

# **Maintenance-Modul "ITSM-StandardWorkflows"**

## **Installation und Konfiguration**

**c.a.p.e. IT® GmbH**

**Version: 17.0.0**

## Impressum

**Titel:** Maintenance-Modul "ITSM-StandardWorkflows"  
**Thema:** Installation und Konfiguration  
**Autoren:** c.a.p.e. IT® GmbH  
**Stichworte:** KIX, KIX Professional, ITSM, Workflows  
**Bemerkung:**  
**Version:** 17.0.0  
**Dateiname:** KIXITSMTicketStatusWorkflows.odt

# Inhaltsverzeichnis

<b>1 Allgemeine Hinweise.....</b>	<b>4</b>
1.1 Dokumenteninformationen.....	4
1.1.1 Zielsetzung.....	4
1.1.2 Abgrenzungen.....	4
1.1.3 Änderungen.....	4
1.1.4 Autoren / Verantwortlich.....	4
1.1.5 Dokumentenbesitzer.....	4
1.1.6 Berechtigte Personen.....	4
<b>2 Allg. Anmerkungen und Voraussetzungen.....</b>	<b>5</b>
2.1 Wichtige Begriffe.....	5
<b>3 Erweiterung KIXITSMTicketStatusWorkflow.....</b>	<b>6</b>
3.1 Tickettypen, -status und Ticketstatusworkflows.....	6
3.1.1 Relevante Ticketstatus.....	6
3.1.2 Workflow „Incident“ + „Incident::Disaster“ .....	7
3.1.3 Workflow „Service Request“ + „Information Request“ .....	8
3.1.4 Workflow „Problem“ .....	9
3.1.5 Workflow „RfC::Standard“ .....	10
3.1.6 Konfigurationsoptionen.....	11
3.2 Speicherung des Abschlusscodes.....	11
3.2.1 Konfigurationsoptionen.....	12
3.3 Ticketaktion "Workaround erfassen".....	13
3.3.1 Konfigurationsoptionen.....	13
3.4 Ticketaktion „Schließen“ .....	14
3.4.1 Konfigurationsoptionen.....	14

# 1 Allgemeine Hinweise

## 1.1 Dokumenteninformationen

### 1.1.1 Zielsetzung

Dieses Dokument beschreibt die im Rahmen des Projektes KIXITSMTicketStatusWorkflows umgesetzten spezifischen Änderungen und Erweiterungen. Es stellt eine grobe Konzeption sowie auch eine kurze Anweisung zur Konfiguration dieser Anpassungen dar.

### 1.1.2 Abgrenzungen

Dieses Dokument stellt keine vollständige Dokumentation zur Administration von KIX / KIX Professional dar.

### 1.1.3 Änderungen

#### 1.1.3.1 Änderungshistorie

Version	Datum	Geänderter Abschnitt	Geändert von
0.1.0	14.03.12	Ersterstellung	Anna Litvinova
0.1.1	02.05.12	Anpassungen	Rene Böhm
1.0.0	11.12.12	OLE-Objekte für Workflows	Torsten Thau
1.1.0	29.08.13	Anpassung für OTRS 3.2	Rene Böhm
1.2.0	04.04.14	Anpassung für OTRS 3.3	Rene Böhm
1.3.0	29.04.15	Anpassung für OTRS 4.0 und andere kl. Änderungen	Ricky Kaiser
1.3.1	16.10.15	Kapitel 3.2 – Hinweis zur Übersetzung des ClosureCode	Rene Böhm
1.4.0	21.12.15	Anpassung für OTRS 5.0	Rene Böhm
17.0.0	05.04.17	Anpassungen für KIX Professional 2017	Rene Böhm

#### 1.1.4 Autoren / Verantwortlich

Nachname, Vorname	Organisation	Funktion
Thau, Torsten	c.a.p.e. IT® GmbH	Bereichsleiter Kundenprojekte
Böhm, Rene	c.a.p.e. IT® GmbH	Bereichsleiter Softwareentwicklung
Kaiser, Ricky	c.a.p.e. IT® GmbH	SW-Entwickler

#### 1.1.5 Dokumentenbesitzer

Organisation	Name	Adresse	Kontakt
c.a.p.e. IT® GmbH		Schönherrstr. 8 09113 Chemnitz	info@cape-it.de +49 371 27095 620

#### 1.1.6 Berechtigte Personen

Nachname, Vorname	Organisation	Funktion	Auth.
Thau, Torsten	c.a.p.e. IT® GmbH	Bereichsleiter Kundenprojekte	rw/ro
Böhm, Rene	c.a.p.e. IT® GmbH	Bereichsleiter Softwareentwicklung	rw/ro
Kaiser, Ricky	c.a.p.e. IT® GmbH	SW-Entwickler	rw/ro

## 2 Allg. Anmerkungen und Voraussetzungen

### 2.1 Wichtige Begriffe

- **ACL – Access Control List:** schränkt verfügbare Aktionen oder Auswahlwerte auf Basis von aktuellen Ticketdaten, angezeigten Funktionen o.ä. ein.
- **Ticket Event / Ticket-Ereignis:** ein bestimmtes Ereignis, welches an einem Ticket stattfindet, kann weitere Aktionen/Automatismen auslösen
- **Generic Agent:** eine periodische laufende Aktion die auf Tickets basiert, ähnlich Cron Jobs oder „Scheduled Tasks“ in MS Windows.
- **Konfigurationsoptionen:** beschreiben relevante SysConfig-Einträge, die das Verhalten der Erweiterung maßgeblich beeinflussen. Eine Änderung geschieht auf eigene Verantwortung.

## 3 Erweiterung KIXITSMTicketStatusWorkflow

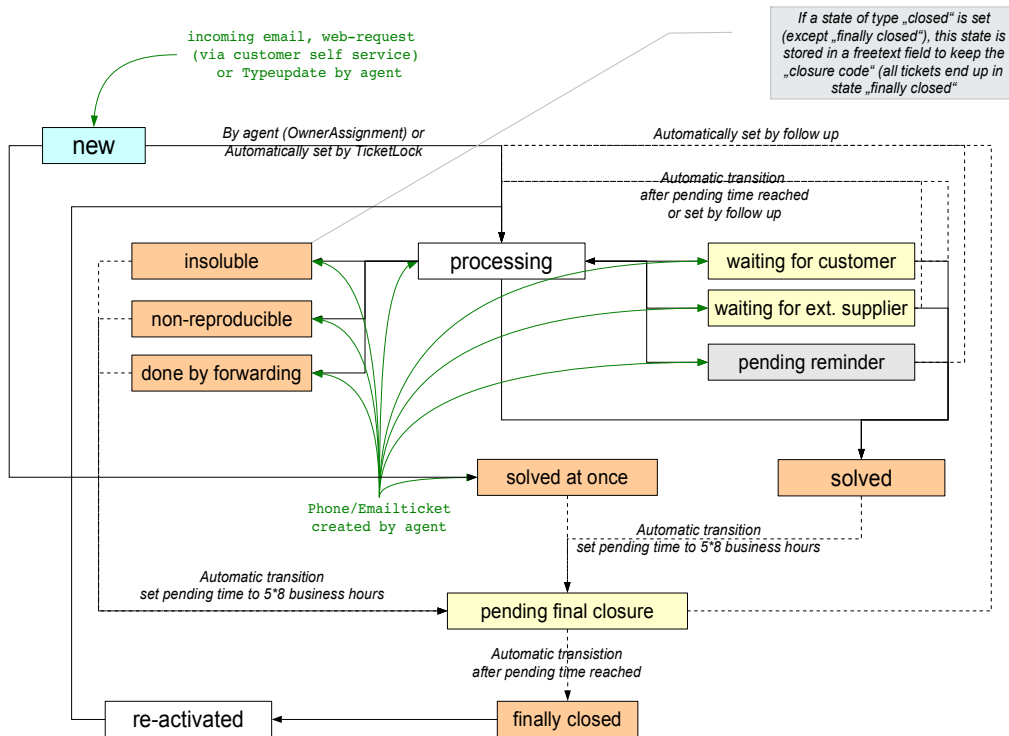
Ziel ist die Abbildung von Tickettyp-spezifischen Statusworkflows. Für jeden Tickettyp existieren passende Status und mögliche Statusübergänge – sowohl automatisch als auch manuell. Die Grundfunktion des Ticketstatus wird von KIX zur Verfügung gestellt. Im Erweiterungspaket wurden die entsprechenden Konfigurationen angepasst. Außerdem wurden zusätzliche, in KIX nicht vorhandene, TicketStateWorkflow-ACLs und Ticket-Events erstellt. Alle noch nicht vorhandenen/deaktivierten Ticketstatus und Tickettypen werden bei der Paketinstallation angelegt/aktiviert. Alle nicht relevanten oder obsoleten Tickettypen sind dabei deaktiviert/umbenannt und stehen nicht mehr zur Auswahl.

### 3.1 Tickettypen, -status und Ticketstatusworkflows

#### 3.1.1 Relevante Ticketstatus

Statusname	Statusyp	Statusname (DE)
closed with workaround	closed	geschlossen mit Workaround
done by forwarding	closed	erledigt durch Weiterleitung
finally closed	closed	geschlossen
insoluble	closed	nicht lösbar
non-reproducible	closed	nicht reproduzierbar
Rejected	closed	abgelehnt
canceled	closed	abgebrochen
solved	closed	gelöst
solved at once	closed	sofort gelöst
solved by change implementation	closed	gelöst durch Changeumsetzung
new	new	neu
problem candidate	new	Problemkandidat
accepted	open	akzeptiert
processing	open	in Bearbeitung
re-activated	open	reaktiviert
workaround documented	open	Workaround dokumentiert
review	open	in Nachkontrolle
pending final closure	pending auto	warte auf finalen Abschluss
waiting for customer	pending auto	warten auf Kundenrückmeldung
waiting for ext. supplier	pending auto	warten auf Rückmeldung ext. DL
pending reminder	pending reminder	warten auf Erinnerung
waiting for change implementation	pending reminder	warten auf Changeumsetzung

### 3.1.2 Workflow „Incident“ + „Incident::Disaster“

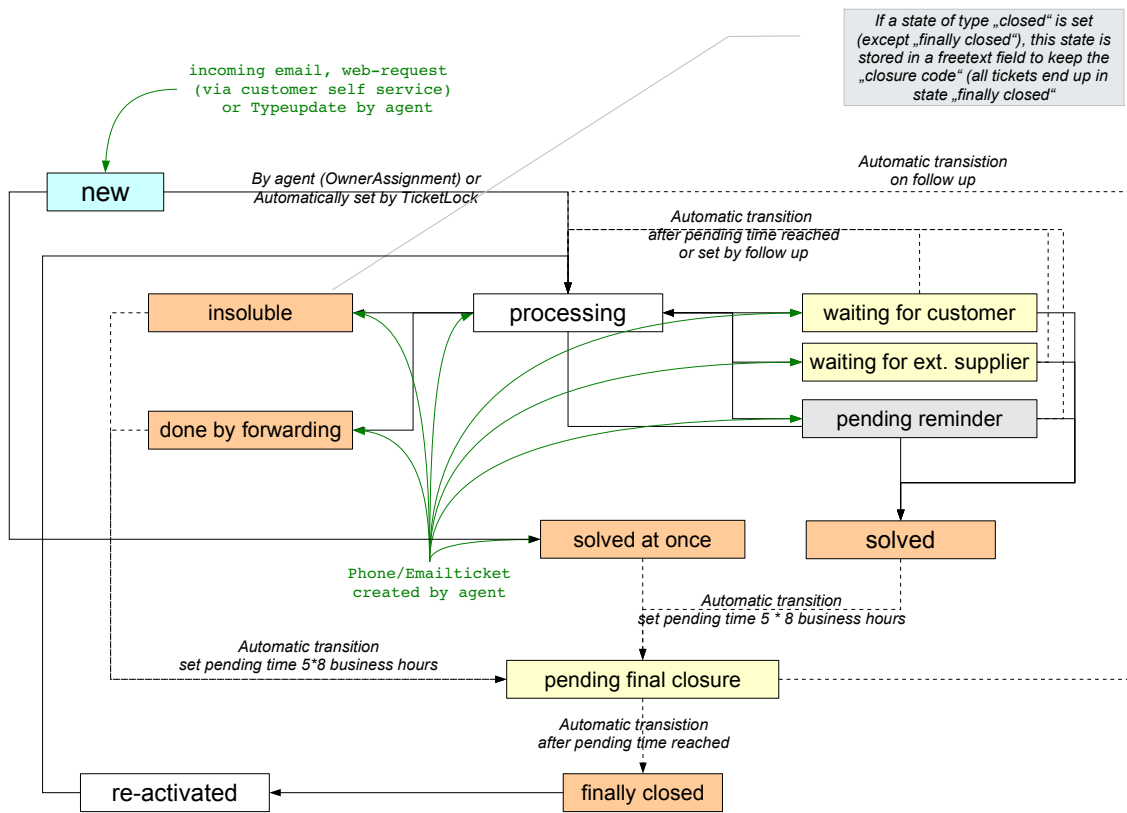


state types: StateType „pending reminder“ StateType „closed“ StateType „open“ StateType „pending auto“ StateType „new“

#### Übersetzungen Statusnamen

new	- „neu“
processing	- „in Bearbeitung“
insoluble	- „nicht lösbar“
non-reproducible	- „nicht reproduzierbar“
done by forwarding	- „erledigt durch Weiterleitung“
pending final closure	- „warte auf finalen Abschluss“
re-activated	- „reaktiviert“
finally closed	- „geschlossen“
solved at once	- „sofort gelöst“
solved	- „gelöst“
waiting for ext. supplier	- „Warten auf Rückmeldung ext. DL“
waiting for customer	- „Warten auf Kundenrückmeldung“
pending reminder	- „Warten auf Erinnerung“

### 3.1.3 Workflow „Service Request“ + „Information Request“



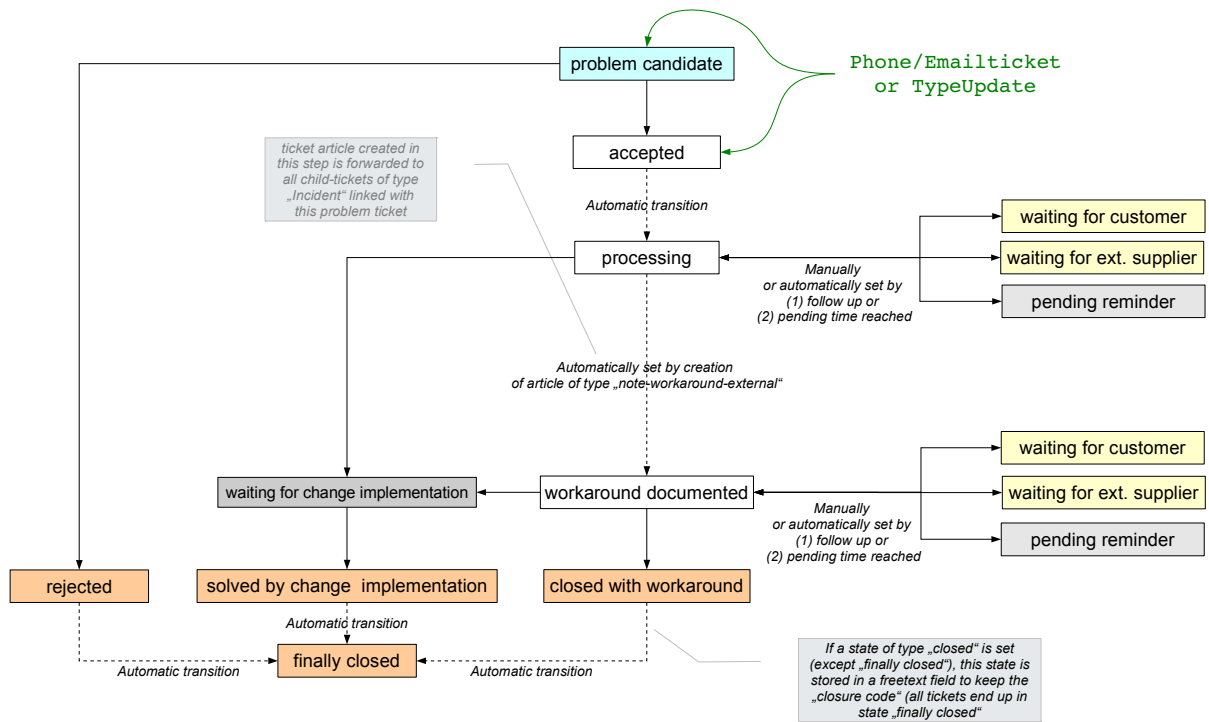
state types: StateType „pending reminder“ StateType „closed“ StateType „open“ StateType „pending auto“ StateType „new“

#### Übersetzungen Statusnamen

new	- „neu“
processing	- „in Bearbeitung“
insoluble	- „nicht lösbar“
non-reproducible	- „nicht reproduzierbar“
done by forwarding	- „erledigt durch Weiterleitung“
pending final closure	- „warte auf finalen Abschluss“
re-activated	- „reaktiviert“
finally closed	- „geschlossen“
solved at once	- „sofort gelöst“
solved	- „gelöst“
waiting for ext. supplier	- „Warten auf Rückmeldung ext. DL“
waiting for customer	- „Warten auf Kundenrückmeldung“
pending reminder	- „Warten auf Erinnerung“



### 3.1.4 Workflow „Problem“

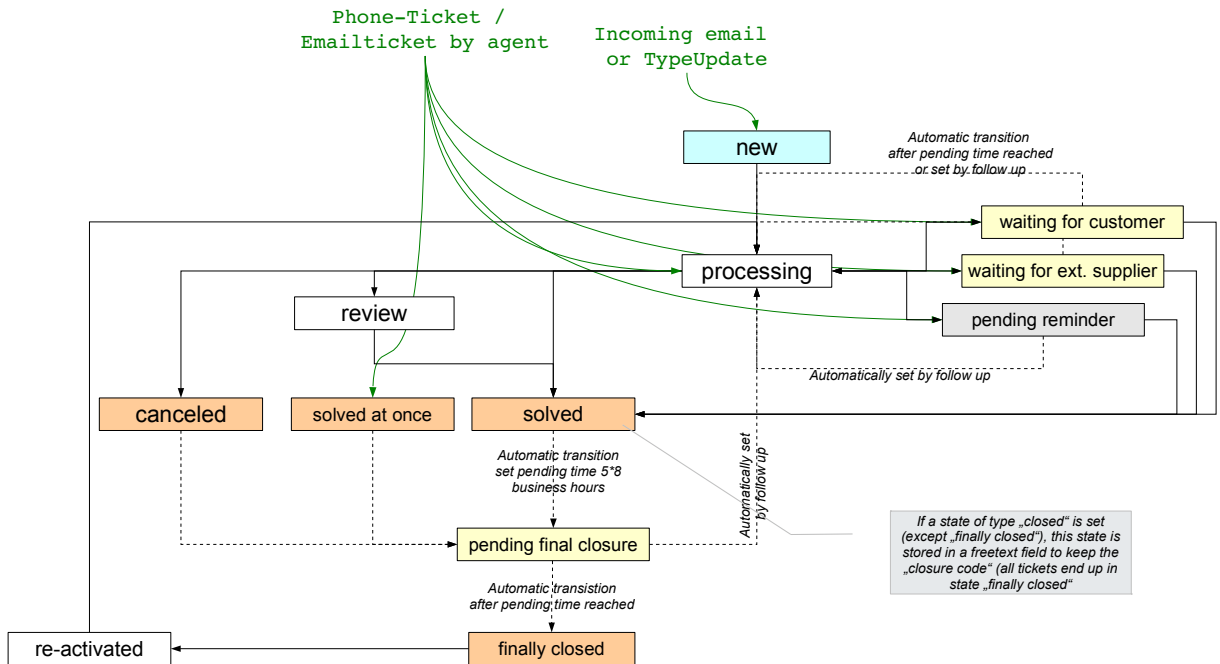


state types: StateType „pending reminder“ StateType „closed“ StateType „open“ StateType „pending auto“ StateType „new“

#### Übersetzungen Statusnamen

problem candidate	- „Problemkandidat“
accepted	- „akzeptiert“
rejected	- „abgelehnt“
workaround documented	- „Workaround dokumentiert“
closed with workaround	- „geschlossen mit Workaround“
solved by change impl.	- „gelöst durch Changeumsetzung“
waiting for change impl.	- „warten auf Changeumsetzung“
processing	- „in Bearbeitung“
finally closed	- „geschlossen“
waiting for ext. supplier	- „Warten auf Rückmeldung ext. DL“
waiting for customer	- „Warten auf Kundenrückmeldung“
pending reminder	- „Warten auf Erinnerung“

### 3.1.5 Workflow „RfC::Standard“



state types: StateType „pending reminder“ StateType „closed“ StateType „open“ StateType „pending auto“ StateType „new“

#### Übersetzungen Statusnamen

new	- „neu“
processing	- „in Bearbeitung“
review	- „in Nachkontrolle“
re-activated	- „reaktiviert“
canceled	- „abgebrochen“
pending final closure	- „warte auf finalen Abschluss“
finally closed	- „geschlossen“
solved at once	- „sofort gelöst“
solved	- „gelöst“
waiting for ext. supplier	- „Warten auf Rückmeldung ext. DL“
waiting for customer	- „Warten auf Kundenrückmeldung“
pending reminder	- „Warten auf Erinnerung“

### 3.1.6 Konfigurationsoptionen

Schlüsselname	Bedeutung
Ticket::Acl::Module ###950_TicketStateWorkflow	Ticket-ACLs zur Definition der möglichen Folgestatus.
TicketStateWorkflow	Einstellungen für den Ticketstatusworkflow, zur Definition der möglichen Folgestatus. Status werden komma-separiert aufgelistet. Als Platzhalter dienen _ANY_ , _PREVIOUS_ und _NONE_.
TicketStateWorkflow::FallbackForPreviousState	Ticketstatus der benutzt wird, falls "_PREVIOUS_" nicht bestimmte werden konnte.
TicketStateWorkflow::DefaultTicketState	Einstellungen für den DefaultTicketstatus, zur Definition der möglichen Folgestatus. Status werden komma-separiert aufgelistet. Als Platzhalter dienen _ANY_ , _PREVIOUS_ und _NONE_.
Ticket::CreateOptions::DefaultTicketType	Vorausgewählter Tickettyp in AgentTicketPhone und AgentTicketEmail
Ticket::EventModulePost ###1-ForceStateChangeOnLock	Ticket Event Modul für automatisches Setzen eines neuen Ticketstatus, nachdem das Ticket gesperrt wurde. Der Schlüssel ist der Tickettyp + "://" + aktueller Status, der Inhalt ist der Status nach dem Sperren.
Ticket::EventModulePost ###700-TicketStateWorkflowTypeUpdate	Aktualisiert Ticketstatus, sofern konfiguriert oder notwendig, nach Änderung des Tickettyps.
TicketStateWorkflow::ForceDefaultTicketState	TicketStateWorkflow::ForceDefaultTicketState
TicketStateWorkflow::PostmasterFollowUpState	Status für ein Ticket, für das ein Follow-up eintrifft (nutzen Sie _PREVIOUS_ als Platzhalter für den letzten Status in der Tickethistorie vor dem aktuellen).
Ticket::PendingAutoStateType	Definiert mögliche Statustypen für Pendng-Tickets.
TicketStateWorkflow::PostmasterFollowUpCheck CustomerIDFrom	Prüft, ob der Absender einer Nachricht als Kundennutzer vorhanden ist und dieselbe Kunden-ID hat, wie am Ticket hinterlegt ist. Ist dem so, wird die Email als "email-external" betrachtet, andernfalls als "email-internal". Standard OTRS-Verhalten, wenn deaktiviert.
TicketStateWorkflowPostmasterFollowUpCheckA gentFrom	Überprüft, ob der Sender als Agent in der DB gespeichert ist, und wenn ja, setzt den Sendertyp auf „agent“.
Ticket::EventModulePost###701-TicketStateWo rkflowAutomaticStateAction	Führt Ticketaktionen nach Statusaktualisierung durch - derzeit beschränkt auf Status- oder Queuwechsel.
TicketStateWorkflowAutomaticStateAction ###NextStateSet	Ermöglicht das automatische Setzen von Status bei Erreichen eines anderen Status.
TicketStateWorkflowAutomaticStateAction ###NextStatePendingOffset	Wenn der automatisch gesetzte Status vom Typ „warten“ ist, wird die hier konfigurierte Zeit als Wartezeit benutzt.
Ticket::StateAfterPending	Definiert, welcher Status gesetzt werden soll, nachdem die Wartezeit erreicht wurde.
Ticket::Type	Aktivieren des Tickettyp-Features
Ticket::Frontend::AgentTicket*###StateType	Mögliche Statustypen bei bestimmten Ticket-Aktionen

## 3.2 Speicherung des Abschlusscodes

Bei Erreichen eines Ticketstatus vom Typ „closed“ („geschlossen“) wird dieser als Abschlusscode in ein dynamisches Feld geschrieben, der Status „finally closed“ ist von diesem Automatismus ausgenommen.

Welches dynamische Feld für die Verwendung des Abschlusscodes verwendet wird ist

konfigurierbar.

**Hinweis:**

Der Abschlusscode wird ohne Übersetzung in der Ticketansicht angezeigt, als 1:1 so, wie er gespeichert wurde. Wenn eine Übersetzung gewünscht ist, kann eine sog. CallMethod für die Ticketansicht in der SysConfig konfiguriert werden.

**Beispiel:**

Erweiterung der Option „Ticket::Frontend::AgentTicketZoom###TicketDataKeys“

```
Key = 2000
Value = CallMethod::TicketObject::GetClosureCodeTranslated
```

Erweiterung der Option „Ticket::Frontend::AgentTicketZoom###TicketDataLabel“

```
Key = CallMethod::TicketObject::GetClosureCodeTranslated
Value = ClosureCode
```

### 3.2.1 Konfigurationsoptionen

Schlüsselname	Bedeutung
Ticket::EventModulePost ###650-TicketWriteClosureCodeToDynamicField	<ul style="list-style-type: none"> <li>De-/Aktiviert Eventhandler zur Hinterlegung des Abschlusscodes in einem dynamischen Feld</li> </ul>
TicketWriteClosureCodeToDynamicField ###DynamicField	<ul style="list-style-type: none"> <li>definiert welches dynamische Feld für die Hinterlegung des Abschlusscodes verwendet werden soll (Defaultwert ist „Closurecode“)</li> </ul>
TicketWriteClosureCodeToDynamicField ###NoClosureCodeStates	<ul style="list-style-type: none"> <li>definiert welche Status vom Typ „close“ nicht als Abschlusscode für die Speicherung in das dynamische Feld betrachtet werden sollen</li> </ul>

### 3.3 Ticketaktion "Workaround erfassen"

Diese Ticketaktion erscheint nur bei Tickettyp „Problem“ im spezifischen Status (siehe Konfiguration für Match::ExcludedAction::HideWorkaroundNote) und ermöglicht das Anlegen eines Artikels analog zur Funktion Notiz mit dem Unterschied des zugeordneten Artikeltyps und einer separaten Ticketaktion. Die Artikeltypen "Workaround (extern)" / "Workaround (intern)" kennzeichnen so Notizen/Artikel, in denen ein Workaround beschrieben ist.

Erfolgt die Anlage eines Workaroundartikels vom Typ „Workaround (extern)“, wird der Status des Problem-Tickets automatisch auf einen konfigurierbaren Wert ('Workaround dokumentiert') gesetzt. Des Weiteren wird dieser Artikel an verknüpfte Kindtickets eines konfigurierbaren Tickettypes weitergegeben. Der weitergegebene Artikel erhält dabei den Typ einer internen Notiz. Auf diese Weise werden Workarounds die in einem Problemticket erarbeitet werden an verbundene Störungstickets weitergegeben.

#### 3.3.1 Konfigurationsoptionen

Schlüsselname	Bedeutung
Ticket::EventModulePost ###800-ProblemWorkaroundNote	<ul style="list-style-type: none"> <li>Workflow-Modul, welches Workaroundartikel und Aktualisierung des Ticketstatus von Problem-Tickets an verknüpfte Tickets weiterleitet.</li> <li>Parameter: <ul style="list-style-type: none"> <li>Event: auslösendes Ticketevent</li> <li>LinkType: zu verfolgender Verknüpfungstyp</li> <li>LinkDirection: zu verfolgende Verknüpfungsrichtung</li> <li>TargetTicketType: Tickettypen an die Weitergabe des Artikels erfolgen soll</li> <li>TriggerTicketType: Tickettyp für den diese Aktion ausgeführt wird</li> <li>TriggerArticleType: Artikeltyp für den diese Aktion ausgeführt wird</li> <li>TriggerTicketNewState: Status des Problem-Tickets nach der Anlage eines Workaroundartikels</li> </ul> </li> </ul>
Frontend::Module ###AgentTicketNoteWorkaroundExternal	<ul style="list-style-type: none"> <li>Frontendmodul-Registration für die Ticket-Aktion "AgentTicketNoteWorkaroundExternal" im Agent-Interface.</li> </ul>
Ticket::Frontend::MenuModule ###420-Note-Workaround-External	<ul style="list-style-type: none"> <li>Anzeige Ticketaktion für eine Workaroundnotiz in der Ticketvorschau.</li> </ul>
Ticket::Frontend:: AgentTicketNoteWorkaroundExternal###*	<ul style="list-style-type: none"> <li>Alle für die Maske AgentTicketNoteWorkaroundExternal notwendigen Konfigurationen, wie z.B. Eingabe Notiz-Feld, Artikeltyp, Statusupdate, Besitzerwechsel etc. sofern diese notwendig sind.</li> <li>Relevante Parameter: <ul style="list-style-type: none"> <li>ArticleTypes</li> <li>Note</li> <li>State und StateDefault</li> <li>...</li> </ul> </li> </ul>
Ticket::Frontend::AgentTicketNoteWorkaroundExternal::KIXSidebarBackend###*	<ul style="list-style-type: none"> <li>De/Aktivierung von Textbausteine, Kunden-Info, Verlinkte Personen für die Maske AgentTicketNoteWorkaroundExternal</li> </ul>
Ticket::ArticleTypelcon	<ul style="list-style-type: none"> <li>Definiert, welche Icons für bestimmte Artikel-Typen</li> </ul>

Schlüsselname	Bedeutung
	verwendet werden.
Ticket::Acl::Module ###803-MatchActionRestrictions	<ul style="list-style-type: none"> <li>Ticket-ACL, um basierend auf Ticketdaten bestimmte Ticket-Aktionen zu deaktivieren.</li> </ul>
Match::ExcludedAction::HideWorkaroundNote	<ul style="list-style-type: none"> <li>Definiert, unter welchen Bedingungen die Ticket-Aktion „Workaround erfassen“ NICHT sichtbar ist. Im Schlüssel sind die Ticket-Eigenschaften enthalten, bei denen die Aktion nicht möglich sein soll.</li> </ul>

### 3.4 Ticketaktion „Schließen“

Diese Ticketfunktion wird nur angezeigt wenn dem aktuellem Ticketstatus ein Folgestatus vom Typ „Geschlossen“ (closed) zugeordnet ist. Dies ist im jeweiligen Ticketstatus-Workflow definiert.

#### 3.4.1 Konfigurationsoptionen

Schlüsselname	Bedeutung
Ticket::Acl::Module ###880-HideAgentTicketClose	<ul style="list-style-type: none"> <li>Ticket-ACL um Funktion "Schließen" auf Basis des aktuellen Ticketstatus einzuschränken.</li> </ul>