



cockpit
IT Service Manager

Tickets - Synchronisation guide

Document FAQ

Table des matières

Introduction.....	3
I.Objectif.....	3
II.Prérequis.....	3
III.Fonctionnement.....	3
IV.Types de connecteur.....	3
Paramétrage portail Cockpit ITSM.....	4
I.Paramétrage.....	4
II.Utilisation.....	4
III.Logs.....	5
Paramétrage XML.....	7
I.Structure.....	7
A.Configuration générale.....	7
B.Configuration du connecteur.....	7
C.Configuration du processus.....	8
II.Configuration des opérations.....	10
A.Configuration de l'opération.....	10
B.Configuration des filtres.....	10
C.Configuration des mappers.....	11
Annexes.....	16
I.Détection des types d'opération.....	16
II.Champs Cockpit ITSM.....	16

Introduction

I. Objectif

- Présenter le fonctionnement du menu « Synchronisation » du module « Tickets ».

II. Prérequis

Pour bien appréhender la synchronisation du module Tickets, il est préférable de :

- Maîtriser le fonctionnement du module Tickets de Cockpit ITSM (priorités, statuts, etc.).
- Avoir des notions de XML (langage de balises génériques).
- Maîtriser le fonctionnement du système externe qui doit être synchronisé avec Cockpit ITSM.

III. Fonctionnement

Les communications entre le portail Cockpit ITSM et le système externe passent par un connecteur et un fichier XML :

- Entrée (Système externe => Cockpit ITSM) : Le connecteur récupère les données des tickets du système externe et les transforme en une liste de clés / valeurs dans un fichier XML, Cockpit ITSM récupère les données et les intègre aux tickets.
- Sortie (Cockpit ITSM => Système externe) : Cockpit envoie au connecteur une liste de combinaisons clés / valeurs dans un fichier XML et la transforme en opérations correspondantes dans le système externe.
- Le connecteur est spécifique pour chaque système externe. Pour synchroniser Cockpit ITSM avec un système externe pour lequel il n'existe pas encore de connecteur, il faut ouvrir un ticket au support pour effectuer la demande d'un nouveau connecteur.

IV. Types de connecteur

Liste des systèmes externes pour lesquels un connecteur existe :

- ServiceNow (ID : « service_now »)
- Jira Software (ID : « jira »)

Paramétrage portail Cockpit ITSM

Menu : Tickets > Configuration du module > Synchronisation

Objectifs : Gérer les systèmes externes avec lesquels les tickets du portail Cockpit ITSM doivent être synchronisés.

I. Paramétrage

Propriétés d'une synchronisation avec un système externe :

Champ	Description
Organisation	Une synchronisation est liée à une organisation, il ne peut avoir qu'une seule synchronisation par organisation. Obligatoire
Application	Texte libre pour indiquer le nom du système externe qui est synchronisé avec le portail Cockpit ITSM, exemple : ServiceNow Obligatoire
Utilisateur final	Indiquer un utilisateur final Cockpit ITSM appartenant à l'organisation et actif. Les tickets synchronisés avec le système externe seront créés dans Cockpit ITSM avec cet utilisateur final. Obligatoire
Statut	Actif / Inactif Quand le statut est inactif, les tickets ne sont pas synchronisés. Quand une synchronisation est désactivée puis réactivée, les tickets créés ou modifiés pendant la période d'inactivité ne sont pas pris en compte. Il est possible de modifier ce fonctionnement et de prendre en compte des actions survenues pendant la période d'inactivité. Si ce besoin existe, prendre contact avec le support Cockpit ITSM.
Données de configuration (XML)	Renseigner dans ce champ directement le code source du fichier XML qui sera utilisé par le connecteur pour communiquer avec le portail Cockpit ITSM et le système externe. Obligatoire

II. Utilisation

Depuis le menu les actions suivantes sont possibles :

- Recharger / Rafraîchir l'affichage : Permet d'afficher les dates et heures des dernières exécutions et derniers échecs.
- Supprimer une synchronisation.
- Copier une synchronisation pour en créer une nouvelle.

Le champ « Dernière exécution » indique la date et l'heure de la dernière synchronisation entre les 2 applications.

Le champ « Dernier échec » indique la date et l'heure de la dernière synchronisation qui n'a pu aboutir.

Si la date de la dernière exécution est plus récente que celle du dernier échec, la synchronisation est techniquement opérationnelle.

III. Logs

Important : Deux cas se présentent :

- Le répertoire des logs se trouve sur la machine hébergeant le portail, si vous êtes en mode SaaS vous n'accédez pas aux logs de synchronisation.
- Si vous êtes en mode Premium, vous pouvez indiquer dans le fichier XML le chemin où vous souhaitez stocker le fichier de logs. Voir le document :

https://faq.cockpit-itsm.com/COCKPIT_FAQ_TICKETING_synchronization_ServiceNow_fr.pdf

Ci-dessous une liste non exhaustive des erreurs qu'il est possible de rencontrer dans les logs.

1. Erreur de connexion

HttpClientErrorException.Unauthorized: 401 Unauthorized

Le portail Cockpit n'a pu se connecter à l'application cible, la connexion est considérée en échec dans le menu de gestion de la synchronisation et sera notée dans le champ « Dernier échec ».

Il faut vérifier tout ce qui en rapport avec l'identification et les accès :

- L'Url d'accès à l'application
- Droits, ID et Mot de passe de l'utilisateur de l'application cible
- etc.

2. Erreur de synchronisation

[TicketSyncJob] Synchronisation stats :

Operations received: 1

Incoming operations successfully synchronized: 0

Outgoing operations successfully synchronized: 0

Incoming operations failed to synchronize: 0

Outgoing operations failed to synchronize: 1

Le champ « Operations received » indique le nombre d'actions à synchroniser entre les 2 systèmes.

Les termes suivants indiquent le sens des éléments à synchroniser :

- « Incoming... » : Du système externe vers Cockpit ITSM
- « Outgoing... » : De Cockpit ITSM vers le système externe

Dans cet exemple il y avait une opération de synchronisation à effectuer de Cockpit ITSM vers le système externe et elle a échoué.

Dans le menu, la synchronisation n'apparaît dans le champ échec mais dans « Dernière exécution », car techniquement la synchronisation a fonctionné, si des éléments n'ont pas été synchronisés, cela provient du paramétrage des systèmes ou du fichier XML.

Les détails des opérations entrantes et sortantes apparaissent également dans la log, ci-dessous quelques exemples :

Unexpected exception while handling ticket log com.cockpit.model.ticket.ticketElements.TicketLog@6f5

Un élément n'a pu être synchronisé, il faut vérifier le paramétrage, notamment si tous les champs associés entre eux dans le fichier XML existent bien dans leurs systèmes respectifs.

[updateTicket] Incoming ticket does not match filters => Skipping ticket line INCXXXXXXXX

Dans cet exemple un élément n'a pas été pris en compte lors de la synchronisation car il ne correspond pas aux filtres paramétrés dans le fichier XML.

Paramétrage XML

Objectifs : Présenter la structure du fichier XML commune à tous les connecteurs. Pour les particularités liées au type de connecteur se reporter aux documents FAQ dédiés à chaque connecteur.

I. Structure

La configuration du fichier XML est généralement structurée en 3 parties :

- La configuration générale : balise <ticketSync>
- La configuration du connecteur : balise <connector>
- La configuration du processus : balise <process>

A. Configuration générale

La balise <ticketSync> contient les éléments suivants :

Paramètre	Description
xmlns	XML Namespace (xmlns) : Ne pas modifier cette valeur, le « xmlns » permet de vérifier la présence de conflit dans le code XML que vous renseignez dans le champ « Données de configuration (XML) ». En cas d'incohérence dans la configuration XML un message d'erreur apparaîtra lors de la sauvegarde, exemple : « XML de configuration invalide: Invalid content was found starting with element 'control' ».
description	Champ texte permettant d'indiquer le nom du système externe. Exemple : « Service Now Madrid »
SyncFrequencyInMinutes	Indique la fréquence de la synchronisation en minutes.
lastUpdateField	Renseigner le champ (un timestamp) permettant d'identifier la dernière opération pour chaque mappage. Cela permet lors des synchronisations de prendre en compte les opérations dans le bon ordre chronologique.

Exemple :

```
<ticketSync xmlns="http://www.cockpit-itsm.com/TicketSyncConfiguration"
description="Système externe" syncFrequencyInMinutes="5">
<connector ... />
<lastUpdateField>Nom_du_Champ</lastUpdateField>
<process ... />
</ticketSync>
```

B. Configuration du connecteur

La balise <connector> contient les éléments suivants :

Paramètre	Description	Valeur par défaut
id	ID correspondant au type de connecteur, cette information est donnée dans la partie Introduction de ce document.	
username	Utilisateur du système externe que le connecteur utilise pour se connecter au système externe.	
password	Mot de passe de l'utilisateur (champ précédent).	
url	URL du système externe utilisée par le connecteur.	
defaultTimeZone	Time Zone par défaut utilisé pour interpréter les champs date du système externe, ne prend pas en compte le time zone de ces champs. La meilleure solution est de tout paramétrer en UTC (champs du système externe et utilisateur final Cockpit ITSM dédié à la création des tickets).	UTC
dateFormat	Format Date / Heure utilisé pour les échanges entre Timestamp.	

Exemple :

```
<connector id="service_now" username="cockpit" password="secret"
url="https://xxxxxx.service-now.com" defaultTimeZone="UTC" dateFormat="yyyy-MM-dd
HH:mm:ss">
  <parameters>
    <entry>
      <string>traceDirectory</string>
      <string>/Users/Username/Temp/Service Now</string>
    </entry>
  </parameters>
</connector>
```

Dans cet exemple le paramétrage correspond à un connecteur « ServiceNow ».

Note : Le paramètre « traceDirectory » permet d'indiquer un chemin pour enregistrer les logs des synchronisations, ce chemin se situe sur la machine hébergeant le portail qui n'est pas accessible en mode SaaS. Par conséquent ce paramètre n'est utile que si vous êtes en mode On premise.

C. Configuration du processus

Il s'agit de processus au sens ITIL du terme, actuellement 3 types de processus sont proposés :

- INCIDENT
- REQUEST (Demande)
- CHANGE (changement)

Ce sont les 3 types de tickets qu'il est possible de créer dans le portail Cockpit ITSM, selon le paramétrage vous pouvez donc choisir de synchroniser un ou plusieurs type de tickets.

La configuration du processus doit être décrite dans la balise « **<process />** ».

Paramètres des processus :

Paramètre	Description
ticketType	Le type de ticket que vous souhaitez synchroniser : <ul style="list-style-type: none"> • INCIDENT • REQUEST • CHANGE
discriminatorField	Attribut permettant d'identifier le champ qui sera utilisé pour trouver le processus renseigné dans « ticketType ». Exemple : « __type__ » correspondant au type de ticket.
discriminatorValue	Valeur utilisée pour identifier le processus. Exemple : « CHANGE » pour les tickets de type « changement ».
ticketTemplate	Modèle de ticket qui sera utilisé pour créer des tickets dans Cockpit ITSM. Utiliser le champ « Reference » visible dans la configuration du modèle.
Paramètres de TicketIdMap	
externalField	Champ permettant d'identifier l'ID du ticket dans le mappage.
cockpitField	Champ permettant d'identifier l'ID du ticket dans le portail Cockpit ITSM.

Exemple :

```
<process ticketType="CHANGE" discriminatorField="__type__" discriminatorValue="CHANGE"
ticketTemplate="Template_Reference">
  <ticketIdMap externalField="XXXX" cockpitField="ticket.externalReference"/>
  <create ... />
  <update ... />
  <close ... />
</process>
```

Dans cet exemple le processus « CHANGE » est utilisé pour tous les tickets ayant pour « __type__ » la valeur « CHANGE ».

La balise <ticketIdMap> permet d'associer les ID d'un ticket :

- L'attribut « externalField » contient l'ID du ticket côté système externe.
- L'attribut « cockpitField » contient « ticket.externalReference » contenant l'ID du ticket dans Cockpit ITSM.

Important : Par convention nous utilisons la notation __KEY__ pour les clés injectées par le connecteur vers le système externe, ceci afin de les différencier des KEY provenant du système externe. Mais ces 2 notations pointent vers une même valeur.

II. Configuration des opérations

Principes :

Les opérations possibles sur les tickets sont :

- CREATE (création d'un ticket)
- UPDATE (ajout d'information sur un ticket)
- CLOSE (clôture d'un ticket)

La configuration d'une opération est composée de :

- La configuration de l'opération.
- Filtres : le but est de filtrer les tickets que l'on ne souhaite pas prendre en compte avant de les soumettre à l'opération.
- Mappers : le but est de transformer les données des tickets dans une description balise / attribut compréhensible par le connecteur.

A. Configuration de l'opération

Détails :

Paramètres des actions CREATE / UPDATE / CLOSE	
Paramètre	Description
direction	Direction pour laquelle l'action est exécutée : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH
attachmentPrivacy	Permet de filtrer les pièces-jointes sortantes selon leur droit d'accès : <ul style="list-style-type: none"> • ALL • PUBLIC • NONE

Exemple :

```
<create direction="BOTH" attachmentPrivacy="ALL">
```

B. Configuration des filtres

Fonctionnement :

- Les filtres sont appliqués un par un dans leur ordre d'apparition dans le code XML.
- Les tickets sont synchronisés si les conditions des filtres sont respectées.
- Un filtre est appliqué sur les tickets uniquement dans la direction indiquée :
 - IN : pour les tickets entrants
 - OUT : pour les tickets sortants
 - BOTH : cette valeur n'est pas valide pour les balises de filtre

- Les conditions du filtre peuvent être inversées.

Détails :

Balise <filter>	
Attribut	Description
direction	À l'action qui a déjà un paramètre « direction », il est possible d'ajouter un filtre sur une direction uniquement : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH (Ne pas utiliser pour les filtres)
Balise <filterRule>	
Attributs	Description
field	Nom du champ recherché
value	Valeur du champ recherché
inverted	Inverse la valeur trouvée

Exemple :

```
<filters>
  <filter direction="IN">
    <filterRule field="__operation__" value="CREATE"/>
    <filterRule field="sys_created_by" value="cockpit" inverted="true"/>
  </filter>
</filters>
```

- Ce filtre s'applique pour les tickets entrants (Système externe => Cockpit ITSM).
- Le filtre prend en compte les opérations de création de tickets.
- Le filtre recherche les tickets créés par l'utilisateur « cockpit » et les exclut avec l'attribut « inverted », il prend donc en compte tous les tickets qui ne sont pas créés par l'utilisateur « cockpit ».

C. Configuration des mappers

Principes :

- Pour chaque opération une liste de « mappers » est paramétrée, le but est de mapper des champs Cockpit ITSM avec les champs du système externe et inversement. Les mappers détaillent les actions qui seront effectuées par l'opération.
- Les mappers sont appliqués un par un par, c'est-à-dire dans leur ordre d'apparition dans le fichier XML, selon les cas le but est de :
 - Modifier les données pour les envoyer au système externe (opérations sortantes).
 - Modifier les tickets Cockpit ITSM pour intégrer les données provenant du système externe (opérations entrantes).

Détails :

Mappers disponibles	
Nom	Description
set	Attribue une valeur fixe à un champ
copy	Copie un champ sans modifier sa valeur
lookup	Associe un champ du système externe à un champ de Cockpit ITSM et inversement via une table de mappage. Actuellement les champs des tickets qui peuvent être synchronisés sont : <ul style="list-style-type: none"> • Statut • Priorité • Application • Environnement • Propriétés spécifiques • Équipe
user	Associe un utilisateur du système externe à un utilisateur Cockpit ITSM
map	Associe un champ à un autre selon une table de mappage statique
template	Associe un champ à un autre via un modèle Velocity. Les données sont disposées dans la variable « \$data ».

Exemple :

```
<mitters>
  <copy externalField="number" cockpitField="ticket.externalReference" direction="IN"/>
  <set externalField="__type__" cockpitField="N/A" value="INCIDENT" direction="OUT"/>
</mitters>
</create>
```

- Pour les opérations entrantes, le champ externalField « number » du système externe est associé au champ cockpitField « ticket.externalReference » (ce sont les ID des tickets).
- Pour les opérations sortantes, le champ externalField « __TYPE__ » est paramétré à la valeur « INCIDENT » (le système externe va donc créer un ticket de type INCIDENT).

Liste détaillée des mappers :

1. Mapper « set »

Configuration du mapper « set »	
Nom	Description
externalField	Champ du système externe
cockpitField	Champ Cockpit ITSM
value	Valeur à attribuer au champ, la valeur est statique.
direction	Direction de l'application du mapper : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH

2. Mapper « copy »

Configuration du mapper « copy »	
Nom	Description
externalField	Champ du système externe
cockpitField	Champ Cockpit ITSM
direction	Direction de l'application du mapper : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH

3. Mapper « lookup »

Configuration du mapper « lookup »	
Nom	Description
externalField	Champ du système externe
cockpitField	Champ Cockpit ITSM
direction	Direction de l'application du mapper : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH
targetType	Type du champ : <ul style="list-style-type: none"> • status • priority • application • environment • specificField • team
valueMap	Table de mappage statique, le symbole (*) peut être utiliser pour accepter tout type de données. Les valeurs pouvant être mappées sont celles des champs listés dans targetType (voir ci-dessus).
valueMap[].externalValue	Valeur du système externe
valueMap[].cockpitValue	Valeur Cockpit ITSM correspondante

4. Mapper « user »

Configuration du mapper « user »	
Nom	Description

externalField	Champ du système externe
cockpitField	Champ Cockpit ITSM
direction	Direction de l'application du mapper : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH
property	Propriété de l'utilisateur Cockpit ITSM à comparer (login, fullName, externalReference). Pour les données entrantes, seul le « login » doit être unique.
userType	Type d'utilisateur possible : <ul style="list-style-type: none"> • CONTACT (Utilisateur final) • OPERATOR

5. Mapper « map »

Le mapper « map » se présente comme le mapper « lookup », sauf que le mapper « lookup » permet le mappage de valeurs Cockpit ITSM (équipe, statut, etc.).

« map » permet le mappage de valeur autres comme des chaînes de caractères, des nombres, des dates, etc.

Configuration du mapper « map »	
Nom	Description
externalField	Champ du système externe
cockpitField	Champ Cockpit ITSM
direction	Direction de l'application du mapper : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH
valueMap	Table de mappage statique, le symbole (*) peut être utiliser pour accepter tout type de données. Les valeurs pouvant être mappées sont celles des champs listés dans targetType (voir ci-dessus).
valueMap[].externalValue	Valeur du système externe
valueMap[].cockpitValue	Valeur Cockpit ITSM correspondante

6. Mapper « template »

Principes :

- Le mapper « template » utilise le langage de modèle Velocity pour injecter des données dans un texte prédéterminé à l'aide de variables.

Exemple : Récupérer le titre ou l’ID d’un ticket pour l’injecter dans le champ titre du ticket de l’autre système.

- Document de référence pour le langage Velocity : <https://velocity.apache.org/engine/2.0/user-guide.html>
- Il est également possible d’ajouter du texte brut.

Détails :

Configuration du mapper « map »	
Nom	Description
externalField	Champ du système externe
cockpitField	Champ Cockpit ITSM
direction	Direction de l’application du mapper : <ul style="list-style-type: none"> • IN (Système externe => Cockpit ITSM) • OUT (Cockpit ITSM => Système externe) • BOTH
<element content>	Indique un modèle Velocity, le modèle doit être intégrer dans une section CDATA.

Exemple :

```

<![CDATA[$data["description"]
Justification
=====
$data["justification"]
Risk and impact analysis
=====
$data["risk_impact_analysis"]
Implementation plan
=====
$data["implementation_plan"]
Test plan
=====
$data["test_plan"]
Backout plan
=====
$data["backout_plan"]
]]>

```

Annexes

I. Détection des types d'opération

Pour savoir si un une opération entrante est une création, une mise à jour ou une clôture, le démarche suivante est utilisée :

- Est-ce que l'ID du ticket entrant est déjà mappé avec un ID Cockpit ?
- Si aucun mappage n'existe, un ticket est créé.
- Si le ticket existe on vérifie s'il s'agit d'une clôture.
- S'il s'agit bien d'une clôture, le ticket est clôturé.
- Dans tous les autres cas il s'agit d'une mise à jour.

II. Champs Cockpit ITSM

Ci-dessous les champs les plus fréquemment utilisés:

Nom	Type	Description
ticket.externalReference	String	External reference of a ticket (Ex : ID de ticket dans ServiceNow)
ticket.status	Lookup	Statut des tickets
ticket.title	String	Titre des tickets
ticket.request	String	Champ requête d'un ticket (création)
ticket.response	String	Champ réponse à un ticket (clôture)
ticket.creationDate	Timestamp	Date de création des tickets
ticket.priority	Lookup	Priorité des tickets
ticket.assignedTeam	Lookup	Équipe en charge des tickets
message	String	Messages échangés dans les tickets
date	Timestamp	Dates des échanges dans les tickets
newStatus	Lookup	Nouveau statut d'un ticket
type	Enum	Les différents types d'échange : <ul style="list-style-type: none"> • ACKNO • ASK_VALID • ASSIGN • CLOSE_DEF • CLOSE_TEMP • CREATION • EMAIL_RECEIVED • EMAIL_SENT • MESSAGE • MIGRATION • REJECT • REOPEN • SAVE • TRANSFER

		<ul style="list-style-type: none">• UNASSIGN• UPDATE_AUTHOR• VALIDATE
--	--	---

Fin du document