



KIX Pro

KIX 18 Administration

Stand: 13.05.2024



Inhaltsverzeichnis

1	Hinweis zur Nutzung des KIX 18 Pro Handbuchs	9
1.1	Zum Verständnis.....	9
2	Installation von KIX Pro	10
2.1	KIX.Cloud Umgebung verwenden	10
2.2	KIX Pro On-Premises verwenden	10
2.3	Installation von Erweiterungen.....	12
3	Jobs - Erweiterte Funktionen	13
3.1	Job Typen in KIX Pro	13
3.2	Initial ausgelieferte Jobs in KIX Pro.....	15
3.3	Einen Job duplizieren.....	19
3.4	Anonymisierung	20
3.4.1	Den Job "Anonymisation" anpassen.....	20
3.5	Assetnummern beim Import ändern.....	22
3.6	LDAP/AD-Synchronisation	24
3.6.1	Synchronisations-Skript oder -Daemon verwenden.....	24
3.6.1.1	Automatische Deaktivierung nicht mehr verfügbarer Kontakte.....	25
3.6.2	Synchronisations-Job einrichten	26
3.6.2.1	Parameter	27
3.6.2.2	Konfigurationshinweise.....	35
3.7	Berichte automatisieren	36
3.7.1	Vorbedingungen.....	36
3.7.2	Beispiel Konfiguration des Jobs	36
3.7.3	Konfiguration der Aktionen	37
3.7.3.1	1. Aktion "Bericht erstellen"	37
3.7.3.2	2. Aktion "Text entnehmen"	37
3.7.3.3	3. Aktion "Macro ausführen"	38
4	System - Erweiterte Funktionen	41
4.1	Analysen	42
4.1.1	FE - Request Logs	43
4.1.1.1	Auswahl der Logfiles	43
4.1.1.2	Metriken.....	43
4.1.1.3	Charts	44



- 4.1.1.4 Tabelle 45
- 4.2 Dynamische Felder in KIX Pro 46
- 4.3 GUI Konfiguration..... 55
 - 4.3.1 Der Explorer 56
 - 4.3.2 Der JSON-Editor 57
- 4.4 Icons 61
- 4.5 Konfigurationstransfer 62
 - 4.5.1 Export von Konfigurationen..... 62
 - 4.5.2 Import von Konfigurationen 64
 - 4.5.3 Hinweise für den Im- und Export..... 67
- 4.6 Migration KIX 17 69
 - 4.6.1 Die Migration durchführen..... 70
 - 4.6.1.1 Schritt 1: PSK-Modus in KIX 17 aktivieren..... 70
 - 4.6.1.2 Schritt 2: Die Migration starten 72
 - 4.6.1.3 Schritt 3: Die Migration abschließen..... 73
 - 4.6.2 Hinweise zur Migration..... 75
- 4.7 Plugins 76
- 4.8 Single Sign-on..... 77
 - 4.8.1 SSO mit Kerberos..... 78
 - 4.8.1.1 Anbindung 78
 - 4.8.1.2 Die Keytab Datei 80
 - 4.8.1.3 Hinweise zu Base64 Codierung 81
 - 4.8.1.4 Verwendung / Client-seitige Voraussetzungen..... 81
 - 4.8.1.5 Häufig auftretende Fehlerursachen 82
 - 4.8.1.6 Verweise 82
 - 4.8.2 SSO mit OpenID Connect..... 83
 - 4.8.2.1 1. Einrichtung Identity Provider (IDP)..... 84
 - 4.8.2.2 2. Einrichtung OAuth2 Profil 84
 - 4.8.2.3 3. Einrichtung Authentifizierungs-Backend 87
- 5 Workflow 92**
 - 5.1 Aktionen..... 93
 - 5.1.1 Arten von Aktionen 93
 - 5.1.1.1 Artikelaktionen 94
 - 5.1.2 Initial ausgelieferte Aktionen in KIX Pro 95
 - 5.1.3 Aktionen anlegen und konfigurieren..... 102
 - 5.1.3.1 Aktion anlegen, bearbeiten, duplizieren und löschen 102
 - 5.1.3.2 Konfiguration einer Aktion..... 103

5.2	Rule Sets	121
5.2.1	Aufbau von Rulesets	122
5.2.2	Reihenfolge der Abarbeitung	124
5.2.2.1	Abarbeitung von Rulesets de-/aktivieren	125
5.2.3	Einen Regelsatz anlegen, bearbeiten oder löschen	126
5.2.4	Regelwerk	128
5.2.5	Übersicht der Befehle	129
5.2.5.1	Eigenschaften des Transaktionsobjekts	132
5.2.5.2	Einschränkung des Nutzungskontextes	133
5.2.5.3	Operatoren	133
5.2.5.4	Bedingungen (if)	135
5.2.5.5	Anweisungen (than)	138
5.2.6	Funktionen	149
5.2.6.1	DynamicFields.contains	149
5.3	Vorlagen	151
5.3.1	Eine Vorlage konfigurieren	153
5.3.1.1	Vorlageninformationen	153
5.3.1.2	Eingabefelder	158
5.3.2	Die Standard-Vorlage	173
5.3.2.1	Initiale Konfiguration der Standard-Vorlage	173
5.3.3	Kontakte anhand der URL auflösen	181
5.4	Vorlagengruppen	182
5.4.1	Duplizieren von Vorlagengruppen	185
6	Services und SLA	187
6.1	Serviceverträge	188
6.1.1	Aktualisierung des Servicebaums	188
6.1.2	Steuerung der Auswahl "Betroffene Services"	189
6.1.3	Betroffene Assets vs. Betroffene Services	191
6.1.4	Kritikalität	191
6.1.5	Serviceverträge im- und exportieren	191
6.1.6	Verwendung post-produktiver Services	192
6.2	Service-Level-Agreements (SLA)	193
6.2.1	SLA Kriterien und -Attribute	193
6.2.2	Schema einer SLA	196
6.2.3	Ein SLA anlegen oder bearbeiten	196
6.2.4	SLA an Tickets	199
6.2.5	SLA an Assets / Geräte-SLA	200
6.2.6	Benachrichtigungen für SLA	200



6.2.7	Verwendung von SLA in KIX Platzhaltern.....	201
6.2.8	Konfiguration der Erfüllungszeit	201
6.2.9	Kalender Konfiguration.....	202
6.3	Reaktionszeit bei Ticketerstellung setzen.....	203
6.3.1	Vorgehen:	203
6.3.2	Die Konfigurationen im Detail	203
6.3.2.1	1. Dynamisches Feld anlegen	203
6.3.2.2	2. Übersetzung anlegen.....	204
6.3.2.3	3. Vorlage erweitern.....	204
6.3.2.4	4. Job anlegen.....	205
7	Self Service Portal	206
7.1	Erreichbarkeit des Self Service Portals	207
7.2	Nutzereinstellungen für das SSP	209
7.2.1	Nutzer-Login einrichten	209
7.2.2	Rollen und Berechtigungen	210
7.3	Sichtbarkeiten im SSP steuern.....	211
7.3.1	Basis-Einstellungen für Sichtbarkeiten.....	212
7.3.2	Sichtbarkeit von Tickets und Artikeln	215
7.3.3	Sichtbarkeit von Follow-ups	216
7.3.4	Sichtbarkeit von Assets.....	217
7.3.5	Sichtbarkeit von FAQ-Artikeln	218
7.3.6	Sichtbarkeit von Dynamischen Feldern	218
7.3.7	Sichtbarkeit von News	220
7.4	Ticketeinstellungen.....	221
7.4.1	Fallback für Ticket-Titel	221
7.4.2	Bearbeitung von Ticket-Kopfattributen ermöglichen (Berechtigung).....	221
7.4.3	Voreinstellungen für Ticketfilter.....	223
7.5	Vorlagen, Aktionen, Rule Sets	225
7.5.1	Bereitstellung von Ticketaktionen	225
7.5.2	Bereitstellung von Artikelaktionen	227
7.5.3	Bereitstellung von Vorlagen	227
7.5.4	Workflow Rulesets	228
7.6	GUI Konfiguration des SSP.....	230
7.6.1	Dashboard Konfigurationen	232
7.6.1.1	Home Dashboard.....	232
7.6.1.2	Ticket-, Asset-, FAQ- Dashboard	232
7.6.2	Konfiguration der Detailansichten	233

7.6.2.1	Ticket Details.....	233
7.6.2.2	Asset Details	234
7.6.2.3	FAQ Details.....	235
7.6.3	Neues Ticket	235
7.6.4	Personal Preferences / Persönl. Einstellungen	236
7.6.4.1	Aktivieren/Deaktivieren der Passwortänderung	236
7.6.4.2	Aktivieren/Deaktivieren der Sprachauswahl	237
7.6.4.3	Konfiguration der Persönlichen Daten.....	237
7.6.5	Konfigurationsbeispiele.....	238
7.6.5.1	Object-information-card-widget	238
7.6.5.2	Tabellenwidget.....	239
7.6.5.3	Eigenes Widget implementieren	241
7.7	Layoutkonfiguration.....	243
7.7.1	Attribute in der Layoutkonfiguration.....	243
7.7.2	Organsiationspezifische Layouts	245
7.7.3	Beispiel Layout-Konfiguration	247
8	News	249
9	Zusätzliche Konfigurationsmöglichkeiten KIX Pro	251
9.1	Konfiguration der Teamansichtsmodi.....	252
9.1.1	Listenansicht.....	253
9.1.2	Kanban-Ansicht.....	254
9.1.3	Kalender-Ansicht.....	255
9.1.4	Kartenansicht.....	256
9.1.4.1	Aktualisierung von Geopositionsdaten	260
9.1.5	De-/aktivieren von Teamansichtsmodi.....	261
9.2	Zeiterfassung	263
9.2.1	Konfigurationsbeispiel.....	265
9.3	KIX Pro REST API	268
9.4	Kind-Tickets in den Ticketdetails auflisten	269
9.4.1	Zum Anzeigen von Kind-Tickets im Tab sind folgende Schritte nötig:.....	270
9.4.2	Dynamisches Feld "ChildTickets" in den Dialog "neues Ticket" integrieren.....	270
9.4.3	Dynamisches Feld "ChildTickets" in den Dialog "Ticket bearbeiten" integrieren	271
9.4.4	SysConfig-Schlüssel aktivieren.....	271
9.5	Objekt-Historie konfigurieren	273
9.6	Konfiguration des Standortexplorers	275
9.6.1	Voraussetzung	275
9.6.1.1	Assetklasse um Attribut "Standort" erweitern.....	276



- 9.6.2 Konfiguration..... 277
 - 9.6.2.1 Explorer..... 277
 - 9.6.2.2 Contentbereich..... 277
 - 9.6.2.3 Sidebar..... 282
- 10 Add-ons 284**
- 10.1 Connect 285
 - 10.1.1 Installation..... 286
 - 10.1.2 Gemeinsamer Funktionsumfang der Connect Add-ons 286
 - 10.1.3 Feldtyp "Data Source" 287
 - 10.1.3.1 Parameter..... 288
 - 10.1.4 Erweiterte Macro Actions 289
 - 10.1.4.1 XSL Transformation..... 289
 - 10.1.4.2 Get Object Data 301
 - 10.1.4.3 Get Item List From Data Source..... 302
 - 10.1.4.4 Get Item From Data Source..... 304
 - 10.1.5 Connect Baramundi 305
 - 10.1.5.1 Voraussetzungen..... 305
 - 10.1.5.2 Verwendung 306
 - 10.1.5.3 Anpassungen 309
 - 10.1.6 Connect Database 311
 - 10.1.6.1 Voraussetzungen..... 311
 - 10.1.6.2 Einrichtung 311
 - 10.1.6.3 Verwendung von Datenquellen 321
 - 10.1.6.4 Verwendung der KIX Datenbank 342
 - 10.1.7 Connect Opsi..... 343
 - 10.1.7.1 Voraussetzungen..... 343
 - 10.1.7.2 Verwendung 343
 - 10.1.7.3 opsi-Synchronisation aktivieren..... 344
 - 10.1.7.4 opsi-Synchronisation zurücksetzen oder ändern..... 345
 - 10.1.7.5 Anpassungen 346
 - 10.1.8 Connect Webservice..... 347
 - 10.1.8.1 KIX als Provider - Spezifische Endpunkte..... 347
 - 10.1.8.2 KIX als Requester - Verwendung von Webhooks..... 357
- 10.2 Add-on "ITIL Practices" 372
 - 10.2.1 Video..... 372
 - 10.2.2 Aktivieren/Deaktivieren von "ITIL Practices"..... 373
 - 10.2.2.1 Voraussetzung 373
 - 10.2.2.2 Installation/System-Update..... 373
 - 10.2.2.3 Das Add-on "ITIL Practices" aktivieren 373
 - 10.2.2.4 Das Add-on "ITIL Practices" zurücksetzen 375

10.2.3	Umsetzung der ITIL Practices mittels Prozessen.....	376
10.2.3.1	Ticketvorlagen.....	376
10.2.3.2	Ticketaktionen.....	377
10.2.3.3	Berechtigungen.....	380
10.2.3.4	Prozess 1: Serviceanfrage.....	381
10.2.3.5	Prozess 2: Störung.....	385
10.2.3.6	Prozess 3: Problem.....	389
10.2.3.7	Prozess 4: Änderungsantrag.....	392
10.2.3.8	Prozess 5: Notfallplan.....	396
10.2.4	Erweiterte Konfigurationen.....	398
10.2.4.1	Aktionen für ITIL Practices.....	399
10.2.4.2	Assets für ITIL Practices.....	401
10.2.4.3	Berichtsdefinitionen für ITIL Practices.....	402
10.2.4.4	Dynamische Felder für ITIL Practices.....	403
10.2.4.5	FAQ für ITIL Practices.....	404
10.2.4.6	Jobs für ITIL Practices.....	405
10.2.4.7	Stammdaten für ITIL Practices.....	406
10.2.4.8	Vorlagen für ITIL Practices.....	407
10.3	Wartungsplan.....	408
10.3.1	Voraussetzung.....	408
10.3.2	Grundlagen.....	409
10.3.2.1	Wartungs-Service.....	410
10.3.2.2	Wartungs-Asset.....	410
10.3.2.3	Wartungs-Plan.....	410
10.3.2.4	Wartungs-Aufgaben.....	411
10.3.2.5	Wartungs-Ticket.....	412
10.3.3	Administration.....	413
10.3.3.1	Installation/System-Update.....	413
10.3.3.2	Berechtigungsrollen.....	413
10.3.3.3	Ticketvorlagen.....	414
10.3.3.4	Update von Wartungsaufgaben.....	415
10.3.3.5	Funktionserweiterungen.....	415
11	Praxis.....	416
11.1	Periodischer Job "Lizenzverlängerung".....	417
11.1.1	Vorbereitung.....	417
11.1.2	Job konfigurieren.....	418
11.1.3	Job ausführen.....	419
11.2	Verwendung von Checklisten.....	421
11.2.1	Checklisten in KIX 18.....	421
11.2.2	Checklisten in der Praxis.....	424



11.2.3	Checklisten in KIX Pro	424
11.2.4	Konfiguration von Checklisten	425
11.2.5	Verweise	426
11.2.6	Konfiguration und Bereitstellung von Checklisten.....	427
11.2.6.1	Checkliste anlegen.....	427
11.2.6.2	Checkliste bereitstellen	433
11.2.6.3	Checklistenstatus anzeigen	440
11.2.6.4	Verweise	449
11.2.7	Datenstruktur von Checklisten.....	450
12	Haftungsausschluss KIX Pro	455
12.1	Haftung für Inhalte.....	455
12.2	Haftung für Links	455
12.3	Urheberrecht.....	456
13	Zweckbestimmung für die Verwendung von KIX Pro im medizinischen Bereich	457



1 Hinweis zur Nutzung des KIX 18 Pro Handbuchs

Im Zuge von Transparenz und effizienter Handhabung behandelt dieses Handbuch ausschließlich die Administration der Zusatzmodule und -funktionen von **KIX Pro der Version 18**. Die Beschreibung aller Grundfunktionen entnehmen Sie bitte dem Handbuch für KIX 18 Start.

Das Handbuch für die Agenten und KIX-Nutzer finden Sie unter <https://docs.kixdesk.com>. Hier finden Sie auch alle weiteren Dokumentationen zu KIX der Versionen 17 und 18.

Wir laden Sie herzlich ein, Teil unserer KIX Community zu werden und uns durch hilfreiche Hinweise bei der Weiterentwicklung von KIX und den Nutzerinformationen zu unterstützen.

 Diese Ausgabe des Handbuchs bezieht sich auf den Releasestand: s. Release Informationen

1.1 Zum Verständnis

In dieser Anleitung wird die nach der Grammatik männliche Form in einem neutralen Sinn verwendet. Es spricht immer alle männlichen, weiblichen und diversen Leser an. Auf Gendervarianten wie "AgentInnen" oder "Nutzer/innen" wird aus Gründen der Lesbarkeit und des Textverständnisses verzichtet. Alle Leserinnen und Diverse bitten wir um Verständnis für diese Vereinfachung im Text.



2 Installation von KIX Pro

KIX Pro ist ein Add-on zu KIX Start mit erweitertem Funktionsumfang. Sie können zwischen verschiedenen Arten der Nutzung von KIX 18 wählen:

- **KIX 18 Cloud:** Anwendung in der KIX.Cloud - ohne Installation
- **KIX 18 On Premises:** Lokale Installation auf dem eigenen Server

2.1 KIX.Cloud Umgebung verwenden

Für die Verwendung einer KIX.Cloud-Umgebung ist keine Installation erforderlich. KIX.Cloud eignet sich daher auch als Testumgebung.

Beantragen Sie unter www.kixdesk.com¹ die Bereitstellung von KIX.Cloud. Wir senden Ihnen per E-Mail den Link zu Ihrem KIX Portal sowie die Zugangsdaten für den Initialbenutzer (admin). Öffnen Sie den Link und loggen Sie sich mit den von uns erhaltenen Zugangsdaten ein.

Nutzen Sie dann den Setup Assistent, um ein neues Admin-Passwort sowie einen "Super User" festzulegen und die Grundkonfiguration von KIX vorzunehmen.

2.2 KIX Pro On-Premises verwenden

Um KIX Pro lokal zu verwenden, müssen Sie lediglich in der Datei `environment` die öffentliche Docker-Registry durch Ihre individuelle Repository-ID ersetzen. Die Repository-ID senden wir Ihnen per E-Mail zu, nachdem Sie KIX Pro abonniert haben.

⚠ Wichtig!

Die Ihnen zugesendete Repository-ID ist ausschließlich für die Nutzung durch Sie bzw. Ihre Organisation bestimmt. Geben Sie daher die Repository-ID nicht weiter und bewahren Sie sie sorgfältig und vor unbefugten Zugriff gesichert auf. Gelangen Unbefugte in Besitz Ihrer Repository-ID, könnten sie Zugang zu Ihrem System und zu Ihren Daten erhalten.

Um KIX Pro zu verwenden, gehen Sie wie folgt vor:

1. Installieren Sie zunächst KIX Start - sofern noch nicht vorhanden. Eine Anleitung dazu finden Sie im Start-Handbuch unter Installation.
2. Öffnen Sie die Datei `environment`
3. Setzen Sie eine Raute (#) An den Anfang der Zeile " `REGISTRY=docker-`
`registry.kixdesk.com/public` "

Damit kommentieren Sie den Verweis auf die öffentliche Registry aus, sodass diese Angaben beim Programmstart unberücksichtigt bleiben.

¹ <http://www.kixdesk.com>



- Ersetzen Sie in der übernächsten Zeile `<YOURREPOSITORYHERE>` durch Ihre individuelle Repository-ID und entfernen Sie die Raute (#) am Anfang der Zeile.
Durch das Entfernen der Raute findet diese Zeile beim Programmstart Berücksichtigung.
- Das Ergebnis sieht wie folgt aus:

```
Angabe der Repository-ID im "environment"
1 # -----
2 # basic configuration
3 # -----
4
5 # the docker registry to use
6 #REGISTRY=docker-registry.kixdesk.com/public
7
8 # the following applies for KIX Pro customers
9 REGISTRY=docker-registry.kixdesk.com/customers/
10 4d11xo22xxxxxxxxxxxxxxxxxxxxxxxxxxxxff29
11
12 # the image tag to use for all application images
13 IMAGE_TAG=stable
14 ...
```

- Führen Sie ein Update aus (s. auch Admin-Handbuch von KIX Start: Installation)
- Beim nächsten Start von KIX können Sie KIX Pro verwenden.
Nutzen Sie dann ggf. den Setup Assistent, um ein neues Admin-Passwort sowie einen "Super User" festzulegen und die Grundkonfiguration von KIX vorzunehmen.

Alternativ können Sie KIX Pro über die Konsole initialisieren. Rufen Sie dazu folgenden Befehl auf:

```
Konsolenkommando zum Initialisieren von KIX Pro

root@dockerhost:/opt/kix-on-premise/deploy/linux/# sed -i.orig 's/REGISTRY=docker-
registry.kixdesk.com/public/REGISTRY=docker-registry.kixdesk.com/
public\nREGISTRY=docker-registry.kixdesk.com/customers/<YOUR_REPOSITORY_ID_HERE>/g'
environment
```

Weiterführende Informationen zur Installation von KIX Pro finden Sie auch im [GitHub](#)²unter:

- Linux: <https://github.com/kix-service-software/kix-on-premise/blob/master/deploy/linux/README.md>
- Windows: <https://github.com/kix-service-software/kix-on-premise/blob/master/deploy/windows/README.md>

² <https://github.com/cape-it>



2.3 Installation von Erweiterungen

Die Zusatzmodule und Erweiterungen von KIX werden in der kundenspezifischen Registry zusammengefügt und stehen direkt zur Verfügung. In der Regel erfolgt dies einen Arbeitstag nachdem die Beauftragung eingegangen und bearbeitet wurde. Die aktualisierten Images werden wie bei einem Versionsupdate installiert:

```
user@DockerHost:/opt/kix-on-premise/deploy/linux# ./stop.sh  
user@DockerHost:/opt/kix-on-premise/deploy/linux# ./update.sh
```

Darüber hinaus sind keine weiteren allgemeinen Schritte erforderlich. Je nach Add-on werden neue Einstellungsmöglichkeiten oder Konfigurations-Assistenten im Admin-Bereich offeriert.

Die Bereitstellung von Add-ons in der KIX Cloud erfolgt über unser Support-Team.

3 Jobs - Erweiterte Funktionen

Menü	KIX > Automatisierung > Jobs
-------------	------------------------------

KIX Pro enthält eine Reihe vorkonfigurierter Jobs und bietet Ihnen die Möglichkeit, Jobs mit erweiterten Funktionen anzulegen. Zudem bietet KIX Pro die Möglichkeit, Jobs zu duplizieren (s. unten).

3.1 Job Typen in KIX Pro

Neben ticketbezogenen Jobs können Sie in KIX Pro auch assetbezogene Jobs und Synchronisations-Jobs anlegen. Die Festlegung dazu erfolgt im Menü *Automatisierung > Jobs* mit der Auswahl des Job-Typs.

Asset

Der Job-Typ "Asset" steht nur in KIX Pro zur Auswahl. Er bietet Ihnen die Möglichkeit, Jobs auf Assets anzulegen. Sie können diesen Job-Typ nutzen, um bspw. die von KIX vergebenen Asset-Nummern nach dem Import von Assets an den eigenen Nummernkreis anzupassen.

Kontakt

Der Job-Typ "Kontakt" wird auf Kontakte angewandt. Diesen können Sie bspw. nutzen, um ein neues Ticket zu erstellen, sobald ein bestimmtes Datum am Kontakt erreicht ist (Start eines Offboardingprozesses bei Ausscheiden eines Mitarbeiters).

Synchronisation

Mithilfe des Job-Typs "Synchronisation" können Sie sowohl die Nutzer- als auch die Kontaktdaten aus LDAP/AD-Verzeichnissen mit KIX synchronisieren. Hierbei holt KIX sich die Daten aus den angegebenen Verzeichnissen, legt sie in KIX ab und synchronisiert diese über einen event- oder zeitbasierten Job. Wie die Einrichtung des Jobs erfolgt, können Sie unter "[Synchronisation \(siehe Seite 24\)](#)" nachlesen.

Berichte

Mithilfe des Job-Typs "Berichte" können Sie automatisiert Berichte generieren und anschließend in einem neuen Ticket versenden. KIX Pro bietet die Möglichkeit, Berichte in verschiedenen Ausgabeformaten wie CSV, JSON, HTML, Excel (XLSX), AtomFeed, XML und PDF zu erstellen. Es ist damit möglich, diese Berichte in andere (externe) Systeme einzulesen und weiterzuverarbeiten. KIX Pro bietet zudem zusätzliche MacroActions, um den Inhalt eines automatisierten Berichts zu extrahieren und direkt im Nachrichtentext eines Tickets zu versenden. Ein Beispiel dazu finden Sie hier, im [Kapitel "Berichte automatisieren"](#). (siehe Seite 36) Die Grundfunktionen zum Erstellen von Berichtsdefinitionen und Berichten sowie zum automatisierten Versand eines Berichts als Ticketanhang finden Sie im Admin-Handbuch von KIX Start (Kapitel KIX Module).

Ticket



Mit dem Job-Typ "Ticket" können Sie Jobs auf Tickets und deren Artikel anlegen. Diese Standardfunktion ist im KIX Start Handbuch im Kapitel Jobs beschrieben.

3.2 Initial ausgelieferte Jobs in KIX Pro

Job	Beschreibung	Hinweise
Anonymisierung	Mit diesem Job können Sie Tickets ab einem vordefinierten Zeitraum anonymisieren lassen, bspw. um sie datenschutzkonform zu archivieren. Sie können festlegen, zu welchem Zeitpunkt oder Event die Ausführung der Anonymisierung erfolgen soll und welche Daten dabei anonymisiert werden. Dieser Job ist bereits für Sie vorbereitet, aber auf "ungültig" gesetzt. Möchten Sie den Job nutzen, setzen Sie ihn auf "gültig" und passen Sie die Konfiguration ggf. weiter an.	Siehe auch Kapitel " Anonymisierung (siehe Seite 20) "
Auto Set Planned Effort (Incident)	Setzt einen Standardwert als Vorgabe für die geplante Erfüllungszeit (Sollzeit). Gilt für Tickets des Typs "Vorfall". Initialer Wert: 30 Minuten	Der Job setzt die im Job definierte Sollzeit automatisch in das Dynamische Feld "PlannedEffort". Sie können die Konfiguration des Jobs ändern, indem Sie bspw.
Auto Set Planned Effort (Service Request)	Setzt einen Standardwert als Vorgabe für die geplante Erfüllungszeit (Sollzeit). Gilt für Tickets des Typs "Service Anfrage". Initialer Wert: 60 Minuten	<ul style="list-style-type: none"> • den Wert der Sollzeit ändern • weitere den Job auslösende Events festlegen • über Filter definieren, unter welchen Voraussetzungen der Job die Sollzeit setzen soll • u.a.m.

Job	Beschreibung	Hinweise
<p>Create FAQ Suggestion</p>	<p>Der Job erstellt aus einem Artikel heraus einen neuen FAQ Eintrag, sofern am Artikel das Dynamische Feld "CreateFAQSuggestion" auf "ja" gesetzt ist (= 1. Aktion).</p> <p>Mit der 2. Aktion wird der Wert des Dynamischen Feld "CreateFAQSuggestion" für eine erneute Verwendung zurückgesetzt (Leerwert).</p> <p>Info: Das Feld "CreateFAQSuggestion" steht nur in den Dialogen "Ticket bearbeiten" und "Ticket schließen" zur Verfügung wenn als Kanal "Notiz" oder "E-Mail" gewählt ist.</p>	<p>Sie können bei Bedarf die Vorgabekonfigurationen in der 1. Aktion ändern, um andere Werte in den FAQ Eintrag zusetzen oder den FAQ Eintrag einer anderen Kategorie zuzuordnen.</p> <p>Initial sind im Job u. a. folgende Parameter vorgegeben:</p> <ul style="list-style-type: none"> • Titel: Betreff des ersten Artikels • Kategorie: Misc • Ursache: Inhalt des ersten Artikels am Ticket • Lösung: Inhalt des den Job auslösenden Artikels • Dynamische Felder: Verwandte Tickets → ID des Tickets (stellt die Verknüpfung zwischen Ticket und FAQ her)
<p>KIX Field Agent - Mobile Processing Rejected</p>	<p>Setzt im Ticket den Bearbeiter und den Sperrstatus im Ticket, wenn in der Field Agent App das Ticket zurückgewiesen wird.</p>	<p>Sie können den zu setzenden Bearbeiter festlegen und weitere Macro-Aktionen hinzufügen.</p>
<p>Put affected assets back into operational state</p>	<p>Dieser Job setzt automatisch den Vorfallstatus "Operational" für alle betroffenen Assets eines bestehenden Tickets wenn:</p> <ul style="list-style-type: none"> • das Ticket vom Typ "Incident" geschlossen ist • oder der Solution-SLA erfüllt ("satisfied") ist • UND keine weiteren offenen/nicht gelösten Incident-Tickets zu diesem Asset existieren. 	<p>Ändern Sie die Konfiguration nicht! Daran sind systemrelevante Funktionen geknüpft.</p>

Job	Beschreibung	Hinweise
Put affected assets into incident state	<p>Dieser Job setzt automatisch den Vorfalstatus "Vorfall" für alle betroffenen Assets eines bestehenden Tickets wenn:</p> <ul style="list-style-type: none"> • ein Ticket vom Typ "Vorfall" erstellt wird • der Solution-SLA noch nicht erfüllt ist ("satisfied") • und eines oder mehrere Assets in "Affected Asset" eingetragen werden. 	<p>Ändern Sie die Konfiguration nicht! Daran sind systemrelevante Funktionen geknüpft.</p>
SLA First Response Time Fulfillment	<p>Dieser Job setzt einen Zeitstempel in ein Ticket, wenn ein Artikel erstellt wird, der im Self Service Portal sichtbar ist. Er definiert damit den Zeitpunkt der ersten Antwort für die SLA-Erfüllung. Sie können den Job nach Bedarf nachkonfigurieren.</p>	<p>Die Aktion "Erfüllungszeit setzen" sollte nicht entfernt werden, sonst wird der Zeitstempel nicht gesetzt.</p>
SLA Lösungszeit-Erfüllung	<p>Dieser Job setzt einen Zeitstempel in ein Ticket, wenn ein Ticket den Status "entfernt", "geschlossen" oder "zusammengefasst" erhält. Er definiert damit den Zeitpunkt der Lösung für die SLA-Erfüllung. Sie können den Job nach Bedarf nachkonfigurieren.</p>	<p>Die Aktion "Erfüllungszeit setzen" sollte nicht entfernt werden, sonst wird der Zeitstempel nicht gesetzt.</p>

Job	Beschreibung	Hinweise
TicketMerge	<p>Dieser Job wird für die initiale Aktion (siehe Seite 93) "Zusammenfassen" benötigt. Diese öffnet einen Dialog, um 2 Tickets zusammenzuführen. Beim Speichern der Aktion wird der Job ausgelöst.</p> <p>Dabei wird</p> <ul style="list-style-type: none"> • das Quellticket in den Status "merged" gesetzt • das Event "TicketMerge" am Quellticket ausgelöst • die in der Aktion konfigurierten Eigenschaften übermittelt <p>Wird in der Aktion der Kanal "E-Mail" oder "Notiz" gewählt, wird auch der Artikel ans Zielticket geschrieben.</p> <p>Besitzt ein Ticket den Status "merged", dann werden an diesem Ticket keine admin-konfigurierbaren Ticket- und Artikelaktionen angezeigt.</p> <p>Die initiale Jobkonfiguration sieht vor, dass die Ausgangstickets ans Zielticket angehängt werden. Dabei bleiben die Werte des Zieltickets bestehen und das Ausgangsticket verliert seine Werte (wie Priorität, Bearbeiter etc.).</p>	<p>Sie können die Aktion nachkonfigurieren und entscheiden, welche Attribute des Quelltickets mit dem Zielticket zusammengeführt werden:</p> <p>1. Aktion: initial "Ticket Merge" (darf nicht geändert werden)</p> <p>Zielticket: Die Ticketnummer des Zieltickets oder der Name des dynamischen Feldes, welche die ID des Zieltickets enthält (initial das zugehörige Dynamische Feld "MergeToTicket (siehe Seite 46) ").</p> <p>Ticket Attribute: Kommaseparierte Liste von Ticketattributen. Die Liste enthält die Attribute des Ausgangstickets, welche ans Zielticket geschrieben werden.(Achtung: Übersetzungen der Attributs-Namen sind nicht möglich)</p> <p>Dynamische Felder: Kommaseparierte Liste der Namen der dynamischen Felder, die ans Zielticket übernommen werden (Achtung: Übersetzungen der Namen sind nicht möglich).</p> <p>Werte für dynamische Felder erzeugen:</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Überschreibt die Werte der aufgeführten Dynamischen Felder im Zielticket mit den Werten des Quelltickets. <input type="checkbox"/> Der Wert eines dynamischen Feldes des Quelltickets wird am Zielticket nur gesetzt, wenn ein solcher Wert dort noch nicht vorhanden ist.



3.3 Einen Job duplizieren

KIX Pro bietet die Möglichkeit, Jobs zu duplizieren. Somit können Sie bestehende Jobs als Vorlage für neue Jobs nutzen, bspw. um verschiedene Aktionen bei gleichen Bedingungen ausführen zu lassen.

In der Detailansicht eines Jobs finden Sie die dazu die Aktion "Duplizieren". Nach Klick auf die Schaltfläche wird ein neuer Job auf Basis des aktuellen Jobs angelegt. Dazu wird ein Formular mit dem Namen "Kopie von [Name des duplizierten Jobs]" geöffnet. Sie können den Namen des Duplikats ändern.

Die Formularfelder des Duplikats sind mit den Werten des duplizierten Jobs gefüllt und können von Ihnen nach Bedarf geändert werden.

Die Macro Actions werden nicht referenziert, sondern neu erstellt. Somit können Sie auch die Macro Actions im Duplikat individuell ändern.

Beim Speichern des Duplikats wird ein eigenständiger neuer Job angelegt, ohne Verweise auf den ursprünglichen Job. Das heißt, eine Änderung im Duplikat ändert nichts am duplizierten Job.

3.4 Anonymisierung

Sie können ausgewählte Daten an Tickets anonymisieren lassen. Beispielsweise, um personenbezogene Daten bei der Archivierung von Tickets DSGVO-konform zu anonymisieren. Die Anonymisierung erfolgt über die zeit- oder eventgesteuerte Ausführung eines Jobs (Menü *Automatisierung* > *Jobs*). Mithilfe des initial ausgelieferten Jobs "Anonymisation" können Sie folgende Ticketdaten anonymisieren:

- Bearbeiter
- Verantwortlicher
- Kontakt
- Organisation
- Historien-Einträge
- Erstell- & Änderungsdatum für Tickets
- Erstell- & Änderungsdatum für Artikel
- an Artikeln hinterlegte E-Mail-Adressen.

Zusätzlich wird am Ticket ein Dynamisches Feld "Anonymise Ticket" ausgeliefert. Der Anonymisierungsjob ist initial so konfiguriert, dass alle Tickets bearbeitet werden, an welchem dieses Feld den Wert "ToDo" hat. Im Auslieferungszustand können Sie den Wert des Feldes nicht manuell setzen, da dieses in keinem Ticketdialog konfiguriert ist. Dies ist eine Sicherheitsmaßnahme, um ungewolltem Datenverlust vorzubeugen. Sie können einen neuen Job konfigurieren, der das Feld "Anonymise Ticket" mit den von Ihnen festgelegten Kriterien befüllt.

3.4.1 Den Job "Anonymisation" anpassen

Der Job "Anonymisation" wird inaktiv ausgeliefert und ist mit Beispiel-Werten gefüllt. Bearbeiten Sie die Konfiguration des Jobs, um diesen an Ihre Bedürfnisse anzupassen.

Im Schritt 1 (Job Informationen):

Legen Sie die Gültigkeit des Jobs fest. Ausgeführt werden nur die auf "gültig" gesetzten Jobs.

- Initiale Konfiguration: "ungültig" (Setzen Sie den Job bei Bedarf auf "gültig").

Im Schritt 2 (Ausführungsplan):

Legen Sie fest, zu welchem Zeitpunkt oder Ereignis die Anonymisierung erfolgen soll. Sie können die Konfiguration ändern oder ergänzen, indem Sie einen Zeitpunkt für die Ausführung des Jobs festlegen oder ein anderes Event wählen, welches den Job auslöst.

- Initiale Konfiguration:
 - Eventgesteuerte Ausführung
 - bei Änderung des Dynamischen Feldes "Anonymised Ticket" ("TicketDynamicFieldUpdate_AnonymiseTicket")

Im Schritt 3 (Filter):

Legen Sie fest, welche Tickets anonymisiert werden sollen. Sie können Filterkriterien hinzufügen oder ändern, um die Auswahl an Tickets einzuschränken.

- Initiale Konfiguration: Der Job wird ausgelöst, wenn das Dynamische Feld "Anonymise Ticket" den Wert "ToDo" hat.

Im Schritt 4 (Aktionen):

Legen Sie fest, welche Ticketdaten mit welchen Inhalte überschrieben werden sollen. Sie können die Konfiguration individuell anpassen und weitere [Aktionen](#) (siehe Seite 93) hinzufügen, indem Sie bspw. alle anonymisierten Tickets ins Team "Archiv" verschieben.

- Initiale Konfiguration:
 - Bearbeiter setzen: 1 (admin)
 - Verantwortlichen setzen: 1 (admin)
 - Kontakt setzen: 1 (admin@localhost)
 - Kunde setzen: 1 (MY_ORGA)
 - Historie löschen: Ersetze "OwnerUpdate", "ResponsibleUpdate", "CustomerUpdate", "TicketLinkAdd", "TicketLinkDelete", "SendAnswer", "SendAgentNotification", "SendCustomerNotification", "EmailAgent", "EmailCustomer", "FollowUp", "Forward", "LoopProtection", "Subscribe" & "Unsubscribe" durch: "Comment replaced by "Anonymisation" job."
 - Erstellt von und Geändert von setzen: 1 (admin)
 - Erstellt von und Geändert von für Artikel setzen:1 (admin)
 - Artikel E-Mail Attribute setzen:
 - Bcc: admin@localhost
 - Cc: admin@localhost
 - Von: admin@localhost
 - An: admin@localhost
 - Dynamisches Feld setzen: AnonymiseTicket: 2
Damit wird das dynamische Feld "Anonymised Ticket" auf den Wert "Done" gesetzt und somit das Ticket als "ist anonymisiert" gekennzeichnet. Das wird in der Oberfläche des Agentenportals angezeigt, sodass Sie auch nach allen anonymisierten Tickets suchen können.

3.5 Assetnummern beim Import ändern

Beim Import werden Assets unter einer von KIX generierten Assetnummer abgelegt. KIX Pro bietet die Möglichkeit, die Assetnummern beim Import durch eigene Identifikatoren zu ersetzen. Dies erfolgt mit einem eventbasierten Job des Typs "Asset".

So konfigurieren Sie einen Job zum Ändern der Assetnummern:

1. Navigieren Sie im Explorer zu *Automatisierung* > *Jobs*. Im Contentbereich wird eine Tabelle geöffnet, welche alle im System angelegten Jobs auflistet.
2. Klicken Sie in der Tabelle auf "Neuer Job". Es wird ein Formular-Dialog geöffnet, in dem Sie schrittweise den Synchronisationsjob anlegen können. Nutzen Sie die kleinen blauen Pfeil- oder Punktschaltflächen, um zum nächsten Schritt zu gelangen oder um zwischen den Schritten zu wechseln.

- **Job Informationen** (Schritt 1):

- Wählen Sie unter "JobTyp" die Option "Asset"
- Vergeben Sie dem Job einen aussagekräftigen Namen.
- Beschreiben Sie den Job im Feld "Kommentar" (optional)
- Setzen Sie den Job auf "gültig".

- **Ausführungsplan** (Schritt 2):

- Wählen Sie unter "Eventgesteuerte Ausführung" das Event "VersionCreate".
Damit wird festgelegt, dass der Job bei jeder Versionsänderung eines Assets, die Nummer ersetzt.

- **Filter** (Schritt 3):

- Setzen Sie Filter, um festzulegen, unter welchen Voraussetzungen der Job ausgeführt wird (optional). Der Job wird nur auf die Assets angewandt, auf die alle Filterkriterien zutreffen.
Werden keine Filter gesetzt, wird der Job für alle importierten Assets ausgeführt.
Es stehen die Asset-Attribute der gewählten Asset-Klasse zur Auswahl.

- **Aktionen** (Schritt 4):

- Wählen Sie als 1. Aktion "Asset Nummer setzen".
- Tragen Sie unter "Asset Nummer" den zu setzenden Identifikator ein. Sie können Buchstaben und Ziffern sowie deren Kombination verwenden.
Platzhalter sind ebenfalls möglich:



Platzhalter	Beschreibung
<KIX_ASSET_CurrentVersion_[AttributeName]>	Referenz auf Attribute aus der Asset Version (z.B. "Name")
<KIX_ASSET_CurrentVersion_[Data_AttributeKey]>	Referenz auf Attribute aus der Asset Klassendefinition
<KIX_ASSET_CurrentVersion_Data_[KEY]>	für XML-Daten (klassenspezifisch)
<KIX_ASSET_CurrentVersion_Name>:: <kix_asset_currentversion_data_roomlabel>::<KIX_ASSET_CurrentVersion_Data_Floor></kix_asset_currentversion_data_roomlabel>	Bespiel-Eintrag, wenn Asset zu Klasse "Raum" gehört

 Hinweise

Ist der zu setzende Wert ein CMDB-weit eindeutiger Indentifikator (Assetnummer), so wird die KIX-Assetnummer durch den Wert ersetzt. Ist eine Assetnummer nicht eindeutig oder kein Wert an betreffendem Attribut gesetzt, wird die bestehende Assetnummer nicht ersetzt und eine Fehlermeldung wird im Log eingetragen (Level "ERROR").

3.6 LDAP/AD-Synchronisation

Sie können die Kontakt- und

Inhalte auf dieser Seite:

- [Synchronisations-Skript oder -Daemon verwenden](#) (siehe Seite 24)
 - [Automatische Deaktivierung nicht mehr verfügbarer Kontakte](#) (siehe Seite 25)
- [Synchronisations-Job einrichten](#) (siehe Seite 26)
 - [Parameter](#) (siehe Seite 27)
 - [Konfigurationshinweise](#) (siehe Seite 35)

Nutzerdaten in KIX Pro mit Daten aus dem LDAP/Active Directory befüllen. Diese Daten dienen der Verwendung als

- Kontakte im Ticket- und Asset-Bereich
- Kundennutzer ("User" mit "IsCustomer")
- Agentennutzer ("User" mit "IsAgent").

Je nach verfügbarer LDAP-Quelle kann die Datensynchronisation auf verschiedenen Wegen erfolgen:

- Synchronisations-Skript bzw. -Daemon via Konsole-Kommando und SysConfig-Schlüssel
- automatisiert über event- oder zeitbasierte Synchronisations-Jobs

Voraussetzung: Mindestens ein LDAP-basierter Verzeichnisdienst, welcher vom Backend des KIX erreichbar ist.

Achtung

Änderungen an den Kontaktdaten müssen dann ausschließlich im LDAP/AD erfolgen. In KIX vorgenommene Änderungen werden mit der Synchronisation überschrieben.

Informationen zur Anbindung von LDAP/AD-Hosts finden Sie auch im KIX Start Handbuch.

3.6.1 Synchronisations-Skript oder -Daemon verwenden

Konsole-Kommando	Console::Command::Maint::Auth::Sync::Synchronize
SysConfig-Schlüssel	Daemon::SchedulerCronTaskManager::Task###AuthSynchronize

Sie können ein Skript bzw. Daemon zur Vereinfachung der AD-Anbindung verwenden, wenn alle Kontakte aus dem LDAP auch als Nutzer in KIX vorhanden sein sollen. Das Einrichten eines LDAP2Contact-Jobs kann somit entfallen.

Das Konsole-Kommando "`Console::Command::Maint::Auth::Sync::Synchronize`" startet ein Synchronisations-Script, welches die im KIX verfügbaren Kontakte ausschließlich aus LDAP-Quellen bezieht und diese gleichzeitig als Nutzer im System zur Verfügung stellt. Bei Ausführung des Kommandos werden alle konfigurierten Auth.-Sync.-Backends abgefragt. Die daraus ermittelten Nutzer und Kontakte werden in KIX angelegt und ihren Berechtigungskontexten sowie -rollen zugeordnet (s. Authentifizierung/Autorisierung und Anbindung Active Directory). Die Konsole finden Sie im Menü *System > Konsole*.

Soll der Abgleich periodisch automatisiert erfolgen, können Sie den SysConfig-Schlüssel:

"`Daemon::SchedulerCronTaskManager::Task###AuthSynchronize`" aktivieren (Menü *System > SysConfig*).

Bitte beachten Sie die Zeitplanung, mit der die Ausführung erfolgt.

Info

Wird der SysConfig-Schlüssel "`Daemon::SchedulerCronTaskManager::Task###AuthSynchronize`" gültig gesetzt, reicht ein Reload der Frontend Konfiguration nicht aus. Es muss der Daemon neu gestartet werden. Nutzen Sie dafür das Konsole-Kommando:

"`Console::command::admin::daemon::Reload`".

3.6.1.1 Automatische Deaktivierung nicht mehr verfügbarer Kontakte

Sie können Script bzw. -Daemon auch für das automatische Deaktivieren von Nutzern und Kontakten verwenden, die nicht mehr in den LDAP-Quellen vorhandenen sind. Dies kann bspw. bei unvollständigen Offboarding-Prozessen vorkommen, in welchen angrenzende Tools nicht informiert werden. Da diese Einträge in KIX nicht mehr aktualisiert werden, findet auch keine Deaktivierung statt. Die Deaktivierung kann manuell erfolgen. Alternativ kann das Kommando

"`Console::Command::Maint::Auth::Sync::Synchronize`" mit der Option "`--invalidate-unsynced`" ausgeführt werden. Das Skript deaktiviert alle Nutzer und Kontakte, die nicht mehr aus einer der Auth.-Sync-Quellen bezogen werden können. Damit Systemzugänge oder manuell angelegte Nutzer diesem Verhalten nicht unterliegen, können ausgewählte Login-Kenner ausgeschlossen werden. Diese Login-Kenner werden im SysConfig-Schlüssel "`Maint::Auth::Sync::Synchronize::Skip`" mittels regulärer Ausdrücke definiert.

Entfernen Sie keinesfalls die initialen Einträge!

3.6.2 Synchronisations-Job einrichten

Sie können spezifische LDAP-Quellen direkt anbinden, um die darin enthaltenen Kontaktdaten periodisch zu synchronisieren ohne auf die Authentifizierungssynchronisation zurückzugreifen. Richten Sie dazu einen Synchronisations-Job wie nachfolgend beschrieben ein.

Pro Job können mehrere LDAP/AD-Hosts in unterschiedlichen Formaten angegeben werden. Zudem können auch Verzeichnisdienste abgefragt werden, die eine Größenbeschränkung der Ergebnismenge verwenden ("max size exceeded"). Durch Angabe der Seitengröße (Request-Page-Size) kann eine blockweise Verarbeitung erfolgen.

Den Synchronisations-Job einrichten:

1. Navigieren Sie im Explorer zu *Automatisierung* > *Jobs*. Im Contentbereich wird eine Tabelle geöffnet, welche alle im System angelegten Jobs auflistet.
2. Klicken Sie in der Tabelle auf "Neuer Job". Es wird ein Formular-Dialog geöffnet, in dem Sie schrittweise den Synchronisationsjob anlegen können. Nutzen Sie die kleinen blauen Pfeil- oder Punkschaltflächen, um zum nächsten Schritt zu gelangen oder um zwischen den Schritten zu wechseln.
 - **Schritt 1** (Job Information):
 - Wählen Sie unter "JobTyp" die Option "Synchronisation".
 - Vergeben Sie dem Job einen aussagekräftigen Namen.
 - Beschreiben Sie den Job im Feld "Kommentar" (optional)
 - Setzen Sie den Job auf "gültig".
 - **Schritt 2** (Ausführungsplan):
 - Wählen Sie aus, wann die Synchronisation erfolgen soll (Wochentage und Uhrzeit).
 - **Schritt 3** (Aktionen):
 - Wählen Sie die Macro-Action "LDAP zu Kontakt" aus. Es werden weitere Eingabefelder angezeigt.
 - tragen Sie die Verbindungsdaten zum LDAP/AD-Server ein sowie die Attribute, welche mit dem LDAP/AD-Server synchronisiert werden sollen (s. Tabelle unten).
 - Sie können optional weitere Hosts angeben.
 - Klicken Sie auf . Eine 2. Action wird geöffnet.
 - Wählen Sie erneut "LDAP zu Kontakt" aus und tragen Sie auch hier die Verbindungsdaten und Attribute ein.
 - Zur besseren Übersicht können Sie die Eingabefelder der Aktionen ein- und ausklappen. Klicken Sie dazu auf die kleine Pfeilschaltfläche neben dem Plus-Symbol.  
 - Speichern Sie den Job mit Klick auf "Speichern". Die Synchronisation wird zum angegebenen Zeitpunkt ausgeführt.

ⓘ Hinweis

Wird ein LDAP zu Kontakt Synchronisations-Job ausgeführt und existiert der Nutzer bereits, dann wird der gefundene Nutzer aktualisiert und der Job fährt mit dem nächsten Nutzereintrag fort.

3.6.2.1 Parameter

Parameter	Bedeutung	Beispiel
Überspringen	Ist hier das Häkchen gesetzt, wird die entsprechende Aktion nicht ausgeführt. Damit können Sie einzelne Aktionen von der Synchronisation (zeitweise) ausschließen, ohne den Job neu zu konfigurieren. Sind mehrere Aktionen im Job konfiguriert (z. B. zwei verschiedene LDAP-Server), laufen die anderen Aktionen normal weiter.	
Host	FQDN welches der LDAP-Server für die Synchronisation nutzt oder komma-getrennte Liste wenn mehrere Hosts angesprochen werden sollen	ldap.example.org ldap1.example.org,ldap2.example.org
BaseDN	Einstiegspunkt in die Verzeichnisstruktur	dc=meinUnternehmen,dc=de

Parameter	Bedeutung	Beispiel
Contact / User Sync Map	<p>JSON-String, der das Mapping von KIX-Attributen zu AD-/LDAP-Attributen beinhaltet. Das Mapping muss folgende Form haben:</p> <pre data-bbox="359 495 1145 678"> { "KIXAttributName" : "LDAPAttributName", "KIXAttributName" : "LDAPAttributName" } </pre> <p>Alle im Mapping angegebenen Attribute müssen existieren. Anderenfalls kann keine korrekte Zuordnung erfolgen. Achten Sie auch auf korrekte Schreibweise.</p> <p>Der Kontakteintrag wird auf Basis der aus dem LDAP/AD ermittelten oder fixierten Werte erstellt bzw. aktualisiert. Die Angabe folgender Attribute ist dazu mindestens erforderlich:</p> <ul data-bbox="375 958 529 1106" style="list-style-type: none"> • Email • Firstname • Lastname • UserLogin <p>Achtung! Sind diese Angaben im LDAP-/AD-Eintrag nicht gesetzt (leer), kann kein Kontakteintrag erzeugt oder aktualisiert werden.</p> <p>Konkrete Werte werden mit vorangestelltem "SET:" angegeben (z. B.: "Email1": "SET:mail@example.com").</p> <p>Für das Zielattribut "Email", sowie für "Email1" bis "Email5" werden die einzelnen Array-Werte automatisch aufgeteilt, wenn im LDAP-Quellattribut ein Array-Wert vorliegt. Eine gesonderte Konfiguration ist nicht erforderlich.</p> <p>Jedes KIX-Attribut kann aus <i>n</i> LDAP-Attributen und Separator-Zeichenketten zusammengesetzt werden. Somit können Array-Attribute im LDAP als konkatenierte Zeichenkette in ein einzelnes Zielattribut geschrieben werden. Voraussetzung: Das Zielattribut muss den gelieferten Datentyp und die Definition des Dynamischen Feldes unterstützen.</p> <p>Die Array-Werte werden als Einzelwert mit Separator-Symbol konkateniert (analog "CONCAT" → s. unten).</p> <p>Hinweis: das Attribut "mail" unterstützt maximal 6 Werte im Array.</p>	<pre data-bbox="1177 383 1428 1921"> { "Email": "mail", "Email1": "SET:mail@example.com", "Email3": "mail", "Email5": "automatedmail", "Title": "title", "Firstname": "givenname", "Lastname": "sn", "Street": "streetAddress", "City": "l", "Zip": "postalCode", "Phone": "telephoneNumber", "Mobile": "mobile", "Fax": "facsimileTelephone Number", "UserLogin": "sAMAccountName" , "IsAgent": "SET:1", "IsCustomer": "SET:0", "PrimaryOrganisatio nID": "department", "OrganisationIDs": ["department", "SET:123"], "DynamicField_XYZ" : "SET:ActiveDirector y1", </pre>

Parameter	Bedeutung	Beispiel
	<ul style="list-style-type: none"> • Direktive: "ARRAYJOIN[<SeparatorString>]: {<ADAttributeName>}" <ul style="list-style-type: none"> • ARRAYJOIN : Befehl zum Erzeugen einer konkatenierten Zeichenkette (Array) • SeparatorString : Angabe des Trennzeichens (Komma, Semikolon, Bindestrich o.a. inkl. eventuellem Leerzeichen). Angabe in eckigen Klammern. • ADAttributeName : Name des LDAP/AD-Attributs, welches konkateniert werden soll. Das Attribut wird in geschweiften Klammern angegeben. • Beispiel: "objectClass" ist ein Attribut, welches im LDAP/AD mehrfach vorkommt. <ul style="list-style-type: none"> • Die Notation: "DynamicField_Source": "ARRAYJOIN[,]:{objectClass}" erzeugt aus allen LDAP/AD-Attributen mit der Bezeichnung "objectClass" ein Array und schreibt dieses als kommaseparierte Liste in das Dynamische Feld "DynamicField_Source". <p>Die Zuordnung zu einer Primärorganisation wird mittels " PrimaryOrganisationID " definiert. Dabei kann neben einem LDAP-/AD-Attribut auch ein fest gesetzter Wert konfiguriert werden (" PrimaryOrganisationID": "SET:123"). "123" wird dabei als ID einer in KIX bekannten Organisation interpretiert. Wird jedoch ein LDAP-/AD-Attribut angegeben ("PrimaryOrganisationID": "department"), wird der Wert des Attributs als Kundennummer interpretiert und per Lookup zu einer KIX-internen Organisations-ID ausgewertet. Wenn nichts gefunden wurde, dann wird der im LDAP/AD-Attribut enthaltene Wert als Organisations-ID interpretiert. Gibt es auch dazu keine Organisation, erfolgt der Fallback auf Organisations-ID 1.</p>	<pre>"DynamicField_ZIPCity": "CONCAT: {postalCode}-{}-DE" }</pre>

Parameter	Bedeutung	Beispiel
	<p>Die Zuordnung zu n weiteren Organisationen wird mittels " <code>OrganisationIDs</code> " definiert. Dazu wird ein Array von LDAP-/AD-Attributnamen oder fest zu setzenden Org.-ID-Zuordnungen angegeben. Dadurch können n LDAP-Attribute oder SET-Werte konfiguriert werden. Das Lookup-Verhalten ist identisch zu " <code>PrimaryOrganisationID</code> ". Bei jedem Abgleich werden die am Kontakt hinterlegten Angaben vollständig ersetzt. Wird die Angabe "OrganisationIDs" im Mapping nicht gepflegt, wird der in <code>PrimaryOrganisationID</code> angegebene Wert automatisch an ggf. schon bestehende Einträge in der Liste der Organisationen angefügt. Es findet dann keine vollständige Ersetzung statt.</p> <p>Soll ein weiterer Nutzer des Systems via Agenten- oder Self Services Portal aktiviert werden, müssen die Parameter " <code>IsAgent</code> " bzw. " <code>IsCustomer</code> " auf "1" gesetzt werden, dazu kann eine fixierte Wertzuweisung eingetragen werden (<code>"IsAgent": "SET:1"</code> bzw. <code>"IsCustomer": "SET:0"</code>).</p> <p>Die weiteren benötigten Berechtigungsrollen sind in der Authentifizierungs- und Autorisierungskonfiguration zu hinterlegen.</p> <p>Dynamische Felder können mit Attributwerten aus dem LDAP/AD belegt werden. Somit können bspw. Dynamische Felder der Objekttypen Kontakt oder Organisation verwendet werden, um</p> <ul style="list-style-type: none"> • an einem Kontakteintrag zu hinterlegen, aus welchem AD der Kontakt stammt • die Bezeichner für Organisations- und Kontaktdaten anderer Systeme auch in KIX bereitzustellen. <p>KIX stellt dafür die Dynamischen Felder "Type" und "Source" (KIX Pro)" bereit.</p>	

Parameter	Bedeutung	Beispiel
	<p>Konkatenation (Verkettung) von LDAP-Attributen: Jedes KIX-Attribut kann mittels " CONCAT " aus <i>n</i> LDAP-Attributen und Separator-Zeichenketten zusammengesetzt werden, bspw. um mehrere LDAP-Attribute in einem KIX-Kontaktattribut zusammenzufassen. Anstelle der direkten Angabe eines LDAP-Attributes werden diese in geschweiften Klammern benannt (z. B. "PrimaryOrganisationID": "CONCAT:{givenName} von der {company}-{department}").</p> <p>Alle Zeichen die nicht innerhalb von geschweiften Klammern enthalten sind, werden direkt in das Kontaktattribut überführt. So wird bspw. " CONCAT:{postalCode}-{l}-DE " zu "09113-Chemnitz-DE".</p>	
UID	<p>Name des LDAP-Attributes, welches zur eindeutigen Identifikation eines LDAP-Eintrages verwendet wird (Nutzer-Login). Sollte kein <i>UserLogin</i> in der Sync Map angegeben werden, wird auf Basis der <i>UID</i> in der KIXDB ein Nutzerlogin erstellt und auf invalid gesetzt. <i>UID</i> und <i>UserLogin</i> können ohne Probleme auf das selbe Attribut zeigen.</p> <p>Tipp: Durch Setzen des Attributs " AuthAttr " im SysConfig-Schlüssel "Authentication###000-Default" können Sie einen alternativen Kenner für das UserLogin setzen. Ist dieser bspw. auf "mail" gesetzt, kann sich ein Nutzer mit seiner im System hinterlegten Mailadresse anmelden. Werden zudem die Nutzerdaten mit dem LDAP per Job synchronisiert, kann der Nutzer sich immer mit seiner aktuellen Mailadresse anmelden, auch wenn diese sich z. B. aufgrund von Namensänderung geändert hat. Sollte die Anmeldung via Mailadresse fehlschlagen, wird die UID/das im System hinterlegte Nutzer Login als Fallback für die Authentifizierung verwendet.</p> <p>Bitte beachten Sie: Die Aktualisierung eines Kontaktattributs "Email" ist nur dann möglich, wenn in KIX ein Nutzereintrag vorhanden ist. Anderenfalls wird bei Änderung einer Email-Adresse im AD/LDAP ein neuer Kontakt angelegt.</p>	UID sAMAccountName

Parameter	Bedeutung	Beispiel
Parameter	<p>JSON-String für LDAP-Verbindungsparameter</p> <p>Default:</p> <pre>{ "port": "389", "version": "3", "timeout": "120", "async": "0" }</pre>	<pre>{ "port": "389", "version": "3", "timeout": "60", "async": "0" }</pre>
Suchnutzer DN	User DN des Nutzers mit dem sich KIX mit dem LDAP Server verbindet, um die Suche durchzuführen	cn=kix,cn=user,dc=meinUnternehmen,dc=de
Suchnutzer Password	Passwort des Nutzers mit dem sich KIX mit dem LDAP Server verbindet, um die Suche durchzuführen	*****
Beschränkung auf GroupDN	Erlaubt die Einschränkung der abzufragenden Einträge auf Objekte, die in der angegebenen Gruppe enthalten sind. Dies kann erforderlich sein, wenn die Gruppenmitgliedschaft nicht über einen direkten Filter auf dem Kontakteintrag eingeschränkt werden kann.	
Access Attribute	Definiert das LDAP-Attribut, das zum Überprüfen der Gruppenmitgliedschaft verwendet wird. Verwendet <code>memberUid</code> , wenn nichts angegeben ist.	member memberUid ...
User Attribute	Legt fest, ob der Distinguished Name des Benutzers ("DN") oder das im Parameter "UID" definierte LDAP-/AD-Attribut zur Überprüfung der Gruppenmitgliedschaft verwendet wird.	DN UID

Parameter	Bedeutung	Beispiel
Immer filtern	<p>Permanenter LDAP Filter, der für alle Anfragen angewandt wird.</p> <p>Beispiel: alle nicht-deaktivierten Nutzerkonten: <code>(&(objectClass=user)(!(userAccountControl:1.2.840.113556.1.4.803:=2)))</code></p> <p>Beispiel: alle Nutzereinträge bei denen "departmentNumber" und "mail" gesetzt ist: <code>(&(objectClass=user)(departmentNumber=*)(mail=*))</code></p> <p>Sie können auch Gruppenfilter eingetragen. Weitere Beispiele finden Sie unter: http://www.selfadsi.de/ldap-filter.htm</p>	<code>(&(mail=*)(sAMAccountName=*))</code>
Zeichensatz	<p>Zeichensatz, in den die LDAP-Suchergebnisse konvertiert werden. Default: utf-8</p>	utf-8
Seitengröße	<p>Sie können bei Bedarf die Gesamtanzahl der zu synchronisierenden Daten in mehrere Datenpakete zu je "n" Datensätzen aufsplitten. Bspw. wenn Sie vom Server die Nachricht erhalten, dass der Antwortdatensatz zu groß ist ("max size exceeded"). Tragen Sie hier die Anzahl der Datensätze ein, die pro Anfrage (Request) übertragen werden. Die Verarbeitung erfolgt dann blockweise in Paketen mit bspw. 20 Datensätzen.</p>	20
Debug	<p>Debug-Ausgaben de-/aktivieren.</p> <p>Ist der Parameter aktiv, werden wesentliche Schritte des LDAP-/AD-Datenabgleichs im <u>Job-Log</u> (Job Historie) vermerkt.</p> <p>Hinweise:</p> <ul style="list-style-type: none"> • Diese Option sollte nur vorübergehend aktiviert werden, da sonst die Historieneinträge sehr umfangreich werden und entspr. Speicherbedarf erzeugen. • Der SysConfig-Schlüssel "<i>Automation::MinimumLogLevel</i>" muss auf den Wert "<code>Debug</code>" gesetzt sein. Anderenfalls werden nur die von der Macro Action gelieferten Fehlermeldungen erfasst, alle anderen Debug-Informationen jedoch nicht. 	yes no

Parameter	Bedeutung	Beispiel
Dry Run	<p>Trockenlauf (Testlauf) de-/aktivieren.</p> <p>Ist der Parameter aktiviert, erfolgt kein tatsächlicher Datenabgleich. Die aus der LDAP-/AD-Quelle ermittelten Daten werden jedoch im Job-Log vermerkt, als würde ein Update stattfinden. Der Datenabgleich kann somit im Vorfeld zu Testzwecken simuliert werden.</p> <p>Hinweis: Diese Option sollte wie Debug nur vorübergehend aktiviert werden, da sonst die Historieneinträge sehr umfangreich werden und entspr. Speicherbedarf erzeugen. Es wird weiterhin empfohlen diese Funktion in Verbindung mit AlwaysFilter zu verwenden um spezifische oder eine geringe Anzahl von LDAP-Einträgen zu prüfen.</p>	

i Info

- Ist im Mapping das User-Attribut "Login" enthalten, wird auch ein User synchronisiert.
- Ein vorhandener Kontakt wird beim Ausführen des Jobs, basierend auf dem Mapping, aktualisiert.
- Ein nicht vorhandener Kontakt wird beim Ausführen des Jobs neu erstellt.

⚠ Hinweise

- Die Nutzeranmeldung ist von der Synchronisation nicht direkt betroffen, jedoch die Funktion des "AuthSync".
- Beim automatischen Import ist auf Datenbeschränkungen des AD zu achten (blockweises Einlesen).
- Single-Point-of-Trust ist das externe System.

3.6.2.2 Konfigurationshinweise

Sie können für bestimmte KIX-Attribute an einem Kontakt/Nutzer einen voreingestellten Wert anstelle eines LDAP-Attributes angeben. Dies erfolgt über das Keyword "SET:", welches vor dem zu setzenden Wert eingebunden wird. Das Keyword ist caseinsensitive.

Beispiel: SyncMap mit fixierten Parametern

```
{  
  "Email": "mail",  
  "Title": "title",  
  "Firstname": "givenname",  
  "Lastname": "sn",  
  "Street": "streetAddress",  
  "City": "l",  
  "Zip": "postalCode",  
  "Phone": "telephoneNumber",  
  "Mobile": "mobile",  
  "Fax": "facsimileTelephoneNumber",  
  "UserLogin": "sAMAccountName",  
  "IsAgent": "SET:1",  
  "IsCustomer": "SET:0",  
  "PrimaryOrganisationID": "department"  
}
```

3.7 Berichte automatisieren

KIX Pro ermöglicht, Berichte in weiteren Ausgabeformaten zu erstellen (Atom Feed, Excel, HTML, JSON, PDF, XML). Durch Extraktion der Berichtsinhalte können diese direkt in den Nachrichtentext eines Ticket eingefügt und/oder als Artikelanhang gesendet werden.

Im nachfolgenden Beispiel wird ein Job vom Typ "Reporting" angelegt, welcher jeden Montagmorgen einen HTML-Bericht erstellt und den Berichtsinhalt im Nachrichtentext des Tickets versendet. Der Job hat folgenden Aufgaben (in erforderlicher Reihenfolge):

1. Jeden Montag um 6:00 Uhr einen Bericht im HTML-Format zu erstellen
2. Die HTML-Tabelle aus dem Bericht extrahieren
3. Ein neues Ticket versenden, welches die HTML-Tabelle des Berichts im Nachrichtentext enthält.

Die Möglichkeit, einen Bericht als Artikelanhang zu versenden, ist bereits in KIX Start gegeben. Ein Beispiel zum Versenden eines Bericht als Artikelanhang finden Sie daher im Admin-Handbuch von KIX Start.

3.7.1 Vorbedingungen

- Der Job benötigt zum Erstellen des Berichts eine bereits existierende Berichtsdefinition als Grundlage. Legen Sie dazu ggf. eine Berichtsdefinition an.
- Da die HTML-Tabelle des Berichts extrahiert werden soll, muss der Bericht als HTML-Datei erstellt werden können. Dazu ist in der Berichtsdefinition als Ausgabeformat "HTML" anzugeben.

3.7.2 Beispiel Konfiguration des Jobs

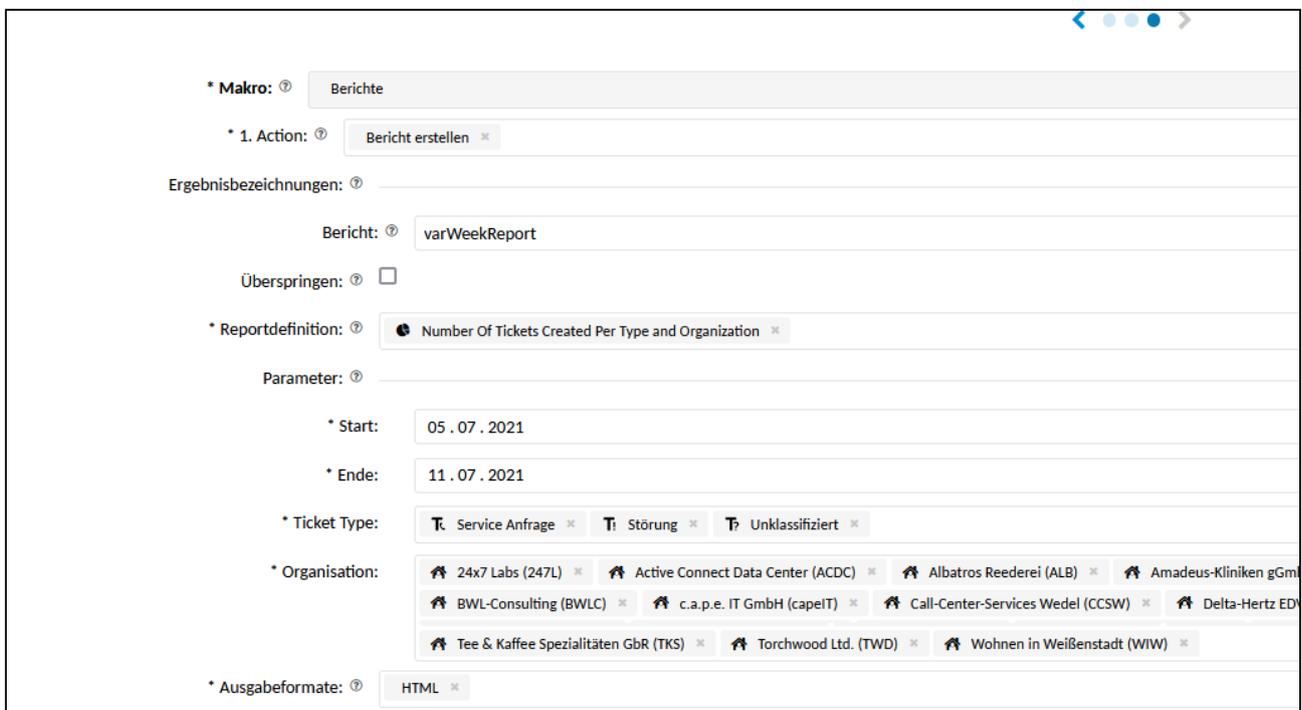
- Job Informationen:
 - Job Typ: Reporting
 - Name: NameDesJobs (z. B. Wochenbericht)
 - Gültigkeit: gültig
- Ausführungsplan:
 - Wochentag(e): Montag
 - Zeit: 6:00 Uhr
 - Events: keine
- Aktionen (in benötigter Reihenfolge)
 1. Aktion: "Bericht erstellen" - Erstellt den Bericht auf Basis seiner Berichtsdefinition
 2. Aktion: "Text entnehmen" - Extrahiert die HTML-Tabelle aus dem erzeugten Bericht.
 3. Aktion: "Macro ausführen" - Führt ein Makro aus. (hier: neues Ticket anlegen).

3.7.3 Konfiguration der Aktionen

3.7.3.1 1. Aktion "Bericht erstellen"

Die Aktion "Bericht erstellen" erstellt den Bericht auf Basis der gewählten Berichtsdefinition. Es werden die in der Berichtsdefinition festgelegten Parameter eingebunden. Tragen Sie die entsprechenden Parameterwerte ein, damit der Job diese Werte beim Erstellen des Berichts setzen kann. Sie können nur Ausgabeformate angeben, die in der Berichtsdefinition festgelegt wurden. Im Beispiel wird der Bericht als HTML ausgegeben, sodass die HTML-Tabelle mit der nächsten Aktion extrahiert werden kann.

Im Beispiel wird der Bericht in der Objektvariablen `varWeekReport` gespeichert (Feld "Bericht"). Wird keine Variable dekliniert, wird der Bericht in der Variablen `Report` gespeichert. Dann kann mit `{Report.Results:n.Attribut}` auf den erstellten Bericht referenziert werden.



3.7.3.2 2. Aktion "Text entnehmen"

Diese Aktion kann mittels RegEx Text extrahieren. Entsprechend dem verwendeten Regulären Ausdruck können sowohl der ganze Text als auch einzelne Textteile extrahiert werden. Der zu extrahierende Text muss bereits existieren (z. B. in einer Variablen).

Im Beispiel wird die HTML-Tabelle des Berichts extrahiert und in der unter "ExtractedText" deklarierten Variable `varTableWeekReport` gespeichert. Auf `varTableWeekReport` wird später in der Aktion "Macro ausführen" referenziert, um den extrahierten Text im Nachrichtentext auszugeben.

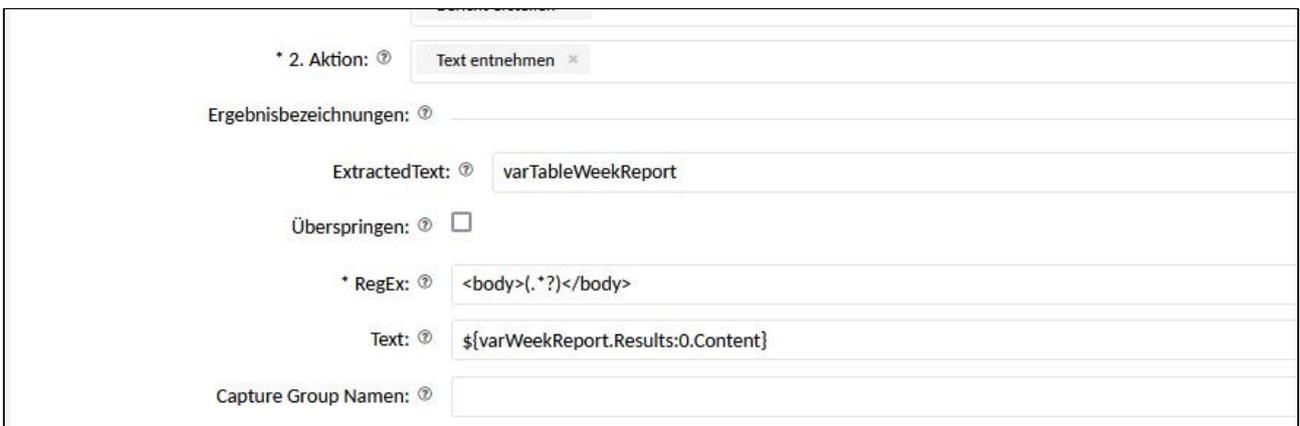
Mit dem unter "RegEx" angegebenen Ausdruck können Sie den zu entnehmenden Text einschränken. Im Beispiel finden keine Einschränkungen statt; der gesamte Body des HTML-Berichts wird extrahiert. Der Ausdruck lautet somit: `<body>(.*)</body>`.

Die unter "Text" angegebene Variable definiert, woher der Text extrahiert wird. Im Beispiel ist dies die in der 1. Aktion deklinierte Objektvariable.

Variablen können komplexe Objekte repräsentieren. Der Zugriff auf die Result-Eigenschaften erfolgt mittels Punkt (.). Ist die Eigenschaft ein Array, kann mittels "`:<index>`" auf ein Element des Arrays zugegriffen werden. Die Variable setzt sich somit wie folgt zusammen:

```
${varWeekReport.Results:0.Content}:
```

- Die Variable `varWeekReport` wird von der Action "Report erstellen" zurückgegeben.
 - Die Variable `varWeekReport` besitzt die Eigenschaft `Results`. `Results` ist ein Array, welches die Berichte in den jeweiligen Ausgabeformaten (HTML, CSV, JSON, XLSX, etc.) beinhaltet.
 - mittels Zugriff auf `:0` wird das erste Ausgabeformat verwendet (hier: HTML)
 - vom Ausgabeformat wird die Eigenschaft `Content` verwendet. Diese bezieht sich auf den gesamten Dateiinhalt. Im Beispiel: das gesamte HTML-Dokument.



* 2. Aktion: **Text entnehmen**

Ergebnisbezeichnungen:

ExtractedText: `varTableWeekReport`

Überspringen:

* RegEx: `<body>(.*)</body>`

Text: `${varWeekReport.Results:0.Content}`

Capture Group Namen:

3.7.3.3 3. Aktion "Macro ausführen"

Diese Aktion führt weitere Macros aus. Im Beispiel wird ein neues Ticket mit folgenden Informationen erstellt:

- die erforderlichen Ticketangaben wie Status, Priorität, Betreff, Kanal etc.
- durch Angabe der Variablen `${varWeekReport.1}` wird der extrahierte Text (s. 2. Aktion "Text entnehmen") im Nachrichteninhalte ausgegeben.



Artikelübersicht (1) Filtern in Liste

Nr.	Neu	Sendertyp	Von	Kanal	Betreff	Erstellt am
1		Agent	Max Müller <m.mueller@example.com>		Wochenbericht	02.07.2021, 11:39

Von: Max Müller <m.mueller@example.com>
Betreff: Wochenbericht

Wochenbericht:
Nachfolgend der aktuelle Wochenbericht

Wochenbericht

Organisation	Unklassifiziert	Störung	Service Anfrage	Total
Mustermann GmbH		1	4	5
My Organisation	11	1		12
c.a.p.e. IT GmbH	1	1		2
Total	12	3	4	19

Info

Die Validierung der Macro Actions erfolgt erst bei Ausführung des Jobs. Hinweise zu eventuellen Fehlern finden Sie im kix.log (Menü *System* > *Logs*) oder im job.log (Tab "Historie" in den Job Details).

4 System - Erweiterte Funktionen

KIX Pro und die KIX Zusatzmodule verfügen über zusätzliche Konfigurationsmöglichkeiten im Menü "System".

Dynamische Felder

Die Übersicht listet alle Dynamischen Felder auf, die initial mit KIX Pro ausgeliefert werden und beschreibt diese. Zudem stehen Ihnen in KIX Pro weitere Feldtypen für die Konfiguration individueller Dynamischer Felder zur Verfügung.

GUI Konfiguration

Für eine bequeme Konfiguration der Benutzeroberfläche des Agentenportals stellt KIX Pro einen JSON-Editor bereit. Der JSON-Editor ermöglicht die Bearbeitung der SysConfig-Schlüssel in aufbereiteter Form inkl. Syntax-Highlighting.

Icons

Im Menü *System > Icons* finden Sie eine Auflistung aller im System vorhandenen Icons. Die Übersicht kann Ihnen helfen, Icons zu finden und Sie können bestehende Icons ändern.

Konfigurationstransfer

KIX Pro ermöglicht den Transfer vollständiger Objekt-Konfigurationen von einer KIX-Umgebung in eine andere. Dadurch können bspw. die Konfigurationen von Test- auf Produktivsysteme oder von On-Premises-Installationen in die KIX Cloud übertragen werden.

Migration

Anwender von KIX 18 Pro können die Daten von KIX 17 nach KIX 18 automatisiert übertragen. Dies erfolgt komfortabel über die Benutzeroberfläche im Menü *System > Migration KIX 17*. Sie müssen in KIX 18 als Nutzer mit Admin-Rechten angemeldet sein, um die Migration durchführen zu können (Rollen: System Admin, Superuser).

Single Sign On via Kerberos

Bei Aufruf des Agenten- oder Self Service Portals kann die Authentifizierung des Nutzers durch Kerberos erfolgen. Der Nutzer muss sich somit nicht zwingend mit Nutzernamen und Passwort an KIX anmelden, wenn er bereits durch sein Login an der Domäne authentifiziert ist (= Single Sign On - Einmalanmeldung).



4.1 Analysen

Im Menü Analyse finden Sie Tools zur Systemanalyse.

4.1.1 FE - Request Logs

Menü	<ul style="list-style-type: none"> • KIX > System > Logs • KIX > System > Analysen > FE-Request Logs
-------------	---

KIX protokolliert die HTTP-Anfragen (Requests) und speichert diese in einem Logfile. Pro Tag wird 1 Request Log geschrieben. Die HTTP-Request-Logfiles finden Sie im Menü *System > Logs > Rubrik: Frontend Server*. Hier können Sie sich die Logfiles anzeigen lassen und auch herunterladen. Die Logfiles liegen im CSV-Format vor. Trennzeichen ist der Tabulator (\t).

Darüber hinaus stellt Ihnen KIX Pro ein Support-Tool zur Verfügung, welches Ihnen einen Überblick über die aktuellen HTTP-Anfragen am System liefert. Dieses finden Sie im Menü *System > Analysen > FE-Request Logs*. Es werden die Metriken der Request-Logs für das Frontend ausgewertet und grafisch für Sie aufbereitet. Sie können Sie einzelne oder mehrere Logfiles auswählen und anhand derer die Performance analysieren.

4.1.1.1 Auswahl der Logfiles

Das Auswahlfeld bietet Ihnen die im Log gespeicherten HTTP-Request-Logfiles zur Auswahl an. Wählen Sie aus, welche Logfiles Sie analysieren möchten. Die gewählten Logfiles werden zusammen ausgewertet und das Ergebnis in den Metriken, Charts und Tabelle angezeigt.

4.1.1.2 Metriken

Die Tabelle liefert Ihnen eine Übersicht über die berechneten Metriken und deren ermittelte Werte:

Metrik	Beschreibung
Total Request Count	Wie viele Requests gibt es insgesamt?
Average Request Duration	Durchschnittliche Laufzeit eines Requests
Fastest Request	Welcher ist der schnellste Request? In Klammern: Dauer des Requests in ms
Slowest Request	Welcher ist der langsamste Request? In Klammern: Dauer des Requests in ms

Metrik	Beschreibung
most frequented resource	Welche Ressource wird am meisten frequentiert/ penetriert? In Klammern: Anzahl der Requests
Requests/Second (last Minute)	Durchschnittliche Anzahl der Requests pro Sekunde innerhalb einer Minute Betrachtet wird die Zeit in den gewählten Logfiles.
Requests/Second (last 5 Minutes)	Durchschnittliche Anzahl der Requests pro Sekunde innerhalb von 5 Minuten Betrachtet wird die Zeit in den gewählten Logfiles.
Requests/Second (last 10 Minutes)	Durchschnittliche Anzahl der Requests pro Sekunde innerhalb von 10 Minuten Betrachtet wird die Zeit in den gewählten Logfiles.

4.1.1.3 Charts

Die Charts visualisieren die berechneten Werte:

Chart	Beschreibung
Request Count	24-Stunden-Chart zur Ermittlung der Spitzenzeiten, gruppiert in 30-Minuten-Paketen. Zeigt an, wie viele Request zu welcher Zeit stattgefunden haben.
Request Duration in ms	Laufzeiten-Chart Anzeige der Dauer der Requests, gruppiert in 50 ms Zeigt die Anzahl der Requests innerhalb eines bestimmten Zeit-Spektrums
Request Count in Last x Minutes	Anzahl an Requests in den letzten 1 5 15 Minuten (relativ bezogen auf die gewählten Logfiles)

4.1.1.4 Tabelle

Die Tabelle listet Ihnen die Logeinträge auf und zeigt zusätzliche Anmerkungen zu den Requests an. Auffälligkeiten werden farblich hervorgehoben:

- gelb: Request dauerte länger als 500 ms
- rot: Fehlerhafter Request

Spalte	Beschreibung
Datum	Zeitpunkt des Logeintrags
Operation	Die verwendete Methode
Status	HTTP-Status-Code zum Erkennen erfolgreicher oder fehlerhafter Operationen
Remarks	<p>Hinweise auf bestimmte Spezifikationen des Requests</p> <ul style="list-style-type: none"> • expand: Request enthält "expand" • include: Request enthält "include" • limit =0: Request enthält kein Limit • no search: Request enthält Filter, aber keine Suche • no searchlimit: Request enthält kein Suchlimit
Duration	Dauer des Requests
Ressource	Anzeige der betroffenen Ressource
Parameter	Anzeige der Parameter im Request

4.2 Dynamische Felder in KIX Pro

KIX Pro und einige Zusatzmodule verwenden zusätzliche Dynamische Felder, welche initial mit KIX Pro ausgeliefert werden. Sie können die Konfiguration dieser Felder im Bedarfsfall anpassen. Ändern Sie jedoch nicht die Feldbezeichnungen. Anderenfalls können die Felder vom System nicht korrekt referenziert und verwendet werden.

Die initial mit KIX Start ausgelieferten Dynamischen Felder finden Sie im Admin-Handbuch von KIX Start.

Beachten Sie beim Einbinden von Dynamische Feldern den Objekttyp! Der Objekttyp legt den Kontext fest, in dem das Dynamische Feld verwendbar ist. So können Dynamische Felder vom Objekttyp "Ticket" ausschließlich in ticketrelevanten Oberflächen eingebunden werden, nicht aber bspw. in FAQs oder Organisationen.

Das Einbinden Dynamischer Felder in die Ticketerstellungsmasken von KIX Pro erfolgt schnell und komfortabel über die Konfiguration von Vorlagen (siehe Seite 151) . Die Konfiguration via SysConfig-Schlüssel kann somit entfallen, dient aber als Fallback. Um Dynamische Felder in den Dialog "Neues Ticket" zu integrieren, können Sie die Vorlage "Default - New Ticket Template" entsprechend anpassen. In allen anderen Oberflächen können Dynamische Felder über die GUI-Konfiguration (siehe Seite 55) eingebunden werden. Orientieren Sie sich hierbei an den Konfigurationsbeispielen von KIX Start.

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
AffectedServices	AssetReference	Ticket	<p>Auswahl der "Betroffenen Services" am Ticket. Je nach Konfiguration (siehe Seite 188) stehen alle Services zur Auswahl oder der Servicebaum. Diese Konfiguration erfolgt in den SysConfig-Schlüsseln "service-contract-configuration" bzw. "service-contract-configuration-ssp".</p> <p>Bei der Migration von KIX 17 nach KIX 18 übernimmt das Feld die Ticket-Service-Zuordnungen.</p>	nicht erforderlich. Bei Bedarf können jedoch die Verwendungsstatus und die maximale Feldanzahl geändert werden.

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
AnonymiseTicket	AssetReference	Ticket	<p>Wird intern zur Kennzeichnung des Anonymisierungsstatus von Tickets zu kennzeichnen.</p> <p>Der Status wird durch den Job "Anonymisation" gesetzt.</p>	nicht erforderlich
ChildTickets	TicketReference	Ticket	<p>Wird für die Verknüpfung von Tickets mit Kind-Tickets (untergeordnete Tickets) verwendet.</p> <p>Agenten können beim Verknüpfen von Tickets die Art der Verknüpfung wählen (Eltern Kind Normal).</p> <p>Die Konfiguration legt fest, welche Eigenschaften Tickets aufweisen müssen, damit sie in der Liste zur Auswahl stehen. Die initiale Konfiguration erlaubt die Eintragung von max. 25 Tickets der Status "neu", "offen", "warten zur Erinnerung und "Warten auf automatisches Schließen". Die Tickets werden als Kind-Tickets verknüpft.</p>	<ul style="list-style-type: none"> • Beschränkung der Anzahl auswählbarer Tickets (Count Max) • Status ergänzen oder entfernen • Beschränkung der auswählbaren Tickets auf bestimmte Tickettypen (TicketTypes)

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
CloseCode	Selection	Ticket	<p>Wird für die Auswahlliste "Abschlusscode" im Schließen-Dialog verwendet.</p> <p>Die Konfiguration definiert den Inhalt die Auswahlliste.</p> <p>Das Feld wird von der Aktion "Schließen" verwendet, welche den Schließen-Dialog öffnet. Die Aktion kann im Menü <i>Ticket > Aktionen</i> nachkonfiguriert werden.</p> <p>In der Komplexsuche kann nach Abschlusscodes gesucht werden.</p>	<ul style="list-style-type: none"> • Änderung der Bezeichnung der Listeneinträge • Ergänzen und Entfernen von Listeneinträgen
CreateFAQSuggestion	Selection	Ticket	<p>Wird im Zusammenhang mit dem Job "Create FAQ Suggestion" verwendet.</p> <p>Das Feld steht in den Dialogen "Ticket bearbeiten" und "Ticket schließen" als Auswahlfeld zur Verfügung, wenn der Kanal "Notiz" oder "E-Mail" gewählt ist.</p> <p>Wird die Option "ja" ausgewählt, erstellt der Job einen neuen FAQ Eintrag unter Angabe der im Ticket verwendeten Informationen (s. auch Jobs - Erweiterte Funktionen (siehe Seite 13))</p>	nicht erforderlich

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
MergeToTicket	TicketReference	Ticket	<p>Wird für das Zusammenfassen von Tickets verwendet, bspw. um eine fehlgeleitete E-Mail an ein Ticket anzuhängen.</p> <p>Das Feld ist in den Dialog der initialen Aktion "Zusammenfassen Merge (siehe Seite 99) " integriert. Es referenziert auf alle Tickets, von denen ein Ticket als Zielticket ausgewählt werden kann. Die Aktion kann bei Bedarf nachkonfiguriert werden (s. Aktionen (siehe Seite 93)).</p> <p>Das Zusammenfassen der Tickets wird durch den initialen Job "TicketMerge" beim Speichern der Aktion ausgelöst. Der Job kann ebenfalls nachkonfiguriert (siehe Seite 13) werden.</p>	Die Konfiguration darf nicht geändert werden!

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
ParentTickets	TicketReference	Ticket	<p>Wird für die Verknüpfung von Tickets mit Eltern-Tickets (übergeordnete Tickets) verwendet. Referenziert auf das Eltern-Ticket.</p> <p>Agenten können beim Verknüpfen von Tickets die Art der Verknüpfung wählen (Eltern Kind Normal).</p> <p>Die Konfiguration legt fest, welche Eigenschaften Tickets aufweisen müssen, damit sie in der Liste zur Auswahl stehen. Die initiale Konfiguration erlaubt die Eintragung von max. 5 Tickets der Status "neu", "offen", "warten zur Erinnerung und "Warten auf automatisches Schließen". Die Tickets werden als Elterntickets verknüpft.</p>	<ul style="list-style-type: none"> • Beschränkung der Anzahl auswählbarer Eltern-Tickets (Count Max) • Status ergänzen oder entfernen • Beschränkung der auswählbaren Tickets auf bestimmte Tickettypen (TicketTypes)

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
PlannedEffort	Text	Ticket	<p>Enthält den Wert für den geplanten Zeitaufwand (Sollzeit). Die Sollzeit bildet die Basis für die Zeiterfassung. Sie definiert, in welcher Zeit ein Ticket bearbeitet werden soll.</p> <p>Wird von den Jobs "Auto Set Planned Effort (Incident)" und "Auto Set Planned Effort (Service Request)" sowie von der Aktion "Geplanter Aufwand" verwendet:</p> <ul style="list-style-type: none"> • Die Jobs setzen den Wert der Sollzeit in das Dynamische Feld. Der vorgegebene Wert kann in den Jobs geändert werden. • Die Aktion "Geplanter Aufwand" öffnet einen Dialog, in welchem das Dynamische Feld inkl. der vom Job gesetzten Sollzeit enthalten ist. Agenten können diese vorgegebene Sollzeit manuell pro Ticket ändern. 	<p>Die Konfiguration darf nicht geändert werden!</p>

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
RelatedTickets	TicketReference	Ticket	<p>Wird in der Ticket-Bearbeiten-Maske für die Auswahl der Verwandten Tickets benötigt.</p> <p>Die Konfiguration legt fest, welche Eigenschaften Tickets aufweisen müssen, damit sie in der Liste zur Auswahl stehen. Die initiale Konfiguration erlaubt die Eintragung von max. 25 Tickets der Status "neu", "offen", "warten zur Erinnerung und "Warten auf automatisches Schließen" erlaubt. Die ausgewählten Tickets werden als "Normal" verknüpft.</p>	<ul style="list-style-type: none"> • Beschränkung der Anzahl auswählbarer Tickets (Count Max) • Status ergänzen oder entfernen • Beschränkung der auswählbaren Tickets auf bestimmte Tickettypen (TicketTypes)
RelatedNewsTickets	TicketReference	News	<p>Wird vom News Modul verwendet, um in News auf verwandte Tickets hinzuweisen.</p>	<p>Initial können maximal 5 Tickets ausgewählt werden. Sie können diese Anzahl nach Erfordernis ändern.</p>

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
Source	Text	Contact	<p>Wird für die Bereitstellung individueller Kontakt-Einträge aus LDAP/AD-Verzeichnisdiensten genutzt.</p> <p>Das Feld kann in der UserSyncMap (siehe Seite 24) angegeben werden, um Kontaktattribute aus dem LDAP/AD aufzunehmen. Enthält das Feld einen Wert, wird dieser in der Kontakt-Detailansicht angezeigt.</p> <p>Das Feld bietet somit die Möglichkeit, bspw.</p> <ul style="list-style-type: none"> • an einem Kontakteintrag zu hinterlegen, aus welchem AD der Kontakt stammt • die Bezeichner für Kontaktdaten anderer Systeme auch in KIX bereitzustellen. 	nicht erforderlich
Satisfaction Points	Selection	Ticket	<p>Auswahlfeld für Bewertungspunkte.</p> <p>Wird im Self Service Portal für die Aktion "Customer Feedback" benötigt. Ermöglicht die Angabe eines Kundenfeedbacks (Ranking).</p>	nicht erforderlich
Satisfaction Remark	Textarea	Ticket	<p>Bemerkung zu den Bewertungspunkten.</p> <p>Wird im Self Service Portal für die Aktion "Customer Feedback" benötigt. Ermöglicht einen Kommentar zum Ranking.</p>	nicht erforderlich

Feldname	Feldtyp	Objekttyp	Beschreibung	Konfigurationsmöglichkeiten
Type	Selection	Organisation	<p>Wird für die Bereitstellung von Organisationsdaten genutzt und ist in den Oberflächen zur Anlage/Bearbeitung von Organisationen enthalten. Enthält das Feld einen Wert, wird dieser in der Detailansicht der Organisation angezeigt.</p> <p>Das Feld kann bspw. in der UserSyncMap (siehe Seite 24) angegeben werden, um Organisationsdaten aus dem LDAP/AD aufzunehmen. Damit ist es möglich</p> <ul style="list-style-type: none"> • an einer Organisation zu hinterlegen, aus welchem AD die Organisation stammt • die Bezeichner für Organisationsdaten anderer Systeme auch in KIX bereitzustellen. 	<ul style="list-style-type: none"> • Festlegen der Anzahl auswählbarer Einträge • Festlegen der auswählbaren Einträge
WorkOrder	Textarea	Ticket	<p>Wird für die Ticketaktion (siehe Seite 93) "Feldagent zuweisen" (AppAssignWorkOrder) benötigt und bildet in diesem Dialog das Eingabefeld "Arbeitsanweisung".</p>	nicht erforderlich

4.3 GUI Konfiguration

Administratoren von KIX Pro können die Benutzeroberfläche des Agentenportals komfortabel im Menü *System > GUI-Konfiguration > Agentenportal* anpassen. Hier sind alle für die GUI-Konfiguration relevanten Schlüssel übersichtlich in einem Explorer (1) aufgeführt. Ein integrierter JSON-Editor (2) erleichtert die Bearbeitung und der Default-Wert des Konfigurationsschlüssels kann parallel zum Vergleichen angezeigt werden (3).

Die Grundlagen und Prinzipien der GUI-Konfiguration entsprechen denen, wie im Admin-Handbuch von KIX 18 Start beschrieben.

Hinweis: Der Explorer enthält nur die Konfigurationsschlüssel, welche für die Anpassung der **Benutzeroberfläche im Agentenportal** relevant sind. Alle weiteren Konfigurationsschlüssel sind im Menü *System > SysConfig* anzupassen.

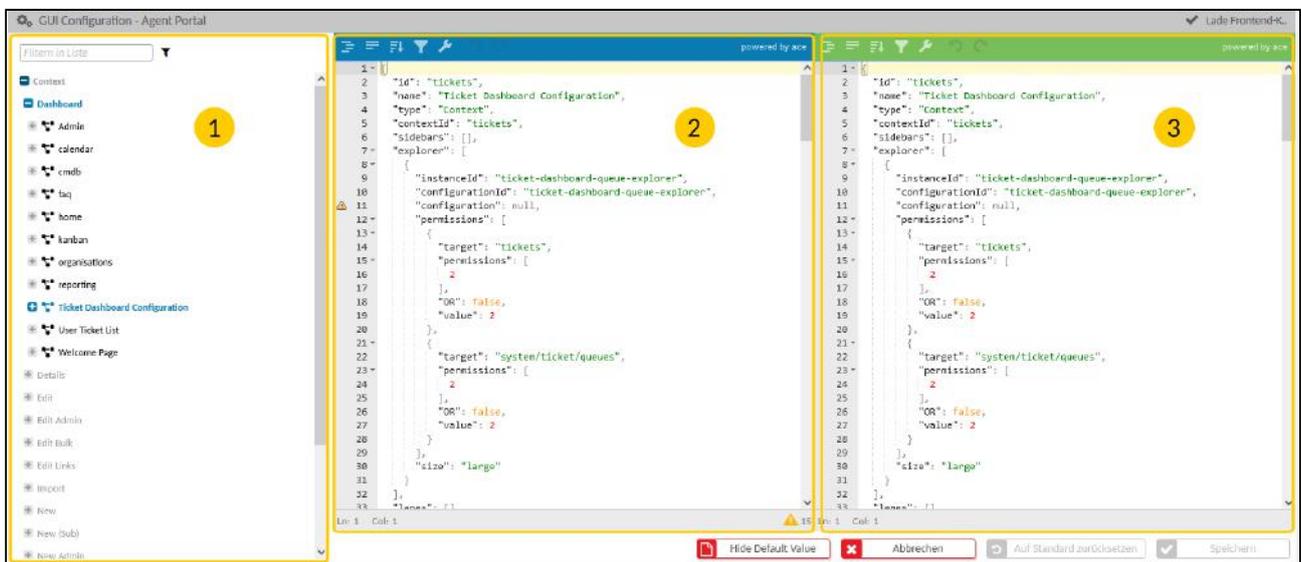


Abb.: Aufbau der GUI-Konfiguration

⚠ Achtung!

Änderungen an der GUI-Konfiguration können schwerwiegende Systemfehler verursachen. Sie sollten daher tiefgreifende Kenntnisse über das System und von JSON besitzen, wenn Sie Änderungen an der GUI-Konfiguration vornehmen.

Wenden Sie sich bitte an unseren Support, wenn Sie Hilfe benötigen oder wenn Sie die Benutzeroberfläche durch uns anpassen lassen möchten.

4.3.1 Der Explorer

Der Explorer enthält die für die Benutzeroberfläche des Agentenportals relevanten Konfigurationen. Die Baumstruktur entspricht der Konfigurationshierarchie, welche der GUI-Konfiguration zugrunde liegt.

Die Wurzeln im Konfigurationen-Baum sind:

- Context
 - Beinhaltet alle Konfigurationen, die einem Kontext zugeordnet sind:
z. B. Dashboards, Detailseiten, Dialoge etc.
- Form
 - Beinhaltet alle Konfigurationen, die einem Formular zugeordnet sind:
z. B. Dialoge "Neues Ticket", "Ticket bearbeiten", etc.
- Not Referenced Configuration
 - Beinhaltet alle Konfigurationen, welche in keinem Kontext oder Form referenziert sind:
z. B. Status des Setup-Assistenten.

Mittels Suchfeld oberhalb des Explorers kann der Baum gefiltert werden. Die Suche basiert auf einer Volltextsuche ohne Wildcards (*). Die Eingabe von "tic" findet sowohl **Article** als auch **Tickets**. Jedem Knoten im Baum sind die Metadaten der Konfiguration (z. B. ID, Name, Typ, etc.) hinterlegt, sodass danach gefiltert werden kann (z. B. "FormField" filtert alle FormField-Konfigurationen). Die Suche nach "modified" findet alle Schlüssel, deren Konfiguration geändert wurde. Modifizierte Konfigurationen werden mit einem in Klammern gesetzten "modified" gekennzeichnet.

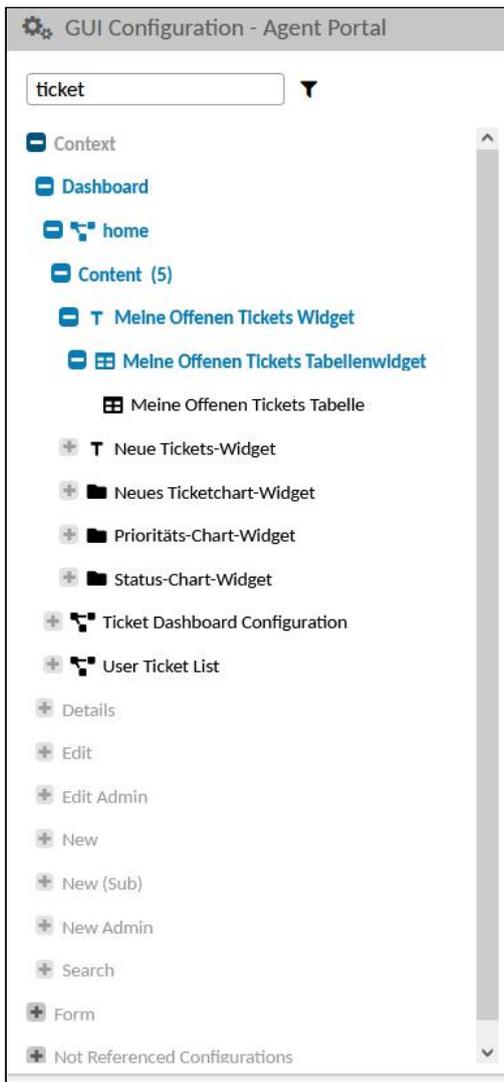
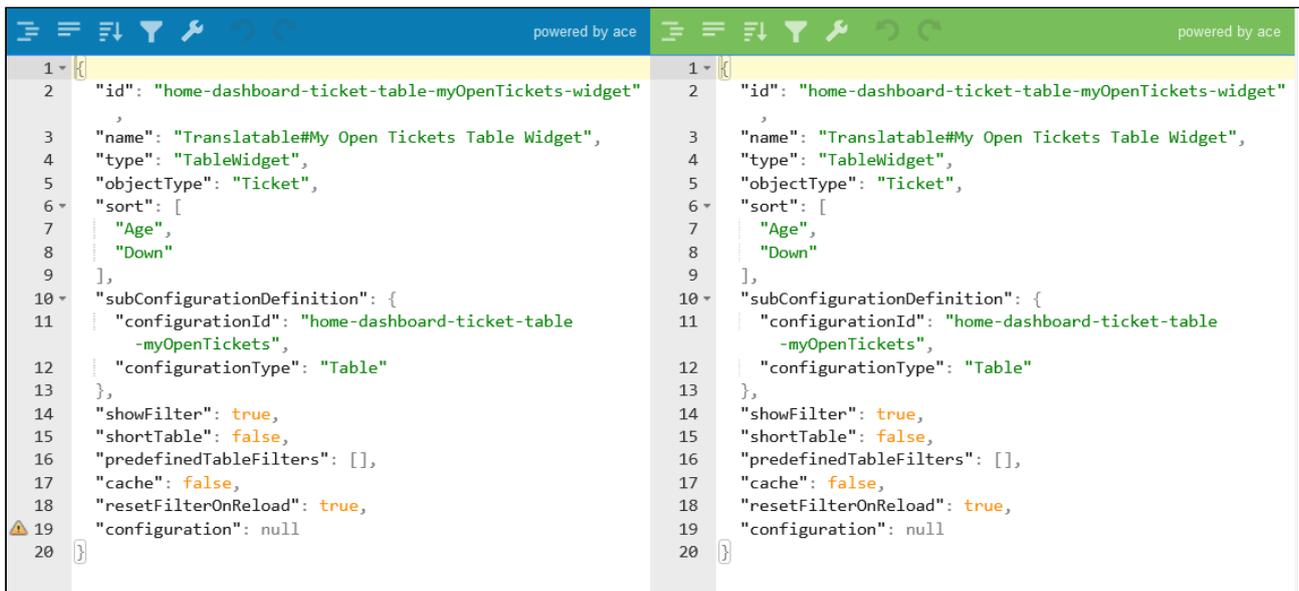


Abb.: Der Konfigurationen-Baum

4.3.2 Der JSON-Editor

Nach Klick auf einen Konfigurationszweig im Explorer, wird dessen Wert im integrierten JSON-Editor angezeigt und kann dort bearbeitet werden. Dies erfolgt in gleicher Weise wie in den Konfigurationsbeispielen im Admin-Handbuch von KIX 18 Start beschrieben.

Die linke Seite des Editors (blau) enthält die aktuelle Konfiguration des gewählten Schlüssels zur Ansicht und Bearbeitung. Die rechte Seite des Editors (grün) enthält die initiale Konfiguration des Schlüssels zum Vergleich. Sie kann bei Bedarf mit Klick auf "Zeige Standardwert" eingeblendet werden.



```

1  {
2  "id": "home-dashboard-ticket-table-myOpenTickets-widget"
3  ,
4  "name": "Translatable#My Open Tickets Table Widget",
5  "type": "TableWidget",
6  "objectType": "Ticket",
7  "sort": [
8  "Age",
9  "Down"
10 ],
11 "subConfigurationDefinition": {
12 "configurationId": "home-dashboard-ticket-table-myOpenTickets",
13 "configurationType": "Table"
14 },
15 "showFilter": true,
16 "shortTable": false,
17 "predefinedTableFilters": [],
18 "cache": false,
19 "resetFilterOnReload": true,
20 "configuration": null
  }
  
```

Abb.: Der JSON-Editor mit parallelem Default-Wert

Zur besseren Lesbarkeit wird der JSON-String mit Einrückungen, Zeilenvorschüben und Syntax-Highlighting dargestellt. Über die Schaltflächen im JSON-Editor kann die Ansicht des Editors geändert werden.

	Formatierte Darstellung des JSON-Strings mit Einrückungen und Zeilenvorschüben
	Verdichtete Darstellung des JSON-Strings ohne Leerzeichen und Zeilenumbrüche
	Sortierung des Inhalts (aufsteigend/absteigend)
	Filtern, Sortieren und Transformieren des Inhalts
	Korrektur des JSON-Strings (Anführungszeichen und Zeilenumbrüche ausgleichen, Kommentare entfernen)
	Letzte Aktion rückgängig machen
	Letzte Aktion wiederherstellen / wiederholen

Die Konfigurationen werden mittels JSON Schema validiert und entsprechende Warnungen und Fehler im Editor angezeigt. Führen Sie den Mauszeiger über das gelbe Ausführungszeichen, um Informationen zum Fehler zu erhalten.

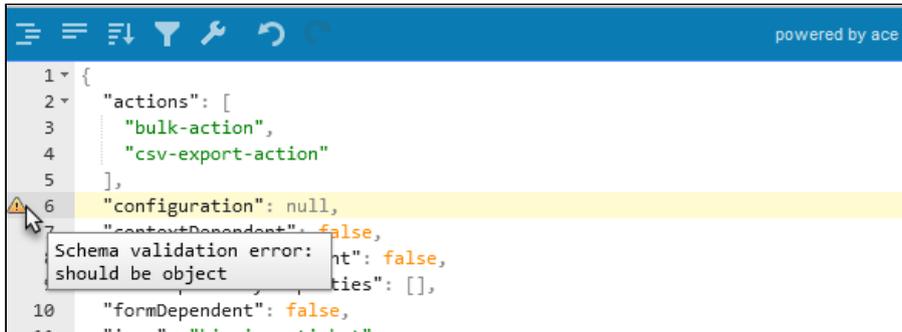


Abb.: Fehlerhinweis im JSON-Editor

Hinweis

Klicken Sie nach Änderungen am Konfigurationsschlüssel auf "Lade Frontend-Konfigurationen neu". Erst danach wird der Frontendcache neu erstellt und der Konfigurationsbaum neu erstellt.

Verwendung der Schaltflächen

Zeige Standardwert (Show Default Value)	Öffnet einen weiteren JSON-Editor (grün) mit dem Default-Wert des Schlüssels zum Vergleichen der Änderungen mit dem Ausgangswert. Änderungen am Wert sind hierüber nicht möglich.
Standardwert ausblenden (Hide Default Value)	Blendet den grünen JSON-Editor aus.
Abbrechen	Verwirft die aktuellen Änderungen ohne zu Speichern.
Auf Standard zurücksetzen	Stellt den Default-Wert wieder her. Danach muss erneut gespeichert werden.  Diese Funktion entbindet Sie jedoch nicht von Ihrer Sorgfaltspflicht im Umgang mit der GUI-Konfiguration.



Speichern	Speichert Ihre Änderungen an der Konfiguration. Das Neu-Laden der Frontend-Konfiguration ist zusätzlich erforderlich.
-----------	--

4.4 Icons

Im Menü *System > Icons* finden Sie die vom System verwendeten Icons. Sie können

- der Übersicht die Bezeichnung der Icons entnehmen, wenn Sie die System-Icons bei der Konfiguration der Benutzeroberfläche oder Sidebar verwenden möchten
- die System-Icons durch eigene Icons ersetzen, wenn Sie unternehmensspezifische Icons nutzen möchten
- das KIX Logo und das KIX Icon durch Ihr Unternehmens-Logo bzw. -Icon ersetzen.

So ändern Sie ein Icon:

1. Navigieren Sie zu *System > Icons*.
2. Klicken Sie in der Tabelle auf das zu ändernde Icon. Es wird ein Formular-Dialog geöffnet, in dem Sie das Icon ändern können.
3. Klicken Sie auf die Schaltfläche "Bilddatei wählen". Es wird ein Dialog für die Navigation in Ihrer Infrastruktur geöffnet.
4. Navigieren Sie zu der Bilddatei, die Sie als neues Icon hochladen möchten.
5. Wählen Sie für den Upload die entsprechende Bilddatei aus.
6. Nach erfolgreichem Upload wird das Bilddatei als Icon neben der Schaltfläche "Bilddatei wählen" angezeigt.

Das Icon kann nun an gewünschter Stelle in KIX verwendet werden.

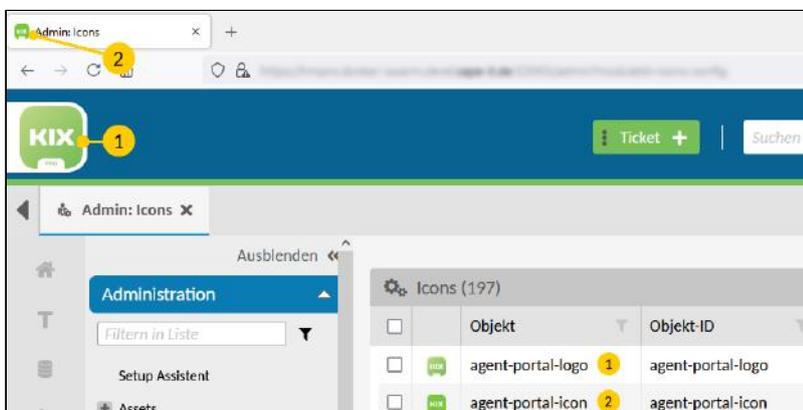
KIX Logo und KIX Icon ändern

Das KIX Logo befindet sich am linken Bildschirmrand (**1**) des Kopfbereichs. Das KIX Icon (**2**) wird als Tab-Icon (Favicon) verwendet.

Sie können sowohl das KIX Logo als auch das KIX Icon des Agentenportals durch Ihr Unternehmens-Logo bzw. -Icon ersetzen. Filtern Sie dazu die Tabelle der Icons nach "agent" und ersetzen Sie die Grafik(en) wie zuvor beschrieben.

1 **Logo:** Objekt "agent-portal-logo"

2 **Icon (Favicon):** Objekt "agent-portal-icon"



4.5 Konfigurationstransfer

Im Menü *System* >

Konfigurationstransfer können Sie vollständige Objekt-Konfigurationen von einer Umgebung in eine andere übertragen. Möglich sind sowohl der Export aus KIX als auch der Import nach KIX.

Dies ermöglicht

- das Übertragen von Konfigurationen von Test- auf Produktivsysteme oder von On-Premises-Installationen in die KIX Cloud
- das Ausrollen vorkonfigurierter Systemeinstellungen durch Support oder Kundenberatung
- den Transfer von Konfigurationen zur Einrichtung, Verwendung und Anzeige von Dynamischen Feldern
- u. a. m.

Sie können die Konfigurationen folgender Objekte im- bzw. exportieren:

- Jobs
- Dynamische Felder
- Reportdefinitionen
- Vorlagen und Vorlagengruppen
- Aktionen
- Teams
- Status und Typen von Tickets
- Systemkonfigurationen
- General Catalog Einträge
- Assetklassen
- FAQ Kategorien
- E-Mail-Filter
- Objekt Icons

Hinweis: Der Im- und Export von Konfigurationen erfolgt aktuell nur im JSON-Format.

4.5.1 Export von Konfigurationen

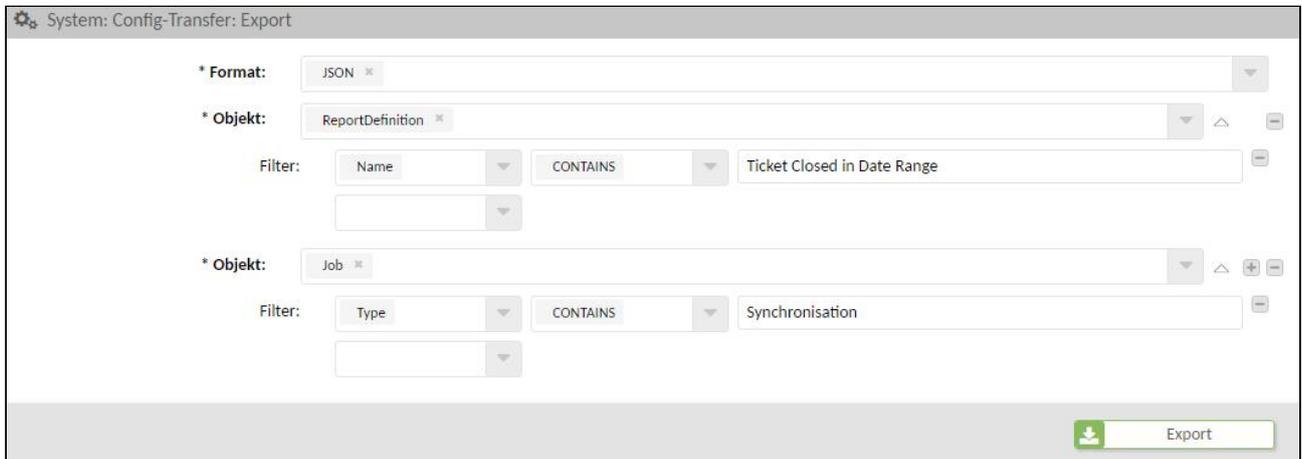
Der Export von Konfigurationen erfolgt im Menü *System* > *Konfigurationstransfer* > *Export*.

Legen Sie im Dialog fest, in welches Ausgabeformat der Export erfolgen soll und welche Datensätze exportiert werden sollen. Die zu exportierenden Datensätze können Sie durch Auswahl des Objekts (z. B. Report Definition) und durch Setzen von Filtern festlegen (Details s. nachfolgende Tabelle).

Inhalte auf dieser Seite:

- [Export von Konfigurationen](#) (siehe Seite 62)
- [Import von Konfigurationen](#) (siehe Seite 64)
- [Hinweise für den Im- und Export](#) (siehe Seite 67)

Der Export startet nach einem Klick auf die "Export"-Schaltfläche. Das Exportieren kann je nach Anzahl der gewählten Objekte und der darin hinterlegten Datensätze einige Zeit in Anspruch nehmen. Im Anschluss wird eine Datei im gewählten Ausgabeformat erzeugt und über die Funktionen des verwendeten Browsers zum Download bereitgestellt.



The screenshot shows the 'System: Config-Transfer: Export' interface. It features two object selection sections. The first section is for 'ReportDefinition' with a filter 'Name' and a value 'Ticket Closed in Date Range'. The second section is for 'Job' with a filter 'Type' and a value 'Synchronisation'. An 'Export' button is visible at the bottom right.

Abb.: Objektauswahl für den Export

Der Formulario dialog enthält folgende Eingabefelder:

Feld	Beschreibung
Format	Wählen Sie aus, in welches Ausgabeformat die Konfiguration exportiert werden soll (aktuell nur JSON).
Objekt	<p>Wählen Sie aus, welchen Objekttyp Sie exportieren möchten.</p> <ul style="list-style-type: none"> • Sie können nacheinander mehrere Objekte wählen. Klicken Sie dazu auf die Plus-Schaltfläche. • Ist kein Objekt gewählt, werden alle verfügbaren Konfigurationen exportiert. Die Plus-Schaltfläche darf dazu noch nicht angeklickt sein. • Es werden sowohl gültige als auch (temporär) ungültige Objekte exportiert.
Filter	<p>Durch Angabe von Filterkriterien können Sie die zu exportierenden Datensätze konkretisieren.</p> <ul style="list-style-type: none"> • Es werden alle Datensätze des Objekts exportiert, die den gewählten Filterkriterien entsprechen. • Die zur Verfügung stehenden Filterkriterien sind vom jeweiligen Objekttyp abhängig. • Wird kein Filter gesetzt, werden alle erlaubten Datensätze des gewählten Objekts exportiert.
	Hinzufügen bzw. Entfernen von Objekten und Filtern

⚠ Wichtig

In den Konfigurationen hinterlegte Berechtigungszuordnungen werden aktuell **nicht** exportiert und müssen manuell gesetzt werden.

4.5.2 Import von Konfigurationen

Der Import von Konfigurationen erfolgt im Menü *System > Konfigurationstransfer > Import* mittels Datei-Upload. Die Datei muss die Datensätze für den Import enthalten und in einem unterstützten Format vorliegen (s. o.: Export -> Format). Der Import kann je nach Dateigröße einen Moment dauern. Beachten Sie auch die unten aufgeführten Hinweise für den Im- und Export!

⚠ Wichtig

- Der Import legt keine neuen Objekte an. Achten Sie daher darauf, dass alle zu importierenden Objekte in KIX existieren, z. B. der im Filter von ObjectActions angegebene Statustyp. Legen Sie ggf. die Objekte zuvor in KIX an.
- Für eine korrekte und vollständige Anzeige bitte nach dem Import mittels Konsolen-Kommando "`Console::Command::Maint::Cache::Delete`" den Cache löschen und eine Ab- und Anmeldung durchführen.

Der Formulario dialog enthält folgende Eingabefelder:

Feld	Beschreibung								
Modus	Der gewählte Modus bestimmt, wie die zu importierenden Datensätze behandelt werden. Beispielsweise, ob bei identischem Identifikator ein neuer Datensatz angelegt wird oder ob ein bestehender Datensatz aktualisiert wird.								
	<table border="1"> <tr> <td>Standard</td> <td>Aktualisierung oder Hinzufügen von Datensätzen, sofern diese vorhanden sind oder nicht (Standard).</td> </tr> <tr> <td>OnlyUpdate</td> <td>Führt nur ein Update bestehender Datensätze aus.</td> </tr> <tr> <td>OnlyAdd</td> <td>Ein Datensatz wird nur hinzugefügt, wenn er nicht existiert.</td> </tr> <tr> <td>ForceAdd</td> <td> <p>Erzwungene Neuanlage:</p> <ul style="list-style-type: none"> • Ein Datensatz wird kopiert, wenn dieser schon existiert. Der Datensatzname erhält ein vorangestelltes "Copy" (außer bei ObjectIcons - s. unten). • Nicht vorhandene Datensätze werden hinzugefügt. </td> </tr> </table>	Standard	Aktualisierung oder Hinzufügen von Datensätzen, sofern diese vorhanden sind oder nicht (Standard).	OnlyUpdate	Führt nur ein Update bestehender Datensätze aus.	OnlyAdd	Ein Datensatz wird nur hinzugefügt, wenn er nicht existiert.	ForceAdd	<p>Erzwungene Neuanlage:</p> <ul style="list-style-type: none"> • Ein Datensatz wird kopiert, wenn dieser schon existiert. Der Datensatzname erhält ein vorangestelltes "Copy" (außer bei ObjectIcons - s. unten). • Nicht vorhandene Datensätze werden hinzugefügt.
Standard	Aktualisierung oder Hinzufügen von Datensätzen, sofern diese vorhanden sind oder nicht (Standard).								
OnlyUpdate	Führt nur ein Update bestehender Datensätze aus.								
OnlyAdd	Ein Datensatz wird nur hinzugefügt, wenn er nicht existiert.								
ForceAdd	<p>Erzwungene Neuanlage:</p> <ul style="list-style-type: none"> • Ein Datensatz wird kopiert, wenn dieser schon existiert. Der Datensatzname erhält ein vorangestelltes "Copy" (außer bei ObjectIcons - s. unten). • Nicht vorhandene Datensätze werden hinzugefügt. 								
Datei	Wählen Sie die hochzuladende Datei aus. Es kann nur 1 Datei hochgeladen werden. Tipp: Sie können die Datei auch via Drag & Drop auf die Schaltfläche ziehen.								

Nach dem Import wird eine Ergebnistabelle mit folgenden Informationen angezeigt:

System: Config-Transfer: Import: Result							
Objekt	Einträge	Ignored	Updated	Added	Force Added	Failed	
DynamicField	27	0	25	0	0	2	
Job	9	0	9	0	0	0	
ObjectAction	11	0	11	0	0	0	
ReportDefinition	6	0	6	0	0	0	
Template	13	0	13	0	0	0	

+ New Import

Abb.: Ergebnistabelle des Imports

Feld	Beschreibung
Objekt	Das Objekt, dessen Datensätze importiert wurden.
Einträge	Anzahl der importierten Datensätze
Ignoriert	Anzahl der ignorierten Datensätze Ignoriert werden Datensätze, wenn sie z. B. bei gewähltem Modus "OnlyAdd" schon existieren oder bei "OnyUpdate" nicht existieren.
Aktualisiert	Anzahl der aktualisierten Datensätze
Hinzugefügt	Anzahl der hinzugefügten Datensätze
Hinzugefügt erzwungen	Anzahl der Datensätze die durch den Modus "ForceAdd" kopiert und mit "Copy" markiert wurden. Dies betrifft grundlegend nur die Datensätze, die schon im System existieren.
Fehlgeschlagen	Anzahl der Datensätze, die nicht importiert werden konnten. Die Gründe dazu finden Sie unter <i>System > Logs</i> .

4.5.3 Hinweise für den Im- und Export

Der Export betrachtet stets die Grundkonfigurationen. Individuelle Parameter (z. B. Icons von Teams) bleiben unberücksichtigt.

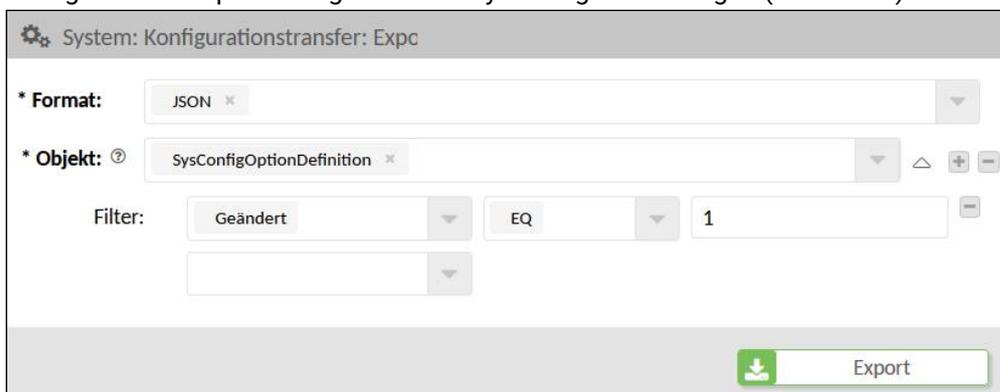
Baumstrukturen (z. B. Teams, Status, Vorlagengruppen) werden komplett exportiert, d. h. inkl. aller ihrer Sub-Kategorien. Es wird stets vom Root-Team ausgegangen. Beim Export von Sub-Kategorien werden auch dessen Eltern-Elemente betrachtet, um die Kategorien beim (Re-)Import korrekt im Baum einordnen zu können.

Team-Konfigurationen:

- Der Kalender wird nur als Kalendernummer (Name) erfasst.
- Icons werden nicht betrachtet.
- Die Absender E-Mail Adressen (Systemadressen) werden aufbereitet exportiert.

SysConfig-Konfigurationen:

- Ermöglicht den Export von SysConfig-Einstellungen, deren Schlüssel bestimmten Namensmustern (wildcard pattern) entspricht
- Ermöglicht den Export aller geänderten SysConfig-Einstellungen ("modified")



- Der Import verarbeitet Konfigurationsdateien (JSON), die zuvor durch den Export erstellt wurden. Nach dem Import ist die geänderte Konfiguration aktiv.
- Bei Verwendung des Modus "ForceAdd" werden existierende Konfigurationen ignoriert, da Konfigurationen nicht kopiert und mit einem vorangestellten "Copy_" versehen werden können.
- **Wichtig:** Beim Import wird die Anwendbarkeit Plug-In-spezifischer Konfigurationen (KIX Connect/ KIX Pro) NICHT überprüft. Die Verantwortung liegt beim ausführenden Administrator.
 - Bei schwerwiegenden Fehlern kann die SysConfig von On-Premises Installationen vollständig auf Standard zurück gesetzt werden. Führen Sie dazu nacheinander die Konsolen-Kommandos: " `Console::Command::Admin::Config::CleanUp` " und " `Console::Command::Maint::Config::Rebuild` " aus. Für Cloud-Umgebungen ist dies nicht möglich.

- Nach dem Import muss das Frontend manuell geladen werden. Navigieren Sie dazu ins Menü *System* > *SysConfig* und klicken Sie in der Übersicht auf "Lade Frontend-Konfigurationen neu".

ObjectIcons

- Dieses Objekt kann kein ForceAdd ausführen bzw. nur teilweise.
 - Ist der Datensatz nicht existent, so wird dieser normal als Add behandelt.
 - Ein Copy ist nicht möglich, da es keinen Namen zum Icon gibt.
- Die Eindeutigkeit wird über die Parameter Object und ObjectID ermittelt.
 - Object: Objektzuweisung für das Icon, wo es angewendet werden soll
 - z. B. TicketState, TicketType, GeneralCatalogItem, MIMEType, usw.
 - ObjectID: Ist die ID des jeweiligen Objekts
- Das heißt, weder an ObjectID noch an Object kann der Prefix "Copy" ergänzt werden, da sonst das Icon nie angezeigt werden kann, weil die Zuweisung fehlerhaft ist.

E-Mail-Filter

- ⚠ Beachten Sie insbesondere bei der Verwendung von IDs, dass zu importierende E-Mail-Header-Werte auf dem Zielsystem nicht plausibilisiert werden!
- Es erfolgt eine Prüfung auf den PostMasterHeader "X-KIX-TicketTemplate".
 - Haben *Match* oder *Set* diesen enthalten, wird der Filter nicht importiert.
 - PostMasterHeader, die das System nicht kennt, werden dem "PostmasterX-Header" ergänzt, sofern es nicht "X-KIX-TicketTemplate" ist.

4.6 Migration KIX 17

Nutzer von KIX 18 Pro können die Daten von KIX 17 nach KIX 18 automatisiert übertragen. Dies erfolgt im Menü *System > Migration KIX 17*. KIX 18 (aktives System) holt sich dazu die Daten vom Quellsystem KIX 17 (passives System) ab. Sie müssen in KIX 18 als Nutzer mit Admin-Rechten angemeldet sein, um die Migration durchführen zu können (Rollen: System Admin, Superuser).

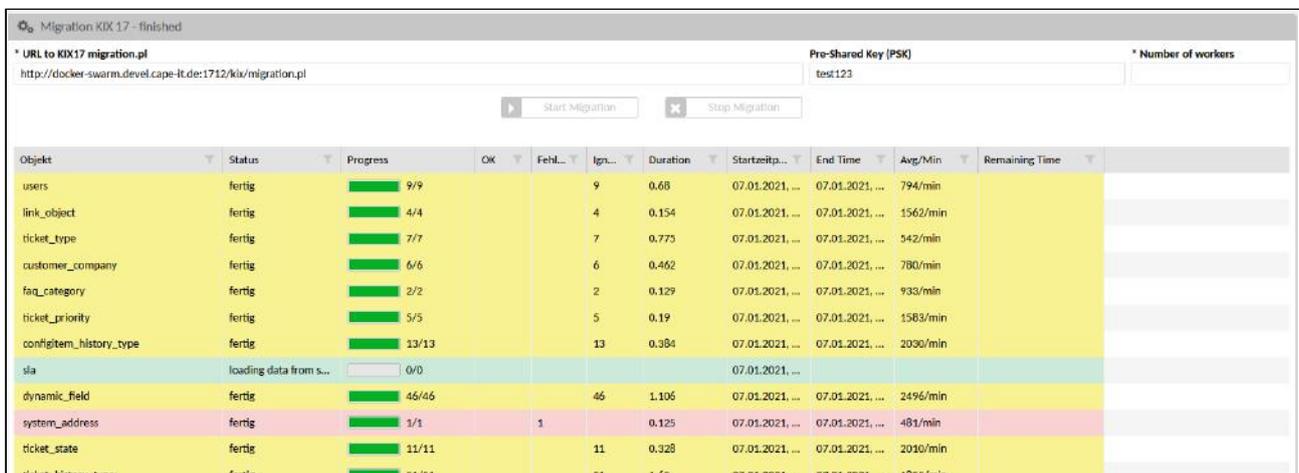
Inhalte auf dieser Seite:

- [Die Migration durchführen](#) (siehe Seite 70)
 - [Schritt 1: PSK-Modus in KIX 17 aktivieren](#) (siehe Seite 70)
 - [Schritt 2: Die Migration starten](#) (siehe Seite 72)
 - [Schritt 3: Die Migration abschließen](#) (siehe Seite 73)
- [Hinweise zur Migration](#) (siehe Seite 75)

Voraussetzung für die Migration sind:

- eine aktuelle Version von KIX 17 (KIX 17.14.0 und höher)
- die Aktivierung des [Pre-Shared Key Modus](#) (siehe Seite 70) in der Konfigurationsdatei von KIX17 mit Angabe des [Pre-Shared Keys](#) (siehe Seite 70) .

Alternativ dazu kann die Migration auch manuell über die Konsole im Menü *System > Konsole* erfolgen. Die Konsole ermöglicht zudem die Migration einzelner Objekte wie Tickets, Organisationen etc. Weiterführende Informationen zur Migration finden Sie im Kapitel Migration KIX 17 > KIX 18 des Start-Handbuches.



Objekt	Status	Progress	OK	Fehl...	Ign...	Duration	Startzeitp...	End Time	Avg/Min	Remaining Time
users	fertig	9/9			9	0.68	07.01.2021, ...	07.01.2021, ...	794/min	
link_object	fertig	4/4			4	0.154	07.01.2021, ...	07.01.2021, ...	1562/min	
ticket_type	fertig	7/7			7	0.775	07.01.2021, ...	07.01.2021, ...	542/min	
customer_company	fertig	6/6			6	0.462	07.01.2021, ...	07.01.2021, ...	780/min	
faq_category	fertig	2/2			2	0.129	07.01.2021, ...	07.01.2021, ...	933/min	
ticket_priority	fertig	5/5			5	0.19	07.01.2021, ...	07.01.2021, ...	1583/min	
configitem_history_type	fertig	13/13			13	0.384	07.01.2021, ...	07.01.2021, ...	2030/min	
sla	loading data from s...	0/0					07.01.2021, ...			
dynamic_field	fertig	46/46			46	1.106	07.01.2021, ...	07.01.2021, ...	2496/min	
system_address	fertig	1/1		1		0.125	07.01.2021, ...	07.01.2021, ...	481/min	
ticket_state	fertig	11/11			11	0.328	07.01.2021, ...	07.01.2021, ...	2010/min	
ticket_history_type	fertig	51/51			51	1.65	07.01.2021, ...	07.01.2021, ...	1855/min	

Abb.: automatisierte Migration von KIX 17 nach KIX 18

4.6.1 Die Migration durchführen

4.6.1.1 Schritt 1: PSK-Modus in KIX 17 aktivieren

KIX 17 und KIX 18 unterstützen einen PSK-Modus (PSK = Pre-Shared-Key). Der PSK ist ein Prüfschlüssel zur Absicherung der Datenübertragung. Er verhindert den ungeschützten bzw. unautorisierten Zugang zur Datenbank. Die Migration erfolgt nur, wenn beiden Systemen dieser vorher vereinbarte Schlüssel bekannt ist. KIX17 weist Anfragen zurück wenn kein oder ein falscher PSK übermittelt wird.

Der PSK wird in der Konfigurationsdatei (Config.pm) von KIX 17 hinterlegt und der Migration als Parameter in KIX 18 mitgegeben.

Achtung

Ist der PSK-Modus aktiviert, ist das System für den Zugriff von Außen geöffnet. Unbefugte Dritte können sich Zugang zur Datenbank verschaffen, wenn ihnen der Schlüssel bekannt ist. Für die Benennung des Schlüssels sind daher die Konventionen zur Vergabe eines sicheren Passwortes bindend!

Löschen Sie nach Abschluss der Migration den Schlüssel in beiden Systemen und deaktivieren Sie den PSK-Modus, um den Zugriff von Außen zu unterbinden!

So aktivieren Sie den PSK-Modus:

1. Öffnen Sie die Konfigurationsdatei im Dateisystem von KIX 17. Diese finden Sie unter: `/opt/kix/Kernel/Config.pm`.
2. Suchen Sie den Abschnitt `"insert your own config settings "here" "`.
3. Fügen Sie innerhalb des Abschnitts folgende Codezeilen ein. Entfernen Sie die Kommentare (`"// PSK[...]"`). Vergeben Sie statt `"my_PSK_Password"` ein eigenes, sicheres Passwort und speichern Sie Ihre Änderungen.

```
$Self->{'Migration::Active'} = 1;           //PSK-Modus aktivieren  
$Self->{'Migration::PSK'} = 'my_PSK_Password'; //PSK benennen
```

```
# The database DSN
$self->{DatabaseDSN} = "DBI:Pg:dbname=$self->{Database};host=$self->{DatabaseHost}";

# ----- #
# insert your own config settings "here" #
# config settings taken from Kernel/Config/Defaults.pm #
# ----- #
$self->{CheckMXRecord} = 0;
$self->{FQDN} = 'kalliope';
$self->{'Migration::Active'} = 1;
$self->{'Migration::PSK'} = 'my_PSK_Password';

# ----- #
# data inserted by installer #
# ----- #
# SDIBIS
```

4.6.1.2 Schritt 2: Die Migration starten

1. Navigieren Sie in KIX 18 Pro zum Menü *System > Migration KIX 17*. Im Contentbereich wird ein Formular geöffnet, in welches Sie die erforderlichen Parameter eintragen können.
2. Tragen Sie folgende Parameter ein:

Feld	Beschreibung
URL zu KIX17 migration.pl	<p>Tragen Sie die URL der Quelle (KIX 17) ein, z. B. "http://myKIX17domain.de/kix/migration.pl". Die Angabe der Domain (z. B. "myKIX17domain.de") ist ausreichend. KIX ergänzt die URL um "http://" und "/kix/migration.pl".</p> <p>Die URL können Sie der Adresszeile im Browser Ihres geöffneten KIX17-Systems entnehmen: z. B. <code>http://192.168.160.200:4711/kix/index.pl</code> oder <code>https://myKIX17domain.de/kix/index.pl</code></p> <p>Die Angabe der URL entspricht der Angabe der <code>"--source-id"</code> in der Konsole. Möchten Sie die durchgeführte Migration anschließend über die Konsole bereinigen, beachten Sie bitte die Hinweise im Starthandbuch (Kapitel "Installation > Migration KIX 17 > KIX 18")</p>
Pre-Shared Key (PSK)	Tragen Sie den in KIX 17 hinterlegten Wert ein, z. B. "start123".

Feld	Beschreibung
Anzahl Worker	<p>Anzahl der parallelen Prozesse, die für die Massenobjekte wie Tickets usw. verwendet werden sollen. Standard: 1</p> <p>Mit der Anzahl der Worker bestimmen Sie die zur Verfügung stehende Systemleistung. Dies hat Einfluss auf die Performance Ihres Servers. Je höher die Anzahl der Worker desto schneller der Migrationsprozess bei stärkerer Belastung des Systems. Vermeiden Sie eine Überlastung Ihres Systems!</p> <p>i Bereits 4 Worker können einen Host zu 100% auslasten. Daher sind KIX.Cloud-Umgebungen auf max. 2 Worker begrenzt.</p>

 Migration KIX 17

<p>* URL zu KIX17 migration.pl http://myKIX17domain.de:4711/kix/migration.pl</p>	<p>Pre-Shared Key (PSK) start123</p>	<p>* Anzahl Worker 1</p>
---	---	-------------------------------------




3. Starten Sie die Migration mit einem Klick auf "Start Migration". Die Tabelle zeigt den Status, Fortschritt und eventuelle Fehler der Übertragung pro Objekt an.
4. Nach erfolgreicher Übertragung sind die Daten aus KIX 17 in KIX 18 enthalten.

Übersicht Assets (501) Filtern in Liste							
<input type="checkbox"/>	A#	Name			Klasse	Geändert am	Geändert...
<input type="checkbox"/>	1722000001	Computer 1001			Migration-Computer	23.11.2020, 08:17	Admin KIX
<input type="checkbox"/>	1722000002	Computer 1002			Migration-Computer	23.11.2020, 08:17	Admin KIX
<input type="checkbox"/>	1722000003	Computer 1003			Migration-Computer	23.11.2020, 08:17	Admin KIX

4.6.1.3 Schritt 3: Die Migration abschließen

1. Pre-Shared Key Modus deaktivieren

Bei aktiviertem PSK-Modus können sich **unbefugte Dritte** Zugang zur Datenbank verschaffen, sobald sie in Besitz des Schlüssels gelangen. Es ist daher zwingend erforderlich, nach Abschluss der Migration (oder nach Abbruch) den Zugriff auf die Datenbank wieder zu unterbinden!

Löschen Sie dazu den Schlüssel und deaktivieren Sie den Pre-Shared Modus wie folgt:

1. Öffnen Sie die Konfigurationsdatei im Dateisystem von KIX 17. Diese finden Sie unter: `/opt/kix/Kernel/Config.pm`.

2. Suchen Sie den Abschnitt "`insert your own config settings "here"`".
 3. Löschen Sie innerhalb des Abschnitts die von Ihnen eingefügten Codezeilen und speichern Sie diese Änderung.
2. **Überprüfen und Bearbeiten der migrierten Daten**
- Überprüfen Sie die übertragenen Daten, Platzhalter, Signaturen etc. und passen Sie diese ggf. an. Sie können diese einzeln manuell oder durch Anwendung eines Jobs automatisiert bearbeiten.
- Die migrierten Daten werden mit einem Migrations-Präfix gekennzeichnet, wenn im Zielsystem bereits eine gleichnamiges Objekt existiert, z. B. Klasse "**Migration-Computer**, **Migration-Nutzername**".

4.6.2 Hinweise zur Migration

Migration abbrechen:

Sie können die Migration unterbrechen. Klicken Sie dazu auf "Stop Migration". Die Hintergrundprozesse werden beendet. Dies kann je nach Migrationsfortschritt etwas Zeit beanspruchen. Beim erneuten Start der Migration werden nur die nicht oder teilweise migrierten Tickets betrachtet.

Fehler bei der Migration

Tritt bei der Migration ein Fehler auf, wird dieser in einer Log-Datei dokumentiert (Menü *System > Logs*). In der Anzeige wird lediglich dokumentiert, dass ein Objekt fehlerhaft war.

Im Falle eines HTTP-Fehlers (z. B. falsche URL) wird der HTTP-Status als Progress-Status angezeigt.

Legende

Die Zeilen sind entsprechend ihres Zustands farbig gekennzeichnet:

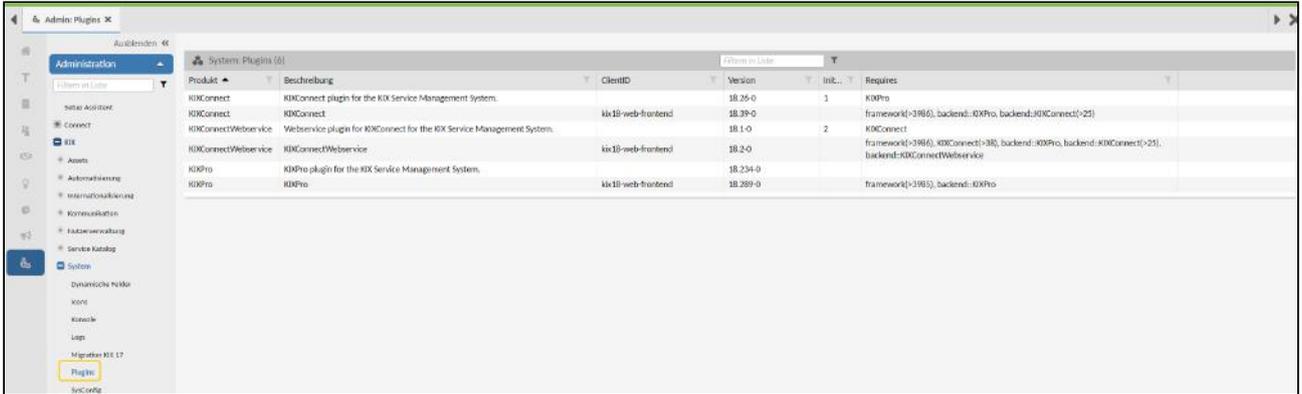
rot	Fehler
gelb	ignoriert
grün	erfolgreich
weiß	wartend

Migrierte Daten

Weiterführende Informationen zu den migrierten Daten finden Sie im KIX 18 Start Admin-Handbuch unter *Installation > Migration KIX 17 > KIX 18*.

4.7 Plugins

Das Menü **KIX > System > Plugins** listet die installierten Plugins auf.



Produkt	Beschreibung	ClientID	Version	Init.	Requires
KIXConnect	KIXConnect plugin for the KIX Service Management System.		18.25-0	1	KIXPro
KIXConnect	KIXConnect	kix18-web-frontend	18.29-0		framework(-3186), backend:KIXPro, backend:KIXConnect(-25)
KIXConnectWebservice	Webservice plugin for KIXConnect for the KIX Service Management System.		18.1-0	2	KIXConnect
KIXConnectWebservice	KIXConnectWebservice	kix18-web-frontend	18.2-0		framework(-3186), KIXConnect(-38), backend:KIXPro, backend:KIXConnect(-25), backend:KIXConnectWebservice
KIXPro	KIXPro plugin for the KIX Service Management System.		18.254-0		
KIXPro	KIXPro	kix18-web-frontend	18.289-0		framework(-3185), backend:KIXPro

Abb.: Übersicht der installierten Plugins



4.8 Single Sign-on

KIX bietet die Möglichkeit der Nutzerauthentifizierung und Autorisierung über Single Sign-on (SSO).

Unterstützt werden aktuell:

- [SSO mit Kerberos \(siehe Seite 78\)](#)
- [SSO mit OpenID Connect \(siehe Seite 83\)](#)

4.8.1 SSO mit Kerberos

Konfigurations schlüssel	Authentication ###00- Kerberos
-------------------------------------	--------------------------------------

Inhalte auf dieser Seite:

- [Anbindung](#) (siehe Seite 78)
- [Die Keytab Datei](#) (siehe Seite 80)
- [Hinweise zu Base64 Codierung](#) (siehe Seite 81)
- [Verwendung / Client-seitige Voraussetzungen](#) (siehe Seite 81)
- [Häufig auftretende Fehlerursachen](#) (siehe Seite 82)
- [Verweise](#) (siehe Seite 82)

Bei Aufruf des Agenten- oder Self Service Portals kann die Authentifizierung des Nutzers durch Kerberos erfolgen. Der Nutzer muss sich somit nicht zwingend mit Nutzernamen und Passwort an KIX

anmelden, wenn er bereits durch sein Login an der Domäne authentifiziert ist (= Single Sign On - Einmalanmeldung).

Voraussetzung dafür ist, dass in Ihrem Unternehmen Kerberos aktiv genutzt wird und ein gegen eine Domäne (z. B. Active Directory/LDAP) authentifizierter Nutzer vorliegt.

Für die Anbindung an KIX benötigen Sie eine **Keytab-Datei**, die Sie von Ihrem Administrator erhalten. Damit KIX die Keytab-Datei direkt in der SysConfig verwenden kann, muss deren Inhalt in Base64-Format hinterlegt werden (empfohlenes Vorgehen). Zum Konvertieren der Keytab -Datei ins Base64-Format können Sie entweder freie Tools wie bspw. base64encode.org³ oder die Linux-Konsole verwenden (s. unten). Alternativ kann die Keytab-Datei in den Backend-Container eingebunden und dann deren Pfad in der SysConfig hinterlegt werden.

4.8.1.1 Anbindung

1. Navigieren Sie zu *System > SysConfig* und öffnen Sie den Konfigurationsschlüssel *Authentication###00-Kerberos*.
2. Kennzeichnen Sie die Authentifizierung als "aktiv" (`Enabled: 1` ,)
3. Hinterlegen Sie unter `Keytab` die ins Base64-Format konvertierte Keytab-Datei.
4. Entfernen oder setzen Sie optionale Parameter (s. unten Abschnitt "Konfiguration")
5. Speichern Sie Ihre Änderungen und klicken Sie auf "Lade Frontendkonfiguration neu".
6. Aktivieren Sie SSO im Frontend
7. Danach kann die Kerberos-Authentifizierung genutzt werden.

³ <http://base64encode.org>

Konfigurationsschlüssel "Authentication###00-Kerberos"

```
[
  {
    "Name": "Kerberos Example",
    "Enabled": 0,
    "Module": "KIXPro::Kernel::System::Auth::Kerberos",
    "Config": {
      "Keytab": "<insert base64 content of keytab file here>",
      "KeytabFile": "(optional)<path to keytab file in filesystem>",
      "Realm": "(optional)<insert your realm here>",
      "krb5.conf": "(optional)<insert base64 content of krb5.conf file here>"
    }
  }
]
```

KIX Konfiguration

Parameter	Beschreibung
Keytab	Inhalt der Keytab-Datei als Base64-String
Keytabfile	Angabe des Dateipfads zur Keytab-Datei, wie sie im Backend-Container eingebunden ist. (Alternative Angabe zu " <code>Keytab</code> ")
Realm	Geben Sie optional den Kerberos Realm an (Name der DNS-Domäne in Großbuchstaben), z.B. @EXAMPLE.ORG Geben Sie den Realm insbesondere an, wenn die Nutzerkennung die Domain enthält, z.B. Nutzernamen: "max.mustermann@example.org". Ohne Angabe des Realm würde die Anmeldung fehl schlagen, da Kerberos den Nutzernamen "max.mustermann" (ohne Domain) zurück liefert.
krb.conf	Kerberos-Konfigurationsdatei als Base64-String (optional)

SSO im Frontend aktivieren

SSO kann für das Agenten- oder das Self Service Portal aktiviert werden. Dazu müssen die jeweiligen Konfigurationen in der environment-Datei Ihrer On-Premise-Konfiguration aktiviert werden. Anschließend ist ein Stack-Neustart erforderlich.

environment

```
SSO_ENABLED=true  
SSO_ENABLED_SSP=true
```

4.8.1.2 Die Keytab Datei

Eine Keytab-Datei ist eine kryptografische, binäre Textdatei, welche die Benutzerkonten inkl. der mit einem Hash verschlüsselten Passwörter enthält. Sie ermöglicht die automatische Anmeldung KIX.

Die Keytab-Datei können Sie bspw. mit dem Kerberos Dienstprogramm *ktpass* erstellen. Informationen hierzu finden Sie bspw. unter: <https://docs.microsoft.com/de-de/windows-server/administration/windows-commands/ktpass>.

Ersetzen Sie im nachfolgenden Beispiel `FQDN`, `DOMAIN`, `USER` und `PASSWORD` durch Ihre entsprechenden Parameter. Beachten Sie dabei Groß- und Kleinschreibung.

Beispiel: Angaben für eine Keytab Datei

```
C:\temp> ktpass -princ HTTP/fqdn@DOMAIN -mapuser USER@DOMAIN -crypto RC4-HMAC-NT  
-ptype KRB5_NT_PRINCIPAL -pass PASSWORD -out c:\temp\kix.keytab
```

⚠ Wichtig

Achten Sie darauf, beim Erstellen der Keytab-Datei einen korrekten Service Principal Namen (SPN) anzugeben, z. B: `HTTP/my.webserver.hostname@DOMAIN.IN.GROSSBUCHSTABEN.ORG`



Besteht die Notwendigkeit mehrere Service Principal Namen (SPN) anzugeben (z.B. wenn "your-kix-agent.example.com" und "your-kix-ssp.example.com" verwendet werden um Agentenportal und SSP unter versch. Hostnamen anzusprechen), werden die Keytabs kombiniert:

Beispiel: Angaben für eine Keytab Datei

```
C:\temp> ktpass -princ HTTP/fqdn@DOMAIN -mapuser USER@DOMAIN -crypto RC4-HMAC-NT  
-ptype KRB5_NT_PRINCIPAL -pass PASSWORD -out c:\temp\kix1.keytab  
C:\temp> ktpass -princ HTTP/fqdn2@DOMAIN -mapuser USER@DOMAIN -crypto RC4-HMAC-NT  
-ptype KRB5_NT_PRINCIPAL -pass PASSWORD -in c:\temp\kix1.keytab -out c:  
\temp\kixAll.keytab
```

4.8.1.3 Hinweise zu Base64 Codierung

Unter **Linux** können Sie dazu folgende Kommandozeile verwenden um die Keytab Base64 zu codieren (Ergebnis in Datei kix.keytab.b64). Der Inhalt kann direkt in Parameter "Keytab" verwendet werden.

Base64-Codierung Keytab File (Linux)

```
SomeLinuxSystem:~# base64 -w 0 ./kix.keytab > ./kix.keytab.b64
```

Unter **Windows** können Sie folgende Kommandozeilen verwenden um die Keytab Base64 zu codieren (Ergebnis in Datei kix.keytab.b64). Für die Verwendung in in Parameter "Keytab" müssen die Zeilenbrüche entfernt werden.

Base64-Codierung Keytab File (Windows)

```
C:\temp> certutil -encode -f kix.keytab kix.keytab.encoded  
C:\temp> type kix.keytab.encoded | find /v "CERTIFICATE" > kix.keytab.b64
```

4.8.1.4 Verwendung / Client-seitige Voraussetzungen

Damit Single Sign On via Firefox funktioniert, muss in diesem die Einstellung "network.negotiate-auth.trusted-uris" den FQDN der KIX-Umgebung oder auch der gesamten Zieldomain beinhalten (z.B. "yourkix.example.com" oder "*.example.com").

Im Edge-Browser muss die Sicherheitsrichtlinie entsprechend konfiguriert werden.

4.8.1.5 Häufig auftretende Fehlerursachen

Sollte SSO mittels Kerberos nicht funktionieren, prüfen Sie zunächst, ob auf dem sich authentifizierenden Windows-Client ein Kerberosticket für den KIX-FQDN vorhanden ist. Verwenden Sie dazu das Kommandozeilentool "*klist*".

Um Problemen vorzubeugen, beachten Sie weiterhin folgende wesentliche Anforderungen beim Einsatz von Kerberos:

- Die Namensauflösung des FQDN muss auch rückwärts funktionieren.
- FQDN muss ein A-Host-Eintrag im DNS sein.
- Systemzeit des KIX (Backends) muss korrekt sein.

4.8.1.6 Verweise

- Kerberos: <https://www.ibm.com/docs/de/was/9.0.5?topic=mechanism-kerberos-krb5-authentication-support-security>
- Service principal name (SPN): <https://www.ibm.com/docs/de/was/9.0.5?topic=server-creating-kerberos-service-principal-name-keytab-file>
- Kerberos Realm: https://web.mit.edu/kerberos/krb5-1.12/doc/admin/realms_config.html

4.8.2 SSO mit OpenID Connect

OpenID Connect (OIDC) ist ein offenes Authentifizierungsprotokoll auf Basis von OAuth2. Es wird von verschiedenen Systemen für die Authentifizierung und Autorisierung der Nutzer verwendet. KIX unterstützt eine Reihe von Identification Providern (IDP), die auf Basis von OpenID Connect arbeiten, z. B.

- GitLab
- Keycloak
- Microsoft Active Directory Federation Service (AD FS)
- Microsoft Entra/Azure.

Somit können berechnigte Nutzer mit den bei einem Identity Provider (IDP) zentral abgelegten Zugangsdaten auf KIX zugreifen, ohne sich mit eigenen Zugangsdaten zusätzlich an KIX anmelden zu müssen. Die Nutzer müssen in KIX bereits als Nutzer für den jeweiligen Kontext (Agentenportal/Self Service Portal) angelegt sein.

Entsprechend der Konfiguration steht im Login-Fenster des Agenten- und/oder Self Service Portals ein separater Button zur Verfügung, mit dem sich der Nutzer via OpenID Connect an KIX anmelden kann. Nach Klick auf den Button erfolgt eine Weiterleitung zum Identity Provider, um die notwendigen Zugangsdaten für KIX abzufragen und den Zugang zu gewähren.

Inhalte auf dieser Seite:

- [1. Einrichtung Identity Provider \(IDP\) \(siehe Seite 84\)](#)
- [2. Einrichtung OAuth2 Profil \(siehe Seite 84\)](#)
 - [Parameter \(siehe Seite 85\)](#)
- [3. Einrichtung Authentifizierungs-Backend \(siehe Seite 87\)](#)
 - [Parameter \(siehe Seite 88\)](#)



The screenshot shows the KIX login interface. At the top left is the KIX logo. The main heading is 'Willkommen bei KIX'. Below this are two input fields: 'Nutzername:' and 'Passwort:'. The password field has a toggle icon for visibility. Below the input fields is a green button with a checkmark and the text 'Anmelden'. At the bottom of the login area is a dark grey button labeled 'OIDC Log-in', which is being pointed to by a mouse cursor. At the very bottom of the page, there is a footer: 'Ein Produkt der KIX Service Software GmbH | Impressum'.

Abb.: SSO mit OpenID Connect

⚠ Hinweise

- Aktuell erfolgt nur eine Authentifizierung durch den IDP. In KIX muss dafür bereits ein Nutzer mit entsprechendem Zugang und Rechten gepflegt sein.
- Ein Datenabgleich der Kontaktdaten, oder die Pflege von Zugängen und Berechtigungen ist bisher NICHT implementiert.
- Sowohl der Browser des Nutzers als auch das KIX-Backend müssen mit dem IDP kommunizieren können.
Gegebenenfalls muss der Proxy im SysConfig-Schlüssel "WebUserAgent::Proxy" konfiguriert werden.



The screenshot shows a configuration window with the following fields:

- Name:** WebUserAgent::Proxy
- Beschreibung:** Defines the connections for http/ftp, via a proxy.
- Gültigkeit:** gültig
- Wert:** http://proxy.example.com:3128/
- Vorgabewert:** (empty)

4.8.2.1 1. Einrichtung Identity Provider (IDP)

Die Einrichtung der Applikation seitens Identification Provider obliegt Ihnen und Ihrem Unternehmen. Informationen dazu finden Sie unter anderem hier:

- Keycloak: https://www.keycloak.org/docs/latest/server_admin/
- GitLab: https://docs.gitlab.com/ee/integration/openid_connect_provider.html
- Microsoft Entra / Azure: <https://learn.microsoft.com/de-de/entra/identity-platform/v2-protocols-oidc>
- Microsoft Active Directory Federation Services 2019 (AD FS): <https://learn.microsoft.com/de-de/windows-server/identity/ad-fs/development/ad-fs-openid-connect-oauth-concepts>

4.8.2.2 2. Einrichtung OAuth2 Profil

Menü	System > OAuth2 Profile
-------------	-------------------------

Richten Sie zunächst ein OAuth2 Profil ein und hinterlegen Sie die vom Identity Provider bereitgestellten Verbindungsparameter.

Weiterführende Informationen s. auch: OAuth2 Profile

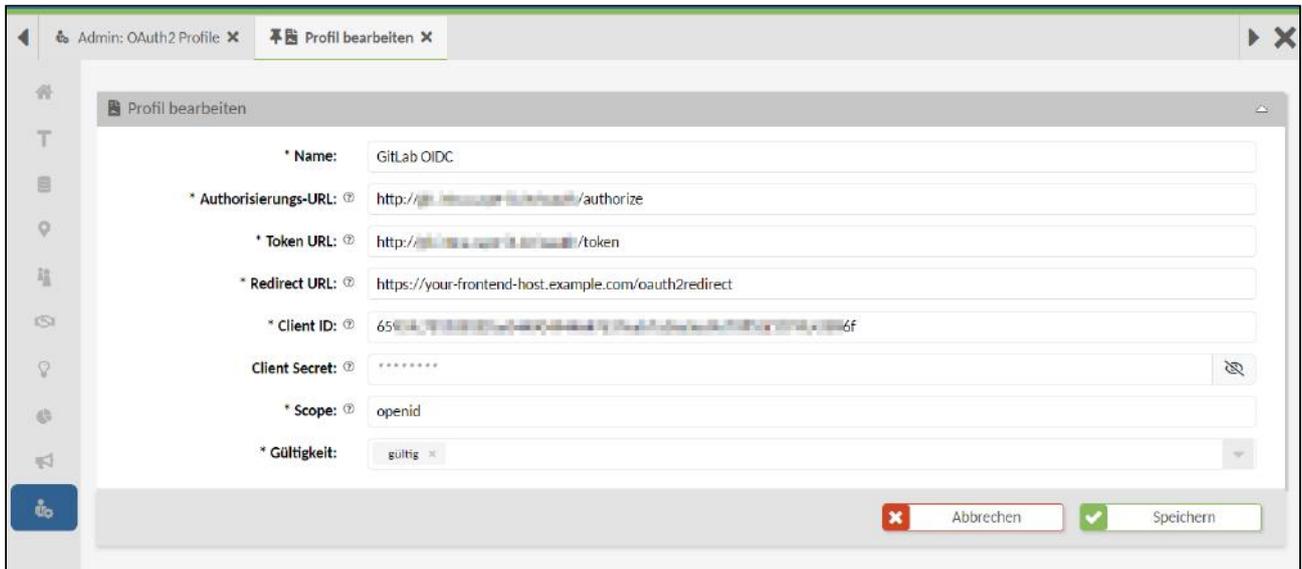


Abb.: Einrichtung eines OIDC Profils

Parameter

Parameter	Beschreibung	Beispiel
Name	Aussagekräftiger Name des Profils. Der Name des Profils wird in der Konfiguration der Authentifizierungsmodule benötigt (s. unten)	GitLab OIDC
Autorisierungs-URL	Tragen Sie die URL für die OIDC Autorisierung ein. Sie wird zum Starten der Anfrage verwendet.	http://your-git-path.example.com/oauth/authorize
Token URL	Tragen Sie die URL für den OIDC Token ein. Sie wird für den Abruf des ID-Tokens durch das Backend verwendet.	http://your-git-path.example.com/oauth/token

Parameter	Beschreibung	Beispiel
Redirect URL	<p>Für die Verwendung mit OIDC ist diese Angabe nicht relevant, da Agenten Portal und Self Service Portal die URLs eigenständig im Authentifizierungsprozess vorgeben.</p> <p>Wenn Sie OIDC zur Anmeldung ab der API verwenden wollen, können Sie hier die URL zur Authentifizierung am Backend hinterlegen.</p>	<ul style="list-style-type: none"> • https://your-frontend-host.example.com/oauth2redirect • https://your-frontend-host.example.com:47111/api/v1/auth
Client ID	Tragen Sie hier die alphanumerische Client-ID ein.	Azure: die Anwendungs-ID (Client)
Client Secret	Tragen Sie hier das Client-Secret ein.	Azure: der Clientschlüssel aus "Zertifikate & Geheimnisse"
Scope	<p>Die Angabe des Geltungsbereichs hängt vom verwendeten System ab.</p> <p>Der Scope sollte immer die Angabe 'openid' enthalten. Je nach IDP und gewünschten Login-Attribut, können weitere Angaben nötig sein.</p>	<p>openid openid profile openid email</p> <ul style="list-style-type: none"> • Microsoft Entra / Azure: <ul style="list-style-type: none"> • Anmeldung per Email: <ul style="list-style-type: none"> • Scope: "openid email" • Anmeldung per Login-Name: <ul style="list-style-type: none"> • Scope: "openid profile" • Login-Attribute: "name" oder "preferred_username" • KeyCloak: <ul style="list-style-type: none"> • Anmeldung per Email: Scope 'openid email'
Gültigkeit	<ul style="list-style-type: none"> • gültig: Das Profil kann verwendet werden. Nutzer können sich hierüber anmelden. • (temporär) ungültig: Das Profil kann nicht verwendet werden. Nutzer können sich hierüber nicht anmelden. 	

4.8.2.3 3. Einrichtung Authentifizierungs-Backend

Menü	KIX > System > SysConfig
Konfigurationsschlüssel	Authentication###000-OIDC

Richten Sie das Authentifizierungs-Backend ein.

Zuvor muss im Menü "System > OAuth2 Profile" mindestens ein OAuth2-Profil angelegt und eingerichtet worden sein.

Sie können mehrere Backends konfigurieren. Geben Sie für jedes Backend das jeweilige OAuth2-Profil an. Achten Sie darauf, dass zu jedem eingerichteten Backend ein entsprechendes OAuth2-Profil vorliegt und die Angaben korrekt sind. Anderenfalls schlägt der Login fehl.

Nach dem Speichern der Konfiguration und Klick auf "Lade Frontendkonfiguration neu" steht den Nutzern am Login-Fenster des Agenten- und Self Service Portals ein zusätzlicher Button für die Anmeldung via OIDC zur Verfügung.

Beispiel für das Authentifizierungs-Backend

```
[
  {
    "Config": {
      "JWKS": {
        "URI": "http://git.example.de/oauth/discovery/keys"
      },
      "LoginAttribute": "nickname",
      "Profile": "GitLab OIDC",
      "Verify": {
        "Audience": "<ClientID>",
        "Issuer": "http://git.example.de"
      }
    },
    "Enabled": 1,
    "Module": "Auth::OIDC",
    "Name": "OIDC Log-in"
  },
  {
    "Config": {
      "JWKS": {
        "URI": "https://login.microsoftonline.com/alphanumerischer-schlüssel/
discovery/v2.0/keys"
      },
      "LoginAttribute": "email",
      "Profile": "Azure",

```

```

    "Verify": {
      "Audience": "<ClientID>",
      "Issuer": "https://login.microsoftonline.com/alphanumeric-
schlüssel/v2.0"
    }
  },
  "Enabled": 1,
  "Module": "Auth::OIDC",
  "Name": "Azure Log-in"
}
]

```

Parameter

Parameter		Beschreibung	Beispiel
JWKS		JSON Web Key Set	
		Diese Angaben dienen der Verifizierung der Signatur des aufgerufenen ID-Tokens.	
	CacheTTL	(Optional) Angabe der Zeit in Sekunden, um angeforderte Schlüssel von URI im Cache zu behalten.	<ul style="list-style-type: none"> • Wenn nicht angegeben, wird 360 (eine Stunde) verwendet. • Zur Deaktivierung des Caches '0' angeben.
	Key	<p>Tragen Sie den Schlüsselstring zur Verifizierung des ID-Tokens ein.</p> <p>Die Angabe hat Vorrang vor "JWKS_URI".</p> <p>Wird hier ein verschlüsseltes Zertifikat angegeben, kann mit "KeyPass" das Passwort zur Entschlüsselung angegeben werden.</p>	PEM Schlüssel oder JWK (JSON Web Key)

Parameter		Beschreibung	Beispiel
	URI	(Optional) Tragen Sie die URI zur Abfrage gültiger Schlüssel ein.	<ul style="list-style-type: none"> Azure: "https://login.microsoftonline.com/alpha-numerischer-schlüssel/discovery/v2.0/keys" GitLab: "http://git.example.de/oauth/discovery/keys"
LoginAttribute	(Optional) Attribut in ID-Token zur Verwendung als Login		email nickname
Profile	<p>Exakter Name des zu verwendenden OAuth2-Profiles.</p> <p>Das Profil muss zuvor unter "OAuth2 Profile" angelegt worden und gültig sein.</p> <p>Existiert das angegebene Profil nicht oder ist ungültig, dann ist die Authentifizierung nicht möglich. Der Login-Button wird nicht angezeigt.</p> <p>Fehlermeldungen finden Sie im Menü "System > Logs".</p>		GitLab OIDC Azure o.a.
Verify	Angaben zur Verifizierung des ID-Tokens		
	Audience	(optional) Name der Zielgruppe, für die der Token ausgestellt wurde. Verwenden Sie "<ClientID>", um die Kunden-ID aus dem Profil zu verwenden.	<ClientID>

Parameter		Beschreibung	Beispiel
	Expiration	(optional) <ul style="list-style-type: none"> • 0 - deaktiviert die Prüfung • 1 - erfordert Prüfung. Wenn nicht gesetzt, wird geprüft, ob der Claim "exp" vorhanden ist.	0 1
	IssueAt	(optional) <ul style="list-style-type: none"> • 0 - deaktiviert die Prüfung • 1 - erfordert Prüfung. Wenn nicht gesetzt, wird geprüft, ob der Claim "iat" vorhanden ist.	0 1
	Issuer	(optional) Name des Ausstellers	<ul style="list-style-type: none"> • Azure: "https://login.microsoftonline.com/alpha-numerischer-schlüssel/v2.0" • GitLab: http://git.example.de.
	Leeway	(optional) Toleranz in Sekunden Toleranz für IssueAt, NotBefore und Expiration	
	NotBefore	(optional) <ul style="list-style-type: none"> • 0 - deaktiviert die Prüfung • 1 - erfordert Prüfung. Wenn nicht gesetzt, wird geprüft, ob der Claim "nbf" vorhanden ist.	0 1
Enabled		De-/Aktiviert das Authentifizierungsbackend	0 1



Parameter		Beschreibung	Beispiel
Module		Für die Nutzung der OIDC-Authentifizierung ist als Modul "Auth::OIDC" anzugeben.	Auth::OIDC
Name		Name des Authentifizierungs-Backends. Dieser steht dann auf dem Login-Button.	OIDC Log-in
UsageContext		(Optional) Definiert, für welchen Nutzungskontext dieses Authentifizierungs-Backend genutzt werden kann. Ist nichts angegeben, wird es sowohl für das Agentenportal als auch für das Self Service Portal genutzt.	Agent Customer

5 Workflow

Unter dem Menü *Workflow* finden Sie erweiterte Funktionen, mit denen Sie das Verhalten und die Eigenschaften von ticketrelevanten Formularen modifizieren und erweitern können. Somit können Sie Tickets an die unterschiedlichsten Erfordernisse und an Ihren Workflow anpassen.

Aktionen

Im Menü *Aktionen* können Sie Ticketaktionen konfigurieren, welche an bestehenden Tickets und deren Artikeln ausgeführt werden. Aktionen ermöglichen die Bearbeitung von Tickets in Abhängigkeit vom aktuellen Ticketzustand und stellen damit die Funktionen für die prozessorientierte Verwendung von KIX bereit.

KIX unterscheidet zwischen Ticketaktionen und Artikelaktionen. Ticketaktionen werden in der Ticketdetailansicht als Schaltflächen in der Aktionszeile angezeigt und stehen je nach Konfiguration im Agentenportal und/oder im Self Service Portal zur Verfügung. Artikelaktionen stehen als seitliches Label am rechten Seitenrand eines Artikels zur Verfügung. Sie können konfigurieren, unter welchen Voraussetzungen wem welche Aktion zur Verfügung steht.

Rule Sets

Im Menü *Rule Sets* können Sie ein Regelwerk für die dynamische Steuerung von Formularinhalten anlegen. Mit diesem Regelwerk definieren Sie, die Bedingungen und das Verhalten der Formularfelder im Ticketdialog sowie in Vorlagen und Aktionen.

Vorlagen

Mit Vorlagen können Sie Templates anlegen, mit denen die Ticketerstellmasken im Agentenportal und im Self Service Portal zum Zeitpunkt des Bedarfs an den jeweiligen Kontext angepasst werden. In einer Vorlage können Sie festlegen, welche Eingabefelder dargestellt werden und Pflichtangaben sind oder welche Vorgabewerte mit der Vorlage gesetzt werden. Es ist damit möglich, an den Meldungstyp angepasste Eingaben zu machen und häufig auftretende Anfragen schneller zu erfassen und effizienter zu bearbeiten.

Vorlagengruppen

Im Menü *Vorlagengruppen* können Sie Kategorien anlegen, um Vorlagen thematisch zusammen zu fassen. Die Zuordnung einer Vorlage zu einer Vorlagengruppe erfolgt beim Anlegen einer Vorlage. Durch Zuordnung einer Vorlagengruppe zu einer übergeordneten Vorlagengruppe können Sie eine Baumstruktur aus Vorlagengruppen erzeugen.

5.1 Aktionen

Im Menü *Workflow > Aktionen* können Sie ticket- und artikelbasierte Aktionen konfigurieren. Aktionen werden stets auf bestehende Tickets und deren Artikel angewendet. Im Unterschied dazu werden Vorlagen nur auf neue Tickets angewendet.

Aktionen sind individuell konfigurierte Funktionen (Macro Actions), die unter bestimmten Voraussetzungen zur Verfügung stehen. Initiale Aktionen sind bspw. die Standardfunktionen wie das Schließen und Bearbeiten von Tickets oder das Beantworten und Weiterleiten von E-Mails.

Mit dem Anlegen eigener Aktionen und der Festlegung, wem wann welche Aktion zur Verfügung steht, können Sie das System an den Workflow Ihres Unternehmens anpassen und den Agenten genau die Funktionen bereitstellen, die sie für die tägliche Serviceerbringung benötigen.

5.1.1 Arten von Aktionen

KIX unterscheidet je nach referenziertem Bezugsobjekt zwischen Ticketaktionen und Artikelaktionen.

Ticketaktionen

Ticketaktionen sind Aktionen, die auf bestehende Tickets angewendet werden. Sie ermöglichen bspw. die Bearbeitung von Tickets in Abhängigkeit vom aktuellen Ticketstatus und stellen somit die Grundlage für eine prozessorientierte Verwendung von KIX bereit. Die Bearbeitungsmöglichkeiten am Ticket können abhängig sein vom Inhalt, Bearbeitungszustand und dem aufrufenden Nutzer.

Ticketaktionen können dialogbasiert oder nicht-dialogbasiert sein:

- **Nicht-dialogbasierte Aktionen** lösen beim Anklicken eine Hintergrundaktion aus. Mit diesen sog. One-Click-Actions können bspw. Tickets mit vordefinierten Werten gefüllt und/oder sofort geschlossen werden. Nicht-dialogbasierte Aktionen können bspw. sein:
 - Urlaubsantrag genehmigen (Genehmigungsstatus setzen und Ticket schließen)
 - Zurück ins Team Servicedesk verschieben (Team "Servicedesk" setzen)
 - Weiterleiten einer Serviceanfrage inkl. Anhänge an den Hersteller mit gleichzeitigem Schließen des Tickets
 - Als Spam einordnen und ins Junk-Team verschieben (Team "Junk" setzen)
 - Warte 2 Tage (Warten-Status mit Angabe der Zeitdifferenz von 2 Tagen setzen)
 - Warte bis zum nächsten Arbeitstag (Warten-Status setzen mit Angabe der Zeitdifferenz 1 Tag und Zielzeit "Beginn Servicezeit").
- **Dialogbasierte Aktionen** öffnen ein zusätzliches Dialogfenster zur Eingabe von Ticketinformationen. Beispielsweise, um in einem konkreten Störfall weitere Ticketinformationen zu hinterlegen oder zum Eintragen der Sollzeit. Welche Formularfelder der Dialog enthält, können Sie individuell [konfigurieren](#) (siehe Seite 102) .

Ticketaktionen können sowohl für das Agentenportal als auch für das Self Service Portal konfiguriert werden. Sie stehen dann als Schaltflächen der Ticketdetailansicht zur Verfügung.

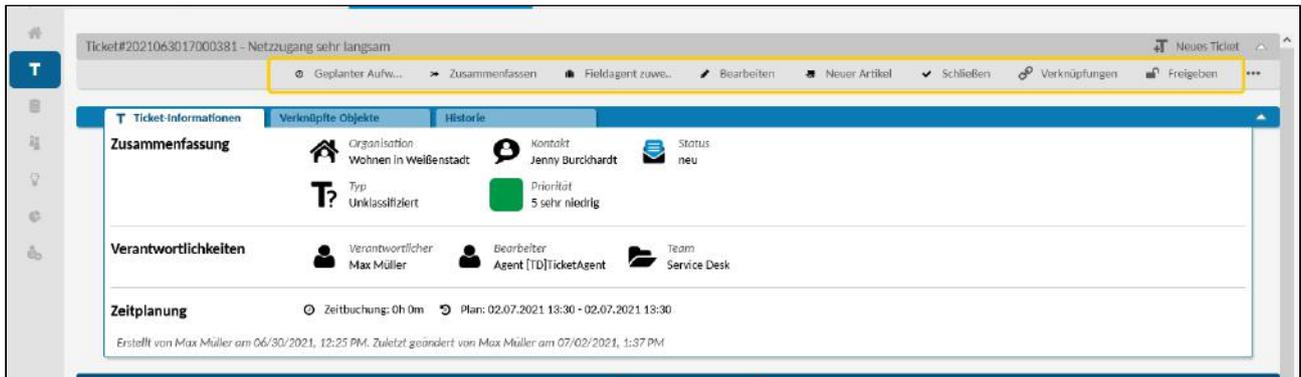


Abb.: Ticketaktionen in der Ticket Detailansicht des Agentenportals



Abb.: Ticketaktionen in der Ticket Detailansicht des Self Service Portals

Damit Ticketaktionen im Self Service Portal verfügbar sind, müssen sie dem Kontext "Kunde" und der Rolle "Customer" zugewiesen werden. Die weitere Konfiguration erfolgt analog der Ticketaktionen für das [Agentenportal](#) (siehe Seite 102) . Weiterführende Informationen zum Self Service Portal: [Self Service Portal](#) (siehe Seite 206) .

5.1.1.1 Artikelaktionen

Artikelaktionen werden immer auf einen bestehenden Artikel angewendet und stehen somit direkt am Artikel zur Verfügung. Zum Beispiel die Aktionen zum Beantworten und Weiterleiten von E-Mails oder zum Teilen des Artikels. Diese Aktionen sind initial vorkonfiguriert und können nach Bedarf angepasst werden. Beispielsweise, um der Antwortmail das betroffene Asset mitzugeben.

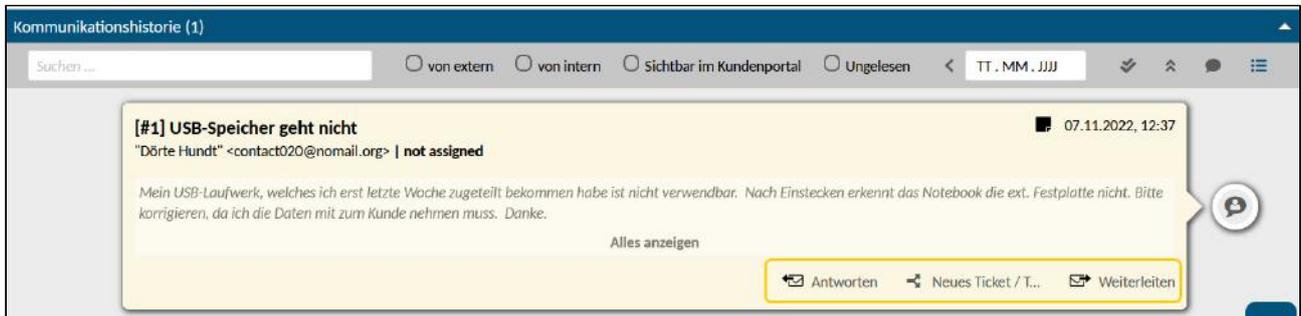


Abb.: Artikelaktionen am Artikel

Am Artikel sind die Aktionen zum Bearbeiten und Löschen eines Artikels vorhanden, jedoch initial deaktiviert (ungültig). Sie können diese Aktionen bei Bedarf aktivieren und nachkonfigurieren, um bspw. Berechtigungen zu definieren.

Sie können weitere Artikelaktionen nach Bedarf anlegen, bspw. um verschiedene Antwortmöglichkeiten für die Weiterleitung oder Beantwortung eines Artikels parat zu haben. Weitere Anwendungsfälle für Artikelaktionen können sein:

- Weiterleiten fehlgeleiteter E-Mails mit Rückmeldung an den Absender mit nur einem Klick
- Weiterleiten von E-Mails an einen externen Dienstleister mit Auswahl der anzuhängenden Informationen bzw. Entfernen von betriebsinterner Informationen sowie gleichzeitigem Schließen des Tickets
- Anlegen mehrerer, unterschiedlicher Weiterleitungs-Vorlagen für verschiedene Dienstleister oder Hersteller. Der Empfänger, betroffener Service/SLA sowie der Status des Tickets sind bereits vorausgewählt, der variable Wartezeitpunkt wird durch den Agenten gesetzt.

Hinweise

- Artikelaktionen, die am Ticket zulässig sind, werden an allen Artikeln angezeigt.
- Artikelaktionen können aktuell nur für das Agentenportal bereitgestellt werden.

5.1.2 Initial ausgelieferte Aktionen in KIX Pro

Aktion	Beschreibung	Hinweise
<p>AppAssignWorkOrder</p>	<p>Die Ticketaktion "Fieldagent zuweisen" öffnet einen Dialog, um dem Ticket eine Arbeitsbeschreibung inkl. Checklisten beizufügen. Diese Arbeitsbeschreibung kann vom Fieldagent in der Field Agent App eingesehen und abgearbeitet werden.</p> <p>Der Dialog enthält initial folgende Dynamische Felder:</p> <ul style="list-style-type: none"> • RiskAssumptionRemark (Gefahrenübergang) • WorkOrder (Arbeitsanweisung) • MobileProcessingChecklist010 (Checkliste 01) • MobileProcessingChecklist020 (Checkliste 02) <p>Hinweis: Erfordert mindestens Field Agent App Version r1.2.x !</p>	<ul style="list-style-type: none"> • Sie können weitere Dynamische Felder in den Dialog integrieren und nicht benötigte Dynamische Felder entfernen. • Sie können die Checklisten nach Bedarf anpassen. Ändern Sie diese direkt in der Konfiguration des jeweiligen Dynamischen Feldes.
<p>ArticleDelete</p>	<p>Mit der Artikelaktion "Löschen" können einzelne Artikel vom Typ <u>Notiz</u> gelöscht werden, bspw. um redundante oder Spam-Artikel zu löschen.</p> <p>Das Löschen kann nicht rückgängig gemacht werden. Daher wird eine Sicherheitsabfrage angezeigt.</p>	<ul style="list-style-type: none"> • Die Aktion ist initial deaktiviert (ungültig). Zum Aktivieren öffnen Sie die Aktion und setzen Sie sie auf "gültig". Danach steht die Aktion am Artikel zur Verfügung. • Sie können die Konfiguration der Aktion ändern, bspw. um den Kreis der Berechtigten einzuschränken.

Aktion	Beschreibung	Hinweise
ArticleForward	<p>Artikelaktion zum "Weiterleiten" von E-Mails.</p> <p>Die Konfiguration der Aktion legt fest, welche Informationen beim Weiterleiten mit versendet werden.</p>	<p>Sie können die Aktion nachkonfigurieren, z. B.:</p> <ul style="list-style-type: none"> • zusätzliche (Dynamische) Felder und Informationen in die E-Mail integrieren. • den Text der E-Mail mit weiteren Inhalten ergänzen und dabei Platzhalter verwenden. • das Versenden von Artikel-Anhängen steuern.
ArticleReply	<p>Artikelaktion zum "Beantworten" von E-Mails.</p> <p>Die Konfiguration der Aktion legt fest, welche Informationen beim Beantworten einer E-Mail versendet werden.</p>	<p>Sie können die Aktion nachkonfigurieren, z. B.:</p> <ul style="list-style-type: none"> • zusätzliche (Dynamische) Felder und Informationen in die Antwort integrieren. • den Text der E-Mail mit weiteren Inhalten ergänzen und dabei Platzhalter verwenden.
ArticleUpdate	<p>Mit der Artikelaktion "Bearbeiten" können einzelne Artikel vom Typ <u>Notiz</u> nachträglich bearbeitet werden, bspw. zum Korrigieren oder Ergänzen von Angaben,</p>	<ul style="list-style-type: none"> • Die Aktion ist initial deaktiviert (ungültig). Zum Aktivieren öffnen Sie die Aktion und setzen Sie sie auf "gültig". Danach steht die Aktion am Artikel zur Verfügung. • Sie können die Konfiguration der Aktion ändern, bspw. um den Kreis der Berechtigten einzuschränken.

Aktion	Beschreibung	Hinweise
Customer Feedback	<p>Ticketaktion, um ein Feedback abzugeben.</p> <p>Wird im Self Service Portal an geschlossenen Tickets bereitgestellt. Kunden können darüber ein Feedback zu einem Ticket abgeben.</p> <p>Verwendet die initialen Dynamischen Felder "SatisfactionPoints" und "SatisfactionRemark".</p>	<p>Sie können die Aktion nachkonfigurieren, bspw. um zusätzliche (Dynamische) Felder und Informationen in die Bewertung zu integrieren.</p>
Follow Up Inquiry	<p>Wird für das Self Service Portal benötigt, um einer Anfrage Informationen hinzuzufügen oder auf eine gestellte Frage zu antworten.</p>	<p>Sie können die Konfiguration ändern.</p>

Aktion	Beschreibung	Hinweise
Merge	<p>Die Ticketaktion "Zusammenfassen" öffnet einen Dialog zum Zusammenfassen zweier Tickets zu einem Ticket. Bspw. um eine fehlgeleitete E-Mail (Quelle) an ein Ticket (Ziel) anzuhängen.</p> <p>Im Dialog eingebunden ist das Dynamische Feld "MergeToTicket". Es referenziert auf alle Tickets, von denen eins als Ziel ausgewählt werden kann und nimmt die ID des gewählten Tickets als Wert auf.</p> <p>Mit dem Speichern der Aktion wird der initiale Job "TicketMerge" ausgelöst, welcher</p> <ul style="list-style-type: none"> • das Quellticket in den Status "merged" setzt • das Event "TicketMerge" am Quellticket auslöst • die in der Aktion konfigurierten Eigenschaften ans Zielticket überträgt • den in der Aktion angelegten Artikel am Zielticket speichert (Kanalauswahl) <p>Besitzt ein Ticket den Status "merged", dann werden an diesem Ticket keine admin-konfigurierbaren Ticket- und Artikelaktionen angezeigt.</p>	<ul style="list-style-type: none"> • Sie können die Aktion nachkonfigurieren und die Ticketeigenschaften angeben, die vom Job "TicketMerge" ans Zielticket übertragen werden sollen. • Sie können weitere Rollen und Filter angeben, um die Verfügbarkeit der Aktion einzuschränken. Die Aktion kann damit nur einem bestimmten Nutzerkreis und/oder unter bestimmten Voraussetzungen zur Verfügung gestellt werden. • Sie können Pre- und Post-Actions definieren, die vor bzw. nach der Aktion ausgeführt werden.

Aktion	Beschreibung	Hinweise
New Ticket/ Split	<p>Über diese Artikelaktion können Agenten aus dem Artikel eines Quelltickets heraus ein neues Ticket erstellen, bspw. um Teilaufgaben zu extrahieren und den jeweiligen Teams zuzuweisen.</p> <p>Sofern nicht anders konfiguriert, werden für das Kind-Ticket die dem Quellticket zugrunde liegende Ticket-Vorlage verwendet und die Artikelinhalte und Anhänge übertragen. Dies können Sie im Schritt 3 der Konfiguration ("Eingabefelder") individuell anpassen.</p>	<ul style="list-style-type: none"> Die im Elternticket gesetzte "Pending time" wird nicht übernommen und muss manuell gesetzt werden.
Planned Effort	<p>Die Ticketaktion "Geplanter Aufwand" öffnet einen Dialog zum Eintragen und Anzeigen der Sollzeit.</p> <p>Die Sollzeit bildet die Basis für die Zeitberechnung am Ticket. Von der Sollzeit wird die Summe der im Ticket gebuchten Zeiten subtrahiert. Die daraus resultierende Zeitdifferenz wird in der Sidebar angezeigt.</p> <p>Der Dialog enthält das Dynamische Feld "PlannedEffort" und zeigt den darin gespeicherten Wert an. Initial wird dieser Wert durch die Jobs "Auto Set Planned Effort (Incident)" und "Auto Set Planned Effort (Service Request)" gesetzt. Sie können die Jobs deaktivieren, wenn die Zeit nicht automatisch gesetzt werden soll.</p> <p>Bei Verwendung der Aktion kann der im Feld gesetzte Wert durch den Ticketverantwortlichen geändert oder eingetragen werden.</p>	<p>Sie können die Konfiguration der Aktion ändern, bspw. um</p> <ul style="list-style-type: none"> weitere Eingabefelder in den Dialog zu integrieren die Verwendung der Aktion durch Setzen weiterer Filter einschränken u.a.m. <p>Hinweis: Entfernen Sie nicht das Eingabefeld "Planned Effort (min)" in Schritt 4 des Assistenten! Anderenfalls kann keine korrekte Zeitberechnung erfolgen.</p>

Aktion	Beschreibung	Hinweise
Ticket Close	<p>Die Aktion "Schließen" öffnet einen Dialog zum Schließen eines Tickets.</p> <p>Die Konfiguration definiert, welche Felder im Dialog enthalten sind und welche Werte beim Schließen des Tickets gesetzt werden.</p> <p>Die initiale Konfiguration enthält das Dynamische Feld "CloseCode", welches die Auswahlliste "Abschlusscode" bereitstellt.</p>	<p>Sie können die Konfiguration ändern, bspw. um weitere Felder in den Ticket-Schließen-Dialog zu integrieren.</p>
Ticket Edit	<p>Die Aktion "Bearbeiten" öffnet einen Dialog zum Bearbeiten eines Tickets.</p> <p>Der Dialog initialisiert die Felder "Bearbeiter" und "Verantwortlicher" nicht, d. h. die Felder werden nach dem Öffnen des Dialogs leer angezeigt. Der am Ticket gesetzte Bearbeiter/ Verantwortliche bleibt jedoch erhalten, sofern er beim Bearbeiten des Tickets nicht geändert wird.</p> <p>Anhand der initial gesetzten Filter, steht die Ticket-Bearbeiten-Aktion nur dann am Ticket zur Verfügung, wenn</p> <ul style="list-style-type: none"> • das Ticket nicht gesperrt ist • das Ticket gesperrt und der ausführende Nutzer gleich dem Bearbeiter (Ticket.Owner) oder dem Verantwortlichen des Tickets (Ticket.Responsible) ist. 	<p>Sie können die Konfiguration ändern, bspw. um die Filterbedingungen für die Bereitstellung der Aktion festzulegen.</p> <p>Hinweis: Die Konfiguration in der Aktion hat Vorrang gegenüber der Konfiguration in der SysConfig. Der SysConfig-Schlüssel "<i>ticket-edit-form-group-data</i>" dient als Fallback.</p>
Ticket New Article	<p>Ticketaktion "Neu" öffnet einen Dialog zum Anlegen eines neuen Artikels.</p>	<p>Sie können die Konfiguration ändern, bspw. um weitere Eingabefelder in den Dialog "Neuer Artikel" zu integrieren.</p>

5.1.3 Aktionen anlegen und konfigurieren

Sie können eigene Ticket- und Artikelaktionen anlegen und konfigurieren. Dabei können Sie festlegen, unter welchen Voraussetzungen welche Aktion wem zur Verfügung steht. Zum Beispiel:

- Der Verantwortliche eines Tickets muss identisch mit dem angemeldeten Benutzer sein.
- Bestimmte Assets müssen betroffen sein.
- Das Ticket muss einem bestimmten Kunden zugeordnet sein.
- u.v.a.m.

Inhalte auf dieser Seite:

- [Aktion anlegen, bearbeiten, duplizieren und löschen](#) (siehe Seite 102)
- [Konfiguration einer Aktion](#) (siehe Seite 103)
 - [Aktionsinformationen](#) (siehe Seite 104)
 - [Filter](#) (siehe Seite 108)
 - [Vorgelagerte Aktionen](#) (siehe Seite 111)
 - [Eingabefelder](#) (siehe Seite 112)
 - [Nachgelagerte Aktionen](#) (siehe Seite 120)

Zudem können Sie die initialen Standard-Aktionen wie bspw. "Neuer Artikel", "Bearbeiten", "Schließen", "Weiterleiten" und "Antworten" individuell nachkonfigurieren.

5.1.3.1 Aktion anlegen, bearbeiten, duplizieren und löschen

So legen Sie eine Aktion an:

1. Navigieren Sie im Explorer zu *Workflow > Aktionen*. Im Contentbereich wird eine Tabelle geöffnet, welche alle im System angelegten Aktionen auflistet.
2. Klicken Sie in der Tabelle auf "Neue Aktion". Es wird ein Formular-Dialog geöffnet, in dem Sie schrittweise eine neue Aktion anlegen und konfigurieren können (s. unten).
3. Speichern Sie die Aktion abschließend mit "Speichern". Die Aktion steht nun in den entsprechenden Ticketoberflächen zur Verfügung.

So bearbeiten Sie eine Aktion:

1. Navigieren Sie im Explorer zu *Workflow > Aktionen*. Im Contentbereich wird eine Tabelle geöffnet, welche alle im System angelegten Aktionen auflistet.
2. Klicken Sie in der Tabelle die zu bearbeitende Aktion an. Es wird ein Formular-Dialog geöffnet, in dem Sie schrittweise die Konfiguration der Aktion bearbeiten können (s. unten).
3. Speichern Sie die Aktion abschließend mit "Speichern". Die geänderte Aktion steht nun in den entsprechenden Ticketoberflächen zur Verfügung.

So duplizieren Sie eine Aktion an:

1. Navigieren Sie im Explorer zu *Workflow > Aktionen*. Im Contentbereich wird eine Tabelle geöffnet, welche alle im System angelegten Aktionen auflistet.
2. Markieren Sie die zu duplizierende Aktion mit einem Häkchen.
3. Klicken Sie in der Tabelle auf "Duplizieren". Es wird ein eigenständiges Duplikat der gewählten Aktion angelegt und geöffnet.
Die Konfiguration des Duplikats entspricht seiner Quelle. Sie können die Konfiguration des Duplikats nach Bedarf ändern (s. unten - Konfiguration einer Aktion).
Der Name des Duplikats lautet standardmäßig "Kopie von [Name der Aktion]". Sie können diesen Namen beibehalten oder das Duplikat anders benennen.
4. Speichern Sie die Aktion abschließend mit "Speichern". Die Aktion steht nun in den entsprechenden Ticketoberflächen zur Verfügung.

So löschen Sie eine Aktion:

1. Navigieren Sie im Explorer zu *Workflow > Aktionen*. Im Contentbereich wird eine Tabelle geöffnet, welche alle im System angelegten Aktionen auflistet.
2. Wählen Sie in der Tabelle die Aktion aus, die gelöscht werden soll. Setzen Sie ein Häkchen an dieser Aktion. Sie können mehrere Aktionen auswählen.
3. Klicken Sie im Tabellenkopf auf die Schaltfläche "Entfernen" und beantworten Sie die Sicherheitsabfrage mit "Ja", wenn Sie sicher sind, dass Sie die Aktion löschen möchten.
Es werden rekursiv alle Macros und Sub-Macros (Pre- und Post-Actions) gelöscht. Sofern jedoch noch interne Referenzen bestehen, kann die Aktion nicht gelöscht werden. Sie erhalten dann einen entsprechenden Hinweis.

Alternativ zum Löschen können Sie eine Aktion auf "ungültig" oder "temporär ungültig" setzen. Die Aktion steht dann in den Ticketoberflächen nicht zur Verfügung. Eine ungültige Aktion kann jederzeit wieder auf "gültig" gesetzt werden.

Tipp

Mit dem Konsolen-Kommando

`Console::Command::Maint::Automation::DeleteNotReferencedMacros` können Sie nicht verwendete Macros löschen (Menü *System > Konsole*). Dabei werden alle Macros gelöscht, die von keiner Pre- und Post-Action referenziert sind und die keine Sub-Macros von anderen Macros sind. Einzelne Macros können vom Löschen ausgeschlossen werden. Geben Sie dazu - ggf. mehrfach - den Parameter `ignore-macro-id` an.

5.1.3.2 Konfiguration einer Aktion

Das Anlegen und Bearbeiten einer Aktion erfolgt schrittweise. Um zum jeweils nächsten Schritt zu gelangen oder um zwischen den einzelnen Schritten zu wechseln, klicken Sie bitte auf die kleinen blauen Pfeilschaltflächen oder auf die dazwischen befindlichen blauen Punkte. Klicken Sie bitte erst zum Schluss auf die "Speichern"-Schaltfläche, wenn Sie alle Parameter eingetragen haben.

Aktionsinformationen

Aktion bearbeiten

Aktionsinformationen

* Nutzungskontext:

* Referenz Objekt:

* Name:

Label:

Icon:

* Verhalten:

Rang:

Nutzereingabe erforderlich:

Kommentar:

* Gültigkeit:

Feld	Beschreibung
Nutzungskontext	<p>Mit dem gewählten Nutzungskontext definieren Sie, in welchem Portal die Aktion zur Verfügung steht.</p> <ul style="list-style-type: none">• Agent: Die Aktion steht im Agentenportal zur Verfügung• Kunde: Die Aktion steht im Self Service Portal zur Verfügung. Zusätzlich wird die Rolle "Customer" benötigt. <p>Soll eine Aktion in beiden Portalen verfügbar sein, so duplizieren Sie eine fertig konfigurierte Aktion und passen Sie im Duplikat den Nutzungskontext und die Rollen entsprechend an.</p> <p>Hinweis: Ab Version 29 kann als Nutzungskontext nur noch Kunde ODER Agent gewählt werden. Für bereits bestehende Aktionen heißt das: Wird eine Aktion bearbeitet, die einst für beide Nutzungskontexte angelegt wurde, wird nur der Nutzungskontext Agent übernommen. Duplizieren Sie daher die Aktion(en) und setzen Sie im Duplikat den Nutzungskontext "Kunde" sowie die Rolle "Customer".</p>

Feld	Beschreibung
Referenz Objekt	<p>Wählen Sie aus, ob die Aktion auf ein Ticket oder auf einen Artikel angewendet werden soll.</p> <ul style="list-style-type: none">• Ticket: Legt eine Ticketaktion an. Ticketaktionen ändern und ergänzen Eigenschaften am Ticket.• Artikel: Legt eine Artikelaktion an. Artikelaktionen ändern und ergänzen Eigenschaften am Artikel. Artikelaktionen werden aktuell nicht vom Self Service Portal unterstützt. Daher ist diese Option nur im Verwendungskontext "Agent" verfügbar.
Name	Interne Bezeichnung für die Aktion. Unter dieser Bezeichnung wird die Aktion im System geführt.
Label	Beschriftung der Schaltfläche in der Programmoberfläche.
Icon	Wählen Sie optional ein Icon aus, mit dem die Aktion grafisch gekennzeichnet wird. Das Icon wird auf der Schaltfläche links neben dem Label angezeigt.

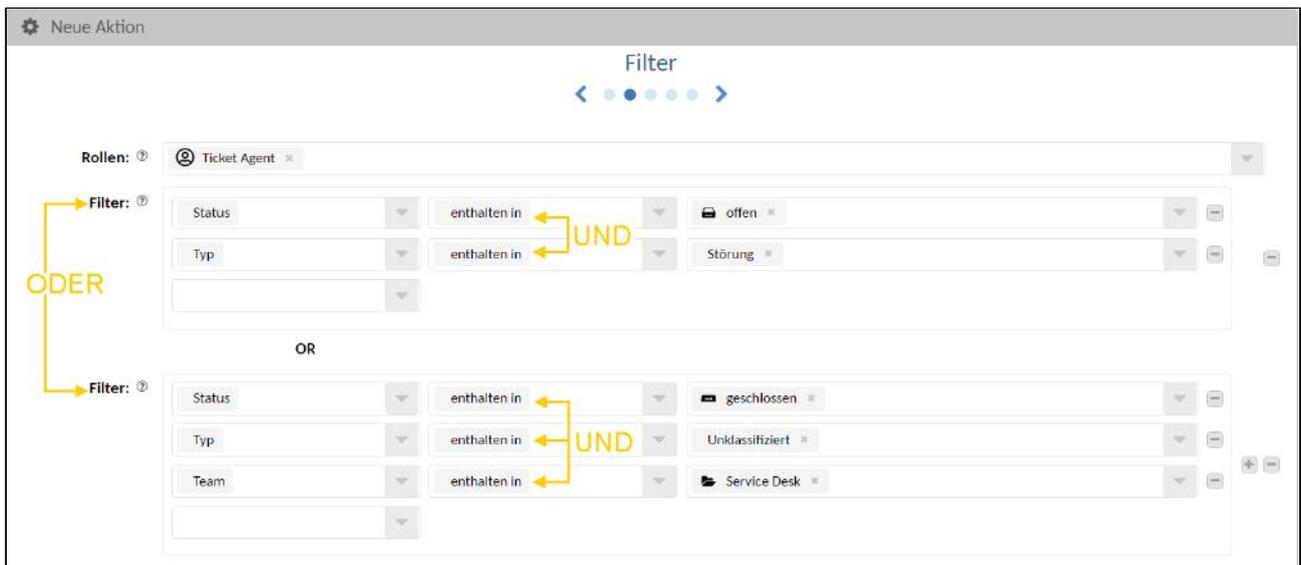
Feld	Beschreibung
Verhalten	<p>Das gewählte Verhalten steuert, was beim Ausführen der Aktion passieren soll. Die Auswahlmöglichkeiten hängen vom gewählten Referenzobjekt ab.</p> <ul style="list-style-type: none"> • Neues Ticket: Erstellt bei Ausführung der Aktion ein neues Ticket. <ul style="list-style-type: none"> • Das neue Ticket kann als Kind-Ticket mit dem Quellticket verknüpft werden. Geben Sie dazu dem Eingabefeld "ParentTickets" den Platzhalter "<KIX_TICKET_TicketID>" mit (s. auch Konfiguration der Artikelaktion "Neues Ticket/Split"). Die entstandenen Kind-Tickets werden in den Ticketdetails des Elterntickets angezeigt. • Pre- und Post-Actions werden nicht ausgeführt. Sie sind daher in der Konfiguration der Aktion deaktiviert. • Sofern nicht anders konfiguriert, werden für das (Kind-)Ticket die dem Quellticket zugrunde liegende Ticketvorlage verwendet und die Artikelinhalte und Anhänge übertragen. Eine am Elternticket gesetzte "Wartezeit" wird nicht übertragen und muss manuell gesetzt werden. • Wird diese Option in bestehenden Aktionen deaktiviert, so werden eventuell gesetzte Pre- und Post-Actions gelöscht und nicht mehr ausgeführt. • Ticket bearbeiten: Nimmt bei Ausführung der Aktion Änderungen an einem bestehenden Ticket vor. Verwenden Sie diese Option, wenn an einem bestehenden Ticket Werte gesetzt oder weitere Eingaben vorgenommen werden sollen. • Neuer Artikel: Erstellt bei Ausführung der Aktion einen neuen Artikel am Ticket. • Artikel Bearbeiten: Nimmt Änderungen am jeweiligen Artikel vor (s. auch initial inaktive Aktion "ArticleUpdate"). <p>Hinweis: Achten Sie bei der Konfiguration eigener Aktionen darauf, dass die neuen Tickets mindestens die erforderlichen Pflichtangaben am Ticket enthalten.</p>

Feld	Beschreibung
Rang	<p>Legt die Reihenfolge der Aktionen-Schaltfläche in der Ticketdetailansicht fest. Eine Aktion mit einem niedrigeren Rang wird weiter links angeordnet. "0" ist ein gültiger Wert. Ist kein Wert angegeben, erhält der Rank den Wert NULL und die Aktion wird links angeordnet. Bei gleicher Prioritätsangabe werden die Aktionen in alphabetischer Reihenfolge angezeigt. Die Standardreihenfolge der Aktionen ist:</p> <ol style="list-style-type: none"> 1. Bearbeiten 2. Neuer Artikel 3. Schließen 4. Andere Aktionen ohne Vorlagen
Nutzereingabe erforderlich	<p>Wählen Sie aus, ob die Aktion eine Nutzereingabe erfordert.</p> <p><input checked="" type="checkbox"/> Generiert eine <u>dialogbasierte Aktion</u>. Wird die Aktion vom Nutzer gewählt, wird ein Dialog geöffnet, in dem weitere Angaben gemacht werden können. Diese werden erst übernommen, wenn der Nutzer auf "Speichern" klickt. In den nachfolgenden Schritten der Aktionen-Konfiguration erstellen Sie das Template für den Dialog und geben dabei an, welche Felder im Dialog angezeigt werden und legen deren Verhalten und Werte fest.</p> <p><input type="checkbox"/> Generiert eine <u>Aktion ohne Dialog</u>. Verwenden Sie diese Option, wenn keine Nutzereingabe erforderlich ist, z.B. um ein Ticket per Knopfdruck zu archivieren.</p> <p>Damit erkennbar ist, dass die Aktion ausgeführt wurde, können Sie optional eine System-Rückmeldung festlegen. Aktivieren Sie dazu die Option "Bestätigung Rückfrage" und hinterlegen Sie unter "Bestätigung Inhalt" den anzuzeigenden Text. Dann erscheint eine Hinweismeldung mit dem ebendiesem Text, sobald die Aktion ausgeführt wurde.</p>
Kommentar	Hinterlegen Sie hier Notizen zur Aktion (optional)
Gültigkeit	Setzen Sie die Aktion auf "gültig". Auf "ungültig" oder "temporär ungültig" gesetzte Aktionen können vom Nutzer nicht genutzt werden.

Filter

Mit dem Setzen von Filtern definieren Sie, unter welchen Voraussetzungen eine Aktion zur Verfügung steht.

Sie können die Filter kombinieren. Die einzelnen Filterblöcke sind ODER-Verknüpft. Die Filter innerhalb eines Filterblocks sind UND-Verknüpft.



Beispiele können sein:

1. Die Ticketaktion "Urlaubsantrag genehmigen" steht nur den Verantwortlichen am Ticket zur Verfügung, wenn im Betreff eines Tickets "Urlaubsantrag" steht. In allen anderen Tickets ist diese Schaltfläche ausgeblendet.
2. Die Artikelaktion "Weiterleiten an Hersteller" steht am Artikel nur dann zur Verfügung, wenn der Bearbeiter einem bestimmten Team angehört und wenn es sich um eine Störung bei ausgewählten Assets handelt.
3. Die Ticketaktion steht nur dann zur Verfügung, wenn der angemeldete Nutzer die Rolle Ticket Agent besitzt und auch der Verantwortliche am Ticket ist.

Feld	Beschreibung
Rollen	<p>Wählen Sie aus, welchen Rollen Sie die Berechtigung zu dieser Aktion erteilen.</p> <p>Damit eine Aktion im Self Service Portal zur Verfügung steht, wird mindestens die Rolle "Customer" benötigt.</p> <p>Verwenden Sie diese Auswahl ausschließlich für Rollen, um deren Berechtigungen hinsichtlich Aktionen zu erweitern. Anderenfalls findet eine Einschränkung der Berechtigungen statt (z. B. bei den Rollen "Super User" und "System Admin").</p> <p>⚠ Beachten Sie die Berechtigungen innerhalb der Rollen!</p> <p>Beispiel: Sie konfigurieren eine Ticket-bearbeiten-Aktion und wählen die Rolle "Ticket Reader" aus. Die Aktion wird nicht angezeigt, weil die Rolle "Ticket Reader" keine Berechtigung für die Ticketbearbeitung besitzt.</p> <div data-bbox="459 884 1423 1182" style="border: 1px solid red; padding: 10px;"><p>⚠ Wichtig!</p><p>Für die gewählten Rollen wird explizit ein "READ" für Templates & Aktionen gesetzt. Vergeben Sie daher diese Berechtigung niemals auf Rollen, die mehr als Leserechte auf Aktionen & Templates haben! Anderenfalls findet eine Einschränkung der Berechtigungen statt (z. B. Superuser, Admin).</p></div>

Feld	Beschreibung
Filter	<p>Durch Setzen von Filtern können Sie die Verfügbarkeit der Aktion optional weiter einschränken. Sie legen damit fest, unter welchen Bedingungen eine Aktion zur Verfügung steht.</p> <p>Basis der Filter ist die gewählte Rolle. Im o. a. Beispiel steht die Aktion nur Mitgliedern der Rolle "Ticket-Agent" zur Verfügung, welche auch als Verantwortliche am Ticket gesetzt sind.</p> <p>Die Angabe der Filter erfolgt analog der Logik in der Komplexsuche (s. Anwenderhandbuch):</p> <ol style="list-style-type: none"> 1. Spalte: Filterattribute für Tickets 2. Spalte: Suchoperatoren (siehe Seite 102) (z. B.: enthalten in, beginnt mit, endet mit, ist gleich, usw.) 3. Spalte: Formularfelder und Werte. <p>Die Aktion steht an allen Tickets zur Verfügung, welche den hier angegebenen Parametern entsprechen. Geben Sie mehrere Filter an, werden die Filter durch eine sog. logische UND-Verknüpfung miteinander verknüpft. Sie können z. B. angeben, dass eine Aktion nur in den Tickets zur Verfügung steht, die den Typ "unklassifiziert" UND den Status "geschlossen" besitzen UND vom Team "Support" sind.</p> <p>Bei gewähltem Bezugsobjekt "Ticket" stehen alle ticketbasierten Filter sowie Dynamischen Felder (außer Typ Checklist) zur Auswahl. Für Dynamische Felder des Typs Text und Textarea steht Ihnen außerdem die Wildcard-Suche zur Verfügung (Suche mit * als Platzhalter für unbekannt Textstellen). Sie können gültige und ungültige Objekte und Attribute auswählen. Die Verwendung von KIX Platzhaltern (z. B. <KIX_CURRENT_UserID>) ist möglich. Platzhalter müssen manuell ins Auswahlfeld eingetragen und mit ENTER bestätigt werden. Zudem können relative Zeitangaben definiert werden (z. B. erstellt innerhalb der letzten 24 Stunden).</p> <p>Hinweis: Werden auf "ungültig" gesetzte Dynamische Felder angegeben, so finden diese <u>keine</u> Verwendung bei der Ausführung.</p>
-	Entfernt den Filter
+	Fügt weitere Filter ein.

Vorgelagerte Aktionen

Vorgelagerte Aktionen (Pre-Actions) sind konfigurierte Macros, die im Hintergrund ausgeführt werden, sobald die Aktion angeklickt wird:

- In dialogbasierten Aktionen:
 - Die vorgelagerten Aktionen werden noch vor dem Öffnen des Dialogfensters ausgeführt.
 - Bricht ein Nutzer eine Aktion ab, erhält er einen Hinweis, dass im Hintergrund eine Aktion ausgeführt wurde.
 - Überspringen Sie diesen Schritt, wenn Sie keine vorgelagerten Aktionen definieren wollen.
- In nicht-dialogbasierten Aktionen (One-Click-Action):
 - Legen Sie fest, was im Hintergrund passieren soll, sobald der Nutzer die Aktion anklickt. Bspw. das Ticket ins "Junk"-Team verschieben, einen Wartestatus oder den Verantwortlichen am Ticket setzen.
 - Nach Ausführung der Aktion erhält der Agent eine entsprechende Hinweismeldung.
 - **Hinweis:** In nicht-dialogbasierten Aktionen finden die "Eingabefelder" in Schritt 4 keine Berücksichtigung; deren Steuerung erfolgt ausschließlich über die Pre- und/oder Post-Actions.

Vorgelagerte Aktionen werden vom Self Service Portal nicht unterstützt. Sie können daher nicht konfiguriert werden, wenn als Nutzungskontext "Kunde" gewählt wurde.

Die Verwendung von Platzhaltern (z. B. <KIX_CURRENT_UserID>) ist möglich. Platzhalter müssen manuell ins Auswahlfeld eingetragen und mit ENTER bestätigt werden.



Hinweise

- Vorgelagerte Aktionen werden mit der Nutzererkennung des aufrufenden Nutzers ausgeführt. Das bedeutet, eine direkte Nutzung der Aktionen nur durch ein Backend-API-Aufruf ist nicht möglich.
- Geben Sie bitte eine Differenz zur Wartezeit in Sekunden an, wenn Sie einen Warten-Status setzen (z. B. "Warten zur Erinnerung" oder "Warten auf erfolgreich schließen"). Damit wird das Datum automatisch gesetzt, sodass das Ticket auch ohne manuelle Zeitangabe geschlossen werden kann.

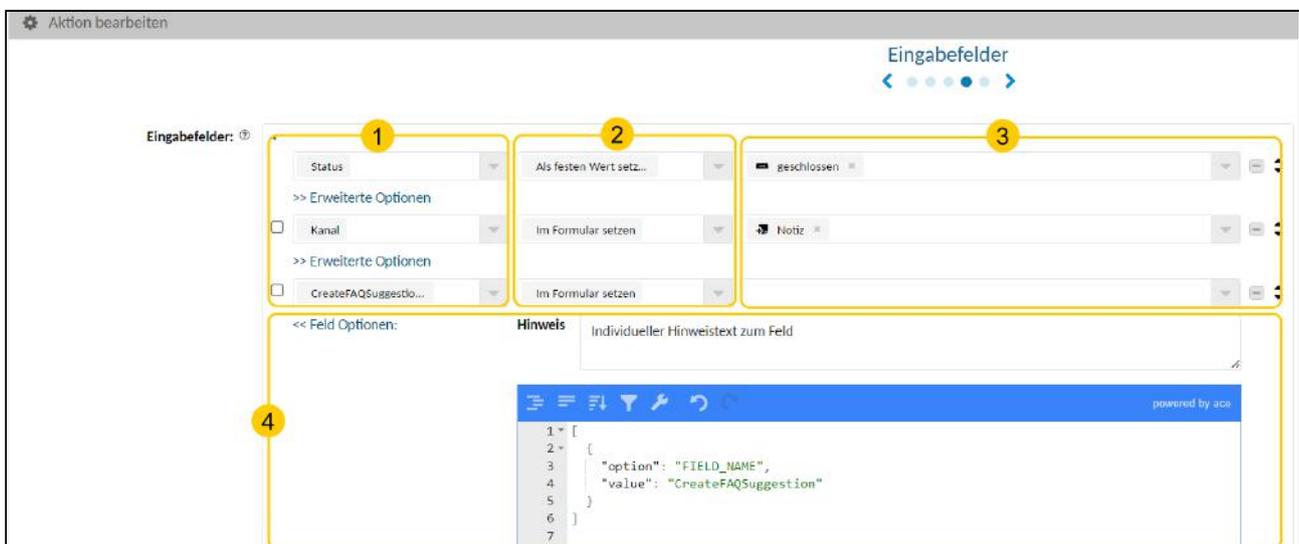
Tipp

Die Macro Action "Loop" kann verwendet werden, um Macros an referenzierten Tickets auszuführen und beliebig zu verschachteln.

Eingabefelder

Legen Sie fest, welche Felder in den Dialog eingebunden werden sollen. Sie gestalten damit den Aufbau und den Inhalt des Dialogfensters in einer dialogbasierten Aktion. Für nicht-dialogbasierte Aktionen kann dieser Schritt übersprungen werden.

Hinweis: Diese Konfiguration findet nur dann Berücksichtigung, wenn in den Aktionsinformationen (Schritt 1) die Option "Nutzereingabe erforderlich" aktiviert ist.



Spalte		Beschreibung
1	Feldauswahl	<p>Wählen Sie hier die Felder aus, die in den Dialog gesetzt werden sollen. Es stehen Ihnen alle Ticket- und Artikelattribute sowie die Dynamischen Felder zur Auswahl.</p> <p><input checked="" type="checkbox"/> Ein Häkchen vor dem Eingabefeld deklariert das Feld als Pflichtfeld.</p> <p>Hinweise zur Feldauswahl:</p> <ul style="list-style-type: none"> • Kanalspezifische Felder (To, CC, BCC, Betreff, Artikelinhalt, Anhänge) benötigen in der Vorlagendefinition zusätzlich das Feld "Kanal". Erst wenn bei Verwendung der Vorlage ein Kanal ausgewählt wurde, werden die kanalspezifischen Felder angezeigt und mit Vorgabewerten befüllt. • Berücksichtigen Sie Berechtigungen: Werden Felder in eine Ticketaktion konfiguriert, auf die der Nutzer (Agent oder Customer) keine Berechtigung hat, so schlägt bei ihm deren Verwendung fehl. • Erhält ein Ticket den Status "Warten auf [...]", wird eine dynamische Zeit in der Zukunft als Vorbelegung für die Zielwartezeit angezeigt. Diese Vorbelegung wird initial aus dem aktuellen Zeitpunkt plus dem in der SysConfig hinterlegten Wert ermittelt. Sie können die Vorbelegung der Zielwartezeit anpassen, indem Sie im Menü <i>System</i> > <i>SysConfig</i> den Schlüssel <code>"Ticket::Frontend::PendingDiffTime"</code> ändern. Bitte beachten: Im Ticket muss eine Zielwartezeit angegeben werden, damit es auch ohne manuelle Änderung der Zeitangabe gespeichert werden kann. • Beachten Sie auch die Hinweise zur Bereitstellung von Ticketaktionen (siehe: Vorlagen, Aktionen, Rule Sets (siehe Seite 225)). • Dynamisches Feld "Betroffene Services" im Self Service Portal: Es erfolgt keine Prüfung von Berechtigungen. Das heißt: Sind Services "als festen Wert" oder "im Hintergrund" in einer Aktion gesetzt, dann wird der Service gesetzt, auch wenn der Self Service Portal Nutzer nicht über die entsprechenden Berechtigungen verfügt.

Spalte		Beschreibung
2	Verhalten	<p>Geben Sie an, wie das Feld bzw. dessen Wert in den Dialog integriert wird:</p> <ul style="list-style-type: none"> • Im Formular setzen: Das Feld wird in der Ticketmaske angezeigt und mit dem bereits am Ticket gesetzten Wert initialisiert. Dieser Wert kann durch einen Vorgabewert im Dialog überschrieben werden. • Als festen Wert setzen: Das Feld und dessen Wert werden in der Ticketmaske angezeigt. Der Wert kann nicht bearbeitet werden. Z. B. um im Ticket einen nicht bearbeitbaren Wert anzuzeigen. • Im Hintergrund setzen: Das Feld und dessen Wert werden in der Ticketmaske nicht angezeigt. Bspw. um Felder gezielt von der Bearbeitung auszuschließen oder um dem Ticket unsichtbare Werte mitzugeben. • Feld leeren: Blendet das Feld im Dialog ein, initialisiert es jedoch mit einem leeren Wert. Damit wird das Rücksetzen von Werten ermöglicht. In Verbindung mit Pflichteingaben wird eine bewusste Entscheidung des Anwenders erzwungen. <p><u>Relative Zeitangaben:</u></p> <p>Datum-/Zeit-Felder können mit einer relativen Initialisierung versehen werden. Folgende Optionen sind möglich:</p> <ul style="list-style-type: none"> • Im Formular setzen (relative Zeit) • Als festen Wert setzen (relative Zeit) • Im Hintergrund setzen (relative Zeit) <p>Verhalten des Dynamischen Felds wie zuvor beschrieben, jedoch Angaben jeweils als relativer Zeitwert (z. B. 10 Minuten). Verfügbar an Dynamischen Feldern der Typen Date und DateTime. Verwendbar bspw. für Wartezeitpunkte mit relativen Zeitdifferenzen oder Tickets mit relativen Zeitpunkten für Plan-Beginn/Plan-Ende.</p> <p>Basierend auf dem Zeitpunkt der Verwendung wird der angegebene Zeitwert zur aktuellen Zeit addiert und als Datum gesetzt (bei jedem Aufruf der Aktion erneut).</p>

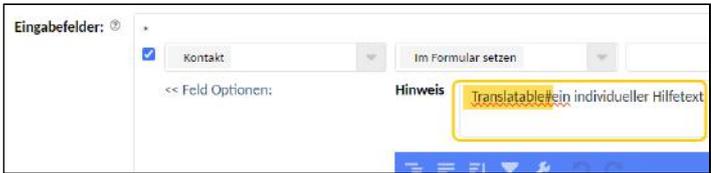


Spalte	Beschreibung
	<p>Die Verwendung von KIX Platzhaltern wird unterstützt, sodass eine in einem Dynamischen Feld (Feldtyp Text) angegebene Zahl als Zeitwert ausgelesen und verwendet werden kann.</p> <p>Die definierten Wartezeitpunkte besitzen eine höhere Priorität als die Einstellung im SysConfig-Schlüssel <i>"Ticket::Frontend::PendingDiffTime"</i></p>

Spalte		Beschreibung
<div style="background-color: yellow; border-radius: 50%; width: 20px; height: 20px; display: flex; align-items: center; justify-content: center; margin: 0 auto;">3</div>	Werte definieren	<p>Geben Sie optional einen Wert an, mit dem das Feld bei Verwendung der Aktion vorbelegt wird. Lassen Sie den Wert frei, wenn Sie dem Feld keinen Wert mitgeben möchten.</p> <p>Die Auswahlmöglichkeiten in Dropdowns stehen im direkten Kontext zu Ihrer Feldauswahl in Spalte 1.</p> <p>Hinweise für die Verwendung von Platzhaltern:</p> <ul style="list-style-type: none"> • Sie können Platzhaltervariablen (siehe Seite 102) angeben, um variable, ticketspezifische Werte ins Feld zu setzen. Tragen Sie den Platzhalter manuell ins Auswahlfeld ein und bestätigen Sie mit ENTER. Platzhalter für Dynamische Felder geben Sie nach folgendem Schema an: <KIX_TICKET_DynamicField_NameDesDynamischenFeldes>. • Beim Bearbeiten von Tickets bezieht sich die Ersetzung von Platzhaltervariablen auf das Quellticket; bei neuen Tickets sind die Felder leer (für Ticketplatzhalter). • Felder, welche auf andere Objekte wie Kontakte, Bearbeiter oder Priorität referenzieren, erwarten eine ID als Wert. Das heißt, bei der Verwendung von KIX Platzhaltern muss die ID angegeben werden (z. B. KIX_CURRENT_UserID>). Bei Dynamischen Feldern, der unter "Key" angegebene Wert. • Damit Platzhalter in Ticketformularen korrekt referenzieren, ist auf eine Unterscheidung und richtige Verwendung von <i>Nutzer</i>-Platzhaltern (KIX_xxx_UserID) und <i>Kontakt</i>-Platzhaltern (KIX_xxx_ContactID) zu achten. So benötigt z. B. das Kontakt-Feld einen Platzhalter, der auf CONTACTS referenziert - auch wenn der Kontakt ein Agent und damit ein Nutzer sein kann. Das bedeutet bspw. für Platzhalter im Template "Default - New Ticket Dialog": <ul style="list-style-type: none"> • <i>Platzhalter für den Kontakt im Ticket:</i> <ul style="list-style-type: none"> • richtig: <KIX_CURRENT_ContactID> • falsch: <KIX_CURRENT_UserID> • Platzhalter für den aktuell angemeldeten Nutzer als Bearbeiter oder Verantwortlicher: <ul style="list-style-type: none"> • richtig: <KIX_CURRENT_UserID>



Spalte	Beschreibung
	<ul style="list-style-type: none">Felder, welche auf andere Objekte wie Kontakte, Bearbeiter oder Priorität referenzieren, erwarten eine ID als Wert. Das heißt, bei der Verwendung von KIX Platzhaltern muss die ID angegeben werden (z. B. KIX_CURRENT_UserID>). Bei Dynamischen Feldern, der unter "Key" angegebene Wert.

Spalte	Beschreibung
<p data-bbox="172 376 225 421">4</p> <p data-bbox="360 376 596 409">Erweiterte Optionen</p>	<p data-bbox="683 376 1382 524">Feld "Hinweis": Sie können einen individuellen Hinweistext zum Feld hinterlegen. Damit können Sie bspw. die Agenten darüber informieren, welche Daten ein Dynamisches Feld erwartet.</p> <p data-bbox="683 553 1390 663">Wenn ein Hinweistext angegeben ist, wird am Feld ein Fragezeichen-Symbol angezeigt. Der Hinweistext wird nach Klick auf das Fragezeichen angezeigt.</p> <div data-bbox="687 683 1398 913" data-label="Image">  </div> <p data-bbox="683 922 1251 956"><i>Abb.: Individueller Hinweistext am Feld "Kontakt"</i></p> <p data-bbox="683 981 1398 1090">Der Hinweistext wird an der Feldkonfiguration der jeweiligen Aktion gespeichert, sodass für jede Aktion ein anderer Hinweistext zum Feld notiert werden kann.</p> <p data-bbox="683 1115 1410 1263">Die initialen Felder einer Vorlage enthalten bereits lokalisierte Hinweistexte, welche über das Weblate bereitgestellt werden. Bei Bedarf können Sie diese durch Ihren individuellen Text ersetzen.</p> <p data-bbox="683 1288 1374 1435">Die Übersetzung Ihrer Texte können Sie im Menü <i>KIX > Internationalisierung > Übersetzung</i> selbst pflegen (s. Übersetzungen). Die zu lokalisierenden Hilfetexte, kennzeichnen Sie mit vorangehendem <code>Translatable#:</code></p> <div data-bbox="687 1464 1398 1637" data-label="Image">  </div> <p data-bbox="683 1646 1426 1718"><i>Abb.: Mit vorangehendem "Translatable#" werden zu übersetzende Texte gekennzeichnet</i></p> <p data-bbox="683 1742 1350 1812">Hinweis: Per Workflow Rule hinzugefügte Felder werden aktuell nicht betrachtet.</p>

Editor: Sie können ein Array weiterer Konfigurationsoptionen für das Feld im JSON-Format angeben, z. B. spezifische Ladeoptionen, den zu ladenden Objekttyp usw.

Abhängig vom gewählten Feld werden verschiedene Optionen unterstützt. Einige Konfigurationsoptionen werden automatisch gesetzt, damit die Felder korrekt funktionieren, z. B. für das Dynamische Feld "Betroffenes Asset".

- **Option:** Angabe des Attributs
- **Value:** Wert des Attributs

Mit Feldoptionen können bspw. auch die im Feld zur Auswahl stehenden Nutzer auf bestimmte Rollen eingegrenzt werden.

Weiterführende Hinweise finden Sie auch im Kapitel [Vorlagen \(siehe Seite 151\)](#) (Abschnitte: Eingabefelder, Hinweise zur Feldauswahl, Hinweise zu den Feldoptionen).

Nachgelagerte Aktionen

Hier können Sie definieren, welche Aktionen direkt im Anschluss an die ausgeführte Aktion ausgeführt werden sollen. Beispielsweise das Ticket schließen, nachdem der Urlaubsantrag abgelehnt wurde. Die Konfiguration erfolgt analog der "Vorgelagerten Aktionen" (s. oben).

Nachgelagerte Aktionen werden im Hintergrund ausgeführt, unmittelbar nach dem in dialogbasierten Aktionen auf die Schaltfläche "Speichern" geklickt wurde. In nicht-dialogbasierten Aktionen werden nachgelagerte Aktionen ausgeführt, sobald die Aktion angeklickt wird.

Nachgelagerte Aktionen werden vom Self Service Portal nicht unterstützt. Sie können daher nicht konfiguriert werden, wenn als Nutzungskontext "Kunde" gewählt wurde.

Sie können KIX-Platzhalter verwenden (z. B. <KIX_CURRENT_UserID>). Schreiben Sie den Platzhalter manuell ins Auswahlfeld und Übernehmen Sie die Eintragung mit ENTER.

Überspringen Sie diesen Schritt, wenn Sie keine nachgelagerten Aktionen definieren wollen.

Hinweise

- Nachgelagerte Aktionen werden mit der Nutzererkennung des aufrufenden Nutzers ausgeführt. Das bedeutet, eine direkte Nutzung der Aktionen nur durch ein Backend-API-Aufruf ist nicht möglich.
- Geben Sie bitte eine Differenz zur Wartezeit in Sekunden an, wenn Sie einen Warten-Status setzen (z. B. "Warten zur Erinnerung" oder "Warten auf erfolgreich schließen"). Damit wird das Datum automatisch gesetzt, sodass das Ticket auch manuelle Zeitangabe geschlossen werden kann.

5.2 Rule Sets

Sie können das Verhalten von Eingabe- und Auswahlfeldern steuern und somit den Workflow innerhalb von Dialogen besser unterstützen. Dies erfolgt im Menü "Workflow > Rule Sets" anhand von individuell gestalteten Regelsätzen (Rulesets). Jeder Regelsatz enthält beliebig viele, frei definierbare Regeln (Rules). Die Regeln definieren die Bedingungen für das Vorhandensein und das Verhalten von Auswahl- und Eingabefeldern in Ticketformularen, Vorlagen und Aktionsdialogen.

Sie können somit bspw. steuern, dass in einem konkreten Dialog unter ganz bestimmten Voraussetzungen

- Felder dynamisch ein- bzw. ausgeblendet werden
- Werte in Felder gesetzt bzw. entfernt werden
- nur zulässige Eingabe- und Auswahlwerte bereitstehen
- ein Feld zum Pflichtfeld wird oder nicht
- u.a.m.

Die Regeln wirken sich direkt auf die initialen Formular-Konfigurationen und auf die von Ihnen konfigurierten Vorlagen und Aktionsdialoge aus und beeinflussen diese. Haben Sie bspw. in einer Vorlage ein Dynamisches Feld eingebunden und blenden Sie dieses mit einer Regel aus, so wird das Dynamische Feld nicht angezeigt, auch wenn es in der Vorlage enthalten ist. Durch Angabe von Bedingungen können Sie zudem festlegen, unter welchen Voraussetzungen das Feld ausgeblendet wird. Somit können Sie das Verhalten, Aussehen und die Inhalte der Formulare/Dialoge granuliert und ganz gezielt steuern.

Inhalte auf dieser Seite:

- [Aufbau von Rulesets](#) (siehe Seite 122)
- [Reihenfolge der Abarbeitung](#) (siehe Seite 124)
 - [Abarbeitung von Rulesets de-/aktivieren](#) (siehe Seite 125)
- [Einen Regelsatz anlegen, bearbeiten oder löschen](#) (siehe Seite 126)
- [Regelwerk](#) (siehe Seite 128)
- [Übersicht der Befehle](#) (siehe Seite 129)
 - [Eigenschaften des Transaktionsobjekts](#) (siehe Seite 132)
 - [Einschränkung des Nutzungskontextes](#) (siehe Seite 133)
 - [Operatoren](#) (siehe Seite 133)
 - [Bedingungen \(if\)](#) (siehe Seite 135)
 - [Anweisungen \(than\)](#) (siehe Seite 138)
- [Funktionen](#) (siehe Seite 149)
 - [DynamicFields.contains](#) (siehe Seite 149)

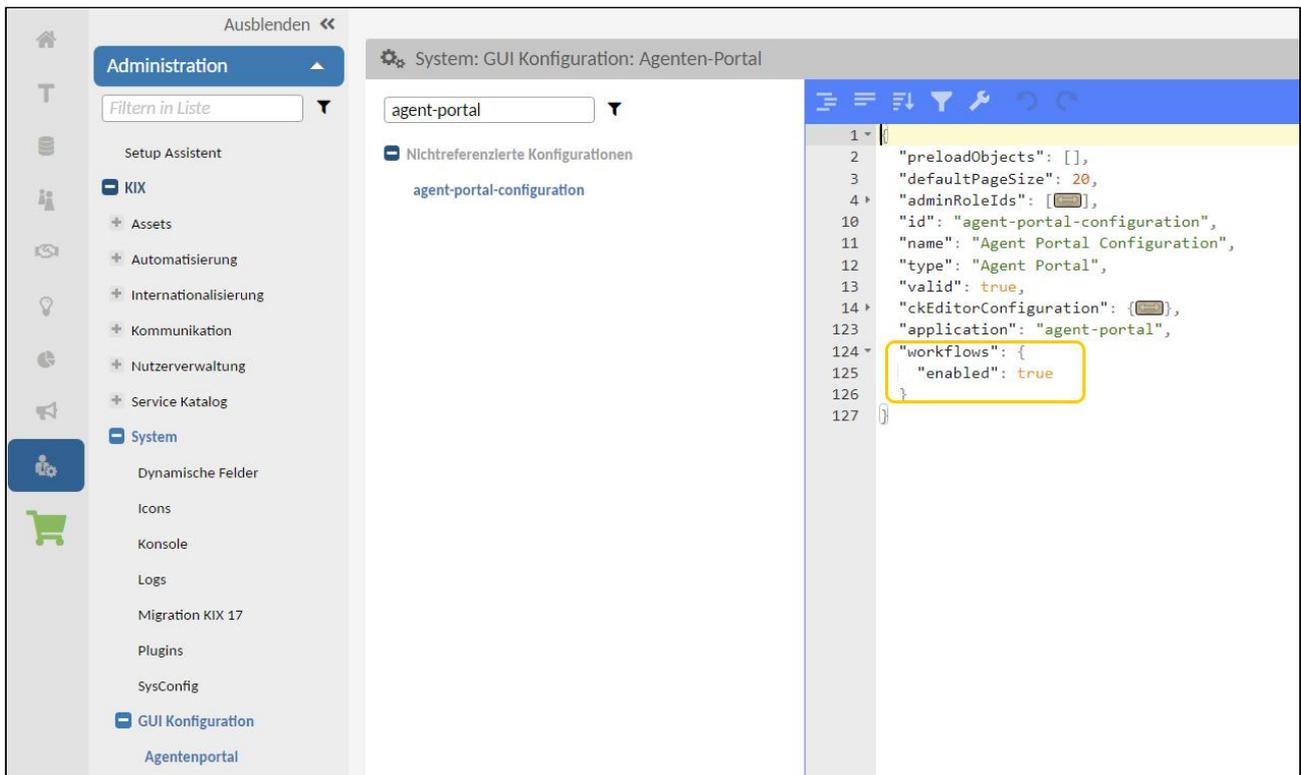


Abb.: Aktivierung Workflow-Evaluation

Achtung!

Regeln können sich - je nach Konfiguration - untereinander ausschließen und miteinander kollidieren! Bedenken Sie daher die Komplexität und arbeiten Sie im Vorfeld gewissenhaft die Regeln und Regelsätze aus, um dies zu vermeiden.

5.2.1 Aufbau von Rulesets

Ein Ruleset ist eine Liste einzelner Regeln (Rules). Jede Regel ist wie folgt aufgebaut:
(Teile in runden Klammern sind optional)

Syntax

```

Rule "<Name>" (on <Object>) (if <Condition>)
  (<Commands>)
End

```

Es gilt:

- Jede Regel beginnt mit dem Schlüsselwort `Rule` .
- Jede Regel endet mit dem Schlüsselwort `End` .

- Eine Regel wird durch eine Leerzeile von der nachfolgenden Regel getrennt.
- Zeilen können Einrückungen enthalten.
- Für eine bessere Lesbarkeit können Aktionen innerhalb einer Regel durch beliebig viele Leerzeichen von den Operanden getrennt werden (s. Beispiel 2).
- `<Commands>` ist eine Liste der Befehle (siehe unten "Übersicht der Befehle").
- Jeder Befehl steht auf einer Zeile.
- Es können Kommentare zwischen den Regeln und zwischen den `<Commands>` geschrieben werden.
- Kommentare beginnen mit dem `#` Zeichen.
- Der `<Name>` der Regel kann beliebig sein und aus mehreren Buchstaben, Wörtern und Ziffern bestehen (nutzbar, um die Regel aussagekräftig zu bezeichnen).
- Der `<Name>` der Regel darf gleichlautend in mehreren Rulesets existieren, jedoch nur einmal pro Ruleset.
- Regeln können optional auf ein spezifisches `<Object>` angewendet werden (aktuell nur auf Tickets; Schlüsselwort: `on`).

Beispiel 1

```
# das ist ein Kommentar zur nachfolgenden Regel
Rule "hier stoppen, unter einer Bedingung" if DB.TicketID == 10
  Stop
End

Rule "hier stoppen" on Ticket
  # stop here!
  Stop
End
```

Beispiel 2

```
# Regel wird ausgeführt, wenn am Ticket der Typ "Störung" gesetzt wird
Rule "Type" on Ticket if TR.TypeID eq "Incident"
  Set Priority "1 very high"
  Set StateID 2
  Set Owner "mmuster"
  Hide DynamicFields.TestFeld
End
```

5.2.2 Reihenfolge der Abarbeitung

Die Abarbeitung der **Rulesets** erfolgt von oben nach unten in alphanumerischer Reihenfolge ihrer Benennung.

<input type="checkbox"/>	Name	Comment	Rule Count	Validity
<input type="checkbox"/>	1 Ruleset		3	valid
<input type="checkbox"/>	A_Ruleset		6	valid
<input type="checkbox"/>	RuleSet 2		4	valid
<input type="checkbox"/>	RuleSet 1		8	valid
<input type="checkbox"/>	RuleSet A		2	valid
<input type="checkbox"/>	RuleSet B		5	valid

Abb.: Reihenfolge der Abarbeitung von Rulesets

Durch Angabe eines vorangehenden oder nachgestellten Buchstabens oder Zahl können Sie die Reihenfolge der Abarbeitung steuern, z. B.:

- A_Ruleset, B_Ruleset, C_Ruleset
- Ruleset1, Ruleset2, Ruleset3
- 1_RulesetA, 2_RulesetA, 1_RulesetB, 2_RulesetB
- etc.

Wir empfehlen Ihnen, im Vorfeld eine eigene Namenskonvention zu erstellen.

Die in einem Ruleset definierten **Regeln** werden von oben nach unten abgearbeitet - sowohl die einzelnen Regeln im Ruleset als auch innerhalb einer Regel. Die Bezeichnung der Regeln ist für die Reihenfolge der Abarbeitung irrelevant. Es zählt die Anordnung der Regeln im Ruleset.

Comment:

* Validity: valid x

Rules:

- 1 Rule "Ticket type Unclassified" on Ticket if TR.TypeID == 1
- 2 PossibleValuesRemove StateID 4
- 3 End
- 4
- 2 Rule "Ticket type Incident" on Ticket if TR.TypeID == 2
- 6 Set PriorityID "5"
- 7 PossibleValuesAdd PriorityID 5
- 8 End
- 9
- 3 Rule "Ticket type Service Request" on Ticket if TR.TypeID == 3
- 11 Set PriorityID "3"
- 12 PossibleValuesAdd PriorityID 3
- 13 End
- 14

Abb.: Reihenfolge der Abarbeitung von Rules innerhalb eines Rulesets

Sowohl Rulesets als auch Regeln kaskadieren. D. h. wird eine Eigenschaft eines Objekts mehrfach angegeben, so gilt die zuletzt angegebene Eigenschaft (außer bei "PossibleValuesAdd" und "PossibleValuesRemove" - s. unten). Achten Sie bei Ihrer Konfiguration darauf, dass sich Regeln nicht unbeabsichtigt beeinflussen oder aushebeln.

Im nachfolgenden Beispiel gewinnt Priorität 5. Die Priorität 2 wird durch Priorität 3 ersetzt und diese wiederum wird durch Priorität 5 ersetzt.

Beispiel: Kaskadierende Regeln

```

Rule "alpha" on Ticket if TR.TypeID == 1
  Set PriorityID 2
End

Rule "beta" on Ticket if TR.TypeID == 1
  Set PriorityID 3
End

Rule "gamma" on Ticket if TR.TypeID == 1
  Set PriorityID 5
End
  
```

5.2.2.1 Abarbeitung von Rulesets de-/aktivieren

SysConfig-Schlüssel	agent-portal-configuration
----------------------------	----------------------------

Die Abarbeitung von Workflow-Rules erzeugt nach jeder Eingabe, die eine Regel startet, eine Eingabesperre zur Prüfung der Eingabewerte. Die Prüfung erfolgt, nachdem ein Eingabe- bzw. Auswahlfeld den Focus verloren hat. In Multiselect-Feldern und in Dynamischen Feldern vom Typ "Tabelle" erfolgt die Prüfung nach Klick auf "Apply". Die Eingabe des Artikelinhalts startet keine Auswertung der Workflow Rules.

Das Verhalten ist nur aktiv, wenn gültige Rulesets definiert sind UND die Workflow-Evaluierung aktiviert ist. Die Aktivierung der Workflow-Evaluierung erfolgt im Menü "System > GUI Konfiguration > Agentenportal" im SysConfig-Schlüssel "agent-portal-configuration".

5.2.3 Einen Regelsatz anlegen, bearbeiten oder löschen

Zum Anlegen bzw. Bearbeiten eines Rulesets, navigieren Sie im Admin Modul ins Menü *Workflow > Rule Sets*. Legen Sie mit Klick auf "Neues Rule Set" ein neues Ruleset an oder klicken Sie ein bestehendes Ruleset, um dieses zur Bearbeitung zu öffnen. Definieren Sie im sich öffnenden Dialog das Ruleset:

Feld	Beschreibung	Beispiel
Name	Tragen Sie hier eine aussagekräftige Bezeichnung für das Ruleset ein.	
Kommentar	Notieren Sie optional einen Kommentar zum Ruleset, bspw. um dessen Funktion zu erläutern.	
Conditions	Geben Sie optional eine globale Bedingung für das gesamte Ruleset an. Das gesamte Ruleset wird nur dann ausgeführt, wenn diese Bedingung zutrifft.	<ul style="list-style-type: none"> • <code>User.isCustomer</code> (gesamtes Ruleset gilt nur für das Self Service Portal) • <code>User.isAgent</code> (gesamtes Ruleset gilt nur für das Agentenportal) • <code>User.isCustomer && TR.TemplateID == 3</code> (gesamtes Ruleset gilt nur im Self Service Portal bei Verwendung der Ticketvorlage mit der ID 3) <p>(s. auch unten: Einschränkung des Nutzungskontextes)</p>

Feld	Beschreibung	Beispiel
Regeln	<p>Definieren Sie hier die einzelnen Regeln gemäß dem o.a. Aufbau und beachten Sie dabei die Reihenfolge der Abarbeitung (von oben nach unten).</p> <p>Die Regeln können sehr gezielt und granuliert definiert werden.</p> <p>Da die vom System vergebenen IDs in je System variieren</p>	<pre>Rule "<Name>" on Ticket if User.isCustomer && TR.TemplateID == 3 <Commands 1,2,3> End Rule "<Name>" on Ticket if User.isAgent && TR.TemplateID == 4 <Commands 4,5,6> End</pre>

 Ruleset bearbeiten

*** Name:**

Kommentar:

*** Gültigkeit:**

Condition:

Regeln:

```

1 Rule "myRule1" on Ticket if TR.DynamicFields.contains(DFSelectionSingle, 1)
2 Show DynamicFields.DFTextArea
3 Enable DynamicFields.DFTextArea
4 Show OwnerID
5 Enable OwnerID
6 Show ResponsibleID
7 Enable ResponsibleID
8 Show SLAID
9 Enable SLAID
10 Show ChannelID
11 Enable ChannelID
12 Show TypeID
13 Enable TypeID
14 End
15
16 Rule "myRule2" on Ticket if TR.DynamicFields.contains(DFSelectionSingle, 2)
17 Required DynamicFields.DFTextArea
18 Show DynamicFields.DFTextArea
19 Enable DynamicFields.DFTextArea
20 End
```

Abb.: Beispiel eines Rulesets

Nicht mehr benötigte Rulesets können gelöscht werden:

- Markieren Sie das betreffende Ruleset mit einem Häkchen
- Klicken Sie auf die Schaltfläche "Löschen".
- Bestätigen Sie die Sicherheitsabfrage mit "Ja", um das Ruleset endgültig zu löschen.

Danach wird das Ruleset nicht mehr angewandt.

5.2.4 Regelwerk

Alle in KIX gespeicherten Daten befinden sich in der Datenbank. Wird ein Formular geöffnet, werden die Daten aus der Datenbank ausgelesen und temporär im Formular angezeigt **1**. Im Formular können diese Daten nun geändert und um weitere Daten ergänzt werden. Erst mit dem Speichern werden die Formulardaten zurück in die Datenbank geschrieben **2**.

Daher unterscheidet das Regelwerk zwischen:

- **Datenbankobjekt:** die in der Datenbank bzw. im Backend bereits gesetzten (Ist-)Werte
- **Transaktionsobjekt:** die in einem geöffneten Dialog angezeigten oder eingegebenen Werte

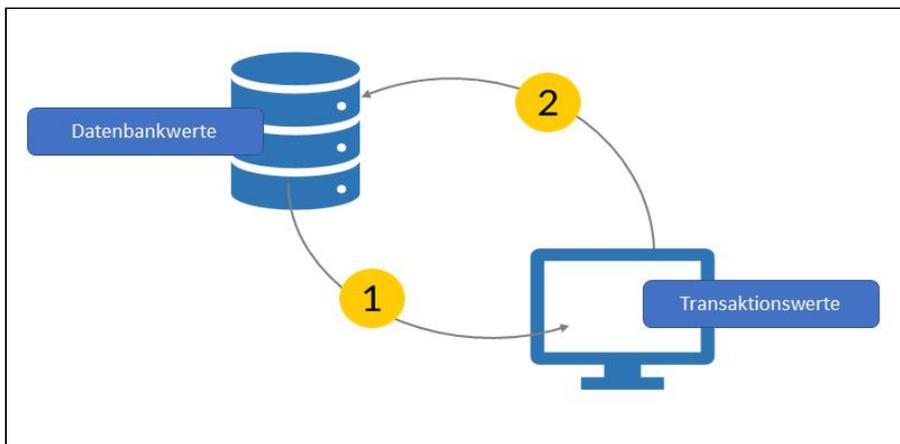


Abb.: Darstellung Unterschied zwischen Datenbank- und Transaktionswerten

Das bedeutet:

- Sprechen Sie Werte an, die in der **Datenbank oder im Backend** gespeichert sind, dann verwenden Sie das Präfix " **DB.** "
- Sprechen Sie Werte an, die in einem **geöffneten Formular** angezeigt oder eingegeben sind, dann verwenden Sie das Präfix " **TR.** "

Beispiel für Transaktionswerte

```
# Ist im geöffneten Ticket-Dialog der Tickettyp "2" (Störung) ausgewählt, dann setze
# Priorität auf "1" (sehr hoch)
Rule "Type" on Ticket if TR.TypeID == 2
  Set PriorityID 1
End
```

Beispiel für Datenbankwerte

```
# Ist am Ticket der Tickettyp "2" (Störung) gespeichert, dann setze die Priorität auf
"1" (sehr hoch)
Rule "Type" on Ticket if DB.TypeID == 2
  Set PriorityID 1
End
```

5.2.5 Übersicht der Befehle

Legende zu den nachfolgenden Übersichten

Platzhalter	Beschreibung	Beispiel
	ODER-Trennzeichen Wird in der Übersicht verwendet, wenn verschiedene Möglichkeiten existieren, z. B. wenn ein Wert ODER eine Liste von Werten angegeben werden kann.	"<Value> <Liste>"
<Array>	Es kann ein Array von Werten angegeben werden.	Set BCC "martin.mustermann@nomail.org", "max.mueller@example.com", "anothermailaddress.example.com"
<Attribute>	Es kann ein KIX-Attribut angegeben werden, s. auch: Übersicht der Ticketattribute (siehe Seite 121)	Set StateID 2
<AttributListe>	Es kann eine Liste von Attributen angegeben werden.	InputOrder PriorityID, StateID, DynamicFields.DFText

Platzhalter	Beschreibung	Beispiel
<Commands>	Eine Liste beliebiger Befehle. Jeder Befehl muss auf einer separaten Zeile stehen.	<pre>Rule ... Show PriorityID Set PriorityID 5 ReadOnly PriorityID End</pre>
<Condition>	<p>Es kann der Ausdruck einer Bedingung angegeben werden.</p> <p>Schlüsselwort: <code>if</code> (s. auch unten "Bedingungen (siehe Seite 135) ")</p> <p>Die Bedingung muss erfüllt sein, damit Regel ausgeführt wird.</p>	<pre>Rule "<Name>" on Ticket if TR.TypeID == 2 && TR.StateID == 3 <Commands> End</pre>
<Count>	Numerischer Wert für eine Anzahl	<pre>MinSelectable DynamicFields.DFSelection 2</pre>
<ErrorMessage>	Text einer Fehlermeldung nach Prüfung der Nutzereingaben mittels RegEx	<pre>RegExp DFText "[A-Za-z0-9]+\$" "Nur Ziffern und Buchstaben möglich"</pre>
<Liste>	Es kann eine Liste von Werten angegeben werden.	<pre>PossibleValuesAdd PriorityID 4,5,6</pre>
<Name>	<p>Der Name der Regel (alphanumerisch)</p> <p>Muss in Hochkommas gesetzt werden. Kann Sonderzeichen und Leerzeichen enthalten.</p>	<pre>Rule "Setze Werte 3, 4 & 5" on Ticket if TR.TypeID == 2</pre>

Platzhalter	Beschreibung	Beispiel
<Object>	<p>Es kann ein KIX-Objekt angegeben werden, auf das die Regel angewendet wird.</p> <p>Schlüsselwort: <code>on</code></p> <p>s. auch: Übersicht der KIX Objekte</p> <p>i Aktuell nur "Ticket" möglich</p>	<pre>Rule "<Name>" on Ticket if TR.TypeID == 2 && TR.StateID == 3 <Commands> End</pre>
<RegEx>	<p>Zur Validierung von Nutzereingaben kann ein Regulärer Ausdruck angegeben werden.</p>	<pre>RegExp DFText "[A-Za-z0-9]+\$" "Nur Ziffern und Buchstaben möglich"</pre>
<Value>	<p>Angabe des zu setzenden Wertes</p>	<pre>Set StateID 2</pre>

5.2.5.1 Eigenschaften des Transaktionsobjekts

Das Transaktionsobjekt kann nicht nur auf KIX-Attribute (z. B. TR.TypeID) zugreifen, sondern auch auf Vorlagen und Aktionen. Damit können Regeln derart eingeschränkt werden, dass sie nur bei Verwendung einer bestimmten Ticketvorlage (Template) oder einer Ticket-/Artikel-Aktion (ObjectAction) angewendet werden.

Dies erfolgt durch Referenz auf die ID bzw. den Namen der Vorlage/Aktion. Beachten Sie daher den in der Vorlage/Aktion gesetzten Nutzungskontext. Wird eine Regel auf eine vom Agentenportal und Self Service Portal gemeinsam genutzte Vorlage/Aktion angewendet (Nutzungskontext Kunde + Agent), so gilt die Regel für beide Portale. Soll diese Regel jedoch nur für das Agentenportal oder für das Self Service Portal gelten, so können Sie entweder 2 separate Vorlagen/Aktionen mit jeweils anderem Nutzungskontext anlegen oder den Nutzungskontext einschränken (s. unten "Einschränkung des Nutzungskontextes").

Beispi: Erweitere Eigenschaften für Transaktionsobjekt

```
Rule "<Name>" on Ticket if TR.ObjectActionName eq "Ticket Edit" && TR.DynamicFields.contains(DFSelection, 1)
    <Commands>
End
```

```
Rule "<Name>" on Ticket if TR.TemplateID == 3 && TR.DynamicFields.contains(DFSelection, 1)
    <Commands>
End
```

Eigenschaft	Beschreibung
TR.TemplateID	ID des verwendeten Templates
TR.ObjectActionID	ID der verwendeten ObjectAction
TR.ObjectActionName	Name der verwendeten ObjectAction Hinweis: Bei Vergleichen Operator <code>eq</code> anstatt <code>==</code> verwenden!

✓ Tipp

Die ID eines Objektes können Sie ermitteln, indem Sie in einer Übersicht den Mauszeiger über das gewünschte Objekt führen. In der Fußzeile Ihres Browsers finden Sie die ID als letzten/rechten Parameter in der angezeigten URL (browserabhängig).

5.2.5.2 Einschränkung des Nutzungskontextes

Für Ticketvorlagen und Aktionen, die vom Agentenportal und Self Service Portal gemeinsam genutzt werden (Nutzungskontext Kunde + Agent), können Sie getrennte Regeln erstellen. Es ist somit möglich, die Ausführung von Regeln auf den jeweiligen Nutzungskontext einzuschränken. Die Einschränkung erfolgt durch Angabe der Eigenschaft des Nutzers:

- **User.isCustomer:** Regel wird auf das Self Service Portal angewendet.
- **User.isAgent:** Regel wird auf das Agentenportal angewendet.

Im folgenden Beispiel wird die Ticketvorlage mit der ID 3 vom Agentenportal und Self Service Portal gemeinsam genutzt. Je nach dem, in welchem Portal die Vorlage genutzt wird, gelten unterschiedliche Regeln.

Beispiel: Einschränkung Nutzungskontext

```
#Regel für Ticketvorlage 3 wird auf den Nutzungskontext 'Customer' (Self Service Portal) eingeschränkt:
```

```
Rule "UsageContext Customer" on Ticket if User.isCustomer && TR.TemplateID == 3 && TR.DynamicFields.contains(DFSelectionSingle, 1)  
  <Commands 1,2,3>  
End
```

```
#Regel für Ticketvorlage 3 wird auf den Nutzungskontext 'Agent' (Agentenportal) eingeschränkt:
```

```
Rule "UsageContext Agent" on Ticket if User.isAgent && TR.TemplateID == 3 && TR.DynamicFields.contains(DFSelectionSingle, 2)  
  <Commands 4,5,6>  
End
```

5.2.5.3 Operatoren

Die Verwendung von Operatoren ermöglicht das Verknüpfen und Vergleichen von Bedingungen. Nachfolgend die wichtigsten Perl-Operatoren.

Beispiel für UND-Operator

```
# Sind am Ticket der Tickettyp "3" (Service Anfrage) UND der Status 3 (Warten) gewählt, dann setze die Priorität auf "4" (niedrig)
```

```
Rule "Type" on Ticket if TR.TypeID == 2 && TR.StateID == 3  
  Set PriorityID 4  
End
```

Operator		Beschreibung
1. Rang	2. Rang	
<code>&&</code>	<code>and</code> <code>AND</code>	Stellt eine logische UND-Verknüpfung zwischen 2 Argumenten her. Regel wird ausgeführt, wenn beide Argumente wahr sind.
<code> </code>	<code>or</code> <code>OR</code>	Stellt eine logische ODER-Verknüpfung zwischen 2 Argumenten her. Regel wird ausgeführt, wenn eins der beiden Argumente wahr ist.
<code>==</code>	<code>eq</code>	Vergleichsoperator "ist gleich" Vergleicht 2 Werte. Gibt wahr zurück, wenn beide Werte exakt gleich sind. <ul style="list-style-type: none"> • Verwenden Sie <code>eq</code>, um Zeichenketten zu vergleichen. <code>eq</code> führt einen Vergleich von Zeichenfolgen durch. Beide Argumente müssen lexikalisch gleich sein (Groß-/ Kleinschreibung beachten). Beispiel: <code>"3.0" eq "3"</code> gibt <code>false</code> zurück • Verwenden Sie <code>==</code>, um Zahlen zu vergleichen. <code>==</code> führt einen numerischen Vergleich durch. Beide Argumente werden in eine Zahl umgewandelt und verglichen. Beispiel: <code>"3.0" == "3"</code> gibt <code>true</code> zurück
<code>!=</code>	<code>ne</code>	Vergleichsoperator "ist ungleich" Vergleicht 2 Werte. Gibt wahr zurück, wenn beide Werte nicht gleich sind. <ul style="list-style-type: none"> • Verwenden Sie <code>ne</code>, um Zeichenketten zu vergleichen. • Verwenden Sie <code>!=</code>, um Zahlen zu vergleichen.

Weiterführende Informationen zu Perl-Operatoren finden Sie bspw. unter: https://de.wikibooks.org/wiki/Perl-Programmierung:_Operatoren

5.2.5.4 Bedingungen (if)

Das Schlüsselwort "if" definiert eine WENN-Bedingung, die erfüllt sein muss, damit die in der Regel angegebenen Befehle abgearbeitet werden. Ist keine Bedingung angegeben, wird die Regel in jedem Fall ausgeführt.

```
Beispiel "if": Wenn die TicketID gleich 10 ist, dann stoppen

Rule "hier stoppen" if DB.TicketID == 10
  Stop
End
```

Schlüsselwort/ Befehl	Beschreibung	Beispiel
DB.<Attribute>	Abarbeitung erfolgt auf Grundlage der Datenbankwerte (s. oben)	
TR.<Attribute>	Abarbeitung erfolgt auf Grundlage der Transaktionswerte (s. oben)	
if !Rule "<Name>" ...	Prüft, ob eine bestimmte Regel NICHT angewendet wurde.	
if Rule "<Name>" ...	Prüft, ob eine bestimmte Regel angewendet wurde.	

Schlüsselwort/ Befehl	Beschreibung	Beispiel
if <Attribute> .isEmpty(<Feld>)	Prüft, ob ein Feld keinen Wert oder eine leere Liste besitzt.  [NULL] oder [""] sind <u>keine</u> leere Liste. Hierbei handelt es sich um einen Wert, auch wenn dieser nicht definiert ist.	Rule "<Name>" if TR.DynamicFields.isEmpty(Category) <Commands> End
if <Attribute>! isEmpty(<Feld>)	Prüft, ob ein Feld <u>irgendeinen</u> Wert oder keine leere Liste besitzt.	Rule "<Name>" if TR.DynamicFields. !isEmpty(Category) <Commands> End
if <Attribute> .contains(<Feld>,<Wert- Liste>)	Prüft, ob irgendeiner der angegebenen Werte in einem Feld enthalten ist. Es kann eine Liste von Werten angegeben werden.	Rule "<Name>" if TR.DynamicFields.contains(Category, [1,2,3,23]) <Commands> End

 **Tipp**

Durch Kombination von " if Rule " mit " if !Rule " kann ein einfaches "IF - ELSEIF - ELSE" erzeugt werden:

Beispiel: Einfaches IF - ELSEIF - ELSE

```
Rule "alpha" on Ticket if TR.ObjectActionName eq "Ticket Edit"
  <Commands>
End

Rule "beta" if Rule "alpha"
  Show DynamicFields.DFText
End

Rule "gamma" if !Rule "alpha"
  Hide DynamicFields.DFText
End
```

Die Prüfung von Bedingungen (Conditions) kann sowohl auf eine Objekt-ID auch auf den jeweils eindeutigen Namen von Objekten erfolgen. Akzeptiert werden die Namen für:

- Team/Queue: der **volle** Team-/Queuename (z.B. "Service Desk")
 - Bei untergeordneten Teams muss auch das übergeordnete Team angegeben werden.
Verwenden Sie doppelte Doppelpunkte als Trennzeichen: z. B. "MainTeamA::SubTeamA1")
- State: der Name des Status
- Priority: der Name der Priorität
- Type: der Name des Tickettyps
- Owner: der Login-Name des Nutzers
- Responsible: der Login-Name des Nutzers
- Contact: die Mailadresse (ohne Realname, nur "localpart@domainpart")
- Organisation: die Kundennummer

Beispiel Bedingung mit Namen

```
Rule "Queue" on Ticket if TR.Queue eq "Service Desk::Monitoring"  
  Set Priority "1 very high"  
  Set State "new"  
  Set Type "Incident"  
  Set Owner "mamu"  
  Set Responsible "mamu"  
  Set Contact "max.mustermann@example.org"  
  Set Organisation "MY_ORGA"  
End
```

Mehrere Bedingungen können miteinander kombiniert werden, indem sie in Klammern gesetzt werden:

Kombinierte Bedingungen

```
Rule "ProzessStatus" on Ticket if (TR.ObjectActionName eq "Freigabe" || TR.ObjectActionName eq "Planung") && (TR.QueueID == 1 || TR.QueueID == 2 || TR.QueueID == 3) && TR.DynamicFields.contains(DFChiffre, "C4711-08A15")  
  <Commands>  
End
```

Bedingungen können als RegEx angegeben werden:

RegEx als Bedingung

```
Rule "Show Dynamic Field" on Ticket if TR.ObjectActionName =~ /^(Ticket Edit|Ticket New Article)$/ && TR.Queue =~ /^(Service Desk|.*Monitoring)$/  
  Show DynamicFields.DynamicFieldName
```

```
    Enable DynamicFields.DynamicFieldName  
End
```

5.2.5.5 Anweisungen (than)

Anweisungen sind ein einzelner Befehl oder eine Liste von Befehlen innerhalb einer Regel. Sie folgen auf eine WENN-Bedingung ("if") und entsprechen einem "than"-Statement gemäß folgender Aussage: WENN eine bestimmte Bedingung erfüllt ist, **DANN** führe folgenden Befehl/folgende Befehle aus.

Jede Zeile enthält genau eine Anweisung. Die Anweisungen werden in der angegebenen Reihenfolge von oben nach unten abgearbeitet.

Beispiel: Liste von Anweisungen

```
# Ist die Ticket ID gleich 10,  
# - DANN: Zeige das Feld Priorität an  
# - DANN: Setze die Priorität 3 (Normal)  
# - DANN: Stoppe die Abarbeitung der Regeln  
Rule "hier stoppen" if DB.TicketID == 10  
    Show PriorityID  
    Set PriorityID 3  
    Stop  
End
```

Einschränken des Wertebereiches:

Schlüsselwort/Befehl	Beschreibung	Beispiele
<p>PossibleValues <Attribute> <Liste></p>	<p>Definiert die möglichen Werte in einem Feld. Nur die hier angegebenen Werte können in ein Feld gesetzt werden.</p> <p>Werte, die Buchstaben enthalten müssen in einfache oder doppelte Hochkommas gesetzt werden ("Text" oder 'Text').</p> <p>Dynamische Felder erwarten stets den Key und nicht den Wert.</p>	<pre>Rule "Tickettyp Incident" on Ticket if TR.TypeID == 2 PossibleValues PriorityID 1,2,3 PossibleValues DynamicField.DFText "Text" PossibleValues DynamicFields.AffectedAsset 17,18 End</pre>
<p>PossibleValuesAdd <Attribute> <Liste></p>	<p>Fügt einem Feld die angegebenen Werte hinzu.</p> <p>Ergänzt damit bspw. weitere Auswahlwerte in einem Selectfeld, wenn zuvor die möglichen Auswahlwerte mit "PossibleValues" eingeschränkt wurden.</p> <p>Hinweis: Wird der Befehl mehrfach verwendet, so werden die zuvor definierten Werte <u>nicht ersetzt</u>, <u>sondern ergänzt</u> (Kombination der Befehle)!</p>	<pre>Rule "Tickettyp Incident" on Ticket if TR.TypeID == 2 PossibleValuesAdd PriorityID 4,5,6 End</pre>

Schlüsselwort/Befehl	Beschreibung	Beispiele
<p>PossibleValuesRemove <Attribute> <Liste></p>	<p>Entfernt die angegebenen Werte aus einem Feld. Reduziert damit bspw. die möglichen Auswahlwerte in einem Selectfeld, wenn die möglichen Auswahlwerte zuvor mit "PossibleValuesAdd" erweitert wurden.</p> <p>Hinweis: Wird der Befehl mehrfach verwendet, so werden die zuvor definierten Werte <u>nicht ersetzt</u>, sondern <u>weiter reduziert</u> (Kombination der Befehle).</p>	<pre>Rule "Tickettyp Unclassified" on Ticket if TR.TypeID == 1 PossibleValuesRemove StateID 4 End</pre>

Info

Für die Commands `PossibleValues`, `PossibleValuesAdd`, `PossibleValuesRemove`, und `Set` müssen Zeichenketten in doppelte Anführungszeichen eingeschlossen ein.
Es können Kommas innerhalb einer Zeichenkette enthalten sein. Diese gelten nicht als Trennzeichen der Werteliste.

Einschränken der Auswählbarkeit von Werten:

Beachten Sie, dass sich einige der nachfolgenden Befehle ausschließen. So kehrt bspw. der Befehl "Hide" den Befehl "Show" um. Das heißt, wird ein Feld mit "Show" angezeigt und danach ein "Hide" auf das Feld angewendet, so beendet das die Anzeige des Feldes. Der letzte Befehl gewinnt.

Schlüsselwort/ Befehl	Beschreibung	Beispiel
MinSelectable <Attribute><Count >	Definiert die Anzahl der Werte, die der Nutzer mindestens auswählen muss, z. B. mindestens 2 von <i>n</i> Werten.	Rule ... MinSelectable DynamicFields.DFSelection 2 End
MaxSelectable <Attribute><Count >	Definiert die Anzahl an Werten, die der Nutzer maximal auswählen darf, z. B. maximal 2 von <i>n</i> Werten; danach ist keine weitere Auswahl möglich ($x \leq n$)	Rule ... MaxSelectable DynamicFields.DFSelection 2 End

Schlüsselwort/ Befehl	Beschreibung	Beispiel
<p>Set <Attribute> "<Value>" "<Liste>" "<Array>"</p>	<p>Setzt einen festen Wert in ein Feld.</p> <p>Erlaubt optional eine Liste von Werten oder ein Array. Zeichenketten müssen in doppelte Hochkommas gesetzt werden.</p> <p>Die Kombination von Zeichenketten und Werten ist möglich - sofern vom Attribut unterstützt.</p> <p>Der Parameter "Set" unterstützt neben den Objekt-IDs auch die Namen von Objekten. Akzeptiert werden für:</p> <ul style="list-style-type: none"> • Team/Queue: der volle Team-/Queuename (z.B. "TeamA::SubteamOne") • State: der Name des Status • Priorität: der Name der Priorität • Type: der Name es Tickettyps • Owner: der Login-Name des Nutzers • Responsible: der Login-Name des Nutzers • Contact: die Mailadresse (ohne Realname, nur "localpart@domainpart") • Organisation: die Kundennummer 	<pre>Rule ... Set PriorityID 5 Set State "New" Set Owner "mamu" Set OrganisationID 2,5,10 Set DynamicFields.DFText "foo", "foo bar", "foo, bar & 2 foobar" Set DynamicFields.DFDateTime "2022-09-01 12:34:56" End</pre>

Schlüsselwort/ Befehl	Beschreibung	Beispiel
	Wird in einer Regel ein Parameter mehrfach angegeben (bspw. als Queue oder QueueID), so gewinnt der letzte Eintrag.	
ReadOnly <Attribute>	Feld wird auf "nur Lesen" gesetzt. Gegenstück und Umkehr des Befehls "Writable" Tipp: In Verbindung mit <code>Set</code> kann ein Wert fix in ein Feld gesetzt werden.	<pre>Rule ... Set PriorityID 5 ReadOnly PriorityID End</pre>
Writeable <Attribute>	Feld wird auf "änderbar" gesetzt Gegenstück und Umkehr des Befehls "ReadOnly".	<pre>Rule ... Writable DynamicFields.DFText End</pre>
Clear <Attribute>	Ein im Feld gesetzter Wert wird entfernt.	<pre>Rule ... Clear DynamicFields.DFSelection End</pre>
Show <Attribute>	Feld wird angezeigt/nicht versteckt Gegenstück und Umkehr des Befehls "Hide" .	<pre>Rule ... Show PriorityID End</pre>

Schlüsselwort/ Befehl	Beschreibung	Beispiel
Hide <Attribute>	<p>Feld wird versteckt (ausgeblendet). Gegenstück und Umkehr des Befehls "Show".</p> <p>Hinweis: Ausgeblendete Felder sind weiterhin aktiv und werden beim Speichern berücksichtigt. Das heißt:</p> <ul style="list-style-type: none"> • ein gesetzter Wert wird im Hintergrund übertragen. • auch ein ausgeblendetes Pflichtfeld erwartet einen Wert. Ohne Wert wird das Formular nicht gespeichert. <p>Damit ein Feld beim Speichern nicht berücksichtigt wird, muss es mit "Disable" deaktiviert werden.</p>	<pre>Rule ... Hide PriorityID End</pre>

Schlüsselwort/ Befehl	Beschreibung	Beispiel
<p>Enable <Attribute></p>	<p>Setzt ein Feld auf "aktiv", sodass es entsprechend seiner Konfiguration normal verwendet werden kann. Gegenstück und Umkehr des Befehls "Disable".</p> <p>Ist das Feld versteckt, wird ein (im Hintergrund) gesetzter Wert übertragen.</p> <p>Benötigt zusätzlich den Befehl "Show", um ein mit "Hide" ausgeblendetes oder mit "Disable" deaktiviertes Feld wieder anzuzeigen.</p> <p>⚠ Führen Sie das "Enable"-Kommando stets am Ende einer Rule aus. Anderenfalls kann es zu inkonsistenten Zuständen kommen. Also erst Feld manipulieren, dann aktivieren.</p>	<pre>Rule ... Show DynamicFields.DFText CountMax DynamicFields.DFText 3 Enable DynamicFields.DFText End</pre>
<p>Disable <Attribute></p>	<p>Feld wird deaktiviert und im Frontend ausgeblendet (implizites "Hide"). (Wert wird NICHT im Hintergrund übertragen) Gegenstück und Umkehr des Befehls "Enable".</p>	<pre>Rule ... Disable DynamicFields.DFText End</pre>
<p>Optional <Attribute></p>	<p>Feld ist optional, d.h. kein Pflichtfeld Gegenstück und Umkehr des Befehls "Required".</p>	<pre>Rule ... Optional DynamicFields.DFTextArea End</pre>

Schlüsselwort/ Befehl	Beschreibung	Beispiel
Required <Attribute>	Feld ist Pflichtfeld Gegenstück und Umkehr des Befehls "Optional".	<pre>Rule ... Required DynamicFields.DFTextArea End</pre>

Beeinflussen der Feldeigenschaften:

Schlüsselwort/Befehl	Beschreibung	Beispiel
CountMax <Attribute> <Count>	<p>Setzt für die Eigenschaft "Anzahl (min)" eines Dynamischen Feldes einen (anderen) Wert.</p> <p>Hinweis: <code>CountMax</code> greift nur bei allen <u>Nicht</u>-Multiselect-Feldern. Um die Auswahl in Multiselect-Feldern (z. B. Selection oder Referenz-Dynamische Felder) zu begrenzen, verwenden Sie den Befehl <code>MaxSelectable</code> bzw. <code>MinSelectable</code>.</p>	<pre>Rule ... CountMax DynamicFields.DFTextArea 3 End</pre>
RegExp <Attribute> "<RegEx>" "<ErrorMessage>"	<p>Regulärer Ausdruck zur Validierung der Nutzereingaben in einem Feld und Text für Fehlermeldung bei invalider Eingabe, z. B. Bedingungen für die Eingabe einer validen IP4- und IP6-Adresse</p>	<pre>Rule ... Show DFText RegExp DFText "(^\\s*(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\\.){3}([0-9] 1[0-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])) \\s*\$) (\\s*(([0-9A-Fa-f]{1,4}:) {7}([0-9A-Fa-f]{1,4} :)) (([0-9A- Fa-f]{1,4}:){6}(:[0-9A-Fa-f] {1,4} ((25[0-5] 2[0-4]\\d 1\\d\\d [1-9]?\\d)(\\. (25[0-5] 2[0-4]\\d 1\\d\\d [1-9]?\\d))){3}) :)) ((([0-9A-</pre>



Schlüsselwort/Befehl	Beschreibung	Beispiel
		<pre>Fa-f]{1,4}:){5}(((:[0-9A-Fa-f]{1,4}){1,2}) :(25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)(\. (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)) {3}) :)) (((:[0-9A-Fa-f]{1,4}:){4}(((:[0-9A-Fa-f]{1,4}){1,3}) (:[0-9A-Fa-f]{1,4})?: (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)(\. (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)) {3})) :)) (((:[0-9A-Fa-f]{1,4}:){3}(((:[0-9A-Fa-f]{1,4}){1,4}) (:[0-9A-Fa-f]{1,4}){0,2}: (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)(\. (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)) {3})) :)) (((:[0-9A-Fa-f]{1,4}:){2}(((:[0-9A-Fa-f]{1,4}){1,5}) (:[0-9A-Fa-f]{1,4}){0,3}: (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)(\. (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)) {3})) :)) (((:[0-9A-Fa-f]{1,4}:){1}(((:[0-9A-Fa-f]{1,4}){1,6}) (:[0-9A-Fa-f]{1,4}){0,4}: (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)(\. (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)) {3})) :)) (:(((:[0-9A-Fa-f]{1,4}){1,7}) (:[0-9A-Fa-f]{1,4}){0,5}: (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)(\. (25[0-5] 2[0-4]\d 1\d\d [1-9]?\d)) {3})) :))) (%.+)?\s*\$))" "Gib eine gültige IP Adresse ein"</pre> <p>End</p>

Schlüsselwort/Befehl	Beschreibung	Beispiel
<p>InputOrder <AttributeList></p>	<p>Definiert die Reihenfolge der Felder im Formular, z. B.</p> <ol style="list-style-type: none"> 1. Feld: Priorität 2. Feld: Status 3. Feld: Dynamisches Feld "DFText" <p>Die benannten Felder werden in angegebener Reihenfolge ganz nach oben an den Anfang des Formulars gesetzt. Alle weiteren Formularfelder werden darunter angeordnet, gemäß der im Template definierten Reihenfolge.</p>	<pre>Rule ... InputOrder PriorityID, StateID, DynamicFields.DFText End</pre>

Allgemeines:

Schlüsselwort/ Befehl	Beschreibung	Beispiel
Stop	<p>Die Abarbeitung der Regeln wird komplett gestoppt. Nachfolgende Regeln und Regelsätze bleiben unberücksichtigt.</p> <p>Mit der nächsten Nutzeraktion beginnt die Abarbeitung der Regeln von Neuem mit dem ersten Regelsatz.</p>	<pre>Rule ... Stop End</pre>

5.2.6 Funktionen

5.2.6.1 DynamicFields.contains

Mit der Funktion `DynamicFields.contains` kann überprüft werden, ob an einem Dynamischen Feld ein bestimmter Wert gesetzt ist. Die Funktion sucht in der Liste der Dynamischen Felder nach dem Namen des Dynamischen Feldes und prüft, ob der angegebene Wert in den Werten des Dynamischen Feldes enthalten ist. Wenn ja, liefert die Funktion 1 zurück. Wenn nein, liefert die Funktion 0 zurück. Es kann stets nur 1 Wert abgefragt werden. Werden mehrere Werte des selben Feldes abgefragt, müssen weitere Regeln oder Bedingungen erstellt werden.

Beachten, Sie dass Zeichenketten in Hochkommas gesetzt werden müssen. Es wird nach genau nach dem angegebenen Begriff gesucht. Groß- und Kleinschreibung wird dabei beachtet. Numerische Werte, z. B. Auswahlwerte in Selectfeldern, erwarten keine Hochkommas.

Syntax:	TR.DynamicFields.contains(<Name des DF>,<Wert des DF>)	DB.DynamicFields.contains(<Name des DF>, <Wert des DF>)
	TR.DynamicFields.contains(<Name des DF>, <"Zeichenkette">)	DB.DynamicFields.contains(<Name des DF>, <"Zeichenkette">)

Beispiel: DynamicFields.contains

```
Rule "Alpha" on Ticket if TR.DynamicFields.contains(DFSelectionSingle, 0)
  MaxSelectable DynamicFields.DFSelectionMulti 1
  PossibleValues DynamicFields.DFSelectionMulti 0,1,2
End
```

```
Rule "Beta" on Ticket if TR.DynamicFields.contains(DFSelectionSingle, 2)
  MaxSelectable DynamicFields.DFSelectionMulti 1
  PossibleValues DynamicFields.DFSelectionMulti 3,4,5
End
```

```
Rule "Gamma" on Ticket if TR.DynamicFields.contains(DFSelectionSingle, 2)
  MaxSelectable DynamicFields.DFSelectionMulti 1
  PossibleValues DynamicFields.DFSelectionMulti 6,7,8,9
End
```

```
Rule "Delta" on Ticket if TR.DynamicFields.contains(DFText, "Hinweis")
  Required DynamicFields.DFTextArea
  Show DynamicFields.DFTextArea
  Enable DynamicFields.DFTextArea
End
```

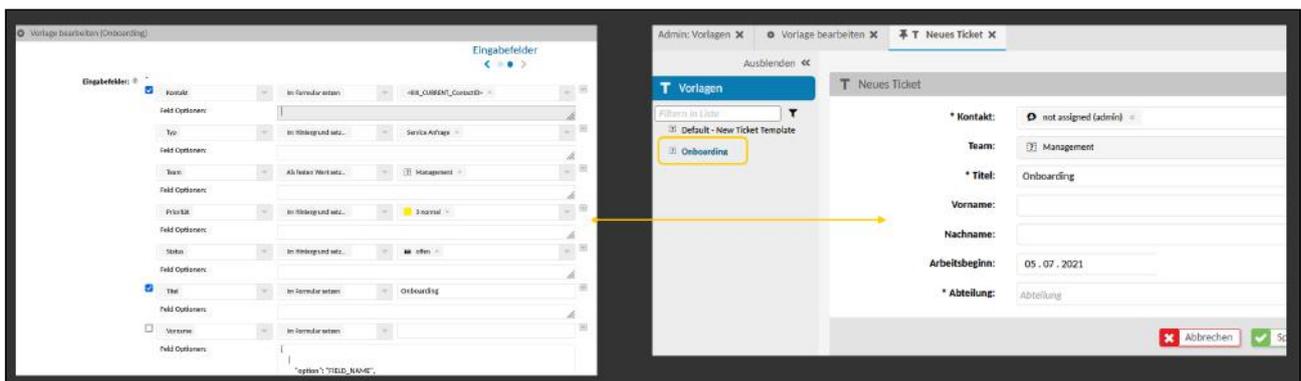
5.3 Vorlagen

Sie können in KIX Pro Vorlagen für die Erstellung neuer Tickets konfigurieren, welche unabhängig von den Vorgaben in der SysConfig sind. Vorlagen sind anwendungsabhängige Templates, mit denen die Ticketerstellmasken im Agentenportal und im Self Service Portal zum Zeitpunkt des Bedarfs an den jeweiligen Kontext angepasst werden. Sie werden daher auch als Ticketvorlagen bezeichnet.

Anwendungsfälle für Vorlagen sind bspw. spezielle Ticketerstellmasken

- für das Einstellen neuer Mitarbeiter
- für Schnelleingaben wie z. B. "Druckerpapier fehlt"
- Urlaubsanträge
- Genehmigungen
- für Checklisten
- u.v.a.m

Somit ist es möglich, an den Meldungstyp angepasste Eingaben zu machen oder spezielle Ticketvorlagen für einen ausgewählten Kundenkreis zu erstellen.



Die Verwaltung der Ticketvorlagen finden Sie unter *Workflow > Vorlagen*. Die Übersicht enthält alle im System vorhandenen Ticketvorlagen. Die Spalte "Nutzungskontext" zeigt, in welchem Portal die Vorlage ausgewählt werden kann:

- **Customer:** Die Vorlage steht im Self Service Portal (Kundenportal) zur Verfügung (erfordert an der Vorlage die Rolle "Customer").
- **Agent:** Die Vorlage steht im Agentenportal zur Verfügung.

Die Spalte "Vorlagengruppen" zeigt, welcher [Vorlagengruppe](#) (siehe Seite 182) die Vorlage zugeordnet ist.

Sie können Vorlagen **duplizieren**, um neue Vorlagen auf Basis bestehender Vorlagen anzulegen. Klicken Sie dazu in der Übersicht auf "Duplizieren". Es wird ein Duplikat der gewählten Vorlage geöffnet. Die Konfiguration des Duplikats entspricht seiner Quelle. Sie können die Konfiguration des Duplikats nach Bedarf ändern (s. unten - Eine Vorlage konfigurieren). Der Name des Duplikats lautet standardmäßig "Kopie von [Name der Vorlage]". Sie können diesen Namen beibehalten oder das Duplikat anders benennen.

Die Vorlagen stehen den Agenten im Agentenportal beim Anlegen eines neuen Tickets in der linken Sidebar zur Auswahl. Im Self Service Portal stehen die Vorlagen oberhalb des Editors zur Auswahl. Anhand der gewählten Vorlage wird die Ticketerstellmaske entsprechend der Vorlagenkonfiguration geändert. Es werden nur die in der Vorlage definierten Felder angezeigt, unabhängig von den Vorgaben der SysConfig-Schlüssel.

Standardmäßig wird die Standard-Ticketvorlage des Nutzers geöffnet. Die Standard-Ticketvorlage kann jeder Nutzer in seinen persönlichen Einstellungen unter "Meine Ticketvorlage" selbst festlegen. Zur Auswahl stehen ihm nur die gültigen Vorlagen, auf die er gemäß seiner Rollenberechtigung Zugriff hat.

Im Self Service Portal können Ticketvorlagen auch per **URL-Direktaufruf** geöffnet werden, z. B.: <https://kix-domain-ssp.kix.cloud/tickets/new?templateId=79>. Das ermöglicht den Aufruf der Vorlage auch aus einem Fremdsystem heraus. Die Vorlage wird direkt nach erfolgreicher Anmeldung des Nutzers geladen. Wird eine nicht vorhandene oder vom Nutzer nicht berechtigte Vorlage aufgerufen, wird der Dialog "Neues Ticket" geöffnet. Die zu ladende Vorlage wird anhand der TemplateID bestimmt (URL-Parameter "templateId"). Die ID eines Templates können Sie der Fußzeile Ihres Browsers entnehmen, wenn Sie in der Vorlagen-Übersicht im Admin Modul den Mauszeiger über den Namen einer Vorlage halten.

Hinweise

- Vorlagen prüfen aktuell keine Abhängigkeiten. Wird bspw. über die Vorlage ein Kunde gesetzt, der nicht zum Kontakt passt, wird der in der Vorlage gesetzte Kunde überschrieben.
- Vorlagen setzen lediglich Werte in bestehende Felder bzw. blenden bestehende Felder im Ticket aus.
- Im Self Service Portal:
 - Die Auswahl bzw. Eingabe des Ticketkontakts ist immer fixiert und kann nicht durch den Kundennutzer erfolgen.
 - Die Auswahl eines Warten-Stauts ist im Self Service Portal nicht möglich, da kein Warten-Zeitpunkt gesetzt werden kann.
- Damit Vorlagen im Self Service Portal zur Verfügung stehen, muss ihnen die Rolle "Customer" zugewiesen sein.
- Manuell geänderte Standard-Vorlagen werden im Update nicht angepasst, um die geänderte Konfiguration nicht zu kompromittieren. Eventuell mit dem Update hinzugekommene Eingabefelder müssen manuell ergänzt werden.

5.3.1 Eine Vorlage konfigurieren

Um eine Vorlage zu konfigurieren oder anzulegen, navigieren Sie zu *Workflow > Vorlagen*. Im Contentbereich wird eine Tabelle geöffnet, welche alle im System angelegten Ticketvorlagen auflistet.

Klicken Sie in der Tabelle auf "Neue Vorlage" oder wählen Sie eine bestehende Vorlage mit einem Klick aus. Es wird ein Assistent geöffnet, in dem Sie die Vorlage schrittweise konfigurieren können.

Klicken Sie entweder auf die blauen Pfeilschaltflächen oder Punkte, um zwischen den einzelnen Schritten zu wechseln. Klicken Sie bitte erst zum Schluss auf "Speichern", um Ihre Änderungen zu übernehmen.

Inhalte auf dieser Seite:

- [Vorlageninformationen](#) (siehe Seite 153)
 - [Parameter](#) (siehe Seite 155)
- [Eingabefelder](#) (siehe Seite 158)
 - [Hinweise zur Feldauswahl](#) (siehe Seite 168)
 - [Hinweise zu den Feldoptionen](#) (siehe Seite 169)

5.3.1.1 Vorlageninformationen

Tragen Sie hier die allgemeinen Informationen zur Vorlage ein.

 Vorlage bearbeiten (Onboarding)

Vorlageninformationen

< ● ● ● >

Vorlageninformationen

* **Nutzungskontext:**

* **Name:**

Icon: **T**

Rang:

Vorlagengruppe:

Kommentar:

Schlagworte:

* **Gültigkeit:**

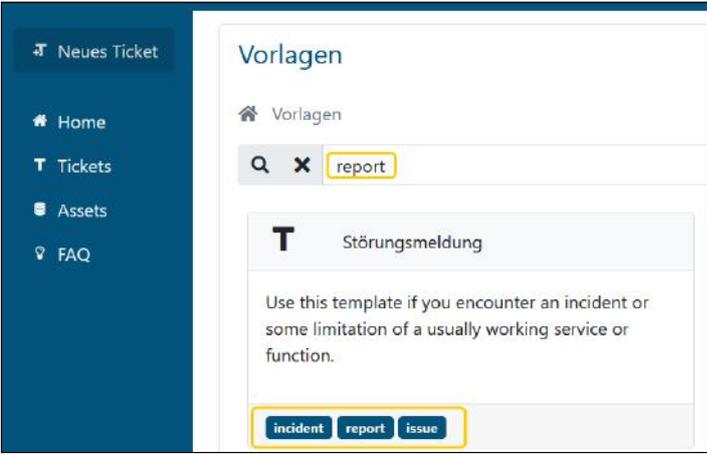
Benötigte Rolle(n)

Rollen:

Parameter

Feld	Beschreibung
Nutzungskontext	<p>Der Nutzungskontext definiert, in welchem Portal die Vorlage zur Auswahl steht (Mehrfachauswahl möglich).</p> <ul style="list-style-type: none"> • Agent: Die Vorlage kann im Agentenportal ausgewählt werden. • Kunde: Die Vorlage kann im Self Service Portal (Kundenportal) ausgewählt werden (Rolle "Customer" erforderlich). • System: Die Vorlage kann von Nutzern des Agenten- oder Self Service Portals <u>nicht</u> genutzt werden. Sie dient der Erstellung von Tickets, welche ausschließlich durch das Backend generiert werden. <p>Die Vorlage kann von Macro Actions (z.B. "Create Ticket from System Template") und anderen Automatismen genutzt werden, z. B. vom Add-on "KIX Maintenance Plan" für das automatisierte Erstellen von Tickets für Wartungsaufgaben.</p> <p>Vorlagen mit ausschließlichem Verwendungskontext "System" stehen im Agentenportal und im Self Service Portal nicht zur Verfügung.</p> <p>Systemvorlagen unterstützen die Verwendung von KIX Platzhaltern, um bei Angabe von Bezugsobjekten deren Informationen aufzulösen.</p>
Name	<p>Tragen Sie eine aussagekräftige Bezeichnung für die Vorlage ein. Unter dieser Bezeichnung können Agenten und Kunden später die Vorlage auswählen.</p>
Icon	<p>Optional können Sie ein Icon wählen, um die Vorlage symbolisch zu kennzeichnen.</p> <p> Ein geändertes Icon wird erst nach Neuansmeldung und Aktualisierung des Browsercaches angezeigt.</p>
Rang	<p>Der Rang bestimmt die Sortierreihenfolge im Self Service Portal:</p> <ol style="list-style-type: none"> 1. Vorlagen basierend auf Rang 2. Vorlagen ohne Rang, sortiert nach Name <p>Im Agentenportal erfolgt die Sortierung alphanumerisch.</p>

Feld	Beschreibung
Verhalten	<p>Legen hier Sie das Verhalten der Vorlage fest:</p> <ul style="list-style-type: none"> • Neu (Default): Wählen Sie diese Option für alle Vorlagen, die nicht vom Add-on "KIX Maintenance Plan Wartungsplan" verwendet werden. Möglich auch für Systemvorlagen, die bspw. von der Macro Action "Ticket Create from System Template" genutzt werden. • Wartung: Diese Option steht nur nach Erwerb des Add-ons "KIX Maintenance Plan Wartungsplan" zur Auswahl. Wählen Sie diese Option, wenn die Vorlage vom Add-on zum generieren der Wartungs-Tickets genutzt werden soll. In der Konfiguration Wartungsplänen können nur Vorlagen mit Nutzungskontext "System" UND Verhalten "Wartung" ausgewählt werden.
Vorlagengruppe	<p>Sie können die Vorlage einer Vorlagengruppe zuordnen und damit die Vorlagen strukturieren. Es stehen nur Vorlagen gültiger Vorlagengruppen zur Verfügung, unabhängig davon, ob die Vorlage selbst gültig ist.</p>
Kommentar	<p>Notieren Sie optional eine detaillierte Beschreibung zur Vorlage. Im Self Service Portal wird der Kommentar im Label der Vorlage angezeigt.</p>

Feld	Beschreibung
Schlagworte	<p>Sie können die Vorlage mit Schlagworten versehen (optional). Trennen Sie die einzelnen Schlagworte durch ein Leerzeichen.</p> <div data-bbox="595 468 999 533" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> Schlagworte: [Ⓢ] incident report issue </div> <p>Die Vorlagen im Self Service Portal sind mit diesen Schlagworten gekennzeichnet. Nutzer des Self Service Portals können Vorlagen nach den Schlagworten suchen.</p> 
Gültigkeit	<p>Setzen Sie die Gültigkeit auf "gültig", damit die Vorlage ausgewählt werden kann.</p> <p>Auf "ungültig" gesetzte Vorlagen stehen in den Portalen nicht zur Auswahl. In bestehenden Tickets bleibt das verwendete Template erhalten.</p>

Feld	Beschreibung
Rollen	<p>Wählen Sie eine oder mehrere Rollen aus, von denen das Template genutzt werden kann. Das Template steht nur Mitgliedern der hier ausgewählten Rollen zur Auswahl. Um Rollen abzuwählen, klicken Sie auf das kleine Kreuz neben der Rollenbezeichnung.</p> <p>Damit eine Vorlage im Self Service Portal zur Verfügung steht, muss die Rolle "Customer" ausgewählt sein.</p> <div style="border: 1px solid red; padding: 10px; margin-top: 10px;"> <p>⚠ Wichtig!</p> <p>Vergeben Sie die Berechtigung auf die Ticketvorlage niemals auf Rollen, die (explizit oder implizit) mehr als Leserechte auf Templates haben (z. B. Superuser, Admin, etc.)! Verwenden Sie die Rollen ausschließlich dazu, um Berechtigungen auf die Verwendung der Ticketvorlage zu erweitern!</p> <p>Hintergrund: Für die hier ausgewählten Rollen wird explizit ein "READ" für Templates (Vorlagen) gesetzt. Wählen Sie hier bspw. die Rolle "Superuser" aus, wird die UPDATE-Berechtigung für Templates <u>überschrieben</u>, sodass der Nutzer das Template nicht mehr bearbeiten kann. Sollte dies dennoch passieren, besteht die Möglichkeit, die Rechte in der Nutzerverwaltung manuell wieder zu ändern.</p> </div>

5.3.1.2 Eingabefelder

Legen Sie hier fest, welche Eingabe- und Auswahlfelder die Vorlage enthalten soll. Bei Auswahl einer Vorlage stehen nur die hier definierten Felder im Ticketformular zur Verfügung.

Zudem können Sie festlegen, welche Werte ins Ticket gesetzt werden sollen. Sobald die Vorlage durch den Nutzer ausgewählt wird, werden die Werte in den Ticketformularfeldern durch die in der Vorlage definierten Werte ersetzt. Eventuelle Standardwerte werden damit überschrieben.

Die Formularfelder im Ticketformular werden in der hier angegebenen Reihenfolge angezeigt. Sie können die Reihenfolge ändern, indem Sie ein Feld mittels der Doppelpfeile am rechten Rand  per Drag & Drop an die gewünschte Position verschieben. Um ein Formularfeld zu entfernen, klicken Sie auf die Minus-Schaltfläche . Weitere Formularfelder fügen Sie hinzu, indem Sie im Dropdown-Feld ein weiteres Eingabefeld auswählen.

Vorlage bearbeiten (Default - New Ticket Template)

Eingabefelder

< ● ● ● >

Eingabefelder: ⓘ

1

- Kontakt ▼
- >> Erweiterte Optionen
- Organisation ▼
- >> Erweiterte Optionen
- AffectedAsset (Betr... ▼
- >> Erweiterte Optionen
- Typ ▼
- >> Erweiterte Optionen
- Priorität ▼
- >> Erweiterte Optionen
- SLA/Service Vereinb... ▼

2

Im Formular setzen ▼

3

Service Anfrage x

3 normal x

<< Feld Optionen: **Hinweis** Mein Hinweistext

powered by ace

```

1 * [
2 * {
3   "option": "OBJECT",
4   "value": "SLA"
5 }
6 * {
7   "option": "SHOW_INVALID",
8   "value": 0
9 }
10 ]

```

4

Spalte		Beschreibung
1	Feldauswahl	<p>Wählen Sie ein Feld aus, auf welches Feld die Aktion angewendet werden soll. Es stehen Ihnen alle Ticket- und Artikelattribute sowie die Dynamischen Felder zur Auswahl.</p> <p>Beachten Sie auch die unten aufgeführten Hinweise zur Feldauswahl.</p> <p><input checked="" type="checkbox"/> Ein Häkchen vor dem Eingabefeld deklariert das Feld als Pflichtfeld.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p>Infos</p> <ul style="list-style-type: none"> • Kanalspezifische Felder (To, CC, BCC, Betreff, Artikelinhalt, Anhänge) benötigen in der Vorlagendefinition zusätzlich das Feld "Kanal". Erst wenn bei Verwendung der Vorlage ein Kanal ausgewählt wurde, werden die kanalspezifischen Felder angezeigt und mit Vorgabewerten befüllt. Wird der Kanal im Hintergrund oder als festen Wert gesetzt, muss bereits in der Vorlagendefinition der Kanal ausgewählt sein. • Berücksichtigen Sie Berechtigungen: Werden Felder in eine Vorlage konfiguriert, auf die der Nutzer (Agent oder Customer) keine Berechtigung hat, so schlägt bei ihm deren Verwendung fehl. </div>

Spalte	Beschreibung
	<ul style="list-style-type: none"> • Erhält ein Ticket den Status "Warten auf [...]", wird eine dynamische Zeit in der Zukunft als Vorbelegung für die Zielwartezeit angezeigt. Diese Vorbelegung wird initial aus dem aktuellen Zeitpunkt plus dem in der SysConfig hinterlegten Wert ermittelt. Sie können die Vorbelegung der Zielwartezeit anpassen, indem Sie im Menü <i>System > SysConfig</i> den Schlüssel "<i>Ticket::Frontend::PendingDiffTime</i>" ändern. <p>Bitte beachten: Im Ticket muss eine Zielwartezeit angegeben werden, damit es auch ohne manuelle Änderung der Zeitangabe gespeichert werden kann.</p> <div style="border: 1px solid #f0e68c; padding: 5px; margin-top: 10px;"> <p> Hinweis</p> <p>Das Self Service Portal (SSP) unterstützt nur den Kanal "Notiz".</p> </div>

Spalte		Beschreibung
2	Verhalten	<p>Legen Sie fest, wie das Feld bei Verwendung der Vorlage angezeigt und mit welchen Werten das Ticket initialisiert wird:</p> <ul style="list-style-type: none"> • Im Formular setzen: Das Feld wird in der Ticketmaske angezeigt und mit dem bereits am Ticket gesetzten Wert initialisiert. Dieser Wert kann durch einen Vorgabewert im Dialog überschrieben werden. Ein in der Vorlage vorgegebener Wert kann im Formular geändert werden (sofern unter Field Options nicht anders konfiguriert). • Als festen Wert setzen: Das Feld und dessen Wert werden in der Ticketmaske angezeigt. Der Wert kann nicht bearbeitet werden. Z. B. um im Ticket einen nicht bearbeitbaren Wert anzuzeigen. • Im Hintergrund setzen: Das Feld und dessen Wert werden in der Ticketmaske nicht angezeigt. Bspw. um Felder gezielt von der Bearbeitung auszuschließen oder um dem Ticket unsichtbare Werte mitzugeben. <ul style="list-style-type: none"> • Wichtig: <u>Artikeltext</u> und <u>Betreff</u> dürfen aus Plausibilitätsgründen nicht auf "im Hintergrund" gesetzt werden. Ziel ist, dass jegliche Kommunikation vor dem Senden für den Anwender sichtbar sein soll. • Feld leeren: blendet das Feld im Dialog ein, initialisiert es jedoch mit einem leeren Wert. Damit wird das Rücksetzen von Werten ermöglicht. In Verbindung mit Pflichteingaben wird eine bewusste Entscheidung des Anwenders erzwungen. <p>Relative Zeitangaben:</p> <p>Datum-/Zeit-Felder können mit einer relativen Initialisierung versehen werden. Folgende Optionen sind möglich:</p> <ul style="list-style-type: none"> • Im Formular setzen (relative Zeit) • Als festen Wert setzen (relative Zeit) • Im Hintergrund setzen (relative Zeit)

Spalte	Beschreibung
	<p>Verhalten des Dynamischen Felds wie vor beschrieben, Angaben jedoch als relativer Zeitwert (z. B. 10 Minuten). Verfügbar an Dynamischen Feldern der Typen Date und DateTime. Verwendbar bspw. für Wartezeitpunkte mit relativen Zeitdifferenzen oder Tickets mit relativen Zeitpunkten für Plan-Beginn/Plan-Ende. In der Vorlage selbst muss somit kein Wartestatus hinterlegt sein.</p> <p>Basierend auf dem Zeitpunkt der Verwendung wird der angegebene Zeitwert zur aktuellen Zeit addiert und als Datum gesetzt (bei jedem Aufruf der Vorlage erneut).</p> <p>Die Verwendung von KIX Platzhaltern wird unterstützt, sodass eine in einem Dynamischen Feld (Feldtyp Text) angegebene Zahl als Zeitwert ausgelesen und verwendet werden kann.</p> <p>Die definierten Wartezeitpunkte besitzen eine höhere Priorität als die Einstellung im SysConfig-Schlüssel "Ticket::Frontend::PendingDiffTime".</p>

Spalte		Beschreibung
<div style="background-color: yellow; border-radius: 50%; width: 20px; height: 20px; display: flex; align-items: center; justify-content: center; margin: 0 auto;">3</div>	Werte definieren	<p>Geben Sie optional einen Wert an, mit dem das Feld bei Verwendung der Vorlage vorbelegt wird. Lassen Sie den Wert frei, wenn Sie dem Feld keinen Wert mitgeben möchten.</p> <p>Die Auswahlmöglichkeiten in Dropdowns stehen im direkten Kontext zu Ihrer Feldauswahl in Spalte 1.</p> <p>Hinweise für die Verwendung von Platzhaltern:</p> <ul style="list-style-type: none"> • Sie können Platzhalter angeben, um variable, ticketspezifische Werte ins Feld zu setzen. Tragen Sie den Platzhalter manuell ins Auswahlfeld ein und bestätigen Sie mit ENTER. Platzhalter für Dynamische Felder geben Sie nach folgendem Schema an: <KIX_TICKET_DynamicField_NameDesDynamischenFeldes>. • Beim Bearbeiten von Tickets bezieht sich die Ersetzung von Platzhaltervariablen auf das Quellticket; bei neuen Tickets sind die Felder leer (für Ticketplatzhalter). • Felder, welche auf andere Objekte wie Kontakte, Bearbeiter oder Priorität referenzieren, erwarten eine ID als Wert. Das heißt, bei der Verwendung von KIX Platzhaltern muss die ID angegeben werden (z. B. KIX_CURRENT_UserID>). Bei Dynamischen Feldern, der unter "Key" angegebene Wert. • Damit Platzhalter in Ticketformularen korrekt referenzieren, ist auf eine Unterscheidung und richtige Verwendung von <i>Nutzer</i>-Platzhaltern (KIX_xxx_UserID) und <i>Kontakt</i>-Platzhaltern (KIX_xxx_ContactID) zu achten. So benötigt z. B. das Kontakt-Feld einen Platzhalter, der auf CONTACTS referenziert - auch wenn der Kontakt ein Agent und damit ein Nutzer sein kann. Das bedeutet bspw. für Platzhalter im Template "Default - New Ticket Dialog": <ul style="list-style-type: none"> • <i>Platzhalter für den Kontakt im Ticket:</i> <ul style="list-style-type: none"> • richtig: <KIX_CURRENT_ContactID> • falsch: <KIX_CURRENT_UserID> • Platzhalter für den aktuell angemeldeten Nutzer als Bearbeiter oder Verantwortlicher:



Spalte	Beschreibung
	<ul style="list-style-type: none">• richtig: <KIX_CURRENT_UserID>• Felder, welche auf andere Objekte wie Kontakte, Bearbeiter oder Priorität referenzieren, erwarten eine ID als Wert. Das heißt, bei der Verwendung von KIX Platzhaltern muss die ID angegeben werden (z. B. KIX_CURRENT_UserID>). Bei Dynamischen Feldern, der unter "Key" angegebene Wert.

Spalte	Beschreibung
<p>4</p> <p>Erweiterte Optionen</p>	<p>Feld "Hinweis": Sie können einen individuellen Hinweistext zum Feld hinterlegen. Damit können Sie bspw. die Agenten darüber informieren, welche Daten ein Dynamisches Feld erwartet.</p> <p>Wenn ein Hinweistext angegeben ist, wird am Feld ein Fragezeichen-Symbol angezeigt. Der Hinweistext wird nach Klick auf das Fragezeichen eingeblendet.</p>  <p><i>Abb.: Individueller Hinweistext am Feld "Kontakt"</i></p> <p>Der Hinweistext wird an der Feldkonfiguration der jeweiligen Vorlage gespeichert, sodass für jede Vorlage ein anderer Hinweistext zum Feld notiert werden kann.</p> <p>Die initialen Felder einer Vorlage enthalten bereits lokalisierte Hinweistexte, welche über das Weblate bereitgestellt werden. Bei Bedarf können Sie diese durch Ihren individuellen Text ersetzen.</p> <p>Die Übersetzung Ihrer Texte können Sie im Menü <i>KIX > Internationalisierung > Übersetzung</i> selbst pflegen (s. Übersetzungen (siehe Seite 153)). Die zu lokalisierenden Hilfetexte, kennzeichnen Sie mit vorangehendem <code>Translatable#:</code></p>  <p><i>Abb.: Mit vorangehendem "Translatable#" werden zu übersetzende Texte gekennzeichnet</i></p> <p>Hinweis: Per Workflow Rule hinzugefügte Felder werden aktuell nicht betrachtet.</p>

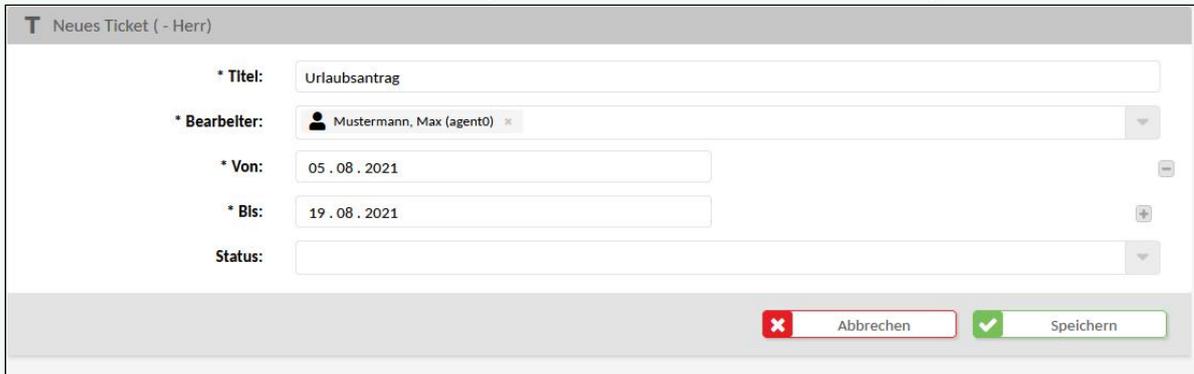
Spalte	Beschreibung
	<p>Editor: Sie können ein Array weiterer Konfigurationsoptionen für das Feld im JSON-Format angeben, z. B. spezifische Ladeoptionen, den zu ladenden Objekttyp usw.</p> <p>Abhängig vom gewählten Feldtyp werden verschiedene Optionen unterstützt. (s. unten: Hinweise zu Feldoptionen). Einige Konfigurationsoptionen werden automatisch gesetzt, damit die Felder korrekt funktionieren, z. B. für das Dynamische Feld " Betroffenes Asset".</p> <ul style="list-style-type: none"> • Option: Angabe des Attributs • Value: Wert des Attributs <p>Setzen Sie die Feldoptionen in eckige Klammern: [...]. Setzen Sie jeden Optionsblock in geschweifte Klammern {...} und trennen Sie mehrere Optionen durch Komma voneinander.</p> <div data-bbox="730 1048 1425 1137" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Beispiel</p> </div> <pre data-bbox="730 1137 1425 1512"> [{ "option": "SHOW_NEW_CONTACT", "value": 1 }, { "option": "SHOW_INVALID", "value": 0 }] </pre> <div data-bbox="730 1518 1425 1702" style="border: 1px solid #c8e6c9; padding: 5px; margin: 10px 0;"> <p> Tipp</p> <p>Nutzen Sie die Konfiguration der Vorlage "Default - New Ticket Template" als Anschauungsbeispiel.</p> </div> <div data-bbox="730 1709 1425 1933" style="border: 1px solid #fff9c4; padding: 5px; margin: 10px 0;"> <p> Hinweis</p> <p>In den Feldoptionen verwendete KIX Platzhalter (siehe Seite 153) werden vom Self Service Portal nicht aufgelöst.</p> </div>

Hinweise zur Feldauswahl

- Im Nutzungskontext "Agent" sind Kontakt, Kunde und Titel in der Konfiguration automatisch immer Pflichtfelder, da diese Felder zum Erstellen von Tickets benötigt werden.
- Im Nutzungskontext "Customer" können Kontakt und Kunde nicht konfiguriert werden. Diese Felder stehen nicht zur Auswahl.
- Ist in der Vorlage kein Kontakt konfiguriert, wird der erstellende Nutzer als Kontakt eingetragen.
- Ist in der Vorlage kein Kunde konfiguriert, wird die PrimaryOrg des Kontakts als Kunde eingetragen.
- Sind in der Vorlage Titel UND Artikelbetreff nicht konfiguriert bzw. angegeben, wird dieser aus "<Vorlagentitel> <YYYY-MM-DD hh:mm of current time>" automatisch generiert.
- Für Ticket-Pflichtfelder, die in der Vorlage nicht angegeben sind, werden folgende SysConfig-Schlüssel als Fallback verwendet:
 - **Typ:** *Ticket::Type::Default*
 - **Team:** *PostmasterDefaultTeam*
 - **Kanal:** *API::Operation::V1::ArticleCreate###Channel*
 - **Priorität:** *PostmasterDefaultPriority*
 - **Status:** *PostmasterDefaultState*
- Dynamisches Feld "Betroffene Services" im Self Service Portal: Es erfolgt keine Prüfung von Berechtigungen.
Das heißt: Sind Services "als festen Wert" oder "im Hintergrund" in einer Vorlage gesetzt, dann wird der Service gesetzt, auch wenn der Self Service Portal Nutzer nicht über die entsprechenden Berechtigungen verfügt.
- Enthält die Vorlage keinen Tickettitel, so wird dieser automatisch gesetzt, bestehend aus: Name der verwendeten Vorlage + Datum + Uhrzeit (z. B.: "Urlaubsantrag 11.11.2022 14:15")

✓ **Tipp**

Enthält die Vorlagenkonfiguration keine Angabe zum Kanal oder werden die Angaben im Hintergrund gesetzt, stehen keine ticket-/artikelspezifischen Eingabefelder zur Verfügung. Das ermöglicht, Tickets ohne Betreff, Artikelinhalt, Anhang usw. zu erstellen, z. B. Urlaubsantrag.



Hinweise zu den Feldoptionen

Abhängig vom Eingabefeld werden in den Feld Optionen verschiedene Attribute unterstützt. Einige Attribute werden im Hintergrund automatisch gesetzt und sind nicht konfigurierbar. (z. B. Organisation wird automatisch entsprechend gewähltem Kontakt gesetzt.). Diese sind nachfolgend nicht aufgeführt.

Spezifische Attribute für **Dropdown-Felder** (Status, Priorität, Bearbeiter, Verantwortlicher, Team, Typ, Organisation und Kontakt):

Attribut	Beschreibung	Beispiel
<p>AUTOCOMPL ETE</p>	<p>Steuert das Autocomplete-Verhalten für die Texteingabe in Auswahlfeldern (z. B. in Referenzfeldern wie "Verwandte Tickets", "Kontakt", u.a.m.)</p> <ul style="list-style-type: none"> • charCount: ¹ Legt fest, wie viele Zeichen eingegeben werden müssen, damit die Suche startet (Default: 3). • delay: Wartezeit in Millisekunden bis die Suche nach Eingabe des letzten Zeichens startet. Der Wert kann verändert werden, wenn die Suche zu früh oder zu spät startet. • limit: Anzeigelimit. Bestimmt die Anzahl der Einträge im Auswahlfeld. Das Limit kann bei Bedarf erhöht werden, wenn aufgrund eingeschränkter Nutzerrechte zu wenig oder keine Einträge geladen werden. • noResultsObjectName: ² Anzuzeigende Objekt-Bezeichnung, wenn keine Objekte gefunden wurden. Durch ein vorangestelltes "Translatable#" wird diese Bezeichnung automatisch übersetzt, sofern Sie ein entsprechendes Pattern im Menü "<i>Internationalisierung > Übersetzung</i>" hinterlegt haben. Wenn nichts angegeben ist, wird die Bezeichnung "Objekte" verwendet. 	<pre>{ "option": "AUTOCOMplete", "value": { "charCount": 2, "delay": 250, "limit": 25, "noResultsObjectName": "Translatable#Contacts" } }</pre>

Attribut	Beschreibung	Beispiel
AUTOCOMPL ETE_PRELOA D_PATTERN	Es wird eine Suche innerhalb der vorgeladenen Werte anhand der unter "value" eingegebenen Zeichenkette durchgeführt.	<pre>{ "option": "AUTOCOMplete_preload_Pattern", "value": "*Meier*" }</pre>
MULTISELEC T	Auswahl mehrerer Items ermöglichen value: 1 0 oder true false	<pre>{ "option": "MULTISELECT", "value": 1 }</pre>
FREETEXT	Eingabe von Freitext erlauben (z. B. bei E-Mail-Feldern) value: 1 0 oder true false	<pre>{ "option": "FREETEXT", "value": 1 }</pre>
COUNT_MIN	Mindestanzahl an Items, die gewählt sein muss. Bei Werten >1 wird das Feld ein Pflichtfeld.	<pre>{ "option": "COUNT_MIN", "value": 1 }</pre>
COUNT_MAX	Maximalanzahl von Items, die gewählt werden dürfen (z. B. maximale Anzahl betroffener Assets)	<pre>{ "option": "COUNT_MAX", "value": 10 }</pre>
SHOW_NEW_ CONTACT	Spezifische Option für das Feld "Kontakt". Entscheidet, ob die Aktion "Neuer Kontakt" angezeigt wird. value: 1 0 oder true false	<pre>{ "option": "SHOW_NEW_CONTACT", "value": 1 }</pre>

Attribut	Beschreibung	Beispiel
USE_OBJECT_SERVICE	Spezifische Option für das Feld "Team". Sorgt dafür, dass Teams als Baum angezeigt werden. value: 1 0 oder true false	<pre data-bbox="898 383 1238 600">{ "option": "USE_OBJECT_SERVICE", "value": 1 }</pre>

5.3.2 Die Standard-Vorlage

In KIX Pro basiert der Dialog "Neues Ticket" auf der initial ausgelieferten Standard-Vorlage "Default - Ticket New Template". Bei Bedarf können Sie diese Vorlage individuell anpassen und eigene Dynamische Felder integrieren (danach ggf. Browserfenster mit F5 aktualisieren).

Das Einbinden Dynamischer Felder in andere Ticketoberflächen wie Ticketdetails oder Home-Dashboard erfolgt über die SysConfig (s. KIX-Start-Handbuch).

Die Konfiguration der Standard-Vorlage basiert auf der Konfiguration im SysConfig-Schlüssel "*ticket-new-form-group-data*". Änderungen an der Standard-Vorlage beeinflussen die Konfiguration im SysConfig-Schlüssel nicht. Wird die Standard-Vorlage auf ungültig gesetzt oder kann sie nicht geladen werden, dient die Konfiguration des SysConfig-Schlüssels als Fallback.

Hinweis: Der Dialog "Ticket bearbeiten" ist eine [Ticketaktion](#) (siehe Seite 93) . Änderungen an diesem Dialog erfolgen daher im Menü *Workflow > Aktionen > Aktion "Ticket Edit"*.

5.3.2.1 Initiale Konfiguration der Standard-Vorlage

⚠ Derzeit besteht keine Möglichkeit, die Konfiguration der Vorlage "Default - Ticket New Template" auf den Auslieferungszustand zurückzusetzen. Ändern Sie daher die Vorlagenkonfiguration mit Bedacht!

Für den Bedarfsfall finden Sie nachfolgend die Ausgangskonfiguration der Vorlage.

Vorlageninformationen

Feld	Wert
Name	Default - New Ticket Template
Nutzungskontext	Agent
Kommentar	Default new ticket template for agents with role Ticket Agent.
Gültigkeit	gültig
Rollen	Ticket Agent

Eingabefelder

Pflichtfeld	Feld	Verhalten	Wert	Field Options	Hilfetext
<input checked="" type="checkbox"/>	Kontakt	Im Formular setzen	[leer]	<pre>[{ "option": "SHOW_NEW_CONTACT", "value": 1 }, { "option": "SHOW_INVALID", "value": 0 }]</pre>	Translatable# Helptext_Tickets_TicketCreate_Contact
<input type="checkbox"/>	Organisation	Im Formular setzen	[leer]	[leer]	Translatable# Helptext_Tickets_TicketCreate_Organisation
<input type="checkbox"/>	Betroffenes Asset	Im Formular setzen	[leer]	<pre>[{ "option": "FIELD_NAME", "value": "AffectedAsset" }]</pre>	Translatable# Helptext_Tickets_TicketCreate_AffectedAsset

Pflichtfeld	Feld	Verhalten	Wert	Field Options	Hilftext
<input checked="" type="checkbox"/>	Typ	Im Formular setzen	[leer]	<pre>[{ "option": "LOADINGOPTIONS", "value": { "filter": [{ "filterType": "AND", "operator": "EQ", "property": "ValidID", "type": "NUMERIC", "value": 1 }] } }]</pre>	Translatable# Helptext_Tickets_TicketCreate_Type
<input type="checkbox"/>	Betroffene Services	Im Formular setzen	[leer]	<pre>[{ "option": "FIELD_NAME", "value": "AffectedServices" }]</pre>	Translatable# Helptext_Tickets_TicketCreate_AffectedServices

Pflichtfeld	Feld	Verhalten	Wert	Field Options	Hilfetext
<input checked="" type="checkbox"/>	Priorität	Im Formular setzen	3 normal	<pre>[{ "option": "LOADINGOPTIONS", "value": { "filter": [{ "filterType": "AND", "operator": "EQ", "property": "ValidID", "type": "NUMERIC", "value": 1 }] } }]</pre>	Translatable# Helptext_Tickets_TicketCreate_Priority
<input type="checkbox"/>	SLA/ Service Vereinbarung	Im Formular setzen	[leer]	<pre>[{ "option": "OBJECT", "value": "SLA" }, { "option": "SHOW_INVALID", "value": 0 }]</pre>	Translatable# Helptext_Tickets_TicketCreate_SLA

Pflichtfeld	Feld	Verhalten	Wert	Field Options	Hilfetext
<input checked="" type="checkbox"/>	Kanal	Im Formular setzen	Notiz	[leer]	Translatable# Helptext_Tickets_TicketCreate_Channel
<input type="checkbox"/>	Im Kundenportal anzeigen	Im Formular setzen	Ja	[leer]	Translatable# Helptext_Tickets_TicketCreate_CustomerVisible
<input type="checkbox"/>	Zeit buchen	Im Formular setzen	[leer]	[{ "option": "STEP", "value": 1 }, { "option": "EXCEPTS_EMPTY", "value": 1 }, { "option": "UNIT_STRING", "value": "Translatable#Minutes" }]	Translatable# Helptext_Tickets_TicketCreate_TimeUnit

Pflichtfeld	Feld	Verhalten	Wert	Field Options	Hilfetext
<input checked="" type="checkbox"/>	Team	Im Formular setzen	[leer]	[{ "option": "USE_OBJECT_SERVICE", "value": 1 }]	Translatable# Helptext_Tickets_TicketCreate_Queue
<input type="checkbox"/>	Bearbeiter	Im Formular setzen	[leer]	[{ "option": "AUTOCOMPLETE", "value": 1 }, { "option": "LOADINGOPTIONS", "value": { "filter": [{ "filterType": "AND", "operator": "EQ", "property": "ValidID", "type": "NUMERIC", "value": 1 }, } "filterType": "AND", "operator":	Translatable# Helptext_Tickets_TicketCreate_Owner

Pflichtfeld Id	Feld	Verhalten	Wert	Field Options	Hilfetext
				<pre> "EQ", "property": "IsAgent", "type": "NUMERIC", "value": 1 }], "query": [["requiredPermission", "TicketRead,TicketCreate"]] }] </pre>	

Pflichtfeld	Feld	Verhalten	Wert	Field Options	Hilfetext
<input checked="" type="checkbox"/>	Status	Im Formular setzen	offen	<pre>[{ "option": "LOADINGOPTIONS", "value": { "filter": [{ "filterType": "AND", "operator": "EQ", "property": "ValidID", "type": "NUMERIC", "value": 1 }] } }]</pre>	Translatable# Helptext_Tickets_TicketCreate_State

5.3.3 Kontakte anhand der URL auflösen

Der Aufruf der Ticketerstell-Maske kann aus externen Anwendungen oder Diensten heraus erfolgen. Dies erfolgt durch Aufruf der entsprechenden URL (z. B. <http://localhost:20001/tickets?new>). Werden der URL zusätzliche Parameter mitgegeben, kann sowohl der Kontakt am neuen Ticket gesetzt werden als auch die gewünschte Ticketvorlage gewählt werden.

Beispiel: Bei Annahme eines Anrufs ruft die Telefonanlage bzw. Telefon-Client die KIX-URL auf und gibt der URL die Informationen zum Anrufer mit. KIX kann diese Informationen auflösen und anhand derer die gewünschte Ticketvorlage öffnen und den entsprechenden Kontakt am neuen Ticket setzen.

Voraussetzung: Die Telefonanlage unterstützt den Aufruf von URLs.

Die URL für den Aufruf muss so aussehen:

- KIX.Cloud: <http://your.agentenportal.kix.cloud/tickets?new&from=0123456789&templateId=2>
- KIX On-Premises: <http://localhost:20001/tickets?new&from=0123456789&templateId=2>

Mehrere Parameter werden durch "&" aneinander gereiht.

Parameter	Beschreibung	Beispiel
from	Erwartet Informationen des Kontakts, z. B. die Telefonnummer. Startet eine Volltextsuche im Kontakt. Der identifizierte Kontakt wird am neu zu erstellenden Ticket eingetragen. Werden mehrere Kontakte gefunden, wird stets der erste gefundene Kontakt am Ticket eingetragen.	http://localhost:20001/tickets?new&from=0123456789
templateId	ID der Ticketvorlage, welche beim URL-Aufruf geladen wird. Tipp: Navigieren Sie zum Ermitteln der ID zu <i>Workflow > Vorlagen</i> . Führen Sie den Mauszeiger über die gewünschte Vorlage und lesen Sie die ID in der Fußzeile des Browsers ab.	http://localhost:20001/tickets?new&from=0123456789&templateId=2

5.4 Vorlagengruppen

Im Menü *Workflow* > *Vorlagengruppen* können Sie Kategorien anlegen, um *Vorlagen* (siehe Seite 151) thematisch zu strukturieren. Die Zuordnung einer Vorlage zu einer Vorlagengruppe erfolgt beim Anlegen einer Vorlage.

Durch Zuordnung einer Vorlagengruppe zu einer übergeordneten Vorlagengruppe können Sie eine Baumstruktur aus Vorlagengruppen erzeugen. Die Sortierung erfolgt alphanumerisch. Sie können beliebig viele Ebenen anlegen. So können Sie einen Servicekatalog für einzelne Nutzergruppen (Agenten und Kunden) erstellen.

Vorlagengruppen können dupliziert werden (s. unten). Bei Bedarf können auch die direkt und indirekt untergeordneten Vorlagengruppen einbezogen werden.

Agenten finden den Vorlagenbaum inkl. der den Gruppen zugeordneten Vorlagen in der linken Sidebar. Dort kann die gewünschte Vorlage beim Anlegen eines neuen Tickets ausgewählt werden.

Durch die Gruppierung liegen auch die im Self Service Portal bereitgestellten Vorlagen in einer gut auffindbaren Struktur vor.

Es werden nur gültige Vorlagengruppen angezeigt, denen eine Vorlage zugeordnet ist. Ein Nutzer sieht nur die Vorlagen, auf die er Berechtigungen hat.

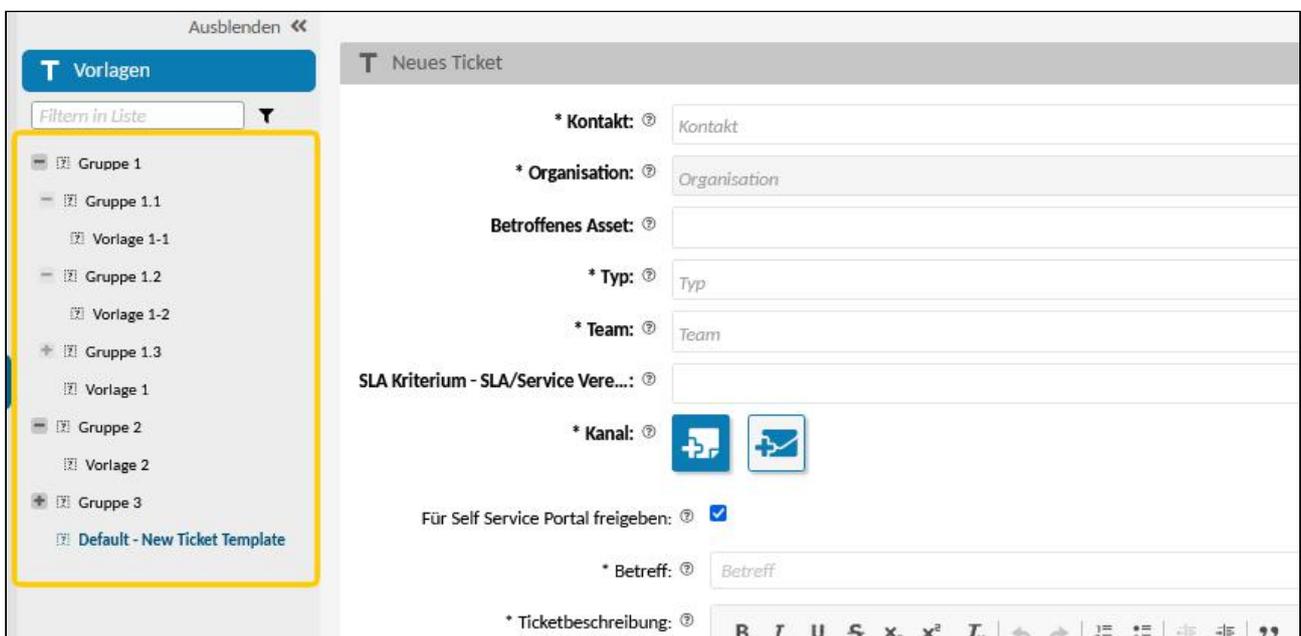


Abb: Die Vorlagengruppen inkl. Vorlagen in der Sidebar beim Anlegen eines neuen Tickets

Zum Anlegen einer neuen Vorlagengruppe navigieren Sie zu *Workflow* > *Vorlagengruppen* und klicken auf "Neue Gruppe". Füllen Sie den sich öffnenden Dialog aus (s. Tabelle unten) und speichern Sie Ihre Eingaben.

Um eine Vorlagengruppe zu bearbeiten, klicken Sie diese und ändern Sie die Angaben nach Erfordernis.



Zum Duplizieren einer Vorlage einschließlich eventuell untergeordneter Vorlagen wählen Sie die Vorlage mit einem Häkchen aus und klicken Sie auf "Duplizieren" (s. unten).

Zum Löschen einer Vorlagengruppe wählen Sie die Vorlagengruppe mit einem Häkchen aus und klicken Sie auf "Löschen". Eine Vorlagengruppe kann nur gelöscht werden, wenn ihr keine Vorlage zugeordnet ist. Ordnen Sie ggf. die betreffenden Vorlagen vorher einer anderen Vorlagengruppe zu.

Der Formular-Dialog besitzt folgende Eingabefelder:

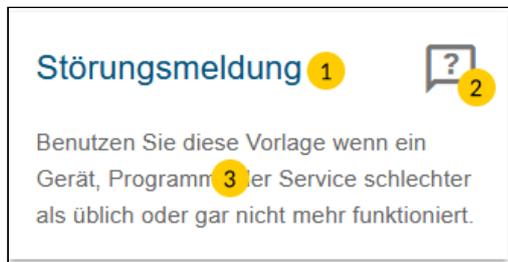


Abb.: Eine Vorlage im SSP

Feld	Beschreibung
1 Name	Bezeichnung der Vorlagengruppe im Agentenportal und im Self Service Portal
2 Icon	Optionales Icon, um die Vorlagengruppe zu kennzeichnen. Im Agentenportal wird das Icon dem Vorlagennamen vorangestellt. i Ein geändertes Icon wird erst nach Neuanmeldung und Aktualisierung des Browsercaches angezeigt.
Rang	Der Rang bestimmt die Sortierreihenfolge im Self Service Portal: <ol style="list-style-type: none"> 1. Vorlagengruppen basierend auf Rang 2. Vorlagengruppen ohne Rang, sortiert nach Name Die Sortierung im Agentenportal erfolgt alphanumerisch.
Übergeordnete Gruppe	Auswahl einer übergeordneten Vorlagengruppe, um einen Strukturbaum an Vorlagengruppen zu erstellen (optional).
3 Kommentar	Optionaler Kommentar, um die Vorlagengruppe näher zu beschreiben.

	Gültigkeit	<ul style="list-style-type: none"> • gültig: Vorlagen dieser Gruppe stehen den Nutzern entsprechend seiner Berechtigungen zur Verfügung. • (temporär) ungültig: Ist eine Vorlagengruppe ungültig, werden untergeordnete Strukturen (Untergruppen und Vorlagen) nicht zur Auswahl angeboten. Die Vorlagen dieser Gruppe stehen somit nicht zur Verfügung, auch wenn die Vorlage selbst gültig ist.
--	------------	---

5.4.1 Duplizieren von Vorlagengruppen

Sie können Vorlagengruppen duplizieren, bspw. um neue Servicekataloge auf Basis bestehender zu erzeugen. Sie können auswählen, ob die der Vorlage untergeordneten Vorlagengruppen ebenfalls dupliziert werden sollen. Es wird nur die Konfiguration der Vorlagengruppen dupliziert; die den Vorlagengruppen zugeordneten Vorlagen werden nicht mit dupliziert.

Zum Duplizieren wählen Sie die betreffende Vorlagengruppe mit einem Häkchen aus. Sie können nur Klicken Sie dann auf "Duplizieren im Kopf der Übersicht. Die Schaltfläche ist erst aktiv, wenn eine Vorlagengruppe angehakt ist. Der sich öffnende Dialog entspricht dem Dialog "Neue Gruppe" mit folgenden zusätzlichen Feldern:

Feld	Beschreibung
Rekursiv kopieren	Wenn angehakt, werden auch die Kopien aller direkt und indirekt untergeordneten Vorlagengruppen erzeugt. Wenn nicht angehakt, wird nur die ausgewählte Vorlagengruppe - ohne Unterelemente - dupliziert.
Präfix für Duplikate	Zur Kennzeichnung der Duplikate werden diese mit einem Präfix versehen. Default ist: "Kopie von". Sie können ein anderes Präfix festlegen.



Feld	Beschreibung
Übergeordnete Gruppe	Legen Sie optional fest, welcher Vorlagengruppe das Duplikat/die Duplikate untergeordnet werden sollen. Ist nichts definiert, entstehen die Duplikate auf der selben Hierarchieebene wie ihre Originale.

6 Services und SLA

Services werden in KIX als Betriebsmittel und Ressourcen zur Erbringung anderer Services aufgefasst. Daher werden sie in der Assetklasse "Service" angelegt und gepflegt. Die Assetklasse "Service" wird initial mit KIX Pro ausgeliefert. Am Ticket können Services im Feld "Betroffene Services" ausgewählt werden (analog zu "Betroffene Assets").

Services können expliziten SLAs zugeordnet werden, sodass diese Zuordnungen automatisiert am Ticket verwendet werden können. Die Zuordnung von Services zu SLAs und die Festlegung, unter welchen Bedingungen sie am Ticket zur Verfügung stehen erfolgt im Modul "Serviceverträge" . Lesen Sie dazu auch im Kapitel "Servicekatalog Management" des KIX 18 Anwender-Handbuchs nach.

Sie können die Auswahl von Services und SLAs am Ticket administrativ steuern. Diese Informationen finden Sie unter [Serviceverträge](#) (siehe Seite 188) im KIX 18 Pro Admin-Handbuch.

6.1 Serviceverträge

Ein Servicevertrag legt fest, welcher Service und welches SLA (Service Level Agreement) unter welchen Bedingungen am Ticket zur Verfügung steht. Das Anlegen und Verwalten der Serviceverträge erfolgt im Modul

Serviceverträge .

Nachfolgend finden Sie

Informationen zu administrativen Konfigurationseinstellungen der Serviceverträge. Eine detaillierte Beschreibung des Moduls und zur Anwendung von Serviceverträgen finden Sie im KIX Pro Anwenderhandbuch.

Inhalt auf dieser Seite:

- [Aktualisierung des Servicebaums](#) (siehe Seite 188)
- [Steuerung der Auswahl "Betroffene Services"](#) (siehe Seite 189)
- [Betroffene Assets vs. Betroffene Services](#) (siehe Seite 191)
- [Kritikalität](#) (siehe Seite 191)
- [Serviceverträge im- und exportieren](#) (siehe Seite 191)
- [Verwendung post-produktiver Services](#) (siehe Seite 192)

6.1.1 Aktualisierung des Servicebaums

Der Servicebaum im Explorer des Moduls *Serviceverträge* enthält alle Assets der Assetklasse "Service" und listet sie hierarchisch auf. Wird ein neues Asset der Klasse "Service" angelegt, wird im Backend der Servicebaum neu aufbereitet.

Alternativ dazu kann der Servicebaum mittels Konsolenkommando neu aufgebaut werden. Starten Sie dazu unter *System > Konsole* das

Kommando `Console::Command::Maint::Service::UpdateServiceCatalog`. Danach ist der Servicebaum aktualisiert.

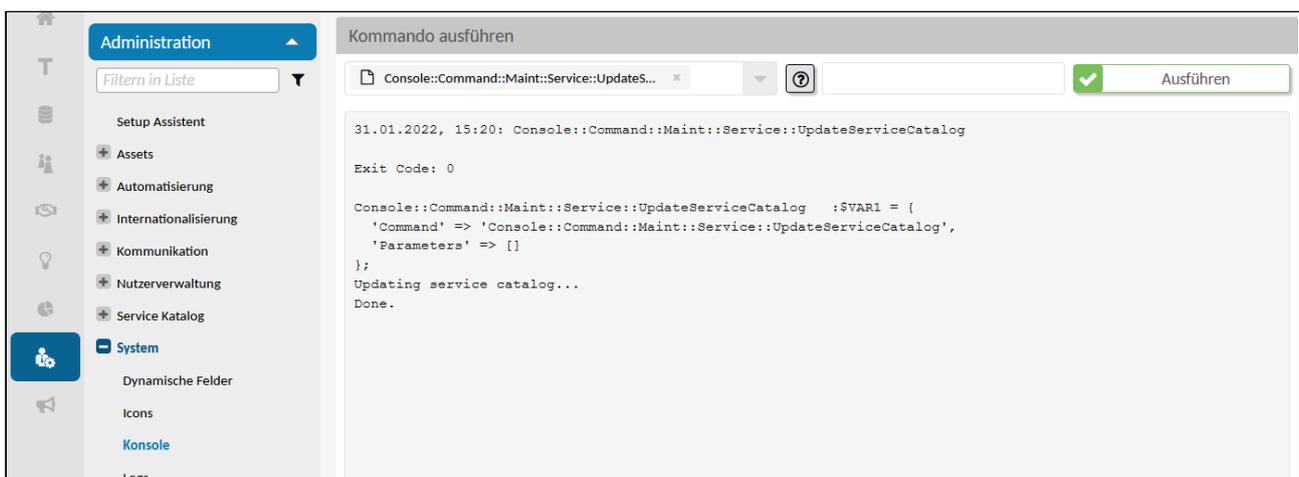


Abb.: Aktualisierung Servicebaum via Konsole

6.1.2 Steuerung der Auswahl "Betroffene Services"

Konfigurationsschlüssel:	Agentenportal	service-contract-configuration
	Self Service Portal	service-contract-configuration-ssp

Mit den o.a. SysConfig-Schlüsseln (Menü "System > SysConfig") können Sie die Auswahl im Dynamischen Feld "Betroffene Services" steuern. Das Verhalten kann getrennt für das Agentenportal und für das Self Service Portal festgelegt werden.

Die Konfiguration wirkt sich grundlegend auf das Dynamische Feld aus, ungeachtet dessen, ob es im Ticket oder in den Konfigurationen von Vorlagen oder Aktionen eingebunden ist.

Konfiguration:

Parameter	Beschreibung
enabled	<p>Aktivierung/Deaktivierung der Serviceverträge im Ticket.</p> <ul style="list-style-type: none"> <code>true</code> (Default): <ul style="list-style-type: none"> In der Auswahl steht der Service-Baum zur Verfügung. Die im Ticket zur Auswahl stehenden Services und SLA werden anhand der im Ticket gesetzten Parameter gefiltert (Kontakt, Organisation, Typ, Priorität). Wird im Ticket einer dieser Parameter geändert, so wird für das Dropdown der Baum neu ermittelt (Alternative: s. oben "Aktualisierung des Servicebaums") Der Filter berücksichtigt auch die in einer Vorlage als "festen Wert" oder "im Hintergrund" gesetzten Werte. <code>false</code>: <ul style="list-style-type: none"> Das Feld "Betroffene Services" entspricht der normalen Suche nach einem Asset (Volltext bzw. mit *). Es werden die Assets der Klasse Service gezogen; die darin definierten Abhängigkeiten werden dabei berücksichtigt. Die Darstellung im Auswahlfeld erfolgt als Liste und nicht als Baumstruktur. Es gilt die Konfiguration des Dynamischen Feldes "Betroffene Services".

Parameter	Beschreibung
singleSelection	<p>Einfach- oder Mehrfachauswahl von Services am Ticket. Die zur Auswahl stehenden Services basieren auf der Konfiguration der Serviceverträge.</p> <ul style="list-style-type: none"> • <code>true</code> : Es kann max. 1 "Betroffener Service" ausgewählt werden. • <code>false</code> (Default): Erlaubt die Mehrfachauswahl von Services.

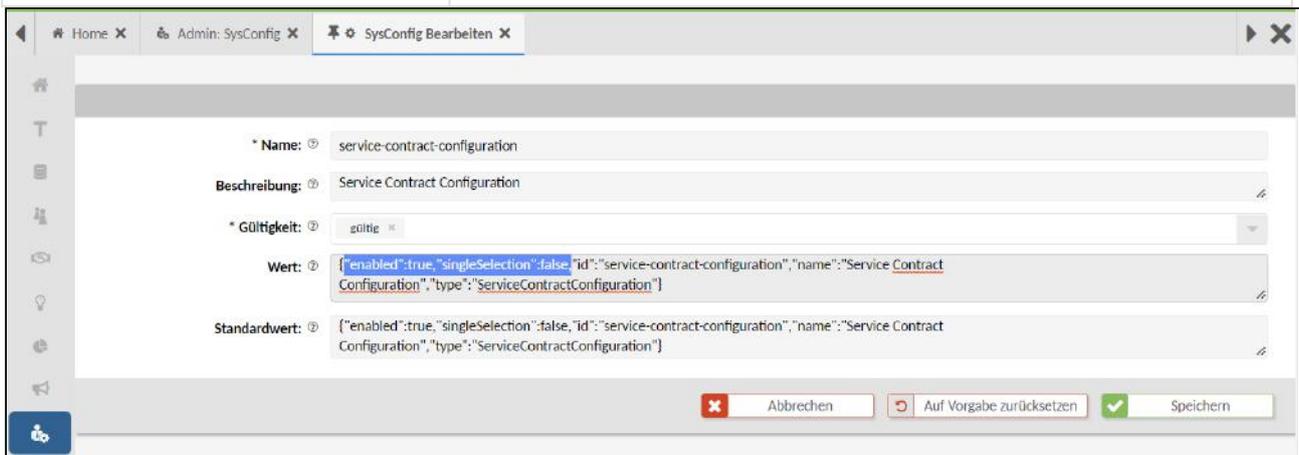


Abb.: SysConfig-Schlüssel "service-contract-configuration"

Hinweis

KIX Pro ermöglicht Ihnen im Menü *Ticket > Vorlagen* die komfortable Anpassung von Vorlagen. Wenn Sie Service Verträge nutzen, ist die Änderung der Feldreihenfolge in der Standard-Vorlage nicht empfehlenswert, da dies zu unplausiblen Vorgehen bei der Ticketerstellung führen könnte. Die initiale Feldreihenfolge resultiert aus der logischen Abfolge der Abhängigkeiten und dem Arbeitsfluss:

- Die Auswahl von Kontakt, Organisation und Tickettyp beeinflussen die Services. Daher stehen diese Felder oberhalb des Feldes "Affected Service".
- Die gewählte Priorität beeinflusst die möglichen SLA. Daher ist die Priorität direkt oberhalb des Feldes "SLA/Service Agreement" angeordnet.

Muss die Feldreihenfolge dennoch geändert werden, beeinflusst dies nicht die Auswahlmöglichkeiten der Services und SLA am Ticket.

6.1.3 Betroffene Assets vs. Betroffene Services

- **AffectedServices:** Es greifen die im Modul "Serviceverträge" definierten Zuordnungen.
 - Default Services sind Services, an denen weder Organisation, Kontakt, Priorität noch Typ hinterlegt sind. Sie sind damit auch für jeden SSP-Nutzer verfügbar.
Wurden Services per "Serviceverträge" eingeschränkt, so können diese nur noch entsprechend ihrer Konfiguration (Kontakt, Organisation, Priorität und Typ) im SSP sowie im AP ausgewählt werden.
- **AffectedAssets:** Es gelten die Zuordnungen und Sichtbarkeiten entsprechend SysConfig-Schlüssel "*AssignedConfigItemsMapping*" (s. auch: Konfigurationseinstellungen und [Sichtbarkeiten im SSP steuern](#) (siehe Seite 211))

6.1.4 Kritikalität

Einem Asset der Klasse Service muss eine Kritikalität zugewiesen werden. Die Kritikalität definiert die Systemrelevanz des Services. Initial kann aus 4 Kritikalitätsstufen ausgewählt werden:

1. existenziell
2. geschäftskritisch
3. Unterstützungsprozess
4. keine/geringfügig.

Sie können diese Kritikalitätsstufen ändern oder weitere hinzufügen. Dies erfolgt im Admin Modul im Menü *Assets > General Catalog*. Wie dem General Catalog Einträge hinzugefügt oder geändert werden, finden Sie im Kapitel General Catalog des Admin Handbuchs von KIX 18 Start.

6.1.5 Serviceverträge im- und exportieren

Sie können Services inkl. deren Zuordnungen zu SLA-Kriterien via CSV-Datei im- und exportieren. Das ermöglicht Ihnen eine externe Massendatenpflege sowie den Transfer von Serviceverträgen von einer KIX 18 Testumgebung in eine KIX 18 Produktivumgebung bzw. von einer On-Premises-Installation in die KIX 18 Cloud.

Öffnen Sie dazu das Modul "Serviceverträge". Exportieren Sie dazu zunächst mindestens 1 bestehenden Servicevertrag als CSV-Datei. Ändern Sie diese nach Erfordernis und importieren Sie sie anschließend ins Zielsystem.

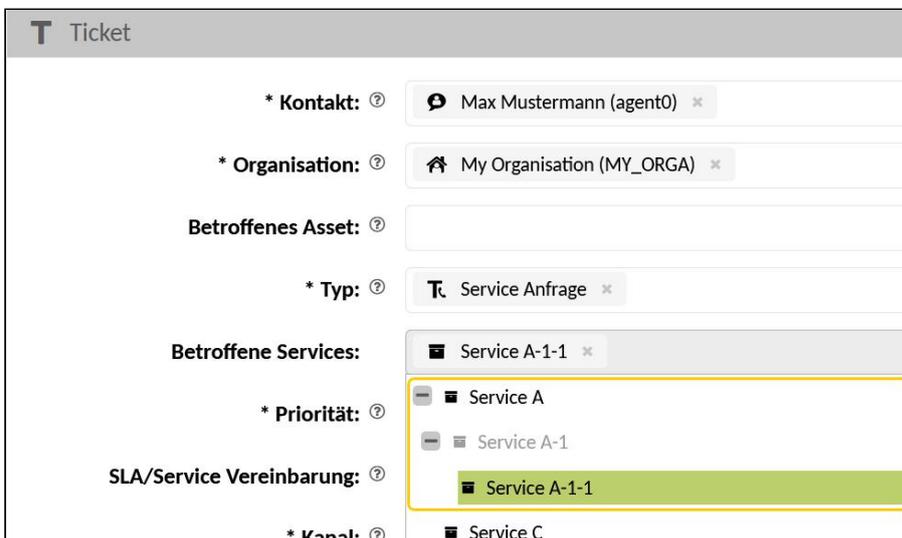
Weiterführende Hinweise im KIX 18 Start Anwenderhandbuch:

- Import/Export durchführen: Allgemeine Funktionserläuterungen
- Hinweise für Import: Serviceverträge - Anlegen und Bearbeiten

6.1.6 Verwendung post-produktiver Services

Ein Service ist post-produktiv, wenn er den Verwendungsstatus "Außer Dienst" oder "Inaktiv" hat (s. auch Konfigurationseinstellungen). Dennoch können post-produktive Services teilweise verwendet werden:

- **Im Modul Asset:** Post-Produktive Services sind - wie alle post-produktiven Assets - nicht sichtbar und können nur über die Komplexsuche gefunden werden.
- **Im Modul Service:** Der Services-Baum enthält alle Services. Post-produktive Services sind ausgegraut gekennzeichnet, können jedoch verwendet werden.
- **Im Ticket:** Post-produktive Services werden ausgegraut angezeigt, sofern sie produktive/auswählbare Sub-Services besitzen. Sie können jedoch nicht ausgewählt werden. Ausgewählt werden können produktive und pre-produktive Services und Sub-Services in Abhängigkeit der im Servicevertrag definierten Bedingungen.



The screenshot shows a 'Ticket' form with the following fields and values:

- * Kontakt:** Max Mustermann (agent0)
- * Organisation:** My Organisation (MY_ORGA)
- Betroffenes Asset:** (empty)
- * Typ:** Service Anfrage
- Betroffene Services:** Service A-1-1
- * Priorität:** (empty)
- SLA/Service Vereinbarung:** (empty)
- * Kanal:** Service C

In the 'Betroffene Services' section, a dropdown menu is open, showing a tree structure of services:

- Service A (greyed out)
- Service A-1 (greyed out)
- Service A-1-1 (highlighted in green)
- Service C (greyed out)

 A yellow box highlights the 'Service A-1-1' option, indicating it is the selected service.

Abb.: Post-produktive und produktive Services im Ticket

6.2 Service-Level-Agreements (SLA)

In KIX Pro können Sie Service Level Agreements (SLA) definieren und diese an Tickets oder Assets zuweisen. SLA werden im Menü *Service Katalog > Service-Level-Agreements (SLA)* angelegt und konfiguriert.

Service Level Agreements sind - vereinfacht gesagt - Vereinbarungen darüber, in welcher Zeit eine Serviceleistung zu erbringen ist.

Ein SLA setzt sich aus zwei SLA-Kriterien zusammen: Reaktion und Lösung. Beide Kriterien basieren aus einer Vorgabedauer in Geschäftsminuten sowie einer Erinnerungsschwelle in Prozent. Beträgt die Vorgabedauer 120 Minuten, muss das Kriterium 120 Geschäftsminuten nach Ticketerstellung erfüllt werden, um keine SLA-Verletzung zu erhalten. Dabei definiert der in KIX gewählte Kalender die Geschäftszeiten.

Wird einem Ticket ein SLA zugeordnet, werden die im SLA definierten Zielvorgaben am Ticket hinterlegt. Dabei werden konkrete Werte für Startzeit, Erinnerungszeitpunkt und Zielzeit berechnet. Diese Angaben stehen dann am Ticket zur Verfügung.

Inhalte auf dieser Seite:

- [SLA Kriterien und -Attribute \(siehe Seite 193\)](#)
- [Schema einer SLA \(siehe Seite 196\)](#)
- [Ein SLA anlegen oder bearbeiten \(siehe Seite 196\)](#)
- [SLA an Tickets \(siehe Seite 199\)](#)
- [SLA an Assets / Geräte-SLA \(siehe Seite 200\)](#)
- [Benachrichtigungen für SLA \(siehe Seite 200\)](#)
- [Verwendung von SLA in KIX Platzhaltern \(siehe Seite 201\)](#)
- [Konfiguration der Erfüllungszeit \(siehe Seite 201\)](#)
- [Kalender Konfiguration \(siehe Seite 202\)](#)

6.2.1 SLA Kriterien und -Attribute

Folgende SLA Kriterien und -Attribute werden an Tickets gespeichert. Diese Finden Sie in der Lane "Service-Level-Agreement (SLA)" in der Detailansicht eines Tickets.

SLA Kriterium - Attribut	Erklärung
SLA/Service Vereinbarung	Name des SLA
<i>Reaktion</i>	<i>Bezieht sich auf die erste Reaktion eines Agenten, nachdem ein Ticket erstellt und das SLA zugewiesen wurde.</i>

SLA Kriterium - Attribut	Erklärung
Startzeit	Startzeitpunkt der Reaktionszeit. Entspricht dem Zeitpunkt, an dem das Ticket erstellt wurde.
Zielzeit	Zielzeitpunkt der Reaktionszeit. Ergibt sich aus dem Zeitpunkt, an dem das Ticket erstellt wurde plus die im SLA eingestellte Zeitspanne für die Reaktionszeit.
Erfüllungszeit	Zeitpunkt, an dem die tatsächliche Reaktion erfolgte. Entspricht dem, was im Job unter "SLA First Response Time Fulfillment" konfiguriert ist. Initial: Zeitpunkt, an dem vom Agenten ein für den Kunden sichtbarer Artikel erstellt wurde (s. unten).
Erinnerungszeit	Erinnerungszeitpunkt der Reaktionszeit. Ergibt sich aus dem Zeitpunkt, an dem das Ticket erstellt wurde, plus dem unter "Benachrichtigen nach" eingestellten Prozentsatz der angegebenen Reaktionsdauer.
Abweichung (Servicezeit)	Absolute Zeitdifferenz zwischen Zielzeit und Erfüllungszeit (beinhaltet auch "über Nacht" oder "Wochenende"). Für "Servicezeit" ist auch der Begriff "Geschäftszeit" gebräuchlich.
Abweichung	Absolute Zeitdifferenz zwischen Zielzeit und Erfüllungszeit in Geschäftsminuten (ohne "Nacht").
Verletzung	"Ja" oder "Nein"; zeigt auf einen Blick, ob die im SLA definierte Verpflichtung eingehalten wurde.
Lösung	<i>Bezieht sich darauf, das Tickets zu erledigen, dem das SLA zugewiesen wurde</i>
Startzeit	Startzeitpunkt der Lösungszeit. Entspricht dem Zeitpunkt, an dem das Ticket erstellt wurde.
Zielzeit	Zielzeitpunkt der Lösungszeit. Ergibt sich aus dem Zeitpunkt, an dem das Ticket erstellt wurde plus die im SLA eingestellte Zeitspanne für die Lösungszeit.

SLA Kriterium - Attribut	Erklärung
Erfüllungszeit	Zeitpunkt, an dem das Ticket tatsächlich gelöst wurde. Entspricht dem, was im Job "SLA Solution Time Fulfillment" konfiguriert ist. Initial: Zeitpunkt, an dem das Ticket geschlossen wird. (s. unten)
Erinnerungszeit	Erinnerungszeitpunkt der Lösungszeit. Ergibt sich aus dem Zeitpunkt, an dem das Ticket erstellt wurde, plus dem unter "Benachrichtigen nach" eingestellten Prozentsatz der angegebenen Reaktionsdauer.
Abweichung (Servicezeit)	Absolute Zeitdifferenz zwischen Zielzeit und Erfüllungszeit (beinhaltet auch "über Nacht" oder "Wochenende"). Für "Servicezeit" ist auch der Begriff "Geschäftszeit" gebräuchlich.
Abweichung	Zeitdifferenz zwischen Zielzeit und Erfüllungszeit in Geschäftsminuten (ohne "Nacht").
Verletzung	"Ja" oder "Nein", zeigt auf einen Blick, ob die im SLA definierte Verpflichtung eingehalten wurde.

Info

Die SLA Kriterien und -Attribute stehen in der Ticket-Komplexsuche, in Filtern für Benachrichtigungen und Jobs, in Job-Aktionen und in Platzhaltern (s. unten) zur Verfügung.

6.2.2 Schema einer SLA

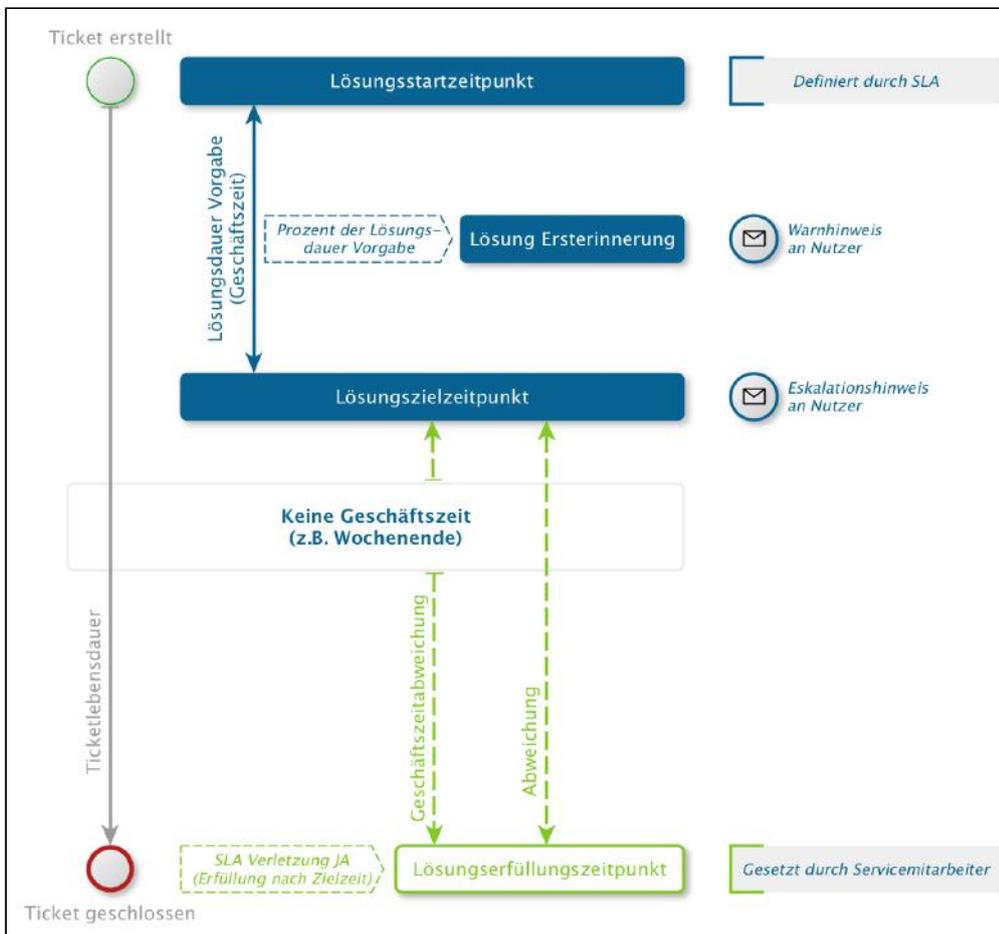


Abb.: Visualisierung SLA Attribute zum Kriterium Lösung

6.2.3 Ein SLA anlegen oder bearbeiten

Im Auslieferungszustand enthält KIX Pro bereits SLAs. Deren Funktion wird in den Bereichen *SLA an Tickets* und *SLA an Assets* weiter unten erläutert. Um ein neues SLA anzulegen, klicken Sie in der Übersichtstabelle auf die Aktion "Neues SLA". Füllen Sie das Formular aus (s. Tabelle unten) und setzen Sie die Gültigkeit auf "gültig". Speichern Sie das SLA mit "Speichern". Ihr neues SLA wird nun in der Übersichtstabelle angezeigt.

Zum Bearbeiten oder um Details zu sehen, klicken Sie in der Übersichtstabelle auf den gewünschten SLA. Es öffnet sich der Dialog "SLA bearbeiten". Ändern Sie die Angaben und übernehmen Sie die Änderungen mit "Speichern". Ihre Änderungen sind sofort wirksam. Ggf. müssen die Agenten ihre Browserseite aktualisieren, damit die Änderungen auch für sie sichtbar sind.

Der Formular-Dialog enthält u.a. folgende Eingabefelder:

Feld	Beschreibung
Name	Geben Sie dem SLA eine aussagekräftige und eindeutige Bezeichnung. (z.B.: "Firma Mayer - Gold - Major Incident", "Firma Mayer - Gold - Service Request")
Kalender	Wählen Sie den Kalender, der als Grundlage für die Berechnung der SLA-Zeiten dient. Sie können die Kalender über die SysConfig konfigurieren (s. unten)
Reaktionsdauer	Geben Sie die Reaktionsdauer in Geschäftsminuten an, die dem SLA zugrunde liegt. Die Eingabe von "0" definiert <u>keine</u> Reaktionsdauer, z. B. wenn Sie nur mit Lösungszeiten und nicht mit Reaktionszeiten arbeiten.
Benachrichtigung	Legen Sie fest, wann Sie in Tickets mit diesem SLA benachrichtigt werden wollen. Beispiel "50%": Ist an einem Ticket mit diesem SLA die Hälfte der oben eingetragenen Zeit vergangen, erhalten Sie eine Erinnerungsbenedachrichtigung.
Aussetzende Tickets	Geben Sie die Status an, die dazu führen, dass die Reaktionszeit am Ticket pausiert (optional).

Feld	Beschreibung
Lösungsdauer	<p>Geben Sie die Lösungsdauer an, die dem SLA zugrunde liegt (in Geschäftsminuten).</p> <p>Die Eingabe von "0" definiert <u>keine</u> Lösungsdauer, z. B. wenn Sie nur mit Reaktionszeiten und nicht mit Lösungszeiten arbeiten.</p>
Benachrichtigung	<p>Legen Sie fest, wann Sie in Tickets mit diesem SLA benachrichtigt werden wollen.</p> <p>Beispiel "50%": Ist an einem Ticket mit diesem SLA die Hälfte der oben eingetragenen Zeit vergangen, erhalten Sie eine Erinnerungsbenedachrichtigung.</p>
Aussetzen des Tickets	<p>Geben Sie die Status an, die dazu führen, dass die Lösungszeit am Ticket pausiert (optional).</p>

Feld	Beschreibung
SLA Startzeitpunkt	<p>Definiert den Start-Zeitpunkt für die Berechnung der SLA-Zeiten.</p> <p>Ist nichts angegeben oder ist das referenzierte Zeit-Feld leer bzw. undefiniert, so wird als Startzeitpunkt der Zeitpunkt der Ticketerstellung gesetzt.</p> <p>Mögliche Auswahlwerte sind:</p> <ul style="list-style-type: none"> • Ticket Create Time (Zeitpunkt der Ticketerstellung) <ul style="list-style-type: none"> • Standard- und Fallback-Wert • SLA Assignment Time (Zeitpunkt, an dem der SLA gesetzt wird) <ul style="list-style-type: none"> • Bei nachträglichem Wechsel des SLA oder Wechsel des Tickets in ein anderes Team kann somit die Lösungs-/Reaktionszeit neu berechnet werden. Die bisherigen Zeiten werden zurückgesetzt. • jedes gültige Dynamische Feld vom Feldtyp "Date/Time" mit Objekttyp "Ticket" <ul style="list-style-type: none"> • Die Startzeit (und damit auch die geforderte Lösungszeit) kann somit von der in einem Dynamischen Feld angegebenen Zeit abhängig gemacht werden. Bei Änderung der Zeitangaben im Dynamischen Feld werden Reaktions- und Lösungszeit neu berechnet. <p>Hinweis: Die Aktualisierung der Zeiten am Ticket erfolgt in festgelegten Zeitabständen. Nach Wechsel/Änderung des SLA oder nach Änderung der Zeit in einem Dynamischen Feld kann es daher vorkommen, dass die Darstellung am Ticket nicht sofort korrekt angezeigt wird. Um die Zeiten manuell zu aktualisieren, können Sie den Konsolen-Befehl <code>" Console::Command::Maint::Ticket::EscalationCheck "</code> ausführen.</p>
Gültigkeit	<ul style="list-style-type: none"> • gültig: Der SLA kann verwendet werden. • ungültig/temporär ungültig: Der SLA kann keinem Ticket zugewiesen werden. Für bestehende Tickets bleibt das SLA erhalten und hat weiterhin Gültigkeit. Werden die Ticket Informationen zur Bearbeitung geöffnet, weist eine rote Feldumrandung auf das Fehlen des SLA hin. Ein neues, gültiges SLA kann ausgewählt werden.

6.2.4 SLA an Tickets

Beim Erstellen oder Bearbeiten eines Tickets sind über ein Dropdown alle gültigen SLA auswählbar. Die SLA Kriterien werden wie oben beschrieben am Ticket gesetzt. Soll ein Ticket ohne SLA erstellt werden, kann das standardmäßig ausgelieferte SLA "Keine Eskalation" gesetzt werden.

Wird am Ticket die SLA "SLA nach betroffenen Service und Priorität" gewählt, werden die Zielkriterien aller auf Basis von Services, Organisation, Kontakt, Typ und Priorität zulässigen SLAs ermittelt und die jeweils kürzeste Zielzeit am Ticket hinterlegt.

Die Reaktions- bzw. Lösungszeit am Ticket kann pausieren, bspw. wenn auf die Rückmeldung des Kunden gewartet wird. Das Pausieren ist an den Status eines Tickets gekoppelt. Welche Status das Pausieren auslösen, können Sie beim Anlegen des SLA festlegen (Feld "Auszusetzende Tickets"). Legen Sie ggf. zusätzliche Ticket-Status an, z. B. "Warten auf Rückantwort" (s. Kapitel Status). Verlässt das Ticket diesen Status, läuft die Zeit für das entsprechende Kriterium weiter. Der mit KIX initial ausgelieferte Job "SLA Time Suspension" prüft den Status und steuert die Anzeige in der Lane "Service-Level-Agreement (SLA)".

6.2.5 SLA an Assets / Geräte-SLA

Um SLA an Assets zu setzen, steht für die Asset-Klassendefinition der Attribut-Typ "SLAReference" zur Verfügung. Ist dieses Attribut in der Klassendefinition eines Assets enthalten, kann beim Erstellen und/oder Bearbeiten pro Asset genau ein SLA gesetzt werden. Damit an einem Ticket der SLA eines oder mehrerer Assets hinterlegt wird, muss der initial mit ausgelieferte SLA "SLA nach betroffenen Asset/s" im Ticket gewählt werden.

Bei Berechnung der konkreten SLA-Zielvorgaben werden alle dem Ticket zugeordneten Assets berücksichtigt (Feld "Betroffene Assets"). Aus allen durch die Assets vorgegebenen SLA Kriterien werden die "strengsten" Kriterien ausgewählt und am Ticket zur Berechnung der Zielvorgaben verwendet. Es ist also möglich, dass die Reaktionszeit von einem Asset und die Lösungszeit eines anderen Assets am Ticket wirksam werden. Damit wird sicher gestellt, dass alle SLA-Vorgaben erfüllt werden können.

6.2.6 Benachrichtigungen für SLA

Für die Konfiguration von Benachrichtigungen, die im Zusammenhang mit SLA stehen, stehen entsprechende Events zur Verfügung. Bspw. um Benachrichtigungen zu versenden, wenn ein SLA Kriterium geändert wird oder ein Ticket zu eskalieren droht. Folgende vier Benachrichtigungen werden initial ausgeliefert:

- *Notify First Response Warning To Team*
 - Wird versendet, wenn an einem Ticket die Erinnerung für die Reaktionszeit des SLA erreicht ist.
 - Empfänger: alle Agenten mit *Update* -Berechtigung für das Ticket.
- *Notify First Response Escalation to Team*
 - Wird versendet, wenn an einem Ticket die Reaktionszeit des den SLA erreicht ist.
 - Empfänger: alle Agenten mit *Update* -Berechtigung für das Ticket.
- *Notify Solution Warning To Team*
 - Wird versendet, wenn an einem Ticket die Erinnerung für die Lösungszeit des SLA erreicht ist.
 - Empfänger: alle Agenten mit *Update* -Berechtigung für das Ticket.
- *Notify Solution Escalation To Team*
 - Wird versendet, wenn an einem Ticket die Lösungszeit des SLA erreicht ist.
 - Empfänger: alle Agenten mit *Update* -Berechtigung für das Ticket.

6.2.7 Verwendung von SLA in KIX Platzhaltern

Die SLA Kriterien können für Platzhalter nach dem Muster "<KIX_TICKET_SLACriteria_Criteria_Attribute>" verwendet werden.

Hinweis

Beim Neu-Erstellen eines Tickets können nur die Platzhalter "KIX_TICKET_SL Aid" und "KIX_TICKET_SLA" verwendet werden.

Hintergrund: Das Setzen der SLA Attribute an Tickets läuft über ein Event. Die SLA Attribute sind also erst vorhanden, wenn das Ticket gespeichert wurde. Vorher kann keine Platzhalter-Ersetzung erfolgen. Das trifft auch zu, wenn beim Bearbeiten eines Tickets das SLA geändert wird.

6.2.8 Konfiguration der Erfüllungszeit

KIX PRO verfügt initial über zwei Jobs mit SLA Kriterien. Mit diesen können Sie konfigurieren, was als "Erfüllung" gilt. Sie können die Konfiguration der Jobs entsprechend Ihrer Geschäftsprozesse anpassen (vgl. Kapitel "Einen Job anlegen oder bearbeiten").

Job	Konfiguration
SLA First Response Time Fulfillment	<p>Dieser Job setzt einen Zeitstempel in ein Ticket, wenn ein Artikel erstellt wird, der im Self Service Portal sichtbar ist. Er definiert damit den Zeitpunkt der ersten Antwort für die SLA-Erfüllung. Sie können den Job nach Bedarf nachkonfigurieren.</p> <p>Die Aktion "Erfüllungszeit setzen" sollte nicht entfernt werden, sonst wird der Zeitstempel nicht gesetzt.</p>
SLA Solution Time Fulfillment	<p>Dieser Job setzt einen Zeitstempel in ein Ticket, wenn ein Ticket den Status "entfernt", "geschlossen" oder "zusammengefasst" erhält. Er definiert damit den Zeitpunkt der Lösung für die SLA-Erfüllung. Sie können den Job nach Bedarf nachkonfigurieren.</p> <p>Die Aktion "Erfüllungszeit setzen" sollte nicht entfernt werden, sonst wird der Zeitstempel nicht gesetzt.</p>

Job	Konfiguration
SLA Time Suspension	<p>Dieser Job veranlasst das Pausieren/Fortsetzen der Reaktions- und Lösungszeit am Ticket anhand des im Ticket gesetzten Status.</p> <p>Welche Status dazu führen, dass Reaktions- bzw. Lösungszeit pausieren, kann beim Anlegen des SLA angegeben werden. Die im Job verwendete Macro Action "SLA Criterion Suspend/Resume" gleicht diese Status mit dem am Ticket gesetzten Status ab und veranlasst das Pausieren/Fortsetzen der Reaktions- bzw. Lösungszeit.</p> <p>Wird am pausierenden Ticket der Status geändert, verlässt das Ticket die Pause; die Zeit für das entsprechende Kriterium läuft weiter. Es erfolgt eine Neuberechnung der Zeit. Die Dauer der Pause wird zur ursprünglichen Zeit hinzugefügt.</p> <p>Die Konfiguration des Jobs sollte nicht geändert werden.</p>

6.2.9 Kalender Konfiguration

KIX wird initial mit 9 Kalendern ausgeliefert (Bezeichnung: Calendar1-9). Die Kalender können über SysConfig-Schlüssel konfiguriert werden (Menü *System* > *SysConfig*). Information zur Konfiguration finden Sie im Admin-Handbuch von KIX 18 Start (s. Kalendereinstellungen).

Hinweis: Wird am SLA kein Kalender hinterlegt, werden die Default-Werte des Systems verwendet.

6.3 Reaktionszeit bei Ticketerstellung setzen

Szenario:

Ein Agent macht beim ankommenden Telefonanruf ein neues Ticket auf. Im Gesprächsverlauf konnte einiges geklärt werden, sodass der Reaktionszeit bereits genüge getan wurde. Diese wird jedoch noch nicht gesetzt, da der aktuelle Mechanismus zum Setzen der Reaktionszeit verlangt, dass der Artikel durch Sender Type "Agent" erstellt wird. Wird ein Ticket im Kanal "Notiz" erstellt, wird jedoch als Sender Type "external" erfasst.

Damit nun der Agent bereits in dem Ticket, welches er gerade anlegt, die Reaktionszeit setzen kann und kein neues "Telefonticket" erstellen muss, kann ein Dynamisches Feld vom Typ "Selection" in die Ticketvorlage integriert werden. Bei Auswahl der Option "ja" wird ein Job ausgelöst, der die Reaktionszeit am Ticket setzt.

Nachfolgend finden Sie eine Beschreibung zum Anlegen des Auswahlfeldes und Integrieren in die Ticketvorlage sowie zum Konfigurieren des auslösenden Jobs.

6.3.1 Vorgehen:

1. Dynamisches Feld anlegen
Das Dynamische Feld dient als "Schalter" zum Auslösen des Jobs (s. Schritt 4).
2. Übersetzung anlegen (optional)
Dient der automatischen Übersetzung des Dynamischen Feldes in der Vorlage.
3. Standard-Ticketvorlage erweitern
Integration des Dynamischen Feldes in der Ticketvorlage.
4. Job anlegen
Dient dem automatischen Setzen der Reaktionszeit am neuen Ticket.

6.3.2 Die Konfigurationen im Detail

6.3.2.1 1. Dynamisches Feld anlegen

Legen Sie im Admin Modul im Menü "System > Dynamische Felder" ein neues Dynamisches Feld vom Typ "Selection" an. Dieses dient als "Schalter" für das Setzen der Reaktionszeit am Ticket. Wird im Auswahlfeld die Option "ja" gewählt, wird der unter 4. konfigurierte Job ausgelöst und am Ticket automatisch die Reaktionszeit gesetzt. Wird die Option "nein" gewählt, wird der Job nicht ausgelöst und auch keine Reaktionszeit gesetzt.

Die Konfiguration des Dynamischen Feldes im Detail. Sie können den im Beispiel verwendeten Namen und das Label auch anders benennen.

- Name: FirstResponseByTicketCreation
- Label: Response by Ticket Creation
- Feld Typ: Selection

- Objekt Typ: Ticket
- Konfiguration:
 - Count Min: 1
 - Count Max: 1
 - Count Default: 1 (optional)
 - Vorgabewert: 0
 - Übersetzbare Werte: 
 - Mögliche Werte (Schlüssel/Wert):
 - 0 / no
 - 1 / yes

Weiterführende Informationen zu Dynamischen Feldern:

- Dynamische Felder anlegen, bearbeiten, einbinden
- Feldtypen Dynamischer Felder

6.3.2.2 2. Übersetzung anlegen

Damit die Beschriftung des Dynamischen Feldes in der Vorlage übersetzt wird, muss dafür ein neues Pattern angelegt werden. Dieser Schritt kann entfallen, wenn eine Übersetzung nicht erforderlich bzw. nicht gewünscht ist.

Zum Anlegen der Übersetzung navigieren Sie im Admin Modul ins Menü "*Internationalisierung > Übersetzungen*". Legen Sie dort ein neues Pattern mit folgenden Parametern an:

- Pattern: Response by Ticket Creation
- Englisch: Response by Ticket Creation
- Deutsch: Reaktion durch Ticketerstellung

Weiterführende Informationen zu Übersetzungen: [Übersetzungen](#) (siehe Seite 203)

6.3.2.3 3. Vorlage erweitern

Damit die Agenten beim Erstellen des Tickets die Reaktionszeit setzen lassen können, muss das unter Schritt 1 angelegte Dynamische Feld in die Vorlage integriert werden. Im Beispiel wird dafür die Standard-Ticketvorlage erweitert. Sie können das Dynamische Feld aber auch in eine andere Vorlage integrieren.

1. Navigieren Sie dazu im Admin Modul ins Menü "*Workflow > Vorlagen*" und öffnen Sie die Vorlage "Default - New Ticket Template" zur Bearbeitung.
2. Gehen Sie zu Schritt 2 "Eingabefelder".
3. Wählen Sie unten im letzten Auswahlfeld das unter Schritt 1 angelegte Dynamische Feld aus und wählen Sie die Option "im Formular setzen".
4. Optional können Sie das Dynamische Feld via Drag & Drop an gewünschter Stelle im Formular platzieren.
5. Speichern Sie die Vorlage abschließend. Danach ist das Dynamische Feld in die Vorlage integriert und kann bei jeder Ticketerstellung verwendet werden.

Weiterführende Informationen zu Ticketvorlagen: [Vorlagen](#) (siehe Seite 151)

6.3.2.4 4. Job anlegen

Das Setzen der Reaktionszeit am Ticket erfolgt automatisiert durch einen Job. Dieser wird ausgelöst, wenn der Agent bei Ticketerstellung im Dynamischen Feld "Response by Ticket Creation" die Option "yes | ja" gewählt hat. Als Erfüllungszeitpunkt wird der Zeitpunkt der Ticketerstellung gesetzt.

Legen Sie dazu im Admin Modul im Menü "*Automatisierung > Jobs*" einen neuen Job mit nachfolgend aufgeführter Konfiguration neu an. Sie können den initialen Job "SLA First Response Time Fulfillment" als Vorlage nutzen:

- Job Informationen
 - Job Typ: Ticket
 - Name: SLA First Response Time Fulfillment - by DF FirstResponseByTicketCreation
 - Kommentar: (optional)
 - Gültigkeit: gültig
- Ausführungsplan
 - Events: TicketCreate
- Filter
 - FirstResponseByTicketCreation - enthalten in - yes | ja
- Aktionen
 - Erfüllungszeit setzen
 - SLA Kriterium: FirstResponse
 - Erzwingen wenn nicht leer: 0

Speichern Sie abschließend den Job. Danach wird am Ticket die Reaktionszeit gesetzt, wenn vom Agent gewünscht.

7 Self Service Portal

⚠ Das SSP1 wird nicht mehr ausgeliefert

Das Self Service Portal 1 ist nicht mehr in KIX Pro enthalten. Wenn noch nicht erfolgt, muss mit Update auf v32 der Umstieg auf Self Service Portal 2 erfolgen.

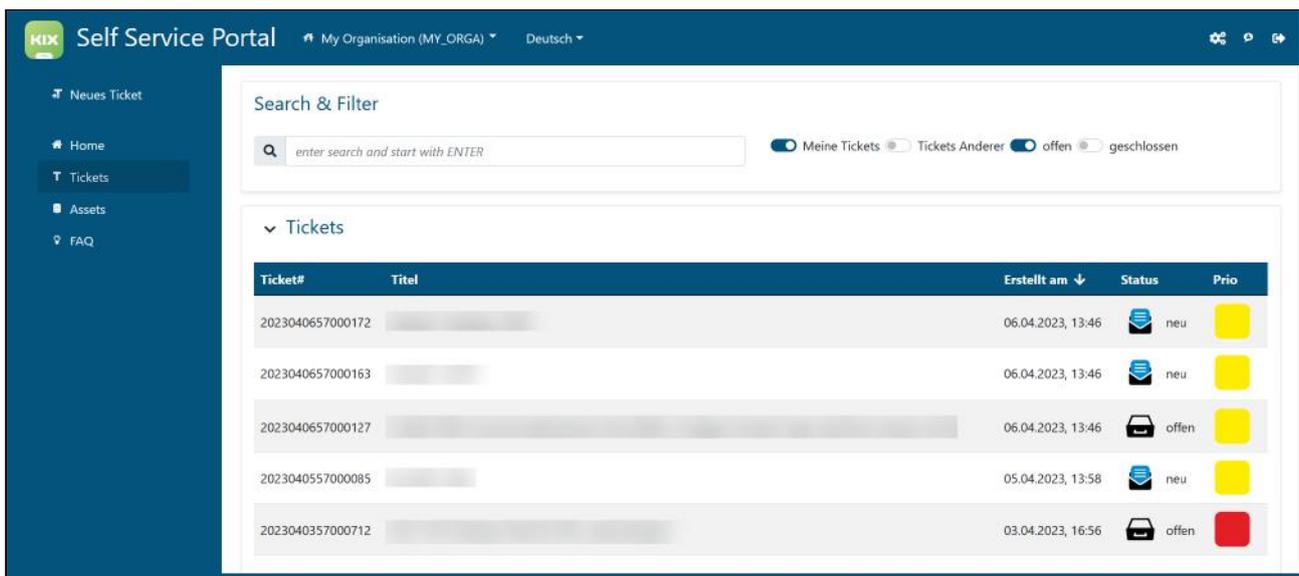
Die Verwendung der Basisberechtigungen ist somit obligatorisch!

Das Self Service Portal (SSP) ist das Kundenportal von KIX und wird mit KIX Pro ausgeliefert - sowohl für On-Premises Installationen als auch für KIX.Cloud.

Über das Self Service Portal können Ihre Kunden Tickets anlegen und bearbeiten und auch FAQ Beiträge und Assets (Betriebsmittel) einsehen.

Das Self Service Portal ist an KIX gekoppelt, sodass eine direkte Zusammenarbeit zwischen Helpdesk und Kunde möglich ist.

Die Administration und Konfiguration des Self Service Portals erfolgt im Admin Modul des Agentenportals (Menü: *KIX > System > GUI Konfiguration > Self Service Portal*).



The screenshot shows the 'Self Service Portal' interface. The top navigation bar includes the KIX logo, 'Self Service Portal', 'My Organisation (MY_ORGA)', and 'Deutsch'. A sidebar on the left contains navigation options: 'Neues Ticket', 'Home', 'Tickets', 'Assets', and 'FAQ'. The main content area features a 'Search & Filter' section with a search input field and toggle switches for 'Meine Tickets', 'Tickets Anderer', 'offen', and 'geschlossen'. Below this is a 'Tickets' section with a table listing ticket details.

Ticket#	Titel	Erstellt am ↓	Status	Prio
2023040657000172	[blurred]	06.04.2023, 13:46	neu	Yellow
2023040657000163	[blurred]	06.04.2023, 13:46	neu	Yellow
2023040657000127	[blurred]	06.04.2023, 13:46	offen	Yellow
2023040557000085	[blurred]	05.04.2023, 13:58	neu	Yellow
2023040357000712	[blurred]	03.04.2023, 16:56	offen	Red

Abb.: Die Ticketübersicht im Self Service Portal



7.1 Erreichbarkeit des Self Service Portals

KIX-Cloud:

Nutzer der KIX.Cloud erreichen das SSP durch Ergänzung der URL des Agentenportals um den Parameter "-ssp", z. B.:

Portal	Erreichbarkeit	Beispiel
Agentenportal	URL der KIX.Cloud Umgebung	https://your.agentenportal.kix.cloud
SSP2	URL der KIX.Cloud Umgebung mit zusätzlicher Angabe von "-ssp"	https:// your.agentenportal-ssp.kix.cloud

On Premises Installationen:

Bei On Premises Installationen ist das SSP über den Aufruf der KIX-URL mit angefügter Portnummer erreichbar:

Portal	Erreichbarkeit	Beispiel
Agentenportal	URL der KIX-Installation mit zusätzlicher Angabe der Portnummer "20001"	https://your.kix.docker-host.org:20001
SSP	URL der KIX-Installation mit zusätzlicher Angabe der Portnummer "20002"	https://your.kix.docker-host.org:20002

Bei Bedarf können Sie die Standard-Ports in der Datei *environment* ändern. Geben Sie die Port-Nummern an, unter denen Sie sich mit KIX verbinden.

Portnummern für das Self Service Portal angeben	
1	[...]
2	# -----
3	# frontend service configuration
4	# -----
5	
6	# the port on the docker host system where the frontend service is listening
7	FRONTEND_PORT=20001
8	FRONTEND_PORT_SSL=20444
9	
10	[...]



⚠ Hinweis

Der Script-Alias mit angefügtem "/ssp" (z. B.: <https://your.kix-docker-host.org:20001/ssp>) wird seit dem 03.11.2023 nicht mehr verwendet!

Weiterführende Informationen finden Sie im [GitHub](#)⁴:

- Linux: <https://github.com/kix-service-software/kix-on-premise/blob/master/deploy/linux/README.md>
- Windows: <https://github.com/kix-service-software/kix-on-premise/blob/master/deploy/windows/README.md>

⁴ <https://github.com/cape-it>

7.2 Nutzereinstellungen für das SSP

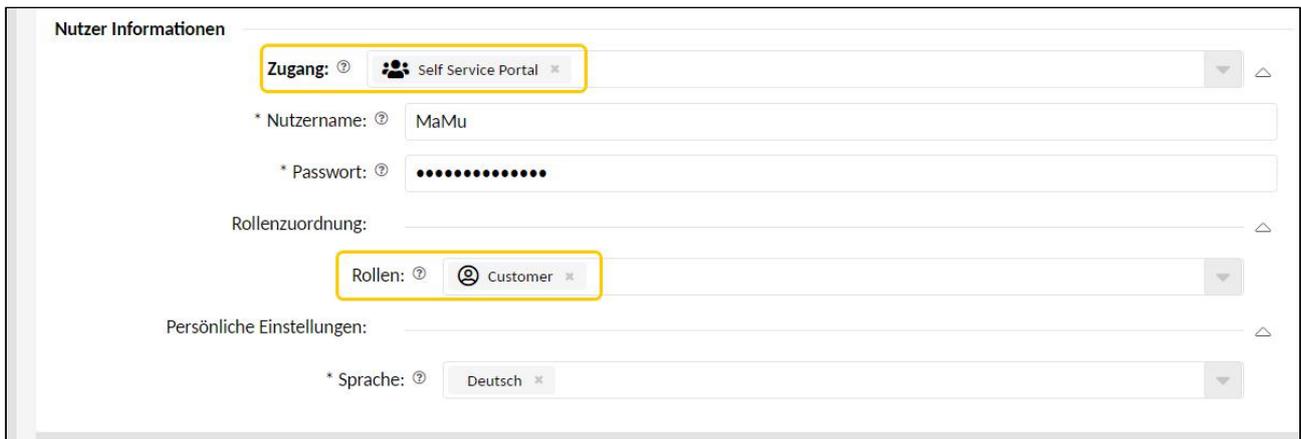
7.2.1 Nutzer-Login einrichten

Voraussetzung für den Zugang zum Self Service Portal ist, dass der Kundenkontakt im Agentenportal als Nutzer mit entsprechender Zugangsberechtigung angelegt ist. Er benötigt mindestens die Rolle "Customer".

Öffnen Sie dazu im *Modul Organisationen* einen bestehenden Kontakt oder legen Sie einen neuen Kontakt an. Für Kundenkontakte, welche bereits Zugang zum Agentenportal besitzen (Nutzer), kann die Einrichtung auch im *Modul Admin > KIX > Nutzerverwaltung > Nutzer* erfolgen.

Wählen Sie unter "Nutzer Informationen" (zusätzlich) das "Self Service Portal" aus und erteilen Sie mindestens die Rolle "Customer". Folgende Formularfelder sind für das Self Service Portal relevant:

Formularfeld	Erklärung
Zugang	Wählen Sie im Dropdown "Self Service Portal" und klicken Sie "Übernehmen". Mit Ihrer Auswahl legen Sie fest, auf welche Portale der Kontakt Zugriff hat.
Nutzername	Vergeben Sie einen eindeutigen Nutzernamen für den Kontakt, mit dem sich dieser im Self Service Portal anmelden kann.
Passwort	Vergeben Sie ein initiales Passwort für den Kontakt, mit dem sich dieser im Self Service Portal anmelden kann. Der Kontakt kann dies im Self Service Portal selbst ändern.
Rollen	Wählen Sie mindestens die Rolle "Customer" aus, damit der Kunde Zugang zum Self Service Portal erhält.
Sprache	Legen Sie die Sprache fest, in der das Self Service Portal für den Kunden angezeigt wird.



Nutzer Informationen

Zugang: Self Service Portal

* Nutzername:

* Passwort:

Rollenzuordnung:

Rollen: Customer

Persönliche Einstellungen:

* Sprache: Deutsch

Abb.: Zugang für das SSP einrichten

Info

Besitzt ein Kundenkontakt Zugang zum Self Service Portal, wird er sowohl als Kontakt (Modul *Organisationen*) als auch als Nutzer (Modul *Admin > Nutzerverwaltung > Nutzer*) angezeigt. Er kann über beide Ansichten aufgerufen und bearbeitet werden.

7.2.2 Rollen und Berechtigungen

Damit ein Kundennutzer Zugang zum Self Service Portal hat, benötigt er mindestens die Rolle "Customer". Dieser Rolle können weitere hinzugefügt werden, um die Berechtigungen des Kundennutzers zu erweitern, z. B. "Customer Manager", "Customer Reader" oder andere (s. auch [Übersicht der vorkonfigurierten Rollen in KIX](#) (siehe Seite 209)).

Jeder Kundennutzer sieht nur die Objekte (Tickets, Assets, FAQ), auf die er mindestens Lese-Berechtigung hat.

Die Anzeige der im Self Service Portal bereitgestellten Informationen und Dynamischen Felder sowie die Auswahlmöglichkeiten in Select-Feldern ist von den jeweiligen Nutzerrechten abhängig.

7.3 Sichtbarkeiten im SSP steuern

Sie können festlegen, welche Informationen ein Kundenkontakt im Self Service Portal sehen kann:

- alle relevanten Tickets vollständig
- nur ausgewählte Artikel eines Tickets
- FAQ Artikel
- vollständige Assets
- ausgewählte Asset-Attribute
- ausgewählte Dynamische Felder
- News

Inhalte auf dieser Seite:

- [Sichtbarkeit von Tickets und Artikeln](#) (siehe Seite 215)
- [Sichtbarkeit von Follow-ups](#) (siehe Seite 216)
- [Sichtbarkeit von Assets](#) (siehe Seite 217)
- [Sichtbarkeit von FAQ-Artikeln](#) (siehe Seite 218)
- [Sichtbarkeit von Dynamischen Feldern](#) (siehe Seite 218)
- [Sichtbarkeit von News](#) (siehe Seite 220)

7.3.1 Basis-Einstellungen für Sichtbarkeiten

Konfigurationsschlüssel	Beschreibung	Beispiel
<p><code>AssignedObjectsMapping</code></p>	<p>Steuert die Zuordnung von Objekten zu Kontakten und/oder Organisationen sowie die Sichtbarkeit von Objekten im SSP.</p> <p>Definiert, anhand welcher Attribute ein abhängiges Objekt (z. B. Ticket) einem relevanten Objekt (z. B. Kontakt) zugeordnet wird (z. B. Tickets aus Kontakt).</p> <p>Im nebenstehenden Beispiel sehen Kontakte nur Tickets der eigenen Organisation und nur Artikel sowie FAQ Artikel, welche mit "Im Kundenportal anzeigen" markiert sind.</p> <p>Struktur:</p> <pre data-bbox="639 1176 1093 1774"> { "RelevantObject": { "DependendObject": { "DependendObjectAttribut": { { "SearchAttribut OR SearchStatic": ["RelevantObjectAttribut OR StaticValue"] } } } } } </pre> <p>Alle Attribute werden mit OR kombiniert.</p>	<pre data-bbox="1125 495 1423 1682"> { "Contact": { "Ticket": { "OrganisationID": { "SearchAttributes": ["RelevantOrganisationID"] } } }, "TicketArticle": { "CustomerVisible": { "SearchStatic": [1] } }, "FAQArticle": { "CustomerVisible": { "SearchStatic": [1] } } } </pre>

Konfigurationsschlüssel	Beschreibung	Beispiel
	<p>Hinweis: Werden E-Mails von bisher unbekanntem Kontakten abgerufen, kann KIX einen neuen Kontakt anlegen und anhand der Mail-Domain die Organisation zuordnen (s. auch Automatische Zuordnung der Organisation zu einem Kontakt). Die Zuordnungsmethode kann im SysConfig-Schlüssel <i>Contact::EventModulePost###800-AutoAssignOrganisation</i> definiert werden. Ist als Zuordnungsmethode "DefaultOrganisation" aktiviert, sollte die Einsichtnahme in die Tickets anderer aus Gründen des Datenschutzes unterbunden werden:</p> <pre data-bbox="639 1037 1093 1413"> { "Contact": { "Ticket": { "ContactID": { "SearchAttributes": ["ID"] } }, [...] } } </pre>	

Konfigurationsschlüssel	Beschreibung	Beispiel
<p><code>AssignedConfigItemsMapping</code></p>	<p>Steuert die Zuordnung von Assets zu Kontakten und/oder Organisationen sowie die Sichtbarkeit von Assets im SSP. Die Zuordnungen können auf Basis der Verwendungsstatus erfolgen.</p> <p>In Kombination mit spezifischen Verwendungsstatus, z.B. "Baseline Configuration" wird die Anzeige von Baseline-Assets oder "Shopping Item"-Versionen von Assets im Self Service Portal möglich.</p> <p>Durch Angabe von Suchkriterien können Sie die Sichtbarkeit auf post- oder pre-produktive Status beschränken. Somit können Sie bspw. ein statisches Suchkriterium auf Basis des Deployment-Status definieren, damit SSP Nutzer nicht selbst zwischen aktuellen Daten und Alt-Daten unterscheiden müssen (für Warenkörbe oder Baseline-Konfigurationen).</p> <p>Möglich sind folgende Attribute:</p> <ul style="list-style-type: none"> • DeploymentState - Erwartet eine Liste der Status-Namen • IncidentState - Erwartet eine Liste der Status-Namen • Name - Erwartet einen String; bei einer Liste wird nur der erste Wert betrachtet • Number - Erwartet einen String; bei einer Liste wird nur der erste Wert betrachtet <p>Bei allen 4 Attributen ist als Suchkriterium möglich:</p>	<pre>{ "Contact": { "Computer": { "SectionOwner::OwnerContact": { "SearchAttributes": ["ID"] }, "DeploymentState": { "SearchStatic": ["Production", "Planned"] }, "SectionOwner::OwnerOrganisation": { "SearchAttributes": ["RelevantOrganisationID"] } } }, ... }</pre>

Konfigurationsschlüssel	Beschreibung	Beispiel
	<ul style="list-style-type: none"> • SearchStatic - Statisches Suchkriterium • SearchAttributes - Sucht nach den angegebenen Attributen (bspw. Kontakt-Attribute) 	

7.3.2 Sichtbarkeit von Tickets und Artikeln

Grundlegend werden alle Tickets des angemeldeten Kundenkontaktes im Self Service Portal angezeigt. Alle Artikel, die der Kontakt selbst anlegt, werden automatisch immer im Self Service Portal angezeigt. Für Tickets und Artikel, die im KIX Agentenportal angelegt werden, kann der Agent entscheiden, ob diese im Self Service Portal sichtbar sind. Die Steuerung erfolgt durch Aktivierung/Deaktivierung der Option "Im Kundenportal anzeigen".

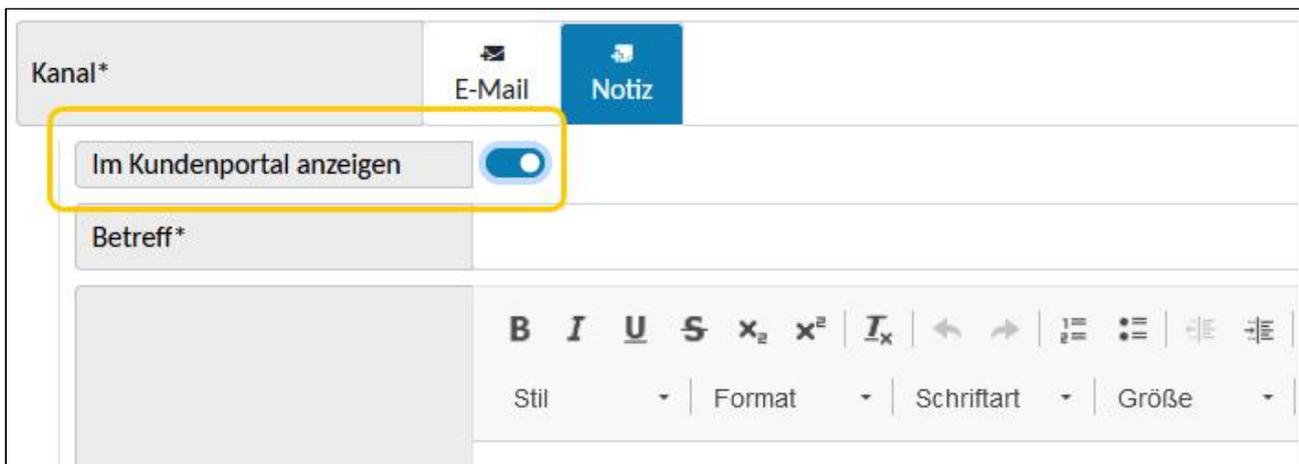


Abb.: Das Ticket wird im Self Service Portal angezeigt

Hinweise

- Wird im Kanal "E-Mail" ein Artikel erstellt, in dem der Kundenkontakt der Empfänger ist, kann das Häkchen "Im Kundenportal anzeigen" nicht entfernt werden.
- Enthält ein Ticket ein dynamisches Feld vom Typ "Checklist" kann dieses im Self Service Portal angezeigt, jedoch nicht bearbeitet werden.

7.3.3 Sichtbarkeit von Follow-ups

Follow-ups sind eingehende Rückantworten auf E-Mails bzw. eingehende Folgemeldungen zu bestehenden Tickets.

Initial sind eingehende E-Mails im SSP nur sichtbar, wenn:

- der Ticketkontakt in der Empfängerliste (To, Cc, Bcc, usw.) enthalten ist.
- der Ticketkontakt der Absender ist (nicht konfigurierbar) .
- eine eingehende Nachricht eine Referenzierte-Antwort auf einen für den Kundennutzer sichtbaren Artikel ist
- der Konfigurationsschlüssel PostMaster::FollowUp::CheckFromOrganisation aktiviert ist und Absender zur Ticket-Organisation gehört
- ein [X-KIX-FollowUp-Header](#) (siehe Seite 211) auf "CustomerVisible" gesetzt ist.

Sie können diese Verhalten nach Bedarf de-/aktivieren, indem Sie die nachfolgenden Konfigurationsschlüssel "gültig" bzw. "(temporär) ungültig" setzen. Initial sind sie aktiv.

Konfigurationsoptionen

Konfigurationsschlüssel	Beschreibung
<p><code>Ticket::EventModulePost###900-ArticleCustomerVisibleByRecipient</code></p>	<p>Steuert die Sichtbarkeit des Artikels im Self Service Portal anhand des E-Mail-Empfängers.</p> <p>Prüft bei Anlegen eines Artikels im Kanal "E-Mail" (z. B. nach Eingang einer E-Mail oder Versenden einer E-Mail aus KIX), ob mindestens einer der E-Mail-Empfänger (To, Cc oder Bcc) dem Ticketkontakt zugeordnet ist.</p> <ul style="list-style-type: none"> • Wenn ja: Das System setzt die Sichtbarkeit des Artikels im SSP automatisch. Und zwar unabhängig davon, ob die Option "Im Kundenportal anzeigen" vom Agent gesetzt wurde oder nicht. • Wenn nein: Die vom Agent gewählte Option für "Im Kundenportal anzeigen" wird nicht verändert. <p>Bei Antwort auf einen Artikel (Artikelaktion "Antworten") wird die Sichtbarkeit im SSP mit dem Wert des Quellartikels initialisiert.</p>

Konfigurationsschlüssel	Beschreibung
<code>Ticket::EventModulePost###910- ArticleCustomerVisibleByReferences</code>	<p>Artikel-Event, welches die Sichtbarkeit im SSP anhand der E-Mail-Referenz steuert.</p> <p>Wird ausgeführt, wenn ein Artikel im Kanal "E-Mail" erstellt wird. Prüft, ob mindestens eine E-Mail-Referenz (References und InReplyTo) die Message-ID eines für den Kunden sichtbaren Artikels im Ticket ist.</p> <ul style="list-style-type: none"> • Wenn Ja: Die Sichtbarkeit des Artikels in der SSP wird automatisch aktiviert (unabhängig von der Einstellung des Agenten). • Wenn Nein: Die vom Bearbeiter gewählte Sichtbarkeit wird nicht geändert.
<code>PostMaster::FollowUp:: CheckFromOrganisation</code>	<p>Vormals: <i>TicketStateWorkflow::PostmasterFollowUpCheckCustomerIDFrom</i></p> <p>Überprüft, ob der Absender einer Folgemeldung/ Rückantwort (Follow-up) in der Kontaktdatenbank enthalten ist und der gleichen Organisation wie das Ticket zugeordnet ist. Ist dies der Fall, wird die E-Mail im Self Service Portal angezeigt.</p> <p>Es wird nur am Ticket des Follow-up auf eine passende Message-ID geprüft. Die Sichtbarkeitsprüfung anhand der Referenz erfolgt durch das Event <i>Ticket::EventModulePost###910- ArticleCustomerVisibleByReferences</i> (s. oben).</p>

7.3.4 Sichtbarkeit von Assets

Grundlegend werden einem Kundenkontakt die Assets angezeigt, denen er als Kontakt zugeordnet ist.

Um die Sichtbarkeit von Asset-Attributen im Self Service Portal zu steuern, steht für klassenspezifische Asset-Attribute in der Klassendefinition die Eigenschaft "CustomerVisible" zur Verfügung.

Standardmäßig werden im Self Service Portal die klassenübergreifenden Attribute angezeigt. Für die klassenspezifischen Attribute ist "Customer Visible = 0", d.h. sie werden nicht angezeigt. Setzen Sie für die Eigenschaften (Formularfelder), die Sie im Self Service Portal anzeigen wollen, das Attribut "CustomerVisible = 1".



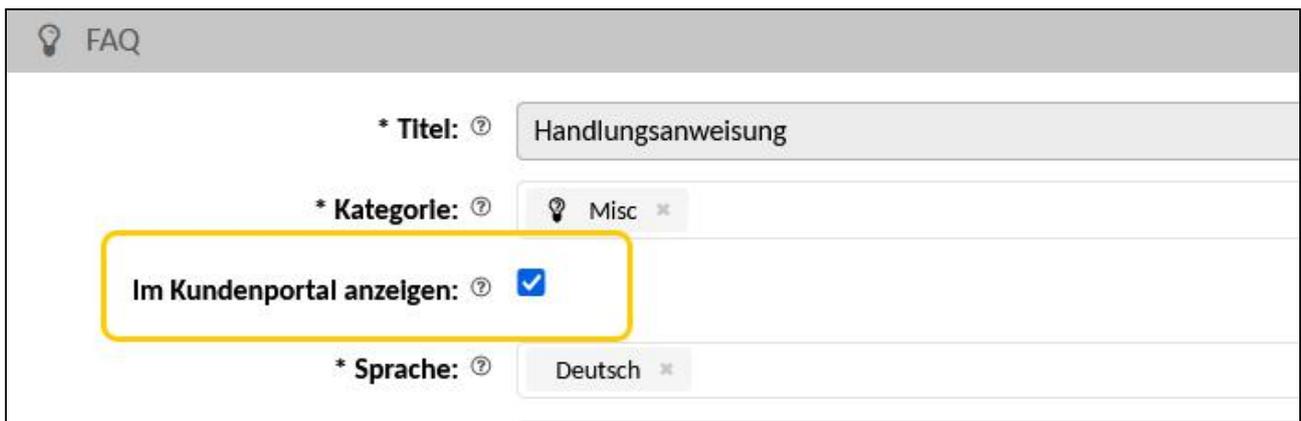
```

* Name: Computer
Icon: [Bilddatei wählen]
* Klassendefinition: [
  {
    Key => 'SectionGeneral',
    Name => 'General Information',
    CustomerVisible => 0,
    Input => {
      Type => 'Dummy',
    },
  },
]
Kommentar:
  
```

Abb.: Mit dem Attribut "Customer Visible" wird die Sichtbarkeit von Assets im SSP gesteuert

7.3.5 Sichtbarkeit von FAQ-Artikeln

Durch Aktivieren/Deaktivieren der Option "Im Kundenportal anzeigen" kann ein Agent beim Anlegen oder Bearbeiten eines FAQ-Artikels entscheiden, ob der FAQ-Artikel im Self Service Portal angezeigt wird. Ist das Häkchen gesetzt, ist der FAQ-Artikel für alle Nutzer des Self Service Portals sichtbar.



FAQ

* Titel: Handlungsanweisung

* Kategorie: Misc

Im Kundenportal anzeigen:

* Sprache: Deutsch

Abb.: Sichtbarkeit im Self Service Portal bei FAQ-Artikeln einstellen

7.3.6 Sichtbarkeit von Dynamischen Feldern

Das Self Service Portal unterstützt die Bereitstellung Dynamischer Felder. Unterstützt werden alle Feldtypen, die auch vom Agentenportal unterstützt werden.

Damit ein Dynamisches Feld im Self Service Portal angezeigt wird, muss in seiner Konfiguration die Option "Im Kundenportal anzeigen" aktiviert sein.

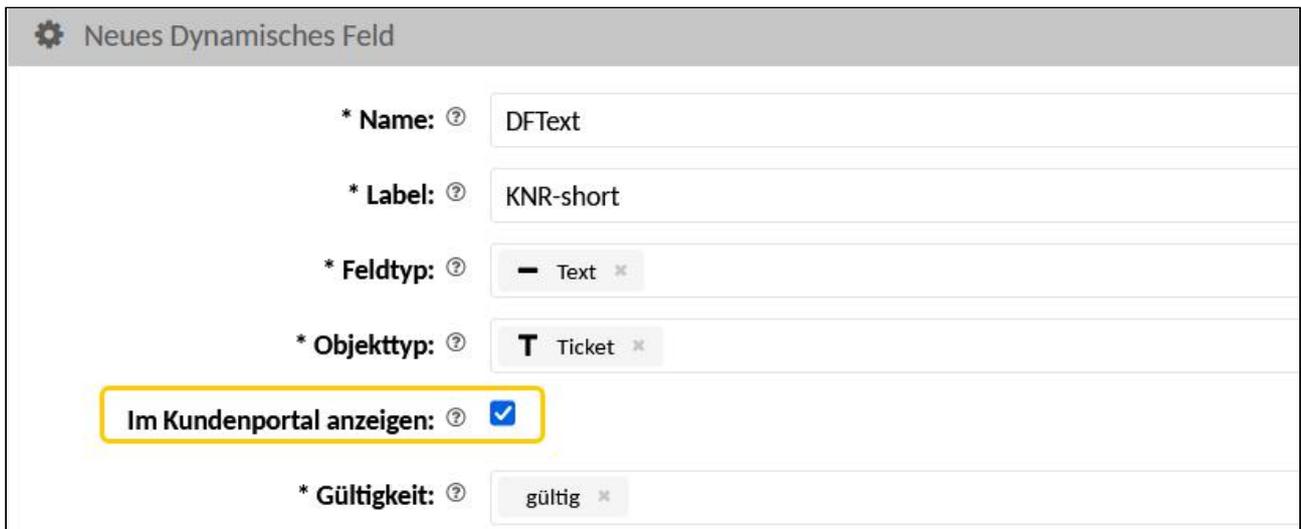


Abb.: Dynamisches Feld im SSP bereitstellen

Mit dem Aktivieren oder Deaktivieren dieser Option legen Sie fest, ob die im Agentenportal bereitgestellten Dynamischen Felder auch im Self Service Portal ersichtlich sind (z. B. in Ticketvorlagen oder Aktionen).

Referenzierende Dynamische Felder (z. B. Typ Ticketreferenz) in Ticketvorlagen oder Aktionen enthalten nur die Objekte zur Auswahl, auf die der angemeldete SSP-Nutzer Rechte hat.

Im Self Service Portal bereitgestellte Dynamische Felder werden oberhalb der Artikelübersicht angezeigt (einklappbarer Bereich "Weitere Informationen"). Dynamische Felder ohne Wert werden nicht angezeigt.

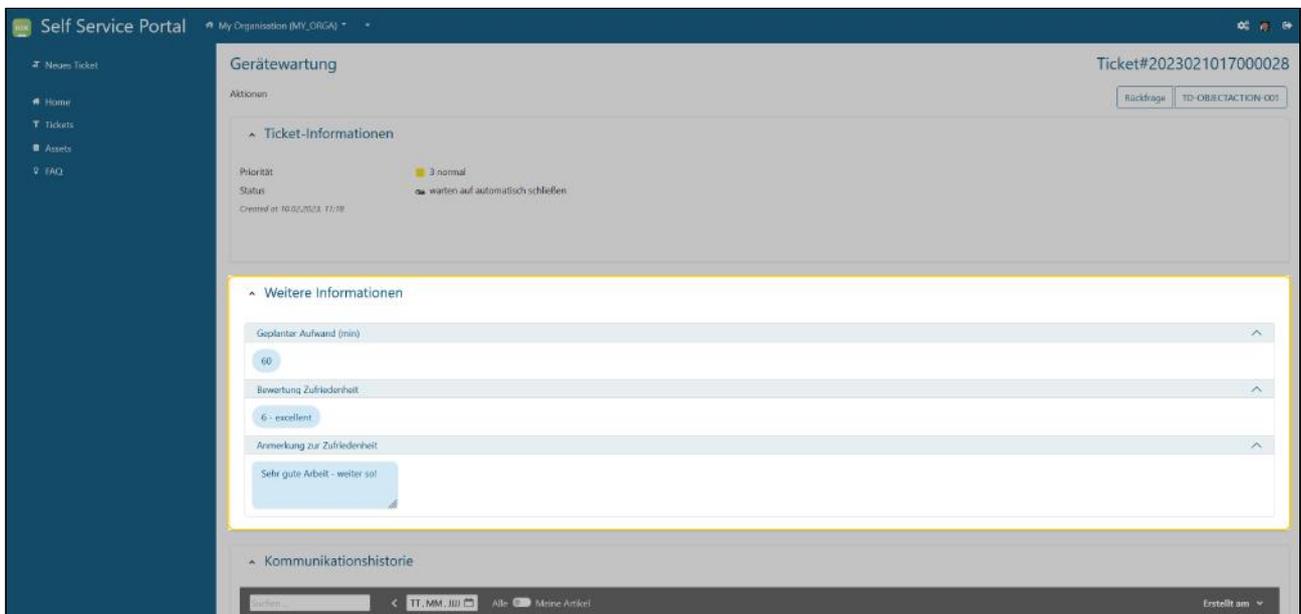


Abb.: im Ticket verwendete Dynamische Felder werden in einem separaten Bereich angezeigt

7.3.7 Sichtbarkeit von News

Sie können den Nutzern des Self Service Portals News und Informationen bereitstellen, um sie bspw. über geplante Systemwartungen zu informieren.

News werden im Modul "News"  angelegt und verwaltet. Damit News im Self Service Portal angezeigt werden, muss ihnen der Nutzungskontext "Kunde" zugewiesen sein. Zudem können Sie festlegen, ob die Anzeige vor und/oder nach dem Login erfolgen soll. Weiterführende Informationen zu News finden Sie im Admin-Pro Handbuch im Abschnitt [News \(siehe Seite 249\)](#) .

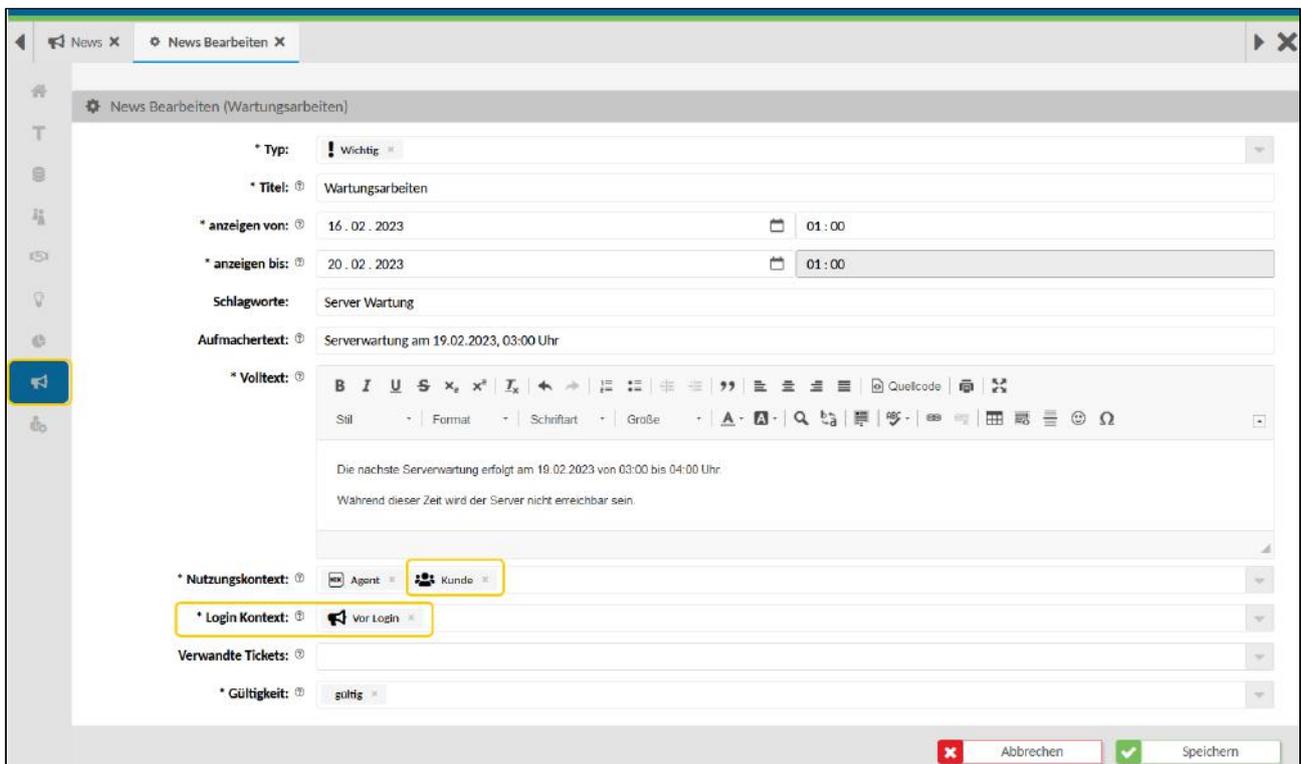


Abb.: News im SSP anzeigen lassen

7.4 Ticketeinstellungen

Inhalte auf dieser Seite:

- [Fallback für Ticket-Titel](#) (siehe Seite 221)
- [Bearbeitung von Ticket-Kopfattributen ermöglichen \(Berechtigung\)](#) (siehe Seite 221)
- [Voreinstellungen für Ticketfilter](#) (siehe Seite 223)

7.4.1 Fallback für Ticket-Titel

Konfigurationsschlüssel	ssp-ticket-title-fallback-pattern
--------------------------------	-----------------------------------

Damit KIX Tickets im System speichern kann, benötigt jedes Ticket einen Titel bzw. einen Betreff. In KIX können jedoch Ticketvorlagen erstellt werden, in denen der Nutzer keinen Tickettitel oder Betreff angeben muss. In diesen Fällen verwendet KIX initial das aktuelle Datum und die Uhrzeit als Fallback.

Soll anstelle dieser Datum-Zeitangabe ein anderer Wert als Fallback dienen, können Sie diesen Wert im SysConfig-Schlüssel "ssp-ticket-title-fallback-pattern" hinterlegen.

7.4.2 Bearbeitung von Ticket-Kopfattributen ermöglichen (Berechtigung)

Inhaber der Rolle "Customer" haben initial nur Lese-Rechte auf die Ticketkopfattribute (Priorität, Status, Titel etc). Um die Bearbeitung der Kopfattribute zu ermöglichen, können Sie der Rolle "Customer" Update-Rechte auf die Kopfattribute einräumen. Klicken Sie dazu auf "Rolle bearbeiten" und setzen Sie an der Property `"/tickets/*{Ticket.`

`[Articles,Changed,ContactID,Created,CreateTimeUnix,DynamicFields,OrganisationID,PriorityID,QueueID,StateID,TypeID,TicketNumber,Title}]` ein Häkchen bei "U - Update".

Rollen-Informationen						
Name:	Customer	Gültigkeit:	gültig	Geändert am:	13.03.2023, 12:11	
Nutzungskontext:	Kunde	Erstellt am:	13.03.2023, 12:11	Geändert von:	not assigned	
Kommentar:	allows the customer to login and read ticket, FAQ and asset data, allows to create FAQ votes, allows to change data of the personal user, allows to set ticket article flags					
Erstellt von:						
not assigned						
Berechtigungen (70)						
Layer	Ziel	Create	Read	Update	Delete	Deny
Object	/faq/articles/*[FAQArticle.CustomerVisible NE 1]	C	R	U	D	DN
Object	/organisations/*[Organisation.ID !IN \$CurrentUser.Contact.OrganisationIDs]	C	R	U	D	DN
Object	/system/dynamicfields/*[DynamicField.CustomerVisible NE 1]	C	R	U	D	DN
Object	/system/objectactions/*[ObjectAction.UsageContext EQ 1]	C	R	U	D	DN
Object	/system/users/*[User.UserID NE \$CurrentUser.UserID]	C	R	U	D	DN
Object	/tickets/*[Ticket.OrganisationID !IN \$CurrentUser.Contact.OrganisationIDs]	C	R	U	D	DN
Object	/tickets/*articles/*[Article.CustomerVisible NE 1]	C	R	U	D	DN
Object	/system/config[SysConfigOption.AccessLevel NE external]	C	R	U	D	DN
Object	/system/config/*[SysConfigOption.AccessLevel NE external]	C	R	U	D	DN
Object	/system/ticket/templates/*[TicketTemplate.CustomerVisible NE 1]	C	R	U	D	DN
Object	/system/templates/*[Template.CustomerVisible NE 1]	C	R	U	D	DN
Property	/system/cmdb/classes/*[ConfigItemClass.{ID,Name}]	C	R	U	D	DN
Property	/tickets/*[Ticket.{Age,Articles.Changed,ContactID,Created,CreateTimeUnix,DynamicFields,OrganisationID,PriorityID,QueueID,StateID,TypeID,TicketNumber,Title}]	C	R	U	D	DN
Property	/tickets/*articles/*[Article.{*,!Bcc,!BccRealname,!TimeUnit,!To,!ToRealname,!Cc,!CcRealname,!From,!FromRealname}]	C	R	U	D	DN
Property	/tickets/*articles/Article.{*,!TimeUnit}	C	R	U	D	DN
Property	/tickets/*Ticket.{ObjectActions}	C	R	U	D	DN

Abb.: Erweiterte Rollenberechtigung auf Update für Customer

Sie können die angegebenen Property's bearbeiten, um ausgewählte Kopfattribute von der Bearbeitung aus- oder einzuschließen. Die angegebenen Property's definieren, auf welche Objekte der "Customer" die gesetzten Berechtigungen hat.

Rolle bearbeiten (Customer)	
Rollen-Informationen * Name: Customer * Nutzungskontext: Kunde Kommentar: allows the customer to login and read ticket, FAQ and asset data, allows to create FAQ votes, allows to change data of the personal user, allows to set ticket article flags * Gültigkeit: gültig	
Berechtigungen	
Resource	/auth <input checked="" type="checkbox"/> <input type="checkbox"/>
Resource	/cmdb <input type="checkbox"/> <input checked="" type="checkbox"/>
Resource	/tickets <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
Resource	/tickets/* <input type="checkbox"/> <input checked="" type="checkbox"/>
Object	/tickets/*[Ticket.OrganisationID !IN \$CurrentUser.Contact.OrganisationIDs] <input type="checkbox"/> <input checked="" type="checkbox"/>
Property	/tickets/*[Ticket.{Age,Articles.Changed,ContactID,Created,CreateTimeUnix,DynamicFields,OrganisationID,PriorityID,QueueID,StateID,TypeID,TicketNumber,Title}] <input type="checkbox"/> <input checked="" type="checkbox"/>
Resource	/tickets/*articles <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
Resource	/tickets/*articles/* <input type="checkbox"/> <input checked="" type="checkbox"/>
Object	/tickets/*articles/*[Article.CustomerVisible NE 1] <input type="checkbox"/> <input checked="" type="checkbox"/>

Abb.: Ticketattribute, auf die der "Customer" Berechtigungen hat.

Bei Bedarf können Sie der Rolle weitere Property's hinzufügen, um die Berechtigungen stärker zu granulieren und auch weitere Rollen anlegen (s. auch Eine Rolle konfigurieren und zuweisen).

Beispiel: Gesplittete Rechte auf die Ticket-Kopfattribute. Es dürfen alle angegebenen Ticketkopfattribute gelesen werden (markierte Zeile unten). Aber nur die Organisation und Dynamische Felder dürfen bearbeitet werden (markierte Zeile oben).

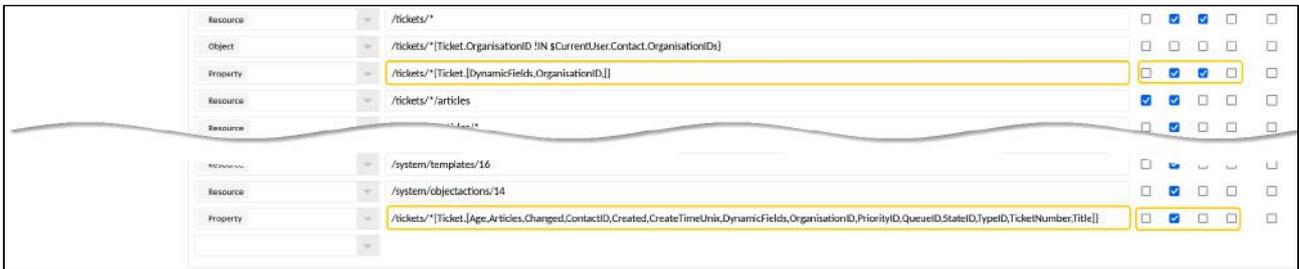


Abb.: Gesplittete Berechtigung für Ticket-Kopfattribute

7.4.3 Voreinstellungen für Ticketfilter

Konfigurationsschlüssel	ssp-ticket-context
--------------------------------	--------------------

Sie können die Standard-Einstellungen der Ticketfilter im Self Service Portal zentral konfigurieren. Sie legen damit fest, welche Filter nach Öffnen des Self Service Portals für alle Kundennutzer initial aktiv bzw. inaktiv sind.

Der Kundennutzer kann die Filter trotzdem entsprechend seines Bedarfs setzen. Seine Einstellungen bleiben bis zum erneuten Login oder Browser-Refresh erhalten.

Um die Ticketfilter zu konfigurieren, gehen Sie wie folgt vor:

- Navigieren Sie im Admin Modul des Agentenportals zu: "System > GUI Konfiguration > Self Service Portal"
- Wählen Sie im Explorer den Eintrag "SSP Ticket Context"
- Klappen Sie im Code ggf. den Parameter "content" zusammen, um eine bessere Übersicht im Code zu erhalten.
- Aktivieren/Deaktivieren Sie in den "filterSettings" die einzelnen Ticketfilter. Setzen Sie dazu die Ticketfilter true (aktiv) oder false (inaktiv).
- Speichern Sie die Änderung.
- Klicken Sie auf "Lade Frontend-Konfigurationen neu".

Danach sind die Ticketfilter im SSP entsprechend Ihrer Konfiguration initial aktiviert bzw. deaktiviert. Gegebenenfalls benötigt das Browserfenster für das Self Service Portal einen Refresh (F5 bzw. STRG+F5), um die Anzeige im Self Service Portal zu aktualisieren.

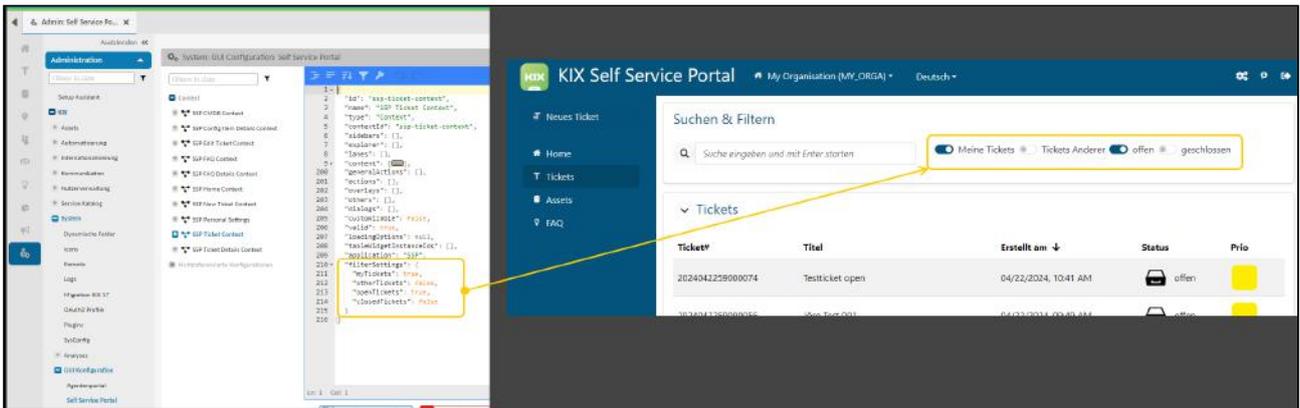
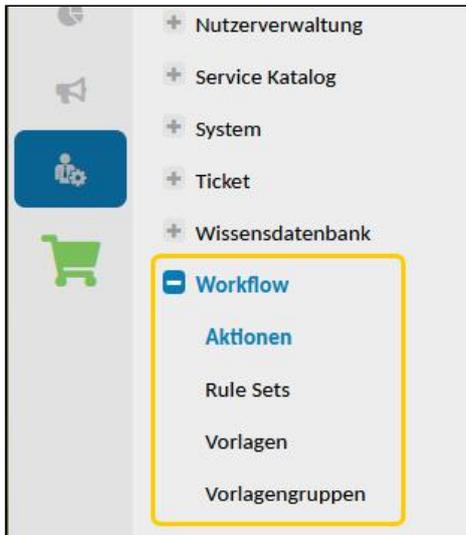


Abb.: Ticketfilter für das Self Service Portal initial de-/aktivieren

7.5 Vorlagen, Aktionen, Rule Sets

Ticketvorlagen, Aktionen und Workflow Rulesets können auch für das Self Service Portal konfiguriert werden. Dazu stehen die Bordmittel des Agentenportals zur Verfügung.



Inhalte auf dieser Seite:

- [Bereitstellung von Ticketaktionen](#) (siehe Seite 225)
- [Bereitstellung von Artikelaktionen](#) (siehe Seite 227)
- [Bereitstellung von Vorlagen](#) (siehe Seite 227)
- [Workflow Rulesets](#) (siehe Seite 228)

Abb.: Die Konfiguration von Vorlagen, Aktonen, Rulesets erfolgt über das Admin Modul des Agentenportals

7.5.1 Bereitstellung von Ticketaktionen

Sie können Ticketaktionen für das Self Service Portal (SSP) freigeben. Die Konfiguration erfolgt analog der [Ticketaktionen](#) (siehe Seite 93) für das Agentenportal. Die für das Self Service Portal freigegebenen Ticketaktionen werden als Schaltflächen im Ticketheader bereitgestellt.

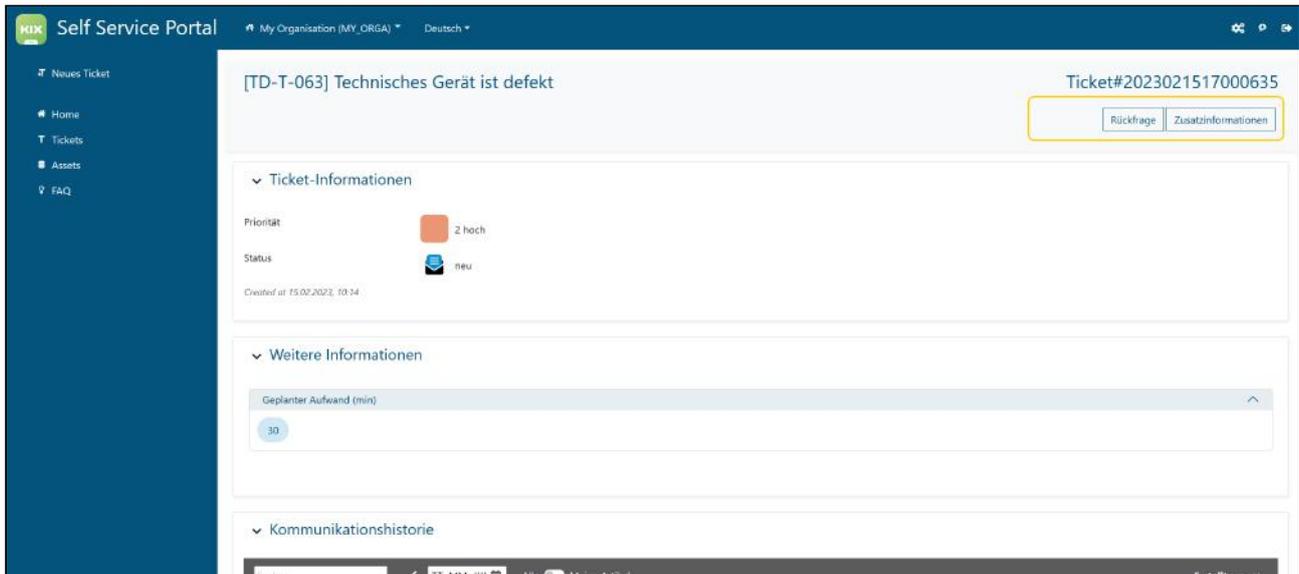


Abb.: Im SSP bereitgestellte Ticketaktionen

Voraussetzungen für die Verwendung einer Ticketaktion im SSP:

- Damit eine Ticketaktion im Self Service Portal zur Verfügung steht, muss diese mindestens dem Nutzungskontext "Kunde" zugeordnet sein.
Ist die Aktion dem Nutzungskontext "Kunde" und "Agent" zugeordnet, steht die Ticketaktion in beiden Portalen zur Verfügung.
- Der Aktion muss im Filter mindestens die Rolle "Customer" zugewiesen sein.
- Die Aktion muss "gültig" sein.
- Für die verwendeten Dynamischen Felder muss in deren Konfiguration die Option "Im Kundenportal anzeigen" aktiviert sein. (s. auch [Sichtbarkeiten im SSP steuern \(siehe Seite 211\)](#))
Ist diese Option nicht aktiviert, so wird das Dynamische Feld im SSP nicht angezeigt, auch wenn es in der Konfiguration der Aktion enthalten ist. Dies ermöglicht eine differenzierte Bereitstellung dynamischer Felder in Ticketaktionen, die gemeinsam von Agentenportal und SSP genutzt werden.

Hinweise für Ticketaktionen im SSP:

- Ticketaktionen für das SSP sind initial auf das Setzen und Bearbeiten Dynamischer Felder sowie auf das Anlegen von Artikeln beschränkt. Die Kopfattribute von Tickets (Status, Typ etc.) können nur bearbeitet werden, wenn die Rolle "Customer" um Update-Rechte erweitert wird (Auswirkungen bedenken!), s. auch [Nutzereinstellungen für das SSP \(siehe Seite 209\)](#) .
- Werden Felder in eine Ticketaktion konfiguriert, auf die der SSP-Nutzer keine Berechtigung hat, so schlägt bei ihm deren Verwendung fehl.
- Es können keine E-Mails aus dem SSP versendet werden, nur Notizen. Daher ist der "Kanal | Chanel" immer auf Notiz gesetzt, auch wenn in der Konfiguration "E-Mail" ausgewählt wurde.
- Pre- und Post-Actions werden vom SSP nicht unterstützt.
- Eine Ticketaktion kann erneut aufgerufen werden, um bspw. Eingaben zu korrigieren (z. B. die Bewertung von 3 auf 4 ändern).
- Der Kanal ist fixiert auf "Notiz" (Aus dem Self Service Portal können keine E-Mails versendet werden).

- Der Artikel-Sendertyp ist fixiert auf "extern" und wird nicht dargestellt.
- Die Option "Im Kundenportal anzeigen" kann nicht durch den SSP-Nutzer geändert werden und wird nicht dargestellt.
- Enthält die Konfiguration keine Angabe zu Kanal/Artikel oder werden diese Angaben im Hintergrund gesetzt, stehen deren Eingabefelder nicht zur Verfügung.
Somit können Aktionen ohne Artikel erstellt werden, z. B. initiale Aktion "Customer Feedback". Diese steht im Self Service Portal an geschlossenen Tickets zur Verfügung und erlaubt dem Kunden eine Bewertung inkl. Bemerkung zu einem Ticket abzugeben.

7.5.2 Bereitstellung von Artikelaktionen

Die Bereitstellung von Artikelaktionen ist aktuell noch nicht möglich.

7.5.3 Bereitstellung von Vorlagen

Sie können Ticketvorlagen für das Self Service Portal konfigurieren. Die Konfiguration erfolgt analog der [Vorlagen](#) (siehe Seite 151) für das Agentenportal.

Die Gruppierung von Vorlagen in Vorlagengruppen ist für das Self Service Portal ebenfalls möglich. Sind die für das SSP freigegebenen Vorlagen einer Vorlagengruppe zugeordnet, so wird diese Vorlagengruppe im SSP angezeigt. Vorlagengruppen und Vorlagen werden oberhalb des Editors zur Auswahl angeboten.

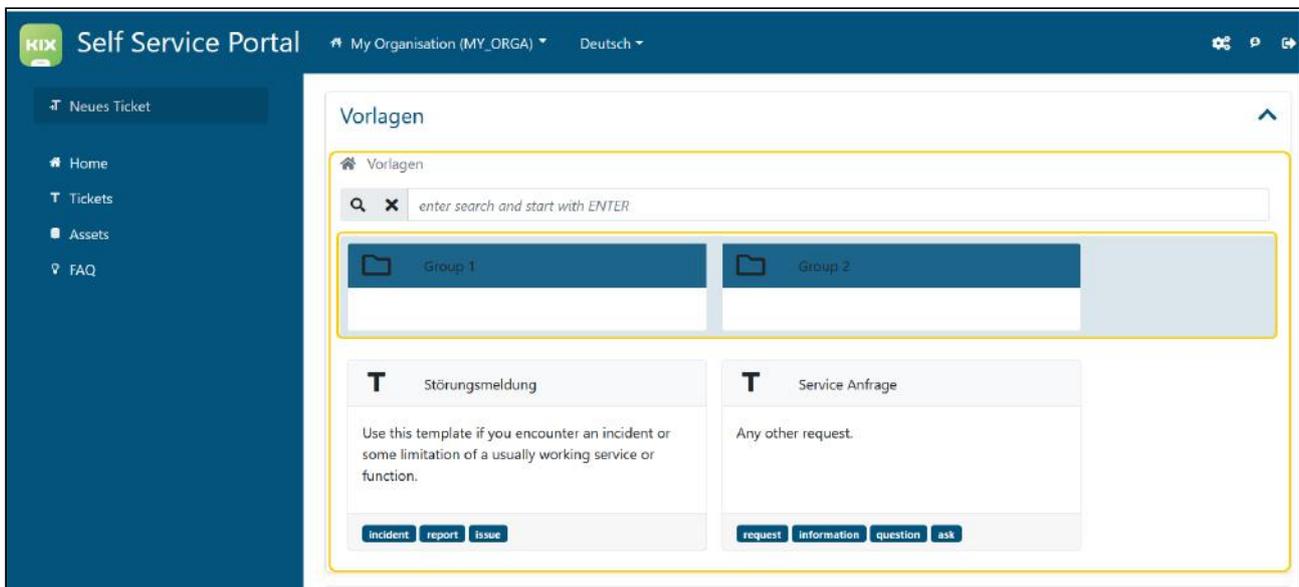


Abb.: Vorlagen und Vorlagengruppen im SSP

Voraussetzungen für die Anzeige der Vorlage im SSP:

- Die Vorlage muss dem Nutzungskontext "Kunde" zugeordnet sein.
Ist die Vorlage dem Nutzungskontext "Kunde" und "Agent" zugeordnet, steht die Ticketaktion in beiden Portalen zur Verfügung.

- Der Vorlage muss mindestens die Rolle "Customer" zugewiesen sein.
- Die Vorlage und ggf. die Vorlagengruppe müssen "gültig" sein.
- Für die verwendeten Dynamischen Felder muss die Option "Im Kundenportal anzeigen" aktiviert sein. (s. auch [Sichtbarkeiten im SSP steuern](#) (siehe Seite 211))

Ist diese Option nicht aktiviert, so wird das Dynamische Feld im SSP nicht angezeigt, auch wenn es in der Vorlagenkonfiguration enthalten ist. Dies ermöglicht eine differenzierte Bereitstellung dynamischer Felder in Ticketvorlagen, die gemeinsam von Agentenportal und SSP genutzt werden.

Hinweise zur Konfiguration von Vorlagen für das SSP:

- Berücksichtigen Sie Berechtigungen: Werden Felder in eine Ticketvorlage konfiguriert, auf die der Kundennutzer keine Berechtigung hat, so schlägt bei ihm deren Verwendung fehl.
- Die Kopfattribute von Tickets (Status, Typ etc.) können nur bearbeitet werden, wenn die Rolle "Customer" um Update-Rechte erweitert wird (Auswirkungen bedenken!) → s. auch [Nutzereinstellungen für das SSP](#) (siehe Seite 209)
- Der Kanal ist fixiert auf "Notiz" (Aus dem Self Service Portal können keine E-Mails versendet werden).
- Der Artikel-Sendertyp ist fixiert auf "extern" und wird nicht dargestellt.
- Die Option "Im Kundenportal anzeigen" kann nicht durch den SSP-Nutzer geändert werden und wird nicht dargestellt.
- Enthält die Vorlagendefinition keine Angabe zu Kanal/Artikel oder werden diese Angaben im Hintergrund gesetzt, stehen deren Eingabefelder nicht zur Verfügung. Somit kann aus einer Vorlage ein Ticket ohne Artikel erstellt werden (z. B. Urlaubsantrag, welcher nur den Bearbeiter und den Zeitraum beinhaltet).

7.5.4 Workflow Rulesets

Sie können das Verhalten von Eingabe- und Auswahlfeldern im Self Service Portal anhand individuell gestalteter Regelsätze (Rulesets) steuern. Die Regeln dafür definieren Sie im Admin Modul des Agentenportals (Menü *Workflow > Rulesets*). Durch Angabe der Bedingung "User.isCustomer" wird die Regel auf das Self Service Portal angewendet. Informationen dazu finden Sie unter [Workflow > Rulesets > Abschnitt "Einschränkung des Nutzungskontextes"](#) (siehe Seite 121) .

Die Workflow-Evaluierung wird nur dann ausgeführt, wenn gültige Rulesets definiert sind UND die Workflow-Evaluierung aktiviert ist. im Auslieferungszustand ist dieses Verhalten aktiviert. Möchten Sie dieses Verhalten deaktivieren, müssen Sie den nachfolgenden Codeabschnitt in der Konfiguration des Self Service Portals einfügen (Menü: "System > GUI Konfiguration > Self Service Portal" > SysConfig-Schlüssel "self-service-portal-configuration")

```
"workflows": {  
  "enabled": false  
}
```



Verweise

- [Ticketaktionen](#) (siehe Seite 93)
- [Vorlagen](#) (siehe Seite 151)
- [Vorlagengruppen](#) (siehe Seite 182)
- [Rule Sets](#) (siehe Seite 121)

7.6 GUI Konfiguration des SSP

Sie können die Benutzeroberfläche des Self Service Portal konfigurieren und damit die dort bereitgestellten Informationen festlegen. Somit können Sie bspw. Dynamische Felder und individuelle Widgets in die Benutzeroberflächen (GUI) des Self Service Portals einbinden oder überflüssige bzw. nicht erwünschte Informationen daraus entfernen.

Inhalte auf dieser Seite:

- [Dashboard Konfigurationen](#) (siehe Seite 232)
 - [Home Dashboard](#) (siehe Seite 232)
 - [Ticket-, Asset-, FAQ- Dashboard](#) (siehe Seite 232)
- [Konfiguration der Detailansichten](#) (siehe Seite 233)
 - [Ticket Details](#) (siehe Seite 233)
 - [Asset Details](#) (siehe Seite 234)
 - [FAQ Details](#) (siehe Seite 235)
- [Neues Ticket](#) (siehe Seite 235)
- [Personal Preferences / Persönl. Einstellungen](#) (siehe Seite 236)
 - [Aktivieren/Deaktivieren der Passwortänderung](#) (siehe Seite 236)
 - [Aktivieren/Deaktivieren der Sprachauswahl](#) (siehe Seite 237)
 - [Konfiguration der Persönlichen Daten](#) (siehe Seite 237)
- [Konfigurationsbeispiele](#) (siehe Seite 238)
 - [Object-information-card-widget](#) (siehe Seite 238)
 - [Tabellenwidget](#) (siehe Seite 239)
 - [Eigenes Widget implementieren](#) (siehe Seite 241)

Die Anpassungen erfolgen wie auch im Agentenportal durch Modifikation der entsprechenden Konfigurationsschlüssel. Die Konfigurationsschlüssel für das Self Service Portal finden Sie im Menü *System > GUI Konfiguration > Self Service Portal*. Alternativ können Sie die Konfigurationsschlüssel im Menü *System > SysConfig* aufrufen und in einem externen Editor (z. B. www.jsonformatter.io)⁵ bearbeiten.

Alle Konfigurationen befinden sich innerhalb der jeweiligen Kontexte. Daher können nur die Kontexte auf oberster Ebene bearbeitet werden (bspw. Home Dashboard, Ticketdetails, Ticketübersicht, etc.) Die einem Kontext untergeordneten Elemente sind nicht konfigurierbar.

⁵ <http://www.jsonformatter.io>)

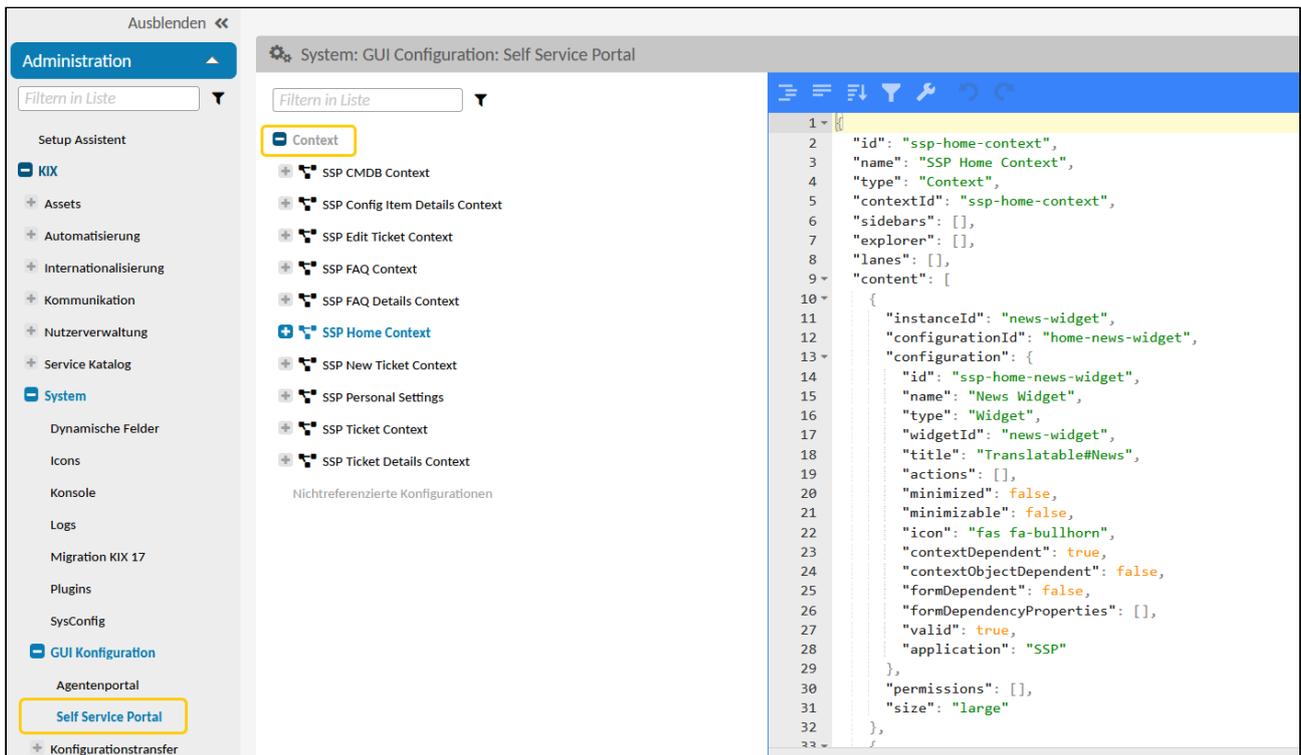


Abb.: Konfiguration der GUI für das Self Service Portal

Bitte beachten:

Klicken Sie nach Änderungen an den Konfigurationen stets auf "Lade Frontendkonfiguration neu", damit die Änderungen ans Self Service Portal übertragen werden. Nutzer des Self Service Portals müssen ggf. ebenfalls das Browserfenster neu laden.

Hinweise:

- Die Bereitstellung und Verwendbarkeit von Attributen und Dynamischen Feldern ist abhängig von den jeweiligen Rechten des angemeldeten Nutzers. Er muss mindestens Lese-Berechtigungen auf das anzuzeigende Objekt haben (z. B. "Betroffenes Asset", Dynamische Felder "Kontakt Referenz" und "Organisations Referenz").

Beispiel:

Ist in die Ticketübersicht eine Spalte zur Anzeige der Betroffenen Assets implementiert, so werden dort nur die Betroffenen Assets angezeigt, auf die der angemeldete Nutzer mindestens Lese-Berechtigungen besitzt.

▼ Tickets					
Ticket#	Betroffenes Asset	Titel	Erstellt am ↓	Status	Prio
2023031517000026	A#175000001 - chprn13-bw	Headset defekt	15.03.2023, 09:50	 offen	
2023031517000017		Headset defekt	15.03.2023, 09:49	 neu	
2023031317000717	A#175000003 - chprn05-color	Drucker druck nicht	13.03.2023, 12:23	 offen	

Abb.: Anzeige der Dynamischen Felder sind abhängig von den Nutzerrechten

- Abhängig von den Benutzerrechten werden auch verschiedene Ticketinformationen im Self Service Portal nicht angezeigt, auch wenn diese bspw. in den Ticketinformationen eingebunden sind (z. B. der Bearbeiter und der Verantwortliche des Tickets).
- Beachten Sie auch die Informationen zu [Sichtbarkeiten im SSP](#) (siehe Seite 211) .

7.6.1 Dashboard Konfigurationen

7.6.1.1 Home Dashboard

SysConfig-Schlüssel	ssp-home-context
----------------------------	------------------

Die Konfiguration umfasst die im Home Dashboard befindlichen Widgets: News, Tickets, Assets, FAQ. Anpassungen dieser Widgets sind nicht erforderlich. Bei Bedarf können Sie jedoch eigene Widgets implementieren (s. unten)

Widget	InstanceID	Konfiguration
News	news-widget	nicht erforderlich
Tickets	home-ticket-widget	nicht erforderlich
Assets	home-asset-widget	nicht erforderlich
FAQ	home-faq-widget	nicht erforderlich

7.6.1.2 Ticket-, Asset-, FAQ- Dashboard

Diese Dashboards beinhalten je eine Übersicht ihrer Objekte. Diese Übersichten sind Tabellen-Widgets, deren Spalten Sie bspw. um Dynamische Felder erweitern können (s. unten).

Die Konfiguration erfolgt jeweils im zum Kontext gehörenden SysConfig-Schlüssel:

Kontext	SysConfig-Schlüssel	InstanceID	Konfiguration
Ticket Dashboard	ssp-ticket-context	ticket-list-widget	möglich
Asset Dashboard	ssp-cmdb-context	asset-list-widget	möglich
FAQ Dashboard	ssp-faq-context	faq-list-widget	möglich

Die Darstellung Dynamischer Felder und Ticketattribute erfolgt in Abhängigkeit der Nutzerrechte.

Im Asset Dashboard können nur Attribute angezeigt werden, die nicht versionsabhängig sind, z. B. Kopfdaten des Assets.

7.6.2 Konfiguration der Detailansichten

7.6.2.1 Ticket Details

Konfigurationsschlüssel	ssp-ticket-details-context
--------------------------------	----------------------------

Die Konfiguration umfasst die in der Ticket Detailansicht enthaltenen Widgets:

Widget	InstanceID	Konfiguration
Ticket Informationen	ticket-information-widget	möglich
Weitere Informationen	ssp-dynamic-fields-widget	nicht erforderlich
Kommunikationshistorie	ticket-article-view	nicht erforderlich

Das Widget "Ticket Informationen" zeigt die Kopfattribute des Tickets wie Priorität, Status und Erstelldatum an. Sie können diesem Objekt-information-card-Widget bei Bedarf weitere Informationen (z. B. Vorfallstatus) hinzufügen (s. unten).

Beachten Sie, dass aufgrund von Benutzerrechten verschiedene Ticketinformationen im Self Service Portal **nicht angezeigt** werden, auch wenn diese in den Ticketinformationen eingebunden sind. Dazu gehören bspw. der Bearbeiter und der Verantwortliche des Tickets.

Es ist nicht erforderlich, Werte Dynamischer Felder in die "Ticket Informationen" zu implementieren. Diese werden im Widget "Weitere Informationen" angezeigt, sofern

- in der Konfiguration des Dynamischen Feldes die Option "Im Kundenportal anzeigen" aktiviert ist
- das Feld einen Wert besitzt
- der Nutzer mindestens Lese-Berechtigung besitzt.

Bei Bedarf können Sie weitere Widgets mit individuellen Inhalten in den Ticketdetails implementieren (s. Konfigurationsmöglichkeiten am Ende der Seite).

Info: Dynamische Felder vom Typ Checklist: Initial werden Checklisten im Widget "Ticket Informationen" nur als Zahlenwert (2/5) dargestellt. Soll hier der Fortschrittsbalken angezeigt werden, setzen Sie den Wert "ComponentID": "dynamic-field-checklist-cell". Im Widget "Weitere Informationen" wird die Checkliste initial als Fortschrittsbalken dargestellt.

Ticketaktionen

In der Ticket Detailansicht stehen Ticketaktionen zur Verfügung. Bei Bedarf können diese vom Kundennutzer verwendet werden, um bspw. Rückfragen an den Helpdesk zu senden oder um zusätzliche Informationen via Dynamische Felder am Ticket zu hinterlegen.

Initial steht im Self Service Portal die Ticketaktion "Rückfrage" bereit, welche einen neuen Artikel am Ticket erzeugt und sinnverwandt ist mit der Ticketaktion "Ticket bearbeiten". Sie können weitere Ticketaktionen konfigurieren und im Self Service Portal bereitstellen (s. auch [Workflow > Aktionen](#) (siehe Seite 93)). Beachten Sie, dass die Kopfattribute des Tickets initial nicht durch den Kundennutzer geändert werden können. Ist dies dennoch erwünscht, müssen Sie der Rolle "Customer" Update-Rechte darauf geben oder ggf. neue Rollen kreieren (s. auch [Nutzereinstellungen für das SSP](#) (siehe Seite 209)).

7.6.2.2 Asset Details

Konfigurationsschlüssel	ssp-config-item-details-context
--------------------------------	---------------------------------

Die Konfiguration umfasst die in der Asset Detailansicht enthaltenen Widgets:

Widget	InstanceID	Konfiguration
Asset Informationen	config-item-information-widget	möglich
Versions Informationen	config-item-version-information-widget	nicht erforderlich

Das Widget "Asset Informationen" zeigt initial die Kopfdaten des gewählten Assets an. Das Widget ist ein Object-information-card-widget, dessen Inhalte Sie nach Bedarf konfigurieren können, um dem Kundennutzer

weitere Asset Informationen bereitzustellen (s. unten). Beachten Sie, dass nur Attribute angezeigt werden können, die nicht versionsabhängig sind, z. B. Kopfdaten des Assets.

Das Widget "Versions Informationen" liefert einen Überblick über die aktuelle Version des gewählten Assets. Es ist fest implementiert und bedarf keiner Änderung.

7.6.2.3 FAQ Details

Konfigurationsschlüssel	ssp-faq-details-context
--------------------------------	-------------------------

Die Konfiguration umfasst die in der FAQ Detailansicht enthaltenen Widgets:

Widget	InstanceID	Konfiguration
FAQ Informationen (Info)	faq-information-widget	möglich
Symptom	faq-symptom-widget	nicht erforderlich
Ursache	faq-cause-widget	nicht erforderlich
Lösung	faq-solution-widget	nicht erforderlich

Das Widget "FAQ Informationen" zeigt initial die Grundinformationen zum FAQ Eintrag an. Das Widget ist ein Object-information-card-widget, dessen Inhalte Sie nach Bedarf konfigurieren können, um dem Kundennutzer weitere Grundinformationen zum FAQ Eintrag bereitzustellen (s. unten).

Die Widgets Symptom, Ursache und Lösung bedürfen keiner Änderung.

7.6.3 Neues Ticket

Konfigurationsschlüssel	ssp-new-ticket-context
--------------------------------	------------------------

Der Dialog zum Anlegen eines neuen Tickets ist fest implementiert. Ein Modifizieren ist nicht erforderlich.

Ticketvorlagen

Zum Anlegen neuer Tickets können Sie unterschiedliche Ticketvorlagen bereitstellen, sodass dem Kundennutzer für jeden Anwendungsfall gesonderte Eingabemasken zur Verfügung stehen. Die Konfiguration von Vorlagen erfolgt im Admin Modul des Agentenportals (s. auch [Workflow > Vorlagen](#) (siehe Seite 151)).

7.6.4 Personal Preferences / Persönl. Einstellungen

Konfigurationsschlüssel	ssp-personal-settings
--------------------------------	-----------------------

Kundennutzer im SSP haben die Möglichkeit, persönliche Systemeinstellungen vorzunehmen. Beispielsweise, um regelmäßig ihr Passwort zu ändern oder die von ihnen bevorzugte Systemsprache einzustellen.



Die Konfiguration des o.g. SysConfig-Schlüssels umfasst die in den Persönlichen Einstellungen enthaltenen Widgets. Sie können festlegen, welche Optionen und Informationen bereitgestellt werden.

Widget	InstanceID	Konfiguration
Passwort ändern	ssp-personal-settings-password-widget	<ul style="list-style-type: none"> • Passwortänderung deaktivieren: Setzen Sie "value" auf "false" • Passwortänderung aktivieren: Setzen de-/aktivieren möglich
Sprache auswählen	ssp-personal-settings-language-widget	de-/aktivieren möglich
Persönliche Daten	ssp-personal-settings-user-info-widget	Konfiguration der Informationen möglich

7.6.4.1 Aktivieren/Deaktivieren der Passwortänderung

InstanceID	ssp-personal-settings-password-widget
-------------------	---------------------------------------

Sie können festlegen, ob dem Kundennutzer die Möglichkeit zum Ändern seines Passwortes zur Verfügung steht. Das Deaktivieren dieser Option kann bspw. erforderlich sein, wenn die Nutzerauthentifizierung der Kundennutzer via LDAP/AD erfolgt und der Nutzer somit sein Passwort nicht im KIX selbst ändern darf.

- Aktivieren: Setzen Sie in der o.g. InstanceID den Wert "value" auf "true".
- Deaktivieren: Setzen Sie in der o.g. InstanceID den Wert "value" auf "false".

```

{
  "instanceId": "ssp-personal-settings-password-widget",
  "configurationId": "ssp-personal-settings-password-widget",
  "configuration": {
    "id": "ssp-personal-settings-password-widget",
    "name": "Change Password",
    "type": "Widget",
    "widgetId": "ssp-personal-settings-password-widget",
    "title": "Translatable#Change Password",
    "actions": [],
    "minimized": false,
    "minimizable": true,
    "icon": "",
    "contextDependent": false,
    "contextObjectDependent": false,
    "formDependent": false,
    "formDependencyProperties": [],
    "value": true,
    "application": "agent-portal"
  }
}

```

Abb.: Aktivierte Passwortänderung

7.6.4.2 Aktivieren/Deaktivieren der Sprachauswahl

InstanceID	ssp-personal-settings-language-widget
-------------------	---------------------------------------

Sie können festlegen, ob der Kundennutzer seine bevorzugte Systemsprache auswählen kann. Die Bereitstellung erfolgt analog des Widgets zur Passwortänderung durch De-/Aktivieren des Widgets.

- Aktivieren: Setzen Sie in der o.g. InstanceID den Wert "value" auf "true".
- Deaktivieren: Setzen Sie in der o.g. InstanceID den Wert "value" auf "false".

7.6.4.3 Konfiguration der Persönlichen Daten

InstanceID	ssp-personal-settings-user-info-widget
-------------------	--

Sie können festlegen, welche Nutzerattribute im Widget angezeigt werden und optional weitere Inhalte bereitstellen, wie bspw. am Kontakt bereitgestellte Dynamische Felder. Die Konfiguration erfolgt wie in jedem anderen Object-information-card-widget (s. unten) durch Hinzufügen oder Entfernen von {"values": [...] }-Code-Blöcken.

Die Bereitstellung Dynamischer Felder ist von den Nutzerrechten abhängig und davon, ob das Dynamische Feld in seiner Konfiguration für das SSP freigegeben ist.

Zudem können Sie die Anzeige des Widgets aktivieren bzw. deaktivieren:

- Aktivieren: Setzen Sie in der o.g. InstanceID den Wert "value" auf "true".

- Deaktivieren: Setzen Sie in der o.g. InstanceID den Wert "value" auf "false".

Tipp: Damit ein Dynamisches Feld in den Persönlichen Daten angezeigt wird, ist es u. U. erforderlich, das Dynamische Feld im Admin Modul erneut zur Bearbeitung zu öffnen und zu speichern und/oder den Konsolenbefehl Console::Command::Maint::Cache::Delete auszuführen.

7.6.5 Konfigurationsbeispiele

Nachfolgend finden Sie Kurzinformationen zur Bereitstellung Dynamischer Felder in Widgets. Ausführliche Informationen zur GUI Konfiguration finden Sie im Admin Handbuch von KIX Start und KIX Pro.

Bitte beachten Sie:

- Im Self Service Portal stehen nicht alle Objekt-Attribute zur Verfügung (z. B.: verfügbar: SLA-Kriterien; nicht verfügbar: SLA-Name)
- Für die Anzeige der Werte benötigt der Customer die entsprechenden Berechtigungen auf Objekteigenschaften (z. B. auf SLAID und SLACriteria) und Ressourcen (z. B. /system/slas).

7.6.5.1 Object-information-card-widget

Um in einem Info-Widget den Wert eines Dynamischen Feldes anzuzeigen, können Sie wie folgt vorgehen:

1. Öffnen Sie den SysConfig-Schlüssel des zu bearbeitenden Kontextes (z. B. ssp-ticket-details-context).
2. Fügen Sie nachfolgenden Codeblock in die Konfiguration des Schlüssels ein.

Platzieren Sie ihn innerhalb des Abschnitt "rows [...]". Die Platzierung bestimmt die Anordnung des Dynamischen Feldes im Widget.

Ersetzen Sie "DFName" durch den Namen des Dynamischen Feldes, welches Sie einbinden möchten.

```
{
  "title": "Translatable#DFName",
  "separator": true,
  "values": [
    [
      {
        "componentId": "ssp-object-detail",
        "componentData": {
          "name": "DFName"
        },
        "conditions": [
          {
            "property": "DynamicFields.DFName",
            "operator": "NE",
            "value": null
          }
        ]
      }
    ]
  ]
}
```

```

    ],
  ],
}

```

3. Fügen Sie den Codeblock ggf. mehrfach ein, wenn Sie weitere Dynamische Felder bereitstellen möchten und verfahren Sie wie unter Schritt 2 beschrieben.
4. Klicken Sie auf Speichern und anschließend auf "Lade Frontendkonfiguration neu".
Danach ist das Dynamische Feld in der Oberfläche integriert, sodass dessen Wert angezeigt werden kann.

Verweise:

- Agentenportal > GUI Konfiguration > Object-information-card-widget
- Ein Object-information-card-widget konfigurieren

7.6.5.2 Tabellenwidget

Um im Self Service Portal einer Tabelle eine weitere Spalte für die Anzeige eines Dynamischen Feldes hinzuzufügen, können Sie wie folgt vorgehen:

1. Öffnen Sie den SysConfig-Schlüssel des zu bearbeitenden Kontextes (z. B. ssp-ticket-context).
2. Fügen Sie nachfolgenden Codeblock in die Konfiguration des Schlüssels ein.
Platzieren Sie ihn innerhalb des Abschnitt " `tableColumns[...]` ". Die Platzierung bestimmt die Anordnung der Spalte innerhalb der Tabelle.
Ersetzen Sie "DFName" durch den Namen des Dynamischen Feldes, welches Sie einbinden möchten.
Ändern Sie optional weitere Spaltenparameter wie bspw. Spaltenbreite (Attribut " `size` ").

```

{
  "id": null,
  "name": null,
  "type": null,
  "property": "DynamicFields.DFName",
  "showText": true,
  "showIcon": false,
  "showColumnName": true,
  "showColumnIcon": false,
  "size": 135,
  "sortable": true,
  "filterable": true,
  "hasListFilter": false,
  "dataType": "STRING",
  "resizable": true,
  "componentId": null,
  "defaultText": null,
  "translatable": false,
}

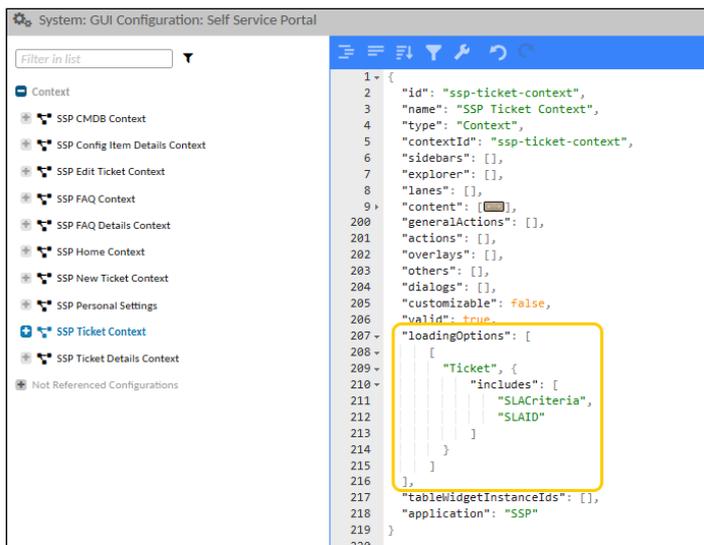
```

```
"titleTranslatable": true,
"useObjectServiceForFilter": false,
"valid": true,
"application": "agent-portal"
},
```

3. Fügen Sie den Codeblock ggf. mehrfach ein, wenn Sie weitere Dynamische Felder bereitstellen möchten und verfahren Sie wie unter Schritt 2 beschrieben.
4. Klicken Sie auf Speichern und anschließend auf "Lade Frontendkonfiguration neu".
Danach ist die Tabelle um die Spalte ergänzt, sodass der Wert des Dynamischen Feldes in der Tabelle angezeigt werden kann.

Hinweise:

- Möchten Sie den Fortschrittsbalken einer Checkliste anzeigen, dann notieren Sie unter "componentID" den Wert "dynamic-field-checklist-cell".
- Zur Darstellung eines SLA Kriteriums verwenden Sie "componentID": "sla-criteria-cell".
- Für die Anzeige der SLA-Kriterien müssen die LoadingOptions für die Tickets um SLAID und SLACriteria erweitert werden:



Verweise:

- Agentenportal > GUI Konfiguration > Tabellen Widget
- Konfiguration von Dashboard-Tabellen
- Konfiguration Tabellen-Widget "Empfohlene FAQ"

7.6.5.3 Eigenes Widget implementieren

Um im Self Service Portal ein individuelles Objekt-information-card-widget zusätzlich einzubinden, können Sie wie folgt vorgehen:

1. Öffnen Sie den SysConfig-Schlüssel des zu bearbeitenden Kontextes (z. B. ssp-ticket-details-context).
2. Fügen Sie nachfolgenden Codeblock in die Konfiguration des Schlüssels ein.
Platzieren Sie ihn innerhalb des Abschnitt " content[...]" . Die Platzierung bestimmt die Anordnung des Widgets im Kontext.

```

{
  "instanceId": "my-ticket-widget",
  "configurationId": "my-ticket-widget",
  "configuration": {
    "id": "my-ticket-widget",
    "name": "My Ticket Widget",
    "type": "Widget",
    "widgetId": "my-ticket-widget",
    "title": "Translatable#My Ticket Widget",
    "actions": [],
    "subConfigurationDefinition": null,
    "configuration": {
      "id": "4711",
      "name": "4711",
      "type": null,
      "valid": true,
      "application": "agent-portal",
      "avatar": [],
      "rows": [
        {
          "title": "Translatable#DF1",
          "separator": true,
          "values": [
            [
              {
                "componentId": "dynamic-field-value",
                "componentData": {
                  "name": "DF1"
                },
                "conditions": [
                  {
                    "property": "DynamicFields.DF1",
                    "operator": "NE",
                    "value": null
                  }
                ]
              }
            ]
          ]
        }
      ]
    }
  ]
}

```

```

    ]
  },
  {
    "values": [
      [
        {
          "componentId": "dynamic-field-value",
          "componentData": {
            "property": "DF2"
          }
        }
      ]
    ],
    "title": "",
    "style": "",
    "separator": false
  }
]
},
"minimized": false,
"minimizable": true,
"icon": "",
"contextDependent": false,
"contextObjectDependent": false,
"formDependent": false,
"formDependencyProperties": [],
"valid": true,
"application": "SSP"
},
"permissions": [],
"size": "large"
},

```

3. Ersetzen Sie "DF1" bzw. "DF2" durch die Namen der Dynamischen Felder, welche im Widget angezeigt werden sollen.
Unter " conditions " können Sie angeben, unter welchen Bedingungen das Feld angezeigt wird. Im o. a. Beispiel darf das DF1 nicht leer sein, damit es angezeigt wird.
4. Sie können dem Abschnitt " rows[...] " weitere Dynamische Felder hinzufügen. Verwenden Sie dazu den Codeblock aus Abschnitt "Object-information-card-widget konfigurieren".
5. Klicken Sie auf Speichern und anschließend auf "Lade Frontendkonfiguration neu".
Danach enthält der Kontext ein weiteres Widget mit individueller Konfiguration.

Verweise:

- Dem Dashboard ein Widget hinzufügen
- Der Sidebar ein Widget hinzufügen

7.7 Layoutkonfiguration

Konfigurationsschlüssel	ssp-layout-configuration
--------------------------------	--------------------------

Inhalte auf dieser Seite:

- [Attribute in der Layoutkonfiguration](#) (siehe Seite 243)
- [Organisationspezifische Layouts](#) (siehe Seite 245)
- [Beispiel Layout-Konfiguration](#) (siehe Seite 247)

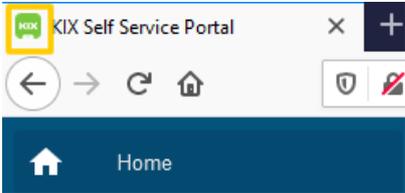
Sie können die Farbdarstellung, den Titel und die verwendeten Logos für das Self Service Portal ändern und damit die Darstellung des Self Service Portals an Ihr Corporate Identity anpassen.

Ändern Sie dazu im Menü *System > GUI Konfiguration > Self Service Portal* die Werte im Schlüssel "ssp-layout-configuration":

- **Farbdarstellung ändern:** Ersetzen Sie die Farbwerte (primary, secondary, accent) durch die von Ihnen gewünschten Farbwerte. Es werden nur Hexadezimalwerte unterstützt.
- **Favicon und Logo ändern:** Konvertieren Sie Ihr Favicon bzw. Logo ins base64-Format und ersetzen Sie damit die initialen Angaben. Es wird nur das base64-Format unterstützt.
Diverse Online-Tools helfen, eigene Grafiken nach base64 zu codieren, z. B.:
 - <https://www.base64-image.de/>
 - <https://www.base64encode.org/>
 - <https://onlinejpgtools.com/convert-jpg-to-base64>
- **Titel ändern:** Ersetzen Sie den unter "title" angegebenen Titel durch den von Ihnen gewünschten Titel.

7.7.1 Attribute in der Layoutkonfiguration

Attribut	Beschreibung
ID	Die ID (Nummer) des Konfigurationsblocks. Sie darf nur 1x im gesamten Schlüssel existieren. Vergeben Sie jedem Konfigurationsblock eine eigene ID, wenn Sie Layouts für mehrere Organisationen konfigurieren.
name	Frei definierbare Bezeichnung des Konfigurationsblocks Sie können hier bspw. beschreiben, für welche Organisation die Konfiguration gilt.

Attribut	Beschreibung
vHostPattern	<p>Regulärer Ausdruck des VHosts (z. B.: "<code>^service4companyA\\.kix\\.de*</code>")</p> <p>KIX prüft den Ausdruck gegen den aktuellen Host der Domain. Wird der VHost nicht gefunden, wird der nächste erreichbare VHost verwendet (i.d.R. der Host der Default-Konfiguration)</p> <p>Beachten Sie: Die Reihenfolge ist entscheidend. KIX arbeitet die Konfiguration von oben nach unten ab; der erste Treffer führt zum Match. D. h.: Ist in der ersten Konfiguration ein RegEx <code>.*</code> angegeben, wird diese Konfiguration immer verwendet.</p>
primaryOrgID	<p>ID der Organisation (optional)</p> <ul style="list-style-type: none"> • Ist eine ID angegeben, wird die Konfiguration auf die Organisation mit dieser ID angewendet. • Ist keine ID angegeben, wird die Konfiguration auf alle Organisationen angewendet, für die keine andere Konfiguration zutrifft (Default und Fallback).
design	<p>Konfigurationsblock für die Layout Definition</p>
favicon	<p>Definiert das Favicon Das Favicon wird als Tab-Icon im Browser angezeigt.</p>  <ul style="list-style-type: none"> • <code>content</code>: base64-String des Favicons (s. oben) • <code>contentType</code>: Datenformat des Favicons (z. B. <code>img/jpg</code>, <code>img/png</code>, <code>xml/svg</code>)
logo	<p>Definiert das Logo Das Logo wird im Anmeldefenster und in der Titelzeile angezeigt.</p>  <ul style="list-style-type: none"> • <code>content</code>: base64-String des Logos (s. oben) • <code>contentType</code>: Datenformat des Logos (<code>img/jpg</code> oder <code>xml/svg</code>)

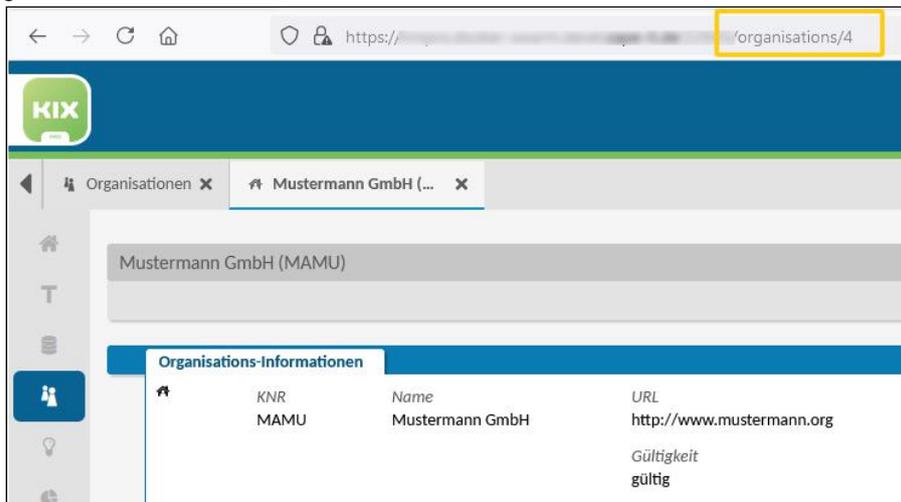
Attribut	Beschreibung																				
colorScheme	<p>Definiert das Farbschema</p> <p>Sie können die Farben ändern. Es werden nur Hexadezimalwerte unterstützt. Die Farbänderungen wirken sich nur auf den "light mode" aus, nicht auf den "dark mode".</p> <p>Defaultwerte:</p> <table border="1"> <thead> <tr> <th>Attribut</th> <th>Beschreibung</th> <th>Farbe</th> <th>Farbcode</th> <th>Verwendung u.a. in</th> </tr> </thead> <tbody> <tr> <td>primary</td> <td>Primärfarbe</td> <td>Blau</td> <td>#04537D</td> <td>Titelzeile, Menüleiste, Überschriften</td> </tr> <tr> <td>secondary</td> <td>Sekundärfarbe</td> <td>Grau</td> <td>#5B5B5B</td> <td>Abbrechen-Button</td> </tr> <tr> <td>accent</td> <td>Akzentfarbe</td> <td>Grün</td> <td>#82C826</td> <td>Label der Vorlagen</td> </tr> </tbody> </table>	Attribut	Beschreibung	Farbe	Farbcode	Verwendung u.a. in	primary	Primärfarbe	Blau	#04537D	Titelzeile, Menüleiste, Überschriften	secondary	Sekundärfarbe	Grau	#5B5B5B	Abbrechen-Button	accent	Akzentfarbe	Grün	#82C826	Label der Vorlagen
Attribut	Beschreibung	Farbe	Farbcode	Verwendung u.a. in																	
primary	Primärfarbe	Blau	#04537D	Titelzeile, Menüleiste, Überschriften																	
secondary	Sekundärfarbe	Grau	#5B5B5B	Abbrechen-Button																	
accent	Akzentfarbe	Grün	#82C826	Label der Vorlagen																	
imprint	<p>Optionaler Link zum Impressum als HTML-String, z. B.:</p> <ul style="list-style-type: none"> <code>"body" : "Impressum"</code> <code>"body" : "Impressum"</code> <p>Der Link wird in der Fußzeile des Self Service Portals angezeigt.</p>																				
title	<p>Definiert den Titel im Self Service Portal, z. B: <code>"title": "KIX Self Service Portal"</code></p> 																				

7.7.2 Organisationspezifische Layouts

Sie können für jede in KIX angelegte Organisation ein eigenes Layout definieren und damit das Layout des Self Service Portals dem Corporate Identity des jeweiligen Unternehmens anpassen:

1. Duplizieren Sie den initialen JSON-Code (ggf. mehrfach) und geben Sie jedem Konfigurationsblock eine eigene ID.
2. Geben Sie jeweils unter "primaryOrgID" die ID der Organisation an, für die diese Konfiguration gilt. Sie können mehrere Organisations-IDs angeben, um ein Layout für mehrere Organisationen zu verwenden.

Die ID einer Organisation können Sie der URL entnehmen, wenn Sie die Detailansicht der Organisation geöffnet haben.



3. Passen Sie in jedem Konfigurationsblock die Farbwerte, Titel, Logos und ggf. den Link zum Impressum an.

Hinweis

KIX arbeitet die Konfiguration von oben nach unten ab und fällt im Bedarfsfall selbständig auf das Standardverhalten (Vorgabewerte) zurück. Platzieren Sie daher die Default-Konfiguration mit vHostPattern ".*" ganz nach unten. Sie dient damit als Fallback und als Standardkonfiguration, falls der Endpunkt nicht erreichbar ist oder keine Antwort liefert.

7.7.3 Beispiel Layout-Konfiguration

Beispielkonfiguration für 2 Organisationen und Default

```
{
  "id": 0,
  "name": "Layout für Kunde A",
  "vHostPattern": "service4companyA.kix...*",
  "primaryOrgId": [23, 24, 25],
  "design": {
    "favIcon": {
      "content": "<base64-string>",
      "contentType": "img/png"
    },
    "logo": {
      "content": "<base64-string>",
      "contentType": "img/jpg"
    },
    "colorScheme": {
      "primary": "#ffcc00",
      "secondary": "#333333",
      "accent": "#f0f0f0"
    },
    "imprint": {
      "body": "<a href=\"https://url.to.your.imprint.com\">Imprint</a>"
    },
    "title": "Customer A – Self Service"
  }
},
{
  "id": 1,
  "name": "Layout für Kunde B",
  "vHostPattern": "service4companyB.kix....*",
  "primaryOrgId": "456",
  "design": {
    "favIcon": {
      "content": "<base64-string>",
      "contentType": "xml/svg"
    },
    "logo": {
      "content": "<base64-string>",
      "contentType": "img/png"
    },
    "colorScheme": {
      "primary": "#333300",
      "secondary": "#336600",
      "accent": "#339900"
    }
  }
},
```



```
"imprint": {
  "body": "<a href=\"https://url.to.your.imprint.com\">Imprint</a>"
},
"title": "Customer B – Self Service"
}
},
{
  "id": 9999999,
  "name": "Default Layout",
  "vHostPattern": ".*",
  "primaryOrgId": [],
  "design": {
    "favIcon": {
      "content": "<base64-string>",
      "contentType": "image/png"
    },
    "logo": {
      "content": "<base64-string>",
      "contentType": "image/png"
    },
    "colorScheme": {
      "primary": "#04537D",
      "secondary": "#5b5b5b",
      "accent": "#82C826"
    },
    "imprint": {
      "body": "<a href=\"https://kixdesk.com\">Impressum</a>"
    },
    "title": "KIX Self Service Portal"
  }
}
```

8 News

News sind Hinweise und Informationen, die dem Nutzer vor und/oder nach dem Login angezeigt werden. Sie können formatierten Text, Tabellen, Links und Bilder beinhalten und zeitgesteuert angezeigt werden. Als Administrator können Sie News bspw. nutzen, um Nutzer über aktuelle Störungen, anstehende Änderungen oder Wartungsarbeiten zu informieren.

Die Verwaltung von News-Meldungen finden Sie im Modul "News" . Agenten mit Rolle "Newsmanager" können dort ebenfalls News-Meldungen verwalten. Eine detaillierte Beschreibung zum Anlegen und Verwalten von News finden Sie daher im Kapitel News Management des KIX Pro Anwenderhandbuchs.

News können wahlweise

- im Agentenportal und/oder Self Service Portal (Nutzungskontext)
- vor und/oder nach dem Nutzerlogin (Login Kontext)

angezeigt werden. Die Anzeige wird gesteuert durch die Auswahl des Kontextes:

Nutzungskontext: Agent + Login Kontext: Vor Login:

- Die News-Meldungen werden oberhalb der Eingabefelder angezeigt.
- Ein Klick auf das Plus-Symbol klappt die Meldung zur vollständigen Ansicht auf.

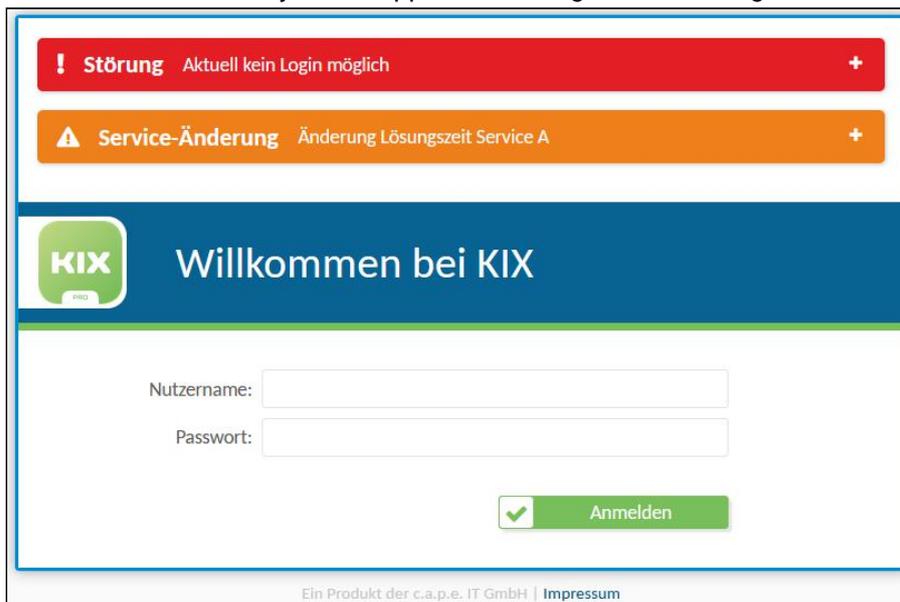


Abb.: News Meldung vor dem Login am Agentenportal

Nutzungskontext: Agent + Login Kontext: Nach Login:

- Die News-Meldungen werden in der Persönlichen Werkzeugleiste des Nutzers angezeigt.
- Kommt es zu Fehlern beim Laden/Erstellen/Updaten/Löschen von Objekten, werden diese ebenfalls angezeigt.
- Die Meldungen werden nach ihrer Art gruppiert.
- Der Klick auf eine Meldung öffnet sie zur vollständigen Anzeige in einem Overlay.

- Einzelne Meldungen und Meldungsgruppen können mit Klick auf das Kreuz aus der Liste entfernt werden.

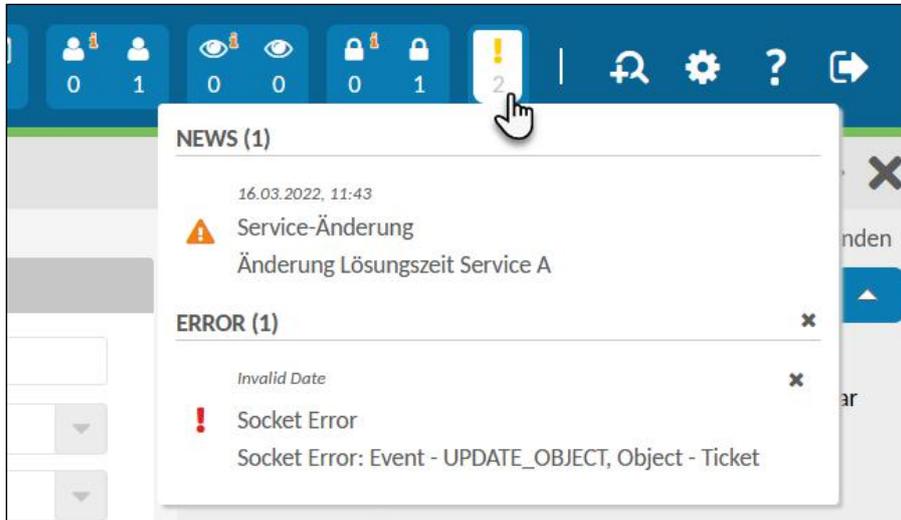


Abb.: News-Meldungen im Agentenportal

Nutzungskontext: Kunde + Login Kontext: Vor Login:



Abb.: News Meldung vor dem Login am Self Service Portal

Nutzungskontext: Kunde + Login Kontext: Nach Login:

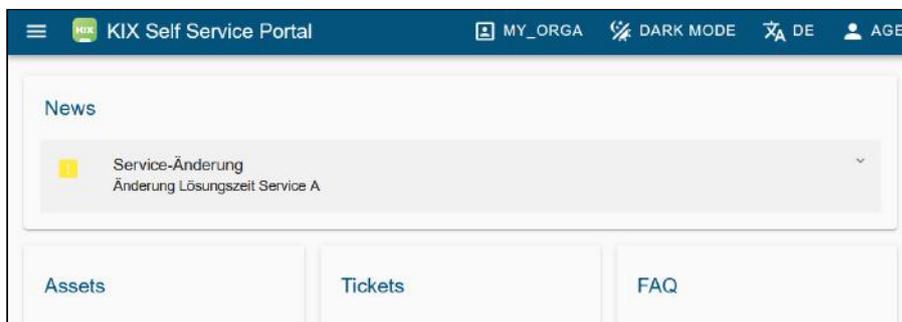


Abb.: News-Meldungen im Self Service Portal



9 Zusätzliche Konfigurationsmöglichkeiten KIX Pro

Die nachfolgenden Kapitel umfassen zusätzliche Konfigurationsmöglichkeiten in den Agenten-Modulen von KIX Pro.

- [Konfiguration der Teamansichtsmodi](#) (siehe Seite 252)
- [Zeiterfassung](#) (siehe Seite 263)
- [KIX Pro REST API](#) (siehe Seite 268)
- [Kind-Tickets in den Ticketdetails auflisten](#) (siehe Seite 269)
- [Objekt-Historie konfigurieren](#) (siehe Seite 273)
- [Konfiguration des Standortexplorers](#) (siehe Seite 275)

9.1 Konfiguration der Teamansichtsmodi

Konfigurationsschlüssel	tickets
--------------------------------	---------

Im Ticketdashboard werden die Tickets sortiert nach Teams angezeigt. Agenten in KIX Pro können hier zwischen verschiedenen Ansichtsmodi wählen:

- Liste
- Kanban
- Kalender
- Landkarte

Inhalte auf dieser Seite:

- [Listenansicht](#) (siehe Seite 253)
- [Kanban-Ansicht](#) (siehe Seite 254)
- [Kalender-Ansicht](#) (siehe Seite 255)
- [Kartenansicht](#) (siehe Seite 256)
- [De-/aktivieren von Teamansichtsmodi](#) (siehe Seite 261)

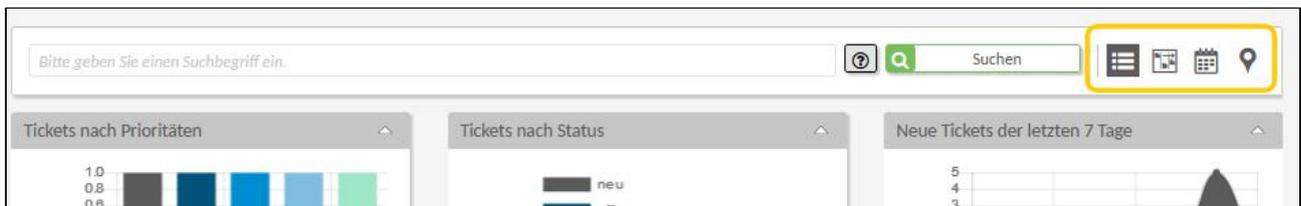


Abb.: Die Teamansichtsmodi im Ticketdashboard

Als Administrator können Sie die Bereitstellung der Ansichtsmodi de-/aktivieren und deren Inhalte konfigurieren. Dies erfolgt im SysConfig-Schlüssel "tickets":

- Abschnitt " `content[]` " : enthält die in der Listenansicht angezeigten Widgets
- Abschnitt " `others[]` " : enthält die Konfigurationsblöcke für die Kanban-, Kalender- und Karten-Ansicht sowie den Konfigurationsblock zum de-/aktivieren der Ansichtsmodi.

```

1  {
2  "id": "tickets",
3  "name": "Ticket Dashboard Configuration",
4  "type": "Context",
5  "contextId": "tickets",
6  "sidebars": [],
7  "explorer": [{}],
8  "lanes": [],
9  "content": [{}],
10 "generalActions": [],
11 "actions": [],
12 "overlays": [],
13 "others": [
14   {
15     "instanceId": "maps",
16     "configurationId": null,
17     "configuration": [{}],
18     "permissions": [],
19     "size": "large"
20   },
21   {
22     "instanceId": "kanban",
23     "configurationId": null,
24     "configuration": [{}],
25     "permissions": [],
26     "size": "large"
27   },
28   {
29     "instanceId": "calendar",
30     "configurationId": null,
31     "configuration": [{}],
32     "permissions": [],
33     "size": "large"
34   },
35   {
36     "instanceId": "ticket-views",
37     "configurationId": "ticket-views",
38     "configuration": [{}],
39     "permissions": [],
40     "size": "large"
41   }
42 ],
43 "dialogs": [],
44 "customizable": false,
45 "valid": true
46 }

```

Abb.: Konfigurationsblöcke der Teamansichtsmodi

Info

Klicken Sie nach Änderungen an der Konfiguration auf "Lade Frontendkonfiguration neu", um die Anzeige im Frontend zu aktualisieren.

9.1.1 Listenansicht

Die Listenansicht ist die Standardansicht im Ticketdashboard. Im Abschnitt "content[]" sind die Chart-Widgets und Tickettabelle eingebunden. Deren Inhalte können Sie im SysConfig-Schlüssel des jeweiligen Widgets (s. Parameter "instanceID") konfigurieren. Näheres dazu finden Sie unter:

- Das Chart Widget
- Das Table-widget
- Konfiguration von Dashboard-Tabellen

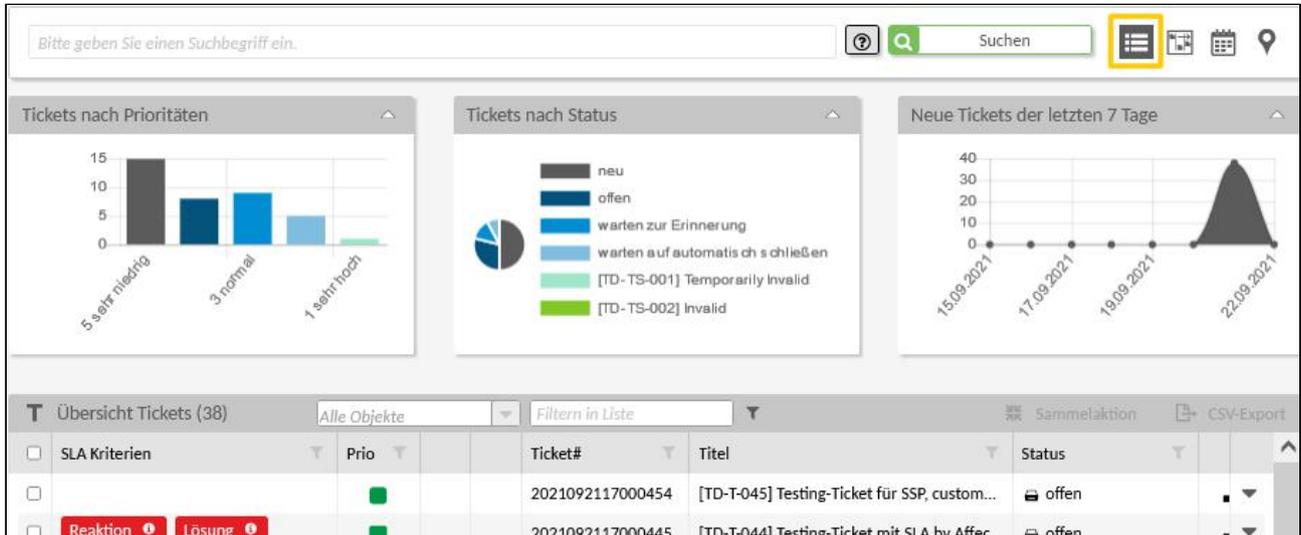


Abb.: Modus Listenansicht

9.1.2 Kanban-Ansicht

Die Kanban-Ansicht ermöglicht eine visuelle Trennung der Tickets nach Bearbeitungsstatus. Die Tickets werden als einzelne Karten dargestellt, welche vom Bearbeiter per Drag & Drop in eine andere Spalte des Kanbanboards verschoben werden können. Der Bearbeitungsstatus des Tickets ändert sich dabei automatisch. Die Karten zeigen verschiedene, konfigurierbare Ticketinformationen an. Als Administrator können Sie festlegen, welche Ticketinformationen in den Karten angezeigt werden sollen.

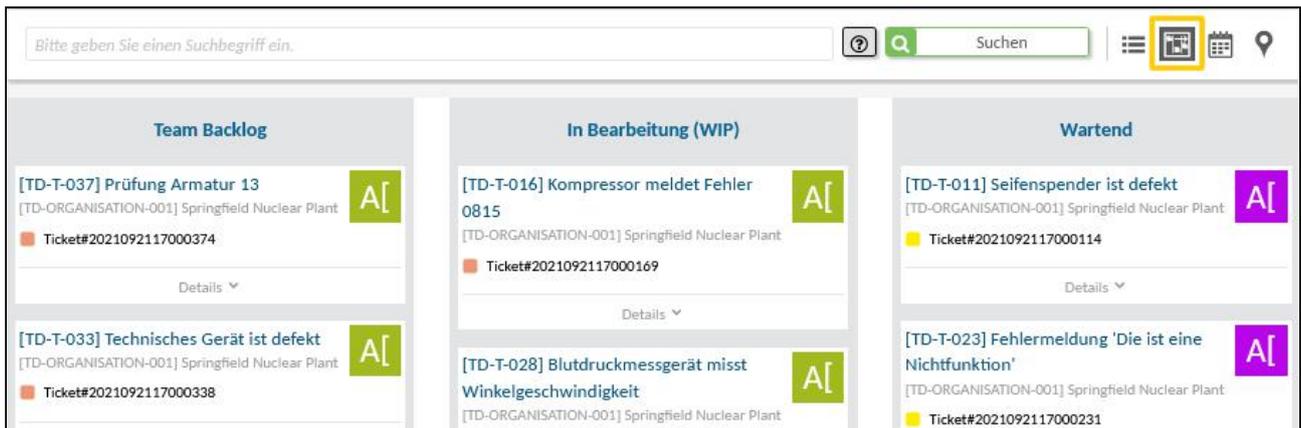


Abb.: Modus Kanban-Ansicht

Die Konfiguration der Karteninhalte erfolgt im Abschnitt "others[]" im Konfigurationsblock

```
{ instanceID": "kanban"... }
```

Parameter	Beschreibung	Werte
cardProperties	Ticketeigenschaften, die im Overlay des Ticketmarkers angezeigt werden sollen. Sie können nicht benötigte Informationen aus der Karte entfernen bzw. weitere Ticketinformationen hinzufügen, z. B. den geplanten Zeitaufwand oder andere Dynamische Felder.	Default: "ContactID", "StateID", "QueueID", "ResponsibleID", "Changed" Beispiel: "ContactID", "StateID", "ResponsibleID", "DynamicFields.PlannedEffort", "DynamicFields.DFxyz"
columns	<p>Spalten des Kanbanboards Default: Team Backlog, In Bearbeitung (WIP), Wartend</p> <p>Sie können dem Kanbanboard weitere Spalten hinzufügen und den Status ändern.</p> <p>id: Definiert je eine Spalte im Kanbanboard. Möglich sind:</p> <ul style="list-style-type: none"> • team-backlog (Team Backlog) • wip (In Bearbeitung) • pending (Wartend) • closed (kürzlich geschlossen) <p>dropState: Definiert den Status, den das Ticket erhalten soll, wenn das Ticket in die entsprechende Spalte verschoben wird. Möglich sind alle im System angelegten Status (Menü <i>Ticket</i> > <i>Status</i>).</p>	<pre>"columns": [{ "id": "team-backlog", "dropState": "new" }, { "id": "wip", "dropState": "open" }, { "id": "pending", "dropState": "pending reminder" }]</pre>

9.1.3 Kalender-Ansicht

In der Kalender-Ansicht werden die Tickets entsprechend ihres Fälligkeitsdatums angezeigt. Wird ein Ticket im Kalender angeklickt, öffnet sich ein Overlay mit einer Karte, welche detaillierte Informationen zum Ticket enthält.

Für die Anzeige der Tickets im Kalender werden standardmäßig die unter "Plan Beginn" und "Plan Ende" hinterlegten Datumsangaben genutzt.

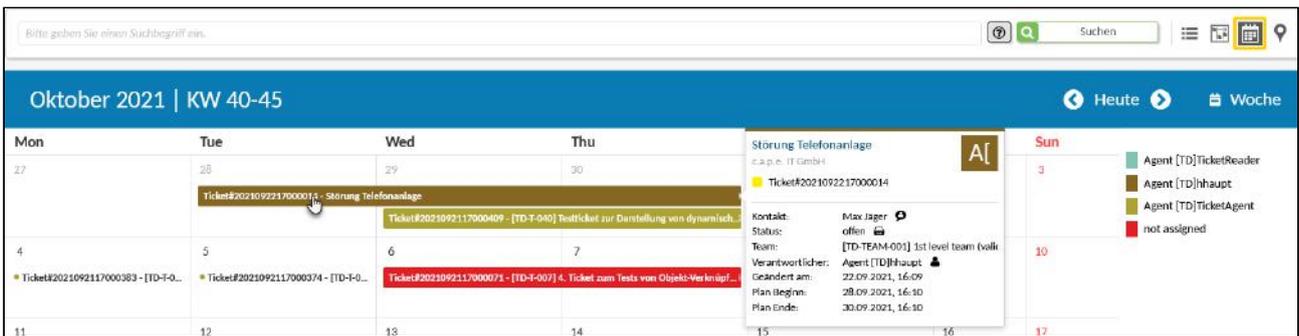


Abb.: Modus Kalender-Ansicht

Sie können den Karteninhalt des Overlays konfigurieren. Die Konfiguration erfolgt im Abschnitt "others[]" im Konfigurationsblock "{ instanceID": "calendar"... }.

Parameter	Beschreibung	Beispiel
properties	<p>Ticketeigenschaften, welche im Karteninhalt (Overlay) angezeigt werden sollen.</p> <p>Sie können nicht benötigte Informationen der Karte entfernen bzw. weitere Ticketinformationen hinzufügen, z. B. den geplanten Zeitaufwand oder eigene Dynamische Felder.</p>	<pre>"properties": ["ContactID", "StateID", "QueueID", "DynamicFields.PlanBegin", "DynamicFields.PlanEnd", "DynamicFields.PlannedEffort", "DynamicFields.DFxyz"]</pre>

9.1.4 Kartenansicht

Die Kartenansicht visualisiert die Einsatzorte für Tickets sowie die aktuelle Position der Agenten. Sie ermöglicht damit die Planung und Steuerung von Außeneinsätzen. Eingehende Meldungen/Tickets können so direkt an den Servicemitarbeiter gegeben werden, welcher sich in der Nähe des Einsatzortes befindet. Dadurch werden effektive Reaktionszeiten verkürzt und Fahrtzeiten reduziert.

Ändern sich die Daten bspw. bei Positionsänderung des Agenten, wird die Karte automatisch aktualisiert. Dazu muss der Agent die Field Agent App mit eingeschaltetem Geo-Tracking auf seinem Mobilgerät verwenden.

Hinweise zur Verwendung der Kartenansicht finden Sie auch im KIX 18 Pro Anwenderhandbuch.

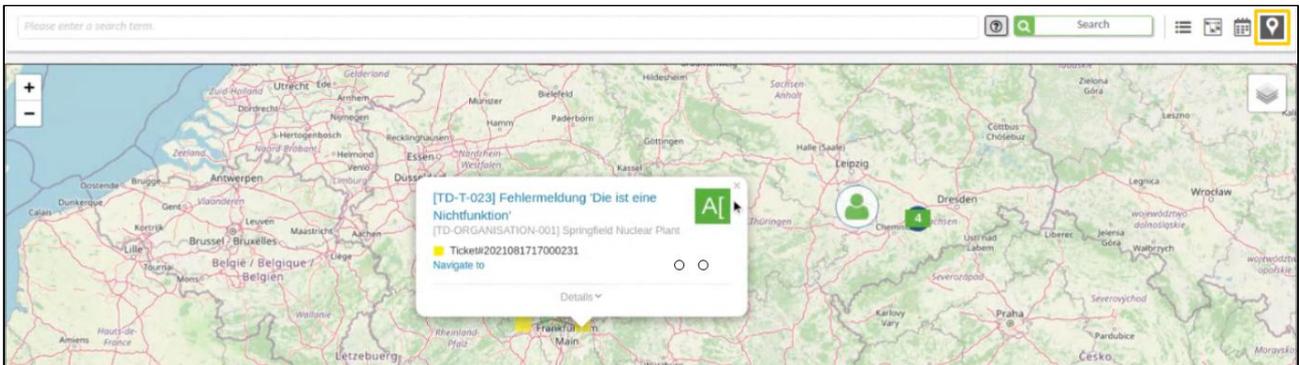


Abb.: Modus Karten-Ansicht

Die Karte verwendet Open Street Maps zur Darstellung. Als Geodaten-Provider zum Auflösen von Adressangaben in Geo-Koordinaten wird <https://nominatim.org> verwendet. Die vorgegebenen Grundlagen sind:

- für die Darstellung eines Tickets in der Karte: die Adressdaten der am jeweiligen Ticket hinterlegten Organisation.
- für die Darstellung der Agentenposition: die Adresse des Agentennutzers.

Die Adressdaten sind in der Konfiguration als KIX Platzhalter angegeben. Sie haben dadurch die Möglichkeit, auch alternative Quellen für Adressdaten wie bspw. Dynamische Felder, Adresse des Agentennutzers, o. ä. zu verwenden. Soll bspw. die Anschrift eines Assets verwendet werden, kann diese mittels eines Automatisierungsjobs und der Macro Action "Asset Attribute abrufen" in ein Dynamisches Feld ins Ticket gesetzt werden. In der Kartenkonfiguration können Sie dann via Platzhalter auf dieses Dynamische Feld referenzieren.

📘 Beschränkung des Vorgabe Geodaten-Providers

Die Auflösung von Geodaten bei Verwendung eines nominatim.org-Services ist auf eine maximale Anzahl von Requests pro Zeiteinheit und IP-Adresse begrenzt (siehe <https://operations.osmfoundation.org/policies/nominatim/>). Ein produktiver Einsatz wird nicht empfohlen. Es besteht das Risiko einer zeitweisen Sperre für Adressanfragen wodurch die Kartenfunktion nicht mehr zur Verfügung steht. Wir empfehlen die Verwendung eines alternativen Geodatenproviders wie <https://opencagedata.com>.

Die Konfiguration der Geo-Positionen erfolgt im Abschnitt "`others[]`" im Konfigurationsblock `"{ instanceID": "maps"... }`.

Parameter	Beschreibung	Werte
initialCoordinates	Initiale Koordinationsdaten, mit der die Karte geladen wird.	<ul style="list-style-type: none"> • Default: 51.324572, 10.519871 (Mitte von Deutschland) • Beispiel: 49.939680368625325, 10.314198730887163 (Europa)
initialZoom	Initiale Zoomstufe, mit der die Karte geladen wird.	<ul style="list-style-type: none"> • Default: 6 • Beispiel: 5
addressDefinition	Definiert einen Adress-String für die Ticketmarker Aufgrund dieser Informationen wird der Ort des Tickets in der Karte angezeigt.	<ul style="list-style-type: none"> • Default: Adresse der am Ticket gesetzten Organisation: "<KIX_ORG_Street>, <KIX_ORG_Zip>, <KIX_ORG_City>" • Beispiel Alternative: "<KIX_TICKET_DynamicField_Einsatzort>"
layerGroups	Wird aktuell noch nicht unterstützt. Aktuell werden nur Layer für Ticket Owner unterstützt.	"OrganisationID", "OwnerID"
markerProperties	Ticketeigenschaften, die im Overlay des Ticketmarkers angezeigt werden sollen. Sie können dem Overlay weitere Ticketinformationen hinzufügen bzw. nicht benötigte Informationen entfernen, z. B. den geplanten Zeitaufwand oder andere Dynamische Felder.	<ul style="list-style-type: none"> • Default: "ContactID", "StateID", "QueueID", "ResponsibleID", "Changed" • Beispiel: "ContactID", "StateID", "ResponsibleID", "DynamicField.PlannedEffort", "DynamicField.DFxyz"

Parameter	Beschreibung	Werte
geoDataProvider	<p>Angabe eines alternativen Geodaten-servers für KIX Cloud Instanzen, z.B. https://opencagedata.com zur Auflösung von Adressdaten und der Darstellung von Einsatzorten (Tickets) und Agentenpositionen in der Kartendarstellung.</p> <ul style="list-style-type: none"> • url: URL der API mit Platzhalter <SEARCH_VALUE>, in welchem der definierte Adress-String eingesetzt wird • latProperty: Array, das den Property-Pfad zu Latitude-Eigenschaft in der Antwort des Geo-Daten-Providers beschreibt. • lonProperty: Array, das den Property-Pfad zu Longitude-Eigenschaft in der Antwort des Geo-Daten-Providers beschreibt. • useTimeout: <code>true</code>, wenn nur 1 Anfrage pro Sekunde an den Geodaten-Provider gesendet werden soll. 	<pre>"geoDataProvider": { "url": "https:// api.opencagedata.com/geocode/ v1/json? q=&key=YOUR_API_TOKEN_HERE", "latProperty": ["results", "0", "geometry", "lat"], "lonProperty": ["results", "0", "geometry", "lng"], "useTimeout": true }</pre>

Parameter	Beschreibung	Werte
geoPositonTTL	<p>Ablaufzeit des Positionsdaten (in Minuten)</p> <p>Die Field Agent App meldet die Geopositionen mit einem Zeitstempel. Hier definieren Sie, nach wieviel Minuten der Zeitstempel als "abgelaufen" gilt und die Positionsdaten verfallen.</p> <p>Die Position des Agenten wird in der Karte angezeigt, solange der Zeitstempel noch nicht abgelaufen ist. Ändert der Agent seine Position, wird diese inkl. eines neuen Zeitstempels übermittelt. Grundlage für die Meldung der Positionsänderung sind die Angaben in den Konfigurationsschlüssen (s. unten: Aktualisierung von Geopositionsdaten). Nach Ablauf des Zeitstempels resultiert die Agentenposition aus dessen Adresse anhand seiner hinterlegten Kontakt- oder Organisationsdaten.</p>	Default: <code>"geoPositionTTL": 480</code>

 **Tipp**

Können Geopositionsdaten nicht aufgelöst werden oder liefert der Geodaten-Provider einen Fehler, können Sie über die Browser-Konsole nach den Gründen recherchieren (Taste F12 - Entwicklerwerkzeuge). Sie erhalten dort einen einen entsprechenden Log-Eintrag.

9.1.4.1 Aktualisierung von Geopositionsdaten

Nutzt ein Servicemitarbeiter die Field Agent App, so kann die App die Geopositionsdaten des mobilen Geräts an das Agentenportal übermitteln. KIX kann anhand der übertragenen Positionsdaten die Agentenposition in der Map-Ansicht des Homedashboards darstellen.

Sie können das Sendeverhalten der App im Menü *System* > *SysConfig* steuern. Dies erfolgt in folgenden Konfigurationsschlüsseln:

Schlüssel	Beschreibung	Werte
KIXMobileApp::GeoPosition::ChangeThreshold	Legt fest, ab welcher Positionsänderung die App die neue Position sendet (Entfernung in Metern).	Natürliche Zahlen >0 Bei ungültigen Eingaben wird der Standardwert verwendet. Default: 1000
KIXMobileApp::GeoPosition::EnabledByDefault	Legt fest, ob die Meldung der Position nach erstmaliger Anmeldung des Nutzers an der App aktiv oder inaktiv ist. Für die weitere Nutzung gelten die Einstellungen in der App.	0 - inaktiv (Default) 1 - aktiv

9.1.5 De-/aktivieren von Teamansichtsmodi

Sie können die Bereitstellung der Teamansichtsmodi global via SysConfig steuern. Nicht benötigte Teamansichtsmodi können somit deaktiviert und bei Bedarf wieder aktiviert werden. Die Schaltflächen deaktivierter Teamansichtsmodi sind ausgeblendet und können somit nicht ausgewählt werden.



Abb.: Bereitstellung Teamansichtsmodi (alle und teilweise)

Der Abschnitt "ticket-views" im Codeblock "others[]" enthält den Parameter "widgets". Hier können Sie für jeden Ansichtsmodus festlegen, ob dieser zur Verfügung steht. Setzen Sie dazu den Parameter "valid" auf true oder false.

```

101 "actions": [],
102 "overlays": [],
103 "others": [
104   {
105     "id": "ticket-views",
106     "name": "Ticket views",
107     "type": "Widget",
108     "widgetId": null,
109     "title": "",
110     "actions": [],
111     "subConfigurationDefinition": null,
112     "configuration": {
113       "id": "ticket-ticket-views-widget-config",
114       "name": "Ticket views Config",
115       "type": "Widget",
116       "widgets": [
117         {
118           "id": "ticket-view-list",
119           "name": "List",
120           "type": "Table",
121           "widgetId": null,
122           "displayName": "Translatable#List",
123           "icon": "kix-icon-legend",
124           "valid": true
125         },
126         {
127           "id": "ticket-view-kanban",
128           "name": "Kanban",
129           "type": "Kanban",
130           "widgetId": "kanban",
131           "displayName": "Translatable#Kanban",
132           "icon": "kix-icon-kanban",
133           "valid": false
134         },
135         {
136           "id": "ticket-view-calendar",
137           "name": "Calendar",
138           "type": "Calendar",
139           "widgetId": "calendar",
140           "displayName": "Translatable#Calendar",
141           "icon": "kix-icon-calendar",
142           "valid": true
143         },
144         {
145           "id": "ticket-view-maps",
146           "name": "Maps",
147           "type": "Maps",
148           "widgetId": "maps",
149           "displayName": "Translatable#Maps",
150           "icon": "kix-icon-tour",
151           "valid": false
152         }
153       ]
154     }
155   }
156 ]
  
```

Abb.: De-/aktivierung Teamansichtsmodi im SysConfig-Schlüssel "tickets"

9.2 Zeiterfassung

Agenten haben in KIX Pro die Möglichkeit, eine einfache Zeiterfassung am Ticket vorzunehmen. Dazu wird am Ticket die geplante Sollzeit (Geplanter Aufwand) festgelegt. Bei jeder Bearbeitung des Tickets kann ein Agent angeben, wieviel Zeit er benötigt hat (Feld "Zeit buchen"). Die Summe der gebuchten Zeiten wird von der geplanten Sollzeit subtrahiert. Die Sollzeit, die Summe der gebuchten Zeiten und die daraus resultierende Differenz werden im Sidebar Widget "Zeitbuchung" angezeigt. Darüber hinaus wird die gebuchte Gesamtzeit in der Ticket-Detailansicht unter "Accounted Time" angezeigt.

Das Setzen der Sollzeit kann auf verschiedenem Wege erfolgen:

1. Manuelles Setzen der Sollzeit über die Aktion "Geplanter Aufwand"
2. Automatisiertes Setzen über Jobs
3. Automatisiertes Setzen unter Verwendung eines Asset-Attributs (siehe unten "Konfigurationsbeispiel")

Die Voraussetzungen und erforderlichen Komponenten für eine einfache Zeiterfassung sind in KIX Pro bereits vorhanden. Sie können die Zeiterfassung direkt nutzen oder individuell konfigurieren.

Für die Zeiterfassung werden folgende Komponenten verwendet:

1. Das **Dynamische Feld** "Planned Effort".

Das Dynamische Feld nimmt die am Ticket gesetzte Sollzeit als Wert auf und ist in KIX Pro initial enthalten. Dieser Wert wird entweder automatisch durch einen Job oder manuell über die Aktion "Planned Effort" gesetzt. Es können positive Zahlenwerte in Minuten eingetragen werden. Die Sollzeit wird in der Sidebar "Zeitbuchung" angezeigt und für die Berechnung der Zeitdifferenz benutzt. Die Konfiguration des Dynamischen Feldes kann nicht geändert werden. Es handelt sich um ein internes Dynamisches Feld.

2. Die **Aktion** "Geplanter Aufwand".

Die Aktion öffnet bei ihrer Verwendung einen Dialog, in welche die Sollzeit für das jeweilige Ticket angegeben werden kann. Sofern nicht anders konfiguriert, steht die Aktion nur den Ticket-Verantwortlichen zur Verfügung. Die Aktion kann individuell nachkonfiguriert werden.

3. **Jobs**

Das Dynamische Feld "Planned Effort" kann automatisiert über einen Job mit der Action "Dynamisches Feld setzen" für eine bestimmte Gruppe von Tickets gefüllt werden. Über die Filter im Job können Sie Bedingungen für die Ausführung des Jobs festlegen, z. B. ein bestimmter Kunde oder Ticket-Typ. Initial werden die Jobs "Auto Set Planned Effort (Incident)" und "Auto Set Planned Effort (Service Request)" ausgeliefert. Für den Tickettyp "Incident" beträgt der Standardwert initial 30 Minuten und für den Tickettyp "Service Request" 60 Minuten. Die Jobs können bei Bedarf individuell nachkonfiguriert werden.

Sie können optional weitere Jobs zum Setzen der Sollzeit anlegen und dabei konfigurieren, unter welchen Voraussetzungen welcher Wert als Sollzeit am Ticket gesetzt wird.

Zudem ist es möglich, die an einem Asset hinterlegte Sollzeit über einen Job auszulesen und ins Ticket zu setzen. Voraussetzung dafür ist, dass ein Eingabefeld für die Sollzeit in die Assetklasse konfiguriert ist.

4. Das **Eingabefeld** "Zeit buchen"

Das Feld wird in Ticket-Formularen und beim Erstellen von Artikeln angeboten. In diesem Feld hinterlegen Agenten die Zeit, die sie am Ticket gearbeitet haben (in Minuten). Das Buchen negativer Werte ist möglich. Die Gesamtheit der gebuchten Zeiten wird initial in der Lane "Ticket-Informationen" angezeigt, sobald Zeiten gebucht sind.

Das Buchen einer Zeit kann auch durch einen Job unter Verwendung der MacroAction "Zeit buchen" erfolgen.

5. Das **Sidebar Widget** "Zeitbuchung".

Im Sidebar Widget wird sowohl die am Ticket geplante Sollzeit als auch die Summe der am Ticket gebuchten Zeiten angezeigt. Die daraus resultierende Zeitdifferenz wird ebenfalls angegeben. Ist die Zeitdifferenz ein negativer Wert, wird die Zeitdifferenz in Rot und mit einem Kreuz versehen dargestellt. Ist das Ergebnis der Zeitdifferenz ein positiver Wert, wird dieser in Grün und mit einem Häkchen versehen dargestellt. Somit ist auf einen Blick erkennbar, wieviel des Planzeitbudgets für eine Ticketbearbeitung noch zur Verfügung steht.

Schema der Zeitbuchung

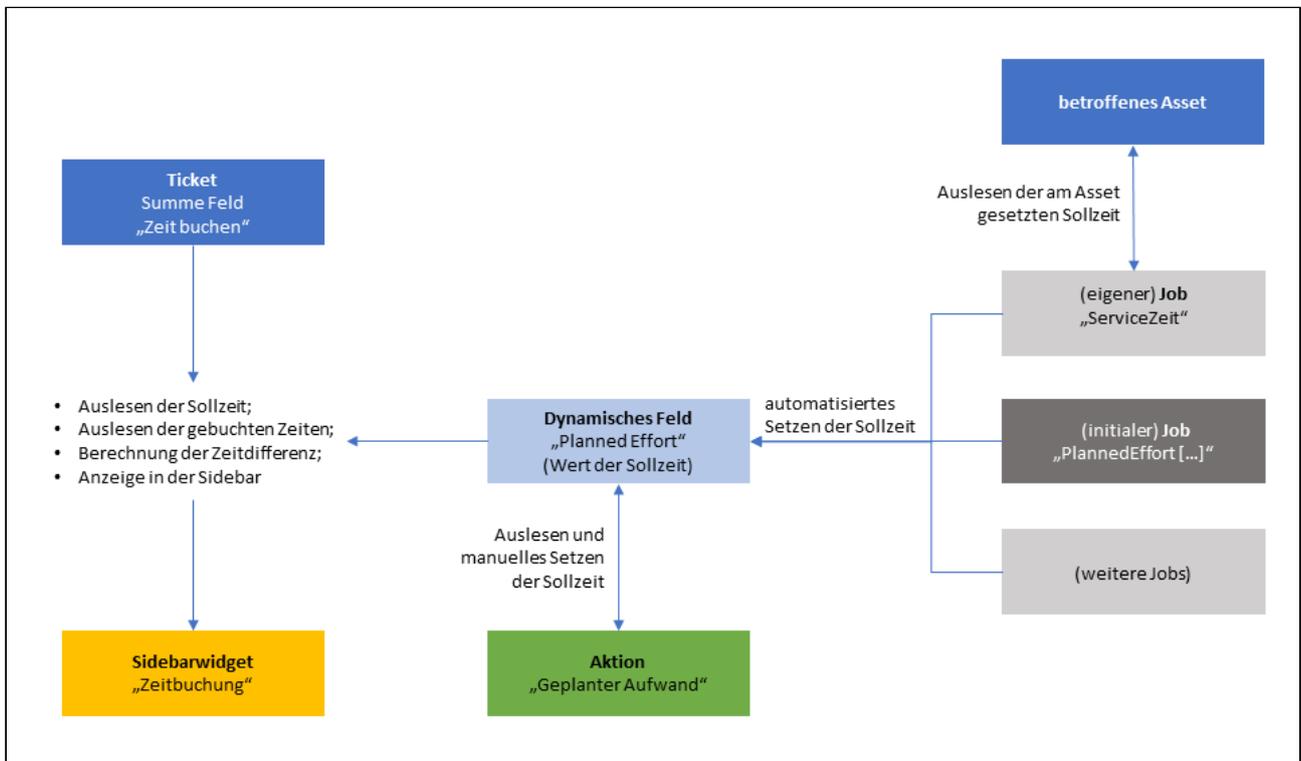


Abb.: Schema der Zeitbuchung

9.2.1 Konfigurationsbeispiel

Dieses Konfigurationsbeispiel zeigt die Verwendung eines Asset-Attributs zum Setzen der Sollzeit. Ziel ist es, dass bei Auswahl eines bestimmten betroffenen Assets automatisch die am Asset hinterlegte Sollzeit am Ticket gesetzt wird. Am Beispiel wird

1. die Asset-Klasse Service um das Attribut "Service Zeit" ergänzt.
2. ein Service "Incident ACDC" erstellt und dort eine Service Zeit von 60 Minuten hinterlegt, da mit dem Kunde "Active Connect Data Center (ACDC)" ein entsprechender Service Vertrag besteht.
3. ein Job konfiguriert, der auf das Erstellen eines Tickets vom Typ "Incident" in Verbindung mit dem Kunde ACDC reagiert. Der Job setzt den am Asset-Attribut "ServiceTime" hinterlegten Wert in das Dynamische Feld "Planned Effort" am Ticket.
4. ein Ticket erstellt, an welchem die Zeitvorgabe des betroffenen Assets genutzt wird.

1. Asset-Klasse "Service" um das Attribut "Service Zeit" ergänzen:

1. Navigieren Sie im Admin Modul zu Assets > Asset Klassen.
2. Öffnen Sie die Klasse "Service" und klicken Sie auf "Bearbeiten". Es wird ein Dialog geöffnet, in dem Sie die Klassendefinition ändern können.
3. Ziehen Sie das Textfeld "Klassendefinition" etwas größer, um den Quellcode besser bearbeiten zu können. Alternativ können Sie auch in einen JSON-Editor (<https://www.jsonformatter.io>) verwenden.

- Fügen Sie folgenden Codeblock an gewünschter Stelle in den Quellcode ein:

```
{
  'CountDefault' => 1,
  'CountMax' => 1,
  'CountMin' => 1,
  'CustomerVisible' => 0,
  'Input' => {
    'Regex' => '^\\d+$',
    'RegexErrorMessage'=> 'Geben Sie eine positive ganze Zahl ein.',
    'Type' => 'Text'
  },
  'Key' => 'ServiceTime',
  'Name' => 'Service Zeit',
  'Searchable' => 1
},
```

- Klicken Sie auf "Speichern", um die Änderung der Klassendefinition zu übernehmen.
- Der Dialog zum Anlegen/Bearbeiten eines Assets enthält nun das Eingabefeld "Service Zeit". Agenten können darin die geplante Sollzeit hinterlegen.

 **Tipps**

- Das Attribut "Service Zeit" kann auch in anderen Asset Klassen ergänzt werden.
- Sie können unter *Internationalisierung > Übersetzungen* ein Pattern für die Attributbezeichnung ("Service Zeit") anlegen. Dadurch wird die Bezeichnung automatisch in die Nutzersprache übersetzt.

2. Service "Incident ACDC" anlegen:

- Legen Sie ein neues Asset der Klasse "Service" an. Klicken Sie dazu auf die Schaltfläche "+Neu". Alternativ können Sie auch im Explorer des Asset Moduls die Klasse "Service" auswählen und dort auf die Schaltfläche "Neues Asset" anklicken.
- Wählen Sie im Tab "Assets" die Klasse "Service".
- Geben Sie dem Asset einen aussagekräftigen Namen (im Beispiel: "Incident ACDC").
- Tragen Sie Verwendungs- und Vorfallstatus ein und optional weitere Parameter.
- Tragen Sie im Feld "Service Zeit" den Wert der Sollzeit in Minuten ein (im Beispiel: 60). Das Feld "Service Zeit" ist das in der Asset Klasse ergänzte Feld.
- Speichern Sie das neu angelegte Asset.

3. Job "AssetServiceZeit" anlegen:

1. Navigieren Sie im Admin Modul zu *Automatisierung > Jobs* und klicken Sie auf "neuer Job".
2. Verwenden Sie im Job folgende Konfiguration:
 - Job Informationen
 - Job Typ: Ticket
 - Name: bspw. "AssetServiceZeit"
 - Gültigkeit: gültig
 - Ausführungsplan
 - Event: TicketDynamicFieldUpdate_ServiceTime, TicketCreate
 - Filter
 - Kunde > enthalten in > Active Connect Data Center (= Kunde ACDC)
 - Aktionen
 - 1. Aktion: Dynamisches Feld setzen
 - Dynamic Field Name: PlannedEffort (= Name des Dynamischen Feldes, welches die Sollzeit am Ticket setzt)
 - Dynamic Field Value: 60 (Zeit in Minuten).
3. Speichern Sie den Job.

Hinweis: Der initial ausgelieferte Job "Auto Set Planned Effort (Incident)" muss "ungültig" gesetzt werden.

4. Ticket erstellen

1. Legen Sie ein neues Ticket an. Klicken Sie dazu auf die Schaltfläche "+Neu".
2. Hinterlegen Sie im sich öffnenden Dialog die erforderlichen Angaben zum Ticket sowie die folgenden Angaben:
 1. Typ: Vorfall
 2. Betroffenes Asset: Incident ACDC.
3. Nach dem Speichern des Tickets ist in der Sidebar "Zeitbuchung" 60 Minuten als "geplanter Aufwand" angegeben.



9.3 KIX Pro REST API

Nachfolgend bieten wir Ihnen die Dokumentation zur REST API von KIX Pro zum Download an.

Im Dokument werden die zusätzlichen Ressourcen und API-Änderungen beschrieben, die durch die Installation von KIX Pro verfügbar sind. Die bestehende KIX-API wird lediglich erweitert, sodass alle in der KIX-API-Dokumentation beschriebenen Grundlagen weiterhin gültig sind und entsprechend verwendet werden müssen.



Hinweis: Datei nach dem Öffnen bitte herunterladen.



9.4 Kind-Tickets in den Ticketdetails auflisten

Konfigurationsschlüssel	ticket-details-parent-child-widget
--------------------------------	------------------------------------

Die Lane "Ticket Informationen" von KIX Pro enthält in den Ticketdetails einen initial deaktivierten Tab zur Anzeige der am Ticket hinterlegten Kind-Tickets inkl. erweiterter Informationen wie Tickettyp, Status und Bearbeiter. Bei Bedarf können Sie diesen Tab im SysConfig-Schlüssel "*ticket-details-parent-child-widget*" aktivieren/deaktivieren (s. unten).

Im Tab werden alle Tickets aufgelistet, welche im [Dynamischen Feld "Kind Tickets"](#) (siehe Seite 46) am Ticket gespeichert sind. Das Dynamische Feld "ChildTickets" wird mit KIX Pro ausgeliefert, ist jedoch initial nicht in die Ticketoberflächen eingebunden. Um Kind-Tickets am Ticket zu speichern, muss es in eine [Ticketvorlage](#) (siehe Seite 151) (z. B. "Default - New Ticket Template") oder [Ticketaktion](#) (siehe Seite 102) (z. B. "Ticket Edit") integriert werden.

Die am Ticket gespeicherten Kind-Tickets werden mit dem aktuellen Ticket als Kind-Ticket verknüpft. Das aktuelle Ticket wird somit zu deren Eltern-Ticket. Die Kind-Tickets werden zudem im Tab "Verknüpfte Objekte" und im Sidebar-Widget "Teilaufgaben" aufgelistet - jedoch mit anderen bzw. reduzierten Informationen.

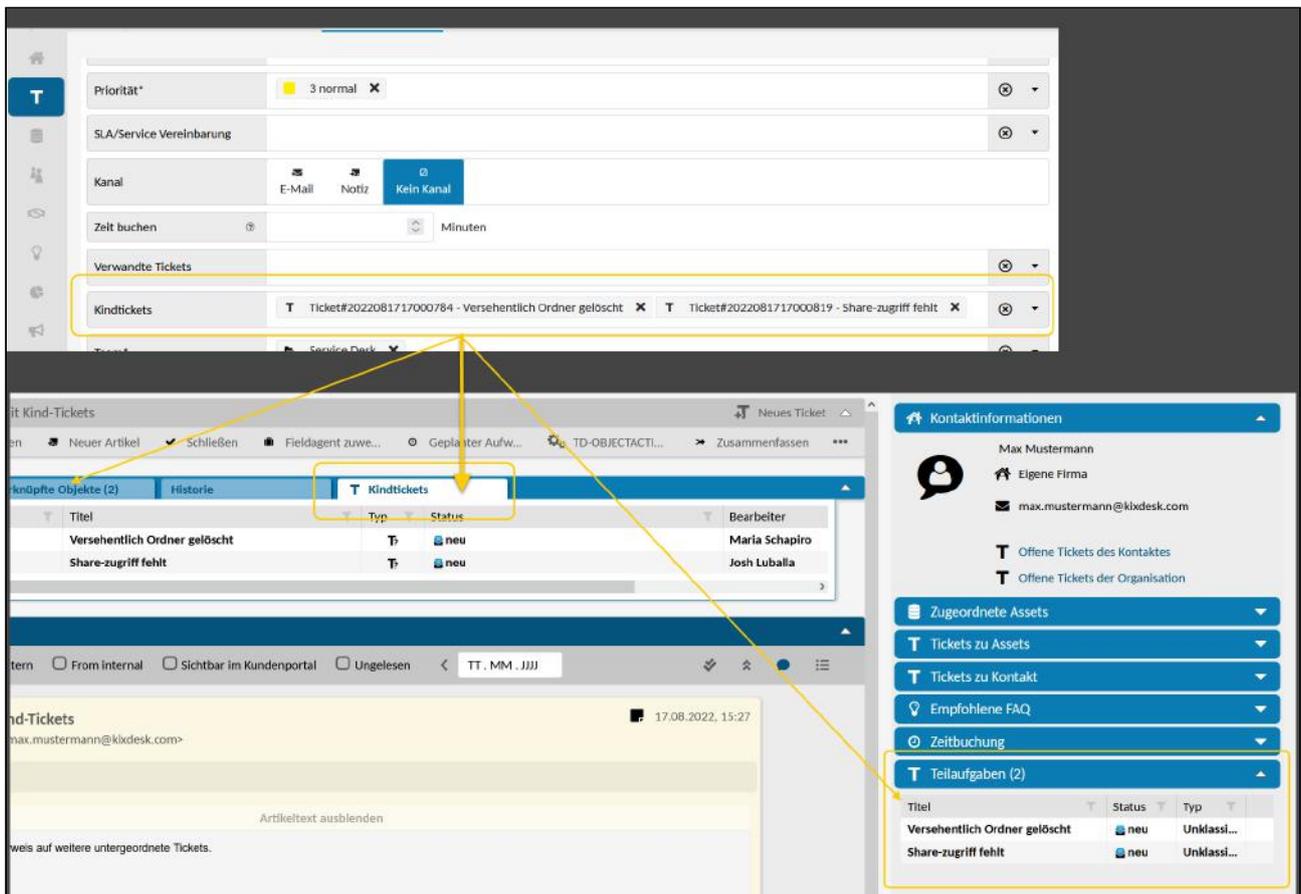


Abb.: Am Ticket hinterlegte Kind-Tickets in den Ticketdetails

9.4.1 Zum Anzeigen von Kind-Tickets im Tab sind folgende Schritte nötig:

1. Dynamisches Feld "ChildTickets" ins Ticket integrieren, damit Agenten Kind-Tickets am Ticket hinterlegen können (z. B. in den Dialog "neues Ticket" und/oder in den Dialog "Ticket bearbeiten").
2. SysConfig-Schlüssel "*ticket-details-parent-child-widget*" aktivieren, damit der initial ausgeblendete Tab "Kindtickets" eingeblendet wird.

9.4.2 Dynamisches Feld "ChildTickets" in den Dialog "neues Ticket" integrieren

Gehen Sie bitte wie folgt vor:

1. Navigieren Sie im Admin Modul von KIX Pro ins Menü *Workflow > Vorlagen*.
2. Öffnen Sie die Vorlage "Default - New Ticket Template" zur Bearbeitung.
Diese Vorlage generiert die Eingabemaske zum Anlegen eines neuen Tickets.
3. Wechseln Sie zum Schritt 2 - "Eingabefelder".
4. Wählen Sie im leeren Dropdown am unteren Ende der Seite das Dynamische Feld "ChildTickets (Kindtickets)" sowie die Option "Im Formular setzen" aus.
Positionieren Sie das Feld optional an einer anderen Stelle im Formular (via Drag & Drop).

5. Speichern Sie die Vorlage.

Danach ist das Dynamische Feld "Kind Tickets" im Dialog "neues Ticket" enthalten, sodass Agenten Kind-Tickets auswählen können.

Alternativ können Sie eine eigene [Vorlage](#) (siehe Seite 151) anlegen und dort das Dynamische Feld einbinden. Dann können Agenten die Kind-Tickets am Ticket speichern, sobald sie diese Vorlage nutzen.

9.4.3 Dynamisches Feld "ChildTickets" in den Dialog "Ticket bearbeiten" integrieren

Gehen Sie bitte wie folgt vor:

1. Navigieren Sie im Admin Modul von KIX Pro ins Menü *Workflow > Aktionen*.
2. Öffnen Sie die Aktion "Ticket Edit" zur Bearbeitung.
Diese Aktion generiert die Eingabemaske zum Bearbeiten eines Tickets.
3. Wechseln Sie zum Schritt 4 - "Eingabefelder".
4. Wählen Sie im leeren Dropdown am unteren Ende der Seite das Dynamische Feld "ChildTickets (Kindtickets)" sowie die Option "Im Formular setzen" aus.
Positionieren Sie das Feld optional an einer anderen Stelle im Formular (via Drag & Drop).
5. Speichern Sie die Aktion.

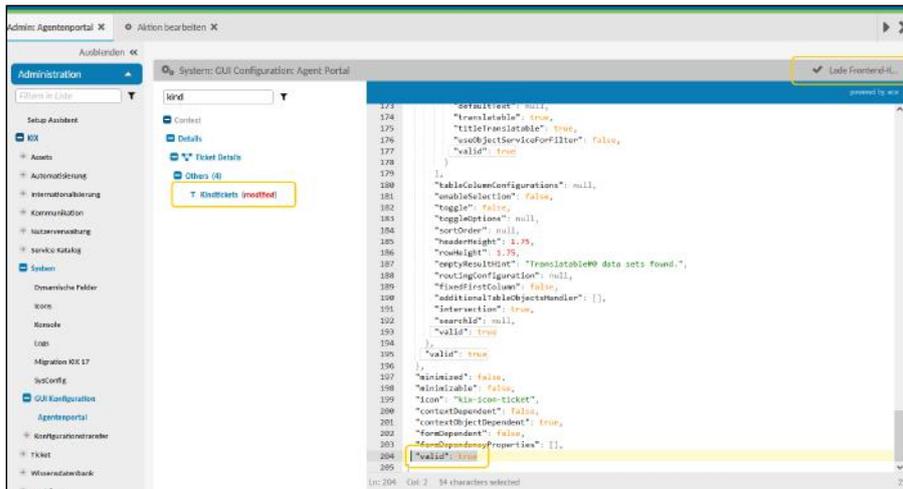
Danach ist das Dynamische Feld "Kind Tickets" im Dialog "Ticket bearbeiten" enthalten, sodass Agenten Kind-Tickets auswählen können.

Alternativ können Sie eine eigene [Ticketaktion](#) (siehe Seite 102) anlegen und dort das Dynamische Feld einbinden. Dann können Agenten die Kind-Tickets am Ticket speichern, sobald sie diese Ticketaktion nutzen.

9.4.4 SysConfig-Schlüssel aktivieren

Gehen Sie bitte wie folgt vor:

1. Navigieren Sie im Admin Modul von KIX Pro ins Menü *System > GUI Konfiguration > Agentenportal*.
2. Suchen Sie nach dem SysConfig-Schlüssel "*ticket-details-parent-child-widget*" (Kindtickets).
Geben Sie dazu bspw. "Kind" ins Suchfeld ein.
3. Scrollen Sie im Editor nach unten bis zum Ende des JSON-Strings.
4. Setzen Sie den letzten Parameter `valid` auf `true`.
Das aktiviert die Anzeige des Tabs "Kindtickets".



5. Speichern Sie die Änderung und klicken Sie abschließend auf "Lade Frontend-Konfigurationen neu".

Danach ist der Tab "Kindtickets" in den Ticketdetails enthalten, sodass die am Ticket gespeicherten "Kind-Tickets" darin aufgeführt werden.

Zum Deaktivieren der Anzeige, setzen Sie den Parameter wieder auf `false` (= Default).

9.5 Objekt-Historie konfigurieren

Konfigurationsschlüssel	ObjectHistoryTypes###Customer ObjectHistoryTypes###Organisation
--------------------------------	--

Die Detailansicht von Kontakten und Organisationen enthält den Tab "Historie". Hier ist ersichtlich, wer welche Änderungen am Kontakt bzw. an der Organisation vorgenommen hat. Protokolliert werden:

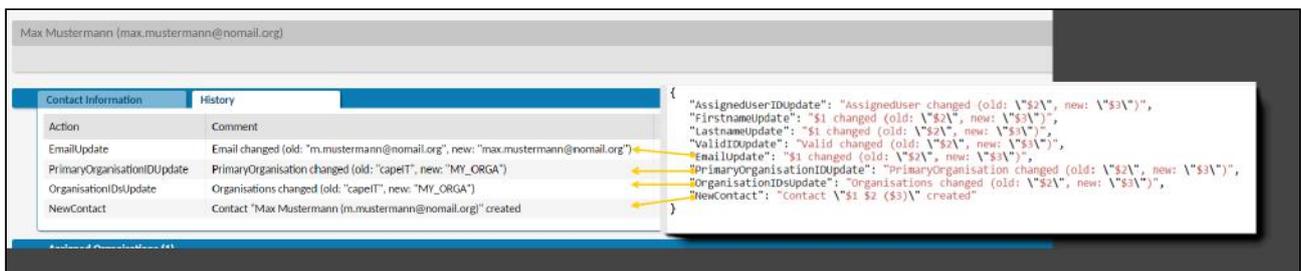
- der Zeitpunkt der Änderung
- der Nutzer, welcher die Änderung vorgenommen hat
- die vorgenommene Änderung.

Wird eine Änderung vorgenommen, speichert KIX die am Objekt geänderten Parameter in Variablen und erstellt daraus einen String, welcher in der Historie in aufbereiteter Form als Kommentar ausgegeben wird: "\$1 geändert (alt: \"\$2\", neu: \"\$3\")".

Variable	Wert
\$1	Attribut, welches geändert wurde
\$2	alter Wert
\$3	neuer Wert

Diesen String können Sie nach Bedarf ändern, bspw. um Updates an einem Dynamischen Feld spezifisch auszugeben. Dies erfolgt im Menü "System > SysConfig" in den o. a. Konfigurationsschlüsseln.

Es muss kein Eintrag für die einzelnen Attribut-Updates vorhanden sein. Es wird dann der Standard (Default) verwendet: <WHAT> changed (old: "<OLDVALUE>", new: "<NEWVALUE>":



The screenshot shows a user interface for a contact named 'Max Mustermann'. It features a 'History' tab with a table of actions and their comments. A configuration window is overlaid on the right, showing a JSON object where each key corresponds to an action in the history table, and the value is a string template using variables \$1, \$2, and \$3 to represent the attribute, old value, and new value respectively.

Abb.: Konfiguration der Objekthistorie

Bitte beachten:

Wenn Sie innerhalb des Strings doppelte Hochkommas verwenden, müssen diese mit vorangehendem Backslash maskiert werden (z. B. "old: \"\$2\" "). Alternativ können Sie einfache Hochkommas verwenden (z. B. "old: '\$2' ")

Soll für ausgewählte Nutzer die Historie nicht angezeigt werden, können Sie dies über die Berechtigungen der entsprechenden Rollen definieren. z. B. Rolle "Customer Reader":

Resource	/contacts/*/history	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Kommentar	-
Resource	/organisations	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Kommentar	-
Resource	/contacts	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Kommentar	-

Abb.: Rolle hat Leserechte auf Kontakte, aber keine Berechtigung auf die Kontakt-Historie

9.6 Konfiguration des Standortexplorers

Konfigurationschlüssel	CMDBL location
-------------------------------	----------------

Agenten mit den Rollen CMDB-Reader bzw. CMDB-Maintainer finden im Modul "Standorte" eine nach Standorten und Assetklassen gruppierte Übersicht der Assets.

Inhalte auf dieser Seite:

- [Voraussetzung](#) (siehe Seite 275)
 - [Assetklasse um Attribut "Standort" erweitern](#) (siehe Seite 276)
- [Konfiguration](#) (siehe Seite 277)
 - [Explorer](#) (siehe Seite 277)
 - [Contentbereich](#) (siehe Seite 277)
 - [Sidebar](#) (siehe Seite 282)

Die Anzeige der in einem Standort, Gebäude oder Raum enthaltenen Assets beachtet explizite, direkte Zuordnungen am Asset. Die Aufstellung der Informationen erfolgt nach den im System hinterlegten Standorten und ihrer Struktur. Dabei wird auf die Vorgabeklassen und -hierarchie "Standort > Gebäude > Raum" gesetzt. Bedarf können Sie die Anzeige individuell konfigurieren.

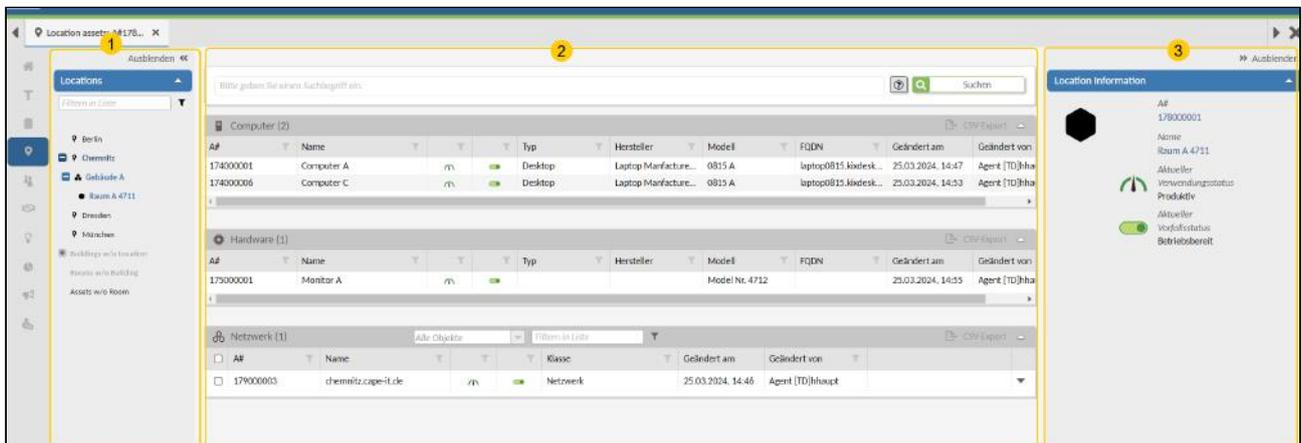


Abb.: Konfigurierte Übersicht der Assets pro Standort

9.6.1 Voraussetzung

Die Übersicht zeigt nur Assets, die einem Standort (Standort, Gebäude, Raum) zugeordnet sind. Nicht alle Assetklassen verfügen initial über das Attribut "Standort" (z. B. Klasse "Netzwerk"). Sollen auch diese Assetklassen im Standortexplorer angezeigt werden, muss das Eingabefeld "Standort" in die betreffende Assetklasse integriert werden.

9.6.1.1 Assetklasse um Attribut "Standort" erweitern

- Navigieren Sie im Admin-Modul zu "Assets > Asset-Klassen".
- Doppelklicken Sie auf die betreffende Assetklasse, um deren Detailansicht zu öffnen.
- Klicken Sie auf "Bearbeiten".
- Fügen Sie den nachfolgenden Codeabschnitt an gewünschter Position in die Klassendefinition ein.
 - Optional: Ändern Sie den Verknüpfungstyp und die Verknüpfungs-Richtung
- Speichern Sie die Änderung. Dabei wird eine neue Version angelegt.

Danach kann den Assets dieser Klasse ein Standort zugewiesen werden.

```
{
  'CountDefault' => 1,
  'CountMax' => 1,
  'CountMin' => 1,
  'CustomerVisible' => 0,
  'Input' => {
    'ReferencedCIClassLinkDirection' => 'Reverse',
    'ReferencedCIClassLinkType' => 'Includes',
    'ReferencedCIClassName' => [
      'Location',
      'Building',
      'Room'
    ],
    'Type' => 'CIClassReference'
  },
  'Key' => 'Location',
  'Name' => 'Location',
  'Searchable' => 1
},
```

Weiterführende Informationen:

- Die Klassendefinition
- Eine Klassendefinition bearbeiten
- Attribute einer Klassendefinition
- Attributtypen in der Klassendefinition

9.6.2 Konfiguration

9.6.2.1 1 Explorer

Der Explorer listet alle im System angelegten Standorte sowie den untergeordneten Gebäuden und Räumen auf. Diese werden im Modul "Assets" angelegt (s. auch KIX 18 Anwenderhandbuch).

Änderungen an der Konfiguration sind in der Regel nicht erforderlich.

9.6.2.2 2 Contentbereich

Der Contentbereich enthält den Gerätebestand für den gewählten Standort nach Assetklassen gruppiert.

Jede Assetklasse wird in einem separaten Tabellenwidget angezeigt. Sie können weitere Widgets hinzufügen (s. Bild oben).

- Öffnen Sie im Menü "System > GUI Konfiguration > Agentenportal" den SysConfig-Schlüssel "CMDBLocation".
- Suchen Sie im Quellcode den Abschnitt "content" und fügen Sie nachfolgenden Codeblock innerhalb dieses Abschnittes ein.
 - Dieser erzeugt ein neues Widget für die Assetklasse "Netzwerk"
 - Platzieren Sie den Codeblock an gewünschter Stelle, z. B. unterhalb des Widgets "Hardware". Die Reihenfolge der Codeblöcke definiert die Reihenfolge in der Ansicht.

```
1  {
2    "instanceId": "kixpro-cmdb-location-network-list-widget",
3    "configurationId": "kixpro-cmdb-location-network-list-
4    widget",
5    "configuration": {
6      "id": "kixpro-cmdb-location-network-list-widget",
7      "name": "Network List",
8      "type": "Widget",
9      "widgetId": "table-widget",
10     "title": "Translatable#Network",
11     "actions": [
12       "bulk-action",
13       "csv-export-action"
14     ],
15     "configuration": {
16       "id": "kixpro-cmdb-location-network-table-widget",
17       "name": "Network Widget",
18       "type": "TableWidget",
19       "objectType": "ConfigItem.9",
20       "sort": null,
21       "subConfigurationDefinition": null,
```

```
21     "headerComponents": null,  
22     "showFilter": true,  
23     "shortTable": null,  
24     "predefinedTableFilters": [  
25         {  
26             "name": "Expired",  
27             "criteria": [  
28                 {  
29                     "property": "CurDeplStateID",  
30                     "operator": "EQ",  
31                     "value": 11,  
32                     "useObjectService": false,  
33                     "useDisplayValue": false  
34                 }  
35             ]  
36         },  
37         {  
38             "name": "Maintenance",  
39             "criteria": [  
40                 {  
41                     "property": "CurDeplStateID",  
42                     "operator": "EQ",  
43                     "value": 13,  
44                     "useObjectService": false,  
45                     "useDisplayValue": false  
46                 }  
47             ]  
48         },  
49         {  
50             "name": "Pilot",  
51             "criteria": [  
52                 {  
53                     "property": "CurDeplStateID",  
54                     "operator": "EQ",  
55                     "value": 14,  
56                     "useObjectService": false,  
57                     "useDisplayValue": false  
58                 }  
59             ]  
60         },  
61         {  
62             "name": "Planned",  
63             "criteria": [  
64                 {  
65                     "property": "CurDeplStateID",  
66                     "operator": "EQ",  
67                     "value": 15,  
68                     "useObjectService": false,  
69                     "useDisplayValue": false  
70                 }  
71             ]  
72         },  
73         {
```

```
74         "name": "Production",
75         "criteria": [
76             {
77                 "property": "CurDeplStateID",
78                 "operator": "EQ",
79                 "value": 16,
80                 "useObjectService": false,
81                 "useDisplayValue": false
82             }
83         ]
84     },
85     {
86         "name": "Repair",
87         "criteria": [
88             {
89                 "property": "CurDeplStateID",
90                 "operator": "EQ",
91                 "value": 17,
92                 "useObjectService": false,
93                 "useDisplayValue": false
94             }
95         ]
96     },
97     {
98         "name": "Review",
99         "criteria": [
100             {
101                 "property": "CurDeplStateID",
102                 "operator": "EQ",
103                 "value": 19,
104                 "useObjectService": false,
105                 "useDisplayValue": false
106             }
107         ]
108     },
109     {
110         "name": "Test/QA",
111         "criteria": [
112             {
113                 "property": "CurDeplStateID",
114                 "operator": "EQ",
115                 "value": 20,
116                 "useObjectService": false,
117                 "useDisplayValue": false
118             }
119         ]
120     }
121 ],
122 "cache": true,
123 "resetFilterOnReload": true,
124 "configuration": null,
125 "valid": true,
126 "application": "agent-portal"
```

```

127     },
128     "minimized": false,
129     "minimizable": false,
130     "icon": "kix-icon-network",
131     "contextDependent": true,
132     "contextObjectDependent": false,
133     "formDependent": false,
134     "formDependencyProperties": [],
135     "valid": true,
136     "application": "agent-portal"
137   },
138   "permissions": [],
139   "size": "large"
140 }

```

- Soll anstatt "Netzwerk" eine andere Assetklasse angezeigt werden, ändern Sie die Parameter `instanceId`, `configurationId`, `name` und `title` (gelbe Markierungen)
 - Ersetzen Sie " `network` " durch den Namen der anzuzeigenden Assetklasse.
 - Anstelle von " `kixpro-cmdb-location-myasset-list-widget` " können Sie auch einen anderen, aber eindeutigen, Bezeichner verwenden.

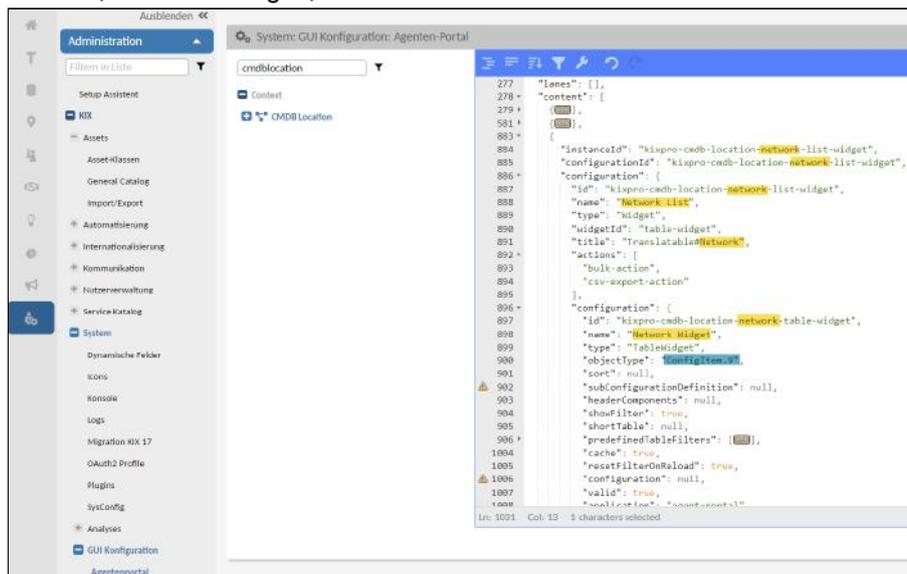
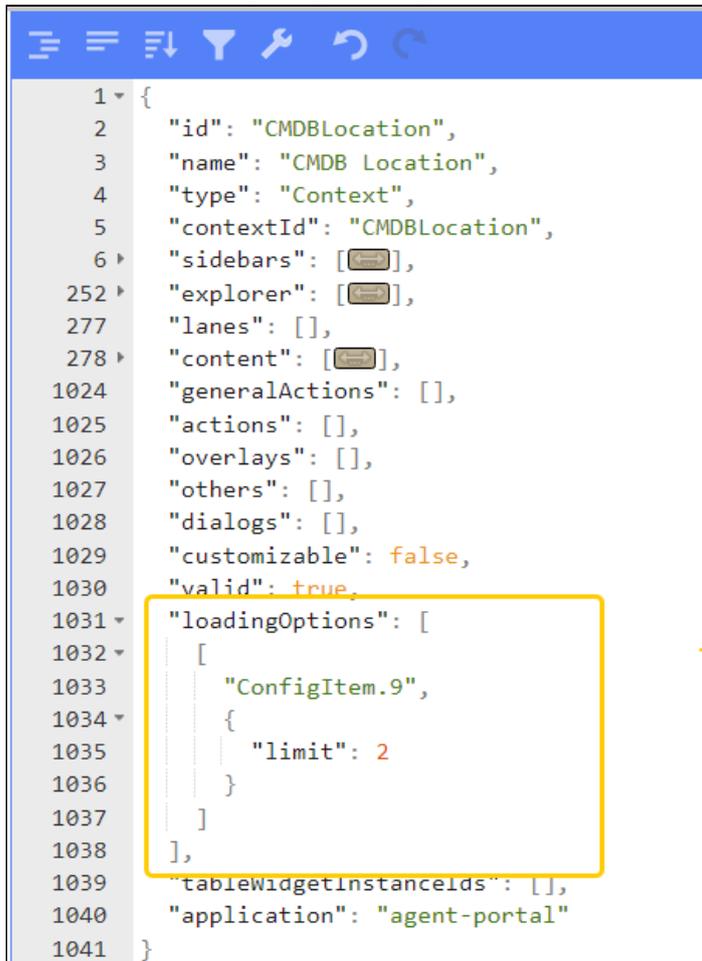


Abb.: Zusätzliches Widget "Netzwerk" im Standortexplorer (Ansicht im externen Editor)

- Ändern Sie ebenfalls den " `objectType` " (blaue Markierung).
 - Geben Sie die ID Assetklasse an, deren Assets im Widget angezeigt werden sollen.
 Tipp: Navigieren Sie im Admin Modul zu "KIX > Assets > Asset-Klassen". Führen Sie die Maus über die betreffende Asset-Klasse und lesen Sie die ID in der Fußzeile des Browsers ab.
Alternativ: Klicken Sie die betreffende Asset-Klasse im Explorer des Asset-Moduls an und lesen Sie die ID in der Adresszeile des Browsers ab.
 - Leerzeichen und Unterstriche sind möglich.

- Die Assetklasse muss mit vorangestelltem "ConfigItem_" angegeben werden (z. B. "ConfigItem_Network oder "ConfigItem_Software").
- Optional können Sie die "loadingOptions" des Contextes ändern (limit, searchLimit, sortOrder usw.)

Als Kenner ist der ObjectType anzugeben, der im Klassen-Widget verwendet wird.



```

1 {
2   "id": "CMDBLocation",
3   "name": "CMDB Location",
4   "type": "Context",
5   "contextId": "CMDBLocation",
6   "sidebars": [{}],
252  "explorer": [{}],
277  "lanes": [],
278  "content": [{}],
1024  "generalActions": [],
1025  "actions": [],
1026  "overlays": [],
1027  "others": [],
1028  "dialogs": [],
1029  "customizable": false,
1030  "valid": true,
1031  "loadingOptions": [
1032    [
1033      "ConfigItem.9",
1034      {
1035        "limit": 2
1036      }
1037    ]
1038  ],
1039  "tableWidgetInstanceIds": [],
1040  "application": "agent-portal"
1041 }
  
```

- Speichern Sie Ihre Änderungen am SysConfig-Schlüssel.
- Klicken Sie auf "Lade Frontendkonfiguration neu", um Ihre Änderungen in der Programmoberfläche anzuzeigen.
Danach ist das zusätzliche Widget im Standortexplorer enthalten und die der Klasse zugeordneten Assets werden darin angezeigt (s. oben: Voraussetzungen).

Weiterführende Infos:

- Externer JSON-Editor, z. B: <https://www.jsonformatter.io/>

- Eine Asset-Klasse anlegen oder bearbeiten
- Grundlagen der GUI-Konfiguration
- Die Konfiguration von Widgets
- Das Table-widget
- Konfigurationsbeispiele GUI-Konfiguration
- Dem Dashboard ein Widget hinzufügen

9.6.2.3 3 Sidebar

Die rechte Sidebar ist ein Object-information-card-widget, dessen anzuzeigende Inhalte Sie individuell konfigurieren können:

- Öffnen Sie im Menü "System > GUI Konfiguration > Agentenportal" den SysConfig-Schlüssel "CMDBLocation".
- Suchen Sie im Quellcode innerhalb des Abschnitts "sidebars" den Abschnitt "rows". Im Abschnitt "rows" werden die im Widget anzuzeigenden Inhalte pro Zeile definiert.
- Definieren Sie unter "componentData" die anzuzeigenden Inhalte und optional unter "conditions" die Bedingungen für die Anzeige.

Beispiel

```
{
  "componentId": "object-avatar-label",
  "componentData": {
    "property": "AnzuzeigenderInhalt"
  },
  "conditions": [
    {
      "property": "<KIX_ASSET_Class>",
      "operator": "NE",
      "value": null,
      "useObjectService": false,
      "useDisplayValue": false
    }
  ],
  "routingConfiguration": {
    "contextId": "config-item-details",
    "objectType": "ConfigItem",
    "contextMode": "DETAILS",
    "objectIdProperty": "ConfigItemID",
    "history": false,
    "resetContext": true,
    "params": null,
  }
}
```



```
"additionalInformation": [],  
  "contextType": "MAIN"  
},  
  "routingObjectId": "<KIX_ASSET_ConfigItemID>"  
},
```

- Speichern Sie Ihre Änderungen.
- Klicken Sie auf "Lade Frontendkonfiguration neu", um Ihre Änderungen in der Programmoberfläche anzuzeigen.

Weiterführende Informationen:

- Das Object-information-card-widget
- Ein Object-information-card-widget konfigurieren
- Die Konfiguration der Sidebar

10 Add-ons

Für KIX Pro können Sie Zusatzmodule erwerben, mit denen Sie die Grundfunktionen von KIX erweitern können. Die Zusatzmodule ermöglichen u. a. den bidirektionalen Datenaustausch mit anderen Informationssystemen wie bspw. Baramundi.

Produktlinie: CONNECT

[Connect Baramundi](#) (siehe Seite 305)

[Connect Database](#) (siehe Seite 311)

[Connect Opsi](#) (siehe Seite 343)

[Connect Webservice](#) (siehe Seite 347)

Produktlinie: ITIL Practices

[Add-on "ITIL Practices"](#) (siehe Seite 372)

[Umsetzung der ITIL Practices mittels Prozessen](#) (siehe Seite 376)

KIX Wartungsplan

[Wartungsplan](#) (siehe Seite 408) (Admin)

[Wartungsplan](#) (Anwender)

10.1 Connect

Unter der Produktlinie Connect bieten wir Ihnen einzeln zu erwerbende Add-ons für KIX Pro an. Die Add-ons ermöglichen Ihnen den Datenaustausch mit Fremdsystemen und die Nutzung externer

Inhalte auf dieser Seite:

- [Installation](#) (siehe Seite 286)
- [Gemeinsamer Funktionsumfang der Connect Add-ons](#) (siehe Seite 286)
- [Feldtyp "Data Source"](#) (siehe Seite 287)
 - [Parameter](#) (siehe Seite 288)
- [Erweiterte Macro Actions](#) (siehe Seite 289)
 - [XSL Transformation](#) (siehe Seite 289)
 - [Parameter](#) (siehe Seite 292)
 - [XSLT-Funktionen](#) (siehe Seite 294)
 - [XSLT-Besonderheiten in KIX](#) (siehe Seite 299)
 - [Get Object Data](#) (siehe Seite 301)
 - [Parameter](#) (siehe Seite 302)
 - [Get Item List From Data Source](#) (siehe Seite 302)
 - [Parameter](#) (siehe Seite 303)
 - [Get Item From Data Source](#) (siehe Seite 304)
 - [Parameter](#) (siehe Seite 304)

Datenbankverbindungen als Datenquelle zur weiteren Verwendung in KIX.

Typische Anwendungsfälle für Connect sind bspw.

- Import von Organisations-, Kontakt- oder Assetdaten aus CRM- oder ERP-Systemen wie Navision, Axapta, Sugar CRM etc.
- Anzeige von Kundendaten, Controlling Informationen aus CRM- oder ERP-Systemen wie Navision, Axapta, Sugar CRM etc.
- Import von Assetdaten aus Endpoint-Management-/Inventarisierungs-/Discovery-Systemen wie Baramundi, i-doit, opsi, etc.
- Anzeige von Warenbeständen zu Artikelnummern ("Warehouse Integration")
- Übernahme von Ticketdaten aus anderen Systemen
- Bereitstellung von Auswahlwerten in Formularfeldern an Tickets, Organisationen, Kontakten oder FAQ-Einträgen aus CRM- oder ERP-Systemen wie Navision, Axapta, Sugar CRM etc.

Den Agenten stehen somit die aktuellen Daten aus anderen Datenbankquellen für die Nutzung in KIX zur Verfügung. Die Daten werden bspw.

- in Dynamischen Feldern an Tickets, Kontakten, Organisationen oder FAQ-Einträgen referenziert
- in Detailansichten oder Dashboards kontextbezogen dargestellt
- als Grundlage für Datenimporte in das KIX-Asset-Management (CMDB) verwendet

- zur Erstellung von Tickets verwendet.

10.1.1 Installation

Nach der Beantragung von Connect stellen wir Ihnen - i.d.R. am nächsten Arbeitstag - ein aktualisiertes Image Ihres Systems bereit. Führen Sie ein System-Update durch, wie im Kapitel [Installation von KIX Pro](#) (siehe Seite 10) beschrieben. Danach ist Connect in Ihr System integriert. Je nach konkretem Erweiterungspaket ist noch eine Aktivierung im Admin-Bereich Ihres KIX erforderlich.

10.1.2 Gemeinsamer Funktionsumfang der Connect Add-ons

Die Add-ons der Connect-Linie greifen auf teilweise gemeinsame Kernfunktionen zurück die nachfolgend beschrieben sind.

- **Feldtyp "Data Source"**
 - Dynamische Felder des Typs "Data Source" ermöglichen, Inhalte aus externen Datenquellen an allen Objekten zu hinterlegen, welche dynamische Felder unterstützen.
 - Löst die in KIX Pro Version 17 existierenden Feldtypen "RemoteDB" und "Invoker" ab.
 - wird nur in [Connect Database](#) (siehe Seite 311) und [Connect Webservice](#) (siehe Seite 347) genutzt
- **Erweiterte Macro Actions**

Macro Action	Beschreibung	Hinweise
Get Item From Datasource	Führt eine Detailanfrage an eine Connect-Datenquelle durch.	Parameter und Details s. Macro Action "Get Item From Data Source" Anwendungsbeispiel: Connect Database (siehe Seite 311) (Verwendung in Macro Actions)
Get Item List From Datasource	Führt eine Listenanfrage an eine Connect-Datenquelle durch.	Parameter und Details s. Macro Action "Get Item List From Data Source" Anwendungsbeispiel: Connect Database (siehe Seite 285)

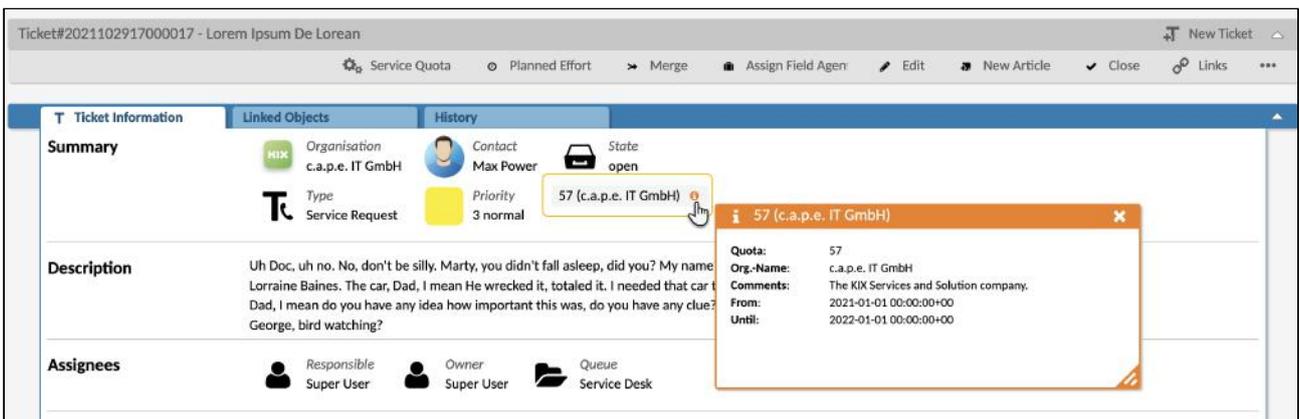
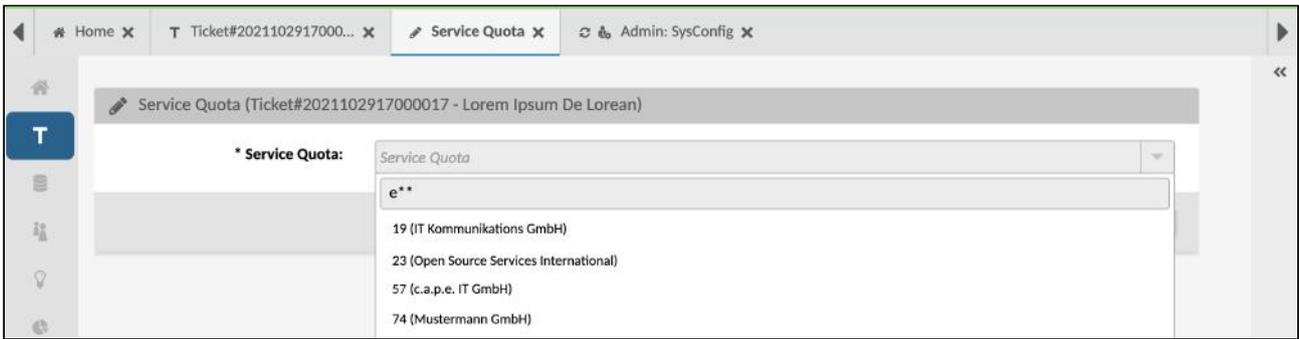
Macro Action	Beschreibung	Hinweise
Get Object Data	Ermittelt den bedarfsgerechten Inhalt von KIX Business Objekten. Z. B. Tickets mit Artikeln und Anhängen, Assets mit Versionsdaten, Organisations- und Kontaktdaten	Parameter und Details s. Macro Action "Get Object Data" Anwendungsbeispiel: Connect Webservice (siehe Seite 347) (Beispiel Redmine Issue)
XSL Transform	Vor- und Aufbereitung von Daten für andere Macro Actions. Erlaubt die Erstellung komplexer Datenstrukturen.	Parameter und Details s. Macro Action "XSL Transform" Anwendungsbeispiel: Connect Webservice (siehe Seite 347) (Beispiel Redmine Issue)

⚠ Hinweis

Einige Macro Actions der Connect-Linie enthalten eine Debug-Option. Wenn Sie das Debugging aktivieren, muss "Automation::MinimumLogLevel" auf "Debug" gesetzt sein. Anderenfalls werden nur die von der Macro Action gelieferten Fehlermeldungen erfasst und in der Job Historie angezeigt, alle anderen Debug-Informationen jedoch nicht.

10.1.3 Feldtyp "Data Source"

Mit Dynamischen Feldern vom Typ "Data Source" können Formularfelder eingebunden werden, deren Wertebereich in Connect-Datenquellen definiert ist. Die Funktionsweise entspricht der des Feldtyps "Asset Reference". So können Einzel- und Mehrfachwerte in einem Formularfeld hinterlegt und die Informationen in einem Overlay eingeblendet werden. Die Bereitstellung und Verwendung dieser Dynamischen Felder in Vorlagen und Aktionen für das Self Service Portal ist ebenfalls möglich - beachten Sie jedoch die hinterlegten Zugriffsbeschränkungen (Rollenzuordnung) an der dem jeweiligen Feld zugeordneten Datenquelle. Die gewünschten Kundennutzerrollen z.B. "Customer", benötigen dann ebenfalls Zugriffsberechtigung auf die Datenquelle.



10.1.3.1 Parameter

Zusätzlich zu den allgemeinen Parametern wird das Verhalten eines Felds vom Typ "Data Source" durch folgende Parameter definiert:

Parameter	Beschreibung	Beispiel
CacheTTL	Definiert, für wieviel Sekunden ein einmal an der Datenquelle angefragter Datensatz im KIX-Cache gehalten wird, bevor eine neue Anfrage an die Datenquelle gesendet wird.	3600
Data Source	Die anzufragende Datenquelle	"Servicequotas (DB)"

Parameter	Beschreibung	Beispiel
Default Display Column	<p>Array von Spalte-Anzeige-Paaren. Definiert den Inhalt des Overlays bei der Darstellung des Dynamischen Feldes in den Detailansichten.</p> <ul style="list-style-type: none"> • Anzeige: Definiert die Beschriftung des Einzelwerts. • Spalte: Definiert, welches Attribut der Detailanfrage an der Datenquelle (bei DB Alias/DB-Spaltenname) für den Anzeigewert genutzt wird. 	<p>Spalte: quota - Anzeige: Quota Spalte: cname - Anzeige: Org.-name Spalte: startdate - Anzeige: From Spalte: enddate - Anzeige: Until</p>
Display Pattern	<p>Definiert, welche Attribute der Listenanfrage für die Darstellung eines Einzeleintrags (View-Mode und Auswahlliste) genutzt werden. Die Attributnamen (DB-Spaltennamen bzw. -Aliase) werden mittels "<" und ">" gekennzeichnet.</p>	<p><quota> (<cname>)</p>

10.1.4 Erweiterte Macro Actions

10.1.4.1 XSL Transformation

XSL Transformation (kurz XSLT) ist eine Sprache zur Transformation von XML-Dokumenten. Diese Macro Action erlaubt die Umwandlung einer eingehenden Datenstruktur in eine andere Datenstruktur. Somit können Wertübersetzungen, Wertextraktionen und Strukturveränderungen vorgenommen werden. KIX verwendet XSLT in Version 1.1.

Zweck der Macro Action ist es, Daten für die Verwendung in anderen Macro Actions vor- oder aufzubereiten, z. B. für "[Webhook Extended](#) (siehe Seite 347)". Sie kann das Ergebnis von XSL-Transformationen als Inhalt im Request verwenden. Gleichmaßen können Ergebnisse von Webanfragen (Responses) für die weitere Verwendung in anderen Macro Actions vorbereitet werden, z. B. für "CreateOrUpdateAsset". Beim Einsatz der XSL-Transformation stehen KIX-eigene Funktionen bereit, um die Möglichkeiten von XSLT zu erweitern.

Die Macro Action übernimmt eine als Macro Variable oder Zeichenkette definierte Datenstruktur, wandelt diese in XML um und wendet darauf die XSL-Transformation auf Basis eines XSLT-Templates an. Das Ergebnis dieser Umwandlung steht dann zur weiteren Verwendung zur Verfügung.



Praktische Anwendungsfälle sind bspw.:

- die Hinterlegung von Fremdsystem-IDs zu Tickets nach deren Übermittlung (Redmine, Confluence, Jira, etc.)
- die Aufbereitung von Asset-Informationen zur Ablage in der KIX-CMDB (Inventarisierung)
- oder die Fehlerbehandlung von Webhook Extended-Aufrufen.

Ein Beispiel zur Erstellung eines Redmine Issues aus einem KIX-Ticket finden Sie unter [Connect Webservice](#) (siehe Seite 347) (Beispiel 2).

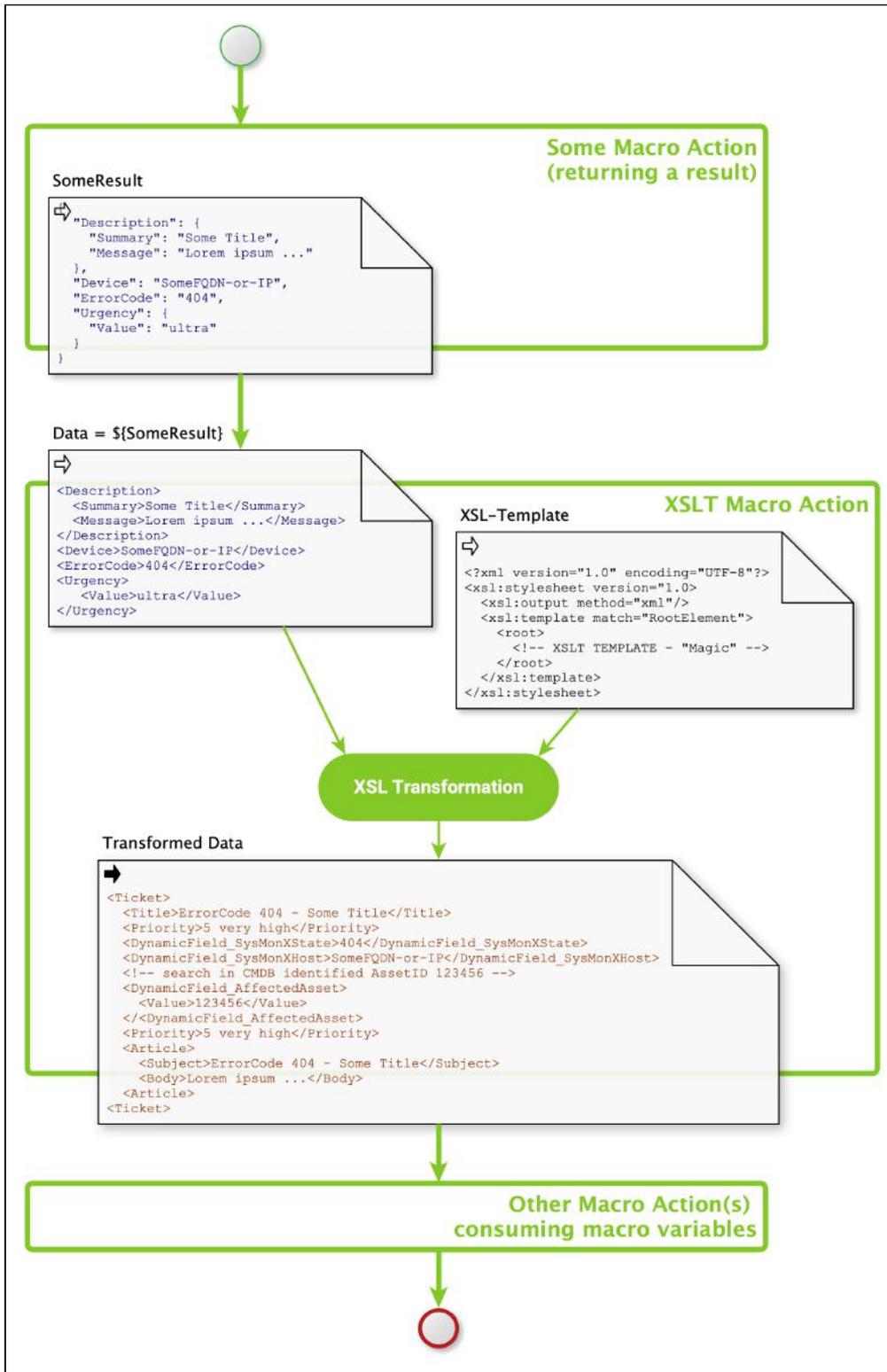


Abb: Schema einer XSL Transformation

Parameter

Parameter	Beschreibung	Beispiel
Data	<p>Das Ausgangsobjekt oder der JSON-String, der transformiert werden soll.</p> <p>Wird ein JSON-String angegeben, werden KIX-Platzhalter unterstützt.</p>	<pre> \${SomeMacroVariable} oder JSON-String, z.B.: { "issue": { "subject": "<KIX_TICKET_Title>", "description": "Lorem ipsum..." } } </pre>
Debug	<p>Debug-Ausgaben de-/aktivieren.</p> <p>Wenn aktiviert, werden die von der Macro Action gelieferten Debug-Informationen in der Job Historie angezeigt.</p> <p>Voraussetzung: Der SysConfig-Schlüssel "<i>Automation::MinimumLogLevel</i>" muss auf den Wert <code>"Debug"</code> gesetzt sein. Anderenfalls werden nur die von der Macro Action gelieferten Fehlermeldungen erfasst, alle anderen Debug-Informationen jedoch nicht.</p>	yes no

Parameter	Beschreibung	Beispiel
<p>XSL Template</p>	<p>Das XSL Template für die vorzunehmende Transformation.</p> <p>Das Ergebnis einer Transformation muss zur weiteren Verwendung im root-Pfad des Dokuments enthalten sein.</p> <p>Das gezeigte Beispiel beinhaltet eine Auswertung des HTTP-Statuscodes und Aufbereitung des HTTP-Response-Inhalts. In Verbindung mit Macro Action "condition" kann eine Fehlerbehandlung eingerichtet werden.</p>	<pre><?xml version="1.0" encoding="UTF-8"?> <xsl:stylesheet version="1.0" xmlns:xsl="http:// www.w3.org/1999/ XSL/Transform" xmlns:KIX="urn:KIX"> <xsl:output method="xml" encoding="utf-8" indent="yes"/> <xsl:template match="RootElement"> <root> <xsl:variable name="HTTPCode" select="HTTPCode"/> <xsl:choose> <xsl:when test="\$HTTPCode='201' "> <RMIssueID><xsl:value- of select="Result/issue/ id"/></RMIssueID> </xsl:when> <xsl:otherwise> <RMIssueID><xsl:text>N ot transfered to Redmine.</xsl:text></ RMIssueID> </xsl:otherwise> </xsl:choose> </root> </xsl:template> </xsl:stylesheet></pre>

Parameter	Beschreibung	Beispiel
XSL Transformed Data	Variablenname für die Ergebnisstruktur der Umwandlung	RMRequestPrepared

XSLT-Funktionen

KIX stellt einige spezifische Funktionen zur Auflösung von Identifikatoren (Ermittlung von Contact-, Organisation-, Asset-IDs aus Namen, Login-Kennern oder Nummern von Objekten) sowie für ergänzende Möglichkeiten in XSLT-Mappings bereit. Der Aufruf der Aktionen bei Anwendung eines XSL-Templates führt zur Ersetzung des Funktionsaufrufs mit dem Ergebnis der Funktion.

Funktion	Beschreibung	mögliche Werte	Rückgabewerte
AccessTokenLookup(String)	Liefert einen OAuth2-Access-Token für das per Parameter benannte OAuth2 Profil.	'Office365'	String
AssetGetIDByName (String)	ID des Assets basierend auf Assetname	'ddsrv009'	String
AssetGetIDByClassAndAttributeValue (String, String, String, String)	ID des Assets basierend auf Assetklasse und einem beliebigen klassenspez. Attribut ('AssetClass', 'AttributeKey', 'SearchValue', 'JoinSeparator', 'SplitSeparator')	'Computer', 'FQDN', 'chnb001.example.com ⁶ , , , , ,'	String
AssetGetIDByClassAndName (String, String)	ID des Assets basierend auf Assetklasse und -name ('AssetClass', 'AssetName')	'Computer', 'chnb001'	String
AssetClassGetIDByName (String)	ID der Assetklasse	'Computer'	String

⁶ <http://chnb001.example.com>

Funktion	Beschreibung	mögliche Werte	Rückgabewerte
AssetGetIDByNumber (String)	ID des Assets basierend auf Assetnummer	'634000001'	String
Base64Decode (String)	Base64-Decodierung der übergebenen Zeichenkette	'U29tZUJhc2U2NFN0cmlyZw=='	String
Base64Encode (String)	Base64-Codierung der übergebenen Daten/ Zeichenkette	'SomeDataString'	String
ConfigGet (String)	Wert des durch "String" bezeichneten SysConfig-Eintrags vom Typ "String"	'SysConfigKey'	String
ContactGetIDByEmail (String)	ID des Kontakts mit der gegebenen Email	'max@example.com ⁷ '	String
ContactGetIDByUserLogin (String)	ID des Kontakts mit dem angegebenen User-Login (nur wenn es einen Nutzer gibt)	'contact075'	String
DeploymentStateGetIDByName (String)	ID des Deployment Status	'Production'	String
GeneralCatalogGetIDByClassAndName (String,String)	ID des General Catalog Eintrags zur GC-Klasse und Anzeigewert ('GC-Class', 'GC-Name')	'ITSM::ConfigItem::Computer::Type', 'Desktop'	String

⁷ <http://example.com>

Funktion	Beschreibung	mögliche Werte	Rückgabewerte
GenerateTOTP(String, String, String, String, String)	<p>Generiert ein "Timebased One Time Password" (TOTP) aus den übergebenen Parametern. Dieses kann via Macro-Variable im Header des WebhookExtended verwendet werden, um TOTP-MFA-Authentifizierungen in Webhook-Aufrufen zu verwenden.</p> <ul style="list-style-type: none"> • Parameter 1: Base32Secret - Das Shared Secret in Base32-Enkodierung • Parameter 2: TimeStep - Angabe für die Gültigkeitsdauer eines Token. Standardwert: 30. Erwartet eine natürliche Zahl. • Parameter 3: Digits - Länge des Tokens. Standardwert: 6. Erlaubt sind: 6, 7 oder 8. • Parameter 4: Algorithm - Verschlüsselungsalgorithmus für den Token. Standardwert: SHA1. Erlaubt sind: 'SHA1', 'SHA256' und 'SHA512' (Großschreibung ist relevant) 	'KRUGKU3FMNZGK5CTORZGS3TH','30','6','SHA1','0'	String

Funktion	Beschreibung	mögliche Werte	Rückgabewerte
	<ul style="list-style-type: none"> Parameter 5: Previous - Erzeugt ein vorheriges/ zukünftigen OTP. Standardwert: 0 		
IncidentStateGetIDByName (String)	ID des Vorfallsstatus	'Operational'	String
MD5Sum (String)	MD5-Summe zu übergebener Zeichenkette	'SomeString'	String
OrganisationGetIDByName (String)	ID der Organisation mit gegebenem Namen	'Harmony Shoal Inc.'	String
OrganisationGetIDByNumber (String)	ID der Organisation mit der gegebenen Nummer	'CN007'	String
PatternMatch (String, String)	<p>Gibt 1 zurück, wenn in der übergebenen Zeichenkette das Muster RegEx enthalten ist.</p> <p>Es werden reguläre Ausdrücke in Perl⁸ verwendet.</p>	'String', 'REGEX'	String
PatternRemove (String, String)	<p>Entfernt aus der übergebenen Zeichenketten den auf RegEx zutreffenden Teil.</p> <p>Es werden reguläre Ausdrücke in Perl⁹ verwendet.</p>	'String', 'REGEX'	String

⁸ <https://perldoc.perl.org/perlre>

⁹ <https://perldoc.perl.org/perlre>

Funktion	Beschreibung	mögliche Werte	Rückgabewerte
PatternReplace (String, String, String)	<p>Ersetzt aus der übergebenen Zeichenketten den auf RegEx zutreffenden Teil.</p> <p>Es werden reguläre Ausdrücke in Perl¹⁰ verwendet.</p>	'String', 'REGEX', 'replacement'	String
SystemDataDelete (String)	Entfernt den vollständigen Eintrag zu einem übergebenen Schlüssel aus KIX, z. B. für Auth-Token	'N2GBearerToken'	String
SystemDataGet (String)	<p>Gibt einen permanent in KIX hinterlegten und durch den Schlüssel identifizierten Wert zurück.</p> <p>Ermöglicht, einen bpsw. in einem früheren "WebhookExtended"-Aufruf erhaltenen Auth-Token in weiteren Aufrufen zu verwenden.</p>	'N2GBearerToken'	String
SystemDataSet (String, String)	<p>Speichert einen Wert zu einem übergebenen Schlüssel permanent in KIX (z. B. Auth-Token), welcher durch eine Nutzer-Passwort-Anfrage generiert wird.</p> <p>Ermöglicht bspw. einen per "WebhookExtended" erhaltenen Auth-Token im System zu speichern.</p>	'N2GBearerToken', '__eyJ0eXAiOiJKV1QiLC...'	String

¹⁰ <https://perldoc.perl.org/perlre>

Funktion	Beschreibung	mögliche Werte	Rückgabewerte
TimeStamp ([String])	Gibt einen Zeitstempel zurück (jetzt+Offset Sekunden) im Format 'YYYY-MM-DD hh:mm:ss' (optional).	'3600'	String
TimeString (String, [String])	Wie "TimeStamp (OffsetInSeconds)", jedoch kann das Format entsprechend strftime ¹¹ definiert werden.	'Format', '3600'	String
UserGetIDByLogin (String)	Ermittelt die ID des Nutzers, der durch den übergeben Loginnamen identifiziert wird.	'mamu'	String
UserGetLoginByID (String)	Ermittelt das Login des Nutzers, der durch die übergeben ID identifiziert wird.	'1'	String

XSLT-Besonderheiten in KIX

Rückgabewerte aus XSL-Funktionen

In der Regel geben XSLT-Funktionen Zeichenketten (Strings) zurück. Diese können direkt innerhalb des XSLT wiederverwendet werden. Werden spezifische XSLT-Funktionen aufgerufen, die komplexe Datenstrukturen zurück geben, kann auf die Einzelelemente der Strukturen zugegriffen werden.

Im nachfolgenden Beispiel wird eine hypothetische Funktion `MyTicketGet()` aufgerufen, die einen Ticket-Hash zurück gibt. Das Beispiel verwendet spezifische Angaben aus dem Hash für die weitere Verarbeitung.

Aufruf im XSLT-Template:

```
<xsl:variable name="Ticket" select="KIX:MyTicketGet(123)"/>
  <Ticket>
    <ContactID><xsl:value-of select="$Ticket//ContactID"/></ContactID>
    <StateName><xsl:value-of select="$Ticket//State"/></StateName>
  </Ticket>
```

¹¹ <https://metacpan.org/pod/POSIX::strftime::GNU>

Ergebnis der JSON-Struktur nach XLS-Transformation:

```
"Ticket": {  
  "ContactID": "1701",  
  "StateName": "open"  
}
```

XML-inkompatible Parameternamen

XSLT ist eine Ausprägung von XML. Daher gelten Beschränkungen hinsichtlich der Zeichen, welche als Namen für Tags verwendet werden können. JSON unterliegt nicht den gleichen Beschränkungen. So sind bspw. in JSON Key-Strings wie "`@name`" möglich. Soll ein Präfix verwendet werden, ist das dem jeweiligen XML-Tag als Attribut "`prefix`" mitzugeben.

So erzeugt bspw.:

```
<id prefix="@">SomeIdentifizierString</id>
```

die JSON-Struktur:

```
"@id": "SomeIdentifizierString", .
```

Typisierung von Ausgehenden Daten

Sollen in ausgehenden Anfragen/Webrequests und dem darin enthaltenen JSON bestimmte Datentypen erzwungen werden, weil bspw. die anzusprechende Schnittstelle eine starke Typisierung fordert (z.B. Boolean oder Number), kann auch dies via XSLT gesteuert werden.

```
<eventSequenceNumber convert="number">
  <xsl:text>1701</xsl:text>
</eventSequenceNumber>
<someBooleanValue convert="boolean">
  <xsl:text>0</xsl:text>
</someBooleanValue>
<someOtherBooleanValue convert="boolean">
  <xsl:text>1</xsl:text>
</someOtherBooleanValue>
```

Folgende JSON-Struktur wird erzeugt:

```
"eventSequenceNumber": 1701,
"someBooleanValue": false,
"someOtherBooleanValue": true,
```

Verweise

- https://de.wikipedia.org/wiki/XSL_Transformation
- https://www.w3schools.com/xml/xsl_intro.asp
- <https://www.tutorialspoint.com/xslt/index.htm>
- <http://www.cheat-sheets.org/saved-copy/XSLT-1.0.pdf>

10.1.4.2 Get Object Data

Die Macro Action "Get Object Data" führt eine KIX-interne Abfrage durch und gibt das ermittelte Objekt mit allen Details zur weiteren Bearbeitung an weitere Automatisierungsschritte zurück. Das kann bspw. in Ticket-Jobs genutzt werden, um alle Ticketdaten - inklusive der zugehörigen Artikel - zu beziehen und zur weiteren Verarbeitung in XSL Transformationen bereit zu stellen. Ein anderer Anwendungsfall ist der Bezug der an einem Ticket referenzierten Daten (Organisation, Kontakt oder Assets). Auf diesem Weg stehen mehr Angaben zur Verfügung, als über die aus Textbausteinen und Antwortvorlagen bekannten Platzhalter möglich sind.

Parameter

Parameter	Beschreibung	Beispiel
Beinhaltet	zu inkludierende Unterobjekte oder aufzulösende IDs - siehe Backend-REST-API Dokumentation	DynamicFields, Priority, Articles, Queue, State, Links
Expands	zu erweiternde Unterobjekte - siehe Backend-REST-API-Dokumentation	Links
Item List	Bezeichner der Macro-Variable für weitere Verarbeitung der Rückgabe	CompanyList
Objekt-ID	ID des zu beziehenden Objekts	123 <code>`\${SomeMacroVariable}`</code>
Object Typ	zu beziehender Businessobjekttyp	Asset Ticket Organisation Contact

10.1.4.3 Get Item List From Data Source

Die Macro Action "Get Item List From Data Source" führt eine Listenanfrage an eine Connect Datenquelle durch. Dabei können freie Suchkriterien und Werte für fixierte Suchattribute angegeben werden. Die Rückgabestruktur ist ein "Array of Hashes", wobei die Attribut-/Spaltennamen bzw. Aliase der Listenanfrage die Schlüssel bilden. Ein komplexes Beispiel ist in der Dokumentation für [Connect Database](#) (siehe [Seite 311](#)) enthalten.

Beispiel Rückgabewert

```
[
  {"ID":11,"cno":"EMA","cname":"EM Automotive GmbH & Co. KG","quota":24},
  {"ID":18,"cno":"HMS","cname":"Hypokrates Medical Services","quota":95}
]
```

Verwendung Rückgabewert

In der Regel wird der Rückgabewert in einer weiteren Macro Action "Loop | Schleife" eingesetzt. Da diese ein Array erwartet, ist keine Aufbereitung des Rückgabewertes erforderlich. Innerhalb dieser Schleife kann auf ID-Werte mittels der Punkt-Notation "``${LoopVariable.ID}``" zugegriffen werden.

Parameter

Parameter	Beschreibung	Beispiel
Data Source	Die anzufragende Datenquelle	"Servicequotas (DB)"
Item List	Bezeichner der Macro-Variable für weitere Verarbeitung der Rückgabe	CompanyList
Parameters	JSON-Struktur der in der Datenquelle anzuwendenden fixierten Parameter	{"Param1": 123, "Param2": "Text Pattern", "Param3": [1,2,3]}
Search	JSON-Struktur der in der Datenquelle anzuwendenden Suchkriterien. Der Aufbau entspricht dem von Filterkriterien in der KIX REST-API	<pre>{ "Item": { "AND": [{ "Field": "id", "Operator": "GT", "Type": "NUMERIC", "Value": "5" }, { "Field": "id", "Operator": "LT", "Type": "NUMERIC", "Value": "9" }] } }</pre>

10.1.4.4 Get Item From Data Source

Die Macro Action "Get Item From Data Source" führt eine Detailanfrage an eine Connect Datenquelle durch. Die Rückgabestruktur ist ein Hash, wobei die Attribut-/Spaltennamen bzw. Aliase der Detailanfrage die Schlüssel bilden. Ein komplexes Beispiel ist in der Dokumentation für [Connect Database](#) (siehe [Seite 311](#)) enthalten.

Beispiel Rückgabewert

```
{
  "ID": 18,
  "cname": "Hypokrates Medical Services",
  "cno": "HMS",
  "quota": 95,
  "comments": "Some generic comment here.",
  "startdate": "2021-01-01 00:00:00+00",
  "enddate": "2022-01-01 00:00:00+00"
}
```

Parameter

Parameter	Beschreibung	Beispiel
Data Source	Die anzufragende Datenquelle	"Servicequotas (DB)"
Item	Bezeichner der Macro-Variable für die weitere Verarbeitung der Rückgabe	CurrCompanyData
Item ID	<p>ID des abzurufenden Datensatzes. Es werden KIX-Platzhalter und Macro Variablen unterstützt.</p> <p>Wird die Macro Action innerhalb einer "Loop" nach einem "Get Item List From Data Source" aufgerufen, kann per Punkt-Notation auf den ID-Wert innerhalb des Hashes der aktuellen Schleifenvariable zugegriffen werden (siehe Spalte Beispiel).</p>	<p>123 <KIX_TICKET_DynamicField_SQID> \$ {SingleValueMacroVariable} \$ {CurrCompany.ID}</p>

10.1.5 Connect Baramundi

Connect Baramundi ist ein Zusatzmodul zu KIX Pro. Es bietet die Möglichkeit, Asset-Daten in KIX auf Basis der Inventarisierungslösung Baramundi bereit zu stellen. Die in einem Baramundi vorhandenen Informationen können so im KIX Service Management System verwendet werden.

KIX stellt dazu eine weitere, separate Assetklasse "Baramundi Device" bereit. Die darin zu importierenden Gerätedaten werden über die Baramundi bConnect Schnittstelle bezogen. Die Synchronisation erfolgt durch den Job "Baramundi - Device Sync". Sowohl Assetklasse als auch Job werden beim Aktivieren der Baramundi-Synchronisation in KIX implementiert.

Inhalte auf dieser Seite:

- [Voraussetzungen](#) (siehe Seite 305)
- [Verwendung](#) (siehe Seite 306)
 - [Baramundi-Synchronisation aktivieren](#) (siehe Seite 307)
 - [Baramundi-Synchronisation zurücksetzen oder ändern](#) (siehe Seite 308)
- [Anpassungen](#) (siehe Seite 309)
 - [De-/Aktivierung Übernahme "LastUser"](#) (siehe Seite 309)
 - [Beschränkung Synchronisierte Baramundi-Geräte](#) (siehe Seite 310)
 - [Weitergehende Anpassungen](#) (siehe Seite 310)

10.1.5.1 Voraussetzungen

Zur Verwendung von Connect Baramundi wird ein HTTP-/S-Zugang zur bConnect-Schnittstelle benötigt. Dazu muss der FQDN und ggf. die Portnummer der bConnect-Schnittstelle angegeben werden. Bei Verwendung von HTTPS dürfen nur korrekte SSL-Zertifikate zum Einsatz kommen. Die Verwendung von ungültigen oder selbst erstellten Zertifikaten kann die Funktion der Schnittstelle negativ beeinflussen.



Weiterhin werden Nutzernamen und Passwörter eines zur Anmeldung an der bConnect-Schnittstelle berechtigten Nutzers benötigt. Zur Verwendung von Nutzernamen und Passwörtern in der Einrichtung/Konfiguration ist aus Nutzernamen und Passwort ein Authentifizierungstoken zu erstellen. Unter Linux können Sie dazu folgende Kommandozeile verwenden (`<YourUserName>` und `<YourPassword>` sind zu ersetzen):

```
SomeLinuxSystem:~# echo -n '<YourUserName>:<YourPassword>' | base64  
PFlvdXJvc2VyTmFtZT46PFlvdXJQYXNzd29yZD4=
```

Unter Windows muss die Zeichenkette "`<YourUserName>:<YourPassword>`" zunächst in eine Datei (hier `UserNamePassword.txt`) geschrieben werden. Diese wird dann Base64-codiert. Für die Anzeige des Authentifizierungstokens entfernen Sie aus dieser Datei alle Zeilen, welche "CERTIFICATE" enthalten. Die folgenden zwei Kommandozeilen bilden das ab:

```
C:\temp> certutil -encode -f UserNamePassword.txt UserNamePasswordB64.txt  
C:\temp> type UserNamePasswordB64.txt | find /v "CERTIFICATE"  
PFlvdXJvc2VyTmFtZT46PFlvdXJQYXNzd29yZD4=
```

10.1.5.2 Verwendung

Die Konfiguration und Aktivierung des Plugins erfolgt nach Installation (KIX On Premise: Neustart des KIX Stacks mittels "start.sh"; KIX Cloud: nach Rückmeldung durch Support) im Admin-Bereich im KIXConnect-Baramundi-Setup:

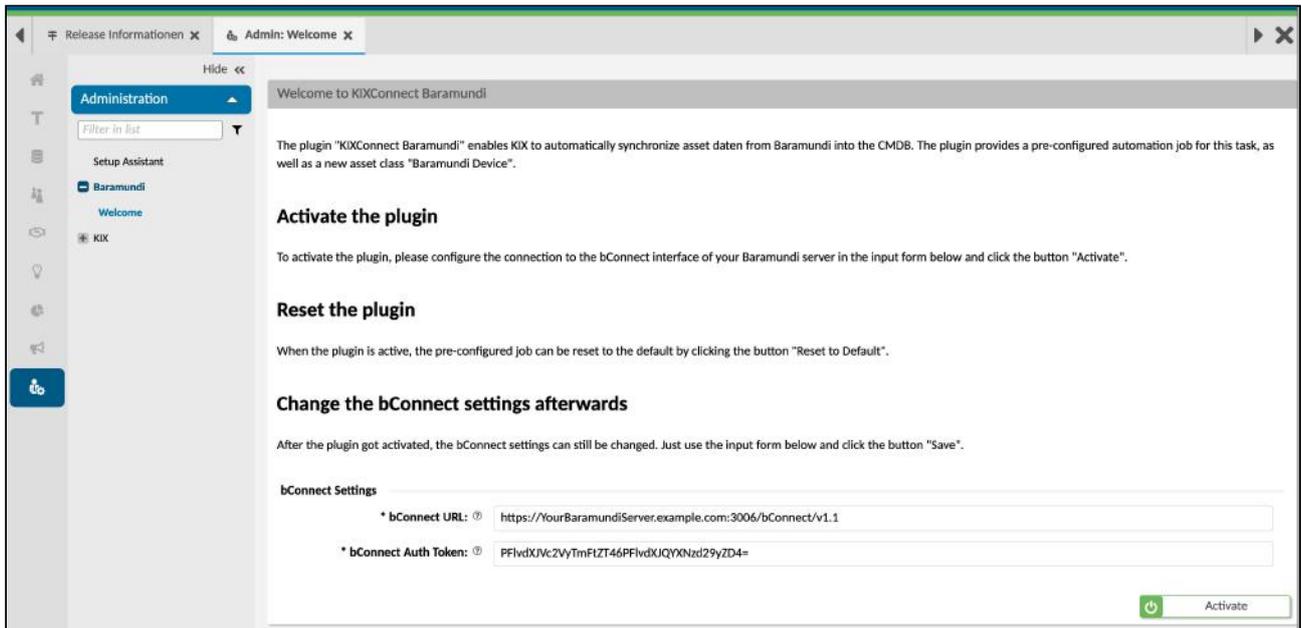


Abb.: Baramundi Setup

Baramundi-Synchronisation aktivieren

Zum Aktivieren des Zusatzmoduls hinterlegen Sie im Admin Modul unter *Baramundi* > *Willkommen* die bConnect-URL (z. B. <https://YourBaramundiServer.example.com:3006/bConnect/v1.1>) sowie den zuvor aufbereiteten Authentifizierungs-Token (s. Voraussetzungen). Klicken Sie anschließend auf "Aktivieren". Die Konfiguration wird gespeichert und legt den Automatisierungsjob "Baramundi - Device Sync" an.

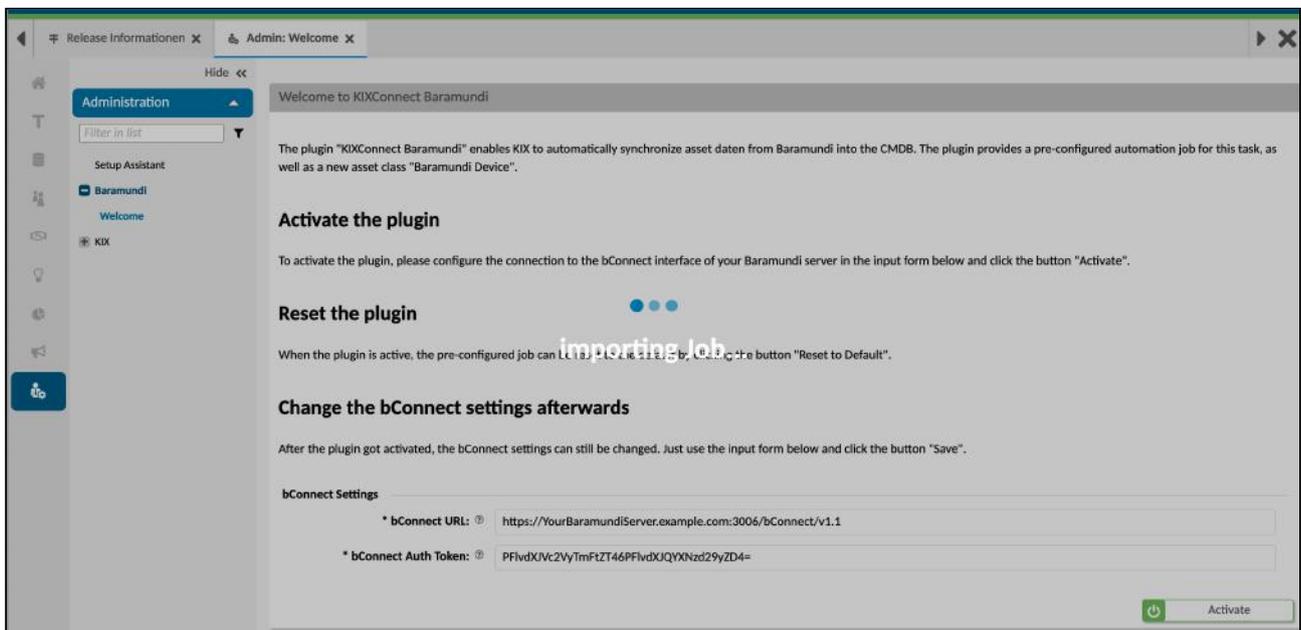


Abb.: Baramundi aktivieren

Für einen ersten Baramundi-Datenimport empfehlen wir Ihnen, anschließend den Job "Baramundi - Device Sync" manuell auszuführen (Admin Modul unter *KIX > Automatisierung > Jobs*). Dies kann je nach Inhalt der Baramundi-Datenquelle einige Minuten in Anspruch nehmen. Soll der Baramundi-Datenimport künftig periodisch ausgeführt werden, ordnen Sie dem Job unter "Ausführungsplan" einen entsprechenden Zeitplan zu, z.B. jeden Tag 03:00 Uhr (s. Einen Job anlegen oder bearbeiten). Verändern Sie nicht die weitere Job-Konfiguration! Danach ist die Konfiguration abgeschlossen.

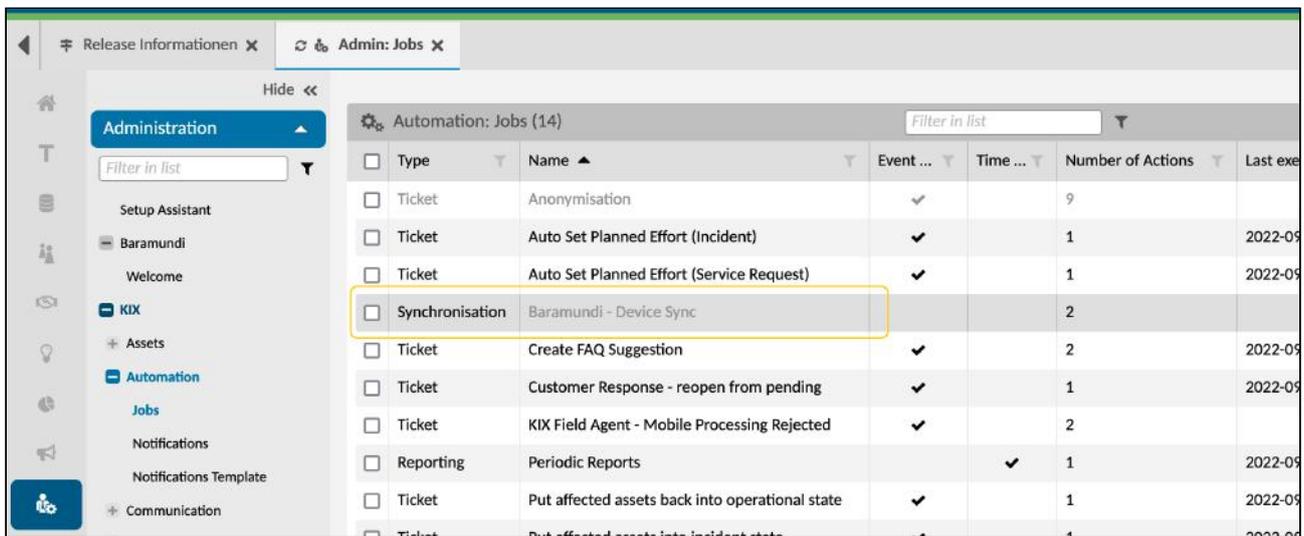


Abb.: Der Baramundi-Synchronisations-Job

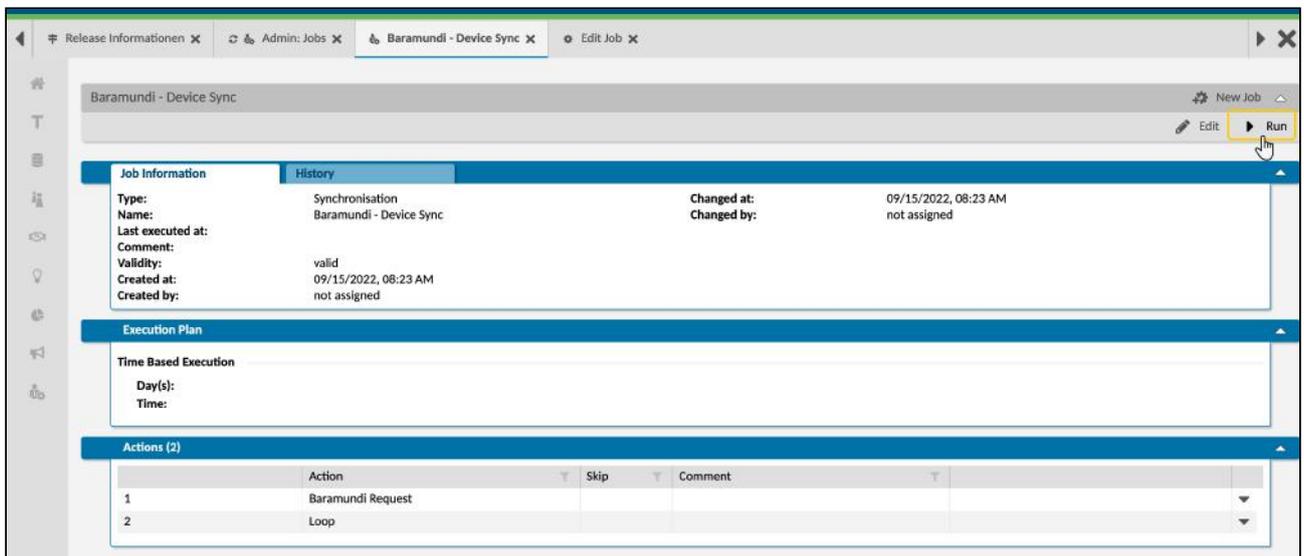


Abb.: Manuelles Starten des Jobs "Baramundi - Device Sync"

Baramundi- Synchronisation zurücksetzen oder ändern

Sollten Sie Anpassungen in der Konfiguration des Jobs vorgenommen haben, um beispielsweise spezifische, erweiterte Angaben zu synchronisieren, können Sie die Vorgabekonfiguration jederzeit mittels des Buttons

"Auf Standard zurücksetzen" wieder auf den Auslieferungszustand zurück setzen. Die Anpassungen werden dabei entfernt.

Nach Aktivierung der Synchronisation, können Sie die Einstellungen zur bConnect-Schnittstelle weiterhin ändern, um beispielsweise eine geänderte URL oder einen neuen Authentifizierungstoken zu hinterlegen. Nehmen Sie dazu die erforderlichen Änderungen im Baramundi-Setup vor und klicken Sie auf "Speichern". Die Änderungen sind bei der nächsten Job-Ausführung aktiv.

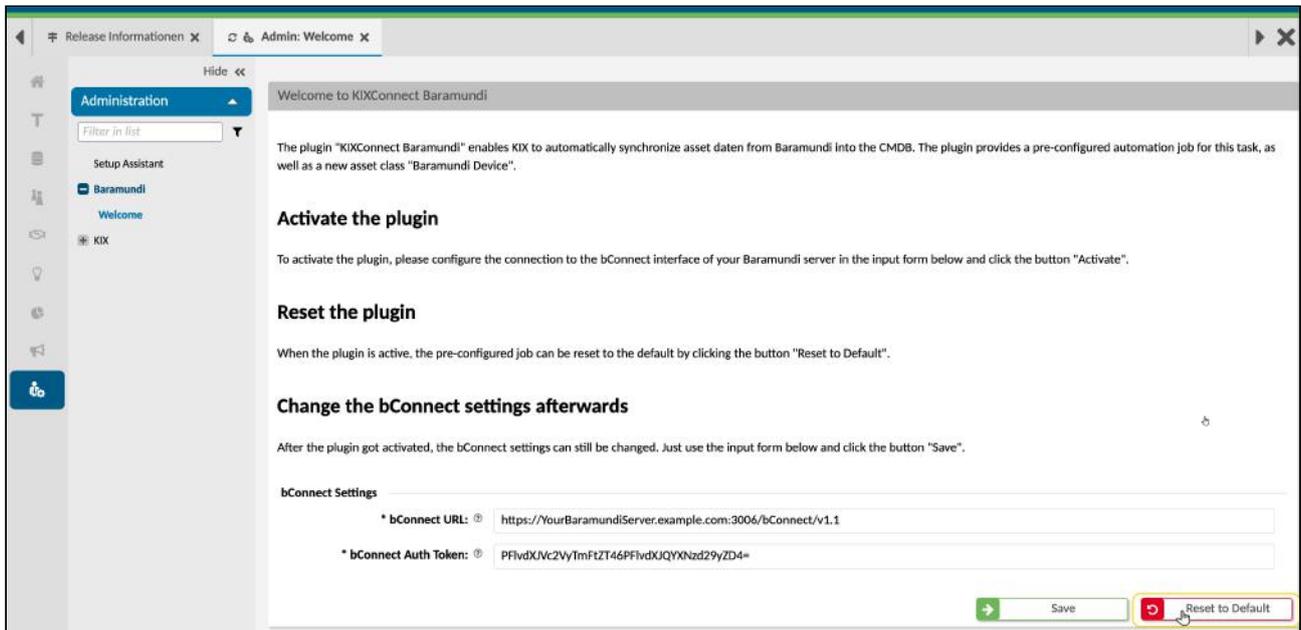


Abb.: Zurücksetzen der Konfiguration

10.1.5.3 Anpassungen

De-/Aktivierung Übernahme "LastUser"

Die Assetklasse "Baramundi Device" und der Datenabgleich importieren neben dem primären Nutzer auch den letzten Nutzer eines Gerätes. Sollte sich dieser sehr häufig ändern, kann es sinnvoll sein, auf den Abgleich zu verzichten. In diesem Fall passen Sie die Konfiguration in KIX wie folgt an:

1. Entfernen Sie den betreffenden Abschnitt aus der Asset-Klassendefinition

Zu entfernender Abschnitt LastUser (Assetklasse)

```

{
  'Key'      => 'LastUserContact',
  'Name'     => 'Last User Contact',
  'Searchable' => '1',
  'Input'    => {

```

```

        'Type'      => 'Contact',
    },
},
{
    'Key'          => 'LastUser',
    'Name'         => 'Last User',
    'Searchable'  => '1',
    'Input'       => {
        'Type'     => 'Text',
    },
},
},

```

- Entfernen Sie den betreffenden Abschnitt aus der Datenaufbereitung im Synchronisations-Job "Baramundi - Device Sync" (die in der Aktion 2 "Schleife" enthaltene Aktion 1 "XSL Transformation")

Zu entfernender Abschnitt LastUser (Datenaufbereitung)

```

<xsl:if test="string-length($LastUserContactID) > 0">
  <LastUserContact>
    <xsl:value-of select="$LastUserContactID"/>
  </LastUserContact>
</xsl:if>

<xsl:if test="string-length($LastUserString) > 0">
  <LastUser>
    <xsl:value-of select="$LastUserString"/>
  </LastUser>
</xsl:if>

```

Beschränkung Synchronisierte Baramundi-Geräte

Um die aus Baramundi abzugleichenden Geräte einzugrenzen, können Sie das Baramundi-Feature "Dynamische Gruppen" nutzen. Definieren Sie eine entsprechende Gruppe in Baramundi, ordnen Sie dieser Gruppe die Geräte zu und ermitteln Sie die "ID" der Gruppe. Soweit uns bekannt, ist dies nur mittels Anfrage an der bConnect-Schnittstelle möglich. Soll die Abfrage in KIX eingegrenzt werden, ergänzen Sie im Job "Baramundi - Device Sync" die abzufragende Ressource "/Endpoints" um den Parameter "DynamicGroup" (Beispiel: "/Endpoints?DynamicGroup=BE7592D3-8E57-41DE-929A-FDA1D993C836")

Weitergehende Anpassungen

Bei weiteren Anpassungswünschen zu den zu verwendenden Daten in der Assetklasse "(Baramundi-) Device" oder einer separaten Betrachtung in verschiedenen Assetklassen, kontaktieren Sie bitte den KIX-Support.

10.1.6 Connect Database

Das Add-on Connect Database für KIX Pro ermöglicht die Nutzung von Datenbankverbindungen als Datenquelle zur weiteren Verwendung in KIX. Für den konkreten Einsatz bedarf es einer entsprechenden Konfiguration. Unterstützt werden die Datenbank-Managementsysteme: PostgreSQL, MariaDB/MySQL oder MS SQL via ODBC.

Inhalte auf dieser Seite:

- [Voraussetzungen](#) (siehe Seite 311)
- [Einrichtung](#) (siehe Seite 311)
 - [1. Datenbankverbindungen einrichten](#) (siehe Seite 312)
 - [Parameter](#) (siehe Seite 313)
 - [2. Datenbank-Datenquellen einrichten](#) (siehe Seite 315)
 - [Schritt 1 - Kopfdaten](#) (siehe Seite 316)
 - [Schritt 2 - Datenbankabfragen](#) (siehe Seite 318)
- [Verwendung von Datenquellen](#) (siehe Seite 321)
 - [In Dynamischen Feldern](#) (siehe Seite 321)
 - [In Sidebar-Tabellen](#) (siehe Seite 325)
 - [In Macro Actions](#) (siehe Seite 330)
- [Verwendung der KIX Datenbank](#) (siehe Seite 342)

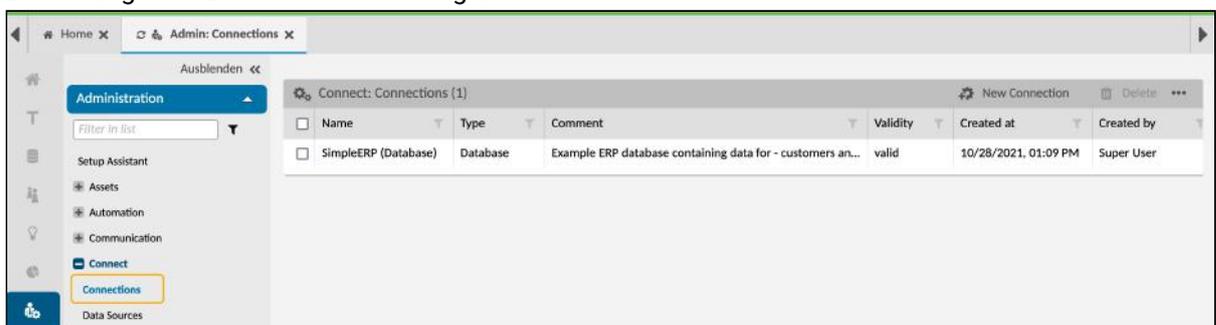
10.1.6.1 Voraussetzungen

Die Anbindung einer Datenbank-Datenquelle erfordert den direkten Zugriff auf die relevante Datenbank mittels Authentifizierung via Nutzernamen und Passwort. Das betreffende DBMS muss dabei via Datenbank-Socket-Verbindung erreichbar sein.

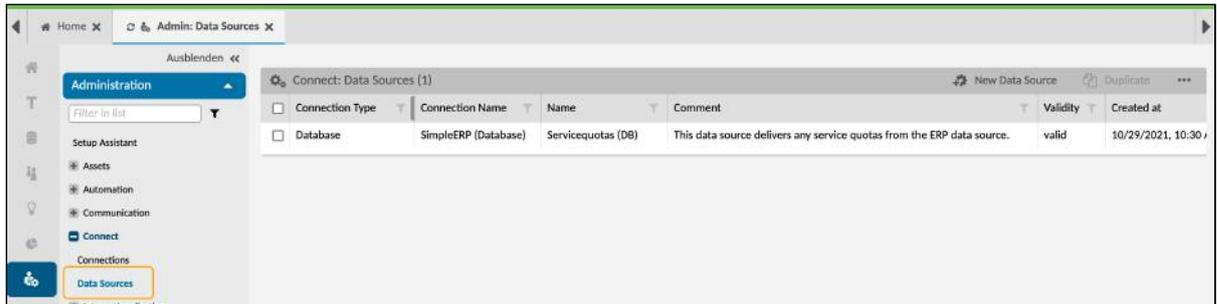
10.1.6.2 Einrichtung

Die Einrichtung der Datenbankverbindungen und der Datenquellen erfolgt im Explorer des Admin-Moduls unter "Connect". Dort finden Sie die Menüpunkte zur

1. Einrichtung einer Datenbankverbindung und



2. Einrichtung der benötigten Datenbank-Datenquellen.



Für die Nutzung einer externen Datenquelle muss beides eingerichtet sein.

1. Datenbankverbindungen einrichten

Die Einrichtung und Konfiguration von Datenbankverbindungen erfolgt schrittweise im Menü *Connect > Verbindungen*. Hinterlegen Sie hier die Verbindungsparameter zur Datenbank. Beim Speichern versucht KIX die Datenbankverbindung aufzubauen. Gelingt dies nicht, werden Sie darauf hingewiesen und können die Einstellungen korrigieren.

Eine Datenbankverbindung benötigt für den Zugang zur Datenbank die Angaben zum Host, den Datenbank-Namen sowie den Nutzernamen und das Passwort, mit dem die Datenbank geschützt sind. Diese Angaben müssen bereitgestellt und im DSN-Feld eingetragen werden. Halten Sie diese Informationen bereit.

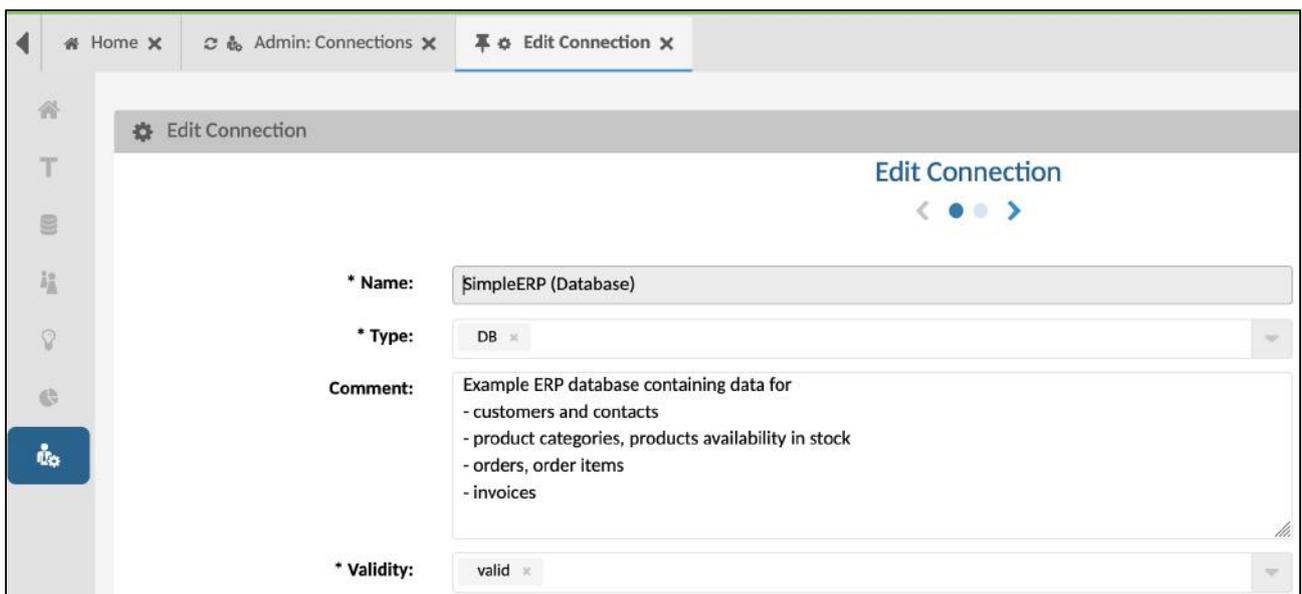


Abb.: Einrichten einer Datenbankverbindung - Schritt 1

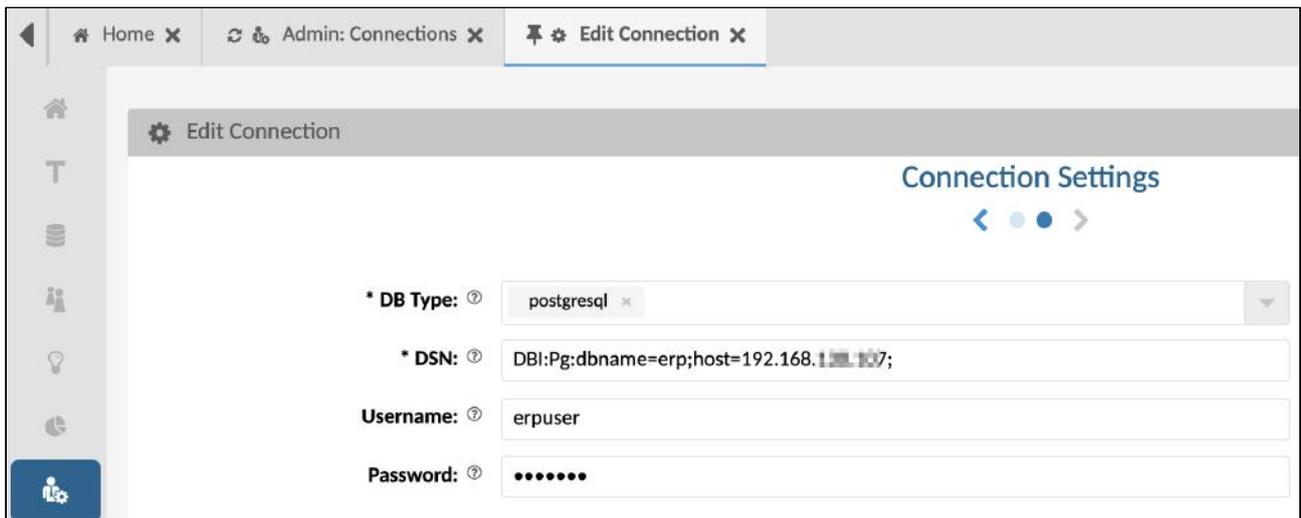


Abb.: Einrichten einer Datenbankverbindung - Schritt 2

Parameter

Parameter	Beschreibung	Beispiel
Name	Der Name der Datenverbindung	SimpleERP (Database)
Typ	Typ der Datenverbindung (wie erfolgt der Zugriff auf die Datenquelle) Die zur Verfügung stehende Auswahl ist abhängig von den installierten Zusatzmodulen.	DB
Kommentar	Optionaler Kommentar	Beispiel ERP Datenbank, enthält Daten zu: <ul style="list-style-type: none"> • Kunden und Kontakten • Produktkategorien, Produktverfügbarkeit im Lager • Bestellungen, Artikelbestellungen • Rechnungen
Gültigkeit	de-/aktiviert Datenquelle für weitere Nutzung	valid invalid

Parameter	Beschreibung	Beispiel
DB-Typ	<p>Datenbanktyp, zu dem verbunden werden soll.</p> <p>Diese Angabe ist relevant, wenn der Typ nicht aus DSN ersichtlich ist.</p>	<p>postgresql mysql mssql</p>
DSN	<p>Für den Datenbankzugriff hinterlegen Sie hier:</p> <ul style="list-style-type: none"> • den Host, auf dem die Datenbank liegt • den Datenbank-Name • die Zugangsdaten zur Datenbank: <ul style="list-style-type: none"> • Nutzer und Passwort. 	<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <pre>DBI:Pg:dbname=erp;host= 192.168.130.107; DBI:Pg:dbname=kix;host=db; DBI:ODBC:Driver={ODBC Driver 17 for SQL Server};Server=<FQDN/ IP>,<Port>;Database=<Datab ase>;UID=<login>;PWD=<pass word></pre> </div> <p>Ersetzen Sie im o. a. Beispiel die folgenden Parameter durch Ihre Werte:</p> <ul style="list-style-type: none"> • <FQDN/IP> • <Port> • <login> • <password> <p>Hinweis: Der Zugriff auf einen MS SQL-Server erfolgt mittels ODBC. Achten Sie im DSN auf die Angabe von Nutzernamen und Passwort. Beide Angaben werden bei ODBC-Verbindungen direkt im DSN definiert. Die Einzelparameter "DB-Nutzer" und "DB-Passwort" sind dann nicht relevant.</p>

Parameter	Beschreibung	Beispiel
DB-Nutzer	Nutzername/Login mit dem der Datenbank-Zugriff erfolgt. Diese Angabe kann bei Zugriff auf einen MS SQL-Server entfallen, da der Nutzername bereits im DSN angegeben wird.	<ul style="list-style-type: none"> erpuser kix
DB-Passwort	Passwort des Nutzers mit dem der Datenbank-Zugriff erfolgt. Diese Angabe kann bei Zugriff auf einen MS SQL-Server entfallen, da das Passwort bereits im DSN angegeben wird.	<ul style="list-style-type: none"> SuperSecretPassword kix

2. Datenbank-Datenquellen einrichten

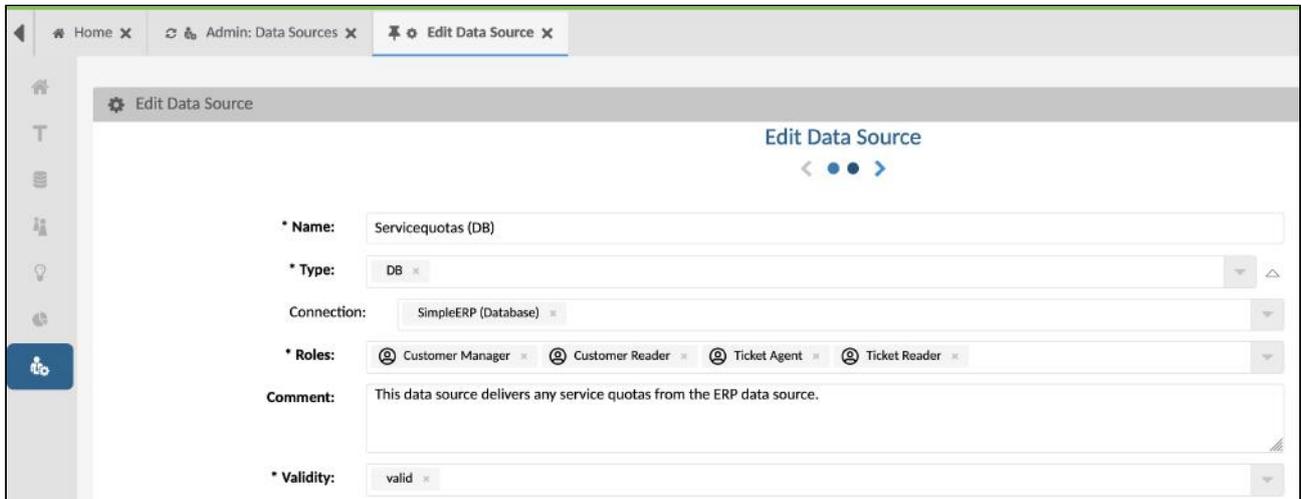
Eine Datenbank-Datenquelle besteht aus allgemeinen Angaben und zwei parametrisierbaren SELECT-Statements, welche einer Datenbankverbindung zugeordnet sind. Beide Abfragen sind dabei nicht auf eine einzelne Datenbanktabelle/View begrenzt, sondern können alle verfügbaren Datenbanktabellen/Views einbinden. Spezielle Einschränkungen, Bedingungen oder Vereinigungen ("JOIN") können direkt in den Abfragen definiert werden.

Die Einrichtung und Konfiguration von Datenbank-Datenquellen erfolgt im Explorer des Admin-Moduls unter *Connect > Datenquellen*.

Für die nachfolgende Beschreibung wird angenommen, dass eine einfache Datenbank mit der Tabelle "Servicequotas (DB)" als Datenquelle dient und die Tabelle folgende Struktur aufweist:

Tabelle/View "servicequotas"				
erp=> \d servicequotas;				
View "public.servicequotas"				
Column	Type	Collation	Nullable	Default
id	integer			
customer_no	character varying(32)			
customer_name	character varying(255)			
comments	character varying(512)			
quota	integer			
startdate	timestamp with time zone			
enddate	timestamp with time zone			
erp=>				

Schritt 1 - Kopfdaten



The screenshot shows the 'Edit Data Source' configuration page. The browser tabs include 'Home', 'Admin: Data Sources', and 'Edit Data Source'. The page title is 'Edit Data Source'. The configuration fields are as follows:

- Name:** Servicequotas (DB)
- Type:** DB
- Connection:** SimpleERP (Database)
- Roles:** Customer Manager, Customer Reader, Ticket Agent, Ticket Reader
- Comment:** This data source delivers any service quotas from the ERP data source.
- Validity:** valid

Abb.: Datenquelle einrichten - Schritt 1

Parameter	Beschreibung	Beispiel
Name	Der Name der Datenquelle.	"Servicequotas (DB)" → apiResourceName: "servicequotas_DB" (s. Info)
Typ	Typ der Datenquelle Die Auswahlliste ist abhängig von den installierten Zusatzmodulen.	DB
Verbindung	Name der zu nutzenden Datenquelle (Auswahlliste)	"Simple ERP (Database)"
Rollen	Schränkt die Verwendung der Datenquelle auf Basis von KIX-Berechtigungsrollen ein. Nur Nutzer mit den hier gewählten Rollenzuordnungen können die Daten aus der KIX-Datenquelle einsehen und im Rahmen ihrer Berechtigungen nutzen.	Customer Manager, Customer Reader, Ticket Agent, Ticket Reader
Kommentar	Optionaler Kommentar	"Diese Datenquelle liefert beliebige Leistungskontingente aus der ERP-Datenquelle."



Parameter	Beschreibung	Beispiel
Gültigkeit	<ul style="list-style-type: none">• gültig: Die Datenquelle einschließlich der darüber bezogenen Daten können in KIX eingesehen und verwendet werden.• (temporär) ungültig: Die Datenquelle ist inaktiv. Es werden keine in der Datenquelle angegebenen Daten bezogenen oder im KIX verwendet.	gültig temporär ungültig ungültig

Info

Der *apiResourceName* wird aus dem vergebenen Namen erstellt. Dieser ist erforderlich für die Ansprache der Datenquelle via der Backend-REST-API von KIX oder für die Einbindung der Datenquelle in Widget- oder Sidebartabellen. Die Bildung des *apiResourceName* erfolgt durch:

- Ersetzung der Großbuchstaben durch Kleinbuchstaben
- Ersetzung der Leerzeichen durch Unterstrich "_"
- Entfernung von ungültigen Zeichen wie Ziffern oder Sonderzeichen. Unterstrich "_" wird akzeptiert.

Die Datenquelle ist in der Rest-API via */datasources/<apiResourceName>* erreichbar.

Schritt 2 - Datenbankabfragen

Bei der Einrichtung von Datenbankabfragen/ SELECT-Statements wird zwischen **Listenanfrage** ("List Select Statement") und **Detailanfrage** ("Item Select Statement") unterschieden. Für beide werden separate Abfragen eingerichtet. Das System unterstützt mit Vorgabewerten, die angepasst werden müssen. Für eine Datenbank-Datenquelle sind beide Angaben erforderlich.

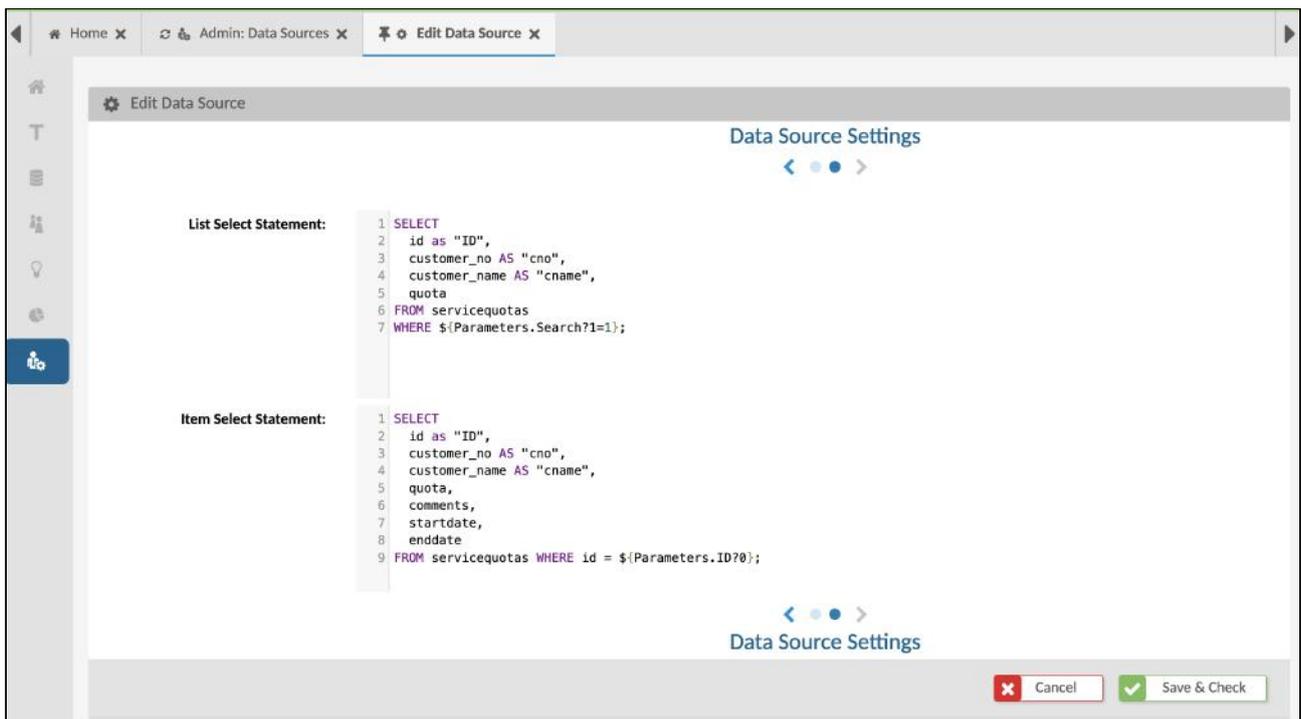


Abb.: Einrichtung Datenquelle - Schritt 2

Parameter	Beschreibung	Beispiel
List Select Statement	<p>Listenanfragen dienen der Ermittlung <u>einer Reihe von Einträgen</u>, die bspw. in einer Feldeingabe zur Auswahl angeboten werden.</p> <p>Typischerweise sind diese kurz gehalten, beinhalten kleinere Ergebnismengen (z.B. nur ID, Name, Erklärung) und haben weiter gefasste Bedingungen, die eine Nutzereingabe für Suchbegriffe (sogenannte Suchkriterien) zulassen.</p> <p>Das Ergebnis einer Listenanfrage wird für die Anzeige von Auswahlwerten in Eingabefeldern oder zur Anzeige eines Einzelwerts genutzt.</p> <p>⚠ Der Alias "ID" muss immer vorhanden sein.</p> <p>Die Notation "<code>\${Parameters.Search?1=1}</code>" ist der Platzhalter für die anzuwendenden Suchkriterien (z.B. bei Eingabe in einem Autocomplete-Feld). Die Fallback-Bedingung (1=1) stellt syntaktisch korrektes SQL sicher, falls kein Suchkriterium verwendet wird.</p> <p>Die Bedingung "<code>AND startdate > '\${Parameters.StartDate?2020-01-01 00:00:00}'</code>" ist eine feste Bedingung die in jeder Abfrage angewandt wird. Wird kein Datum übergeben verwendet die Abfrage automatisch "2020-01-01 00:00:00" als Startdatum.</p>	<pre>SELECT id as "ID", customer_no AS "cno", customer_name AS "cname", quota FROM servicequotas WHERE \${Parameters.Search?1=1}, AND startdate > '\${Parameters.StartDate?2020-01-01 00:00:00}';</pre>

Parameter	Beschreibung	Beispiel
Item Select Statement	<p>Detailanfragen dienen der Anzeige <u>eines konkreten Eintrags</u> aus der Datenquelle.</p> <p>Sie sind relevant, wenn alle Angaben eines konkreten Eintrags benötigt werden. Dabei werden i. d. R. mehrere Spalten bezogen und z. B. in einer <u>Sidebar</u> als Tabelleneinträge oder an einem Feldwert als <u>Info-Popup</u> dargestellt.</p> <p>Im Beispiel umfasst die Detailanfrage zusätzlich den konkreten Wert der Quota und in welchem Zeitbereich sie gilt. Entsprechend werden mehr Spalten in der Abfrage zurück geliefert.</p> <p>Ein Einzeleintrag wird immer auf Basis der ID-Spalte identifiziert. Entsprechend ist die Bedingung " <code>id = \${Parameters.ID?0}</code> "; " immer anzugeben. Sollte kein Schlüsselwert zur Abfrage übermittelt werden wird ein Eintrag mit " <code>id=0</code> " gesucht. Dieser ist i. d. R. nicht vorhanden ist und liefert daher kein Ergebnis liefert. Damit wird syntaktisch korrektes SQL sichergestellt.</p>	<pre>SELECT id as "ID", customer_no AS "cno", customer_name AS "cname", quota, comments, startdate, enddate FROM servicequotas WHERE id = \${Parameters.ID?0};</pre>

⚠ Umbenennung von Datenquellen

Das Umbenennen von Datenquellen ist jederzeit möglich. Alle bestehenden Verwendungen erfordern daraufhin manuelle Anpassungen. Dies betrifft insbesondere die Konfiguration der GUI (SysConfig), Ticketvorlagen und Aktionen. Der alte `apiResourceName` muss dann durch den neuen `apiResourceName` ersetzt werden.

10.1.6.3 Verwendung von Datenquellen

In Dynamischen Feldern

Szenario:

An einem Ticket soll erfasst werden, welche Quota gilt. Dazu wird ein dynamisches Feld eingerichtet, welches von Agenten gesetzt und in Detailansichten mit der Darstellung von Einzelangaben zur Quota eingesehen werden soll.

Um dies zu erreichen, ist im KIX folgendes zu konfigurieren:

1. Einrichtung Datenbankverbindung und Datenquelle (die Datenbank-Tabelle)
2. Anlegen des Dynamischen Felds "ServiceQuota"
 1. Bei Eingabe in dem Feld sollen nur Einträge offeriert werden, die dem Suchmuster entsprechen **und** die nach einen vorgegebenen Stichtag gültig wurden.
3. Einrichtung einer Ticketaktion "(Setze) Service Quota"
4. Konfiguration der GUI zur Darstellung des Dynamischen Feldes in den Ticketdetails

1. Einrichtung von Datenbankverbindung und Datenquelle (die Datenbank-Tabelle)

Es wird vorausgesetzt, dass die oben beschriebene [Datenquelle](#) (siehe Seite 315) "Servicequotas (DB)" bereits eingerichtet wurde.

2. Anlegen des Dynamischen Felds "ServiceQuota" (Menü System > Dynamische Felder)

- Name: "ServiceQuota"
- Label: "Service Quota"
- Feldtyp: Data Source
- Objekttyp: "Ticket"
- SSP-Sichtbar: nein
- Konfiguration:
 - Data Source: "Servicequotas (DB)"
 - Display Pattern: "<quota> (<cname>)"
 - Default Display Columns
 - 1: "quota" / "Quota"
 - 2: "cname" / "Org.-name"
 - 3: "comments" / "Comments"
 - 4: "startdate" / "From"
 - 5: "enddate" / "Until"
 - Cache TTL: 0
 - Count Min: 1
 - Count Max: 1

3. Einrichtung einer Ticketaktion "(setze) Service Quota" (Menü *Ticket* > *Aktionen*)

- Schritt 1 - Aktionsinformationen
 - Referenz Objekt: Ticket
 - Name: "ServiceQuota"
 - Label: "Service Quota"
 - Nutzungskontext: Agent
 - Nutzereingabe erforderlich: ja
- Schritt 2 - Filter
 - Rollen: Ticket Agent
- Schritt 3 - Pre Actions
 - leer
- Schritt 4 - Eingabefelder
 - "Service Quota" - set in mask
 - für Beschränkung der offerierten Auswahlen durch eine Suche nach dem Org.-Namen werden "Erweiterte Optionen" definiert:
 - (1) den Abschnitt "LOADINGOPTIONS" ergänzen um Angaben zum "filter". "<SEARCH_VALUE>" dient als Platzhalter für das durch den Anwender später eingegebene Suchmuster.
 - (2) den Abschnitt "LOADINGOPTIONS" ergänzen um Angaben zum "query". es werden dadurch nur die Einträge offeriert, deren Startdatum nicht früher als 23.04.2022 liegt. Es können auch Platzhalter wie <KIX_TICKET_DynamicField_SomeDateTimeField> oder auch <SEARCH_VALUE> verwendet werden.

```

{
  "option": "LOADINGOPTIONS",
  "value": {
    "expands": [],
    "filter": [
      {
        "filterType": "OR",
        "operator": "LIKE",
        "property": "cname",
        "type": "STRING",
        "value": "<SEARCH_VALUE>"
      }
    ],
    "query": [
      [ "StartDate" , "2022-04-23 00:00:00" ]
    ],
    "includes": [],
    "limit": null,
    "sortOrder": null
  }
},

```

- Schritt 5 - Post Actions
 - leer

4. Darstellung des Dynamischen Feldes in den Ticketdetails

- Navigieren Sie im Explorer zu *System > SysConfig* und ggf. weiter zu *GUI Konfiguration*.
- Suchen Sie den Schlüssel "ticket-details-info-card" und binden Sie dort das Dynamische Feld "ServiceQuota" ein (s. auch Werte Dynamischer Felder anzeigen). Sie können nachfolgenden Codeblock verwenden.
- Laden Sie anschließend das Frontend neu.

Snippet ticket-details-info-card

```

{
  "componentId": "dynamic-field-value",
  "componentData": {
    "name": "ServiceQuota"
  },
  "conditions": [
    {
      "property": "DynamicFields.ServiceQuota",
      "operator": "NE",
      "value": null
    }
  ]
}

```

```
} ,
```

Das Dynamische Feld "ServiceQuota" ist nun in den Ticketdetails enthalten, sodass dessen Wert angezeigt werden kann. Sie können das Dynamische Feld auch in andere Ansichten einbinden (s. Konfigurationsbeispiele GUI-Konfiguration).

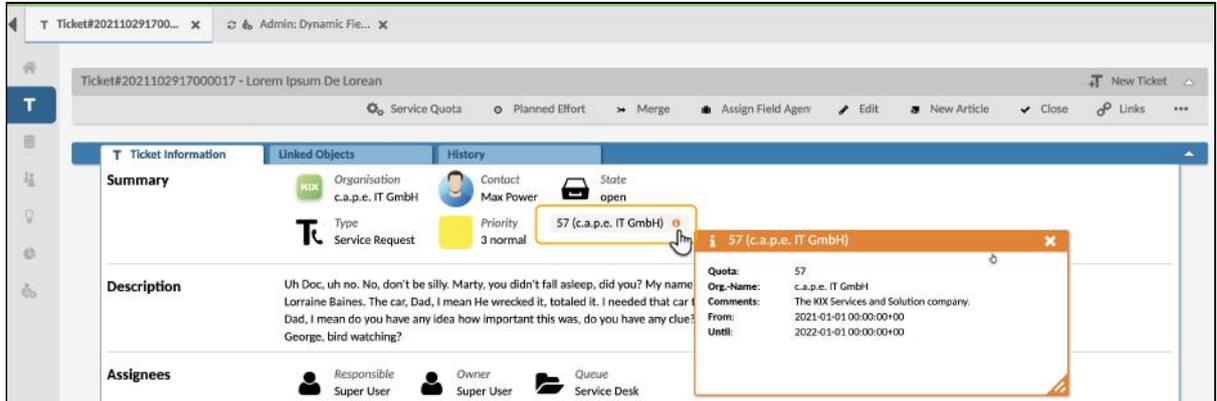
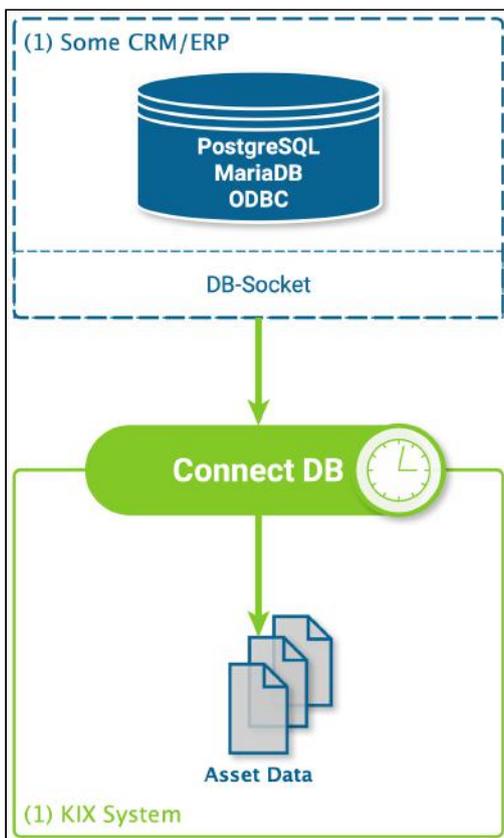


Abb.: Anzeige des Dynamischen Feldes "ServiceQuota" in den Ticketdetails



In Sidebar-Tabellen

Szenario:

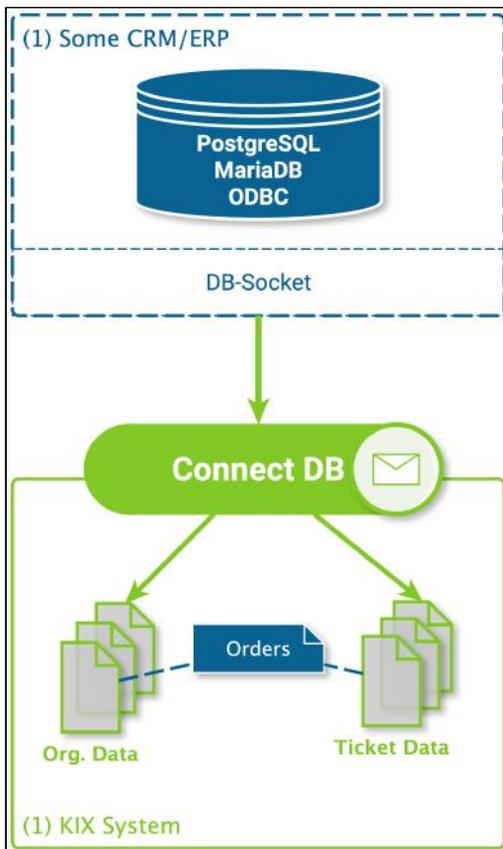
Zu einer Organisation sollen an jedem Ticket der Organisation alle Bestellungen eingesehen werden können. In KIX sind die Kundennummern bekannt, die auch im ERP für Kundenorganisationen verwendet werden.

Um dies zu erreichen, ist im KIX folgendes zu konfigurieren.

1. Einrichtung der Datenbankverbindung und der Datenquelle (die Datenbank-Tabelle)
2. Einrichtung Sidebarwidget "Order List" in der Ticketdetailansicht

Für die nachfolgende Beschreibung wird angenommen, dass eine einfache Datenbank mit den Tabellen "orders" und "customers" als Datenquelle dient. Es wird ferner angenommen, dass die Tabellen folgende Struktur aufweisen:

Tabelle/View orders					
erp=> \d orders;					
Table "public.orders"					
Column	Type	Collation	Nullable	Default	
id	integer		not null	generated always as	
customer_id	integer				
order_number	character varying(255)		not null		
description	character varying(512)		not null		
order_date	date		not null		
...					
erp=> \d customers;					
Table "public.customers"					
Column	Type	Collation	Nullable	Default	
id	integer		not null	generated always as	
customer_name	character varying(255)		not null		
customer_no	character varying(32)		not null		
addr_street	character varying(255)				
addr_city	character varying(128)				
addr_zip	character varying(16)				
addr_country	character varying(128)				
url	character varying(255)		not null		
comments	character varying(512)		not null		
...					
erp=>					



1. Einrichtung der Datenbankverbindung und der Datenquelle

Es wird vorausgesetzt, dass die **oben beschriebene Datenbankverbindung** (siehe Seite 312) **"SimpleERP (Database)"** bereits eingerichtet wurde.

Navigieren Sie zu *Connect > Datenquellen* und nehmen Sie folgende Konfiguration zur Einrichtung der Datenquelle vor:

- Name: "OrderList (DB)"
- Typ: DB
- Verbindung: "SimpleERP (DB)"
- Rollen: "Customer Manager", "Customer Reader", "Ticket Agent", "Ticket Reader"
- Kommentar: -
- Gültigkeit: gültig

- List Select Statement:

```

List Select Statement

SELECT
  o.id AS "ID",
  c.customer_no,
  o.order_number AS "OrderNumber",
  o.order_date AS "Date"
FROM
  orders o LEFT JOIN customers c ON (o.customer_id = c.id)
WHERE ${Parameters.Search?1=0};

```

Die Fallback-WHERE-Bedingung "1=0" hat zur Folge, dass keine Einträge gelistet werden wenn keine gültige Suchanfrage übergeben wird.

- Item Select Statement:

```

Item Select Statement

SELECT
  o.id AS "ID",
  c.customer_no,
  o.order_number AS "OrderNumber",
  o.description AS "Description",
  o.order_date AS "Date"
FROM
  orders o LEFT JOIN customers c ON (o.customer_id = c.id)
WHERE o.id = ${Parameters.ID?0};

```

2. Einrichtung Dynamisches Feld "OrderListSidebar" (Menü System > Dynamische Felder)

- Name: "OrderListSidebar"
- Label: "Order List Sidebar"
- Feldtyp: Data Source
- Objekttyp: "Ticket"
- SSP-Sichtbar: nein
- Konfiguration:
 - Data Source: "OrderList (DB)"
 - Display Pattern: "<quota> (<cname>)"
 - Default Display Columns
 - 1: "OrderNumber" / "OrderNumber"
 - 2: "Date" / "Date"
 - 3: "Description" / "Description"
 - 4: "c.customer_no" / "Customer Number"

- Cache TTL: 0
- Count Min: 1
- Count Max: 1

Das Feld dient nur zum Tragen der Konfiguration der Datenquelle und der Vorgabespalten in der Sidebar, sofern diese nicht weitere oder weniger Spalten definiert. Das Feld muss nicht eingesetzt werden um einen Wert am Ticket zu hinterlegen.

3. Einrichtung Sidebarwidget "Order List" in Ticketdetailansicht

- Öffnen Sie unter *System > SysConfig > GUI-Konfiguration* den Schlüssel "ticket-details"
- Ergänzen Sie den nachfolgenden Codeblock im Array " `sidebars` ":

```

CONFIGKEYHERE

{
  "instanceId": "ticket-details-datasource-orderlist",
  "configuration": {
    "id": "ticket-details-datasource-orderlist",
    "name": "Order List from DataSource",
    "type": "Widget",
    "widgetId": "table-widget",
    "title": "Translatable#Order List",
    "actions": [],
    "subConfigurationDefinition": null,
    "configuration": {
      "id": "ticket-details-datasource-orderlist-widget",
      "name": "orders from data source",
      "type": "TableWidget",
      "objectType": "DataSourceRequestItem",
      "sort": [],
      "subConfigurationDefinition": null,
      "configuration": {
        "id": "ticket-details-datasource-orderlist-table",
        "name": "orders from datasource",
        "type": "Table",
        "objectType": "DataSourceRequestItem",
        "loadingOptions": {
          "filter": [
            {
              "filterType": "OR",
              "operator": "LIKE",
              "property": "c.customer_no",
              "type": "STRING",
              "value": "<KIX_ORG_Number>"
            }
          ]
        },
        "sortOrder": null,
        "query": null,
      }
    }
  }
}

```

```

    "limit": 35
  },
  "specificLoadingOptions": {
    "dfName": "OrderListSidebar"
  },
  "displayLimit": 15,
  "tableColumns": [
    {
      "name": null,
      "property": "OrderNumber",
      "showText": true,
      "showIcon": true,
      "showColumnTitle": true,
      "showColumnIcon": false,
      "size": 250,
      "sortable": true,
      "filterable": true,
      "hasListFilter": false,
      "dataType": "STRING",
      "resizable": true,
      "defaultText": "",
      "translatable": true,
      "titleTranslatable": true,
      "useObjectServiceForFilter": false
    },
    {
      "name": null,
      "property": "Date",
      "showText": true,
      "showIcon": true,
      "showColumnTitle": true,
      "showColumnIcon": false,
      "size": 100,
      "sortable": true,
      "filterable": true,
      "hasListFilter": true,
      "dataType": "STRING",
      "resizable": true,
      "defaultText": "",
      "translatable": true,
      "titleTranslatable": true,
      "useObjectServiceForFilter": false
    }
  ],
  "headerHeight": 1.75,
  "rowHeight": 1.75,
  "emptyResultHint": "Translatable#0 orders found.",
  "fixedFirstColumn": false
},
"showFilter": false,
"shortTable": false,
"predefinedTableFilters": [],

```

```

    "cache": false,
    "resetFilterOnReload": true
  },
  "minimized": false,
  "minimizable": true,
  "icon": "fas fa-shopping-cart",
  "contextDependent": false
},
"size": "large",
"permissions": []
},

```

Tipp

Dieses Beispiel zeigt die Verwendung von festen Parametern zur Einschränkung einer Listenanfrage. Als Parameter können auch variable (durch den Nutzer änderbare) Eingabewerte in Dialogen dienen. Damit können abhängige Eingaben über mehrere Felder abgebildet werden, z.B. "Produktlinie → Produkt → Model" oder "Auftrag → Auftragsposition".

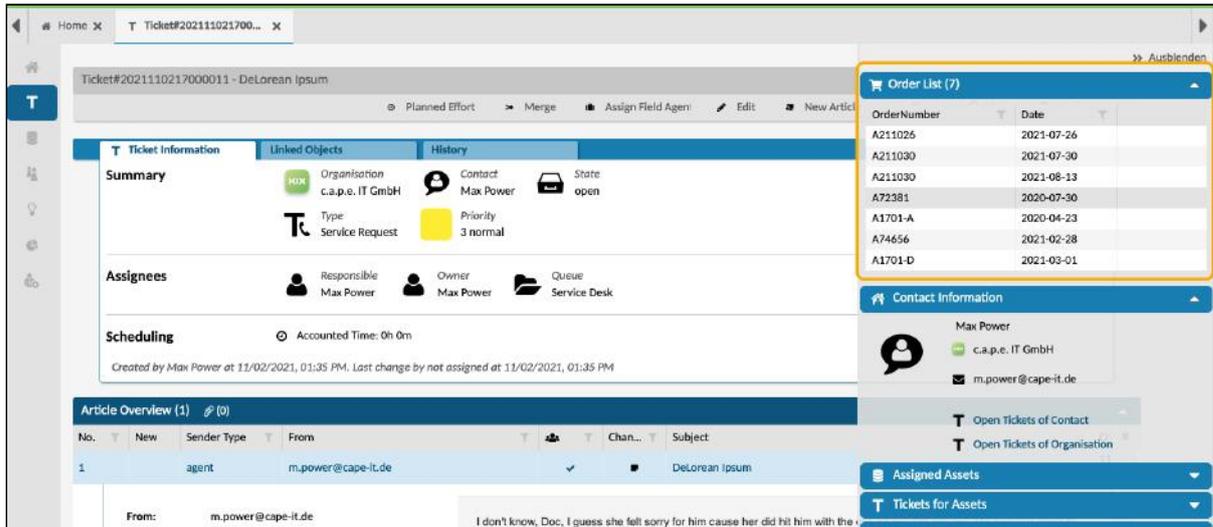


Abb.: Sidebarwidget "Order List" in den Ticketdetails

In Macro Actions

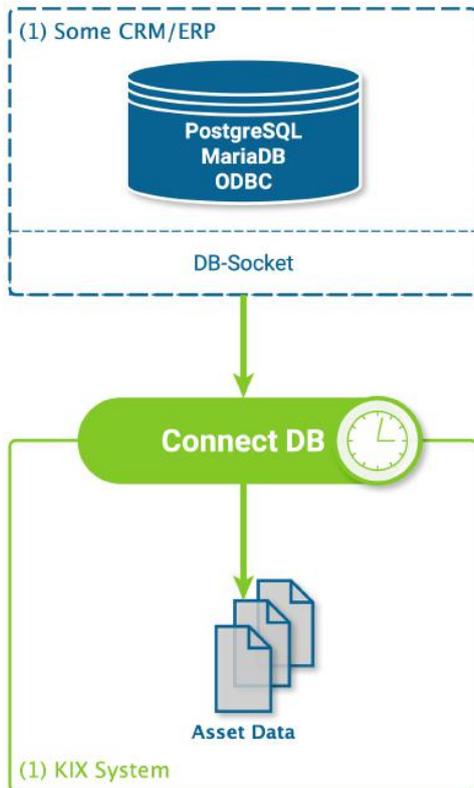
Szenario "Inventarisierung Asset Management"

In der KIX Asset-Datenbank gibt es eine Assetklasse "Quota", in deren Einträgen die Anzahl der pro Kunde und Servicevertrag inkludierten Service-Calls beschrieben wird. Die Informationen wie viele Calls zulässig sind, liegen im ERP in einer einfachen Datenbanktabelle mit den Spalten Kundennummer, Kundenname, Calls u.a. vor. Diese Daten sollen aus der ERP-Datenbank in die KIX-Asset-Datenbank importiert und aktuell gehalten werden.

Um dies zu erreichen, ist im KIX folgendes zu konfigurieren.

1. Erstellung der Assetklasse "Quota"

2. Einrichtung der Datenbankverbindung und der Datenquelle (die Datenbank-Tabelle)
3. Einrichtung des Automatisierungsjobs



Ausschnitt Quelldaten

```
kix=# SELECT * FROM servicequotas;
 id | number | name | comments | quota |
-----+-----+-----+-----+-----
 1 | MY_ORGA | My Organisation | | 100 |
2021-01-01 00:00:00+00 | 2022-01-01 00:00:00+00
 3 | MAMU | Mustermann GmbH | | 60 |
2021-01-01 00:00:00+00 | 2022-01-01 00:00:00+00
 4 | TWD | Torchwood Ltd. | | 39 |
2021-01-01 00:00:00+00 | 2022-01-01 00:00:00+00
 5 | FKLHZ | Fackelholz AG | | 36 |
2021-01-01 00:00:00+00 | 2022-01-01 00:00:00+00
 6 | capeIT | c.a.p.e. IT GmbH | | 71 |
2021-01-01 00:00:00+00 | 2022-01-01 00:00:00+00
 7 | HRMSHL | Harmony Shoal | | 45 |
2021-01-01 00:00:00+00 | 2022-01-01 00:00:00+00
 8 | ITKOM | IT Kommunikations GmbH | | 43 |
2021-01-01 00:00:00+00 | 2022-01-01 00:00:00+00
...
```

1. Erstellung der Assetklasse "Quota"

Als Voraussetzung muss eine Assetklasse "Quota" eingerichtet sein. Dazu wird die nachfolgende, einfache Assetklassen-Definition angenommen (s. auch "Eine Asset-Klasse anlegen").

Assetklasse "Quota"

```
[
  {
    Key          => 'OrgNumber',
    Name         => 'Org. Number',
    Searchable   => 1,
    CustomerVisible => 1,
    Input        => {
      Type => 'Text',
    },
  },
  {
    Key          => 'Quota',
    Name         => 'Quota',
    Searchable   => 1,
    CustomerVisible => 1,
    Input        => {
      Type => 'Text',
    },
  }
]
```

```

    },
  }
];

```

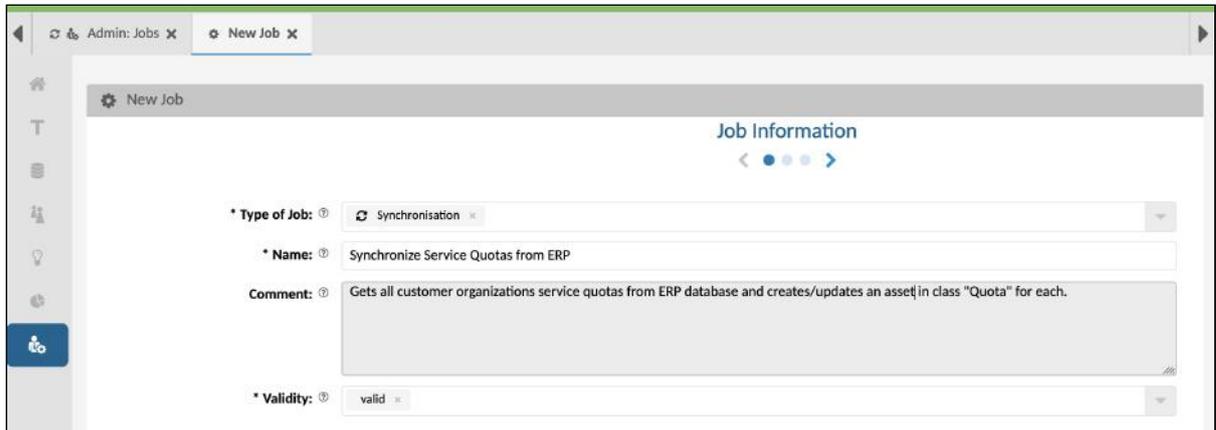
2. Einrichtung Datenbankverbindung und Datenquelle

Im Weiteren wird vorausgesetzt, dass die oben beschriebene [Datenverbindung](#) (siehe [Seite 312](#)) "SimpleERP (Database)" sowie die [Datenquelle](#) (siehe [Seite 315](#)) "Servicequotas (DB)" bereits eingerichtet wurden.

3. Einrichtung Automatisierungsjob

Die Aufgabe des Automatisierungsjobs ist es, alle Servicequotas abzufragen und für jeden Eintrag ein Asset in der Asset-Datenbank zu erstellen oder dieses zu aktualisieren, wenn vorhanden. Damit geprüft werden kann, ob ein Asset bereits existiert, wird der Name einer Servicequota als eindeutig innerhalb der Assetklasse vorausgesetzt. Da die Angaben der ERP-Datenbank nicht direkt in eine Asset-Struktur eingefügt werden können, müssen diese zuvor aufbereitet werden. Es ergibt sich die folgende Liste an Macro Actions:

1. Action: Get List Of Items From Data Source
2. Action: Loop → Macrotyp "Synchronisation"
 1. Get Item From Data Source
 2. XSL Transformation
 3. Create or update asset
3. Als Job Type wird "Synchronisation" verwendet, alle weiteren Job-Parameter (Name, Ausführungsplan etc) können individuell definiert werden.



The screenshot shows the 'New Job' configuration interface. The 'Job Information' section is active, displaying the following fields:

- Type of Job:** Synchronisation
- Name:** Synchronize Service Quotas from ERP
- Comment:** Gets all customer organizations service quotas from ERP database and creates/updates an asset in class "Quota" for each.
- Validity:** valid

Abb.: Konfiguration Schritt 1 - Job Information

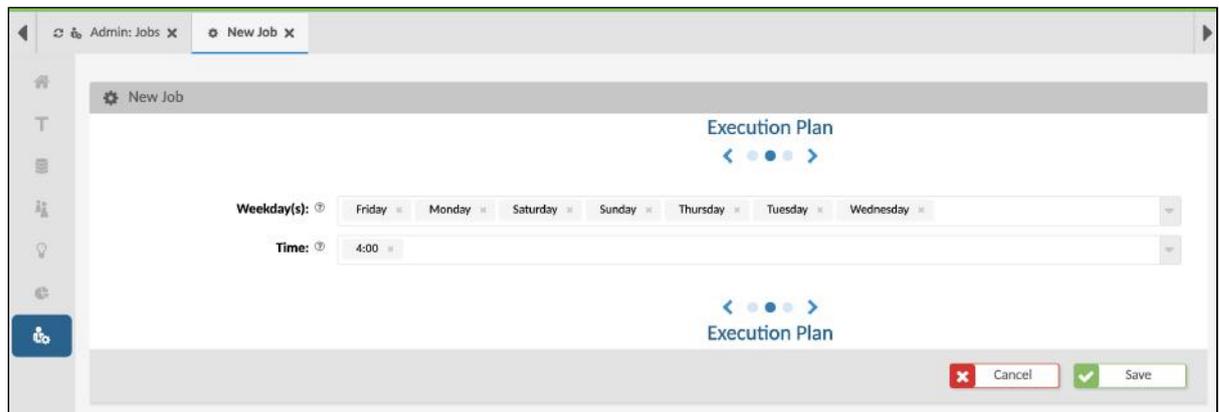


Abb.: Konfiguration Schritt 2 - Ausführungsplan

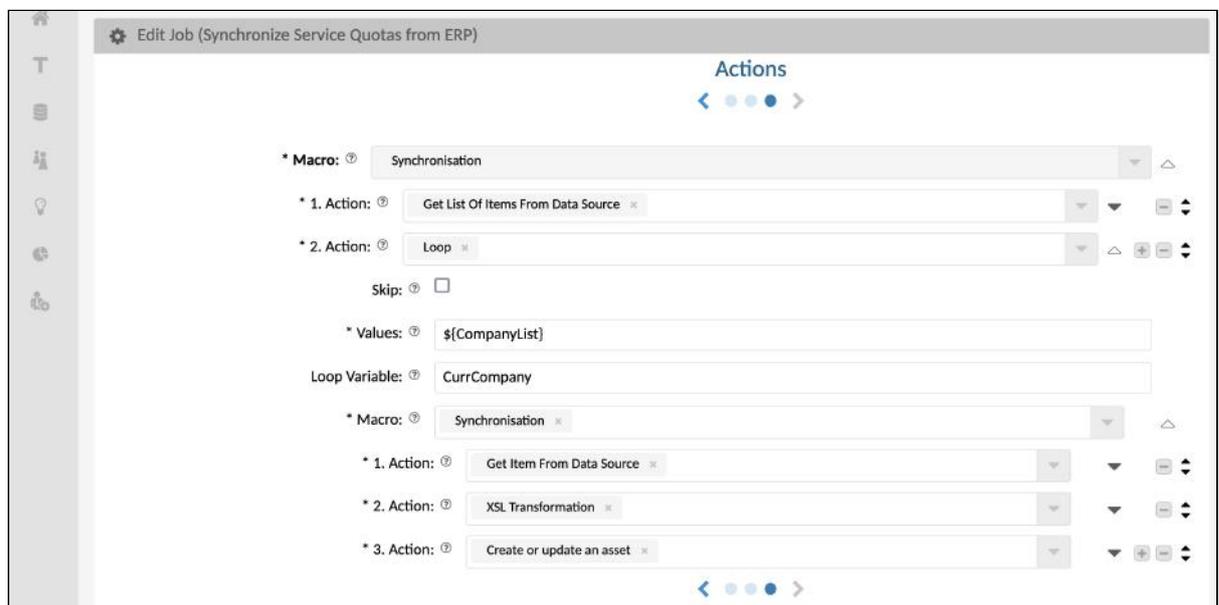


Abb.: Übersicht der nachfolgend anzulegenden Macro Actions

4. Macro Action 1 - Get List Of Items From Data Source
(Parameter der Action s. unter: [Connect](#) (siehe Seite 285))

Edit Job (Synchronize Service Quotas from ERP)

Actions

* Macro: Synchronisation

* 1. Action: Get List Of Items From Data Source

Result names:

ItemList: CompanyList

Skip:

* Data Source: Servicequotas (DB)

Search:

```
1- {
2-   "Item": {
3-     "AND": [
4-       {
5-         "Field": "customer_no",
6-         "Operator": "LIKE",
7-         "Type": "STRING",
8-         "Value": "%"
9-       }
10-     ]
11-   }
12- }
```

Ln: 1 Col: 1

Parameters:

```
1- {}
```

Ln: 1 Col: 1

Debug:

* 2. Action: Loop

Cancel Save

- Action: "Get List Of Items From Data Source"
- ItemList: CompanyList
- Data Source: "Servicequotas (DB)"

- Search:

```

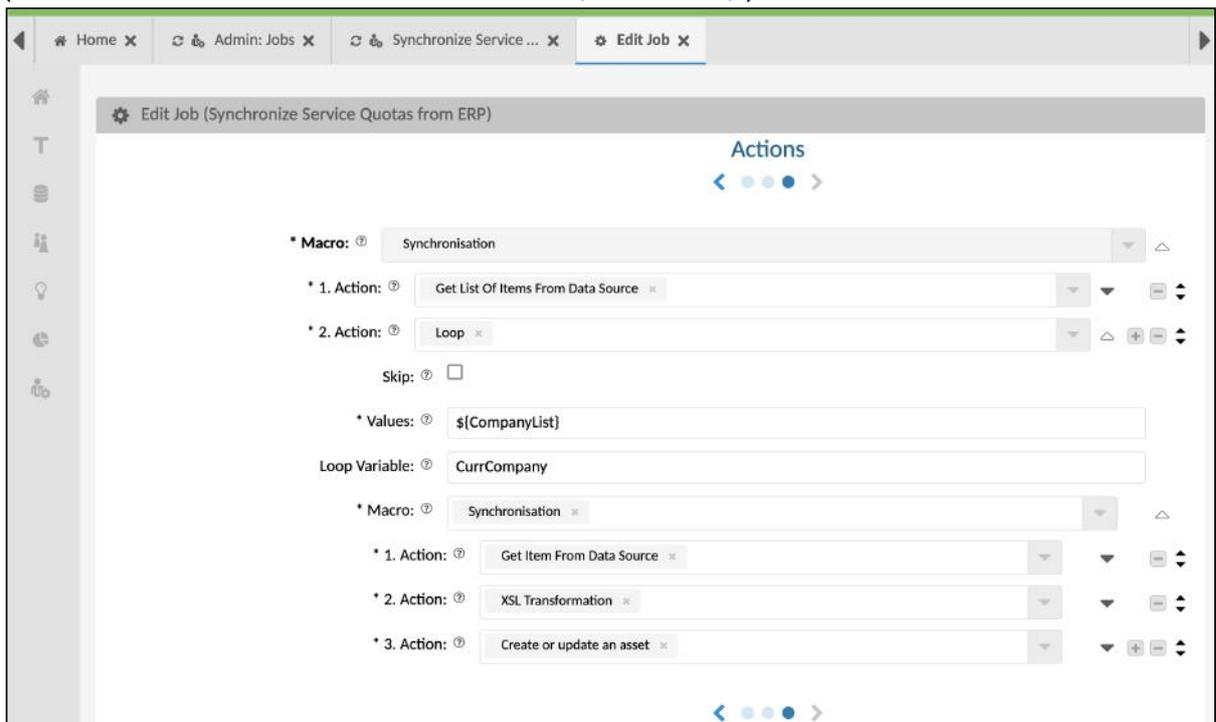
Search

{
  "Item": {
    "AND": [
      {
        "Field": "customer_no",
        "Operator": "LIKE",
        "Type": "STRING",
        "Value": "%"
      }
    ]
  }
}

```

- Parameters
 - {} (leer/keine Parameter)

5. Macro Action 2 - Loop → Macrotyp "Synchronization"
(Infos zur Action s. [Übersicht der Macro Actions](#) (siehe Seite 311))

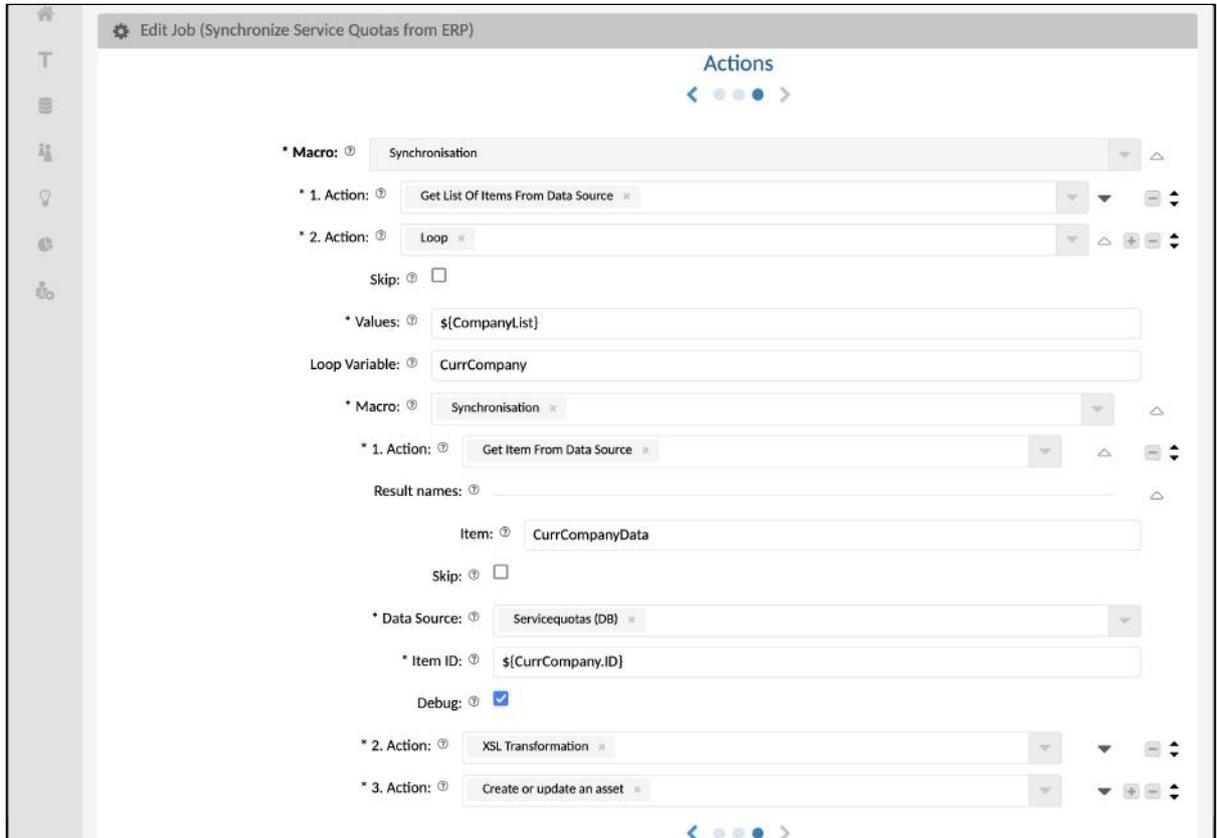


The screenshot shows the 'Edit Job' configuration for 'Synchronize Service Quotas from ERP'. The 'Actions' section is expanded to show a macro named 'Synchronisation'. The macro configuration is as follows:

- Macro:** Synchronisation
- 1. Action:** Get List Of Items From Data Source
- 2. Action:** Loop
 - Skip:
 - Values: \$(CompanyList)
 - Loop Variable: CurrCompany
- 3. Action:** Synchronisation (nested macro)
 - 1. Action: Get Item From Data Source
 - 2. Action: XSL Transformation
 - 3. Action: Create or update an asset

- Action: Loop
- Macro: Synchronisation

6. Macro Action 2.1 - Get Item From Data Source
(Parameter der Action s. unter: [Connect](#) (siehe Seite 285))

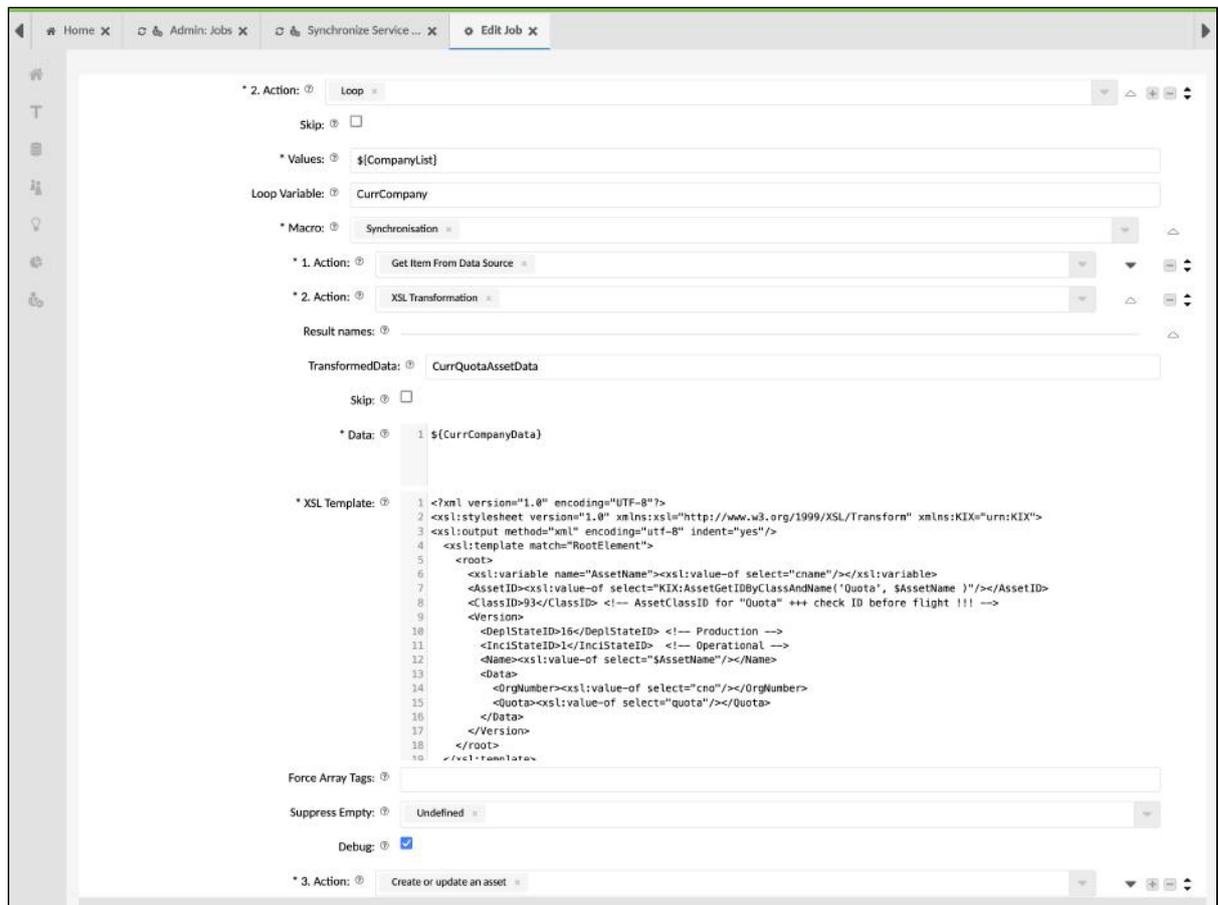


The screenshot displays the configuration for a macro action within a job. The job is named 'Synchronisation'. The macro contains three actions:

- Action 1:** 'Get List Of Items From Data Source'. It is followed by a 'Loop' action.
- Loop Action:** Configured with 'Values: \${CompanyList}' and 'Loop Variable: CurrCompany'. It includes a 'Skip' checkbox which is unchecked.
- Action 2:** 'Get Item From Data Source'. It is configured with:
 - Result names: (empty)
 - Item: CurrCompanyData
 - Skip: (unchecked)
 - Data Source: Servicequotas (DB)
 - Item ID: \${CurrCompany.ID}
 - Debug: (checked)
- Action 3:** 'XSL Transformation' followed by 'Create or update an asset'.

- ⚠ Struktur beachten: gehört zu Macro unter Loop
- Action: Get Item From Data Source
- Item: CurrCompanyData
- Data Source: "Servicequotas (DB)"
- Item ID: `${CurrCompany.ID}`

7. Macro Action 2.2 - XSL Transformation
(Parameter der Action s. unter: [Connect](#) (siehe Seite 285))



* 2. Action: Loop

Skip:

* Values:

Loop Variable:

* Macro: Synchronisation

* 1. Action: Get Item From Data Source

* 2. Action: XSL Transformation

Result names:

TransformedData:

Skip:

* Data:

* XSL Template:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:KIX="urn:KIX">
3 <xsl:output method="xml" encoding="utf-8" indent="yes"/>
4 <xsl:template match="RootElement">
5 <root>
6 <xsl:variable name="AssetName"><xsl:value-of select="cname"/></xsl:variable>
7 <AssetID><xsl:value-of select="KIX:AssetGetIDByClassAndName('Quota', $AssetName)"/></AssetID>
8 <ClassID>93</ClassID> <!-- AssetClassID for "Quota" +++ check ID before flight !!! -->
9 <Version>
10 <DeplStateID>16</DeplStateID> <!-- Production -->
11 <InciStateID>1</InciStateID> <!-- Operational -->
12 <Name><xsl:value-of select="$AssetName"/></Name>
13 <Data>
14 <OrgNumber><xsl:value-of select="cno"/></OrgNumber>
15 <Quota><xsl:value-of select="quota"/></Quota>
16 </Data>
17 </Version>
18 </root>
19 </xsl:template>

```

Force Array Tags:

Suppress Empty:

Debug:

* 3. Action: Create or update an asset

- ⚠ Struktur beachten: gehört zu Macro unter Loop
- Action: XSL Transformation
- TransformedData: CurrQuotaAssetData
- Data: `$(CurrCompanyData)`

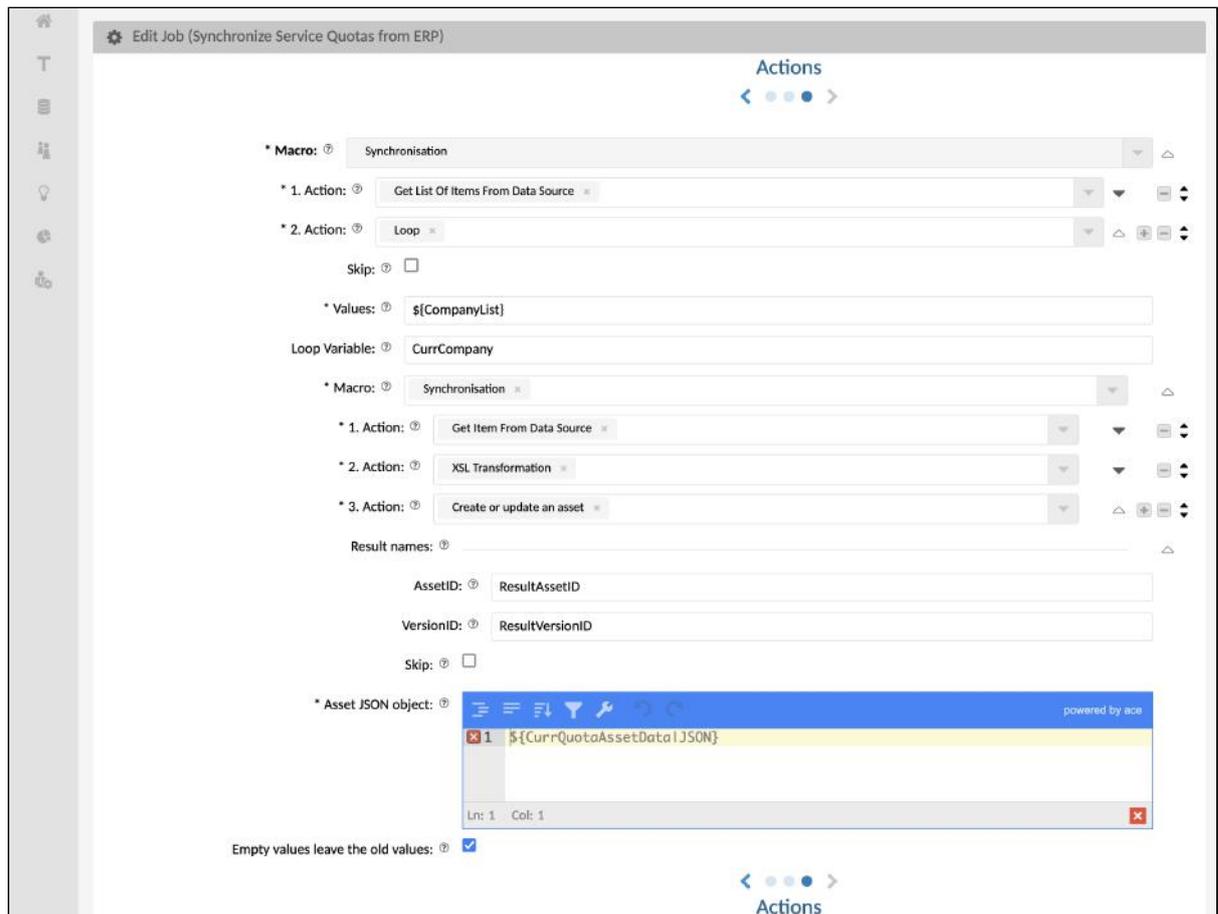
- XSL Template:

XSLT-Mapping Erzeugung Assetdaten "Quota"

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/
Transform" xmlns:KIX="urn:KIX">
<xsl:output method="xml" encoding="utf-8" indent="yes"/>
  <xsl:template match="RootElement">
    <root>
      <xsl:variable name="AssetName"><xsl:value-of select="cname"/></xsl:
variable>
      <AssetID><xsl:value-of select="KIX:AssetGetIDByClassAndName('Quota'
, $AssetName )"/></AssetID>
      <ClassID>93</ClassID> <!-- AssetClassID for "Quota" +++ check ID
before flight !!! -->
      <Version>
        <DeplStateID>16</DeplStateID> <!-- Production -->
        <InciStateID>1</InciStateID> <!-- Operational -->
        <Name><xsl:value-of select="$AssetName"/></Name>
        <Data>
          <OrgNumber><xsl:value-of select="cno"/></OrgNumber>
          <Quota><xsl:value-of select="quota"/></Quota>
        </Data>
      </Version>
    </root>
  </xsl:template>
</xsl:stylesheet>
```

- Force Array Tags: -
- Suppress Empty: Undefined
- Debug: no

8. Macro Action 2.3 - Create or update asset
(Infos zur Action s. [Übersicht der Macro Actions](#) (siehe Seite 311))



Edit Job (Synchronize Service Quotas from ERP)

Actions

* Macro: Synchronisation

* 1. Action: Get List Of Items From Data Source

* 2. Action: Loop

Skip:

* Values: `$(CompanyList)`

Loop Variable: CurrCompany

* Macro: Synchronisation

* 1. Action: Get Item From Data Source

* 2. Action: XSL Transformation

* 3. Action: Create or update an asset

Result names:

AssetID: ResultAssetID

VersionID: ResultVersionID

Skip:

* Asset JSON object: `$(CurrQuotaAssetData | JSON)`

Empty values leave the old values:

- ⚠ Struktur beachten: gehört zu Macro unter Loop
- AssetID: ResultAssetID (oder leer)
- VersionID: ResultVersionID (oder leer)
- Asset JSON Object: `$(CurrQuotaAssetData | JSON)`

Ergebnis

Nach einer manuellen oder automatischen Ausführung des Jobs werden Assets angelegt bzw. aktualisiert:



The screenshot displays the 'Asset Explorer' interface in KIX. The left sidebar shows a navigation tree with categories like Building, Computer, Hardware, Location, Network, Quota (960), Room, Service, and Software. The main area shows a table of assets under the heading 'Overview Assets (40)'. The table has columns for 'A#', 'Name', 'Class', 'Changed at', and 'Changed by'. Three assets are visible in the list, each with a detailed view below it showing 'Incident state', 'Deployment state', 'Org. Number', and 'Quota'.

A#	Name	Class	Changed at	Changed by
179300001	Mustermann GmbH	Quota	10/29/2021, 04:4...	not assigned
179300010	Amadeus-Kliniken gGmbH	Quota	10/29/2021, 04:4...	not assigned
179300011	EM Automotive GmbH & Co. KG	Quota	10/29/2021, 04:4...	not assigned
179300012	Active Connect Data Center	Quota	10/29/2021, 04:4...	not assigned
179300013	Olympic Logistik GmbH & Co KG	Quota	10/29/2021, 04:4...	not assigned
179300014	Otto-Schweizer-Land-Verkehrsunternehmen GmbH	Quota	10/29/2021, 04:4...	not assigned

Details for selected assets:

- Asset 179300011:** Incident state: Operational, Deployment state: Production, Org. Number: EMA, Quota: 27
- Asset 179300012:** Incident state: Operational, Deployment state: Production, Org. Number: ACDC, Quota: 88

10.1.6.4 Verwendung der KIX Datenbank

Sie können auch die KIX-eigene Datenbank anbinden. Die dafür erforderlichen Zugangsdaten können - ausgehend vom Docker Host - aus der Backend-Konfiguration ermittelt werden:

Ermittlung DB-Zugangsdaten

```
# access KIX backend service...
user@dockerhost:/opt/kix-on-premise/deploy/linux$ docker exec -it kixpro-backend-1
bash

# check currently used database setting in KIX configuration...
I have no name!@4be1ec32c580:/opt/kix$ cat /opt/kix/config/01main | grep Database
'DatabaseHost'    => 'db',
'Database'        => 'kix',
'DatabaseUser'    => 'kix',
'DatabasePw'      => 'Password',
'Database::Type' => 'postgresql',
'DatabaseDSN'     =>
'DBI:Pg:dbname=<KIX_CONFIG_Database>;host=<KIX_CONFIG_DatabaseHost>;",
I have no name!@4be1ec32c580:/opt/kix$

# resulting DSN...
DBI:Pg:dbname=kix;host=db;
```

Hinweis

Beachten Sie, dass die Verwendung von KIX-eigenen Strukturen von Updates beeinflusst werden kann. Sollten eigene DB-Strukturen angelegt werden (z. B. Views), versehen Sie diese konsequent mit einem spezifischem Präfix, z. B. "acme_orderlists".

10.1.7 Connect Opsi

Connect Opsi ist ein Zusatzmodul zu KIX Pro. Es bietet die Möglichkeit, Asset-Daten in KIX auf Basis der Inventarisierungslösung opsi (<https://www.opsi.org>) zu erfassen. Die in opsi vorhandenen Informationen können so im KIX Service Management System verwendet werden.

KIX stellt dazu eine weitere, separate Assetklasse "Opsi Device" bereit. Die darin zu importierenden Gerätedaten werden über die opsi bConnect Schnittstelle bezogen. Die Synchronisation erfolgt durch den Job "Opsi - Device Sync". Sowohl die Assetklasse als auch der Job werden beim Aktivieren der opsi-Synchronisation in KIX implementiert.

Inhalte auf dieser Seite:

- [Voraussetzungen](#) (siehe Seite 343)
- [Verwendung](#) (siehe Seite 343)
- [opsi-Synchronisation aktivieren](#) (siehe Seite 344)
- [opsi-Synchronisation zurücksetzen oder ändern](#) (siehe Seite 345)
- [Anpassungen](#) (siehe Seite 346)

10.1.7.1 Voraussetzungen

Zur Verwendung von Connect Opsi wird ein HTTP-/S-Zugang zu opsi benötigt. Dazu muss der FQDN und ggf. die Portnummer angegeben werden. Bei Verwendung von HTTPS dürfen nur korrekte SSL-Zertifikate zum Einsatz kommen. Die Verwendung von ungültigen oder selbst erstellten Zertifikaten kann die Funktion der Schnittstelle negativ beeinflussen.

Weiterhin werden Nutzernamen und Passwort eines zur Anmeldung berechtigten opsi-Nutzers benötigt.

10.1.7.2 Verwendung

Die Bereitstellung von Connect Opsi erfolgt

- für KIX On Premise: durch Neustart des KIX Stacks mittels "start.sh"
- für KIX Cloud: nach Rückmeldung durch den KIX-Support.

Die Konfiguration und Aktivierung des Plugins erfolgt nach der Installation im Admin-Bereich im Connect Opsi Setup.

Welcome to Connect Opsi

The plugin "Connect Opsi" enables KIX to automatically synchronize asset data from Opsi into the CMDB. The plugin provides a pre-configured automation job for this task, as well as a new asset class "Opsi Device".

Activate the plugin

To activate the plugin, please configure the connection to the RPC interface of your Opsi server in the input form below and click the button "Activate".

Reset the plugin

When the plugin is active, the pre-configured job can be reset to the default by clicking the button "Reset to Default".

Change the RPC settings afterwards

After the plugin got activated, the RPC settings can still be changed. Just use the input form below and click the button "Save".

RPC Settings

* RPC User:

* RPC User Password:

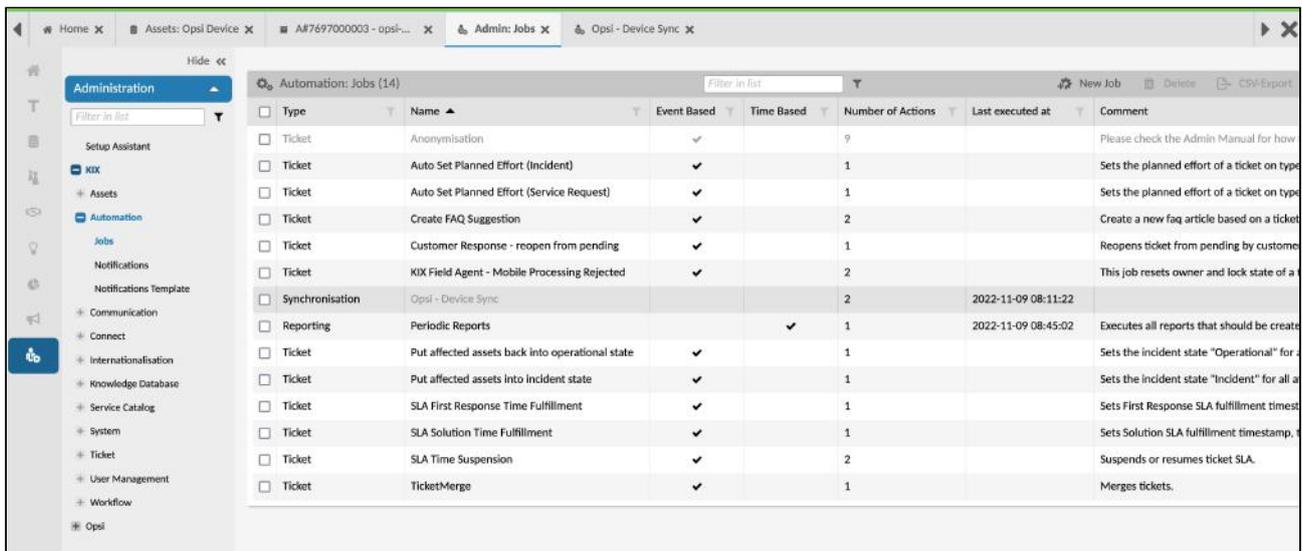
* RPC URL:

Abb.: opsi Setup

10.1.7.3 opsi-Synchronisation aktivieren

Zum Aktivieren der Synchronisation hinterlegen Sie im Admin Modul unter *Opsi > Willkommen* die opsi-URL (z.B. <https://YourOpsiServer.example.com:4447/rpc>) sowie Nutzernamen und Passwort (s. Voraussetzungen). Klicken Sie anschließend auf "Aktivieren". Die Konfiguration wird gespeichert und legt den Automatisierungsjob "Opsi - Device Sync" an.

Für einen ersten Datenimport aus opsi empfehlen wir Ihnen, anschließend den Job "Opsi - Device Sync" manuell auszuführen (Admin Modul unter *KIX > Automatisierung > Jobs*). Dies kann je nach Inhalt der opsi-Datenquelle einige Minuten in Anspruch nehmen. Soll der Datenimport künftig periodisch ausgeführt werden, ordnen Sie dem Job unter "Ausführungsplan" einen entsprechenden Zeitplan zu, z. B. jeden Tag 03:00 Uhr (s. Einen Job anlegen oder bearbeiten). Verändern Sie nicht die weitere Job-Konfiguration! Danach ist die Konfiguration abgeschlossen.



Type	Name	Event Based	Time Based	Number of Actions	Last executed at	Comment
<input type="checkbox"/>	Ticket Anonymisation	✓		9		Please check the Admin Manual for how
<input type="checkbox"/>	Ticket Auto Set Planned Effort (Incident)	✓		1		Sets the planned effort of a ticket on type
<input type="checkbox"/>	Ticket Auto Set Planned Effort (Service Request)	✓		1		Sets the planned effort of a ticket on type
<input type="checkbox"/>	Ticket Create FAQ Suggestion	✓		2		Create a new faq article based on a ticket
<input type="checkbox"/>	Ticket Customer Response - reopen from pending	✓		1		Reopens ticket from pending by custome
<input type="checkbox"/>	Ticket KIX Field Agent - Mobile Processing Rejected	✓		2		This job resets owner and lock state of a
<input type="checkbox"/>	Synchronisation Opsl - Device Sync			2	2022-11-09 08:11:22	
<input type="checkbox"/>	Reporting Periodic Reports		✓	1	2022-11-09 08:45:02	Executes all reports that should be create
<input type="checkbox"/>	Ticket Put affected assets back into operational state	✓		1		Sets the incident state "Operational" for a
<input type="checkbox"/>	Ticket Put affected assets into incident state	✓		1		Sets the incident state "Incident" for all a
<input type="checkbox"/>	Ticket SLA First Response Time Fulfillment	✓		1		Sets First Response SLA fulfillment timest
<input type="checkbox"/>	Ticket SLA Solution Time Fulfillment	✓		1		Sets Solution SLA fulfillment timestamp,
<input type="checkbox"/>	Ticket SLA Time Suspension	✓		2		Suspends or resumes ticket SLA.
<input type="checkbox"/>	Ticket TicketMerge	✓		1		Merges tickets.

Abb.: Der opsi-Synchronisations-Job

10.1.7.4 opsi-Synchronisation zurücksetzen oder ändern

Sollten Sie Anpassungen in der Konfiguration des Jobs vorgenommen haben, um beispielsweise spezifische, erweiterte Angaben zu synchronisieren, können Sie die Vorgabekonfiguration jederzeit mittels des Buttons "Auf Standard zurücksetzen" wieder auf den Auslieferungszustand zurück setzen. Die spezifischen Anpassungen werden dadurch entfernt.

Nach Aktivierung der Synchronisation, können Sie die Einstellungen zur opsi-Schnittstelle weiterhin ändern, um beispielsweise eine geänderte URL oder neue Anmeldedaten zu hinterlegen. Nehmen Sie dazu die erforderlichen Änderungen im Connect Opsl Setup vor und klicken Sie auf "Speichern". Die Änderungen sind bei der nächsten Job-Ausführung aktiv.

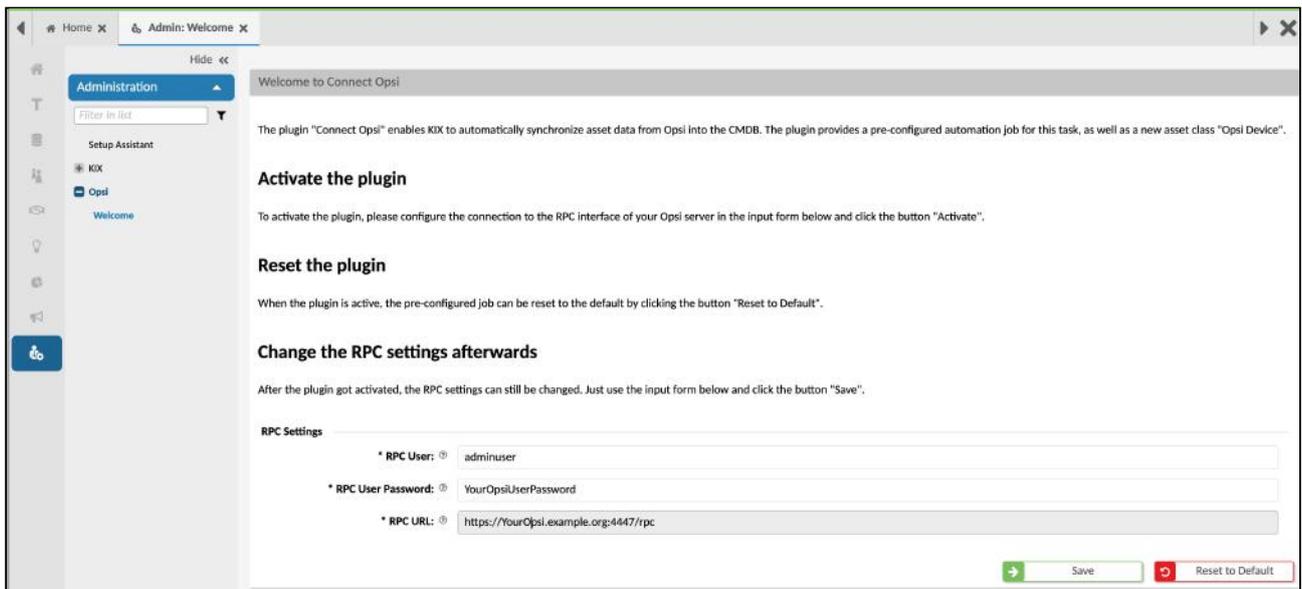


Abb.: Zurücksetzen der Konfiguration

10.1.7.5 Anpassungen

Bei Anpassungswünschen zu den zu verwendenden Daten in der Assetklasse "Opsi Device" oder einer separaten Betrachtung in verschiedenen Assetklassen, kontaktieren Sie uns bitte.

10.1.8 Connect Webservice

Inhalte auf dieser Seite:

- [KIX als Provider - Spezifische Endpunkte](#) (siehe Seite 347)
 - [Übersicht der Endpunkte](#) (siehe Seite 348)
 - [Einrichtung eines Endpunkts](#) (siehe Seite 349)
 - [1. Definition eines Endpunktes](#) (siehe Seite 349)
 - [2. Endpunktaktionen und -antworten](#) (siehe Seite 351)
 - [Zugriff auf erhaltene Parameter](#) (siehe Seite 353)
 - [Verwendung eines Endpunkts](#) (siehe Seite 355)
- [KIX als Requester - Verwendung von Webhooks](#) (siehe Seite 357)
 - [Voraussetzungen](#) (siehe Seite 357)
 - [Erweiterte Macro Actions](#) (siehe Seite 357)
 - [Webhook Extended](#) (siehe Seite 357)
 - [Parameter](#) (siehe Seite 357)
 - [Beispiel: Confluence Seite erstellen](#) (siehe Seite 360)
 - [Beispiel: Redmine Issue erstellen](#) (siehe Seite 362)
 - [Post Attachment as Form Data](#) (siehe Seite 367)
 - [Parameter](#) (siehe Seite 367)
 - [Variablen und Erweiterte Filter](#) (siehe Seite 369)
 - [Anwendung in Macros](#) (siehe Seite 370)
 - [Erweiterte Operationen/Filter](#) (siehe Seite 370)

10.1.8.1 KIX als Provider - Spezifische Endpunkte

Menü

Connect > Endpoints

Neben der universellen REST-API von KIX können mit dem Add-on Connect Webservice spezifische Endpunkte mit konkreten Funktionen oder, im Gegensatz zur REST-API, mit erweiterten Funktionen eingerichtet werden.

Anwendungsszenarien können sein:

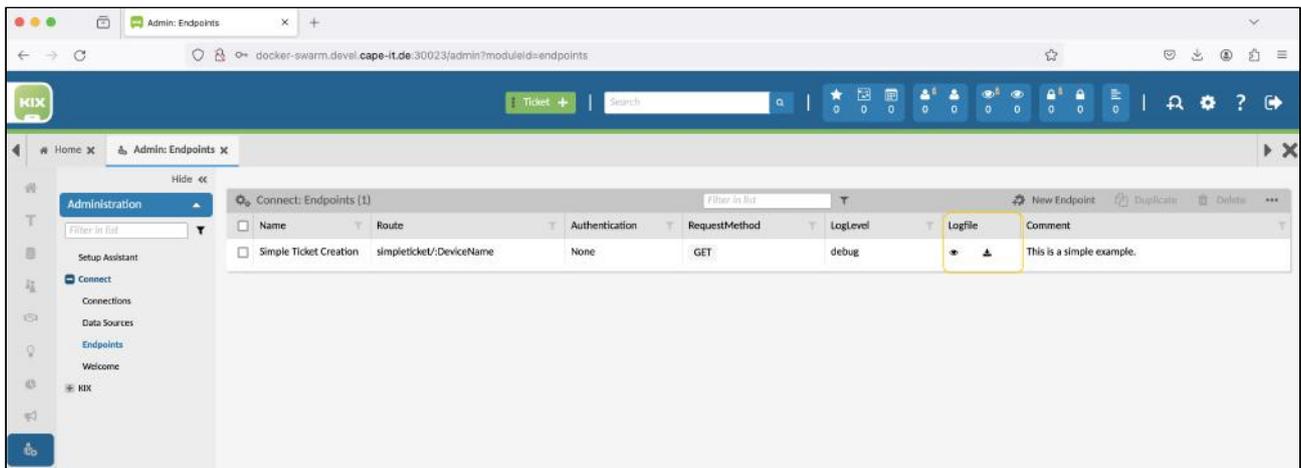
- Schnittstelle für die Aktualisierung von Kontakt-/Organisationen
 - Das auslösende System hat keine Kenntnis über die KIX-internen Identifikatoren oder darf nur ausgewählte Informationen bereit stellen. So kann beispielsweise nur die E-Mail-Adresse eines Kontaktes oder die Nummer der Organisation bekannt sein, jedoch soll eine geänderte

Anschrift, Telefonnummer oder Nachname erfasst werden. Die Aufgabe des Endpunktes ist es, zum übermittelten Kenner den passenden Eintrag zu ermitteln und zu aktualisieren.

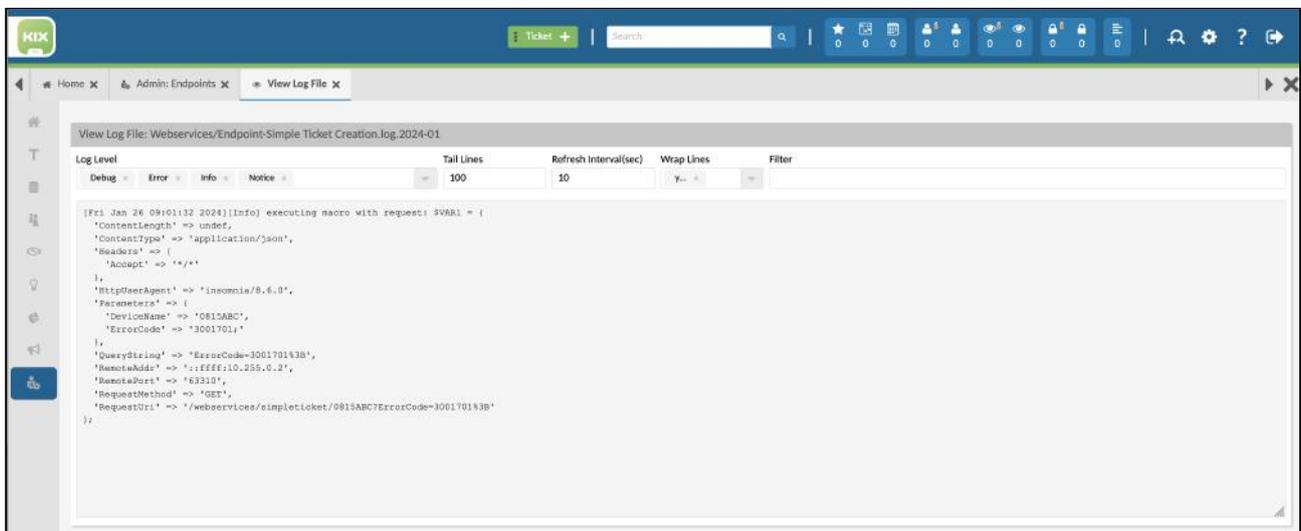
- Schnittstelle für die Aktualisierung von Asset-Daten
 - Die Meldung von freiem Speicher oder dem zuletzt angemeldeten Benutzer kann direkt an KIX erfolgen, ohne dass auf eine Synchronisation der Daten durch KIX gewartet werden muss. Kritische Informationen können damit schneller und einfach am Asset bereit gestellt werden.
- Schnittstelle für das Monitoring von Meldungen
 - IT-Monitoring-Services können Störungen direkt an KIX melden. Die Meldung beinhaltet bspw. einen Geräte-Identifikator und den aktuellen Fehlerzustand. Mit den erweiterten Makro Actions von Connect Webservice kann nicht nur ein Ticket erstellt, sondern bei wiederholten Meldungen auch aktuelle Informationen zu einem bestehenden, offenen Ticket ergänzt werden. Eine Anbindung per Mail ist nicht erforderlich.
- Schnittstelle für Gerätedaten bzw. Erstellung von Tickets aus unvollständigen Eingaben
 - Technische Monitoring-Tools oder Geräte können Fehlerzustände an eine definierte URL melden. Übermittelt werden in der Regel nur die Gerätenummer und ein Fehlercode. Die Aufgabe des Endpunktes ist es, die übermittelten Daten zu vervollständigen, sodass ein Ticket mit allen notwendigen Informationen erstellt oder ein bereits bestehender Vorgang zum Gerät aktualisiert werden kann.
- Ergänzung von Informationen an Tickets
 - Mittels einer einfachen URL kann direkt eine Information ("Flag") an einem Ticket gesetzt werden. Der Klick auf die URL setzt ohne Aufruf des Self Service oder Agentenportals eine Rückmeldung ab.
 - Ein konkretes Beispiel sind niedrigschwellige Genehmigungen oder Ablehnungen. Der Empfänger einer E-Mail mit einer solchen URL kann diese anklicken und somit einen Antrag direkt ablehnen oder freigeben.
 - ⚠ **Wichtig:** Der Versand der URL in einer E-Mail birgt Unsicherheiten. Jeder mit Zugriff auf die URL könnte eine Zustimmung auslösen. Daher sollte dieses Verfahren nur bei begrenzten Risiken eingesetzt werden.

Übersicht der Endpunkte

Eine Übersicht aller konfigurierten Endpunkte finden Sie im Admin Modul unter *Connect > Endpoints*. Die Übersicht enthält zudem den Zugriff auf das jeweilige endpunktspezifische Log.



Das Log kann in der Oberfläche eingesehen oder heruntergeladen werden.



Einrichtung eines Endpunkts

Eigene Endpunkte können Sie im Admin Modul unter *Connect > Endpoints > New Endpoint* eingerichtet. Die Einrichtung erfolgt in zwei Schritten:

1. Definition des Endpunktes
2. Definition der auszuführenden Aktion und Antworten

i Nachfolgend wird anhand eines vereinfachten Beispiels ein Endpunkt eingerichtet. Wir greifen das o. g. Anwendungsszenario "Erstellung von Tickets aus unvollständigen Eingaben" auf. Das Beispiel dient lediglich der Veranschaulichung und erhebt keinen Anspruch auf produktive Einsatzfähigkeit.

1. Definition eines Endpunktes

Folgende Angaben werden in der Definition des Endpunktes hinterlegt:

Parameter	Beschreibung	Beispiel
Name	Name des Endpunkts	Simple Ticket Creation
Route	Der Endpunkt als Unterpunkt zur REST-API /api/v1/webservices/<EndpointRoute>	simpleticket/:DeviceName
Authentication	Definiert, ob der Endpunkt eine Authentifizierung verlangt (aktuell noch keine verfügbar). Als Workaround kann dies über Parameterauswertung in der Macro-Verarbeitung realisiert werden.	none
Request Methode	Soll ein Proxy-Server angesprochen werden?	GET POST PUT PATCH DELETE OPTIONS HEAD
Log Level	Detailtiefe des Logs	error notice info debug
Comment	Kurzbeschreibung des Endpunktes, z.B. Verwendungszweck, wie Prozesszuordnung oder relevante Gegenseite.	This is a simple example.
Gültigkeit	De-/Aktivieren des Endpunktes	gültig (temporär) ungültig

Mittels des Platzhalters ":DeviceName" kann ein Parameterwert innerhalb der URL und ohne Angabe eines Parameternamens übermittelt werden. Im Beispiel wird der Parameter "DeviceName" für die Erstellung des Tickets verwendet. Sonstige Parameter werden beim Übermitteln der URL wie üblich abgetrennt. Das heißt, mit Fragezeichen, gefolgt vom Parameternamen und dem Wert (z. B.: ...webservices/simpleticket/0815ABC?ErrorCode=3001701;)

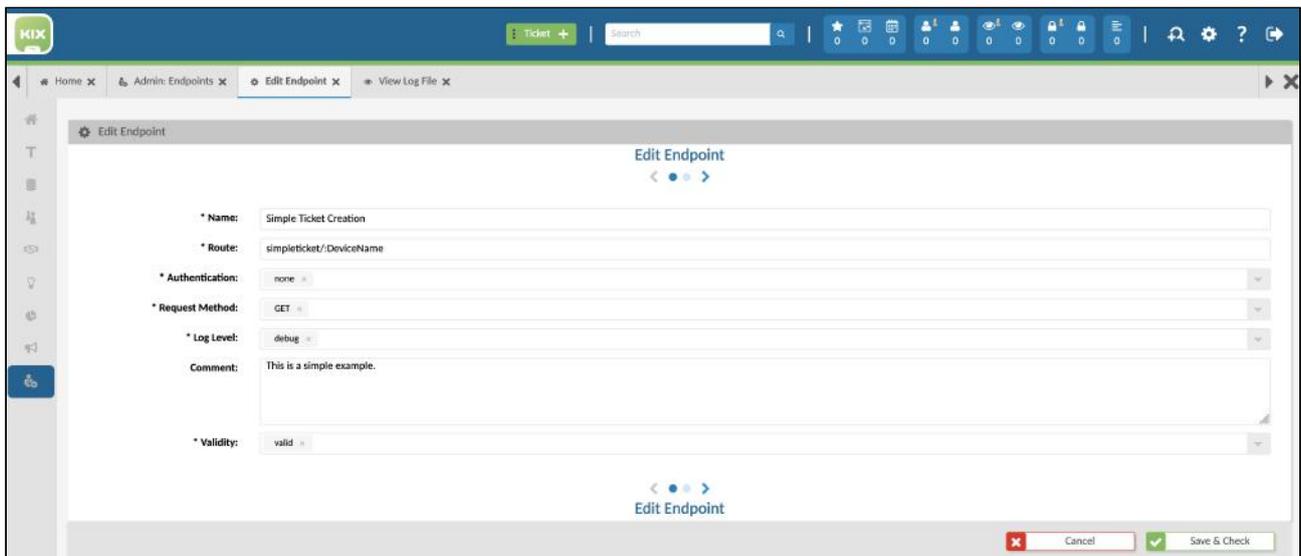


Abb.: Definition des Beispiel-Endpunkts "Simple Ticket Creation"

2. Endpunktaktionen und -antworten

Endpunktaktionen sind grundsätzlich wie Macros in Jobs aufgebaut und offerieren die gleichen Möglichkeiten. Um auf die Inhalte eines Requests wie URL-Parameter oder den Inhalt eines POST-Requests zuzugreifen, wird die Macro-Variable `${Request}` bereit gestellt. Diese beinhaltet alle Angaben des erhaltenen Requests.

Ein einfacher Endpunkt ist ein (erweitertes) Echo. Dazu verwendet das Endpunkt-Macro ausschließlich die Aktion "Set Response". Diese Macro Action erlaubt die Definition des HTTP-Status Codes, der HTTP-Header sowie des zu übermittelnden Inhaltes. Die einfachste Ausprägung ist, mit einem Statuscode ohne weitere Angaben zu antworten.

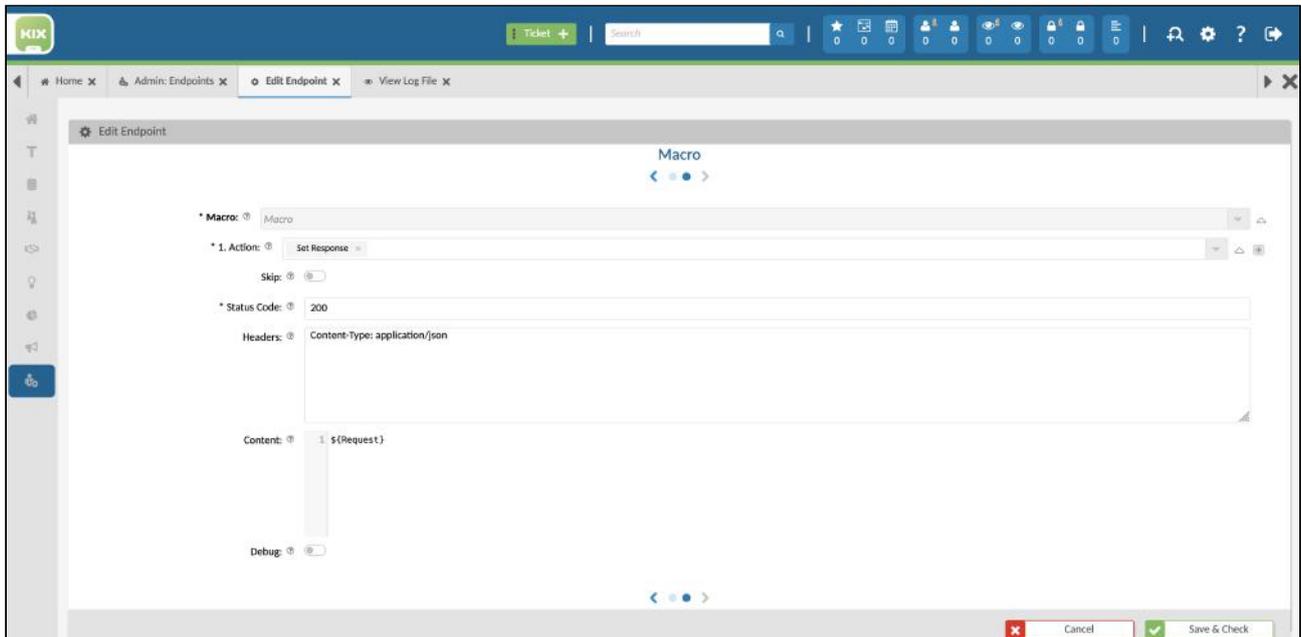


Abb.: Konfiguration des "Simple Ticket Creation"-Endpunkts als Echo

Bei Verwendung des Endpunkts wird somit die vollständige Anfrage als Antwort zurück geliefert:

Aufruf eines Echo-Endpunkts (Verwendung jq für Formatierung)

```
shell>$ curl --request GET \
--url 'http://yourkix-api.kix.cloud/api/v1/webservices/simpleticket/0815ABC?
ErrorCode=3001701' \
--header 'Content-Type: application/json' \
| jq
{
  "Parameters": {
    "ErrorCode": "3001701;",
    "DeviceName": "0815ABC"
  },
  "RequestMethod": "GET",
  "ContentType": "application/json",
  "HttpUserAgent": "curl/8.1.2",
  "RequestUri": "/webservices/simpleticket/0815ABC?ErrorCode=3001701",
  "ContentLength": null,
  "Headers": {
    "Accept": "*/*"
  },
  "RemoteAddr": "::ffff:10.255.0.2",
  "RemotePort": "63873",
  "QueryString": "ErrorCode=3001701%3B"
}
```

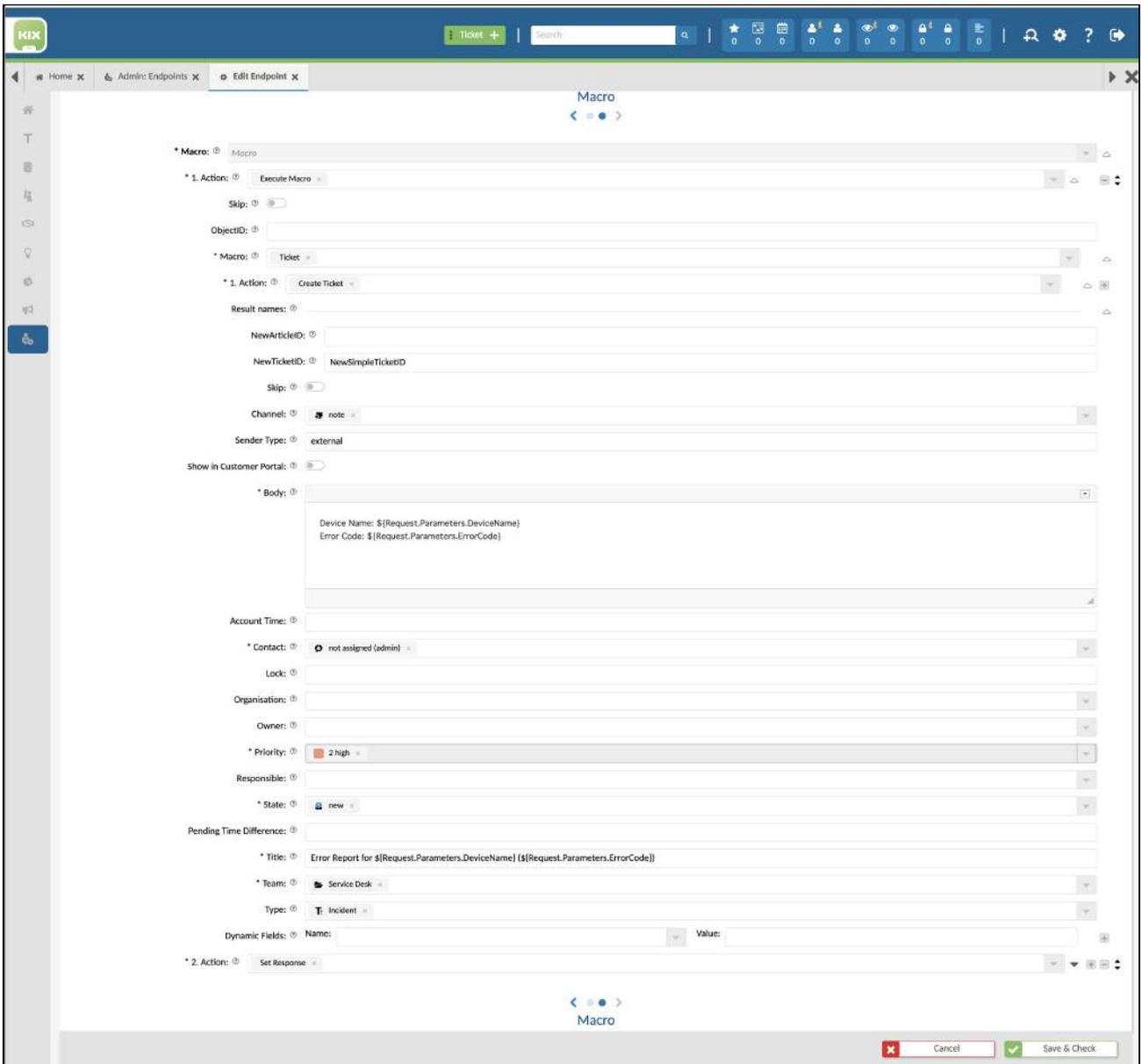
Zugriff auf erhaltene Parameter

Auf die Inhalte des Requests kann mit der Macro-Variable `${Request}` zugegriffen werden. Ist in der Anfrage ein Parameter XYZ enthalten, so wird dieser mittels `${Request.Parameters.XYZ}` abgerufen. Gleiches gilt für Inhalte des Bodies eines POST-/PUT-/PATCH-Requests.

Im folgenden Beispiel soll ein Ticket erstellt und dabei die Angaben aus dem Request genutzt werden. Dazu wird die Macro-Definition angepasst. Die erste Action ist nun der Aufruf eines Macros vom Typ Ticket, welches wiederum die TicketCreate-Aktion benutzt und deren Rückgabewert als Antwort auf den Endpunkt-Aufruf liefert.

Macro-Konfiguration:

- 1. Aktion: Execute Macro
 - Macro: Ticket
 - 1. Aktion: Ticket erstellen
 - NewTicketID: NewSimpleTicketID
 - Title: Fehler Bericht für `${Request.Parameters.DeviceName}` (`${Request.Parameters.ErrorCode}`)
 - Artikeltext:
Geräte Name: `${Request.Parameters.DeviceName}`
Fehler Code: `${Request.Parameters.ErrorCode}`
 - Kontakt: admin
 - Kanal: note
 - Sender Typ: external
 - Priorität: 2 high
 - Team: Service Desk
 - Typ: Incident
- 2. Aktion: Set Response
 - Status Code: 200
 - Headers: Content-Type: application/json
 - Content: `{"ID": "${NewSimpleTicketID}"}`



The screenshot displays the configuration for a macro endpoint in the KIX Administration tool. The macro is titled "Macro" and is configured with the following settings:

- Macro:** Macro
- 1. Action:** Execute Macro
 - Skip:
 - ObjectID:
 - Macro: Ticket
 - 1. Action: Create Ticket
 - Result names:
 - NewArticleID:
 - NewTicketID: NewSimpleTicketID
 - Skip:
 - Channel: note
 - Sender Type: external
 - Show in Customer Portal:
 - Body:


```
Device Name: ${Request.Parameters.DeviceName}
Error Code: ${Request.Parameters.ErrorCode}
```
 - Account Time:
 - Contact: not assigned (admin)
 - Locks:
 - Organisation:
 - Owner:
 - Priority: 2 high
 - Responsible:
 - State: new
 - Pending Time Difference:
 - Title: Error Report for \${Request.Parameters.DeviceName} (\${Request.Parameters.ErrorCode})
 - Team: Service Desk
 - Type: Incident
 - Dynamic Fields: Name: Value:
- 2. Action:** Set Response

Abb.: Konfiguration Endpunkt-Makro mit Parameterzugriff (Teil 1)

(Für eine bessere Übersicht wurden nicht relevante Parameter der Macro-Action "Ticket Create" ausgeblendet.)

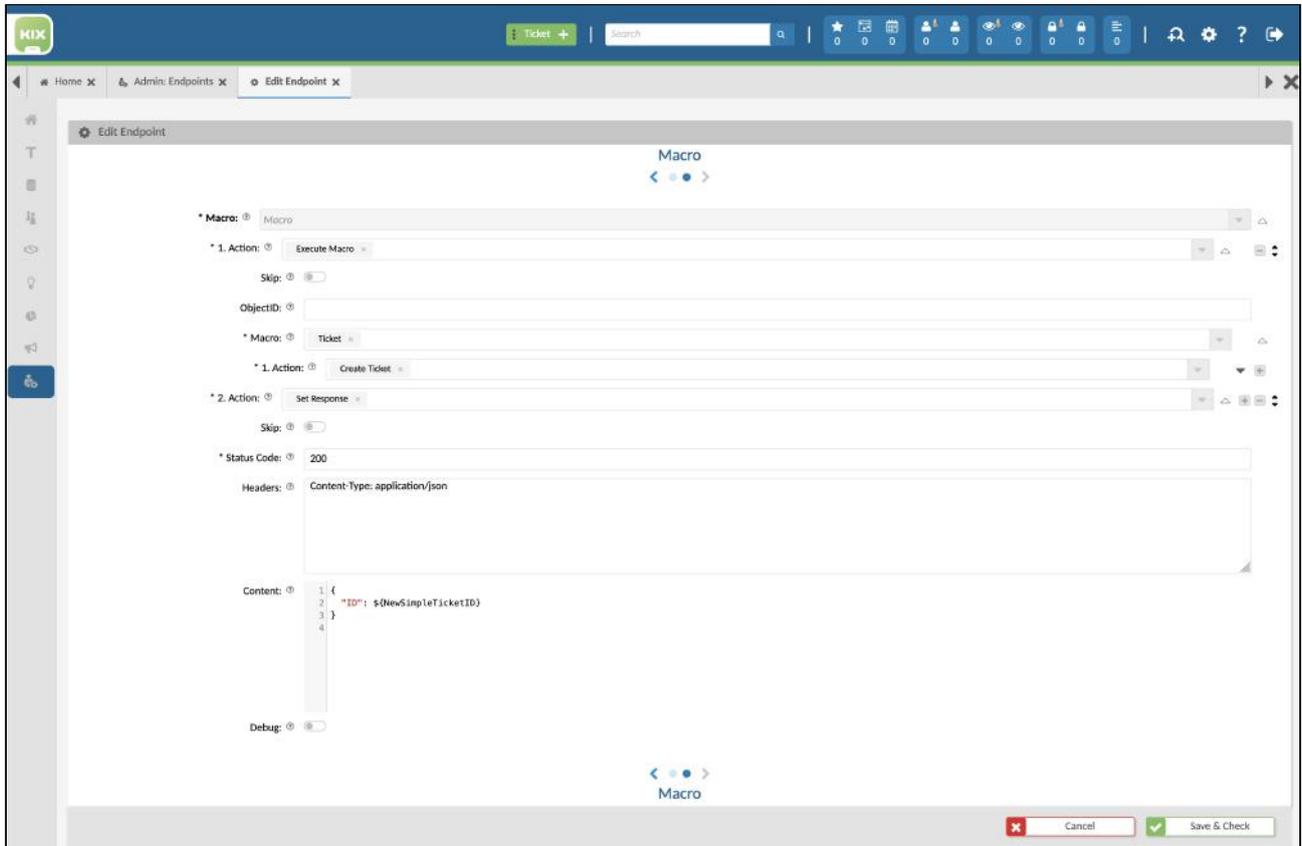


Abb.: Konfiguration Endpunkt-Makro mit Parameterzugriff (Teil 2)

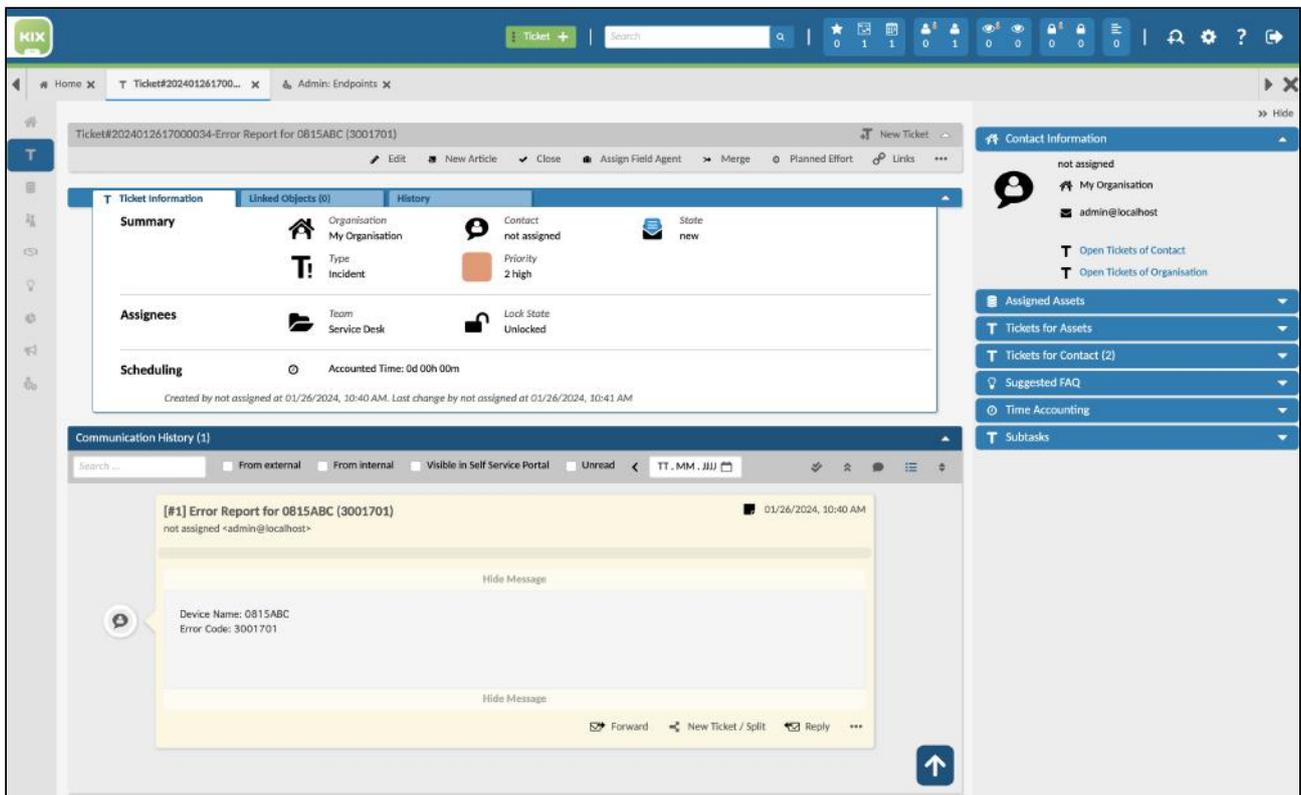
Verwendung eines Endpunkts

Ein zuvor eingerichteter Endpunkt kann über die REST-API unter `"/api/v1/webservices/<EndpointRoute>"` aufgerufen werden. Ist die REST-API bspw. unter `https://yourkix-api.kix.cloud` erreichbar und wurde ein Endpunkt mit der Route `"/simpleticket"` eingerichtet, so kann dieser unter der URL `https://yourkix-api.kix.cloud/api/v1/webservices/simpleticket` aufgerufen werden. Dabei muss die HTTP-Methode verwendet werden, die der Endpunkt-Konfiguration entspricht. Im Beispiel von oben soll der Aufruf `"GET https://yourkix-api.kix.cloud/api/v1/webservices/simpleticket/0815ABC?ErrorCode=3001701"` ein Ticket zum Gerät `"0815ABC"` mit dem Fehlercode `"3001701"` erstellen.

Aufruf des Simple-Ticket-Endpunkts (Verwendung jq für Formatierung)

```
shell> curl --request GET \
  --url 'http://yourkix-api.kix.cloud/api/v1/webservices/simpleticket/0815ABC?
  ErrorCode=3001701' \
  --header 'Content-Type: application/json' \
  | jq
{
  "ID": 3
}
```

Das Ergebnis des Aufrufes ist ein neues Ticket:



The screenshot displays the KIX ticket management interface. The main content area shows a ticket titled "Ticket#202401261700034-Error Report for 0815ABC (3001701)". The ticket details are as follows:

- Summary:** Organisation: My Organisation, Contact: not assigned, State: new, Type: Incident, Priority: 2 high.
- Assignees:** Team: Service Desk, Lock State: Unlocked.
- Scheduling:** Accounted Time: 0d 00h 00m.

The communication history shows a message from "not assigned" at 10:40 AM on 01/26/2024, with the content: "Device Name: 0815ABC, Error Code: 3001701".

Abb.: Neues Ticket, welches durch den Endpunkt erstellt wurde

✓ Tipp

In Verbindung mit den Möglichkeiten weiterer Macro Actions wie ObjectGet, XSLTransform, Set Dynamic Field, CreateOrUpdateAsset, CreateOrUpdateContact, CreateOrUpdateOrganisation, usw. können Sie die Funktionen der Schnittstelle weiter spezialisieren und auf den konkreten Anwendungsfall anpassen.

10.1.8.2 KIX als Requester - Verwendung von Webhooks

Das Add-on Connect Webservice ermöglicht die Einbindung, d.h. Nutzung von HTTP/s-Webservices in KIX. Dabei können die Webservices sowohl Datenquelle als auch Ziel für die aktive Datenübermittlung sein. Gängige Synonyme für diese Funktionen sind auch "ausgehende Webhooks" oder "Invoker".

In Unterscheidung zu [Connect Database \(siehe Seite 311\)](#) wird kein direkter Datenbankzugriff genutzt, sondern der Zugriff auf eine Webschnittstelle via HTTP/s. Darüber hinaus kann Connect Webservice auch Informationen an die angebundenen Webservices übermitteln, statt diese nur zu konsumieren. Typische Szenarien sind:

- Übermittlung von Informationen an dritte Vorgangs-/Ticketsysteme, wie bspw. Jira, Redmine, Trello, BMC, etc.
- Auslösen von Softwareauslieferungen durch Endpoint-Managementsysteme wie Baramundi, ido-IT, OPSI, etc.
- Übermittlung von Abrechnungsdaten aus Tickets an CRP- oder ERP-Systeme wie Navision, Axapta, Sugar CRM etc.

Voraussetzungen

Die technische Voraussetzung ist das Vorhandensein einer HTTP/S-Verbindung vom KIX-Backend zum Zielsystem einschließlich der Zugangsdaten (falls erforderlich) mit entsprechenden Berechtigungen.

Erweiterte Macro Actions

Das Add-on Connect Webservice enthält zusätzliche Macro Actions für die Kommunikation mit anderen Webservices.

Webhook Extended

Die Macro Action "Webhook Extended" ermöglicht die aktive Kommunikation mit einem anderem Webservice.

Mögliche Auslöser der Macro Action können spezifische Änderungen an einem Ticket oder Asset sein. Mittels periodischer Jobs können auch Listen von Einträgen abgerufen und verarbeitet werden. Konkrete Anwendungen sind bspw. die Inventarisierung der Asset-Datenbank von KIX oder die proaktive Erstellung von Tickets.

Die Macro Action funktioniert ähnlich der im Basisprodukt enthaltenen Macro Action "Webhook". Sie unterscheidet sich darin, dass komplexe, mehrstufige Datenstrukturen als Nutzlast im Aufruf (Request) übergeben werden können. Zudem kann die Antwort (Response) des aufgerufenen Webservices entgegen genommen und für weitere Macro Actions (z. B. [XSL Transformation](#), [Get Object Data \(siehe Seite 285\)](#)) weiterverwendet werden.

Parameter

Parameter	Beschreibung	Beispiel
Response	Variablenname für die Ergebnisstruktur des Webrequests. Es wird ein strukturiertes Objekt zurückgegeben, welches HTTPCode, Status, Content, Header (Key-Value-Paare) und Result mit dem Ergebnis als JSON-Objekt enthält.	-
URL	Die aufzurufende URL. Es werden KIX-Platzhalter unterstützt.	-
Method	die Aufrufmethode (HTTP-Verb)	GET POST PUT PATCH DELETE OPTIONS HEAD
Use Proxy	Soll ein Proxy-Server angesprochen werden?	yes no
Proxy	Welcher Proxy Server soll angesprochen werden? Ist nichts angegeben, wird die allgemeine Proxy-Einstellung aus der Systemkonfiguration verwendet (Option "WebUserAgent::Proxy").	http://proxy.example.com:3128
Headers	zu übermittelnde HTTP-Header	Content-Type: application/json
Content	Der zu übermittelnde Inhalt ("payload"). Kann als JSON-Zeichenkette oder als Macro-Variable angegeben werden. Es werden KIX-Platzhalter unterstützt.	<pre>{ "issue": { "project_id": 1, "subject": "<KIX_TICKET_Title>", "description": "Lorem ipsum..." } }</pre> <p>oder</p> <pre>\${SomeMacroVariable}</pre>

Parameter	Beschreibung	Beispiel
Debug	Erweitertes Logging für den Aufruf de-/aktivieren. Tipp: Debug-Informationen können auch der Job-Historie entnommen werden.	yes no

JSON-Quotierung bei Platzhalterinhalten

Bei der Verwendung von Platzhaltern im Content ist darauf zu achten, dass deren Inhalte bereits JSON-quotiert vorliegen. Das ist insbesondere bei mehrzeiligen Inhalten **nicht** der Fall. Sollte solch ein Platzhalter benutzt werden, kann dieser zunächst einer Macro-Variablen zugewiesen und diese nachgehend JSON-quotiert eingefügt werden.

1. Macro Action ("Assign Variable")

Variable := "ArticleBody"

Value := "<KIX_FIRST_Body>"

2. Macro Action ("Webhook Extended")

```
{
  "issue": {
    "project_id": 1,
    "subject": "<KIX_TICKET_Title>",
    "description": "{$ArticleBody|JSON}"
  }
}
```

Beispiel: Confluence Seite erstellen

Szenario:

Erstellung einer neuen Confluence Seite, sobald ein Ticket vom Typ "Problem" mit dem Close Code "Known Error" geschlossen wird. So können bspw. bekannte Fehler direkt kommuniziert und für alle einsehbar dokumentiert werden.

Das Beispiel erstellt die neue Seite im Confluence als Unterseite zu einer bestehenden Seite mit der ID 123456 im Bereich "BPTTEST". Dabei ist es gleich, welche Antwort Confluence liefert ("fire and forget"). Als Authentifizierungsmöglichkeit wird Basic Auth, d. h. eine Kombination aus Nutzernamen und Passwort genutzt.

Vorbereitung:

- Dynamisches Feld "Close Code" um Wert "Known Error" erweitern
- Dynamisches Feld "Workaround" anlegen und in die Ticketaktion "Close" [integrieren](#) (siehe Seite 102)
- Confluence-Zugangsdaten besorgen
 - im Beispiel: Username / Password
- ID der Seite im Confluence ermitteln, unter der die zu erstellende Seite angesiedelt sein soll
 - im Beispiel 123456
- Erzeugung eines HTTP-Headers für Basic Auth
 - der Wert wird als HTTP-Header benötigt, um sich am Confluence anzumelden
z. B. mittels eines Kommandozeilentools "base64":

```
someLinuxUnixSystem:~# echo 'Authorization: Basic "$(echo -n  
'Username:Password' | base64)'  
Authorization: Basic VXNlck5hbWU6UGFzc3dvcjQ=
```

Konfiguration des Jobs:

1. Grundkonfiguration:
 - Job Typ: "Ticket"
 - alle weiteren Job-Parameter (Name, Ausführungsplan, Filter etc) können individuell definiert werden
2. Macro Action 1 - Artikelinhalt in Macro-Variable schreiben
 - Action: Set Variable
 - Variable: ArticleBody
 - Wert: <KIX_ARTICLE_Body>

3. Macro Action 2 - Aufruf des Webservice

- Action: Webhook Extended
- Response: -
- URL: `http://confluence.intra.example.com:8092/rest/api/content/`
- Method: POST
- Use Proxy: -
- Proxy: -
- Headers
Content-Type: `application/json`
Authorization: `Basic VXNlcm5hbWU6UGFzc3dvcmQ=`
- Content:

Content

```
{
  "type": "page",
  "space": {
    "key": "BPTTEST"
  },
  "ancestors": [
    {
      "type": "page",
      "id": "123456"
    }
  ],
  "title": "Known Error <KIX_TICKET_TicketNumber> - <KIX_TICKET_Title>",
  "body": {
    "storage": {
      "value": "<h1>Description</h1>${ArticleBody|JSON} <br/>
<h2>Workaround</h2> <KIX_TICKET_DynamicField_Workaround>",
      "representation": "storage"
    }
  }
}
```

Verweise:

- <https://developer.atlassian.com/server/confluence/confluence-rest-api-examples/>
- https://en.wikipedia.org/wiki/Basic_access_authentication

Beispiel: Redmine Issue erstellen

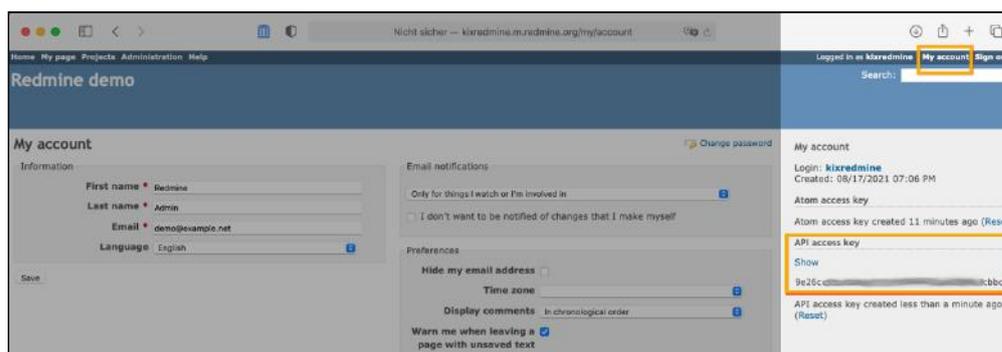
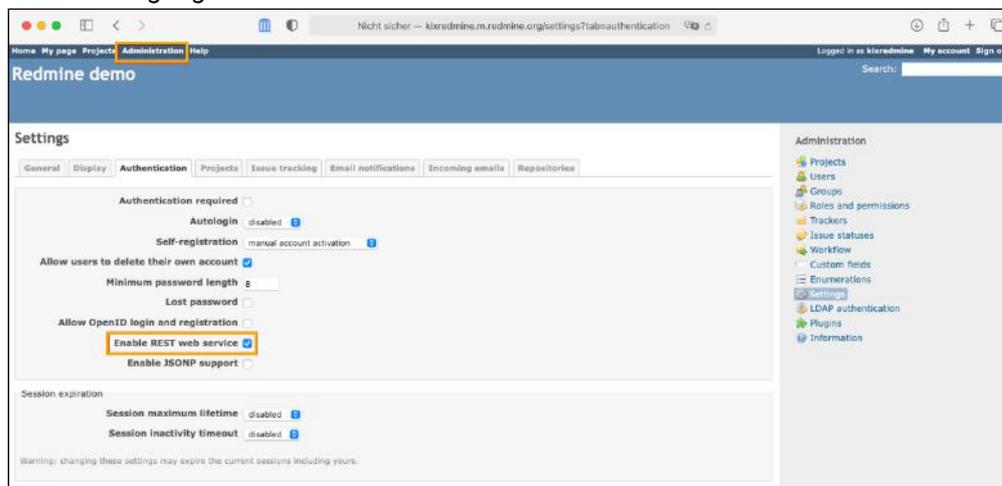
Szenario:

Erstellung eines Issues in Redmine aus einem KIX-Ticket, sobald ein Ticket vom Typ "Incident" mit dem Solution Code "Processing in Redmine" versehen wird. Die inhaltliche Bearbeitung des Vorgangs erfolgt im Projektmanagement Tool "Redmine". Als Authentifizierung wird ein zuvor ermittelter Authentifizierungstoken genutzt.

Das Beispiel erstellt ein neues Issue in Redmine mit dem Titel und dem Inhalt des ersten Artikels des auslösenden KIX-Tickets. Auslöser des Jobs ist eine Änderung im Dynamischen Feld "Close Code" mit Filterung auf den Wert "Processing in Redmine". Die von Redmine erstellte IssueID wird in KIX in einem Dynamischen Feld hinterlegt. Kann kein Issue erstellt werden, wird ein Hinweis hinterlegt.

Vorbereitung:

1. Redmine-Zugang einrichten und AuthToken ermitteln



2. KIX Dynamisches Feld "Close Code (siehe Seite 46) " um Wert "Processing in Redmine" erweitern.
3. Dynamisches Feld "RedmineIssueID" in KIX einrichten und zur Verwendung in der Ticketaktion "Bearbeiten" bereitstellen (siehe Seite 102) .

4. Dynamisches Feld "Project" in KIX einrichten und zur Verwendung in der Ticketaktion "Bearbeiten" bereit stellen.

Konfiguration des Jobs:

Die Job-Konfiguration enthält im Schritt 4 ("Actions") mehrere Macro Actions, die aufeinander aufbauen:

1. Get Object Data - Ermitteln der Daten für den Web-Request
2. XSL Transformation - Aufbereitung des Webrequest-Inhalts
3. Webhook Extended - Aufruf des Webservice
4. XSL Transformation - Verarbeitung der Response
5. Set Dynamic Field - Speichern der Redmine-IssueID

Die Konfiguration im Detail:

1. Grundkonfiguration:
 - Als Job Typ wird "Ticket" verwendet,
 - alle weiteren Job-Parameter (Name, Ausführungsplan, Filter etc.) können individuell definiert werden.
2. Macro Action 1 - Ermitteln der Daten für den Web-Request
 - Action: Get Object Data
 - ObjectData: TicketContentUnprepared
 - Objekttyp: Ticket
 - Objekt-ID: <KIX_TICKET_ID>
 - Includes: DynamicFields,Priority,Articles
 - Expands: -/leer
3. Macro Action 2 - Aufbereitung des Webrequest-Inhalts
 - Action: XSL Transformation
 - TransformedData: RMRequestPrepared
 - Data: `${TicketContentUnprepared}`
 - XSL Template:

XSL Template

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/
Transform" xmlns:KIX="urn:KIX">
  <xsl:output method="xml" encoding="utf-8" indent="yes"/>
  <xsl:template match="RootElement">
    <root>

      <!-- This is the custom part... -->
```

```

<xsl:variable name="ProjectName" select="DynamicFields[Name =
'Project']/DisplayValue"/>

<issue>

  <!-- translate project name in KIX to Redmine project id -->
  <project_id>
    <xsl:choose>
      <xsl:when test="$ProjectName='Sample Project'">1</xsl:when>
      <xsl:when test="$ProjectName='Other Project'">2</xsl:when>
      <xsl:otherwise>1</xsl:otherwise>
    </xsl:choose>
  </project_id>

  <!-- translate KIX priority to Redmine priority -->
  <priority_id>
    <xsl:choose>
      <xsl:when test="Priority='1 very high'">6</xsl:when>
      <xsl:when test="Priority='2 high'">7</xsl:when>
      <xsl:when test="Priority='3 normal'">1</xsl:when>
      <xsl:when test="Priority='4 low'">5</xsl:when>
      <xsl:when test="Priority='5 very low'">8</xsl:when>
      <xsl:otherwise>1</xsl:otherwise>
    </xsl:choose>
  </priority_id>

  <!-- compose subject "<TicketTitle> (T#<TicketNumber>)" -->
  <subject>
    <xsl:value-of select="Title"/>
    <xsl:text disable-output-escaping="yes">(T#</xsl:text>
    <xsl:value-of select="TicketNumber"/>)
  </subject>

  <!-- just submit first article and a timestamp-->
  <description>
    <xsl:value-of select="Articles[1]/Body"/>
    <TimeStamp>
    <xsl:value-of select="KIX:TimeStamp()"/>
  </description>

</issue>
<!-- ...E0 custom part. -->

</root>
</xsl:template>
</xsl:stylesheet>

```

- Force Array Tags: -
- Suppress Empty: Empty Hash
- Debug: no

5. Macro Action 4 - Verarbeitung der Response

- Action: XSL Transformation
- TransformedData: RMResponsePrepared
- Data: `${RMResponse}`
- XSL Template

XSL Template

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/
Transform" xmlns:KIX="urn:KIX">
  <xsl:output method="xml" encoding="utf-8" indent="yes"/>
  <xsl:template match="RootElement">
    <root>

      <xsl:variable name="HTTPCode" select="HTTPCode"/>
      <xsl:choose>
        <xsl:when test="$HTTPCode='201'">
          <RMIssueID><xsl:value-of select="Result/issue/id"/></RMIssueID>
        </xsl:when>
        <xsl:otherwise>
          <RMIssueID><xsl:text>Not transfered to Redmine.</xsl:text></
RMIssueID>
        </xsl:otherwise>
      </xsl:choose>

    </root>
  </xsl:template>
</xsl:stylesheet>
```

- Force Array Tags: -
- Suppress Empty: Empty Hash
- Debug: no

6. Macro Action 5 - Speichern der Redmine-IssueID

- Action: Set Dynamic Field
- Dynamic Field Name: "RedmineIssueID"
- Dynamic Field Value: `${RMResponsePrepared.RMIssueID}`

Info

Zur Fehleranalyse können Sie den Macro-Action-Parameter "Debug" aktivieren. Folgendes XSL-Template sorgt dafür, dass die eingehenden Parameter direkt, d. h. ohne Transformation, wieder ausgegeben werden:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:KIX="urn:KIX">
  <xsl:output method="xml" encoding="utf-8" indent="yes"/>
  <xsl:template match="/">
    <xmp>
      <xsl:copy-of select="*" />
    </xmp>
  </xsl:template>
</xsl:stylesheet>
```

Verweise

- <https://www.redmine.org>
- https://www.redmine.org/projects/redmine/wiki/Rest_api
- https://www.redmine.org/projects/redmine/wiki/Rest_api#Authentication

Post Attachment as Form Data

Die Macro Action "Post Attachment as Form Data" ermöglicht das Übermitteln von Anhängen an andere Webservices, die Multipart/FormData empfangen können. Beispielsweise, um aus einem Ticket oder bei Anlage eines Tickets ein Jira-Issue zu erstellen, welches auch die Anhänge enthält, die am Artikel hinterlegt sind.

Parameter

Parameter	Beschreibung	Beispiel
Response	Variablenname für die Ergebnisstruktur des Webrequests. Es wird ein strukturiertes Objekt zurückgegeben, welches HTTPCode, Status, Content, Header (Key-Value-Paare) und Result mit dem Ergebnis als JSON-Objekt enthält.	MyAttachmentResponse
URL	Die aufzurufende URL. Es werden KIX-Platzhalter unterstützt.	http://192.168.188.48:3001/rest/api/2/issue/KIX-123/attachments
Use Proxy	Soll ein Proxy-Server angesprochen werden?	yes no
Proxy	Welcher Proxy Server soll angesprochen werden? Ist nichts angegeben, wird die allgemeine Proxy-Einstellung aus der Systemkonfiguration verwendet (Option "WebUserAgent::Proxy").	http://proxy.example.com:3128
Headers	zu übermittelnde HTTP-Header	X-Atlassian-Token: nocheck Authorization: Basic VXNlck5hbWU6UGFzc3dvcmQ=

Parameter	Beschreibung	Beispiel
Attachment Data	<p>Der zu übermittelnde Inhalt. Hier wird i.d.R. eine Macro Variable verwendet, die ein Objekt mit den Strukturen Content, ContentType und Filename enthalten muss, z.B.:</p> <pre>SomeMacroVariableWithStructure = { "Content": "<base64 encoded content here>", "ContentType": "image/png", "Filename": "sample.png" }</pre> <p>Weiterführende Hinweise: siehe auch Macro Actions "Objekt zusammenstellen", "Bericht erstellen" (siehe Seite 347)</p>	<pre>\$ {SomeMacroVariableWithStructure}</pre>
Debug	<p>Erweitertes Logging für den Aufruf de-/aktivieren. Tipp: Debug-Informationen können auch der Job-Historie entnommen werden.</p>	yes no

Variablen und Erweiterte Filter

Use Case: Es wird ein spezifischer Endpoint konfiguriert, der ein Array von Assets, Kontakten, Organisationen oder Ticketdaten entgegen nimmt. Diese Daten werden in einer Macro Action "Schleife" verarbeitet um daraus jeweils das betreffende KIX-Objekt zu erstellen (Asset, Kontakt, Organisation oder Ticket). Die IDs der innerhalb der Schleife zu erstellenden Objekte sollen gesammelt werden, um nach der Abarbeitung **eine** Sammelantwort/Response mit allen erzeugten IDs zurück zu geben. Zum Sammeln der IDs wird ein Array benötigt.

Anwendung in Macros

Macro-Variablen können mehrere Werte aufnehmen und somit ein Array bilden. Die einzelnen Werte werden dabei durch Komma getrennt (s. auch Variablen):

```
#{Variable1,Variable2}
```

Ist einer der übergebenen Parameter bereits ein Array, so werden die beiden Arrays konkateniert:

```
Variable1 := ['Test1.1','Test1.2']
Variable2 := ['Test2.1','Test2.2']
#{Variable1,Variable2} == ['Test1.1','Test1.2','Test2.1','Test2.2']
```

Erweiterte Operationen/Filter

Für erweiterte Array-Operationen werden mit KIX Connect Webservice zusätzliche Variablen-Filter bereit gestellt.

Filter	Beschreibung	Beispiele
VariableUtil.Push	Wandelt die Variablen in ein Array (falls es noch keines ist) und fügt den übergebenen Wert am Ende des Arrays an.	Variable1 := ['Test1.1','Test1.2'] Variable2 := ['Test2.1','Test2.2'] #{Variable1} VariableUtil.Push(Variable2) == ['Test1.1','Test1.2',['Test2.1','Test2.2']]
VariableUtil.Pop	Wandelt die Variablen in ein Array (falls es noch keines ist) und entfernt den Wert am Ende des Array	Variable1 := ['Test1.1','Test1.2'] #{Variable1 Pop} == ['Test1.1']
VariableUtil.Shift	Wandelt die Variablen in ein Array (falls es noch keines ist) und entfernt den Wert am Anfang des Array	Variable1 := ['Test1.1','Test1.2'] #{Variable1 Shift} == ['Test1.2']

Filter	Beschreibung	Beispiele
VariableUtil.Unshift	Wandelt die Variablen in ein Array (falls es noch keines ist) und fügt den übergebenen Wert am Anfang des Arrays an	<pre>Variable1 := ['Test1.1','Test1.2'] Variable1 := ['Test2.1','Test2.2'] \${Variable1} VariableUtil.Unshift(Variable2) == [['Test2.1','Test2.2'],'Test1.1','Test1.2,']</pre>

⚠ Hinweis

Die Funktionen verändern nicht die eigentliche Variable, sondern gelten nur für den entsprechenden Aufruf! Soll das Ergebnis persistent sein, muss die Macro Action "Variable setzen" verwendet werden.

10.2 Add-on "ITIL Practices"

Zum effektiven Arbeiten mit Tickets bedarf es Möglichkeiten, über Tickets zu kommunizieren sowie Informationen auszutauschen und zu erfassen. KIX in Verbindung mit dem Add-on "ITIL Practices" liefert Ihnen die notwendigen Mittel zur Automatisierung bestimmter Abläufe, bei denen Aufgaben - basierend auf Ereignissen, Zeit und definierten Prozessen - abgearbeitet werden. Für jeden Tickettyp existieren passende Status und mögliche Statusübergänge – sowohl automatisch als auch manuell. Das dahinter liegende Prozessmanagement führt Kundennutzer und Agenten durch die Ticketerstellung und stellt sicher, dass Tickets jederzeit definierte Abläufe durchlaufen und am Ende abgeschlossen werden. Wird also in einem Ticket ein bestimmter Status ausgewählt, erfolgen automatisch weitere Prozessschritte.



KIX Pro 18 wurde als SERVIEW CERTIFIED TOOL für 15 von 19 möglichen ITIL® 4-Practices ausgezeichnet. Um allen Anwendern den Einsatz von ITIL® 4 so einfach wie möglich zu machen, wurde das Add-on "**ITIL® Practices**" entwickelt.

"ITIL Practices" ist ein Add-on für KIX Pro 18 und wird ab Version 26 unterstützt. Es umfasst eine Reihe bereits vorkonfigurierter Berichte, Assetklassen, FAQ-Kategorien, Ticketvorlagen und Aktionen, mit denen Problemfälle, Services und Managementaufgaben nach ITIL®4 bearbeitet werden können. Das Add-on stellt sicher, dass alle erforderlichen Informationen am richtigen Ort verfügbar sind und die Ansprechpartner jederzeit über ihre Aufgaben und die weiteren Arbeitsschritte

informiert sind.

Die Nutzung von Prozessen macht Arbeitsabläufe nachvollziehbar und verringert Fehlerquellen. Zudem haben Agenten und Kunden jederzeit alle Informationen im gesamten Prozessverlauf im Blick. Die daraus resultierende Verkürzung von Bearbeitungs- und Durchlaufzeiten sorgt für eine Erhöhung der Produktivität bei gleichzeitiger Reduzierung der Kosten. Die auf Sie zugeschnittene Planung und Steuerung verkürzt Stillstandzeiten, reduziert Lagerbestände und spart somit weitere Kosten ein.

Für folgende Prozesse bringt das Add-on "ITIL Practices" bereits vorkonfigurierte Ticketvorlagen und Aktionen mit:

- Serviceanfragen / Service Requests
- Störungsanzeigen / Incidents
- Problem-Meldungen / Problems
- Änderungen / Changes
- Notfallpläne / Emergency Plans

10.2.1 Video

Einen ausführlichen Showcase zum Add-on "ITIL Practices" finden Sie auch in unserem Youtube-Kanal:

<https://www.youtube.com/watch?v=M8Ds6TGIEWo> 



10.2.2 Aktivieren/Deaktivieren von "ITIL Practices"

10.2.2.1 Voraussetzung

Als Grundlage für die Nutzung des Add-ons "ITIL Practices" benötigen Sie KIX 18 Pro, welches mindestens auf Version 26 läuft.

10.2.2.2 Installation/System-Update

Nach der Beauftragung stellen wir Ihnen - in der Regel am nächsten Arbeitstag - ein aktualisiertes Image Ihres Systems bereit.

Bitte führen Sie, falls Sie KIX Pro on-premises betreiben, anschließend ein System-Update durch:

```
user@DockerHost:/opt/kix-on-premise/deploy/linux# ./stop.sh  
user@DockerHost:/opt/kix-on-premise/deploy/linux# ./update.sh
```

Danach ist das Add-on in Ihr System integriert und kann durch Sie aktiviert werden.

Die Bereitstellung des Add-ons in der KIX Cloud erfolgt über unser Support-Team.

10.2.2.3 Das Add-on "ITIL Practices" aktivieren

Nach dem System-Update bzw. nach Bereitstellung in der Cloud finden Sie das Add-on "ITIL Practices" im Explorer des Admin-Moduls. Klicken Sie dazu auf der Willkommenseite auf "Aktivieren/Activate". Es erfolgt eine Sicherheitsabfrage, ob Sie die Konfiguration des Add-ons jetzt aktivieren möchten. Bestätigen Sie diese.

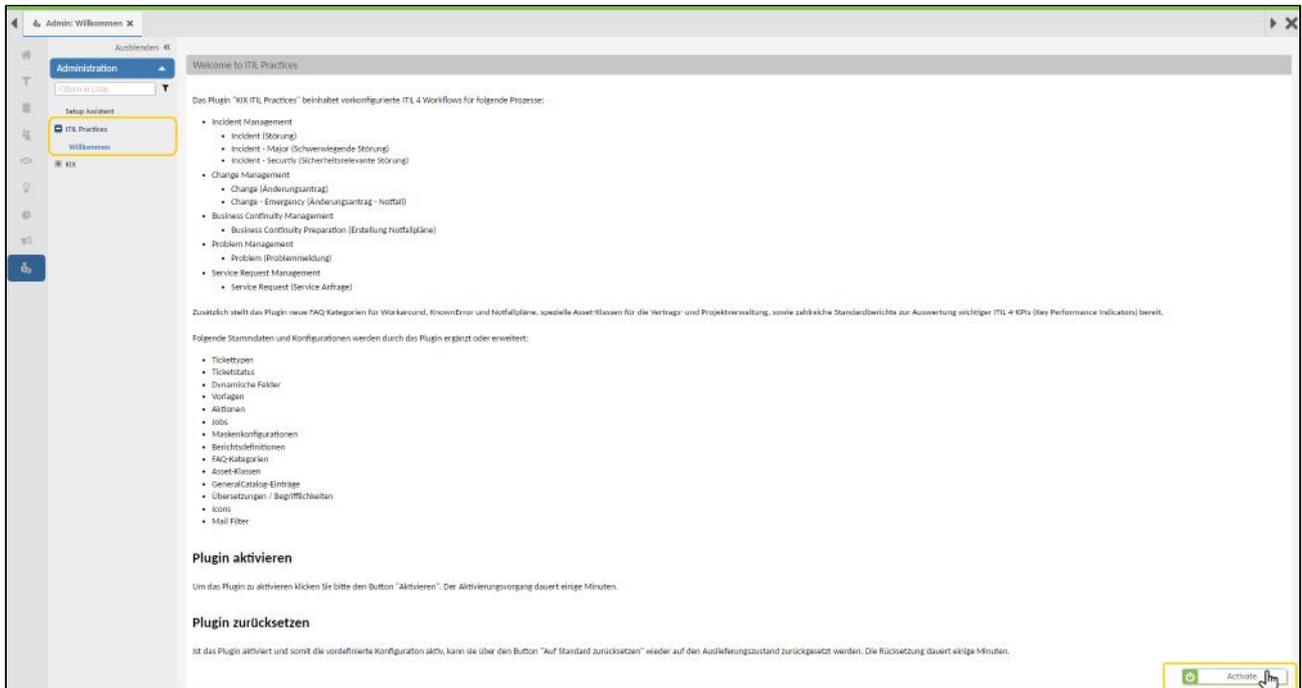


Abb.: Add-on "ITIL Practices" aktivieren

Anschließend startet KIX den Import. Dieser Vorgang kann einige Minuten dauern. Die Installationsroutine zeigt Ihnen, welche Pakete importiert werden. Abschließend wird kurz eine Erfolgsmeldung eingeblendet.

Danach ist das Add-on "ITIL Practices" in KIX integriert. Weitere allgemeine Schritte sind nicht erforderlich. Die mit dem Add-on ausgelieferten Bestandteile können Sie direkt verwenden oder mit nur wenigen Schritten an Ihren spezifischen Anwendungsfall anpassen.

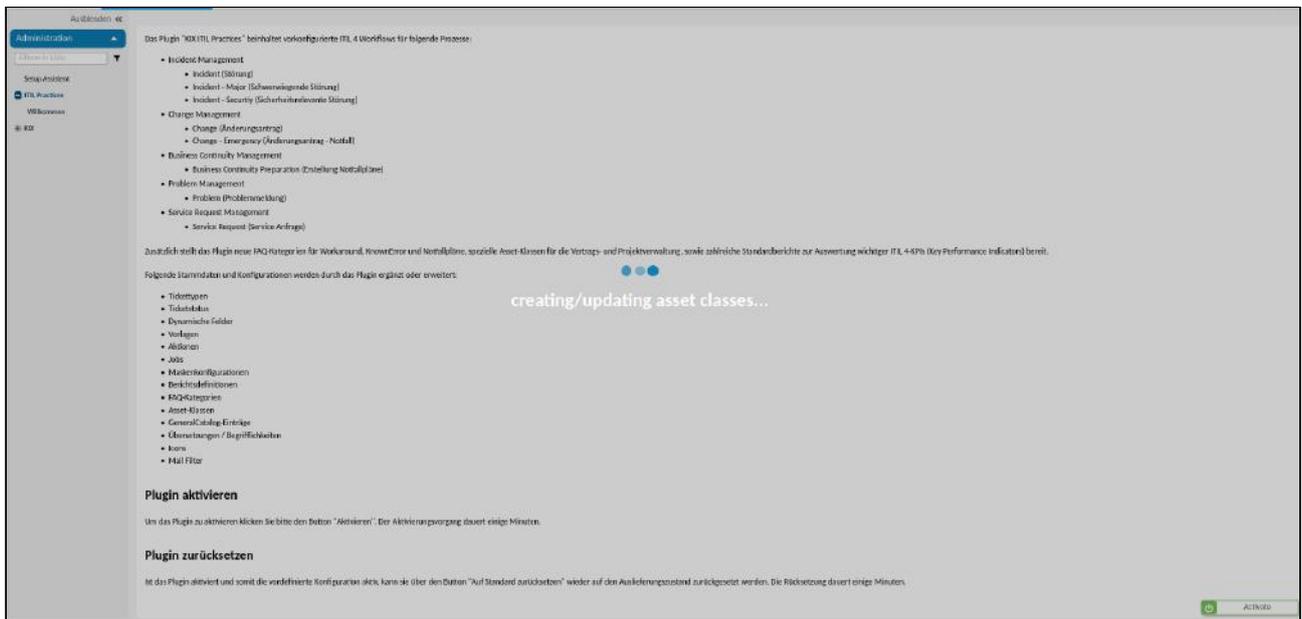


Abb.: Add-on "ITIL Practices" Setup

Mit der Aktivierung des Add-ons wird KIX um weitere Konfigurationen, Objekte und Stammdaten ergänzt, die für die Prozesse erforderlich sind. Zudem werden bestehende Konfigurationen angepasst.

 **Wichtig!**

Bei der Aktivierung werden verschiedene Standard-Konfigurationen überschrieben. Wenn Sie bereits Standard-Konfigurationen geändert haben, **sichern Sie diese bitte vorher!**

Folgende Stammdaten und Konfigurationen werden durch das Add-on ergänzt oder erweitert:

- Tickettypen
- Ticketstatus
- Dynamische Felder
- Vorlagen
- Aktionen
- Jobs
- Ansichtskonfigurationen
- Berichtsdefinitionen
- FAQ-Kategorien
- Asset-Klassen
- General Catalog-Einträge
- Übersetzungen / Begrifflichkeiten
- Icons
- E-Mail-Filter

10.2.2.4 Das Add-on "ITIL Practices" zurücksetzen

Sie können das Add-on "ITIL Practices" auf den Auslieferungszustand zurücksetzen. Klicken Sie dazu auf "Auf Vorgabe zurücksetzen". Das Zurücksetzen des Systems dauert einige Minuten.

Alle Konfigurationen, die durch das Add-On ausgeliefert werden und einen unveränderten Bezeichner/Namen haben, werden auf die Vorgabewerte des Add-Ons zurückgesetzt. Ausgenommen davon sind Übersetzungen und Rollen-Zuordnungen (betrifft Berichte, Ticketvorlagen und Aktionen). Erstellte Tickets, Artikel und Berichte usw. bleiben unverändert erhalten.

10.2.3 Umsetzung der ITIL Practices mittels Prozessen

Das Add-on "ITIL Practices" stellt eine Reihe bereits fertig konfigurierter Berichte, Ticketvorlagen und Aktionen bereit, mit denen Problemfälle, Servicemeldungen und Änderungsanträge nach ITIL®4 bearbeitet werden können. Eine Übersicht dazu finden Sie im Kapitel "Erweiterte Konfiguration".

Folgende vorbereitete ITIL®4 Workflows (Prozesse) werden mit dem Add-on ausgeliefert:

- Serviceanfrage / Service Request Management
 - Service Request (Service Anfrage)
- Störung / Incident Management
 - Incident (Störung)
 - Incident - Major (Schwerwiegende Störung)
 - Incident - Security (Sicherheitsrelevante Störung)
- Problemmeldung / Problem Management
 - Problem (Problem-Meldung)
- Änderungsantrag / Change Management
 - Change (Änderungsantrag)
 - Change - Emergency (Änderungsantrag - Notfall)
- Notfallplan / Business Continuity Management
 - Business Continuity Preparation (Erstellung Notfallpläne)

10.2.3.1 Ticketvorlagen

Das Add-on beinhaltet für jeden der o. g. Prozesse eine separat konfigurierte Ticketvorlage. Diese finden Sie beim Anlegen eines neuen Tickets unter der Vorlagengruppe "IT Aufträge". Die gewählte Vorlage bestimmt den weiteren Prozessverlauf.

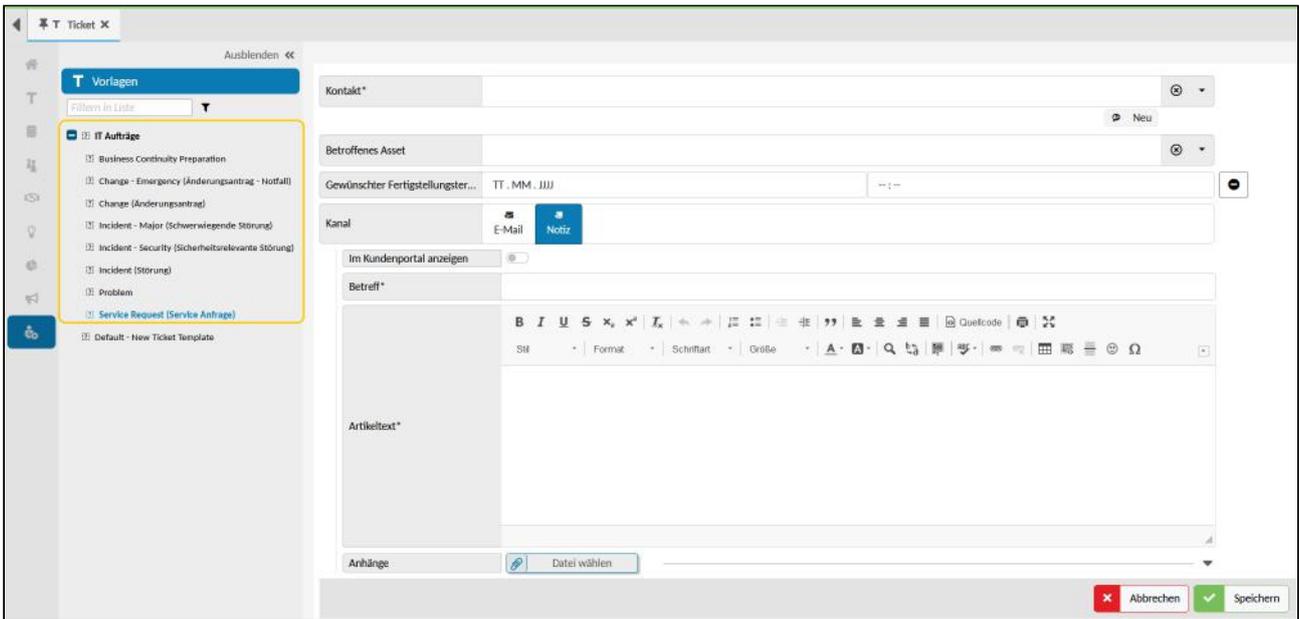


Abb.: Ticketvorlagen für ITIL Practices im Agentenportal

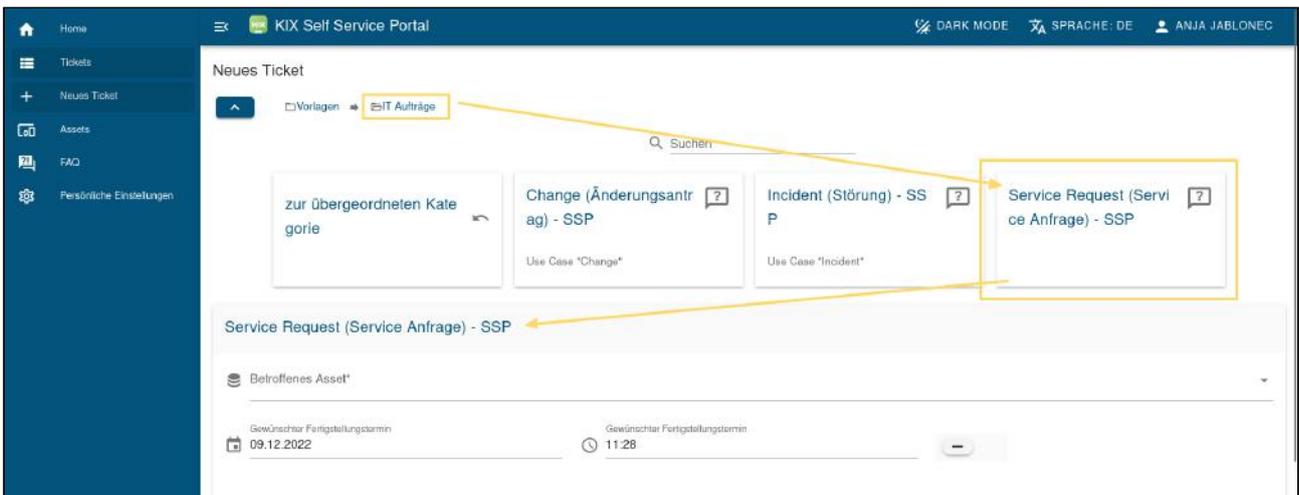


Abb.: Ticketvorlagen, die das Add-on "ITIL Practices" im SSP mitbringt

Eine Übersicht über die im Add-on enthaltenen Ticketvorlagen sowie weitere Hinweise finden Sie unter: [Vorlagen für ITIL Practices \(siehe Seite 407\)](#)

10.2.3.2 Ticketaktionen

Das Add-on "ITIL Practices" ändert die Konfigurationen verschiedener Standard-Ticketaktionen. So werden bspw. die Aktionen "Ticket bearbeiten" oder "Schließen" ausgeblendet, um einen korrekter Prozessverlauf sicherzustellen.

Zudem implementiert "ITIL Practices" eine Reihe von Ticketaktionen, die für die Bearbeitung erforderlich sind. Wann und unter welchen Voraussetzungen eine Aktion am Ticket zur Verfügung steht, ist vom gewählten Prozess und vom Prozessverlauf abhängig. Der Prozessverlauf orientiert sich zudem am Status des Tickets. Das Add-on bringt dafür mehrere neue Ticketstatus mit.

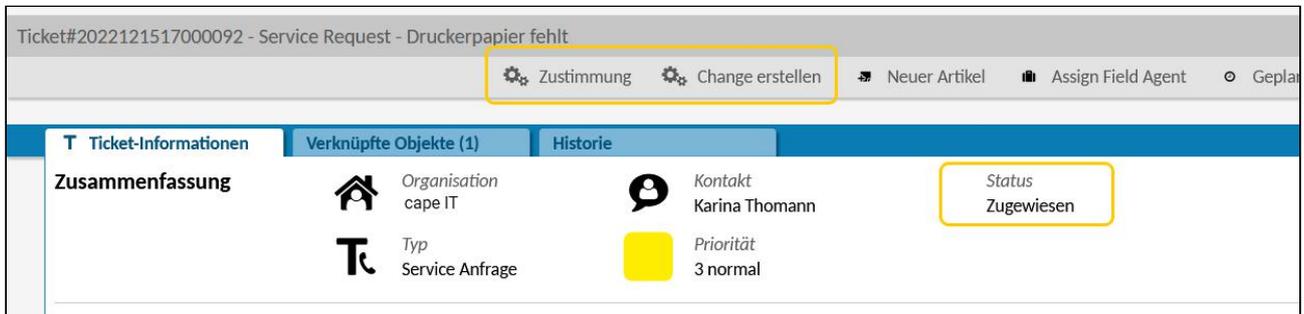


Abb.: Vom Ticketstatus abhängende Ticketaktionen

Die Konfiguration der Aktionen berücksichtigt im Auslieferungszustand nur Standardnutzer. Sie können die Aktionen jedoch individuell anpassen, um Berechtigungen einzuschränken bzw. zu erweitern oder auch, um weitere Dynamische Felder in Dialogen zu integrieren.

Eine Übersicht über die im Add-on enthaltenen Ticketaktionen finden Sie unter: [Aktionen für ITIL Practices](#) (siehe Seite 399)

Bearbeitungsstand

Die mit dem Add-on ausgelieferten dialogbasierte Ticketaktionen enthalten alle das Feld "Bearbeitungsstand". Der Bearbeitungsstand ist ein Dynamisches Feld vom Typ Checkliste. Jeder Punkt auf der Checkliste entspricht einem Prozessschritt, der unter Angabe eines Zustands beantwortet werden kann. Je nach gewähltem Zustand ändert sich der Bearbeitungsstand (erkennbar am Fortschrittsbalken) und der Ticketstatus.

Status	Bedeutung	Bearbeitungsstand
OK	Okay / Erledigt / Genehmigt	Wird hochgezählt, da gültige Antwort.
NOK	Nicht Okay / Nicht ausgeführt / Nicht genehmigt	Wird hochgezählt, da gültige Antwort.
PENDING	Warten-Status / Warten auf Rückmeldung / Zurückgestellt	Wird nicht hochgezählt, da Schwebezustand.
n.a.	keine Angabe / Angabe nicht erforderlich / keine Aussage getroffen	Wird hochgezählt, da z. B. "keine Aussage getroffen" auch eine gültige Antwort ist.

Status	Bedeutung	Bearbeitungsstand
-	unbeantwortet / keine Auswahl am Checklistenpunkt getroffen	Wird nicht hochgezählt.

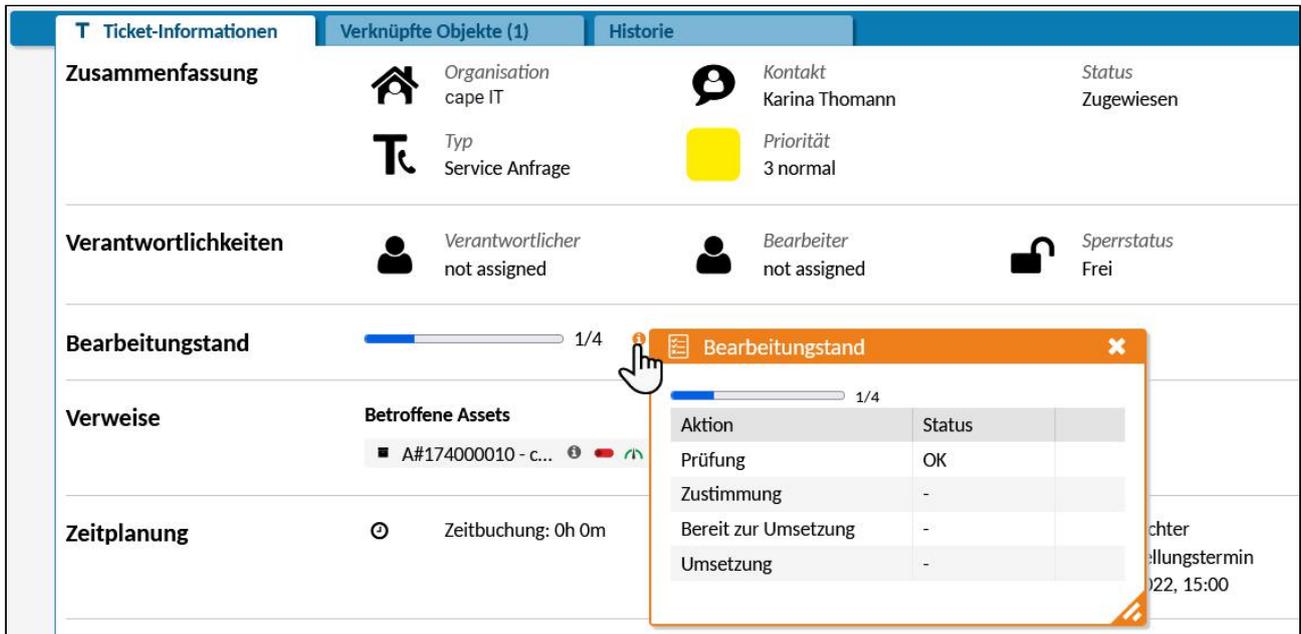
Im Beispiel einer Serviceanfrage erfolgt zunächst die Prüfung des Sachverhalts. Um diesen zu bestätigen, erhält der Checklistenpunkt "Prüfung" den Zustand OK. Der Prozessschritt der Prüfung ist somit erledigt und das Ticket wechselt in den nächsten Prozessschritt. Analog verhält es sich bei allen weiteren Prozessschritten (Checklistenpunkten), sodass das Ticket im Prozessverlauf vorangetrieben wird. Sind alle Prozessschritte abgearbeitet, kann das Ticket abgeschlossen werden. Wird ein Prozessschritt mit NOK, -, n.a. oder mit PENDING beantwortet, so wird der Prozess nicht mehr vorangetrieben.

Die am Ticket verfügbaren Aktionen sind abhängig vom jeweiligen Bearbeitungsstand. Das bedeutet, am Ticket stehen stets die Aktionen bereit, die für den jeweiligen Prozessschritt relevant sind (s. o. Ticketaktionen).

Hinweis: Der Prozessverlauf kann nicht rückgängig gemacht werden. Eine Änderung am Status des Tickets ist rückwirkend nicht möglich; der gespeicherte Bearbeitungsstand bleibt erhalten, auch wenn die einzelnen Zustände geändert werden. Damit wird ein korrekter Prozessverlauf sichergestellt. Ausnahme: Die Ablehnung einer Lösung führt zurück zum Prozessschritt "Zustimmung".

⚠ Für jeden Prozess bringt das Add-on "ITIL Practices" eine separate Checkliste "Bearbeitungsstand" mit. Die Konfigurationen dieser Checklisten dürfen **nicht geändert** werden. Anderenfalls funktionieren die daran gekoppelten Jobs nicht mehr korrekt, sodass der gesamte Prozessverlauf zerstört wird.

Der Bearbeitungsstand eines Tickets ist auch am Fortschrittsbalken in der Ticket Detailansicht ersichtlich. Ein Klick auf das Info-Symbol zeigt die Details an.



The screenshot shows the 'Ticket-Informationen' tab with various fields. The 'Bearbeitungsstand' section features a progress bar at 1/4. A modal window titled 'Bearbeitungsstand' is open, showing a table of actions and their statuses.

Aktion	Status
Prüfung	OK
Zustimmung	-
Bereit zur Umsetzung	-
Umsetzung	-

Abb.: Bearbeitungsstand eines Tickets in der Ticket Detailansicht

10.2.3.3 Berechtigungen

Die mit dem Add-on bereitgestellten Konfigurationen für Jobs, Aktionen, Berichtsdefinitionen etc. verwenden stets die initialen Standardwerte (Rollen, Nutzer, Teams usw.). Gegebenenfalls müssen Sie Werte in den Konfigurationen anpassen, bspw. um zusätzliche Berechtigungen zu erteilen.

Weiterführende Hinweise finden Sie auch im den Kapitel "Erweiterte Konfiguration".



10.2.3.4 Prozess 1: Serviceanfrage

Bei einer Serviceanfrage handelt es sich i.d.R. um eine Unterbrechung einer Serviceverfügbarkeit. Eine Serviceanfrage kann bspw. sein:

- Die Erledigung von Aufgaben durch andere Abteilungen
 - Bereitstellung von Verbrauchsmaterialien
 - Beschaffung von Hardware
- Kundenanfragen
 - das Zurücksetzen eines Kennworts
 - Zugriff auf Anwendungen.

Ablaufschema einer Serviceanfrage

KIX behandelt eine Serviceanfrage nach folgendem Schema und stellt mit dem Add-on "ITIL Practices" die dafür benötigten Ticketvorlagen, Ticketstatus, Aktionen etc. bereit.

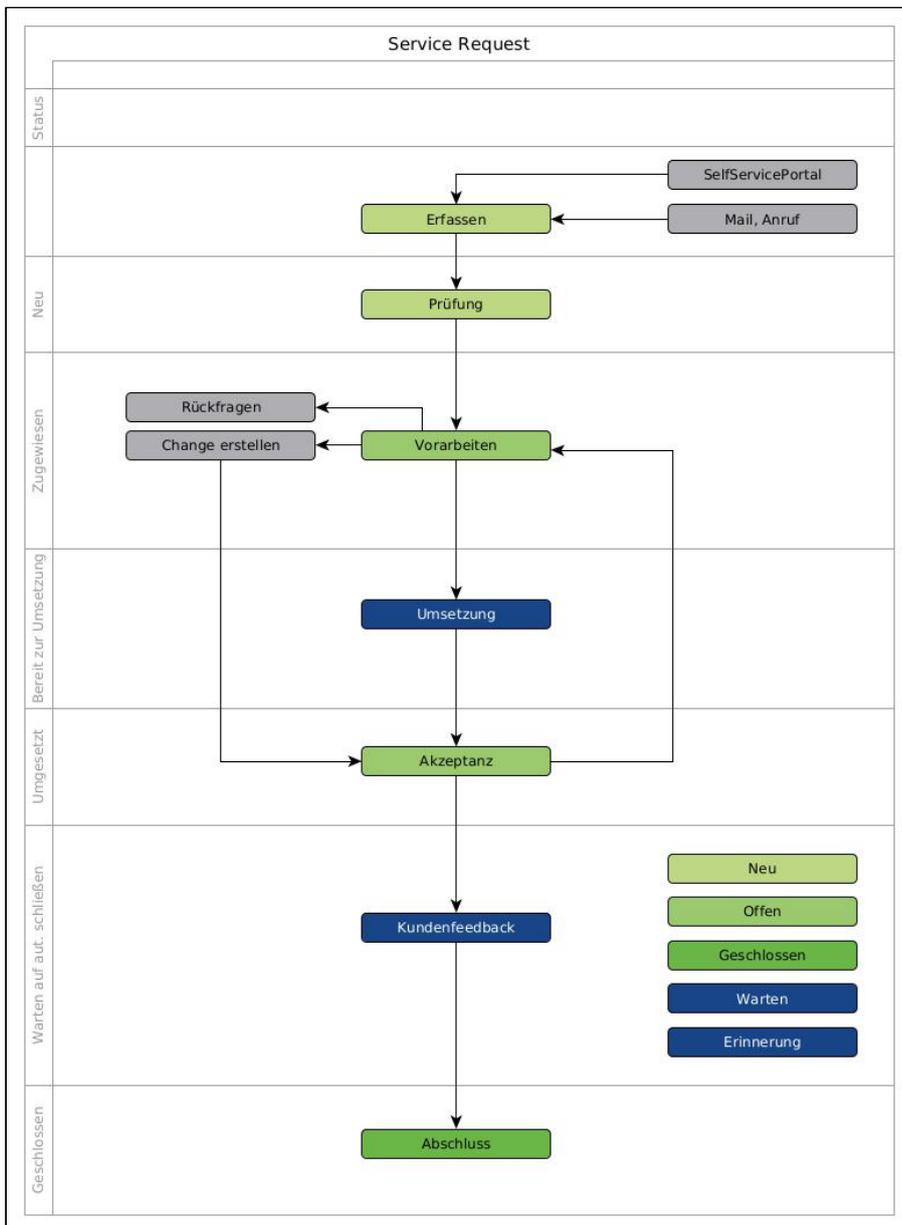


Abb.: Ablaufschema des Prozesses "Service Anfrage"

1. Erfassen

Zunächst wird die Serviceanfrage (z. B. Bereitstellung von Verbrauchsmaterial) erfasst und ein neues Ticket dazu erstellt. Zur Erfassung der Serviceanfrage dient die Ticketvorlage "Service Request (Service Anfrage)".

Mit dem Speichern wechselt das Ticket in den nächsten Prozessschritt "Prüfung" und erhält die Default-Werte für Priorität (normal) und Status (neu).

2. Prüfung

Bei der Prüfung wird der Sachverhalt geprüft und dem zuständigen Bearbeiter zugewiesen. Dazu steht die Aktion "Unter Review" am Ticket zur Verfügung. Sie öffnet einen Dialog, in dem u. a. der zuständige Bearbeiter ausgewählt und die Priorität neu gesetzt werden kann.

Erhält der Prozessschritt "Prüfung" den Zustand OK, wird mit dem Speichern das Ticket

- in den nächsten Prozessschritt "Vorarbeiten" überführt
- dem Team des gewählten Bearbeiters zugewiesen
- der Ticketstatus auf "zugewiesen" gesetzt.

3. Vorarbeiten

Ist die Serviceanfrage zugewiesen, kann deren Umsetzung vorbereitet werden. Möglicherweise ist eine Beschaffung und somit ein weiteres Ticket (Kind) erforderlich (s. unten "Abzweig zum Change Prozess"). Das Ticket befindet sich so lange in einem Warten-Status. Ist die Beschaffung erfolgt, kann auch die Umsetzung erfolgen. Dazu steht am Ticket die Aktion "Bereit zur Umsetzung" zur Verfügung (One-Klick-Aktion ohne Dialog). Die Verwendung der Aktion gibt das OK für die Umsetzung, sodass das Ticket in den nächsten Prozessschritt überführt wird. Der Ticketstatus bleibt dabei auf "Vorarbeiten".

4. Umsetzung

Sind die nötigen Vorarbeiten (z. B. Bestellung des Verbrauchsmaterials) abgeschlossen, kann die Serviceanfrage umgesetzt werden. Am Ticket steht dafür die Aktion "Umsetzung" zur Verfügung. Sie öffnet einen Dialog, um das Ergebnis und/oder Notizen zur Umsetzung am Ticket zu hinterlegen. Beispielsweise, um bei einer fehlgeschlagenen Umsetzung die Gründe dafür zu notieren.

Wird die Umsetzung mit OK beantwortet, erhält das Ticket den Status "umgesetzt". Im nächsten Prozessschritt kann die Lösung akzeptiert oder abgelehnt werden.

5. Akzeptanz

Ist die Umsetzung erfolgt (z. B. das Verbrauchsmaterial ist ausgeliefert), kann die Lösung akzeptiert (z. B. richtiges Verbrauchsmaterial) oder aber auch abgelehnt (z. B. falsches Verbrauchsmaterial) werden. Am Ticket stehen dazu die gleichnamigen Ticketaktionen zur Verfügung.

Wird die Lösung abgelehnt, wird das Ticket auf den 3. Prozessschritt (Bereit zur Umsetzung) zurückgesetzt, sodass der Prozessverlauf erneut starten kann.

Wird die Lösung akzeptiert, gilt der Prozess als abgeschlossen. An dieser Stelle kann ein Kundenfeedback eingeholt werden. Das Ticket wird in den Status "warten auf automatisch schließen" gesetzt.



Abzweig zum Change Prozess

Wird am Ticket die Aktion "Change erstellen" bereitgestellt, ermöglicht dies einen Prozess-Abzweig. Bspw. wenn für eine Serviceanfrage Beschaffungsmaßnahmen (z. B. Bestellung von Verbrauchsmaterial) erforderlich sind. Die Aktion erstellt ein weiteres Ticket (Kind) und verknüpft dieses mit dem Quellticket (Eltern).

10.2.3.5 Prozess 2: Störung

Beim Prozess einer Störung handelt es sich i.d.R. um die Unterbrechung eines Services oder um ein ungeplantes Ereignis, welches zum Ausfall eines Services führt. Ziel ist es, den normalen Servicebetrieb schnellstmöglich wieder herzustellen.

Eine Störung kann bspw. sein:

- Funktionsbeeinträchtigungen eines Webservers
- eine unterbrochene Strom- oder Datenverbindung
- ein von einem Gerät gemeldete Fehlermeldung
- Ausfall einer Geschäftsanwendung
- ein Cyberangriff
- defekte Hardware oder Betriebsmittel
- u.v.a.m.

Ablaufschema einer Störung

Wird eine Störung gemeldet, kann in KIX ein Ticket nach dem Prozess "Störung/Incident" angelegt werden. Der Prozess einer Störung verläuft dabei nach folgendem Schema:

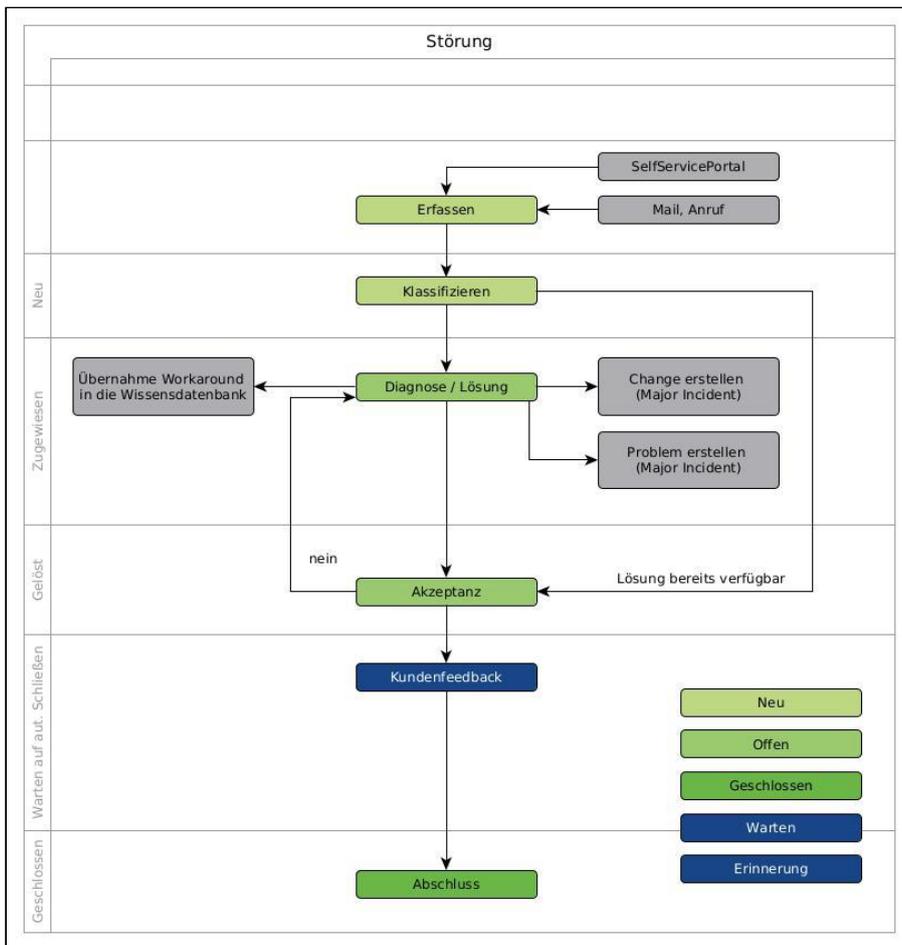


Abb.: Ablaufschema des Prozesses "Störung"

1. Erfassen

Das Add-on "ITIL Practices" unterscheidet nach der Art einer Störung:

- Incident (Störung).
- Incident - Major (Schwerwiegende Störung)
- Incident - Security (Sicherheitsrelevante Störung)

Das Add-on stellt zu jeder dieser Störungsarten eine gleichnamige Ticketvorlage bereit.

Mit dem Speichern erhält das Ticket die Default-Werte für Priorität (normal) und Status (neu). Im nächsten Prozessschritt - beim Klassifizieren des Tickets - kann das Ticket anders bewertet werden.

2. Klassifizieren

Durch Klassifizieren wird der Schweregrad der Störung angegeben und die Dringlichkeit des Tickets eingestuft. Diese richtet sich nach den Auswirkungen der Störung. Am Ticket ist dazu die Aktion "Klassifizierung" verfügbar. Im sich öffnenden Dialog können u. a. die Auswirkungen klassifiziert, die Priorität geändert, die Symptome beschrieben und das Ticket dem zuständigen Bearbeiter zugewiesen werden. Die Priorität des Tickets kann nach dem Klassifizieren nicht mehr geändert werden.

Nach dem Speichern erhält das Ticket den Status "zugewiesen" und wechselt in den nächsten Prozessschritt "Diagnose" oder "Lösung".

3. Diagnose

Durch Diagnose wird die Ursache der Störung benannt. Am Ticket ist dazu die Aktion "Diagnose" verfügbar. Ist die Diagnose mit OK bestätigt, erhält das Ticket den Status "in Arbeit". Die Aktion "Diagnose" steht weiterhin am Ticket zur Verfügung, sodass Änderungen bspw. an der Auswirkung oder Ergänzungen zu Symptom/Ursache vorgenommen werden können.

Der Prozessschritt "Diagnose" kann bei vorhandener Lösung übersprungen werden, bspw. wenn die Störung bereits behoben wurde. Verwenden Sie dann die Ticketaktion "Lösung".

4. Lösung

Am Ende des Störungs-Prozesses steht die Lösung. Am Ticket ist dazu eine gleichnamige Ticketaktion verfügbar. Im sich öffnenden Dialog können u. a. angegeben werden:

- Lösungsbeschreibung: z. B. durchgeführte Arbeiten, Hinweise zur Nichtdurchführbarkeit etc.
- Workaround: Beschreibung eines alternativen Vorgehens oder einer Behelfslösung. Ist ein Workaround angegeben, wird automatisch ein FAQ Eintrag erstellt.
- Abschlusscode: Spezifizierung der Lösung (abgeschlossen, nicht abgeschlossen usw.)

Ist ein Abschlusscode gewählt und der Bearbeitungsstand der Lösung mit OK beantwortet, erhält das Ticket den Status "gelöst" und wechselt in den nächsten Prozessschritt "Akzeptanz".

5. Akzeptanz

Ist die Störung gelöst (positiv oder negativ), kann die Lösung abgelehnt oder akzeptiert werden. Am Ticket stehen dazu die gleichnamigen Ticketaktionen zur Verfügung.

- Wird die Lösung akzeptiert, gilt der Prozess als erfolgreich abgeschlossen. Das Ticket wird in den Status "warten auf automatisch schließen" gesetzt.
- Wird die Lösung abgelehnt, wird das Ticket auf den 3. Prozessschritt (Diagnose/Lösung) zurückgesetzt. Das Ticket erhält den Status "zugewiesen", sodass der Prozessverlauf erneut starten kann.



Abzweig zum Change Prozess

Wird am Ticket die Aktion "Change erstellen" bereitgestellt, ermöglicht dies einen Prozess-Abzweig zum Change Management, bspw. wenn Beschaffungsmaßnahmen erforderlich sind (z. B. Bestellung neuer Servertechnik). Die Aktion erstellt ein weiteres Ticket (Kind) und verknüpft dieses mit dem Quellticket (Eltern).

Abzweig zum Problem Prozess

Die Ticketaktion "Problem erstellen" steht bei schwerwiegenden Störungen (Incident - Major) zur Verfügung. Sie ermöglicht einen Prozess-Abzweig zum Problem Management, bspw. wenn ein häufig wiederkehrender Defekt Ursache für die Störung ist. Die Aktion öffnet einen Dialog zum Anlegen eines neuen Artikels und erstellt parallel dazu ein weiteres Ticket vom Typ "Problem".

10.2.3.6 Prozess 3: Problem

Bei einem Problem handelt es sich i.d.R. um die (mögliche) Ursache für (wiederkehrende) Störungen (Incident). Das Problem Management kümmert sich um die Identifizierung und Verwaltung dieser Ursachen. Ziele sind hierbei die dauerhafte Problemlösung und Vermeidung kostspieliger Störungen sowie Sicherung des Wissens über (bekannte) Probleme.

Ein Problem kann bspw. sein:

- eine geänderte Konfigurationsdatei
- Ursachen für einen beschädigten Datenbankeintrag
- Softwarefehler
- u.v.a.m.

Ablaufschema des Prozesses Problem

Liegt basierend auf den Richtwerten von ITIL®4 ein Problem vor, kann in KIX ein entsprechendes Prozessticket "Problem" angelegt werden. Das Problem Management in KIX verläuft dabei nach folgendem Schema:

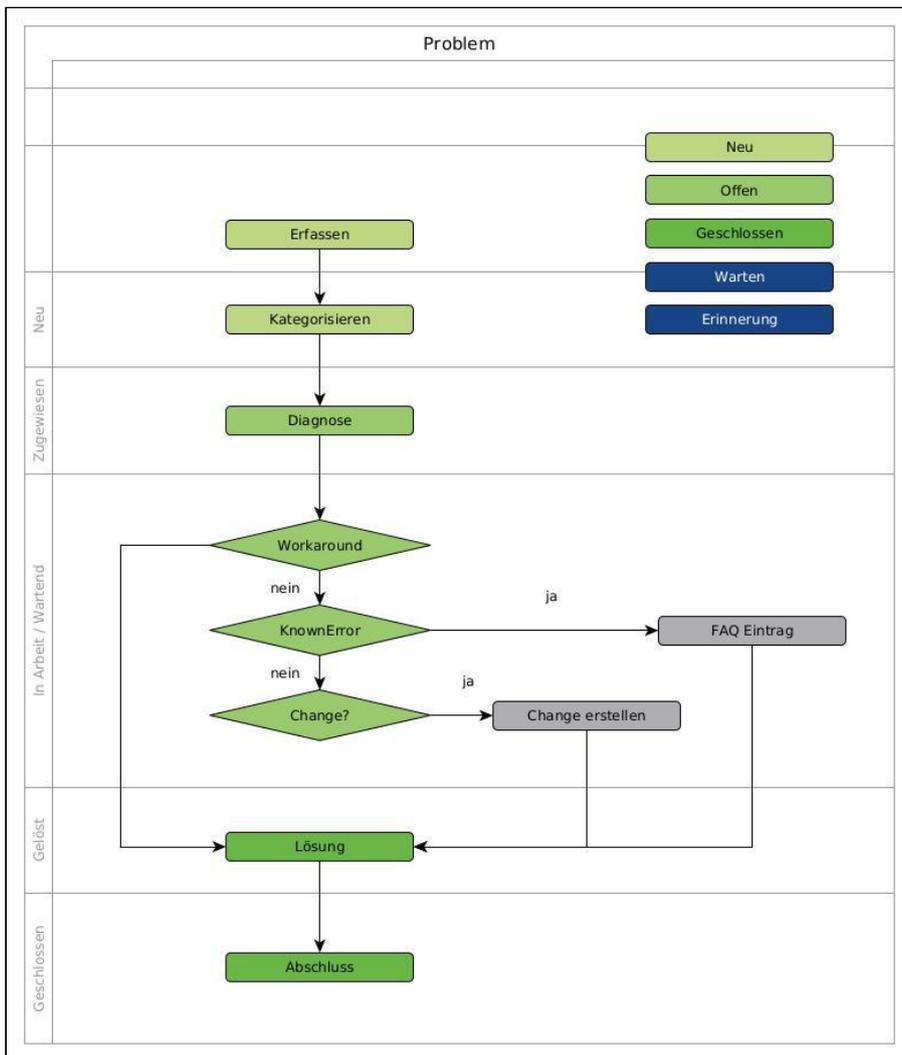


Abb.: Ablaufschema des Prozesses "Problem"

1. Erfassen

Die Erfassung eines Problems kann auf verschiedenen Wegen erfolgen:

- manuell: unter Verwendung der Ticketvorlage "Problem"
- automatisiert: bei Verwendung der Ticketaktion "Problem" (Prozess: schwerwiegende Störung) .

Das erstellte Ticket erhält die Default-Werte für Priorität (normal) und Status (neu) und wechselt in den nächsten Prozessschritt "Kategorisierung".

2. Kategorisieren

Das Kategorisieren des Tickets hilft, das Problem zu bewerten und ihm eine Steuergröße zu geben. Am Ticket existiert dazu die Aktion "Kategorisieren". Sie öffnet einen Dialog indem u. a. Priorität, Auswirkungen

und Fehlerkategorie bestimmt werden können. Die Auswirkungen richten sich nach der Art der Beeinträchtigung des Services. Die Fehlerkategorie dient der Einordnung des Problems in Themenbereiche.

Wird die Kategorisierung mit OK beantwortet, erhält das Ticket den Status "zugewiesen" und wechselt in den nächsten Prozessschritt "Diagnose".

3. Diagnose

Die Diagnose dient dem Identifizieren des Problems und dessen Bestätigung bzw. Ablehnung. Am Ticket existiert dazu die Aktion "Diagnose". Das Ergebnis der Diagnose wird im Feld "Problemstatus" angegeben.

Ist dies erfolgt und die Diagnose mit OK bestätigt, erhält das Ticket mit dem Speichern den Status "in Arbeit".

Die Aktion "Diagnose" steht weiterhin am Ticket zur Verfügung, sodass Änderungen bspw. am Problemstatus oder Ergänzungen zu Symptom/Ursache vorgenommen werden können.

Der Prozessschritt "Diagnose" kann bei vorhandener Lösung übersprungen werden. Verwenden Sie dann die Ticketaktion "Lösung".

4. Known Error

Ein Known Error ist ein Fehler mit bekannter Ursache. Für solche Fehler kann ein FAQ-Eintrag erzeugt werden. Dafür steht am Ticket die Aktion "Known Error" zur Verfügung. Sie erstellt anhand der im Ticket hinterlegten Informationen automatisch - ohne Dialog - einen FAQ Eintrag und verknüpft diesen mit dem Quellticket.

5. Lösung

Konnte das Problem gelöst werden oder existiert ein Workaround zum Problem, kann das Ticket geschlossen werden. Dies erfolgt unter Verwendung der Ticketaktion "Lösung". Im sich öffnenden Dialog können u. a. angegeben werden:

- Lösungsbeschreibung: z. B. durchgeführte Arbeiten, Hinweise zur Nichtdurchführbarkeit etc.
- Workaround: Beschreibung eines alternativen Vorgehens oder einer Behelfslösung. Ist ein Workaround angegeben, wird automatisch ein FAQ Eintrag erstellt.
- Problemstatus: Der Problemstatus muss angegeben werden, wenn der Prozessschritt "Diagnose" übersprungen wurde.

Ist der Problemstatus gesetzt und der Bearbeitungsstand der Lösung mit OK beantwortet, wird das Ticket direkt geschlossen. Der Prozessschritt der Akzeptanz entfällt.

Abzweig zum Change Prozess

Wird am Ticket die Aktion "Change erstellen" bereitgestellt, ermöglicht dies einen Prozess-Abzweig zum Change Management. Die Aktion öffnet einen Dialog zum Anlegen eines neuen Artikels und erstellt parallel dazu ein weiteres Ticket vom Typ "Change". Beide Tickets werden als Eltern-Kind-Beziehung miteinander verknüpft.



10.2.3.7 Prozess 4: Änderungsantrag

Dem Change Management werden organisatorische oder infrastrukturelle Veränderungen zugeordnet, welche direkte oder indirekte Auswirkungen auf Services haben können.

Ein Change kann bspw. sein:

- der Austausch eines Geräts
- Einführung neuer Services
- u. a. m.

Ablaufschema eines Änderungsantrags

KIX behandelt einen Change Prozess nach folgendem Schema und stellt mit dem Add-on "ITIL Practices" die dafür benötigten Ticketvorlagen, Ticketstatus, Aktionen etc. bereit.

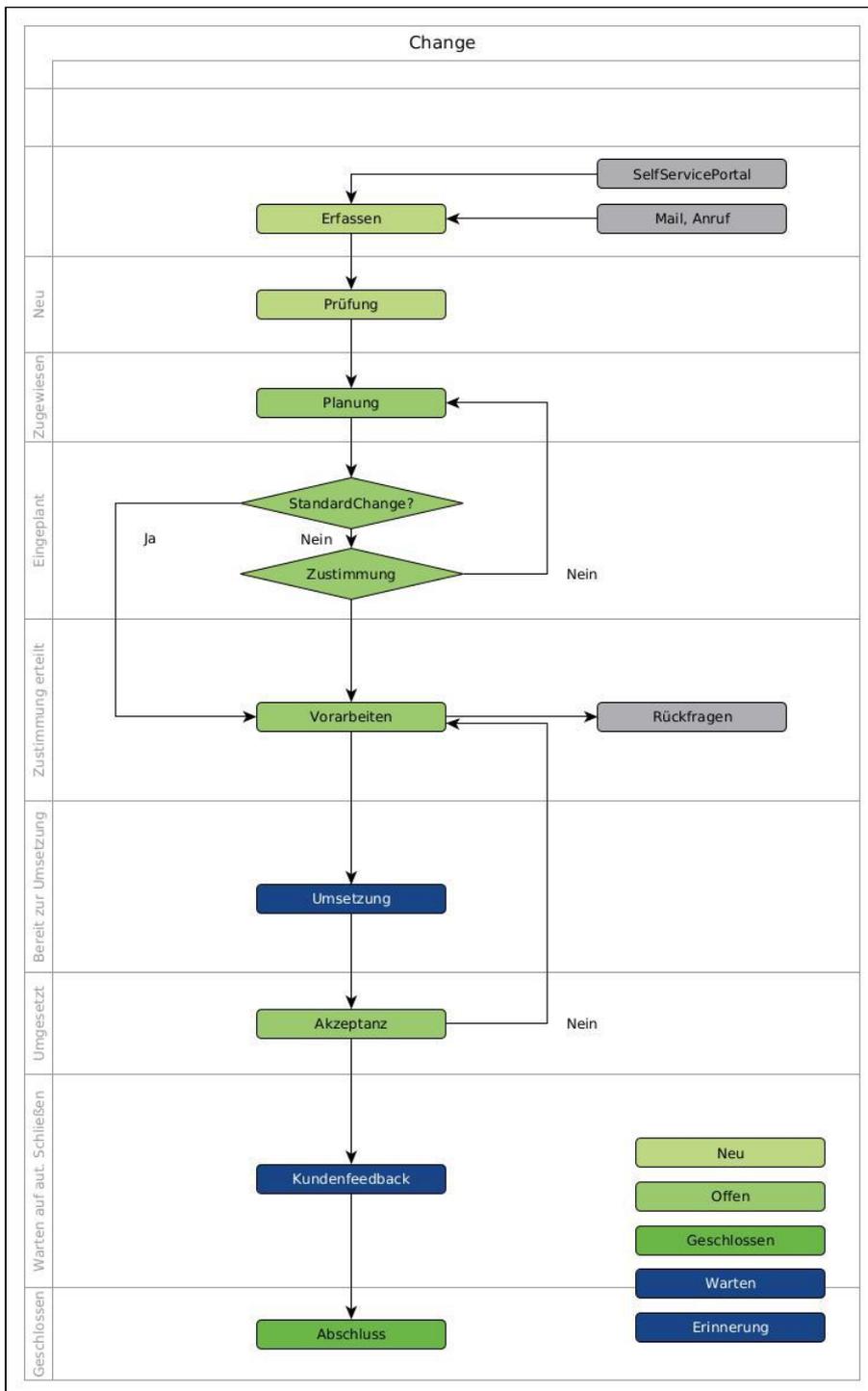


Abb.: Ablaufschema eines "Change" Prozesses in KIX

1. Erfassen

Das Add-on "ITIL Practices" unterscheidet nach der Art eines Änderungsantrags:

- Change (Änderungsantrag): übliche Änderungen mit niedrigem oder erhöhtem Risiko

- Change Emergency (Änderungsantrag - Notfall): Änderungen aufgrund eines unerwarteten Fehlers, der umgehend behoben werden muss

Das Add-on stellt zu jeder dieser Störungsarten eine gleichnamige Ticketvorlage bereit. Der Ablauf ist für beide Change-Prozesse gleich.

Im Bereich Geschäftszenario können

- die *Gründe* für den Antrag
- die *Vorteile*, die die Änderung mitbringt
- voraussichtliche *Umsetzungskosten*
- und mögliche *Alternativen*

notiert werden.

Die Erfassung eines Änderungsantrags kann auf verschiedenen Wegen erfolgen:

- manuell: unter Verwendung der entsprechenden Ticketvorlage
- automatisiert: bei Verwendung der Ticketaktion "Change erstellen".

Das erstellte Ticket erhält die Default-Werte für Priorität (normal) und Status (neu) und wechselt in den nächsten Prozessschritt "Prüfung".

2. Prüfung

Bei der Prüfung wird der Sachverhalt geprüft und mögliche Risiken eingeschätzt. Dazu steht die Aktion "Prüfung" am Ticket zur Verfügung. Im sich öffnenden Dialog können mehrere Risiken notiert werden.

Die Auswahl im Feld "Standard Change" klassifiziert die Änderung:

- Standard Change - ja:
 - Übliche Änderungen, risikoarm, häufig wiederkehrend, vorab genehmigt (bspw. Austausch eines Headsets)
 - Überspringt den Prozessschritt "Zustimmung"
- Standard Change - nein:
 - Änderungen mit erhöhtem Risiko oder hohem finanziellen Aufwand (bspw. Austausch des Backupserver)
 - Erfordert Zustimmung

Wird der Prozessschritt mit OK abgeschlossen, erhält das Ticket den Status "zugewiesen" und wechselt in den nächsten Prozessschritt "Planung".

3. Planung

Bei der Planung erfolgt eine Einschätzung der Kosten und die Betrachtung eventueller Möglichkeiten, die Änderungen rückgängig zu machen. Dies kann unter Verwendung der Ticketaktion "Planung" angegeben werden.

Wird der Prozessschritt mit OK abgeschlossen, erhält das Ticket den Status "eingeplant" und wechselt in den nächsten Prozessschritt "Zustimmung".

4. Zustimmung

Für Änderungsanträge mit erhöhtem Risiko oder finanziellem Aufwand bedarf es vor der Umsetzung einer Zustimmung. Die Ticketaktion "Zustimmung" steht daher an Tickets zur Verfügung, die kein Standard-Change sind (s. Prüfung). Die Aktion öffnet einen Dialog, in dem im Bearbeitungsstand die Zustimmung erteilt werden kann.

Bei Zustimmung (OK) wechselt das Ticket in den Status "Zustimmung erteilt" und in den nächsten Prozessschritt "Bereit zur Umsetzung".

Bei Ablehnung (NOK) verbleibt das Ticket im Prozessschritt "Zustimmung" bis diese erteilt wurde und erhält den Status "eingeplant".

5. Bereit zur Umsetzung

Ist der Änderungsantrag genehmigt, kann dessen Umsetzung vorbereitet werden. Das Ticket befindet sich in einem Wartestatus bis bspw. alle Rückfrage geklärt sind und wechselt nach Klick auf die Ticketaktion "Bereit zur Umsetzung" in den nächsten Prozessschritt "Umsetzung".

6. Umsetzung

Nachdem bspw. der Austausch der Hardware vollständig erfolgt ist, gilt die Umsetzung als abgeschlossen. Am Ticket existiert dazu die Ticketaktion "Umsetzung". Sie öffnet einen Dialog zur Dokumentation der Umsetzung.

Wird die Umsetzung mit OK beantwortet, erhält das Ticket den Status "umgesetzt". Im nächsten Prozessschritt kann die Lösung abgelehnt oder akzeptiert werden.

7. Akzeptanz

Abschließend kann das Ergebnis der Umsetzung abgelehnt oder akzeptiert werden. Am Ticket stehen dazu die gleichnamigen Ticketaktionen zur Verfügung.

- Wird die Lösung abgelehnt, wird das Ticket auf den 3. Prozessschritt (Planung) zurückgesetzt, sodass der Prozessverlauf erneut starten kann.
- Wird die Lösung akzeptiert, gilt der Prozess als abgeschlossen. Das Ticket wird in den Status "warten auf automatisch schließen" gesetzt.

10.2.3.8 Prozess 5: Notfallplan

Notfallpläne dienen dazu, bei Unterbrechungen von Geschäftsabläufen oder Services den Geschäftsbetrieb am Laufen zu halten. Anhand der als FAQ vorliegenden Notfallpläne kann rasch Abhilfe geschaffen werden.

Ablaufschema für Notfallpläne

KIX behandelt Notfallpläne nach folgendem Schema und stellt mit dem Add-on "ITIL Practices" die dafür benötigten Ticketvorlagen, Ticketstatus, Aktionen etc. bereit.

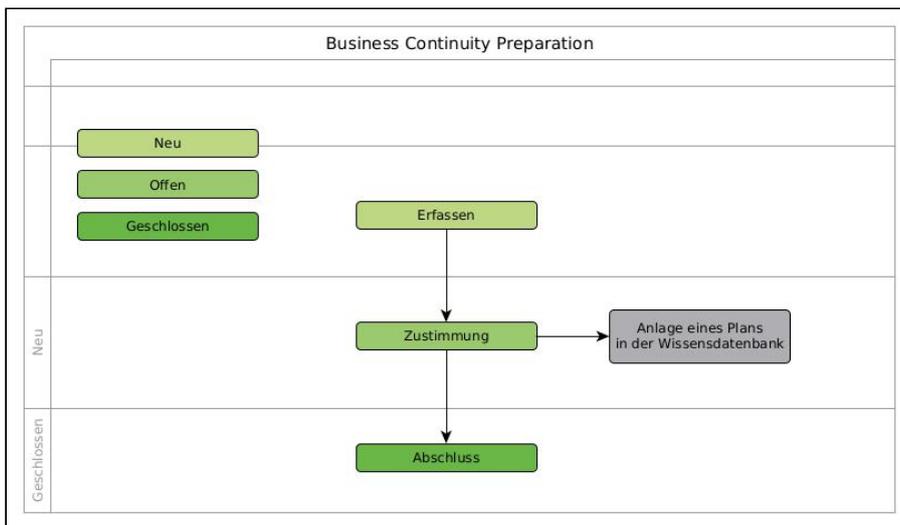


Abb.: Ablaufschema Prozess "Notfallpläne"

1. Erfassen

Zum Erfassen eines Notfallplans dient die Ticketvorlage "Business Continuity Preparation". In sich öffnenden Dialog können u. a. die im Notfall abzuarbeitenden Aufgabenschritte hinterlegt werden.

Nach dem Speichern erhält das Ticket die Default-Werte für Priorität (normal) und Status (neu). Im nächsten Prozessschritt kann die Zustimmung erteilt werden.

2. Zustimmung

Damit der vorbereitete Notfallplan in den FAQ hinterlegt wird, muss diesem zugestimmt werden. Dazu existiert am Ticket die Aktion "Zustimmung". Sie öffnet einen Dialog, in dem die Zustimmung erteilt werden kann oder abgelehnt werden kann.

Bei Zustimmung wird ein neuer FAQ Eintrag erstellt.

Bei Ablehnung bleibt das Ticket im Prozessschritt "Zustimmung", sodass der Notfallplan geändert und zu einem späteren Zeitpunkt akzeptiert werden kann.



3. Abschluss

Durch die Zustimmung zu einem Notfallplan erhält dieser den Status "erstellt". Variierend je nach kundenspezifischer Definition sind im Anschluss Maßnahmen und Arbeitsschritte zu ergänzen, um den Notfall zu beheben. Das bildet den Abschluss dieses Prozesses.

10.2.4 Erweiterte Konfigurationen

KIX Pro enthält bereits zahlreiche Grundfunktionen für Ihr IT Service Management. Darauf aufbauend erhalten Sie mit dem Add-on "ITIL Practices" viele nützliche Konfigurationen, um einen schnellen Einstieg in ITIL® 4 zu erhalten. Diese Konfigurationen können Sie schnell und einfach in Ihre Prozesse integrieren und flexibel Ihre individuellen Bedürfnisse anpassen.

Zu den erweiterten Konfigurationen zählen Elemente aus den folgenden Bereichen:

- Aktionen
- Assets
- Berichtsdefinitionen
- Dynamische Felder
- FAQs
- Jobs
- Stammdaten
- Vorlagen

10.2.4.1 Aktionen für ITIL Practices

Aktionen sind individuell konfigurierte Funktionen (Macro Actions), die unter bestimmten Voraussetzungen am Ticket oder am Artikel zur Verfügung stehen.

Das Add-on "ITIL Practices" bringt - bezogen auf den jeweiligen Prozess - folgende Ticket-Aktionen mit:

- **Allgemein:**
 - Lösung abgelehnt
 - Lösung akzeptiert
 - Lösung akzeptiert/abgelehnt

- **Störung:**
 - Klassifizieren
 - Lösung
 - Diagnose
 - Change erstellen
 - Problem erstellen

- **Problem:**
 - Kategorisierung
 - Diagnose
 - Known Error
 - Lösung
 - Change erstellen

- **Änderungsantrag**
 - Prüfung
 - Zustimmung
 - Planung
 - Bereit zur Umsetzung
 - Umsetzung

- **Serviceanfrage**
 - Prüfung
 - Bereit zur Umsetzung
 - Umsetzung
 - Change erstellen



Im Menü *KIX > Workflow > Aktionen* des Admin Moduls können Sie diese Aktionen nachkonfigurieren und neue ticket- oder artikelbasierte Aktionen anlegen. Mit dem Anlegen eigener Aktionen und der Festlegung, wem wann welche Aktion zur Verfügung steht, können Sie das System an den Workflow Ihres Unternehmens anpassen und somit am Ticket genau die Funktionen bereitstellen, die für die tägliche Serviceerbringung benötigt werden. Eine Übersicht, welche Aktionen initial in KIX Pro ausgeliefert werden, finden Sie im Admin Handbuch im Kapitel "Aktionen".

Info: Aktionen werden stets auf bestehende Tickets und deren Artikel angewendet. Im Unterschied dazu werden Vorlagen nur auf neue Tickets angewendet.

10.2.4.2 Assets für ITIL Practices

Asset-Klassen

Alle im Unternehmen befindlichen Betriebsmittel und Betriebsmittelinformationen wie Gebäude, Maschinen, Gerätschaften oder auch Verträge werden als Assets bezeichnet und als solche im System angelegt. Um diese strukturiert in KIX abzubilden, werden sie Asset-Klassen zugeordnet. So werden bspw. alle Gebäude im Unternehmen unter der Asset-Klasse „Gebäude“ zusammengefasst, alle Computer im Unternehmen unter der Asset-Klasse „Computer“ usw. Die Verwaltung der Asset-Klassen finden Sie im Menü *KIX > Assets > Asset-Klassen*.

Das Add-on "ITIL Practices" bringt weitere Asset-Klassen für die Vertrags- und Projektverwaltung mit:

- Vertrag
- Vertrag (AV)
- Projekt

General Catalog

Der General Catalog ist ein Katalog, in welchem unterschiedlichste Werte abgelegt werden können, um sie bspw. in der Klassendefinition einer Asset-Klasse wiederzuverwenden. General Catalog Einträge können bspw. sein: Attribute von Asset-Klassen, Ereigniszustände von Assets, Baugruppenreihen, Modelltypen, Gebäudearten usw. Den General Catalog finden Sie im Menü *KIX > Assets > General Catalog*.

Das Add-on "ITIL Practices" bringt folgende weitere General Catalog-Einträge mit:

- `ITSM::ConfigItem::AffectedData`
- `ITSM::ConfigItem::AffectedPersons`
- `ITSM::ConfigItem::PlaceOfDelivery`

10.2.4.3 Berichtsdefinitionen für ITIL Practices

Das Modul "Berichte" ermöglicht die Erstellung individueller Berichte für die Auswertung von Kennzahlen.

Die Berichtsdefinitionen können auf Basis von SQL Select-Abfragen erstellt werden. Agenten (mit Rolle "Report User") können anschließend auf diese Berichtsdefinitionen zugreifen und eigenständig Berichte im gewünschten Ausgabeformat erzeugen.

Grundlegende Informationen zu diesem Thema finden Sie im Admin Handbuch KIX Start im Kapitel "Berichtswesen".

Ergänzend zu den initial ausgelieferten Berichtsdefinitionen enthält das Add-on "ITIL Practices" weitere Berichtsdefinitionen zur Auswertung wichtiger ITIL®4 Schlüsselindikatoren:

- Number of Accounted Time: Created In Date Range for User and Team per Ticket
- Number of Tickets: Created Per Type and Priority
- List of Assets: Changed in Date Range per Class and IncidentState
- List of Tickets: SLA Violation In Date Range per Organisation and State
- Number of Assets: Per Class and IncidentState
- Number of Tickets: Closed In Date Range of Type "Incident" per CloseCode
- Number of Tickets: Closed In Date Range per SatisfactionPoints
- Number of Tickets: Created In Date Range of Type "Problem" per ProblemState
- Number of Tickets: Created In Date Range per Service and Impact
- Number of Tickets: SLA Violation In Date Range per Organisation and State

10.2.4.4 Dynamische Felder für ITIL Practices

Dynamische Felder sind individuelle Eingabe- und Ausgabefelder, die nach Bedarf angelegt und in den Oberflächen integriert werden können. Sie ermöglichen das Speichern von zusätzlichen Informationen sowohl am Ticket als auch an Organisationen, Kontakten oder FAQ.

Das Add-on "ITIL Practices" bringt folgende Dynamischen Felder mit, welche von den mit dem Add-on ausgelieferten Funktionen und Ticketvorlagen verwendet werden.

- Approval
- BackoutPlan
- BusinessCase
- Causes
- ChangeProcessState
- ChangeReference
- CostEstimation
- CreateChangeTicket
- CreateProblemTicket
- DesiredDeadline
- DisasterRecoveryPlan
- ErrorCategory
- Impact
- IncidentProcessState
- ProblemProcessState
- ProblemState
- Risks
- ServiceRequestProcessState
- Solution
- SolutionAccepted
- SolutionAcceptedComment
- StandardChange
- Summary
- Symptoms
- WorkAround

Die Verwaltung von Dynamischen Feldern finden Sie im Menü *KIX > System > Dynamische Felder*.

Durch Konfiguration der entsprechenden SysConfig-Schlüssel (Menü *KIX > System > SysConfig* bzw. *KIX > System > GUI Konfiguration > Agentenportal*) werden die Dynamischen Felder an gewünschter Stelle in die Benutzeroberfläche integriert.

Im KIX Start Admin Handbuch finden Sie weiterführende Informationen zu Dynamischen Feldern und deren Integration.



10.2.4.5 FAQ für ITIL Practices

Die Wissensdatenbank enthält eine Zusammenstellung häufig benötigter Informationen sowie Antworten auf häufig gestellte Fragen (FAQ). Hier können auch Notfallpläne, Workarounds und bekannte Fehler (Known Errors) abgelegt werden, sodass diese im Bedarfsfall schnell zur Verfügung stehen.

Das Add-on "ITIL Practices" ermöglicht es, direkt aus den Prozessen heraus Informationen als FAQ Einträge abzulegen. Dazu bringt "ITIL Practices" nachfolgende FAQ Kategorien mit:

- Known Error
- Notfallplan
- Workaround

Die Verwaltung von FAQ-Kategorien finden Sie im Admin Modul im Menü *KIX > Wissensdatenbank > FAQ Kategorien*. Administratoren können dort bei Bedarf weitere FAQ Kategorien anlegen.

10.2.4.6 Jobs für ITIL Practices

Das Add-on "ITIL Practices" stellt eine Reihe von Jobs bereit, welche in Abhängigkeit des Bearbeitungsstands den Fortschritt der Prozesse vorantreiben und FAQ Einträge erstellen. Jobs werden im Menü *KIX > Automatisierung > Jobs angelegt und konfiguriert*.

 **Wichtig!**

Ändern Sie die Jobkonfigurationen nur im Ausnahmefall. Anderenfalls kann es zu Störungen im Prozessverlauf kommen.

Mit dem Add-on "ITIL Practices" erhalten Sie, bezogen auf den jeweiligen Prozess, folgende neue Jobs:

- Change: 1 Set State on Review
- Change: 2 Set State on Planning
- Change: 3 Set State on Approval
- Change: 4 Set State on Ready For Implementation
- Change: 5 Set State on Implementation
- Incident: Set State on Classification
- Incident: Set State on Diagnosis
- Incident: Set State on Resolution
- Problem: Set State on Categorization
- Problem: Set State on Diagnosis
- Problem: Set State on Resolution
- ServiceRequest: 1 Set State on Review
- ServiceRequest: 3 Set State on Implementation
- Set State on Solution Accepted
- Create Change
- Create Problem
- Create FAQ Suggestion (Workaround)

10.2.4.7 Stammdaten für ITIL Practices

Stammdaten bezeichnen in der Informatik und der Betriebswirtschaftslehre Daten, die Grundinformationen über betrieblich relevante Objekte enthalten, die zur laufenden Verarbeitung in Geschäftsprozessen erforderlich sind.

Sowohl KIX Start als auch KIX Pro bringen eine Reihe von Stammdaten, wie eine Auswahl an Tickettypen oder Ticketstatus mit. Darauf aufbauend ergänzt das Add-on "ITIL Practices" diese Stammdaten um folgende Werte:

- **Ticket Typen:**
 - Problem
 - Change
 - Change (Emergency)
 - Incident (Major)
 - Incident (Security)
 - Business Continuity Preparation

- **Ticket Status:**
 - approved (open)
 - assigned (open)
 - implemented (open)
 - ready for implementation (pending)
 - in process (pending)
 - planned (open)
 - solved (open)
 - known error (open)
 - in validation

10.2.4.8 Vorlagen für ITIL Practices

Das Add-on "ITIL Practices" bringt für jede Prozessart (Serviceanfrage, Störung, Problem, Änderungsantrag, Notfallplan) eine extra konfigurierte Ticketvorlage mit:

- Service Request (Service Anfrage) - **Agent**
- Service Request (Service Anfrage) - **Customer/SSP**
- Problem
- Incident (Störung) - **Agent**
- Incident (Störung) - **Customer/SSP**
- Incident - Security (Sicherheitsrelevante Störung)
- Incident - Major (Schwerwiegende Störung)
- Change (Änderungsantrag) - **Agent**
- Change (Änderungsantrag) - **Customer/SSP**
- Change - Emergency (Änderungsantrag - Notfall)
- Business Continuity Preparation

Diese Vorlagen sind alle der Vorlagengruppe "IT Aufträge" zugeordnet.

Sie können die Konfiguration der Ticketvorlagen ändern, bspw. um Berechtigungen zu ändern oder um weitere Dynamische Felder zu integrieren. Die Verwaltung der Ticketvorlagen finden Sie unter *KIX >Workflow > Vorlagen*.

Achtung

Der **Bearbeitungsstand** ist ein Dynamisches Feld vom Typ Checkliste. An dieses sind viele Jobs, Aktionen usw. gekoppelt. **Entfernen** Sie daher den Bearbeitungsstand **nicht** aus den Ticketvorlagen und **ändern** Sie **nicht** die Konfiguration des Dynamischen Feldes! Anderenfalls können schwerwiegende Fehler in den Prozessabläufen entstehen!

Info

Mit dem Speichern werden am Ticket automatisch Status, Team, Priorität usw. gesetzt. Diese Werte resultieren aus den im System hinterlegten Default-Werten. Bei Bedarf können Sie die Default-Werte in den SysConfig-Schlüsseln *Ticket::Queue::Default*, *Ticket::State::Default*, *Ticket::Priority::Default* ändern.

10.3 Wartungsplan

Das Add-on "KIX Wartungsplan" unterstützt Sie bei wiederkehrenden Wartungsaufgaben. Es ermöglicht Ihnen die Verwaltung wiederkehrender Aufgaben zu den im Asset-Management dokumentierten Geräten, Verträgen oder Betriebsmittel.

Haupteinsatzszenarien sind:

- Planung, Ausführung und Dokumentation von Wartungen
- Erinnerung/Verlängerung von Verträgen, Zertifikaten, Genehmigungen und gesetzlichen Prüffristen, bspw.:
 - regelmäßige Prüfungen gemäß BGV/GUV-V A 3
 - regelmäßige Inspektionen von Maschinen, Fahrzeugen, Produktionsanlagen
- Darstellung von Nichtverfügbarkeiten von Betriebsmitteln
- Automatisches Erstellen von Aufgabenbeschreibungen und Wartungsaufträgen
- Digitales Gerätebuch/Objektlebenslauf

Basierend auf Ihren frei definierten Wartungsplänen generiert das Add-on die anstehenden Wartungsaufgaben und erstellt zum Zeitpunkt der Planungsfälligkeit die entsprechenden Tickets. Übersichtliche Listen und Kalender informieren Sie über die anstehenden, geplanten und abgeschlossenen Wartungsaufgaben. Somit behalten Sie den Überblick über alle Wartungsaufgaben und können diese der Fälligkeit entsprechend und vollständig dokumentiert durchführen.

10.3.1 Voraussetzung

Als Grundlage für die Nutzung des Add-ons "KIX Wartungsplan" benötigen Sie KIX 18 Pro, welches mindestens auf Version 29 läuft.

Inhalte auf dieser Seite:

- [Voraussetzung](#) (siehe Seite 408)
- [Grundlagen](#) (siehe Seite 409)
 - [Wartungs-Service](#) (siehe Seite 410)
 - [Wartungs-Asset](#) (siehe Seite 410)
 - [Wartungs-Plan](#) (siehe Seite 410)
 - [Wartungs-Aufgaben](#) (siehe Seite 411)
 - [Wartungs-Ticket](#) (siehe Seite 412)
- [Administration](#) (siehe Seite 413)
 - [Installation/System-Update](#) (siehe Seite 413)
 - [Berechtigungsrollen](#) (siehe Seite 413)
 - [Ticketvorlagen](#) (siehe Seite 414)
 - [Update von Wartungsaufgaben](#) (siehe Seite 415)
 - [Funktionserweiterungen](#) (siehe Seite 415)

10.3.2 Grundlagen

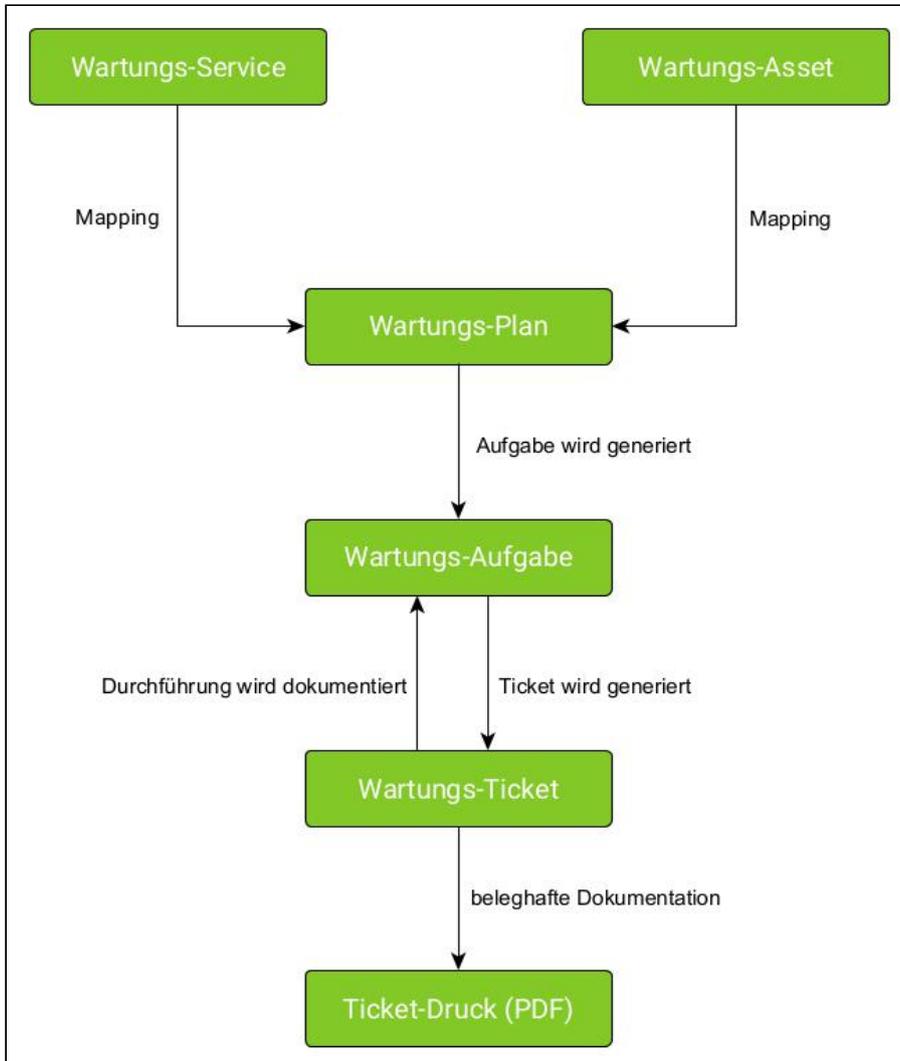


Abb.: Ablaufschema Add-on "KIX Wartungsplan"

Jede Wartung ist ein Dienst im Servicekatalog. Daher werden die Wartungsservices in der Assetklasse "Service" angelegt, sodass diese anschließend im Explorer des Moduls "Wartungsplan" zur Verfügung stehen .

Im Modul "Wartungsplan" werden die Wartungspläne erstellt und dabei die Wartungsservices mit den entsprechenden Assets verknüpft. KIX erstellt anhand der definierten Wartungspläne einmal am Tag die daraus resultierenden Wartungsaufgaben.

Im Wartungsplan ist u.a. definiert, zu welchem Zeitpunkt die Tickets für die durchzuführende Wartung erstellt werden sollen. Zu diesem Zeitpunkt generiert KIX die Tickets zur Durchführung der Wartung. Administratoren können die dafür vom System zu verwendenden Ticketvorlagen frei konfigurieren.

Die Durchführung der Wartung wird an der Wartungsaufgabe dokumentiert, indem u.a. der jeweilige Status und die Daten zu Fälligkeit und Fertigstellung hinterlegt werden.

10.3.2.1 Wartungs-Service

Ein Wartungsservice definiert, *WAS* die Wartung beinhaltet. Er beschreibt die am Asset auszuführende Leistung (z. B. alle rotierenden Getriebeteile fetten).

Jeder Wartungsservice ist ein Asset der Assetklasse "Service" und somit Teil des Servicebaums. Daher werden Wartungsservices als Assets der Assetklasse "Service" angelegt.

Nach dem Anlegen eines Wartungsservices müssen Sie ggf. folgende Konsole-Befehle (Menü "System > Konsole"), ausführen, damit der Wartungsservice im Servicebaum des Moduls "Wartungsplan" angezeigt wird:

- `Console::Command::Maint::Service::UpdateServiceCatalog`
- `Console::Command::Maint::Cache::Delete`

10.3.2.2 Wartungs-Asset

Ein Wartungs-Asset definiert das *WO* der Wartung. Wartungs-Assets sind alle Assets, die einer (regelmäßigen) Wartung zu unterziehen sind. Das kann jedes im System vorhandene Asset sein, aber auch ein Service (z. B. Prüfung Lebenszyklus).

10.3.2.3 Wartungs-Plan

Der Wartungsplan ist das Herzstück der Planung. Er definiert das *WIE* der Wartung. Hier erfolgt das Mapping (Verknüpfung) von Wartungsservice und Wartungs-Asset und es wird festgelegt, *WAS WO* und *WANN* geplant ist. Wartungspläne werden im Modul "Wartungsplan" angelegt und verwaltet.

Mapping

Ein Wartungsplan ist im einfachsten Fall dadurch gekennzeichnet, dass ein Wartungs-Service und ein Wartungs-Asset existieren und einander zugeordnet werden. Die Zuordnung kann erfolgen durch

- explizite Zuordnung eines Wartungsservices zu einem konkreten Asset
- Zuordnung eines Wartungsservices zu verschiedenen Assets, die bestimmte Bedingungen erfüllen (z. B. alle Assets der Klasse Computer, des Herstellers XYZ, seit 01.01.2023 in Betrieb)

Aufgrund dieser Flexibilität werden auch zukünftige Assets, die die Kriterien erfüllen, im Wartungsplan berücksichtigt, ohne dass der Wartungsplan angepasst werden muss.

Zeitplanung

Im Wartungsplan kann der Zeitplan festgelegt werden:

- wie oft die Wartung durchgeführt werden soll (Periodizität)
- die Dauer der Vorausplanung (Projection Days)
- der Zeitpunkt der Ticketerstellung (Planning Days)
- der Gültigkeitszeitraum des Wartungsplans
- der Planungstyp
 - Zeitplanungstreu (Shedule Based)
 - Fertigstellungstreu (Completion Based)

Ticketvorlagen

Bei der Einrichtung des Wartungsplans kann die zu verwendende Ticketvorlage ausgewählt werden. Die Ticketvorlage wird vom System verwendet, um zum gegebenen Zeitpunkt die Wartungs-Tickets zur Einplanung der Wartungsaufgabe zu generieren. Entsprechend der Vorlagenkonfiguration werden somit das beauftragte Team, die Inhalte der Aufgabenbeschreibung, FAQ-Verweise u. a. Angaben für die Umsetzung der Wartung definiert (s. auch Abschnitt unten).

10.3.2.4 Wartungs-Aufgaben

KIX prüft einmal täglich die Liste der Wartungspläne. Es wird geprüft, welche Aufgaben es für welche Wartungspläne innerhalb der nächsten n Tage gibt. Im Ergebnis dieser Prüfung werden die Wartungsaufgaben ermittelt und erstellt. Die Wartungsaufgaben werden sowohl im Modul "Wartungsplan" (Liste und Kalender) als auch an den jeweiligen Assets und Services angezeigt. Diese Prüfung kann jederzeit auch manuell ausgelöst werden, z. B. nach Änderungen an einem Wartungsplan.

Wartungsaufgaben (5)								
<input type="checkbox"/>	Service	Status	Fälligkeitsdatum (P...)	Fälligkeitsdatum F...	Ticketnummer	Ticketstatus	Tickettitel	Fertigstellungster...
<input type="checkbox"/>	Service 1	abgebrochen	23.06.2023	07.07.2023	2023063017000028	geschlossen	Maintenance Service ...	
<input type="checkbox"/>	Service 1	erledigt	16.06.2023	30.06.2023	2023063017000019	geschlossen	Maintenance Service ...	30.06.2023, 12:58
<input type="checkbox"/>	Service 1	planned	30.06.2023	14.07.2023	2023063017000037	neu	Maintenance Service ...	
<input type="checkbox"/>	Service 1	projected	07.07.2023	21.07.2023				
<input type="checkbox"/>	Service 1	projected	14.07.2023	28.07.2023				

Abb.: Liste der Wartungsaufgaben am Asset

i Info

Bitte beachten Sie: Wartungsaufgaben sind keine Tickets! Wartungsaufgaben informieren lediglich über geplante Wartungen und dienen der Wartungsdokumentation. Die Tickets für die Durchführung der Wartung werden zu einem späteren Zeitpunkt (wie im Wartungsplan definiert) erstellt.

Die Wartungsaufgaben sind das Kernelement der Wartungsdokumentation (Wartungshistorie). Hier werden alle Informationen zusammengeführt:

- das konkrete Asset, welches gewartet wird/wurde
- den konkreten Wartungsservice, der ausgeführt wird/wurde
- das Datum der Planungsfälligkeit
 - An diesem Tag sollte das Ticket erstellt werden.
- das geplante Ausführungsdatum
 - An diesem Tag sollte die Wartung durchgeführt werden.
- der tatsächliche Ausführungszeitpunkt
 - Zeitpunkt, an dem die Wartung tatsächlich ausgeführt wurde.
- der Status der Wartungsaufgabe
 - projiziert | projected: Erstellte Wartungsaufgabe, die voraussichtlich in einem Wartungsticket münden wird.
 - geplant | planned: Aufgabe, für die bereits ein Wartungsticket erstellt wurde.
 - erledigt | done: Wartungsticket wurde geschlossen (nicht abgebrochen).
 - abgebrochen | canceled: Wartungsticket wurde abgebrochen.

10.3.2.5 Wartungs-Ticket

Wartungstickets sind Tickets, welche das System unter Anwendung der am Wartungsplan hinterlegten Angaben generiert. Sie dienen der Einplanung und anschließenden Durchführung der Wartungsaufgabe.

Der Mechanismus zur Prüfung und Erstellung der Wartungsaufgaben überprüft auch die Fälligkeit der Wartungsaufgaben und erstellt einige Tage vor Ablauf der Fälligkeit (wie im Wartungsplan definiert) die zugehörigen Wartungstickets.

Wartungstickets werden angezeigt:

- am jeweiligen Asset
- am jeweiligen Service
- im Modul "Wartungsplan" (Kalender und Liste)
- im Ticketdashboard

Nach dem Erstellen des Wartungstickets wechselt der Status der Wartungsaufgabe von "projected | projiziert" zu "planned | geplant".

Nach erfolgter Durchführung der Wartung kann das Ticket geschlossen werden. Mit dem Schließen des Tickets werden der gewählte Abschlusscode und - sofern "erledigt" - der Fertigstellungszeitpunkt zurück an die Wartungsaufgabe geschrieben.

Für die belegte Dokumentation der Wartung kann das Wartungsticket als PDF heruntergeladen und bei Bedarf gedruckt und abgeheftet werden.

10.3.3 Administration

Der administrative Aufwand zur Einrichtung und Pflege des Add-ons "KIX Wartungsplan" ist gering. Vom Admin sind lediglich auszuführen:

- Installation (sofern KIX als On-Premises-Lösung genutzt wird)
- Vergeben der Berechtigungsrollen
- Einrichtung einer oder mehrerer Ticketvorlagen auf deren Basis die Wartungs-Tickets erstellt werden.

Das Anlegen von Assets und Services sowie die Einrichtung der Wartungspläne erfolgt durch den Anwender in den Modulen Assets bzw. Wartungsplan. Eine ausführliche Beschreibung dazu finden Sie im Anwenderhandbuch.

10.3.3.1 Installation/System-Update

Nach der Beauftragung stellen wir Ihnen - in der Regel am nächsten Arbeitstag - ein aktualisiertes Image Ihres Systems bereit.

Falls Sie KIX Pro on-premises betreiben, führen Sie bitte anschließend ein System-Update durch:

```
user@DockerHost:/opt/kix-on-premise/deploy/linux# ./stop.sh  
user@DockerHost:/opt/kix-on-premise/deploy/linux# ./update.sh
```

Danach ist das Add-on in Ihr System integriert und steht als zusätzliches Modul im linken Modulmenü zur Verfügung .

Die Bereitstellung des Add-ons in der KIX Cloud erfolgt über unser Support-Team.

10.3.3.2 Berechtigungsrollen

Mit dem Add-on "KIX Wartungsplan" werden weitere, ergänzende Berechtigungsrollen ausgeliefert. Diese werden den betreffenden Nutzern zusätzlich erteilt.

- Maintenance Reader:
 - kann Wartungspläne einsehen, aber nicht bearbeiten
 - benötigt zusätzlich die Rollen Agent User, Ticket Reader und Asset Reader
- Maintenance Manager:
 - kann Wartungspläne einsehen, bearbeiten und deaktivieren
 - benötigt zusätzlich die Rollen Agent User, Ticket Agent, Asset Maintainer

10.3.3.3 Ticketvorlagen

Die Ticketvorlage wird vom System verwendet, um zum Zeitpunkt der Planungsfälligkeit die Wartungstickets zu generieren, sodass die Ausführung der Wartungsaufgabe eingeplant werden kann. Der Zeitpunkt der Planungsfälligkeit ist im Wartungsplaner hinterlegt.

Je nach Konfiguration werden das beauftragte Team, die Inhalte der Aufgabenbeschreibung, FAQ-Verweise u. a. Angaben für die Umsetzung der Wartung in der Vorlage definiert.

Damit die Vorlage vom Add-on "KIX Wartungsplan" verwendet wird, muss sie dem **Nutzungskontext "System"** sowie dem **Verhalten "Wartung"** zugeordnet sein.

Für jeden Wartungsplan kann eine separate Ticketvorlage konfiguriert werden. Mehrere Wartungspläne können auch eine gemeinsame Ticketvorlage nutzen.

Die Konfiguration der Ticketvorlage erfolgt genau wie bei allen anderen Vorlagen im Admin Modul im Menü "Workflow > Vorlagen". Welche Eingabefelder in den künftigen Tickets enthalten sein sollen und mit welchen Daten das Ticket initialisiert wird, kann in der Vorlage frei konfiguriert werden. Die Verwendung von KIX Platzhaltern in der Vorlage ist möglich (z. B.:

<KIX_TICKET_DynamicField_AffectedAsset_Object_0_Attribute_0_Value>. Checklisten können in den Eingabefeldern über MobileProcessingChecklist eingefügt werden.

Häufig sind Assets auch mit FAQ-Einträgen verknüpft. Damit diese FAQ-Einträge auch am Wartungsticket verfügbar sind, wurden die Dynamischen Felder um den Feldtyp "FAQ Referenz" erweitert. Somit können Sie Vorlagen anlegen, die Verweise zu FAQ-Einträgen enthalten. Dies dient auch der Bereitstellung von FAQ-Einträgen in der Field Agent App.

Sofern im Wartungsplan keine oder eine unvollständige Ticketvorlage angegeben ist, werden am Ticket folgende Attributwerte gesetzt:

Attribut	Wert
Titel	"Maintenance <Maintenance Service Name> - <Maintenance Asset Name>"
Status	gemäß SysConfig-Schlüssel "Ticket::State::Default"
Team	gemäß SysConfig-Schlüssel "Ticket::Queue::Default"
Priorität	gemäß SysConfig-Schlüssel "Ticket::Priority::Default"
Typ	Wartung
Artikel/Kontakt	Ohne Vorlagenverwendung bleiben diese Angaben leer

Unabhängig von den Angaben in der Ticketvorlage werden am Ticket immer gesetzt:

Attribut	Wert
Betroffener Service	der entsprechende Wartungsservice
geplanter Aufwand	Aufwand aus Service
Plan Beginn	Starttermin gem. Wartungsplan
Plan Ende	Enddatum gem. Wartungsplan
Typ	Wartung

10.3.3.4 Update von Wartungsaufgaben

Konfigurationsschlüssel	Daemon::SchedulerCronTaskManager::Task###UpdateTaskList
--------------------------------	---

Das Add-on "KIX Wartungsplan" enthält einen Automatismus, der täglich ein Update aller Wartungs-Services durchführt (Daemon-Cron-Task + Console Command). Dieses Update erfolgt i. d. R. um 04:00 Uhr. Bei Bedarf können Sie dieses Update im SysConfig-Schlüssel `"Daemon::SchedulerCronTaskManager::Task###UpdateTaskList"` deaktivieren.

10.3.3.5 Funktionserweiterungen

Das Add-on "KIX Wartungsplan" ergänzt KIX um folgende Einträge:

- Ergänzung der General Catalog-Klasse "ITSM::ConfigItem::Service::Type" um den Wert "Maintenance (periodic)"
- Zusätzliche Berechtigungsrollen "Maintenance Reader" und "Maintenance Manager" (im Kontext "Agent")
- Zusätzlicher Tickettyp "Maintenance"

11 Praxis

Die nachfolgenden Kapitel enthalten eine Sammlung von Praxisbeispielen, Verwendungsmöglichkeiten und beispielhaften Use Cases. Mit diesem Service wollen wir Ihnen eine Reihe von Informationen an die Hand geben, um KIX etwas besser zu verstehen, Anwendungsmöglichkeiten zu erkennen und bei Ihnen die Lust am Ausprobieren wecken.

- [Periodischer Job "Lizenzverlängerung"](#) (siehe Seite 417)
- [Verwendung von Checklisten](#) (siehe Seite 421)
 - [Konfiguration und Bereitstellung von Checklisten](#) (siehe Seite 427)
 - [Datenstruktur von Checklisten](#) (siehe Seite 450)

11.1 Periodischer Job "Lizenzverlängerung"

In diesem UseCase wird ein Job erstellt, der ein Ticket generiert, sobald die Lizenz eines Software-Assets innerhalb der nächsten 1 bis 3 Wochen abläuft.

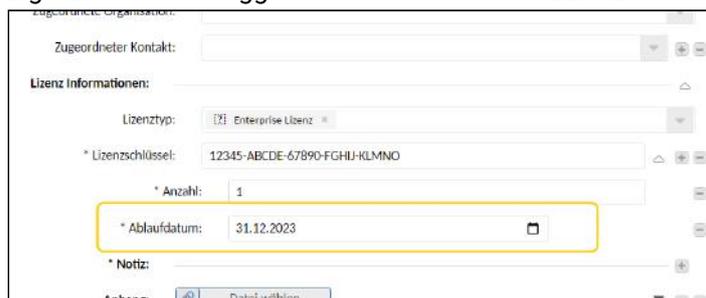
Dazu prüft der Job alle Assets der Klasse "Software" und kontrolliert das angegebene Lizenz-Ablaufdatum. Liegt dieses Datum innerhalb der nächsten 1 bis 3 Wochen, wird ein Ticket generiert und dem zuständigen Team zugeordnet. Im Ticket sind das betroffene Asset sowie dessen Name und ID angegeben, sodass der Agent auf einen Blick erkennt, um welches Asset es sich handelt.

Hinweis: Dieser UseCase kann nur mit KIX Pro abgebildet werden, da in KIX Start keine Jobs auf Assets ausgeführt werden können.

11.1.1 Vorbereitung

1. An den betreffenden Assets der Klasse "Software" muss das Ablaufdatum der Lizenz hinterlegt sein, da dieses Datum im Job geprüft wird.

Ergänzen Sie daher ggf. das Ablaufdatum der Lizenz für eine oder mehrere Software-Assets.



2. Das Attribut "Ablaufdatum | Expiration Date" ist initial nicht durchsuchbar und steht somit in den Job-Filtern nicht zur Auswahl. Die Asset-Klassendefinition der Klasse "Software" muss daher angepasst werden:

1. Navigieren Sie im Admin Modul zu `KIX > Assets > Asset-Klassen`
2. Klicken Sie die Klasse "Software" an, um die Detailansicht zu öffnen und klicken Sie dann auf "Bearbeiten".
Der Dialog "Assetklasse bearbeiten" wird geöffnet.
3. Vergrößern Sie das Textfeld "Klassendefinition", um den Code einfacher bearbeiten zu können. Optional können Sie den Code in einen (Text-)Editor Ihrer Wahl kopieren.
4. Nutzen Sie die Suchfunktion Ihres Browsers oder Editors und suchen Sie nach dem Attribut "Expiration Date".
5. Setzen Sie ein Komma nach der letzten Zeile und fügen Sie den Parameter `'Searchable'` hinzu.
`=> 1`

Damit wird das Attribut "Ablaufdatum" auf durchsuchbar gesetzt.

```

* Class Definition: ⓘ
'CountDefault' => 0,
'CountMax' => 1,
'CountMin' => 0,
'CustomerVisible' => 0,
'Input' => {
  'Required' => 1,
  'Type' => 'Date',
  'YearPeriodFuture' => 10,
  'YearPeriodPast' => 20
},
'Key' => 'ExpirationDate',
'Name' => 'Expiration Date',
'Searchable' => 1
}

```

6. Speichern Sie die Änderung.

11.1.2 Job konfigurieren

1. Job Informationen

1. Job Typ: Asset

i Mit Ausführung des Jobs wird zwar ein Ticket erstellt, aber da der Job auf Assets angewendet wird, ist als Job-Typ "Asset" anzugeben.

2. Name, Kommentar: Felder können individuell belegt werden.

3. Gültigkeit: gültig

2. Ausführungsplan

Konfigurieren Sie einen Zeitpunkt Ihrer Wahl. Es empfiehlt sich eine zeitgesteuerte Ausführung, z. B.

1. Wochentag(e): Montag

2. Zeit: 10:00 (Uhr)

3. Filter

Setzen Sie die Filter, um die Software-Assets zu ermitteln, deren Lizenz innerhalb der nächsten 1 bis 3 Wochen abläuft.

1. Klasse - enthalten in - Software

2. Ablaufdatum - innerhalb - der nächsten - 1 - Wochen - bis - der nächsten - 3 - Wochen
(Kann das Attribut "Ablaufdatum" nicht gewählt werden, ist es nicht durchsuchbar (s. Vorbereitung))

4. Aktionen

1. 1. Action: Variable Setzen

Mittels Platzhalter wird vom auslösenden Asset die Asset-Nr. und dessen Name ermittelt und gemeinsam mit Text als String in einer Variable gespeichert.

i Variablen können mehrere Attribute aufnehmen und Arrays bilden. Die Kombination aus Text und Platzhaltern ist ebenfalls möglich.

a. Variable: `varCurrentAsset`

b. Wert: `A#: <KIX_ASSET_Number> (<KIX_ASSET_Name>)`

2. 2. Action: Macro ausführen

a. Macro: Ticket

i. Aktion: Ticket anlegen

i Die nachfolgend angegebenen Parameter können Sie nach Bedarf ändern.

A. Kanal: Notiz

B. Artikeltext:

*Die Software-Lizenz läuft innerhalb der nächsten 14 Tage aus.
Bitte Lizenz erneuern.*

Asset: `${varCurrentAsset}`
(Asset-ID: `${RootObjectID}`)

C. Kontakt: Martin Mustermann

D. Organisation: My Organisation

E. Priorität: hoch

F. Status: neu

G. Titel: Erneuerung Software-Lizenz erforderlich: `${varCurrentAsset}`

H. Team: Service Desk

I. Dynamische Felder:

Die Variable `${RootObjectID}` ermittelt das Asset, welches den Job auslöst und setzt dieses als "Betroffenes Asset" am Ticket.

I. Name: Betroffenes Asset

II. Wert: `${RootObjectID}`

11.1.3 Job ausführen

Der Job wird zum konfigurierten Zeitpunkt ausgeführt und erstellt ein Ticket zur Verlängerung der Lizenz. Als "Betroffenes Asset" ist das den Job auslösende Objekt gesetzt (`${RootObjectID}`). Dies entspricht dem Asset, dessen Software-Lizenz verlängert werden muss.

Zudem enthält der Artikeltext und der Ticketbetreff die Asset-Nummer und den Namen des betreffenden Assets, sodass der Agent auf einen Blick erkennt, um welches Asset es sich handelt. Diese Informationen wurden in der Variable "`varCurrentAsset`" gespeichert und beim Generieren des Tickets an entsprechender Stelle ausgegeben.

The screenshot displays a KIX ticket interface. At the top, the ticket title is "Ticket#2023121817000807-Software licence renewal required: A#: 1710000001 (Office 365)". Below the title, there are tabs for "Ticket Information", "Linked Objects (1)", and "History". The "Ticket Information" tab is active, showing a summary of the ticket with the following details:

- Summary:** Organisation: [TD-ORGANISATION-001] Springfield Nuclear Plant; Contact: Homer Simpson; State: new; Type: Unclassified; Priority: 2 high.
- Assignees:** Team: Service Desk; Lock State: Unlocked.
- References:** Affected Assets: A#1710000001 - Office 365.
- Scheduling:** Accounted Time: 0d 00h 00m.

At the bottom of the ticket information, it states: "Created by not assigned at 12/18/2023, 03:38 PM. Last change by not assigned at 12/18/2023, 03:38 PM".

Below the ticket information is the "Communication History (1)" section. It shows a search bar and filters for "From external", "From internal", "Visible in Self Service Portal", and "Unread". The communication history shows a message from "tt.mm.jjjj" dated "12/18/2023, 03:38 PM". The message content is:

[#1] Software licence renewal required: A#: 1710000001 (Office 365)
not assigned <admin@localhost> | not assigned

The message body contains the text: "The software licence will expire within the next 14 days. Please renew your licence." and a link to the affected asset: "Asset: A#: 1710000001 (Office 365) (Asset-ID: 25)".

Abb.: Ein durch den Job erstelltes Ticket

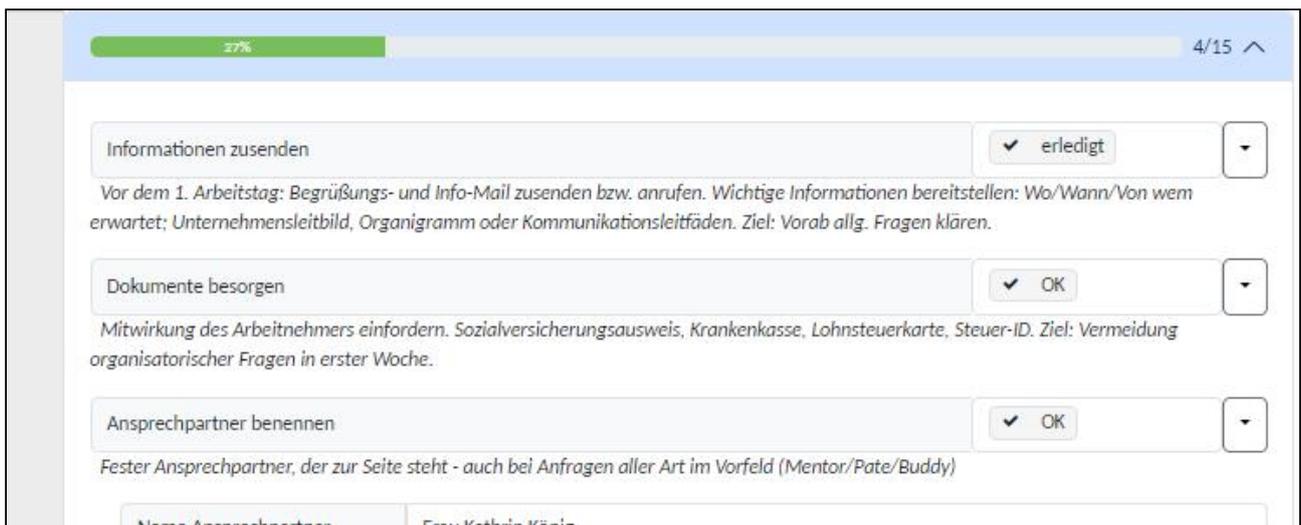
11.2 Verwendung von Checklisten

11.2.1 Checklisten in KIX 18

Checklisten sind in eines der wichtigsten Hilfsmittel bei der täglichen Arbeit und ihre Einsatzmöglichkeiten sind denkbar vielfältig: Sie werden als Einkaufs- oder To-do-Listen aber auch für Fragenkataloge oder Prüflisten verwendet. Mit KIX 18 haben Sie die freie Wahl beim Erstellen der benötigten Checklisten, sodass für jeden Anwendungsfall und für jeden Verwendungszweck die passende Checkliste konfiguriert werden kann. Der Bearbeitungsstand einer Checkliste ist anhand eines Fortschrittsbalkens jederzeit ersichtlich.

Inhalte dieser Seite:

- [Checklisten in KIX 18](#) (siehe Seite 421)
- [Checklisten in der Praxis](#) (siehe Seite 424)
- [Checklisten in KIX Pro](#) (siehe Seite 424)
- [Konfiguration von Checklisten](#) (siehe Seite 425)
- [Verweise](#) (siehe Seite 426)



The screenshot shows a portion of a KIX 18 ticket form. At the top, there is a progress bar indicating 27% completion and a page indicator '4/15'. The checklist contains three items:

- Informationen zusenden**: Status 'erledigt' (checked).
- Dokumente besorgen**: Status 'OK' (checked).
- Ansprechpartner benennen**: Status 'OK' (checked).

Below the third item, there is a text input field for the contact name, with 'Frau Kathrin König' entered.

Abb.: Teil einer Checkliste im Ticketformular

In KIX 18 können beliebig viele Checklisten vom Administrator angelegt und am Ticket bereitgestellt werden. Im Zuge der Ticketbearbeitung können Agenten diese Checklisten teilweise oder ganz abarbeiten. Wird das Ticket an einen anderen Agenten oder Team weitergeleitet, kann die Abarbeitung der Checkliste dort durch diese Agenten fortgesetzt werden.

Jede Checkliste besteht aus beliebig vielen, frei definierten Listenpunkten. Jeder Listenpunkt kann wiederum weitere, ihm untergeordnete Listenpunkte enthalten. Die einzelnen Listenpunkte können in beliebiger Reihenfolge und zu jedem Zeitpunkt abgearbeitet werden, da sie direkt am Ticket gespeichert sind. Jeder Listenpunkt besteht aus einem kurzen Titel zur Benennung ¹ und einem optionalen Hinweistext zur

näheren Erläuterung des Listenpunkts ². Unklarheiten und Nachfragen beim Abarbeiten der Checkliste werden so vermieden.



Abb.: Listenpunkte einer Checkliste

Ein Listenpunkt kann entweder ein Textfeld ³ oder ein Auswahlfeld (Dropdown) ⁴ sein.

Textfeld: Textfelder werden verwendet, um Notizen, Antworten, Begründungen oder Hinweise zu notieren. Je nach Erfordernis können einzeilige oder mehrzeilige Textfelder genutzt werden. Sie dienen bspw. dazu, um

- die Antworten des Befragten während einer telefonischen Umfrage
- die Gründe einer Ablehnung im Genehmigungsprozess
- Hinweise für den Servicetechniker zum Reparatur- oder Installationsauftrag
- Namen oder Telefonnummern
- und andere Notizen

schriftlich festzuhalten.

Sobald ein Textfeld Zeichen enthält (Text oder Leerzeichen), wird der Fortschritt der Checkliste hochgezählt.

Auswahlfeld: Auswahlfelder (Dropdown) dienen dem Setzen des Status und kennzeichnen den Bearbeitungsstandes eines jeden Listenpunkts. In der Checklisten-Konfiguration können die Status und deren Bezeichnungen individuell festgelegt werden. Ebenso, welcher Status den Fortschritt der Checkliste weiter hochzählt. Sind in der Checklisten-Konfiguration keine individuellen Status angegeben, wird folgender Standard verwendet:

Zustand	Bedeutung	Fortschritt
OK	Okay / Erledigt / Genehmigt	wird hochgezählt, da gültige Antwort
NOK	Nicht Okay / Nicht ausgeführt / Nicht genehmigt	wird hochgezählt, da gültige Antwort
PENDING	Warten-Status / Warten auf Rückmeldung / Zurückgestellt	wird nicht hochgezählt, da Schwebezustand
n.a.	keine Angabe / Angabe nicht erforderlich / keine Aussage getroffen	wird hochgezählt, da z. B. "keine Aussage getroffen" auch eine gültige Antwort ist

Zustand	Bedeutung	Fortschritt
-	unbeantwortet / keine Auswahl am Checklistenpunkt getroffen	wird nicht hochgezählt

Wie weit eine Checkliste abgearbeitet ist, kann anhand eines Fortschrittbalkens und/oder eines numerischen Wertes (2/10) in der Ticket Detailansicht sowie in den Dashboard-Tabellen visualisiert werden. Zusätzlich wird in der Ticket Detailansicht ein Info-Symbol neben dem Fortschrittsbalken angezeigt. Mit auf das Info-Symbol werden die Details zum aktuellen Bearbeitungsstand angezeigt. Wird der Mauszeiger über ein Label oder ein Icon geführt, werden der Beschreibungstext bzw. Checklisten-Status eingeblendet.

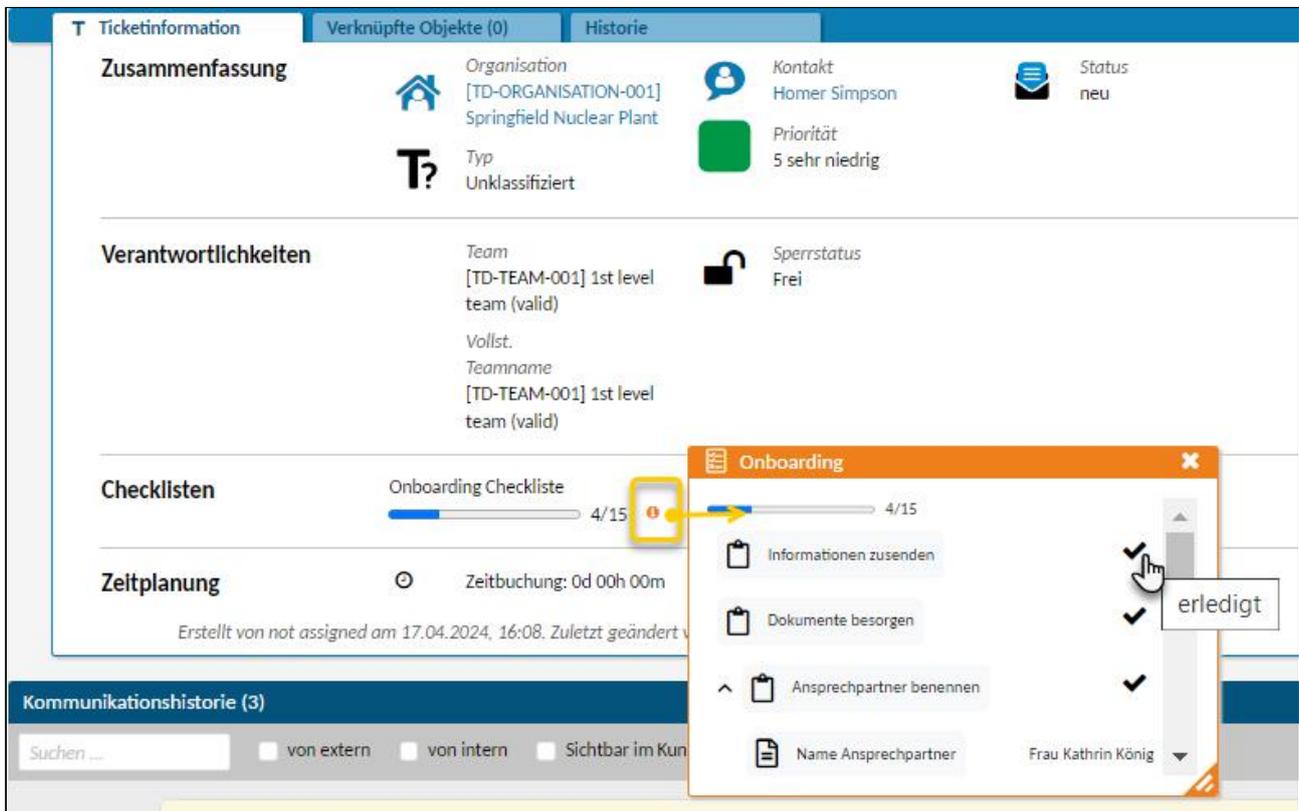


Abb.: Checklistenfortschritt in der Ticket Detailansicht

11.2.2 Checklisten in der Praxis

In KIX können beliebig viele, individuell konfigurierte Checklisten angelegt und in den Ticketoberflächen bereitgestellt werden. Dadurch können die unterschiedlichsten Szenarien abgebildet werden, zum Beispiel:

1. Die Checkliste für Servicetechniker als Leitlinie für eine vollständige Serviceerbringung, um die Einhaltung von Qualitätsstandards zu garantieren.
Der Servicetechniker kann die Checkliste Punkt für Punkt abarbeiten, beantworten und ggf. einen Kommentar zu einzelnen Punkten notieren.
2. Die Checkliste für die Abbildung der zu erledigenden Aufgaben in einem Onboarding-Prozess.
Durchläuft das Ticket mehrere Abteilungen, so kann jede Abteilung ihre Arbeitsschritte dokumentieren und abhaken. Der aktuelle Stand des Prozessfortschritts ist somit für jeden Ticketbearbeiter sofort ersichtlich.
Ein Konfigurationsbeispiel solch einer Checkliste finden Sie hier: [Konfiguration und Bereitstellung von Checklisten](#) (siehe Seite 427)
3. Die Checkliste als Gesprächsleitfaden oder Fragebogen für Support oder Vertrieb.
Durch diese gezielten Fragestellungen können Agenten einen gemeldeten Schaden oder eine Störung lokalisieren und konkretisieren.
4. Die Checkliste zur Auflistung aller erforderlichen Dokumente und Nachweise, die zusammen mit einer von Ihnen produzierten Maschine oder technischen Anlage ausgeliefert werden.
So wird sichergestellt, dass alle geforderten Dokumente und Nachweise wie Produktdatenblätter, Sicherheitshinweise, Lieferantendokumentation usw. dem Produkt beiliegen und den rechtlichen Anforderungen entsprechen wird.
5. Die Checkliste zur Erfassung der Prozess-Schritte von Genehmigungsverfahren oder im Auftragsmanagement.
Durch Abhaken der einzelnen Schritte und Hinterlegen von Notizen wird sichergestellt, dass nichts übersehen wird und warum welche Entscheidungen getroffen wurden.
Ein Anwendungsbeispiel zum Auftragsmanagement finden Sie im KIX Start - Anwenderhandbuch unter: Praxis > Das Auftragsmanagement.

11.2.3 Checklisten in KIX Pro

Im Zusammenspiel mit den erweiterten Funktionen von KIX Pro finden Checklisten auch bei der Bereitstellung individuell konfigurierter [Ticketvorlagen](#) (siehe Seite 151) sowohl für das Agenten- als auch für das Self Service Portal Verwendung. Das ermöglicht Ihnen, eine Vielzahl anwendungsspezifischer Ticketformulare bereitzustellen: z. B. ein Ticketformular für Bestellungen, ein anderes Ticketformular für Störmeldungen, ein weiteres Ticketformular für das Onboarding usw... Jedes einzelne dieser Ticketformulare

können Sie so konfigurieren, dass es sowohl die dem Anwendungsfall entsprechenden Eingabefelder als auch eine oder mehrere anwendungsspezifisch konfigurierte Checkliste/n enthält.

Ebenso können in KIX Pro eine Vielzahl an [Ticketaktionen](#) (siehe Seite 93) angelegt und - unter Berücksichtigung von Berechtigungen - am Ticket bereitgestellt werden. Auch diese können Checklisten enthalten, deren Konfiguration konkret auf die Funktion der jeweiligen Ticketaktion abgestimmt ist. Ticketaktionen werden immer dann aufgerufen, wenn der Sachverhalt dies erfordert. So kann bspw. an einem Störungs-Ticket bei Bedarf ein zusätzlicher Fragebogen am Ticket hinterlegt werden, wenn es sich um einen Havariefall handelt.

An jedem Ticket können mehrere Ticketaktionen und Checklisten Anwendung finden. So kann bspw. an einem Auftrags-Ticket

- das Team "Einkauf" seine Einkaufs-Checkliste mit den zu bestellenden Produkten am Ticket verwenden und danach
- das Team "Verkauf" bei der Auslieferung seine Checkliste mit den auszuliefernden Produkten und Dokumenten ausfüllen und abschließend
- das Team "After-Sales-Management" einen Fragebogen zur Kundenzufriedenheit nutzen.

Auch die kombinierte Verwendung von Ticketvorlagen und Ticketaktionen ist in KIX Pro möglich, wobei beide jeweils unterschiedlich konfigurierte Checklisten enthalten können.

11.2.4 Konfiguration von Checklisten

Checklisten in KIX sind Dynamische Felder vom Typ "Checklist". Wie bei allen Dynamischen Feldern sind die in einer Checkliste ausgefüllten Daten immer auch an das jeweilige Ticket gebunden, in welchem die Daten eingetragen wurden.

Die Konfiguration von Checklisten erfolgt durch den Administrator im Modul "Admin" im Menü *System > Dynamische Felder*.

Die Bereitstellung einer Checkliste in den Oberflächen erfolgt wie bei jedem anderen Dynamischen Feld:

- KIX Start: Durch Anpassungen der entsprechenden Konfigurationsschlüssel im Menü *System > SysConfig*.
- KIX Pro: Durch Integration des Dynamischen Feldes in eine [Ticketvorlage](#) (siehe Seite 151) oder [Aktion](#) (siehe Seite 93) . Oder durch Anpassen der entsprechenden Konfigurationsschlüssel unter Verwendung des JSON-Editors im Menü *System > GUI-Konfiguration > Agentenportal*. Dieser ermöglicht eine komfortablere Bearbeitung Konfigurationsschlüssel.

Eine Auflistung der wichtigsten Konfigurationsschlüssel inkl. eines kompletten Anwendungsbeispiels zum Anlegen und Bereitstellen einer Checkliste finden Sie unter: [Konfiguration und Bereitstellung von Checklisten](#) (siehe Seite 427) .

Nach der Bereitstellung ist das Aussehen und Verhalten der Checkliste abhängig vom Kontext, in dem sie eingebunden wurde:

- In den Dialogen "neues Ticket" und "Ticket bearbeiten" sowie in Ticketvorlagen oder Aktionen wird sie als bearbeitbare Liste angezeigt.
- In den Ticketdetails und Dashboard-Tabellen wird sie - je nach gewählter Komponente - als Fortschrittsbalken oder als numerischer Wert (2/10) angezeigt.

Zudem können Agenten über die Anpassung ihres Home Dashboard die Anzeige des Checklistenstatus in einer Dashboard-Tabelle individuell ein- oder ausschalten. Eine Beschreibung zur Personalisierung des Home Dashboards finden Sie im KIX Start Anwender-Handbuch (Start) unter: Home Dashboard > Home Dashboard personalisieren.

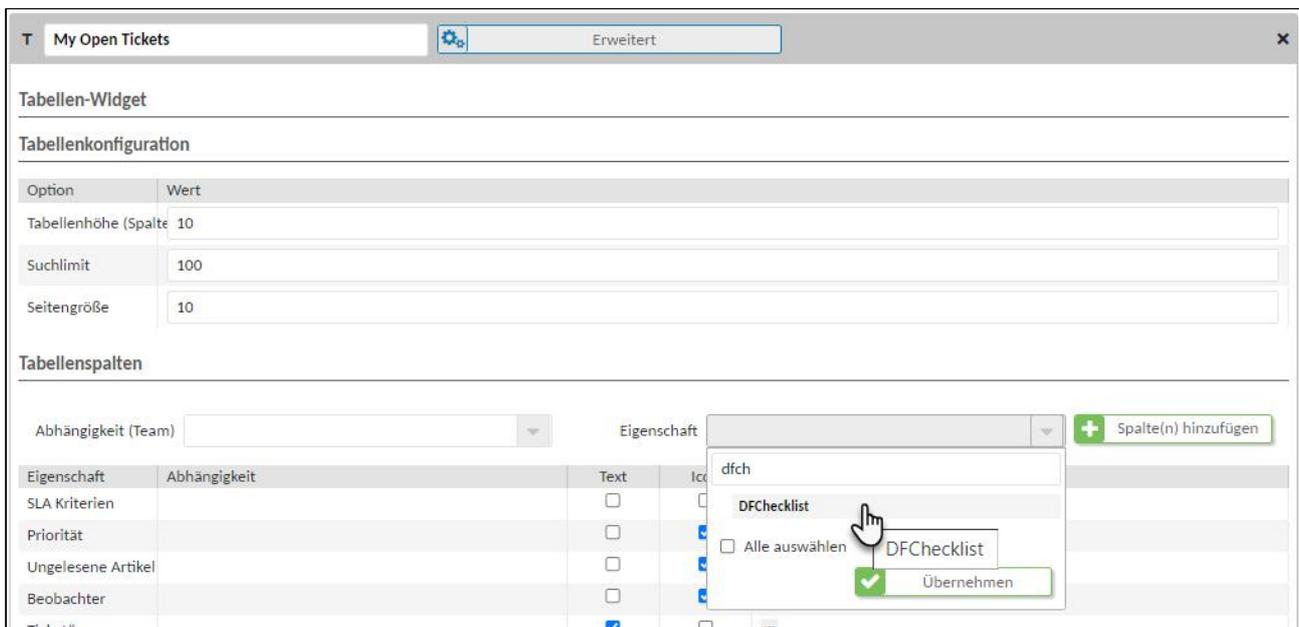


Abb.: Eine Checkliste dem Home Dashboard hinzufügen

11.2.5 Verweise

In den Admin-Handbüchern von KIX Start und KIX Pro finden Sie weiterführende Informationen zu Dynamischen Feldern, dem Feldtyp "Checkliste" sowie zu Ticketvorlagen und Ticketaktionen:

- Dynamische Felder
- Feldtypen Dynamischer Felder
- Ein Dynamisches Feld einbinden
- Werte Dynamischer Felder anzeigen
- Konfiguration von Dashboard-Tabellen
- [Ticketaktionen](#) (siehe Seite 93)
- [Vorlagen](#) (siehe Seite 151)
- Anwendungsbeispiel: [Konfiguration und Bereitstellung von Checklisten](#) (siehe Seite 427)

11.2.6 Konfiguration und Bereitstellung von Checklisten

Checklisten in KIX sind Listen mit vordefinierten Aufgabenstellungen, welche von den Agenten zum Zeitpunkt ihrer Wahl ganz oder teilweise abgearbeitet werden können. Administratoren können im Admin Modul unter *System > Dynamische Felder* beliebig viele, unterschiedlich konfigurierte Checklisten anlegen und sie

anschließend über die Konfiguration der Benutzeroberfläche in Ticketdialogen, in der Ticket Detailansicht und/oder in Dashboard Tabellen bereitstellen.

Das nachfolgende Komplett-Beispiel zeigt, wie eine Checkliste konfiguriert und in die verschiedenen Oberflächen integriert wird. Als Anwendungsszenario dient ein Pre-Boarding-Prozess, welcher die zu erledigenden Aufgaben bei Neueinstellung von Mitarbeitern abbildet (beispielhafte Kurzform).

Folgende Handlungsschritte werden im Beispiel durchgeführt:

1. **Checkliste anlegen:** Erstellen und Konfigurieren eines Dynamischen Feldes vom Typ "Checkliste"
2. **Checkliste bereitstellen:** Integration der Checkliste in die Dialoge
 - "neues Ticket"
 - "Ticket bearbeiten"
3. **Anzeige** des Checklistenstatus
 - in der Ticket Detailansicht
 - im Home Dashboard

Info: Jede im System angelegte Checkliste ist ein eigenständiges Dynamisches Feld. Sie können daher frei wählen, ob und wo Sie die Checkliste bereitstellen oder anzeigen möchten und daher auch nur Teile dieses Beispiels nutzen.

11.2.6.1 Checkliste anlegen

Eine Checkliste in KIX ist ein Dynamisches Feld vom Typ "Checklist". Sie wird wie jedes Dynamische Feld im Menü *System > Dynamische Felder* angelegt und konfiguriert. Die Konfiguration einer Checkliste besteht aus

Inhalte auf dieser Seite:

- [Checkliste anlegen](#) (siehe Seite 427)
 - [Konfiguration](#) (siehe Seite 429)
 - [Item-Konfiguration](#) (siehe Seite 429)
 - [Checklisten-Status](#) (siehe Seite 432)
- [Checkliste bereitstellen](#) (siehe Seite 433)
 - [Bereitstellung im Dialog "neues Ticket"](#) (siehe Seite 435)
 - [Bereitstellung im Dialog "Ticket bearbeiten"](#) (siehe Seite 439)
- [Checklistenstatus anzeigen](#) (siehe Seite 440)
 - [Anzeige in den Ticketdetails](#) (siehe Seite 440)
 - [Anzeige im Home Dashboard](#) (siehe Seite 445)
- [Verweise](#) (siehe Seite 449)

den Basisdaten wie Name, Label, Feld- und Objekttyp und der Konfiguration der einzelnen Listenpunkte (Items). Jeder Listenpunkt kann weitere, ihm untergeordnete Listenpunkte enthalten.

Alle Listenpunkte müssen eine eindeutige (alphanumerische) ID besitzen. Sie kann frei vergeben werden, darf jedoch nur einmal innerhalb einer Checkliste vorkommen. Jeder Listenpunkt besteht aus einem kurzen Titel zur Benennung und einer optionalen Beschreibung als Hinweistext zum Listenpunkt.

Ein Listenpunkt kann entweder

- ein einzeiliges Textfeld (Text) oder
- ein mehrzeiliges Textfeld (TextArea) oder
- eine Auswahlliste (ChecklistState)

sein. Textfelder werden verwendet, um Notizen, Antworten, Begründungen oder Hinweise zu notieren. Auswahlfelder werden verwendet, um den aktuellen Status eines Listenpunkts zu dokumentieren (z. B.: OK, NOK, PENDING, n.a.). Diese Festlegung wird unter Eingabe definiert. Weiterhin kann jeder Listenpunkt optional mit einem Wert vorbelegt sein, welcher vom Agenten zum Zeitpunkt der Verwendung geändert werden kann.

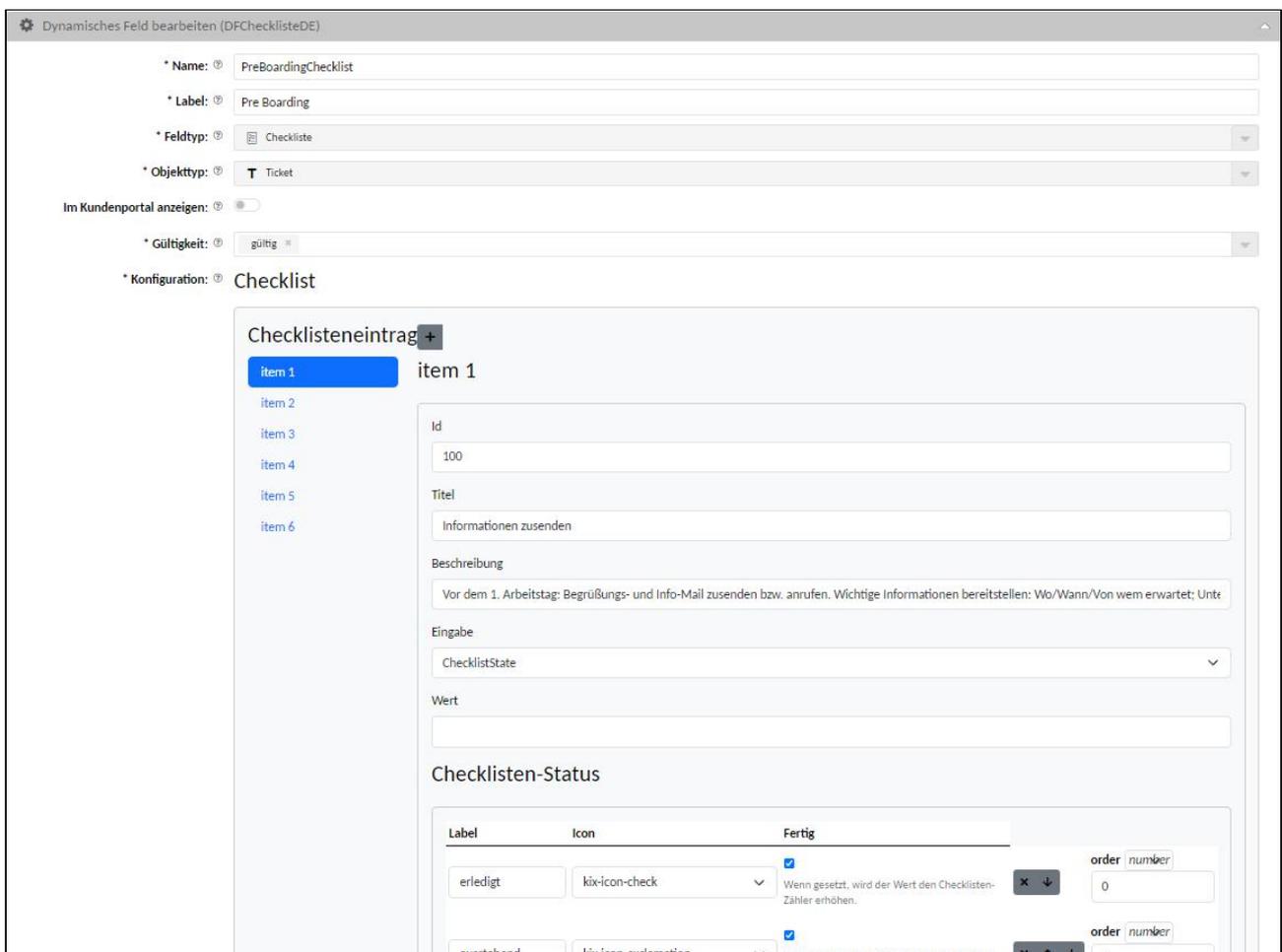


Abb.: Checklistenkonfiguration

Konfiguration

Für das Beispiel des Pre-Boarding-Prozesses legen Sie die Checkliste wie folgt an:

1. Navigieren Sie ins Menü *System > Dynamische Felder*
2. Klicken auf "Neues Feld".
3. Erstellen Sie im sich öffnenden Dialog die Checkliste mit folgenden Basisdaten:
 - **Name:** PreBoardingChecklist
 ⓘ Notieren Sie sich den Namen. Sie benötigen ihn später für die Integration in die Oberflächen.
 - **Label:** Pre Boarding
 - **Feldtyp:** Checkliste
 - **Objekttyp:** Ticket
 - **Im Kundenportal anzeigen:** nein/deaktiviert
 - **Gültigkeit:** gültig
4. Legen Sie pro (Sub-)Listeneintrag ein Item an.
 - Klicken Sie dazu auf das Plus-Symbol neben "Checklisteneintrag".
Pro Klick wird ein weiterer Checklistenpunkt (Item) hinzugefügt.
 - Klicken Sie auf das Plus-Symbol neben "Sub", um einem Listeneintrag einen untergeordneten Listeneintrag hinzuzufügen.
Pro Klick wird ein weiterer untergeordneter Checklistenpunkt (Item) hinzugefügt.
5. Konfigurieren Sie die einzelnen Listeneinträge wie im Abschnitt "Item-Konfiguration" angegeben.
6. Speichern Sie abschließend Ihre Konfiguration. Danach ist die Checkliste angelegt und kann in den Oberflächen bereitgestellt werden.

Item-Konfiguration

Die angegebenen Werte orientieren sich am Anwendungsbeispiel des Pre-Boarding-Prozesses. Sie können die Werte an Ihren konkreten Bedarf anpassen.

Item	Sub-Item	ID	Sub-Item	Titel	Beschreibung	Eingabe	Wert
1		100		Informationen zusenden	Vor dem 1. Arbeitstag: Begrüßungs- und Info-Mail zusenden bzw. anrufen. Wichtige Informationen bereitstellen: Wo/Wann/Von wem erwartet; Unternehmensleitbild, Organigramm oder Kommunikationsleitfäden. Ziel: Vorab allg. Fragen klären.	Checklist State	--
2		200		Dokumente besorgen	Mitwirkung des Arbeitnehmers einfordern. Sozialversicherungsausweis, Krankenkasse, Lohnsteuerkarte, Steuer-ID. Ziel: Vermeidung organisatorischer Fragen in erster Woche.	Checklist State	--
3		300		Ansprechpartner benennen	Fester Ansprechpartner, der zur Seite steht - auch bei Anfragen aller Art im Vorfeld (Mentor/Pate/Buddy)	Checklist State	--
	1	310		Name Ansprechpartner	--	Text	--

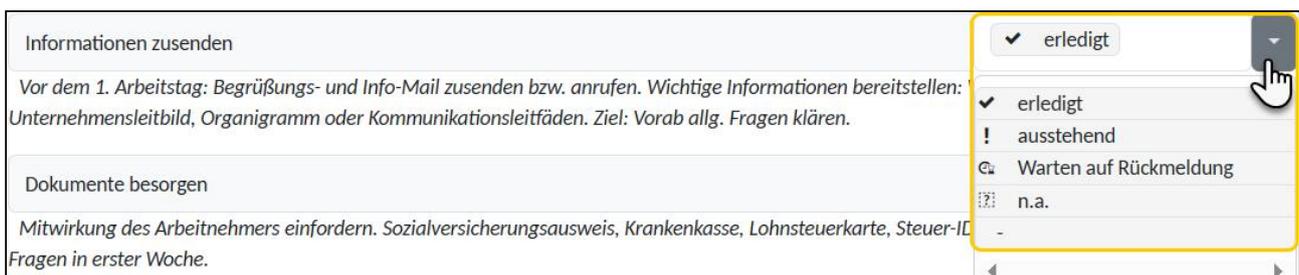
Item	Sub-Item	ID	Sub-Item	Titel	Beschreibung	Eingabe	Wert
4		400		Arbeitsplatz einrichten	Arbeitsplatz muss am 1. Arbeitstag eingerichtet sein. Dazu gehören: Bereitstellung Passwörter, Druckerzugang, Telefon, Bürostuhl/ Schreibtisch, sonstige Hilfsmittel. Ziel: Vermeidung technischer Blocker in erster Woche.	Checklist State	--
	1	410		Passwörter bereitgestellt?	--	Checklist State	--
	2	420		Drucker Zugang bereitgestellt?	--	Checklist State	--
	3	430		Telefon bereitgestellt?	--	Checklist State	--
	4	440		Arbeitsplatzrechner bereitgestellt?			
			441	Desktop-PC	--	Checklist State	
			442	Laptop	--		
	4	440		Bürostuhl und Bürotisch bereitgestellt?	--	TextArea	--
	5	450		Sonstige Büromittel	--	Checklist State	--

Item	Sub-Item	ID	Sub-Item	Titel	Beschreibung	Eingabe	Wert
5		500		Einarbeitungsplan erstellen	Muss frühzeitig, vor Arbeitsbeginn, stehen. Inhalt: detaillierten Arbeitsplan für erste Woche; Aufgaben; künftige Projekte. Ziel: Chaos und Leerlauf verhindern.	Checklist State	---
	1	510		URL Einarbeitungsplan	---	Text	---
6		600		Wohnungssuche organisieren	Unterstützung Auswärtiger bei Wohnungssuche, Umzug oder Behördengängen. Ziel: Vermeidung Ablenkung, Ermöglichen Konzentration auf Einarbeitung.	Checklist State	---

Checklisten-Status

Punkte des Typs "ChecklistState" innerhalb der Checkliste erzeugen Auswahlfelder. Agenten können eine der angebotenen Optionen wählen, um den Status des Checklistenpunkts anzugeben.

Pro Checklistenpunkt können Sie beliebig viele, frei definierte Auswahlwerte konfigurieren. Ist nichts angegeben, werden die Standardwerte verwendet (OK, NOK, PENDING, n.a., -).



Informationen zusenden

Vor dem 1. Arbeitstag: Begrüßungs- und Info-Mail zusenden bzw. anrufen. Wichtige Informationen bereitstellen: Unternehmensleitbild, Organigramm oder Kommunikationsleitfäden. Ziel: Vorab allg. Fragen klären.

Dokumente besorgen

Mitwirkung des Arbeitnehmers einfordern. Sozialversicherungsausweis, Krankenkasse, Lohnsteuerkarte, Steuer-ID Fragen in erster Woche.

- ✓ erledigt
- ✓ erledigt
- ! ausstehend
- 🕒 Warten auf Rückmeldung
- 📄 n.a.
-

Abb.: Status des Checklistenpunkts auswählen

Parameter	Beschreibung	Beispiele
Label	Definiert den zur Auswahl stehenden Wert. Sie können die Beschriftungen der Auswahlwerte frei definieren.	<ul style="list-style-type: none"> • erledigt ausstehend Warten auf Rückmeldung durch Kunden • 230 V 400 V • 36 °C 97 F
Icon	Optionales Icon, das dem Auswahlwert vorangestellt wird. Zur Auswahl stehen die KIX-Icons sowie eine Reihe von Fontawesome-Icons.	<ul style="list-style-type: none"> • kix-icon-check • fas fa-thermometer
Done	Definiert, ob der Auswahlwert den Fertigstellungsgrad der Checkliste erhöht.	
	Löschen/Verschieben des Auswahlwerts nach oben bzw. unten	
order[type]	Bestimmt die Reihenfolge des Auswahlwertes in der Field Agent App. Sonst nicht erforderlich.	
	Fügt weitere Felder für die Konfiguration von Auswahlwerten hinzu.	

11.2.6.2 Checkliste bereitstellen

Nachdem das Dynamischen Feld vom Typ "Checkliste" angelegt und konfiguriert wurde, kann es in den gewünschten Oberflächen bereitgestellt werden.

Bereitstellung in KIX Start

Die Bereitstellung in KIX Start erfolgt über die Konfiguration der entsprechenden SysConfig-Schlüssel. Jeder Kontext, in dem das Feld eingebunden wird, hat seinen eigenen Konfigurationsschlüssel. Die wichtigsten sind:

Konfigurationsschlüssel	Konfigurationsobjekt	Darstellung der Checkliste
ticket-new-form-group-data	Dialog "Neues Ticket"	Listenpunkte als Eingabe- und Auswahlfelder
ticket-edit-form-group-data	Dialog "Ticket bearbeiten"	Listenpunkte als Eingabe- und Auswahlfelder
ticket-details-info-card	Ticket Detailansicht	Fortschrittsbalken/numerischer Wert inkl. Details
home	Home Dashboard Tabellen	Fortschrittsbalken/numerischer Wert ohne Details

Zur leichteren Bearbeitung empfehlen wir die Verwendung eines externen JSON-Editors (z. B. www.jsonformatter.io¹²). Kopieren Sie den Wert des Schlüssels in den Editor und bearbeiten Sie ihn dort. Minimieren Sie im Editor den bearbeiteten Wert und kopieren Sie ihn zurück nach KIX.

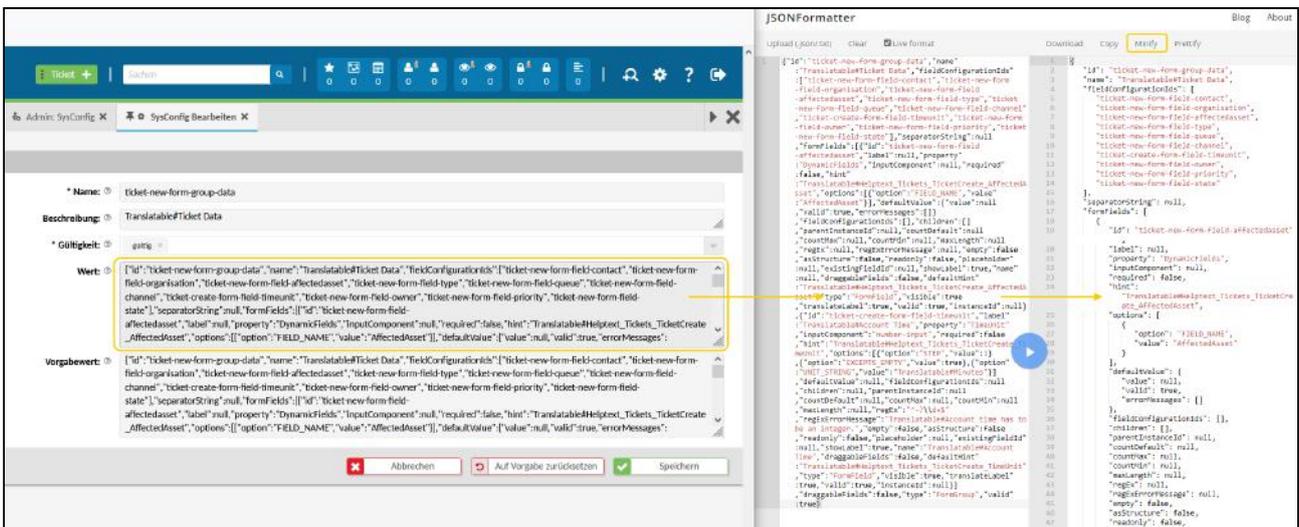


Abb.: SysConfig-Schlüssel in externen JSON-Editor kopieren

Bereitstellung in KIX Pro

In KIX Pro werden Dynamische Felder bequem über [Ticketaktionen](#) (siehe Seite 102) und [Vorlagen](#) (siehe Seite 151) bereitgestellt. Das Dynamische Feld muss dazu lediglich in der jeweiligen Vorlage/Aktion in einem Dropdown ausgewählt werden.

12 <http://www.jsonformatter.io>.

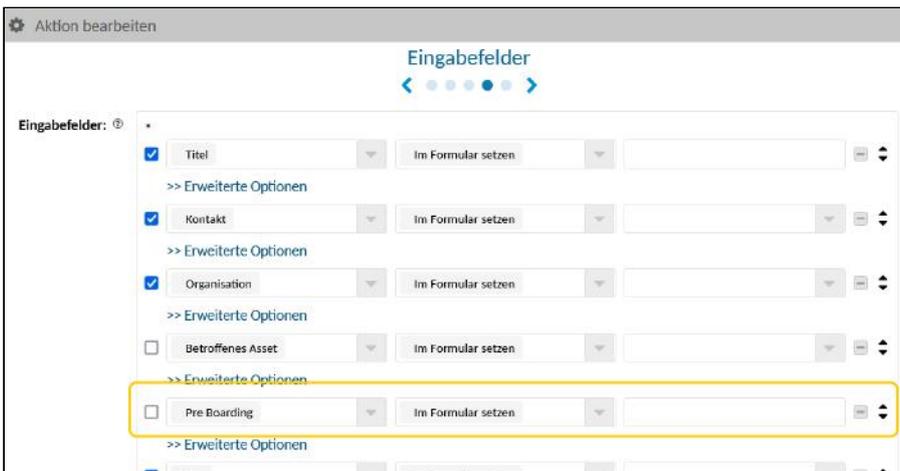


Abb.: Dynamisches Feld (Checkliste) der Ticketaktion "Ticket bearbeiten" hinzufügen und platzieren

Die im SysConfig-Schlüssel hinterlegte Konfiguration dient als Fallback. Zum Bearbeiten eines SysConfig-Schlüssels können KIX Pro Nutzer den integrierten [JSON-Editor](#) (siehe Seite 55) im Menü *System > GUI Konfiguration > Agentenportal* nutzen. Die Verwendung des externen JSON-Editors einschließlich der Schritte zum Hin- und Her-Kopieren des Wertes entfallen somit.

Bereitstellung im Dialog "neues Ticket"

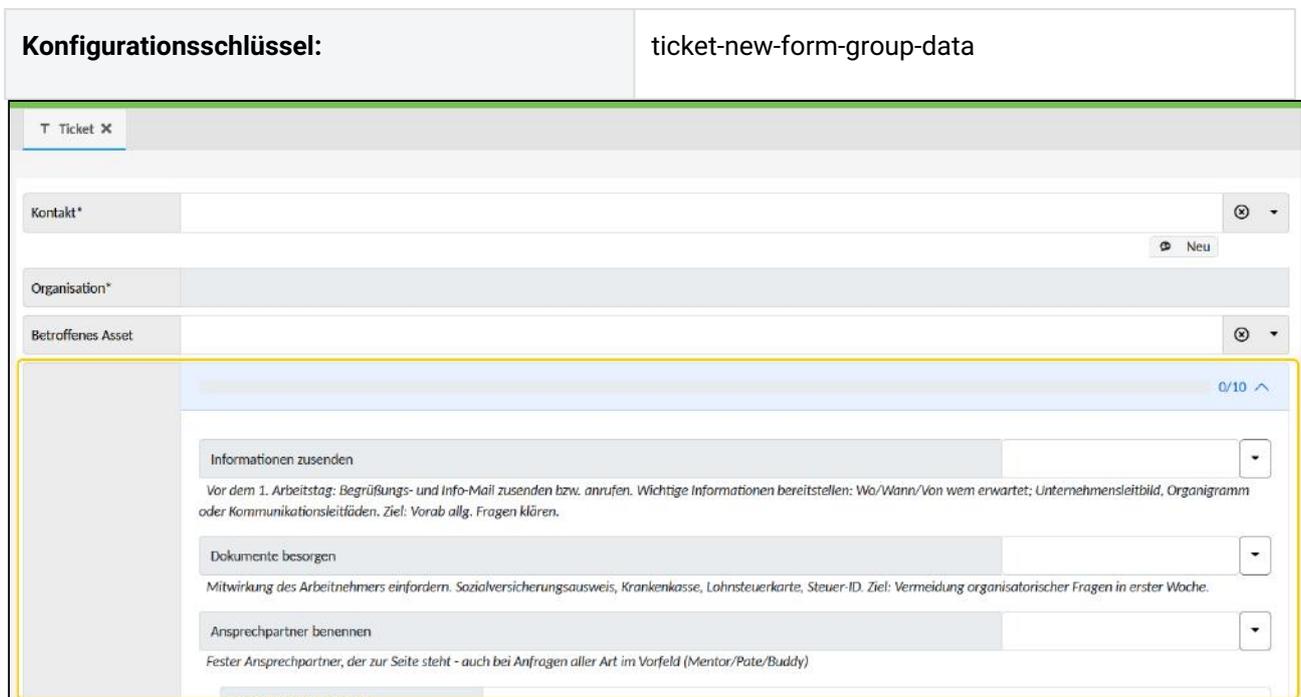


Abb.: Die integrierte Checkliste im Dialog "neues Ticket"

Vorgehensweise für KIX Pro:

In KIX Pro wird der Dialog "neues Ticket" über die Standard-Vorlage "Default - New Ticket Template" gebildet. Fügen Sie dort wie folgt die Checkliste ein:

- Navigieren Sie zu *Workflow > Vorlagen* und öffnen Sie die Vorlage "Default - New Ticket Template" zur Bearbeitung.
- Gehen Sie zu Schritt 2 "Eingabefelder". Am unteren Ende des Dialoges finden Sie ein freies Auswahlfeld. Wählen Sie dort das Dynamische Feld "Pre Boarding" aus.
- Verschieben Sie das Feld optional via Drag & Drop an die gewünschte Stelle im Formular und klicken Sie abschließend auf "Speichern". Danach ist die Checkliste im Dialog "neues Ticket" enthalten.

Vorgehensweise für KIX Start:

Um die Checkliste im Dialog "neues Ticket" bereitzustellen, gehen Sie wie nachfolgend beschrieben vor. Die Checkliste ist dann in jedem neuen Ticket enthalten.

1. Navigieren Sie zu *System > SysConfig*. Öffnen Sie den Schlüssel "*ticket-new-form-group-data*". In diesem Schlüssel ist der Aufbau der Ticketerstellmaske definiert.
2. Kopieren Sie den Quellcode vom Feld "Wert" in einen JSON-Editor. In der maximierten Ansicht können Sie den Quellcode leichter bearbeiten.
3. Fügen Sie den nachfolgenden Quellcode (ohne Kommentare!) in den Codeblock unter `"formFields:[...]"` ein.

Platzieren Sie ihn zwischen 2 Codeabschnitte. Jeder Codeabschnitt beginnt und endet stets mit einer geschweiften Klammer. Zwei aufeinander folgende Codeabschnitte sind durch Komma voneinander getrennt.

Sie können die unter "id" und "value" angegebenen Werte durch die von Ihnen verwendeten Bezeichnungen ersetzen.

```

1      {
2          "id": "ticket-new-form-field-dfpreboardingchecklist", //
           eindeutige ID des Feldes im Schlüssel
3          "label": null,
4          "property": "DynamicFields",
5          "inputComponent": null,
6          "required": false,           //Pflichtfeld: ja/nein
7          "hint": "Translatable#Pre Boarding",
8          "options": [
9              {
10                 "option": "FIELD_NAME",
11                 "value": "PreBoardingChecklist" //Name des Dynamischen
           Feldes wie oben angelegt
12             }
13         ],
14         "defaultValue": {
15             "value": null,

```

```
16         "valid": true,  
17         "errorMessages": []  
18     },  
19     "fieldConfigurationIds": [],  
20     "children": [],  
21     "parentInstanceId": null,  
22     "countDefault": null,  
23     "countMax": null,  
24     "countMin": null,  
25     "maxLength": null,  
26     "regEx": null,  
27     "regExErrorMessage": null,  
28     "empty": false,  
29     "asStructure": false,  
30     "readonly": false,  
31     "placeholder": null,  
32     "existingFieldId": null,  
33     "showLabel": true,  
34     "name": null,  
35     "draggableFields": false,  
36     "defaultHint": "Translatable#optionaler_Hilfetext",  
37     "type": "FormField",  
38     "visible": true,  
39     "translateLabel": true,  
40     "instanceId": null  
41 }
```

```

{
  "id": "ticket-new-form-group-data",
  "name": "Translatable#Ticket Data",
  "fieldConfigurationIds": [],
  "separatorString": null,
  "formFields": [
    {
    },
    {
    },
    {
      "id": "ticket-new-form-field-dfpreboardingchecklist",
      "label": null,
      "property": "DynamicFields",
      "inputComponent": null,
      "required": false,
      "hint": "Translatable#Pre_Boarding",
      "options": [
        {
          "option": "FIELD_NAME",
          "value": "PreBoardingChecklist"
        }
      ],
      "defaultValue": {
        "value": null,
        "valid": true,
        "errorMessages": []
      },
      "fieldConfigurationIds": [],
      "children": [],
      "parentInstanceId": null,
      "countDefault": null,
      "countMax": null,
      "countMin": null,
      "maxLength": null,
      "regEx": null,
      "regExErrorMessage": null,
      "empty": false,
      "asStructure": false,
      "readonly": false,
      "placeholder": null,
      "existingFieldId": null,
      "showLabel": true,
      "name": null,
      "draggableFields": false,
      "defaultHint": "Translatable#optionaler_Hilfetext",
      "type": "FormField",
      "visible": true,
      "translateLabel": true,
      "instanceId": null
    },
    {
      "id": "ticket-create-form-field-timeunit",

```

- Ergänzen Sie im Codeblock "fieldConfigurationIDs" die ID des Dynamischen Feldes (s. Codeblock oben). Die Reihenfolge der Feld-IDs bestimmt die Reihenfolge der Formularfelder in der Ticketerstellmaske.

```

{
  "id": "ticket-new-form-group-data",
  "name": "Translatable#Ticket Data",
  "fieldConfigurationIds": [
    "ticket-new-form-field-contact",
    "ticket-new-form-field-organisation",
    "ticket-new-form-field-affectedasset",
    "ticket-new-form-field-dfpreboardingchecklist",
    "ticket-new-form-field-type",
    "ticket-new-form-field-queue",
    "ticket-new-form-field-channel",
    "ticket-create-form-field-timeunit",
    "ticket-new-form-field-owner",
    "ticket-new-form-field-priority",
    "ticket-new-form-field-state"
  ],
  "separatorString": null,
  "formFields": [

```

5. Minimieren Sie im Editor den Quellcode, um unnötige Leerzeichen und Zeilenumbrüche zu entfernen und kopieren Sie den Quellcode in die Zwischenablage.
6. Fügen Sie den Quellcode aus der Zwischenablage zurück nach KIX ins Feld "Wert" und übernehmen Sie die Änderungen mit "Speichern".
7. Klicken Sie auf "Lade Frontend Konfiguration neu", damit die Checkliste im Dialog "neues Ticket" angezeigt wird.



Bereitstellung im Dialog "Ticket bearbeiten"

Konfigurationsschlüssel:	ticket-edit-form-group-data
---------------------------------	-----------------------------

Damit die einzelnen Listenpunkte beim Bearbeiten des Tickets abgearbeitet werden können, muss die Checkliste auch im Dialog "Ticket bearbeiten" bereitgestellt werden.

Vorgehensweise für KIX Start: Öffnen Sie den SysConfig-Schlüssel "*ticket-edit-form-group-data*" und verfahren Sie wie oben unter "Bereitstellung im Dialog 'neues Ticket' " beschrieben.

Vorgehensweise für KIX Pro: Navigieren Sie zu *Ticket > Aktionen* und öffnen Sie die Aktion "Ticket Edit" zur Bearbeitung. Am unteren Ende des sich öffnenden Dialoges finden Sie ein freies Auswahlfeld. Wählen Sie dort das Dynamische Feld "Pre Boarding" aus. Verschieben Sie das Feld optional via Drag & Drop an die

gewünschte Stelle im Formular und klicken Sie abschließend auf "Speichern". Danach ist die Checkliste im Dialog "Ticket bearbeiten" enthalten.

11.2.6.3 Checklistenstatus anzeigen

Der Status einer Checkliste ändert sich mit dem Fortschreiten ihrer Bearbeitung. Er kann in Form eines Fortschrittsbalkens visualisiert werden. Der Fortschrittsbalken kann entsprechend des Bedarfs in den verschiedenen Oberflächenstrukturen von KIX bereitgestellt werden. Die Bereitstellung erfolgt in KIX Start und in KIX Pro über die Konfiguration der jeweiligen SysConfig-Schlüssel. Jeder Kontext hat seinen eigenen Konfigurationsschlüssel (s. oben).

Für die Bearbeitung der Konfigurationsschlüssel empfehlen wir Nutzern von KIX Start die Verwendung eines externen JSON-Editors.¹³ Nutzer von KIX Pro können den JSON-Editor im Menü *System > GUI Konfiguration > Agentenportal* nutzen (s. oben).

Anzeige in den Ticketdetails

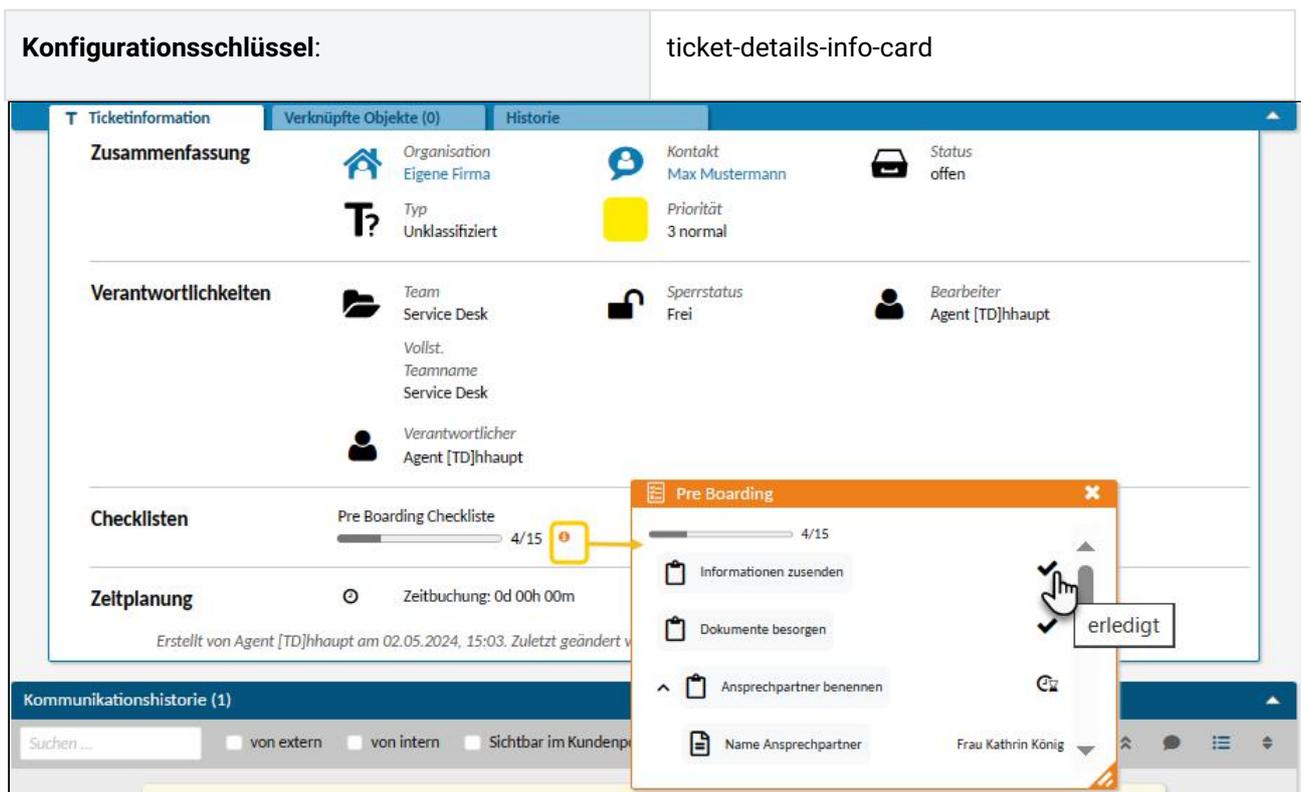


Abb.: Der Checklistenstatus in den Ticketdetails

Vorgehensweise:

¹³ <http://www.jsonformatter.io>.

1. **KIX Start:** Navigieren Sie zu *System > SysConfig* und öffnen Sie den Schlüssel *"ticket-details-info-card"*. In diesem Schlüssel ist die Anzeige der Lane "Ticket Informationen" konfiguriert.
KIX Pro: Navigieren Sie zu *System > GUI Konfiguration > Agentenportal* und suchen Sie den Schlüssel *"ticket-details-info-card"*.

2. **KIX Start:** Kopieren Sie den Quellcode vom Feld "Wert" in einen JSON-Editor. In der maximierten Ansicht können Sie den Quellcode leichter bearbeiten.

3. **KIX Start + KIX Pro:** Kopieren Sie den nachfolgenden Codeblock an gewünschter Stelle in einen `values`-Block. Mit Ihrer Platzierung legen Sie fest, an welcher Stelle der Fortschrittsbalken angezeigt wird.
Ersetzen Sie ggf. den Namen des Dynamischen Feldes (PreBoardingChecklist) durch den von Ihnen verwendeten Namen.
Hinweis: Die Angaben der `conditions` ist optional. Diese definieren die Voraussetzungen für die Anzeige der Checkliste in den Ticketdetails. Im Beispiel wird die Checkliste nicht angezeigt, solange an ihr kein Wert gesetzt ist.

```
[
  {
    "text": "Pre Boarding Checklist",
    "textStyle": "font-weight:bold;margin-bottom:0.5rem",
    "icon": "kix-icon-ci",
    "componentId": "dynamic-field-value",
    "componentData": {
      "name": "PreBoardingChecklist"
    },
    "conditions": [
      {
        "property": "DynamicFields.PreBoardingChecklist",
        "operator": "NE",
        "value": null,
        "useObjectService": false,
        "useDisplayValue": false
      }
    ]
  }
]
```

```

{
  "id": "ticket-details-info-card",
  "name": "Ticket Info Widget",
  "type": "Widget",
  "widgetId": "object-information-card-widget",
  "title": "Translatable#Ticket Information",
  "actions": [],
  "subConfigurationDefinition": null,
  "configuration": {
    "id": "1644499906890",
    "name": "1644499906890",
    "type": null,
    "avatar": [],
    "rows": [
      {
        "title": "Translatable#Summary",
        "style": "",
        "separator": true,
        "values": [
          [
            {type: "button"},
            {type: "button"}
          ]
        ]
      },
      {
        "componentId": "dynamic-field-value",
        "componentData": {
          "name": "PreBoardingChecklist"
        },
        "conditions": [
          {
            "property": "DynamicFields.PreBoardingChecklist",
            "operator": "NE",
            "value": null
          }
        ]
      }
    ]
  }
}

```

Variante:

Sie können einen separaten Abschnitt in den Ticketdetails anlegen, welcher eine oder mehrere Checklisten enthält. Verwenden Sie dazu nachfolgenden Codeblock und fügen Sie ihn an gewünschter Stelle in den `rows` -Block ein.

Anzeige numerischer Wert (2/10 - 2 von 10))

```

{
  "title": "Translatable#Checklists",
  "style": "",
  "separator": true,
  "values": [
    [
      {
        "text": "Pre Boarding Checklist",
        "textStyle": "font-weight:bold;margin-bottom:0.5rem",

```

```

"icon": "kix-icon-ci",
"componentId": "dynamic-field-value",
"componentData": {
  "name": "PreBoardingChecklist"
},
"conditions": [
  {
    "property": "DynamicFields.PreBoardingChecklist",
    "operator": "NE",
    "value": null,
    "useObjectService": false,
    "useDisplayValue": false
  }
]
},
[
  {
    "text": "Optionale 2. Checkliste",
    "textStyle": "font-weight:bold;margin-bottom:0.5rem",
    "icon": "kix-icon-ci",
    "componentId": "dynamic-field-value",
    "componentData": {
      "name": "NameDerCheckliste"
    },
    "conditions": [
      {
        "property": "DynamicFields.NameDerCheckliste",
        "operator": "NE",
        "value": null,
        "useObjectService": false,
        "useDisplayValue": false
      }
    ]
  }
]
],

```

```

{
  "id": "ticket-details-info-card",
  "name": "Ticket Info Widget",
  "type": "Widget",
  "widgetId": "object-information-card-widget",
  "title": "Translatable#Ticket Information",
  "actions": [],
  "subConfigurationDefinition": null,
  "configuration": {
    "id": "1644499906890",
    "name": "1644499906890",
    "type": null,
    "avatar": [],
    "rows": [
      {},
      {},
      {}
    ]
  }
}

```

```

{
  "title": "Translatable#Checklists",
  "style": "",
  "separator": true,
  "values": [
    [
      {
        "text": "Pre Boarding Checklist",
        "textStyle": "font-weight:bold;margin-bottom:0.5rem",
        "icon": "kix-icon-ci",
        "componentId": "dynamic-field-value",
        "componentData": {
          "name": "PreBoardingChecklist"
        }
      },
      "conditions": [
        {
          "property": "DynamicFields.PreBoardingChecklist",
          "operator": "NE",
          "value": null,
          "useObjectService": false,
          "useDisplayValue": false
        }
      ]
    }
  ],
  [
    {
      "text": "Optionale 2. Checkliste",
      "textStyle": "font-weight:bold;margin-bottom:0.5rem",
      "icon": "kix-icon-ci",
      "componentId": "dynamic-field-value",
      "componentData": {
        "name": "NameDerCheckliste"
      }
    }
  ]
]
}

```

4. KIX Start: Minimieren Sie im Editor den Quellcode, um unnötige Leerzeichen und Zeilenumbrüche zu entfernen und kopieren Sie den Quellcode in die Zwischenablage.
5. KIX Start: Fügen Sie den Quellcode aus der Zwischenablage zurück ins Feld "Wert".
6. KIX Start + KIX Pro: Übernehmen Sie die Änderungen mit Klick auf "Speichern".
7. KIX Start + KIX Pro: Klicken Sie auf "Lade Frontend Konfiguration", um die Anzeige in den Ticketdetails zu aktualisieren.



Anzeige im Home Dashboard

Agenten können in KIX Start und KIX Pro die Tabellen in ihren Dashboards selbst anpassen und darüber auch die im System angelegten Dynamische Felder als zusätzliche Spalte(n) ein- oder ausblenden. Somit kann jeder Agent für sich entscheiden, ob und in welcher Tabelle er den Checklistenstatus angezeigt bekommen möchte. Die Grundkonfiguration der Tabelle(n) wird dadurch nicht geändert, nur die benutzerdefinierte Ansicht im Home Dashboard.

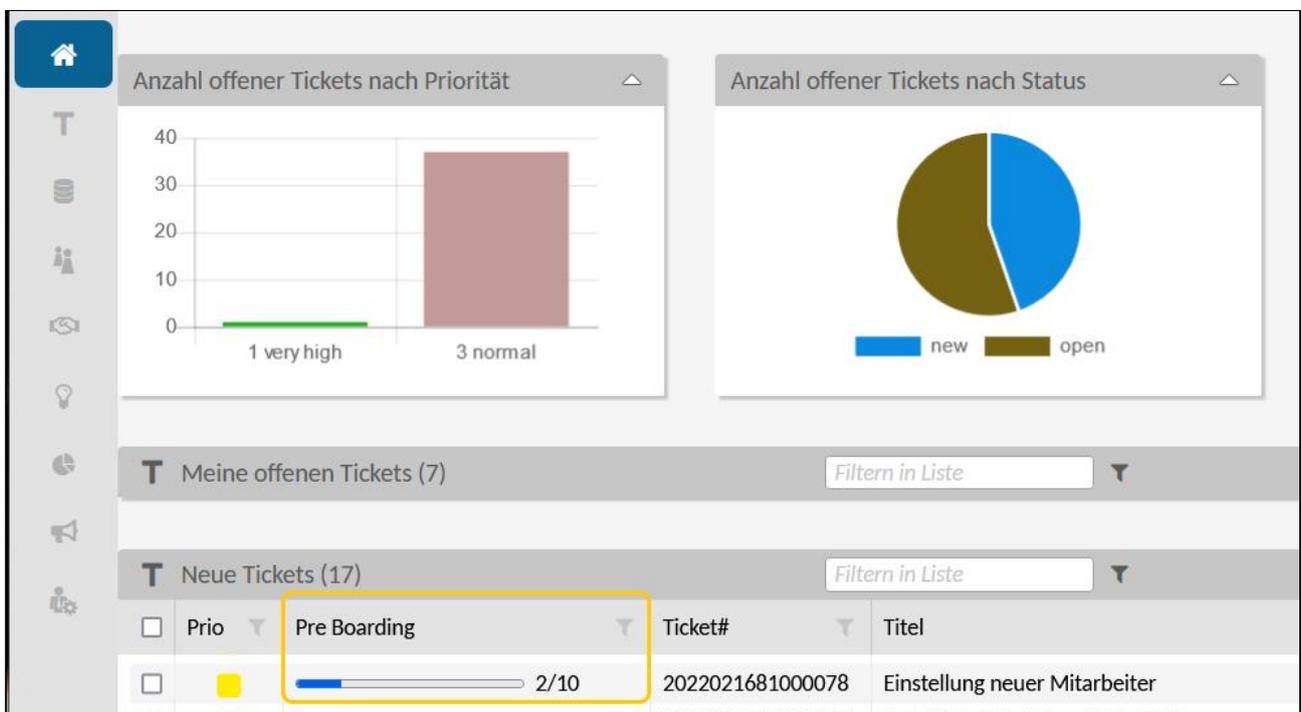


Abb.: Der Checklistenstatus im Home Dashboard

Zum Anpassen klicken Sie im Home Dashboard am unteren Bildschirmrand auf "Dashboard anpassen".

2022082917001064	Suche in CSV exportieren	Application Support
2022082917001037	CMDB-Explorer zeigt keine Einträge	Application Support
2022082917000921	Druck schlägt fehl	Service Desk

[Dashboard anpassen](#)

Abb.: Schaltfläche "Dashboard personalisieren"

In der sich öffnenden Ansicht kann durch hinzufügen einer Tabellenspalte die Checkliste und damit der Fortschrittsbalken in die gewünschte Tabelle integriert werden. Per Drag & Drop kann die Spalte in die gewünschte Reihenfolge verschoben werden.

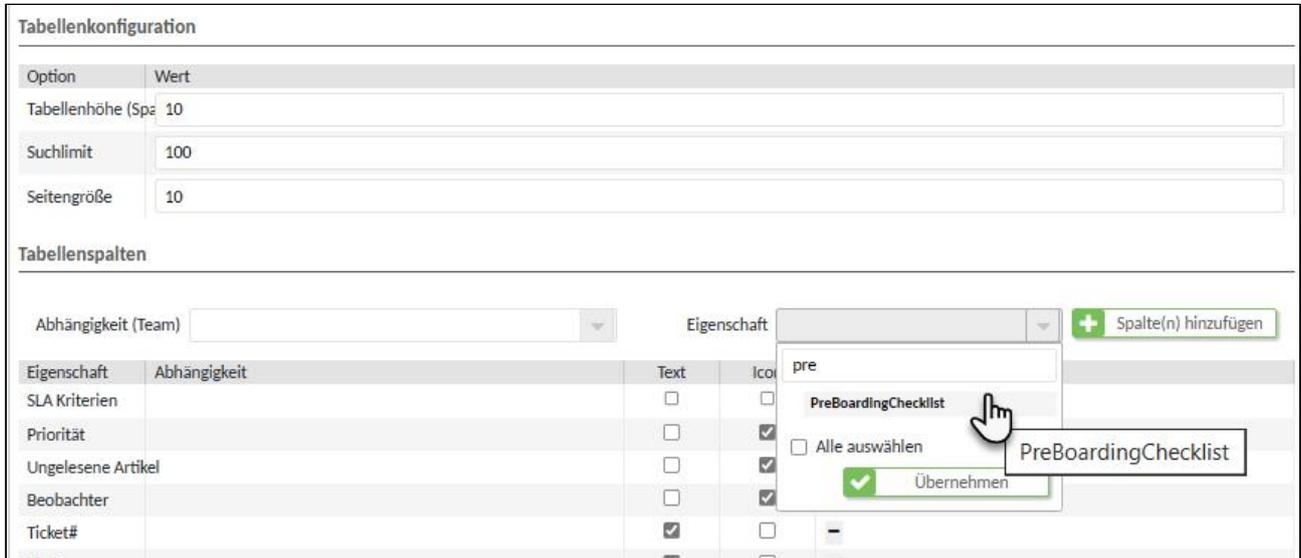


Abb.: Integration des Checklistenstatus in die Konfiguration des Home Dashboards

Damit nicht nur der numerische Wert des Checklistenstatus (2/10) in der Tabelle angezeigt wird, sondern auch der Fortschrittsbalken, muss in der erweiterten Ansicht noch die Komponent-ID "dynamic-field-checklist-cell" angegeben werden. Klicken Sie dazu auf "Erweitert":

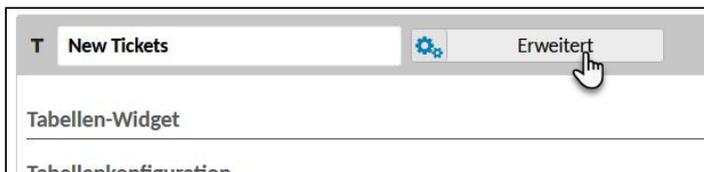


Abb.: Schaltfläche "Erweitert" in der Personalisierung des Home Dashboards

Eigenschaft	Text	Icon	Spaltentitel...	Spaltens...	Sortierbar	Filterbar	Filterliste	Größe in...	Übersetz...	Übersetz...	Größe (Pixel)	Komponenten-ID	Spaltentitel
Priorität	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	65		
Pre Boarding	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	225	dynamic-field-checklist-cell	
Ticket#	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	135		

Abb.: Ergänzung der Komponent-ID

Weiterführende Informationen finden Sie auch im Anwenderhandbuch von KIX Start: Home Dashboard > Home Dashboard personalisieren

Alternative

Konfigurationsschlüssel	home-dashboard-ticket-table-new
--------------------------------	---------------------------------

Alternativ zur Personalisierung des Home Dashboards kann der Checklistenfortschritt auch über die Konfiguration des SysConfig-Schlüssels "*home-dashboard-ticket-table-new*" in die Tabelle "neue Tickets" integriert werden. Dazu wird der Tabelle eine weitere Spalte hinzugefügt, welche den Fortschrittsbalken enthält. Diese globale Änderung am SysConfig-Schlüssel ändert die Grundkonfiguration der Tabelle im Home Dashboard für alle Agenten.

Vorgehensweise

1. KIX Start: Navigieren Sie zu *System > SysConfig*.
KIX Pro: Navigieren Sie zu *System > GUI Konfiguration > Agentenportal*
2. KIX Start + KIX Pro: Öffnen Sie den Schlüssel "*home-dashboard-ticket-table-new*".
In diesem Schlüssel ist der Aufbau der Tabelle definiert.
3. KIX Start: Öffnen Sie den Schlüssel und kopieren Sie den Quellcode vom Feld "Wert" in einen JSON-Editor.
Die rechte Spalte enthält den Quellcode in maximierter Ansicht zur leichteren Bearbeitung.
4. KIX Start + KIX Pro: Fügen Sie den nachfolgenden Codeblock (ohne Kommentare!) in den Editor ein.
 - Scrollen Sie zum Abschnitt `tableColumns[...]`. Darin sind die Tabellenspalten definiert.
 - Platzieren Sie den Codeblock an gewünschter Stelle zwischen zwei bereits vorhandene Tabellenspalten. Eine Tabellenspalte beginnt und endet mit je einer geschweiften Klammer. Zwei aufeinander folgende Tabellenspalten sind durch Komma voneinander getrennt. Die Platzierung im Schlüssel legt fest, an welcher Position die Spalte in der Tabelle eingefügt wird.

```
{
  "id": null,
  "name": null,
  "type": null,
  "property": "DynamicFields.PreBoardingChecklist",
  "showText": true,
  "showIcon": false,
  "showColumnTitle": true,
  "showColumnIcon": true,
  "size": 225,
  "sortable": true,
  "filterable": true,
  "hasListFilter": false,
  "dataType": "STRING",
  "resizable": true,
  "componentId": "dynamic-field-checklist-cell",
  "defaultText": null,
```

```
"translatable": true,
"titleTranslatable": true,
"useObjectServiceForFilter": false
},
```

```
{
  "id": "home-dashboard-ticket-table-new",
  "name": "Translatable#New Tickets Table",
  "type": "Table",
  "objectType": "Ticket",
  "loadingOptions": {},
  "displayLimit": 10,
  "tableColumns": [
    {},
    {
      "id": null,
      "name": null,
      "type": null,
      "property": "DynamicFields.PreBoardingChecklist",
      "showText": true,
      "showIcon": false,
      "showColumnName": true,
      "showColumnIcon": true,
      "size": 200,
      "sortable": true,
      "filterable": true,
      "hasListFilter": false,
      "dataType": "STRING",
      "resizable": true,
      "componentId": "dynamic-field-checklist-cell",
      "defaultText": null,
      "translatable": true,
      "titleTranslatable": true,
      "useObjectServiceForFilter": false
    }
  ]
}
```

5. KIX Start: Minimieren Sie im Editor den Quellcode, um unnötige Leerzeichen und Zeilenumbrüche zu entfernen und kopieren Sie den Quellcode in die Zwischenablage.
6. KIX Start: Fügen Sie den Quellcode aus der Zwischenablage zurück in den SysConfig-Schlüssel ein.
7. KIX Start + KIX Pro: Übernehmen Sie die Änderungen mit "Speichern".
8. Klicken Sie auf "Lade Frontend Konfiguration", um die Tabelle im Home Dashboard zu aktualisieren. Die Agenten müssen Ihr Frontend ebenfalls neu laden. Danach ist die Spalte in der Dashboard Tabelle "neue Tickets" eingefügt und der Fortschrittsbalken wird angezeigt.

11.2.6.4 Verweise

- Dynamische Felder:
 - Dynamische Felder
 - Feldtypen Dynamischer Felder
 - Ein Dynamisches Feld einbinden
 - Werte Dynamischer Felder anzeigen.
- Ticketvorlagen und -aktionen in KIX Pro:
 - [Vorlagen](#) (siehe Seite 151)
 - [Aktionen](#) (siehe Seite 93)
 - [Aktionen anlegen und konfigurieren](#) (siehe Seite 102)
- JSON-Editor
 - <http://jsonformatter.io>
 - KIX Pro: [JSON-Edito](#) (siehe Seite 55) r
- Anpassung Home Dashboard: Home Dashboard > Home Dashboard personalisieren

11.2.7 Datenstruktur von Checklisten

Grundsätzlich bestehen Dynamische Felder vom Typ "Checkliste" aus JSON-Inhalten, die den vollständigen Inhalt der Checkliste beschreiben. Dadurch ergeben sich erweiterte Möglichkeiten für die Verwendung und Konfiguration von Checklisten.

Die Datenstruktur entspricht folgendem Aufbau (in Anlehnung an Erweiterte Backus-Naur-Form (EBNF)):

Struktur Checklisten-Werte

```

CheckList := [ <CheckListItem>+ ];

CheckListItem := {
  "id": "intvalue",
  "title": "Some string describing the task",
  "input": "TextArea | Text | ChecklistState"
  "inputStates":[
    {
      "value": "TitleState1",
      "order": 0 | 1 | 2 | 3 | ...,
      "icon": "kix-icon-xyz | fontawesome-icon",
      "done": true | false
    },
    {
      "value": "TitleState2",
      "order": 0 | 1 | 2 | 3 | ...,
      "icon": "kix-icon-xyz | fontawesome-icon",
      "done": true | false
    },
    #optional further States
    {...}
  ],
  "value": "- | pending | OK | NOK | n.a. | ... or just a string for Text-inputs...",
  "description": "Optional description for the task.",
  "sub": [ optional<CheckList> ]
}

```

Das heißt, bei Erstellung von Aktionen oder Vorlagen kann durch Zuweisung eines JSON-Strings eine Checkliste vollständig neu gesetzt werden. Eine Anpassung des Dynamischen Feldes ist dazu nicht erforderlich. Dies kann auch automatisiert durch einen Job (Konfiguration im Menü: *Automatisierung > Jobs*) erfolgen.

In Verbindung mit dem Variablenfilter "**jq**" können bestehende Checklisten verändert werden, um bspw. automatisiert Checklisten-Aufgaben hinzuzufügen, abzuhaken oder zu entfernen (s. auch <https://jqplay.org>).

Beispiel einer Checkliste

```

1  [
2    {
3      "id": "100",
4      "title": "Announce shut off",
5      "description": "Announce shut off to all possibly affected personell
6      directly or indirectly working with the device.",
7      "input": "ChecklistState",
8      "inputSates":[
9        {
10         "value": "done",
11         "order": 1,
12         "icon": "kix-icon-check",
13         "done": true
14       },
15       {
16         "value": "not completed",
17         "order": 1,
18         "icon": "kix-icon-exclamation",
19         "done": true
20       },
21       {
22         "value": "pending",
23         "order": 1,
24         "icon": "fa-times"
25         "done": false
26       }
27     ],
28     "value": "-"
29   },
30   {
31     "id": "200",
32     "title": "Identify energy source(s)",
33     "description": "Check for connected external and internal energy
34     sources.",
35     "input": "ChecklistState",
36     "inputSates":[
37       {
38         "value": "OK",
39         "order": 1,
40         "icon": "kix-icon-check",
41         "done": true
42       },
43       {
44         "value": "NOK",
45         "order": 1,
46         "icon": "kix-icon-exclamation",
47         "done": true
48     }
49   }
50 ]

```

```

48         "value": "pending",
49         "order": 1,
50         "icon": "fa-times"
51         "done": false
52     }
53 ],
54     "value": "-"
55 },
56 {
57     "id": "300",
58     "title": "Isolate energy source(s)",
59     "description": "Document measures you took in order to isolate energy
60 source(s).",
61     "input": "ChecklistState",
62     "inputSates":[
63         {
64             "value": "OK",
65             "order": 1,
66             "icon": "kix-icon-check",
67             "done": true
68         },
69         {
70             "value": "NOK",
71             "order": 1,
72             "icon": "kix-icon-exclamation",
73             "done": true
74         },
75         {
76             "value": "pending",
77             "order": 1,
78             "icon": "fa-times"
79             "done": false
80         }
81     ],
82     "value": "-",
83     "sub": [
84         {
85             "id": "310",
86             "title": "Isolation by",
87             "description": "",
88             "input": "Text",
89             "value": ""
90         }
91     ],
92     {
93         "id": "400",
94         "title": "Lock & Tag energy source(s)",
95         "description": "Lock energy sources to avoid accidental re-energizing
96 while working on the device. Tag the device as out of order due to
97 maintenance actions.",
98         "input": "ChecklistState",
99         "inputSates":[

```

```

98     {
99         "value": "OK",
100        "order": 1,
101        "icon": "kix-icon-check",
102        "done": true
103    },
104    {
105        "value": "NOK",
106        "order": 1,
107        "icon": "kix-icon-exclamation",
108        "done": true
109    },
110    {
111        "value": "pending",
112        "order": 1,
113        "icon": "fa-times"
114        "done": false
115    }
116 ],
117 "value": "-",
118 "sub": [
119     {
120         "id": "410",
121         "title": "Information by",
122         "input": "TextArea",
123         "value": ""
124     }
125 ]
126 },
127 {
128     "id": "500",
129     "title": "Ensure that equipment isolation is effective",
130     "description": "Before starting maintenance or repair tasks ensure
131 that isolation is working.",
132     "input": "ChecklistState",
133     "inputStates": [
134         {
135             "value": "OK",
136             "order": 1,
137             "icon": "kix-icon-check",
138             "done": true
139         },
140         {
141             "value": "NOK",
142             "order": 1,
143             "icon": "kix-icon-exclamation",
144             "done": true
145         },
146         {
147             "value": "pending",
148             "order": 1,
149             "icon": "fa-times"
150             "done": false

```



```
150     }  
151   ],  
152   "value": ""  
153 }  
154 ]
```

12 Haftungsausschluss KIX Pro

12.1 Haftung für Inhalte

Die Inhalte unserer Seiten und Dokumente wurden mit größter Sorgfalt erstellt. Dennoch kann für eventuelle technische oder redaktionelle Fehler oder Auslassungen in diesem Dokument keine Haftung übernommen werden. Dies gilt auch für jegliche Neben- oder Folgeschäden, die möglicherweise aus Bereitstellung, Funktion oder Gebrauch dieses Materials entstehen.

Hinweise bezüglich Gestaltung, Ergänzungen und eventuell vorhandener Fehler können Sie jederzeit unserem Supportteam (<https://forum.kixdesk.com>) übermitteln. Gern werden wir sinnvolle Anregungen und Verbesserungen aufgreifen und umsetzen.

Alle in diesem Werk mitgeteilten Daten, Merkmale und Beschreibungen können sich jederzeit und ohne besondere Ankündigungen ändern. Personennamen und Firmenbezeichnungen sind frei erfunden. Eventuelle Übereinstimmungen mit real existierenden Personen und Unternehmen sind rein zufällig.

Als Diensteanbieter sind wir gemäß § 7 Abs.1 TMG für eigene Inhalte auf diesen Seiten nach den allgemeinen Gesetzen verantwortlich. Nach §§ 8 bis 10 TMG sind wir als Diensteanbieter jedoch nicht verpflichtet, übermittelte oder gespeicherte fremde Informationen zu überwachen oder nach Umständen zu forschen, die auf eine rechtswidrige Tätigkeit hinweisen.

Verpflichtungen zur Entfernung oder Sperrung der Nutzung von Informationen nach den allgemeinen Gesetzen bleiben hiervon unberührt. Eine diesbezügliche Haftung ist jedoch erst ab dem Zeitpunkt der Kenntnis einer konkreten Rechtsverletzung möglich. Bei Bekanntwerden von entsprechenden Rechtsverletzungen werden wir diese Inhalte umgehend entfernen.

12.2 Haftung für Links

Unser Angebot enthält Links zu externen Webseiten Dritter, auf deren Inhalte wir keinen Einfluss haben. Deshalb übernehmen wir für diese fremden Inhalte keine Gewähr. Für die Inhalte der verlinkten Seiten ist stets der jeweilige Anbieter oder Betreiber der Seiten verantwortlich. Die verlinkten Seiten wurden zum Zeitpunkt der Verlinkung auf mögliche Rechtsverstöße überprüft. Rechtswidrige Inhalte waren zum Zeitpunkt der Verlinkung nicht erkennbar.

Eine permanente inhaltliche Kontrolle der verlinkten Seiten ist ohne konkrete Anhaltspunkte einer Rechtsverletzung nicht zumutbar. Bei Bekanntwerden von Rechtsverletzungen werden wir derartige Links umgehend entfernen.



12.3 Urheberrecht

Die durch die Seitenbetreiber erstellten Inhalte unterliegen dem deutschen Urheberrecht. Die Vervielfältigung, Bearbeitung, Verbreitung und jede Art der Verwertung außerhalb der Grenzen des Urheberrechtes bedürfen der schriftlichen Zustimmung des jeweiligen Autors bzw. Erstellers.

Soweit die Inhalte auf dieser Seite nicht vom Betreiber erstellt wurden, werden die Urheberrechte Dritter beachtet. Insbesondere werden Inhalte Dritter als solche gekennzeichnet. Sollten Sie trotzdem auf eine Urheberrechtsverletzung aufmerksam werden, bitten wir um einen entsprechenden Hinweis. Bei Bekanntwerden von Rechtsverletzungen werden wir derartige Inhalte umgehend entfernen.

Copyright ©2024

KIX Service Software GmbH, Chemnitz

Hergestellt im Auftrag der KIX Service Software GmbH, Chemnitz

Programmentwicklung: KIX Service Software GmbH, Schönherrstraße 8, 09113 Chemnitz

Dokumentation: bluescript, Zeppelinstraße 9, 08451 Crimmitschau

13 Zweckbestimmung für die Verwendung von KIX Pro im medizinischen Bereich

KIX ist weder dafür geeignet, bestimmt, noch zugelassen, Krankheiten, Verletzung oder Behinderungen zu erkennen, verhüten, überwachen, behandeln, zu lindern oder zu kompensieren. KIX dient auch nicht der Untersuchung, der Ersetzung oder der Veränderung des anatomischen Aufbaus oder eines physiologischen Vorgangs. Zudem darf KIX nicht dafür eingesetzt werden, unmittelbar ein diagnostisches oder therapeutisches Produkt zu steuern. KIX ist ferner nicht dazu bestimmt, mit einem Medizinprodukt so verwendet zu werden, dass dieses Medizinprodukt seine Zweckbestimmung erfüllen kann.

KIX ist im medizinischen Bereich ausschließlich für Verwaltungs- und Datenbankaufgaben zu verwenden. Unter Beachtung der vorstehenden Einschränkungen kann KIX daher im medizinischen Umfeld angewendet werden

- für reine Dokumentationszwecke, wie etwa:
 - das übergreifende Management von Equipment in Form von Verwaltung und Inventarisierung der Gerätedaten (Metadaten der Geräte) wie Namen, IP-Adressen, Seriennummern, Verantwortliche, Garantiezeiten, Dienstleister, Betriebsdokumentation, Lizenzinformationen, Kostenstellen sowie der Verwaltung / Organisation der Benutzer, Geräte-Einweisungen;
 - die zentrale Dokumentation aller Tätigkeiten und Änderungen in der IT etwa aufgrund von durchgeführten Instandhaltungs- oder anderen Serviceleistungen (bspw. Medizingeräte-Logbuch);
 - zum Aufbau einer Wissensdatenbank.
- zur Automatisierung und Vereinfachung allgemeiner Verwaltungsabläufe, wie etwa:
 - im Service und technischen Kundendienst, beispielsweise im IT-Service (Störungen, Änderungen, Wartung);
 - in der Haustechnik (Störungen, Änderungen, Reinigung) oder der Medizingerätetechnik.
- für Überwachungszwecke und Kalenderfunktionen, etwa:
 - von zentralen IT-Diensten (Netzwerk, Mail, Datenserver, SAP,...);
 - sowie von Störungs- und Bedarfsmeldungen an die IT, Haustechnik, Medizingerätetechnik;
 - die Planung von regelmäßigen Wartungsarbeiten und die Erinnerung an Wechsel von Verschleißteilen;
 - die Organisation von regelmäßigen Bestellungen und die Einsatzplanung der Service-Techniker.

KIX dient nicht dazu, das Funktionieren von Medizinprodukten zu ermöglichen oder zu gewährleisten und darf zu diesem Zweck auch nicht eingesetzt werden. Sofern KIX im Rahmen der vorstehenden Funktionalitäten zudem einen Datenaustausch in Form einer Schnittstellenfunktion zulässt, weisen wir darauf hin, dass KIX nicht zur Veränderung oder sonstige Einflussnahme auf Daten in medizinisch oder therapeutisch relevanter Art und Weise genutzt werden darf.



Eine Verwendung von KIX im medizinischen Bereich ist nur innerhalb der oben dargelegten Bereiche zulässig. Für eine darüber hinausgehende oder abweichende Verwendung übernimmt die KIX Service Software GmbH keinerlei Haftung.