ticket::ViewHistory	458
Frontend::Agent::Ticket::ViewMailbox	458
Frontend::Agent::Ticket::ViewMerge	460
Frontend::Agent::Ticket::ViewMove	462
Frontend::Agent::Ticket::ViewNote	464
Frontend::Agent::Ticket::ViewOwner	471
Frontend::Agent::Ticket::ViewPending	478
Frontend::Agent::Ticket::ViewPhoneInbound	485
Frontend::Agent::Ticket::ViewPhoneNew	489
Frontend::Agent::Ticket::ViewPhoneOutbound	492
Frontend::Agent::Ticket::ViewPrint	496
Frontend::Agent::Ticket::ViewPriority	496
Frontend::Agent::Ticket::ViewQueue	503
Frontend::Agent::Ticket::ViewResponsible	506
Frontend::Agent::Ticket::ViewSearch	513
Frontend::Agent::Ticket::ViewStatus	523
Frontend::Agent::Ticket::ViewZoom	524
Frontend::Agent::TicketOverview	527
Frontend::Agent::ToolBarModule	531

Frontend::Customer	535
Frontend::Customer::ModuleMetaHead	537
Frontend::Customer::ModuleRegistration	538
Frontend::Customer::Preferences	541
Frontend::Customer::Ticket::ViewNew	543
Frontend::Customer::Ticket::ViewPrint	548
Frontend::Customer::Ticket::ViewSearch	548
Frontend::Customer::Ticket::ViewZoom	551
Frontend::Queue::Preferences	555
Frontend::SLA::Preferences	556
Frontend::Service::Preferences	556
C. Danksagungen	557
D. GNU Free Documentation License	559
0. PREAMBLE	559
1. APPLICABILITY AND DEFINITIONS	559
2. VERBATIM COPYING	560
3. COPYING IN QUANTITY	560
4. MODIFICATIONS	561
5. COMBINING DOCUMENTS	562
6. COLLECTIONS OF DOCUMENTS	562
7. AGGREGATION WITH INDEPENDENT WORKS	562
8. Übersetzung	563
9. TERMINATION	563
10. FUTURE REVISIONS OF THIS LICENSE	563
How to use this License for your documents	563

Tabellenverzeichnis

3.1. Die folgenden Perl-Module werden von OTRS benötigt.	19
3.2. Die verschiedenen Skripte für die cron-Jobs von OTRS.	29
5.1. Standardmäßig vorhandene Gruppen in OTRS	48
5.2. Mit OTRS-Gruppen verknüpfte Rechte	50
5.3. Ereignisse für automatische Antworten	68
5.4. Funktion der verschiedenen X-OTRS-Header	77
A.1. Mailinglisten	157

Beispiele

5.1. Aussortierung von Spammails in eine bestimmte Queue	79
7.1. .fetchmailrc	90
7.2. Beispiel-Jobs für das Filtermodul Kernel::System::PostMaster::Filter::Match	90
7.3. Beispiel-Job für das Filtermodul Kernel::System::PostMaster::Filter::CMD	91
11.1. Konfiguration eines Datenbank-Kunden-Backends	98
11.2. Firmen-Tickets mit einem Datenbank-Backend	102
11.3. Konfiguration eines LDAP Kunden-Backends	102
11.4. Verwendung von Firmentickets mit einem LDAP-Backend	104
11.5. Gleichzeitige Einbindung mehrerer verschiedener Kunden Backends	105
11.6. Agentenauthentifizierung gegen ein Datenbank-Backend	108
11.7. Agentenauthentifizierung gegen ein LDAP Backend	108
11.8. Agentenauthentifizierung über HTTPBasic	110
11.9. Agentenauthentifizierung gegen ein Radius-Backend	110
11.10. Kundenauthentifizierung gegen ein Datenbank-Backend	111
11.11. Kundenauthentifizierung gegen ein LDAP Backend	111
11.12. Kundenauthentifizierung über HTTPBasic	112
11.13. Kundenauthentifizierung gegen ein Radius-Backend	113
18.1. ACL, die das Verschieben von Tickets in eine Queue nur bei einer Ticket-Priorität von 5 erlaubt.	132
18.2. ACL, die das Schließen von Tickets in der raw-Queue verbietet und den Schließen-Schalter ausblendet.	132
18.3. ACL, die den Status für alle Agenten entfernt und Ihn nur noch für eine Gruppe zur Verfügung stellt.	133
19.1. Festlegen einer Wertereihe bei einem Element	144
19.2. Festlegen einer Wertereihe bei zwei Elementen	144

Vorwort

Dieses Buch soll das Leben von OTRS-Administratoren und -Neulingen erleichtern.

Dieses Kapitel beschreibt die Installation, Konfiguration und Administration von OTRS. Das erste Drittel des Textes beschreibt die Hauptfunktionalität der Software, während der Rest als Referenz aller vorhandenen Konfigurationseinstellungen dient.

Da das Buch versucht, sich an den Bedürfnissen von OTRS Administratoren und OTRS Neulingen zu orientieren und da die Qualität der folgenden Kapitel so hoch wie möglich sein soll, sind wir auf Ihr Feedback angewiesen. Bitteteilen Sie uns mit, wenn Sie Abschnitte in diesem Buch vermissen, wenn Dinge für Sie unverständlich erklärt sind oder auch wenn Sie Rechtschreib-, Tipp- oder Grammatikfehler in diesem Buch entdecken. Jede Art von Rückmeldung ist ausdrücklich erwünscht und sollte durch einen Eintrag auf <http://bugs.otrs.org> an uns gerichtet werden, da sie so nicht verloren geht und direkt beim zuständigen Ansprechpartner landet. Wir bedanken uns schon jetzt für jede Art von Mithilfe!

Kapitel 1. Trouble-Ticket-Systeme - die Grundlagen

In diesem Abschnitt soll kurz die grundlegende Idee, die hinter Trouble-Tickets im Allgemeinen und Trouble-Ticket-Systemen im Speziellen steht, erläutert werden. An einem kleinen Beispiel wird gezeigt, wo die Vorteile dieser Systeme liegen.

Was ist ein Trouble-Ticket-System, und warum benötigen Sie eins?

Das folgende Beispiel soll verdeutlichen, was ein Trouble Ticket System ist und wie Sie damit in Ihrem Unternehmen Zeit und Geld einsparen können.

Nehmen wir an, dass Max Mustermann Fabrikant ist und Videorekorder produziert. Da die Programmierung der Videorekorder sehr unübersichtlich und kompliziert ist, wenden sich die Kunden von Herrn Mustermann gerne und häufig mit Supportanfragen per Mail an ihn. An manchen Tagen kann Herr Mustermann der Mailflut kaum Herr werden und so kommt es, dass seine Kunden sich einige Zeit gedulden müssen, bis die Antwort mit der rettenden Lösung bei ihnen eintrifft. Manchen Kunden dauert dies jedoch zu lange, eine weitere E-Mail mit dem gleichen Inhalt wird an Herrn Mustermann geschickt. Die E-Mails mit den Supportanfragen werden alle in eine INBOX weitergeleitet, wie sie von fast allen E-Mailprogrammen verwendet wird.

An manchen Tagen ist die Anfragewelle besonders groß und Herr Mustermann sieht sich außerstande, alle Mails noch in einem vertretbaren Zeitrahmen zu beantworten. Aus diesem Grund kommandiert er seine Entwickler Meier und Schulze zur Bearbeitung der Supportanfragen ab. Da von allen das gleiche System benutzt wird, greifen alle auf die gleiche INBOX und daher auch auf die gleichen Mails zu. Meier und Schulze haben jedoch keine Ahnung, dass manch ein Kunde in seiner Not gleich zwei E-Mails verfasst und an Herrn Mustermann geschickt hat. So kommt es vor, dass Meier die erste Mail mit einem anderen Ratschlag beantwortet als Schulze der sich im selben Moment der zweiten Nachricht des gleichen Kunden annimmt. Das Ergebnis ist, dass der Kunde unterschiedliche Antworten bekommt. Darüber hinaus hat Herr Mustermann keinen Einblick darüber, welcher Mitarbeiter wann was welchem Kunden gesagt hat, welche Probleme besonders häufig auftreten und wie groß sein gesamter Aufwand für den Kundensupport ist.

Von einem Kollegen erfährt Herr Mustermann schließlich, dass es Trouble-Ticket-Systeme gibt, die genau die Probleme lösen, die Herr Mustermann mit dem Support für seine Kunden hat. Herr Mustermann entscheidet sich für das offene Trouble-Ticket-Request System OTRS und installiert dieses System auf einem Rechner, der über einen Webserver sowohl für seine Mitarbeiter als auch über das Internet erreichbar ist. Von nun an werden die Hilferufe der Kunden nicht mehr länger an seine private INBOX, sondern direkt an den Mail-Account für OTRS weitergeleitet. OTRS hat eine Schnittstelle zur INBOX für die Supportanfragen, so dass alle ankommenden E-Mails automatisch ins Trouble Ticket System eingespeist werden. Unabhängig ob Herr Mustermann nun gerade anwesend ist oder nicht, generiert OTRS eine automatische Antwort und teilt dem Kunden mit, dass seine E-Mail angekommen ist und so schnell wie möglich bearbeitet wird. Dabei wird eine eindeutige Trouble Ticket Nummer vergeben. Der Kunde ist glücklich, dass sein Flehen schnell erhört wurde und wartet gespannt auf eine Antwort. Sowohl Herr Mustermann als auch die Entwickler Meier und Schulze können nun über einen beliebigen Internetbrowser und die Weboberfläche von OTRS auf die Supportanfragen zugreifen und diese einzeln beantworten. Da die Tickets bei Beantwortung gesperrt werden, wird keine Nachricht versehentlich zweimal erstellt.

Stellen wir uns vor, dass Herr Schmidt eine Anfrage ans System gestellt hat und Herr Meier diese kurz und knapp beantwortet. Herrn Schmidt reicht diese Antwort jedoch nicht aus und so antwortet er auf

die Lösungsmail am folgenden Tag. Herr Meier ist jedoch gerade mit anderen Dingen beschäftigt, so dass sich Herr Mustermann der Sache annimmt. Über die History-Funktion von OTRS kann er jetzt auf alle vergangenen E-Mails von Herrn Schmidt und Herrn Meier zugreifen, deren Inhalt abfragen und eine ausführlichere Antwort versenden. Herr Schmidt erhält nun die Lösung für sein Problem, weiß aber nicht, dass diese von unterschiedlichen Personen stammt.

Natürlich ist dies nur ein sehr kleiner Einblick in die Funktionalitäten von Trouble Ticket Systemen. Da Herr Mustermann eine kleine Firma führt, erhält er vielleicht nur wenige E-Mails mit Supportanfragen pro Tag, die er vielleicht noch ganz überschaulich mit seiner normalen Mailsoftware handhaben kann und somit kein Trouble Ticket System braucht. Wenn aber der neue DVD-Rekorder in die Regale kommt, werden es vielleicht schon 500 oder in ein paar Jahren schon 10.000 Nachrichten pro Tag sein. Spätestens dann rechnet sich der Einsatz von Trouble-Ticket-Systemen wie OTRS.

Was ist ein Trouble-Ticket?

Ein Trouble-Ticket lässt sich im Wesentlichen mit einem Krankenblatt eines Krankenhauspatienten vergleichen. Bei der erstmaligen Einlieferung in das Krankenhaus wird das Krankenblatt im Zuge der Anamnese neu angelegt. Jeder Arzt trägt nun seine Diagnose sowie die verordnete Therapie und Medikation ein und dokumentiert deren Erfolg. Das Krankenblatt gibt nun einen schnellen Überblick, gewährleistet eine schnelle Einarbeitung und verhindert eine Mehrfachdosierung von Medikamenten. Ist die Krankheit besiegt und der Patient entlassen, wird das Krankenblatt archiviert.

Im OTRS werden Trouble Tickets, also die Krankenblätter aus dem obigen Beispiel, als normale E-Mails behandelt und gespeichert. Schickt z. B. ein Kunde eine Anfrage an das Trouble Ticket System, wird das Krankenblatt eingerichtet, ein neues Ticket wird geöffnet. Die Antwort eines Mitarbeiters auf die Anfrage kann als Eintrag eines Arztes gesehen werden, eine erneute Antwort bzw. Anfrage des Kunden auf das selbe Ticket als Veränderung oder Erweiterung des Krankheitsbildes. Ein Ticket gilt als erledigt bzw. geschlossen, wenn eine Antwort auf die Anfrage an den Kunden zurückgesendet wurde oder das Ticket über das System als geschlossen markiert wird. Antwortet ein Kunde auf ein bereits geschlossenes Ticket, so wird es erneut geöffnet und die neuen Informationen ergänzt. Um die Konsistenz der Daten sicherzustellen, werden alle Tickets mit all ihren spezifischen Informationen archiviert und verbleiben im System. Durch die Speicherung der Tickets als ganznormale E-Mails ist es möglich, dass diese auch E-Mail-Anhänge enthalten können. Zusätzlich zu den normalen Informationen einer E-Mail lassen sich beliebige Notizen zu jedem Ticket hinzufügen. Die Tickets selbst werden auf der Festplatte bzw. in einer Datenbank archiviert, ebenso zusätzliche Meta-Informationen des Tickets wie Notizen, an der Beantwortung des Tickets beteiligte Mitarbeiter, Zeit und Datum der Bearbeitung, Bearbeitungsdauer usw. Eine Sortierung oder eine Suche über den Datenbestand wird mit Hilfe aller vorhandenen Informationen zu den Tickets realisiert.

Kapitel 2. OTRS - Open Ticket Request System

In diesem Abschnitt werden die Features des Open Ticket Request Systems (OTRS) vorgestellt. Des Weiteren wird näher auf die Systemanforderungen von OTRS eingegangen und erläutert, wie Kontakt zur OTRS-Community aufgenommen werden kann bzw. wie kommerzieller Support erhältlich ist.

Grundlagen

Das Open Ticket Request System (OTRS) ist eine Webapplikation, die mit jedem HTML kompatiblen Browser benutzt werden kann. OTRS verzichtet bewusst auf aktive Webkomponenten wie z. B. Javaskript, JavaApplets und Flash. Dadurch wird es möglich, das System nicht nur über einen PC, sondern auch von Mobiltelefonen und anderen Handcomputern, welche HTML unterstützen, zu benutzen. Als Client kann jedes Betriebssystem eingesetzt werden.

OTRS ist in mehrere Komponenten aufgeteilt. Als Basis wird ein Framework benötigt, der alle grundlegenden Funktionen des Systems und das Trouble Ticket System enthält. An zusätzlichen Komponenten können ein Webmailer, ein Dateimanager, ein Contentmanager, ein Kalender und ein Modul zur Ausgabe verschiedener Systemstatus bequem über die Weboberfläche nachinstalliert werden.

Features

OTRS bietet viele verschiedene Features. Die folgende Aufzählung gibt einen Überblick über die wichtigsten Eigenschaften und Fähigkeiten des zentralen OTRS-Frameworks.

Die Features von OTRS

- Web-Interface:
 - Einfache und intuitive Bedienung mit einem Web-Browser.
 - Auf aktive Inhalte wie Flash oder Java-Applets wird in der Weboberfläche bewusst verzichtet. Das System kann somit mit nahezu allen Browsern benutzt werden, auch von Mobiltelefonen oder Handcomputern aus.
 - Eine Web-Oberfläche zur Administration des Systems ist vorhanden.
 - Ein Webinterface für die Mitarbeiter (Agenten) zur Bearbeitung von Kundenanfragen ist verfügbar.
 - Eine Weboberfläche für Kunden, über die Nachrichten an zuständige Agenten geschickt werden können und der Status eigener Tickets abgerufen werden kann, ist vorhanden.
 - Unterstützung für verschiedene Oberflächen-Layouts (Themes).
 - Unterstützung vieler Sprachen.
 - Eigene Anpassungen der Ausgabe-Vorlagen sind möglich (DTL).
 - Mehrfach-Anhänge von Emails sind über die Weboberfläche möglich.
- E-Mail-Schnittstelle:

- Unterstützung für E-Mail-Anhänge (MIME)
- Automatische Umwandlung von HTML- in reine Text-Nachrichten (höhere Sicherheit vor schädlichen Inhalten und schneller durchsuchbar).
- Filterung von E-Mails über eigene X-Header-Einträge oder Mailadressen, z. B. für die Aussortierung von Spam.
- PGP-Support, Erstellung und Import eigener Zertifikate, verschlüsselter und signierter Mails, Anzeige von verschlüsselten und signierten Nachrichten.
- Unterstützung für die Verschlüsselung und Anzeige von SMIME-Nachrichten.
- Automatisierte Antworten (auto responder) für die Benachrichtigung von Kunden, abhängig von der Queue konfigurierbar.
- E-Mail-Benachrichtigungen für Agenten über neue Tickets, Follow-ups oder freigegebene Tickets.
- Follow-Ups anhand von Reference- oder In-Reply-To-Headern, automatisierte Zuordnung von Follow-Ups über den Ticket-Bezeichner im Betreff, Mailbody oder in Text-Anhängen.
- Tickets:
 - Erweiterte Queue-Ansicht, Übersicht über alle Anfragen innerhalb einer Queue.
 - Sperren von Tickets.
 - Erstellung eigener Antwortvorlagen.
 - Erstellung eigener auto responder, abhängig von der Queue.
 - Ticket-History, Übersicht über die komplette Entwicklung eines Tickets, Änderungen der Ticketstatus, Übersicht über die verschiedenen Aktionen für ein Ticket usw.
 - Druckansicht für Tickets, Export als PDF möglich.
 - Hinzufügen eigener (interner oder externer) Notizen zu einem Ticket (eigener Text und Dateianhänge).
 - Möglichkeit zum Zoomen von Tickets.
 - Definition von ACLs (access control lists) für Tickets.
 - Tickets können an andere E-Mail-Adressen weiter- oder umgeleitet werden (forwarding, bouncing).
 - Verschieben von Tickets zwischen verschiedenen Queues.
 - Festlegen der Priorität für Tickets.
 - Erfassung der Bearbeitungsdauer für Tickets.
 - Anstehende Aufgaben für ein Ticket festlegen (pending features).
 - Massenoperationen auf Tickets sind möglich (bulk features).
 - Automatische und zeitgesteuerte Aktionen können mit Hilfe eines sog. "GenericAgent" auf Tickets ausgeführt werden.

- Volltextsuche über den gesamten Ticketbestand und Export der Suchergebnisse als PDF.
- System:
 - OTRS läuft unter vielen Betriebssystemen (Linux, Solaris, AIX, FreeBSD, OpenBSD, Mac OS 10.x, Microsoft Windows).
 - Unterstützung von ASP (active service providing).
 - Verknüpfung von Objekten wie z. B. Tickets, FAQ-Einträgen o.ä. innerhalb des Systems.
 - Einbindung externer Datenquellen für die Kundendaten, z. B. über AD, eDirectory oder OpenLDAP).
 - Festlegen einer eigenen Kennzeichnung für Tickets, z. B. Call#, Ticket#, Request# o.ä.
 - Festlegen einer eigenen Nummerierung für Tickets.
 - Unterstützung verschiedener Datenbanktypen für die zentrale Datenbank von OTRS (MySQL, PostgreSQL, Oracle MSSQL, DB2, usw.).
 - Framework für die Erstellung von Statistiken über die Web-Oberfläche, Export und Import von Reports und Statistikmodulen, Export von Statistiken als PDF ist möglich.
 - Unicode-Unterstützung für Front- und Back-End.
 - Die Authentifikation für Agenten oder Kunden kann unabhängig voneinander über eine Datenbank, LDAP, HTTPAuth oder Radius realisiert werden.
 - Unterstützung von Benutzer-Accounts, Benutzergruppen und Rollen.
 - Unterstützung verschiedener Zugriffsrechte, z. B. auf Queues oder Systembereiche.
 - Die Erstellung von Standardantworten ist möglich.
 - Unter-Queues werden unterstützt.
 - Anreden und Signaturen können abhängig von der Queue definiert werden.
 - E-Mail-Benachrichtigungen für Administratoren.
 - Bekanntgabe von Informationen zu Updates über die Weboberfläche oder via E-Mail.
 - Festlegen von Ablaufristen für problematische Tickets.
 - Benutzerabhängige Unterstützung für verschiedene Zeitzonen.
 - Einfache Einbindung eigener Addons und Module mit Hilfe der OTRS API.
 - Einfache Erstellung eigener Front-Ends, z. B. X11, Console usw.

Die wichtigsten neuen Features von OTRS 3.0

Kontext

- Benutzerzentriertes Redesign der graphischen Benutzerschnittstelle, welches einen Übergang zu dynamischen Technologien wie AJAX, XHTML und CSS beinhaltet.

Hervorhebung für "neue Tickets" und "neue Artikel"

- Dieses neue Feature wurde für Tickets und Artikel implementiert. Es erlaubt einem Agenten, auf einen Blick zu erkennen, ob zu bestimmten Tickets oder auch innerhalb eines Artikels neue, ungelesene Artikel gibt. Sie profitieren durch erhöhte Transparenz und kürzere Reaktionszeiten.

Optimierte Volltextsuche

- Das neue Such-Feature erlaubt es, die vorhandenen Informationen flexibel zu durchforsten. Die Optionen reichen von einfachem Suchwort bis hin zu komplexen, multi-Wort-fähigen booleschen Suchoperationen, in denen man verschiedene Operatoren verwenden kann. Somit stehen Ihnen gemäß Ihrer Anforderungen anpassbare Suchmöglichkeiten zur Verfügung.

Neue Ticket-Zoom-Ansicht

- Die neue, AJAX-basierte Ticket-Zoom-Ansicht erlaubt die Darstellung komplexer und verknüpfter Informationen in Echtzeit unter Beibehaltung der aktuellen Arbeitsumgebung des Agenten. Letzterer profitiert von verbesserter Orientierung und daraus folgend höherer Arbeitseffizienz.

Globale Ticketübersichten

- Die seit OTRS 2.4 wohlbekannten Ticketübersichten wurden für verbesserte Interaktivität optimiert. Je nach Anwendungsszenario können Ihre Agenten die Ticketübersicht leicht wechseln. Klein, Mittel und Groß stehen zur Verfügung und stellen jeweils eine unterschiedliche Detailtiefe bereit.

Zugänglichkeit

- Das Redesign berücksichtigt die anerkannten Zugänglichkeits-Standards WCAG und WAI-ARIA. Dadurch können behinderte Nutzer besser mit OTRS arbeiten. Die Bestimmungen des Abschnitt 508 des US Rehabilitation Act werden erfüllt.

Neues Kunden-Interface

- Das Kunden-Web-Interface kann in das Intranet Ihrer Organisation integriert werden und ist eng mit dem neuen OTRS-System verzahnt.

Archivierungs-Feature

- Mit OTRS 3.0 können Sie Tickets archivieren. Dadurch verkürzen sich Zeiten für Suchabfragen und deren Ergebnisauswertung.

Neue Features von OTRS 2.4

Lizenzwechsel auf die AGPL Version 3

- Warum AGPL statt GPL? AGPL und GPL sind bis auf einen einzigen Punkt identisch. Wird Software im Bereich SaaS eingesetzt, greift bei der AGPL das Copyleft, bei der GPL nicht. Wir möchten, dass auch im wachsenden SaaS-Umfeld Weiterentwicklungen an OTRS in die Community zurückfließen.

Warum v3 statt v2? Die GPL v2 ist in die Jahre gekommen und hat insbesondere außerhalb der USA diverse rechtliche Unsicherheiten. Die GPLv3 bzw. AGPLv3 ist unter den Open Source Lizenzen diejenige Copyleft-Lizenz, die sowohl den Copyright-Inhabern als auch dem Anwender den besten Schutz und die größte Rechtssicherheit bietet.

Neues Management-Dashboard

- Der Bedarf an einer systemweiten, personalisierten und aktuellen Präsentation wichtiger Informationen hat zur Entwicklung des Management-Dashboards geführt. Dafür können Plugins erstellt werden, die Inhalte von Erweiterungsmodulen zusätzlich zu den standardmäßig verfügbaren Informationen darstellen. Standard-Plugins sind:
- Ticket-Volumen (neu & offen) der letzten 24h, 48h und 72h
- Kalender mit anstehenden Ereignissen (Eskalationen, automatische Ticketentsperrungen usw.)
- Systemweite Übersicht der Ticketverteilung in den Queues
- Zeit für die erste Reaktion / Lösungszeit von Queues
- RSS-Integration

Weitere Standard-Reports

- Die zusätzlichen Standardreports von OTRS 2.4 sind:
- Erstellte Tickets
- Geschlossene Tickets
- SLA-Analyse
- Benötigte Bearbeitungszeit pro Kunde / pro Queue
- Analyse der Lösungszeit pro Kunde / pro Queue
- Analyse der Antwortzeit pro Kunde / pro Queue

Neues Master/Slave Ticket-Feature

- Die Master Slave Funktion erlaubt Ihnen, Tickets zum gleichen Thema zu markieren, um sie über das Master-Ticket gebündelt abzarbeiten. Die Slave-Tickets werden automatisch gelöst und Ihre Kunden mit der im Master-Ticket hinterlegten Antwort benachrichtigt, wenn das Master-Ticket auf gelöst gesetzt wird.

Alle Slave-Tickets, 'erben' die folgenden Aktionen ihres Masters:

- Status ändern
- Email-Antworten
- Freitextfelder verändern
- Anmerkungen
- Wartezeit ändern
- Priorität ändern
- Besitzer ändern
- Verantwortlichen ändern

HTML E-Mail Unterstützung

- Erstellen und formatieren Sie E-Mails, Notizen & Benachrichtigungen nun auch im Rich-Text-Format (HTML-Format) oder binden Sie Bilder über der WYSIWYG-Editor ein.

Neue Out-Of-Office-Funktion

- Definieren Sie über Ihre persönlichen Einstellungen den Zeitraum Ihrer Abwesenheit. Aktivierung hat folgende Auswirkungen:

In Auswahllisten, die der Zuweisung eines Besitzers oder Verantwortlichen dienen, wird die Abwesenheitszeit und -dauer hinter Ihrem Namen dargestellt.

Bei FollowUps, wird das Ticket vollautomatisch "entsperrt" und das Bearbeitungs-Team der Queue erhält eine Benachrichtigung.

Neue Ticket-Übersichten und globale Sammelaktion

- Flexibel zur Laufzeit gestaltbare Ticket-Übersichten erleichtern Ihnen die tägliche die Arbeit. Ob tabellarische Übersicht oder Ticket-Vorschau, ein Small / Medium / Large View Icon erlaubt es Ihnen, sich je nach Einsatzszenario Ihre Übersichten, z. B. den Queue View, Status View, Escalation View oder Suchergebnisse individuell einzurichten.

Die systemweit in allen Ticket-Übersichten integrierte Sammelaktion erlaubt Ihnen die gleichzeitige Bearbeitung mehrerer Tickets.

Postmaster-Filter erkennen Follow-Ups zu intern weitergeleiteten Nachrichten

- Derzeit werden E-Mail-Antworten auf weitergeleitete Artikel in OTRS als "email-extern" aufgenommen. Das Problem dabei ist, dass diese Antworten vom Kunden im Web-Interface eingesehen werden können. Zwar wäre es möglich, E-Mails generell als "email-intern" zu klassifizieren, doch dann können Kunden nicht mehr angemessen bedient werden, weil sie ihre Tickets im Web-Interface nicht mehr korrekt nachverfolgen könnten. Mit diesem neuen Feature können E-Mail-Antworten zurückverfolgt werden, und der Artikeltyp "email-extern" oder "email-intern" wird je nach dem Typ des ursprünglichen Artikels automatisch gesetzt.

Konfigurierbare Event-basierte Benachrichtigungen

- Um eine höhere Flexibilität für E-Mail-Benachrichtigungen zu erreichen, wurde ein komplett neuer Benachrichtigungs-Mechanismus implementiert. Dieser ermöglicht es, event-basierte Benachrichtigungen an Agenten, Kunden oder dedizierte Email-Adressen zu versenden.

Somit ist es einfach möglich, nur beim Schließen eines Tickets eine E-Mail-Benachrichtigung an den Kunden eines Tickets zu senden. Oder z. B. beim Erstellen von VIP Tickets für einen bestimmten Kundenkreis mit sehr hoher Priorität, eine bestimmte E-Mail-Adresse (z. B. ein Bereitschaftshandy) zu benachrichtigen. Die dabei jeweils zu berücksichtigenden Events (z. B. Erstellung neuer Tickets, Ticketstatus-Update, Queue-Update, Erstellung neuer Artikel, usw.) und Inhalt der Benachrichtigungen (inkl. OTRS-Platzhalter) sind frei über ein Web-Interface konfigurierbar.

Leserechte und Benachrichtigungen auf Tickets der Watchliste

- Mit der "Read-Only"-Feature haben Sie Leserechte auf ein Ticket in Ihrer Watchliste, auch wenn das Ticket in eine Queue verschoben wurde, in der Sie keine Leserechte haben. Über Ihre persönlichen Einstellungen haben Sie die Möglichkeit zu definieren, ob Sie bei beobachteten Tickets, wie ein Besitzer oder Verantwortlicher über Änderungen per E-Mail benachrichtigt werden soll.

Secure SMTP-Unterstützung

- Beim Empfang von E-Mails unterstützt OTRS bislang POP3, POP3s, IMAP und IMAPs. Beim Versand neben lokalen Mail-Transfer-Agents (z. B. Sendmail, Postfix, Exim) und SMTP nun auch SMTPS (Secure-SMTP).

Neue Features von OTRS 2.3

Performance

- Datenbank- und Codeverbesserungen führen zu einem allgemeinen Performance-Gewinn von bis zu 20%.
- Es wurde eine indizierte Volltextsuche hinzugefügt. Standardmässig ist dieses Feature deaktiviert, da zusätzlicher Festplattenplatz erforderlich ist. Der erwartete Performance-Gewinn liegt bei 50%.
- Reduziertes Neu-Laden von Seiten durch AJAX-Technologie.
- Anstatt die Eskalationszeiten während der Laufzeit ständig neu zu berechnen, werden sie nur noch dann neu berechnet, wenn sie sich durch ein Event in OTRS verändern. Die Eskalationszeiten werden im Ticket-Objekt gespeichert, dadurch können externe Reporting-Tools direkt darauf zugreifen. Ausserdem wird die Performance signifikant erhöht.

Such-Funktionalität

- Unterstützung von logischen Ausdrücken: Die Ticket-, Kunden-, und die FAQ-Suche unterstützen logische Ausdrücke, wie die Operatoren UND, ODER, !, als auch Klammern, um Ausdrücke zu strukturieren.
- Die Suche nach Ticket-Nummern direkt aus dem Browser heraus, mithilfe des OpenSearch features (OpenSearch Format).
- Die Suche nach dem Ticket-Titel im Agenten-Interface und im GenericAgent.
- Die Suche nach Schließ-Zeiten von Tickets im Agenten-Interface und im GenericAgent.

Ticket-Inhalt und Verschieben von Tickets

- Aus-/zuklappen von Artikeln: Die Artikel-Ansicht kann ausgeklappt werden, um alle Artikel auf einmal zu sehen. Der aktuelle Artikel bleibt im Fokus, und der vorherige und nachfolgende Artikel werden zusätzlich angezeigt.
- Strukturierter Artikel-Baum: Die Artikel-Baum-Ansicht wird jetzt als Tabelle dargestellt.
- Das Ausdrucken von Artikeln ist jetzt möglich.
- Der Ticket-Titel von verknüpften Tickets wird als Mouse-Over Effekt angezeigt.
- Verknüpfte Tickets werden nun durchgestrichen dargestellt.
- Beim Verschieben von Tickets können mehrere Dateien als Anhänge hinzugefügt werden.

Ticket FreeText und FreeTime Features

- Beim Teilen eines Tickets werden nun alle FreeText und FreeTime Felder in das neue Ticket kopiert.
- Ticket-FreeTime Felder können als Pflichtfeld definiert werden.

- Eine URL kann als Wert für FreeText-Felder konfiguriert werden, die dann als Link in einem Ticket dargestellt wird.
- Die Email-Header X-OTRS-TicketTime und X-OTRS-FollowUp-TicketTime wurden hinzugefügt.

Unterstützung von IMAP, IMAPS und POP3S

- Mit OTRS 2.3 werden zusätzlich zu POP3 die Email-Protokolle POP3S, IMAP und IMAPS unterstützt.

Sicherheit

- Im Falle eines vergessenen Passworts sendet OTRS mit Hilfe des "Password Reset Links" eine E-mail an den Benutzer. Nach dem der Benutzer auf den Bestätigungs-Link in der Mail klickt, wird das Passwort in einer zweiten E-mail verschickt.

Benachrichtigungen und Eskalationen

- Alle Agenten mit Leseberechtigung für eine bestimmte Queue können für Benachrichtigungen ausgewählt werden.
- Eine neue Eskalationsübersicht wurde hinzugefügt. Sie zeigt alle Tickets, sortiert nach der verbleibenden Zeit bis zur Eskalation.

Neue Features von OTRS 2.2

Neue Features von OTRS 2.2

- Services und SLAs: Auf dem Weg hin zu einem IT-Servicemanagement-Werkzeug, wurden in OTRS 2.2 die neuen Attribute 'Service' und 'Service Level Agreements (SLA)' integriert. Bei der Erstellung eines neuen Tickets kann der Antragsteller einen Service (z. B. Email-Service) und ein zugehöriges SLA auswählen. SLA Attribute sind "response time" (Antwortzeit), "update time" (Updatezeit) und "solution time" (Lösungszeit). Diese Attribute können vom IT-Service für Benachrichtigungen oder für die Eskalation von Tickets herangezogen werden, um bestehende SLAs einzuhalten. Service- und SLA-spezifische Informationen innerhalb der Header neuer Emails können weiterhin mit Hilfe des PostMasterFilter-Moduls ausgewertet werden.
- Unterstützung nativer Tickettypen: Verschiedene Tickettypen können nun über das Admin-Interface verwaltet werden. Die Benutzung von Freitext-Feldern für die Spezifikation von Tickettypen ist somit nicht mehr notwendig. Installationen, die Freitextfelder für die Klassifikation von Tickettypen verwenden, müssen nicht migriert werden. Dieses neue Feature wird ebenfalls im ticketinhalt und auch in der Druckansicht für Agenten und Kunden angezeigt und kann über das Agenten-Interface angepasst werden.
- Unterstützung multipler Authentifikation-Quellen: Durch die Unterstützung multipler vertrauenswürdiger Authentifikations-Quellen wird es möglich, mehrere Backends für die Anmeldung von Kunden und Agenten einzubinden. Schlägt die Authentifikation für die erste Quelle fehl (z. B. für ein LDAP-Backend), kann als 2. Quelle ein weiteres Backend (z. B. eine SQL-Datenbank) eingebunden werden.
- Support von Kennworten mit verschiedener Verschlüsselung: Die verschiedenen unterstützten und erlaubten Verschlüsselungen für Agenten- und Kundenkennworte sind:
 - unix_crypt()
 - md5()

- plain()

Die Standardeinstellung ist 'unix_crypt()'. Für ein laufendes System ist ein Wechsel ohne großen Aufwand von unix_crypt() zu md5() möglich, es muss lediglich der entsprechende Konfigurationsparameter angepasst werden.

- Änderung der OTRS-internen Struktur zur Speicherung der Kundendaten: Die Speicherung der Kundendaten wurde restrukturiert und in die Objekte "CustomerCompany" und "CustomerUser" aufgeteilt. Firmenspezifische Daten (Company) wie z. B. der Firmenname, die Firmenadresse, usw. werden nun gegenüber den personenspezifischen Daten (Vor- und Nachname, Anrede, etc.) gesondert gepflegt.

Diese Informationen werden wie gewohnt innerhalb des Agent-Interfaces angezeigt.

- Erweitertes OPM-Format: OPM Pakete werden nun mit einer zusätzlichen Beschreibung ausgeliefert. Die einleitende Seite für ein OPM Paket informiert den Benutzer über neue Features und über die Schritte, die nach der Installation des Paketes nötig sind.
- PostMaster-Admin-Interface: Durch diese Erweiterung kann genauer definiert werden, welche Email-Adressen vom PostMasterFilter-Modul behandelt und in Betracht gezogen werden sollen.
- Verbesserung des SysConfig-Interfaces: Durch interne Umstrukturierungen des SysConfig-Interfaces wurde die Performance stark erhöht. Die Seiten werden nun viel flüssiger aufgebaut wodurch ein viel schnelleres Arbeiten mit dem Tool möglich wird.

Neue Features von OTRS 2.1

Neue Features von OTRS 2.1

- Stabile Unterstützung von Microsoft SQL Server: von dieser Version an bietet OTRS eine Integration zu MS SQL Server Datenbanken. OTRS unterstützt alle populären Betriebssysteme wie Linux, Windows, Mac OS X wie auch alle wichtigen Datenbanksysteme wie MS SQL, MySQL, PostgreSQL, Oracle und DB2.
- Multi-Kalender-Funktionalität: Dieses Feature vereinfacht das Arbeiten in verschiedenen Umgebungen mit verschiedenen Zeitzonen. Die relevanten Arbeitsstunden des Systems können abhängig von der Zeitzone festgelegt werden, die Spezifikation von Urlaubs- und Feiertagen ist abhängig von der Region möglich.
- Statistik-Framework: Integration eines komplett überarbeiteten Statistik-Frameworks der es erlaubt, nahezu jede Art von Report über die Weboberfläche zu erstellen. Die Erstellung und Ansicht von Statistiken kann pro Benutzer, Gruppe und / oder Rolle freigeschaltet werden. Der Ex- und Import von bereits vorhandenen oder neuen Statistiken ist möglich, Statistikmodule aus früheren OTRS-Versionen können weiter verwendet werden.
- Optimierte Synchronisation von Autorisierungen von LDAP-Diensten. Damit wird die Anbindung aller bekannten LDAP-Verzeichnisdienste wie MS Active Directory, OpenLDAP, Novell eDirectory, Oracle Internet Directory usw. erleichtert. Ein nützliches Feature, welches beim Verwalten der Autorisierungen in der lokalen Datenbank von OTRS Zeit und Mühe spart.
- PDF-Generator: Die Druck-Ansicht von Tickets, Statistiken und Suchergebnisse können nun als PDF-Datei exportiert werden.
- FAQ-Modul: Das FAQ-Modul von OTRS wurde komplett neu implementiert und aus dem zentralen OTRS-Framework herausgenommen, so dass eine Installation als separates Modul möglich ist. Das

neue FAQ-Modul bietet erweiterte Such- und Navigationsfunktionen, eine verbesserte Administration und eine Möglichkeit zur Bewertung von einzelnen FAQ-Einträgen.

- Unterstützung hierarchischer Team-Strukturen: Mit Hilfe der neu eingeführten Überwachungs- und Verantwortlichkeitsfunktionen für Tickets, wird die strukturierte Bearbeitung von Tickets in Teams vereinfacht. Das Verantwortlichkeits-Feature ermöglicht es einem Agenten den Ticket-Besitz abzugeben, jedoch trotzdem noch alle Schreibrechte am Ticket zu behalten. Das Überwachungs-Features ermöglicht die Verfolgung der Aktivitäten für ein Ticket, ohne selbst Eigentümer des Tickets zu sein.
- Optimierte Erkennung von Follow-Ups: Neben der Auswertung der Ticketnummer im Betreff einer Nachricht kann OTRS 2.1 nun auch nach Ticket-Bezeichnern im Body einer Mail und innerhalb von Textanhängen suchen und so ggfs. ein Follow-Up automatisiert zuordnen.

Hardware und Software-Anforderungen

OTRS kann unter vielen Betriebssystemen installiert werden. Neben Linux und den verschiedenen Unix-Derivaten (z. B. OpenBSD oder FreeBSD), läuft OTRS auch unter allen Windows-Versionen. Bezüglich der Hardware empfiehlt es sich, mindestens einen 2 GHz Pentium Xeon oder vergleichbare CPU, 2 GB RAM und eine 160 GB Festplatte zu verwenden.

Für den Betrieb von OTRS werden einige externe Software-Komponenten benötigt. Ein Web- sowie ein Datenbankserver und eine funktionierende Perl-Installation mit einigen Zusatzmodulen sind die Grundvoraussetzungen für ein funktionierendes System. Der Webserver und Perl müssen auf der gleichen Maschine installiert sein, auf der später auch OTRS ausgeführt werden soll. Das Datenbank-Back-End kann auf der lokalen oder auf einer entfernten Maschine installiert werden.

Für den Webserver wird empfohlen, auf apache 2 zurück zu greifen. Vor allem durch die Erweiterung der apache-Konfiguration um das mod_perl-Modul, kann die Geschwindigkeit von OTRS enorm gesteigert werden. Prinzipiell sollte aber jeder Webserver, der die Ausführung von Perl-Skripten unterstützt, für den Betrieb von OTRS geeignet sein.

Als Datenbank-Back-End eignen sich MySQL, PostgreSQL, Oracle, MSSQL oder DB2. Ein Vorteil der Benutzung von MySQL ist, dass OTRS über einen Webinstaller die Datenbank und Tabellen automatisch anlegen kann.

Für Perl gilt mindestens die Version 5.8.8 zu verwenden. Es werden einige Zusatzmodule benötigt, die Sie entweder direkt über die Shell von Perl und CPAN oder mit Hilfe des Paketmanagers (yast, apt-get) Ihres Betriebssystems einspielen müssen.

Software-Anforderungen

Perl

- Perl 5.8.8 oder höher

Webserver

- Apache2 und mod_perl2 (mod_perl2 wird wegen der hohen Geschwindigkeit empfohlen)
- Webserver mit CGI support (CGI nicht empfohlen)
- IIS6 oder höher

Datenbanken

- MySQL 4.1 oder höher

- PostgreSQL 8.0 oder höher
- Oracle 10g oder höher
- DB2 8 oder höher
- MSSQL 2000 oder höher

Im Abschnitt für die Installation der für OTRS benötigten Perl-Module wird beschrieben, wie Sie Perl-Module manuell einspielen können.

Wenn Sie ein bereits vorgefertigtes OTRS-Paket für Ihr Betriebssystem zur Installation verwenden (rpm, Windows-Installer), sollten die benötigten Perl-Module automatisch installiert werden.

Community

Um OTRS hat sich in den letzten Jahren eine große Community gebildet. Über Mailinglisten tauschen sich Anwender und Entwickler zu den verschiedensten Themen rund um das Trouble Ticket System aus. Behandelt werden Fragen rund um die Installation, Konfiguration, Benutzung, Lokalisation und Entwicklung. Fehler in der Software können über ein Bug-Tracking-System gemeldet werden und erreichen so die zuständigen Entwickler bzw. gehen nicht verloren, so dass schnell Fixes bereit gestellt werden können.

Die Homepage der OTRS-Community ist: <http://www.otrs.org> [<http://www.otrs.org>] (siehe Abb. 2.1 unten).

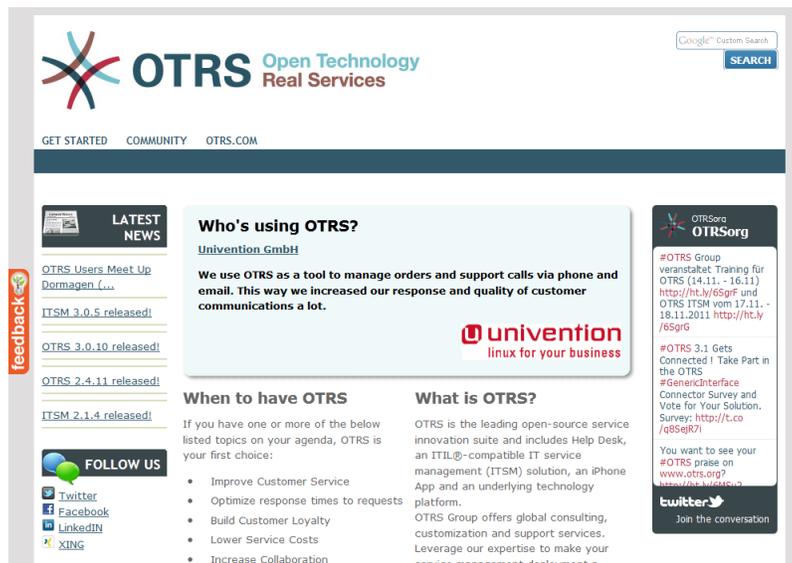


Abb. 2.1. Homepage der OTRS-Community.

Kommerzieller Support und Dienstleistungen für OTRS

Kommerzieller Support für OTRS ist erhältlich. Die können die aktuellen Angebote auf der Website der OTRS Gruppe finden, der Firma hinter OTRS: <http://www.otrs.com/de/solutions> [<http://www.otrs.com/de/solutions>].

Die OTRS Gruppe bietet Support, Anpassungen, Beratung und Schulungen für *den OTRS Help Desk* [<http://www.otrs.com/de/products/help-desk/>] und *OTRS ITSM* [<http://www.otrs.com/de/products/itsm/>] an. Zusätzlich sind *Best Practice Product Editions* [<http://www.otrs.com/solutions/>] erhältlich. Mit letzteren unterstützt die OTRS Gruppe Organisationen dabei, OTRS für ihre jeweilige Umgebung anzupassen, zum Einsatz zu bringen und zu optimieren. Weiterhin bietet die OTRS Gruppe gehostete Versionen an, inklusive *OTRS OnDemand* [<http://www.otrs.com/de/solutions/ondemand/>] und *Managed OTRS* [<http://www.otrs.com/de/solutions/managed-otrs/>].

Detailliertere Informationen über die OTRS Gruppe können Sie unter <http://www.otrs.com> [<http://www.otrs.com>] finden oder über die Kontaktadresse *sales at otrs.com* [[mailto:sales at otrs.com](mailto:sales@otrs.com)] erfragen.

Kapitel 3. Installation

Dieser Abschnitt beschreibt die Installation und die grundlegende Einrichtung des zentralen OTRS Frameworks. Dabei wird die Installation von OTRS aus dem Quellcode erläutert, wie auch über Binärpakete wie RPM oder ausführbare Dateien für Windows.

Dieses Kapitel beschreibt die Einrichtung des Web- und Datenbankservers, die Schnittstelle zwischen OTRS und der Datenbank, das Einspielen einzelner Perl-Module, das Setzen der richtigen Berechtigungen, die Einrichtung der OTRS-eigenen cron-Jobs sowie grundlegende Einstellungen in den OTRS-Konfigurationsdateien.

Folgen Sie den Schritten in diesem Kapitel, um OTRS auf Ihrem Server zu installieren. Dann können Sie sich über die Weboberfläche anmelden und das System administrieren.

Der einfache Weg - Installation fertiger Pakete

Der einfachste und komfortableste Weg ein lauffähiges OTRS zu installieren ist sicherlich, auf bereits vorgefertigte Pakete zurück zu greifen. Viele bereits vorgefertigte Installations-Pakete sind im Download-Bereich unter <http://www.otrs.org> zu finden. Da der Aufwand viel zu groß wäre, die Installation aller dort aufgeführten Pakete in dieser Dokumentation anzuführen, soll im Folgenden nur näher auf die Installation von OTRS unter SUSE Linux, Debian und Microsoft Windows eingegangen werden. Sehen Sie unter der o.g. URL nach, ob auch für Ihr Betriebssystem ein fertiges Installations-Paket vorhanden ist und greifen Sie nur auf die manuelle Installation zurück, wenn Sie keine andere Möglichkeit haben.

Installation des RPM auf einem SuSE-Linux-Server

Dieser Abschnitt beinhaltet die Anleitung für die Installation von OTRS unter SUSE Linux. Getestet wurden die Versionen auf allen aktuellen Versionen von SLES und OpenSUSE. Bevor Sie mit der Installation beginnen, sehen Sie bitte unter <http://www.otrs.org/downloads> nach, ob eine aktuellere Version von OTRS als .rpm-Datei vorliegt. Sollte dies der Fall sein, verwenden Sie bitte diese neuere Version.

Installieren Sie OTRS mittels `yast (yast2)` oder der Kommandozeile und `rpm`, je nach Vorliebe. Beachten Sie jedoch, dass OTRS einige Perl-Module benötigt, die nicht standardmäßig in einer typischen SUSE-Installation enthalten sind. `yast` sollte die bessere Wahl sein, da es alle Abhängigkeiten automatisch beachtet und auflösen kann.

Sollten Sie den Weg über die Kommandozeile mit `rpm` bevorzugen, so müssen Sie die Perl-Module manuell vor Beginn der Installation von OTRS installieren. Angenommen Sie haben die Datei `otrs.rpm` im Verzeichnis `/tmp` gespeichert, dann geben Sie zur Installation von OTRS folgenden Befehl ein.

Hinweis: Wenn Sie es auf SLES9 oder OES installieren, dann folgen Sie bitte dieser Anleitung, um die fehlenden perl-GD Pakete zu installieren: <http://faq.otrs.org/otrs/public.pl?FAQID=49>.

```
linux:~ # rpm -i /tmp/otrs-xxx.rpm
otrs
#####
Check OTRS user (/etc/passwd)... otrs exists.
```

Next steps:

```
[SuSEconfig]
Execute 'SuSEconfig' to configure the web server.
```

```
[start Apache and MySQL]
Execute 'rcapache restart' and 'rcmysql start' in case they don't
run.

[install the OTRS database]
Use a web browser and open this link:
http://localhost/otrs/installer.pl

[OTRS services]
Start OTRS 'rcotrs start-force' (rcotrs {start|stop|status|restart|
start-force|stop-force}).

Have fun!

Your OTRS Team
http://otrs.org/
```

```
linux:~ #
```

Skript 3.1. Kommando für die Installation von OTRS.

Nach der Installation des rpm's ist es notwendig, SuSEconfig zu starten. Geben Sie hierzu Folgendes ein.

```
linux:~ # SuSEconfig
Starting SuSEconfig, the SuSE Configuration Tool...
Running in full featured mode.
Reading /etc/sysconfig and updating the system...
Executing /sbin/conf.d/SuSEconfig.aaa_at_first...
Executing /sbin/conf.d/SuSEconfig.apache...
Including /opt/otrs/scripts/apache-httpd.include.conf
Executing /sbin/conf.d/SuSEconfig.bootsplash...
Executing /sbin/conf.d/SuSEconfig.doublecheck...
Executing /sbin/conf.d/SuSEconfig.guile...
Executing /sbin/conf.d/SuSEconfig.hostname...
Executing /sbin/conf.d/SuSEconfig.ispell...
Executing /sbin/conf.d/SuSEconfig.perl...
Executing /sbin/conf.d/SuSEconfig.permissions...
Executing /sbin/conf.d/SuSEconfig.postfix...
Setting up postfix local as MDA...
Setting SPAM protection to "off"...
Executing /sbin/conf.d/SuSEconfig.profiles...
Finished.
linux:~ #
```

Skript 3.2. Ausführung des Kommandos SuSEconfig.

Die Installation des OTRS-rpm ist abgeschlossen. Starten Sie nun Ihren Webserver neu, um die Änderungen in der Konfiguration zu übernehmen.

```
linux:~ # rcapache restart
Shutting down httpd                               done
Starting httpd [ PERL ]                           done
linux:~ #
```

Skript 3.3. Neustart des Webservers.

Der nächste Schritt ist das Aufsetzen der Datenbank wie in dem Abschnitt über die Datenbankkonfiguration beschrieben.

Installation von OTRS auf einem CentOS-System

Im OTRS-Wiki finden Sie detaillierte Anweisungen für das Aufsetzen unter CentOS, die auch für die Installation unter RedHat gelten, weil beide Systeme auf derselben Basis aufsetzen: http://wiki.otrs.org/index.php?title=Installation_of_OTRS_3.0b1_on_CentOS_5.5 [http://wiki.otrs.org/index.php?title=Installation_of_OTRS_3.0b1_on_CentOS_5.5].

Installation von OTRS auf einem Debian-System

Im OTRS-Wiki finden Sie detaillierte Anweisungen für die Installation unter Debian: http://wiki.otrs.org/index.php?title=Installation_on_Debian_5.04_lenny [http://wiki.otrs.org/index.php?title=Installation_on_Debian_5.04_lenny].

Installation von OTRS auf einem Ubuntu-System

Im OTRS-Wiki finden Sie detaillierte Anweisungen für die Installation unter Ubuntu: [http://wiki.otrs.org/index.php?title=Installation_on_Ubuntu_Lucid_Lynx_\(10.4\)](http://wiki.otrs.org/index.php?title=Installation_on_Ubuntu_Lucid_Lynx_(10.4)) [[http://wiki.otrs.org/index.php?title=Installation_on_Ubuntu_Lucid_Lynx_\(10.4\)](http://wiki.otrs.org/index.php?title=Installation_on_Ubuntu_Lucid_Lynx_(10.4))].

Installation von OTRS auf Microsoft Windows-Systemen

Die Installation von OTRS unter Microsoft Windows ist denkbar einfach. Laden Sie den auf <http://www.otrs.org/downloads/> bereit gestellten Installer herunter und speichern Sie die Datei. Anschließend führen Sie den Installer einfach aus und folgen den einzelnen Installationsschritten. Danach können Sie sich als OTRS-Administrator am System anmelden und dieses weiter konfigurieren. Um sich als Administrator anzumelden, verwenden Sie den Benutzernamen 'root@localhost' und das voreingestellte Passwort 'root'.

Warnung

Bitte ändern Sie das Passwort für den Benutzer 'root@localhost' umgehend.

Wichtig

Der Windows-Installer für OTRS beinhaltet bereits alle Komponenten, die für den Betrieb von OTRS benötigt werden. D.h., es wird zusätzlich zum eigentlichen OTRS der apache2 Webserver, MySQL, Perl mit den für OTRS benötigten Modulen und cron für Windows installiert. Aus diesem Grund ist es empfehlenswert OTRS über den Installer nur auf solchen Windowssystemen zu installieren, auf denen noch kein apache2 bzw. ein anderer Webserver und kein MySQL läuft.

Manuelle Installation (Linux, Unix)

Vorbereiten der manuellen Installation

Wenn Sie OTRS manuell über die Quellen installieren möchten oder müssen, laden Sie sich zuerst das aktuelle Archiv herunter. Sie finden die entsprechenden .tar.gz- oder .tar.bz2-Dateien im Downloadbereich auf <http://www.otrs.org/downloads> [<http://www.otrs.org/downloads/>].

Entpacken Sie das Archiv mit Hilfe von **tar** z. B. in das Verzeichnis `/opt` und benennen Sie das entpackte Verzeichnis von `'otrs-3.0.x'` in `'otrs'` um:

```
linux:/opt# tar xf /tmp/otrs-3.0.tar.gz
linux:/opt# mv otrs-3.0 otrs
linux:/opt# ls
otrs
linux:/opt#
```

Skript 3.4. Erste Schritte bei der Installation von OTRS.

Da die Skripte von OTRS später nicht mit root-Rechten laufen sollen, muss im nächsten Schritt ein Benutzer für OTRS im System angelegt werden. Dieser Benutzer sollte als Homeverzeichnis das Verzeichnis erhalten, in das gerade die Quellen von OTRS entpackt wurden, also `/opt/otrs`. Wird der Webserver unter einem anderen Benutzer als dem OTRS-User betrieben, so muss der neue OTRS-Benutzer noch zur Gruppe des Webserver-Users hinzugefügt werden.

```
linux:/opt# useradd -r -d /opt/otrs/ -c 'OTRS user' otrs
linux:/opt# usermod -G nogroup otrs
linux:/opt#
```

Skript 3.5. Anlegen eines neuen Nutzers 'otrs' und Hinzufügung zu desselben zu einer Gruppe.

Im nächsten Schritt werden einige Demo-Konfigurationsdateien innerhalb der entpackten Quelldateien bzw. innerhalb des Homeverzeichnisses des OTRS-Benutzers kopiert. Die Dateien befinden sich in den Verzeichnissen `/opt/otrs/Kernel` bzw. `/opt/otrs/Kernel/Config` und haben die Endung `.dist`.

```
linux:/opt# cd otrs/Kernel/
linux:/opt/otrs/Kernel# cp Config.pm.dist Config.pm
linux:/opt/otrs/Kernel# cd Config
linux:/opt/otrs/Kernel/Config# cp GenericAgent.pm.dist GenericAgent.pm
```

Skript 3.6. Kopieren von Beispieldateien.

Zum Abschluss der Vorbereitungen werden noch die richtigen Zugriffsrechte für die Dateien des Ticket Systems gesetzt. Dazu kann das Skript **SetPermissions.sh** verwendet werden, das sich im Verzeichnis `bin` innerhalb des Homeverzeichnisses des OTRS-Benutzers befindet. Das Skript kann mit folgenden Parametern aufgerufen werden:

```
otrs.SetPermissions.pl {Homedirectory des OTRS Benutzers} {OTRS Benutzer}
{Webserver Benutzer} [Gruppe des OTRS Benutzers] [Gruppe des Webserver
Benutzers]
```

Läuft Ihr Webserver mit den Benutzerrechten des Benutzers `'otrs'`, dann lautet das Kommando also `otrs.SetPermissions.pl /opt/otrs --otrs-user=otrs --web-user=otrs`. Unter SUSE Linux wird der Webserver mit dem Benutzer `wwwrun` betrieben. Auf Debian-basierten Systemen ist dies `www-data`. Geben Sie hier das Kommando `otrs.SetPermissions.pl /opt/otrs --otrs-user=otrs --web-user=wwwrun --otrs-group=nogroup --web-group=www` ein, um die korrekten Zugriffsrechte zu setzen.

Installation der für OTRS benötigten Perl-Module

Für den Betrieb von OTRS werden einige Perl-Module benötigt. Wenn Sie OTRS manuell einrichten, müssen Sie wahrscheinlich einige dieser Module per Hand nachinstallieren. Dies können Sie entweder über

den Paketmanager ihrer Distribution erledigen (yast, apt-get), oder, wie in diesem Kapitel beschrieben, direkt über die Shell von Perl und CPAN. Wir empfehlen die Verwendung des Paketmanagers, wenn möglich.

Tabelle 3.1. Die folgenden Perl-Module werden von OTRS benötigt.

Name	Beschreibung
DBI	Dieses Modul wird von OTRS für die Verbindung zum Datenbank-Backend benötigt.
DBD::mysql	Modul zum Verbindungsaufbau zum MySQL-Datenbank-Backend (nur benötigt bei Verwendung von MySQL).
DBD::pg	Modul zum Verbindungsaufbau zum PostgreSQL-Datenbank-Backend (nur benötigt bei Verwendung von PostgreSQL).
Digest::MD5	Ermöglicht die Verwendung des md5-Algorithmus.
CSS::Minifier	Komprimiert eine CSS-Datei und schreibt das Ergebnis in eine andere Datei.
Crypt::PasswdMD5	Stellt MD5-basierte kryptographische Funktionen bereit.
MIME::Base64	Kodiert / dekodiert Zeichenketten in Base64, z. B. für E-Mail-Anhänge.
JavaScript::Minifier	Komprimiert eine JavaScript-Datei und schreibt das Ergebnis in eine andere Datei.
Net::DNS	Perl-Interface zum DNS-System.
LWP::UserAgent	Verarbeitet HTTP-Anfragen.
Net::LDAP	LDAP-Schnittstelle für Perl (nur benötigt, wenn LDAP verwendet wird).
GD	Schnittstelle zur Gd Graphics Library. Wird nur benötigt, wenn das Statistikmodul von OTRS verwendet werden soll.
GD::Text, GD::Graph, GD::Graph::lines, GD::Text::Align	Text- und Grafikwerkzeuge für die Benutzung zusammen mit der GD Graphics Library. Diese Komponenten werden nur benötigt, wenn das Statistikmodul von OTRS verwendet werden soll.
PDF::API2, Compress::Zlib	Diese Module werden für die Generierung der PDF-Ausgaben von Statistiken, Suchergebnissen oder Druckansicht eines Tickets benötigt.

Sie können herausfinden, welche Module noch zu installieren sind, indem Sie das Skript **otrs.checkModules** verwenden. Das Skript finden sie im Verzeichnis `bin` innerhalb des Homeverzeichnis des Benutzers 'otrs'.

Beachten Sie, dass einige Module optional sind.

```
linux:~# cd /opt/otrs/bin/
linux:/opt/otrs/bin# ./otrs.CheckModules.pl
  o CGI.....ok (v3.49)
```

```
o Crypt::PasswdMD5.....ok (v1.3)
o CSS::Minifier.....ok (v0.01)
o Date::Format.....ok (v2.24)
o Date::Pcalc.....ok (v1.2)
o DBI.....ok (v1.609)
o DBD::mysql.....ok (v4.013)
o Digest::MD5.....ok (v2.36_01)
o Encode::HanExtra.....ok (v0.23)
o GD.....ok (v2.44)
  o GD::Text.....ok (v0.86)
  o GD::Graph.....ok (v1.44)
  o GD::Graph::lines.....ok (v1.15)
  o GD::Text::Align.....ok (v1.18)
o IO::Scalar.....ok (v2.110)
o IO::Wrap.....ok (v2.110)
o JavaScript::Minifier.....ok (v1.05)
o JSON.....ok (v2.21)
  o JSON::PP.....ok (v2.27003)
  o JSON::XS.....Not installed! (Optional - Install
it for faster AJAX/JavaScript handling.)
o LWP::UserAgent.....ok (v5.829)
o Mail::Internet.....ok (v2.06)
o Mail::POP3Client.....ok (v2.18 )
  o IO::Socket::SSL.....ok (v1.31)
o MIME::Base64.....ok (v3.07_01)
o MIME::Tools.....ok (v5.428)
o Net::DNS.....ok (v0.65)
o Net::POP3.....ok (v2.29)
o Net::IMAP::Simple.....ok (v1.1916)
  o Net::IMAP::Simple::SSL.....ok (v1.3)
o Net::SMTP.....ok (v2.31)
  o Authen::SASL.....ok (v2.15)
  o Net::SMTP::SSL.....ok (v1.01)
o Net::LDAP.....ok (v0.4001)
o PDF::API2.....ok (v0.73)
  o Compress::Zlib.....ok (v2.008)
o SOAP::Lite.....ok (v0.712)
o Text::CSV.....ok (v1.18)
  o Text::CSV_PP.....ok (v1.26)
  o Text::CSV_XS.....Not installed! (Optional -
Optional, install it for faster CSV handling.)
o XML::Parser.....ok (v2.36)
linux:/opt/otrs/bin#
```

Skript 3.7. Überprüfung benötigter Module.

Wenn möglich, sollten Sie die fehlenden Module über das Paketsystem Ihrer Linux-Distribution installieren. Dadurch werden die Pakete automatisch aktualisiert, wenn neue Versionen verfügbar sind oder Sicherheitslücken geschlossen wurden. Bitte schauen Sie dazu in die Dokumentation Ihrer Distribution. Wenn ihr Paket nicht (in der korrekten Version) im Paketsystem nicht verfügbar ist, können Sie es über das CPAN installieren, das Comprehensive Perl Archive Network.

Um eines der oben aufgeführten Module mit Hilfe von CPAN zu installieren, geben Sie als root das Kommando **perl -e shell -MCPAN** ein. Perl wird im interaktiven Modus gestartet und das CPAN Modul

wird geladen. Ist CPAN bereits ordentlich konfiguriert, können Sie die für OTRS benötigten Module mit Hilfe des Kommandos **install** gefolgt vom Modulnamen einrichten. CPAN weist darauf hin, wenn Abhängigkeiten zwischen einzelnen Modulen nicht erfüllt sind und schlägt automatisch die zusätzlich benötigten Module für die Installation vor.

Führen Sie weiterhin die beiden Befehle **perl -cw bin/cgi-bin/index.pl** und **perl -cw bin/PostMaster.pl** aus, nach dem Sie in das Verzeichnis `/opt/otrs` gewechselt sind. Wird bei beiden Befehlen die Meldung "syntax OK" angezeigt, verfügt Ihre Perl-Installation über alle von OTRS benötigten Module und Sie können im nächsten Schritt mit der Einrichtung des Webservers beginnen.

```
linux:~# cd /opt/otrs
linux:/opt/otrs# perl -cw bin/cgi-bin/index.pl
cgi-bin/installer.pl syntax OK
linux:/opt/otrs# perl -cw bin/otrs.PostMaster.pl
otrs.PostMaster.pl syntax OK
linux:/opt/otrs#
```

Skript 3.8. Syntaxprüfung.

Konfiguration des Apache-Webservers

In diesem Abschnitt wird beschrieben, wie der Apache-Webserver grundlegend für OTRS eingerichtet werden muss. Der Webserver sollte cgi- bzw. Perl-Skripte ausführen können, andernfalls ist kein Betrieb von OTRS möglich. Überprüfen Sie die Konfigurationsdateien Ihres Webservers und stellen Sie fest, ob das cgi-Modul geladen wird: Wenn Ihr Webserver die Ausführung von cgi-Skripten unterstützt, sollte eine Zeile ähnlich der folgenden zu finden sein.

```
LoadModule cgi_module /usr/lib/apache2/modules/mod_cgi.so
```

Um die Oberfläche von OTRS bequem erreichen zu können, wird ein Alias- und ein ScriptAlias-Eintrag angelegt. Für die meisten Installationen des apache Webservers gilt, dass ein Verzeichnis mit dem Namen `conf.d` vorhanden ist, unter Linux ist es meist unterhalb des Verzeichnisses `/etc/apache` bzw. `/etc/apache2` zu finden. Wechseln Sie als root in dieses Verzeichnis und kopieren Sie die passende Konfigurationsvorlage in `/opt/otrs/scripts/apache2-httpd.include.conf` in die Datei `otrs.conf`.

Starten Sie Ihren Webserver neu, um die neue Konfiguration zu laden. Auf den meisten Systemen lässt sich der Webserver über den Befehl **/etc/init.d/apache2 restart** neu starten.

```
linux:/etc/apache2/conf.d# /etc/init.d/apache2 restart
Forcing reload of web server: Apache2.
linux:/etc/apache2/conf.d#
```

Skript 3.11. Neustart des Webservers.

Das war bereits die grundlegende Konfiguration des Webservers für OTRS.

Zur Verbesserung der Performance können sie `mod_cgi` deaktivieren und den Apache-Webserver stattdessen für die Benutzung von `mod_perl` konfigurieren.

Stellen Sie bitte sicher, daß `mod_perl` installiert und geladen ist. Andernfalls wird der Webserver-Dienst trotzdem starten, in diesem Fall wird jedoch `mod_cgi` verwendet, wenn es noch aktiviert ist (Deaktivierung wird empfohlen).

Suchen Sie im Verzeichnis `/etc/apache*` nach `mod_perl.so` (siehe Skript 3.12 unten) um herauszufinden, ob es bereits geladen wird, oder nicht.

```
#:/ grep -Rn mod_perl.so /etc/apache*
```

Skript 3.12. Suche nach mod_perl.

Wenn Sie das o. a. geeignete Startskript verwenden, und `mod_perl` geladen wird, kann das Skript `/opt/otrs/scripts/apache2-perl-startup.pl` verwendet werden, um die Perl-Module bereits beim Start des Webservers einmalig vorzuladen. Damit werden die Antwortzeiten kürzer, was die Performance insgesamt verbessert.

Einrichten der Datenbank

Der einfache Weg - Einrichtung der Datenbank mit Hilfe des Webinstallers (nur für MySQL)

Wenn Sie MySQL als Datenbank-Backend verwenden, können Sie den Web-Installer von OTRS verwenden: <http://localhost/otrs/installer.pl> [<http://localhost/otrs/installer.pl>].

Der Webinstaller wird gestartet. Folgen Sie den Anweisungen auf dem Bildschirm.

1. Lesen Sie die Informationen über die OTRS-Niederlassungen und klicken Sie auf "Weiter".

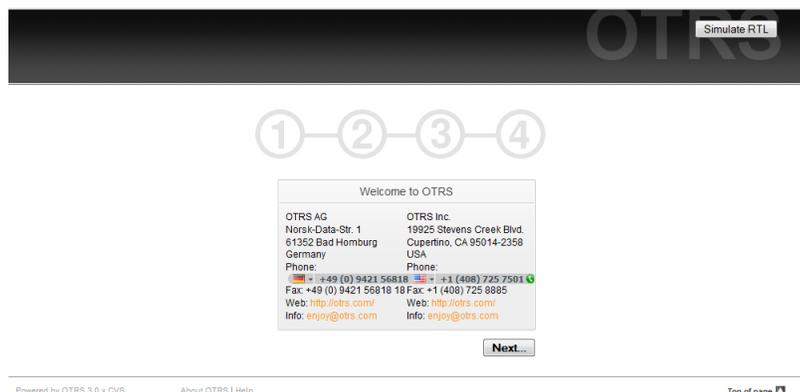


Abb. 3.1. Willkommensbildschirm.

2. Lesen Sie die GNU Affero General Public License und akzeptieren Sie sie durch Klick auf "Akzeptieren".

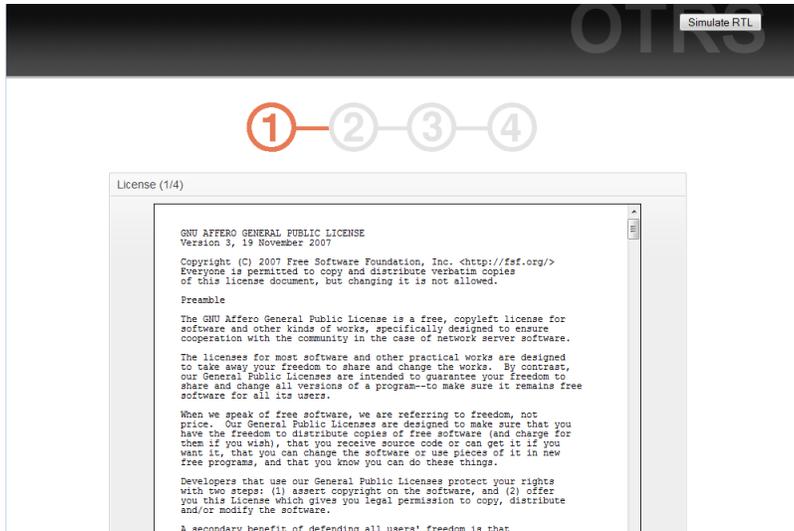


Abb. 3.2. GNU Affero General Public License.

3. Geben Sie Benutzernamen und Passwort für den Administrator an, den DNS-Namen des Servers, auf dem OTRS läuft, und den zu verwendenden Datenbanktyp. Prüfen Sie danach die Einstellungen.

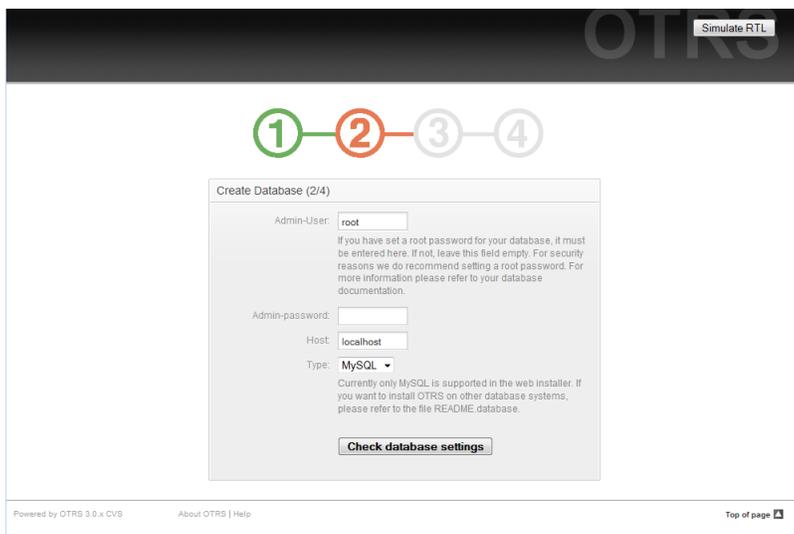


Abb. 3.3. Datenbank-Grundeinstellungen.

Wenn die Prüfung erfolgreich war, erhalten Sie einen Hinweis. Drücken Sie "Ok" zum Fortfahren.

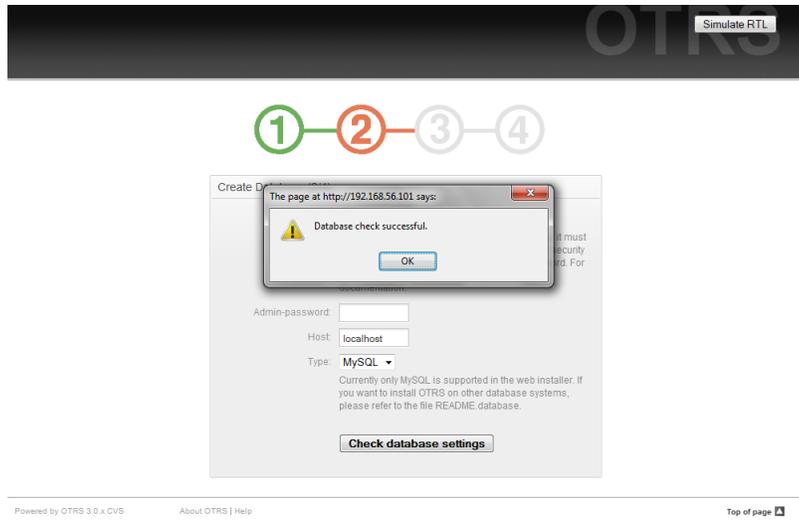


Abb. 3.4. Hinweis auf erfolgreiche Prüfung.

4. Erstellen Sie einen neuen Datenbankbenutzer, wählen Sie einen Namen für die Datenbank und klicken Sie auf "Weiter".

Warnung

Man sollte nie Standard-Passwörter verwenden. Bitte ändern Sie das Standardpasswort der OTRS-Datenbank!

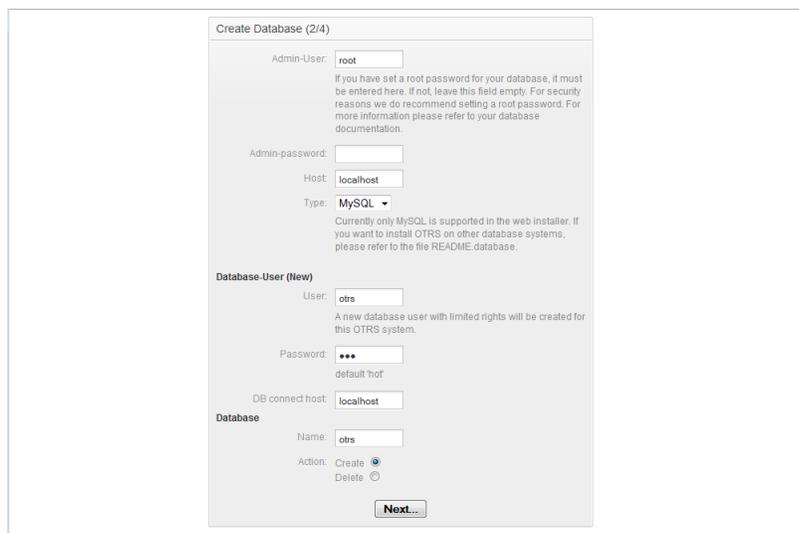


Abb. 3.5. Datenbankeinstellungen.

Wenn der Datenbankbenutzer und die Datenbank erfolgreich angelegt wurden, erhalten Sie eine Bestätigung. Klicken Sie auf 'Weiter', um fortzufahren.

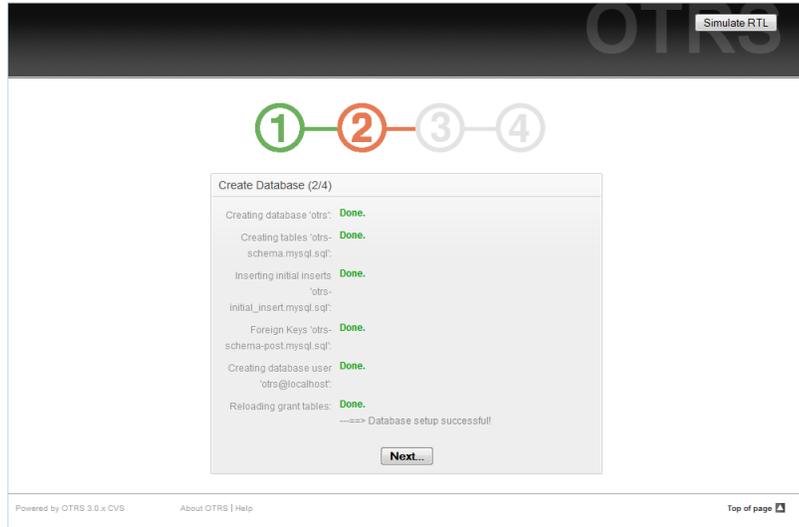


Abb. 3.6. Bestätigung der erfolgreichen Datenbankeinrichtung.

5. Geben Sie alle benötigten Systemeinstellungen an und klicken Sie "Weiter".

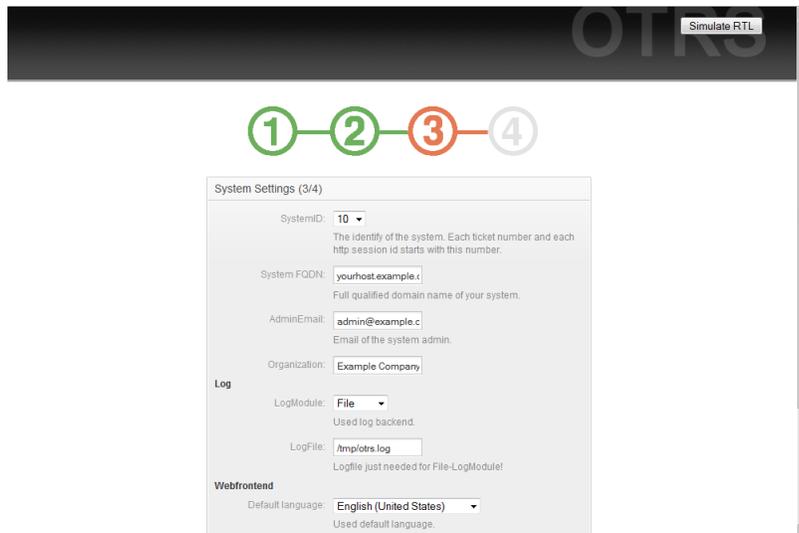


Abb. 3.7. Systemeinstellungen.

6. Wenn Sie wollen, können Sie an dieser Stelle eingehende und ausgehende E-Mail-Verbindungen einrichten, oder diesen Schritt überspringen.

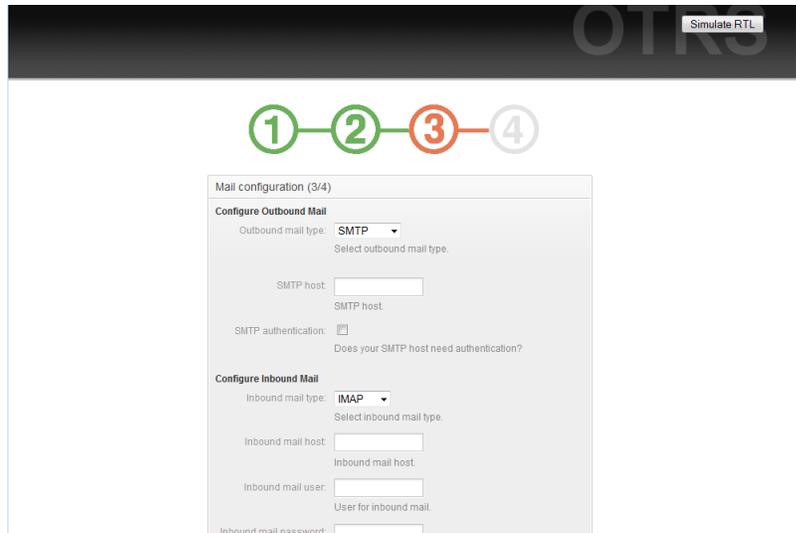


Abb. 3.8. E-Mail-Konfiguration.

7. Starten Sie den OTRS-Dienst neu, damit die neuen Konfigurationseinstellungen verwendet werden.

```
linux:~ # rcotrs restart-force
Shutting down OTRS
  Disable /opt/otrs/bin/otrs.PostMaster.pl ... done.
no crontab for otrs
Shutting down cronjobs ... failed!
Shutting down OTRS (completely)
  Shutting down Apache ... done.
  Shutting down MySQL ... done.

done
Starting OTRS (completely)
  Starting Apache ... done.
  Starting MySQL ... done.
Starting OTRS
  Checking Apache ... done.
  Checking MySQL ... done.
  Checking database connect... (It looks Ok!).
  Enable /opt/otrs/bin/otrs.PostMaster.pl ... done.
  Checking otrs spool dir... done.
  Creating cronjobs (source /opt/otrs/var/cron/*) ... done.

-->> http://linux.example.com/otrs/index.pl <<--

done

done
linux:~ #
```

Skript 3.13. Neustart des OTRS-Dienstes.

Herzlichen Glückwunsch! Die Installation ist abgeschlossen, jetzt können Sie mit dem System arbeiten. Um sich anzumelden, verwenden Sie bitte die Adresse `http://localhost/otrs/index.pl` [`http://`

localhost/otrs/index.pl] in ihrem Webbrowser. Melden Sie sich als Administrator an, mit dem Benutzernamen 'root@localhost' und dem Passwort 'root'. Anschließend können Sie das System gemäß Ihrer Anforderungen konfigurieren.

Warnung

Bitte ändern Sie das Passwort für den Benutzer 'root@localhost' umgehend.

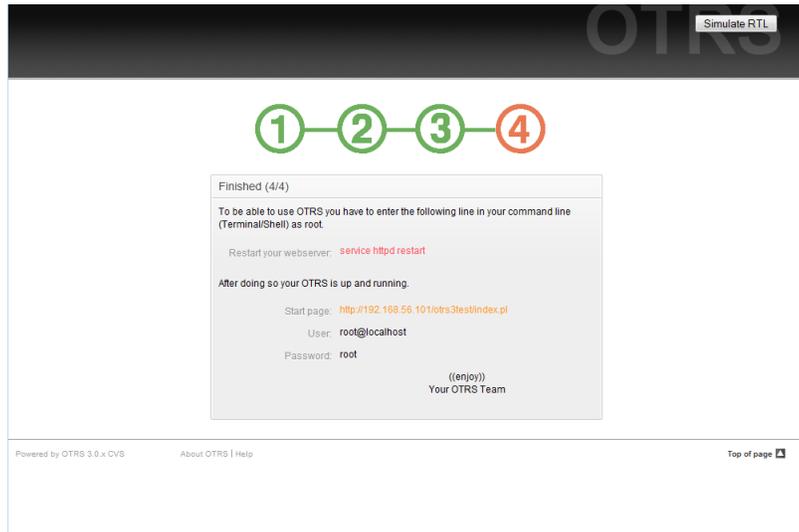


Abb. 3.9. Letzte Schritte zur Installation von OTRS.

Manuelle Installation der OTRS Datenbank

Wenn Sie den Webinstaller nicht einsetzen können, kann die Datenbank für OTRS auch manuell eingerichtet werden. Skripte mit dem Datenbankschema und den SQL-Statements finden Sie im Verzeichnis `scripts/database` innerhalb des Homeverzeichnisses des Benutzers 'otrs'.

```
linux:~# cd /opt/otrs/scripts/database/
linux:/opt/otrs/scripts/database# ls
otrs-initial_insert.db2.sql          otrs-schema.mysql.sql
otrs-schema.oracle.sql
otrs-initial_insert.mssql.sql        otrs-schema-post.db2.sql
otrs-initial_insert.mysql.sql        otrs-schema.postgresql.sql
otrs-initial_insert.oracle.sql
otrs-initial_insert.postgresql.sql   otrs-schema-post.mssql.sql
otrs-initial_insert.xml              otrs-schema-post.mysql.sql
otrs-schema.db2.sql                 otrs-schema-post.oracle.sql
  otrs-schema-post.postgresql.sql
otrs-schema.mssql.sql               otrs-schema.xml
linux:/opt/otrs/scripts/database#
```

Skript 3.14. Für die Datenbankeinstellung benötigte Dateien.

Für die verschiedenen Datenbanktypen sind mehrere .sql-Dateien vorhanden, die nacheinander abgearbeitet werden müssen.

Die OTRS-Datenbank manuell Schritt für Schritt anlegen

1. Anlegen der Datenbank für OTRS: Legen Sie mit Hilfe Ihres Datenbankinterfaces bzw. Ihrer Datenbankoberfläche die Datenbank an, die später von OTRS verwendet werden soll.
2. Erstellen der Tabellen: Mit Hilfe der `otrs-schema.Datenbanktyp.sql`-Dateien (z. B. `otrs-schema.oracle.sql`, `otrs-schema.postgresql.sql`, usw.) können Sie die Tabellen innerhalb der OTRS Datenbank erzeugen.
3. Einfügen der vom System benötigten Daten: Damit OTRS richtig funktioniert, müssen einige Daten in verschiedene Tabellen geschrieben werden (z. B. die verschiedenen Ticketstatus, Ticket- und Benachrichtigungstypen, etc.). Verwenden Sie entweder die `dateiotrs-initial_insert.mysql.sql`, `otrs-initial_insert.db2.sql`, `otrs-initial_insert.oracle.sql`, `otrs-initial_insert.postgresql.sql` oder `otrs-initial_insert.mssql.sql` zum Einspielen der Daten, je nachdem welche Datenbank verwendet wird.
4. Erzeugen von "foreign keys" auf andere Tabellen: Abschließend müssen noch die "foreign keys" erstellt werden, über die die verschiedenen Tabellen in der OTRS Datenbank voneinander abhängen. Dies kann mit Hilfe der `otrs-schema-post.Datenbanktyp.sql`-Dateien erreicht werden (z. B. `otrs-schema-oracle.post.sql`, `otrs-schema-post.postgresql.sql`, usw.).

Nachdem Sie die Datenbank angelegt haben, sollten Sie die Zugriffsrechte dafür setzen und z. B. sicherstellen, dass nur ein bestimmter Benutzer ohne Datenbank-Administrationsrechte Zugriff auf die OTRS Datenbank hat. Je nachdem, welche Datenbank Sie einsetzen, unterscheiden sich hier die Vorgehensweisen, es sollte jedoch möglich sein dies mit Hilfe Ihres Datenbankinterfaces bzw. Ihrer Datenbankoberfläche zu erledigen.

Wurden die nötigen Einstellungen für die Datenbank vorgenommen, muss nun noch dem Ticket-System mitgeteilt werden, welche Datenbank es verwenden soll. Öffnen Sie die Datei `Kernel/Config.pm` innerhalb des Homeverzeichnis des OTRS Benutzers und passen Sie die dafür vorgesehenen Parameter an.

```
# DatabaseHost
# (The database host.)
$self->{'DatabaseHost'} = 'localhost';

# Database
# (The database name.)
$self->{Database} = 'otrs';

# DatabaseUser
# (The database user.)
$self->{DatabaseUser} = 'otrs';

# DatabasePw
# (The password of database user.)
$self->{DatabasePw} = 'some-pass';
```

Skript 3.15. Anzupassende Einstellungen.

Einrichten der von OTRS benötigten cron-Jobs

Damit OTRS voll funktioniert, werden einige cron-Jobs benötigt. Die cron-Jobs sollten mit denselben Benutzerrechten ausgeführt werden, die auch für die restlichen OTRS-Skripte vergeben wurden, d.h. die cron-Jobs sollten in die crontab des Benutzers 'otrs' eingetragen werden.

Alle Skripte für die verschiedenen cron-Jobs befinden sich im Verzeichnis `var/cron` innerhalb des Homeverzeichnis des Benutzers 'otrs'.

```
linux:~# cd /opt/otrs/var/cron
linux:/opt/otrs/var/cron# ls
aaa_base.dist          generic_agent.dist
  rebuild_ticket_index.dist
cache.dist             pending_jobs.dist    session.dist
fetchmail.dist         postmaster.dist      unlock.dist
generic_agent-database.dist  postmaster_mailbox.dist
linux:/opt/otrs/var/cron#
```

Skript 3.16. Für die Erstellung der Cronjobs benötigte Dateien.

Alle Skripte tragen die Endung `.dist` und sollten zunächst so umkopiert werden, dass keine Endung mehr vorhanden ist.

```
linux:/opt/otrs/var/cron# for foo in *.dist; do cp $foo `basename
  $foo .dist`; done
linux:/opt/otrs/var/cron# ls
aaa_base          generic_agent-database.dist
  rebuild_ticket_index
aaa_base.dist     generic_agent.dist
  rebuild_ticket_index.dist
cache             pending_jobs          session
cache.dist       pending_jobs.dist    session.dist
fetchmail         postmaster            unlock
fetchmail.dist   postmaster.dist      unlock.dist
generic_agent     postmaster_mailbox
generic_agent-database  postmaster_mailbox.dist
linux:/opt/otrs/var/cron#
```

Skript 3.17. Kopieren und Umbenennen der für die Erstellung der Cronjobs benötigten Dateien.

Die folgende Tabelle gibt eine kurze Übersicht über die Aufgabe der verschiedenen Skripte.

Tabelle 3.2. Die verschiedenen Skripte für die cron-Jobs von OTRS.

Skript	Funktion
aaa_base	Über dieses Skript werden die grundlegenden Einstellungen für die crontab des Benutzers 'otrs' festgelegt.
cache	Löscht abgelaufene Cache-Einträge von der Festplatte. Leert den Cache von CSS- und JavaScript-Dateien.

Skript	Funktion
fetchmail	Falls Nachrichten mit Hilfe von fetchmail in das System eingespeist werden sollen, kann dieses Skript verwendet werden.
generic_agent	Mit Hilfe dieses Skripts werden die Jobs des GenericAgents ausgeführt, die über eigene Konfigurationsdateien festgelegt wurden.
generic_agent-database	Mit Hilfe dieses Skripts werden die Jobs des GenericAgents ausgeführt, die über dem Administrations-Bereich innerhalb von "GenericAgent" angelegt wurden.
pending_jobs	Mit Hilfe dieses Skripts wird das System auf "wartende" (pending) Tickets überprüft.
postmaster	Mit Hilfe dieses Skripts wird die Nachrichten-Warteschlange von OTRS überprüft und noch nicht verarbeitete Nachrichten werden im System gespeichert bzw. zugestellt.
postmaster_mailbox	Mit Hilfe dieses Skripts werden die verschiedenen pop3-Konten abgefragt, die im Administrations-Bereich innerhalb von "PostMaster POP3 Account" eingerichtet wurden.
rebuild_ticket_index	Mit Hilfe dieses Skripts wird der Ticket-Index für die Queue-Ansicht neu erzeugt, wodurch die Anzeige beschleunigt wird.
session	Über dieses Skript werden alte und nicht mehr gültige Session-IDs entfernt.
unlock	Mit Hilfe dieses Skripts wird die Freigabe von Tickets innerhalb des Systems ermöglicht.

Für die Einrichtung aller cron-Jobs kann das Skript `bin/Cron.sh` verwendet werden, das sich im Homeverzeichnis des OTRS-Benutzers befindet. Dem Skript muss beim Aufruf ein Parameter übergeben werden. Dieser Parameter legt fest, ob die cron-Jobs installiert, deinstalliert oder neu gestartet werden. Es sind folgende Parameter zulässig:

```
Cron.sh { start } { stop } { restart } [ OTRS-Benutzer ]
```

Da die cron-Jobs für den Benutzer 'otrs' angelegt werden sollen, muss das Skript von diesem Benutzer ausgeführt werden. Sind Sie z. B. als Benutzer root am System angemeldet, können Sie mit Hilfe des Kommandos `su otrs` zum OTRS-Benutzer wechseln. Nehmen Sie also die Installation wie folgt vor.

Warnung

Bitte beachten Sie, dass durch die Verwendung von `Cron.sh` evtl. andere Cron-Jobs des OTRS-Benutzers überschrieben bzw. gelöscht werden. Um weitere, nicht von OTRS benötigte Cron-Jobs für den OTRS-Benutzer zu installieren, erweitern Sie bitte `Cron.sh` dementsprechend.

```
linux:/opt/otrs/var/cron# cd /opt/otrs/bin/
linux:/opt/otrs/bin# su otrs
linux:~/bin$ ./Cron.sh start
/opt/otrs/bin
```

```
Cron.sh - start/stop OTRS cronjobs
Copyright (C) 2001-2009 OTRS AG, http://otrs.org/
(using /opt/otrs) done
linux:~/bin$ exit
exit
linux:/opt/otrs/bin#
```

Skript 3.18. Installation der Cronjobs.

Mit Hilfe des Kommandos **crontab -l -u otrs**, das Sie als root ausführen können, wird die crontab-Datei des Benutzers 'otrs' angezeigt und Sie können überprüfen, ob alle Einträge vorhanden sind.

```
linux:/opt/otrs/bin# crontab -l -u otrs
# --
# cron/aaa_base - base crontab package
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# Who gets the cron emails?
MAILTO="root@localhost"

# --
# cron/cache - delete expired cache
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# delete expired cache weekly (Sunday mornings)
20 0 * * 0 $HOME/bin/otrs.CacheDelete.pl --expired >> /dev/null
30 0 * * 0 $HOME/bin/otrs.LoaderCache.pl -o delete >> /dev/null

# --
# cron/fetchmail - fetchmail cron of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# fetch every 5 minutes emails via fetchmail
#*/5 * * * * /usr/bin/fetchmail -a >> /dev/null

# --
# cron/generic_agent - otrs.GenericAgent.pl cron of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# start generic agent every 20 minutes
*/20 * * * * $HOME/bin/GenericAgent.pl >> /dev/null
# example to execute GenericAgent.pl on 23:00 with
# Kernel::Config::GenericAgentMove job file
#0 23 * * * $HOME/bin/otrs.GenericAgent.pl -c
"Kernel::Config::GenericAgentMove" >> /dev/null
```

```
# --
# cron/generic_agent - GenericAgent.pl cron of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# start generic agent every 10 minutes
*/10 * * * * $HOME/bin/otrs.GenericAgent.pl -c db >> /dev/null
# --
# cron/pending_jobs - pending_jobs cron of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# check every 120 min the pending jobs
45 */2 * * * * $HOME/bin/otrs.PendingJobs.pl >> /dev/null
# --
# cron/postmaster - postmaster cron of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# check daily the spool directory of OTRS
#10 0 * * * * test -e /etc/init.d/otrs & /etc/init.d/otrs cleanup
>> /dev/null; test -e /etc/rc.d/init.d/otrs && /etc/rc.d/init.d/otrs
cleanup >> /dev/null
10 0 * * * * $HOME/bin/otrs.CleanUp.pl >> /dev/null
# --
# cron/postmaster_mailbox - postmaster_mailbox cron of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# fetch emails every 10 minutes
*/10 * * * * $HOME/bin/otrs.PostMasterMailbox.pl >> /dev/null
# --
# cron/rebuild_ticket_index - rebuild ticket index for OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# just every day
01 01 * * * * $HOME/bin/otrs.RebuildTicketIndex.pl >> /dev/null

# --
# cron/session - delete old session ids of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# delete every 120 minutes old/idle session ids
55 */2 * * * * $HOME/bin/otrs.DeleteSessionIDs.pl --expired >> /dev/null

# --
```

```
# cron/unlock - unlock old locked ticket of the OTRS
# Copyright (C) 2001-2011 OTRS AG, http://otrs.org/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# unlock every hour old locked tickets
35 * * * * $HOME/bin/otrs.UnlockTickets.pl --timeout >> /dev/null

linux:/opt/otrs/bin#
```

Skript 3.19. Crontab-Datei.

Upgrade des OTRS Frameworks

Diese Hinweise sind für Leute gedacht, die von OTRS 2.4 auf 3.0 upgraden, und gilt für RPM-basierte und manuelle Upgrades (tarball).

Wenn Sie eine ältere Version von OTRS einsetzen, müssen sie zuerst dem Upgrade-Pfad zur 2.4 folgen (1.1->1.2->1.3->2.0->2.1->2.2->2.3->2.4->>3.0 ...).

Wenn Sie ein "Patchlevel-Update" durchführen wollen, z. B. von OTRS 3.0.3 auf 3.0.4, dann sollten Sie die Schritte 8, 9, 10 und 13 - 16 auslassen.

1. Stoppen Sie alle relevanten Dienste.

z. B. (hängt von eingesetzten Diensten ab):

```
shell> /etc/init.d/cron stop
shell> /etc/init.d/postfix stop
shell> /etc/init.d/apache stop
```

2. Sichern Sie folgende Daten in \$OTRS_HOME (Standard: OTRS_HOME=/opt/otrs):

- Kernel/Config.pm
- Kernel/Config/GenericAgent.pm
- Kernel/Config/Files/ZZZAuto.pm
- var/*

3. Sichern Sie die Datenbank.

4. Neues System aufsetzen (optional). Wenn möglich, installieren Sie zunächst auf einer separaten Testmaschine.

5. Installieren Sie das neue Release (tar oder RPM).

- Mit dem Tarball:

```
shell> cd /opt
shell> tar -xzf otrs-x.x.x.tar.gz
```

```
shell> ln -s otrs-x.x.x otrs
```

Alte Configdateien wiederherstellen.

- Kernel/Config.pm
- Kernel/Config/GenericAgent.pm
- Kernel/Config/Files/ZZZAuto.pm
- Per RPM:

```
shell> rpm -Uvh otrs-x.x.x-01.rpm
```

In diesem Fall stellt das RPM-Update die alten Konfigurationsdateien automatisch wieder her.

6. Eigene Themes

Hinweis: Die OTRS-Themes von 2.4 sind nicht zu 3.0 kompatibel, verwenden Sie ihre alten Themes daher nicht weiter! (Themes befinden sich in /opt/otrs/Kernel/Output/HTML/*/*.dtl)

7. Setzen Sie die Datei-Berechtigungen.

wenn der Tarball benutzt wird, führen Sie

```
shell> cd /opt/otrs/  
shell> bin/otrs.SetPermissions.pl
```

aus, um die Berechtigungen zu setzen, die für Ihr System benötigt werden.

8. Wenden Sie die Datenbankänderungen an (Teil 1/2):

```
shell> cd /opt/otrs/  
  
# MySQL:  
shell> cat scripts/DBUpdate-to-3.0.mysql.sql | mysql -p -f -u root  
otrs  
# PostgreSQL:  
shell> cat scripts/DBUpdate-to-3.0.postgresql.sql | psql otrs
```

9. Führen Sie das Migrationsskript aus (als OTRS-Benutzer, nicht als root!):

Sie müssen das Migrationsskript ausführen, um die Daten auf die neue Datenbankstruktur zu übernehmen. Bitte führen Sie

```
shell> scripts/DBUpdate-to-3.0.pl
```

aus.

10.Datenbankänderungen anwenden (Teil 2/2):

```
# MySQL:
shell> cat scripts/DBUpdate-to-3.0-post.mysql.sql | mysql -p -f -u
root otrs
# PostgreSQL:
shell> cat scripts/DBUpdate-to-3.0-post.postgresql.sql | psql otrs
```

11.Systemkonfiguration aktualisieren und Caches löschen. Bitte führen Sie aus:

```
shell> bin/otrs.RebuildConfig.pl
shell> bin/otrs.DeleteCache.pl
```

12.Starten Sie Ihre Dienste neu.

z. B. (hängt von eingesetzten Diensten ab):

```
shell> /etc/init.d/cron stop
shell> /etc/init.d/postfix stop
shell> /etc/init.d/apache stop
```

13.Prüfen Sie auf mögliche Kodierungsprobleme:

Mit OTRS 3.0 wurde der Standardzeichensatz von "ISO-8859-1" auf "utf-8" umgestellt.

Von dieser Änderung sind Sie nur betroffen, wenn sie in Kernel/Config.pm keinen Zeichensatz eingestellt haben (alle Installationen, die unter Verwendung des Web-Installers angelegt wurden, haben dort einen konfigurierten Wert), und sie den vorkonfigurierten Wert nicht in AdminSysconfig geändert haben.

Wenn Sie mit dem neuen Zeichensatz Probleme feststellen sollten, dann fügen Sie diese Zeile zu Kernel/Config.pm hinzu:

```
$Self->{'DefaultCharset'} = 'iso-8859-1';
```

Generell ist "utf-8" der empfohlene Zeichensatz für den Betrieb von OTRS, und die Umstellung von "iso-8859-1" sollte problemlos verlaufen.

Hinweis: Wir empfehlen, alle existierenden nicht-UTF-8-Installationen bereits jetzt mit dem Update auf 3.0 auf UTF-8 umzustellen. Andere Zeichensätze sind ab sofort im Status "veraltet / nicht empfohlen".

OTRS 3.1 wird nur noch UTF-8 als internen Zeichensatz unterstützen.

14.Auf angepasste Frontend-Modulregistrierungen prüfen

Seit OTRS 3.0 können Applikationsmodule ihr eigenes spezielles CSS und/oder Javascript haben. Dieses wird als Teil der sog. Frontend-Modulregistrierung der Module in der SysConfig angegeben. Das kann Probleme beim Upgrade verursachen, weil lokal geänderte Frontend-Modulregistrierungen die Updates der neuen Standardkonfiguration nicht erhalten.

Ob Sie betroffen sind, können sie herausfinden, indem Sie sich `Kernel/Config/Files/ZZZAuto.pm` ansehen und nach Einträgen wie

```
$Self->{'Frontend::Module'}->{'AgentStats'} = { ... };
```

suchen

Diese Einträge sollten auf ihren Standardwert zurückgesetzt werden, in Admin -> SysConfig (in diesem Fall: 'Frontend::Module###AgentStats') mit dem kleinen "Zurücksetzen"-Knopf neben dem Optionsfeld "Aktiv". Dann werden die Einträge zurückgesetzt auf ihren Standardwert und tauchen demzufolge nicht mehr in `Kernel/Config/Files/ZZZAuto.pm` auf. Nach dieser Maßnahme können Sie die Einträge wieder anpassen, falls erforderlich.

Dieselbe Prozedur muss für den Eintrag "Frontend::ToolBarModule###1-Ticket::TicketSearchFulltext" durchgeführt werden, weil dieses auch modulspezifisches CSS verwendet. Wenn diese Einstellung auf Ihrem System nicht aktiv ist, können Sie diesen Schritt überspringen.

15.Auf angepasste PreferencesGroups-Einträge prüfen

In OTRS 3.0 gab es einige Korrekturen in den PreferencesGroups-Konfigurationseinträgen. Die Schlüssel 'Activ' und 'Colum' wurden in 'Active' und 'Column' geändert. Sie müssen sich darum nur kümmern, wenn Sie solche Einträge in Ihrem System angepasst haben, ansonsten werden sie automatisch aktualisiert.

Ob Sie betroffen sind, können sie herausfinden, indem Sie sich `Kernel/Config/Files/ZZZAuto.pm` ansehen und nach Einträgen wie

```
$Self->{'CustomerPreferencesGroups'}->{'RefreshTime'} = {
  'Activ' => '0',
  'Colum' => 'Frontend',
  'Data' => {
    '' => 'off',
    '10' => '10 minutes',
    '15' => '15 minutes',
    '2' => ' 2 minutes',
    '5' => ' 5 minutes',
    '7' => ' 7 minutes'
  },
  'Desc' => 'Select your QueueView refresh time.',
  'Label' => 'QueueView refresh time',
  'Module' => 'Kernel::Output::HTML::PreferencesGeneric',
  'PrefKey' => 'UserRefreshTime',
  'Prio' => '4000'
};
```

suchen. Diese müssen wie folgt verändert werden:

```
$Self->{'CustomerPreferencesGroups'}->{'RefreshTime'} = {
  'Active' => '0',
```

```

'Column' => 'Frontend',
'Data' => {
  '' => 'off',
  '10' => '10 minutes',
  '15' => '15 minutes',
  '2' => ' 2 minutes',
  '5' => ' 5 minutes',
  '7' => ' 7 minutes'
},
'Desc' => 'Select your QueueView refresh time.',
'Label' => 'QueueView refresh time',
'Module' => 'Kernel::Output::HTML::PreferencesGeneric',
'PrefKey' => 'UserRefreshTime',
'Prio' => '4000'
};

```

16.Feld-Umbenennung im Kunden-Datenbankbackend

Das Feld 'salutation' in der eingebauten Kundendatenbank wurde umbenannt in das angemessenere 'title'. Falls Sie die eingebaute Datenbank als Backend für Kundenverwaltung benutzen, und Sie die Konfiguration angepasst haben, z. B. weil Sie Felder zu der Datenbank hinzugefügt haben, oder die Unterstützung für Kunden-Unternehmen aktiviert haben, sollten Sie das Mapping in Ihrer Kernel/Config.pm anpassen:

Verändern Sie diese Zeile:

```
[ 'UserSalutation', 'Salutation', 'salutation', 1, 0, 'var',
'', 0 ],
```

folgendermaßen:

```
[ 'UserTitle', 'Title', 'title', 1, 0, 'var',
'', 0 ],
```

Davon können auch Anreden oder andere Templates betroffen sein, in dem Sie die Zeichenkette <OTRS_*_UserSalutation> verwendet haben. Wenn Sie diese Zeichenkette verwendet haben UND die interne Kundendatenbank einsetzen, sollten Sie <OTRS_CURRENT_UserTitle> stattdessen verwenden.

17.Optional: Tickets als gelesen markieren

In OTRS 3.0 gibt es eine neue Funktion: neue Tickets und neue Artikel, die ein Agent noch nicht gelesen hat, werden als 'ungelesen' gekennzeichnet. Für alle Tickets, die vor dem Upgrade auf OTRS 3.0 erstellt wurden, fehlt die 'gelesen'-Information jedoch, daher werden diese Tickets als 'ungelesen' dargestellt. Wenn das Sie stört, können Sie ein Skript verwenden, welches alle Tickets und alle Artikel im System für alle Agenten mit Leserechten auf 'gelesen' setzt. Beachten Sie, dass dieses Skript eine ganze Weile für seine Arbeit benötigen kann!

```
shell> bin/otrs.MarkTicketAsSeen.pl
```

18.Gut gemacht!

Upgrade mit dem Windows-Installer

Derzeit gibt es keinen automatischen Upgrademechanismus für OTRS-Installationen, die mit dem Windows-Installer erstellt wurden. Das Upgrade besteht im Wesentlichen aus einem Backup der Datenbank und des Filesystems, dem Deinstallieren von OTRS, der Installation der neuere Version, der Wiederherstellung der Datenbank und dem Ausführen der Upgrade-Prozedur.

Das Upgrade wird beschrieben in FAQ# 4200351 [<http://faq.otrs.org/otrs/public.pl?Action=PublicFAQ;ItemID=351>], und es gibt zudem ein informatives YouTube-Video [<http://www.youtube.com/watch?v=sf0R-reMTWc>] available.

Kapitel 4. Erste Schritte

Dieser Abschnitt soll einen ersten Überblick über die Funktionsweise von OTRS und den Aufbau der Weboberfläche des Systems geben. Es werden die Unterschiede zwischen Mitarbeitern (Agents), Kunden (Customer) und Administratoren erklärt. Anschließend wird die erste Anmeldung als OTRS-Administrator durchgeführt und erläutert, was mit Hilfe der Benutzereinstellungen für jeden Account innerhalb des Systems festgelegt werden kann.

Agenten-Weboberfläche

Die Mitarbeiter bearbeiten über die Weboberfläche des Systems die Anfragen der Kunden, erstellen neue Tickets für andere Mitarbeiter oder Kunden, legen Tickets über Telefongespräche mit Kunden an, schreiben FAQ-Einträge, bearbeiten Kundendaten usw.

Angenommen, Ihr Host mit der OTRS-Installation ist über die Domain `http://www.example.com` [`http://www.example.com/`] erreichbar, so kann der Login-Bildschirm für die Mitarbeiter und somit auch für den OTRS-Administrator über die URL `http://www.example.com/otrs/index.pl` [`http://www.example.com/otrs/index.pl`] aufgerufen werden.

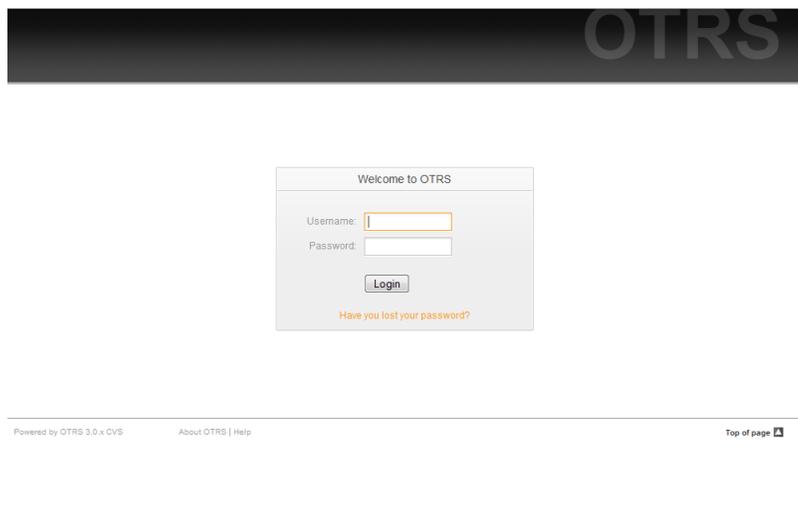


Abb. 4.1. Login-Bildschirm der Agenten-Weboberfläche.

Kunden-Weboberfläche

Kunden können über das speziell für sie vorhandene Webinterface von OTRS einen eigenen Kundenaccount anlegen, die eigenen Tickets einsehen, Tickets erstellen und bearbeiten, die Account-Einstellungen anpassen usw.

Gemäß dem obigen Beispiel kann der Login-Bildschirm für die Kunden über die URL `http://www.example.com/otrs/customer.pl` [`http://www.example.com/otrs/customer.pl`] geladen werden.

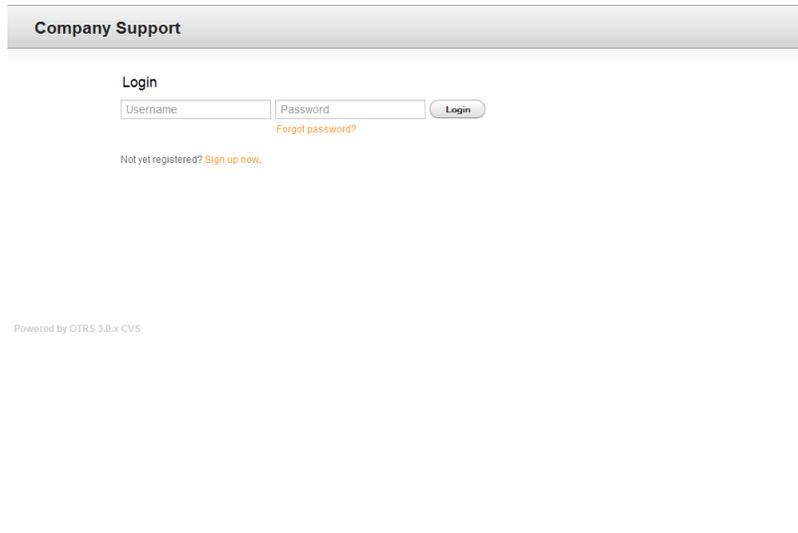


Abb. 4.2. Login-Bildschirm der Kunden-Weboberfläche.

Öffentliche Weboberfläche

Neben diesen beiden Bereichen der Weboberfläche verfügt OTRS weiterhin über ein Interface, welches vom FAQ-Modul bereit gestellt wird (muss separat installiert werden). Es ist öffentlich, also ohne Anmeldung, zugänglich und bietet Zugriff auf das FAQ-System.

Das öffentliche Web-Interface ist über die URL <http://www.example.com/otrs/faq.pl> [<http://www.example.com/otrs/faq.pl>] erreichbar. Ebenfalls ist ein Zugriff auf die FAQ über die URL <http://www.example.com/otrs/public.pl> [<http://www.example.com/otrs/public.pl>] möglich

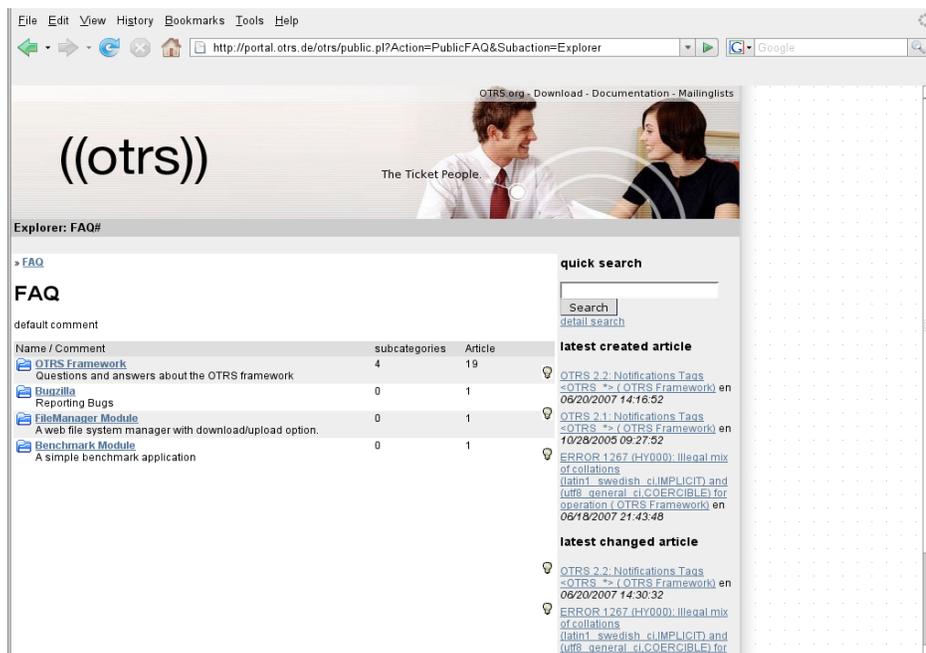


Abb. 4.3. Öffentliche Weboberfläche.

Die erste Anmeldung

Der Zugriff auf den Login-Bildschirm wird im Abschnitt *Agenten-Weboberfläche* beschrieben. Hier haben Sie die Möglichkeit, einen Benutzernamen und ein Kennwort anzugeben. Um sich als OTRS-Administrator anzumelden, verwenden Sie als Benutzername "root@localhost" und als Kennwort "root".

Warnung

Diese Zugangsdaten werden bei jeder OTRS-Installation standardmäßig vergeben. Da das Kennwort für den OTRS-Administrator somit öffentlich bekannt ist, sollten Sie es schnellstmöglich ändern! Sie können dies nach der Anmeldung als OTRS-Administrator über die Benutzereinstellungen vornehmen.

Wollen Sie sich nicht als OTRS-Administrator anmelden, geben Sie einfach den Benutzernamen und das Kennwort Ihres normalen OTRS-Accounts in die dafür vorgesehenen Eingabefelder ein.

Falls Sie einmal ihr Kennwort vergessen haben sollten, können Sie sich automatisch vom System ein neues Kennwort an die Mailadresse schicken lassen, die für Ihren OTRS-Account im System hinterlegt ist. Geben Sie dazu im unteren Bereich des Login-Bildschirms den Benutzernamen Ihres Accounts an.

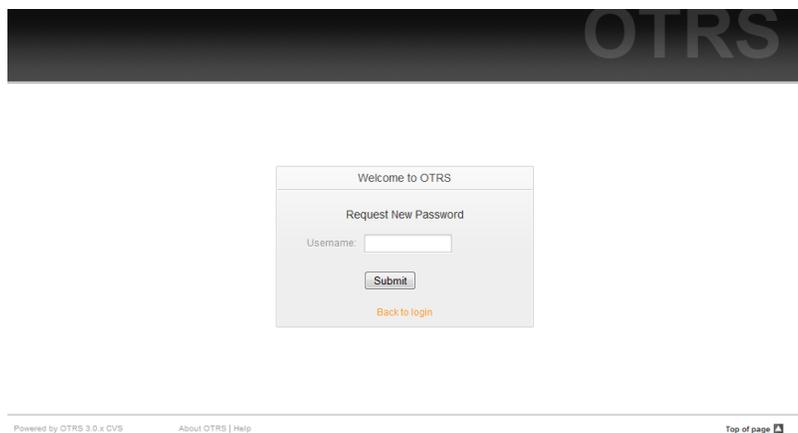


Abb. 4.4. Neues Passwort anfordern.

Agenten-Weboberfläche - ein Überblick

Nachdem Sie sich erfolgreich am System angemeldet haben, wird die Oberfläche von OTRS geladen. Standardmäßig befinden Sie sich nach der Anmeldung im Dashboard. Dieses Dashboard ist frei anpassbar. Zu Beginn zeigt es Ihre gesperrten Tickets und bietet Übersichten über fällige, eskalierte, neue und offene Tickets, neben anderen Dingen.

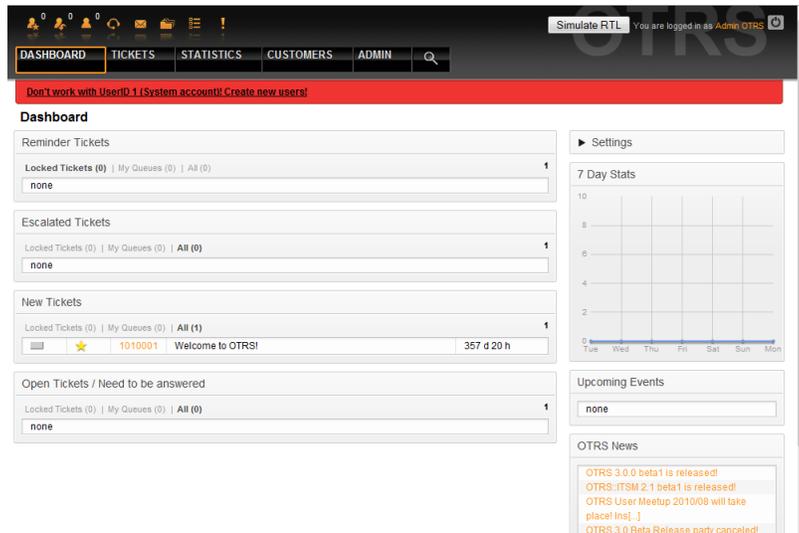


Abb. 4.5. Dashboard der Agenten-Weboberfläche.

Um die Übersichtlichkeit zu erhöhen, wurde die Oberfläche von OTRS in verschiedene Bereiche aufgeteilt. Oben finden sich einige allgemeine Informationen, wie der aktuelle Benutzername, der Abmelden-Knopf, und die Menge der gesperrten Tickets.

Unter diesen Icons befindet sich die Navigationsleiste. Sie zeigt Knöpfe, mit denen Sie die verschiedenen Bereiche und Module des Systems ansteuern und globale Aktionen auslösen können. Der Dashboard-Knopf führt zum Dashboard. Wenn sie den Ticket-Knopf drücken, erhalten Sie ein Untermenü mit verschiedenen Operationen, wie den Ticketansichten, der Erstellung neuer Tickets oder der Ticketsuche. Der Statistik-Knopf ist auch ein Menü, in dem Sie zwischen einer Übersicht der verfügbaren Statistiken, der Neuerstellung oder dem Import von Statistiken auswählen können. Der Kunden-Knopf führt in die Kunden-Verwaltung. Durch Druck auf den Admin-Knopf haben Sie Zugriff auf alle Administrator-Module, wo Sie z. B. neue Agenten, Queues usw. erstellen können. Zudem gibt es einen Suchknopf, mit dem man Suchvorgänge ausführen kann.

Wenn Zusatzmodule installiert werden, dann können diese weitere Menüpunkte zur Navigation hinzufügen.

Die rote Leiste unterhalb des Navigationsbereiches zeigt verschiedene Systemnachrichten an. Wenn Sie als OTRS-Administrator-Benutzer angemeldet sind, erhalten Sie hier den Hinweis, dass Sie mit diesem Benutzer nicht regulär arbeiten sollten.

Unter dem Haupttitel des Dashboards befinden sich verschiedene Unterbereiche, jeweils in einer eigenen Box. Diese können per Drag'n'Drop mit der Maus gezogen und losgelassen werden, um sie innerhalb ihrer Spalte neu anzuordnen. Dazu müssen Sie den Mauszeiger in den Kopfbereich der Boxen bewegen.

In der linken Spalte sehen Sie Daten über Tickets, unterteilt in: Erinnerung, eskaliert, neu und offen. In jeder Kategorie können Sie alle Tickets sehen, auf die Sie Rechte haben, wie viele Tickets Sie gesperrt haben und wie viele in "Meine Queues" liegen. "Meine Queues" sind Queues die Sie in Ihrer Benutzer-Konfiguration markieren können als Queues, die speziell überwacht werden sollen.

Auf der rechten Seite befindet sich der Bereich "Einstellungen". Klicken Sie darauf um den Bereich zu öffnen und die verschiedenen Einstellmöglichkeiten zu sehen. Sie können dann Einstellungen verändern und Ihre Änderungen speichern. Dieser Bereich ist fixiert und kann nicht mit der Maus verschoben werden.

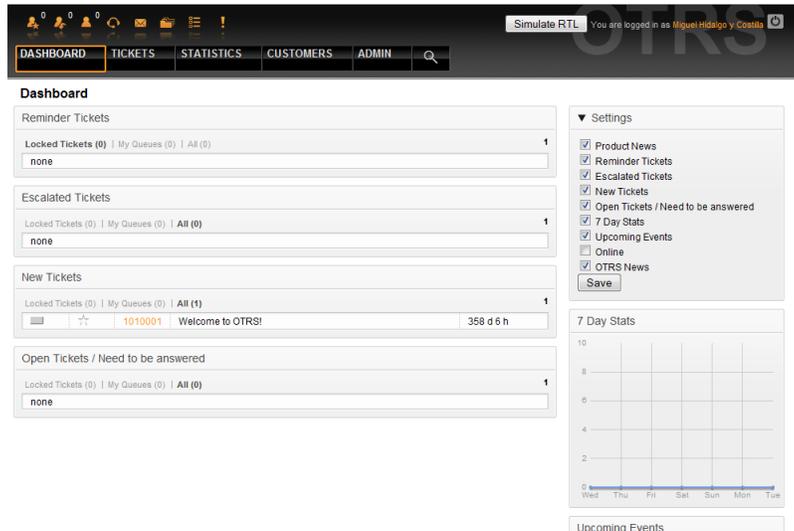


Abb. 3.5. Dashboard-Einstellungen.

Unter den Einstellungen sehen Sie die Statistik der Tickets der letzten 7 Tage. Zudem gibt es einen Abschnitt für anstehende Ereignisse und Neuigkeiten zu OTRS.

Im unteren Bildschirmbereich sehen sie die Fußzeile. Sie enthält u. A. Links auf die offizielle OTRS-Website und zum Seitenanfang.



Abb. 4.7. Fußzeile.

Was verbirgt sich hinter dem Begriff Queue?

Normalerweise werden E-Mails in einer INBOX gespeichert und verwaltet. Eine INBOX ist eine große Datei, in der alle E-Mails aneinandergereiht werden. Neue E-Mails werden einfach an das Ende der INBOX angehängt. Das E-Mail-Programm, welches Sie zum Lesen und Bearbeiten Ihrer Nachrichten benutzen, liest die INBOX-Datei aus und bereitet den Inhalt für Sie als Nutzer auf.

Eine Queue in OTRS ist ein Mechanismus, mit dessen Hilfe viele Tickets gespeichert und verwaltet werden können, also auch eine Art INBOX. Als Anwender ist es völlig unwichtig zu wissen, wo oder wie das Ticket gesichert ist. Wichtig ist nur, zu wissen, welcher Queue das Ticket zugeordnet wurde. Anwender, also die sog. Agents (z. B. die Mitarbeiter ihrer Supportabteilung), können nun Tickets zwischen den Queues verschieben! Warum aber sollten sie das tun?

Gehen wir zur praktischeren Erklärung noch mal von Max Mustermanns Unternehmen aus dem Abschnitt ein Beispiel für ein Trouble Ticket System aus. Max Mustermann hat nach seinem anfänglichen Support-Chaos OTRS installiert, und er und seine Mitarbeiter nutzen das System zur Bearbeitung der Anfragen für die Videorekorder.

Eine Queue, in die alle Anfragen einsortiert werden, reicht in dieser Situation aus. Nach einiger Zeit bringt Max Mustermann einen DVD-Player auf den Markt, der von den Kunden gut angenommen wird. Doch auch zu diesem Gerät laufen immer mehr Anfragen in das Ticket System und die Verwaltung der E-Mails mit einer Queue wird immer unübersichtlicher.

Deshalb entschließt sich Max Mustermann nach einiger Zeit, sein Supportsystem weiter zu optimieren. Er richtet zwei neue Queues ein, so dass er nun insgesamt drei Queues in OTRS definiert hat. Die

erste und schon länger vorhandene Queue wird zur Eingangsqueue, in die erst mal alle Mails wandern, umfunktioniert. Daneben gibt es jetzt noch die neuen Queues "Videorekorder" und "DVD-Player".

Herr Mustermann beauftragt Frau Müller als sog. Dispatcherin tätig zu werden und mehrmals am Tag die Mails in der Eingangsqueue zu sichten und sie, je nach Inhalt, der Queue "Videorecorder" oder der Queue "DVD-Player" zuzuordnen. Herr Meier bearbeitet ab jetzt nur noch die Anfragen in der Videorekorder-Queue, Herr Schulze geht nur noch auf die Anfragen innerhalb der DVD-Player-Queue ein. Beide haben auf die jeweils anderen zwei Queues keinen Zugriff. Herr Mustermann kümmert sich weiter wie gewohnt um alle Arten von Anfragen und darf auf alle drei Queues zugreifen.

Weil OTRS Zugriffskontrolle für Agenten, Gruppen und Rollen unterstützt, ist es denkbar einfach, Queues einzurichten, die nur von bestimmten Agenten benutzt werden können. Herr Mustermann könnte auch ein anderes Szenario umsetzen, um seine Tickets in die verschiedenen Queues zu bekommen, mit Hilfe von Filterregeln, oder falls zwei verschiedene E-Mail-Adressen verwendet werden, müsste Frau Müller nur die E-Mails in die Queues zuordnen, die nicht automatisch einsortiert werden konnten.

Das Sortieren von Mails in verschiedene Queues schafft also Ordnung und mehr Übersicht in der täglichen Mailflut, deshalb sind Queues sehr wichtig für OTRS. Durch die Einteilung der Mitarbeiter (agents) in verschiedene Benutzergruppen mit differenzierten Zugriffsrechten auf die einzelnen Queues, kann die Abarbeitung der Anfragen weiter optimiert werden. Mit Hilfe von Queues können Sie die Struktur Ihres Unternehmens abbilden bzw. einzelne Geschäftsvorgänge abgrenzen. So könnte Max Mustermann neben seinem Support-Queues für die verschiedenen Geräte für Bestellungen eine Queue mit dem Namen "Sales" anlegen und als Unter-Queues "Anfragen", "Angebote", "Bestellungen" usw. definieren, um den Bestellvorgang zu optimieren.

Je besser und strukturierter ein Support-System organisiert ist, desto weniger Zeit und letztlich auch finanzielle Mittel müssen dafür aufgebracht werden. Queues und Unter-Queues helfen bei der Strukturierung bzw. bei der Abbildung von Abläufen.

Benutzereinstellungen

Die Einstellungen eines Accounts lassen sich mit Hilfe der Benutzereinstellungen den eigenen Wünschen entsprechend anpassen. Dabei spielt keine Rolle, ob man als Mitarbeiter, Kunde oder Administrator am System angemeldet ist. Die Benutzereinstellungen sind über den Link auf den Benutzernamen erreichbar, Kunden müssen den "Einstellungen"-Link verwenden.

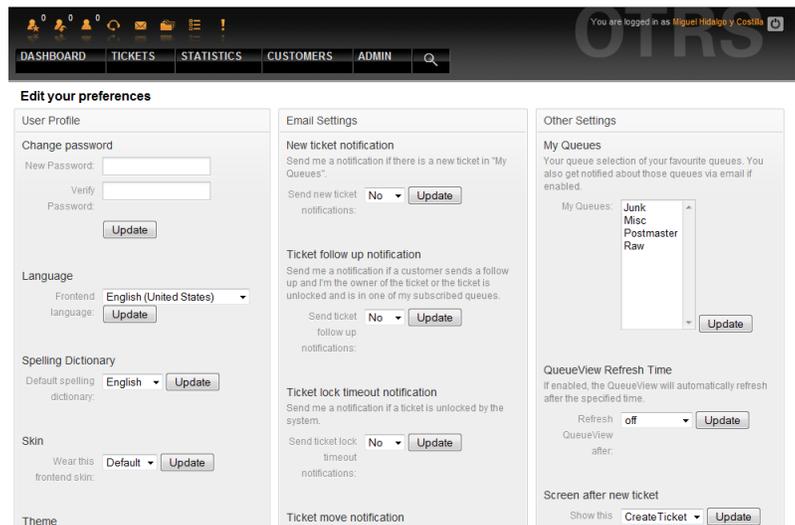


Abb. 4.8. Benutzereinstellungen des Agenten.

Ein Agent kann 3 verschiedene Kategorien von Einstellungen vornehmen: Benutzerprofil, E-Mail-Einstellungen und andere Einstellungen. Folgende Einstellungen sind möglich:

Benutzerprofil

- Aktuelles Passwort ändern.
- Sprache der Benutzeroberfläche wechseln.
- Oberflächen-Skin auswählen.
- Oberflächen-Theme auswählen.
- Abwesenheitszeit aktivieren und konfigurieren.

E-Mail-Einstellungen

- Ereignisse auswählen, bei denen eine E-Mail-Benachrichtigung durch das System erfolgen soll.

Andere Einstellungen

- Wählen Sie die Queues aus, die sie in "Meine Queues" überwachen wollen.
- Aktualisierungszeit der Queue-Ansicht einstellen.
- Bildschirm auswählen, der nach Erstellung eines Tickets angezeigt wird.

Company Personal Support Simulate RTL

New Ticket My Tickets Company Tickets Search Preferences Logout

Interface language

Your language: English (United States)

Number of displayed tickets

Max. displayed tickets: 25

Ticket overview

Refresh interval: off

Change password

New Password:

Verify Password:

S/MIME Certificate

S/MIME Certificate Upload

Powered by OTRS 3.0.x CVS [About OTRS Help](#)

Abb. 4.8. Benutzereinstellungen des Kunden.

Ein Kunde kann die Sprache der Benutzeroberfläche wechseln, die Aktualisierungszeit der Ticketübersichten verändern und die max. Anzahl angezeigter Tickets pro Seite einstellen. Zusätzlich kann das Passwort geändert werden.

Kapitel 5. Der Administrationsbereich von OTRS

Grundlagen

OTRS-Administratoren verwenden die Admin-Seite in der Weboberfläche von OTRS um das System zu konfigurieren - Agenten, Kunden und Queues hinzufügen, Ticket- und E-Mail-Einstellungen ändern und Zusatzpakete wie FAQ und ITSM installieren und vieles mehr ist damit möglich.

Agenten, die Mitglied der "admin"-Gruppe sind, können über den Link "Admin" innerhalb der Navigationsleiste der Agenten-Weboberfläche den Administrationsbereich von OTRS aufrufen.

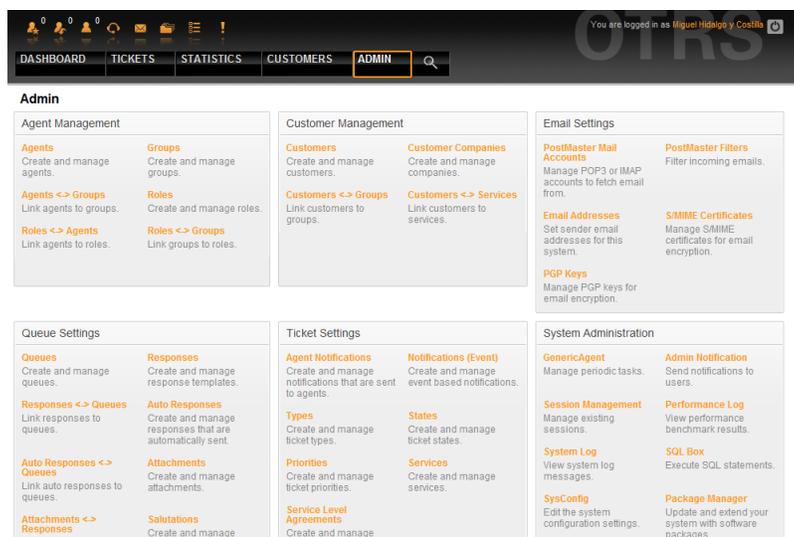


Abb. 3.1. OTRS Administrationsbereich.

Benutzer, Gruppen und Rollen

Benutzer

Über den Link "Benutzer" gelangen Sie in die Benutzerverwaltung von OTRS. Hier können Sie Benutzer anlegen, bearbeiten und deaktivieren. Weiterhin lassen sich einige grundlegende Einstellungen für den Benutzer festlegen, z. B. die Oberflächensprache oder das Anzeigeschema.

Anmerkung

Ein OTRS-Benutzer kann deaktiviert, aber nicht gelöscht werden. Um einen Benutzer zu deaktivieren, setzen Sie die Einstellung für "Gültig" auf "ungültig" oder "ungültig-temporär".

Der Administrationsbereich von OTRS

Agent Management

Actions

[Add agent](#)

Filter for Agents

Note

Agents will be needed to handle tickets.
Attention: Dont forget to add a new agent to groups and/or roles!

USERNAME	NAME	EMAIL	LAST LOGIN	VALID	CHANGED	CREATED
fcoxavier.mina	Francisco Xavier Mina	fcoxavier.mina@freedom.com		valid	08/03/2010 10:38	08/03/2010 10:38
miguel.hidalgo	Miguel Hidalgo y Costilla	miguel.hidalgo@freedom.com	08/03/2010 06:04	valid	08/03/2010 06:01	08/03/2010 06:01
root@localhost	Admin OTRS	root@localhost	08/03/2010 05:59	valid	08/03/2010 05:57	08/03/2010 05:57

Powered by OTRS 3.0.x CVS Top of page

Abb. 5.2. Benutzerverwaltung.

Um einen neuen Benutzer anzulegen, klicken Sie auf den "Benutzer hinzufügen"-Knopf, geben Sie die benötigten Daten an und klicken Sie auf Absenden.

Agent Management

Actions

[Go to overview](#)

Note

Agents will be needed to handle tickets.
Attention: Dont forget to add a new agent to groups and/or roles!

Add Agent

Title:

Firstname:

Lastname:

Username:

Password:

Email:

Valid:

Language:

Spelling Dictionary:

Skin:

Theme:

Out Of Office Time: On Off

Start: 08 / 03 / 2010

Abb. 5.3. Hinzufügen eines Benutzers.

Nachdem Sie einen neuen Benutzer angelegt haben, muss dieser einer Gruppe bzw. einer Rolle zugewiesen werden. Sie werden nach dem Anlegen eines neuen Benutzers automatisch auf die Bildschirmmaske für die Zuweisung eines Benutzers in Gruppen weitergeleitet. Weitere Informationen über Gruppen und Rollen finden Sie in den Abschnitten Gruppen und Rollen dieses Kapitels.

Gruppen

Jeder Mitarbeiter mit einem Account im OTRS, sollte mindestens einer Benutzergruppe angehören. In einer Neuinstallation sind drei vordefinierte Gruppen verfügbar.

Tabelle 5.1. Standardmäßig vorhandene Gruppen in OTRS

Gruppe	Beschreibung
admin	Gruppe für die Benutzer mit Administrationsrechten.
stats	Benutzer in dieser Gruppe dürfen lesend (ro) oder schreibend (rw) auf das Statistikmodul von OTRS zugreifen, d.h. sie können Statistiken einsehen oder auch erstellen.
users	Dies ist die Gruppe, in die normale Mitarbeiter aufgenommen und mit den kompletten Rechten ausgestattet werden sollten. Dadurch wird für die Mitarbeiter das normale Arbeiten im System ermöglicht, der Zugriff auf alle Funktionen rund um Tickets ist gegeben.

Anmerkung

In einer neuen Installation ist die Gruppe "users" leer und der Benutzer "root@localhost" ist Mitglied der Gruppen "admin" und "stats".

Sie können durch Klick auf "Gruppen" im Administrationsbereich in die Gruppenverwaltung gelangen.

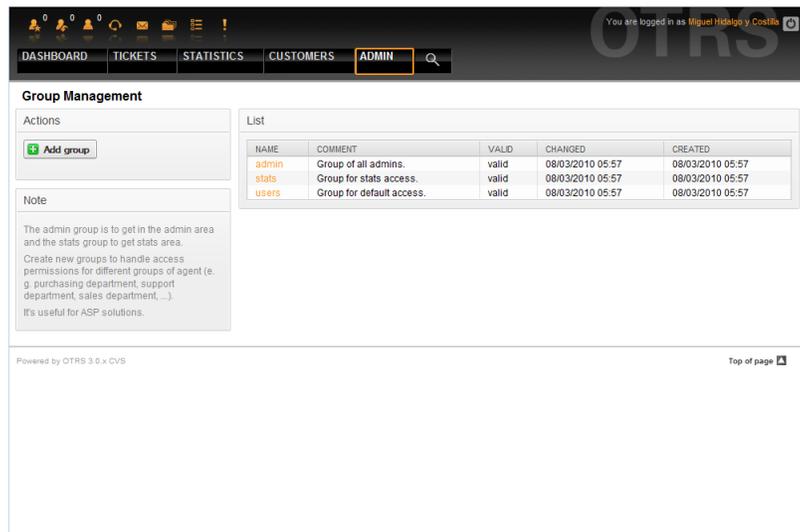


Abb. 5.4. Gruppenverwaltung.

Anmerkung

In OTRS können Gruppen deaktiviert, aber nicht gelöscht werden. Deaktivieren Sie eine Gruppe, indem Sie für "Gültig" den Wert entweder auf "ungültig" oder auf "ungültig-temporär" setzen.

Um einen Benutzer einer Gruppe zuzuweisen bzw. die Gruppenmitgliedschaft eines Benutzers zu ändern, kann der Link "Benutzer <-> Gruppen" im Administrationsbereich genutzt werden.

Der Administrationsbereich von OTRS

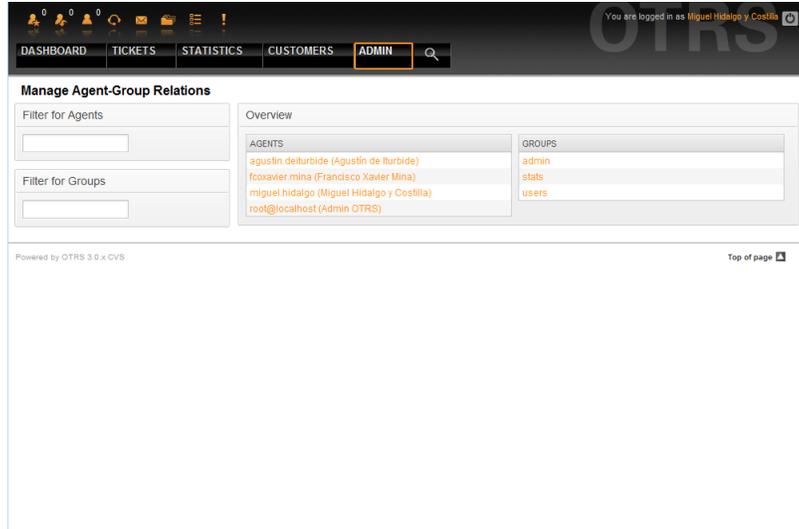


Abb. 5.5. Benutzer-Gruppen-Zuordnung.

Im unteren Bereich des Bildschirms wird eine Übersicht angezeigt, die alle Benutzer und Gruppen auflistet. Indem Sie auf einen Benutzernamen klicken, bekommen Sie dessen Gruppenzugehörigkeiten angezeigt und können diese ändern. Bei der Auswahl einer Gruppe werden alle Benutzer aufgelistet, die sich in dieser Gruppe befinden.

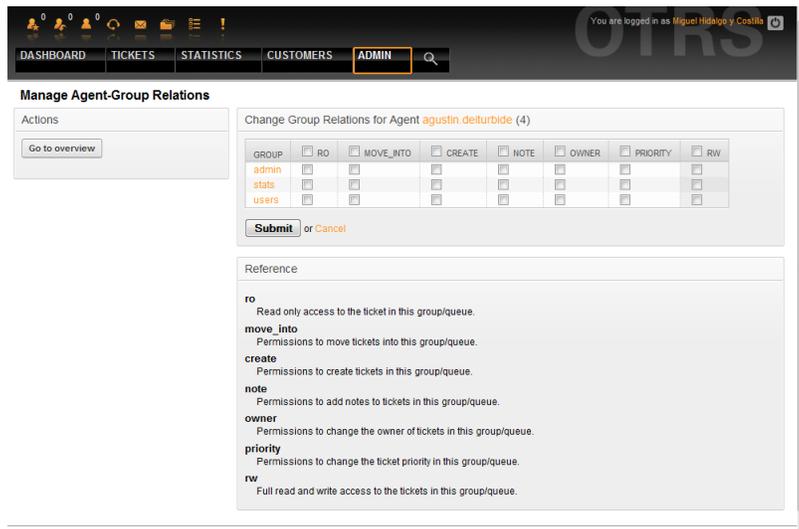


Abb. 5.6. Bearbeiten der Gruppenzugehörigkeiten eines Benutzers.

Der Administrationsbereich von OTRS

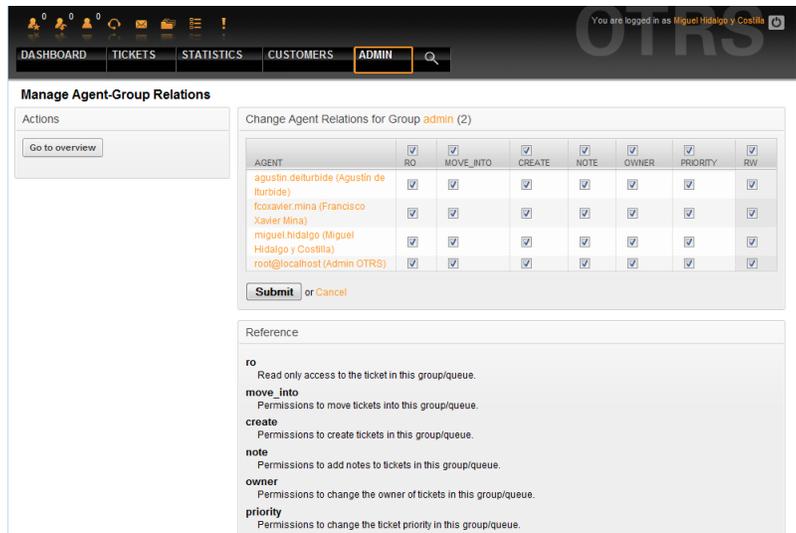


Abb. 5.7. Bearbeiten der Benutzerzuordnungen einer Gruppe.

Jede Gruppe hat zugehörige Berechtigungen, und jeder Agent, der Mitglied dieser Gruppe ist, kann eine Kombination dieser Rechte erhalten. Eine Liste dieser Berechtigungen wird in Tabelle 5-2 gezeigt.

Tabelle 5.2. Mit OTRS-Gruppen verknüpfte Rechte

Berechtigung	Beschreibung
ro	Nur Lesezugriff auf die Tickets bzw. Beiträge dieser Gruppe bzw. der Gruppe zugewiesenen Queues oder Bereiche.
Verschieben in (move into)	Recht zum Verschieben von Tickets oder Beiträgen innerhalb der Queues bzw. Bereiche dieser Gruppe.
erstellen	Recht zum Erstellen von Tickets oder Beiträgen in den Queues, bzw. Bereichen dieser Gruppe.
Besitzer (owner)	Recht zum Ändern des Eigentümers von Tickets, bzw. Beiträgen in den der Gruppe zugewiesenen Queues bzw. Bereiche.
Priorität	Recht zum Ändern der Priorität von Tickets, bzw. Beiträgen in den der Gruppe zugewiesenen Queues bzw. Bereiche.
rw	Voller Lese- und Schreibzugriff auf alle Inhalte der dieser Gruppe zugewiesenen Queues, bzw. Bereiche.

Anmerkung

Standardmäßig listet die Queue-Ansicht nur Tickets in Queues auf, auf welche der Benutzer *rw*-Rechte hat, d. h. Tickets, an denen der Benutzer arbeiten muss. Wenn Sie dieses Verhalten ändern wollen, können Sie `Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets` auf *Ja* setzen.

Rollen

Rollen sind ein sehr nützliches und mächtiges Feature in OTRS, um schnell und einfach die Vergabe von Zugriffsrechten für viele Benutzer vorzunehmen. Vor allem bei großen und komplexen Installationen mit vielen Benutzern, Gruppen und Queues, zahlt sich dieses Feature schnell aus und erspart dem OTRS-Administrator viel Zeit und Arbeit. Das nachfolgende Beispiel stellt ein mögliches Szenario vor.

Um den Nutzen von Rollen zu verdeutlichen, stellen Sie sich die Situation vor, dass Sie ein OTRS-System mit 100 Benutzern verwalten. 90 Benutzer haben Zugriff auf eine Queue namens Support, die mehrere themenspezifische Unter-Queues enthält und in der die Support-Anfragen Ihrer Kunden landen. Die restlichen Queues des Systems sind für die 90 Supporter nicht zugänglich, dies wurde durch Gruppenzugriffsrechte so festgelegt. Die übrigen 10 Benutzer haben Zugriff auf alle Queues im System. Sie sortieren falsch einsortierte Mails aus, behalten die "Raw"-Queue im Auge und verschieben Spam-Mails in die "Junk"-Queue.

Im Rahmen einer Unternehmensumstrukturierung wird eines Tages zusätzlich eine Abteilung eröffnet, die Produkte verkaufen soll. Es müssen Angebote, Auftragsbestätigungen und Rechnungen erstellt, Anfragen bearbeitet, Bestellungen ans Lager weitergeleitet und Stornierungen entgegen genommen werden. Ein Teil der bisherigen Mitarbeiter soll in verschiedenen Bereichen der neuen Abteilung tätig werden und Sie als OTRS-Administrator haben nun die Aufgabe die neuen Queues anzulegen, die erweiterten Zugriffsrechte anzupassen und diese für die einzelnen Benutzer zu ändern.

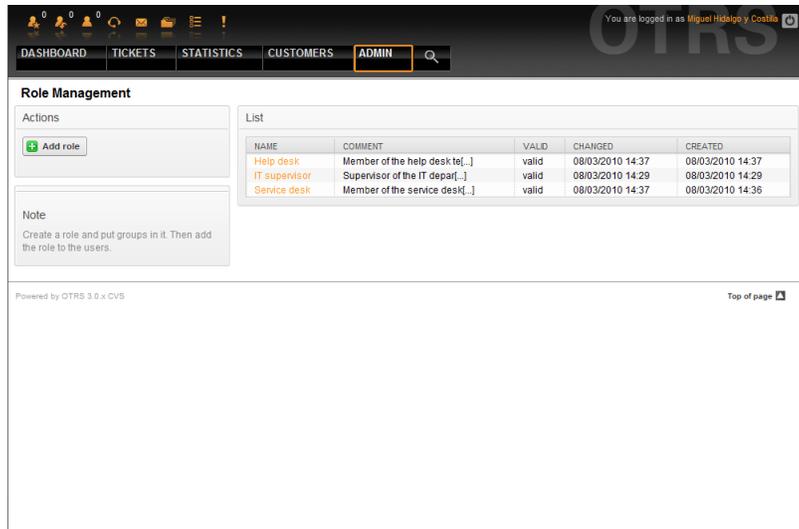
Da es mühsam und viel zu umständlich wäre, für einen Teil aller 100 Benutzer einzeln die Zugriffsrechte zu ändern, richten Sie Rollen ein die mit Hilfe von Gruppenberechtigungen die verschiedenen Zugriffsrechte regeln. Anschließend ändern Sie für die entsprechenden Benutzer auf einmal die Zugriffsberechtigungen, indem Sie diese der entsprechenden Rolle zuweisen. Beim Anlegen neuer Benutzer müssen Sie nicht mehr einzeln die Gruppen und Zugriffsrechte einstellen, auch hier genügt die Verknüpfung des neuen Benutzers mit einer Rolle.

Anmerkung

Rollen sind wirklich hilfreich für die Verwaltung großer OTRS-Installationen. Allerdings sollten Sie nicht Gruppen und Rollen gleichzeitig einsetzen, um die Wartung nicht zu kompliziert zu machen. Sollten Sie sich also entscheiden, auf Rollen zu setzen, empfehlen wir Ihnen, die Verwaltung für die Benutzer-Gruppen-Zuordnungen abzuschalten, indem Sie `Frontend::Module###AdminUserGroup` in der `SysConfig` deaktivieren. Dadurch werden existierende Gruppenzuordnungen natürlich nicht entfernt!

Sie können die Rollenverwaltung durch Klick auf "Rollen" im Administrationsbereich erreichen.

Der Administrationsbereich von OTRS



The screenshot shows the OTRS Role Management interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN. The ADMIN tab is selected. Below the navigation bar, the page title is "Role Management". On the left, there is an "Actions" section with a green "Add role" button and a "Note" section with instructions: "Create a role and put groups in it. Then add the role to the users." On the right, there is a "List" table with the following data:

NAME	COMMENT	VALID	CHANGED	CREATED
Help desk	Member of the help desk te[...]	valid	08/03/2010 14:37	08/03/2010 14:37
IT supervisor	Supervisor of the IT depart[...]	valid	08/03/2010 14:29	08/03/2010 14:29
Service desk	Member of the service desk[...]	valid	08/03/2010 14:37	08/03/2010 14:36

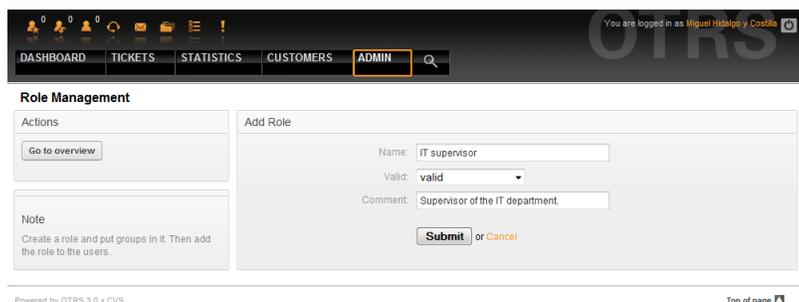
At the bottom of the page, it says "Powered by OTRS 3.0.x CVS" and "Top of page".

Abb. 5.8. Rollenverwaltung.

Anmerkung

In OTRS können Rollen deaktiviert, aber nicht gelöscht werden. Deaktivieren Sie eine Rolle, indem Sie für "Gültig" den Wert entweder auf "ungültig" bzw. "ungültig-temporär" setzen.

Eine Übersicht aller Rollen im System erscheint, um einen Eintrag zu bearbeiten, klicken Sie einfach auf den Rollennamen. In einer Neuinstallation gibt es keine vordefinierten Rollen. Um eine anzulegen, klicken Sie auf den "Rolle hinzufügen"-Knopf und geben Sie die erforderlichen Daten an.



The screenshot shows the OTRS Role Management interface with the "Add Role" form open. The form has the following fields:

- Name:
- Valid:
- Comment:

At the bottom of the form, there are "Submit" and "Cancel" buttons. On the left, there is an "Actions" section with a "Go to overview" button and a "Note" section with instructions: "Create a role and put groups in it. Then add the role to the users." At the bottom of the page, it says "Powered by OTRS 3.0.x CVS" and "Top of page".

Abb. 5.9. Anlegen einer neuen Rolle.

Um einen Überblick über alle Rollen und Agenten im System zu erhalten, klicken Sie auf den Link "Rollen <-> Benutzer" im Administrationsbereich. Sie können die Filter verwenden, um einen bestimmten Eintrag zu finden. Wenn Sie die Rollenzordnungen eines Benutzers ändern wollen, klicken Sie auf den Benutzernamen. Um die Benutzerzuordnungen einer Rolle zu verändern, klicken Sie auf die Rolle.

Der Administrationsbereich von OTRS

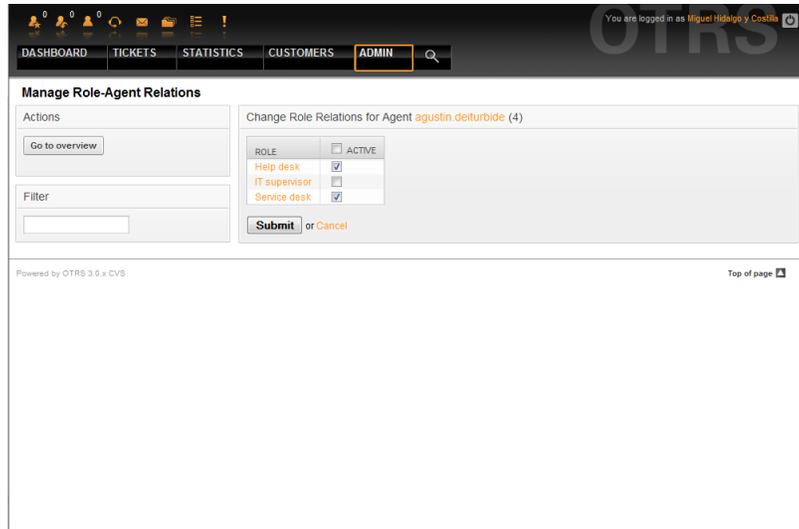


Abb. 5.10. Bearbeiten der Rollenzuordnungen eines Agenten.

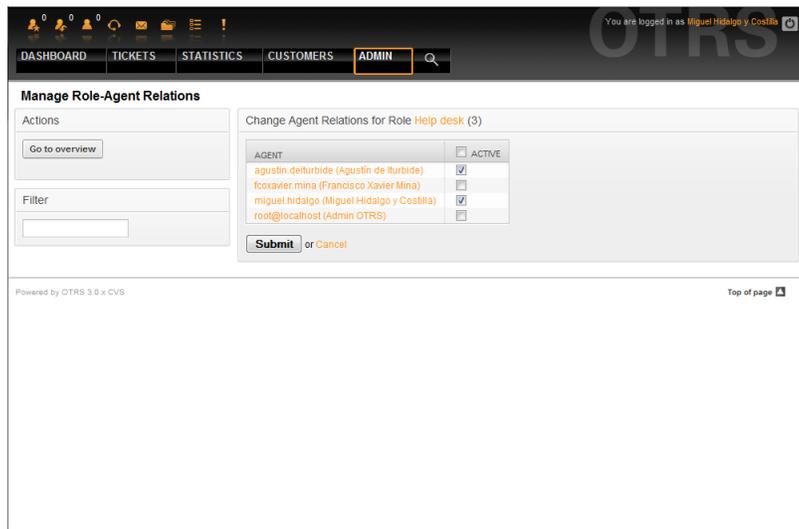


Abb. 5.11. Bearbeiten der Benutzerzuordnungen einer Rolle.

Um eine Übersicht aller Rollen und Gruppen im System zu erhalten, verwenden Sie den Link "Rollen <-> Gruppen" im Administrationsbereich.

Der Administrationsbereich von OTRS

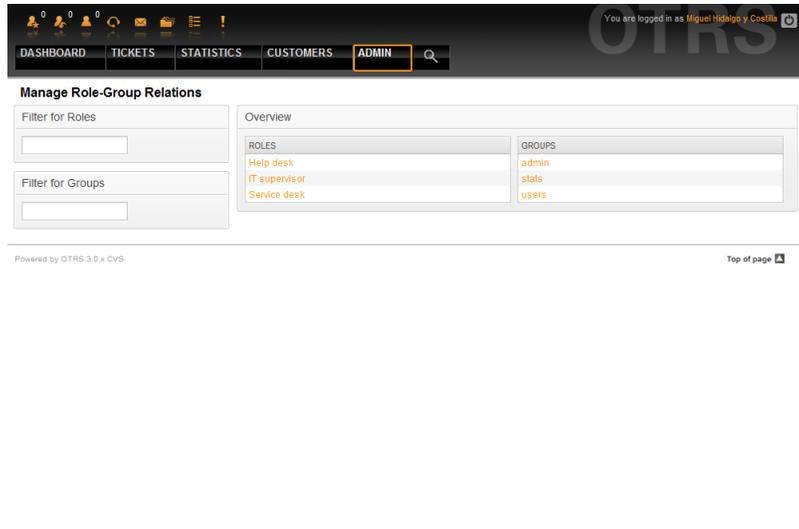


Abb. 5.12. Rollen-Gruppen-Zuordnungen verwalten.

Klicken Sie auf eine Gruppe oder Role, um die jeweiligen Zuordnungen zu verändern.

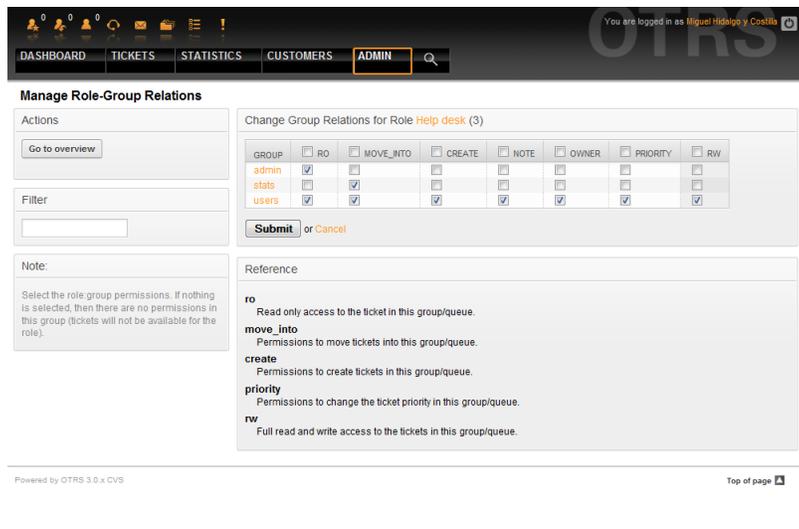


Abb. 5.13. Gruppenzuordnungen einer Rolle verändern.

Der Administrationsbereich von OTRS

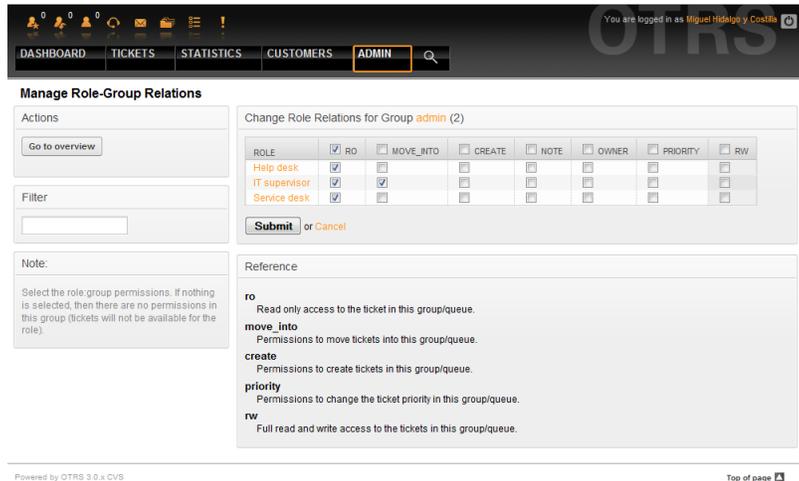


Abb. 5.14. Rollenzuordnungen für eine Gruppe verändern.

Kundenbenutzer und Kundengruppen

Kunden

OTRS unterstützt, wie bereits erwähnt, verschiedene Arten von Benutzern. Über den Link "Kunden", den Sie im Admin-Bereich von OTRS finden, können Sie die Benutzerdaten der im System angelegten Kunden verwalten. Ein Kunde kann sich mit Hilfe seines Accounts in das vom Ticket-System bereitgestellte Webinterface für Kunden einloggen, um dort die eigenen Tickets einzusehen, neue Tickets zu verfassen, usw. Weiterhin wird ein Kunden-Account vom System für die Historie von Tickets benötigt.

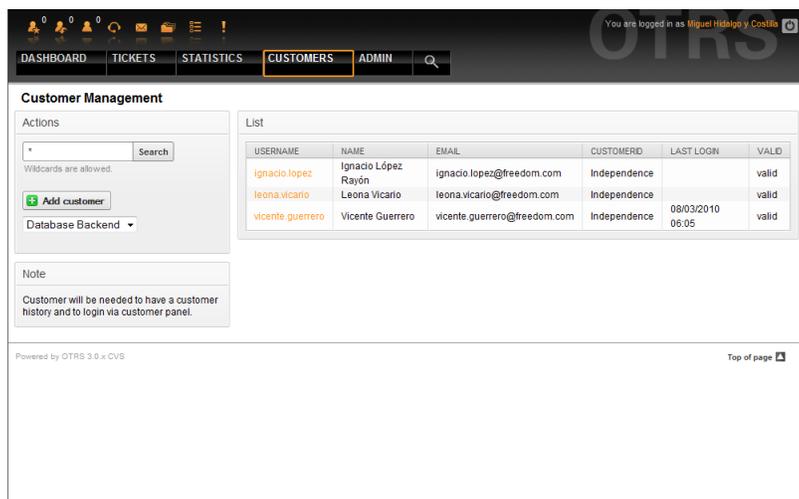
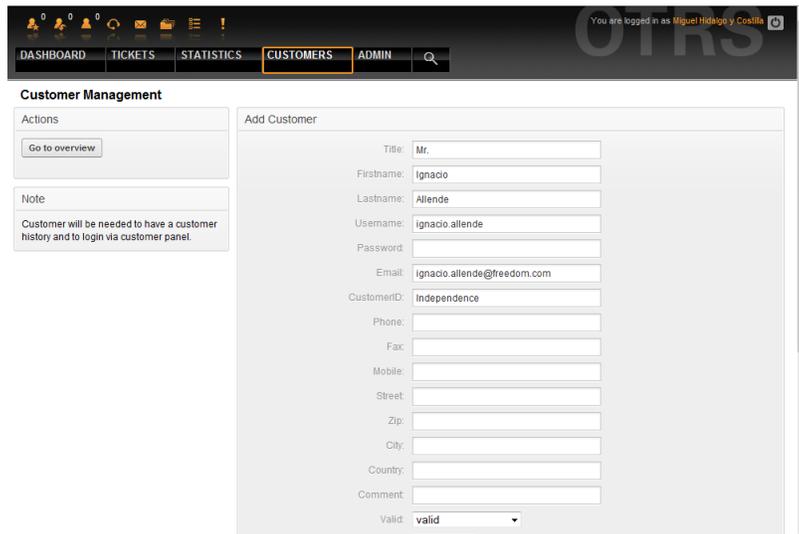


Abb. 5.14. Kundenverwaltung.

Neben der Möglichkeit in der Datenbank nach einem bestimmten Kunden zu suchen, kann das Backend umgestellt werden, über das auf die Kundendaten zugegriffen wird. In OTRS lassen sich mehrere Datenbanken mit Kundendaten einbinden, genauere Informationen hierzu finden Sie im Abschnitt Einbinden externer Backends für Agents und Customer.

Um einen neuen Kunden hinzuzufügen, klicken Sie auf "Kunde hinzufügen". Einige Felder sind Pflichtfelder, müssen also ausgefüllt werden.



The screenshot shows the OTRS web interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS (which is highlighted), and ADMIN. Below the navigation bar, the page title is "Customer Management". On the left side, there is a sidebar with "Actions" (containing a "Go to overview" button) and a "Note" section with the text: "Customer will be needed to have a customer history and to login via customer panel." The main content area is titled "Add Customer" and contains a form with the following fields: Title (Mr.), Firstname (Ignacio), Lastname (Allende), Username (ignacio.allende), Password, Email (ignacio.allende@freedom.com), CustomerID (Independence), Phone, Fax, Mobile, Street, Zip, City, Country, Comment, and a "Valid" dropdown menu set to "valid".

Abb. 5.14. Hinzufügen eines Kunden.

Der Kunde kann mit Benutzername und Passwort auf das System zugreifen. Die Kundennummer wird vom System benötigt, um den Kunden und seine Tickets zu erkennen. Da E-Mail-Adressen eindeutig sind, können sie als ID verwendet werden.

Anmerkung

In OTRS können Kunden deaktiviert, aber nicht gelöscht werden. Sie können einen Kunden deaktivieren, indem Sie den Wert für "Gültig" auf "ungültig" oder "ungültig-temporär" setzen.

Kundengruppen

Kunden können auch Gruppen zugeordnet werden. Das ist nützlich, wenn Kunden nur auf bestimmte Queues zugreifen können sollen. Legen Sie zuerst die benötigten Gruppen in der Gruppenverwaltung an. Fügen Sie dann die Queues hinzu und ordnen Sie diese der neuen Gruppe zu.

Im nächsten Schritt aktivieren Sie mit Hilfe des Konfigurationsparameters CustomerGroupSupport die Unterstützung für Kundengruppen. Mit Hilfe des Parameters CustomerGroupAlwaysGroups legen Sie fest, welchen Gruppen ein neu angelegter Kundenbenutzer automatisch zugeordnet werden soll.

Über den Link "Kunden <-> Gruppen" können Sie nun die Zuordnung der Kundenbenutzer in die gewünschten Gruppen vornehmen.

Der Administrationsbereich von OTRS

Powered by OTRS 3.0.x CVS Top of page

Abb. 5.15. Verwaltung der Kunden-Gruppen-Zuordnungen.

Klicken Sie auf den Kunden- oder Gruppennamen, um die jeweiligen Zuordnungen zu bearbeiten.

Powered by OTRS 3.0.x CVS Top of page

Abb. 5.16. Gruppenzuordnungen eines Kunden bearbeiten.

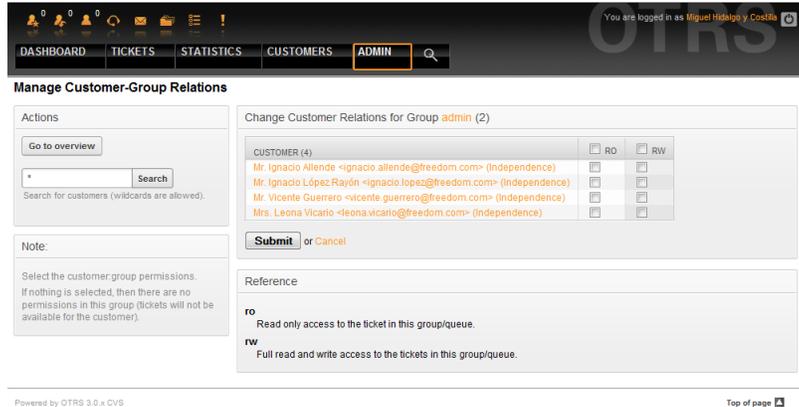


Abb. 5.17. Kundenzuordnungen einer Gruppe bearbeiten.

Queues

Über den Link "Queue" innerhalb des Admin-Bereiches von OTRS können Sie die Queues Ihres Systems verwalten. Nach einer Standard-Installation sind bereits die Queues "Junk", "Misc", "Postmaster" und "Raw" im System angelegt. "Raw" ist die Default-Queue, in ihr landen alle neuen Tickets, so lange kein Filter definiert wurde. "Junk" kann z. B. zum Aussortieren von Spam-Mails genutzt werden.

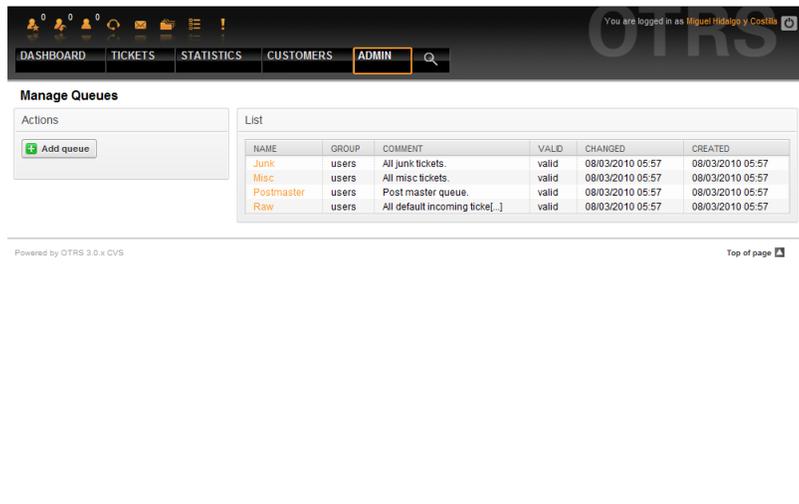


Abb. 5.18. Queue-Verwaltung.

Hier können Sie Queues anlegen und verändern. Zusätzlich zum Namen der neuen Queue kann angegeben werden, für welche Benutzergruppe die Queue bereitgestellt werden und ob die neue Queue eine Unter-Queue (sub queue) von einer bereits in Ihrem System vorhandenen Queue sein soll.

The screenshot shows the 'Add Queue' form in the OTRS Admin interface. The form is titled 'Add Queue' and is located under the 'Manage Queues' section. The 'Name' field is set to 'Support'. The 'Group' is set to 'admin'. The 'Unlock timeout minutes' is set to 1220. There are three escalation time settings, each with a 'Notify by' dropdown and explanatory text below it:

- Escalation - first response time (minutes): 60 (Notify by: -)
- Escalation - update time (minutes): 180 (Notify by: -)
- Escalation - solution time (minutes): 300 (Notify by: -)

Abb. 5.19. Hinzufügen einer Queue.

Wurde ein Ticket von einem Agenten gesperrt, so können Sie mit Hilfe des Freigabezeit-Intervalls festlegen, wann ein Ticket wieder automatisch vom System freigegeben werden soll. So können auch die anderen Mitarbeiter wieder auf dieses Ticket zugreifen und es bearbeiten.

Drei Arten von Eskalationszeiten können pro Queue eingestellt werden:

Eskalation - Zeit für die erste Reaktion

- Nach der Erstellung wird dieses Ticket eskalieren, wenn die hier definierte Zeit verstreicht, ohne dass eine Kommunikation zum Kunden (per E-Mail oder Telefonanruf) stattgefunden hat.

Eskalation - Aktualisierungszeit

- Wenn ein Artikel vom Kunden hinzugefügt wird, als E-Mail-Antwort oder über die Kundenoberfläche, wird die Aktualisierungszeit zurückgesetzt. Wenn anschließend für die definierte Zeitdauer keine Rückmeldung an den Kunden erfolgt, eskaliert das Ticket.

Eskalation - Lösungszeit

- Wenn das Ticket nicht innerhalb der definierten Zeitdauer geschlossen werden kann, eskaliert es.

Weiterhin können Sie festlegen, dass bei einem Follow-Up auf ein Ticket wieder der Mitarbeiter Eigentümer dieses Tickets wird, der zuletzt als Eigentümer im System für dieses Ticket vermerkt war. Dies stellt sicher, dass die Nachfrage eines Kunden zuerst bei demselben Mitarbeiter landet, der sich zuletzt um dieses Ticket gekümmert hat.

Der Parameter für die Systemadresse legt fest, mit welcher Absenderadresse Mails aus dieser Queue versendet werden sollen. Mit Hilfe der Parameter für Anrede und Signatur kann eingestellt werden, welche Vorgaben hier standardmäßig bei Antworten auf Tickets in dieser Queue genutzt werden. In den Abschnitten E-Mail-Adressen, Anreden und Signaturen erfahren Sie mehr über die Einrichtung dieser Parameter.

Anmerkung

In OTRS können Queues deaktiviert, aber nicht gelöscht werden. Sie können eine Queue deaktivieren, indem Sie den Wert für "Gültig" auf "ungültig" oder "ungültig-temporär" setzen.

Anreden, Signaturen, Anlagen und Antwortvorlagen

Anreden

Ein Textbaustein für eine Antwortvorlage ist die Anrede. Anreden können einer Queue zugeordnet werden, wie im Abschnitt zu den Queues beschrieben, nur so werden Sie bei Antworten für Tickets aus dieser Queue verwendet. Über den Link "Anreden" innerhalb des Admin-Bereiches, können Sie die verschiedenen Anreden Ihres Systems verwalten.

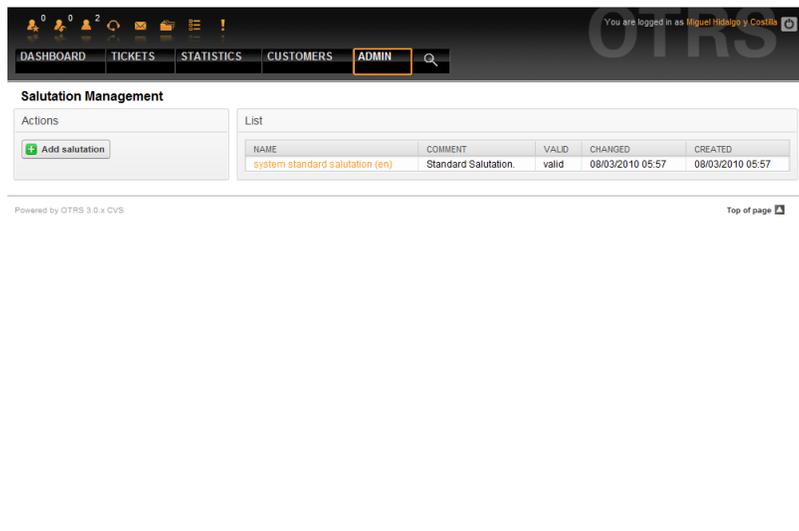


Abb. 5.20. Verwaltung der Anreden.

Nach einer Standardinstallation von OTRS sind bereits eine Anrede im System gespeichert, "system standard salutation (en)".

Um eine neue Anrede anzulegen, verwenden Sie den Knopf "Anrede hinzufügen" und geben Sie die erforderlichen Daten an.

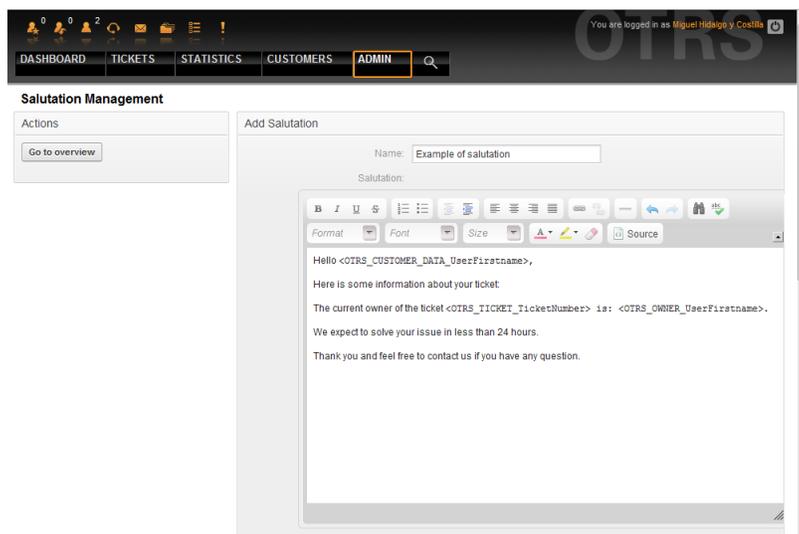


Abb. 5.21. Anrede hinzufügen.

In Anreden können Variablen verwendet werden. Wenn Sie auf ein Ticket antworten, werden die Variablennamen durch die zugehörigen Werte ersetzt.

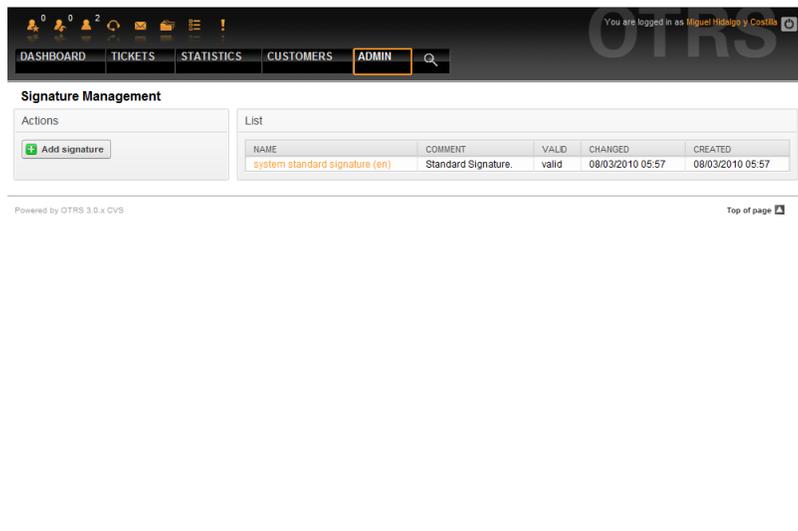
Im unteren Teil der Bildschirmmaske zum Ändern oder Anlegen einer Anrede finden Sie die verschiedenen OTRS-Variablen, die Sie für die dynamische Aufbereitung eines Anredentextes verwenden können. Bauen Sie z. B. in Ihrem Anredentext die Variable <OTRS_LAST_NAME> ein, so wird diese später durch den Nachnamen der Person ersetzt, deren Ticket Sie beantworten.

Anmerkung

In OTRS können Anreden deaktiviert, aber nicht gelöscht werden. Sie können eine Anrede deaktivieren, indem Sie den Wert für "Gültig" auf "ungültig" oder "ungültig-temporär" setzen.

Signaturen

Ein weiterer Textbaustein für eine Antwortvorlage, ist die Signatur. Signaturen können einer Queue zugeordnet werden, wie im Abschnitt zu den Queues beschrieben, nur so werden Sie bei Antworten für Tickets aus dieser Queue verwendet. Über den Link "Signaturen" innerhalb des Admin-Bereiches, können Sie die verschiedenen Signaturen Ihres Systems verwalten.



The screenshot shows the OTRS Administration interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN (which is highlighted). Below the navigation bar, the 'Signature Management' section is visible. It contains an 'Actions' panel with a green 'Add signature' button. To the right of the actions panel is a table with the following data:

NAME	COMMENT	VALID	CHANGED	CREATED
system standard signature (en)	Standard Signature.	valid	08/03/2010 05:57	08/03/2010 05:57

At the bottom of the page, there is a footer that reads 'Powered by OTRS 3.0.x CVS' and a 'Top of page' link.

Abb. 5.22. Signaturverwaltung.

Nach einer Standardinstallation von OTRS ist bereits eine Signatur im System vorhanden, "system standard signature (en)".

Verwenden Sie den Knopf "Signatur hinzufügen", um eine neue Signatur anzulegen.

Der Administrationsbereich von OTRS

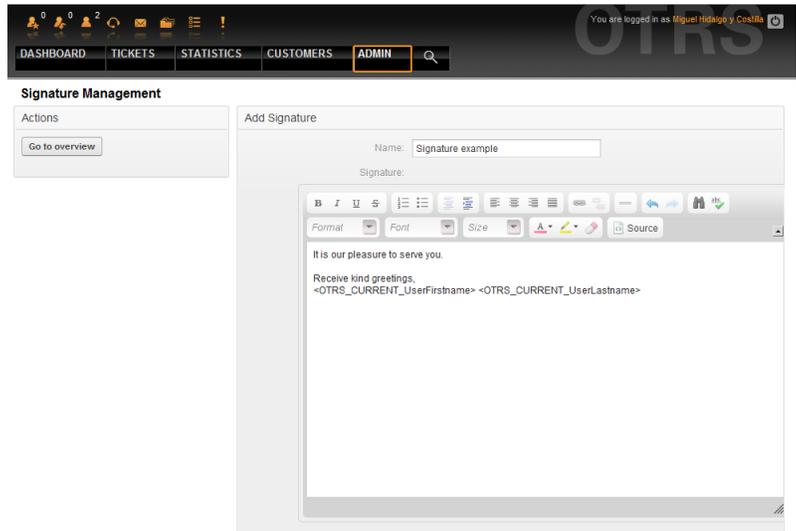


Abb. 5.23. Signatur hinzufügen.

Auch in Signaturen können dynamische Inhalte eingebaut werden, wie etwa Vor- und Nachname des Agenten. Dies geschieht, genauso wie bei den Anreden, mit Hilfe verschiedener OTRS-Variablen, die in den Text der Signatur integriert werden können. Ein variabler Textteil könnte z. B. der Name des Mitarbeiters sein, der das Ticket beantwortet hat (<OTRS_LAST_NAME>).

Anmerkung

In OTRS können Signaturen deaktiviert, aber nicht gelöscht werden. Sie können eine Signatur deaktivieren, indem Sie den Wert für "Gültig" auf "ungültig" oder "ungültig-temporär" setzen.

Anlagen

Ein weitere optionaler Teil einer Antwortvorlage kann eine Anlage sein. Diese wird bei der Benutzung einer Antwortvorlage als Anhang an das zu beantwortende Ticket angehängt, kann aber über eine Checkbox im Antworten-Bildschirm für Tickets leicht deaktiviert werden.

Über den "Anlagen"-Link im Admin-Bereich können Sie neue Anlagen in das System integrieren.

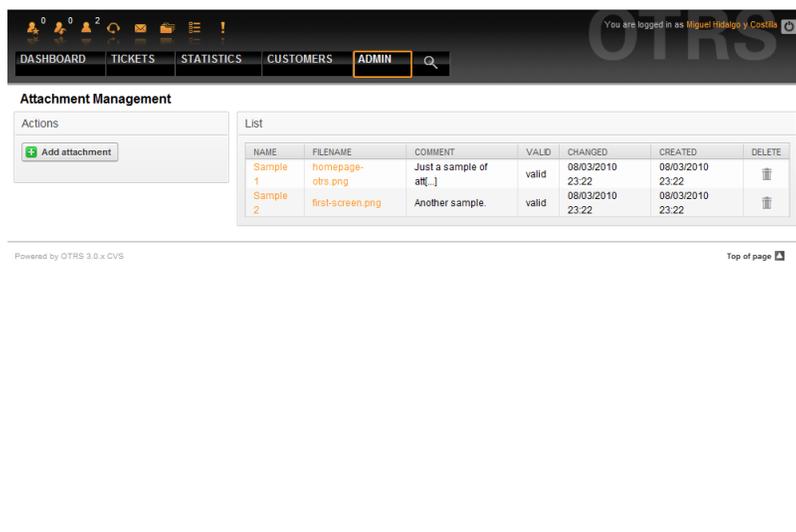


Abb. 5.24. Verwaltung der Anlagen.

Verwenden Sie den "Anlage hinzufügen"-Knopf, um eine neue Anlage zu erstellen.

The screenshot shows the OTRS Administration interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN. The ADMIN tab is selected. Below the navigation bar, the 'Attachment Management' section is displayed. On the left, there is an 'Actions' box with a 'Go to overview' button. The main area is titled 'Add Attachment' and contains a form with the following fields: 'Name' (Sample 3), 'Attachment' (C:\Intel\Logs\IntelGFX.log) with a 'Browse...' button, 'Valid' (valid), and 'Comment' (Just another sample). At the bottom of the form are 'Submit' and 'Cancel' buttons. The footer of the page indicates 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

Abb. 5.25. Anlage hinzufügen.

Wenn eine Anlage erstellt wurde, kann sie einer oder mehreren Antworten zugeordnet werden. Folgen Sie dazu dem Link "Anlagen <-> Antworten" im Administrationsbereich.

The screenshot shows the OTRS Administration interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN. The ADMIN tab is selected. Below the navigation bar, the 'Manage Responses <-> Attachments Relations' section is displayed. The 'Overview' table is shown with two columns: 'RESPONSES' and 'ATTACHMENTS'. The 'RESPONSES' column contains three entries: 'empty answer', 'More details needed', and 'test answer'. The 'ATTACHMENTS' column contains three entries: 'Sample 1 (homepage-otrs.png)', 'Sample 2 (first-screen.png)', and 'Sample 3 (IntelGFX.log)'. The footer of the page indicates 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

RESPONSES	ATTACHMENTS
empty answer	Sample 1 (homepage-otrs.png)
More details needed	Sample 2 (first-screen.png)
test answer	Sample 3 (IntelGFX.log)

Abb. 5.26. Zuordnung von Anlagen zu Antworten.

Klicken Sie auf den Namen einer Anlage oder einer Antwortvorlage, um die jeweiligen Zuordnungen zu verändern.

Der Administrationsbereich von OTRS

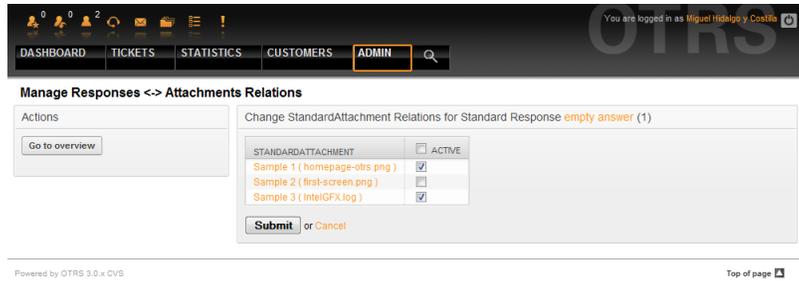


Abb. 5.27. Anlagenzuordnungen einer Antwortvorlage verändern.

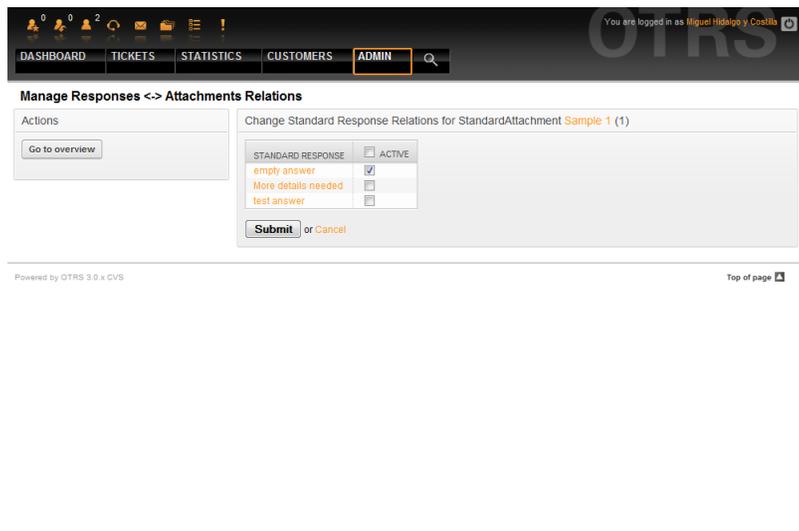


Abb. 5.28. Antwortvorlagen-Zuordnungen einer Anlage verändern.

Antwortvorlagen

Um das Antworten auf Tickets zu beschleunigen und das Aussehen beantworteter Tickets zu vereinheitlichen, können in OTRS Antwortvorlagen erstellt werden. Jede Antwortvorlage kann einer oder mehreren Queues bzw. Unter-Queues zugeordnet werden, es können mehrere Antwortvorlagen für jede Queue definiert werden.

Nach einer Standardinstallation ist für jede Queue die Antwortvorlage "empty answer" vorhanden. Im Admin-Bereich kann über den Link "Antworten" auf die im System gespeicherten Antwortvorlagen zugegriffen und diese bearbeitet werden.

Der Administrationsbereich von OTRS

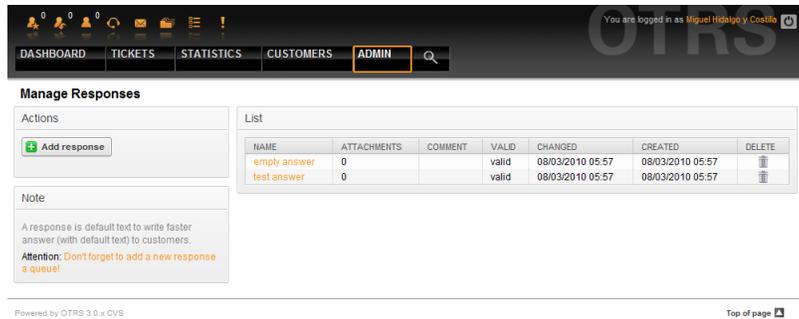


Abb. 5.29. Verwaltung der Antwortvorlagen.

Verwenden Sie den "Antwort hinzufügen"-Knopf zur Erstellung einer neuen Antwortvorlage.

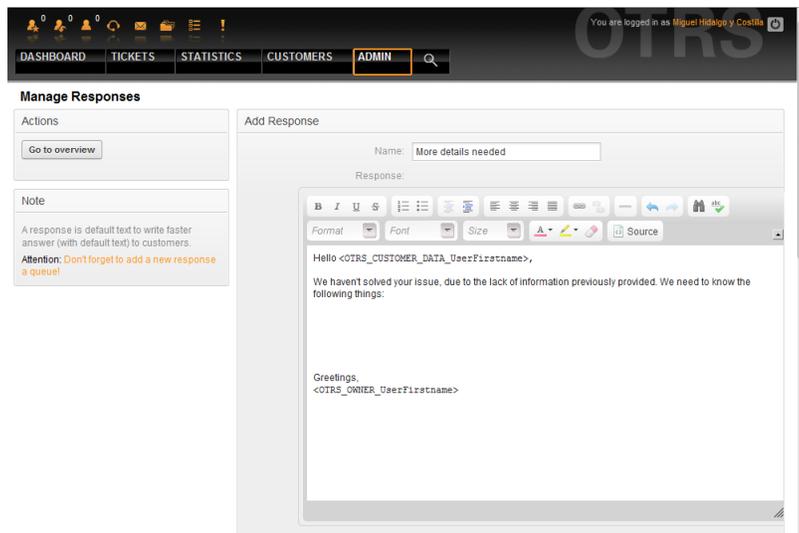


Abb. 5.30. Antwortvorlage hinzufügen.

Um eine Antwortvorlage einer oder mehreren Queues zuzuweisen, aktivieren Sie im Admin-Bereich den Link "Antworten <-> Queues". Sie können die Filter verwenden, um bestimmte Elemente zu finden.

Der Administrationsbereich von OTRS

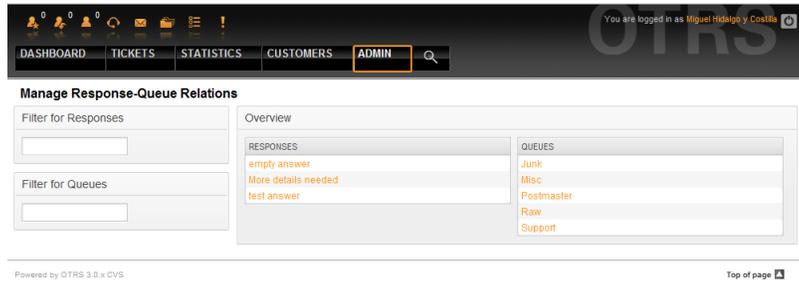


Abb. 5.31. Queue-Antwort-Zuordnungen verwalten.

Klicken Sie auf den Namen einer Queue oder Antwortvorlage, um die jeweiligen Zuordnungen zu bearbeiten.

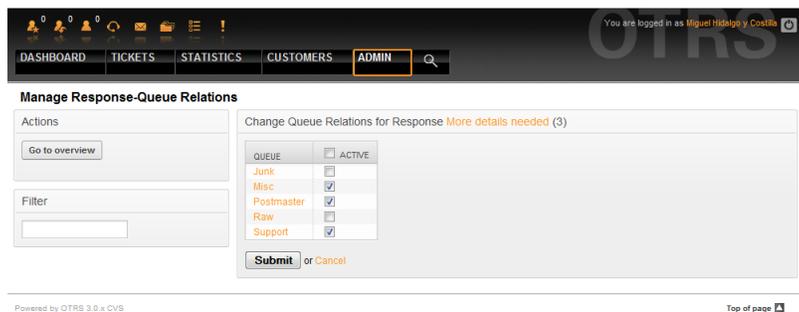


Abb. 5.32. Queue-Zuordnungen einer Antwortvorlage ändern.

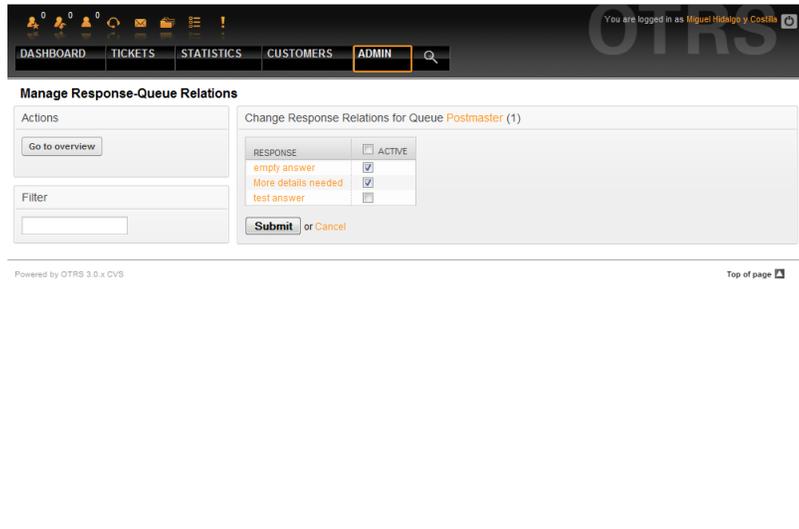


Abb. 5.33. Antwort-Zuordnungen einer Queue ändern.

Die Struktur einer Antwort ist intuitiv. Sie enthält eine mit der Queue verknüpfte Anrede, den Text der Antwort, den zitierten Ticket-Text und schließlich die Signatur der Queue.

Automatische Antworten

OTRS bietet die Möglichkeit, automatische Antworten an Kundenbenutzer zu verschicken. Automatische Antworten sind an bestimmte Ereignisse im System gebunden, z. B. an das Anlegen eines neuen Tickets in einer Queue, wenn ein Followup eines Tickets stattfindet, wenn ein Ticket geschlossen oder vom System zurückgewiesen wird. Über den Link "Auto Antworten" innerhalb des Admin-Bereiches erreichen Sie die Verwaltung der automatischen Antworten.

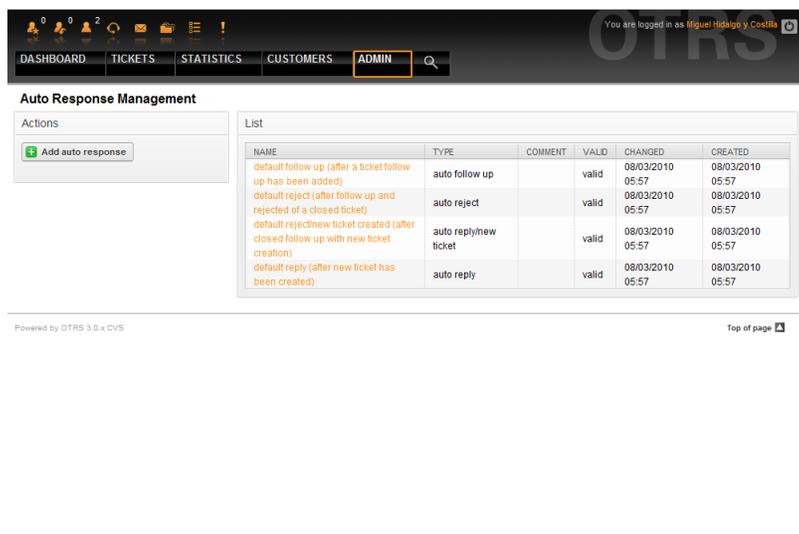


Abb. 5.34. Verwaltung automatischer Antworten.

Verwenden Sie den Knopf "Automatische Antwort hinzufügen", um eine neue anzulegen.

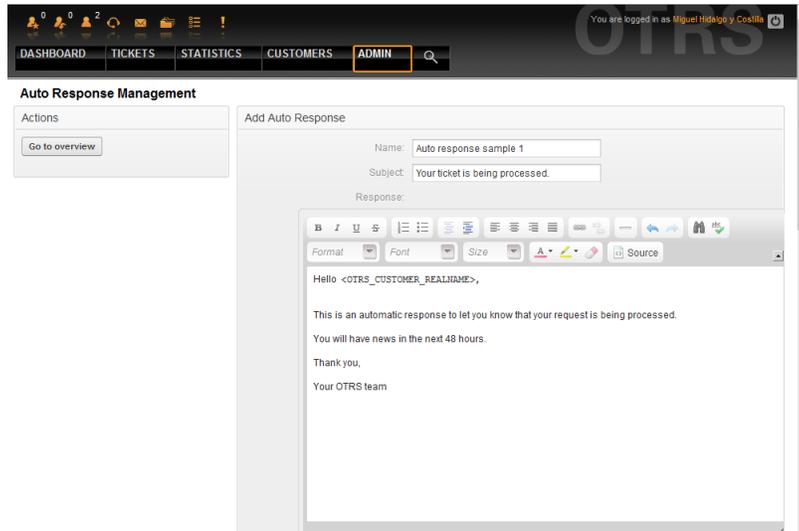


Abb. 5.35. Automatische Antwort hinzufügen.

Für die Betreffzeile und den Text von automatischen Antworten kann genauso wie bei Signaturen oder Anreden, der Inhalt mit Hilfe von OTRS-Variablen dynamisch erzeugt werden. So werden über die Variable <OTRS_CUSTOMER_EMAIL[5]> die ersten 5 Zeilen der an das System gesendeten E-Mail in die automatische Antwort eingefügt, oder durch <OTRS_CUSTOMER_FROM> die From-Zeile. Die Anmerkungen im unterem Bereich der Bildschirmmaske zur Verwaltung der automatischen Antworten listen alle OTRS-Variablen auf, die verwendet werden können.

Für jede automatische Antwort können Sie angeben, durch welches Ereignis sie ausgelöst werden soll. Folgende Ereignisse sind in einer Standardinstallation verfügbar:

Tabelle 5.3. Ereignisse für automatische Antworten

Name	Beschreibung
auto reply	Dieses Ereignis tritt ein, wenn ein neues Ticket in einer Queue angelegt wird.
auto reply/new ticket	Dieses Ereignis tritt ein, wenn ein bereits geschlossenes Ticket, z. B. durch die Antwort eines Kunden, mit einer neuen Ticketnummer erneut geöffnet wird.
auto follow up	Dieses Ereignis tritt ein, wenn ein Follow up für ein bereits vorhandenes Ticket eintrifft.
auto reject	Dieses Ereignis tritt ein, wenn ein Ticket vom System zurückgewiesen wird.
auto remove	Dieses Ereignis tritt ein, wenn ein Ticket vom System entfernt wird.

Anmerkung

In OTRS können automatische Antworten deaktiviert, aber nicht gelöscht werden. Zum Deaktivieren einer automatischen Antwort können Sie den Wert für "Gültig" auf "ungültig" oder "ungültig-temporär" setzen.

Der Administrationsbereich von OTRS

Um eine automatische Antwort einer oder mehreren Queues zuzuweisen, folgen Sie im Admin-Bereich den Link "Auto Antworten <-> Queues". Dort sind für jede Queue die verschiedenen Ereignistypen aufgelistet und es kann eine Zuordnung einer Auto-Antwort vom gleichen Ereignistyp durchgeführt bzw. entfernt werden.

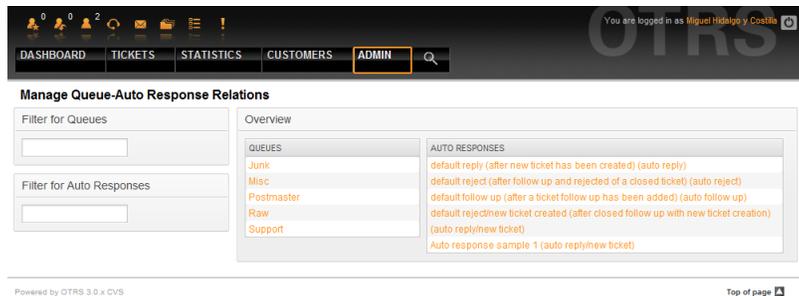


Abb. 5.36. Zuordnungen von *Queues* zu automatischen Antworten verwalten.

Um die Zuordnungen von automatischen Antworten zu Queues zu verändern, klicken sie auf den Namen der Queue. Durch einen Klick auf den Namen einer automatischen Antwort können Sie diese direkt bearbeiten.

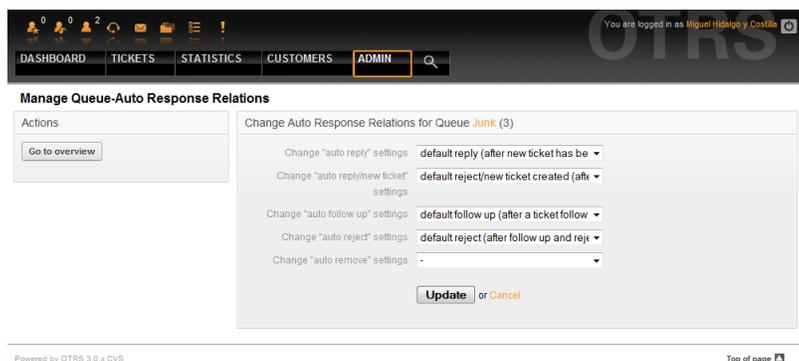


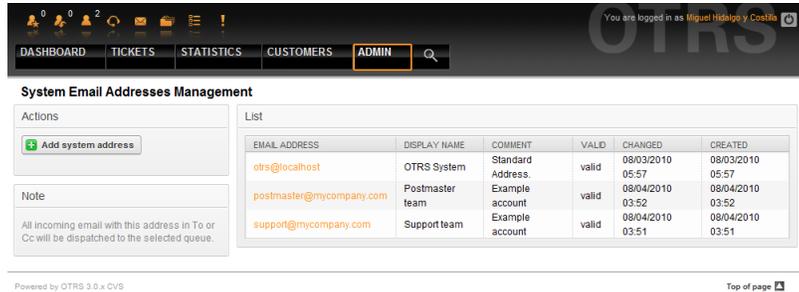
Abb. 5.37. Zuordnung von automatischen Antworten zu einer *Queue* bearbeiten.

E-Mail-Adressen

Um aus OTRS heraus E-Mails verschicken zu können, benötigen Sie mindestens eine gültige Mailadresse. Da in vielen Fällen eine Mailadresse nicht ausreicht, ist OTRS auch in der Lage, mit mehreren

Der Administrationsbereich von OTRS

Mailadressen zu arbeiten. Eine Queue mit mehreren E-Mail-Adressen verknüpft werden, und umgekehrt. Die Adresse, die für ausgehende Nachrichten einer Queue benutzt werden soll, kann bei Erstellung oder Bearbeitung der Queue ausgewählt werden. Verwenden Sie den Link "E-Mail-Adressen" im Administrationsbereich, um die Adressen des Systems zu verwalten.

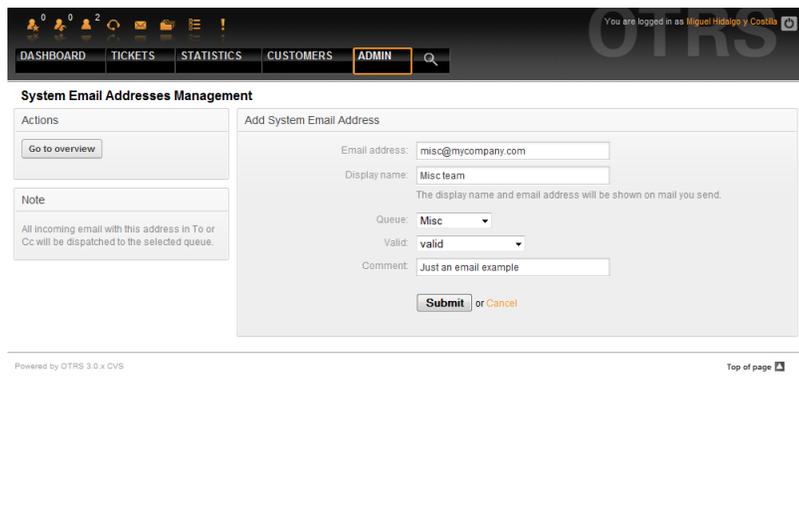


The screenshot shows the OTRS administration interface for "System Email Addresses Management". At the top, there is a navigation bar with "ADMIN" selected. Below the navigation bar, there are "Actions" and "List" sections. The "List" section contains a table with the following data:

EMAIL ADDRESS	DISPLAY NAME	COMMENT	VALID	CHANGED	CREATED
otrs@localhost	OTRS System	Standard Address.	valid	08/03/2010 05:57	08/03/2010 05:57
postmaster@mycompany.com	Postmaster team	Example account	valid	08/04/2010 03:52	08/04/2010 03:52
support@mycompany.com	Support team	Example account	valid	08/04/2010 03:51	08/04/2010 03:51

Abb. 5.38. Verwaltung der E-Mail-Adressen des Systems.

Im Formular zur Verwaltung der E-Mail-Adressen können Sie u.a. direkt auswählen, mit welcher Queue oder Unter-Queue die neue Mailadresse verknüpft werden soll. Diese Verknüpfung ist wichtig, da so eingehende Mails anhand der Adresse im To: Feld der entsprechenden Queue zugewiesen werden können.



The screenshot shows the "Add System Email Address" form in the OTRS administration interface. The form includes the following fields:

- Email address:
- Display name:
- Queue:
- Valid:
- Comment:

At the bottom of the form, there are "Submit" and "Cancel" buttons. A note below the form states: "The display name and email address will be shown on mail you send."

Abb. 5.39. Hinzufügen einer System-E-Mail-Adresse.

Anmerkung

In OTRS können E-Mail-Adressen deaktiviert, aber nicht gelöscht werden. Sie können eine Adresse deaktivieren, indem Sie den Wert für "Gültig" auf "ungültig" oder "ungültig-temporär" setzen.

Benachrichtigungen

Kunden und Mitarbeiter können im Einstellungs-Bereich ihres Accounts festlegen, bei welchen Ereignissen Sie automatisch vom System per Mail benachrichtigt werden wollen.

Über den "Benachrichtigungen"-Link im Admin-Bereich erreichen Sie die Verwaltung der Benachrichtigungen. Mit den Filtern können Sie bestimmte Einträge finden.

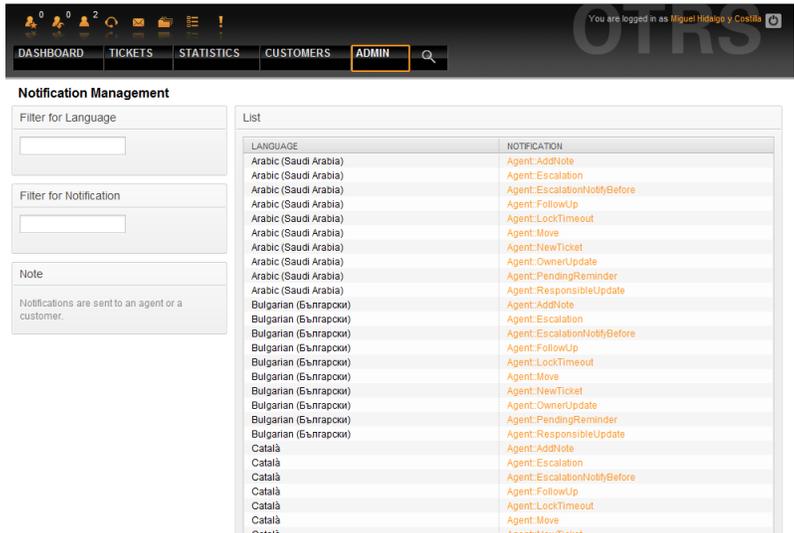


Abb. 5.40. Verwaltung der Benachrichtigungen.

Den Betreff und Text der Benachrichtigungen können Sie anpassen. Wählen Sie die gewünschte Benachrichtigung aus der Liste durch Klick aus. Beachten Sie: in jeder Sprache gibt es eine Benachrichtigung mit demselben Namen.

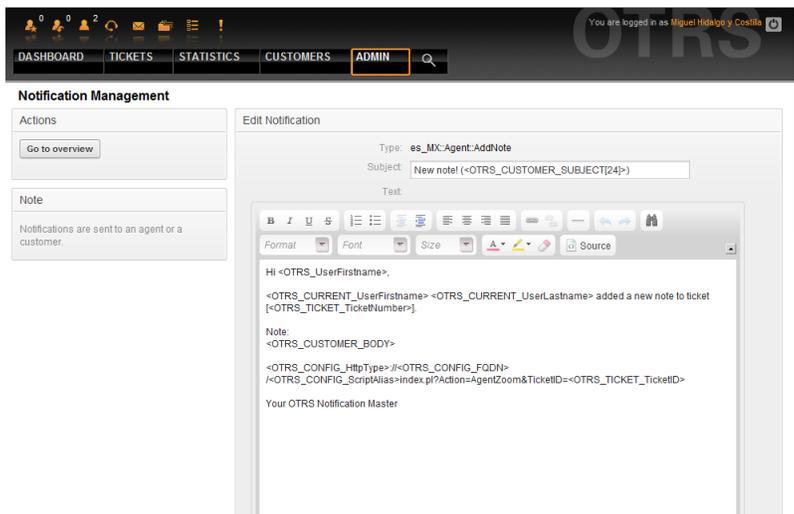


Abb. 5.41. Bearbeiten einer Benachrichtigung.

Auch innerhalb der Benachrichtigungen können die Textinhalte mit Hilfe der OTRS-Variablen dynamisch aufbereitet werden. Innerhalb der Anmerkungen im unteren Bereich der Bildschirmmaske zur

Der Administrationsbereich von OTRS

Benachrichtigungen-Verwaltung werden die verschiedenen zur Verfügung stehenden Variablen und deren Verwendung aufgelistet und näher erklärt.

Ebenso ist es möglich, Benachrichtigungen zu erstellen, die auf Ereignissen basieren. Hier können Sie detailliert einstellen, wann diese Benachrichtigung geschickt werden soll. Zudem kann ein weites Spektrum von Parametern konfiguriert werden, wie: Empfängergruppen(n), Benutzer, Rollen, E-Mail-Adressen; Ereignistyp, der die Benachrichtigung auslöst, Ticket-Typ, Status, Priorität, Sperre, Service, SLA, usw.

Um eine Übersicht aller ereignisbasierten Benachrichtigungen zu erhalten, klicken Sie auf "Benachrichtigungen (Ereignis)" im Administrationsbereich.

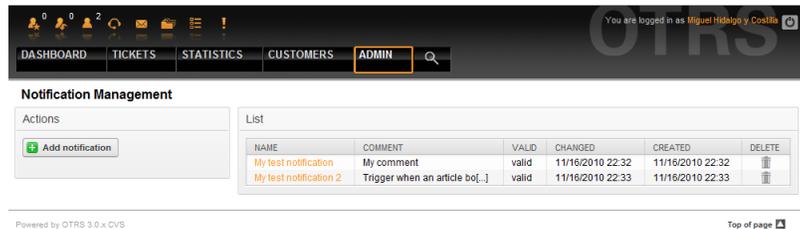


Abb. 5.42. Ereignisbasierte Benachrichtigungen verwalten.

Erstellen Sie eine neue Benachrichtigung durch Klick auf den "Hinzufügen"-Knopf.

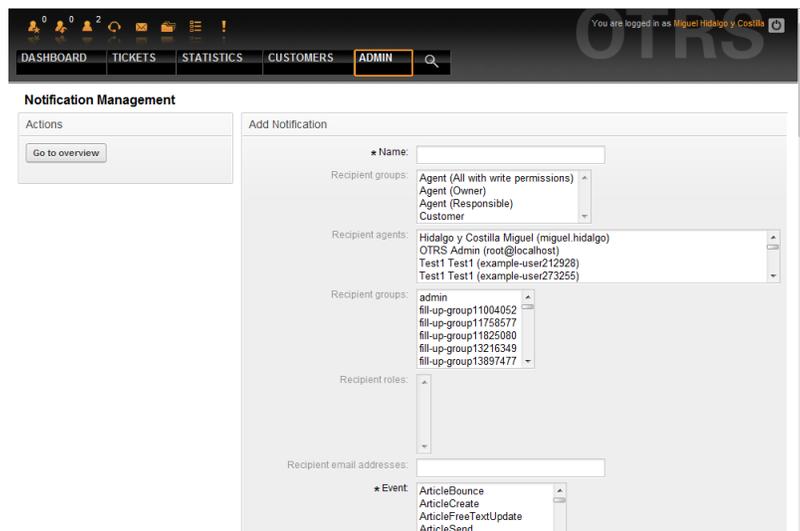


Abb. 5.43. Ereignisbasierte Benachrichtigung einrichten.

Auch in den ereignisbasierten Benachrichtigungen können Sie mit den speziellen Variablen dynamische Texte erzeugen. Die Referenz der Variablen sehen Sie im unteren Bildschirmbereich.

S/MIME

Mit OTRS ist es möglich, Tickets mit Hilfe von SMIME zu ent- und verschlüsseln bzw. Tickets zu signieren. Bevor SMIME allerdings systemweit genutzt werden kann, muss das Feature mit Hilfe einiger Konfigurationsparameter aktiviert und konfiguriert werden.

Über den "SMIME"-Link im Admin-Bereich von OTRS erreichen Sie die Verwaltung der SMIME-Zertifikate. Es können Zertifikate und private Fingerprints hinzugefügt und entfernt werden und eine Suche in den Zertifikaten ist möglich.

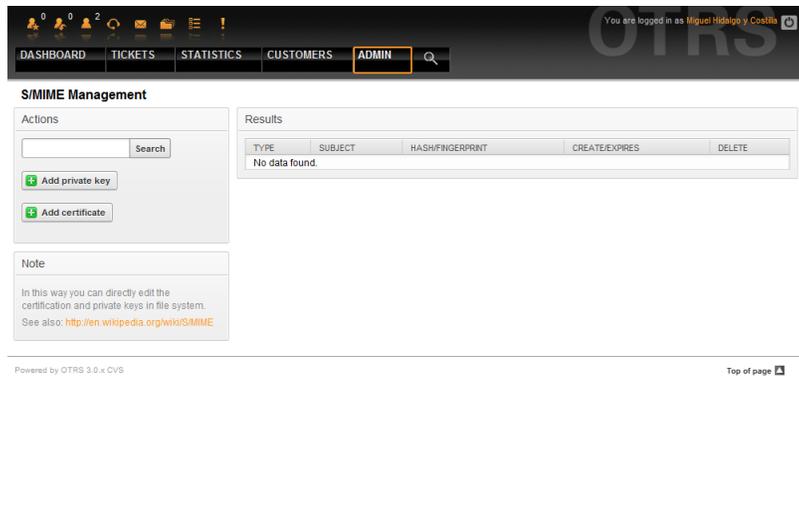


Abb. 5.44. S/MIME-Verwaltung.

PGP

Mit OTRS ist es möglich, Tickets mit Hilfe von PGP zu ent- und verschlüsseln bzw. zu signieren. Bevor PGP allerdings systemweit genutzt werden kann, muss das Feature mit Hilfe einiger Konfigurationsparameter aktiviert und konfiguriert werden.

Über den "PGP"-Link im Admin-Bereich von OTRS erreichen Sie die Verwaltung des Schlüsselrings Ihres Systems. Es können Schlüssel und Signaturen hinzugefügt und entfernt werden und eine Suche innerhalb des Schlüsselrings ist möglich.

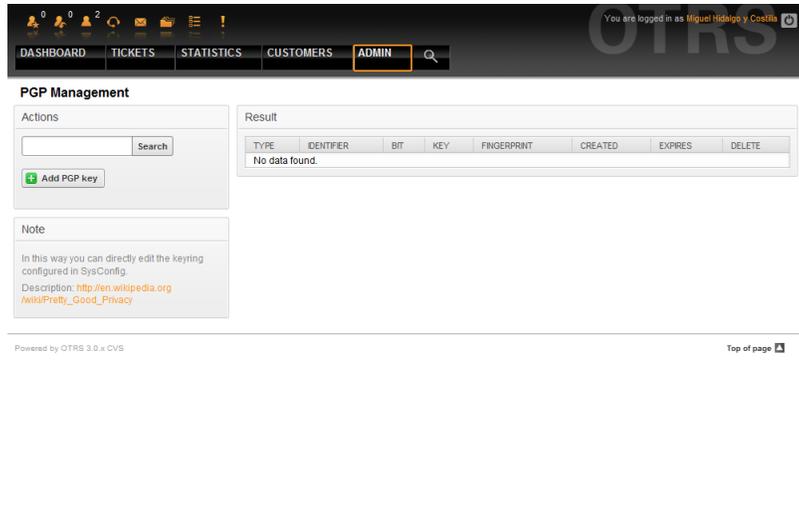


Abb. 5.45. PGP-Verwaltung.

Status

Über den Link "Status" können Sie die verschiedenen Ticket-Status bearbeiten, die Sie in OTRS verwenden möchten.

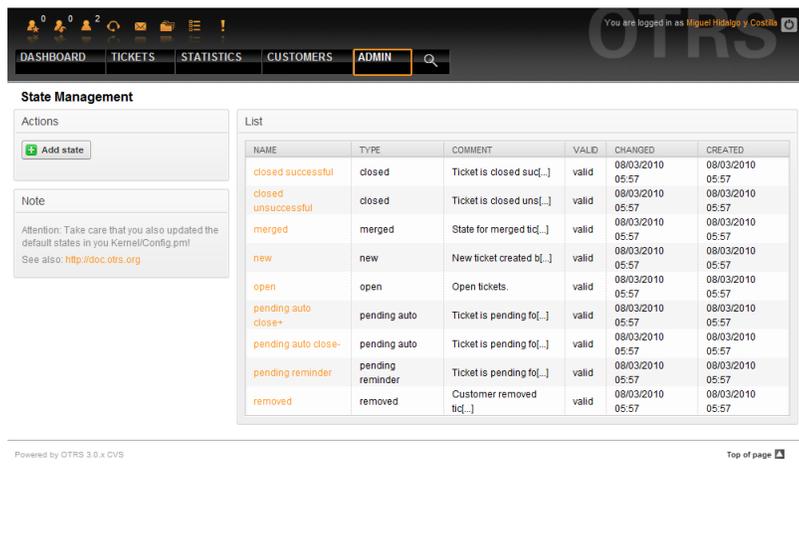


Abb. 5.46. Status-Verwaltung.

In einer Standardinstallation sind folgende Status definiert:

- erfolgreich geschlossen
- erfolglos geschlossen
- zusammengefasst
- neu
- öffnen

- warten auf erfolgreich schließen
- warten auf erfolglos schließen
- warten zur Erinnerung
- entfernt

Jeder Status ist mit einem Typ verküpft. Auch bei dem Neuanlegen eines Status muss ein Typ angegeben werden. Die standardmäßig verfügbaren Status-Typen sind:

- geschlossen
- zusammengefasst
- neu
- öffnen
- warten auto
- warten zur Erinnerung
- entfernt

Die SysConfig (Verwaltung der Systemkonfiguration)

In der SysConfig werden die meisten Konfigurationseinstellungen von OTRS vorgenommen.

Über den "SysConfig"-Link im Admin-Bereich von OTRS erreichen sie die grafische Administrationsoberfläche. Über diese können Sie eigene Konfigurationsdateien in das System integrieren bzw. Ihre persönlichen Änderungen an der Standardkonfiguration in eine Datei sichern. Nahezu alle Konfigurationsparameter des OTRS Frameworks und der zusätzlich installierten Module können eingesehen und geändert werden, die Navigation durch die Vielzahl der Konfigurationsparameter wurde durch die Einteilung in Module und Gruppen übersichtlich aufgeteilt. Weiterhin ist es möglich, eine Suche über alle Konfigurationsparameter hinweg durchzuführen, so dass einzelne Parameter schnell aufgefunden und bearbeitet werden können.

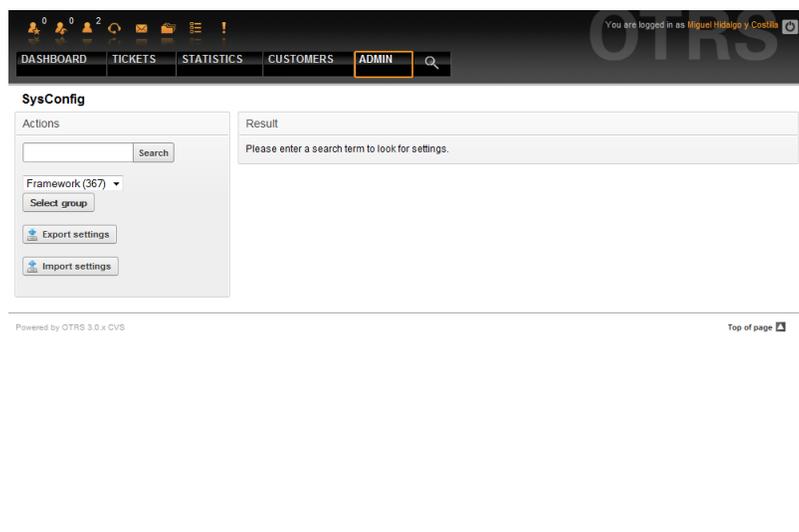


Abb. 5.47. Das grafische Konfigurationsfrontend von OTRS (SysConfig).

Das Kapitel Nähere Beschreibung der grafischen Administrationsoberfläche geht ausführlicher auf das grafische Konfigurations-Frontend ein.

Einrichten von Mail-Konten

Es gibt mehrere Möglichkeiten, Mails in OTRS einzuspeisen. Neben der Nutzung des PostMaster.pl-Skripts, welches die Mails direkt in das System piped, können auch Mail-Konten abgefragt werden, an die Mails für Ihr Ticket-System gehen. Über den Link "PostMaster Mail Account" innerhalb des Admin-Bereiches erreichen Sie die Verwaltung der Mail-Konten, die OTRS automatisch auf neue Nachrichten überprüfen soll. OTRS unterstützt dabei die Mail-Protokolle POP3, POP3S, IMAP und IMAPS.

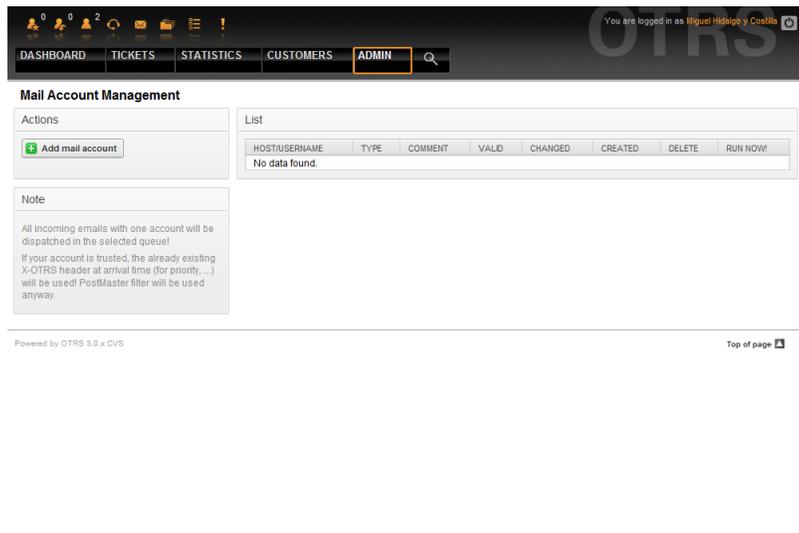


Abb. 5.48. Verwaltung von Mail-Konten.

Im Kapitel über die Mailkonfiguration für OTRS wird die Einrichtung der Mail-Konten näher erläutert.

Einkommende Nachrichten filtern

Damit Sie automatisch Mails vorsortieren können bzw. ungewünschte Mails gar nicht in die Queues des Ticket-Systems gelangen, können Filterregeln für eingehende Nachrichten festgelegt werden. Dabei spielt es keine Rolle, ob die Mails über Mail-Konten oder mit Hilfe des PostMaster.pl Skriptes ins System gelangen. Der Link "PostMaster Filter" innerhalb des Admin-Bereiches von OTRS ruft die Verwaltung für die Filterregeln auf.

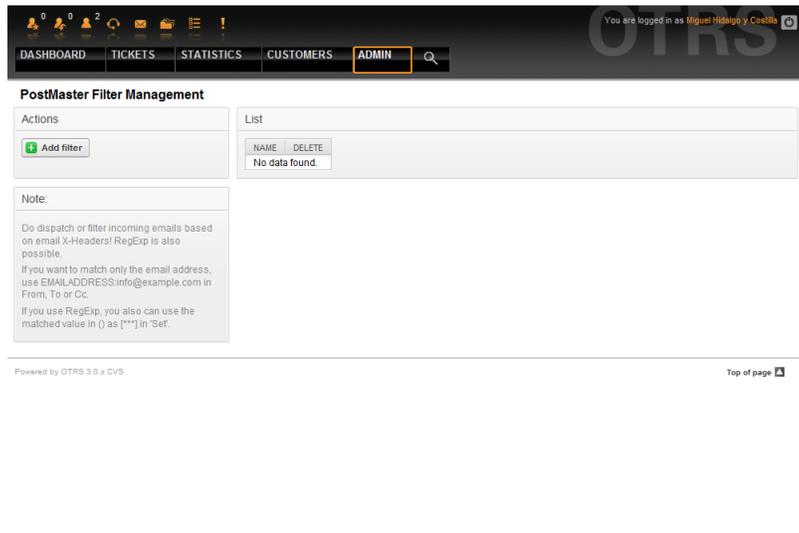


Abb 5.49. Verwaltung von Filterregeln für eingehende E-Mails.

Eine Filterregel besteht aus einem oder mehreren Kriterien die erfüllt werden müssen, damit die Filterregel greift, und aus einer oder mehreren Aktionen die ausgeführt werden, nachdem die Filterkriterien erfüllt wurden. Sie können für die Kopfzeilen und den Body einer Nachricht Filterregeln festlegen, also z. B. nach bestimmten Headern suchen oder nach Zeichenketten bzw. regulären Ausdrücken innerhalb der Mail. Alle Aktionen für eine Filterregel werden über sog. X-OTRS-Header gesteuert, die in die Mail eingefügt werden. Das Ticket-System wertet die X-OTRS-Header aus und nimmt die entsprechenden Aktionen vor. Mit den X-OTRS-Headern kann z. B. die Priorität einer Nachricht geändert werden, die Mail in eine bestimmte Queue einsortiert oder die Mail komplett verworfen werden, usw. Die folgende Tabelle listet die verschiedenen X-OTRS-Header, deren zulässige Werte und die Bedeutung der einzelnen Header auf.

Hinweis: Sie können auch X-OTRS-FollowUp-* header für Follow-Up-E-Mails verwenden.

Tabelle 5.4. Funktion der verschiedenen X-OTRS-Header

Name	Mögliche Werte	Beschreibung
X-OTRS-Priority:	1 very low, 2 low, 3 normal, 4 high, 5 very high	Legt die Priorität eines Tickets fest.
X-OTRS-Queue:	Name einer Queue des Systems.	Legt die Queue fest, in die das Ticket einsortiert werden soll. Wird mit dem X-OTRS-Queue-Header eine Queue voreingestellt, so hat diese Einstellung Vorrang vor allen anderen Filterregeln, die sich auf Queues beziehen.
X-OTRS-Lock:	lock, unlock	Setzt den "gesperrt"-Status eines Tickets.
X-OTRS-Ignore:	Yes oder True	Wird dieser Header gesetzt, wird die Mail komplett ignoriert und gelangt somit nie als Ticket in das System.
X-OTRS-State:	new, open, closed successful, closed unsuccessful, ...	Setzt den nächsten Status eines Tickets.

Der Administrationsbereich
von OTRS

Name	Mögliche Werte	Beschreibung
X-OTRS-State-PendingTime:	Z. B. 2010-11-20 00:00:00	Setzt die Wartezeit eines Tickets (Sie sollten dann ebenfalls einen "Warten"-Status per X-OTRS-State setzen).
X-OTRS-Type:	default (systemabhängig)	Setzt den Typ eines Tickets (wenn Ticket::Type aktiviert ist).
X-OTRS-Service:	(systemabhängig)	Setzt den Service eines Tickets (wenn Ticket::Service aktiv ist).
X-OTRS-SLA:	(systemabhängig)	Setzt das Service-Level-Abkommen eines Tickets (wenn Ticket::Service aktiv ist).
X-OTRS-CustomerUser:	CustomerUser	Legt den Kunden-Benutzer fest, dem das Ticket zugeordnet werden soll.
X-OTRS-CustomerNo:	CustomerNo	Legt die Kunden-ID fest, die dem Ticket zugeordnet werden soll.
X-OTRS-ArticleKey(1 2 3):	Zusätzlicher Info-Key für den Artikel.	Speichert einen zusätzlichen Info-Key für den Artikel.
X-OTRS-ArticleValue(1 2 3):	Zusätzlicher Info-Value.	Speichert einen zusätzlichen Info-Value für den Artikel.
X-OTRS-SenderType:	agent, system, customer	Legt die Art des Ticket-Absenders fest.
X-OTRS-ArticleType:	email-external, email-internal, email-notification-ext, email-notification-int, phone, fax, sms, webrequest, note-internal, note-external, note-report	Legt den Typ des Artikles für das eingehende Ticket fest.
X-OTRS-TicketKey(1 2 ... 8):	Zusätzlicher Info-Key	Speichert einen zusätzlichen Info-Key für das Ticket.
X-OTRS-TicketValue(1 2 ... 8):	Zusätzlicher Info-Value	Speichert einen zusätzlichen Info-Value für das Ticket.
X-OTRS-Loop:	True	Ist dieser X-OTRS-Header gesetzt, wird keine automatische Antwort an den Absender des neuen Tickets geschickt, z. B. um Mailschleifen zu vermeiden.

Sie müssen für jede Regel einen Namen vergeben. Darunter legen Sie in der Sektion für "Treffer" die verschiedenen Kriterien fest, nach denen gefiltert werden soll. Wählen Sie aus den Listboxen für "Header 1", "Header 2" usw. aus, an welcher Stelle in der eingehenden Nachricht Sie suchen möchten und tragen Sie rechts neben dem entsprechenden Header den Wert ein, nach dem gesucht werden soll. In der Sektion "Setzen" legen Sie die Aktionen fest, die ausgeführt werden sollen, wenn die Filterkriterien zutreffen. Hier können Sie für "Header1", "Header 2" usw. den entsprechenden X-OTRS-Header auswählen und rechts daneben den entsprechenden Wert eintragen.

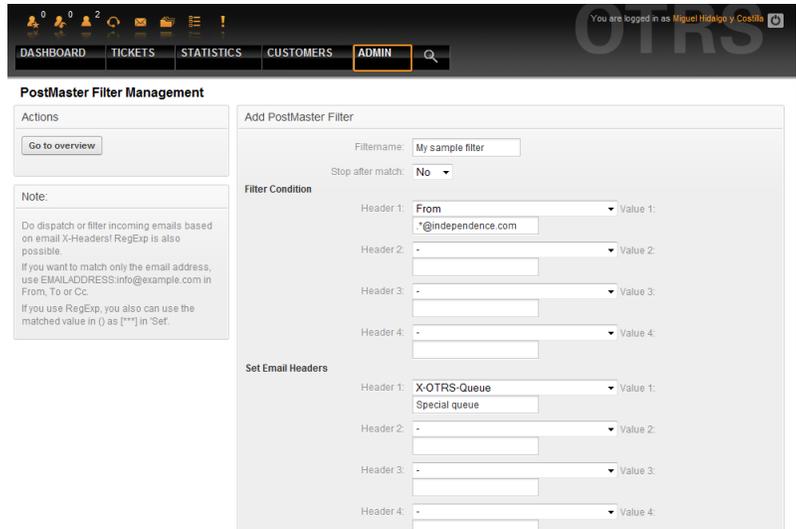


Abb. 5.50. Filter-Regeln für eingehende Nachrichten anlegen.

Beispiel 5.1. Aussortierung von Spammails in eine bestimmte Queue

Eine nützliche Filterregel könnte sein, alle Mails, die von SpamAssassin als Spam markiert wurden, automatisch in die Queue "Junk" einsortieren zu lassen. Spamassassin fügt bei allen überprüften Mails die Kopfzeile "X-Spam-Flag" hinzu. Wird die Mail als Spam markiert, wird dieser Header auf "Yes" gesetzt. Das Filterkriterium lautet also "X-Spam-Flag: Yes". Um nun eine Filterregel mit diesem Kriterium zu erzeugen, tragen Sie hierzu als Name für die Filterregel z. B. "spam-mails" ein und wählen in der Sektion "Treffer" für "Header 1" den Header "X-Spam-Flag:" aus der Listbox aus. Rechts daneben fügen Sie als Wert "Yes" hinzu. Das Filterkriterium wurde somit festgelegt. Um nun die Einsortierung der von spamassassin als Spam klassifizierten Mails in die Queue "Junk" zu erzwingen, wählen Sie in der Sektion "Setzen" für "Header 1" den Eintrag "X-OTRS-Queue:" aus und tragen als Wert rechts daneben "Junk" ein. Abschließend wird mit Hilfe des "Hinzufügen"-Buttons die neue Regel gespeichert und aktiviert, beim nächsten Abruf der POP3-Konten bzw. für die nächste an das System gesendete Nachricht wird die Filterregel abgearbeitet.

Es gibt weitere Module, die zum Filtern eingehender E-Mails genutzt werden und bei komplexeren Installationen nützlich sein können.

Ausführen von automatisierten Jobs mit Hilfe des GenericAgents

Der GenericAgent ist ein Tool zum automatischen Ausführen von Aufgaben, die normalerweise ein richtiger Agent manuell durchführen müsste. Er kann z. B. bestimmte Tickets in einer Queue schließen oder Benachrichtigungen für eskalierte Tickets versenden.

Um zum GenericAgent zu gelangen, verwenden Sie die Verknüpfung "GenericAgent" in der Administrationsoberfläche von OTRS. Eine Liste verfügbarer Jobs wird angezeigt, welche bearbeitet, ausgeführt oder gelöscht werden können.

Der Administrationsbereich von OTRS

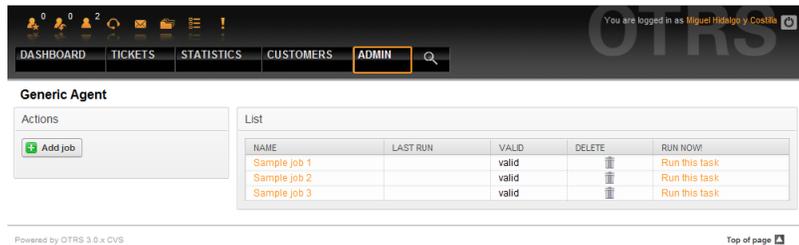


Abb. 5.51. Jobliste des GenericAgent.

Über die Bildschirmmaske zum Erstellen eines neuen Jobs kann der Zeitplan für die Ausführung des Jobs eingestellt werden. Desweiteren kann über verschiedene Kriterien festgelegt werden, welche Tickets vom Job erfasst werden sollen. Schließlich ist es möglich, die neuen Eigenschaften der vom Job betroffenen Tickets einzustellen.

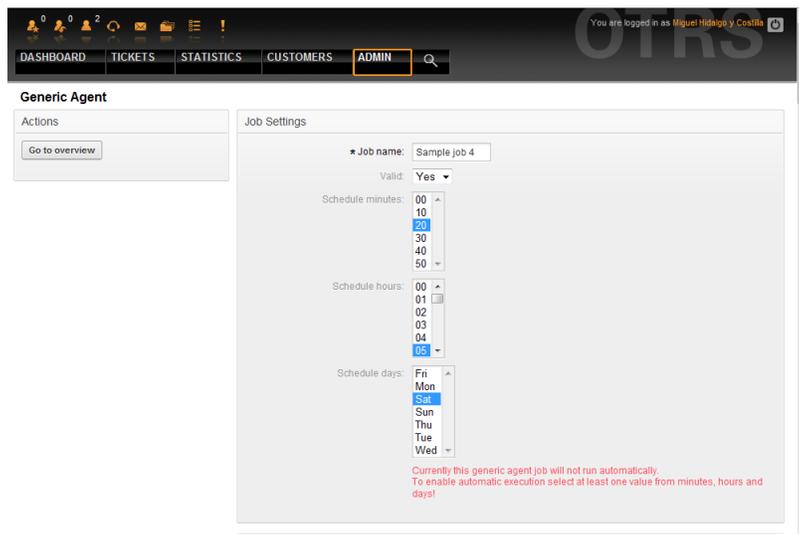


Abb. 5.52. Anlegen eines GenericAgent-Jobs.

Nachdem der Job gespeichert wurde, werden alle Tickets aufgeführt, die durch den Job verändert werden. Diese Auflistung dient zur Übersicht, ob der Job richtig funktioniert, es werden noch keine Veränderungen vorgenommen. Erst nach der Übernahme des Jobs in die Job-Liste, wird der Job aktiviert.

Administrator-E-Mail

Um als OTRS-Administrator eine Mail an bestimmte Benutzer (Agenten) oder Benutzergruppen im System zu versenden, folgen Sie dem Link "Admin Notification".

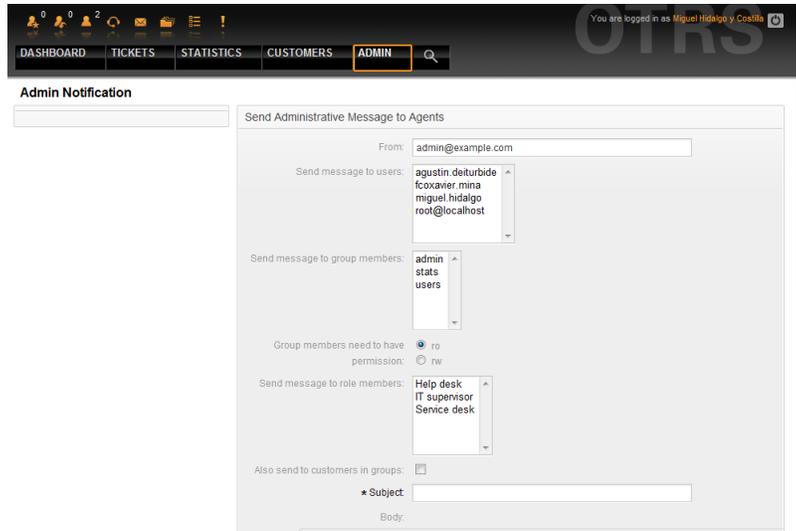


Abb. 5.53. Administrator-Benachrichtigung.

Es wird ein Formular geladen, in das Sie die Absenderadresse, den Betreff und den Inhalt der Administratoren-Mitteilung eintragen können. Weiterhin können Sie aus der Tabelle auswählen, an welche Benutzer und / oder Benutzergruppen bzw. Rollen die Mitteilung gesendet werden soll.

Sitzungsverwaltung

Um eine Übersicht über die gerade im System angemeldeten Benutzer und deren Sitzungseigenschaften zu erhalten, folgen Sie dem Link "Sitzungsverwaltung" der Administrationsoberfläche.

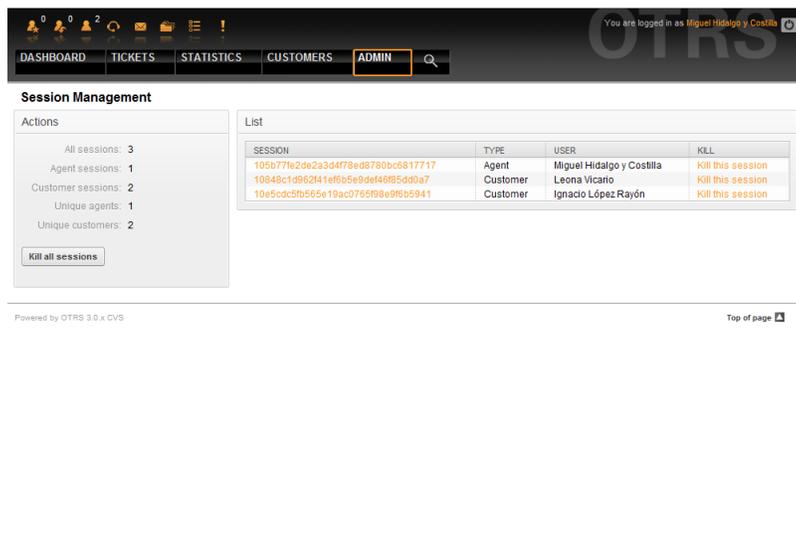


Abb. 5.54. Sitzungsverwaltung.

Innerhalb der Sitzungsverwaltung werden allgemeine Informationen zu allen Sitzungen im System ausgegeben, also z. B. wie viele Sitzungen gerade insgesamt aktiv sind oder wie viele Agenten- und Kunden-Sitzungen laufen, usw. Es besteht die Möglichkeit mit Hilfe des "Alle Sitzungen löschen"-Schalters die Sitzungen aller angemeldeten Benutzer zu beenden. Weiterhin können detaillierte Informationen für jede einzelne Sitzung abgerufen und einzelne Sitzungen gelöscht werden.

Der Administrationsbereich von OTRS

The screenshot shows the OTRS Admin interface with the 'ADMIN' tab selected. The main content area is titled 'Session Management' and displays 'Detail View for SessionID : 105b77fe2de2a3d4f78ed8780bc6817717 - Miguel Hidalgo y Costilla'. On the left, there are two buttons: 'Go to overview' and 'Kill this session'. The main area contains a table with the following data:

KEY	VALUE
ChangeTime	2010-08-03 06:01:50
CreateTime	2010-08-03 06:01:50
OutOfOfficeEndDay	4
OutOfOfficeEndMonth	8
OutOfOfficeEndYear	2010
OutOfOfficeStartDay	3
OutOfOfficeStartMonth	8
OutOfOfficeStartYear	2010
SessionID	105b77fe2de2a3d4f78ed8780bc6817717
UserChallengeToken	44305769205c0299e0c213b1b072eb6
UserEmail	miguel.hidalgo@freedom.com
UserFirstname	Miguel
UserID	2
UsersGroupRo[admin]	Yes
UsersGroupRo[stats]	Yes
UsersGroupRo[users]	Yes
UsersGroup[admin]	Yes
UsersGroup[stats]	Yes
UsersGroup[users]	Yes
UserLanguage	en
UserLastLogin	1280935411
UserLastLoginTimestamp	2010-08-04 10:23:31
UserLastPw	xxxxxxx
UserLastRequest	1280935483
UserLastname	Hidalgo y Costilla
UserLogin	miguel.hidalgo
UserPw	xxxxxxx

Abb. 5.55. Sitzungsdetails.

System Log

Der Link "System Log" ermöglicht es, die letzten Logeinträge des Ticket-Systems über die Weboberfläche einzusehen.

The screenshot shows the OTRS Admin interface with the 'ADMIN' tab selected. The main content area is titled 'System Log'. On the left, there is a 'Note' box with the text: 'Here you will find log information about your system.' The main area contains a table with the following data:

TIME	PRIORITY	FACILITY	MESSAGE
Wed Aug 4 10:23:51 2010	notice	OTRS-CGI-10	Removed SessionID 10b2c771b4115052ca665a7559e997c7b3.
Wed Aug 4 10:23:31 2010	notice	OTRS-CGI-10	User: miguel.hidalgo authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:23:12 2010	notice	OTRS-CGI-10	Removed SessionID 102d146b9b92158482b55deb742695ab7.
Wed Aug 4 10:12:36 2010	notice	OTRS-CGI-10	CustomerUser: ignacio.lopez Authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:12:17 2010	notice	OTRS-CGI-10	CustomerUser: leona.vicario Authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:11:05 2010	notice	OTRS-CGI-10	Removed SessionID 107bdfaba2f0ed4abe496b1f69fc31921.
Wed Aug 4 10:11:05 2010	notice	OTRS-CGI-10	User: agustin.delturbide authentication ok (REMOTE_ADDR: 192.168.56.1).

Abb. 5.56. System Log.

Ein Logeintrag setzt sich aus der Zeit, der Priorität, der betroffenen Systemkomponente und der eigentlichen Meldung zusammen.

Anmerkung

Die System Logs können nur auf Unix- oder Linux-Systemen über das Web-Interface eingesehen werden, unter Windows-Betriebssystemen können Sie stattdessen die Datei [install_dir]otrs\var\log\otrs.log mit einem Text-Editor öffnen.

SQL-Abfragen mit Hilfe der SQL Box

Über den Link "SQL Box" kann eine Bildschirmmaske aufgerufen werden, die es ermöglicht direkt mit SQL-Statements den Inhalt von Tabellen der OTRS-Datenbank abzurufen. Es sind nur SELECT-Abfragen möglich, d.h. die Tabellen können auf diesem Weg nicht verändert werden.

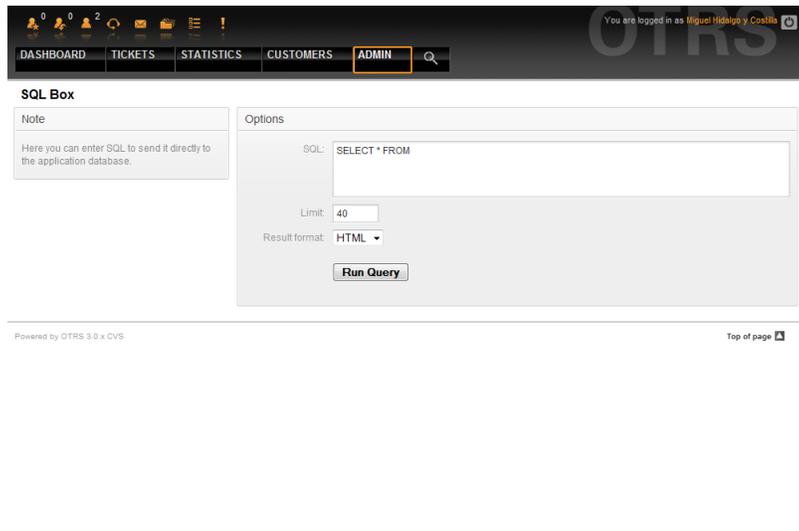


Abb. 5.57. SQL Box.

Paketverwaltung

Mit der "Paketverwaltung" in der Administrationsoberfläche können Sie Pakete installieren und verwalten, die die Funktionalität von OTRS erweitern. Eine Liste von verfügbaren Erweiterungsmodulen von OTRS finden Sie unter Zusätzliche Applikationen.

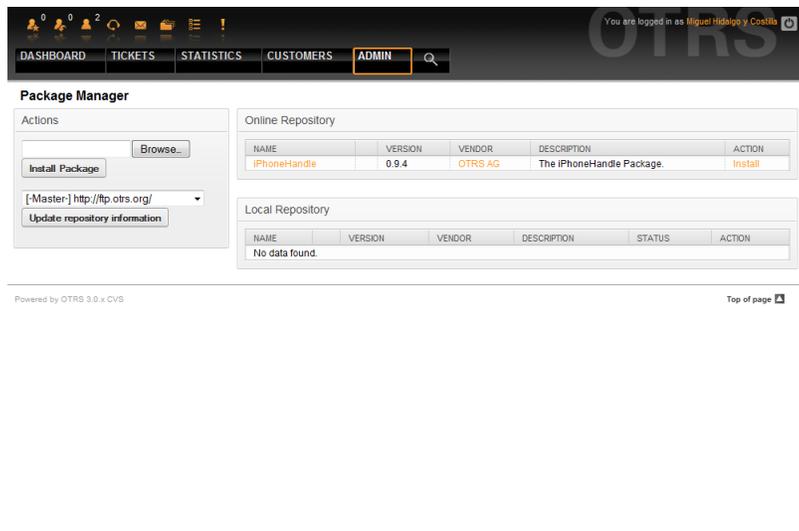


Abb. 5.58. Paketverwaltung.

In der Paketverwaltung können Sie die bereits installierten Pakete mit ihrer Versionsinformation einsehen.

Sie können Pakete von einem entfernten Server installieren, indem Sie das Paketverzeichnis im Abschnitt *Online-Paketverzeichnis* auswählen und auf *Verzeichnisinformation aktualisieren* klicken. Die verfügbaren Pakete werden rechts tabellarisch dargestellt. Um eines zu installieren, klicken Sie auf "Installieren". Nach der Installation wird das Paket im Abschnitt *Lokales Verzeichnis* dargestellt.

Um ein installiertes Paket zu aktualisieren, können Sie die Funktion "Aktualisieren" benutzen, die bei installierten Paketen angezeigt wird, von denen eine neuere Version verfügbar ist. Klicken Sie einfach auf "Aktualisieren", und die neue Version wird in Ihrem System installiert.

In manchen Fällen, wenn beispielsweise ein OTRS-System nicht mit dem Internet verbunden ist, können Sie auch Pakete installieren, die lokal heruntergeladen wurden. Klicken Sie auf den *Datei auswählen*-Knopf in der "Aktionen"-Liste, und wählen Sie die .opm-Datei auf Ihrem Speichermedium aus. Verwenden Sie dann *Paket installieren*. Nach der Installation wird das Paket im Abschnitt *Lokales Verzeichnis* angezeigt. Auf dieselbe Weise können Sie ein bereits installiertes Paket aktualisieren.

In besonderen Fällen ist es erforderlich, die Paketverwaltung zu konfigurieren, zum Beispiel für die Verwendung eines Proxy-Servers, oder um auf ein lokales Verzeichnis zuzugreifen. Verwenden Sie dazu die Einstellungen unter `Framework:Core::Package`.

Kapitel 6. Systemkonfiguration

Die Konfigurationsdateien von OTRS

Alle Konfigurationsdateien des OTRS-Frameworks befinden sich innerhalb des Verzeichnisses `Kernel` bzw. in Unterverzeichnissen dieses `Directory`s. Bis auf die Datei `Kernel/Config.pm` sollten Sie keine Konfigurationsdatei manuell verändern, da alle anderen Dateien beim Updaten des Systems überschrieben werden und so Ihre eigenen Einstellungen verloren gehen. Übertragen Sie lediglich die Parameter aus den anderen Dateien nach `Kernel/Config.pm` und passen Sie die Parameter Ihren Wünschen entsprechend an.

Die Datei `Kernel/Config/Defaults.pm` enthält die Konfigurationsparameter für den OTRS-Framework. In ihr finden Sie grundlegende Einstellungen wie die Mailkonfiguration, die Datenbankbindung, Standardsprache o.ä. In der Datei `Kernel/Config/Files/Ticket.pm` sind alle Konfigurationsparameter für das Ticketsystem aufgeführt.

Das Verzeichnis `Kernel/Config/Files` enthält weitere Konfigurationsdateien, die beim Starten von OTRS eingelesen werden. Sind zusätzliche Module wie der Filemanager oder der Webmailer installiert, liegen die Konfigurationsdateien dieser Applikationen ebenfalls in `Kernel/Config/Files`.

Wird die Web-Oberfläche von OTRS aufgerufen, werden die `xml`-Dateien in `Kernel/Config/Files` in alphabetischer Reihenfolge ausgelesen und die Einstellungen des Frameworks und der evtl. zusätzlich installierten Applikationen geladen. Anschließend werden die Einstellungen in den Dateien `Kernel/Config/Files/ZZZAAuto.pm` und `Kernel/Config/Files/ZZZAuto.pm` ausgewertet. Beide Dateien werden vom grafischen Konfigurations-Frontend angelegt und sollten auf keinem Fall manuell geändert werden. Zuletzt wird die Datei `Kernel/Config.pm` mit den von Ihnen individuell angepassten Konfigurationsparametern ausgewertet, so dass auf jeden Fall Ihre eigenen Einstellungen geladen werden.

Konfiguration des Systems mit Hilfe des grafischen Konfigurations-Frontends

Seit OTRS 2.0 können nahezu alle Konfigurationsparameter des Frameworks oder der ggf. zusätzlich installierten Applikationen bequem über ein grafisches Frontend angepasst werden. Melden Sie sich als OTRS-Administrator an und folgen Sie im Administrationsbereich den Link "SysConfig", um das grafische Konfigurations-Frontend zu laden.

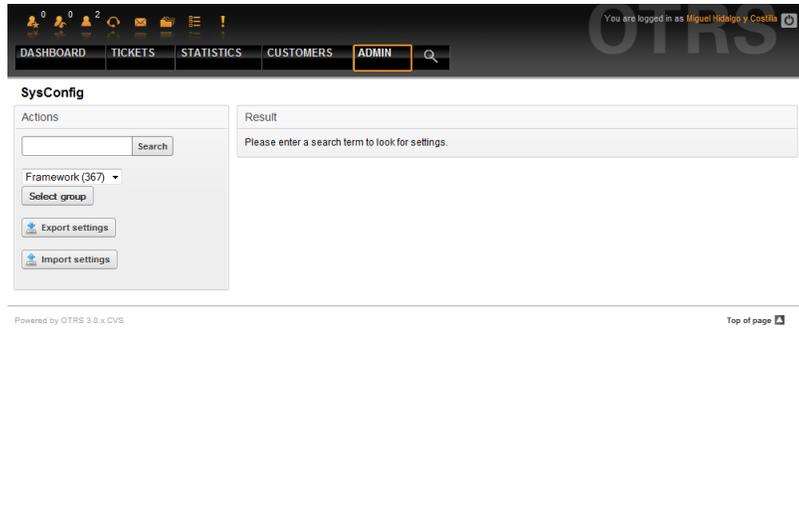


Abb. 6.1. Die grafische System-Konfigurationsoberfläche.

Da OTRS mittlerweile über mehr als 600 verschiedene Konfigurationsparameter verfügt, bietet das Konfigurations-Frontend mehrere Möglichkeiten zur schnellen Auffindung der gewünschten Einstellung. Es kann über alle Konfigurationsparameter hinweg nach einem bestimmten Stichwort gesucht werden. Bei der Suche werden neben dem Namen des Konfigurationsparameters auch die Beschreibungen ausgewertet, eine Einstellung kann also auch gefunden werden, wenn ihr Name nicht bekannt ist.

Weiterhin wurden die verschiedenen Konfigurationsparameter in Haupt- und Untergruppen unterteilt. Die Hauptgruppe stellt die Applikation dar, für die der Konfigurationsparameter zuständig ist, also z. B. "Framework" für das OTRS-Framework oder "Ticket" für das Ticketsystem. Die Untergruppen einer Hauptgruppe können eingesehen werden, indem die Gruppe bzw. Applikation aus der dazu vorgesehenen Listbox ausgewählt und der "Zeigen"-Knopf gedrückt wird.

Für jeden einzelnen Konfigurationsparameter kann über eine Checkbox festgelegt werden, ob er vom System beachtet werden soll oder nicht. Wird eine Einstellung verändert, kann die Änderung mit Hilfe des "Aktualisieren"-Buttons übernommen werden. Eine Einstellung kann mit Hilfe des "Rücksetzen"-Schalters auf ihren Default-Wert zurückgesetzt werden.

Für die Sicherung aller von Ihnen vorgenommenen Änderungen, kann eine .pm-Datei heruntergeladen werden, die alle vom Standard abweichenden Konfigurationsparameter Ihres Systems enthält. Dieselbe Datei können Sie ebenfalls über die Konfigurationsoberfläche eines frisch installierten Systems zurückspielen und so alle Einstellungen wiederherstellen.

Anmerkung

Die Einstellungen für die Datenbankanbindung können aus Sicherheitsgründen nicht über das grafische Konfigurations-Frontend geändert werden und müssen manuell in die Datei `Kernel/Config.pm` eingefügt werden.

Kapitel 7. E-Mails versenden/ empfangen

E-Mails versenden

Via Sendmail (Standard)

OTRS ist in der Lage, E-Mails via Sendmail (z. B. Sendmail [<http://www.sendmail.org/>], Postfix [<http://www.postfix.org/>], Qmail [<http://www.qmail.org/>] oder Exim [<http://www.exim.org/>]) zu versenden. Die Standard-Konfiguration sollte gleich ohne Probleme funktionieren.

Die Konfiguration kann über die grafische Administrationsoberfläche (Framework::Core::Sendmail) vorgenommen werden

Via SMTP server/smarthost

Wenn kein sendmail-Binary zur Verfügung steht, kann OTRS E-Mails via SMTP (Simple Mail Transfer Protocol / RFC 821 [<http://www.ietf.org/rfc/rfc821.txt>]) oder Secure SMTP versenden. Diese Möglichkeit kann hauptsächlich auf Nicht-Unix-Plattformen (z. B. Win32) genutzt werden.

Die SMTP-Einstellungen können über die SysConfig vorgenommen werden (Framework::Core::Sendmail). Wenn SMTPS nicht als Option auftaucht, fehlen benötigte Perl-Module. Lesen Sie in diesem Fall im Abschnitt "Installation benötigter Perl-Module" nach.

E-Mails empfangen

E-Mail-Konten, die über die OTRS-Oberfläche konfiguriert werden

OTRS ist in der Lage, E-Mails von POP3, POP3S, IMAP und IMAPS E-Mail-Konten zu empfangen.

Konfigurieren Sie Ihre Mail-Konten im Admin-Bereich von OTRS im Bereich "PostMaster Mail Account".

Beim Anlegen eines neuen Mail-Accounts muss der Mail-Server, ein Login und ein Kennwort angegeben werden. Zusätzlich muss der Mail-Server-Typ angegeben werden, es werden die Protokolle POP3, POP3S, IMAP und IMAPS unterstützt. Wenn Sie den gewünschten Typ nicht als Option zur Verfügung haben, fehlen benötigte Perl-Module. Lesen Sie in diesem Fall im Abschnitt "Installation benötigter Perl-Module" nach.

The screenshot shows the OTRS Admin interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN. The ADMIN tab is selected. Below the navigation bar, there is a section titled 'Mail Account Management'. On the left, there are two boxes: 'Actions' with a 'Go to overview' button, and 'Note' with a warning message: 'All incoming emails with one account will be dispatched in the selected queue! If your account is trusted, the already existing X-OTRS header at arrival time (for priority, ...) will be used! PostMaster filter will be used anyway.' The main area is titled 'Add Mail Account' and contains a form with the following fields: Type (POP3), Username (miquel.hidalgo), Password (masked with dots), Host (mail.independencia.com), Example (mail.example.com), Trusted (Yes), Dispatching (Dispatching by email To: field), Queue (Postmaster), Valid (valid), and Comment (Example of mail account). At the bottom of the form are 'Submit' and 'Cancel' buttons. The footer of the page indicates 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

Abb 7.1. Einen E-Mail-Account hinzufügen.

Wählen Sie für "Vertraut" den Wert "Ja" aus, dann werden die sog. X-OTRS-Header-Einträge ausgewertet und angewendet, sofern derartige Header-Einträge in einer abgerufenen Nachricht vorhanden sind. Da mit Hilfe der X-OTRS-Header einige Dinge am System beeinflusst werden können, sollten Sie "Vertraut" nur auf "Ja" setzen, wenn Sie genau wissen, von welchen Absendern die abgerufenen Nachrichten stammen. X-OTRS-Header werden vom Modul für die Nachrichtenfilterung in OTRS benutzt, die X-OTRS-Header werden in dieser Tabelle näher beschrieben. Eventuell eingerichtete Filterregeln werden auf jeden Fall abgearbeitet, egal welcher Wert für "Vertraut" ausgewählt wurde.

Weiterhin können Sie die Verteilung der abgerufenen Mails durch die Angabe steuern, ob die neuen Nachrichten nach dem To-Feld oder nach der Queue im System einsortiert werden sollen. Wählen Sie "Verteilung nach ausgewählter Queue" aus, landen die abgerufenen Mails auf jeden Fall in der Queue, die zusätzlich in der dafür vorgesehenen Listbox angegeben werden kann. Dabei spielt keine Rolle, an welche Adresse die Mail geschickt wurde. Wählen Sie "Verteilung nach To: Feld" aus, wird überprüft, welcher Queue die Adresse zugeordnet ist, an die die abgerufene Mail gesendet wurde. Die Zuordnung einer Mailadresse zu einer Queue kann über die Mailadressen Verwaltung vorgenommen werden. Existiert eine Zuordnung der Adresse im To: Feld zu einer Queue innerhalb des Systems, wird die abgerufene Nachricht in die entsprechende Queue einsortiert. Kann keine Zuordnung gefunden werden, landet das Ticket in der Standard-Queue des Systems (Raw), die mit Hilfe des Konfigurationsparameters PostmasterDefaultQueue eingestellt werden kann.

Die Daten zu allen Mail-Konten werden in der Datenbank von OTRS gespeichert. Das Skript `otrs.PostMasterMailbox.pl`, welches sich im Verzeichnis `bin` innerhalb des OTRS-Homeverzeichnisses befindet, fragt die Einstellungen ab und holt die Mails von den einzelnen Mail-Konten. Sie können dieses Skript manuell ausführen, um zu prüfen, ob alle E-Mail-Einstellungen korrekt funktionieren.

Auf einer Standardinstallation holt der Cron-Job `postmaster_mailbox` die E-Mails alle 10 Minuten ab. Das Kapitel "Einrichten der von OTRS benötigten cron-Jobs" beschreibt das Zusammenspiel zwischen OTRS und cron ausführlicher.

Anmerkung

Beim Abholen löscht OTRS die E-Mails vom POP oder IMAP server. Es besteht keine Möglichkeit, die Mails dort zu erhalten. Wenn Sie dies wünschen, erstellen Sie am Besten

Weiterleitungsregeln auf Ihrem Mail-Server. Lesen Sie hierzu bitte in der Dokumentation ihres Mail-Servers nach.

Via Kommandozeilen-Programm und z. B. procmail (otrs.PostMaster.pl)

OTRS ist in der Lage, E-Mails über ein Kommandozeilen-Programm (`bin/otrs.PostMaster.pl`) zu empfangen. Es nimmt die Mails über STDIN entgegen und fügt sie direkt in das OTRS-System ein. Das bedeutet, dass E-Mails im OTRS angezeigt werden, wenn der MDA (mail delivery agent, z. B. procmail) die E-Mails an `bin/otrs.PostMaster.pl` weiterleitet.

Um `bin/otrs.PostMaster.pl` ohne MDA zu testen, führen Sie die Kommandos in Skript 7.1 aus.

```
linux:/opt/otrs# cd bin
linux:/opt/otrs/bin# cat ../doc/sample_mails/test-email-1.box | ./
otrs.PostMaster.pl
linux:/opt/otrs/bin#
```

Skript 7.1. PostMaster-Test ohne MDA.

Wird die E-Mail in der Queue-Ansicht angezeigt, sind Ihre Einstellungen in Ordnung.

Procmail ist in der Linux-Umgebung ein sehr bekannter E-Mail-Filter, der höchstwahrscheinlich auf Ihrem System installiert sein wird. Falls nicht, erhalten Sie auf der *procmail Homepage* [<http://www.procmail.org/>] weitere Informationen.

Um procmail einzurichten (benötigt einen für procmail konfigurierten MDA (z. B. sendmail, postfix, exim oder qmail)), kann die Datei `.procmailrc.dist` aus dem OTRS-Homeverzeichnis verwendet werden. Kopieren Sie `.procmailrc.dist` nach `.procmailrc` und nehmen Sie die in Skript 7.2 dargestellten Änderungen vor.

```
SYS_HOME=$HOME
PATH=/bin:/usr/bin:/usr/local/bin
# --
# Pipe all email into the PostMaster process.
# --
:0 :
| $SYS_HOME/bin/otrs.PostMaster.pl
```

Skript 7.2. Konfiguration von procmail für OTRS.

Alle an den lokalen OTRS-Benutzer gesendeten E-Mails werden an `bin/otrs.PostMaster.pl` weitergeleitet und dadurch im Ticket-System gespeichert.

E-Mails via POP3 oder IMAP und fetchmail für otrs.PostMaster.pl empfangen

Um E-Mails von Ihrem Mailserver via POP3 oder IMAP für den OTRS-Rechner/lokalen OTRS-Benutzer und procmail abzuholen, benutzen Sie fetchmail [<http://fetchmail.berlios.de/>].

Anmerkung

Voraussetzung ist eine funktionierende SMTP-Konfiguration auf dem OTRS-Rechner.

Eine Beispielkonfiguration finden Sie in der Datei `.fetchmailrc.dist` im Homeverzeichnis von OTRS. Kopieren Sie diese Datei nach `.fetchmailrc` und erweitern Sie die Datei um die Daten Ihrer E-Mail-Accounts (siehe Beispiel 7-1).

Beispiel 7.1. `.fetchmailrc`

```
#poll (mailserver) protocol POP3 user (user) password (password) is
  (localuser)
poll mail.example.com protocol POP3 user joe password mama is otrs
```

Vergessen Sie nicht, die Zugriffsrechte von `.fetchmailrc` auf 710 zu setzen.

Wird das Kommando "**fetchmail -a** mit dieser Konfiguration ausgeführt, werden alle E-Mails auf das lokale OTRS-Konto weitergeleitet. Richten Sie ggf. einen Cronjob ein, wenn die E-Mails regelmäßig geholt werden sollen.

Filterung/Verteilung über PostMaster-Module (für komplexere Verteilungsszenarien)

Falls die `bin/otrs.PostMaster.pl` oder `bin/otrs.PostMasterMailbox.pl` Methoden verwendet werden, können X-OTRS-Header mit Hilfe der PostMaster-Filtermodule in die eingehenden Mails eingefügt bzw. bereits vorhandene X-OTRS-Header verändert werden. Mit Hilfe von X-OTRS-Headern kann das Ticket System bestimmte Aktionen für Mails ausführen, z. B. diese in eine bestimmte Queue einsortieren, sie einem bestimmten Kunden zuordnen, die Priorität ändern usw. Eine nähere Beschreibung der X-OTRS-Header finden Sie im Kapitel zum Einrichten von Mail-Accounts über den Administrations-Bereich von OTRS.

Es gibt verschiedene Standard-Filtermodule:

Anmerkung

Der Jobname (z. B. `$Self->{"PostMaster::PreFilterModule"}->{"Jobname"}`) muss eindeutig sein!

`Kernel::System::PostMaster::Filter::Match` ist ein Standard-Modul, um einige E-Mail-Header (z. B. From, To, Subject) zu prüfen und dann den neuen E-Mail-Header zu setzen (z. B. X-OTRS-Ignore: yes oder X-OTRS-Queue: spam).

Beispiel 7.2. Beispiel-Jobs für das Filtermodul `Kernel::System::PostMaster::Filter::Match`

```
# Job Name: 1-Match
# (block/ignore all spam email with From: noreply@)
$Self->{'PostMaster::PreFilterModule'}->{'1-Match'} = {
  Module => 'Kernel::System::PostMaster::Filter::Match',
  Match => {
    From => 'noreply@',
  },
  Set => {
    'X-OTRS-Ignore' => 'yes',
  },
}
```

```
};
# Job Name: 2-Match
# (sort emails with From: sales@example.com and Subject: **ORDER**
# into queue 'Order')
$self->{'PostMaster::PreFilterModule'}->{'2-Match'} = {
  Module => 'Kernel::System::PostMaster::Filter::Match',
  Match => {
    To => 'sales@example.com',
    Subject => '**ORDER**',
  },
  Set => {
    'X-OTRS-Queue' => 'Order',
  },
};
```

Kernel::System::PostMaster::Filter::CMD ist ein Standard-Modul, um die E-Mails an ein externes Kommando zu leiten. Dessen Ergebniss wird an STOUT ausgegeben und, falls das Ergebnis aus STDOUT true ist, wird ein neuer E-Mail-Header gesetzt (z. B. X-OTRS-Ignore: yes oder X-OTRS-Queue: spam).

Beispiel 7.3. Beispiel-Job für das Filtermodul Kernel::System::PostMaster::Filter::CMD

```
# Job Name: 5-SpamAssassin
# (SpamAssassin example setup, ignore spam emails)
$self->{'PostMaster::PreFilterModule'}->{'5-SpamAssassin'} = {
  Module => 'Kernel::System::PostMaster::Filter::CMD',
  CMD => '/usr/bin/spamassassin | grep -i "X-Spam-Status: yes"',
  Set => {
    'X-OTRS-Ignore' => 'yes',
  },
};
```

Natürlich ist es auch möglich, eigene PostMaster-Filtermodule zu entwickeln.

Kapitel 8. Zeitabhängige Funktionen in OTRS

Arbeitszeiten, Feiertage und Zeitzonen für das System festlegen

Einige Funktionen in OTRS, wie das automatische Entsperren von Tickets, setzen eine korrekte Konfiguration von Arbeitszeiten, Zeitzonen und Feiertagen voraus. Sie können diese in der SysConfig definieren, in Framework > Core::Time. Ebenso können Sie verschiedene Typen von Arbeitszeiten, Feiertagen und Zeitzonen als separate "Kalender" in Framework > Core::Time::Calendar1 bis Framework > Core::Time::Calendar9 definieren. Kalender können auf Queue- oder SLA-Ebene definiert werden. Das heißt, dass Sie z. B. einen Kalender mit 5x8 Arbeitsstunden pro Woche für Ihren "Standard"-SLA definieren können, zusätzlich jedoch auch einen mit 7x24 für Ihren "Gold"-SLA. Ebenso kann der Kalender für Ihre Queue "Support-USA" ein anderes Arbeitszeitfenster haben als ihre Queue "Support-Japan". OTRS kann bis zu 99 verschiedene Kalender verwalten.

Arbeitszeiten

Setzen Sie die Arbeitszeiten für Ihr System in der SysConfig fest, in Framework>Core::Time::TimeWorkingHours oder in der Konfiguration Ihrer Kalender. OTRS unterstützt eine Granularität von einer Stunde. Um z. B. 8:00 bis 18:00 als Arbeitszeit festzulegen, müssen die Auswahlkästchen 8, 9, 10 .. 17 aktiviert werden.

Nur während dieser Stunden können Tickets eskalieren, Benachrichtigungen zu Erinnerungs-Tickets versendet oder Tickets automatisch freigegeben werden. Weiterhin werden auch nur diese Stunden in die Berechnung der Eskalationszeit und der Zeit für die automatische Freigabe mit einbezogen.

Feiertage mit festem Datum

Feiertage deren Datum jedes Jahr gleich ist, können dem System in TimeVacationDays oder im entsprechenden Abschnitt in den Kalendern 1-9 bekannt gemacht werden.

Während der hier festgelegten Tage werden keine zeitabhängigen Aktionen oder Berechnungen auf Tickets im system ausgeführt.

Anmerkung

In OTRS sind die *deutschen* Feiertage voreingestellt.

TimeVacationDaysOneTime

Freie oder Feiertage, für die sich jährlich das Datum ändert, können in TimeVacationDaysOneTime angegeben werden.

Während der hier festgelegten Tage werden keine zeitabhängigen Aktionen oder Berechnungen auf Tickets im system ausgeführt.

Anmerkung

OTRS hat für diese Feiertage keine Voreinstellung. Sie müssen diese, wie z. B. Ostern, dem System also selbst bekannt machen.

Automatische Ticket-"Freigabe" (unlock)

Gesperrte Tickets können automatisch vom System freigegeben werden. Diese Funktion kann z. B. dann nützlich sein, wenn sich ein Agent im Urlaub befindet und noch Tickets gesperrt hat, die bearbeitet werden sollen / müssen. Dieses Feature entsperrt Tickets nach einer bestimmten Zeit, um sicherzustellen, dass gesperrte Tickets nicht vergessen werden und andere Agenten darauf zugreifen können.

Die Zeit, nach der gesperrte Tickets automatisch freigegeben werden, kann in den Einstellungen jeder Queue festgelegt werden. Mit Hilfe des Moduls `bin/otrs.UnlockTickets.pl`, das als cron-Job regelmäßig ausgeführt werden sollte, wird die automatische Freigabe von Tickets umgesetzt.

Benachrichtigungen für entsperrte Tickets werden nur zu den Benutzern versendet, die die jeweilige Queue in "Meine Queues" eingetragen und die Benachrichtigung für entsperrte Tickets in ihren persönlichen Einstellungen aktiviert haben.

Tickets werden entsperrt wenn folgende Bedingungen zutreffen:

- In der Queue des Tickets ist ein *Freigabe-Zeitintervall* definiert.
- Das Ticket ist *gesperrt*.
- Der Status des Tickets ist *offen*".

Die Freigabezeit wird zurückgesetzt, wenn ein Agent einen neuen externen Artikel zum Ticket hinzufügt. Das kann einer der folgenden Artikeltypen sein: *email-external*, *phone*, *fax*, *sms*, or *note-external*.

Die Freigabezeit wird auch zurückgesetzt, wenn der letzte Artikel von einem Agent stammt und dann der Kunde einen hinzufügt, per E-Mail oder Weboberfläche.

Zuletzt wird die Freigabezeit auch dann zurückgesetzt, wenn das Ticket einem anderen Agenten zugewiesen wird.

Kapitel 9. Ticket-Verantwortlicher und Beobachten von Tickets

Seit OTRS 2.1 kann zusätzlich zum Besitzer ein Verantwortlicher für das Ticket festgelegt werden. Zudem können alle Aktivitäten rund um das Ticket beobachtet werden. Diese beiden Funktionalitäten sind mit den TicketResponsible- und dem TicketWatcher-Funktionen implementiert, und erleichtern das Zuweisen von Aufgaben und die Arbeit in hierarchischen Teamstrukturen.

Ticket-Verantwortlicher

Die Funktion "Verantwortlicher" eines Tickets macht es möglich, das Ticket einem anderen als dem aktuellen Besitzer zur Bearbeitung zu übergeben. Ein Benutzer, der ein Ticket gesperrt hat, kann es einem anderen Agenten, der nicht der Ticketbesitzer ist, zur Beantwortung übergeben. Nach der Bearbeitung kann der erste Benutzer die Ticketverantwortung vom zweiten wieder entfernen.

Mit dem Konfigurationsparameter `Ticket::Responsible` kann die Funktion `TicketResponsible` aktiviert werden. Dadurch werden drei neue Icons in ihrer Toolbar erscheinen.

Die Ticketverantwortung kann übergeben werden, indem das Ticket aufgerufen und die Funktion "Verantwortlicher" in der Aktionsleiste genutzt wird.

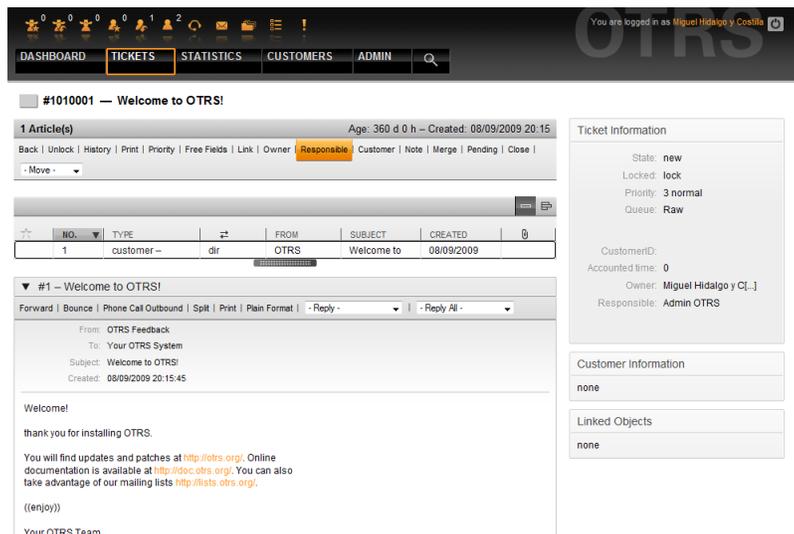


Abb. 9.1. Verantwortlichen eines Tickets ändern.

Nach dem Klick auf "Verantwortlicher" öffnet sich ein Popup-Fenster, in dem die Änderung durchgeführt werden kann. Damit kann auch dem neuen verantwortlichen Benutzer eine Nachricht geschickt werden.

Ticket-Verantwortlicher und Beobachten von Tickets

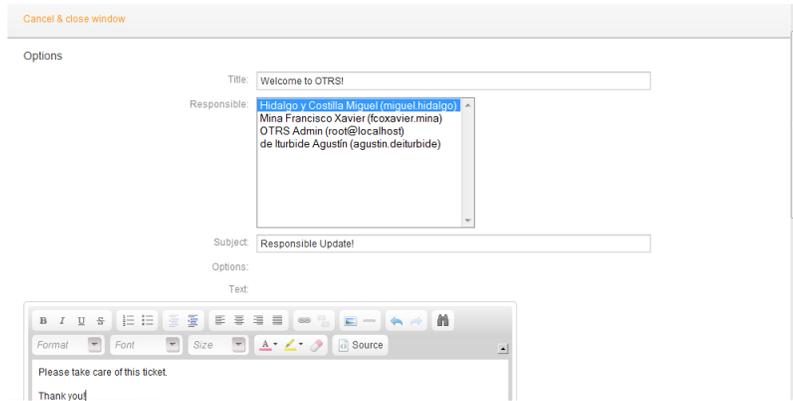


Abb. 9.2. *Popup-Dialog zum Ändern des Ticket-Verantwortlichen.*

Die Liste aller Tickets, für die ein Benutzer verantwortlich ist, kann durch die Ansicht "Verantwortliche Tickets" eingesehen werden. Diese kann über 3 Icons in der Toolbar aufgerufen werden, sobald das TicketResponsible-Feature aktiviert wird.

Tickets beobachten

Sollen innerhalb des Ticket-Systems lediglich bestimmte Tickets z. B. von einem Abteilungsleiter beobachtet und nicht selbst bearbeitet werden, so kann dies seit OTRS 2.1 mit Hilfe des TicketWatcher-Features realisiert werden.

Das TicketWatcher-Feature kann mit Hilfe des Konfigurations-Parameters Ticket::Watcher aktiviert werden. Weiterhin können über Ticket::WatcherGroup eine oder mehrere Benutzergruppen festgelegt werden, die die Erlaubnis haben sollen Tickets zu beobachten.

Um ein Ticket zu beobachten, muss der Inhalt des Tickets aufgerufen und im Menü für die verschiedenen Aktionen für das Ticket der "Beobachten"-Schalter betätigt werden.

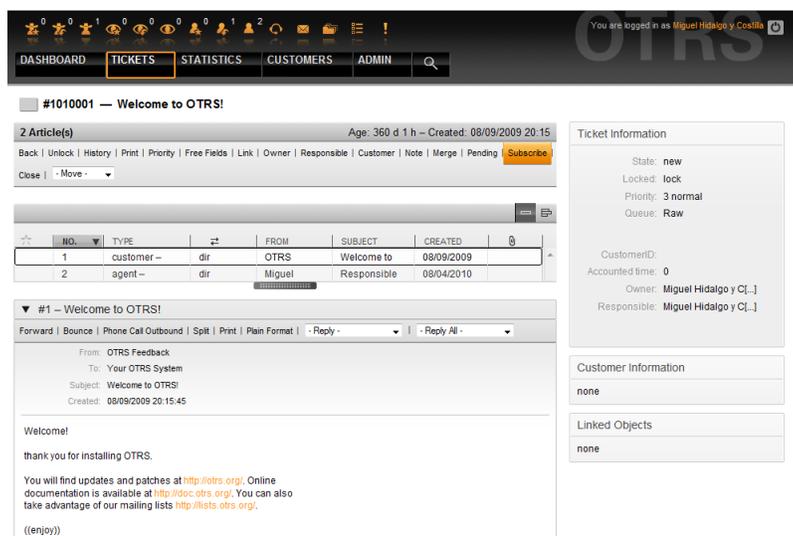


Abb. 9.3. *Ticket beobachten.*

Die Beobachtung eines Tickets wird aufgehoben, indem in der Inhaltsanzeige eines Tickets im Menü für die möglichen Aktionen der "Nicht beobachten"-Schalter gedrückt wird.

Ticket-Verantwortlicher und Beobachten von Tickets

#1010001 — Welcome to OTRS!

2 Article(s) Age: 360 d 2 h — Created: 08/09/2009 20:15
 Back | Unlock | History | Print | Priority | Free Fields | Link | Owner | Responsible | Customer | Note | Merge | Pending |

Unsubscribe Close | - Move -

NO.	TYPE	FROM	SUBJECT	CREATED
1	customer —	dir	OTRS	Welcome to
2	agent —	dir	Miguel	Responsible

▼ #1 — Welcome to OTRS!

Forward | Bounce | Phone Call Outbound | Split | Print | Plain Format | - Reply - | - Reply All -

From: OTRS Feedback
 To: Your OTRS System
 Subject: Welcome to OTRS!
 Created: 08/09/2009 20:15:45

Welcome!

thank you for installing OTRS.

You will find updates and patches at <http://otrs.org/>. Online documentation is available at <http://doc.otrs.org/>. You can also take advantage of our mailing lists <http://lists.otrs.org/>.

((enjoy))

Ticket Information
 State: new
 Locked: lock
 Priority: 3 normal
 Queue: Raw

CustomerID:
 Accounted time: 0
 Owner: Miguel Hidalgo y Costi...
 Responsible: Miguel Hidalgo y Costi...

Customer Information
 none

Linked Objects
 none

Abb. 9.4. Ticket nicht mehr beobachten.

Die Liste aller zur Beobachtung ausgewählten Tickets wird über die Ansicht "Beobachtete Tickets" erreicht. Diese kann über 3 Icons in der Toolbar aufgerufen werden, sobald das TicketWatcher-Feature aktiviert wird.

My Watched Tickets: All

All 1 | New Article 0 | Pending 0 | Reminder Reached 0

Bulk

TICKET#	AGE	FROM / SUBJECT	STATE	LOCKED	QUEUE	OWNER	CUSTOMERID
1010001	367 d 6 h	OTRS Feedback: Welcome to OTRS!	new	lock	Raw	Miguel Hidalgo y Costi...	

1-1 of 1 - Page: 1 | S M L

Powered by OTRS 3.0.x CVS Top of page

Abb. 9.5. Ansicht "Beobachtete Tickets".

Kapitel 10. Einrichtung der PDF-Ausgabe

Dieser Abschnitt behandelt die Einstellungen für die PDF-Druckausgabe von OTRS.

Wenn sie die Aktion "Drucken" in OTRS verwenden, wird eine formatierte PDF-Datei erzeugt. Sie können dies mit dem Konfigurationsparameter `PDF` deaktivieren. In diesem Fall wird stattdessen eine HTML-Ausgabe erzeugt.

Sie können das Erscheinungsbild der generierten Dateien anpassen, indem Sie ihr eigenes Logo in `PDF::LogoFile` hinterlegen. Mit `PDF::PageSize` lässt sich die Standard-Seitengröße der PDF-Datei einstellen. Schließlich kann mit `PDF::MaxPages` die maximale Anzahl an Seiten festgelegt werden, die eine PDF-Datei höchstens beinhalten soll. Das ist nützlich, um versehentliches Drucken überlanger Dokumente zu verhindern.

Für die Generierung von PDF-Dateien müssen die CPAN-Module `PDF::API2` und `Compress::Zlib` installiert werden. Bei vielen Distributionen sind diese als Paket verfügbar und können bequem über den jeweiligen Paketmanager installiert werden, falls dies jedoch nicht möglich ist muss der direkte Weg über CPAN gewählt werden. Das Kapitel Installation der für OTRS benötigten Perl-Module beschreibt, wie die Installation der Perl-Module durchzuführen ist.

Kapitel 11. Einbinden externer Backends

Kundenbenutzerdaten

OTRS ist in der Lage, mit verschiedenen Kundendaten (insbesondere Login, E-Mail, Telefon) umzugehen. Diese Informationen können im Agenten-Interface angezeigt und für das Kunden-Interface verwendet werden. Weiterhin werden die Daten für die Authentifizierung der Kunden am System benötigt.

Die benutzten/angezeigten Kundendaten sind frei konfigurierbar, es gibt jedoch folgende benötigte Optionen die unbedingt vorhanden sein müssen, damit OTRS ordnungsgemäß funktioniert:

- Benutzer-Login
- E-Mail-Adresse
- Benutzer-Kunden-ID

Verwenden Sie die Konfigurationsparameter aus Skript 11.1 in Ihrer Datei `Kernel/Config.pm`, wenn Sie Kundeninformationen anzeigen möchten.

```
# Ticket::Frontend::CustomerInfo*
# (show customer info on Compose (Phone and Email), Zoom and
# Queue view)
$self->{'Ticket::Frontend::CustomerInfoCompose'} = 1;
$self->{'Ticket::Frontend::CustomerInfoZoom'} = 1;
$self->{'Ticket::Frontend::CustomerInfoQueue'} = 0;
```

Skript 11.1. Konfigurationsparameter in `Kernel/Config.pm`.

Kundenbenutzer Backend

Es existieren zwei Kundenbenutzer Back-ends, DB und LDAP. Falls Sie bereits ein Kundenverzeichnis (z. B. SAP, ...) haben, ist es natürlich möglich, dafür ein eigenes Back-end zu schreiben.

Datenbank (Standard)

Beispiel 11-1 zeigt die Konfiguration eines Kundenbenutzer-Backends, welches die Kundendaten in der OTRS-Datenbank speichert.

Beispiel 11.1. Konfiguration eines Datenbank-Kunden-Backends

```
# CustomerUser (customer database backend and settings)
$self->{'CustomerUser'} = {
    Name => 'Database Datasource',
    Module => 'Kernel::System::CustomerUser::DB',
    Params => {
```

```

        # if you want to use an external database, add the required
settings
#       DSN => 'DBI:odbc:yourdsn',
#       DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
#       User => '',
#       Password => '',
        Table => 'customer_user',
    },
# customer unique id
CustomerKey => 'login',
# customer #
CustomerID => 'customer_id',
CustomerValid => 'valid_id',
    CustomerUserListFields => ['first_name', 'last_name', 'email'],
    CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
    CustomerUserSearchPrefix => '',
    CustomerUserSearchSuffix => '*',
    CustomerUserSearchListLimit => 250,
    CustomerUserPostMasterSearchFields => ['email'],
    CustomerUserNameFields => ['title', 'first_name', 'last_name'],
    CustomerUserEmailUniqCheck => 1,
#   # show not own tickets in customer panel, CompanyTickets
#   CustomerUserExcludePrimaryCustomerID => 0,
#   # generate auto logins
#   AutoLoginCreation => 0,
#   AutoLoginCreationPrefix => 'auto',
#   # admin can change customer preferences
#   AdminSetPreferences => 1,
#   # cache time to live in sec. - cache any database queries
#   CacheTTL => 0,
#   # just a read only source
#   ReadOnly => 1,
    Map => [
        # note: Login, Email and CustomerID needed!
        # var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly, http-link-target
        [ 'UserTitle',      'Title',      'title',      1, 0, 'var',
'', 0 ],
        [ 'UserFirstname', 'Firstname', 'first_name', 1, 1, 'var',
'', 0 ],
        [ 'UserLastname',  'Lastname',  'last_name',  1, 1, 'var',
'', 0 ],
        [ 'UserLogin',     'Username',  'login',     1, 1, 'var',
'', 0 ],
        [ 'UserPassword',  'Password',  'pw',        0, 0, 'var',
'', 0 ],
        [ 'UserEmail',     'Email',     'email',     1, 1, 'var',
'', 0 ],

#       [ 'UserEmail',      'Email', 'email',      1, 1, 'var',
'$Env{"CGIHandle"}?Action=AgentTicketCompose&ResponseID=1&TicketID=
$Data{"TicketID"}&ArticleID=$Data{"ArticleID"}', 0 ],
        [ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var',
'', 0 ],

```

```
#      [ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0,
'var', '', 0 ],
      [ 'UserPhone',      'Phone',      'phone',      1, 0,
'var', '', 0 ],
      [ 'UserFax',        'Fax',        'fax',        1, 0,
'var', '', 0 ],
      [ 'UserMobile',     'Mobile',     'mobile',     1, 0,
'var', '', 0 ],
      [ 'UserStreet',     'Street',     'street',     1, 0,
'var', '', 0 ],
      [ 'UserZip',        'Zip',        'zip',        1, 0,
'var', '', 0 ],
      [ 'UserCity',       'City',       'city',       1, 0,
'var', '', 0 ],
      [ 'UserCountry',    'Country',    'country',    1, 0,
'var', '', 0 ],
      [ 'UserComment',    'Comment',    'comments',   1, 0,
'var', '', 0 ],
      [ 'ValidID',        'Valid',      'valid_id',   0, 1,
'int', '', 0 ],
    ],
    # default selections
    Selections => {
      UserTitle => {
        'Mr.' => 'Mr.',
        'Mrs.' => 'Mrs.',
      },
    },
  };
```

Falls Sie die Kundendaten anpassen möchten, ändern Sie in der Datenbank die Tabellenspalten oder fügen Sie weitere hinzu (im folgenden Beispiel wird ein Feld für die Raumnummer hinzugefügt).

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 116 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD room VARCHAR (250);
Query OK, 1 rows affected (0.01 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> quit
Bye
linux:~#
```

Skript 11.1. Ein Feld zur Tabelle customer_user hinzufügen.

Danach fügen Sie Ihre eigenen Spalten dem MAP Array in der Datei Kernel/Config.pm hinzu:

```
# var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly
[... ]
[ 'UserRoom',      'Room',      'room',      0, 1, 'var', '',
0 ],
```

Skript 11.3. Hinzufügen eines Feldes in Kernel/Config.pm file.

Natürlich können Sie all diese Kundeninformationen dann auch über das Admin-Interface bzw. die Kundenverwaltung pflegen.

Kunden mit multiplen IDs (Firmen-Tickets)

Es ist möglich, einem Kunden mehr als nur eine Kundennummer zuzuweisen. Dies kann z. B. dann sinnvoll sein, wenn ein Kunde auf Tickets anderer Kunden zugreifen muss, z. B. der Abteilungsleiter auf die Tickets der Mitarbeiter seiner Abteilung. Hat ein Kunde Zugriff auf Tickets anderer Kunden, verwendet man in OTRS das sog. Firmen Ticket Feature. Im Kunden-Interface können diese Tickets über den "Firmen-Ticket" Link eingesehen werden.

Um Firmen-Tickets zu verwenden, muss die customer_user Tabelle in der OTRS Datenbank um eine Spalte erweitert werden, in die später die Kundennummern eingetragen werden, auf die ein Kunde zusätzlich zu den eigenen Tickets Zugriff haben soll.

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 124 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD customer_ids VARCHAR (250);
Query OK, 1 rows affected (0.02 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> quit
Bye
linux:~#
```

Skript 11.4. Hinzufügen eines Feldes customer_ids zur Tabelle customer_user.

Danach fügen Sie die neue Spalte dem MAP Array in der Datei Kernel/Config.pm hinzu.

```
# var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly
```

```
[...]
[ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var',
'', 0 ],
```

Skript 11.5. Hinzufügen des Feldes UserCustomerIDs zu Kernel/Config.pm.

Die Spalte für die Multi-Kundennummern kann ab nun über das Admin-Interface bzw. über die Kundenverwaltung gepflegt werden.

Um nun den Zugriff für einen Kunden auf die Tickets anderer Kunden zu ermöglichen, tragen Sie in die neue Spalte die IDs der Kunden ein, auf deren Tickets der Zugriff ermöglicht werden soll. Die einzelnen IDs trennen Sie durch ein Semikolon.

Beispiel 11.2. Firmen-Tickets mit einem Datenbank-Backend

Angenommen es sind die Kunden A, B und C im System angelegt. A soll mit Hilfe von Firmen Tickets über das Kunden-Interface Zugriff auf die Tickets von B und C haben, B und C sollen jedoch jeweils nur ihre eigenen Tickets einsehen und bearbeiten können.

Um dieses Setup zu realisieren, ändern Sie wie oben beschrieben die customer_user Tabelle in der OTRS Datenbank und das Mapping in Kernel/Config.pm. Anschließend laden Sie über die Kundenverwaltung die Einstellungen des Kunden A und tragen bei "Kundennummern" die Werte "B;C;" ein.

LDAP

Falls Sie ein existierendes LDAP Verzeichnis mit Ihren Kundenbenutzern haben, können Sie dieses auch mit OTRS nutzen.

Beispiel 11.3. Konfiguration eines LDAP Kunden-Backends

```
# CustomerUser
# (customer ldap backend and settings)
$self->{CustomerUser} = {
    Name => 'LDAP Data Source',
    Module => 'Kernel::System::CustomerUser::LDAP',
    Params => {
        # ldap host
        Host => 'bay.csuhayward.edu',
        # ldap base dn
        BaseDN => 'ou=seas,o=csuh',
        # search scope (one|sub)
        SSCOPE => 'sub',
        # The following is valid but would only be necessary if the
        # anonymous user does NOT have permission to read from the
LDAP tree
        UserDN => '',
        UserPw => '',
        # in case you want to add always one filter to each ldap
query, use
        # this option. e. g. AlwaysFilter => '(mail=*)' or
AlwaysFilter => '(objectclass=user)'
        AlwaysFilter => '',
```

```

        # if your frontend is e. g. iso-8859-1 and the charset of your
        # ldap server is utf-8, use these options.
#       SourceCharset => 'utf-8',
#       DestCharset  => 'iso-8859-1',
        # if both your frontend and your LDAP are unicode, use
this:
#       SourceCharset => 'utf-8',
#       DestCharset  => 'utf-8',
        # Net::LDAP new params (if needed - for more info see
perldoc Net::LDAP)
        Params => {
            port => 389,
            timeout => 120,
            async => 0,
            version => 3,
        },
    },
    # customer unique id
    CustomerKey => 'uid',
    # customer #
    CustomerID => 'mail',
    CustomerUserListFields => ['cn', 'mail'],
    CustomerUserSearchFields => ['uid', 'cn', 'mail'],
    CustomerUserSearchPrefix => '',
    CustomerUserSearchSuffix => '*',
    CustomerUserSearchListLimit => 250,
    CustomerUserPostMasterSearchFields => ['mail'],
    CustomerUserNameFields => ['givenname', 'sn'],
    # show not own tickets in customer panel, CompanyTickets
    CustomerUserExcludePrimaryCustomerID => 0,
    # add an ldap filter for valid users (expert setting)
#   CustomerUserValidFilter => '(!((description=locked)))',
    # administrator can't change customer preferences
    AdminSetPreferences => 0,
#   # cache time to live in sec. - cache any database queries
#   CacheTTL => 0,
    Map => [
        # note: Login, Email and CustomerID are mandatory!
        # var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly
        [ 'UserTitle',      'Title',      'title',      1, 0,
'var', '', 0 ],
        [ 'UserFirstname',  'Firstname',  'givenname',  1, 1,
'var', '', 0 ],
        [ 'UserLastname',  'Lastname',  'sn',        1, 1,
'var', '', 0 ],
        [ 'UserLogin',     'Username',  'uid',       1, 1,
'var', '', 0 ],
        [ 'UserEmail',     'Email',    'mail',      1, 1,
'var', '', 0 ],
        [ 'UserCustomerID', 'CustomerID', 'mail',      0, 1,
'var', '', 0 ],
#       [ 'UserCustomerIDs', 'CustomerIDs', 'second_customer_ids', 1,
0, 'var', '', 0 ],

```

```

        [ 'UserPhone',      'Phone',      'telephonenumber', 1, 0,
'var', '', 0 ],
        [ 'UserAddress',   'Address',   'postaladdress', 1, 0,
'var', '', 0 ],
        [ 'UserComment',   'Comment',   'description', 1, 0,
'var', '', 0 ],
    ],
};

```

Falls Sie in Ihrem LDAP Verzeichnis weitere Informationen zu Ihren Kunden gespeichert haben und mit OTRS darauf zugreifen möchten, erweitern Sie das MAP Array in `Kernel/Config.pm` bzw. entfernen nicht gewünschte Einträge.

```

# var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly
[...]
[ 'UserPhone',      'Phone',      'telephonenumber', 1, 0, 'var',
'', 0 ],

```

Skript 11.6. Hinzufügen neuer Felder zu Kernel/Config.pm.

Kunden mit multiplen IDs (Firmen-Tickets)

Um Firmentickets mit einem LDAP zu nutzen, muss ein Feld zum LDAP-Verzeichnis hinzugefügt werden, welches die IDs enthält, die für einen Kunden zugänglich sein sollen.

Wenn das neue Feld erstellt wurde, muss es zum MAP Array in `Kernel/Config.pm` hinzugefügt werden.

```

# var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly
[...]
[ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var',
'', 0 ],

```

Skript 11.7. Hinzufügen eines neuen Feldes in Kernel/Config.pm.

Das Feld muss direkt im LDAP-Verzeichnis bearbeitet werden, weil OTRS von LDAP nur lesen kann, nicht jedoch darauf schreiben.

Um nun den Zugriff für einen Kunden auf die Tickets anderer Kunden zu ermöglichen, tragen Sie in das neue LDAP-Feld die IDs der Kunden ein, auf deren Tickets der Zugriff ermöglicht werden soll. Die einzelnen IDs trennen Sie durch ein Semikolon.

Beispiel 11.4. Verwendung von Firmentickets mit einem LDAP-Backend

Angenommen es sind die Kunden A, B und C im System angelegt. A soll mit Hilfe von Firmen Tickets über das Kunden-Interface Zugriff auf die Tickets von B und C haben, B und C sollen jedoch jeweils nur ihre eigenen Tickets einsehen und bearbeiten können.

Um dieses Setup zu realisieren, ändern Sie wie oben beschrieben das LDAP Verzeichnis und das Mapping in `Kernel/Config.pm`. Anschließend tragen Sie im LDAP Verzeichnis innerhalb der Einstellungen für den Kunden A für CustomerIDs die Werte "B;C;" ein.

Verwenden mehrerer Kunden Backends

Soll mehr als nur ein Back-end mit verschiedenen Kundendaten verwendet werden (z. B. gleichzeitig DB und LDAP), so ist dies ebenfalls mit OTRS möglich. In einem solchen Fall muss der CustomerUser Parameter für jedes Back-end um eine Nummer erweitert werden, z. B. "CustomerUser1", "CustomerUser2", usw.

Beispiel 11.5. Gleichzeitige Einbindung mehrerer verschiedener Kunden Backends

In der folgenden KOnfiguration verwendet OTRS gleichzeitig ein Datenbank- und ein LDAP-Kunden-Backend.

```
# 1. Customer user backend: DB
# (customer database backend and settings)
$self->{CustomerUser1} = {
    Name => 'Customer Database',
    Module => 'Kernel::System::CustomerUser::DB',
    Params => {
        # if you want to use an external database, add the
        # required settings
        # DSN => 'DBI:odbc:yourdsn',
        # DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
        # User => '',
        # Password => '',
        Table => 'customer_user',
    },
    # customer unique id
    CustomerKey = 'login',
    # customer #
    CustomerID = 'customer_id',
    CustomerValid = 'valid_id',
    CustomerUserListFields => ['first_name', 'last_name', 'email'],
    CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
    CustomerUserSearchPrefix => '',
    CustomerUserSearchSuffix => '*',
    CustomerUserSearchListLimit => 250,
    CustomerUserPostMasterSearchFields => ['email'],
    CustomerUserNameFields => ['title', 'first_name', 'last_name'],
    CustomerUserEmailUniqCheck => 1,
    # # show not own tickets in customer panel, CompanyTickets
    # CustomerUserExcludePrimaryCustomerID => 0,
    # # generate auto logins
    # AutoLoginCreation => 0,
    # AutoLoginCreationPrefix => 'auto',
    # # admin can change customer preferences
    # AdminSetPreferences => 1,
    # # cache time to live in sec. - cache any database queries
    # CacheTTL => 0,
    # # just a read only source
    # ReadOnly => 1,
    Map => [
```

```

        # note: Login, Email and CustomerID needed!
        # var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly, http-link-target
    [ 'UserTitle',      'Title',      'title',      1, 0, 'var',
'', 0 ],
    [ 'UserFirstname', 'Firstname', 'first_name', 1, 1, 'var',
'', 0 ],
    [ 'UserLastname',  'Lastname',  'last_name',  1, 1, 'var',
'', 0 ],
    [ 'UserLogin',     'Username',  'login',     1, 1, 'var',
'', 0 ],
    [ 'UserPassword',  'Password',  'pw',        0, 0, 'var',
'', 0 ],
    [ 'UserEmail',     'Email',     'email',     1, 1, 'var',
'', 0 ],
    [ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var',
'', 0 ],
    [ 'UserPhone',     'Phone',     'phone',     1, 0, 'var',
'', 0 ],
    [ 'UserFax',       'Fax',       'fax',       1, 0, 'var',
'', 0 ],
    [ 'UserMobile',    'Mobile',    'mobile',    1, 0, 'var',
'', 0 ],
    [ 'UserStreet',    'Street',    'street',    1, 0, 'var',
'', 0 ],
    [ 'UserZip',       'Zip',       'zip',       1, 0, 'var',
'', 0 ],
    [ 'UserCity',      'City',      'city',      1, 0, 'var',
'', 0 ],
    [ 'UserCountry',   'Country',   'country',   1, 0, 'var',
'', 0 ],
    [ 'UserComment',   'Comment',   'comments',  1, 0, 'var',
'', 0 ],
    [ 'ValidID',       'Valid',     'valid_id',  0, 1, 'int',
'', 0 ],
    ],
    # default selections
    Selections => {
        UserTitle => {
            'Mr.' => 'Mr.',
            'Mrs.' => 'Mrs.',
        },
    },
};

```

```

# 2. Customer user backend: LDAP
# (customer ldap backend and settings)
$self->{CustomerUser2} = {
    Name => 'LDAP Datasource',
    Module => 'Kernel::System::CustomerUser::LDAP',
    Params => {
        # ldap host
        Host => 'bay.csuhayward.edu',
        # ldap base dn

```

```

BaseDN => 'ou=seas,o=csuh',
# search scope (one|sub)
SSCOPE => 'sub',
# # The following is valid but would only be necessary if the
# # anonymous user does NOT have permission to read from the
LDAP tree
  UserDN => '',
  UserPw => '',
  # in case you want to add always one filter to each ldap
query, use
  # this option. e. g. AlwaysFilter => '(mail=*)' or
AlwaysFilter => '(objectclass=user)'
  AlwaysFilter => '',
  # if both your frontend and your LDAP are unicode, use this:
#   SourceCharset => 'utf-8',
#   DestCharset   => 'utf-8',
  # if your frontend is e. g. iso-8859-1 and the character set
of your
  # ldap server is utf-8, use these options:
#   SourceCharset => 'utf-8',
#   DestCharset => 'iso-8859-1',

# Net::LDAP new params (if needed - for more info see perldoc
Net::LDAP)
  Params => {
    port => 389,
    timeout => 120,
    async => 0,
    version => 3,
  },
},
# customer unique id
CustomerKey => 'uid',
# customer #
CustomerID => 'mail',
CustomerUserListFields => ['cn', 'mail'],
CustomerUserSearchFields => ['uid', 'cn', 'mail'],
CustomerUserSearchPrefix => '',
CustomerUserSearchSuffix => '*',
CustomerUserSearchListLimit => 250,
CustomerUserPostMasterSearchFields => ['mail'],
CustomerUserNameFields => ['givenname', 'sn'],
# show not own tickets in customer panel, CompanyTickets
CustomerUserExcludePrimaryCustomerID => 0,
# add a ldap filter for valid users (expert setting)
# CustomerUserValidFilter => '(!(description=locked))',
# admin can't change customer preferences
AdminSetPreferences => 0,
Map => [
  # note: Login, Email and CustomerID needed!
  # var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly
  [ 'UserTitle',      'Title',      'title',          1, 0,
'var', '', 0 ],

```

```

        [ 'UserFirstname', 'Firstname', 'givenname',      1, 1,
'var', '', 0 ],
        [ 'UserLastname', 'Lastname', 'sn',              1, 1,
'var', '', 0 ],
        [ 'UserLogin',    'Username', 'uid',              1, 1,
'var', '', 0 ],
        [ 'UserEmail',    'Email',    'mail',             1, 1,
'var', '', 0 ],
        [ 'UserCustomerID', 'CustomerID', 'mail',         0, 1,
'var', '', 0 ],
#        [ 'UserCustomerIDs', 'CustomerIDs', 'second_customer_ids', 1,
0, 'var', '', 0 ],
        [ 'UserPhone',    'Phone',    'telephonenumber', 1, 0,
'var', '', 0 ],
        [ 'UserAddress',  'Address',  'postaladdress',   1, 0,
'var', '', 0 ],
        [ 'UserComment',  'Comment',  'description',    1, 0,
'var', '', 0 ],
    ],
};

```

Es können bis zu 10 Kunden Back-ends gleichzeitig eingebunden werden. Über die Kundenverwaltung in OTRS ist der Zugriff auf die verschiedenen Back-ends möglich.

Backends für die Authentifizierung von Agenten und Kunden

OTRS bietet die Möglichkeit Agenten und Kunden über verschiedene Back-ends zu authentifizieren.

Authentifizierungs-Backends für Agenten

Datenbank (Standard)

Das Backend für die Authentifizierung von Agenten, welches OTRS standardmäßig verwendet, ist die OTRS-Datenbank. Die Agenten können innerhalb des Admin-Bereiches in der Benutzerverwaltung angelegt und bearbeitet werden.

Beispiel 11.6. Agentenauthentifizierung gegen ein Datenbank-Backend

```
$Self->{'AuthModule'} = 'Kernel::System::Auth::DB';
```

LDAP

Falls ein LDAP Verzeichnis mit Ihren Agenten-Benutzerdaten verfügbar ist, können Sie das LDAP Modul für die Authentifizierung Ihrer Agenten nutzen. Dieses Modul greift nur lesend auf die Daten im LDAP Verzeichnis zu, d.h. die Daten können nicht mit OTRS bearbeitet werden, es können also keine Agenten mit Hilfe der Benutzerverwaltung von OTRS angelegt oder bearbeitet werden.

Beispiel 11.7. Agentenauthentifizierung gegen ein LDAP Backend

```

# This is an example configuration for an LDAP auth. backend.
# (Make sure Net::LDAP is installed!)
$Self->{'AuthModule'} = 'Kernel::System::Auth::LDAP';
$Self->{'AuthModule::LDAP::Host'} = 'ldap.example.com';
$Self->{'AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';
$Self->{'AuthModule::LDAP::UID'} = 'uid';

# Check if the user is allowed to auth in a posixGroup
# (e. g. user needs to be in a group xyz to use otrs)
$Self->{'AuthModule::LDAP::GroupDN'} =
    'cn=otrsallow,ou=posixGroups,dc=example,dc=com';
$Self->{'AuthModule::LDAP::AccessAttr'} = 'memberUid';
# for ldap posixGroups objectclass (just uid)
# $Self->{'AuthModule::LDAP::UserAttr'} = 'UID';
# for non ldap posixGroups objectclass (with full user dn)
# $Self->{'AuthModule::LDAP::UserAttr'} = 'DN';

# The following is valid but would only be necessary if the
# anonymous user do NOT have permission to read from the LDAP tree
$Self->{'AuthModule::LDAP::SearchUserDN'} = '';
$Self->{'AuthModule::LDAP::SearchUserPw'} = '';

# in case you want to add always one filter to each ldap query, use
# this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter =>
# '(objectclass=user)'
$Self->{'AuthModule::LDAP::AlwaysFilter'} = '';

# in case you want to add a suffix to each login name, then
# you can use this option. e. g. user just want to use user but
# in your ldap directory exists user@domain.
# $Self->{'AuthModule::LDAP::UserSuffix'} = '@domain.com';

# Net::LDAP new params (if needed - for more info see perldoc
# Net::LDAP)
$Self->{'AuthModule::LDAP::Params'} = {
    port => 389,
    timeout => 120,
    async => 0,
    version => 3,
};

Mit den folgenden Konfigurationsparametern können die Benutzerdaten der Agenten aus dem LDAP
in die lokale OTRS Datenbank synchronisiert werden. Dies reduziert die Zugriffe auf ihr LDAP
Verzeichnis, entlastet den Server mit den LDAP Daten und beschleunigt die Anmeldung an OTRS. Die
Synchronisierung der Daten findet bei der ersten Anmeldung des Agenten statt, trotz der synchronisierten
Daten bleibt ihr LDAP Verzeichnis die letzte Instanz bei der Anmeldung. D.h. wird ein User im LDAP
Verzeichnis gelöscht oder deaktiviert, klappt die Anmeldung an OTRS nicht. Ebenfalls müssen die Daten
für einen Agenten weiterhin direkt im LDAP Verzeichnis gepflegt werden.

# agent data sync against ldap
$Self->{'AuthSyncModule'} = 'Kernel::System::Auth::Sync::LDAP';
$Self->{'AuthSyncModule::LDAP::Host'} = 'ldap://ldap.example.com/';
$Self->{'AuthSyncModule::LDAP::BaseDN'} = 'dc=otrs, dc=org';

```

```

$Self->{'AuthSyncModule::LDAP::UID'} = 'uid';
$Self->{'AuthSyncModule::LDAP::SearchUserDN'} = 'uid=sys, ou=user,
  dc=otrs, dc=org';
$Self->{'AuthSyncModule::LDAP::SearchUserPw'} = 'some_pass';
$Self->{'AuthSyncModule::LDAP::UserSyncMap'} = {
  # DB -> LDAP
  UserFirstname => 'givenName',
  UserLastname  => 'sn',
  UserEmail     => 'mail',
};
[...]

# AuthSyncModule::LDAP::UserSyncInitialGroups
# (sync following group with rw permission after initial create of
# first agent
# login)
$Self->{'AuthSyncModule::LDAP::UserSyncInitialGroups'} = [
  'users',
];

```

Skript 11.8. Synchronisation der Benutzerdaten aus dem LDAP-Verzeichnis in die OTRS-Datenbank.

HTTPBasicAuth für Agenten

Falls Sie eine "single sign on"-Lösung für Ihre Agenten implementieren möchten, benutzen Sie http basic authentication (für alle Ihre Systeme) und aktivieren Sie das HTTPBasicAuth Modul.

Beispiel 11.8. Agentenauthentifizierung über HTTPBasic

```

# This is an example configuration for an apache ($ENV{REMOTE_USER})
# auth. backend. Use it if you want to have a single login through
# apache http-basic-auth
$Self->{'AuthModule'} = 'Kernel::System::Auth::HTTPBasicAuth';

# Note:
#
# If you use this module, you should use as fallback
# the following configuration settings if the user is not authorized
# apache ($ENV{REMOTE_USER})
$Self->{LoginURL} = 'http://host.example.com/not-authorized-for-
otrs.html';
$Self->{LogoutURL} = 'http://host.example.com/thanks-for-using-
otrs.html';

```

Radius

Mit den folgenden Einstellungen kann die Authentifizierung von Agenten gegen einen Radius-Server realisiert werden.

Beispiel 11.9. Agentenauthentifizierung gegen ein Radius-Backend

```
# This is example configuration to auth. agents against a radius
server
$Self->{'AuthModule'} = 'Kernel::System::Auth::Radius';
$Self->{'AuthModule::Radius::Host'} = 'radiushost';
$Self->{'AuthModule::Radius::Password'} = 'radiussecret';
```

Authentifizierungs-Backends für Kunden

Datenbank (Standard)

Das Backend für die Authentifizierung von Kunden, welches OTRS standardmäßig verwendet, ist die OTRS-Datenbank. Die Kundendaten können über das Interface zur Verwaltung von Kunden angelegt und bearbeitet werden.

Beispiel 11.10. Kundenauthentifizierung gegen ein Datenbank-Backend

```
# This is the auth. module against the otrs db
$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::DB';
$Self->{'Customer::AuthModule::DB::Table'} = 'customer_user';
$Self->{'Customer::AuthModule::DB::CustomerKey'} = 'login';
$Self->{'Customer::AuthModule::DB::CustomerPassword'} = 'pw';
##$Self->{'Customer::AuthModule::DB::DSN'} =
  "DBI:mysql:database=customerdb;host=customerdbhost";
##$Self->{'Customer::AuthModule::DB::User'} = "some_user";
##$Self->{'Customer::AuthModule::DB::Password'} = "some_password";
```

LDAP

Falls ein LDAP Verzeichnis mit Ihren Kundenbenutzern verfügbar ist, können Sie das LDAP Modul für die Authentifizierung Ihrer Kunden nutzen. Dieses Modul greift nur lesend auf die Daten im LDAP Verzeichnis zu, d.h. die Daten können nicht mit OTRS bearbeitet werden, es können also keine Kunden mit Hilfe der Kundenverwaltung von OTRS angelegt oder bearbeitet werden.

Beispiel 11.11. Kundenauthentifizierung gegen ein LDAP Backend

```
# This is an example configuration for an LDAP auth. backend.
# (make sure Net::LDAP is installed!)
$Self->{'Customer::AuthModule'} =
  'Kernel::System::CustomerAuth::LDAP';
$Self->{'Customer::AuthModule::LDAP::Host'} = 'ldap.example.com';
$Self->{'Customer::AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';
$Self->{'Customer::AuthModule::LDAP::UID'} = 'uid';

# Check if the user is allowed to auth in a posixGroup
# (e. g. user needs to be in a group xyz to use otrs)
$Self->{'Customer::AuthModule::LDAP::GroupDN'} =
  'cn=otrsallow,ou=posixGroups,dc=example,dc=com';
$Self->{'Customer::AuthModule::LDAP::AccessAttr'} = 'memberUid';
# for ldap posixGroups objectclass (just uid)
$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'UID';
# for non ldap posixGroups objectclass (full user dn)
```

```

$$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'DN';

# The following is valid but would only be necessary if the
# anonymous user does NOT have permission to read from the LDAP tree
$Self->{'Customer::AuthModule::LDAP::SearchUserDN'} = '';
$Self->{'Customer::AuthModule::LDAP::SearchUserPw'} = '';

# in case you want to add always one filter to each ldap query, use
# this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter =>
# '(objectclass=user)'
$Self->{'Customer::AuthModule::LDAP::AlwaysFilter'} = '';

# in case you want to add a suffix to each customer login name, then
# you can use this option. e. g. user just want to use user but
# in your ldap directory exists user@domain.
$$Self->{'Customer::AuthModule::LDAP::UserSuffix'} = '@domain.com';

# Net::LDAP new params (if needed - for more info see perldoc
# Net::LDAP)
$Self->{'Customer::AuthModule::LDAP::Params'} = {
    port => 389,
    timeout => 120,
    async => 0,
    version => 3,
};

```

HTTPBasicAuth für Kunden

Falls Sie eine "single sign on"-Lösung für Ihre Kunden implementieren möchten, benutzen Sie HTTPBasic Authentication (für alle Ihre Systeme) und aktivieren Sie das HTTPBasicAuth Modul (kein OTRS-Login mehr benötigt).

Beispiel 11.12. Kundenauthentifizierung über HTTPBasic

```

# This is an example configuration for an apache ($ENV{REMOTE_USER})
# auth. backend. Use it if you want to have a single login through
# apache http-basic-auth
$Self->{'Customer::AuthModule'} =
    'Kernel::System::CustomerAuth::HTTPBasicAuth';

# Note:
# If you use this module, you should use the following
# config settings as fallback, if user isn't login through
# apache ($ENV{REMOTE_USER})
$Self->{'CustomerPanelLoginURL'} = 'http://host.example.com/not-
authorised-for-otrs.html';
$Self->{'CustomerPanelLogoutURL'} = 'http://host.example.com/thanks-for-
using-otrs.html';

```

Radius

Mit den folgenden Einstellungen kann die Authentifizierung von Kunden gegen einen Radius-Server realisiert werden.

Beispiel 11.13. Kundenauthentifizierung gegen ein Radius-Backend

```
# This is a example configuration to auth. customer against a radius
server
$Self->{'Customer::AuthModule'} = 'Kernel::System::Auth::Radius';
$Self->{'Customer::AuthModule::Radius::Host'} = 'radiushost';
$Self->{'Customer::AuthModule::Radius::Password'} = 'radiussecret';
```

Kunden-Selbstregistrierung anpassen

Es ist möglich, die Kunden-Selbstregistrierung für neue Kunden über "customer.pl" anzupassen. Somit können Sie mehr optionale oder benötigte Felder (z. B. Adresse, Ort, Telefonnummer) hinzufügen.

In folgenden Beispiel wird ein benötigtes Feld für die Raumnummer hinzugefügt.

Anpassen der Weboberfläche

Damit im Webinterface das zusätzliche Feld für die Raumnummer angezeigt wird, muss die zuständige dtl-Datei angepasst werden. Editieren Sie Kernel/Output/HTML/Standard/CustomerLogin.dtl und fügen Sie das gewünschte Feld hinzu.

```
[...]
<div class="NewLine">
  <label for="Room">$Text{"Room{CustomerUser}"}</label>
  <input title="$Text{"Room Number"}" name="Room" type="text"
    id="UserRoom" maxlength="50" />
</div>
[...]
```

Skript 11.9. Anzeigen eines neuen Feldes in der Weboberfläche.

Kunden-Mapping

Zusätzlich muss das Kunden-Mapping um den Eintrag für die Raumnummer erweitert werden. Dazu werden zuerst die Einstellungen für "CustomerUser" aus der Datei Kernel/Config/Defaults.pm in die Datei Kernel/Config.pm übertragen. Anschließend wird das Kunden-Mapping um das Raumnummer-Feld erweitert.

```
# CustomerUser
# (customer database backend and settings)
$Self->{CustomerUser} = {
  Name => 'Database Backend',
  Module => 'Kernel::System::CustomerUser::DB',
  Params => {
    # if you want to use an external database, add the
    # required settings
    # DSN => 'DBI:odbc:yourdsn',
    # DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
    # User => '',
    # Password => '',
    Table => 'customer_user',
```

```

    },
    # customer unique id
    CustomerKey => 'login',
    # customer #
    CustomerID => 'customer_id',
    CustomerValid => 'valid_id',
    CustomerUserListFields => ['first_name', 'last_name', 'email'],
#   CustomerUserListFields => ['login', 'first_name', 'last_name',
'customer_id', 'email'],
    CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
    CustomerUserSearchPrefix => '',
    CustomerUserSearchSuffix => '*',
    CustomerUserSearchListLimit => 250,
    CustomerUserPostMasterSearchFields => ['email'],
    CustomerUserNameFields => ['title', 'first_name', 'last_name'],
    CustomerUserEmailUniqCheck => 1,
#   # show not own tickets in customer panel, CompanyTickets
#   CustomerUserExcludePrimaryCustomerID => 0,
#   # generate auto logins
#   AutoLoginCreation => 0,
#   AutoLoginCreationPrefix => 'auto',
#   # admin can change customer preferences
#   AdminSetPreferences => 1,
#   # cache time to live in sec. - cache database queries
#   CacheTTL => 0,
#   # just a read only source
#   ReadOnly => 1,
    Map => [

        # note: Login, Email and CustomerID needed!
        # var, frontend, storage, shown (1=always,2=lite), required,
storage-type, http-link, readonly, http-link-target
        [ 'UserTitle',      'Title',      'title',      1, 0, 'var',
'', 0 ],
        [ 'UserFirstname', 'Firstname', 'first_name', 1, 1, 'var',
'', 0 ],
        [ 'UserLastname',  'Lastname',  'last_name',  1, 1, 'var',
'', 0 ],
        [ 'UserLogin',     'Username',  'login',     1, 1, 'var',
'', 0 ],
        [ 'UserPassword',  'Password',  'pw',        0, 0, 'var',
'', 0 ],
        [ 'UserEmail',     'Email',     'email',     1, 1, 'var',
'', 0 ],
        [ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var',
'', 0 ],
        [ 'UserPhone',     'Phone',     'phone',     1, 0, 'var',
'', 0 ],
        [ 'UserFax',       'Fax',       'fax',       1, 0, 'var',
'', 0 ],
        [ 'UserMobile',    'Mobile',    'mobile',    1, 0, 'var',
'', 0 ],
        [ 'UserRoom',     'Room',     'room',     1, 0, 'var',
'', 0 ],
    ],

```

```

        [ 'UserStreet',      'Street',      'street',      1, 0, 'var',
'', 0 ],
        [ 'UserZip',        'Zip',         'zip',         1, 0, 'var',
'', 0 ],
        [ 'UserCity',       'City',        'city',        1, 0, 'var',
'', 0 ],
        [ 'UserCountry',    'Country',     'country',     1, 0, 'var',
'', 0 ],
        [ 'UserComment',    'Comment',     'comments',    1, 0, 'var',
'', 0 ],
        [ 'ValidID',        'Valid',       'valid_id',    0, 1, 'int',
'', 0 ],
    ],
    # default selections
    Selections => {
        UserTitle => {
            'Mr.' => 'Mr.',
            'Mrs.' => 'Mrs.',
        },
    },
};

```

Skript 11.10. Anpassung des MAP-Arrays.

Anpassen der Kunden-Tabelle in der OTRS-Datenbank

Abschließend muss eine neue Spalte zur "customer_user" Tabelle in der OTRS Datenbank hinzugefügt werden, in der die Raumnummer gespeichert werden kann.

```

linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 6 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD room VARCHAR (200);
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> quit
Bye
linux:~#

```

Skript 11.11. Hinzufügen einer neuen Spalte zur Tabelle customer_user.

Alle benötigten Anpassungen sind durchgeführt und das Feld für die Raumnummer sollte nun im Kunden-Interface (customer.pl) angezeigt und verwendet werden können. Neue Kunden müssen nun ihre

Raumnummer angeben, wenn sie einen neuen Zugang einrichten wollen. Wird mod_perl eingesetzt, sollte der Webserver neu gestartet werden um die Änderungen zu übernehmen.

Kapitel 12. Status

Vordefinierte Ticket-Status

OTRS erlaubt es Ihnen, die Ticket-Status zu verändern oder neue Status hinzuzufügen. Hierbei gibt es zwei wichtige Optionen: den Namen des Status und Typ des Status.

Die standardmäßig voreingestellten Status lauten: 'neu', 'offen', 'erfolgreich geschlossen', 'erfolglos geschlossen', 'merged', 'entfernt', 'warten auf erfolgreich schließen', 'warten auf erfolglos schließen' und 'warten zur Erinnerung'.

Neu

Tickets sind normalerweise nur in diesem Status, wenn sie von eingehenden E-Mails neu erzeugt werden.

Öffnen

Der Standardstatus für Tickets, die Queues und Agents zugeordnet sind.

Warten zur Erinnerung

Nachdem die Wartezeit abgelaufen ist, erhält der Ticketbesitzer eine E-Mail zu dem Ticket. Wenn es nicht gesperrt ist, wird die Erinnerung an alle Agenten der Queue verschickt. Erinnerungs-Benachrichtigungen werden nur innerhalb der Arbeitszeiten versendet, und werden alle 24 Stunden wiederholt, bis der Status des Tickets vom Agenten verändert wird. Die Zeit, die dieses Ticket in diesem Status verbringt, wird dennoch bei der Berechnung der Eskalationszeiten einbezogen.

Warten auf erfolglos schließen

Wenn die Wartezeit abgelaufen ist, wird das Ticket auf "erfolglos geschlossen" gesetzt. Die Zeit, die dieses Ticket in diesem Status verbringt, wird dennoch bei der Berechnung der Eskalationszeiten einbezogen.

Warten auf erfolgreich schließen

Wenn die Wartezeit abgelaufen ist, wird das Ticket auf "erfolgreich geschlossen" gesetzt. Die Zeit, die dieses Ticket in diesem Status verbringt, wird dennoch bei der Berechnung der Eskalationszeiten einbezogen.

Zusammengefügt

Status von Tickets, die mit anderen Tickets zusammengefügt wurden.

Erfolgreich geschlossen

Endstatus für Tickets, die erfolgreich gelöst werden konnten. Je nach Konfiguration kann es unmöglich sein, geschlossene Tickets wieder zu eröffnen.

Erfolglos geschlossen

Endstatus für Tickets, die NICHT erfolgreich gelöst werden konnten. Je nach Konfiguration kann es unmöglich sein, geschlossene Tickets wieder zu eröffnen.

Anpassen von Ticket-Status

Jeder Status besteht aus einem Namen ("state-name") und einem Typen ("state-type"). Der Name ist frei wählbar und kann über das Admin-Interface von OTRS angepasst werden, die Statustypen müssen direkt in der Datenbank geändert werden. Letzteres wird jedoch nicht empfohlen und kann zu unerwarteten Ergebnissen führen. Z. B. hängen die Berechnung der Eskalationszeiten und auch die Funktion zum automatischen Entsperren von Tickets von bestimmten Statustypen ab.

Im Admin-Interface können Sie innerhalb der Einstellungen für "Status" neue Status für die vorhandenen Statustypen hinzufügen oder ändern. Beachten Sie, dass Sie bei Änderungen am Status "neu - new" auch die entsprechenden Änderungen in der Konfigurationsdatei `Kernel/Config.pm` bzw. mit Hilfe des grafischen Konfigurations-Front-End vornehmen müssen. Folgende Änderungen müssen für einen geänderten "neu"-Status vorgenommen werden:

```
[...]
# PostmasterDefaultState
# (The default state of new tickets.) [default: new]
$self->{PostmasterDefaultState} = 'new';

# CustomerDefaultState
# (default state of new customer tickets)
$self->{CustomerDefaultState} = 'new';
[...]
```

Script 12.1. Anpassung der Einstellungen von Kernel/Config.pm.

Möchten Sie einen neuen Statustyp hinzufügen, müssen Sie zuerst die `ticket_status-type`-Tabelle in der OTRS Datenbank mit Hilfe eines entsprechenden Datenbankclient anpassen.

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 23 to server version: 5.0.16-Debian_1-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> insert into ticket_state_type (name,comments) values
('own','Own
state type');
Query OK, 1 row affected (0.00 sec)

mysql> quit
Bye
linux:~#
```

Script 12.2. Anpassung der Statustypen.

Nun können Sie den neuen Statustyp verwenden. Nachdem ein Status mit diesem Statustyp verknüpft wurde, muss auch die OTRS-Konfiguration angepasst werden, damit der neue Status korrekt verwendet werden kann. Passen Sie die Folgenden Optionen in der SysConfig an:

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateDefault - um den Standardstatus für Telefontickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateType - um die verfügbaren Status für Telefontickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateDefault - um den Standardstatus für E-Mail-Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateType - um die verfügbaren Status für E-Mail-Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###State - um den Standardstatus für neue Telefonartikel einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###StateType - um die verfügbaren Status für neue Telefonartikel einzustellen

Ticket:Frontend::Agent::Ticket::ViewMove:Ticket::DefaultNextMoveStateType - um den Standardstatus für das Verschieben von Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateDefault - um den Standardstatus für das Bouncen eines Tickets einzustellen

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateType - um die verfügbaren Status für das Bouncen von Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateDefault - um den Standardstatus in einer Sammelaktion einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateType - um die verfügbaren Status in einer Sammelaktion einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewClose > StateDefault - um den Standardstatus für das Schließen von Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewClose > StateType - um die verfügbaren Status für das Schließen von Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateDefault - um den Standardstatus für Antworten einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateType - um die verfügbaren Status für das Antworten einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - um den Standardstatus für das Weiterleiten von Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - um die verfügbaren Status für das Weiterleiten von Tickets einzustellen.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - um den Standardstatus für das Bearbeiten der Freitextfelder festzulegen.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - um die verfügbaren Status für das Bearbeiten der Freitextfelder festzulegen.

Ticket -> Core::PostMaster > PostmasterDefaultState - um den Status von Tickets festzulegen, die aus E-Mails neu erzeugt werden.

Ticket -> Core::PostMaster > PostmasterFollowUpState - um den Status von Tickets festzulegen, nachdem ein Followup eingegangen ist.

Ticket -> Core::PostMaster > PostmasterFollowUpStateClosed - um den Status von bereits geschlossenen Tickets festzulegen, nachdem ein Followup eingegangen ist.

Ticket -> Core::Ticket > ViewableStateType - um die Ticketstatus festzulegen, die in verschiedenen Bereichen des Systems zur Anzeige kommen, z. B. in der Queue-Ansicht.

Ticket -> Core::Ticket > UnlockStateType - um die verfügbaren Status für das Entsperren von Tickets einzustellen.

Ticket -> Core::Ticket > PendingReminderStateType - um die Status für Erinnerungstickets festzulegen.

Ticket -> Core::Ticket > PendingAutoStateType - um die Status für Tickets mit "Warten auto" festzulegen.

Ticket -> Core::Ticket > StateAfterPending - um den Status von Tickets mit "Warten auto" einzustellen, deren Wartezeit abgelaufen ist.

Kapitel 13. Anpassen der Ticket-Prioritäten

OTRS enthält 5 vordefinierte Ticketprioritäten, die mit dem "Prioritäten"-Modul im Administrationsbereich verwaltet werden können. Wenn Sie eine angepasste Prioritätenliste erstellen, bedenken Sie bitte, dass diese in der Prioritätsauswahl-Liste alphabetisch sortiert dargestellt werden. In der Queue-Ansicht ordnet OTRS die Prioritäten nach internen Datenbank-IDs.

Anmerkung

In OTRS können Prioritäten deaktiviert, aber nicht gelöscht werden. Wenn Sie eine Priorität deaktivieren wollen, setzen Sie den Wert für "Gültig" auf "ungültig" oder "ungültig-temporär".

Wichtig

Wenn eine Priorität hinzugefügt oder geändert wurde, müssen Sie ggf. einige Werte in der SysConfig anpassen:

- Ticket:Core::Postmaster::PostmasterDefaultPriority - definiert die Standardpriorität für einkommende E-Mails.
- Ticket:Frontend::Agent:Ticket::ViewPhoneNew:Priority - definiert die Standardpriorität für neue Telefontickets im Agentenbereich.
- Ticket:Frontend::Agent:Ticket::ViewEmailNew:Priority - definiert die Standardpriorität für neue E-Mail-Tickets im Agentenbereich.
- Ticket:Frontend::Customer:Ticket::ViewNew:PriorityDefault - definiert die Standardpriorität für neue Tickets im Kundenbereich.

Kapitel 14. Erstellung eigener Themes

Für OTRS können verschiedene Themes angelegt werden, also verschiedene Layouts zur Gestaltung der Web-Oberfläche. Dazu müssen Sie die vorhandenen Templates ändern und Ihren Wünschen entsprechend anpassen.

Mehr Informationen über die Syntax und den Aufbau von Templates finden Sie im Entwickler-Handbuch auf <http://doc.otrs.org> innerhalb des Kapitels zu den *Templates* [<http://doc.otrs.org/developer/3.0/en/html/hacking.html#TemplatingMechanism>].

Führen Sie beispielsweise folgende Schritte durch um ein neues Theme namens "Company" zu erstellen:

1. Erstellen Sie das Verzeichnis `Kernel/Output/HTML/Company` und kopieren Sie die Dateien, die Sie für das Company-Theme anpassen möchten, aus `Kernel/Output/HTML/Standard` in das neu erstellte Verzeichnis.

Wichtig

Kopieren Sie nur diejenigen Dateien, die Sie verändern möchten. Fehlende Dateien wird OTRS automatisch vom Standard-Theme laden. Dadurch werden Systemaktualisierungen vereinfacht.

2. Passen Sie die Dateien im Verzeichnis `Kernel/Output/HTML/Company` Ihren Wünschen entsprechend an.
3. Um das neue Theme OTRS bekannt zu machen, fügen Sie es unter `Frontend::Themes` zur `SysConfig` hinzu.

Ab nun sollten sie das neue Theme nutzen und über Ihre persönlichen Einstellungen aktivieren können.

Warnung

Nehmen Sie keine Änderungen an den Theme-Dateien vor, die mit OTRS installiert wurden, da diese bei einem Update verloren gehen können. Erstellen Sie eigene Themes ausschließlich nach dem oben beschriebenen Verfahren.

Kapitel 15. Übersetzung des OTRS-Frontends

OTRS bietet Mehrsprachigkeit in der Weboberfläche an.

Wie der Lokalisationsmechanismus des OTRS Frameworks arbeitet und welche Möglichkeiten er bietet, entnehmen Sie bitte dem Kapitel Language Translations [<http://doc.otrs.org/developer/3.0/en/html/contributing.html#translate>] aus dem Developer Handbuch auf <http://doc.otrs.org>. Dieses Kapitel beschreibt, wie eigene Übersetzungen eingepflegt und komplett neue Sprachen hinzugefügt werden können.

Kapitel 16. PGP

In OTRS können ausgehende E-Mails mit Hilfe von PGP signiert oder verschlüsselt werden. Ebenfalls ist es möglich verschlüsselte Nachrichten zu entschlüsseln. Die Ver- und Entschlüsselung mit PGP wird mit Hilfe des GPL-Werkzeugs GnuPG vorgenommen. Zur Einrichtung sind die folgenden Schritte notwendig:

1. Erste Aufgabe ist es, das entsprechende GnuPG-Software-Paket zu installieren, welches bei den meisten Linux-Distributionen mitgeliefert wird. Dies sollte mit Hilfe des jeweiligen Paketmanagers leicht durchgeführt werden können.
2. Im zweiten Schritt muss das soeben installierte GnuPG zur Benutzung für OTRS konfiguriert werden. Dies geschieht auf der Kommandozeilenebene durch einen Aufruf von GnuPG, der die notwendigen Verzeichnisse anlegt und den privaten Schlüssel erzeugt. Der Aufruf muss als Benutzer 'otrs' durchgeführt werden.

```
linux:~# su otrs
linux:/root$ cd
linux:~$ pwd
/opt/otrs
linux:~$ gpg --gen-key
gpg (GnuPG) 1.4.2; Copyright (C) 2005 Free Software Foundation,
Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: directory `/opt/otrs/.gnupg' created
gpg: new configuration file `/opt/otrs/.gnupg/gpg.conf' created
gpg: WARNING: options in `/opt/otrs/.gnupg/gpg.conf' are not yet
active during t
his run
gpg: keyring `/opt/otrs/.gnupg/secring.gpg' created
gpg: keyring `/opt/otrs/.gnupg/pubring.gpg' created
Please select what kind of key you want:
  (1) DSA and Elgamal (default)
  (2) DSA (sign only)
  (5) RSA (sign only)
Your selection? 1
DSA keypair will have 1024 bits.
ELG-E keys may be between 1024 and 4096 bits long.
What keysize do you want? (2048)
Requested keysize is 2048 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0)
Key does not expire at all
Is this correct? (y/N) y
```

You need a user ID to identify your key; the software constructs the user ID

from the Real Name, Comment and Email Address in this form:
"Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: Ticket System
Email address: support@example.com
Comment: Private PGP Key for the ticket system with address support@example.com

You selected this USER-ID:
"Ticket System (Private PGP Key for the ticket system with address support@examp
le.com) <support@example.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? O
You need a Passphrase to protect your secret key.

Passphrase: secret
Repeat passphrase: secret

We need to generate a lot of random bytes. It is a good idea to perform some other action (type on the keyboard, move the mouse, utilize the disks) during the prime generation; this gives the random number generator a better chance to gain enough entropy.

+++++++.....+++++.....+++++
+++++++.
+++++++.....+++++.....+++++.....>
+++++++>+++++
.....>+++++<+++++.....+++++

Not enough random bytes available. Please do some other work to give the OS a chance to collect more entropy! (Need 280 more bytes)

+++++++.....+++++.....+++++.....+++++.....+++++.....+++++
+++++++.....+++++
+++++++.....+++++.....+++++.....+++++
+.....+++++
.....>+++++.....>+++++
+.....>+++++
+<+++++.....>+++++

.....+++++^^^
gpg: /opt/otrs/.gnupg/trustdb.gpg: trustdb created
gpg: key 7245A970 marked as ultimately trusted
public and secret key created and signed.

gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
pub 1024D/7245A970 2006-02-03

```

Key fingerprint = 2ED5 BC36 D2B6 B055 7EE1 5833 1D7B F967
7245 A970
uid Ticket System (Private gpg key for ticket
system with address support@example.com) <support@example.com>
sub 2048g/52B97069 2006-02-03

linux:~$

```

Skript 16.1. Konfiguration von GnuPG.

Wie man sehen kann, genügt es bei den meisten Fragen die Vorgabe mit <EINGABE> zu bestätigen. Lediglich die Angabe zur Person des Schlüsselbesitzers ist zu vervollständigen bzw. zum Ende hin ist an der mit (passphrase) gekennzeichneten Stelle die Passphrase für den zu generierenden Schlüssel einzugeben.

3. Im nächsten Schritt muss OTRS auf die Verwendung von PGP vorbereitet werden. Suchen Sie in SysConfig nach "PGP" und wählen Sie danach die Untergruppe Crypt::PGP aus.

In der nun angezeigten Maske sollte zum einen " PGP aktiviert werden (die erste Option). Danach sollte überprüft werden, ob der voreingestellte Pfad zum Programm gpg der tatsächlichen Installation entspricht.

Die nächste Einstellung (PGP::Options) muss ggf. modifiziert werden. Es handelt sich um die Parameter, mit denen OTRS das Programm gpg aufruft. Hier ist insbesondere die Option für die Lage des GnuPG-Konfigurationsverzeichnis des OTRS-Benutzers otrs wichtig. Im Beispiel ist dies: /opt/otrs/.gnupg). Dieses Verzeichnis wurde im Schritt 1 automatisch von GnuPG angelegt.

Über die letzte Option können die Schlüssel-Werte-Paare für die ID(s) und Passphrase(n) der eigenen PGP-Schlüssel dem Ticket System bekannt gemacht werden. Noch einmal genauer: da andere Kommunikationspartner an das Ticket-System (oder besser: an den Mail-Eingang des Systems) E-Mails mit dessen öffentlichen Schlüssel verschlüsselt schicken, kann OTRS mit dem/n in dieser Option angegebenen privaten Schlüssel(n) solchermaßen verschlüsselte Mails entschlüsseln.

Woher bekommt man die ID des eigenen Schlüssels? Die ID steckt schon in der Ausgabe der Schlüsselgenerierung (siehe Schritt 1). Man kann die ID aber auch als Benutzer 'otrs' jederzeit über die Kommandozeile ermitteln:

```

linux:~# su otrs
linux:/root$ cd
linux:~$ pwd
/opt/otrs
linux:~$ gpg --list-keys
/opt/otrs/.gnupg/pubring.gpg
-----
pub 1024D/7245A970 2006-02-03
uid Ticket System (Private gpg key for ticket
system with address support@example.com) <support@example.com>
sub 2048g/52B97069 2006-02-03

linux:~$

```

Skript 16.2. Die ID des eigenen privaten Schlüssels ermitteln.

Die ID des Schlüssels befindet sich in der Zeile, die mit sub beginnt und ist eine 8-stellige hexadezimale Kennung (im Beispiel lautet sie "52B97069"). Die für die Option geforderte Passphrase ist dieselbe, die beim Schlüsselgenerieren in Schritt 1 verwendet wurde.

Nach Eingabe all dieser Angaben können sie mit dem Aktualisieren-Button gespeichert werden. OTRS ist jetzt für das Empfangen mit PGP verschlüsselter E-Mails konfiguriert.

4. Der letzte Schritt ist der Import des öffentlichen PGP-Schlüssels eines Kunden. Dadurch wird sicher gestellt, dass aus OTRS heraus verschlüsselte Mails an den jeweiligen Kunden gesendet werden können. Es bestehen zwei Möglichkeiten für den Import.

Zum einem können über das Modul zur Verwaltung der Kunden die öffentlichen PGP-Schlüssel beim Anlegen/Bearbeiten des jeweiligen Kunden im System hinterlegt werden.

Die zweite Möglichkeit bietet das System in den PGP-Einstellungen innerhalb des Admin-Bereiches. In dieser Konfigurationsmaske ist im rechten Teil eine Liste der bereits dem System zur Verfügung stehenden öffentlichen Schlüssel zu sehen. Im Regelfall sollte nach der obigen Einrichtung hier schon der öffentliche Schlüssel des Ticket-Systems selbst zu sehen sein. Im linken Teil besteht neben der Suche die Möglichkeit, neue Schlüssel als Schlüsseldatei ins System zu laden.

Sowohl für das Hinzufügen eines Schlüssels über die Kunden- als auch über die PGP-Verwaltung gilt, dass die Dateien mit den Schlüsseln PGP/GnuPG-konforme Schlüssel sein müssen. Normalerweise wird der Schlüssel als "ASCII armored key"-Datei vorliegen, welches problemlos von OTRS verarbeitet werden kann.

Kapitel 17. S/MIME

Die Einrichtung der Verschlüsselung mit S/MIME scheint auf den ersten Blick ein bisschen komplizierter als die PGP-Einrichtung zu sein, da für das OTRS-System erst einmal eine Certification Authority (CA) eingerichtet werden muss. Ansonsten ist das Vorgehen ähnlich wie bei PGP, OTRS konfigurieren, eigenes Zertifikat einrichten, ggf. fremde Public-Zertifikate importieren, usw.

Die S/MIME-Konfiguration geschieht zu einem großen Teil außerhalb der OTRS-Web-Oberfläche und sollte als otrs-Benutzer (bzw. als der Benutzer mit dessen Rechten OTRS läuft) in einer Shell durchgeführt werden. Da die MIME-Konfiguration unter Linux im wesentlichen auf SSL (openssl basiert, sollte zuerst sichergestellt werden, dass das openssl-Paket installiert ist. Mit dem openssl-Paket kommt ein Skript, `CA.pl`, mit welchem die wichtigsten Schritte zur Zertifikatserstellung bewältigt werden können. Damit dieser Vorgang einfacher wird, sollte zuerst herausgefunden werden, wo sich das Skript `CA.pl` im Dateisystem befindet. Danach sollte diese Stelle der Einfachheit halber temporär in den Suchpfad der Shell übernommen werden.

```
otrs@linux:~> rpm -ql openssl | grep CA
/usr/share/ssl/misc/CA.pl
otrs@linux:~> export PATH=$PATH:/usr/share/ssl/misc
otrs@linux:~> which CA.pl
/usr/share/ssl/misc/CA.pl
otrs@linux:~> mkdir tmp; cd tmp
otrs@linux:~/tmp>
```

Skript 17.1. Konfiguration von S/MIME.

Im Beispiel sieht man auch, dass ein temporäres Verzeichnis `~/tmp` angelegt wurde, in welchem die Zertifikatsgenerierung durchgeführt wird.

Im einzelnen sind zur Zertifikatserzeugung folgende Schritte durchzuführen, die einzelnen Schritte sind in der Kommandozeile auszuführen. Sollte ein beglaubigtes SSL-Zertifikat für die Verschlüsselung bereits vorhanden sein, so sollte natürlich dieses verwendet werden. Dann können die nun folgenden Schritte übersprungen werden. Der beschriebene Ablauf geht davon aus, dass der OTRS-Administrator sich das SSL-Zertifikat zu Test- und Lernzwecken selbst anlegen muss.

1. Anlegen einer eigenen Certification Authority für SSL. Diese wird benötigt, um die Anfrage für ein eigenes SSL-Zertifikat zu beglaubigen.

```
otrs@linux:~/tmp> CA.pl -newca
CA certificate filename (or enter to create)

Making CA certificate ...
Generating a 1024 bit RSA private key
.....+++++
.....+++++
writing new private key to './demoCA/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be
  incorporated
into your certificate request.
```

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

```
Country Name (2 letter code) [AU]:DE
State or Province Name (full name) [Some-State]:OTRS-state
Locality Name (eg, city) []:OTRS-town
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Your
company
```

```
Organizational Unit Name (eg, section) []:
```

```
Common Name (eg, YOUR name) []:OTRS Admin
```

```
Email Address []:otrs@your-domain.tld
```

```
otrs@linux:~/tmp> ls -la demoCA/
```

```
total 8
```

```
-rw-r--r--  1 otrs otrs 1330 2006-01-08 17:54 cacert.pem
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 certs
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 crl
-rw-r--r--  1 otrs otrs   0 2006-01-08 17:53 index.txt
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 newcerts
drwxr-xr-x  2 otrs otrs  80 2006-01-08 17:54 private
-rw-r--r--  1 otrs otrs  17 2006-01-08 17:54 serial
```

```
otrs@linux:~/tmp>
```

Skript 17.2. Anlegen einer Certification Authority für SSL.

2. Erzeugen der Zertifikatsanfrage.

```
otrs@linux:~/tmp> CA.pl -newreq
```

```
Generating a 1024 bit RSA private key
```

```
.....++++++
```

```
....++++++
```

```
writing new private key to 'newreq.pem'
```

```
Enter PEM pass phrase:
```

```
Verifying - Enter PEM pass phrase:
```

```
-----
```

You are about to be asked to enter information that will be incorporated

into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

```
Country Name (2 letter code) [AU]:DE\keyreturn
```

```
State or Province Name (full name) [Some-State]:OTRS-state
```

```
Locality Name (eg, city) []:OTRS-town
```

```
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Your
company
```

```
Organizational Unit Name (eg, section) []:
```

```
Common Name (eg, YOUR name) []:OTRS admin
Email Address []:otrs@your-domain.tld
```

Please enter the following 'extra' attributes
to be sent with your certificate request

A challenge password []:

An optional company name []:

Request (and private key) is in newreq.pem

```
otrs@linux:~/tmp> ls -la
```

```
total 4
```

```
drwxr-xr-x  6 otrs otrs  232 2006-01-08 17:54 demoCA
```

```
-rw-r--r--  1 otrs otrs 1708 2006-01-08 18:04 newreq.pem
```

```
otrs@linux:~/tmp>
```

Skript 17.3. Erstellen einer Zertifikatsanfrage.

3. Die Zertifikatsanfrage durch die CA signieren lassen. Die Zertifikatsanfrage kann entweder durch die selbst angelegte CA signiert (= beglaubigt) werden. Allerdings ist es natürlich seriöser, wenn das eigene SSL-Zertifikat von einer fremden, externen und ihrerseits beglaubigten CA beglaubigt wird.

```
otrs@linux:~/tmp> CA.pl -signreq
```

```
Using configuration from /etc/ssl/openssl.cnf
```

```
Enter pass phrase for ./demoCA/private/cakey.pem:
```

```
Check that the request matches the signature
```

```
Signature ok
```

```
Certificate Details:
```

```
Serial Number:
```

```
fd:85:f6:9f:14:07:16:c8
```

```
Validity
```

```
Not Before: Jan  8 17:04:37 2006 GMT
```

```
Not After  : Jan  8 17:04:37 2007 GMT
```

```
Subject:
```

```
countryName           = DE
```

```
stateOrProvinceName  = OTRS-state
```

```
localityName         = OTRS-town
```

```
organizationName     = Your Company
```

```
commonName           = OTRS administrator
```

```
emailAddress          = otrs@your-domain.tld
```

```
X509v3 extensions:
```

```
X509v3 Basic Constraints:
```

```
CA:FALSE
```

```
Netscape Comment:
```

```
OpenSSL Generated Certificate
```

```
X509v3 Subject Key Identifier:
```

```
01:D9:1E:58:C0:6D:BF:27:ED:37:34:14:D6:04:AC:C4:64:98:7A:22
```

```
X509v3 Authority Key Identifier:
```

```
keyid:10:4D:8D:4C:93:FD:2C:AA:9A:B3:26:80:6B:F5:D5:31:E2:8E:DB:A8
```

```
DirName:/C=DE/ST=OTRS-state/L=OTRS-town/O=Your
```

```
Company/
```

```
CN=OTRS admin/emailAddress=otrs@your-domain.tld
```

```
serial:FD:85:F6:9F:14:07:16:C7
```

```
Certificate is to be certified until Jan  8 17:04:37 2007 GMT (365
days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
Signed certificate is in newcert.pem
otrs@linux:~/tmp>
```

Skript 17.4. Signieren der Zertifikatsanfrage.

4. Mit der signierten Zertifikatsanfrage das eigene Zertifikat und alle dazugehörigen Dateien erzeugen.

```
otrs@linux:~/tmp> CA.pl -pkcs12 "OTRS Certificate"
Enter pass phrase for newreq.pem:
Enter Export Password:
Verifying - Enter Export Password:
otrs@linux:~/tmp> ls -la
total 12
drwxr-xr-x  6 otrs otrs  328 2006-01-08 18:04 demoCA
-rw-r--r--  1 otrs otrs 3090 2006-01-08 18:13 newcert.p12
-rw-r--r--  1 otrs otrs 3791 2006-01-08 18:04 newcert.pem
-rw-r--r--  1 otrs otrs 1708 2006-01-08 18:04 newreq.pem
otrs@linux:~/tmp>
```

Skript 17.5. Neues Zertifikat erstellen.

Nach der Durchführung dieser Schritte ist es notwendig die Einrichtung von S/MIME in OTRS abzuschließen.

Die OTRS-seitige Einrichtung erfolgt aus dem Admin-Bereich, Block System über den Punkt SMIME. Falls die generelle S/MIME-Unterstützung in OTRS noch nicht aktiviert wurde, weist die Maske den OTRS-Administrator beim Aufruf darauf hin und bietet einen bequemen Link zur Einrichtung.

Über die SysConfig kann die generelle S/MIME-Unterstützung eingeschaltet und konfiguriert werden. Diese Einrichtung findet man unter der SysConfig-Gruppe Crypt::SMIME.

Hier können sie S/MIME-Unterstützung aktivieren, und die Pfade zum OpenSSL-Kommando und zu den Zertifikaten angeben. Die vorhin erzeugte Schlüsseldatei muss in diesem Verzeichnis gespeichert werden, sonst kann OpenSSL sie nicht verwenden.

Im nächsten Schritt geht es (zurück) zur S/MIME-Konfiguration im Administrationsbereich. Dort können nun sowohl der private Schlüssel(Key) bzw. die privaten Schlüssel des OTRS-Systems, als auch die öffentlichen Schlüssel anderer Kommunikationspartner ins System importiert werden. Tragen Sie hier den öffentlichen Schlüssel ein, der zu Beginn dieses Abschnitts erzeugt und dann in OTRS hinzugefügt wurde.

Selbstverständlich können aber auch alle öffentlichen S/MIME-Schlüssel der Kommunikationspartner über das Modul zur Kundenverwaltung beim Anlegen/Bearbeiten des jeweiligen Kunden in das System importiert werden.

Kapitel 18. Access Control Lists (ACLs)

Seit OTRS 2.0 können Access Control Lists (ACLs) verwendet werden, um den Zugriff auf Tickets, Module, Queues, usw. zu steuern bzw. um in bestimmten Situationen Aktionen auf Tickets (schließen, verschieben, usw.) zu beeinflussen. ACLs können als Ergänzung zum bestehenden Berechtigungssystem der Rollen und Gruppen verwendet werden. Mit ACLs lassen sich, basierend auf Ticket-Attributen, rudimentäre Workflows innerhalb des Systems abbilden.

ACLs können noch nicht über das SysConfig-Interface erstellt werden, sie müssen direkt in die Datei `Kernel/Config.pm` eingetragen werden. Im Folgenden einige Beispiele:

Beispiel 18.1. ACL, die das Verschieben von Tickets in eine Queue nur bei einer Ticket-Priorität von 5 erlaubt.

```
# ticket acl
$self->{TicketAcl}->{'ACL-Name-2'} = {
    # match properties
    Properties => {
        # current ticket match properties
        Ticket => {
            Queue => ['Raw'],
            Priority => ['5 very high'],
        }
    },
    # return possible options (white list)
    Possible => {
        # possible ticket options (white list)
        Ticket => {
            Queue => ['Alert'],
        },
    },
};
```

Beispiel 18.2. ACL, die das Schließen von Tickets in der raw-Queue verbietet und den Schließen-Schalter ausblendet.

```
$self->{TicketAcl}->{'ACL-Name-1'} = {
    # match properties
    Properties => {
        # current ticket match properties
        Ticket => {
            Queue => ['Raw'],
        }
    },
    # return possible options (white list)
    Possible => {
        # possible ticket options (white list)
        Ticket => {
            State => ['new', 'open', 'pending reminder'],
        },
    },
};
```

```

# possible action options
Action => {
  AgentTicketLock => 1,
  AgentTicketZoom => 1,
  AgentTicketClose => 0,
  AgentTicketPending => 1,
  AgentTicketNote => 1,
  AgentTicketHistory => 1,
  AgentTicketPriority => 1,
  AgentTicketFreeText => 1,
  AgentTicketHistory => 1,
  AgentTicketCompose => 1,
  AgentTicketBounce => 1,
  AgentTicketTicketPrint => 1,
  AgentTicketForward => 1,
  AgentTicketTicketLink => 1,
  AgentTicketPrint => 1,
  AgentTicketPhone => 1,
  AgentTicketCustomer => 1,
  AgentTicketOwner => 1,
},
};

```

Beispiel 18.3. ACL, die den Status für alle Agenten entfernt und Ihn nur noch für eine Gruppe zur Verfügung stellt.

```

$Self->{TicketAcl}->{'ACL-Name-5'} = {
  # match properties
  Properties => {
    # current ticket match properties (match always)
  },
  # return possible options
  PossibleNot => {
    # possible ticket options
    Ticket => {
      State => ['closed successful'],
    },
  },
};

```

In Skript 18.1 finden Sie eine Liste aller Parameter, die für ACLs verwendet werden können.

```

# ticket acl
$Self->{TicketAcl}->{'ACL-Name-Test'} = {
  # match properties
  Properties => {
    # current action match properties
    Frontend => {
      Action => ['AgentTicketPhone', 'AgentTicketEmail'],
    },
    # current user match properties

```

```

User => {
  Group_rw => [
    'hotline',
  ],
},
# current user match properties
Ticket => {
  Queue => ['Raw'],
  State => ['new', 'open'],
  Priority => ['some priority'],
  Lock => ['lock'],
  CustomerID => ['some id'],
  CustomerUserID => ['some id'],
  TicketFreeKey1 => ['some key'],
  TicketFreeKey2 => ['some key'],
  # ...
  TicketFreeKey8 => ['some key'],
  TicketFreeText1 => ['some value'],
  TicketFreeText2 => ['some value'],
  # ...
  TicketFreeText8 => ['some value'],
}
},
# return possible options (white list)
Possible => {
  # possible ticket options (white list)
  Ticket => {
    Queue => ['Hotline', 'Koordination'],
    State => => ['some state'],
    Priority => ['5 very high'],
    TicketFreeKey1 => ['some key'],
    TicketFreeKey2 => ['some key'],
    # ...
    TicketFreeKey8 => ['some key'],
    TicketFreeText1 => ['some value'],
    TicketFreeText2 => ['some value'],
    # ...
    TicketFreeText8 => ['some value'],
  },
  # possible action options (white list)
  Action => {
    AgentTicketLock => 1,
    AgentTicketZoom => 1,
    AgentTicketClose => 1,
    AgentTicketPending => 0,
    AgentTicketNote => 1,
    AgentTicketHistory => 0,
    AgentTicketPriority => 1,
    AgentTicketFreeText => 0,
    AgentTicketHistory => 1,
    AgentTicketCompose => 1,
    AgentTicketBounce => 1,
    AgentTicketTicketPrint => 0,
    AgentTicketForward => 1,
  }
}

```

```
        AgentTicketTicketLink => 1,
        AgentTicketPrint => 1,
        AgentTicketPhone => 1,
        AgentTicketCustomer => 1,
        AgentTicketOwner => 0,
    },
},
# remove options (black list)
PossibleNot => {
    # possible ticket options (black list)
    Ticket => {
        Queue => ['Hotline', 'Koordination'],
        State => ['closed', 'removed'],
    },
},
};
```

Skript 18.1. Verwendbare Parameter für ACLs.

Kapitel 19. Statistikmodul

Das OTRS-Statistikmodul enthält Funktionen, um betriebliche Kennziffern zu überwachen und angepasste Reports der Verwendung von OTRS zu erstellen. In OTRS wird der Begriff "Stat/Statistik" für einen Report benutzt, der verschiedene Indikatoren abbildet.

Eine sinnvolle Konfiguration des OTRS-Statistikmoduls ist verbunden mit einer Vielzahl von Anforderungen und Überlegungen. Darin eingeschlossen sind die auszuwertenden Module von OTRS, Benutzerrechte, zu berechnende Indikatoren und ihre Komplexitätsgrade, eine einfache Konfiguration des Statistikmoduls, Geschwindigkeit und Effizienz der Berechnungen, und die Unterstützung verschiedener Ausgabevarianten.

Um komplexe Statistiken abzubilden, besteht ferner die Möglichkeit statische Elemente zu integrieren (Dateien, die die Funktionalität des Statistik-Moduls um ganz spezifische Anforderungen ergänzen).

Bedienung des Moduls durch den Agent

Wenn man als Agent angemeldet ist, erscheint der Menüpunkt "Statistik" mit verschiedenen Unterpunkten, siehe Abb. 19.1.

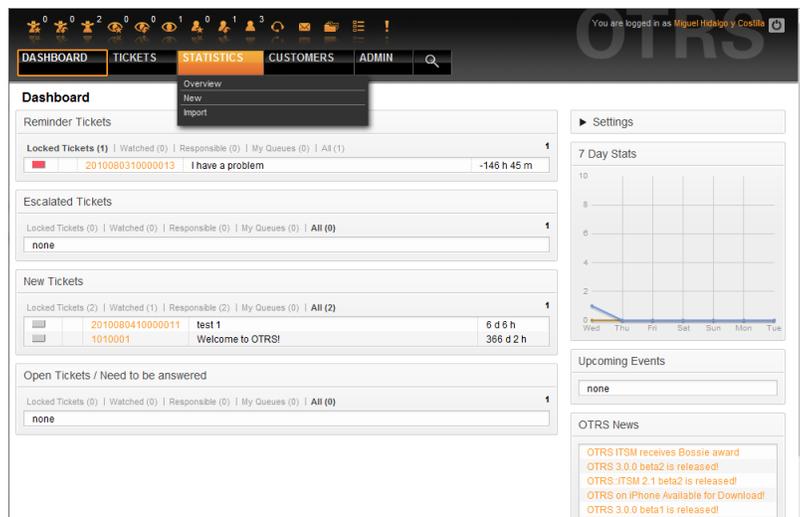


Abb. 3.7. Statistikmenü.

Folgende Optionen sind im Statistikmenü verfügbar:

- *Übersicht.* Zeigt eine Liste vorkonfigurierter Reports an.
- *Neu.* Benötigt "rw"-Rechte.
- *Importieren.* Benötigt "rw"-Rechte.

Übersicht

Die Auswahl von "Statistik" und des Unterpunktes "Übersicht" ruft den Übersichtsbildschirm auf. Hier wird dem Agenten eine Liste von vorkonfigurierten Reports präsentiert, die er verwenden kann.

STAT#	TITLE	OBJECT	DESCRIPTION
10001	List of the most time-consuming tickets	Ticketlist	List of tickets closed last month which required [..]
10002	Changes of status in a monthly overview		Monthly overview, which reports status changes per[...]
10003	List of tickets created last month	Ticketlist	List of all tickets created last month. Order by ag[...]
10004	List of open tickets, sorted by time left until solution deadline expires	Ticketlist	List of open tickets, sorted by time left until sol[...]
10005	List of tickets closed, sorted by solution time	Ticketlist	List of tickets closed last month, sorted by solu[...]
10006	List of open tickets, sorted by time left until escalation deadline expires	Ticketlist	List of open tickets, sorted by time left until es[...]
10007	List of tickets closed last month	Ticketlist	List of all tickets closed last month. Order by ag[...]
10008	Overview about all tickets in the system	TicketAccumulation	Current state of all tickets in the system without[...]
10009	List of tickets closed, sorted by response time	Ticketlist	List of tickets closed last month, sorted by respo[...]
10010	List of open tickets, sorted by time left until response deadline expires	Ticketlist	List of open tickets, sorted by time left until re[...]
10011	New Tickets	TicketAccumulation	Total number of new tickets per day and queue whic[...]

Abb. 19.2. Übersicht der Standardstatistiken.

Folgende Informationen sind den in der Übersicht aufgelisteten Statistiken zu entnehmen:

- *Stat#*. Eindeutige Statistiknummer.
- *Titel*. Titel der Statistik.
- *Objekt*. Objekt, das für die Generierung der Statistik verwendet wird. Wenn es sich um eine statische Statistik handelt, wird kein Objekt angezeigt, da kein dynamisches Objekt zur Generierung verwendet wird.
- *Beschreibung*. Ein Auszug aus der Beschreibung der Statistik.

Wenn das Statistikmodul installiert ist, enthält es einige vorkonfigurierte Reports. Diese können in der Übersicht aufgerufen werden. Wenn die Liste zu lang für eine Seite wird, kann der Agent durch die verschiedenen Seiten blättern. Die Liste kann durch klick auf den Spaltenkopf wie gewünscht geordnet werden. Um einen bestimmten Report zu erzeugen, klicken Sie auf die Statistiknummer des Reports in der Übersicht. Dadurch gelangen Sie in die Ansichtseite des Reports.

Ansehen und Generieren von Statistiken

Aufgabe der "Ansehen"-Oberfläche ist es, dem Agent einen Überblick über die Vorkonfiguration der Statistik zu ermöglichen.

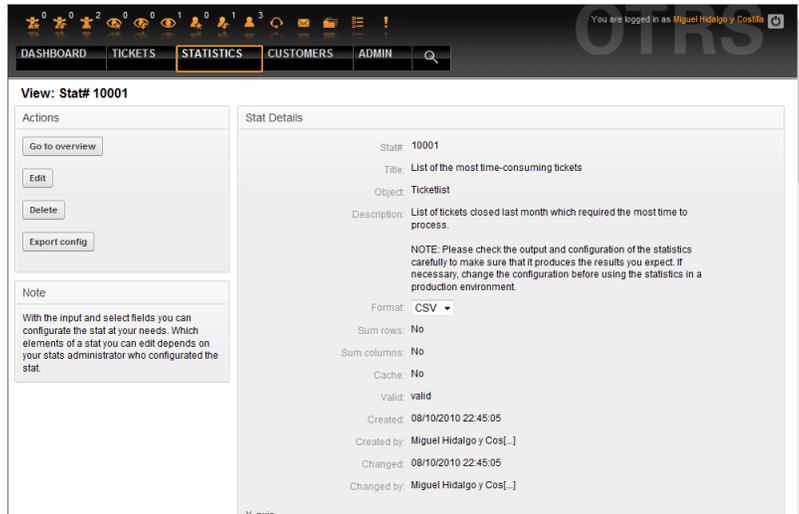


Abb. 19.3. Ansicht einer Statistik.

Konfigurationseinstellungen für einen Report können in den Optionen der Ansichtseite eingestellt werden. Der Ersteller des Reports und jeder Agent mit den benötigten Rechten kann diese Einstellungen vornehmen.

Konkret wird auf dieser Seite folgendes angezeigt:

- **Verfügbare Aktionen:**
 - *Zurück zur Übersicht.* Zurück zur Übersichtsliste der Reports.
 - *Bearbeiten.* Die aktuelle Statistik bearbeiten (benötigt "rw"-Rechte).
 - *Löschen.* Die aktuelle Statistik löschen (benötigt "rw"-Rechte).
 - *Konfiguration exportieren.* Exportieren einer Statistikkonfiguration mittels Dateidownload ("rw"-Rechte erforderlich).

Verwendung: Mittels der Export- und Importfunktionen können Statistiken bequem auf Testsystemen entworfen und getestet werden, um dann anschließend einfach in das Produktivsystem integriert zu werden.

- **Report-Details:**
 - *Statistik-Nr..* Die Nummer der Statistik.
 - *Titel.* Titel der Statistik.
 - *Objekt.* Objekt, das für die Generierung der Statistik verwendet wird.
 - *Beschreibung.* Ein Auszug aus der Beschreibung der Statistik.
 - *Format.* Ausgabeformat der Statistik. Hierbei stehen je nach Konfiguration folgende Ausgabeformate zur Verfügung:
 - CSV.
 - Drucken.

- Liniendiagramm.
- Balkendiagramm.
- Balkendiagramm (horizontal).
- Punktdiagramm
- Linienpunktdiagramm.
- Flächendiagramm.
- Tortendiagramm.
- *Grafikgröße*. Mögliche Größe der Grafik bzw. des Diagramms. Diese Auswahl bzw. Anzeige erscheint nur, wenn die Statistikvorkonfiguration ein Diagramm vorsieht bzw. ermöglicht. Alle generell verwendbaren Grafikgrößen werden durch den otrs-Admin in der SysConfig definiert. Beim Konfigurieren der Statistiken kann dann der Ersteller alle relevanten Formate vorselektieren.
- *Zeilensummierung*. Gibt an, ob die Statistik durch eine Spalte ergänzt wird, deren Zellen die Summe der jeweiligen Zeile darstellt.
- *Spaltensummierung*. Angabe, ob die Statistik durch eine Zeile ergänzt wird, deren Zellen die Summe der jeweiligen Spalte beinhaltet.
- *Cache*. Gibt an, ob die errechneten Statistiken im Dateisystem gecacht werden.
- *Gültig*. Ist eine vorkonfigurierte Statistik erst im Entstehen oder darf sie aus bestimmten Gründen temporär nicht mehr ausgeführt werden so wird das Element "Gültig" auf "ungültig" gesetzt. Ferner wird dadurch die Schaltfläche "Start" rechts unten am Ende des Block ausgeblendet. Ein Generieren der Statistik ist somit nicht möglich.
- *Erstellt*. Erstellungszeit der Statistik.
- *Erstellt von*. Agent, der die Statistik erstellt hat.
- *Geändert*. Letzte Änderungszeit der Statistik.
- *Geändert von*. Agent, der die Statistik zuletzt geändert hat.
- *X-Achse*. Diese Option erlaubt es dem Agenten, die X- und Y-Achsen zu vertauschen (Diese Funktion muss vom OTRS-Administrator freigeschaltet werden).
- Nach den allgemeinen Angaben sieht der Agent die Angaben zur eigentlichen Statistik. Hierbei kommen grundsätzlich zwei verschiedene Anzeigen zum Einsatz:
 - *Anzeige statischer Statistiken*. Statische Report-Generatoren können in die Statistik integriert werden (siehe unten).

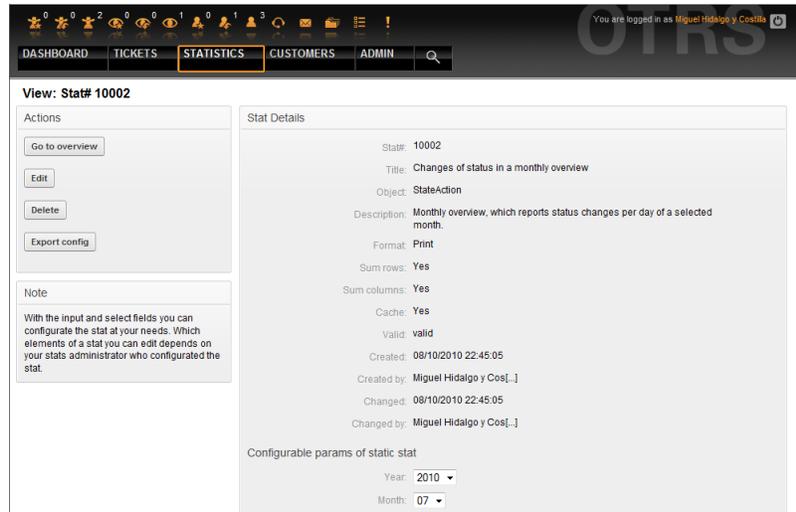


Abb. 19.4. Ansicht der statischen Statistik.

- *Anzeige dynamischer Statistiken* Dynamische Statistiken können auf zweierlei Weise angezeigt werden:
 - *Unveränderbare Einstellungen.* In diesem Fall hat der Ersteller der Statistik dem Agent, der eine Statistik generieren will, keinen Freiraum für Veränderungen gelassen.
 - *Veränderbare Einstellungen.* Bei derartigen Statistiken ist es dem Agent noch erlaubt die vorkonfigurierte Statistik zu verändern.

Um eine Statistik zu generieren, muss abschließend lediglich der "Start"-Button am rechten unteren Ende des Blocks gedrückt werden. Sollte einmal der "Start"-Button nicht sichtbar sein, kann dies zwei Ursachen haben:

1. die Statistik wurde auf ungültig gesetzt und damit deaktiviert.
2. die Statistik wurde nicht sauber konfiguriert und ist noch nicht lauffähig. Ist dies der Fall, findet man die nötigen Informationen im Notification-Bereich von OTRS (unterhalb der Navigationsleiste).

Bei falschen Einstellungen auf der Ansichtseite wird nach Drücken des "Start"-Button wieder die Ansichtseite gezeigt und im Notificationbereich darauf hingewiesen, welche Eingaben falsch waren.

Bearbeiten / Neu

Agenten mit Schreibrechten können existierende Report-Konfigurationen bearbeiten, indem sie die Bearbeitungsmaske des Statistikmoduls aufrufen. Alternativ können sie einen neuen Report erstellen. Die zugehörigen Masken können folgendermaßen aufgerufen werden:

1. Durch den Knopf "Bearbeiten" in der Statistikansicht.
2. Durch den "Neu"-Link im Statistikmenü der Navigationsleiste, oder durch den Knopf "Hinzufügen" in der Übersichtsseite.

Das Bearbeiten der Statistiken erfolgt in vier Schritten mit Hilfe eines Assistenten bzw. Wizards:

1. Allgemeine Angaben zur Statistik.
2. Festlegen des Elements für die X-Achse.

3. Festlegen der Wertereihen.

4. Festlegen der Einschränkungen des Reports.

Die Punkte 2-4 werden nur für dynamisch zusammengestellte Statistiken benötigt. Benutzt man eine statische Statistik ist man mit dem Einfügen der allgemeinen Angaben in Punkt 1. bereits fertig.

Für alle Eingabeoberflächen des Statistik-Moduls gilt: Im Anschluss an das jeweilige Eingabeformular findet der Benutzer Informationen zur Bedienung der Seite.

Sollte es dennoch zu fehlerhaften Eingaben kommen, wird die zuvor bearbeitete Oberfläche wieder geladen. Dabei wird die Seite um Informationen zur fehlerhaften Eingabe ergänzt. Diese Informationen findet man im Notificationbereich von OTRS. Erst nachdem ein Formular korrekt ausgefüllt wurde, gelangt man zur nächsten Eingabeoberfläche.

1. *Allgemeine Angaben.* Dies ist die erste Seite des Assistenten.

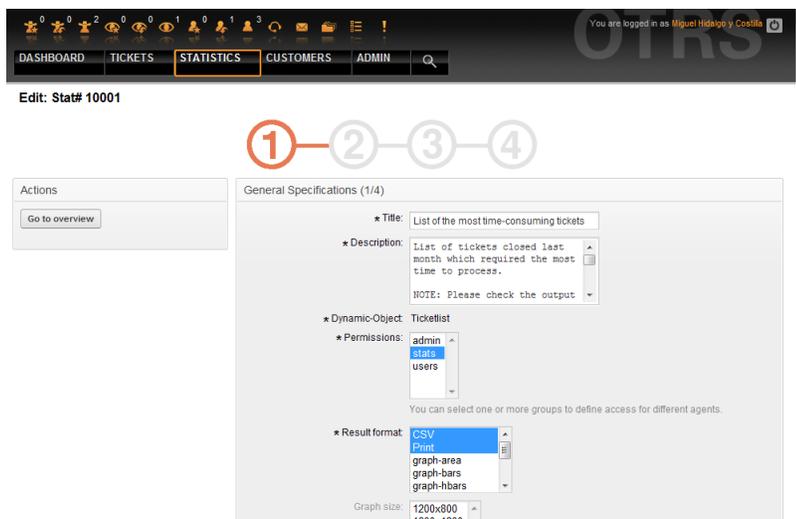


Abb. 19.5. Allgemeine Angaben zur Statistik bearbeiten.

Folgende allgemeine Angaben und Einstellungen der Statistik können geändert werden:

- *Titel.* Er sollte kurz und prägnant die Aufgabe der Statistik wiedergeben.
- *Beschreibung.* Hier werden Informationen eingetragen, die einen tieferen Einblick in die Aufgabenstellung, Art der Konfigurationsparameter, usw. erlauben.
- *Dynamisches Objekt.* Sollte die OTRS-Installation über mehrere dynamische Objekte verfügen, kann hier das gewünschte Objekt ausgewählt werden. Die Objekte sind jeweils auf die Anforderungen des entsprechenden Moduls ausgelegt.
- *Statische Datei.* An dieser Stelle kann eine gewünschte statische Datei ausgewählt werden. Diese Auswahl wird für gewöhnlich nicht erscheinen, da nur die statischen Dateien angezeigt werden, die noch keiner Statistik zugeordnet wurden! Wenn der Punkt "Statische Datei" angezeigt wird, ist es wichtig im Formular durch die Auswahl des Optionsfelds die gewünschte Generierungsart auszuwählen (Dynamisch durch ein dynamisches Objekt oder statisch durch eine Datei). Wird eine statische Datei ausgewählt entfallen die Eingabeoberflächen 2-4, da die statische Datei die nötigen Konfigurationen bereits mitbringt.

- *Rechtevergabe.* Durch die Rechtevergabe kann man regeln, welche Gruppen (und dadurch welche Agents) die vorkonfigurierten Statistiken später ansehen und generieren können. So können die verschiedenen Statistiken auf die verschiedenen Abteilungen und Arbeitsgruppen aufgeteilt werden, die diese benötigen. Eine Mehrfachauswahl ist dabei möglich.

Beispiel 1: Die Gruppe "stats" wurde ausgewählt. Der Report kann von allen Nutzern verwendet werden, die mindestens "ro"-Rechte für die Gruppe "stats" haben. Diese Berechtigung ist die Voreinstellung.

Beispiel 2: Es wurde eine Gruppe namens "sales" selektiert. Alle Benutzer die ro-Rechte in der Gruppe "sales" haben können nun diese Statistik im Ansichtsmodus sehen und generieren. Anderen Benutzern, die in der Lage sind Statistiken zu generieren wird dann diese Statistik nicht angeboten.

- *Format.* Ausgabeformat der Statistik. Hierbei stehen je nach Konfiguration folgende Ausgabeformate zur Verfügung:
 - CSV.
 - Drucken.
 - Liniendiagramm.
 - Balkendiagramm.
 - Balkendiagramm (horizontal).
 - Punktdiagramm.
 - Linienpunktdiagramm.
 - Flächendiagramm.
 - Tortendiagramm.
- *Grafikgröße.* Hier kann ausgewählt werden, in welchen Größen (Pixel) die Diagramme ausgegeben werden dürfen. Diese Auswahl ist nur sinnvoll, wenn unter dem Punkt "Format" ein grafisches Ausgabeformat ausgewählt wurde. Alle generell verwendbaren Grafikgrößen werden durch den OTRS-Admin in der SysConfig definiert. Bei der Konfiguration der Statistiken kann der Ersteller alle relevanten Formate vorselektieren.
- *Zeilensummierung.* Wird im "Ja" ausgewählt, wird die Statistik durch eine Spalte ergänzt, deren Zellen die Summe der jeweiligen Zeile darstellt.
- *Spaltensummierung.* Angabe, ob die Statistik durch eine Zeile ergänzt wird, deren Zellen die Summe der jeweiligen Spalte beinhaltet.
- *Cache.* Wird im "Ja" ausgewählt, werden die errechneten Statistiken im Dateisystem gecacht. Dies spart Rechenleistung und Zeit, sollte genau diese Statistik wieder aufgerufen werden. Diese Funktion ist jedoch nur sinnvoll, wenn sichergestellt ist, dass sich der Inhalt der Statistik nicht mehr ändert. Dies ist genau zu prüfen, bevor man die Cachingfunktion nutzt.

Caching wird automatisch verhindert, wenn die Statistik keinerlei Zeitangaben enthält (es kommen laufend neue Daten hinzu) oder wenn eine Zeitangabe in der Zukunft liegt.

Sollte eine Statistik zwischenzeitlich wieder bearbeitet werden, werden alle gecachten Daten wieder gelöscht.

- *Gültig*. Ist eine vorkonfigurierte Statistik erst im Entstehen oder darf sie aus bestimmten Gründen temporär nicht mehr ausgeführt werden so wird das Element "Gültig" auf "ungültig" gesetzt. Ferner wird dadurch die Schaltfläche "Start" rechts unten am Ende des Block ausgeblendet. Ein Generieren der Statistik ist somit nicht möglich.

2. *Definition des Elements für die X-Achse*. Hier wird eingestellt welches Element für die Darstellung der X-Achse verwendet wird, bzw. bei Tabellen, welche Spaltenbezeichnung die Statistik bekommt.

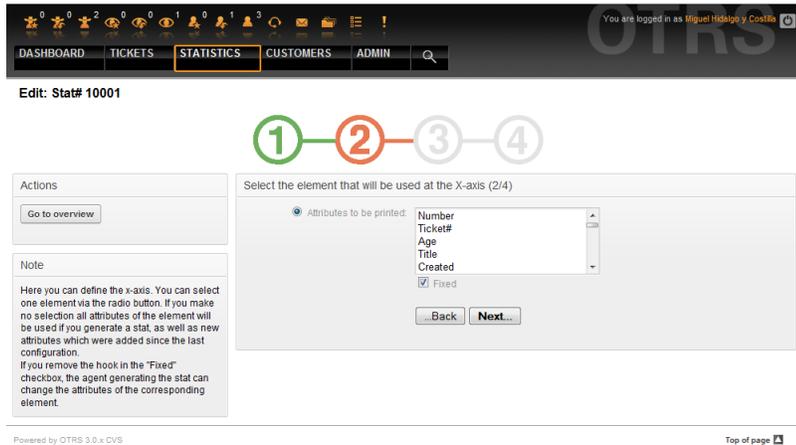


Abb. 19.6. Definition des Elements für die X-Achse.

Als erstes wählt man ein Element per Optionsfeld aus. Anschließend müssen zwei oder mehr Attribute des Elements selektiert werden. Werden keine Attribute des Elements ausgewählt, werden alle Attribute verwendet, auch solche die nach der Konfiguration der Statistik hinzukommen.

Entfernt man den Haken "Fixiert" in der Checkbox, kann der Agent, der die Statistik erstellt, die Attribute des entsprechenden Elements in der "Ansehen"-Oberfläche verändern.

Eine Besonderheit stellen Zeitelemente dar, hier ist die Zeitspanne und die Skalierung anzugeben. Die Art und Anzahl der Elemente ergibt sich durch das verwendete dynamische Objekt und ist von Objekt zu Objekt verschieden.

Wurden alle Eingaben richtig getätigt, gelangt man nach der Auswahl des "Weiter" Schalters zu dem Formular "Wertereihen". Ebenfalls gibt es die Möglichkeit, einen Bearbeitungsschritt zurück zu gehen und nochmals die Oberfläche "Allgemeine Angaben" zu bearbeiten.

3. *Wertereihen*.

Im dritten Schritt der Vorkonfiguration einer Statistik werden die Wertereihen der Statistik festgelegt. Sie bilden später die einzelnen Grafen des Diagramms (grafische Darstellung) oder die einzelnen Reihen (tabellarische Darstellung).

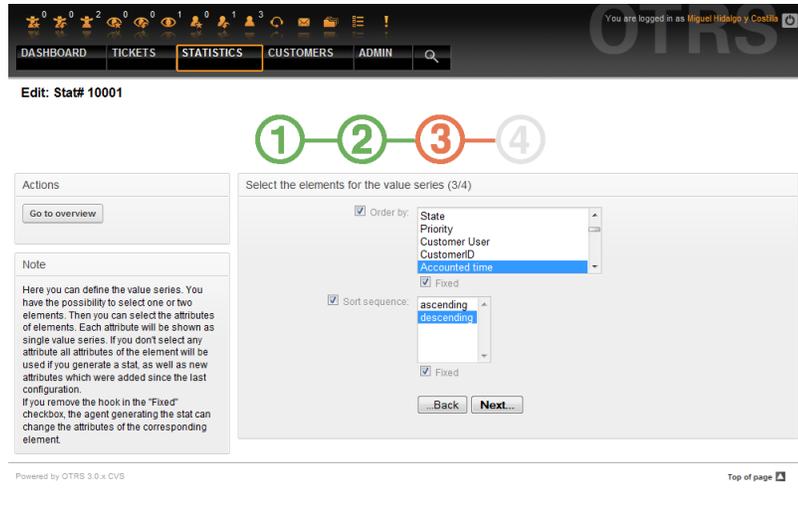


Abb. 19.7. Festlegen der Wertereihen.

Wird ein Element selektiert, entspricht jedes ausgewählte Attribut einer Wertereihe.

Beispiel 19.1. Festlegen einer Wertereihe bei einem Element

Element Queue:

- Wertereihe 1 = Raw
- Wertereihe 2 = Junk
-

Werden zwei Elemente selektiert, wird jedes ausgewählte Attribut des ersten Elements kombiniert mit einem Attribut des zweiten Elements zu einer Wertereihe.

Beispiel 19.2. Festlegen einer Wertereihe bei zwei Elementen

Element 1 Queue, Element 2 Status:

- Wertereihe 1 = Raw - offen
- Wertereihe 2 = Raw - erfolgreich geschlossen
- Wertereihe 3 = Junk - offen
- Wertereihe 4 = Junk - erfolgreich geschlossen

Die Auswahl von drei oder mehr Elementen ist nicht sinnvoll und wird deshalb durch eine Fehlermeldung verhindert.

Ferner gelten noch die gleichen Bedingungen zur Auswahl der Attribute und zur "Fixiert" Checkbox wie bei der "X-Achsen" Auswahl.

- Werden keine Attribute des Elements ausgewählt, werden alle Attribute verwendet, auch solche die nach der Konfiguration der Statistik hinzukommen.

- Entfernt man den Haken "Fixiert" in der Checkbox, kann der Agent der die Statistik erstellt, die Attribute des entsprechenden Elements verändern.
4. *Festlegen der Einschränkungen.* Vierter und letzter Schritt bei der Vorkonfiguration ist das Festlegen der Einschränkungen. Hier ist es möglich die Ergebnisse einer Statistik durch die Angabe von Bedingungen einzuschränken. Vergleichbar ist dies mit Eingaben in eine Such-Oberfläche.

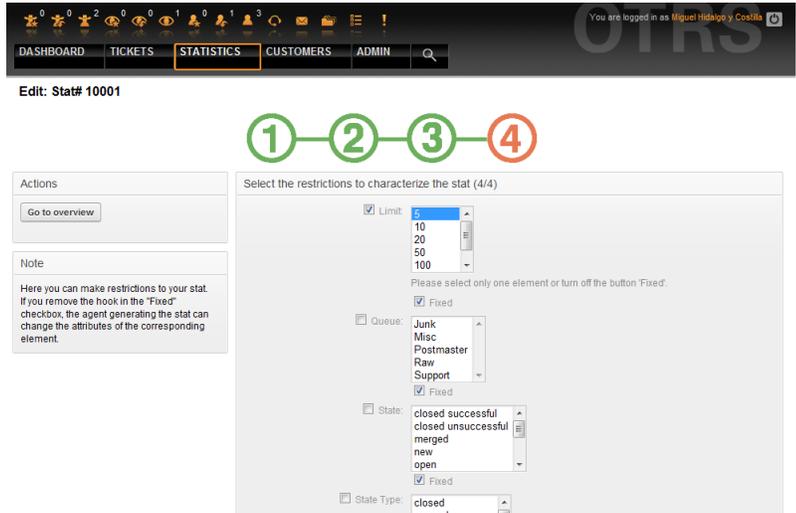


Abb. 19.8. Festlegen der Einschränkungen.

Nachdem man alle Einschränkungen getätigt hat, drückt man den Button "Abschließen". Dadurch beendet man die Vorkonfiguration der Statistik und gelangt in die "Ansehen" Oberfläche.

Importieren

Durch die Auswahl des "Import"-Menüpunkts im Statistikmenü der Navigationsleiste oder durch den "Import"-Knopf in der Übersichtsmaske gelangt man in die Importoberfläche ("rw"-Rechte erforderlich).

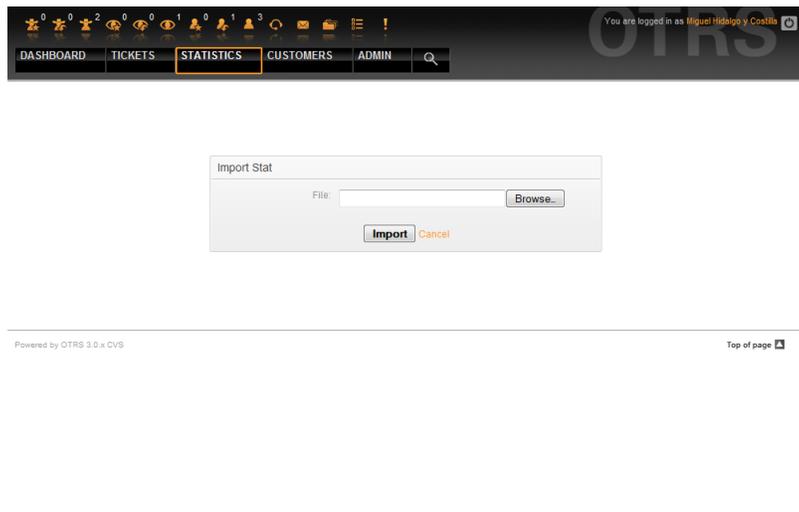


Abb. 19.9. Die Import-Oberfläche.

Durch sie können Statistiken importiert werden. Dies ist im Zusammenhang mit der Exportfunktion des Moduls eine sehr hilfreiche Funktionalität. So können Statistiken bequem auf Testsystemen entworfen und getestet werden, um sie später auf dem Produktivsystem zu importieren.

Der Import geschieht dabei ganz einfach durch einen Dateiapload. Anschließend gelangt man automatisch in die Detailansicht der importierten Statistik.

Verwaltung des Statistik-Moduls durch den OTRS-Administrator

In diesem Abschnitt werden die Aufgaben und Möglichkeiten des OTRS-Administrators in Bezug auf das Statistikmodul angesprochen.

Rechte, Gruppen und Queues

Durch die Installation des Statistikmoduls werden keine neuen Queues oder/und Gruppen angelegt.

Durch die Standardkonfiguration der Modulregistrierung haben alle Agenten, die Rechte in der "stats"-Gruppe haben, Zugriff auf das Statistikmodul.

Zugangsmöglichkeiten aufgrund der Rechte:

- *rw*. Darf Statistiken konfigurieren.
- *ro*. Darf vorkonfigurierte Statistiken generieren.

Ob nun dem jeweiligen Agenten, der später vorkonfigurierte Statistiken generieren darf, *ro*-Rechte in der "stats" Gruppe zugewiesen werden oder in der Modulregistrierung der SysConfig die dem Agent entsprechenden Gruppen ergänzt werden, bleibt dem OTRS-Administrator und dessen Anforderungen überlassen.

Die SysConfig (Verwaltung der Systemkonfiguration)

Die SysConfig-Gruppen `Framework:Core::Stats`, `Framework:Core::Stats::Graph` und `Framework:Frontend::Agent::Stats` enthalten alle Konfigurationsparameter zur grundlegenden Einstellung des Statistik-Moduls. Weiterhin regelt der Konfigurationsparameter `$Self->{'Frontend::Module'}->{'AgentStats'}` die Anordnung und Registrierung der Module und Icons innerhalb des Statistik-Moduls selbst.

Verwaltung des Statistik-Moduls durch den System-Administrator

Grundsätzlich ist bei der Bedienung, Konfiguration und Wartung des Statistikmoduls kein System-Administrator notwendig. An dieser Stelle sollen nur einige Hintergrundinformationen für den System-Administrator gegeben werden.

Anmerkung

Pfadangaben beziehen sich auf Unterverzeichnisse des OTRS-Homeverzeichnisses (meist `/opt/otrs`).

Datenbanktabelle

Alle Statistikkonfigurationen werden in XML umgesetzt und verwaltet. Dies bedeutet unter anderem, dass alle Statistikkonfigurationen in der Datenbanktabelle "xml_storage" gespeichert werden. Diese Tabelle wird auch von anderen Modulen benutzt, deren Inhalte im XML-Format dargestellt werden.

Auflistung aller Dateien

Folgende Dateien sind für die fehlerfreie Funktion des Statistikmoduls notwendig.

- Kernel/System/Stats.pm
- Kernel/Modules/AgentStats.pm
- Kernel/System/CSV.pm
- Kernel/Output/HTML/Standard/AgentStatsOverview.dtl
- Kernel/Output/HTML/Standard/AgentStatsDelete.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditSpecification.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditRestrictions.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditXaxis.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditValueSeries.dtl
- Kernel/Output/HTML/Standard/AgentStatsImport.dtl
- Kernel/Output/HTML/Standard/AgentStatsPrint.dtl
- Kernel/Output/HTML/Standard/AgentStatsView.dtl
- Kernel/System/Stats/Dynamic/Ticket.pm
- bin/otrs.GenerateStats.pl

Caching

Bei der Vorkonfiguration von Statistiken kann eingestellt werden, ob die Ergebnisse einer Statistik gecached werden oder nicht. Statistikergebnisse, die in den Cache geschrieben werden sollen, werden als Datei im Verzeichnis `var/tmp` der OTRS-Installation gespeichert (meist `/opt/otrs/var/tmp`).

Alle gecachten Statistiken sind dabei durch das Präfix "Stats" zu erkennen.

Sollten diese Dateien verloren gehen, ist dies nicht weiter schlimm. Beim nächsten Aufruf dieser Statistik kann das Statistikmodul diese Datei nicht mehr finden und muss deshalb die Statistik neu errechnen, was etwas Zeit beanspruchen kann.

otrs.GenerateStats.pl

Diese Datei befindet sich im Verzeichnis `bin/`. Durch sie ist es möglich, Statistiken auf der Kommandozeile zu generieren.

Exemplarisch soll hierzu folgender Kommandozeilenaufruf angegeben werden:

```
bin> perl otrs.GenerateStats.pl -n 10004 -o /output/dir
```

Skript 19.1. Generieren einer Statistik auf der Kommandozeile.

Dabei wird eine Statistik aus der Statistikkonfiguration "Stat# 10004" generiert und als csv in das Verzeichnis /output/dir gespeichert.

Ferner kann die generierte Statistik auch per Email versandt werden. Mehr Informationen erhält man wie in Skript 19.2 geschildert.

```
bin> perl otrs.GenerateStats.pl --help
```

Skript 19.2. Aufruf der Hilfe von otrs.GenerateStats.pl.

Automatisiertes Generieren von Statistiken - Cron

Sinnvoll ist das Generieren von Statistiken per Hand über die Kommandozeile natürlich nicht unbedingt, da das Statistikmodul eine geeignete grafische Oberfläche bietet. Interessant wird die Generierung von Statistiken per Kommandozeile aber in Kombination mit einem Cronjob.

Folgendes Szenario ist denkbar: Ein Abteilungsleiter wünscht pünktlich zum ersten eines Monats eine Statistik zum abgelaufenen Monat. Durch die Kombination Cronjob und Kommandozeilenaufruf ist man in der Lage automatisiert zu Beginn eines Monats dem Abteilungsleiter die gewünschte Statistik per Email zukommen zu lassen.

Statische Statistiken

Das Statistikmodul bietet die Möglichkeit statische Statistiken zu generieren. Für jede statische Statistik gibt es eine Datei, in der die Inhalte der Statistik genau definiert sind.

Vorteil dieser Art von Statistik ist, dass dadurch sehr komplexe Statistiken erzeugt werden können. Der Nachteil ist wiederum die mangelnde Flexibilität dieser Statistiken.

Gespeichert werden diese Dateien unter folgendem Pfad: `Kernel/System/Stats/Static/`.

Wiederverwerten alter statischer Statistiken

Bereits OTRS 1.3 und 2.0 boten die Möglichkeit Statistiken zu generieren. Eine Vielzahl der Statistiken, die für OTRS-Versionen 1.3 und 2.0 speziell auf Kundenwunsch entwickelt wurden, können wiederverwendet werden.

Diese Dateien müssen hierzu lediglich aus dem Pfad `Kernel/System/Stats/` in den Pfad `Kernel/System/Stats/Static/` verschoben werden. Ferner muss die Paketangabe des jeweiligen Skripts um `::Static` ergänzt werden.

Das folgende Beispiel zeigt die Erweiterung des Paketnamens.

```
package Kernel::System::Stats::AccountedTime;
```

```
package Kernel::System::Stats::Static::AccountedTime;
```

Standardstatistiken

"Man muss das Rad nicht immer neu erfinden."

Deshalb liefert das Statistikmodul automatisch Standardstatistiken mit. Statistiken, die für alle OTRS-Benutzer interessant sind, werden zukünftig als Defaultstatistik im Statistik-Modul-Paket ergänzt. Gespeichert werden die Defaultstatistiken im XML-Format des Statistikmoduls im Verzeichnis `scripts/test/sample/`.

Kapitel 20. Zusätzliche Applikationen

Zusätzlich zum OTRS-Framework haben Sie die Möglichkeit, weitere Applikationen aus einem Online-Verzeichnis über das Admin-Interface (Paketmanager) zu installieren. Dieser lädt Applikationen von einem Online-Verzeichnis herunter und löst die Paketabhängigkeiten auf. Es ist zudem möglich, Pakete in Form lokaler Dateien zu installieren, z. B. für den Fall, dass Ihr OTRS keine Internetverbindung hat.

FAQ

Mit Hilfe des FAQ-Systems kann eine Wissensdatenbank aufgebaut und verwaltet werden. Einträge lassen sich nur intern oder auch extern, d.h. für alle Kunden oder komplett öffentlich, freischalten. Einträge können nach Sprache oder nach Kategorien erstellt und sortiert werden.

Kapitel 21. Leistungsverbesserung

Hier finden Sie eine Liste verschiedener Techniken der Leistungssteigerung Ihrer OTRS-Installation, einschließlich Konfiguration, Programmierung, Speichernutzung und mehr.

OTRS

Es gibt verschiedene Ansätze zur Leistungssteigerung von OTRS.

TicketIndexModule

Zur Verfügung stehen zwei Backend-Module für den Ticket Index:

- `Kernel::System::Ticket::IndexAccelerator::RuntimeDB` (Standard), generiere jede Queue-Ansicht dynamisch aus der Ticket Tabelle. Sie werden keine Probleme mit der Leistung bekommen bis zu etwa 60.000 Tickets (oder 6000 offenen) in Ihrem System.
- `Kernel::System::Ticket::IndexAccelerator::StaticDB`, das leistungsfähigste Modul. Es sollte ab 80.000 Tickets oder mehr als 6000 offenen eingesetzt werden. Benutzt eine extra `ticket_index` Tabelle, arbeitet wie eine Ansicht (View). Führen Sie `bin/otrs.RebuildTicketIndex.pl` zum erstmaligen Aufbau des Index aus.

Sie können den `IndexAccelerator` per `SysConfig` einstellen.

TicketStorageModule

Es stehen zwei Module für das Speichern der Tickets und Artikel bereit:

- `Kernel::System::Ticket::ArticleStorageDB` (Standard), speichert Anhänge u. A. in der Datenbank. Merke: Benutzen Sie diese Option nicht für größere Systeme.

Pro: Ist der Benutzer, unter dem der Webserver läuft, nicht der Benutzer 'otrs', können Sie mit diesem Modul Dateiberechtigungsprobleme vermeiden.

Contra: Es ist nicht wirklich ratsa., Anhänge in Ihrer Datenbank zu speichern. Achten Sie darauf, dass Ihre Datenbank das kann. Für MySQL setzen Sie in dessen Konfiguration bspw. "`set-variable = max_allowed_packet=8M`", um 8 MB große Objekte zu speichern (Standard ist 2M).

- `Kernel::System::Ticket::ArticleStorageFS`, speichert Anhänge u. A. im lokalen Filesystem ab. Merke: Benutzen Sie dies für große Installationen.

Pro: Schneller!

Contra: Der Benutzer, unter dem der Webserver läuft, sollte der Benutzer 'otrs' sein (Dateisystemberechtigungen!). Wenn Sie mehrere OTRS-Frontendserver haben, müssen Sie sicherstellen, dass das Dateisystem gemeinsam genutzt wird. Sie können es z. B. auf ein NFS-Share oder vorzugsweise ein SAN oder eine vergleichbare Lösung legen.

Merke: Sie können während des Betriebs von einem Backend auf das andere wechseln. Stellen Sie dazu das Backend in der `SysConfig` um, und führen Sie `otrs.ArticleStorageSwitch.pl` aus, um die Artikel aus der Datenbank ins Dateisystem zu überführen oder umgekehrt. Sie können die `-s` und `-d` Option

verwenden, um das Quell- und Zielbackend anzugeben. Der Prozess kann eine ganze Weile dauern, je nach Menge der Artikel sowie System- und Netzwerkleistung.

```
shell> bin/otrs.ArticleStorageSwitch.pl -s ArticleStorageDB -d  
ArticleStorageFS
```

Skript 21.1 Wechsel des TicketStorage-Backends von Datenbank zu Dateisystem.

Tickets archivieren

Da OTRS als revisionssicheres System betrieben werden kann, ist das Löschen von geschlossenen Tickets möglicherweise nicht empfehlenswert. Daher haben wir eine Funktion implementiert, mit der Sie Tickets archivieren können.

Kokret ist darunter zu verstehen, dass Tickets, die bestimmte Kriterien erfüllen, als "archiviert" markiert werden. Diese Tickets werden dann bei regulären Suchabfragen oder von GenericAgent-Jobs nicht mehr erfasst. Somit muss sich das System mit einer großen Ticketmenge nicht mehr befassen, weil dann nur noch die "aktuellsten" Tickets betrachtet werden. Das kann auf großen Systemen eine signifikante Performanceverbesserung bewirken.

Befolgen Sie folgende Schritte, um die Archivierungsfunktion zu nutzen:

1. Archivsystem in der SysConfig aktivieren

Wählen Sie in der SysConfig die Gruppe `Ticket` aus. In `Core::Ticket` finden Sie die Option `Ticket::ArchiveSystem`, die standardmäßig auf "Nein" steht. Ändern Sie diese auf "Ja" und speichern Sie die Änderung ab.

2. Anlegen eines GenericAgent-Jobs

Wählen Sie im Administrationsbereich den "GenericAgent" aus und legen Sie dort einen neuen Job an.

a. Job-Einstellungen

Geben Sie dem Job einen geeigneten Namen und angemessene Optionen.

b. Ticket-Filter

Der Ticketfilter ist eine Ticketsuche, die Tickets nach bestimmten Kriterien auswählt. Es könnte empfehlenswert sein, nur Tickets zu archivieren, die seit einigen Monaten im Status "geschlossen" sind.

c. Ticket-Aktion

Im Abschnitt "Ticket-Aktion" werden Sie eine Aktion "Ausgewählte Tickets archivieren" finden. Wählen Sie dort "Tickets archivieren" aus.

d. Job speichern

Am Ende der Seite finden Sie einen Knopf zum Speichern des Jobs.

e. Betroffene Tickets

Das System wird dann alle Tickets anzeigen, die beim Ausführen des GenericAgent-Jobs archiviert werden.

3. Ticketsuche

Wenn Sie nun nach Tickets suchen, werden standardmäßig nur Tickets gefunden, die nicht archiviert sind. Wenn Sie auch in archivierten Tickets suchen wollen, fügen Sie "Archivsuche" zu Ihren Suchkriterien hinzu.

Datenbank

Einstellungen sind immer spezifisch für die jeweils eingesetzte Datenbank. Bei Problemen lesen Sie die Dokumentation und fragen Sie Ihren Datenbankadministrator.

MySQL

Wenn Sie den Tabellentyp MyISAM (Standard) benutzen, und einen großen Teil einer Tabelle gelöscht haben, oder wenn Sie sehr viele Änderungen an einer Tabelle mit Zeilen variabler Länge vorgenommen haben (Tabellen mit VARCHAR, BLOB oder TEXT Spalten), sollten Sie die Datendateien mit dem "optimize" Kommando behandeln.

Dies bietet sich an, wenn MySQL viel CPU Zeit braucht. Optimieren Sie die Tabellen ticket, ticket_history und article.

```
shell$ mysql -u user -p database
mysql$ optimize table ticket;
mysql$ optimize table ticket_history;
mysql$ optimize table article;
```

Skript 21.2. Optimierung von Datenbanktabellen.

PostgreSQL

PostgreSQL konfigurieren Sie am besten in der postgresql.conf Datei in Ihrem PostgreSQL Datenverzeichnis. Hier gibt es Hilfe dazu:

- <http://www.revsys.com/writings/postgresql-performance.html> [http://www.revsys.com/writings/postgresql-performance.html]
- <http://varlena.com/GeneralBits/Tidbits/perf.html> [http://varlena.com/GeneralBits/Tidbits/perf.html]
- http://varlena.com/GeneralBits/Tidbits/annotated_conf_e.html [http://varlena.com/GeneralBits/Tidbits/annotated_conf_e.html]

Ist die Leistung immer noch nicht genügend, empfehlen wir, Fragen auf der "PostgreSQL Performance Mailing Liste" (<http://www.postgresql.org/community/lists/> [http://www.postgresql.org/community/lists/]) zu stellen. Die Teilnehmer der PostgreSQL Liste sind sehr freundlich und können wahrscheinlich helfen.

Webserver

Natürlich empfehlen wir mod_perl 2.0 (<http://perl.apache.org/>). Es ist sehr viel schneller (etwa um den Faktor 100) als pures CGI, braucht aber auch mehr Speicher. Ihr httpd wird mit mod_perl also größer sein.

Persistente Datenbankverbindungen

Sie können die Datenbankverbindung bereits beim Start des Webservers herstellen lassen. Dies spart ebenso Zeit (siehe auch README.webserver).

Vorgeladene Module - startup.pl

Nutzen Sie das Start Skript `scripts/apache2-perl-startup.pl`, um die Perl Module vorzuladen (siehe README.webserver). Dadurch wird der Webserver schneller und braucht weniger Speicher.

Perl Module bei Änderung neu laden

Standardmäßig wird `Apache::Reload` in `scripts/apache2-httpd.include.conf` eingesetzt. Deaktivieren Sie es und die Geschwindigkeit steigt um etwa 8%. Ab nun müssen Sie den Webserver neu starten, wenn Sie irgendetwas ändern! Wichtig: Dies hat zur Folge, dass der OTRS-Paket-Manager nicht mehr über das Web-Interface bedient werden kann sondern nur noch auf der Kommandozeile - `bin/otrs.PackageManager.pl`.

Die richtige Strategie wählen

Bei wirklich großen Installationen (über 1000 neue Tickets am Tag, über 40 Agenten) ist es eine sehr gute Idee, den Artikel "Choosing the Right Strategy" (in englisch) zu lesen (<http://perl.apache.org/docs/1.0/guide/strategy.html>).

mod_gzip/mod_deflate

Falls Ihre Bandbreite ein wenig schmal sein sollte, benutzen Sie `mod_deflate` für Apache2. Eine HTML-Seite von 45k wird `mod_gzip/mod_deflate` auf etwa 7k zusammendrücken. Allerdings wird dadurch die Last auf dem Server erhöht.

Kapitel 22. Datensicherung

In diesem Kapitel wird beschrieben, wie alle relevanten Daten der OTRS-Installation gesichert und wieder hergestellt werden können.

Datensicherung

Bei einem Backup gibt es zwei Arten von Datensicherung, die Applikation (z. B. `/opt/otrs/`) und die Datenbank.

Um Backups zu vereinfachen, wird ein `scripts/backup.pl` mitgeliefert, das alle benötigten Komponenten sichert.

```
linux:/opt/otrs# cd scripts/  
linux:/opt/otrs/scripts# ./backup.pl --help  
backup.pl <Revision 1.1> - backup script  
Copyright (c) 2001-2005 Martin Edenhofer <martin@otrs.org>  
usage: backup.pl -d /data_backup/ [-c bzip2|gzip] [-r 30] [-t  
nofullbackup]  
linux:/opt/otrs/scripts#
```

Skript 22.1. Aufruf der Hilfe des Backup-Mechanismus.

Ein Backup kann also z. B. mit folgendem Befehl erstellt werden:

```
linux:/opt/otrs/scripts# ./backup.pl -d /backup/  
Backup /backup//2010-09-07_14-28/Config.tar.gz ... done  
Backup /backup//2010-09-07_14-28/Application.tar.gz ... done  
Dump MySQL rdbms ... done  
Compress SQL-file... done  
linux:/opt/otrs/scripts#
```

Abb. 22.2. Erstellen einer Datensicherung.

Alle Daten wurden in das Verzeichnis `/backup/2010-09-07_14-28/` gesichert und dort, getrennt nach Art der Daten, in einzelne `.tar.gz`-Dateien gespeichert.

```
linux:/opt/otrs/scripts# ls /backup/2010-09-07_14-28/  
Application.tar.gz Config.tar.gz DatabaseBackup.sql.gz  
linux:/opt/otrs/scripts#
```

Skript 22.3. Prüfung der gesicherten Dateien.

Wiederherstellung der Daten

Um ein Backup wieder einzuspielen, müssen die Applikation (z. B. nach `/opt/otrs/`) und die Datenbank wieder hergestellt werden.

Um Backups einspielen zu können, wird ein `scripts/restore.pl` mitgeliefert, das die benötigten Komponenten wieder zurück sichert. Es unterstützt MySQL und PostgreSQL.

```
linux:/opt/otrs/scripts# ./restore.pl --help
restore.pl <Revision 1.1> - restore script
Copyright (c) 2001-2005 Martin Edenhofer <martin@otrs.org>
usage: restore.pl -b /data_backup/<TIME>/ -d /opt/otrs/
linux:/opt/otrs/scripts#
```

Skript 22.4. Aufruf der Hilfe des Restore-Mechanismus.

Daten, die im Verzeichnis /backup/2005-09-12_14-28/ gesichert wurden, können bei einer OTRS-Installation in /opt/otrs folgendermaßen zurückgespielt werden.

```
linux:/opt/otrs/scripts# ./restore.pl -b /backup/2010-09-07_14-28 -d /
opt/otrs/
Restore /backup/2010-09-07_14-28//Config.tar.gz ...
Restore /backup/2010-09-07_14-28//Application.tar.gz ...
create MySQL
decompresses SQL-file ...
cat SQL-file into MySQL database
compress SQL-file...
linux:/opt/otrs/scripts#
```

Skript 22.5. Wiederherstellung des Systems.

Anhang A. Weitere Quellen

Wir unterstützen Sie mit den aktuellsten Informationen über OTRS und geben Ihnen die Möglichkeit, uns mit Feedback zu versorgen.

Homepage OTRS.org

Sie finden unsere Projekt-Homepage unter <http://www.otrs.org/>.

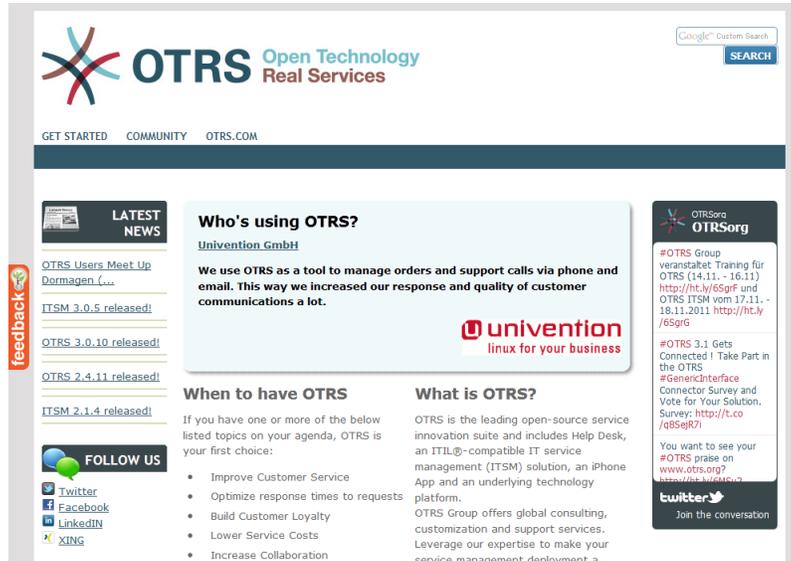


Abb. A-1. OTRS-Homepage.

Mailinglisten

Die folgende Tabelle gibt einen Überblick darüber, welche Community-Mailinglisten es derzeit für OTRS gibt und wie diese zu erreichen sind:

Tabelle A.1. Mailinglisten

Name	Beschreibung	Homepage
announce@otrs.org	Ankündigungen zu OTRS (neue Versionen, Updates, ...) mit wenig Verkehr.	http://lists.otrs.org/cgi-bin/listinfo/announce [http://lists.otrs.org/cgi-bin/listinfo/announce]
otrs@otrs.org	Benutzer-Fragen und Diskussionen rund um OTRS (englischsprachig) mit mittel bis viel Verkehr.	http://lists.otrs.org/cgi-bin/listinfo/otrs [http://lists.otrs.org/cgi-bin/listinfo/otrs]
ç	Benutzer-Fragen und Diskussionen rund um OTRS (deutschsprachig) mit mittel bis viel Verkehr.	http://lists.otrs.org/cgi-bin/listinfo/otrs-de [http://lists.otrs.org/cgi-bin/listinfo/otrs-de]

Name	Beschreibung	Homepage
dev@otrs.org	Mailingliste für die Entwickler von OTRS mit wenig Verkehr.	http://lists.otrs.org/cgi-bin/listinfo/dev [http://lists.otrs.org/cgi-bin/listinfo/dev]
doc-de@otrs.org	Deutsche Diskussion über die OTRS-Dokumentation mit wenig Verkehr.	http://lists.otrs.org/cgi-bin/listinfo/doc-de [http://lists.otrs.org/cgi-bin/listinfo/doc-de]
i18n@otrs.org	Englische Mailingliste mit wenig Verkehr für Übersetzungs- und Lokalisierungsfragen. Wenn Sie OTRS übersetzen oder übersetzen wollen oder Probleme mit einer unserer Applikationen in einer internationalen Umgebung haben, sollten Sie sich in diese Liste eintragen.	http://lists.otrs.org/cgi-bin/listinfo/i18n [http://lists.otrs.org/cgi-bin/listinfo/i18n]
cvs-log@otrs.org	Benachrichtigungen über geänderte CVS-Dateien im Quellcode von OTRS mit sehr viel Verkehr.	http://lists.otrs.org/cgi-bin/listinfo/cvs-log [http://lists.otrs.org/cgi-bin/listinfo/cvs-log]

Um eine Liste zu abonnieren besuchen Sie folgenden Link: <http://lists.otrs.org/>.

Fehler berichten

Um uns Fehler in OTRS mitzuteilen, besuchen Sie uns unter: <http://bugs.otrs.org/>. Bitte unterscheiden Sie zwischen Fehlermeldungen und Konfigurationsproblemen. Letztere sind Probleme, die beim Einrichten eines Systems auftreten können, oder allgemeine Fragen zur Verwendung von OTRS. Fehlerberichte im Bugtracker sollten nur für Probleme im Quellcode von OTRS eingestellt werden, oder um Verbesserungsvorschläge zu unterbreiten. Alle Ihre Fehlerberichte und Verbesserungsvorschläge sind im Bugtracker herzlich willkommen.

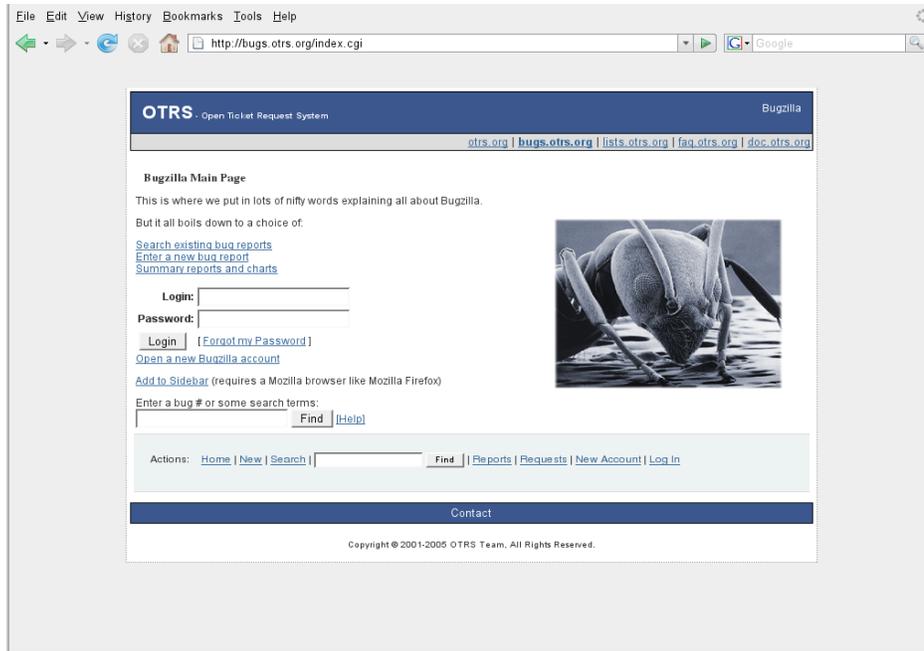


Abb. A-2. Fehlerverfolgungssystem.

Für Konfigurationsprobleme sollten Sie entweder den *kommerziellen Support von OTRS.com* [<http://www.otrs.com/en/support/>] oder die öffentlichen Mailinglisten verwenden.

Wenn Sie uns Fehler berichten, tragen Sie zur Verbesserung des Produkts bei. Wir freuen uns auf Ihre Mitwirkung!

Kommerzieller Support

Für Dienstleistungen (Support, Consulting, Entwicklung und Schulungen) können Sie die Firma hinter OTRS kontaktieren, die OTRS AG. Sie hat Niederlassungen in Deutschland, USA, Mexiko, Holland und weiteren Ländern. Schauen Sie sich die Website an, um Kontaktinformationen zu erhalten: <http://www.otrs.com/de/corporate-navigation/contact/> [<http://www.otrs.com/de/corporate-navigation/contact/>].

Anhang B. Referenz der Konfigurationsoptionen

Framework

Core

SecureMode

Description	Value
Description:	Disables the web installer (http://yourhost.example.com/otrs/installer.pl), to prevent the system from being hijacked. If set to "No", the system can be reinstalled and the current basic configuration will be used to pre-populate the questions within the installer script. If not active, it also disables the GenericAgent, PackageManager and SQL Box (to avoid the use of destructive queries, such as DROP DATABASE, and also to steal user passwords).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SecureMode'} = '0';</code>

Frontend::DebugMode

Description	Value
Description:	Enables or disable the debug mode over frontend interface.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::DebugMode'} = '0';</code>

ConfigLevel

Description	Value
Description:	Sets the configuration level of the administrator. Depending on the config level, some sysconfig options will be not shown. The config levels are in in ascending order: Expert, Advanced, Beginner. The higher the config level is (e.g. Beginner is the highest), the less likely is it that the user can accidentally configure the system in a way that it is not usable any more.
Group:	Framework

Description	Value
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ConfigLevel'} = '100';</code>

ProductName

Description	Value
Description:	Defines the name of the application, shown in the web interface, tabs and title bar of the web browser.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ProductName'} = 'OTRS';</code>

SystemID

Description	Value
Description:	Defines the system identifier. Every ticket number and http session string contain this ID. This ensures that only tickets which belong to your system will be processed as follow-ups (useful when communicating between two instances of OTRS).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SystemID'} = '10';</code>

FQDN

Description	Value
Description:	Defines the fully qualified domain name of the system. This setting is used as a variable, OTRS_CONFIG_FQDN which is found in all forms of messaging used by the application, to build links to the tickets within your system.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'FQDN'} = 'yourhost.example.com';</code>

HttpType

Description	Value
Description:	Defines the type of protocol, used by the web server, to serve the application. If https protocol will be used instead of plain http, it must be specified here. Since this has no affect on the web server's settings or behavior, it will not change the method of access to the application and, if it is wrong, it will not prevent you from logging into the application. This setting is used as a variable, OTRS_CONFIG_HttpType which is found in all forms of messaging used by the application, to build links to the tickets within your system.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'HttpType'} = 'http';</code>

ScriptAlias

Description	Value
Description:	Sets the prefix to the scripts folder on the server, as configured on the web server. This setting is used as a variable, OTRS_CONFIG_ScriptAlias which is found in all forms of messaging used by the application, to build links to the tickets within the system.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ScriptAlias'} = 'otrs/';</code>

AdminEmail

Description	Value
Description:	Defines the system administrator's email address. It will be displayed in the error screens of the application.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AdminEmail'} = 'admin@example.com';</code>

Organization

Description	Value
Description:	Company name for the customer web interface. Will also be included in emails as an X-Header.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Organization'} = 'Example Company';</code>

DefaultLanguage

Description	Value
Description:	Defines the default front-end language. All the possible values are determined by the available language files on the system (see the next setting).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultLanguage'} = 'en';</code>

DefaultUsedLanguages

Description	Value
Description:	Defines all the languages that are available to the application. The Key/Content pair links the front-end display name to the appropriate language PM file. The "Key" value should be the base-name of the PM file (i.e. de.pm is the file, then de is the "Key" value). The "Content" value should be the display name for the front-end. Specify any own-defined language here (see the developer documentation http://doc.otrs.org/ for more information). Please remember to use the HTML equivalents for non-ASCII characters (i.e. for the German oe = o umlaut, it is necessary to use the ö symbol).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultUsedLanguages'} = { 'ar_SA' => 'Arabic (Saudi Arabia)', 'bg' => 'Bulgarian (&#x0411;&#x044a;&#x043b;&#x0433;&#x0430;&#x0440;&#x0441;&#x0433; 'cs' => 'Czech (&#x010c;esky)', 'ct' => 'Catal&agrave;',</code>

Description	Value
	<pre>'da' => 'Dansk', 'de' => 'Deutsch', 'el' => 'Greek (&#x0395;&#x03bb;&#x03bb;&#x03b7;&#x03bd;&#x03b9;&#x03ba;&#x03a1; 'en' => 'English (United States)', 'en_CA' => 'English (Canada)', 'en_GB' => 'English (United Kingdom)', 'es' => 'Espa&ntilde;ol', 'es_MX' => 'Espa&ntilde;ol (M&eacute;xico)', 'et' => 'Eesti', 'fa' => 'Persian (&#x0641;&#x0627;&#x0631;&#x0633;&#x0649;)', 'fi' => 'Suomi', 'fr' => 'Fran&ccedil;ais', 'hi' => 'Hindi', 'hr' => 'Hrvatski', 'hu' => 'Magyar', 'it' => 'Italiano', 'ja' => 'Japanese (&#x65e5;&#x672c;&#x8a9e)', 'lt' => "Lietuvi\x{173} kalba", 'lv' => 'Latvijas', 'nb_NO' => 'Norsk bokm&aring;l', 'nl' => 'Nederlands', 'pl' => 'Polski', 'pt' => 'Portugu&ecirc;s', 'pt_BR' => 'Portugu&ecirc;s Brasileiro', 'ru' => 'Russian (&#x0420;&#x0443;&#x0441;&#x0441;&#x043a;&#x0438;&#x0439;)', 'sk_SK' => 'Slovak (Sloven&#x010d;ina)', 'sr_Cyrl' => "Serbian Cyrillic (\x{441}\x{440}\x{43f}\x{441}\x{43a}\x{438})", 'sr_Latn' => 'Serbian Latin (Srpski)', 'sv' => 'Svenska', 'tr' => 'T&uuml;rker&ccedil;e', 'uk' => 'Ukrainian (&#x0423;&#x043a;&#x0440;&#x0430;&#x0457;&#x043d;&#x0441;&#x0441; 'vi_VN' => 'Vietnam (Vi&#x0246;t Nam)', 'zh_CN' => 'Chinese (Sim.) (&#x7b80;&#x4f53;&#x4e2d;&#x6587;)', 'zh_TW' => 'Chinese (Tradi.) (&#x6b63;&#x9ad4;&#x4e2d;&#x6587;)' };</pre>

DefaultTheme

Description	Value
Description:	Defines the default front-end (HTML) theme to be used by the agents and customers. The default themes are Standard and Lite. If you like, you can add your own theme. Please refer the administrator manual located at http://doc.otrs.org/ .
Group:	Framework

Description	Value
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultTheme'} = 'Standard';</code>

DefaultTheme::HostBased

Description	Value
Description:	It is possible to configure different themes, for example to distinguish between agents and customers, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid theme on your system. Please see the example entries for the proper form of the regex.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'DefaultTheme::HostBased'} = { 'host1\\.example\\.com' => 'SomeTheme1', 'host2\\.example\\.com' => 'SomeTheme2' };</code>

CheckMXRecord

Description	Value
Description:	Makes the application check the MX record of email addresses before sending an email or submitting a telephone or email ticket.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckMXRecord'} = '1';</code>

CheckMXRecord::Nameserver

Description	Value
Description:	Defines the address of a dedicated DNS server, if necessary, for the "CheckMXRecord" look-ups.
Group:	Framework
SubGroup:	Core

Description	Value
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CheckMXRecord::Nameserver'} = 'ns.example.com';</code>

CheckEmailAddresses

Description	Value
Description:	Makes the application check the syntax of email addresses.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckEmailAddresses'} = '1';</code>

CheckEmailValidAddress

Description	Value
Description:	Defines a regular expression that excludes some addresses from the syntax check (if "CheckEmailAddresses" is set to "Yes"). Please enter a regex in this field for email addresses, that aren't syntactically valid, but are necessary for the system (i.e. "root@localhost").
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckEmailValidAddress'} = '^ (root@localhost admin@localhost)\$';</code>

CheckEmailInvalidAddress

Description	Value
Description:	Defines a regular expression that filters all email addresses that should not be used in the application.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckEmailInvalidAddress'} = '@(example)\.\.(...)\$';</code>

CGILogPrefix

Description	Value
Description:	Specifies the text that should appear in the log file to denote a CGI script entry.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CGILogPrefix'} = 'OTRS-CGI';</code>

DemoSystem

Description	Value
Description:	Runs the system in "Demo" mode. If set to "Yes", agents can change preferences, such as selection of language and theme via the agent web interface. These changes are only valid for the current session. It will not be possible for agents to change their passwords.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DemoSystem'} = '0';</code>

SwitchToUser

Description	Value
Description:	Allows the administrators to login as other users, via the users administration panel.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SwitchToUser'} = '0';</code>

NotificationSenderName

Description	Value
Description:	Specifies the name that should be used by the application when sending notifications. The sender name is used to build the complete display name for the notification master (i.e. "OTRS Notification

Description	Value
	Master" otrs@your.example.com). Notifications are messages such as en::Customer::QueueUpdate or en::Agent::Move.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'NotificationSenderName'} = 'OTRS Notification Master';</pre>

NotificationSenderEmail

Description	Value
Description:	Specifies the email address that should be used by the application when sending notifications. The email address is used to build the complete display name for the notification master (i.e. "OTRS Notification Master" otrs@your.example.com). You can use the OTRS_CONFIG_FQDN variable as set in your configuration, or choose another email address. Notifications are messages such as en::Customer::QueueUpdate or en::Agent::Move.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'NotificationSenderEmail'} = 'otrs@<OTRS_CONFIG_FQDN>';</pre>

System::Customer::Permission

Description	Value
Description:	Defines the standard permissions available for customers within the application. If more permissions are needed, you can enter them here. Permissions must be hard coded to be effective. Please ensure, when adding any of the afore mentioned permissions, that the "rw" permission remains the last entry.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'System::Customer::Permission'} = ['ro', 'rw'];</pre>

LanguageDebug

Description	Value
Description:	Debugs the translation set. If this is set to "Yes" all strings (text) without translations are written to STDERR. This can be helpful when you are creating a new translation file. Otherwise, this option should remain set to "No".
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'LanguageDebug'} = '0';</code>

Secure::DisableBanner

Description	Value
Description:	If enabled, the OTRS version tag will be removed from the HTTP headers.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Secure::DisableBanner'} = '0';</code>

StandardResponse2QueueByCreating

Description	Value
Description:	List of default StandardResponses which are assigned automatically to new Queues upon creation.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'StandardResponse2QueueByCreating'} = [</code> <code>''</code> <code>];</code>

Core::LinkObject

LinkObject::ViewMode

Description	Value
Description:	Determines the way the linked objects are displayed in each zoom mask.

Description	Value
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LinkObject::ViewMode'} = 'Simple';</code>

LinkObject::Type###Normal

Description	Value
Description:	Defines the link type 'Normal'. If the source name and the target name contain the same value, the resulting link is a non-directional one; otherwise, the result is a directional link.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LinkObject::Type'}->{'Normal'} = { 'SourceName' => 'Normal', 'TargetName' => 'Normal' };</code>

LinkObject::Type###ParentChild

Description	Value
Description:	Defines the link type 'ParentChild'. If the source name and the target name contain the same value, the resulting link is a non-directional one; otherwise, the result is a directional link.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LinkObject::Type'}->{'ParentChild'} = { 'SourceName' => 'Parent', 'TargetName' => 'Child' };</code>

LinkObject::TypeGroup###0001

Description	Value
Description:	Defines the link type groups. The link types of the same group cancel one another. Example: If ticket A is linked per a 'Normal' link with ticket B,

Description	Value
	then these tickets could not be additionally linked with link of a 'ParentChild' relationship.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::TypeGroup'}->{'0001'} = ['Normal', 'ParentChild'];</pre>

Core::Log

LogModule

Description	Value
Description:	Defines the log module for the system. "File" writes all messages in a given logfile, "SysLog" uses the syslog daemon of the system, e.g. syslogd.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LogModule'} = 'Kernel::System::Log::SysLog';</pre>

LogModule::SysLog::Facility

Description	Value
Description:	If "SysLog" was selected for LogModule, a special log facility can be specified.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LogModule::SysLog::Facility'} = 'user';</pre>

LogModule::SysLog::LogSock

Description	Value
Description:	If "SysLog" was selected for LogModule, a special log sock can be specified (on solaris you may need to use 'stream').

Description	Value
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::SysLog::LogSock'} = 'unix';</code>

LogModule::SysLog::Charset

Description	Value
Description:	If "SysLog" was selected for LogModule, the charset that should be used for logging can be specified.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::SysLog::Charset'} = 'iso-8859-1';</code>

LogModule::LogFile

Description	Value
Description:	If "file" was selected for LogModule, a logfile must be specified. If the file doesn't exist, it will be created by the system.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::LogFile'} = '/tmp/otrs.log';</code>

LogModule::LogFile::Date

Description	Value
Description:	Adds a suffix with the actual year and month to the OTRS log file. A logfile for every month will be created.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'LogModule::LogFile::Date'} = '0';</code>

Core::MIME-Viewer

MIME-Viewer###application/excel

Description	Value
Description:	Specifies the path to the converter that allows the view of Microsoft Excel files, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/excel'} = 'xlhtml';</code>

MIME-Viewer###application/msword

Description	Value
Description:	Specifies the path to the converter that allows the view of Microsoft Word files, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/msword'} = 'wvWare';</code>

MIME-Viewer###application/pdf

Description	Value
Description:	Specifies the path to the converter that allows the view of PDF documents, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/pdf'} = 'pdftohtml -stdout -i';</code>

MIME-Viewer###text/xml

Description	Value
Description:	Specifies the path to the converter that allows the view of XML files, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'MIME-Viewer'}->{'text/xml'} = '<OTRS_CONFIG_Home>/scripts/tools/xml2html.pl';</pre>

Core::MirrorDB

Core::MirrorDB::DSN

Description	Value
Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, specify the DSN to this database.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Core::MirrorDB::DSN'} = 'DBI:mysql:database=mirrordb;host=mirrordbhost';</pre>

Core::MirrorDB::User

Description	Value
Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, the user to authenticate to this database can be specified.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Core::MirrorDB::User'} = 'some_user';</pre>

Core::MirrorDB::Password

Description	Value
Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, the password to authenticate to this database can be specified.

Description	Value
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Core::MirrorDB::Password'} = 'some_password';</pre>

Core::PDF

PDF

Description	Value
Description:	Enables PDF output. The CPAN module PDF::API2 is required, if not installed, PDF output will be disabled.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PDF'} = '1';</pre>

PDF::LogoFile

Description	Value
Description:	Specifies the path of the file for the logo in the page header (gif jpg png, 700 x 100 pixel).
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PDF::LogoFile'} = '<OTRS_CONFIG_Home>/var/logo-otrs.png';</pre>

PDF::PageSize

Description	Value
Description:	Defines the standard size of PDF pages.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'PDF::PageSize'} = 'a4';</code>

PDF::MaxPages

Description	Value
Description:	Defines the maximum number of pages per PDF file.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::MaxPages'} = '100';</code>

PDF::TTFontFile###Proportional

Description	Value
Description:	Defines the path and TTF-File to handle proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'Proportional'} = 'DejaVuSans.ttf';</code>

PDF::TTFontFile###ProportionalBold

Description	Value
Description:	Defines the path and TTF-File to handle bold proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalBold'} = 'DejaVuSans-Bold.ttf';</code>

PDF::TTFontFile###ProportionalItalic

Description	Value
Description:	Defines the path and TTF-File to handle italic proportional font in PDF documents.

Description	Value
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalItalic'} = 'DejaVuSans-Oblique.ttf';</code>

PDF::TTFontFile###ProportionalBoldItalic

Description	Value
Description:	Defines the path and TTF-File to handle bold italic proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalBoldItalic'} = 'DejaVuSans-BoldOblique.ttf';</code>

PDF::TTFontFile###Monospaced

Description	Value
Description:	Defines the path and TTF-File to handle monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'Monospaced'} = 'DejaVuSansMono.ttf';</code>

PDF::TTFontFile###MonospacedBold

Description	Value
Description:	Defines the path and TTF-File to handle bold monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'MonospacedBold'} = 'DejaVuSansMono-Bold.ttf';</code>

PDF::TTFontFile###MonospacedItalic

Description	Value
Description:	Defines the path and TTF-File to handle italic monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'MonospacedItalic'} = 'DejaVuSansMono-Oblique.ttf';</code>

PDF::TTFontFile###MonospacedBoldItalic

Description	Value
Description:	Defines the path and TTF-File to handle bold italic monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'MonospacedBoldItalic'} = 'DejaVuSansMono-BoldOblique.ttf';</code>

Core::Package

Package::FileUpload

Description	Value
Description:	Enables file upload in the package manager frontend.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Package::FileUpload'} = '1';</code>

Package::RepositoryRoot

Description	Value
Description:	Defines the location to get online repository list for additional packages. The first available result will be used.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Package::RepositoryRoot'} = ['http://ftp.otrs.org/pub/otrs/misc/packages/ repository.xml', 'http://otrs.org/repository.xml'];</code>

Package::RepositoryList

Description	Value
Description:	Defines the list of online repositories. Another installations can be used as repository, for example: Key="http://example.com/otrs/public.pl? Action=PublicRepository;File=" and Content="Some Name".
Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Package::RepositoryList'} = { 'ftp://ftp.example.com/pub/otrs/misc/packages/' => '[Example] ftp://ftp.example.com/' };</code>

Package::RepositoryAccessRegExp

Description	Value
Description:	Defines the IP regular expression for accessing the local repository. You need to enable this to have access to your local repository and the package::RepositoryList is required on the remote host.
Group:	Framework
SubGroup:	Core::Package
Valid:	0

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'Package::RepositoryAccessRegExp'} = '127\\.0\\.0\\.1';</code>

Package::Timeout

Description	Value
Description:	Sets the timeout (in seconds) for package downloads. Overwrites "WebUserAgent::Timeout".
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Package::Timeout'} = '15';</code>

Package::Proxy

Description	Value
Description:	Fetches packages via proxy. Overwrites "WebUserAgent::Proxy".
Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Package::Proxy'} = 'http:// proxy.sn.no:8001/';</code>

Core::PerformanceLog

PerformanceLog

Description	Value
Description:	Enables performance log (to log the page response time). It will affect the system performance. Frontend::Module###AdminPerformanceLog must be enabled.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'PerformanceLog'} = '0';</code>

PerformanceLog::File

Description	Value
Description:	Specifies the path of the file for the performance log.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PerformanceLog::File'} = '<OTRS_CONFIG_Home>/var/log/Performance.log';</code>

PerformanceLog::FileMax

Description	Value
Description:	Defines the maximum size (in MB) of the log file.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PerformanceLog::FileMax'} = '25';</code>

Core::SOAP

SOAP::User

Description	Value
Description:	Defines the username to access the SOAP handle (bin/cgi-bin/rpc.pl).
Group:	Framework
SubGroup:	Core::SOAP
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SOAP::User'} = 'some_user';</code>

SOAP::Password

Description	Value
Description:	Defines the password to access the SOAP handle (bin/cgi-bin/rpc.pl).
Group:	Framework

Description	Value
SubGroup:	Core::SOAP
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SOAP::Password'} = 'some_pass';</code>

Core::Sendmail

SendmailModule

Description	Value
Description:	Defines the module to send emails. "Sendmail" directly uses the sendmail binary of your operating system. Any of the "SMTP" mechanisms use a specified (external) mailserver. "DoNotSendEmail" doesn't send emails and it is useful for test systems.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule'} = 'Kernel::System::Email::Sendmail';</code>

SendmailModule::CMD

Description	Value
Description:	If "Sendmail" was selected as SendmailModule, the location of the sendmail binary and the needed options must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule::CMD'} = '/usr/sbin/sendmail -i -f';</code>

SendmailModule::Host

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, the mailhost that sends out the mails must be specified.
Group:	Framework
SubGroup:	Core::Sendmail

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule::Host'} = 'mail.example.com';</code>

SendmailModule::Port

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, the port where your mailserver is listening for incoming connections must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::Port'} = '25';</code>

SendmailModule::AuthUser

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, an username must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::AuthUser'} = 'MailserverLogin';</code>

SendmailModule::AuthPassword

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, a password must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::AuthPassword'} = 'MailserverPassword';</code>

SendmailBcc

Description	Value
Description:	Sends all outgoing email via bcc to the specified address. Please use this only for backup reasons.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'SendmailBcc'} = '';</code>

SendmailNotificationEnvelopeFrom

Description	Value
Description:	If set, this address is used as envelope from header in outgoing notifications. If no address is specified, the envelope from header is empty.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailNotificationEnvelopeFrom'} = '';</code>

SendmailEncodingForce

Description	Value
Description:	Forces encoding of outgoing emails (7bit 8bit quoted-printable base64).
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailEncodingForce'} = 'base64';</code>

Core::Session

SessionModule

Description	Value
Description:	Defines the module used to store the session data. With "DB" the frontend server can be splitted from the db server. "FS" is faster.

Description	Value
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionModule'} = 'Kernel::System::AuthSession::DB';</code>

SessionName

Description	Value
Description:	Defines the name of the session key. E.g. Session, SessionID or OTRS.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionName'} = 'Session';</code>

SessionCheckRemoteIP

Description	Value
Description:	Turns on the remote ip address check. It should be set to "No" if the application is used, for example, via a proxy farm or a dialup connection, because the remote ip address is mostly different for the requests.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionCheckRemoteIP'} = '1';</code>

SessionDeleteIfNotRemoteID

Description	Value
Description:	Deletes a session if the session id is used with an invalid remote IP address.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'SessionDeleteIfNotRemoteID'} = '1';</code>

SessionMaxTime

Description	Value
Description:	Defines the maximal valid time (in seconds) for a session id.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionMaxTime'} = '57600';</code>

SessionMaxIdleTime

Description	Value
Description:	Sets the inactivity time (in seconds) to pass before a session is killed and a user is logged out.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionMaxIdleTime'} = '21600';</code>

SessionDeleteIfTimeToOld

Description	Value
Description:	Deletes requested sessions if they have timed out.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionDeleteIfTimeToOld'} = '1';</code>

SessionUseCookie

Description	Value
Description:	Makes the session management use html cookies. If html cookies are disabled or if the client browser disabled html cookies, then the system will work as usual and append the session id to the links.

Description	Value
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionUseCookie'} = '1';</code>

SessionUseCookieAfterBrowserClose

Description	Value
Description:	Stores cookies after the browser has been closed.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionUseCookieAfterBrowserClose'} = '0';</code>

SessionCSRFProtection

Description	Value
Description:	Protection against CSRF (Cross Site Request Forgery) exploits (for more info see http://en.wikipedia.org/wiki/Cross-site_request_forgery).
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionCSRFProtection'} = '1';</code>

SessionDir

Description	Value
Description:	If "FS" was selected for SessionModule, a directory where the session data will be stored must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'SessionDir'} = '<OTRS_CONFIG_Home>/var/sessions';</code>

SessionTable

Description	Value
Description:	If "DB" was selected for SessionModule, a table in database where session data will be stored must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionTable'} = 'sessions';</code>

SessionTableID

Description	Value
Description:	If "DB" was selected for SessionModule, a column for the identifiers in session table must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionTableID'} = 'session_id';</code>

SessionTableValue

Description	Value
Description:	If "DB" was selected for SessionModule, a column for the values in session table must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionTableValue'} = 'session_value';</code>

SessionSaveLoginPasswd

Description	Value
Description:	Saves the login and password on the session table in the database, if "DB" was selected for SessionModule.

Description	Value
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionSaveLoginPasswd'} = '0';</code>

CustomerPanelSessionName

Description	Value
Description:	Defines the name of the key for customer sessions.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSessionName'} = 'CSID';</code>

Core::SpellChecker

SpellChecker

Description	Value
Description:	Enables spell checker support.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellChecker'} = '0';</code>

SpellCheckerBin

Description	Value
Description:	Install ispell or aspell on the system, if you want to use a spell checker. Please specify the path to the aspell or ispell binary on your operating system.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'SpellCheckerBin'} = '/usr/bin/ispell';</code>

SpellCheckerDictDefault

Description	Value
Description:	Defines the default spell checker dictionary.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellCheckerDictDefault'} = 'english';</code>

SpellCheckerIgnore

Description	Value
Description:	Defines a default list of words, that are ignored by the spell checker.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellCheckerIgnore'} = ['www', 'webmail', 'https', 'http', 'html', 'rfc'];</code>

Core::Stats

Stats::StatsHook

Description	Value
Description:	Sets the stats hook.
Group:	Framework
SubGroup:	Core::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::StatsHook'} = 'Stat#';</code>

Stats::StatsStartNumber

Description	Value
Description:	Start number for statistics counting. Every new stat increments this number.
Group:	Framework
SubGroup:	Core::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::StatsStartNumber'} = '10000';</code>

Stats::MaxXaxisAttributes

Description	Value
Description:	Defines the default maximum number of X-axis attributes for the time scale.
Group:	Framework
SubGroup:	Core::Stats
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Stats::MaxXaxisAttributes'} = '1000';</code>

Core::Stats::Graph

Stats::Graph::t_margin

Description	Value
Description:	Specifies the top margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::t_margin'} = '10';</code>

Stats::Graph::l_margin

Description	Value
Description:	Specifies the left margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Stats::Graph::l_margin'} = '10';</code>

Stats::Graph::b_margin

Description	Value
Description:	Specifies the bottom margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::b_margin'} = '10';</code>

Stats::Graph::r_margin

Description	Value
Description:	Specifies the right margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::r_margin'} = '20';</code>

Stats::Graph::bgclr

Description	Value
Description:	Specifies the background color of the picture.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::bgclr'} = 'white';</code>

Stats::Graph::transparent

Description	Value
Description:	Makes the picture transparent.
Group:	Framework
SubGroup:	Core::Stats::Graph

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::transparent'} = '0';</code>

Stats::Graph::fgclr

Description	Value
Description:	Specifies the border color of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::fgclr'} = 'black';</code>

Stats::Graph::boxclr

Description	Value
Description:	Specifies the background color of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::boxclr'} = 'white';</code>

Stats::Graph::accentclr

Description	Value
Description:	Specifies the border color of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::accentclr'} = 'black';</code>

Stats::Graph::legendclr

Description	Value
Description:	Specifies the text color of the legend.

Description	Value
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legendclr'} = 'black';</code>

Stats::Graph::textclr

Description	Value
Description:	Specifies the text color of the chart (e. g. caption).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::textclr'} = 'black';</code>

Stats::Graph::dclrs

Description	Value
Description:	Defines the colors for the graphs.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::dclrs'} = ['red', 'green', 'blue', 'yellow', 'black', 'purple', 'orange', 'pink', 'marine', 'cyan', 'lgray', 'lblue', 'lyellow', 'lgreen', 'lred', 'lpurple', 'lorange',</code>

Description	Value
	<code>'lbrown'</code> <code>];</code>

Stats::Graph::line_width

Description	Value
Description:	Defines the boldness of the line drawn by the graph.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::line_width'} = '1';</code>

Stats::Graph::legend_placement

Description	Value
Description:	Defines the placement of the legend. This should be a two letter key of the form: 'B[LCR] R[TCB]'. The first letter indicates the placement (Bottom or Right), and the second letter the alignment (Left, Right, Center, Top, or Bottom).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_placement'} = 'BC';</code>

Stats::Graph::legend_spacing

Description	Value
Description:	Defines the spacing of the legends.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_spacing'} = '4';</code>

Stats::Graph::legend_marker_width

Description	Value
Description:	Defines the width of the legend.
Group:	Framework

Description	Value
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_marker_width'} = '12';</code>

Stats::Graph::legend_marker_height

Description	Value
Description:	Defines the height of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_marker_height'} = '8';</code>

Core::Time

TimeInputFormat

Description	Value
Description:	Defines the date input format used in forms (option or input fields).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeInputFormat'} = 'Option';</code>

TimeShowAlwaysLong

Description	Value
Description:	Shows time in long format (days, hours, minutes), if set to "Yes"; or in short format (days, hours), if set to "No".
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeShowAlwaysLong'} = '0';</code>

TimeZone

Description	Value
Description:	Sets the system time zone (required a system with UTC as system time). Otherwise this is a diff time to the local time.
Group:	Framework
SubGroup:	Core::Time
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'TimeZone'} = '+0';</code>

TimeZoneUser

Description	Value
Description:	Sets the user time zone per user (required a system with UTC as system time and UTC under TimeZone). Otherwise this is a diff time to the local time.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'TimeZoneUser'} = '0';</code>

TimeZoneUserBrowserAutoOffset

Description	Value
Description:	Sets the user time zone per user based on java script / browser time zone offset feature at login time.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'TimeZoneUserBrowserAutoOffset'} = '1';</code>

CalendarWeekDayStart

Description	Value
Description:	Define the start day of the week for the date picker.
Group:	Framework
SubGroup:	Core::Time
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'CalendarWeekDayStart'} = '1';</code>

TimeVacationDays

Description	Value
Description:	Adds the permanent vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDays'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } }; </pre>

TimeVacationDaysOneTime

Description	Value
Description:	Adds the one time vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDaysOneTime'} = { '2004' => { '1' => { '1' => 'test' } } }; </pre>

TimeWorkingHours

Description	Value
Description:	Defines the hours and week days to count the working time.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', </pre>

Description	Value
	<pre> '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'] }; </pre>

TimeShowCompleteDescription

Description	Value
Description:	Shows time use complete description (days, hours, minutes), if set to "Yes"; or just first letter (d, h, m), if set to "No".
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeShowCompleteDescription'} = '0';</code>

Core::Time::Calendar1

TimeZone::Calendar1Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar1Name'} = 'Calendar Name 1';</pre>

TimeZone::Calendar1

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar1'} = '0';</pre>

TimeVacationDays::Calendar1

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar1'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => {</pre>

Description	Value
	<pre>'1' => 'International Workers\' Day' } };</pre>

TimeVacationDaysOneTime::Calendar1

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar1'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

TimeWorkingHours::Calendar1

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar1'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], };</pre>

Description	Value
	<pre>'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11',</pre>

Description	Value
	<pre>'12', '13', '14', '15', '16', '17', '18', '19', '20'] };</pre>

Core::Time::Calendar2

TimeZone::Calendar2Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar2Name'} = 'Calendar Name 2';</pre>

TimeZone::Calendar2

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar2'} = '0';</pre>

TimeVacationDays::Calendar2

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDays::Calendar2'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } }; </pre>

TimeVacationDaysOneTime::Calendar2

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDaysOneTime::Calendar2'} = { '2004' => { '1' => { '1' => 'test' } } }; </pre>

TimeWorkingHours::Calendar2

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours::Calendar2'} = { 'Fri' => [</pre>

Description	Value
	<pre>'8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12',</pre>

Description	Value
	<pre>'13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'] };</pre>

Core::Time::Calendar3

TimeZone::Calendar3Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar3Name'} = 'Calendar Name 3';</code>

TimeZone::Calendar3

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar3'} = '0';</code>

TimeVacationDays::Calendar3

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDays::Calendar3'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } }; </pre>

TimeVacationDaysOneTime::Calendar3

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDaysOneTime::Calendar3'} = { '2004' => { '1' => { '1' => 'test' } } }; </pre>

TimeWorkingHours::Calendar3

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours::Calendar3'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', </pre>

Description	Value
	<pre> '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20']]; </pre>

Core::Time::Calendar4

TimeZone::Calendar4Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'TimeZone::Calendar4Name'} = 'Calendar Name 4';</code>

TimeZone::Calendar4

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar4'} = '0';</code>

TimeVacationDays::Calendar4

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDays::Calendar4'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } }; </pre>

TimeVacationDaysOneTime::Calendar4

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar4

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar4'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

TimeWorkingHours::Calendar4

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar4'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18',</pre>

Description	Value
	<pre>'19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'] };</pre>

Core::Time::Calendar5

TimeZone::Calendar5Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar5Name'} = 'Calendar Name 5';</code>

TimeZone::Calendar5

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar5'} = '0';</code>

TimeVacationDays::Calendar5

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeVacationDays::Calendar5'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => {</code>

Description	Value
	<pre>'1' => 'International Workers\' Day' } };</pre>

TimeVacationDaysOneTime::Calendar5

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar5'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

TimeWorkingHours::Calendar5

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar5'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], };</pre>

Description	Value
	<pre>'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11',</pre>

Description	Value
	<pre>'12', '13', '14', '15', '16', '17', '18', '19', '20'] };</pre>

Core::Time::Calendar6

TimeZone::Calendar6Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar6Name'} = 'Calendar Name 6';</pre>

TimeZone::Calendar6

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar6'} = '0';</pre>

TimeVacationDays::Calendar6

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDays::Calendar6'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } }; </pre>

TimeVacationDaysOneTime::Calendar6

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDaysOneTime::Calendar6'} = { '2004' => { '1' => { '1' => 'test' } } }; </pre>

TimeWorkingHours::Calendar6

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours::Calendar6'} = { 'Fri' => [</pre>

Description	Value
	<pre>'8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12',</pre>

Description	Value
	<pre>'13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'] };</pre>

Core::Time::Calendar7

TimeZone::Calendar7Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar7Name'} = 'Calendar Name 7';</code>

TimeZone::Calendar7

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar7'} = '0';</code>

TimeVacationDays::Calendar7

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar7'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

TimeVacationDaysOneTime::Calendar7

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar7'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

TimeWorkingHours::Calendar7

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours::Calendar7'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', </pre>

Description	Value
	<pre> '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20']]; </pre>

Core::Time::Calendar8

TimeZone::Calendar8Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'TimeZone::Calendar8Name'} = 'Calendar Name 8';</code>

TimeZone::Calendar8

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar8'} = '0';</code>

TimeVacationDays::Calendar8

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDays::Calendar8'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } }; </pre>

TimeVacationDaysOneTime::Calendar8

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar8

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar8'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

TimeWorkingHours::Calendar8

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar8'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18',</pre>

Description	Value
	<pre>'19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'] };</pre>

Core::Time::Calendar9

TimeZone::Calendar9Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar9Name'} = 'Calendar Name 9';</pre>

TimeZone::Calendar9

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar9'} = '0';</pre>

TimeVacationDays::Calendar9

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar9'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => {</pre>

Description	Value
	<pre>'1' => 'International Workers\' Day' } };</pre>

TimeVacationDaysOneTime::Calendar9

Description	Value
Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar9'} = { '2004' => { '1' => { '1' => 'test' } } } };</pre>

TimeWorkingHours::Calendar9

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar9'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], }</pre>

Description	Value
	<pre>'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11',</pre>

Description	Value
	<pre>'12', '13', '14', '15', '16', '17', '18', '19', '20'] };</pre>

Core::Web

Frontend::WebPath

Description	Value
Description:	Defines the URL base path of icons, CSS and Java Script.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::WebPath'} = '/otrs-web/';</code>

Frontend::ImagePath

Description	Value
Description:	Defines the URL image path of icons for navigation.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::ImagePath'} = '<OTRS_CONFIG_Frontend::WebPath>skins/Agent/ default/img/';</code>

Frontend::CSSPath

Description	Value
Description:	Defines the URL CSS path.
Group:	Framework
SubGroup:	Core::Web

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CSSPath'} = '<OTRS_CONFIG_Frontend::WebPath>css/';</code>

Frontend::JavaScriptPath

Description	Value
Description:	Defines the URL java script path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::JavaScriptPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/';</code>

Frontend::RichText

Description	Value
Description:	Uses richtext for viewing and editing: articles, salutations, signatures, standard responses, auto responses and notifications.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichText'} = '1';</code>

Frontend::RichTextPath

Description	Value
Description:	Defines the URL rich text editor path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/thirdparty/ ckeditor-3.6.2/';</code>

Frontend::RichTextWidth

Description	Value
Description:	Defines the width for the rich text editor component. Enter number (pixels) or percent value (relative).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextWidth'} = '620';</code>

Frontend::RichTextHeight

Description	Value
Description:	Defines the =hHeight for the rich text editor component. Enter number (pixels) or percent value (relative).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextHeight'} = '320';</code>

Frontend::RichText::DefaultCSS

Description	Value
Description:	Defines the default CSS used in rich text editors.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichText::DefaultCSS'} = 'font-family:Geneva,Helvetica,Arial,sans-serif; font-size: 12px;';</code>

DefaultViewNewLine

Description	Value
Description:	Automated line break in text messages after x number of chars.
Group:	Framework
SubGroup:	Core::Web
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'DefaultViewNewLine'} = '90';</code>

DefaultViewLines

Description	Value
Description:	Sets the number of lines that are displayed in text messages (e.g. ticket lines in the QueueZoom).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultViewLines'} = '6000';</code>

Frontend::AnimationEnabled

Description	Value
Description:	Turns on the animations used in the GUI. If you have problems with these animations (e.g. performance issues), you can turn them off here.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::AnimationEnabled'} = '1';</code>

AttachmentDownloadType

Description	Value
Description:	Allows choosing between showing the attachments of a ticket in the browser (inline) or just make them downloadable (attachment).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AttachmentDownloadType'} = 'attachment';</code>

WebMaxFileUpload

Description	Value
Description:	Defines the maximal size (in bytes) for file uploads via the browser.

Description	Value
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'WebMaxFileUpload'} = '16777216';</code>

WebUploadCacheModule

Description	Value
Description:	Selects the module to handle uploads via the web interface. "DB" stores all uploads in the database, "FS" uses the file system.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'WebUploadCacheModule'} = 'Kernel::System::Web::UploadCache::DB';</code>

Frontend::Output::FilterText###AAAURL

Description	Value
Description:	Defines the filter that processes the text in the articles, in order to highlight URLs.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::Output::FilterText'}->{'AAAURL'} = { 'Module' => 'Kernel::Output::HTML::OutputFilterTextURL' };</code>

Frontend::Themes

Description	Value
Description:	Activates the available themes on the system. Value 1 means active, 0 means inactive.
Group:	Framework
SubGroup:	Core::Web
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Themes'} = { 'Lite' => '0', 'Standard' => '1' };</pre>

Frontend::Output::FilterText###OutputFilterTextAutoLink

Description	Value
Description:	Defines a filter to process the text in the articles, in order to highlight predefined keywords.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::FilterText'}- >{'OutputFilterTextAutoLink'} = { 'Module' => 'Kernel::Output::HTML::OutputFilterTextAutoLink' };</pre>

Frontend::Output::OutputFilterTextAutoLink###CVE

Description	Value
Description:	Defines a filter for html output to add links behind CVE numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self- >{'Frontend::Output::OutputFilterTextAutoLink'}- >{'CVE'} = { 'RegExp' => ['(CVE CAN)\{3,4}\{2,6}'], 'URL1' => { 'Description' => 'Mitre', 'Image' => 'http://cve.mitre.org/favicon.ico', 'Target' => '_blank', 'URL' => 'http://cve.mitre.org/cgi-bin/ cvename.cgi?name=<MATCH1>-<MATCH2>-<MATCH3>' } };</pre>

Description	Value
	<pre> }, 'URL2' => { 'Description' => 'Google', 'Image' => 'http://www.google.de/favicon.ico', 'Target' => '_blank', 'URL' => 'http://google.com/search?q=<MATCH1>- <MATCH2>-<MATCH3>' }, 'URL3' => { 'Description' => 'US-CERT NVD', 'Image' => 'http://nvd.nist.gov/favicon.ico', 'Target' => '_blank', 'URL' => 'http://nvd.nist.gov/nvd.cfm? cvename=<MATCH1>-<MATCH2>-<MATCH3>' } }; </pre>

Frontend::Output::OutputFilterTextAutoLink###Bugtraq

Description	Value
Description:	Defines a filter for html output to add links behind bugtraq numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre> \$self- >{'Frontend::Output::OutputFilterTextAutoLink'}- >{'Bugtraq'} = { 'RegExp' => ['Bugtraq[\\s\\w\\t]*?ID[\\s\\w\\t]*?:[\\s\\w\\ \\t]*?(\\d{2,8})', 'Bugtraq[\\s\\w\\t]*?ID[\\s\\w\\t]*?(\\ \\d{2,8})', 'Bugtraq[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})', 'Bugtraq[\\s\\w\\t]*?(\\d{2,8})', 'BID[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})', 'BID[\\s\\w\\t]*?(\\d{2,8})'], 'URL1' => { 'Description' => 'Security Focus', 'Image' => 'http://www.securityfocus.com/ favicon.ico', 'Target' => '_blank', 'URL' => 'http://www.securityfocus.com/bid/ <MATCH1>/info' }, </pre>

Description	Value
	<pre>'URL2' => { 'Description' => 'Google', 'Image' => 'http://www.google.de/favicon.ico', 'Target' => '_blank', 'URL' => 'http://google.com/search?q=<MATCH>' } };</pre>

Frontend::Output::OutputFilterTextAutoLink###MSBulletins

Description	Value
Description:	Defines a filter for html output to add links behind MSBulletin numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'MSBulletins'} = { 'RegExp' => ['MS[^A-Za-z]{0,5}(\\d\\d).?(\\d{2,4})'], 'URL1' => { 'Description' => 'Microsoft Technet', 'Image' => 'http://www.microsoft.com/favicon.ico', 'Target' => '_blank', 'URL' => 'http://www.microsoft.com/technet/security/bulletin/MS<MATCH1>-<MATCH2>.aspx' }, 'URL2' => { 'Description' => 'Google', 'Image' => 'http://www.google.de/favicon.ico', 'Target' => '_blank', 'URL' => 'http://google.com/search?q=MS<MATCH1>-<MATCH2>' } };</pre>

Frontend::Output::OutputFilterTextAutoLink###Setting1

Description	Value
Description:	Define a filter for html output to add links behind a defined string. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In

Description	Value
	this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self- >{'Frontend::Output::OutputFilterTextAutoLink'}- >{'Setting1'} = { 'RegExp' => ['RegExp'], 'URL1' => { 'Description' => 'Description', 'Image' => 'right-small.png', 'Target' => '_blank', 'URL' => 'URL' }, 'URL2' => { 'Description' => 'Description', 'Image' => 'Image', 'Target' => '_blank', 'URL' => 'URL' } }; </pre>

Frontend::Output::OutputFilterTextAutoLink###Setting2

Description	Value
Description:	Defines a filter for html output to add links behind a defined string. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self- >{'Frontend::Output::OutputFilterTextAutoLink'}- >{'Setting2'} = { 'RegExp' => ['RegExp'], 'URL1' => { 'Description' => 'Description', 'Image' => 'right-small.png', </pre>

Description	Value
	<pre> 'Target' => '_blank', 'URL' => 'URL' }, 'URL2' => { 'Description' => 'Description', 'Image' => 'Image', 'Target' => '_blank', 'URL' => 'URL' }, 'URL3' => { 'Description' => 'Description', 'Image' => 'Image', 'Target' => '_blank', 'URL' => 'URL' } }; </pre>

Loader::Enabled::CSS

Description	Value
Description:	If enabled, OTRS will deliver all CSS files in minified form. WARNING: If you turn this off, there will likely be problems in IE 7, because it cannot load more than 32 CSS files.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Enabled::CSS'} = '1';</code>

Loader::Enabled::JS

Description	Value
Description:	If enabled, OTRS will deliver all JavaScript files in minified form.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Enabled::JS'} = '1';</code>

Loader::Agent::CommonCSS###000-Framework

Description	Value
Description:	List of CSS files to always be loaded for the agent interface.

Description	Value
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Loader::Agent::CommonCSS'}->{'000-Framework'} = ['Core.Reset.css', 'Core.Default.css', 'Core.Header.css', 'Core.OverviewControl.css', 'Core.OverviewSmall.css', 'Core.OverviewMedium.css', 'Core.OverviewLarge.css', 'Core.Footer.css', 'Core.PageLayout.css', 'Core.Form.css', 'Core.Table.css', 'Core.Widget.css', 'Core.WidgetMenu.css', 'Core.TicketDetail.css', 'Core.Tooltip.css', 'Core.Dialog.css', 'Core.Print.css']; </pre>

Loader::Agent::CommonCSS::IE7###000-Framework

Description	Value
Description:	List of IE7-specific CSS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Loader::Agent::CommonCSS::IE7'}->{'000-Framework'} = ['Core.Default.IE7.css', 'Core.Header.IE7.css', 'Core.OverviewControl.IE7.css', 'Core.OverviewSmall.IE7.css', 'Core.OverviewMedium.IE7.css', 'Core.OverviewLarge.IE7.css', 'Core.PageLayout.IE7.css', 'Core.Form.IE7.css', 'Core.Widget.IE7.css', 'Core.WidgetMenu.IE7.css', 'Core.Tooltip.IE7.css',]; </pre>

Description	Value
	'Core.Dialog.IE7.css', 'Core.TicketDetail.IE7.css', 'Core.Table.IE7.css'];

Loader::Agent::CommonCSS::IE8###000-Framework

Description	Value
Description:	List of IE8-specific CSS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonCSS::IE8'}->{'000-Framework'} = ['Core.OverviewSmall.IE8.css'];</pre>

Loader::Agent::CommonJS###000-Framework

Description	Value
Description:	List of JS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonJS'}->{'000-Framework'} = ['thirdparty/json/json2.js', 'thirdparty/jquery-1.6.4/jquery.js', 'thirdparty/jquery-ui-1.8.16/jquery-ui.js', 'thirdparty/jquery-validate-1.9/ jquery.validate.js', 'thirdparty/stacktrace-0.3/stacktrace.js', 'Core.JavaScriptEnhancements.js', 'Core.Debug.js', 'Core.Data.js', 'Core.Config.js', 'Core.Exception.js', 'Core.JSON.js', 'Core.AJAX.js', 'Core.UI.js', 'Core.UI.IE7Fixes.js', 'Core.UI.Accordion.js', 'Core.UI.Datepicker.js',</pre>

Description	Value
	<pre>'Core.UI.Resizable.js', 'Core.UI.Table.js', 'Core.UI.Accessibility.js', 'Core.UI.RichTextEditor.js', 'Core.UI.Dialog.js', 'Core.UI.ActionRow.js', 'Core.UI.Popup.js', 'Core.Form.js', 'Core.Form.ErrorTooltips.js', 'Core.Form.Validate.js', 'Core.App.js', 'Core.Agent.js', 'Core.Agent.Search.js'];</pre>

Loader::Customer::CommonCSS###000-Framework

Description	Value
Description:	List of CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonCSS'}->{'000- Framework'} = ['Core.Reset.css', 'Core.Default.css', 'Core.Form.css', 'Core.Dialog.css', 'Core.Tooltip.css', 'Core.Login.css', 'Core.Control.css', 'Core.Table.css', 'Core.TicketZoom.css', 'Core.Print.css'];</pre>

Loader::Customer::CommonCSS::IE6###000-Framework

Description	Value
Description:	List of IE6-specific CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre>\$Self->{'Loader::Customer::CommonCSS::IE6'}- >{'000-Framework'} = ['Core.IE6.css', 'Core.Tooltip.IE6.css'];</pre>

Loader::Customer::CommonCSS::IE7###000-Framework

Description	Value
Description:	List of IE7-specific CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonCSS::IE7'}- >{'000-Framework'} = ['Core.IE7.css', 'Core.Tooltip.IE7.css', 'Core.Dialog.IE7.css'];</pre>

Loader::Customer::CommonCSS::IE8###000-Framework

Description	Value
Description:	List of IE8-specific CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonCSS::IE8'}- >{'000-Framework'} = [];</pre>

Loader::Customer::CommonJS###000-Framework

Description	Value
Description:	List of JS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre> \$Self->{'Loader::Customer::CommonJS'}->{'000-Framework'} = ['thirdparty/jquery-1.6.4/jquery.js', 'thirdparty/jquery-validate-1.9/jquery.validate.js', 'thirdparty/jquery-ui-1.8.16/jquery-ui.js', 'thirdparty/stacktrace-0.3/stacktrace.js', 'Core.Debug.js', 'Core.Data.js', 'Core.Exception.js', 'Core.App.js', 'Core.JavaScriptEnhancements.js', 'Core.Config.js', 'Core.UI.js', 'Core.UI.IE7Fixes.js', 'Core.UI.Accessibility.js', 'Core.UI.Dialog.js', 'Core.UI.RichTextEditor.js', 'Core.UI.Datepicker.js', 'Core.UI.Popup.js', 'Core.Form.js', 'Core.Form.ErrorTooltips.js', 'Core.Form.Validate.js', 'Core.Customer.js']; </pre>

Loader::Agent::DefaultSelectedSkin

Description	Value
Description:	The agent skin's InternalName which should be used in the agent interface. Please check the available skins in Frontend::Agent::Skins.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Loader::Agent::DefaultSelectedSkin'} = 'default'; </pre>

Loader::Customer::SelectedSkin::HostBased

Description	Value
Description:	It is possible to configure different skins, for example to distinguish between different customers, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid skin on your system. Please see the example entries for the proper form of the regex.

Description	Value
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{ 'Loader::Customer::SelectedSkin::HostBased' } = { 'host1\\.example\\.com' => 'Someskin1', 'host2\\.example\\.com' => 'Someskin2' };</pre>

Core::WebUserAgent

WebUserAgent::Timeout

Description	Value
Description:	Sets the timeout (in seconds) for http/ftp downloads.
Group:	Framework
SubGroup:	Core::WebUserAgent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{ 'WebUserAgent::Timeout' } = '15';</pre>

WebUserAgent::Proxy

Description	Value
Description:	Defines the connections for http/ftp, via a proxy.
Group:	Framework
SubGroup:	Core::WebUserAgent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{ 'WebUserAgent::Proxy' } = 'http:// proxy.sn.no:8001/';</pre>

Crypt::PGP

PGP

Description	Value
Description:	Enables PGP support. When PGP support is enabled for signing and securing mail, it is HIGHLY recommended that the web server be run as the OTRS user.

Description	Value
	Otherwise, there will be problems with the privileges when accessing <code>.gnupg</code> folder.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP'} = '0';</code>

PGP::Bin

Description	Value
Description:	Defines the path to PGP binary.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP::Bin'} = '/usr/bin/gpg';</code>

PGP::Options

Description	Value
Description:	Sets the options for PGP binary.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP::Options'} = '--homedir /opt/otrs/.gnupg/ --batch --no-tty --yes';</code>

PGP::Key::Password

Description	Value
Description:	Sets the password for private PGP key.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self->{'PGP::Key::Password'} = { '488A0B8F' => 'SomePassword', 'D2DF79FA' => 'SomePassword' };</pre>

PGP::TrustedNetwork

Description	Value
Description:	Set this to yes if you trust in all your public and private pgp keys, even if they are not certified with a trusted signature.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PGP::TrustedNetwork'} = '0';</pre>

PGP::Log

Description	Value
Description:	Configure your own log text for PGP.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PGP::Log'} = { 'BADSIG' => 'The PGP signature with the keyid has not been verified successfully.', 'ERRSIG' => 'It was not possible to check the PGP signature, this may be caused by a missing public key or an unsupported algorithm.', 'EXPKEYSIG' => 'The PGP signature was made by an expired key.', 'GOODSIG' => 'Good PGP signature.', 'KEYREVOKED' => 'The PGP signature was made by a revoked key, this could mean that the signature is forged.', 'NODATA' => 'No valid OpenPGP data found.', 'NO_PUBKEY' => 'No public key found.', 'REVKEYSIG' => 'The PGP signature was made by a revoked key, this could mean that the signature is forged.', 'SIGEXPIRED' => 'The PGP signature is expired.', 'SIG_ID' => 'Signature data.',</pre>

Description	Value
	<pre>'TRUST_UNDEFINED' => 'This key is not certified with a trusted signature!.', 'VALIDSIG' => 'The PGP signature with the keyid is good.' };</pre>

Crypt::SMIME

SMIME

Description	Value
Description:	Enables S/MIME support.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME'} = '0';</code>

SMIME::Bin

Description	Value
Description:	Defines the path to open ssl binary. It may need a HOME env (<code>\$ENV{HOME} = '/var/lib/wwwrun';</code>).
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::Bin'} = '/usr/bin/openssl';</code>

SMIME::CertPath

Description	Value
Description:	Specifies the directory where SSL certificates are stored.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::CertPath'} = '/etc/ssl/certs';</code>

SMIME::PrivatePath

Description	Value
Description:	Specifies the directory where private SSL certificates are stored.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'SMIME::PrivatePath'} = '/etc/ssl/private';</pre>

Frontend::Admin::AdminCustomerUser

AdminCustomerUser::RunInitialWildcardSearch

Description	Value
Description:	Runs an initial wildcard search of the existing customer users when accessing the AdminCustomerUser module.
Group:	Framework
SubGroup:	Frontend::Admin::AdminCustomerUser
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'AdminCustomerUser::RunInitialWildcardSearch'} = '1';</pre>

Frontend::Admin::ModuleRegistration

Frontend::Module###Admin

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'Admin'} = { 'Description' => 'Admin-Area', 'Group' => [</pre>

Description	Value
	<pre> 'admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.css'], 'CSS_IE7' => ['Core.Agent.AdminIE7.css'], 'JavaScript' => ['Core.Agent.Admin.SysConfig.js'] }, 'NavBar' => [{ 'AccessKey' => 'a', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=Admin', 'LinkOption' => '', 'Name' => 'Admin', 'NavBar' => 'Admin', 'Prio' => '10000', 'Type' => 'Menu' }], 'NavBarModule' => { 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin' }, 'NavBarName' => 'Admin', 'Title' => '' }; </pre>

Frontend::Module###AdminInit

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminInit'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarName' => '', 'Title' => 'Init' }; </pre>

Description	Value
	};

Frontend::Module###AdminUser

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminUser'} = { 'Description' => 'Create and manage agents.', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage agents.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Agents' }; </pre>

Frontend::Module###AdminGroup

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', </pre>

Description	Value
	<pre> 'Name' => 'Groups', 'Prio' => '150' }, 'NavBarName' => 'Admin', 'Title' => 'Groups' }; </pre>

Frontend::Module###AdminUserGroup

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminUserGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link agents to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents <-> Groups', 'Prio' => '200' }, 'NavBarName' => 'Admin', 'Title' => 'Agents <-> Groups' }; </pre>

Frontend::Module###AdminCustomerUser

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminCustomerUser'} = { 'Description' => 'Edit Customers', 'Group' => [</pre>

Description	Value
	<pre> 'admin', 'users'], 'GroupRo' => [''], 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => 'c', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AdminCustomerUser;Nav=Agent', 'LinkOption' => '', 'Name' => 'Customers', 'NavBar' => 'Customers', 'Prio' => '9000', 'Type' => 'Menu' }], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Create and manage customers.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customers', 'Prio' => '300' }, 'NavBarName' => 'Customers', 'Title' => 'Customers' }; </pre>

Frontend::Module###AdminCustomerCompany

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}- >{'AdminCustomerCompany'} = { 'Description' => 'Admin', </pre>

Description	Value
	<pre> 'Group' => ['admin', 'users'], 'GroupRo' => [''], 'NavBar' => [{ 'AccessKey' => 'c', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AdminCustomerCompany;Nav=Agent', 'LinkOption' => '', 'Name' => 'Companies', 'NavBar' => 'Companies', 'Prio' => '9100', 'Type' => 'Menu' }], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Create and manage companies.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customer Companies', 'Prio' => '310' }, 'NavBarName' => 'Admin', 'Title' => 'Customer Companies' }; </pre>

Frontend::Module###AdminCustomerUserGroup

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}- >{'AdminCustomerUserGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { </pre>

Description	Value
	<pre> 'Block' => 'Customer', 'Description' => 'Link customers to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customers <-> Groups', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Customers <-> Groups' }; </pre>

Frontend::Module###AdminCustomerUserService

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}- >{'AdminCustomerUserService'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Link customers to services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customers <-> Services', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Customers <-> Services' }; </pre>

Frontend::Module###AdminRole

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Frontend::Module'}->{'AdminRole'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage roles.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Roles', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'Roles' }; </pre>

Frontend::Module###AdminRoleUser

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminRoleUser'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link agents to roles.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents <-> Roles', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'Agents <-> Roles' }; </pre>

Frontend::Module###AdminRoleGroup

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminRoleGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link roles to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Roles <-> Groups', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'Roles <-> Groups' }; </pre>

Frontend::Module###AdminSMIME

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSMIME'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Manage S/MIME certificates for email encryption.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'S/MIME Certificates', 'Prio' => '1100' }, 'NavBarName' => 'Admin', 'Title' => 'S/MIME Management' }; </pre>

Description	Value
	};

Frontend::Module###AdminPGP

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPGP'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Manage PGP keys for email encryption.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PGP Keys', 'Prio' => '1200' }, 'NavBarName' => 'Admin', 'Title' => 'PGP Key Management' }; </pre>

Frontend::Module###AdminMailAccount

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminMailAccount'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', </pre>

Description	Value
	<pre>'Description' => 'Manage POP3 or IMAP accounts to fetch email from.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PostMaster Mail Accounts', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Mail Accounts' };</pre>

Frontend::Module###AdminPostMasterFilter

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AdminPostMasterFilter'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Filter incoming emails.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PostMaster Filters', 'Prio' => '200' }, 'NavBarName' => 'Admin', 'Title' => 'PostMaster Filters' };</pre>

Frontend::Module###AdminEmail

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Frontend::Module'}->{'AdminEmail'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Send notifications to users.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Admin Notification', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Admin Notification' }; </pre>

Frontend::Module###AdminSession

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSession'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Manage existing sessions.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Session Management', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Session Management' }; </pre>

Frontend::Module###AdminPerformanceLog

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPerformanceLog'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.PerformanceLog.css'] }, 'NavBarModule' => { 'Block' => 'System', 'Description' => 'View performance benchmark results.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Performance Log', 'Prio' => '550' }, 'NavBarName' => 'Admin', 'Title' => 'Performance Log' }; </pre>

Frontend::Module###AdminLog

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminLog'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'View system log messages.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', </pre>

Description	Value
	<pre>'Name' => 'System Log', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'System Log' };</pre>

Frontend::Module###AdminSelectBox

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminSelectBox'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Execute SQL statements.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'SQL Box', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'SQL Box' };</pre>

Frontend::Module###AdminPackageManager

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminPackageManager'} = { 'Description' => 'Software Package Manager', 'Group' => [</pre>

Description	Value
	<pre> 'admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Update and extend your system with software packages.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Package Manager', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'Package Manager' }; </pre>

Frontend::Module###AdminGenericInterfaceDebugger

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceDebugger'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'CSS_IE7' => ['Core.Agent.Admin.GenericInterface.IE7.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceDebugger.js'] }, 'Title' => 'GenericInterface Debugger GUI' }; </pre>

Frontend::Module###AdminGenericInterfaceWebservice

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceWebservice'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceWebservice.js'] }, 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Create and manage web services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Web Services', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'GenericInterface Web Service GUI' }; </pre>

Frontend::Module###AdminGenericInterfaceTransportHTTPSOAP

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceTransportHTTPSOAP'} = { 'Description' => 'Admin', 'Group' => ['admin'], </pre>

Description	Value
	<pre>'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'] }, 'Title' => 'GenericInterface TransportHTTPS SOAP GUI' };</pre>

Frontend::Module###AdminGenericInterfaceWebserviceHistory

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceWebserviceHistory'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceWebserviceHistory.js'] }, 'Title' => 'GenericInterface Webservice History GUI' };</pre>

Frontend::Module###AdminGenericInterfaceOperationDefault

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceOperationDefault'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceOperation.js'] }, 'Title' => 'GenericInterface Operation GUI' }; </pre>

Frontend::Module###AdminGenericInterfaceInvokerDefault

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceInvokerDefault'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceInvoker.js'] }, 'Title' => 'GenericInterface Invoker GUI' }; </pre>

Frontend::Module###AdminGenericInterfaceMappingSimple

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceMappingSimple'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'CSS_IE7' => ['Core.Agent.Admin.GenericInterface.IE7.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceMappingSimple.js'] }, 'Title' => 'GenericInterface Webservice Mapping GUI' }; </pre>

Frontend::Agent

AgentLogo

Description	Value
Description:	The logo shown in the header of the agent interface. The URL to the image can be a relative URL to the skin image directory, or a full URL to a remote web server.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'AgentLogo'} = { 'StyleHeight' => '67px', 'StyleRight' => '38px', 'StyleTop' => '-4px', 'StyleWidth' => '244px', 'URL' => 'skins/Agent/default/img/logo_bg.png' }; </pre>

AgentLoginLogo

Description	Value
Description:	The logo shown on top of the login box of the agent interface. The URL to the image must be relative URL to the skin image directory.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'AgentLoginLogo'} = { 'StyleHeight' => '100px', 'URL' => 'skins/Agent/default/img/ loginlogo_default.png' };</pre>

LoginURL

Description	Value
Description:	Defines an alternate URL, where the login link refers to.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'LoginURL'} = 'http://host.example.com/ login.html';</pre>

LogoutURL

Description	Value
Description:	Defines an alternate URL, where the logout link refers to.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'LogoutURL'} = 'http://host.example.com/ thanks-for-using-otrs.html';</pre>

PreApplicationModule###AgentInfo

Description	Value
Description:	Defines a useful module to load specific user options or to display news.

Description	Value
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PreApplicationModule'}->{'AgentInfo'} = 'Kernel::Modules::AgentInfo';</code>

InfoKey

Description	Value
Description:	Defines the key to be checked with Kernel::Modules::AgentInfo module. If this user preferences key is true, the message is accepted by the system.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'InfoKey'} = 'wpt22';</code>

InfoFile

Description	Value
Description:	File that is displayed in the Kernel::Modules::AgentInfo module, if located under Kernel/Output/HTML/Standard/AgentInfo.dtl.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'InfoFile'} = 'AgentInfo';</code>

LostPassword

Description	Value
Description:	Activates lost password feature for agents, in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'LostPassword'} = '1';</code>

ShowMotd

Description	Value
Description:	Shows the message of the day on login screen of the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ShowMotd'} = '0';</code>

NotificationSubjectLostPasswordToken

Description	Value
Description:	Defines the subject for notification mails sent to agents, with token about new requested password.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSubjectLostPasswordToken'} = 'New OTRS password request';</code>

NotificationBodyLostPasswordToken

Description	Value
Description:	Defines the body text for notification mails sent to agents, with token about new requested password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'NotificationBodyLostPasswordToken'} = 'Hi <OTRS_USERFIRSTNAME>, You or someone impersonating you has requested to change your OTRS password. </pre>

Description	Value
	<p>If you want to do this, click on the link below. You will receive another email containing the password.</p> <pre><OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>index.pl? Action=LostPassword;Token=<OTRS_TOKEN></pre> <p>If you did not request a new password, please ignore this email.</p> <pre>';</pre>

NotificationSubjectLostPassword

Description	Value
Description:	Defines the subject for notification mails sent to agents, about new password.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'NotificationSubjectLostPassword'} = 'New OTRS password';</pre>

NotificationBodyLostPassword

Description	Value
Description:	Defines the body text for notification mails sent to agents, about new password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'NotificationBodyLostPassword'} = 'Hi <OTRS_USERFIRSTNAME>, Here\'s your new OTRS password. New password: <OTRS_NEWPW> You can log in via the following URL: <OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>index.pl ';</pre>

OpenMainMenuOnHover

Description	Value
Description:	If enabled, the first level of the main menu opens on mouse hover (instead of click only).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'OpenMainMenuOnHover'} = '0';</code>

Loader::Agent::Skin###000-default

Description	Value
Description:	Default skin for interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Loader::Agent::Skin'}->{'000-default'} = { 'Description' => 'This is the default orange - black skin.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'default', 'VisibleName' => 'Default' };</code>

Loader::Agent::Skin###001-ivory

Description	Value
Description:	Balanced white skin by Felix Niklas.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Loader::Agent::Skin'}->{'001-ivory'} = { 'Description' => 'Balanced white skin by Felix Niklas', 'HomePage' => 'www.felixniklas.de', 'InternalName' => 'ivory', 'VisibleName' => 'Ivory' };</code>

Loader::Agent::Skin###001-slim

Description	Value
Description:	Experimental "Slim" skin which tries to save screen space for power users.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Loader::Agent::Skin'}->{'001-slim'} = { 'Description' => 'Experimental "Slim" skin which tries to save screen space for power users.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'slim', 'VisibleName' => 'Slim' }; </pre>

Loader::Agent::DefaultSelectedSkin::HostBased

Description	Value
Description:	It is possible to configure different skins, for example to distinguish between different agents, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid skin on your system. Please see the example entries for the proper form of the regex.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self- >{'Loader::Agent::DefaultSelectedSkin::HostBased'} = { 'host1\\.example\\.com' => 'SomeSkin1', 'host2\\.example\\.com' => 'SomeSkin2' }; </pre>

Frontend::Agent::Dashboard

DashboardBackend###0000-ProductNotify

Description	Value
Description:	Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually.

Description	Value
	"CacheTTLLocal" defines the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0000-ProductNotify'} = { 'Block' => 'ContentLarge', 'CacheTTLLocal' => '1440', 'Default' => '1', 'Description' => 'News about OTRS releases!', 'Group' => 'admin', 'Module' => 'Kernel::Output::HTML::DashboardProductNotify', 'Title' => 'Product News', 'URL' => 'http://otrs.org/product.xml' }; </pre>

DashboardBackend###0400-UserOnline

Description	Value
Description:	Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" defines the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0400-UserOnline'} = { 'Block' => 'ContentSmall', 'CacheTTLLocal' => '5', 'Default' => '0', 'Description' => '', 'Filter' => 'Agent', 'Group' => '', 'IdleMinutes' => '60', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardUserOnline', 'ShowEmail' => '1', 'SortBy' => 'UserLastname', 'Title' => 'Online' } </pre>

Description	Value
	};

DashboardBackend###0410-RSS

Description	Value
Description:	Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0410-RSS'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '360', 'Default' => '1', 'Description' => '', 'Group' => '', 'Limit' => '6', 'Module' => 'Kernel::Output::HTML::DashboardRSS', 'Title' => 'OTRS News', 'URL' => 'http://otrs.org/rss/' }; </pre>

DashboardBackend###0200-Image

Description	Value
Description:	Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0200-Image'} = { 'Block' => 'ContentLarge', 'Default' => '1', 'Description' => 'Some picture description!', 'Group' => '', 'Height' => '140', 'Link' => 'http://otrs.org/', }; </pre>

Description	Value
	<pre>'LinkTitle' => 'http://otrs.org/', 'Module' => 'Kernel::Output::HTML::DashboardImage', 'Title' => 'A picture', 'URL' => 'http://www.otrs.com/uploads/pics/ jointhecommunity_02.jpg', 'Width' => '198' };</pre>

DashboardBackend###0210-MOTD

Description	Value
Description:	Shows the message of the day (MOTD) in the agent dashboard. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0210-MOTD'} = { 'Block' => 'ContentLarge', 'Default' => '1', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardMOTD', 'Title' => 'Message of the Day' };</pre>

DashboardBackend###0300-IFrame

Description	Value
Description:	Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0300-IFrame'} = { 'Align' => 'left', 'Block' => 'ContentLarge', 'Default' => '1', 'Description' => 'Some description!',</pre>

Description	Value
	<pre>'Frameborder' => '1', 'Group' => '', 'Height' => '800', 'Link' => 'http://otrs.org/', 'LinkTitle' => 'OTRS.org/', 'Marginheight' => '5', 'Marginwidth' => '5', 'Module' => 'Kernel::Output::HTML::DashboardIFrame', 'Scrolling' => 'auto', 'Title' => 'A Website', 'URL' => 'http://www.otrs.org/', 'Width' => '1024' };</pre>

Frontend::Agent::LinkObject

Frontend::AgentLinkObject::WildcardSearch

Description	Value
Description:	Starts a wildcard search of the active object after the link object mask is started.
Group:	Framework
SubGroup:	Frontend::Agent::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self-> >{'Frontend::AgentLinkObject::WildcardSearch'} = '0';</pre>

Frontend::Agent::ModuleMetaHead

Frontend::HeaderMetaModule###100-Refresh

Description	Value
Description:	Defines the module to generate html refresh headers of html sites.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleMetaHead
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::HeaderMetaModule'}->{'100-Refresh'} = { 'Module' => 'Kernel::Output::HTML::HeaderMetaRefresh'</pre>

Description	Value
	};

Frontend::Agent::ModuleNotify

Frontend::NotifyModule###200-UID-Check

Description	Value
Description:	Defines the module to display a notification in the agent interface, if the system is used by the admin user (normally you shouldn't work as admin).
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'200-UID-Check'} = { 'Module' => 'Kernel::Output::HTML::NotificationUIDCheck' };</pre>

Frontend::NotifyModule###300-ShowAgentOnline

Description	Value
Description:	Defines the module that shows all the currently logged in agents in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'300-ShowAgentOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationAgentOnline', 'ShowEmail' => '1' };</pre>

Frontend::NotifyModule###400-ShowCustomerOnline

Description	Value
Description:	Defines the module that shows all the currently logged in customers in the agent interface.
Group:	Framework

Description	Value
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'400-ShowCustomerOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationCustomerOnline', 'ShowEmail' => '1' };</pre>

Frontend::NotifyModule###500-OutofOffice-Check

Description	Value
Description:	Defines the module to display a notification in the agent interface, if the agent is logged in while having out-of-office active.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'500-OutofOffice-Check'} = { 'Module' => 'Kernel::Output::HTML::NotificationOutofOfficeCheck', };</pre>

Frontend::NotifyModule###900-Generic

Description	Value
Description:	Defines the module that shows a generic notify in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'900-Generic'} = { 'File' => '<OTRS_CONFIG_Home>/var/notify.txt', 'Link' => 'http://www.otrs.com', 'Module' => 'Kernel::Output::HTML::NotificationGeneric', 'Priority' => 'Warning', 'Text' => 'The OTRS Website' };</pre>

Frontend::Agent::ModuleRegistration

Frontend::Module###Logout

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'Logout'} = { 'Description' => 'Logout', 'NavBarName' => '', 'Title' => '' };</pre>

Frontend::Module###AgentDashboard

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentDashboard'} = { 'Description' => 'Agent Dashboard', 'Loader' => { 'JavaScript' => ['thirdparty/flot-0.7/excanvas.js', 'thirdparty/flot-0.7/jquery.flot.js', 'Core.UI.Chart.js', 'Core.UI.DnD.js', 'Core.Agent.Dashboard.js'] }, 'NavBar' => [{ 'AccessKey' => 'd', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AgentDashboard', 'LinkOption' => '', 'Name' => 'Dashboard', 'NavBar' => 'Dashboard', 'Prio' => '50', }] }</pre>

Description	Value
	<pre> 'Type' => 'Menu' }], 'NavBarName' => 'Dashboard', 'Title' => '' }; </pre>

Frontend::Module###AgentPreferences

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentPreferences'} = { 'Description' => 'Agent Preferences', 'NavBarName' => 'Preferences', 'Title' => '' }; </pre>

Frontend::Module###PictureUpload

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'PictureUpload'} = { 'Description' => 'Picture upload module', 'NavBarName' => 'Ticket', 'Title' => 'Picture-Upload' }; </pre>

Frontend::Module###AgentSpelling

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentSpelling'} = { 'Description' => 'Spell checker', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => '', 'Title' => 'Spell Checker' }; </pre>

Frontend::Module###SpellingInline

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'SpellingInline'} = { 'Description' => 'Spell checker', 'NavBarName' => '', 'Title' => 'Spell Checker' }; </pre>

Frontend::Module###AgentBook

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentBook'} = { 'Description' => 'Address book of CustomerUser sources', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', </pre>

Description	Value
	<pre> 'Core.Agent.TicketAction.js'] }, 'NavBarName' => '', 'Title' => 'Address Book' }; </pre>

Frontend::Module###AgentLinkObject

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentLinkObject'} = { 'Description' => 'Link Object', 'NavBarName' => '', 'Title' => 'Link Object' }; </pre>

Frontend::Module###AgentInfo

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentInfo'} = { 'Description' => 'Generic Info module', 'NavBarName' => '', 'Title' => 'Info' }; </pre>

Frontend::Module###AgentSearch

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentSearch'} = { 'Description' => 'Global Search Module', 'NavBarName' => '', 'Title' => 'Search' };</pre>

CustomerFrontend::Module###SpellingInline

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'SpellingInline'} = { 'Description' => 'Spell checker', 'NavBarName' => '', 'Title' => 'Spell Checker' };</pre>

Frontend::Module###AgentHTMLReference

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentHTMLReference'} = { 'Description' => 'HTML Reference', 'Group' => ['users'], 'GroupRole' => ['users'], 'Loader' => { 'CSS' => ['Core.Agent.HTMLReference.css'] } };</pre>

Description	Value
	<pre>'NavBarName' => '', 'Title' => 'HTML Reference' };</pre>

Frontend::Module###AgentStats

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentStats'} = { 'Description' => 'Stats', 'Group' => ['stats'], 'GroupRo' => ['stats'], 'Loader' => { 'JavaScript' => ['Core.Agent.Stats.js'] }, 'NavBar' => [{ 'AccessKey' => '', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AgentStats;Subaction=Overview', 'LinkOption' => '', 'Name' => 'Statistics', 'NavBar' => 'Stats', 'Prio' => '8500', 'Type' => 'Menu' }, { 'AccessKey' => '', 'Block' => '', 'Description' => 'Overview', 'GroupRo' => ['stats'], 'Link' => 'Action=AgentStats;Subaction=Overview', 'LinkOption' => '', 'Name' => 'Overview',</pre>

Description	Value
	<pre> 'NavBar' => 'Stats', 'Prio' => '100', 'Type' => '' }, { 'AccessKey' => '', 'Block' => '', 'Description' => 'New', 'Group' => ['stats'], 'Link' => 'Action=AgentStats;Subaction=Add', 'LinkOption' => '', 'Name' => 'New', 'NavBar' => 'Stats', 'Prio' => '200', 'Type' => '' }, { 'AccessKey' => '', 'Block' => '', 'Description' => 'Import', 'Group' => ['stats'], 'Link' => 'Action=AgentStats;Subaction=Import', 'LinkOption' => '', 'Name' => 'Import', 'NavBar' => 'Stats', 'Prio' => '300', 'Type' => '' }], 'NavBarName' => 'Stats', 'Title' => 'Stats' }; </pre>

Frontend::Agent::NavBarModule

Frontend::NavBarModule###6-CustomerCompany

Description	Value
Description:	Frontend module registration (disable company link if no company feature is used).
Group:	Framework
SubGroup:	Frontend::Agent::NavBarModule
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'Frontend::NavBarModule'}->{'6- CustomerCompany'} = { 'Module' => 'Kernel::Output::HTML::NavBarCustomerCompany' }; </pre>

Frontend::Agent::Preferences

PreferencesTableValue

Description	Value
Description:	Defines the name of the column to store the data in the preferences table.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'PreferencesTableValue'} = 'preferences_value'; </pre>

PreferencesTableUserID

Description	Value
Description:	Defines the name of the column to store the user identifier in the preferences table.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'PreferencesTableUserID'} = 'user_id'; </pre>

PreferencesView

Description	Value
Description:	Sets the display order of the different items in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'PreferencesView'} = ['User Profile', </pre>

Description	Value
	'Email Settings', 'Other Settings'];

PreferencesGroups###Password

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'Password'} = { 'Active' => '1', 'Area' => 'Agent', 'Column' => 'User Profile', 'Label' => 'Change password', 'Module' => 'Kernel::Output::HTML::PreferencesPassword', 'PasswordMaxLoginFailed' => '0', 'PasswordMin2Characters' => '0', 'PasswordMin2Lower2UpperCharacters' => '0', 'PasswordMinSize' => '0', 'PasswordNeedDigit' => '0', 'PasswordRegExp' => '', 'Prio' => '0500' }; </pre>

PreferencesGroups###SpellDict

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view. Take care to maintain the dictionaries installed in the system in the data section.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'SpellDict'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { 'deutsch' => 'Deutsch', 'english' => 'English' }, 'DataSelected' => 'english', }; </pre>

Description	Value
	<pre>'Key' => 'Default spelling dictionary', 'Label' => 'Spelling Dictionary', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSpellDict', 'Prio' => '2000' };</pre>

PreferencesGroups###Comment

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Comment'} = { 'Active' => '0', 'Block' => 'Input', 'Column' => 'Other Settings', 'Data' => '\$Env{"UserComment"}', 'Key' => 'Comment', 'Label' => 'Comment', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserComment', 'Prio' => '6000' };</pre>

PreferencesGroups###Language

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Language'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Frontend language', 'Label' => 'Language', 'Module' => 'Kernel::Output::HTML::PreferencesLanguage', 'PrefKey' => 'UserLanguage',</pre>

Description	Value
	'Prio' => '1000' };

PreferencesGroups###Skin

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Skin'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Wear this frontend skin', 'Label' => 'Skin', 'Module' => 'Kernel::Output::HTML::PreferencesSkin', 'PrefKey' => 'UserSkin', 'Prio' => '2000' };</pre>

PreferencesGroups###Theme

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Theme'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Frontend theme', 'Label' => 'Theme', 'Module' => 'Kernel::Output::HTML::PreferencesTheme', 'PrefKey' => 'UserTheme', 'Prio' => '3000' };</pre>

PreferencesGroups###OutOfOffice

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.

Description	Value
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'OutOfOffice'} = { 'Active' => '1', 'Block' => 'OutOfOffice', 'Column' => 'User Profile', 'Key' => '', 'Label' => 'Out Of Office Time', 'Module' => 'Kernel::Output::HTML::PreferencesOutOfOffice', 'PrefKey' => 'UserOutOfOffice', 'Prio' => '4000' }; </pre>

PreferencesGroups###TimeZone

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'TimeZone'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Time Zone', 'Label' => 'Time Zone', 'Module' => 'Kernel::Output::HTML::PreferencesTimeZone', 'PrefKey' => 'UserTimeZone', 'Prio' => '5000' }; </pre>

PreferencesGroups###CSVSeparator

Description	Value
Description:	Gives end users the possibility to override the separator character for CSV files, defined in the translation files.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'CSVSeparator'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Data' => { ' ' => ' ', ',' => ',', ';' => ';', '\\t' => 'tab', ' ' => ' ' }, 'DataSelected' => '0', 'Desc' => 'Select the separator character used in CSV files (stats and searches). If you don\'t select a separator here, the default separator for your language will be used.', 'Key' => 'CSV Separator', 'Label' => 'CSV Separator', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserCSVSeparator', 'Prio' => '4000' }; </pre>

Frontend::Agent::SearchRouter

Frontend::SearchDefault

Description	Value
Description:	Search backend default router.
Group:	Framework
SubGroup:	Frontend::Agent::SearchRouter
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::SearchDefault'} = 'Action=AgentTicketSearch;Subaction=AJAX'; </pre>

Frontend::Agent::Stats

Stats::SearchPageShown

Description	Value
Description:	Defines the default maximum number of search results shown on the overview page.
Group:	Framework

Description	Value
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::SearchPageShown'} = '20';</code>

Stats::DefaultSelectedDynamicObject

Description	Value
Description:	Defines the default selection at the drop down menu for dynamic objects (Form: Common Specification).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::DefaultSelectedDynamicObject'} = 'Ticket';</code>

Stats::DefaultSelectedPermissions

Description	Value
Description:	Defines the default selection at the drop down menu for permissions (Form: Common Specification).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::DefaultSelectedPermissions'} = ['stats'];</code>

Stats::DefaultSelectedFormat

Description	Value
Description:	Defines the default selection at the drop down menu for stats format (Form: Common Specification). Please insert the format key (see Stats::Format).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::DefaultSelectedFormat'} = [</code>

Description	Value
	'Print', 'CSV'];

Stats::SearchLimit

Description	Value
Description:	Defines the search limit for the stats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::SearchLimit'} = '500';</code>

Stats::Format

Description	Value
Description:	Defines all the possible stats output formats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Format'} = { 'CSV' => 'CSV', 'GD::Graph::area' => 'graph-area', 'GD::Graph::bars' => 'graph-bars', 'GD::Graph::hbars' => 'graph-hbars', 'GD::Graph::lines' => 'graph-lines', 'GD::Graph::linespoints' => 'graph-lines-points', 'GD::Graph::pie' => 'graph-pie', 'GD::Graph::points' => 'graph-points', 'Print' => 'Print' };</code>

Stats::GraphSize

Description	Value
Description:	Sets the size of the statistic graph.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre>\$Self->{'Stats::GraphSize'} = { '1200x800' => '1200x800', '1600x1200' => '1600x1200', '800x600' => '800x600' };</pre>

Stats::TimeType

Description	Value
Description:	Sets the time type which should be shown.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::TimeType'} = 'Extended';</pre>

Stats::ExchangeAxis

Description	Value
Description:	Allows agents to exchange the axis of a stat if they generate one.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::ExchangeAxis'} = '0';</pre>

Stats::UseAgentElementInStats

Description	Value
Description:	Allows agents to generate individual-related stats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::UseAgentElementInStats'} = '0';</pre>

Stats::CustomerIDAsMultiSelect

Description	Value
Description:	Shows all the customer identifiers in a multi-select field (not useful if you have a lot of customer identifiers).

Description	Value
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Stats::CustomerIDAsMultiSelect'} = '1';</code>

Frontend::Customer

CustomerHeadline

Description	Value
Description:	The headline shown in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerHeadline'} = 'Example Company Support';</code>

CustomerLogo

Description	Value
Description:	The logo shown in the header of the customer interface. The URL to the image can be a relative URL to the skin image directory, or a full URL to a remote web server.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerLogo'} = { 'StyleHeight' => '50px', 'StyleRight' => '25px', 'StyleTop' => '2px', 'StyleWidth' => '135px', 'URL' => 'skins/Customer/default/img/logo.png' };</code>

CustomerPanelUserID

Description	Value
Description:	Defines the user identifier for the customer panel.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelUserID'} = '1';</code>

CustomerGroupSupport

Description	Value
Description:	Activates support for customer groups.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerGroupSupport'} = '0';</code>

CustomerGroupAlwaysGroups

Description	Value
Description:	Defines the groups every customer user will be in (if CustomerGroupSupport is enabled and you don't want to manage every user for these groups).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerGroupAlwaysGroups'} = ['users'];</code>

CustomerPanelLoginURL

Description	Value
Description:	Defines an alternate login URL for the customer panel..
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'CustomerPanelLoginURL'} = 'http://host.example.com/cgi-bin/login.pl';</code>

CustomerPanelLogoutURL

Description	Value
Description:	Defines an alternate logout URL for the customer panel.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelLogoutURL'} = 'http://host.example.com/cgi-bin/login.pl';</code>

Frontend::CustomerUser::Item###1-GoogleMaps

Description	Value
Description:	Defines a customer item, which generates a google maps icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::CustomerUser::Item'}->{'1-GoogleMaps'} = { 'Attributes' => 'UserStreet;UserCity;UserCountry;', 'CSS' => 'Core.Agent.CustomerUser.GoogleMaps.css', 'CSSClass' => 'GoogleMaps', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserStreet;UserCity;', 'Target' => '_blank', 'Text' => 'Location', 'URL' => 'http://maps.google.com/maps?z=7&q=' };</code>

Frontend::CustomerUser::Item###2-Google

Description	Value
Description:	Defines a customer item, which generates a google icon at the end of a customer info block.
Group:	Framework

Description	Value
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::CustomerUser::Item'}->{'2- Google'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Google.css', 'CSSClass' => 'Google', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'Google', 'URL' => 'http://google.com/search?q=' }; </pre>

Frontend::CustomerUser::Item###2-LinkedIn

Description	Value
Description:	Defines a customer item, which generates a LinkedIn icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::CustomerUser::Item'}->{'2- LinkedIn'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Linkedin.css', 'CSSClass' => 'LinkedIn', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'LinkedIn', 'URL' => 'http://www.linkedin.com/commonSearch? type=people&keywords=' }; </pre>

Frontend::CustomerUser::Item###3-XING

Description	Value
Description:	Defines a customer item, which generates a XING icon at the end of a customer info block.
Group:	Framework

Description	Value
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::CustomerUser::Item'}->{'3- XING'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Xing.css', 'CSSClass' => 'Xing', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'XING', 'URL' => 'https://www.xing.com/app/search? op=search;keywords=' }; </pre>

CustomerPanelPreApplicationModule###CustomerAccept

Description	Value
Description:	This module and its PreRun() function will be executed, if defined, for every request. This module is useful to check some user options or to display news about new applications.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerPanelPreApplicationModule'}- >{'CustomerAccept'} = 'Kernel::Modules::CustomerAccept'; </pre>

CustomerPanel::InfoKey

Description	Value
Description:	Defines the key to check with CustomerAccept. If this user preferences key is true, then the message is accepted by the system.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerPanel::InfoKey'} = 'CustomerAccept1'; </pre>

CustomerPanel::InfoFile

Description	Value
Description:	Defines the path of the shown info file, that is located under Kernel/Output/HTML/Standard/CustomerAccept.dtl.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanel::InfoFile'} = 'CustomerAccept';</code>

CustomerPanelLostPassword

Description	Value
Description:	Activates lost password feature for customers.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelLostPassword'} = '1';</code>

CustomerPanelCreateAccount

Description	Value
Description:	Enables customers to create their own accounts.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelCreateAccount'} = '1';</code>

CustomerPanelSubjectLostPasswordToken

Description	Value
Description:	Defines the subject for notification mails sent to customers, with token about new requested password.
Group:	Framework
SubGroup:	Frontend::Customer

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSubjectLostPasswordToken'} = 'New OTRS password request';</code>

CustomerPanelBodyLostPasswordToken

Description	Value
Description:	Defines the body text for notification mails sent to customers, with token about new requested password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelBodyLostPasswordToken'} = 'Hi <OTRS_USERFIRSTNAME>, You or someone impersonating you has requested to change your OTRS password. If you want to do this, click on this link. You will receive another email containing the password. <OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl? Action=CustomerLostPassword;Token=<OTRS_TOKEN> If you did not request a new password, please ignore this email. '; </pre>

CustomerPanelSubjectLostPassword

Description	Value
Description:	Defines the subject for notification mails sent to customers, about new password.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'CustomerPanelSubjectLostPassword'} = 'New OTRS password';</code>

CustomerPanelBodyLostPassword

Description	Value
Description:	Defines the body text for notification mails sent to customers, about new password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelBodyLostPassword'} = 'Hi <OTRS_USERFIRSTNAME>, New password: <OTRS_NEWPW> <OTRS_CONFIG_HttpType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl '; </pre>

CustomerPanelSubjectNewAccount

Description	Value
Description:	Defines the subject for notification mails sent to customers, about new account.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelSubjectNewAccount'} = 'New OTRS Account!'; </pre>

CustomerPanelBodyNewAccount

Description	Value
Description:	Defines the body text for notification mails sent to customers, about new account.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre> \$Self->{'CustomerPanelBodyNewAccount'} = 'Hi <OTRS_USERFIRSTNAME>, You or someone impersonating you has created a new OTRS account for you. Full name: <OTRS_USERFIRSTNAME> <OTRS_USERLASTNAME> User name: <OTRS_USERLOGIN> Password : <OTRS_USERPASSWORD> You can log in via the following URL. We encourage you to change your password via the Preferences button after logging in. <OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl '; </pre>

Loader::Customer::Skin###000-default

Description	Value
Description:	Default skin for OTRS 3.0 interface.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Loader::Customer::Skin'}->{'000-default'} = { 'Description' => 'This is the default orange - black skin for OTRS 3.0.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'default', 'VisibleName' => 'Default' }; </pre>

Loader::Customer::SelectedSkin

Description	Value
Description:	The customer skin's InternalName which should be used in the customer interface. Please check the available skins in Frontend::Customer::Skins.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Loader::Customer::SelectedSkin'} = 'default';</code>

Frontend::Customer::Auth

Customer::AuthModule

Description	Value
Description:	Defines the module to authenticate customers.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::DB';</code>

Customer::AuthModule::DB::CryptType

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the crypt type of passwords must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CryptType'} = 'md5';</code>

Customer::AuthModule::DB::Table

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the name of the table where your customer data should be stored must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::Table'} = 'customer_user';</code>

Customer::AuthModule::DB::CustomerKey

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the name of the column for the CustomerKey in the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CustomerKey'} = 'login';</code>

Customer::AuthModule::DB::CustomerPassword

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the column name for the CustomerPassword in the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CustomerPassword'} = 'pw';</code>

Customer::AuthModule::DB::DSN

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the DSN for the connection to the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::DSN'} = 'DBI:mysql:database=customerdb;host=customerdbhost';</code>

Customer::AuthModule::DB::User

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, a username to connect to the customer table can be specified.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::User'} = 'some_user';</code>

Customer::AuthModule::DB::Password

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, a password to connect to the customer table can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::Password'} = 'some_password';</code>

Customer::AuthModule::DB::Type

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, a database driver (normally autodetection is used) can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::Type'} = 'mysql';</code>

Customer::AuthModule::HTTPBasicAuth::Replace

Description	Value
Description:	If "HTTPBasicAuth" was selected for Customer::AuthModule, you can specify to strip leading parts of user names (e. g. for domains like example_domain\user to user).
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::HTTPBasicAuth::Replace'} = 'example_domain\\\\\\';</pre>

Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp

Description	Value
Description:	If "HTTPBasicAuth" was selected for Customer::AuthModule, you can specify (by using a RegExp) to strip parts of REMOTE_USER (e. g. for to remove trailing domains). RegExp-Note, \$1 will be the new Login.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp'} = '^(.+?)@.+?\$';</pre>

Customer::AuthModule::LDAP::Host

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, the LDAP host can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::Host'} = 'ldap.example.com';</pre>

Customer::AuthModule::LDAP::BaseDN

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, the BaseDN must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0

Description	Value
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';</code>

Customer::AuthModule::LDAP::UID

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, the user identifier must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::UID'} = 'uid';</code>

Customer::AuthModule::LDAP::GroupDN

Description	Value
Description:	If "LDAP" was selected for Customer::Authmodule, you can check if the user is allowed to authenticate because he is in a posixGroup, e.g. user needs to be in a group xyz to use OTRS. Specify the group, who may access the system.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::GroupDN'} = 'cn=otrsallow,ou=posixGroups,dc=example,dc=com';</code>

Customer::AuthModule::LDAP::AccessAttr

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, you can specify access attributes here.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::AccessAttr'} = 'memberUid';</code>

Customer::AuthModule::LDAP::UserAttr

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, user attributes can be specified. For LDAP posixGroups use UID, for non LDAP posixGroups use full user DN.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{ 'Customer::AuthModule::LDAP::UserAttr' } = 'UID' ;</code>

Customer::AuthModule::LDAP::SearchUserDN

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule and your users have only anonymous access to the LDAP tree, but you want to search through the data, you can do this with a user who has access to the LDAP directory. Specify the username for this special user here.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{ 'Customer::AuthModule::LDAP::SearchUserDN' } = 'cn=binduser,ou=users,dc=example,dc=com' ;</code>

Customer::AuthModule::LDAP::SearchUserPw

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule and your users have only anonymous access to the LDAP tree, but you want to search through the data, you can do this with a user who has access to the LDAP directory. Specify the password for this special user here.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{ 'Customer::AuthModule::LDAP::SearchUserPw' } = 'some_password' ;</code>

Customer::AuthModule::LDAP::AlwaysFilter

Description	Value
Description:	If "LDAP" was selected, you can add a filter to each LDAP query, e.g. (mail=*), (objectclass=user) or (!objectclass=computer).
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{ 'Customer::AuthModule::LDAP::AlwaysFilter' } = '(!objectclass=computer)';</pre>

Customer::AuthModule::LDAP::UserSuffix

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule and if you want to add a suffix to every customer login name, specify it here, e. g. you just want to write the username user but in your LDAP directory exists user@domain.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{ 'Customer::AuthModule::LDAP::UserSuffix' } = '@domain.com';</pre>

Customer::AuthModule::LDAP::Params

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule and special paramaters are needed for the Net::LDAP perl module, you can specify them here. See "perldoc Net::LDAP" for more information about the parameters.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{ 'Customer::AuthModule::LDAP::Params' } = { 'async' => '0', 'port' => '389', 'timeout' => '120', 'version' => '3' };</pre>

Customer::AuthModule::LDAP::Die

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, you can specify if the applications will stop if e. g. a connection to a server can't be established due to network problems.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::Die'} = '1';</code>

Customer::AuthModule::Radius::Host

Description	Value
Description:	If "Radius" was selected for Customer::AuthModule, the radius host must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::Radius::Host'} = 'radiushost';</code>

Customer::AuthModule::Radius::Password

Description	Value
Description:	If "Radius" was selected for Customer::AuthModule, the password to authenticate to the radius host must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::Radius::Password'} = 'radiussecret';</code>

Customer::AuthModule::Radius::Die

Description	Value
Description:	If "Radius" was selected for Customer::AuthModule, you can specify if the applications will stop if e. g. a connection to a server can't be established due to network problems.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::Radius::Die'} = '1';</pre>

Frontend::Customer::ModuleMetaHead

CustomerFrontend::HeaderMetaModule###1-Refresh

Description	Value
Description:	Defines the module to generate html refresh headers of html sites, in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleMetaHead
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::HeaderMetaModule'}->{'1-Refresh'} = { 'Module' => 'Kernel::Output::HTML::HeaderMetaRefresh' };</pre>

Frontend::Customer::ModuleNotify

CustomerFrontend::NotifyModule###1-ShowAgentOnline

Description	Value
Description:	Defines the module that shows the currently logged in agents in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowAgentOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationAgentOnline', 'ShowEmail' => '1' };</pre>

CustomerFrontend::NotifyModule###1-ShowCustomerOnline

Description	Value
Description:	Defines the module that shows the currently logged in customers in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowCustomerOnline'} = { 'Module' => 'Kernel::Output::HTML::NotificationCustomerOnline', 'ShowEmail' => '1' }; </pre>

Frontend::Customer::ModuleRegistration

CustomerFrontend::Module###Logout

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::Module'}->{'Logout'} = { 'Description' => 'Logout of customer panel', 'NavBarName' => '', 'Title' => '' }; </pre>

CustomerFrontend::Module###CustomerPreferences

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self->{'CustomerFrontend::Module'}- >{'CustomerPreferences'} = { 'Description' => 'Customer preferences', 'NavBarName' => '', 'Title' => 'Preferences' };</pre>

CustomerFrontend::Module###CustomerAccept

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}- >{'CustomerAccept'} = { 'Description' => 'To accept login information, such as an EULA or license.', 'NavBarName' => '', 'Title' => 'Info' };</pre>

CustomerFrontend::Module###PictureUpload

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}- >{'PictureUpload'} = { 'Description' => 'Picture upload module', 'NavBarName' => 'Ticket', 'Title' => 'Picture-Upload' };</pre>

Frontend::Customer::Preferences

PreferencesTable

Description	Value
Description:	Defines the name of the table, where the customer preferences are stored.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PreferencesTable'} = 'user_preferences';</code>

PreferencesTableKey

Description	Value
Description:	Defines the column to store the keys for the preferences table.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PreferencesTableKey'} = 'preferences_key';</code>

CustomerPreferences

Description	Value
Description:	Defines the parameters for the customer preferences table.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerPreferences'} = { 'Module' => 'Kernel::System::CustomerUser::Preferences::DB', 'Params' => { 'Table' => 'customer_preferences', 'TableKey' => 'preferences_key', 'TableUserID' => 'user_id', 'TableValue' => 'preferences_value' } };</pre>

CustomerPreferencesView

Description	Value
Description:	Sets the order of the different items in the customer preferences view.
Group:	Framework

Description	Value
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerPreferencesView'} = ['User Profile', 'Other Settings'];</pre>

CustomerPreferencesGroups###Password

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Password'} = { 'Active' => '1', 'Area' => 'Customer', 'Column' => 'Other Settings', 'Label' => 'Change password', 'Module' => 'Kernel::Output::HTML::PreferencesPassword', 'PasswordMin2Characters' => '0', 'PasswordMin2Lower2UpperCharacters' => '0', 'PasswordMinSize' => '0', 'PasswordNeedDigit' => '0', 'PasswordRegExp' => '', 'Prio' => '1000' };</pre>

CustomerPreferencesGroups###Language

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Language'} = {</pre>

Description	Value
	<pre>'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Your language', 'Label' => 'Interface language', 'Module' => 'Kernel::Output::HTML::PreferencesLanguage', 'PrefKey' => 'UserLanguage', 'Prio' => '2000' };</pre>

CustomerPreferencesGroups###Theme

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Theme'} = { 'Active' => '0', 'Column' => 'User Profile', 'Key' => 'Select your frontend Theme.', 'Label' => 'Theme', 'Module' => 'Kernel::Output::HTML::PreferencesTheme', 'PrefKey' => 'UserTheme', 'Prio' => '1000' };</pre>

CustomerPreferencesGroups###TimeZone

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'TimeZone'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Time Zone', 'Label' => 'Time Zone',</pre>

Description	Value
	<pre>'Module' => 'Kernel::Output::HTML::PreferencesTimeZone', 'PrefKey' => 'UserTimeZone', 'Prio' => '5000' };</pre>

CustomerPreferencesGroups###PGP

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'PGP'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Key' => 'PGP Key Upload', 'Label' => 'PGP Key', 'Module' => 'Kernel::Output::HTML::PreferencesPGP', 'PrefKey' => 'UserPGPKey', 'Prio' => '10000' };</pre>

CustomerPreferencesGroups###SMIME

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'SMIME'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Key' => 'S/MIME Certificate Upload', 'Label' => 'S/MIME Certificate', 'Module' => 'Kernel::Output::HTML::PreferencesSMIME', 'PrefKey' => 'UserSMIMEKey', 'Prio' => '11000' };</pre>

Frontend::Public

PublicFrontend::CommonParam###Action

Description	Value
Description:	Defines the default value for the action parameter for the public frontend. The action parameter is used in the scripts of the system.
Group:	Framework
SubGroup:	Frontend::Public
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PublicFrontend::CommonParam'}->{'Action'} = 'PublicDefault';</pre>

Frontend::Public::ModuleRegistration

PublicFrontend::Module###PublicDefault

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Public::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PublicFrontend::Module'}- >{'PublicDefault'} = { 'Description' => 'PublicDefault', 'NavBarName' => '', 'Title' => 'PublicDefault' };</pre>

PublicFrontend::Module###PublicRepository

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Public::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PublicFrontend::Module'}- >{'PublicRepository'} = { 'Description' => 'PublicRepository',</pre>

Description	Value
	<pre>'NavBarName' => '', 'Title' => 'PublicRepository' };</pre>

Ticket

Core

OTRSEscalationEvents::DecayTime

Description	Value
Description:	The duration in minutes after emitting an event, in which the new escalation notify and start events are suppressed.
Group:	Ticket
SubGroup:	Core
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'OTRSEscalationEvents::DecayTime'} = '1440';</pre>

Core::FulltextSearch

Ticket::SearchIndexModule

Description	Value
Description:	Helps to extend your articles full-text search (From, To, Cc, Subject and Body search). Runtime will do full-text searches on live data (it works fine for up to 50.000 tickets). StaticDB will strip all articles and will build an index after article creation, increasing fulltext searches about 50%. To create an initial index use "bin/otrs.RebuildFulltextIndex.pl".
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::SearchIndexModule'} = 'Kernel::System::Ticket::ArticleSearchIndex::RuntimeDB';</pre>

Ticket::SearchIndex::Attribute

Description	Value
Description:	Configures the full-text index. Execute "bin/otrs.RebuildFulltextIndex.pl" in order to generate a new index.

Description	Value
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::SearchIndex::Attribute'} = { 'WordCountMax' => '1000', 'WordLengthMax' => '30', 'WordLengthMin' => '3' };</pre>

Ticket::EventModulePost###98-ArticleSearchIndex

Description	Value
Description:	Builds an article index right after the article's creation.
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'98- ArticleSearchIndex'} = { 'Event' => '(ArticleCreate ArticleUpdate)', 'Module' => 'Kernel::System::Ticket::Event::ArticleSearchIndex' };</pre>

Core::LinkObject

LinkObject::PossibleLink###0200

Description	Value
Description:	Links 2 tickets with a "Normal" type link.
Group:	Ticket
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::PossibleLink'}->{'0200'} = { 'Object1' => 'Ticket', 'Object2' => 'Ticket', 'Type' => 'Normal' };</pre>

LinkObject::PossibleLink###0201

Description	Value
Description:	Links 2 tickets with a "ParentChild" type link.
Group:	Ticket
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'LinkObject::PossibleLink'}->{'0201'} = { 'Object1' => 'Ticket', 'Object2' => 'Ticket', 'Type' => 'ParentChild' }; </pre>

Core::PostMaster

PostmasterMaxEmails

Description	Value
Description:	Maximal auto email responses to own email-address a day (Loop-Protection).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'PostmasterMaxEmails'} = '40'; </pre>

PostMasterMaxEmailSize

Description	Value
Description:	Maximal size in KBytes for mails that can be fetched via POP3/POP3S/IMAP/IMAPS (KBytes).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'PostMasterMaxEmailSize'} = '16384'; </pre>

PostMasterReconnectMessage

Description	Value
Description:	The "bin/PostMasterMailAccount.pl" will reconnect to POP3/POP3S/IMAP/IMAPS host after the specified count of messages.

Description	Value
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostMasterReconnectMessage'} = '20';</code>

LoopProtectionModule

Description	Value
Description:	Default loop protection module.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LoopProtectionModule'} = 'Kernel::System::PostMaster::LoopProtection::DB';</code>

LoopProtectionLog

Description	Value
Description:	Path for the log file (it only applies if "FS" was selected for LoopProtectionModule and it is mandatory).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LoopProtectionLog'} = '<OTRS_CONFIG_Home>/var/log/LoopProtection';</code>

PostmasterAutoHTML2Text

Description	Value
Description:	Converts HTML mails into text messages.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterAutoHTML2Text'} = '1';</code>

PostmasterFollowUpSearchInReferences

Description	Value
Description:	Executes follow up checks on In-Reply-To or References headers for mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInReferences'} = '0';</code>

PostmasterFollowUpSearchInBody

Description	Value
Description:	Executes follow up mail body checks in mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInBody'} = '0';</code>

PostmasterFollowUpSearchInAttachment

Description	Value
Description:	Executes follow up mail attachments checks in mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInAttachment'} = '0';</code>

PostmasterFollowUpSearchInRaw

Description	Value
Description:	Executes follow up plain/raw mail checks in mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInRaw'} = '0';</code>

PostmasterUserID

Description	Value
Description:	Specifies user id of the postmaster data base.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterUserID'} = '1';</code>

PostmasterDefaultQueue

Description	Value
Description:	Defines the postmaster default queue.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultQueue'} = 'Raw';</code>

PostmasterDefaultPriority

Description	Value
Description:	Defines the default priority of new tickets.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultPriority'} = '3 normal';</code>

PostmasterDefaultState

Description	Value
Description:	Defines the default state of new tickets.
Group:	Ticket

Description	Value
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultState'} = 'new';</code>

PostmasterFollowUpState

Description	Value
Description:	Defines the state of a ticket if it gets a follow-up.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpState'} = 'open';</code>

PostmasterFollowUpStateClosed

Description	Value
Description:	Defines the state of a ticket if it gets a follow-up and the ticket was already closed.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PostmasterFollowUpStateClosed'} = 'open';</code>

PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner

Description	Value
Description:	Sends agent follow-up notification only to the owner, if a ticket is unlocked (the default is to send the notification to all agents).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner'} = '0';</code>

PostmasterX-Header

Description	Value
Description:	Defines all the X-headers that should be scanned.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'PostmasterX-Header'} = ['From', 'To', 'Cc', 'Reply-To', 'ReplyTo', 'Subject', 'Message-ID', 'Message-Id', 'Resent-To', 'Resent-From', 'Precedence', 'Mailing-List', 'List-Id', 'List-Archive', 'Errors-To', 'References', 'In-Reply-To', 'X-Loop', 'X-Spam-Flag', 'X-Spam-Status', 'X-Spam-Level', 'X-No-Loop', 'X-Priority', 'Importance', 'X-Mailer', 'User-Agent', 'Organization', 'X-Original-To', 'Delivered-To', 'Envelope-To', 'Return-Path', 'X-OTRS-Loop', 'X-OTRS-Info', 'X-OTRS-Priority', 'X-OTRS-Queue', 'X-OTRS-Lock', 'X-OTRS-Ignore', 'X-OTRS-State', 'X-OTRS-State-PendingTime', 'X-OTRS-Type', 'X-OTRS-Service', </pre>

Referenz der Konfigurationsoptionen

Description	Value
	'X-OTRS-SLA', 'X-OTRS-CustomerNo', 'X-OTRS-CustomerUser', 'X-OTRS-ArticleKey1', 'X-OTRS-ArticleKey2', 'X-OTRS-ArticleKey3', 'X-OTRS-ArticleValue1', 'X-OTRS-ArticleValue2', 'X-OTRS-ArticleValue3', 'X-OTRS-SenderType', 'X-OTRS-ArticleType', 'X-OTRS-TicketKey1', 'X-OTRS-TicketKey2', 'X-OTRS-TicketKey3', 'X-OTRS-TicketKey4', 'X-OTRS-TicketKey5', 'X-OTRS-TicketKey6', 'X-OTRS-TicketKey7', 'X-OTRS-TicketKey8', 'X-OTRS-TicketKey9', 'X-OTRS-TicketKey10', 'X-OTRS-TicketKey11', 'X-OTRS-TicketKey12', 'X-OTRS-TicketKey13', 'X-OTRS-TicketKey14', 'X-OTRS-TicketKey15', 'X-OTRS-TicketKey16', 'X-OTRS-TicketValue1', 'X-OTRS-TicketValue2', 'X-OTRS-TicketValue3', 'X-OTRS-TicketValue4', 'X-OTRS-TicketValue5', 'X-OTRS-TicketValue6', 'X-OTRS-TicketValue7', 'X-OTRS-TicketValue8', 'X-OTRS-TicketValue9', 'X-OTRS-TicketValue10', 'X-OTRS-TicketValue11', 'X-OTRS-TicketValue12', 'X-OTRS-TicketValue13', 'X-OTRS-TicketValue14', 'X-OTRS-TicketValue15', 'X-OTRS-TicketValue16', 'X-OTRS-TicketTime1', 'X-OTRS-TicketTime2', 'X-OTRS-TicketTime3', 'X-OTRS-TicketTime4', 'X-OTRS-TicketTime5', 'X-OTRS-TicketTime6', 'X-OTRS-FollowUp-Priority', 'X-OTRS-FollowUp-Queue', 'X-OTRS-FollowUp-Lock',

Referenz der Konfigurationsoptionen

Description	Value
	<pre>'X-OTRS-FollowUp-State', 'X-OTRS-FollowUp-State-PendingTime', 'X-OTRS-FollowUp-Type', 'X-OTRS-FollowUp-Service', 'X-OTRS-FollowUp-SLA', 'X-OTRS-FollowUp-ArticleKey1', 'X-OTRS-FollowUp-ArticleKey2', 'X-OTRS-FollowUp-ArticleKey3', 'X-OTRS-FollowUp-ArticleValue1', 'X-OTRS-FollowUp-ArticleValue2', 'X-OTRS-FollowUp-ArticleValue3', 'X-OTRS-FollowUp-SenderType', 'X-OTRS-FollowUp-ArticleType', 'X-OTRS-FollowUp-TicketKey1', 'X-OTRS-FollowUp-TicketKey2', 'X-OTRS-FollowUp-TicketKey3', 'X-OTRS-FollowUp-TicketKey4', 'X-OTRS-FollowUp-TicketKey5', 'X-OTRS-FollowUp-TicketKey6', 'X-OTRS-FollowUp-TicketKey7', 'X-OTRS-FollowUp-TicketKey8', 'X-OTRS-FollowUp-TicketKey9', 'X-OTRS-FollowUp-TicketKey10', 'X-OTRS-FollowUp-TicketKey11', 'X-OTRS-FollowUp-TicketKey12', 'X-OTRS-FollowUp-TicketKey13', 'X-OTRS-FollowUp-TicketKey14', 'X-OTRS-FollowUp-TicketKey15', 'X-OTRS-FollowUp-TicketKey16', 'X-OTRS-FollowUp-TicketValue1', 'X-OTRS-FollowUp-TicketValue2', 'X-OTRS-FollowUp-TicketValue3', 'X-OTRS-FollowUp-TicketValue4', 'X-OTRS-FollowUp-TicketValue5', 'X-OTRS-FollowUp-TicketValue6', 'X-OTRS-FollowUp-TicketValue7', 'X-OTRS-FollowUp-TicketValue8', 'X-OTRS-FollowUp-TicketValue9', 'X-OTRS-FollowUp-TicketValue10', 'X-OTRS-FollowUp-TicketValue11', 'X-OTRS-FollowUp-TicketValue12', 'X-OTRS-FollowUp-TicketValue13', 'X-OTRS-FollowUp-TicketValue14', 'X-OTRS-FollowUp-TicketValue15', 'X-OTRS-FollowUp-TicketValue16', 'X-OTRS-FollowUp-TicketTime1', 'X-OTRS-FollowUp-TicketTime2', 'X-OTRS-FollowUp-TicketTime3', 'X-OTRS-FollowUp-TicketTime4', 'X-OTRS-FollowUp-TicketTime5', 'X-OTRS-FollowUp-TicketTime6'];</pre>

PostMaster::PreFilterModule###1-Match

Description	Value
Description:	Module to filter and manipulate incoming messages. Block/ignore all spam email with From: noreply@ address.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'PostMaster::PreFilterModule'}->{'1-Match'} = { 'Match' => { 'From' => 'noreply@' }, 'Module' => 'Kernel::System::PostMaster::Filter::Match', 'Set' => { 'X-OTRS-Ignore' => 'yes' }, 'StopAfterMatch' => '0' }; </pre>

PostMaster::PreFilterModule###2-Match

Description	Value
Description:	Module to filter and manipulate incoming messages. Get a 4 digit number to ticket free text, use regex in Match e. g. From => '(.+?)@.+?', and use () as [***] in Set =>.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'PostMaster::PreFilterModule'}->{'2-Match'} = { 'Match' => { 'Subject' => 'SomeNumber:(\\d\\d\\d\\d)' }, 'Module' => 'Kernel::System::PostMaster::Filter::Match', 'Set' => { 'X-OTRS-TicketKey-1' => 'SomeNumber', 'X-OTRS-TicketValue-1' => '[***]' }, 'StopAfterMatch' => '0' }; </pre>

PostMaster::PreFilterModule###3-NewTicketReject

Description	Value
Description:	Blocks all the incoming emails that do not have a valid ticket number in subject with From: @example.com address.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'PostMaster::PreFilterModule'}->{'3- NewTicketReject'} = { 'Match' => { 'From' => '@example.com' }, 'Module' => 'Kernel::System::PostMaster::Filter::NewTicketReject', 'Set' => { 'X-OTRS-Ignore' => 'yes' }, 'StopAfterMatch' => '0' }; </pre>

PostMaster::PreFilterModule::NewTicketReject::Sender

Description	Value
Description:	Defines the sender for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self- >{'PostMaster::PreFilterModule::NewTicketReject::Sender'} = 'noreply@example.com'; </pre>

PostMaster::PreFilterModule::NewTicketReject::Subject

Description	Value
Description:	Defines the subject for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre>\$Self- >{'PostMaster::PreFilterModule::NewTicketReject::Subject'} = 'Email Rejected';</pre>

PostMaster::PreFilterModule::NewTicketReject::Body

Description	Value
Description:	Defines the body text for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'PostMaster::PreFilterModule::NewTicketReject::Body'} = ' Dear Customer, Unfortunately we could not detect a valid ticket number in your subject, so this email can\'t be processed. Please create a new ticket via the customer panel. Thanks for your help! Your Helpdesk Team ';</pre>

PostMaster::PreFilterModule###4-CMD

Description	Value
Description:	CMD example setup. Ignores emails where external CMD returns some output on STDOUT (email will be piped into STDIN of some.bin).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'4-CMD'} = { 'CMD' => '/usr/bin/some.bin', 'Module' => 'Kernel::System::PostMaster::Filter::CMD', 'Set' => { 'X-OTRS-Ignore' => 'yes' } };</pre>

PostMaster::PreFilterModule###5-SpamAssassin

Description	Value
Description:	Spam Assassin example setup. Ignores emails that are marked with SpamAssassin.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'PostMaster::PreFilterModule'}->{'5-SpamAssassin'} = { 'CMD' => '/usr/bin/spamassassin grep -i "X-Spam-Status: yes"', 'Module' => 'Kernel::System::PostMaster::Filter::CMD', 'Set' => { 'X-OTRS-Ignore' => 'yes' } }; </pre>

PostMaster::PreFilterModule###6-SpamAssassin

Description	Value
Description:	Spam Assassin example setup. Moves marked mails to spam queue.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'PostMaster::PreFilterModule'}->{'6-SpamAssassin'} = { 'CMD' => '/usr/bin/spamassassin grep -i "X-Spam-Status: yes"', 'Module' => 'Kernel::System::PostMaster::Filter::CMD', 'Set' => { 'X-OTRS-Queue' => 'spam' } }; </pre>

PostMaster::PreFilterModule###000-MatchDBSource

Description	Value
Description:	Module to use database filter storage.
Group:	Ticket
SubGroup:	Core::PostMaster

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'000-MatchDBSource'} = { 'Module' => 'Kernel::System::PostMaster::Filter::MatchDBSource' };</pre>

PostMaster::PostFilterModule###000-FollowUpArticleTypeCheck

Description	Value
Description:	Module to check if arrived emails should be marked as email-internal (because of original forwarded internal email it college). ArticleType and SenderType define the values for the arrived email/article.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PostFilterModule'}->{'000-FollowUpArticleTypeCheck'} = { 'ArticleType' => 'email-internal', 'Module' => 'Kernel::System::PostMaster::Filter::FollowUpArticleTypeCheck', 'SenderType' => 'customer' };</pre>

SendNoAutoResponseRegExp

Description	Value
Description:	If this regex matches, no message will be send by the autoresponder.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'SendNoAutoResponseRegExp'} = '(MAILER-DAEMON postmaster abuse)@.+?\.\.+?';</pre>

Core::Stats

Stats::DynamicObjectRegistration###Ticket

Description	Value
Description:	Module to generate ticket statistics.

Description	Value
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}- >{'Ticket'} = { 'Module' => 'Kernel::System::Stats::Dynamic::Ticket' };</pre>

Stats::DynamicObjectRegistration###TicketList

Description	Value
Description:	Determines if the statistics module may generate ticket lists.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}- >{'TicketList'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketList' };</pre>

Stats::DynamicObjectRegistration###TicketAccountedTime

Description	Value
Description:	Module to generate accounted time ticket statistics.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}- >{'TicketAccountedTime'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketAccountedTime' };</pre>

Stats::DynamicObjectRegistration###TicketSolutionResponseTime

Description	Value
Description:	Module to generate ticket solution and response time statistics.

Description	Value
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}- >{'TicketSolutionResponseTime'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketSolutionResponseTime' };</pre>

Core::Ticket

Ticket::Hook

Description	Value
Description:	The identifier for a ticket, e.g. Ticket#, Call#, MyTicket#. The default is Ticket#.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Hook'} = 'Ticket#';</pre>

Ticket::HookDivider

Description	Value
Description:	The divider between TicketHook and ticket number. E.g ': '.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::HookDivider'} = ' ';</pre>

Ticket::SubjectSize

Description	Value
Description:	Max size of the subjects in an email reply.
Group:	Ticket
SubGroup:	Core::Ticket

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectSize'} = '100';</code>

Ticket::SubjectRe

Description	Value
Description:	The text at the beginning of the subject in an email reply, e.g. RE, AW, or AS.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectRe'} = 'Re';</code>

Ticket::SubjectFwd

Description	Value
Description:	The text at the beginning of the subject when an email is forwarded, e.g. FW, Fwd, or WG.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectFwd'} = 'Fwd';</code>

Ticket::SubjectFormat

Description	Value
Description:	The format of the subject. 'Left' means '[TicketHook#:12345] Some Subject', 'Right' means 'Some Subject [TicketHook#:12345]', 'None' means 'Some Subject' and no ticket number. In the last case you should enable <code>PostmasterFollowupSearchInRaw</code> or <code>PostmasterFollowUpSearchInReferences</code> to recognize followups based on email headers and/or body.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::SubjectFormat'} = 'Left';</code>

Ticket::CustomQueue

Description	Value
Description:	Name of custom queue. The custom queue is a queue selection of your preferred queues and can be selected in the preferences settings.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomQueue'} = 'My Queues';</code>

Ticket::NewArticleIgnoreSystemSender

Description	Value
Description:	Ignore article with system sender type for new article feature (e. g. auto responses or email notifications).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NewArticleIgnoreSystemSender'} = '0';</code>

Ticket::ChangeOwnerToEveryone

Description	Value
Description:	Changes the owner of tickets to everyone (useful for ASP). Normally only agent with rw permissions in the queue of the ticket will be shown.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ChangeOwnerToEveryone'} = '0';</code>

Ticket::Responsible

Description	Value
Description:	Enables ticket responsible feature, to keep track of a specific ticket.

Description	Value
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Responsible'} = '0';</code>

Ticket::ResponsibleAutoSet

Description	Value
Description:	Automatically sets the owner of a ticket as the responsible for it (if ticket responsible feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::ResponsibleAutoSet'} = '1';</code>

Ticket::Type

Description	Value
Description:	Allows defining new types for ticket (if ticket type feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Type'} = '0';</code>

Ticket::Service

Description	Value
Description:	Allows defining services and SLAs for tickets (e. g. email, desktop, network, ...), and escalation attributes for SLAs (if ticket service/SLA feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Service'} = '0';</code>

Ticket::ArchiveSystem

Description	Value
Description:	Activates the ticket archive system to have a faster system by moving some tickets out of the daily scope. To search for these tickets, the archive flag has to be enabled in the ticket search.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ArchiveSystem'} = '0';</code>

Ticket::CustomerArchiveSystem

Description	Value
Description:	Activates the ticket archive system search in the customer interface.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerArchiveSystem'} = '0';</code>

Ticket::NumberGenerator

Description	Value
Description:	Selects the ticket number generator module. "AutoIncrement" increments the ticket number, the SystemID and the counter are used with SystemID.counter format (e.g. 1010138, 1010139). With "Date" the ticket numbers will be generated by the current date, the SystemID and the counter. The format looks like Year.Month.Day.SystemID.counter (e.g. 200206231010138, 200206231010139). With "DateChecksum" the counter will be appended as checksum to the string of date and SystemID. The checksum will be rotated on a daily basis. The format looks like Year.Month.Day.SystemID.Counter.CheckSum (e.g. 2002070110101520, 2002070110101535). "Random" generates randomized ticket numbers in the format "SystemID.Random" (e.g. 100057866352, 103745394596).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator'} = 'Kernel::System::Ticket::Number::DateChecksum';</code>

Ticket::NumberGenerator::MinCounterSize

Description	Value
Description:	Sets the minimal ticket counter size (if "AutoIncrement" was selected as TicketNumberGenerator). Default is 5, this means the counter starts from 10000.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator::MinCounterSize'} = '5';</code>

Ticket::NumberGenerator::CheckSystemID

Description	Value
Description:	Checks the SystemID in ticket number detection for follow-ups (use "No" if SystemID has been changed after using the system).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator::CheckSystemID'} = '1';</code>

Ticket::CounterLog

Description	Value
Description:	Log file for the ticket counter.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CounterLog'} = '<OTRS_CONFIG_Home>/var/log/TicketCounter.log';</code>

Ticket::IndexModule

Description	Value
Description:	IndexAccelerator: to choose your backend TicketViewAccelerator module. "RuntimeDB" generates each queue view on the fly from ticket table (no performance problems up to approx. 60.000 tickets in total and 6.000 open tickets in the system). "StaticDB" is the most powerful module, it uses an extra ticket-index table that works like a view (recommended if more than 80.000 and 6.000 open tickets are stored in the system). Use the script "bin/otrs.RebuildTicketIndex.pl" for initial index update.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::IndexModule'} = 'Kernel::System::Ticket::IndexAccelerator::RuntimeDB';</pre>

Ticket::StorageModule

Description	Value
Description:	Saves the attachments of articles. "DB" stores all data in the database (not recommended for storing big attachments). "FS" stores the data on the filesystem; this is faster but the webserver should run under the OTRS user. You can switch between the modules even on a system that is already in production without any loss of data.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::StorageModule'} = 'Kernel::System::Ticket::ArticleStorageDB';</pre>

ArticleDir

Description	Value
Description:	Specifies the directory to store the data in, if "FS" was selected for TicketStorageModule.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'ArticleDir'} = '<OTRS_CONFIG_Home>/var/ article';</pre>

Ticket::EventModulePost###100-ArchiveRestore

Description	Value
Description:	Restores a ticket from the archive (only if the event is a state change, from closed to any open available state).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'100-ArchiveRestore'} = { 'Event' => 'TicketStateUpdate', 'Module' => 'Kernel::System::Ticket::Event::ArchiveRestore' }; </pre>

Ticket::EventModulePost###110-AcceleratorUpdate

Description	Value
Description:	Updates the ticket index accelerator.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'110-AcceleratorUpdate'} = { 'Event' => 'TicketStateUpdate TicketQueueUpdate TicketLockUpdate', 'Module' => 'Kernel::System::Ticket::Event::TicketAcceleratorUpdate' }; </pre>

Ticket::EventModulePost###120-ForceOwnerResetOnMove

Description	Value
Description:	Resets and unlocks the owner of a ticket if it was moved to another queue.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'120-ForceOwnerResetOnMove'} = { </pre>

Description	Value
	<pre>'Event' => 'TicketQueueUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceOwnerReset' };</pre>

Ticket::EventModulePost###130-ForceStateChangeOnLock

Description	Value
Description:	Forces to choose a different ticket state (from current) after lock action. Define the current state as key, and the next state after lock action as content.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'130-ForceStateChangeOnLock'} = { 'Event' => 'TicketLockUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceState', 'new' => 'open' };</pre>

Ticket::EventModulePost###140-ResponsibleAutoSet

Description	Value
Description:	Automatically sets the responsible of a ticket (if it is not set yet) after the first owner update.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'140-ResponsibleAutoSet'} = { 'Event' => 'TicketOwnerUpdate', 'Module' => 'Kernel::System::Ticket::Event::ResponsibleAutoSet' };</pre>

Ticket::EventModulePost###150-TicketPendingTimeReset

Description	Value
Description:	Sets the PendingTime of a ticket to 0 if the state is changed to a non-pending state.

Description	Value
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'150- TicketPendingTimeReset'} = { 'Event' => 'TicketStateUpdate', 'Module' => 'Kernel::System::Ticket::Event::TicketPendingTimeReset' }; </pre>

Ticket::EventModulePost###500-NotificationEvent

Description	Value
Description:	Sends the notifications which are configured in the admin interface under "Notfication (Event)".
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'500- NotificationEvent'} = { 'Event' => '', 'Module' => 'Kernel::System::Ticket::Event::NotificationEvent', 'Transaction' => '1' }; </pre>

Ticket::EventModulePost###900-EscalationIndex

Description	Value
Description:	Updates the ticket escalation index after a ticket attribute got updated.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'900- EscalationIndex'} = { 'Event' => 'TicketSLAUpdate TicketQueueUpdate TicketStateUpdate TicketCreate ArticleCreate', 'Module' => 'Kernel::System::Ticket::Event::TicketEscalationIndex' }; </pre>

Description	Value
	};

Ticket::EventModulePost###900-EscalationStopEvents

Description	Value
Description:	Ticket event module that triggers the escalation stop events.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'900- EscalationStopEvents'} = { 'Event' => 'TicketSLAUpdate TicketQueueUpdate TicketStateUpdate ArticleCreate', 'Module' => 'Kernel::System::Ticket::Event::TriggerEscalationStopEvents' }; </pre>

Ticket::EventModulePost###910-ForceUnlockOnMove

Description	Value
Description:	Forces to unlock tickets after being moved to another queue.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::EventModulePost'}->{'910- ForceUnlockOnMove'} = { 'Event' => 'TicketQueueUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceUnlock' }; </pre>

Ticket::EventModulePost###920-TicketArticleNewMessageUpdate

Description	Value
Description:	Update Ticket "Seen" flag if every article got seen or a new Article got created.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Ticket::EventModulePost'}->{'920- TicketArticleNewMessageUpdate'} = { 'Event' => 'ArticleCreate ArticleFlagSet', 'Module' => 'Kernel::System::Ticket::Event::TicketNewMessageUpdate' }; </pre>

Ticket::CustomModule###001-CustomModule

Description	Value
Description:	Overloads (redefines) existing functions in Kernel::System::Ticket. Used to easily add customizations.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::CustomModule'}->{'001- CustomModule'} = 'Kernel::System::Ticket::Custom'; </pre>

Ticket::ViewableSenderTypes

Description	Value
Description:	Defines the default viewable sender types of a ticket (default: customer).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::ViewableSenderTypes'} = ['\customer\']; </pre>

Ticket::ViewableLocks

Description	Value
Description:	Defines the viewable locks of a ticket. Default: unlock, tmp_lock.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::ViewableLocks'} = [</pre>

Description	Value
	<pre>'\unlock\'', '\tmp_lock\'];</pre>

Ticket::ViewableStateType

Description	Value
Description:	Defines the valid state types for a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::ViewableStateType'} = ['new', 'open', 'pending reminder', 'pending auto'];</pre>

Ticket::UnlockStateType

Description	Value
Description:	Defines the valid states for unlocked tickets. To unlock tickets the script "bin/otrs.UnlockTickets.pl" can be used.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::UnlockStateType'} = ['new', 'open'];</pre>

Ticket::PendingNotificationOnlyToOwner

Description	Value
Description:	Sends reminder notifications of unlocked ticket after reaching the reminder date (only sent to ticket owner).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::PendingNotificationOnlyToOwner'} = '0';</code>

Ticket::PendingNotificationNotToResponsible

Description	Value
Description:	Disables sending reminder notifications to the responsible agent of a ticket (Ticket::Responsible needs to be activated).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingNotificationNotToResponsible'} = '0';</code>

Ticket::PendingReminderStateType

Description	Value
Description:	Defines the state type of the reminder for pending tickets.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingReminderStateType'} = ['pending reminder'];</code>

Ticket::PendingAutoStateType

Description	Value
Description:	Determines the possible states for pending tickets that changed state after reaching time limit.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingAutoStateType'} = ['pending auto'];</code>

Description	Value
];

Ticket::StateAfterPending

Description	Value
Description:	Defines which states should be set automatically (Content), after the pending time of state (Key) has been reached.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::StateAfterPending'} = { 'pending auto close+' => 'closed successful', 'pending auto close-' => 'closed unsuccessful' };</pre>

System::Permission

Description	Value
Description:	Standard available permissions for agents within the application. If more permissions are needed, they can be entered here. Permissions must be defined to be effective. Some other good permissions have also been provided built-in: note, close, pending, customer, freetext, move, compose, responsible, forward, and bounce. Make sure that "rw" is always the last registered permission.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'System::Permission'} = ['ro', 'move_into', 'create', 'note', 'owner', 'priority', 'rw'];</pre>

Ticket::Permission###1-OwnerCheck

Description	Value
Description:	Module to check the owner of a ticket.
Group:	Ticket

Description	Value
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Permission'}->{'1-OwnerCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::OwnerCheck', 'Required' => '0' }; </pre>

Ticket::Permission###2-ResponsibleCheck

Description	Value
Description:	Module to check the agent responsible of a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Permission'}->{'2-ResponsibleCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::ResponsibleCheck', 'Required' => '0' }; </pre>

Ticket::Permission###3-GroupCheck

Description	Value
Description:	Module to check if a user is in a special group. Access is granted, if the user is in the specified group and has ro and rw permissions.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Permission'}->{'3-GroupCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::GroupCheck', 'Required' => '0' }; </pre>

Description	Value
	};

Ticket::Permission###4-WatcherCheck

Description	Value
Description:	Module to check the watcher agents of a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'4-WatcherCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::WatcherCheck', 'Required' => '0' };</pre>

CustomerTicket::Permission###1-GroupCheck

Description	Value
Description:	Module to check the group permissions for the access to customer tickets.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'1- GroupCheck'} = { 'Granted' => '0', 'Module' => 'Kernel::System::Ticket::CustomerPermission::GroupCheck', 'Required' => '1' };</pre>

CustomerTicket::Permission###2-CustomerUserIDCheck

Description	Value
Description:	Grants access, if the customer ID of the ticket matches the customer user's ID and the customer user has group permissions on the queue the ticket is in.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'2-CustomerUserIDCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerUserIDCheck', 'Required' => '0' };</pre>

CustomerTicket::Permission###3-CustomerIDCheck

Description	Value
Description:	Module to check customer permissions.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'3-CustomerIDCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerIDCheck', 'Required' => '0' };</pre>

Ticket::DefineEmailFrom

Description	Value
Description:	Defines how the From field from the emails (sent from answers and email tickets) should look like.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::DefineEmailFrom'} = 'SystemAddressName';</pre>

Ticket::DefineEmailFromSeparator

Description	Value
Description:	Defines the separator between the agents real name and the given queue email address.

Description	Value
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::DefineEmailFromSeparator'} = 'via';</code>

CustomerNotifyJustToRealCustomer

Description	Value
Description:	Sends customer notifications just to the mapped customer. Normally, if no customer is mapped, the latest customer sender gets the notification.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerNotifyJustToRealCustomer'} = '0';</code>

AgentSelfNotifyOnAction

Description	Value
Description:	Specifies if an agent should receive email notification of his own actions.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AgentSelfNotifyOnAction'} = '0';</code>

Core::TicketACL

Ticket::Acl::Module###1-Ticket::Acl::Module

Description	Value
Description:	ACL module that allows closing parent tickets only if all its children are already closed ("State" shows which states are not available for the parent ticket until all child tickets are closed).
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	0

Description	Value
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Acl::Module'}->{'1-Ticket::Acl::Module'} = { 'Module' => 'Kernel::System::Ticket::Acl::CloseParentAfterClosedChilds', 'State' => ['closed successful', 'closed unsuccessful'] }; </pre>

TicketACL::Default::Action

Description	Value
Description:	Default ACL values for ticket actions.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TicketACL::Default::Action'} = {}; </pre>

Core::TicketBulkAction

Ticket::Frontend::BulkFeature

Description	Value
Description:	Enables ticket bulk action feature for the agent frontend to work on more than one ticket at a time.
Group:	Ticket
SubGroup:	Core::TicketBulkAction
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::BulkFeature'} = '1'; </pre>

Ticket::Frontend::BulkFeatureGroup

Description	Value
Description:	Enables ticket bulk action feature only for the listed groups.
Group:	Ticket
SubGroup:	Core::TicketBulkAction
Valid:	0

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::BulkFeatureGroup'} = ['admin', 'users'];</pre>

Core::TicketDynamicFieldDefault

Ticket::EventModulePost###TicketDynamicFieldDefault

Description	Value
Description:	Event module registration. For more performance you can define a trigger event (e. g. Event => TicketCreate). This is only possible if all Ticket dynamic fields need the same event.
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}- >{'TicketDynamicFieldDefault'} = { 'Module' => 'Kernel::System::Ticket::Event::TicketDynamicFieldDefault', 'Transaction' => '1' };</pre>

Ticket::TicketDynamicFieldDefault###Element1

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element1'} = { 'Event' => 'TicketCreate', 'Name' => 'Field1', 'Value' => 'Default' };</pre>

Ticket::TicketDynamicFieldDefault###Element2

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otros.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element2'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element3

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otros.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element3'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element4

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otros.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault

Description	Value
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element4'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element5

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element5'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element6

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element6'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Description	Value
	};

Ticket::TicketDynamicFieldDefault###Element7

Description	Value
Description:	Configures a default TicketDynsicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element7'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element8

Description	Value
Description:	Configures a default TicketDynsicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element8'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element9

Description	Value
Description:	Configures a default TicketDynsicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".

Description	Value
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element9'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element10

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otros.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element10'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element11

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otros.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element11'} = { 'Event' => '', </pre>

Description	Value
	<pre>'Name' => '', 'Value' => '' };</pre>

Ticket::TicketDynamicFieldDefault###Element12

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otsr.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element12'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

Ticket::TicketDynamicFieldDefault###Element13

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otsr.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element13'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

Ticket::TicketDynamicFieldDefault###Element14

Description	Value
Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and

Description	Value
	"Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element14'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element15

Description	Value
Description:	Configures a default TicketDynmicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element15'} = { 'Event' => '', 'Name' => '', 'Value' => '' }; </pre>

Ticket::TicketDynamicFieldDefault###Element16

Description	Value
Description:	Configures a default TicketDynmicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}- >{'Element16'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

Core::TicketWatcher

Ticket::Watcher

Description	Value
Description:	Enables or disables the ticket watcher feature, to keep track of tickets without being the owner nor the responsible.
Group:	Ticket
SubGroup:	Core::TicketWatcher
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Watcher'} = '0';</pre>

Ticket::WatcherGroup

Description	Value
Description:	Enables ticket watcher feature only for the listed groups.
Group:	Ticket
SubGroup:	Core::TicketWatcher
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::WatcherGroup'} = ['admin', 'users'];</pre>

Frontend::Admin::ModuleRegistration

Frontend::Module###AdminQueue

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminQueue'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage queues.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Queues', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Queues' }; </pre>

Frontend::Module###AdminResponse

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage response templates.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Responses', 'Prio' => '200' }, 'NavBarName' => 'Admin', 'Title' => 'Responses' }; </pre>

Frontend::Module###AdminQueueResponses

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminQueueResponses'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link responses to queues.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Responses <-> Queues', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'Responses <-> Queues' }; </pre>

Frontend::Module###AdminAutoResponse

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminAutoResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage responses that are automatically sent.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Auto Responses', </pre>

Description	Value
	<pre>'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Auto Responses' };</pre>

Frontend::Module###AdminQueueAutoResponse

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AdminQueueAutoResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link queues to auto responses.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Auto Responses <-> Queues', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Auto Responses <-> Queues' };</pre>

Frontend::Module###AdminAttachment

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminAttachment'} = { 'Description' => 'Admin', 'Group' => [</pre>

Description	Value
	<pre> 'admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage attachments.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Attachments', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'Attachments' }; </pre>

Frontend::Module###AdminResponseAttachment

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}- >{'AdminResponseAttachment'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link attachments to responses templates.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Attachments <-> Responses', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'Attachments <-> Responses' }; </pre>

Frontend::Module###AdminSalutation

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSalutation'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage salutations.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Salutations', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'Salutations' }; </pre>

Frontend::Module###AdminSignature

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSignature'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage signatures.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Signatures', 'Prio' => '900' }, 'NavBarName' => 'Admin', </pre>

Description	Value
	'Title' => 'Signatures' };

Frontend::Module###AdminSystemAddress

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminSystemAddress'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Set sender email addresses for this system.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Email Addresses', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'Email Addresses' }; </pre>

Frontend::Module###AdminNotification

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminNotification'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { </pre>

Description	Value
	<pre> 'Block' => 'Ticket', 'Description' => 'Create and manage notifications that are sent to agents.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agent Notifications', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Agent Notifications' }; </pre>

Frontend::Module###AdminNotificationEvent

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}- >{'AdminNotificationEvent'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage event based notifications.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Notifications (Event)', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Notifications (Event)' }; </pre>

Frontend::Module###AdminService

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminService'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Services', 'Prio' => '900' }, 'NavBarName' => 'Admin', 'Title' => 'Services' }; </pre>

Frontend::Module###AdminSLA

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSLA'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage Service Level Agreements (SLAs).', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Service Level Agreements', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'Service Level Agreements' }; </pre>

Frontend::Module###AdminType

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminType'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage ticket types.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Types', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'Types' }; </pre>

Frontend::Module###AdminState

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminState'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage ticket states.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'States', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'States' }; </pre>

Description	Value
	};

Frontend::Module###AdminPriority

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPriority'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage ticket priorities.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Priorities', 'Prio' => '850' }, 'NavBarName' => 'Admin', 'Title' => 'Priorities' }; </pre>

Frontend::Module###AdminGenericAgent

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminGenericAgent'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', </pre>

Description	Value
	<pre>'Description' => 'Manage periodic tasks.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'GenericAgent', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'GenericAgent' };</pre>

Frontend::Agent

Ticket::Frontend::PendingDiffTime

Description	Value
Description:	Time in seconds that gets added to the actual time if setting a pending-state (default: 86400 = 1 day).
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::PendingDiffTime'} = '86400';</code>

Ticket::Frontend::ListType

Description	Value
Description:	Shows existing parent/child queue lists in the system in the form of a tree or a list.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ListType'} = 'tree';</code>

Ticket::Frontend::TextAreaEmail

Description	Value
Description:	Permitted width for compose email windows.
Group:	Ticket
SubGroup:	Frontend::Agent

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TextAreaEmail'} = '82';</code>

Ticket::Frontend::TextAreaNote

Description	Value
Description:	Permitted width for compose note windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TextAreaNote'} = '78';</code>

Ticket::Frontend::CustomerInfoCompose

Description	Value
Description:	Shows the customer user information (phone and email) in the compose screen.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoCompose'} = '1';</code>

Ticket::Frontend::CustomerInfoComposeMaxSize

Description	Value
Description:	Max size (in characters) of the customer information table (phone and email) in the compose screen.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoComposeMaxSize'} = '22';</code>

Ticket::Frontend::CustomerInfoZoom

Description	Value
Description:	Shows the customer user's info in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoZoom'} = '1';</code>

Ticket::Frontend::CustomerInfoZoomMaxSize

Description	Value
Description:	Maximum size (in characters) of the customer information table in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoZoomMaxSize'} = '22';</code>

Ticket::Frontend::CustomerInfoQueueMaxSize

Description	Value
Description:	Maximum size (in characters) of the customer info table in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoQueueMaxSize'} = '18';</code>

Ticket::Frontend::AccountTime

Description	Value
Description:	Activates time accounting.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AccountTime'} = '1';</code>

Ticket::Frontend::TimeUnits

Description	Value
Description:	Sets the preferred time units (e.g. work units, hours, minutes).
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TimeUnits'} = '(work units)';</code>

Ticket::Frontend::NeedAccountedTime

Description	Value
Description:	Defines if time accounting is mandatory in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NeedAccountedTime'} = '0';</code>

Ticket::Frontend::BulkAccountedTime

Description	Value
Description:	Defines if time accounting must be set to all tickets in bulk action.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::BulkAccountedTime'} = '1';</code>

Ticket::Frontend::NeedSpellCheck

Description	Value
Description:	Defines if composed messages have to be spell checked in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NeedSpellCheck'} = '0';</pre>

Ticket::Frontend::NewOwnerSelection

Description	Value
Description:	Shows an owner selection in phone and email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NewOwnerSelection'} = '1';</pre>

Ticket::Frontend::NewResponsibleSelection

Description	Value
Description:	Show a responsible selection in phone and email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NewResponsibleSelection'} = '1';</pre>

Ticket::Frontend::NewQueueSelectionType

Description	Value
Description:	Defines the receipt target of the phone ticket and the sender of the email ticket ("Queue" shows all queues, "SystemAddress" displays all system addresses) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NewQueueSelectionType'} = 'Queue';</pre>

Ticket::Frontend::NewQueueSelectionString

Description	Value
Description:	Determines the strings that will be shown as receipt (To:) of the phone ticket and as sender (From:) of the email ticket in the agent interface. For Queue as NewQueueSelectionType "<Queue>" shows the names of the queues and for SystemAddress "<Realname> <<Email>>" shows the name and email of the receipt.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NewQueueSelectionString'} = '<Queue>';</pre>

Ticket::Frontend::NewQueueOwnSelection

Description	Value
Description:	Determines which options will be valid of the receipt (phone ticket) and the sender (email ticket) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NewQueueOwnSelection'} = { '1' => 'First Queue!', '2' => 'Second Queue!' };</pre>

Ticket::Frontend::ShowCustomerTickets

Description	Value
Description:	Shows customer history tickets in AgentTicketPhone, AgentTicketEmail and AgentTicketCustomer.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ShowCustomerTickets'} = '1';</code>

NewTicketInNewWindow::Enabled

Description	Value
Description:	If enabled, TicketPhone and TicketEmail will be open in new windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NewTicketInNewWindow::Enabled'} = '0';</code>

CustomerDBLink

Description	Value
Description:	Defines an external link to the database of the customer (e.g. 'http://yourhost/customer.php?CID=\$Data{"CustomerID"}' or ").
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerDBLink'} = '\$Env{"CGIHandle"}?Action=AgentTicketCustomer;TicketID=\$Data{"TicketID"}';</code>

CustomerDBLinkTarget

Description	Value
Description:	Defines the target attribute in the link to external customer database. E.g. 'target="cdb"'.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerDBLinkTarget'} = '';</code>

Frontend::CommonObject###QueueObject

Description	Value
Description:	Path of the file that stores all the settings for the QueueObject object for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonObject'}->{'QueueObject'} = 'Kernel::System::Queue';</code>

Frontend::CommonObject###TicketObject

Description	Value
Description:	Path of the file that stores all the settings for the TicketObject for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonObject'}->{'TicketObject'} = 'Kernel::System::Ticket';</code>

Frontend::CommonParam###Action

Description	Value
Description:	Defines the default used Frontend-Module if no Action parameter given in the url on the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'Action'} = 'AgentDashboard';</code>

Frontend::CommonParam###QueueID

Description	Value
Description:	Default queue ID used by the system in the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'QueueID'} = '0';</code>

Frontend::CommonParam###TicketID

Description	Value
Description:	Default ticket ID used by the system in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'TicketID'} = '';</code>

Frontend::Agent::CustomerSearch

Ticket::Frontend::CustomerSearchAutoComplete###Active

Description	Value
Description:	Enables or disables the autocomplete feature for the customer search in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerSearchAutoComplete'}->{'Active'} = '1';</code>

Ticket::Frontend::CustomerSearchAutoComplete###MinQueryLength

Description	Value
Description:	Sets the minimum number of characters before autocomplete query is sent.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerSearchAutoComplete'}- >{'MinQueryLength'} = '2';</pre>

Ticket::Frontend::CustomerSearchAutoComplete###QueryDelay

Description	Value
Description:	Delay time between autocomplete queries in milliseconds.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerSearchAutoComplete'}- >{'QueryDelay'} = '100';</pre>

Ticket::Frontend::CustomerSearchAutoComplete###MaxResultsDisplayed

Description	Value
Description:	Sets the number of search results to be displayed for the autocomplete feature.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerSearchAutoComplete'}- >{'MaxResultsDisplayed'} = '20';</pre>

Ticket::Frontend::CustomerSearchAutoComplete::DynamicWidth

Description	Value
Description:	Determines if the search results container for the autocomplete feature should adjust its width dynamically.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerSearchAutoComplete::DynamicWidth'} = '1';</pre>

Frontend::Agent::Dashboard

DashboardBackend###0100-TicketPendingReminder

Description	Value
Description:	Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0100- TicketPendingReminder'} = { 'Attributes' => 'TicketPendingTimeOlderMinutes=1;StateType=pending reminder;SortBy=PendingTime;OrderBy=Down;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All tickets with a reminder set where the reminder date has been reached', 'Filter' => 'Locked', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'UntilTime', 'Title' => 'Reminder Tickets' }; </pre>

DashboardBackend###0110-TicketEscalation

Description	Value
Description:	Parameters for the dashboard backend of the ticket escalation overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0110-TicketEscalation'} = { 'Attributes' => 'TicketEscalationTimeOlderMinutes=1;SortBy=EscalationTime;Order 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All escalated tickets', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'EscalationTime', 'Title' => 'Escalated Tickets' }; </pre>

DashboardBackend###0120-TicketNew

Description	Value
Description:	Parameters for the dashboard backend of the new tickets overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2);. "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0120-TicketNew'} = { 'Attributes' => 'StateType=new;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All new tickets, these tickets have not been worked on yet', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'Age', 'Title' => 'New Tickets' } </pre>

Description	Value
	};

DashboardBackend###0130-TicketOpen

Description	Value
Description:	Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0130-TicketOpen'} = { 'Attributes' => 'StateType=open;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All open tickets, these tickets have already been worked on, but need a response', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'Age', 'Title' => 'Open Tickets / Need to be answered' }; </pre>

DashboardBackend###0250-TicketStats

Description	Value
Description:	Parameters for the dashboard backend of the ticket stats of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self->{'DashboardBackend'}->{'0250-TicketStats'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '30', 'Closed' => '1', 'Created' => '1', 'Default' => '1', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardTicketStatsGeneric', 'Permission' => 'rw', 'Title' => '7 Day Stats' };</pre>

DashboardBackend###0260-TicketCalendar

Description	Value
Description:	Parameters for the dashboard backend of the ticket calendar of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0260- TicketCalendar'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '2', 'Default' => '1', 'Group' => '', 'Limit' => '6', 'Module' => 'Kernel::Output::HTML::DashboardCalendar', 'OwnerOnly' => '', 'Permission' => 'rw', 'Title' => 'Upcoming Events' };</pre>

Frontend::Agent::ModuleMetaHead

Frontend::HeaderMetaModule###2-TicketSearch

Description	Value
Description:	Module to generate html OpenSearch profile for short ticket search in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleMetaHead
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::HeaderMetaModule'}->{'2-TicketSearch'} = { 'Action' => 'AgentTicketSearch', 'Module' => 'Kernel::Output::HTML::HeaderMetaTicketSearch' };</pre>

Frontend::Agent::ModuleNotify

Frontend::NotifyModule###5-Ticket::TicketEscalation

Description	Value
Description:	Module to show notifications and escalations (ShownMax: max. shown escalations, EscalationInMinutes: Show ticket which will escalation in, CacheTime: Cache of calculated escalations in seconds).
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'5-Ticket::TicketEscalation'} = { 'CacheTime' => '40', 'EscalationInMinutes' => '120', 'Module' => 'Kernel::Output::HTML::NotificationAgentTicketEscalation', 'ShownMax' => '25' };</pre>

Frontend::Agent::ModuleRegistration

Frontend::Module###AgentTicketQueue

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketQueue'} = { 'Description' => 'Overview of all open Tickets', 'Loader' => { 'CSS' => ['Core.AgentTicketQueue.css'] }, 'NavBar' => [{ 'AccessKey' => 'o', 'Block' => '', 'Description' => 'Overview of all open Tickets', 'Link' => 'Action=AgentTicketQueue', 'LinkOption' => '', 'Name' => 'Queue view', 'NavBar' => 'Ticket', 'Prio' => '100', 'Type' => '' }, { 'AccessKey' => 't', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AgentTicketQueue', 'LinkOption' => '', 'Name' => 'Tickets', 'NavBar' => 'Ticket', 'Prio' => '200', 'Type' => 'Menu' }], 'NavBarName' => 'Ticket', 'Title' => 'QueueView' }; </pre>

Frontend::Module###AgentTicketPhone

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketPhone'} = { 'Description' => 'Create new phone picket', 'Loader' => { </pre>

Description	Value
	<pre> 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => 'n', 'Block' => '', 'Description' => 'Create new phone ticket (inbound)', 'Link' => 'Action=AgentTicketPhone', 'LinkOption' => '', 'Name' => 'New phone ticket', 'NavBar' => 'Ticket', 'Prio' => '200', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'New phone ticket' }; </pre>

Frontend::Module###AgentTicketPhoneOutbound

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}- >{'AgentTicketPhoneOutbound'} = { 'Description' => 'Phone Call', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Phone-Ticket' }; </pre>

Frontend::Module###AgentTicketPhoneInbound

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketPhoneInbound'} = { 'Description' => 'Incoming Phone Call', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Phone-Ticket' }; </pre>

Frontend::Module###AgentTicketEmail

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketEmail'} = { 'Description' => 'Create new email ticket', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => 'e', 'Block' => '', 'Description' => 'Create new email ticket and send this out (outbound)', 'Link' => 'Action=AgentTicketEmail', 'LinkOption' => '', 'Name' => 'New email ticket', 'NavBar' => 'Ticket', 'Prio' => '210', 'Type' => '' }] }; </pre>

Description	Value
	<pre>], 'NavBarName' => 'Ticket', 'Title' => 'New email ticket' };</pre>

Frontend::Module###AgentTicketSearch

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketSearch'} = { 'Description' => 'Search Ticket', 'NavBar' => [{ 'AccessKey' => 's', 'Block' => '', 'Description' => 'Search Tickets', 'Link' => 'Action=AgentTicketSearch', 'LinkOption' => 'onclick="window.setTimeout(function() {Core.Agent.Search.OpenSearchDialog(\'AgentTicketSearch \');}, 0); return false;"', 'Name' => 'Search', 'NavBar' => 'Ticket', 'Prio' => '300', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Search' };</pre>

Frontend::Module###AgentTicketMailbox

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self->{'Frontend::Module'}- >{'AgentTicketMailbox'} = { 'Description' => 'compat module for AgentTicketMailbox to AgentTicketLockedView', 'NavBarName' => 'Ticket', 'Title' => '' };</pre>

Frontend::Module###AgentTicketLockedView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AgentTicketLockedView'} = { 'Description' => 'Locked Tickets', 'NavBarName' => 'Ticket', 'Title' => 'Locked Tickets' };</pre>

Frontend::Module###AgentTicketResponsibleView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AgentTicketResponsibleView'} = { 'Description' => 'Responsible Tickets', 'NavBarName' => 'Ticket', 'Title' => 'Responsible Tickets' };</pre>

Frontend::Module###AgentTicketWatchView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketWatchView'} = { 'Description' => 'Watched Tickets', 'NavBarName' => 'Ticket', 'Title' => 'Watched Tickets' }; </pre>

Frontend::Module###AgentCustomerSearch

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentCustomerSearch'} = { 'Description' => 'AgentCustomerSearch', 'NavBarName' => 'Ticket', 'Title' => 'AgentCustomerSearch' }; </pre>

Frontend::Module###AgentTicketStatusView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketStatusView'} = { 'Description' => 'Overview of all open tickets', 'NavBar' => [{ 'AccessKey' => 'v', 'Block' => '', 'Description' => 'Overview of all open Tickets.', 'Link' => 'Action=AgentTicketStatusView', </pre>

Description	Value
	<pre> 'LinkOption' => '', 'Name' => 'Status view', 'NavBar' => 'Ticket', 'Prio' => '110', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Status view' }; </pre>

Frontend::Module###AgentTicketEscalationView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketEscalationView'} = { 'Description' => 'Overview of all escalated tickets', 'NavBar' => [{ 'AccessKey' => 'e', 'Block' => '', 'Description' => 'Overview Escalated Tickets', 'Link' => 'Action=AgentTicketEscalationView', 'LinkOption' => '', 'Name' => 'Escalation view', 'NavBar' => 'Ticket', 'Prio' => '120', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Escalation view' }; </pre>

Frontend::Module###AgentZoom

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentZoom'} = { 'Description' => 'compat module for AgentZoom to AgentTicketZoom', 'NavBarName' => 'Ticket', 'Title' => '' }; </pre>

Frontend::Module###AgentTicketZoom

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketZoom'} = { 'Description' => 'Ticket Zoom', 'Loader' => { 'JavaScript' => ['thirdparty/jquery-tablesorter-2.0.5/ jquery.tablesorter.js', 'Core.UI.Table.Sort.js', 'Core.Agent.TicketZoom.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Zoom' }; </pre>

Frontend::Module###AgentTicketAttachment

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketAttachment'} = { 'Description' => 'To download attachments', </pre>

Description	Value
	<pre>'NavBarName' => 'Ticket', 'Title' => '' };</pre>

Frontend::Module###AgentTicketPlain

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPlain'} = { 'Description' => 'Ticket plain view of an email', 'NavBarName' => 'Ticket', 'Title' => 'Plain' };</pre>

Frontend::Module###AgentTicketNote

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketNote'} = { 'Description' => 'Ticket Note', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Note' };</pre>

Frontend::Module###AgentTicketMerge

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketMerge'} = { 'Description' => 'Ticket Merge', 'NavBarName' => 'Ticket', 'Title' => 'Merge' };</pre>

Frontend::Module###AgentTicketPending

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AgentTicketPending'} = { 'Description' => 'Ticket Pending', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Pending' };</pre>

Frontend::Module###AgentTicketWatcher

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AgentTicketWatcher'} = { 'Description' => 'A TicketWatcher Module',</pre>

Description	Value
	<pre>'NavBarName' => 'Ticket-Watcher', 'Title' => 'Ticket-Watcher' };</pre>

Frontend::Module###AgentTicketPriority

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AgentTicketPriority'} = { 'Description' => 'Ticket Priority', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Priority' };</pre>

Frontend::Module###AgentTicketLock

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketLock'} = { 'Description' => 'Ticket Lock', 'NavBarName' => 'Ticket', 'Title' => 'Lock' };</pre>

Frontend::Module###AgentTicketMove

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketMove'} = { 'Description' => 'Ticket Move', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Move' }; </pre>

Frontend::Module###AgentTicketHistory

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}-> {'AgentTicketHistory'} = { 'Description' => 'Ticket History', 'NavBarName' => 'Ticket', 'Title' => 'History' }; </pre>

Frontend::Module###AgentTicketOwner

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketOwner'} = { 'Description' => 'Ticket Owner', 'Loader' => { 'JavaScript' => [</pre>

Description	Value
	<pre> 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Owner' }; </pre>

Frontend::Module###AgentTicketResponsible

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketResponsible'} = { 'Description' => 'Ticket Responsible', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Responsible' }; </pre>

Frontend::Module###AgentTicketCompose

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketCompose'} = { 'Description' => 'Ticket Compose email Answer', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', }; </pre>

Description	Value
	'Title' => 'Compose' };

Frontend::Module###AgentTicketBounce

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketBounce'} = { 'Description' => 'Ticket Compose Bounce Email', 'NavBarName' => 'Ticket', 'Title' => 'Bounce' }; </pre>

Frontend::Module###AgentTicketForward

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}-> {'AgentTicketForward'} = { 'Description' => 'Ticket Forward Email', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Forward' }; </pre>

Frontend::Module###AgentTicketCustomer

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketCustomer'} = { 'Description' => 'Ticket Customer', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Customer' }; </pre>

Frontend::Module###AgentTicketClose

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketClose'} = { 'Description' => 'Ticket Close', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Close' }; </pre>

Frontend::Module###AgentTicketFreeText

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AgentTicketFreeText'} = { 'Description' => 'Ticket FreeText', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Free Fields' }; </pre>

Frontend::Module###AgentTicketPrint

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketPrint'} = { 'Description' => 'Ticket Print', 'NavBarName' => 'Ticket', 'Title' => 'Print' }; </pre>

Frontend::Module###AgentTicketBulk

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketBulk'} = { 'Description' => 'Ticket bulk module', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] } }; </pre>

Description	Value
	<pre>] }, 'NavBarName' => 'Ticket', 'Title' => 'Bulk-Action' }; </pre>

Frontend::Agent::Preferences

PreferencesGroups###NewTicketNotify

Description	Value
Description:	Parameters for the NewTicketNotify object in the preferences view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'NewTicketNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if there is a new ticket in "My Queues".', 'Key' => 'Send new ticket notifications', 'Label' => 'New ticket notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendNewTicketNotification', 'Prio' => '1000' }; </pre>

PreferencesGroups###FollowUpNotify

Description	Value
Description:	Parameters for the FollowUpNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'FollowUpNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a customer sends a follow up and I\'m the owner of the ticket or the ticket is unlocked and is in one of my subscribed queues.', 'Key' => 'Send ticket follow up notifications', 'Label' => 'Ticket follow up notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendFollowUpNotification', 'Prio' => '2000' }; </pre>

PreferencesGroups###LockTimeoutNotify

Description	Value
Description:	Parameters for the LockTimeoutNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}- >{'LockTimeoutNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a ticket is unlocked by the system.', 'Key' => 'Send ticket lock timeout notifications', 'Label' => 'Ticket lock timeout notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendLockTimeoutNotification', 'Prio' => '3000' }; </pre>

Description	Value
	};

PreferencesGroups###MoveNotify

Description	Value
Description:	Parameters for the MoveNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'MoveNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a ticket is moved into one of "My Queues".', 'Key' => 'Send ticket move notifications', 'Label' => 'Ticket move notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendMoveNotification', 'Prio' => '4000' }; </pre>

PreferencesGroups###WatcherNotify

Description	Value
Description:	Parameters for the WatcherNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'WatcherNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' } }; </pre>

Description	Value
	<pre> }, 'DataSelected' => '0', 'Desc' => 'Send me the same notifications for my watched tickets that the ticket owners will get.', 'Key' => 'Send ticket watch notifications', 'Label' => 'Ticket watch notification', 'Module' => 'Kernel::Output::HTML::PreferencesTicketWatcher', 'PrefKey' => 'UserSendWatcherNotification', 'Prio' => '5000' }; </pre>

PreferencesGroups###CustomQueue

Description	Value
Description:	Parameters for the CustomQueue object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'PreferencesGroups'}->{'CustomQueue'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Desc' => 'Your queue selection of your favorite queues. You also get notified about those queues via email if enabled.', 'Key' => 'My Queues', 'Label' => 'My Queues', 'Module' => 'Kernel::Output::HTML::PreferencesCustomQueue', 'Permission' => 'ro', 'Prio' => '1000' }; </pre>

PreferencesGroups###RefreshTime

Description	Value
Description:	Parameters for the RefreshTime object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'PreferencesGroups'}->{'RefreshTime'} = { </pre>

Description	Value
	<pre>'Active' => '1', 'Column' => 'Other Settings', 'Data' => { '0' => 'off', '10' => '10 minutes', '15' => '15 minutes', '2' => ' 2 minutes', '5' => ' 5 minutes', '7' => ' 7 minutes' }, 'DataSelected' => '0', 'Desc' => 'If enabled, the different overviews (Dashboard, LockedView, QueueView) will automatically refresh after the specified time.', 'Key' => 'Refresh Overviews after', 'Label' => 'Overview Refresh Time', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserRefreshTime', 'Prio' => '2000' };</pre>

PreferencesGroups###TicketOverviewSmallPageShown

Description	Value
Description:	Parameters for the pages (in which the tickets are shown) of the small ticket overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}- >{'TicketOverviewSmallPageShown'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Data' => { '10' => '10', '15' => '15', '20' => '20', '25' => '25', '30' => '30', '35' => '35' }, 'DataSelected' => '25', 'Key' => 'Ticket limit per page for Ticket Overview "Small"', 'Label' => 'Ticket Overview "Small" Limit', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric',</pre>

Description	Value
	<pre>'PrefKey' => 'UserTicketOverviewSmallPageShown', 'Prio' => '8000' };</pre>

PreferencesGroups###TicketOverviewMediumPageShown

Description	Value
Description:	Parameters for the pages (in which the tickets are shown) of the medium ticket overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}- >{'TicketOverviewMediumPageShown'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Data' => { '10' => '10', '15' => '15', '20' => '20', '25' => '25', '30' => '30', '35' => '35' }, 'DataSelected' => '20', 'Key' => 'Ticket limit per page for Ticket Overview "Medium"', 'Label' => 'Ticket Overview "Medium" Limit', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewMediumPageShown', 'Prio' => '8100' };</pre>

PreferencesGroups###TicketOverviewPreviewPageShown

Description	Value
Description:	Parameters for the pages (in which the tickets are shown) of the ticket preview overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'PreferencesGroups'}->{'TicketOverviewPreviewPageShown'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Data' => { '10' => '10', '15' => '15', '20' => '20', '25' => '25', '30' => '30', '35' => '35' }, 'DataSelected' => '15', 'Key' => 'Ticket limit per page for Ticket Overview "Preview"', 'Label' => 'Ticket Overview "Preview" Limit', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewPreviewPageShown', 'Prio' => '8200' }; </pre>

PreferencesGroups###CreateNextMask

Description	Value
Description:	Parameters for the CreateNextMask object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'CreateNextMask'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Data' => { '' => 'CreateTicket', 'AgentTicketZoom' => 'TicketZoom' }, 'DataSelected' => '', 'Key' => 'Show this screen after I created a new ticket', 'Label' => 'Screen after new ticket', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserCreateNextMask', 'Prio' => '3000' }; </pre>

Frontend::Agent::SearchRouter

Frontend::Search###Ticket

Description	Value
Description:	Search backend router.
Group:	Ticket
SubGroup:	Frontend::Agent::SearchRouter
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Search'}->{'Ticket'} = { '^AgentTicket' => 'Action=AgentTicketSearch;Subaction=AJAX' }; </pre>

Frontend::Agent::Ticket::ArticleAttachmentModule

Ticket::Frontend::ArticleAttachmentModule###1-Download

Description	Value
Description:	Shows a link to download article attachments in the zoom view of the article in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleAttachmentModule
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self-> >{'Ticket::Frontend::ArticleAttachmentModule'}-> >{'1-Download'} = { 'Module' => 'Kernel::Output::HTML::ArticleAttachmentDownload' }; </pre>

Ticket::Frontend::ArticleAttachmentModule###2-HTML-Viewer

Description	Value
Description:	Shows a link to access article attachments via a html online viewer in the zoom view of the article in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleAttachmentModule
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre>\$Self- >{'Ticket::Frontend::ArticleAttachmentModule'}- >{'2-HTML-Viewer'} = { 'Module' => 'Kernel::Output::HTML::ArticleAttachmentHTMLViewer' };</pre>

Frontend::Agent::Ticket::ArticleComposeModule

Ticket::Frontend::ArticleComposeModule###1-SignEmail

Description	Value
Description:	Module to compose signed messages (PGP or S/MIME).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleComposeModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleComposeModule'}- >{'1-SignEmail'} = { 'Module' => 'Kernel::Output::HTML::ArticleComposeSign' };</pre>

Ticket::Frontend::ArticleComposeModule###2-CryptEmail

Description	Value
Description:	Module to crypt composed messages (PGP or S/MIME).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleComposeModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleComposeModule'}- >{'2-CryptEmail'} = { 'Module' => 'Kernel::Output::HTML::ArticleComposeCrypt' };</pre>

Frontend::Agent::Ticket::ArticleViewModule

Ticket::Frontend::ArticleViewModule###1-PGP

Description	Value
Description:	Agent interface article notification module to check PGP.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModule
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-PGP'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckPGP' }; </pre>

Ticket::Frontend::ArticleViewModule###1-SMIME

Description	Value
Description:	Agent interface module to check incoming emails in the Ticket-Zoom-View if the S/MIME-key is available and true.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModule
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-SMIME'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckSMIME' }; </pre>

Frontend::Agent::Ticket::ArticleViewModulePre

Ticket::Frontend::ArticlePreViewModule###1-PGP

Description	Value
Description:	Agent interface article notification module to check PGP.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModulePre
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::ArticlePreViewModule'}->{'1-PGP'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckPGP' }; </pre>

Ticket::Frontend::ArticlePreviewModule###1-SMIME

Description	Value
Description:	Agent interface article notification module to check S/MIME.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModulePre
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticlePreviewModule'}->{'1-SMIME'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckSMIME' };</pre>

Frontend::Agent::Ticket::MenuModule**Ticket::Frontend::MenuModule###000-Back**

Description	Value
Description:	Shows a link in the menu to go back in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'000-Back'} = { 'Action' => '', 'Description' => 'Back', 'Link' => '\$Env{"LastScreenOverview"};TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Back', 'PopupType' => '', 'Target' => '' };</pre>

Ticket::Frontend::MenuModule###100-Lock

Description	Value
Description:	Shows a link in the menu to lock/unlock tickets in the ticket zoom view of the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'100-Lock'} = { 'Action' => 'AgentTicketLock', 'Module' => 'Kernel::Output::HTML::TicketMenuLock', 'Name' => 'Lock', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###200-History

Description	Value
Description:	Shows a link in the menu to access the history of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'200-History'} = { 'Action' => 'AgentTicketHistory', 'Description' => 'Shows the ticket history', 'Link' => 'Action=AgentTicketHistory;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'History', 'PopupType' => 'TicketHistory', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###210-Print

Description	Value
Description:	Shows a link in the menu to print a ticket or an article in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'210-Print'} = { 'Action' => 'AgentTicketPrint', 'Description' => 'Print this ticket', 'Link' => 'Action=AgentTicketPrint;TicketID=\$QData{"TicketID"}', 'LinkParam' => 'target="print"', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Print', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###300-Priority

Description	Value
Description:	Shows a link in the menu to see the priority of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'300-Priority'} = { 'Action' => 'AgentTicketPriority', 'Description' => 'Change the ticket priority', 'Link' => 'Action=AgentTicketPriority;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Priority', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###310-FreeText

Description	Value
Description:	Shows a link in the menu to add a free text field in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'310-FreeText'} = { 'Action' => 'AgentTicketFreeText', 'Description' => 'Change the free fields for this ticket', 'Link' => 'Action=AgentTicketFreeText;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Free Fields', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###320-Link

Description	Value
Description:	Shows a link in the menu that allows linking a ticket with another object in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'320-Link'} = { 'Action' => 'AgentLinkObject', 'Description' => 'Link this ticket to other objects', 'Link' => 'Action=AgentLinkObject;SourceObject=Ticket;SourceKey=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Link', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###400-Owner

Description	Value
Description:	Shows a link in the menu to see the owner of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'400-Owner'} = { 'Action' => 'AgentTicketOwner', 'Description' => 'Change the owner for this ticket', 'Link' => 'Action=AgentTicketOwner;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Owner', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###410-Responsible

Description	Value
Description:	Shows a link in the menu to see the responsible agent of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'410-Responsible'} = { 'Action' => 'AgentTicketResponsible', 'Description' => 'Change the responsible person for this ticket', 'Link' => 'Action=AgentTicketResponsible;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuResponsible', 'Name' => 'Responsible', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###420-Customer

Description	Value
Description:	Shows a link in the menu to see the customer who requested the ticket in the ticket zoom view of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'420- Customer'} = { 'Action' => 'AgentTicketCustomer', 'Description' => 'Change the customer for this ticket', 'Link' => 'Action=AgentTicketCustomer;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Customer', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###420-Note

Description	Value
Description:	Shows a link in the menu to add a note in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'420- Note'} = { 'Action' => 'AgentTicketNote', 'Description' => 'Add a note to this ticket', 'Link' => 'Action=AgentTicketNote;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Note', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###430-Merge

Description	Value
Description:	Shows a link in the menu that allows merging tickets in the ticket zoom view of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'430-Merge'} = { 'Action' => 'AgentTicketMerge', 'Description' => 'Merge into a different ticket', 'Link' => 'Action=AgentTicketMerge;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Merge', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###440-Pending

Description	Value
Description:	Shows a link in the menu to set a ticket as pending in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'440-Pending'} = { 'Action' => 'AgentTicketPending', 'Description' => 'Set this ticket to pending', 'Link' => 'Action=AgentTicketPending;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Pending', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###448-Watch

Description	Value
Description:	Shows a link in the menu for subscribing / unsubscribing from a ticket in the ticket zoom view of the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'448-Watch'} = { 'Action' => 'AgentTicketWatcher', 'Module' => 'Kernel::Output::HTML::TicketMenuTicketWatcher', 'Name' => 'Watch', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###450-Close

Description	Value
Description:	Shows a link in the menu to close a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'450-Close'} = { 'Action' => 'AgentTicketClose', 'Description' => 'Close this ticket', 'Link' => 'Action=AgentTicketClose;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Close', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###460-Delete

Description	Value
Description:	Shows a link in the menu to delete a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'460-Delete'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Delete this ticket', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Delete', 'PopupType' => '', 'Target' => '' }; </pre>

Ticket::Frontend::MenuModule###470-Spam

Description	Value
Description:	Shows a link to set a ticket as spam in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'470-Spam'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Mark as Spam!', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Spam', 'PopupType' => '', 'Target' => '' }; </pre>

Frontend::Agent::Ticket::MenuModulePre

Ticket::Frontend::PreMenuModule###100-Lock

Description	Value
Description:	Shows a link in the menu to lock / unlock a ticket in the ticket overviews of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'100-Lock'} = { 'Action' => 'AgentTicketLock', 'Module' => 'Kernel::Output::HTML::TicketMenuLock', 'Name' => 'Lock', 'PopupType' => '', 'Target' => '' };</pre>

Ticket::Frontend::PreMenuModule###200-Zoom

Description	Value
Description:	Shows a link in the menu to zoom a ticket in the ticket overviews of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'200-Zoom'} = { 'Action' => 'AgentTicketZoom', 'Description' => 'Look into a ticket!', 'Link' => 'Action=AgentTicketZoom;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Zoom', 'PopupType' => '', 'Target' => '' };</pre>

Ticket::Frontend::PreMenuModule###210-History

Description	Value
Description:	Shows a link in the menu to see the history of a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'210- History'} = { 'Action' => 'AgentTicketHistory', 'Description' => 'Shows the ticket history!', 'Link' => 'Action=AgentTicketHistory;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'History', 'PopupType' => 'TicketHistory', 'Target' => '' }; </pre>

Ticket::Frontend::PreMenuModule###300-Priority

Description	Value
Description:	Shows a link in the menu to set the priority of a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'300- Priority'} = { 'Action' => 'AgentTicketPriority', 'Description' => 'Change the ticket priority!', 'Link' => 'Action=AgentTicketPriority;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Priority', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::PreMenuModule###420-Note

Description	Value
Description:	Shows a link in the menu to add a note to a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'420-Note'} = { 'Action' => 'AgentTicketNote', 'Description' => 'Add a note to this ticket', 'Link' => 'Action=AgentTicketNote;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Note', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::PreMenuModule###440-Close

Description	Value
Description:	Shows a link in the menu to close a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'440-Close'} = { 'Action' => 'AgentTicketClose', 'Description' => 'Close this ticket', 'Link' => 'Action=AgentTicketClose;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Close', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

Ticket::Frontend::PreMenuModule###445-Move

Description	Value
Description:	Shows a link in the menu to move a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'445-Move'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Change queue!', 'Module' => 'Kernel::Output::HTML::TicketMenuMove', 'Name' => 'Move' }; </pre>

Ticket::Frontend::PreMenuModule###450-Delete

Description	Value
Description:	Shows a link in the menu to delete a ticket in every ticket overview of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'450-Delete'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Delete this ticket', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Delete', 'PopupType' => '', 'Target' => '' }; </pre>

Ticket::Frontend::PreMenuModule###460-Spam

Description	Value
Description:	Shows a link in the menu to set a ticket as spam in every ticket overview of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'460-Spam'} = { 'Action' => 'AgentTicketMove', </pre>

Description	Value
	<pre>'Description' => 'Mark as Spam!', 'Link' => 'Action=AgentTicketMove;TicketID= \$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Spam', 'PopupType' => '', 'Target' => '' };</pre>

Frontend::Agent::Ticket::ViewBounce

Ticket::Frontend::AgentTicketBounce###Permission

Description	Value
Description:	Required permissions to use the ticket bounce screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}- >{'Permission'} = 'bounce';</pre>

Ticket::Frontend::AgentTicketBounce###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket bounce screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}- >{'RequiredLock'} = '1';</pre>

Ticket::Frontend::AgentTicketBounce###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateDefault'} = 'closed successful';</pre>

Ticket::Frontend::AgentTicketBounce###StateType

Description	Value
Description:	Defines the next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateType'} = ['open', 'closed'];</pre>

Ticket::Frontend::BounceText

Description	Value
Description:	Defines the default ticket bounced notification for customer/sender in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::BounceText'} = 'Your email with ticket number "<OTRS_TICKET>" is bounced to "<OTRS_BOUNCE_TO>". Contact this address for further information.';</pre>

Frontend::Agent::Ticket::ViewBulk

Ticket::Frontend::AgentTicketBulk###RequiredLock

Description	Value
Description:	Automatically lock and set owner to current Agent after selecting for an Bulk Action.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'RequiredLock'} = '1';</pre>

Ticket::Frontend::AgentTicketBulk###TicketType

Description	Value
Description:	Sets the ticket type in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'TicketType'} = '1';</pre>

Ticket::Frontend::AgentTicketBulk###Owner

Description	Value
Description:	Sets the ticket owner in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Owner'} = '1';</pre>

Ticket::Frontend::AgentTicketBulk###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Responsible'} = '1';</code>

Ticket::Frontend::AgentTicketBulk###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'State'} = '1';</code>

Ticket::Frontend::AgentTicketBulk###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</code>

Ticket::Frontend::AgentTicketBulk###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	0
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateDefault'} = 'open';</code>

Ticket::Frontend::AgentTicketBulk###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Priority'} = '1';</code>

Ticket::Frontend::AgentTicketBulk###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'PriorityDefault'} = '3 normal';</code>

Ticket::Frontend::AgentTicketBulk###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'ArticleTypeDefault'} = 'note-internal';</code>

Ticket::Frontend::AgentTicketBulk###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}- >{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</pre>

Frontend::Agent::Ticket::ViewClose

Ticket::Frontend::AgentTicketClose###Permission

Description	Value
Description:	Required permissions to use the close ticket screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}- >{'Permission'} = 'close';</pre>

Ticket::Frontend::AgentTicketClose###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the close ticket screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}- >{'RequiredLock'} = '1';</pre>

Ticket::Frontend::AgentTicketClose###TicketType

Description	Value
Description:	Sets the ticket type in the close ticket screen of the agent interface (Ticket::Type needs to be activated).

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'TicketType'} = '0';</pre>

Ticket::Frontend::AgentTicketClose###Service

Description	Value
Description:	Sets the service in the close ticket screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Service'} = '0';</pre>

Ticket::Frontend::AgentTicketClose###Owner

Description	Value
Description:	Sets the ticket owner in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Owner'} = '0';</pre>

Ticket::Frontend::AgentTicketClose###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'OwnerMandatory'} = '0';</code>

Ticket::Frontend::AgentTicketClose###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Responsible'} = '0';</code>

Ticket::Frontend::AgentTicketClose###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'State'} = '1';</code>

Ticket::Frontend::AgentTicketClose###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateType'} = ['closed'];</code>

Ticket::Frontend::AgentTicketClose###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateDefault'} = 'closed successful';</pre>

Ticket::Frontend::AgentTicketClose###Note

Description	Value
Description:	Allows adding notes in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Note'} = '1';</pre>

Ticket::Frontend::AgentTicketClose###Subject

Description	Value
Description:	Sets the default subject for notes added in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Subject'} = '\$Text{"Close"}';</pre>

Ticket::Frontend::AgentTicketClose###Body

Description	Value
Description:	Sets the default body text for notes added in the close ticket screen of the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Body'} = '';</code>

Ticket::Frontend::AgentTicketClose###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InvolvedAgent'} = '0';</code>

Ticket::Frontend::AgentTicketClose###InformAgent

Description	Value
Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InformAgent'} = '0';</code>

Ticket::Frontend::AgentTicketClose###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

Ticket::Frontend::AgentTicketClose###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

Ticket::Frontend::AgentTicketClose###Priority

Description	Value
Description:	Shows the ticket priority options in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Priority'} = '0';</pre>

Ticket::Frontend::AgentTicketClose###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'PriorityDefault'} = '3 normal';</pre>

Ticket::Frontend::AgentTicketClose###Title

Description	Value
Description:	Shows the title fields in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Title'} = '0';</pre>

Ticket::Frontend::AgentTicketClose###HistoryType

Description	Value
Description:	Defines the history type for the close ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryType'} = 'AddNote';</pre>

Ticket::Frontend::AgentTicketClose###HistoryComment

Description	Value
Description:	Defines the history comment for the close ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryComment'} = '%Close';</pre>

Ticket::Frontend::AgentTicketClose###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket close screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewCompose

Ticket::Frontend::AgentTicketCompose###Permission

Description	Value
Description:	Required permissions to use the ticket compose screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'Permission'} = 'compose';</pre>

Ticket::Frontend::AgentTicketCompose###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket compose screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'RequiredLock'} = '1';</pre>

Ticket::Frontend::AgentTicketCompose###StateDefault

Description	Value
Description:	Defines the default next state of a ticket if it is composed / answered in the ticket compose screen of the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateDefault'} = 'open';</pre>

Ticket::Frontend::AgentTicketCompose###StateType

Description	Value
Description:	Defines the next possible states after composing / answering a ticket in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateType'} = ['open', 'closed', 'pending auto', 'pending reminder'];</pre>

Ticket::Frontend::ResponseFormat

Description	Value
Description:	Defines the format of responses in the ticket compose screen of the agent interface (\$QData{"OrigFrom"} is From 1:1, \$QData{"OrigFromName"} is only realname of From).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ResponseFormat'} = '\$QData{"Salutation"} \$QData{"StdResponse"} \$QData{"Signature"} \$TimeShort{"\$QData{"Created"}} - \$QData{"OrigFromName"} \$Text{"wrote"}: \$QData{"Body"} ';</pre>

Ticket::Frontend::Quote

Description	Value
Description:	Defines the used character for email quotes in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::Quote'} = '>';</code>

Ticket::Frontend::ComposeAddCustomerAddress

Description	Value
Description:	Adds customers email addresses to recipients in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ComposeAddCustomerAddress'} = '1';</code>

Ticket::Frontend::ComposeReplaceSenderAddress

Description	Value
Description:	Replaces the original sender with current customer's email address on compose answer in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ComposeReplaceSenderAddress'} = '0';</code>

Ticket::Frontend::ComposeExcludeCcRecipients

Description	Value
Description:	Uses Cc recipients in reply Cc list on compose an email answer in the ticket compose screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ComposeExcludeCcRecipients'} = '0';</pre>

Ticket::Frontend::AgentTicketCompose###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket compose screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewCustomer

Ticket::Frontend::AgentTicketCustomer###Permission

Description	Value
Description:	Required permissions to change the customer of a ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCustomer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'Permission'} = 'customer';</pre>

Ticket::Frontend::AgentTicketCustomer###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required to change the customer of a ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCustomer

Description	Value
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'RequiredLock'} = '0';</code>

Frontend::Agent::Ticket::ViewEmailNew

Ticket::Frontend::AgentTicketEmail###Priority

Description	Value
Description:	Sets the default priority for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Priority'} = '3 normal';</code>

Ticket::Frontend::AgentTicketEmail###ArticleType

Description	Value
Description:	Sets the default article type for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'ArticleType'} = 'email-external';</code>

Ticket::Frontend::AgentTicketEmail###SenderType

Description	Value
Description:	Sets the default sender type for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'SenderType'} = 'agent';</code>

Ticket::Frontend::AgentTicketEmail###Subject

Description	Value
Description:	Sets the default subject for new email tickets (e.g. 'email Outbound') in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Subject'} = '';</pre>

Ticket::Frontend::AgentTicketEmail###Body

Description	Value
Description:	Sets the default text for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketEmail###StateDefault

Description	Value
Description:	Sets the default next ticket state, after the creation of an email ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateDefault'} = 'open';</pre>

Ticket::Frontend::AgentTicketEmail###StateType

Description	Value
Description:	Determines the next possible ticket states, after the creation of a new email ticket in the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</pre>

Ticket::Frontend::AgentTicketEmail###HistoryType

Description	Value
Description:	Defines the history type for the email ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryType'} = 'EmailAgent';</pre>

Ticket::Frontend::AgentTicketEmail###HistoryComment

Description	Value
Description:	Defines the history comment for the email ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryComment'} = '';</pre>

Ticket::Frontend::AgentTicketEmail###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket email screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewEscalation

Ticket::Frontend::AgentTicketEscalationView###ViewableTicketsPage

Description	Value
Description:	Shows all open tickets (even if they are locked) in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'ViewableTicketsPage'} = '50';</pre>

Ticket::Frontend::AgentTicketEscalationView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'SortBy::Default'} = 'EscalationTime';</pre>

Ticket::Frontend::AgentTicketEscalationView###Order::Default

Description	Value
Description:	Defines the default ticket order (after priority sort) in the escalation view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'Order::Default'} = 'Up';</pre>

Frontend::Agent::Ticket::ViewForward

Ticket::Frontend::AgentTicketForward###Permission

Description	Value
Description:	Required permissions to use the ticket forward screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'Permission'} = 'forward';</pre>

Ticket::Frontend::AgentTicketForward###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket forward screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RequiredLock'} = '1';</pre>

Ticket::Frontend::AgentTicketForward###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after being forwarded, in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateDefault'} = 'closed successful';</pre>

Ticket::Frontend::AgentTicketForward###StateType

Description	Value
Description:	Defines the next possible states after forwarding a ticket in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</pre>

Ticket::Frontend::AgentTicketForward###ArticleTypeDefault

Description	Value
Description:	Defines the default type of forwarded message in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypeDefault'} = 'email-external';</pre>

Ticket::Frontend::AgentTicketForward###ArticleTypes

Description	Value
Description:	Specifies the different article types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypes'} = ['email-external', 'email-internal'];</pre>

Ticket::Frontend::AgentTicketForward###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket forward screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewFreeText

Ticket::Frontend::AgentTicketFreeText###Permission

Description	Value
Description:	Required permissions to use the ticket free text screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Permission'} = 'rw';</pre>

Ticket::Frontend::AgentTicketFreeText###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket free text screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText

Description	Value
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RequiredLock'} = '0';</code>

Ticket::Frontend::AgentTicketFreeText###TicketType

Description	Value
Description:	Sets the ticket type in the ticket free text screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'TicketType'} = '1';</code>

Ticket::Frontend::AgentTicketFreeText###Service

Description	Value
Description:	Sets the service in the ticket free text screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Service'} = '1';</code>

Ticket::Frontend::AgentTicketFreeText###Owner

Description	Value
Description:	Sets the ticket owner in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Owner'} = '0';</code>

Ticket::Frontend::AgentTicketFreeText###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'OwnerMandatory'} = '0';</pre>

Ticket::Frontend::AgentTicketFreeText###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Responsible'} = '0';</pre>

Ticket::Frontend::AgentTicketFreeText###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'State'} = '0';</pre>

Ticket::Frontend::AgentTicketFreeText###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket free text screen of the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</pre>

Ticket::Frontend::AgentTicketFreeText###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateDefault'} = 'open';</pre>

Ticket::Frontend::AgentTicketFreeText###Note

Description	Value
Description:	Allows adding notes in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Note'} = '0';</pre>

Ticket::Frontend::AgentTicketFreeText###Subject

Description	Value
Description:	Defines the default subject of a note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Subject'} = '\$Text{"Note"}';</pre>

Ticket::Frontend::AgentTicketFreeText###Body

Description	Value
Description:	Defines the default body of a note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketFreeText###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InvolvedAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketFreeText###InformAgent

Description	Value
Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InformAgent'} = '0';</code>

Ticket::Frontend::AgentTicketFreeText###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypeDefault'} = 'note-internal';</code>

Ticket::Frontend::AgentTicketFreeText###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</code>

Ticket::Frontend::AgentTicketFreeText###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Priority'} = '0';</code>

Ticket::Frontend::AgentTicketFreeText###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'PriorityDefault'} = '3 normal';</pre>

Ticket::Frontend::AgentTicketFreeText###Title

Description	Value
Description:	Shows the title fields in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Title'} = '1';</pre>

Ticket::Frontend::AgentTicketFreeText###HistoryType

Description	Value
Description:	Defines the history type for the ticket free text screen action, which gets used for ticket history.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryType'} = 'AddNote';</pre>

Ticket::Frontend::AgentTicketFreeText###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket free text screen action, which gets used for ticket history.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryComment'} = '%FreeText';</code>

Ticket::Frontend::AgentTicketFreeText###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket free text screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'DynamicField'} = {};</code>

Frontend::Agent::Ticket::ViewHistory

Ticket::Frontend::HistoryOrder

Description	Value
Description:	Shows the ticket history (reverse ordered) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewHistory
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::HistoryOrder'} = 'normal';</code>

Frontend::Agent::Ticket::ViewMailbox

Ticket::Frontend::AgentTicketLockedView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the locked ticket view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketLockedView'}- >{'SortBy::Default'} = 'Age';</pre>

Ticket::Frontend::AgentTicketLockedView###Order::Default

Description	Value
Description:	Defines the default ticket order in the ticket locked view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketLockedView'}- >{'Order::Default'} = 'Up';</pre>

Ticket::Frontend::AgentTicketResponsibleView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the responsible view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsibleView'}- >{'SortBy::Default'} = 'Age';</pre>

Ticket::Frontend::AgentTicketResponsibleView###Order::Default

Description	Value
Description:	Defines the default ticket order in the responsible view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'Order::Default'} = 'Up';</pre>

Ticket::Frontend::AgentTicketWatchView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the watch view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'SortBy::Default'} = 'Age';</pre>

Ticket::Frontend::AgentTicketWatchView###Order::Default

Description	Value
Description:	Defines the default ticket order in the watch view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'Order::Default'} = 'Up';</pre>

Frontend::Agent::Ticket::ViewMerge

Ticket::Frontend::AgentTicketMerge###Permission

Description	Value
Description:	Required permissions to use the ticket merge screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'Permission'} = 'rw';</pre>

Ticket::Frontend::AgentTicketMerge###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket merge screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RequiredLock'} = '1';</pre>

Ticket::Frontend::MergeText

Description	Value
Description:	When tickets are merged, the customer can be informed per email by setting the check box "Inform Sender". In this text area, you can define a pre-formatted text which can later be modified by the agents.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MergeText'} = 'Your email with ticket number "<OTRS_TICKET>" is merged to "<OTRS_MERGE_TO_TICKET>".';</pre>

Ticket::Frontend::AutomaticMergeText

Description	Value
Description:	When tickets are merged, a note will be added automatically to the ticket which is no longer active. In this text area you can define this text (This text cannot be changed by the agent).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AutomaticMergeText'} = 'Merged Ticket <OTRS_TICKET> to <OTRS_MERGE_TO_TICKET>.';</code>

Frontend::Agent::Ticket::ViewMove

Ticket::Frontend::MoveType

Description	Value
Description:	Determines if the list of possible queues to move to ticket into should be displayed in a dropdown list or in a new window in the agent interface. If "New Window" is set you can add a move note to the ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::MoveType'} = 'form';</code>

Ticket::Frontend::AgentTicketMove###State

Description	Value
Description:	Allows to set a new ticket state in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'State'} = '1';</code>

Ticket::DefaultNextMoveStateType

Description	Value
Description:	Defines the next state of a ticket after being moved to another queue, in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::DefaultNextMoveStateType'} = ['open', 'closed'];</code>

Ticket::Frontend::AgentTicketMove###Priority

Description	Value
Description:	Shows the ticket priority options in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Priority'} = '0';</pre>

Ticket::Frontend::AgentTicketMove###NextScreen

Description	Value
Description:	Determines the next screen after the ticket is moved. LastScreenOverview will return to search results, queueview, dashboard or the like, LastScreenView will return to TicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'NextScreen'} = 'LastScreenView';</pre>

Ticket::Frontend::AgentTicketMove###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket move screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Subject'} = '\$Text{"Change Queue"}';</pre>

Ticket::Frontend::AgentTicketMove###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket move screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}- >{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketMove###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket move screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}- >{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewNote

Ticket::Frontend::AgentTicketNote###Permission

Description	Value
Description:	Required permissions to use the ticket note screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'Permission'} = 'note';</pre>

Ticket::Frontend::AgentTicketNote###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket note screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote

Description	Value
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RequiredLock'} = '0';</code>

Ticket::Frontend::AgentTicketNote###TicketType

Description	Value
Description:	Sets the ticket type in the ticket note screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'TicketType'} = '0';</code>

Ticket::Frontend::AgentTicketNote###Service

Description	Value
Description:	Sets the service in the ticket note screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Service'} = '0';</code>

Ticket::Frontend::AgentTicketNote###Owner

Description	Value
Description:	Sets the ticket owner in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Owner'} = '0';</code>

Ticket::Frontend::AgentTicketNote###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'OwnerMandatory'} = '0';</pre>

Ticket::Frontend::AgentTicketNote###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Responsible'} = '0';</pre>

Ticket::Frontend::AgentTicketNote###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'State'} = '0';</pre>

Ticket::Frontend::AgentTicketNote###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</pre>

Ticket::Frontend::AgentTicketNote###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'StateDefault'} = 'open';</pre>

Ticket::Frontend::AgentTicketNote###Note

Description	Value
Description:	Allows adding notes in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'Note'} = '1';</pre>

Ticket::Frontend::AgentTicketNote###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'Subject'} = '\$Text{"Note"}';</pre>

Ticket::Frontend::AgentTicketNote###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketNote###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'InvolvedAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketNote###InformAgent

Description	Value
Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'InformAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketNote###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

Ticket::Frontend::AgentTicketNote###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</pre>

Ticket::Frontend::AgentTicketNote###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Priority'} = '0';</pre>

Ticket::Frontend::AgentTicketNote###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket note screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'PriorityDefault'} = '3 normal';</pre>

Ticket::Frontend::AgentTicketNote###Title

Description	Value
Description:	Shows the title fields in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'Title'} = '0';</pre>

Ticket::Frontend::AgentTicketNote###HistoryType

Description	Value
Description:	Defines the history type for the ticket note screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}- >{'HistoryType'} = 'AddNote';</pre>

Ticket::Frontend::AgentTicketNote###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket note screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'HistoryComment'} = '%Note';</code>

Ticket::Frontend::AgentTicketNote###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket note screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'DynamicField'} = {};</code>

Frontend::Agent::Ticket::ViewOwner

Ticket::Frontend::AgentTicketOwner###Permission

Description	Value
Description:	Required permissions to use the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Permission'} = 'owner';</code>

Ticket::Frontend::AgentTicketOwner###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket owner screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RequiredLock'} = '0';</code>

Ticket::Frontend::AgentTicketOwner###TicketType

Description	Value
Description:	Sets the ticket type in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}- >{'TicketType'} = '0';</pre>

Ticket::Frontend::AgentTicketOwner###Service

Description	Value
Description:	Sets the service in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}- >{'Service'} = '0';</pre>

Ticket::Frontend::AgentTicketOwner###Owner

Description	Value
Description:	Sets the ticket owner in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}- >{'Owner'} = '1';</pre>

Ticket::Frontend::AgentTicketOwner###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'OwnerMandatory'} = '1';</code>

Ticket::Frontend::AgentTicketOwner###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Responsible'} = '0';</code>

Ticket::Frontend::AgentTicketOwner###State

Description	Value
Description:	If a note is added by an agent, sets the state of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'State'} = '0';</code>

Ticket::Frontend::AgentTicketOwner###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}- >{'StateType'} = ['open', 'pending reminder', 'pending auto'];</pre>

Ticket::Frontend::AgentTicketOwner###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}- >{'StateDefault'} = 'open';</pre>

Ticket::Frontend::AgentTicketOwner###Note

Description	Value
Description:	Allows adding notes in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}- >{'Note'} = '1';</pre>

Ticket::Frontend::AgentTicketOwner###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}- >{'Subject'} = '\$Text{"Owner Update"}!';</pre>

Ticket::Frontend::AgentTicketOwner###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketOwner###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InvolvedAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketOwner###InformAgent

Description	Value
Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InformAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketOwner###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket owner screen of a zoomed ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

Ticket::Frontend::AgentTicketOwner###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

Ticket::Frontend::AgentTicketOwner###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Priority'} = '0';</pre>

Ticket::Frontend::AgentTicketOwner###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'PriorityDefault'} = '3 normal';</pre>

Ticket::Frontend::AgentTicketOwner###Title

Description	Value
Description:	Shows the title fields in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Title'} = '0';</pre>

Ticket::Frontend::AgentTicketOwner###HistoryType

Description	Value
Description:	Defines the history type for the ticket owner screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryType'} = 'AddNote';</pre>

Ticket::Frontend::AgentTicketOwner###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket owner screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryComment'} = '%Owner';</code>

Ticket::Frontend::AgentTicketOwner###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket owner screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'DynamicField'} = {};</code>

Frontend::Agent::Ticket::ViewPending

Ticket::Frontend::AgentTicketPending###Permission

Description	Value
Description:	Required permissions to use the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Permission'} = 'pending';</code>

Ticket::Frontend::AgentTicketPending###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket pending screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RequiredLock'} = '1';</code>

Ticket::Frontend::AgentTicketPending###TicketType

Description	Value
Description:	Sets the ticket type in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'TicketType'} = '0';</pre>

Ticket::Frontend::AgentTicketPending###Service

Description	Value
Description:	Sets the service in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Service'} = '0';</pre>

Ticket::Frontend::AgentTicketPending###Owner

Description	Value
Description:	Sets the ticket owner in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Owner'} = '0';</pre>

Ticket::Frontend::AgentTicketPending###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'OwnerMandatory'} = '0';</code>

Ticket::Frontend::AgentTicketPending###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Responsible'} = '0';</code>

Ticket::Frontend::AgentTicketPending###State

Description	Value
Description:	If a note is added by an agent, sets the state of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'State'} = '1';</code>

Ticket::Frontend::AgentTicketPending###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateType'} = ['pending reminder', 'pending auto'];</pre>

Ticket::Frontend::AgentTicketPending###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateDefault'} = 'pending reminder';</pre>

Ticket::Frontend::AgentTicketPending###Note

Description	Value
Description:	Allows adding notes in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Note'} = '1';</pre>

Ticket::Frontend::AgentTicketPending###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Subject'} = '\$Text{"Pending"}!';</pre>

Ticket::Frontend::AgentTicketPending###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketPending###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InvolvedAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketPending###InformAgent

Description	Value
Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InformAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketPending###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket pending screen of a zoomed ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

Ticket::Frontend::AgentTicketPending###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

Ticket::Frontend::AgentTicketPending###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Priority'} = '0';</pre>

Ticket::Frontend::AgentTicketPending###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'PriorityDefault'} = '3 normal';</code>

Ticket::Frontend::AgentTicketPending###Title

Description	Value
Description:	Shows the title fields in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Title'} = '0';</code>

Ticket::Frontend::AgentTicketPending###HistoryType

Description	Value
Description:	Defines the history type for the ticket pending screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryType'} = 'AddNote';</code>

Ticket::Frontend::AgentTicketPending###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket pending screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryComment'} = '%Pending';</code>

Ticket::Frontend::AgentTicketPending###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket pending screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'DynamicField'} = {};</code>

Frontend::Agent::Ticket::ViewPhoneInbound

Ticket::Frontend::AgentTicketPhoneInbound###Permission

Description	Value
Description:	Required permissions to use the ticket phone inbound screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Permission'} = 'phone';</code>

Ticket::Frontend::AgentTicketPhoneInbound###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket phone inbound screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'RequiredLock'} = '0';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###ArticleType

Description	Value
Description:	Defines the default type of the note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'ArticleType'} = 'phone';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###SenderType

Description	Value
Description:	Defines the default sender type for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'SenderType'} = 'customer';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###Subject

Description	Value
Description:	Defines the default subject for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'Subject'} = '\$Text{"Phone call"}!';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###Body

Description	Value
Description:	Defines the default note body text for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###State

Description	Value
Description:	Defines the default ticket next state after adding a phone note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'State'} = 'open';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###StateType

Description	Value
Description:	Next possible ticket states after adding a phone note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'</pre>

Description	Value
];

Ticket::Frontend::AgentTicketPhoneInbound###HistoryType

Description	Value
Description:	Defines the history type for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'HistoryType'} = 'PhoneCallCustomer';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'HistoryComment'} = '';</pre>

Ticket::Frontend::AgentTicketPhoneInbound###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket phone inbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneInbound'}- >{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewPhoneNew

Ticket::Frontend::AgentTicketPhone###Priority

Description	Value
Description:	Sets the default priority for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Priority'} = '3 normal';</pre>

Ticket::Frontend::AgentTicketPhone###ArticleType

Description	Value
Description:	Sets the default article type for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'ArticleType'} = 'phone';</pre>

Ticket::Frontend::AgentTicketPhone###SenderType

Description	Value
Description:	Sets the default sender type for new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SenderType'} = 'customer';</pre>

Ticket::Frontend::AgentTicketPhone###Subject

Description	Value
Description:	Sets the default subject for new phone tickets (e.g. 'Phone call') in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Subject'} = '';</code>

Ticket::Frontend::AgentTicketPhone###Body

Description	Value
Description:	Sets the default note text for new telephone tickets. E.g 'New ticket via call' in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Body'} = '';</code>

Ticket::Frontend::AgentTicketPhone###StateDefault

Description	Value
Description:	Sets the default next state for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'StateDefault'} = 'open';</code>

Ticket::Frontend::AgentTicketPhone###StateType

Description	Value
Description:	Determines the next possible ticket states, after the creation of a new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}- >{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</pre>

Ticket::Frontend::AgentTicketPhone###HistoryType

Description	Value
Description:	Defines the history type for the phone ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}- >{'HistoryType'} = 'PhoneCallCustomer';</pre>

Ticket::Frontend::AgentTicketPhone###HistoryComment

Description	Value
Description:	Defines the history comment for the phone ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}- >{'HistoryComment'} = '';</pre>

Ticket::Frontend::AgentTicketPhone###SplitLinkType

Description	Value
Description:	Sets the default link type of splitted tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SplitLinkType'} = { 'Direction' => 'Target', 'LinkType' => 'ParentChild' };</pre>

Ticket::Frontend::AgentTicketPhone###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket phone screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewPhoneOutbound

Ticket::Frontend::AgentTicketPhoneOutbound###Permission

Description	Value
Description:	Required permissions to use the ticket phone outbound screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Permission'} = 'phone';</pre>

Ticket::Frontend::AgentTicketPhoneOutbound###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket phone outbound screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'RequiredLock'} = '1';</pre>

Ticket::Frontend::AgentTicketPhoneOutbound###ArticleType

Description	Value
Description:	Defines the default type of the note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'ArticleType'} = 'phone';</pre>

Ticket::Frontend::AgentTicketPhoneOutbound###SenderType

Description	Value
Description:	Defines the default sender type for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'SenderType'} = 'agent';</pre>

Ticket::Frontend::AgentTicketPhoneOutbound###Subject

Description	Value
Description:	Defines the default subject for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'Subject'} = '\$Text{"Phone call"}!';</pre>

Ticket::Frontend::AgentTicketPhoneOutbound###Body

Description	Value
Description:	Defines the default note body text for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketPhoneOutbound###State

Description	Value
Description:	Defines the default ticket next state after adding a phone note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'State'} = 'closed successful';</pre>

Ticket::Frontend::AgentTicketPhoneOutbound###StateType

Description	Value
Description:	Next possible ticket states after adding a phone note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'StateType'} = [</pre>

Description	Value
	'open', 'pending auto', 'pending reminder', 'closed'];

Ticket::Frontend::AgentTicketPhoneOutbound###HistoryType

Description	Value
Description:	Defines the history type for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryType'} = 'PhoneCallAgent';

Ticket::Frontend::AgentTicketPhoneOutbound###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryComment'} = '';

Ticket::Frontend::AgentTicketPhoneOutbound###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket phone outbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}- >{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewPrint

Ticket::Frontend::AgentTicketPrint###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket print screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPrint
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPrint'}- >{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewPriority

Ticket::Frontend::AgentTicketPriority###Permission

Description	Value
Description:	Required permissions to use the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}- >{'Permission'} = 'priority';</pre>

Ticket::Frontend::AgentTicketPriority###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket priority screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RequiredLock'} = '1';</code>

Ticket::Frontend::AgentTicketPriority###TicketType

Description	Value
Description:	Sets the ticket type in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'TicketType'} = '0';</code>

Ticket::Frontend::AgentTicketPriority###Service

Description	Value
Description:	Sets the service in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Service'} = '0';</code>

Ticket::Frontend::AgentTicketPriority###Owner

Description	Value
Description:	Sets the ticket owner in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Owner'} = '0';</code>

Ticket::Frontend::AgentTicketPriority###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'OwnerMandatory'} = '0';</pre>

Ticket::Frontend::AgentTicketPriority###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Responsible'} = '0';</pre>

Ticket::Frontend::AgentTicketPriority###State

Description	Value
Description:	If a note is added by an agent, sets the state of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'State'} = '0';</pre>

Ticket::Frontend::AgentTicketPriority###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket priority screen of a zoomed ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateType'} = ['open', 'pending reminder', 'pending auto'];</pre>

Ticket::Frontend::AgentTicketPriority###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateDefault'} = 'open';</pre>

Ticket::Frontend::AgentTicketPriority###Note

Description	Value
Description:	Allows adding notes in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Note'} = '1';</pre>

Ticket::Frontend::AgentTicketPriority###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket priority screen of a zoomed ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Subject'} = '\$Text{"Priority Update"}!';</pre>

Ticket::Frontend::AgentTicketPriority###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketPriority###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InvolvedAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketPriority###InformAgent

Description	Value
Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority

Description	Value
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InformAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketPriority###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

Ticket::Frontend::AgentTicketPriority###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

Ticket::Frontend::AgentTicketPriority###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Priority'} = '1';</code>

Ticket::Frontend::AgentTicketPriority###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'PriorityDefault'} = '3 normal';</code>

Ticket::Frontend::AgentTicketPriority###Title

Description	Value
Description:	Shows the title fields in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Title'} = '0';</code>

Ticket::Frontend::AgentTicketPriority###HistoryType

Description	Value
Description:	Defines the history type for the ticket priority screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryType'} = 'AddNote';</code>

Ticket::Frontend::AgentTicketPriority###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket priority screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryComment'} = '%Priority';</pre>

Ticket::Frontend::AgentTicketPriority###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket priority screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewQueue**Ticket::Frontend::AgentTicketQueue###StripEmptyLines**

Description	Value
Description:	Strips empty lines on the ticket preview in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'StripEmptyLines'} = '0';</pre>

Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets

Description	Value
Description:	Shows all both ro and rw queues in the queue view.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'ViewAllPossibleTickets'} = '0';</pre>

Ticket::Frontend::AgentTicketQueue###HighlightAge1

Description	Value
Description:	Sets the age in minutes (first level) for highlighting queues that contain untouched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge1'} = '1440';</pre>

Ticket::Frontend::AgentTicketQueue###HighlightAge2

Description	Value
Description:	Sets the age in minutes (second level) for highlighting queues that contain untouched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge2'} = '2880';</pre>

Ticket::Frontend::AgentTicketQueue###Blink

Description	Value
Description:	Activates a blinking mechanism of the queue that contains the oldest ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'Blink'} = '1';</pre>

Ticket::Frontend::AgentTicketQueue###QueueSort

Description	Value
Description:	Sorts the tickets (ascendingly or descendingly) when a single queue is selected in the queue view and after the tickets are sorted by priority. Values: 0 = ascending (oldest on top, default), 1 = descending (youngest on top). Use the QueueID for the key and 0 or 1 for value.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'QueueSort'} = { '3' => '0', '7' => '1' };</pre>

Ticket::Frontend::AgentTicketQueue###SortBy::Default

Description	Value
Description:	Defines the default sort criteria for all queues displayed in the queue view, after sort by priority is done.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'SortBy::Default'} = 'Age';</pre>

Ticket::Frontend::AgentTicketQueue###Order::Default

Description	Value
Description:	Defines the default sort order for all queues in the queue view, after priority sort.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}- >{'Order::Default'} = 'Up';</pre>

Frontend::Agent::Ticket::ViewResponsible

Ticket::Frontend::AgentTicketResponsible###Permission

Description	Value
Description:	Required permissions to use the ticket responsible screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Permission'} = 'responsible';</pre>

Ticket::Frontend::AgentTicketResponsible###RequiredLock

Description	Value
Description:	Defines if a ticket lock is required in the ticket responsible screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'RequiredLock'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###TicketType

Description	Value
Description:	Sets the ticket type in the ticket responsible screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'TicketType'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###Service

Description	Value
Description:	Sets the service in the ticket responsible screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Service'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###Owner

Description	Value
Description:	Sets the ticket owner in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Owner'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'OwnerMandatory'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Responsible'} = '1';</pre>

Ticket::Frontend::AgentTicketResponsible###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'State'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'StateType'} = ['open', 'pending reminder', 'pending auto'];</pre>

Ticket::Frontend::AgentTicketResponsible###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'StateDefault'} = 'open';</pre>

Ticket::Frontend::AgentTicketResponsible###Note

Description	Value
Description:	Allows adding notes in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Note'} = '1';</pre>

Ticket::Frontend::AgentTicketResponsible###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Subject'} = '\$Text{"Responsible Update"}!';</pre>

Ticket::Frontend::AgentTicketResponsible###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket responsible screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Body'} = '';</pre>

Ticket::Frontend::AgentTicketResponsible###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'InvolvedAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###InformAgent

Description	Value
Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'InformAgent'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket responsible screen of the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'ArticleTypeDefault'} = 'note-internal';</pre>

Ticket::Frontend::AgentTicketResponsible###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

Ticket::Frontend::AgentTicketResponsible###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Priority'} = '0';</pre>

Ticket::Frontend::AgentTicketResponsible###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket responsible screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'PriorityDefault'} = '3 normal';</pre>

Ticket::Frontend::AgentTicketResponsible###Title

Description	Value
Description:	Shows the title fields in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'Title'} = '1';</pre>

Ticket::Frontend::AgentTicketResponsible###HistoryType

Description	Value
Description:	Defines the history type for the ticket responsible screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::AgentTicketResponsible'}- >{'HistoryType'} = 'AddNote';</pre>

Ticket::Frontend::AgentTicketResponsible###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket responsible screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}- >{'HistoryComment'} = '%Responsible';</pre>

Ticket::Frontend::AgentTicketResponsible###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket responsible screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}- >{'DynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewSearch

Ticket::Frontend::AgentTicketSearch###ExtendedSearchCondition

Description	Value
Description:	Allows extended search conditions in ticket search of the agent interface. With this feature you can search w. g. with this kind of conditions like "(key1&&key2)" or "(key1 key2)".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}- >{'ExtendedSearchCondition'} = '1';</pre>

Ticket::Frontend::AgentTicketSearch###SearchLimit

Description	Value
Description:	Maximum number of tickets to be displayed in the result of a search in the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchLimit'} = '2000';</code>

Ticket::Frontend::AgentTicketSearch###SearchPageShown

Description	Value
Description:	Number of tickets to be displayed in each page of a search result in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchPageShown'} = '40';</code>

Ticket::Frontend::AgentTicketSearch###SearchViewableTicketLines

Description	Value
Description:	Number of lines (per ticket) that are shown by the search utility in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchViewableTicketLines'} = '10';</code>

Ticket::Frontend::AgentTicketSearch###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting of the ticket search result of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SortBy::Default'} = 'Age';</pre>

Ticket::Frontend::AgentTicketSearch###Order::Default

Description	Value
Description:	Defines the default ticket order in the ticket search result of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Order::Default'} = 'Down';</pre>

Ticket::Frontend::AgentTicketSearch###SearchArticleCSVTree

Description	Value
Description:	Exports the whole article tree in search result (it can affect the system performance).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchArticleCSVTree'} = '0';</pre>

Ticket::Frontend::AgentTicketSearch###SearchCSVData

Description	Value
Description:	Data used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchCSVData'} = ['TicketNumber', 'Age',</pre>

Description	Value
	<pre>'Created', 'Closed', 'FirstLock', 'FirstResponse', 'State', 'Priority', 'Queue', 'Lock', 'Owner', 'UserFirstname', 'UserLastname', 'CustomerID', 'CustomerName', 'From', 'Subject', 'AccountedTime', 'ArticleTree', 'SolutionInMin', 'SolutionDiffInMin', 'FirstResponseInMin', 'FirstResponseDiffInMin'];</pre>

Ticket::Frontend::AgentTicketSearch###ArticleCreateTime

Description	Value
Description:	Includes article create times in the ticket search of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'ArticleCreateTime'} = '0';</pre>

Ticket::Frontend::AgentTicketSearch###Defaults###Fulltext

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Fulltext'} = '';</pre>

Ticket::Frontend::AgentTicketSearch###Defaults###TicketNumber

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketNumber'} = '';</pre>

Ticket::Frontend::AgentTicketSearch###Defaults###Title

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Title'} = '';</pre>

Ticket::Frontend::AgentTicketSearch###Defaults###From

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'From'} = '';</pre>

Ticket::Frontend::AgentTicketSearch###Defaults###To

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'To'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###Cc

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Cc'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###Subject

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Subject'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###Body

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Body'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###CustomerID

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerID'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###CustomerUserLogin

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerUserLogin'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###StateIDs

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'StateIDs'} = [];</code>

Ticket::Frontend::AgentTicketSearch###Defaults###QueueIDs

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'QueueIDs'} = [];</code>

Ticket::Frontend::AgentTicketSearch####Defaults####TicketCreateTimePoint

Description	Value
Description:	Default data to use on attribute for ticket search screen. Example: "TicketCreateTimePointFormat=year;TicketCreateTimePointStart=Last;TicketCreateTimePoint
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimePoint'} = '';</pre>

Ticket::Frontend::AgentTicketSearch####Defaults####TicketCreateTimeSlot

Description	Value
Description:	Default data to use on attribute for ticket search screen. Example: "TicketCreateTimeStartYear=2010;TicketCreateTimeStartMonth=10;TicketCreateTimeStartDay
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimeSlot'} = '';</pre>

Ticket::Frontend::AgentTicketSearch####Defaults####TicketChangeTimePoint

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimePoint'} = '';</pre>

Ticket::Frontend::AgentTicketSearch####Defaults####TicketChangeTimeSlot

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimeSlot'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###TicketCloseTimePoint

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimePoint'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###TicketCloseTimeSlot

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimeSlot'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreateTimePoint

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimePoint'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreateTimeSlot

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimeSlot'} = '';</code>

Ticket::Frontend::AgentTicketSearch###Defaults###SearchInArchive

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'SearchInArchive'} = '';</code>

Ticket::Frontend::CustomerTicketSearch###SearchArticleCSVTree

Description	Value
Description:	Exports the whole article tree in search result (it can affect the system performance).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchArticleCSVTree'} = '0';</code>

Ticket::Frontend::AgentTicketSearch###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket search screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'DynamicField'} = {};</pre>

Ticket::Frontend::AgentTicketSearch###Defaults###DynamicField

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen. Example: a text, 1, DynamicField_Field1StartYear=2002;DynamicField_Field1StartMonth=12;DynamicField_Field
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'DynamicField'} = {};</pre>

Ticket::Frontend::AgentTicketSearch###SearchCSVDynamicField

Description	Value
Description:	Dynamic Fields used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchCSVDynamicField'} = {};</pre>

Frontend::Agent::Ticket::ViewStatus

Ticket::Frontend::AgentTicketStatusView###ViewableTicketsPage

Description	Value
Description:	Shows all open tickets (even if they are locked) in the status view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre> \$self- >{'Ticket::Frontend::AgentTicketStatusView'}- >{'ViewableTicketsPage'} = '50'; </pre>

Ticket::Frontend::AgentTicketStatusView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the status view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<pre> \$self- >{'Ticket::Frontend::AgentTicketStatusView'}- >{'SortBy::Default'} = 'Age'; </pre>

Ticket::Frontend::AgentTicketStatusView###Order::Default

Description	Value
Description:	Defines the default ticket order (after priority sort) in the status view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<pre> \$self- >{'Ticket::Frontend::AgentTicketStatusView'}- >{'Order::Default'} = 'Down'; </pre>

Frontend::Agent::Ticket::ViewZoom

Ticket::Frontend::PlainView

Description	Value
Description:	Shows a link to see a zoomed email ticket in plain text.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::PlainView'} = '0';</code>

Ticket::Frontend::ZoomExpand

Description	Value
Description:	Shows all the articles of the ticket (expanded) in the zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomExpand'} = '0';</code>

Ticket::Frontend::ZoomExpandSort

Description	Value
Description:	Shows the articles sorted normally or in reverse, under ticket zoom in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomExpandSort'} = 'normal';</code>

Ticket::ZoomAttachmentDisplayCount

Description	Value
Description:	Shows a count of icons in the ticket zoom, if the article has attachments.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ZoomAttachmentDisplayCount'} = '20';</code>

Ticket::ZoomTimeDisplay

Description	Value
Description:	Displays the accounted time for an article in the ticket zoom view.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	0
Required:	0
Config-Setting:	<code>\$\$Self->{'Ticket::ZoomTimeDisplay'} = '0';</code>

Ticket::Frontend::TicketArticleFilter

Description	Value
Description:	Activates the article filter in the zoom view to specify which articles should be shown.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$\$Self->{'Ticket::Frontend::TicketArticleFilter'} = '0';</code>

Ticket::Frontend::HTMLArticleHeightDefault

Description	Value
Description:	Set the default height (in pixels) of inline HTML articles in AgentTicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$\$Self->{'Ticket::Frontend::HTMLArticleHeightDefault'} = '100';</code>

Ticket::Frontend::HTMLArticleHeightMax

Description	Value
Description:	Set the maximum height (in pixels) of inline HTML articles in AgentTicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::HTMLArticleHeightMax'} = '2500';</code>

Ticket::Frontend::ZoomRichTextForce

Description	Value
Description:	Show article as rich text even if rich text writing is disabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomRichTextForce'} = '0';</code>

Ticket::Frontend::AgentTicketZoom###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket zoom screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'DynamicField'} = {};</code>

Frontend::Agent::TicketOverview

Ticket::Frontend::Overview###Small

Description	Value
Description:	Allows having a small format ticket overview (CustomerInfo => 1 - shows also the customer information).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::Overview'}->{'Small'} = {</code>

Description	Value
	<pre>'CustomerInfo' => '1', 'Module' => 'Kernel::Output::HTML::TicketOverviewSmall', 'Name' => 'Small', 'NameShort' => 'S' };</pre>

Ticket::Frontend::OverviewSmall###ColumnHeader

Description	Value
Description:	Shows either the last customer article's subject or the ticket title in the small format overview.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewSmall'}- >{'ColumnHeader'} = 'LastCustomerSubject';</pre>

Ticket::Frontend::Overview###Medium

Description	Value
Description:	Allows having a medium format ticket overview (CustomerInfo => 1 - shows also the customer information).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview'}->{'Medium'} = { 'CustomerInfo' => '0', 'Module' => 'Kernel::Output::HTML::TicketOverviewMedium', 'Name' => 'Medium', 'NameShort' => 'M', 'TicketActionsPerTicket' => '0' };</pre>

Ticket::Frontend::Overview###Preview

Description	Value
Description:	Shows a preview of the ticket overview (CustomerInfo => 1 - shows also Customer-Info, CustomerInfoMaxSize max. size in characters of Customer-Info).

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview'}->{'Preview'} = { 'CustomerInfo' => '0', 'CustomerInfoMaxSize' => '18', 'DefaultPreViewLines' => '25', 'DefaultViewNewLine' => '90', 'Module' => 'Kernel::Output::HTML::TicketOverviewPreview', 'Name' => 'Preview', 'NameShort' => 'L', 'StripEmptyLines' => '0', 'TicketActionsPerTicket' => '0' };</pre>

Ticket::Frontend::Overview::PreviewArticleSenderTypes

Description	Value
Description:	Defines which article sender types should be shown in the preview of a ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::Overview::PreviewArticleSenderTypes'} = { 'agent' => '1', 'customer' => '1', 'system' => '1' };</pre>

Ticket::Frontend::Overview::PreviewArticleTypeExpanded

Description	Value
Description:	Defines which article type should be expanded when entering the overview. If nothing defined, latest article will be expanded.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview::PreviewArticleTypeExpanded'} = '';</pre>

Ticket::Frontend::OverviewSmall###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket small format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewSmall'}->{'DynamicField'} = {};</pre>

Ticket::Frontend::OverviewMedium###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket medium format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewMedium'}->{'DynamicField'} = {};</pre>

Ticket::Frontend::OverviewPreview###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket preview format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewPreview'}->{'DynamicField'} = {};</pre>

Frontend::Agent::ToolBarModule

Frontend::ToolBarModule###1-Ticket::AgentTicketQueue

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'1- Ticket::AgentTicketQueue'} = { 'AccessKey' => 'q', 'Action' => 'AgentTicketQueue', 'CssClass' => 'QueueView', 'Link' => 'Action=AgentTicketQueue', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Queue view', 'Priority' => '1010010' }; </pre>

Frontend::ToolBarModule###2-Ticket::AgentTicketStatus

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'2- Ticket::AgentTicketStatus'} = { 'AccessKey' => 'o', 'Action' => 'AgentTicketStatusView', 'CssClass' => 'StatusView', 'Link' => 'Action=AgentTicketStatusView', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Status view', 'Priority' => '1010020' }; </pre>

Frontend::ToolBarModule###3-Ticket::AgentTicketEscalation

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'3- Ticket::AgentTicketEscalation'} = { 'AccessKey' => 'w', 'Action' => 'AgentTicketEscalationView', 'CssClass' => 'EscalationView', 'Link' => 'Action=AgentTicketEscalationView', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Escalation view', 'Priority' => '1010030' }; </pre>

Frontend::ToolBarModule###4-Ticket::AgentTicketPhone

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'4- Ticket::AgentTicketPhone'} = { 'AccessKey' => 'l', 'Action' => 'AgentTicketPhone', 'CssClass' => 'PhoneTicket', 'Link' => 'Action=AgentTicketPhone', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'New phone ticket', 'Priority' => '1020010' }; </pre>

Frontend::ToolBarModule###5-Ticket::AgentTicketEmail

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'5- Ticket::AgentTicketEmail'} = { </pre>

Description	Value
	<pre>'AccessKey' => '1', 'Action' => 'AgentTicketEmail', 'CssClass' => 'EmailTicket', 'Link' => 'Action=AgentTicketEmail', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'New email ticket', 'Priority' => '1020020' };</pre>

Frontend::ToolBarModule###6-Ticket::TicketResponsible

Description	Value
Description:	Agent interface notification module to see the number of tickets an agent is responsible for.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'6- Ticket::TicketResponsible'} = { 'CssClass' => 'Responsible', 'CssClassNew' => 'Responsible New', 'CssClassReached' => 'Responsible Reached', 'Module' => 'Kernel::Output::HTML::ToolBarTicketResponsible', 'Priority' => '1030010' };</pre>

Frontend::ToolBarModule###7-Ticket::TicketWatcher

Description	Value
Description:	Agent interface notification module to see the number of watched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'7- Ticket::TicketWatcher'} = { 'CssClass' => 'Watcher', 'CssClassNew' => 'Watcher New', 'CssClassReached' => 'Watcher Reached', 'Module' => 'Kernel::Output::HTML::ToolBarTicketWatcher', 'Priority' => '1030020' };</pre>

Frontend::ToolBarModule###8-Ticket::TicketLocked

Description	Value
Description:	Agent interface notification module to check the used charset.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'8- Ticket::TicketLocked'} = { 'CssClass' => 'Locked', 'CssClassNew' => 'Locked New', 'CssClassReached' => 'Locked Reached', 'Module' => 'Kernel::Output::HTML::ToolBarTicketLocked', 'Priority' => '1030030' }; </pre>

Frontend::ToolBarModule###9-Ticket::TicketSearchProfile

Description	Value
Description:	Agent interface module to access search profiles via nav bar.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'9- Ticket::TicketSearchProfile'} = { 'Block' => 'ToolBarSearchProfile', 'Description' => 'Search-Template', 'MaxWidth' => '40', 'Module' => 'Kernel::Output::HTML::ToolBarTicketSearchProfile', 'Name' => 'Search-Template', 'Priority' => '1990010' }; </pre>

Frontend::ToolBarModule###10-Ticket::TicketSearchFulltext

Description	Value
Description:	Agent interface module to access fulltext search via nav bar.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0

Description	Value
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'10-Ticket::TicketSearchFulltext'} = { 'Block' => 'ToolBarSearchFulltext', 'CSS' => 'Core.Agent.Toolbar.FulltextSearch.css', 'Description' => 'Fulltext-Search', 'Module' => 'Kernel::Output::HTML::ToolBarTicketSearchFulltext', 'Name' => 'Fulltext-Search', 'Priority' => '1990020', 'Size' => '10' }; </pre>

Frontend::Customer

Ticket::Frontend::CustomerTicketOverviewSortable

Description	Value
Description:	Controls if customers have the ability to sort their tickets.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::CustomerTicketOverviewSortable'} = ''; </pre>

Ticket::Frontend::CustomerTicketOverviewCustomEmptyText

Description	Value
Description:	Custom text for the page shown to customers that have no tickets yet.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::CustomerTicketOverviewCustomEmptyText'} = { 'Button' => 'Create your first ticket', 'Text' => 'Please click the button below to create your first ticket.', 'Title' => 'Welcome!' }; </pre>

Description	Value
	};

Frontend::CustomerUser::Item###9-OpenTickets

Description	Value
Description:	Customer item (icon) which shows the open tickets of this customer as info block.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::CustomerUser::Item'}->{'9-OpenTickets'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Open;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Open Tickets' }; </pre>

CustomerFrontend::CommonObject###QueueObject

Description	Value
Description:	Path of the file that stores all the settings for the QueueObject object for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerFrontend::CommonObject'}->{'QueueObject'} = 'Kernel::System::Queue'; </pre>

CustomerFrontend::CommonObject###TicketObject

Description	Value
Description:	Path of the file that stores all the settings for the TicketObject for the customer interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::CommonObject'}- >{'TicketObject'} = 'Kernel::System::Ticket';</pre>

CustomerFrontend::CommonParam###Action

Description	Value
Description:	Defines the default used Frontend-Module if no Action parameter given in the url on the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::CommonParam'}- >{'Action'} = 'CustomerTicketOverview';</pre>

CustomerFrontend::CommonParam###TicketID

Description	Value
Description:	Default ticket ID used by the system in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::CommonParam'}- >{'TicketID'} = '';</pre>

Frontend::Customer::ModuleMetaHead

CustomerFrontend::HeaderMetaModule###2-TicketSearch

Description	Value
Description:	Module to generate html OpenSearch profile for short ticket search in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleMetaHead
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::HeaderMetaModule'}- >{'2-TicketSearch'} = { 'Action' => 'CustomerTicketSearch', 'Module' => 'Kernel::Output::HTML::CustomerHeaderMetaTicketSearch' }; </pre>

Frontend::Customer::ModuleRegistration

CustomerFrontend::Module###CustomerTicketOverview

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::Module'}- >{'CustomerTicketOverview'} = { 'Description' => 'Overview of customer tickets', 'NavBar' => [{ 'AccessKey' => 'm', 'Block' => '', 'Description' => 'My Tickets', 'Link' => 'Action=CustomerTicketOverview;Subaction=MyTickets', 'LinkOption' => '', 'Name' => 'My Tickets', 'NavBar' => '', 'Prio' => '110', 'Type' => '' }, { 'AccessKey' => 'c', 'Block' => '', 'Description' => 'Company Tickets', 'Link' => 'Action=CustomerTicketOverview;Subaction=CompanyTickets', 'LinkOption' => '', 'Name' => 'Company Tickets', 'NavBar' => '', 'Prio' => '120', 'Type' => '' }], }; </pre>

Description	Value
	<pre>'NavBarName' => 'Ticket', 'Title' => 'Overview' };</pre>

CustomerFrontend::Module###CustomerTicketMessage

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}- >{'CustomerTicketMessage'} = { 'Description' => 'Create tickets', 'NavBar' => [{ 'AccessKey' => 'n', 'Block' => '', 'Description' => 'Create new Ticket', 'Link' => 'Action=CustomerTicketMessage', 'LinkOption' => '', 'Name' => 'New Ticket', 'NavBar' => '', 'Prio' => '100', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'New Ticket' };</pre>

CustomerFrontend::Module###CustomerTicketZoom

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}- >{'CustomerTicketZoom'} = { 'Description' => 'Ticket zoom view', 'Loader' => { 'JavaScript' => [</pre>

Description	Value
	<pre> 'Core.Customer.TicketZoom.js', 'Core.UI.Popup.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Zoom' }; </pre>

CustomerFrontend::Module###CustomerTicketPrint

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'CustomerFrontend::Module'}-> {'CustomerTicketPrint'} = { 'Description' => 'Customer Ticket Print Module', 'NavBarName' => '', 'Title' => 'Print' }; </pre>

CustomerFrontend::Module###CustomerZoom

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'CustomerFrontend::Module'}-> {'CustomerZoom'} = { 'Description' => 'compat mod', 'NavBarName' => '', 'Title' => '' }; </pre>

CustomerFrontend::Module###CustomerTicketAttachment

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::Module'}- >{'CustomerTicketAttachment'} = { 'Description' => 'To download attachments', 'NavBarName' => '', 'Title' => '' }; </pre>

CustomerFrontend::Module###CustomerTicketSearch

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::Module'}- >{'CustomerTicketSearch'} = { 'Description' => 'Customer ticket search', 'NavBar' => [{ 'AccessKey' => 's', 'Block' => '', 'Description' => 'Search', 'Link' => 'Action=CustomerTicketSearch', 'LinkOption' => '', 'Name' => 'Search', 'NavBar' => '', 'Prio' => '300', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Search' }; </pre>

Frontend::Customer::Preferences

CustomerPreferencesGroups###ShownTickets

Description	Value
Description:	Defines all the parameters for the ShownTickets object in the customer preferences of the customer interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerPreferencesGroups'}- >{'ShownTickets'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { '15' => '15', '20' => '20', '25' => '25', '30' => '30' }, 'DataSelected' => '25', 'Key' => 'Tickets per page', 'Label' => 'Number of displayed tickets', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserShowTickets', 'Prio' => '4000' }; </pre>

CustomerPreferencesGroups###RefreshTime

Description	Value
Description:	Defines all the parameters for the RefreshTime object in the customer preferences of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerPreferencesGroups'}- >{'RefreshTime'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { '' => 'off', '10' => '10 minutes', '15' => '15 minutes', '2' => ' 2 minutes', '5' => ' 5 minutes', '7' => ' 7 minutes' }, 'DataSelected' => '', 'Key' => 'Refresh interval', </pre>

Description	Value
	<pre>'Label' => 'Ticket overview', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserRefreshTime', 'Prio' => '4000' };</pre>

Frontend::Customer::Ticket::ViewNew

Ticket::Frontend::CustomerTicketMessage###NextScreenAfterNewTicket

Description	Value
Description:	Determines the next screen after new customer ticket in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{ 'Ticket::Frontend::CustomerTicketMessage' }->{ 'NextScreenAfterNewTicket' } = 'CustomerTicketOverview';</pre>

Ticket::Frontend::CustomerTicketMessage###Priority

Description	Value
Description:	Allows customers to set the ticket priority in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{ 'Ticket::Frontend::CustomerTicketMessage' }->{ 'Priority' } = '1';</pre>

Ticket::Frontend::CustomerTicketMessage###PriorityDefault

Description	Value
Description:	Defines the default priority of new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'PriorityDefault'} = '3 normal';</pre>

Ticket::Frontend::CustomerTicketMessage###Queue

Description	Value
Description:	Allows customers to set the ticket queue in the customer interface. If this is set to 'No', QueueDefault should be configured.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'Queue'} = '1';</pre>

Ticket::Frontend::CustomerTicketMessage###QueueDefault

Description	Value
Description:	Defines the default queue for new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'QueueDefault'} = 'Postmaster';</pre>

Ticket::Frontend::CustomerTicketMessage###Service

Description	Value
Description:	Allows customers to set the ticket service in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'Service'} = '1';</pre>

Ticket::Frontend::CustomerTicketMessage###SLA

Description	Value
Description:	Allows customers to set the ticket SLA in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'SLA'} = '1'; </pre>

Ticket::Frontend::CustomerTicketMessage###StateDefault

Description	Value
Description:	Defines the default state of new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'StateDefault'} = 'new'; </pre>

Ticket::Frontend::CustomerTicketMessage###ArticleType

Description	Value
Description:	Defines the default type for article in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'ArticleType'} = 'webrequest'; </pre>

Ticket::Frontend::CustomerTicketMessage###SenderType

Description	Value
Description:	Sender type for new tickets from the customer interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'SenderType'} = 'customer';</pre>

Ticket::Frontend::CustomerTicketMessage###HistoryType

Description	Value
Description:	Defines the default history type in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'HistoryType'} = 'WebRequestCustomer';</pre>

Ticket::Frontend::CustomerTicketMessage###HistoryComment

Description	Value
Description:	Comment for new history entries in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'HistoryComment'} = '';</pre>

CustomerPanelSelectionType

Description	Value
Description:	Defines the receipt target of the tickets ("Queue" shows all queues, "SystemAddress" displays all system addresses) in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'CustomerPanelSelectionType'} = 'Queue';</code>

CustomerPanelSelectionString

Description	Value
Description:	Determines the strings that will be shown as receipt (To:) of the ticket in the customer interface. For Queue as CustomerPanelSelectionType, "<Queue>" shows the names of the queues, and for SystemAddress, "<Realname><<Email>>" shows the name and email of the receipt.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelSelectionString'} = '<Queue>';</code>

CustomerPanelOwnSelection

Description	Value
Description:	Determines which queues will be valid for ticket's recipients in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelOwnSelection'} = { 'Junk' => 'First Queue', 'Misc' => 'Second Queue' };</code>

CustomerPanel::NewTicketQueueSelectionModule

Description	Value
Description:	Module for To-selection in new ticket screen in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanel::NewTicketQueueSelectionModule'} =</code>

Description	Value
	'Kernel::Output::HTML::CustomerNewTicketQueueSelectionGeneric';

Ticket::Frontend::CustomerTicketMessage###DynamicField

Description	Value
Description:	Dynamic fields options shown in the ticket message screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required. NOTE. If you want to display these fields also in the ticket zoom of the customer interface, you have to enable them in CustomerTicketZoom###AttributesView.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self- >{'Ticket::Frontend::CustomerTicketMessage'}- >{'DynamicField'} = {};</pre>

Frontend::Customer::Ticket::ViewPrint

Ticket::Frontend::CustomerTicketPrint###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket print screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewPrint
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketPrint'}- >{'DynamicField'} = {};</pre>

Frontend::Customer::Ticket::ViewSearch

Ticket::CustomerTicketSearch::SearchLimit

Description	Value
Description:	Maximum number of tickets to be displayed in the result of a search in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::CustomerTicketSearch::SearchLimit'} = '5000';</pre>

Ticket::CustomerTicketSearch::SearchPageShown

Description	Value
Description:	Number of tickets to be displayed in each page of a search result in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::CustomerTicketSearch::SearchPageShown'} = '40';</pre>

Ticket::CustomerTicketSearch::SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in a ticket search of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::CustomerTicketSearch::SortBy::Default'} = 'Age';</pre>

Ticket::CustomerTicketSearch::Order::Default

Description	Value
Description:	Defines the default ticket order of a search result in the customer interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::CustomerTicketSearch::Order::Default'} = 'Down';</code>

Ticket::Frontend::CustomerTicketSearch###ExtendedSearchCondition

Description	Value
Description:	Allows extended search conditions in ticket search of the customer interface. With this feature you can search w. g. with this kind of conditions like "(key1&&key2)" or "(key1 key2)".
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'ExtendedSearchCondition'} = '1';</code>

Ticket::Frontend::CustomerTicketSearch###SearchCSVData

Description	Value
Description:	Data used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVData'} = ['TicketNumber', 'Age', 'Created', 'Closed', 'State', 'Priority', 'Lock', 'CustomerID', 'CustomerName', 'From', 'Subject'];</code>

Ticket::Frontend::CustomerTicketSearch###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket search screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'DynamicField'} = {};</pre>

Ticket::Frontend::CustomerTicketSearch###SearchOverviewDynamicField

Description	Value
Description:	Dynamic fields shown in the ticket search overview results screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchOverviewDynamicField'} = {};</pre>

Ticket::Frontend::CustomerTicketSearch###SearchCSVDynamicField

Description	Value
Description:	Dynamic Fields used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVDynamicField'} = {};</pre>

Frontend::Customer::Ticket::ViewZoom

Ticket::Frontend::CustomerTicketZoom###NextScreenAfterFollowUp

Description	Value
Description:	Determines the next screen after the follow up screen of a zoomed ticket in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'NextScreenAfterFollowUp'} = 'CustomerTicketOverview';</code>

Ticket::Frontend::CustomerTicketZoom###ArticleType

Description	Value
Description:	Defines the default type of the note in the ticket zoom screen of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'ArticleType'} = 'webrequest';</code>

Ticket::Frontend::CustomerTicketZoom###SenderType

Description	Value
Description:	Defines the default sender type for tickets in the ticket zoom screen of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'SenderType'} = 'customer';</code>

Ticket::Frontend::CustomerTicketZoom###HistoryType

Description	Value
Description:	Defines the history type for the ticket zoom action, which gets used for ticket history in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryType'} = 'FollowUp';</code>

Ticket::Frontend::CustomerTicketZoom###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket zoom action, which gets used for ticket history in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryComment'} = '';</pre>

Ticket::Frontend::CustomerTicketZoom###Priority

Description	Value
Description:	Allows customers to change the ticket priority in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'Priority'} = '1';</pre>

Ticket::Frontend::CustomerTicketZoom###PriorityDefault

Description	Value
Description:	Defines the default priority of follow up customer tickets in the ticket zoom screen in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'PriorityDefault'} = '3 normal';</pre>

Ticket::Frontend::CustomerTicketZoom###State

Description	Value
Description:	Allows choosing the next compose state for customer tickets in the customer interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'State'} = '1';</pre>

Ticket::Frontend::CustomerTicketZoom###StateDefault

Description	Value
Description:	Defines the default next state for a ticket after customer follow up in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateDefault'} = 'open';</pre>

Ticket::Frontend::CustomerTicketZoom###StateType

Description	Value
Description:	Defines the next possible states for customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateType'} = ['open', 'closed'];</pre>

Ticket::Frontend::CustomerTicketZoom###AttributesView

Description	Value
Description:	Shows the activated ticket attributes in the customer interface (0 = Disabled and 1 = Enabled).
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'AttributesView'} = { 'Owner' => '0', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '0', 'SLA' => '0', 'Service' => '0', 'State' => '1', 'Type' => '0' }; </pre>

Ticket::Frontend::CustomerTicketZoom###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket zoom screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'DynamicField'} = {}; </pre>

Frontend::Queue::Preferences

QueuePreferences###Comment2

Description	Value
Description:	Parameters of the example queue attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::Queue::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'QueuePreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the queue comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::QueuePreferencesGeneric', 'PrefKey' => 'Comment2', }; </pre>

Description	Value
	'Rows' => '5' };

Frontend::SLA::Preferences

SLAPreferences###Comment2

Description	Value
Description:	Parameters of the example SLA attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::SLA::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'SLAPreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the sla comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::SLAPreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' }; </pre>

Frontend::Service::Preferences

ServicePreferences###Comment2

Description	Value
Description:	Parameters of the example service attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::Service::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'ServicePreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the service comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::ServicePreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' }; </pre>

Anhang C. Danksagungen

Wie die meisten OpenSource-Projekte möchten auch wir vielen Leuten für Ihre Hilfe herzlich danken. Dies ist eine offene Liste in der wir sicher den Einen oder Aderen unabsichtlich vergessen haben, sorry dafür! Meldet Euch einfach, wenn Euer Name hier fehlt!

Folgende Personen haben das Projekt besonders unterstützt bzw. helfen immernoch aktiv mit:

- Robert Kehl, der den Win32-Installer für die 1.x Versionen alleine kreiert hat. Dank Dir, Robert!
- Torsten Werner, der sich aktiv um den Debian Installer kümmert und dafür sorgt, dass OTRS auch bequem unter Debian verfügbar ist. Vielen Dank, Torsten!
- Nils Jeppe (Mirror Hamburg, Germany), Bryan Fullerton (Mirro Toronto, Canada), Eberhard Moenkeberg (Mirror Göttingen, Deutschland), Timo Dreger (Mirror Düsseldorf, Deutschland) und Netmonic (Mirror Wien, Österreich), die unseren ftp-Server spiegeln. Vielen Dank an Euch, durch Eure Hilfe ist immer ein schneller Download von OTRS möglich!
- Anja Schneider, die kräftig bei der Übersetzung dieses Admin Handbuches mitgeholfen und die deutschen Texte korrigiert hat. Vielen Dank, Anja, für deine Geduld und deine Mithilfe!
- Aus den OTRS Mailinglisten schöpfen wir viele Ideen für Weiterentwicklungen, erhalten Patches oder Fixes für Fehler und die Benutzer bekommen sehr guten Support. Vielen Dank an alle auf den Listen für Eure aktive Mithilfe!

Weiterhin möchten wir den folgenden Personen herzlich danken:

- Martin Scherbaum
- Carsten Gross
- Harald Müller
- Stefan Schmidt
- Milisav Radmanic
- Uli Hecht
- Norman Walsh
- Heiko Baumann
- Atif Ghaffar
- Pablo Ruiz Garcia
- Dan Rau
- Christoph Kaulich
- Mark Jackson
- Diane Shieh
- Bernard Choppy

- Carl Bailey
- Phil Davis
- Edwin D. Vinas
- Lars Müller
- Vladimir Gerdjikov
- Fred van Dijk
- Sebastien Guilbaud
- Wiktor Wodecki
- Arnold Ligtoet
- Antti Kämäräinen
- Nicolas Goralski
- Gilberto Cezar de Almeida
- Jorge Becerra
- Eddie Urenda
- Stella Power
- Andreas Haase
- Reiner Keller
- Covert Jake
- Moshe Leibovitch
- Björn Jacke
- Remo Catelotti
- Alfons Obermeyer
- Michael Rech
- Danie Theron
- Richard Paradies
- Art Powell

Anhang D. GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document, that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters), and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and

straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats, suitable for input to text formatters. A copy made in an otherwise Transparent file format, whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include: plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include: PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location, containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously, at no charge, using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location, until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version, as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or

all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License, for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

Übersetzung

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections, in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.