



FORCE Bridge API ERP-Interface

Version 5.12

Handbuch



Dokument: Handbuch – FORCE Bridge API:
ERP-Interface



Freigabedatum: 19.05.2023



Dokumentversion: 1



Autor: FORCAM GmbH

Inhaltsverzeichnis

1	Über dieses Dokument	5
1.1	Zielgruppe	5
2	Einführung FORCE Bridge API.....	6
2.1	Technisches Grundwissen	6
2.1.1	Erkundbare API	6
2.1.2	Collections	7
2.1.3	Versionierung.....	7
2.1.4	HTTP-Anfragen.....	7
2.1.5	HTTP-Statuscodes	8
2.1.6	Authentifizierung.....	8
2.2	FORCE Bridge API verwenden	9
2.2.1	OAuth-Token anfordern	10
2.2.2	POST-Anfragen versenden.....	10
3	Konfiguration Webservices	12
3.1	URLs.....	12
3.2	API-Clients	13
3.3	Job Scheduling	15
3.4	Kundenspezifische Einstellungen	16
4	FORCE Bridge API als digitales Abbild einer Produktionsstätte.....	17
4.1	Grundkonzepte und Objekte in der Produktion.....	17
4.1.1	Einheiten aus der Produktion	17
4.1.2	Ressourcen der FORCE Bridge API.....	20
4.2	Schichten und geplante Betriebszeiten von Arbeitsplätzen	21
4.2.1	Domänenspezifisches Wissen.....	21
4.2.2	Arbeitsplatz-Ressourcen in FORCE Bridge API.....	22
5	Callbacks und Ereignisse	25
5.1	Callbacks	25
5.1.1	Callback-Anfrage an den externen Webservice	25
5.1.2	Wiederholungen	25
5.1.3	Ein Callback erstellen	25
5.1.4	Alle registrierten Callbacks abrufen	27
5.1.5	Einen registrierten Callback löschen	27

5.1.6	Ereignistypen	28
5.1.7	Ereignisse empfangen.....	28
5.1.8	Selbst definierte Ereignisse an Applikationen oder Drittsysteme senden	28
6	ERP-Interface – Daten und Events via XML.....	30
7	ERP-Download- und ERP-Upload-Datendienst	31
8	Konfiguration ERP-Interface	32
8.1	ERP-Download	32
8.2	ERP-Upload	33
8.2.1	ERP-Upload-Verbindungskonfiguration.....	34
8.3	Filter – Ungenutzte geschlossene AVOs	35
8.4	ERP-Steuerschlüssel	36
8.5	Anbindung über Infor ION	36
8.6	Zuweisungsverfahren der AVOs auf Arbeitsplatzgruppen.....	36
8.6.1	Stückzeitfaktor.....	38
8.7	Arbeitsplatz-spezifische Konfiguration.....	39
8.7.1	ERP-Upload Aktivierung.....	39
8.7.2	ERP Zykluszeit	40
8.7.3	Ausschuss melden an ERP.....	40
9	Datenstrukturen	43
9.1	XSL Basis-Datentypen	43
9.2	XSL Komplexe Datentypen.....	44
10	Download	46
10.1	Download Basis-Datentypen	46
10.1.1	Zeitfelder	46
10.1.2	Multilinguale Texte	47
10.1.3	Datensatz-Aktion	48
10.1.4	ERP-Schlüssel	49
10.1.5	Material	49
10.1.6	Benutzerdatenfelder.....	50
10.2	Auftragsdaten	51
10.2.1	Auftragskopf	51
10.2.2	Sachmerkmale des Materials.....	53
10.2.3	Arbeitsvorgänge.....	54
10.2.4	Komponenten	59

10.2.5	Fertigungshilfsmittel	60
10.3	Schichtdaten	61
10.4	Personaldaten	63
10.5	Daten zu Gemeinkosten	64
10.6	Lagerdaten aus Lagerverwaltungssystem	67
10.7	Antwort auf Mengenanfragen	68
11	Upload	69
11.1	Upload Basis-Datentypen	70
11.1.1	ERP-Schlüssel	70
11.1.2	Nachrichten an ERP-System	70
11.2	Nachrichten zu Arbeitsvorgängen (AVOs)	71
11.2.1	Wechsel einer AVO-Phase	72
11.2.2	AVO -Mengenbuchungen und Begründungen	72
11.2.3	AVO-Dauer	73
11.2.4	Komponenten Mengen	73
11.2.5	Zusammenbau von Komponenten	74
11.3	AVO-Zeitplanung	75
11.4	Gemeinkosten-Meldungen	75
11.5	Lagerdaten an Lagerverwaltungssystem	75
11.6	Mengenanfragen	77
11.7	Serialisierung und Korrekturfähigkeit	77
12	Anhang	78
12.1	Abkürzungen und Begriffe	78

1 Über dieses Dokument

Dieses Dokument beschreibt die Verwendung der FORCE Bridge API und des ERP-Interfaces für Daten und Events via XML.

- ❗ Aus Gründen der besseren Lesbarkeit wird im Text verallgemeinernd das generische Maskulinum verwendet. Diese Formulierungen umfassen jedoch gleichermaßen alle Geschlechter und sprechen alle gleichberechtigt an.

1.1 Zielgruppe

Dieses Handbuch setzt Kenntnisse im Umgang mit der FORCE MES FLEX (im Folgenden nur noch MES FLEX genannt), FORCE Bridge API und dem ERP-Interface voraus. Sollten Sie dazu keine oder wenige Kenntnisse haben, nehmen Sie sich die Zeit, sich mit den Grundlagen vertraut zu machen.

- ❗ Wir empfehlen Ihnen die Nutzung unserer Academy: <https://forcam.com/academie/>
Die FORCAM Academy bietet das Wissen zum effektiven Einsatz der Methoden für die digitale Transformation und der Technologien für die Smart Factory.
Unser Institutsteam begleitet Sie auf Basis von Lean Manufacturing und TPM-Methoden, Veränderungen im Unternehmen einzuleiten und die Technologien richtig einzusetzen.

2 Einführung FORCE Bridge API

FORCE Bridge API ist die Programmierschnittstelle zwischen den Objekten der Produktionsumgebung und der IT-Infrastruktur eines Produktionswerks.

FORCE Bridge API dient als Brücke zwischen dem realen Produktionsstandort, den IT-Systemen und den Anwendungen, die für eine effektive und effiziente Organisation eingesetzt werden.


Die Industrie IoT-Plattform erfüllt zwei Funktionen:

- Die FORCE Bridge API liefert ein vollständiges digitales Abbild einer Produktionsanlage mit ihren relevanten Objekten samt ihren Zuständen. Objekte sind reale Einheiten wie Personen, Maschinen und Werkzeuge, aber auch abstrakte Einheiten wie Produktionsaufträge oder Vorgänge.
- Die FORCE Bridge API gewährleistet die organisatorische Interoperabilität¹ von Personen, Produktionsanlagen und IT-Systemen einer Produktionsanlage zur Maximierung der Ressourceneffektivität und Prozesseffizienz.

2.1 Technisches Grundwissen

FORCE Bridge API verbindet Ihre Anwendung mit einer IoT-Plattform. Ihre Anwendung kann die API verwenden, um auf Daten zuzugreifen, Befehle an die FORCE Runtime zu senden oder um auf Ereignisse zu warten, die im System auftreten.

- Der komplette Zugriff auf die API erfolgt über HTTP(S). Anforderungs-Payloads werden als *application/json* formatiert. Antwort-Payloads sind als *application/json* oder *application/hal+json* formatiert.
- **OAuth2** wird für alle Authentifizierungen verwendet. Alle API-Anforderungen müssen authentifiziert werden, ansonsten gibt es eine „401 Unauthorized“-Fehlerantwort (siehe **Authentication**).
- FORCE Bridge API ist eine **erkundbare API** und stellt Ressourcen bereit.
- Jede Ressource wird durch ihren universellen eindeutigen Bezeichner (UUID = universal unique identifier) identifiziert.

 Die komplette Swagger-Spezifikation der FORCE Bridge API ist **hier** verfügbar.

2.1.1 Erkundbare API

Antwort-Payloads werden mit Hyperlinks unter Verwendung der **Hypermedia Application Language** (HAL) versehen. HAL ist ein einfaches Format, das eine Reihe von Konventionen zum Ausdruck von Hyperlinks in JSON bereitstellt. Im Grunde ist es JSON mit Hyperlinks. Dies ermutigt Entwickler, Anwendungen zu erstellen, die diese Hyperlinks anstatt hart kodierter URLs verwenden, um zwischen Ressourcen zu navigieren.

Die folgenden *Felder* sind bei jeder Ressource zu erwarten:

- *properties* – reguläre Eigenschaften der Ressource
- *_links* – JSON-Objekt mit einem Satz von URL-Pfaden relativ zur API-Basis-URL
- *_embedded* – JSON-Objekt mit anderen Ressourcen, die in die aktuelle Ressource eingebettet sind

¹ Interoperabilität beschreibt die Fähigkeit, Informationen zwischen zwei unterschiedlichen Systemen auszutauschen und nutzbar zu machen.

2.1.2 Collections

Ein Satz von Ressourcen ist eine *Collection*.

2.1.3 Versionierung


Die API wird durch Bezeichner im URI überarbeitet.

Die Version der API wird nur dann revidiert (erhöht), wenn eine wesentliche Änderung in einem der Webservices auftritt. Wesentliche Änderungen sind:

- Entfernung einer JSON-Eigenschaft
- Neubenennung einer JSON-Eigenschaft
- Allgemeine Umstrukturierung des Designs bestehender Darstellungen

Wesentliche Änderungen sind nicht:

- JSON-Eigenschaften hinzufügen
- Neue Webservices hinzufügen

-  Die Versionierung ist global. Das bedeutet, dass ein Webservice dieselbe Antwort für verschiedene Versionen ausgeben kann, wenn an diesem spezifischen Webservice keine Änderung vorgenommen wurde.

Möglicherweise fügt eine neue Version neue JSON-Eigenschaften zur Antwort hinzu. Dadurch wird die Webservice-Version nicht erhöht. Um die Abwärtskompatibilität zu gewährleisten, sollte jeder Client, der mit der FORCE Bridge API kommuniziert, nur bei Bedarf neue Eigenschaften verarbeiten.

2.1.4 HTTP-Anfragen

Nach Möglichkeit verwendet die API für jede Aktion geeignete **HTTP-Methoden**.

Methode	Beschreibung
GET	Für das Abrufen von Ressourcen
POST	Zum Erstellen von Ressourcen oder Ausführen einer Aktion, u.a.
PUT	Für das Ersetzen von Ressourcen
DELETE	Für das Löschen von Ressourcen

Tabelle 1: HTTP-Methoden

Schlüssel	Wert
authorization	Authentifizierungstoken
accept header	<i>application/json</i> oder <i>application/hal+json</i>
accept language	<i>en-GB</i> (Standard), <i>en-US</i> , <i>de-DE</i> , <i>zh-CN</i> , etc. (alle Sprachen, die in der Workbench konfiguriert sind)

Tabelle 2: http-Methoden: Header



2.1.5 HTTP-Statuscodes

Die API gibt, für jede gestellte Anfrage entsprechende **HTTP-Statuscodes** zurück.

Code	Beschreibung
200 OK	Erfolg
201 Created	Die Anfrage war erfolgreich und es wurde eine neue Ressource erstellt. Die Standort-Überschrift der Antwort enthält den URI der neuen Ressource.
202 Accepted	Die Anfrage wurde akzeptiert und ist in Bearbeitung, aber noch nicht abgeschlossen.
204 No content	Die Anfrage war erfolgreich, aber der Server gibt keinen Inhalt zurück. Für die meisten DELETE-Anfragen ist dies die häufigste Antwort.
400 Bad request	Die Anfrage war ungültig oder kann nicht bearbeitet werden. Eine begleitende Fehlermeldung erklärt mehr.
401 Unauthorized	Die Authentifizierungsberechtigung fehlte oder war falsch.
403 Forbidden	Die Anfrage wird erkannt, aber sie wurde abgelehnt oder der Zugriff ist nicht erlaubt. Eine begleitende Fehlermeldung erklärt mehr.
404 Not found	Die angeforderte URI ist ungültig oder die angeforderte Ressource (z. B. ein Ereignis) existiert nicht.
415 Unsupported media type	The request entity has a content type that the server does not support.
500 Internal server error	Es ist ein unerwarteter Fehler aufgetreten.
504 Gateway timeout	Der als Gateway oder Proxy fungierende Server hat keine rechtzeitige Antwort von dem Upstream-Server erhalten, auf den er zugreifen musste, um die Anfrage zu vervollständigen.

2.1.6 Authentifizierung

Die FORCE Bridge API verwendet das OAuth 2.0-Protokoll zur Authentifizierung.

-  OAuth ist nur sicher, wenn es über das TLS/SSL-Protokoll verwendet wird, daher müssen alle Anfragen und API-Endpunkte HTTPS verwenden.
-  Token sind Passwörter: Die Client-ID, das Client-Geheimnis und die Zugriffstoken Ihrer Anwendung sollten als genauso sensibel wie Passwörter betrachtet und nicht weitergegeben werden.

Beim Zugriff auf die FORCE Bridge API, folgen alle Anwendungen einem Grundmuster. Im Allgemeinen werden diese drei Schritte befolgt:

- Zugriffstoken beziehen
 - Um eine Authentifizierungsanfrage an die API zu stellen, muss Ihre Anwendung zunächst unter Mithilfe des Client Credentials Flow, ein OAuth Zugriffstoken erhalten.

- Zugriffstoken für die Ausführung von Anfragen verwenden
Nachdem Ihre Anwendung ein Zugriffstoken erhalten hat und API-Anfragen stellt, sendet sie die *access_token* -Eigenschaft in einer Token-Antwort in einem **Bearer authorization header** (deutsch: Träger-Authorisierungs-Überschrift).
- Zugriffstoken beim Ablauf aktualisieren
Zugriffstoken haben eine begrenzte Lebensdauer. Wenn Ihre Anwendung nach Ablauf der Lebensdauer eines Zugriffstokens eine Anfrage stellen muss, muss sie ein neues Zugriffstoken erhalten.

2.2 FORCE Bridge API verwenden

Im Gegensatz zu allen anderen HTTP-Methoden muss POST für verschiedene Zwecke verwendet werden, um in bestimmten Situationen unbegrenzte Aufrufe zu vermeiden.

Im engeren Sinne wird POST verwendet, um eine oder mehrere neue Ressourcen zu erzeugen. Dazu wird die entsprechende Sammlung mit POST aufgerufen:

POST/{collection}

Nach dem Hypermedia-Prinzip der RESTful-Architektur wird bestimmt, ob eine Ressource erstellt werden kann oder nicht, wenn die zugehörige Sammlung einen entsprechenden Hyperlink hat. Zum Beispiel bietet die FORCE Bridge API in der Sammlung der Werkzeug-Montageaufträge einen Hyperlink zur folgenden Methode, mit der Anwendungen neue Werkzeuge erstellen können:

POST/tools/assemblyOrders

Alle Eigenschaften des Werkzeugmontageauftrags werden mit dem Methodenaufwurf an die IoT-Plattform übertragen, die wiederum mit einer Darstellung der neu angelegten Ressource antwortet, wenn der Aufruf erfolgreich war. Insbesondere wird in der Antwort die UUID des neu erstellten Werkzeugs bekannt gegeben.

Eine weitere Verwendung von POST ist das Ändern des Ressourcenzustands. Dazu wird die betreffende Einzelressource mit POST und einer Angabe des neuen Zustands aufgerufen:

POST/{collection}/{id}/{state}

Die FORCE Bridge API ermöglicht es einem Werkzeugverwaltungssystem, den Zustand eines Werkzeug-Montageauftrags z. B. durch den folgenden Aufruf zu setzen:

POST/tools/assemblyOrders/{toolAssemblyOrderId}/{toolAssemblyOrderStateId}

Obwohl auch die PUT- oder PATCH-Methode für diesen Zweck verwendet werden könnte, wären die Aufrufe wesentlich umständlicher, da bei PUT auch Eigenschaften übergeben werden müssen, die nicht geändert werden. Andererseits ist es schwierig, dies mit dem Hypermedia-Konzept einer **REST**-Architektur zu vereinbaren. Eine REST-Architektur verlangt, dass einer Anwendung, die eine Ressource repräsentiert, alle möglichen Zustandsübergänge als Hyperlinks präsentiert werden und dass jeder Zustandsübergang durch das Folgen eines Hyperlinks ausgelöst wird. Es ist möglich, den folgenden Verweis in HAL zu erstellen:

```

1. "updateToolAssemblyOrderState": {
2.   "method": "POST",
3.   "embeddable": false,
4.   "href": ".../tools/assemblyOrders/E446BB6B0C084CB09D0BC0319A8A1F1C/AVAILABLE"
5. }

```

Es wird ggf. auch die POST- anstelle der GET-Methode verwendet. Der Grund dafür ist, dass die GET-Methode nicht geeignet ist, eine große Anzahl von Filterparametern beim Aufrufen zu setzen. Dieses Problem tritt auf, wenn die Sammlungen von Fertigungsaufträgen oder Vorgängen explizit nach hunderten von Auftragsnummern gefiltert werden sollen. Für den Abruf beider Collections ist daher auch die POST-Methode vorgesehen. Die folgende Notation wird verwendet, um zu zeigen, dass eine solche POST-Methode für den Abruf einer Sammlung und nicht für die Erstellung einer neuen Ressource vorgesehen ist:

POST{collection}/search

2.2.1 OAuth-Token anfordern

Um ein Zugriffstoken zu erhalten, das den Zugriff auf die FORCE Bridge API ermöglicht, müssen Anwendungen den **Client Credentials Flow der OAuth2-Spezifikation** verwenden.

Um einen Autorisierungs-Header zu erstellen:

1. URL-kodieren Sie die Client-ID und das Client-Secret Ihrer Anwendung gemäß **RFC 1738**.
2. Verketteten Sie die kodierte Client-ID, ein Doppelpunkt-Zeichen (:) und das kodierte Client-Secret in eine einzige Zeichenfolge.
3. Fügen Sie "Basic" an die Zeichenfolge an.
4. Kodieren Sie die Zeichenfolge aus dem vorherigen Schritt in Base64.

Beispiel

Header: { Authorisation : Basic xyz Content-Type: application/x-www-form-urlencoded }
 Body: 'grant_type=client_credentials&scope =read write'

Beispiel-Code

```

{
    // How to create Authorization String
    "var encodedClientID = encodeURI(clientID); //encode with URLEncode"
    "var encodedClientSecret = encodeURI(clientSecret); //encode with URLEncode"
    "var pair = encodedClientID+':'+encodedClientSecret; //combine Strings with Semicolon"
    "var encodedPair = window.btoa(pair); //encode with Base64"
    "var authorizationValue = 'Basic' +encodedPair; //add Basic to the String"
}

```

2.2.2 POST-Anfragen versenden

Die Token-URL ist:

POST http(s)://\$HOST:\$PORT/ffwebservices/oauth/token

Name	Wert
Authorization	"Basic" + Base64 verschlüsselte clientid:clientsecret vom Vorschritt
Content type	application/x-www-urlencoded


Tabelle 3: POST-Request: Header

Name	Typ	Wert
grant_type	<i>string</i>	Erforderlich. Wert sollte <i>client_credentials</i> sein
scope	<i>string</i>	Leerzeichenbegrenzte Zeichenfolge der gewünschten Bereiche

Tabelle 4: POST-Request: Parameter

Name	Beschreibung
read	Erteilt die Berechtigung zum Aufruf von HTTP GET
write	Erteilt die Berechtigung zum Aufruf von HTTP POST, PUT, DELETE

Tabelle 5: POST-Request: Scopes

 Ablauf des Tokens: Anwendungen sollten unter Berücksichtigung der Möglichkeit geschrieben werden, dass ein gewährter Token möglicherweise nicht mehr funktioniert. Die Anwendung muss sich erneut authentifizieren, um neue Zugriffstoken zu erhalten. Die Zeit, bis ein Token abläuft, wird mit der Token-Antwort zurückgegeben.

CURL

```
curl -X POST http(s)://$HOST:$PORT/ffwebservices/oauth/token \  
--header "accept:application/x-www-urlencode" \  
-d "client_id=$CLIENT_ID" \  
-d "client_secret=$CLIENT_SECRET" \  
-d "grant_type=client_credentials" \  
-d "scope=read%20write"
```

Beispiel-Anfrage

```
{  
  "//access_token must be requested first"  
  "method": "POST",  
  "headers": "{",  
  "content-type": "application/json",  
  "Authorization": "'Bearer ' +access_token"  
}
```

Beispiel-Antwort

```
{  
  "access_token": "537517ab-faa3-4ad2-8ae5-37ff91ffb7c0",  
  "token_type": "bearer",  
  "expires_in": 42523,  
  "scope": "read write"  
}
```

3 Konfiguration Webservices

Sollen Daten über die FORCE Bridge API übertragen werden, muss die Verbindung zum Ziel- bzw. Quellsystem konfiguriert werden. Dies geschieht in der Workbench unter der Kachel Konfigurationen.

3.1 URLs

Pfad: Konfigurationen > Module > Webservices > URLs

Mit der **Webservices**-Konfiguration werden Webservices von externen Systemen und deren Verwendung konfiguriert. Dazu müssen die Verbindungsdaten, des jeweiligen Systems, im Bereich URLs konfiguriert werden.

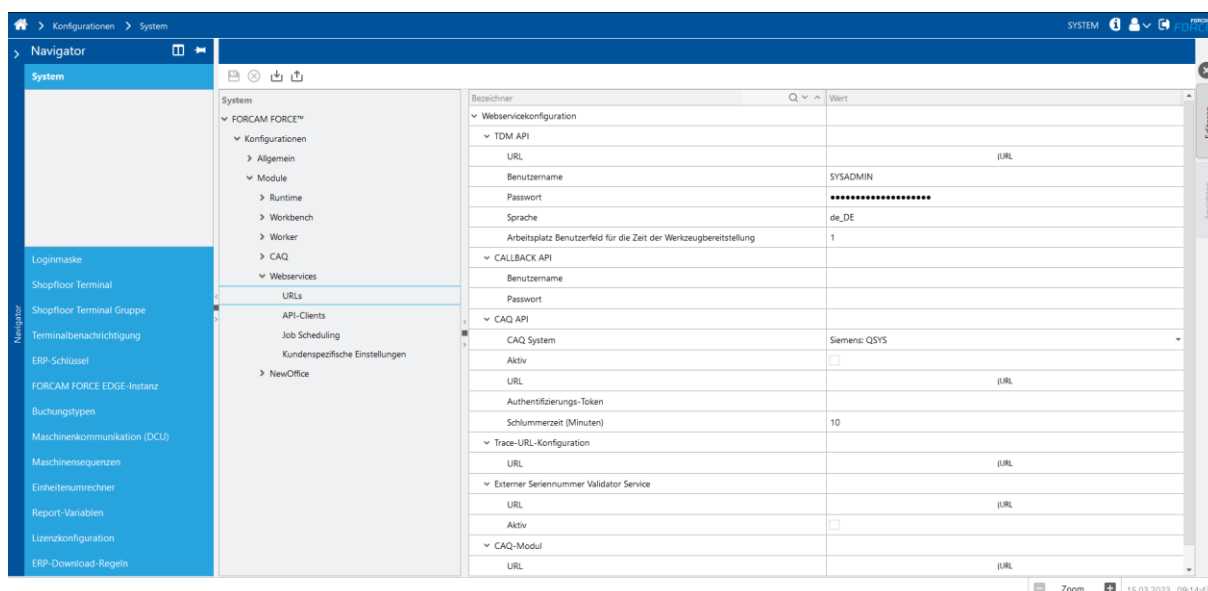


Bild 1: Konfiguration: Webservices > URLs

Bezeichnung	Erklärung
TDM API	
URL	URL des TDM-Systems
Benutzername	Username für Basic Authentifizierung
Passwort	Passwort für Basic Authentifizierung
Sprache	Verwendete Sprache für Nachrichten an TDM
Arbeitsplatz Benutzerfeld für die Zeit der Werkzeugbereitstellung	Nummer des Felds, in dem die Zeit für die Werkzeugbereitstellung hinterlegt ist
CALLBACK API	

Bezeichnung	Erklärung
Benutzername	Globaler Username für Basic Authentifizierung bei Callbacks
Passwort	Globales Passwort für Basic Authentifizierung bei Callbacks
CAQ API	
CAQ System	Auswahl des externen CAQ-Systems (aktuell QSYS und CAQ.net)
Aktiv	Anbindung aktiv
URL	URL des CAQ-Systems
Authentifizierungs-Token	Token für die Authentifizierung beim CAQ-System
Schlummerzeit (Minuten)	Zeitintervall für die erneute Benachrichtigung des Werkers bei negativer Prüfungs-Quittierung
TRACE-URL-Konfiguration	
URL	URL des Trace-Systems
Externer Seriennummer Validator Service	
URL	URL des Dienstes zur Validierung der Track & Trace Seriennummern
Aktiv	Soll Dienst genutzt werden
CAQ-Modul	
URL	URL des Dienstes zur Integration von externen CAQ-Systemen über Node-RED
Aktiv	Soll Dienst genutzt werden

Tabelle 6: Konfiguration: Webservices > URLs

3.2 API-Clients

Pfad: Konfigurationen > Module > Webservices > API-Clients > OpenAPI

Hier erfolgt die Konfiguration von API-Clients und entsprechenden Rechten für die Verwendung der FORCE Bridge API.

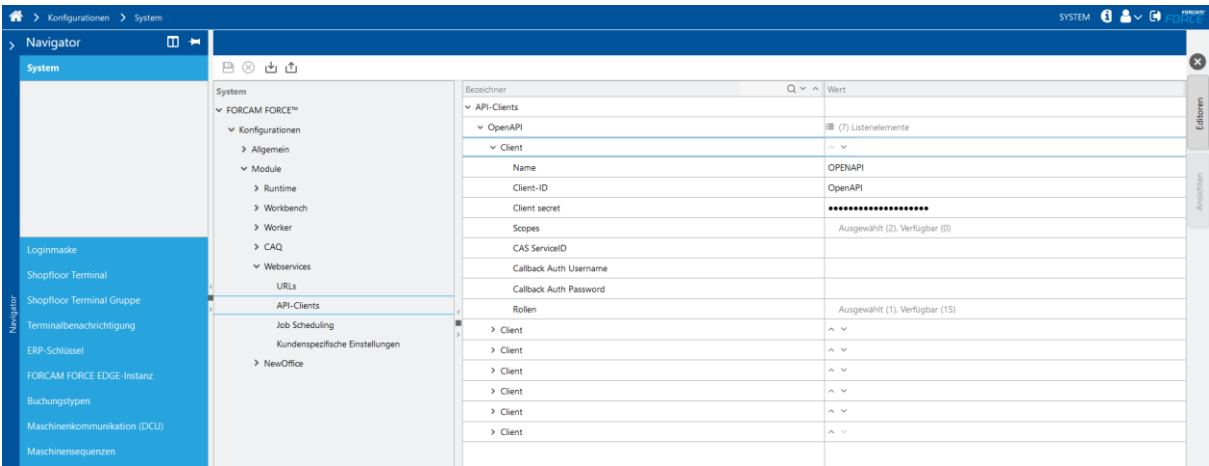


Bild 2: Konfiguration: API-Clients

Bezeichnung	Erklärung
Client	
Name	Sprechender Name des API-Clients
Client-ID	ID des API-Clients
Client secret	Secret des API-Clients
Scopes	Scopes für die der API-Client berechtigt ist (read / write)
CAS Service ID	CAS Service ID bei Verwendung von OAuth 2 Authorization grant
Callback Auth Username	Client spezifischer Username für Basic Authentifizierung bei Callbacks
Callback Auth Password	Client spezifisches Passwort für Basic Authentifizierung bei Callbacks
Rollen	Rollen des API-Clients

Tabelle 7: Konfiguration: API-Clients

Rolle	Wirkung
ADMIN	Clientzugriff auf alles, außer Mitarbeiterdaten
CAQ	Clientzugriff auf die CAQ-Schnittstelle
COMMAND_CALLBACK	Clientzugriff auf die Command- und Callback-Schnittstelle
CUSTOMIZED	Clientzugriff auf die Customized API. Diese wird für die Shopfloor Connectivity benötigt (weitere Informationen im Handbuch - Shopfloor Connectivity)
ERP_DOWNLOAD	Clientzugriff auf die ERP-Download-Schnittstelle (Datenfluss vom ERP-System in die MES FELX)
ERP_UPLOAD	Clientzugriff auf die ERP-Upload-Schnittstelle (Datenfluss von der MES FELX in das ERP-System)
STAFF_MEMBER_MINIMAL	Clientzugriff auf Minimalinformationen von Personaldaten
STAFF_RESPONSIBILITY	Vollständiger Clientzugriff auf Personaldaten
TDM	Clientzugriff auf TDM-Daten
Einträge mit dem Zusatz ‚_DOCS‘	Zugriff auf die Schnittstellenbeschreibung der entsprechenden Schnittstelle in der Swagger-UI

Tabelle 8: API-Client: Beschreibung der Rollen

3.3 Job Scheduling

Pfad: Konfigurationen > Module > Webservices > Job Scheduling

Hier erfolgt die Konfiguration von Jobs, die im Hintergrund laufen und mit externen Systemen kommunizieren.

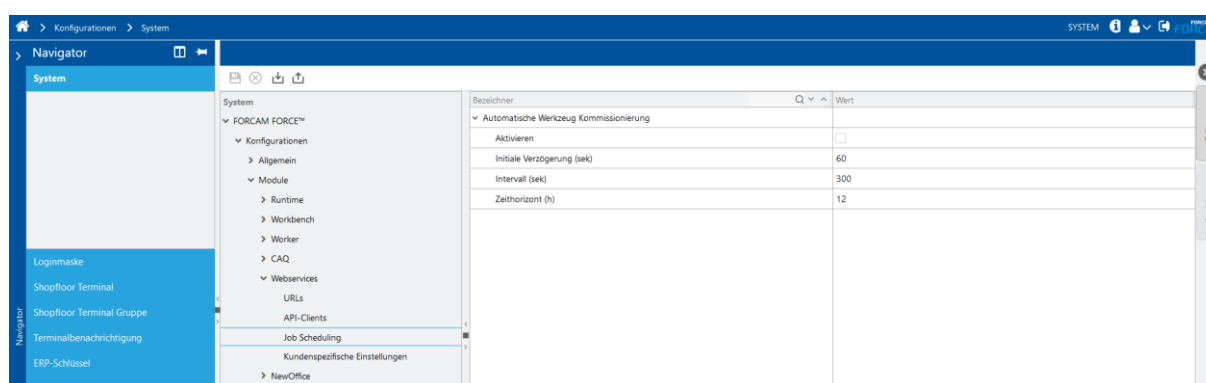


Bild 3: Konfiguration: Job Scheduling

Bezeichnung	Erklärung
Aktivieren	Aktivieren der automatischen Werkzeug-Kommissionierung mit TDM
Initiale Verzögerung (sek)	Initiale Verzögerung der automatischen Werkzeug-Kommissionierung nach Start von ffwebservices
Intervall (sek)	Intervall, in dem die automatische Werkzeug-Kommissionierung ausgeführt wird
Zeithorizont (h)	Zeithorizont für Vorgänge für die automatische Werkzeug-Kommissionierung

Tabelle 9: Konfiguration: Job Scheduling

3.4 Kundenspezifische Einstellungen

Pfad: Konfigurationen > Module > Webservices > Kundenspezifische Einstellungen

Hier können Kundenspezifische Einstellungen konfiguriert werden, welche über die entsprechende API (*GET/masterData/customerSpecificSettings*) bereitgestellt werden.

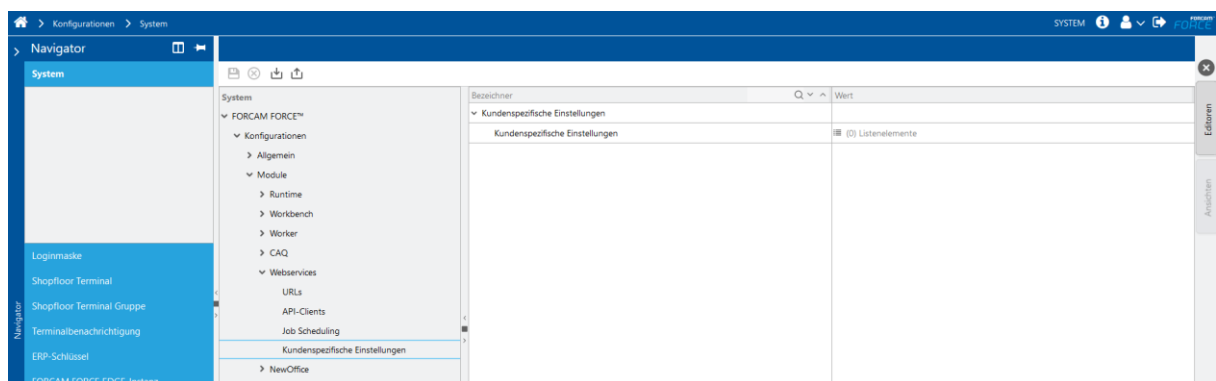


Bild 4:Konfiguration: Kundenspezifische Einstellungen

Bezeichnung	Erklärung
UUID	Wert für „id“ der kundenspezifischen Einstellung
Wert	Wert für „value“ der kundenspezifischen Einstellung
Thema	Wert für „topic“ der kundenspezifischen Einstellung
Beschreibung	Wert für „description“ der kundenspezifischen Einstellung

Tabelle 10: Konfiguration: Kundenspezifische Einstellungen

4 FORCE Bridge API als digitales Abbild einer Produktionsstätte

In diesem Kapitel werden die Konzepte und Objekte einer Produktionsstätte beschrieben. Es wird erklärt, wie sie in der FORCE Bridge API dargestellt werden und wie auf sie zugegriffen und mit ihnen gearbeitet wird.

4.1 Grundkonzepte und Objekte in der Produktion

Ziel des IoT ist es, ein digitales Abbild der realen Welt und ihrer Objekte zu schaffen. Diese Objekte fungieren einerseits als konkrete materielle Gegenstände und andererseits als abstrakte Konzepte. Ein IoT für den Bereich der Produktion erzeugt ein digitales Abbild sowohl der Maschinen und Werkzeuge, die in der Produktionsstätte eingesetzt werden, als auch der Konzepte, die den Produktionsalltag dominieren, wie z. B. Produktionsaufträge und Arbeitsvorgänge.

4.1.1 Einheiten aus der Produktion

Fertigungsaufträge

Wenn ein bestimmtes Produkt an einem bestimmten Punkt mit einer bestimmten Menge produziert werden soll, gibt das Produktionsplanungssystem die Erstellung eines Fertigungsauftrags aus. Die Vorlage für den Fertigungsauftrag ist ein Arbeitsplan, der jeden Schritt (Vorgang) enthält, der zur Herstellung des Endprodukts erforderlich ist. Der Arbeitsplan legt fest, an welcher Art von Arbeitsplatz oder Maschinen der Vorgang ausgeführt werden soll, sowie wie viel Zeit und welche Komponenten benötigt werden.

Die Erstellung eines Fertigungsauftrags läuft folgendermaßen ab:

- Das Unternehmen erhält einen oder mehrere Kundenaufträge zur Herstellung eines bestimmten Produkts oder möchte eine bestimmte Menge davon auf Lager produzieren.
- Im ERP-System wird ein neuer Fertigungsauftrag mit einer eigenen Nummer angelegt, der alle Informationen aus dem Arbeitsplan enthält.
- Der Fertigungsauftrag enthält zusätzlich zu den Angaben im Arbeitsplan die Angabe der Menge des zu produzierenden Produkts, diese wird als Sollmenge bezeichnet.
- Schließlich wird aus dem Liefertermin ein Eckendtermin für den Fertigungsauftrag erzeugt und das Ergebnis der Materialbedarfsplanung ist ein Eckstarttermin. Der Starttermin entspricht dem Zeitpunkt, zu dem alle für die Bearbeitung des Fertigungsauftrags benötigten Komponenten verfügbar sind.

Arbeitsvorgänge

Ein Arbeitsvorgang (AVO) beschreibt einen einzelnen Schritt, der zur Bearbeitung eines Fertigungsauftrags erforderlich ist. Jeder Arbeitsvorgang produziert ein bestimmtes Material in einer vorgegebenen Menge (Zielmenge des Arbeitsvorgangs). Dabei handelt es sich entweder um ein Material, das in einem nachfolgenden Arbeitsvorgang als Komponente weiterverarbeitet wird, oder im Falle eines letzten Arbeitsvorgangs, um das Endprodukt des Fertigungsauftrags.

Zu den im Arbeitsvorgang gespeicherten Standardwerten gehören:

- Das zu fertigende Material.

- Die Sollmenge ist die Menge des zu produzierenden Materials. Diese kann sich von der Sollmenge des Fertigungsauftrags unterscheiden (z. B. 4 Türen, die in einem Arbeitsvorgang gefertigt werden müssen, um letztendlich ein Automobil zu produzieren).
- Die Mengeneinheit der Sollmenge (z. B. Stück, Meter, Kilogramm, etc.).
- Der Arbeitsplatz oder eine Gruppe von gleichwertigen Arbeitsplätzen, an denen der AVO bearbeitet werden soll.
- Die Sollzeit pro Einheit, d. h. die Zeit, die für die Herstellung einer Einheit des zu produzierenden Materials benötigt wird.
- Die Sollbearbeitungszeit ist die Zeit, die für die gesamte Bearbeitungszeit des AVOs verwendet wird. Sie ist das Produkt aus Sollmenge und Sollzeit pro Einheit.
- Die Soll-Rüstzeit ist die Zeit, die für das Rüsten des Arbeitsplatzes eingeplant ist.
- Die Soll-Abrüstzeit ist die Zeit, die für den Abbau des Arbeitsplatzes eingeplant ist.
- Die Soll-Transportzeit ist die Zeit, die für den Transport zum nächsten Arbeitsplatz eingeplant ist.
- Die Soll-Liegezeit ist die Zeit, die für Funktionen wie Kühlen, Trocknen oder Aushärten des zu produzierenden Materials eingeplant ist.
- Die Komponenten, die benötigt werden, um das Material zu produzieren.
- Die Fertigungsressourcen und Hilfsmittel (z. B. Werkzeuge, Geräte oder Dokumente), die am Arbeitsplatz benötigt werden, um den Arbeitsvorgang auszuführen.

Arbeitsvorgangsphasen

Die Arbeitsvorgangsphasen beschreiben den Zustand eines Arbeitsvorgangs; von der Freigabe durch das ERP-System bis zur Fertigstellung und Rückbuchung der Ausgangsmengen des Arbeitsvorgangs an das ERP-System.

Arbeitsplätze

Als Minimum wird jeder Arbeitsvorgang an einem Arbeitsplatz ausgeführt. Ein Arbeitsplatz kann ein einfacher Handarbeitsplatz oder ein Arbeitsplatz mit einer automatisierten Fertigungslinie (Maschine) sein.

Arbeitsplatz-Gruppen

Arbeitsplätze werden in ERP-Systemen oft als ein einzelner Arbeitsplatz mit mehreren Kapazitäten hinterlegt. In FORCE Bridge API wird jeder Arbeitsplatz einzeln angezeigt und dann einer bestimmten Kapazitätsgruppe bzw. Produktionslinie zugeordnet.

- Kapazitätsgruppen sind Arbeitsplätze, an denen das gleiche Produktionsverfahren ausgeführt wird, und daher zusammen gruppiert werden. Zum Beispiel gehört ein Spritzgießarbeitsplatz zur Kapazitätsgruppe Spritzgießmaschinen.
- Produktionslinien sind verschiedene Arbeitsplätze, die kollektiv einen kompletten Produktionsprozess eines Produkts abbilden. Hierbei werden Prozesse der Linienfertigung oder Massenfertigung verwendet. Dabei sind die verschiedenen Stationen der Produktionslinie nicht durch zeitliche Taktung des Produktionszyklus verbunden, sondern Pufferspeicher fangen die Zeitunterschiede bei der Bearbeitung zwischen benachbarten Stationen auf. So kann z. B. ein fehlerhaftes Produkt zur Nacharbeit aus dem Produktionsablauf herausgenommen werden, ohne den Produktionsprozess zu stören.

Jeder Arbeitsplatz kann genau einer Kapazitätsgruppe und maximal einer Fertigungslinie zugeordnet werden.

Mitarbeiter

Die Mitarbeiter sind das Produktionspersonal, das in Schichten eingeteilt ist und den jeweiligen Arbeitsplätzen zugeordnet werden kann. Die Mitarbeiter tragen neben ihren Personaldaten zusätzliche Informationen wie z. B. Qualifikationen, gebuchter Urlaub etc., die für die Planung und den Ablauf der Produktion wichtig sind.

Werkzeuge

Produktionswerkzeuge werden an Maschinenarbeitsplätzen benötigt, um einen Arbeitsvorgang fachgerecht durchzuführen. Ein Werkzeug ist z. B. die Spritzgussform einer Spritzgießanlage. Das Werkzeug hat einen Lagerort, einen aktuellen Einsatzort und eine Lebensdauer. Die Lebensdauer beschreibt, wie viel Zeit verbleibt, bis das Werkzeug ausgetauscht werden muss.

Dokumente

Für die Ausführung eines Arbeitsvorgangs werden Dokumente benötigt, die in Produktionsordnern abgelegt werden. Diese Dokumente stammen in der Regel aus dem Product-Lifecycle-System. Bei den Dokumenten kann es sich um NC-Programme zur Steuerung von Werkzeugmaschinen handeln, aber auch um andere Dokumente wie Prüfpläne, Montageanleitungen oder Montagezeichnungen. Werden Dokumente geändert, legt die IoT-Plattform automatisch eine neue Version an. Alle Dokumente, die zu einem Arbeitsvorgang gehören, werden im Shopfloor Terminal angezeigt. Von dort können auch die NC-Programme aus dem Fertigungsordner in die Maschinensteuerung geladen werden.

Geräte

Geräte sind alle mit der IoT-Plattform verbundene Geräte. Dazu gehören insbesondere die speicherprogrammierbaren Steuerungen (SPS) der angeschlossenen Maschinen und Produktionsanlagen. Diese sind fest einem bestimmten Arbeitsplatz zugeordnet.

Mit `GET/workplaces/{workplaceId}/devices` können alle Geräte abgerufen werden, die an einem Arbeitsplatz oder einer Maschine vorhanden sind. Zu den Geräten können alle aufgezeichneten Prozess- und Sensordaten abgefragt werden.

4.1.2 Ressourcen der FORCE Bridge API

FORCE Bridge API bildet das komplette digitale Abbild einer Produktionsanlage ab. Der RESTful ²-Webservice kann direkt über einen Webbrowser aufgerufen werden, um mit den in der API bereitgestellten Ressourcen zu kommunizieren.

Operation	>
Production Order	>
Workplace	>
Workplace Group	>
Staff Member	>
Tool	>
Device	>
Document	∨
<div> <div>GET</div> <div>/api/v1/documents/folder/ Get all folders</div> <div>🔒</div> </div>	
<div> <div>POST</div> <div>/api/v1/documents/folder/ Create a folder</div> <div>🔒</div> </div>	
<div> <div>GET</div> <div>/api/v1/documents/folder/{folderId} Get a folder</div> <div>🔒</div> </div>	

Bild 5: Zugriff auf API-Ressourcen über die Swagger UI

Eine technische Beschreibung der angebotenen Schnittstellen der FORCE Bridge API erhält man im SwaggerHub.

Einstiegspunkt und Top-Level-Ressourcen

RESTful hat generell nur einen *Einstiegspunkt* (API-Basis), von woher die anderen, ausgehenden Ressourcen abgerufen werden können. Um die API-Ressourcen finden zu können, ist es daher wichtig, dass der Uniform Resource Indicator (URI) mit den REST-Clients korrekt kommuniziert wird.

In diesem Abschnitt werden die folgenden Top-Level-Ressourcen näher beschrieben:

- *productionOrders*/ Produktionsaufträge
- *operations*/ Vorgänge der Fertigungsaufträge
- *workplaces*/ Arbeitsplätze (Maschinen oder Handarbeitsplätze), an denen Vorgänge ausgeführt werden
- *workplaceGroups*/ Kapazitätsgruppen (funktional ähnliche Maschinen) oder Produktionslinien
- *staffMembers*/ Die Mitarbeiter, die für die Ausführung von Operationen bereitstehen

² RESTful sind Anwendungen oder Webservices, die Zustände und Funktionen als eine Sammlung von Ressourcen anbieten und so die Interoperabilität zwischen Computersystemen im Internet ermöglichen.

- *tools*/ Die verfügbaren Werkzeuge für die Ausführung von Operationen
- *documents*/ Für die gesamte Produktionsanlage wichtige Dokumente
- *devices*/ Geräte wie z. B. speicherprogrammierbare Steuerungen von Maschinen

Anlegen / Ändern / Löschen von Daten über die FORCE Bridge API

Folgende Ressourcen können über die FORCE Bridge API angelegt werden:

- *productionOrders*/ Produktionsaufträge
- *operations*/ Vorgänge der Fertigungsaufträge
- *staffMembers*/ Die Mitarbeiter, die für die Ausführung von Operationen bereitstehen
- *workplaces/shifts*/ Arbeitsplatz-Schichten
- *materials*/ Material der Aufträge und Arbeitsvorgänge

Dabei sind die Vorgänge Sub-Ressourcen der Fertigungsaufträge.

Aufträge und Arbeitsvorgänge können aber auch separat in Einzeltransaktionen angelegt, geändert oder gelöscht werden. Diese Transaktionen werden synchron ausgeführt, sodass anhand der Antwort erkennbar ist, ob die Transaktion erfolgreich war.

Referenzen zu anderen Ressourcen, wie ERP-Kontext und Material müssen in der Request-Payload mittels UUID angegeben werden. Diese Ressourcen müssen bereits existieren bzw. vorab angelegt werden.

4.2 Schichten und geplante Betriebszeiten von Arbeitsplätzen

4.2.1 Domänenspezifisches Wissen

Arbeitsplatzschichten

Der Begriff Schichtarbeit beschreibt eine Arbeitsstruktur, bei der Mitarbeiter nacheinander und nach einem bestimmten Zeitplan an der gleichen Tätigkeit arbeiten, so dass sie innerhalb eines bestimmten Zeitrahmens zu unterschiedlichen Zeiten ihre Arbeit verrichten. Ein Betrieb setzt Schichtarbeit ein, wenn es aus verschiedenen Gründen notwendig ist, außerhalb der üblichen Tagesarbeitszeit zu arbeiten oder wenn Betriebs- oder Bereitschaftsdienste außerhalb der üblichen Tagesarbeitszeit erforderlich sind.

Für Arbeitsplätze werden Schichten definiert, z. B. im Dreischichtbetrieb sind dies meist Frühschicht, Spätschicht und Nachtschicht. Dieses Konzept beschreibt ein Schichtmodell.

Wenn eine der Schichten an einem bestimmten Tag ausfallen soll, wird die Schicht als Leerlaufzeit oder arbeitsfreie Schicht gekennzeichnet. Die anderen Schichten sind Arbeitsschichten. Innerhalb einer Arbeitsschicht können sogenannte Schichtpausen definiert werden, während derer die Produktion angehalten wird.

Geplante Wartungszeiten

Dies sind Zeiten, in denen Maschinenarbeitsplätze gewartet werden. Handarbeitsplätze sind davon nicht betroffen. Das Ziel geplanter Wartungszeiten ist es, Störungen zu reduzieren, damit Maschinen nicht unerwartet ausfallen, was wiederum eine außerplanmäßige Wartung auslösen würde. Diese Zeiten finden in regelmäßigen Wartungsintervallen statt.

Geplante Betriebszeiten

Die planmäßige Betriebszeit ist der Zeitraum, in dem ein Arbeitsplatz laut Schichtplanung und Wartungsplanung in Produktion sein soll. Dazu gehören alle Arbeitsschichten abzüglich der Schichtpausen und der geplanten Wartung.

4.2.2 Arbeitsplatz-Ressourcen in FORCE Bridge API

Die oben genannten Objekte befinden sich unter den Teilressourcen von Arbeitsplätzen bzw. einem bestimmten Arbeitsplatz.

Workplace			^
GET	/workplaces	Retrieve the collection of workplaces.	✓ 🔒
GET	/workplaces/shifts	Retrieve the collection of workplace shifts.	✓ 🔒
GET	/workplaces/shifts/{shiftId}	Retrieve a workplace shift.	✓ 🔒
GET	/workplaces/{workplaceId}	Retrieve a workplace.	✓ 🔒
GET	/workplaces/{workplaceId}/customFields	Request workplace custom fields	✓ 🔒
GET	/workplaces/{workplaceId}/customFields/{customFieldId}	Request workplace specific custom field	✓ 🔒
GET	/workplaces/{workplaceId}/devices	Retrieve the devices that belong to a workplace.	✓ 🔒
GET	/workplaces/{workplaceId}/operatingStateSummary	Retrieve the operating state summary of a workplace.	✓ 🔒
GET	/workplaces/{workplaceId}/quantitySummary	Retrieve the quantity summary of a workplace.	✓ 🔒
GET	/workplaces/{workplaceId}/recordedOperatingStates	Retrieve the recorded operating states of a workplace.	✓ 🔒
GET	/workplaces/{workplaceId}/recordedOperationPhases	Retrieve the recorded operation phases of a workplace and their durations.	✓ 🔒
GET	/workplaces/{workplaceId}/recordedOutputQuantities	Retrieve the recorded output quantities of a workplace.	✓ 🔒
GET	/workplaces/{workplaceId}/scheduledMaintenanceTimes	Retrieve the scheduled maintenance times of a workplace.	✓ 🔒
GET	/workplaces/{workplaceId}/scheduledOperatingTimes	Retrieve the originally scheduled operating times of a workplace.	✓ 🔒

Bild 6: Arbeitsplatz-Ressourcen, wie in der Swagger UI dargestellt

GET workplaces/shifts

Schichten eines Arbeitsplatzes anfragen.

GET workplaces/shifts/{shiftId}

Details für eine bestimmte Schicht anfragen.

GET workplace/{workplaceId}/scheduledMaintenanceTimes

Anfrage der geplanten Wartungszeiten eines Arbeitsplatzes.

GET workplaces/{workplaceId}/scheduledOperatingTimes

Arbeitsplatz geplante Betriebszeiten anfragen. Das digitale Abbild in FORCE Bridge API.


Betriebszustände und Arbeitsplatzstatus

Betriebszustände werden definiert, um festzustellen, wie der aktuelle Zustand eines Arbeitsplatzes ist. Mögliche Betriebszustände eines Arbeitsplatzes sind z. B. Produktion, Maschinenstörung, Werkzeugschaden, etc.

Zu jedem Betriebszustand gehört ein Arbeitsplatzstatus, um festzustellen, ob ein Arbeitsplatz gerade arbeitet oder gestoppt ist. Sie sind wie folgt definiert.

Arbeitsplatzstatus	Beschreibung
DOWNTIME	Der Arbeitsplatz steht still.
PRODUCTION	Der Arbeitsplatz produziert.

Tabelle 11: Arbeitsplatzstatus

-  Die Betriebszustände können in der MES FLEX frei konfiguriert werden. Dadurch kann jeder Arbeitsplatz individuell angepasste Betriebszustände haben.

Arbeitsvorgangsphasen (AVO-Phasen)

Zwischen der Freigabe eines AVOs und seiner Fertigstellung mit abschließender Rückmeldung an das ERP-System treten in der Produktion die nachfolgenden Phasen auf. Diese werden entweder am Shopfloor Terminal gemeldet oder von der MES FLEX automatisch aktualisiert.

Phase	Beschreibung
RELEASED	Der Produktionsauftrag eines AVOs wurde vom ERP-System freigegeben, aber der AVO wurde noch nicht gestartet oder im Detail geplant.
DISPATCHED	Der AVO wurde zu einem bestimmten Arbeitsplatz delegiert.
SETUP	Der AVO wurde an einem Arbeitsplatz angemeldet und dieser Arbeitsplatz ist für die Bearbeitung des AVOs gerüstet
PROCESSING	Der AVO wird bearbeitet.
INTERRUPTED	Der AVO wurde unterbrochen.
COMPLETED	Der AVO wurde vollständig abgeschlossen.
CLOSED	Der AVO wurde abgeschlossen, und eine abschließende Bestätigung wurde an das ERP-System gesendet.

Tabelle 12: AVO-Phasen

Werksdaten erfassen

Erfasste Zeitreihen

GET operations/{operationId}/recordedOperatingStates

Enthält alle Betriebszustände, die während des Arbeitsvorgangs aufgetreten sind

GET operations/{operationId}/recordedOperationPhases

Enthält alle AVO-Phasen, die beim Arbeitsvorgang aufgetreten sind

GET operations/{operationId}/recordedOutputQuantities

Enthält alle Outputmengen (Gutmenge, Nacharbeit, Ausschuss), die bei dem AVO produziert wurden

Verdichtete Werte

GET operations/{operationId}/recordedOperatingStates

GET operations/{operationId}/recordedOperationPhases

Maschinendatenerfassung

Istwerte

GET workplaces/{workplaceId}/recordedOperatingStates

Enthält alle Betriebszustände, die am Arbeitsplatz aufgetreten sind.

GET workplaces/{workplaceId}/recordedOperationPhases

Enthält alle AVO-Phasen, die am Arbeitsplatz aufgetreten sind.


GET workplaces/{workplaceId}/recordedOutputQuantities

Enthält alle Outputmengen (Gutmenge, Nacharbeit, Ausschuss), die am Arbeitsplatz produziert wurden.

Verdichtete Werte

GET workplaces/{workplaceId}/recordedOperatingStates

GET workplaces/{workplaceId}/recordedOperationPhases

 Jede Ressource enthält die Istwerte – oder für die Zusammenfassungen – die verdichteten Istwerte, die zur Berechnung der KPIs verwendet werden.

5 Callbacks und Ereignisse

Mit der FORCE Bridge API können Applikationen und Drittsysteme über Ereignisse via HTTP oder MQTT benachrichtigt werden. Dafür muss man sich registrieren. Dazu wird ein Callback (auch Webhook genannt) per POST erstellt. Dieser gibt die Art des Ereignisses an, über das die Applikation oder das Drittsystem benachrichtigt werden soll, und die URL, unter der es erreicht werden kann.

Für die Benachrichtigung über MQTT muss ein Topic definiert werden, das immer mit *external* beginnen muss, um von den intern in der MES FLEX verwendeten Topics zu unterscheiden:

```
"url": "mqtt://{host}:1883/external/{any topic}"
```

Drittsysteme, die nicht über MQTT, sondern über HTTP über Ereignisse informiert werden sollen, benötigen einen eigenen HTTP-Server. In diesem Fall ist der Callback selbst ein HTTP-POST-Request mit JSON-Daten, der wie folgt an die URL des zu spezifizierenden HTTP-Servers gesendet wird:

```
"url": "http://{host}:{port}/{path}"
```

5.1 Callbacks

5.1.1 Callback-Anfrage an den externen Webservice

Immer, wenn ein Ereignis eintritt, das mit einer bestehenden Registrierung übereinstimmt, werden die Ereignisdaten über eine HTTP-POST-Anfrage an die angegebene URL gesendet.

5.1.2 Wiederholungen

Wenn ein Callback fehlschlägt (z. B. wenn das System, das den registrierten Endpunkt bereitstellt, ausfällt), werden alle 10 Sekunden Wiederholungsversuche ausgelöst, bis die maximale Anzahl von Wiederholungsversuchen erreicht ist. Wenn die maximale Anzahl von Wiederholungsversuchen erreicht ist, wird das Callback verworfen und es werden keine weiteren Wiederholungsversuche für dieses Callback gesendet. Wiederholungen bedeuten, dass Callbacks möglicherweise nicht in der richtigen Reihenfolge gesendet werden. Daher muss der Zeitstempel des Ereignisses, das den Callback ausgelöst hat, vom empfangenden System berücksichtigt werden. Wenn die maximale Anzahl der fehlgeschlagenen Callbacks erreicht ist, wird die älteste verworfen, auch wenn die maximale Anzahl der Wiederholungsversuche noch nicht erreicht ist.

5.1.3 Ein Callback erstellen

Um über Ereignisse benachrichtigt zu werden, muss ein Callback erstellt werden. Der Callback wird für einen bestimmten EventType (wie z. B. ERP) registriert, um alle Nachrichten dieses Typs zu erhalten. Ggf. kann eine weitere Filterung über den EventName erfolgen, wenn Benachrichtigungen über ein bestimmtes Event eines Event Typs erfolgen soll.

Die Erstellung eines Callbacks geschieht durch das Aufrufen von POST-Callbacks mit dem folgenden Inhalt im Anfragetext:

```
6. {
7.   "name": "string",
8.   "objectFilter": [
9.     {
10.      "name": "string",
11.      "value": "string"
12.    }
13.  ],
14.  "eventType": "COMMAND",
15.  "eventName": "string",
16.  "url": "string",
17.  "maxRedeliveryAttempts": 100,
18.  "maxUnconfirmedMessages": 5000
19. }
```

Der Parameter *maxRedeliveryAttempts* definiert die Anzahl der erneuten Zustellversuche, wenn Callback-Benachrichtigungen fehlschlagen.

Mit der Einstellung des Parameters *maxUnconfirmedMessages* wird die maximale Anzahl fehlgeschlagener Callback-Benachrichtigungen eingestellt. Wenn die maximale Anzahl erreicht ist, wird die älteste Benachrichtigung verworfen, auch wenn die maximale Anzahl an Wiederholungen für die Benachrichtigung noch nicht erreicht ist.

Ein Callback kann mit *maxUnconfirmedMessages = 0* registriert werden, um sicherzustellen, dass keine Nachrichten verloren gehen. Intern werden Nachrichten an eine dynamische MQTT-Queue gesendet, die alle Nachrichten speichert, bevor sie an das endgültige Ziel gesendet werden.

⚠ Für Callbacks mit *maxUnconfirmedMessages = 0* muss sichergestellt werden, dass diese irgendwann auch tatsächlich zugestellt werden, sonst stapeln sich Nachrichten in der Persistent Queue und bringen den Message Broker irgendwann zum Erliegen.

⚠ Durch das Löschen eines Callbacks werden auch alle damit verbundenen Queues gelöscht.

Für die Einstellung *maxUnconfirmedMessages = 0*, gelten die folgenden Regeln für Wiederholungsversuche:

- Der Wiederholungsmechanismus wird standardmäßig 10 Sekunden nach einer fehlgeschlagenen Zustellung gestartet oder wenn die Zustellung länger als 10 Sekunden dauert
- Wiederholungen werden mit einem exponentiellen Backoff ausgelöst. (Beginnend mit 200 ms bis 10 Sekunden standardmäßig)
- Nachrichten werden immer noch aus der Queue entfernt, nachdem *maxRedeliveryAttempts* erreicht sind.
- Für das Abbrechen des Wiederholungsmechanismus gibt es folgende zwei Möglichkeiten:
 - Löschung des Callbacks
 - Neustart der ffwebservice-Anwendung

⚠ Wenn ein Callback mit *maxUnconfirmedMessages = 0* und *maxRedeliveryAttempts = 0* registriert wird, wird die Zustellung von Nachrichten für immer wiederholt. Wenn diese Konfiguration verwendet wird, wird die Nachrichtenreihenfolge auch bei Übermittlungsfehlern beibehalten.

In der Anfrage müssen mindestens zwei Angaben gemacht werden:

- Ereignistyp, über den informiert werden soll
- URL, an der die Information über ein Ereignis gesendet werden soll

Ereignisse, die sich auf einen bestimmten Typ von Primärressourcen beziehen, können bei der Erstellung des Callbacks auf eine Teilmenge oder eine einzelne Primärressource eingeschränkt

werden. Dazu werden Objektfilter verwendet, die als eine Feldanordnung von Namen/Werte-Paaren übergeben werden.

Name gibt eine Eigenschaft des Referenzobjekts an.

Wert ist ein Wert, den die Eigenschaft haben muss, um das Filterkriterium anwenden zu können. In der aktuellen Version wird nur die Eigenschaft *ID* unterstützt, d. h. die Referenzobjekte eines Ereignistyps können nur nach UUIDs gefiltert werden.

Beispiele

CURL Beispiel mit HTTP

```
1. curl -X POST http://$HOST:$PORT/ffwebservice/api/v2/callbacks \
2.   --header "Content-Type: application/json" \
3.   --header "Accept: application/json" \
4.   --header "Authorization: Bearer $TOKEN" \
5.   -d '{
6.     "objectFilter": [
7.       {
8.         "name": "id",
9.         "value": "DA892519507444768080F0E81879513"
10.      }
11.    ],
12.    "eventName": "Operation 100 has changed phase",
13.    "eventType": "OPERATION_PHASE_CHANGED",
14.    "url": "http://third-party-system:24080/webserver"
15.  }
```

CURL Beispiel mit MQTT

```
1. curl -X POST http://$HOST:$PORT/ffwebservice/api/v2/callbacks \
2.   --header "Content-Type: application/json" \
3.   --header "Accept: application/json" \
4.   --header "Authorization: Bearer $TOKEN" \
5.   -d '{
6.     "objectFilter": [
7.       {
8.         "name": "id",
9.         "value": "DA892519507444768080F0E81879513"
10.      }
11.    ],
12.    "eventName": "Operation 100 has changed phase",
13.    "eventType": "OPERATION_PHASE_CHANGED",
14.    "url": "mqtt://third-party-system:1883/external/OperationPhaseChanged"
15.  }
```

5.1.4 Alle registrierten Callbacks abrufen

Um alle registrierten Callbacks zu erhalten, muss eine Anfrage an *callbacks/* via *GET* gesendet werden.

Beispiel CURL

```
1. curl -X GET "http://$HOST:$PORT/ffwebservice/api/v1/callbacks/" \
2.   --header "Content-Type: application/json" \
3.   --header "Accept: application/json" \
```

5.1.5 Einen registrierten Callback löschen

Um einen registrierten Callback zu löschen, muss eine Anfrage über *callbacks/{callbackId}* via *DELETE* gesendet werden, wobei *callbackId* die *ID* des Callbacks ist.

Beispiel CURL

```
1. curl -X DELETE "http://$HOST:$PORT/ffwebservice/api/v1/callbacks/$CALLBACK_UUID" \  
2.   --header "Content-Type: application/json" \  
3.   --header "Accept: application/json" \  

```

5.1.6 Ereignistypen

Eine Liste aller verfügbaren Callbacks werden im Swagger unter </ffwebservices/docs/index.html#callback-api> und Befehle unter </ffwebservices/docs/index.html#command-api> gefunden.

Im Swagger sind alle Ereignistypen aufgelistet, für die eine Callback-Registrierung möglich ist. Sie werden als verdichtete Informationen von der MES FLEX als Folge bestimmter anderer Einzelereignisse bereitgestellt. Darüber hinaus werden zahlreiche Ereignisse generiert, die über Details informieren.

Die Ereignistypen *COMMAND*, *SHOP_FLOOR_TERMINAL* und *INTERNAL* sind für interne Ereignisse der MES FLEX reserviert.

5.1.7 Ereignisse empfangen

Tritt ein Ereignis ein (im folgenden Beispiel: *OPERATION_PHASE_CHANGED*), für das sich ein Drittsystem per HTTP-Callback registriert hat, erhält der HTTP-Server des Drittsystems beispielsweise die folgende POST-Anfrage:

```
<curl -X POST  
--header 'Content-Type: application/json'  
-d '{  
  "properties" : {  
    "callbackId" : "B1B1BFB60ECF4DB7B095F096992FB9FA",  
    "timestamp" : "2019-03-06T11:04:16.786Z",  
    "data" : {  
      "currentOperationPhase" : "PROCESSING",  
      "previousOperationPhase" : "INTERRUPTED",  
      "operationId" : "2A05AAEF15FD41A8A4A36C45516871AA"  
    },  
    "objectId" : "DA892519507444768080F0E81879513",  
    "objectType" : "OPERATION",  
    "eventType" : "OPERATION_PHASE_CHANGED",  
    "eventName" : "Operation 100 has changed phase"  
  },  
  "links" : {  
    "callback" : {  
      "method" : "GET",  
      "embeddable" : true,  
      "href" : "http://localhost:24080/ffwebservices/api/v2/callbacks/B1B1BFB60ECF4DB7B095F096992FB9FA"  
    }  
  }  
}' 'CALLBACK_URL'>
```

Es ist möglich, eine Basic-Auth-Authentifizierung für Callbacks zu konfigurieren. Wenn ein Ereignis eintritt, für das sich eine Applikation über MQTT registriert hat, erhält die Applikation eine MQTT-Nachricht.

5.1.8 Selbst definierte Ereignisse an Applikationen oder Drittsysteme senden

Eine Applikation kann Ereignisse auslösen, die von anderen Applikationen oder Drittsystemen per Callback empfangen werden können. Dazu müssen POST-Ereignisse mit dem folgenden Inhalt im Anfragetext aufgerufen werden:

```
{
  "data": {},
  "eventName": "string",
  "eventType": "EXTERNAL",
  "objectId": "string",
  "objectType": "OPERATION",
  "timestamp": "2019-02-23T17:42:59.429Z"}
```

Im Anfragetext muss der Ereignistyp immer EXTERNAL sein.

Der Objekttyp kann leer sein oder einen der folgenden Werte haben:

- *OPERATION*
- *PRODUCTION_ORDER*
- *WORKPLACE, STAFF_MEMBER*
- *TOOL*

Im Datenfeld können beliebig viele Informationen mit dem Ereignis übertragen werden. Der Ereignisname und die Objekt-ID sind optional.

Beispiel CURL

```
1. curl -X POST https://$HOST:$PORT/ffwebservice/api/v2/events"
2.
3.   -H "accept: */*"
4.   -H "authorization: Bearer $TOKEN
5.   -H "Content-Type: application/json"
6.   -d "{
7.       \"data\": {\"myData1\" : \"123\", \"myData2\" : \"456\"},
8.       \"eventName\": \"MY_EVENT\",
9.       \"eventType\": \"EXTERNAL\",
10.      \"objectId\": \"4DDE054C036C4F8CAB424C7FB70E86B9\",
11.      \"objectType\": \"TOOL\",
12.      \"timestamp\": \"2019-03-01T07:24:37.756Z\"
13.    }
```

6 ERP-Interface – Daten und Events via XML

Für die Art der Kommunikationsverbindung zwischen MES FLEX und einem ERP-System gibt es mehrere Optionen bzw. Alternativen.

Zum einen kann die Kommunikation direkt über die Runtime von MES FLEX mit Hilfe des offenen und standardisierte Verfahrens der XSL-Transformation (XSLT) erfolgen. Die XSLT ist Bestandteil der XSL (Extensible Stylesheet Language) und führt eine anwendungsspezifische Transformation der übermittelten Datenstrukturen durch. Dieses Verfahren kommt z. B. bei der Anbindung von ERP-Systemen zum Einsatz, die Datensätze im SAP IDoc-Format austauschen und wird im Handbuch zu **ERP Interface & Daten und Events via SAP IDoc** näher beschrieben.

Fortschrittlicher, sicherer und wesentlich verlässlicher ist die alternative Anbindung über die von MES FLEX bereitgestellte Programmierschnittstelle FORCE Bridge API, eine RESTful-Schnittstelle zu MES FLEX. Diese Variante ist in den vorhergehenden Kapiteln beschrieben worden und Grundlage für den Einsatz des ERP-Interfaces.

Bei beiden Verfahren werden Datensätze in Form von XML-Datensätzen über das HTTP- bzw. HTTPS-Protokoll ausgetauscht. Dabei kommt ein http-POST-Request mit den XML-Datensätzen als Payload zum Einsatz.

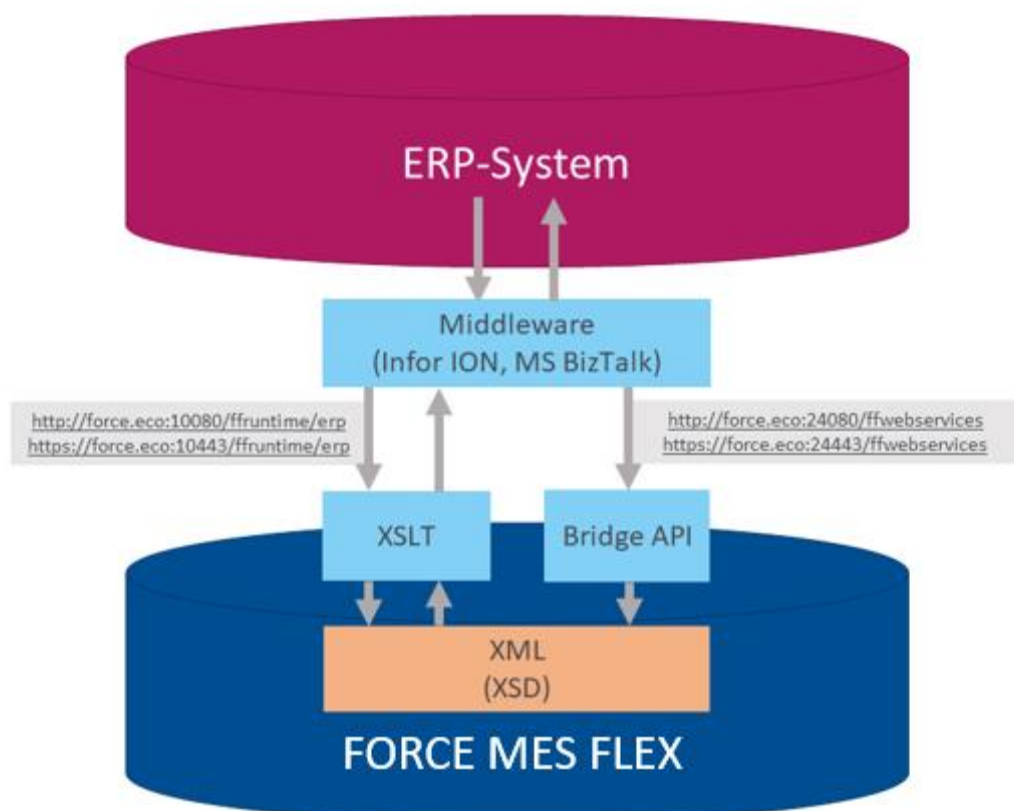


Bild 7: Alternative Verbindungen ERP-System und FORCE MES FLEX

Zusätzlich kann für die Anbindung an das ERP auch Middleware wie z. B. Infor ION oder Microsoft BIZ Talk eingesetzt werden. Die dort vorhandene Funktionalität kann genutzt werden, um die benötigten Datensätze in die MES FLEX interne XML-Struktur zu transformieren.

7 ERP-Download- und ERP-Upload-Datendienst

Für die Kommunikation zwischen MES FLEX und dem verbundenen ERP-System steht ein Download- und Upload-Datendienst zur Verfügung.

Der Download-Datendienst versorgt die MES FLEX Rule-Engine mit Stammdaten und Auftragsdaten. Die eintreffenden Daten (Data Message Stream) werden dabei asynchron verarbeitet.

Der Upload-Datendienst dient zur Rückmeldung von Mengen, Zeiten und Zuständen (Statusmeldungen) von Aufträgen (Event Message Stream). Bei Dauern und Mengen werden hier zusätzlich zu den relativen Änderungen auch die absoluten Werte übertragen.

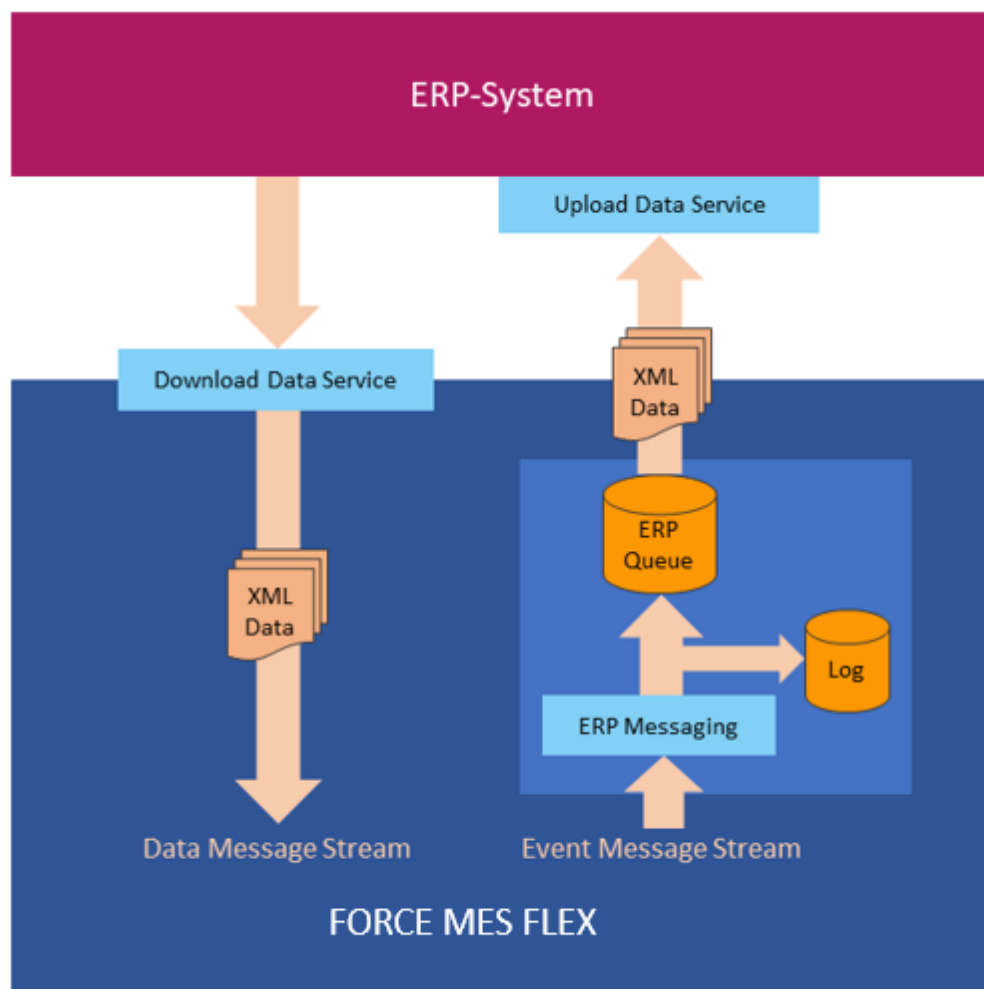


Bild 8: ERP-Download- und -Upload-Datendienst

Um einem Datenverlust bei Systemausfällen vorzubeugen, kommen bei der Datenübertragung im Upload-Datendienst persistent queues zum Einsatz. Ausschließlich erfolgreich an das ERP übermittelte Nachrichten werden zur Queue hin quittiert. Dadurch ist eine Ausfallsicherheit gewährleistet. Fehlgeschlagene Übertragungen verbleiben in der Queue.

Zur Rückverfolgbarkeit der kommunizierten Nachrichten dient außerdem eine Log-Datei, die laufend befüllt wird.

8 Konfiguration ERP-Interface

Sollen Daten via FORCE Bridge API übertragen werden, muss über die Workbench (siehe Kapitel 0) ein User mit Passwort und entsprechenden ERP-Benutzerrollen für die API-Client-Verbindung konfiguriert werden.

8.1 ERP-Download

Pfad: Konfigurationen > Module > Runtime > ERP > ERP-Download

Diese Konfiguration steuert das Mapping von Nachrichten eines ERP-Systems. Dabei wird das spezifische SAP-IDOC Format in das generische ERP XML Format transformiert. Die Transformation geschieht über XSLT.

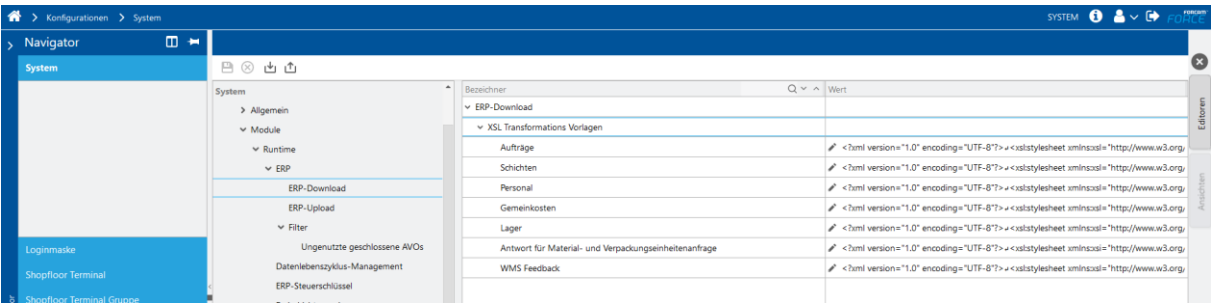


Bild 9: Konfiguration ERP-Download

Bezeichnung	Erklärung
validation	Definiert, ob eingehende Daten gemäß XSD-Schema und entsprechender Aktion (ADD, DELETE) validiert werden sollen. Wenn aktiv werden nicht gültige Daten abgelehnt und nicht verarbeitet.
XSL Transformers Vorlagen	
Aufträge	XSLT, welches das Mapping von IDOC auf ERP XML für Aufträge definiert
Schichten	XSLT, welches das Mapping von IDOC auf ERP XML für Schichten definiert
Personal	XSLT, welches das Mapping von IDOC auf ERP XML für Personal definiert
Gemeinkosten	XSLT, welches das Mapping von IDOC auf ERP XML für Gemeinkosten definiert
Lager	XSLT, welches das Mapping von IDOC auf ERP XML für Lager definiert
WMS Feedback	XSLT, welches das Mapping von IDOC auf ERP XML für WMS Feedback definiert
Antwort für Material- und Verpackungseinheitenanfrage	XSLT, welches das Mapping von IDOC auf ERP XML für Material- und Verpackungseinheit definiert

Tabelle 13: Konfiguration ERP-Download

8.2 ERP-Upload

Pfad: Konfigurationen > Module > Runtime > ERP > ERP-Upload

Diese Konfiguration legt die Verbindungs-Parameter zu ERP-Systemen für den ERP-Upload fest.

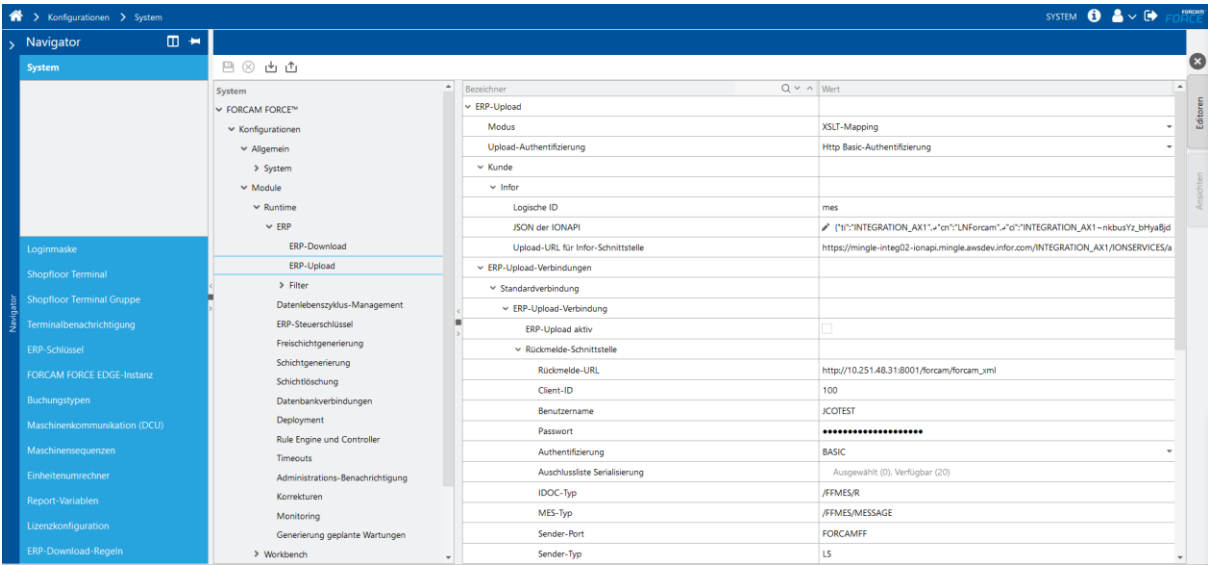


Bild 10: Konfiguration ERP-Upload

Bezeichnung	Erklärung
Modus	XSLT-Mapping für SAP IDOC Upload Generisch für XML-Upload
Upload Authentifizierung	HTTP Basic Auth Infor spezifische Authentifizierung gemäß IONAPI Definition
Kunde	
Infor: Logische ID	Logische ID für Infor Integration
Infor: JSON der IONAPI	IONAPI JSON für Infor Integration
Infor: Upload URL für Infor Schnittstelle	URL des Upload Endpoints für Infor Integration

Tabelle 14: Konfiguration ERP-Upload

8.2.1 ERP-Upload-Verbindungskonfiguration

Die *Standardverbindung* gilt zunächst für alle ERP-Schlüssel.

Bezeichnung	Erklärung
ERP-Upload aktiv	Flag: Sollen ERP-Meldungen geschickt werden
Rückmeldes-Schnittstelle	
Rückmelde URL	URL des Upload Endpoints für Standard ERP Integration
Client ID	Client-ID die bei SAP-Integration im IDOC-Header als Wert für MANDT übermittelt wird.
Benutzername	Username für BASIC Authentifizierung
Passwort	Passwort für BASIC Authentifizierung
Ausschlussliste	Konfigurierte IDOCs werden nicht serialisiert
IDOC-Typ	Client-ID die bei SAP-Integration im IDOC-Header als Wert für IDOCTYP übermittelt wird.
MES-Typ	Client-ID die bei SAP-Integration im IDOC-Header als Wert für MESTYP übermittelt wird.
Sender-Port	Client-ID die bei SAP-Integration im IDOC-Header als Wert für SNDPOR übermittelt wird.
Sender-Typ	Client-ID die bei SAP-Integration im IDOC-Header als Wert für SNDPRT übermittelt wird.
Sender-Nummer	Client-ID die bei SAP-Integration im IDOC-Header als Wert für SNDPRN übermittelt wird.
Empfänger-Port	Client-ID die bei SAP-Integration im IDOC-Header als Wert für RCVPOR übermittelt wird.
Empfänger-Typ	Client-ID die bei SAP-Integration im IDOC-Header als Wert für RCVPRT übermittelt wird.
Empfänger-Nummer	Client-ID die bei SAP-Integration im IDOC-Header als Wert für RCVPNR übermittelt wird.
IDOC-Vorlage	XSLT mit der das IDOC aus dem ERP XML erzeugt wird
Serialisierungsschnittstelle	
Serialisierung aktiv	Flag: Serialisierung aktiv
Serialisierungsmodus	Vorgangsbezogene Serialisierung oder Arbeitsplatzbezogene Serialisierung
URL zur Synchr. der Serialisierung	URL zur Abfrage der Kanal/Zähler Nummern von SAP via SOAP Request
Benutzername	Username für Basic Authentifizierung zur Abfrage der Kanal/Zähler Nummern von SAP via SOAP Request
Passwort	Passwort für Basic Authentifizierung zur Abfrage der Kanal/Zähler Nummern von SAP via SOAP Request
Authentifizierung	Verwendete Authentifizierung (BASIC, NONE) zur Abfrage der Kanal/Zähler Nummern von SAP via SOAP Request
Synchr. SOAP-Request Vorlage	Vorlage zur Abfrage der Kanal/Zähler Nummern von SAP via SOAP Request

Bezeichnung	Erklärung
Kleinste Kanalnummer (inklusive)	Kleinste Kanalnummer zur Abfrage der Kanal/Zähler Nummern von SAP via SOAP Request
Größte Kanalnummer (inklusive)	Größte Kanalnummer zur Abfrage der Kanal/Zähler Nummern von SAP via SOAP Request
Größte Zählernummer	Größte bei der Serialisierung verwendete Zählernummer
Präfix	Für Serialnummern verwendetes Präfix

Tabelle 15: Konfiguration ERP-Upload-Verbindung

Weitere Verbindungen können für bestimmte ERP-Schlüssel definiert werden.

Bezeichnung	Erklärung
Mandant	Mandant für den die Verbindung definiert wird.
Buchungskreis	Buchungskreis für den die Verbindung definiert wird.
Werk	Werk für das die Verbindung definiert wird.
System-ID	System-ID für die die Verbindung definiert wird.

Tabelle 16: Konfiguration weiterer Verbindungen

8.3 Filter – Ungenutzte geschlossene AVOs

Pfad: Konfigurationen > Module > Runtime > ERP > Filter > Ungenutzte geschlossene AVOs

Diese Konfiguration steuert einen Filter, der es erlaubt, die technisch (TABG) abgeschlossenen Arbeitsvorgänge zu ändern, wenn ihre letzte Statusänderung innerhalb der konfigurierten Anzahl von Tagen erfolgte.

Beispiel: Ein Arbeitsvorgang wurde vor zwei Tagen abgeschlossen und der konfigurierte Wert im Filter ist 2 Tage, dann kann der AVO nicht im ERP geändert werden. Zumindest wird die Verarbeitung in der Schnittstelle zwischen MES FLEX und ERP abgewiesen.

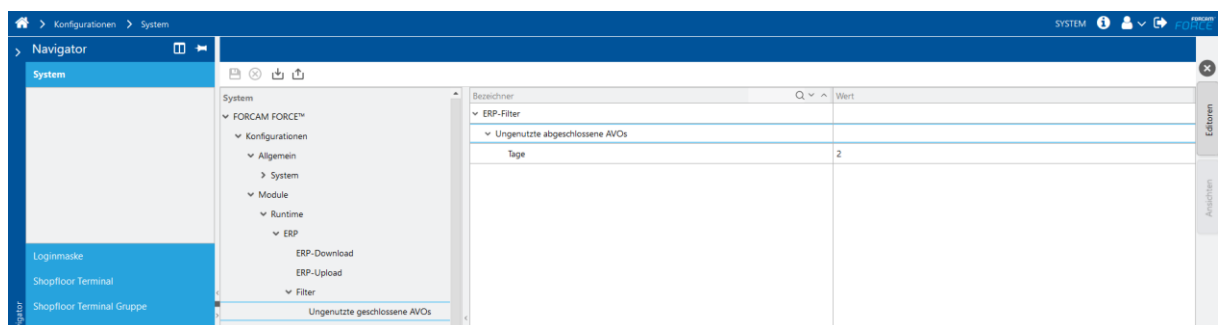


Bild 11: Konfiguration Filter - Ungenutzte geschlossene AVOs

Bezeichnung	Erklärung
Tage	Anzahl der Tage in der Vergangenheit, an denen TABG-Vorgänge aus dem ERP aktualisiert werden können.

Tabelle 17: Konfiguration: ERP Filter

8.4 ERP-Steuerschlüssel

Pfad: Konfigurationen > Module > Runtime > ERP-Steuerschlüssel

Jeder AVO hat einen für ihn definierten Steuerschlüssel. Wenn eine Änderung an einem AVO vorgenommen wird, wird ein DomainAttributeChangeCommand für den Steuerschlüssel erzeugt. Dieser Befehl wird verwendet, um ein OperationRescheduleERPDTO an das ERP-System zu senden. Die Konfiguration steuert, ob diese Funktion aktiv ist.

8.5 Anbindung über Infor ION

Kommt bei der Anbindung Infor ION zum Einsatz, muss folgende Konfiguration in der Workbench von MES FLEX vorgenommen werden:

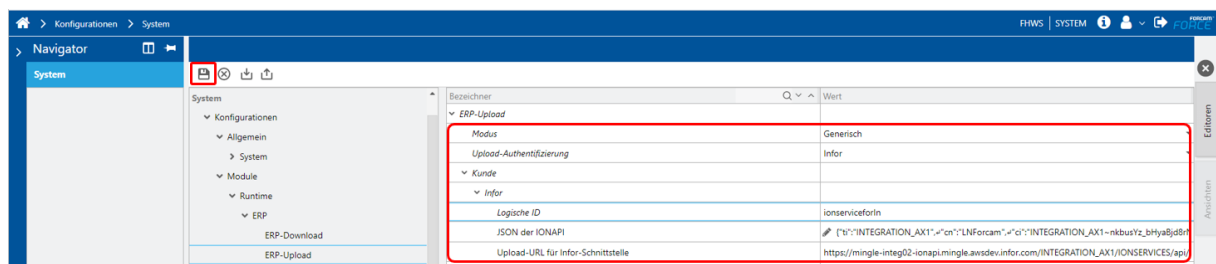


Bild 12: Konfiguration Upload-Datendienstes für Infor ION

Um den ERP-Upload-Datendienst für Infor ION zu konfigurieren:

1. Setzen Sie den **Modus** auf **Generisch**.
2. Setzen Sie die **Upload-Authentifizierung** auf **Infor**.
3. Tragen Sie die vom Infor ERP vorgegebene **Logische ID** ein.
4. Fügen Sie das vom Infor ERP vorgegebene **JSON der IONAPI**-Dokument ein.
5. Fügen Sie die **Upload-URL für Infor-Schnittstelle** eintragen.
6. Speichern Sie die Daten.

8.6 Zuweisungsverfahren der AVOs auf Arbeitsplatzgruppen

Im Regelfall werden einzelne AVOs innerhalb eines Fertigungsauftrags über das Attribut **TargetWorkplace** einem bestimmten Arbeitsplatz zugewiesen. MES FLEX bietet jedoch auch eine erweiterte Funktion für die dynamische Zuweisung der AVOs auf Arbeitsplätze innerhalb einer Gruppierung.

Arbeitsplätze können in der Workbench über die Funktion der **Arbeitsplatz-Hierarchie** beliebig hierarchisch angeordnet werden (weitere Informationen Handbuch zu **Stammdaten und Systemkonfiguration**).

Für eine dynamische Zuweisung der AVOs ist dabei die ERP-Hierarchie nutzbar. Auf der untersten Ebene können dort Arbeitsplätze als Arbeitsplatzgruppe zusammengefasst werden. Übernehmen Arbeitsplätze in einer solchen Gruppierung die gleiche Funktion, nennt man die Gruppierung auch

Kapazitätsgruppe. Für eine Kapazitätsgruppe kann festgelegt werden, wie die dynamische Zuweisung der AVOs freigegebener Aufträge erfolgen soll.

Verfahren	Beschreibung
Auto-Split	Die AVOs werden auf die Arbeitsplätze innerhalb der Kapazitätsgruppe bzgl. der zu produzierenden Sollmenge aufgesplittet (Vorgangssplit). Dabei erhält jeder Arbeitsplatz möglichst den gleichen Anteil der zu produzierenden Sollmenge. Ist dabei die Sollmenge nicht durch die Anzahl der Arbeitsplätze teilbar, so wird die Restmenge auf einzelne Arbeitsplätze verteilt. Z. B. führt dies bei einer Sollmenge von 100 Stück und 3 Arbeitsplätzen zu einer Aufsplittung von 33 + 33 + 34 Stück.
Pull	Die AVOs stehen zuerst allen Arbeitsplätzen innerhalb der Kapazitätsgruppe zur Verfügung. Der AVO kann dann manuell (z. B. durch einen Vorarbeiter) einem Arbeitsplatz innerhalb der Kapazitätsgruppe zugewiesen werden, oder der erste Arbeitsplatz, der die Bearbeitung starten möchte, weist sich den AVO selbst zu. Der zugewiesene AVO verschwindet in der Shopfloor-Anzeige aller anderen Arbeitsplätze. Wird ein zugewiesener AVO abgebrochen, so steht er wieder allen zur Verfügung und das Verfahren beginnt von vorne.

Tabelle 18: Verfahren bei der Zuweisung der AVOs pro Kapazitätsgruppe

Entsprechend kann in der Workbench das Zuweisungsverfahren gewählt werden.

Pfad: Stammdaten > Arbeitsplatz > Arbeitsplatz-Hierarchie > Hierarchie-Baum > ERP – ERP

Um AVOs einer Arbeitsplatzgruppe im Verfahren Auto-Split oder Pull zuzuweisen:

1. Wählen Sie im Hierarchie-Baum den entsprechenden Knoten in der untersten Hierarchie-Ebene an.
2. Öffnen Sie in der Spalte **Attribute** das Drop-down-Menü.
→ Ein Dialogfenster wird geöffnet.
3. Wählen Sie beim Attribut **Verhalten beim Split** den Wert **Auto-Split**.
Oder
Wählen Sie beim Attribut **Verhalten beim Split** den Wert **Pull**.
4. Bestätigen Sie die Auswahl mit **Ok**.

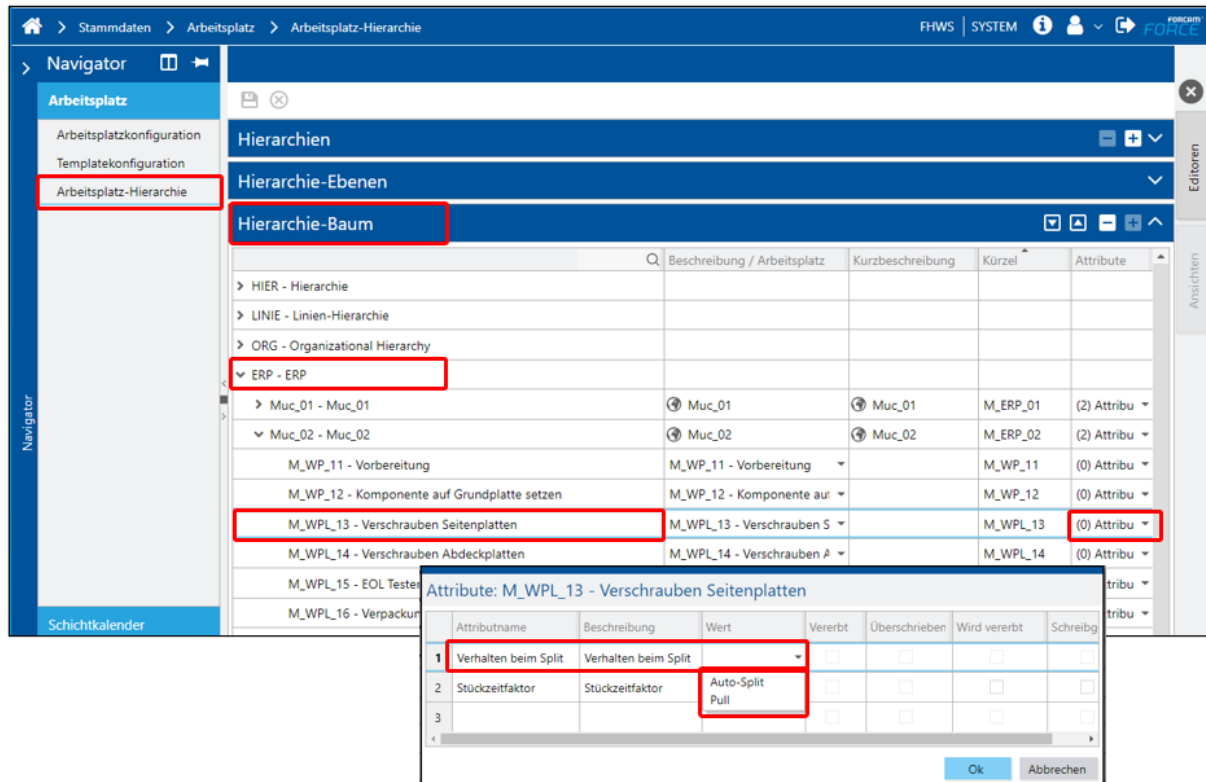


Bild 13: Auswahl des Verfahrens bei der Zuweisung der Vorgänge

8.6.1 Stückzeitfaktor

- i** Der **Stückzeitfaktor** kann nur in Kombination mit dem Verfahren **Auto-Split** angewendet werden.

Er wird immer verwendet, wenn die Stückzeit am Arbeitsplatz nicht als Zeit pro Stück, sondern als Zeit pro s Stück angegeben ist. Dann kann mit dem Stückzeitfaktor s „gegengesteuert“ werden, um die Zeit je produzierter Einheit (TIME_PER_UNIT) wieder korrekt zu berechnen:

$$\text{Zeit pro Einheit} = \text{Stückzeit} * \text{Einheit} / \text{Stückzeitfaktor}$$

In einer Arbeitsplatzgruppe von Maschinen unterschiedlicher Technologie kann es vorkommen, dass diese unterschiedliche Bearbeitungsgeschwindigkeiten vorweisen. Die Stückzeit einer langsameren Maschine kann so über einen Stückzeitfaktor $s < 1$ nachjustiert werden und eine schnellere Maschine über einen Stückzeitfaktor $s > 1$ (beispielsweise kann ein Wert von 2.0 für eine doppelt schnell produzierende Maschine oder 0.75 für eine 25% langsamere Maschine).

- i** Der Default-Wert des Stückzeitfaktors ist 1 und führt ebenso wie ein Wert von 0 zu keiner Anpassung.

Pfad: Stammdaten > Arbeitsplatz > Arbeitsplatz-Hierarchie > Hierarchie-Baum > ERP – ERP

Um den Stückzeitfaktor festzulegen:

1. Wählen Sie im Hierarchie-Baum den entsprechenden Knoten in der untersten Hierarchie-Ebene an.
2. Öffnen Sie in der Spalte **Attribute** das Drop-down-Menü.
→ Ein Dialogfenster wird geöffnet.
3. Tragen Sie beim Attribut **Stückzeitfaktor** den gewünschten **Wert** ein.

- Bestätigen Sie die Auswahl mit **Ok**.

Attribute: M_WPL_13 - Verschrauben Seitenplatten

	Attributname	Beschreibung	Wert	Vererbt	Überschrieben	Wird vererbt	Schreibgeschü
1	Verhalten beim Spli	Verhalten beim Spli		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Stückzeitfaktor	Stückzeitfaktor	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bild 14: Einstellung des Stückzeitfaktors

8.7 Arbeitsplatz-spezifische Konfiguration

8.7.1 ERP-Upload Aktivierung

Pfad: Stammdaten > Arbeitsplatz > Arbeitsplatzkonfiguration

Der ERP-Upload-Datendienst kann über die Einstellung **ERP-Upload aktiv** auch pro Arbeitsplatz aktiviert bzw. deaktiviert werden.

- Eine zwischenzeitliche Deaktivierung wird in der Regel bei der Neuanlage eines Arbeitsplatzes verwendet. Erst zum Zeitpunkt des wirksamen Freischalten des neuen Arbeitsplatzes wird dessen ERP-Upload aktiviert. So können Doppelbuchungen in der Übergangsphase verhindert werden.

Stammdaten > Arbeitsplatz > Arbeitsplatzkonfiguration

FHWS | SYSTEM |

Navigator

- Arbeitsplatz
 - Arbeitsplatzkonfiguration
 - Template-Konfiguration
 - Arbeitsplatz-Hierarchie
- Schichtkalender
- Betriebsdatenerfassung
- Montageplan

Stammdaten

Anzahl Arbeitsplätze: 35/100

Arbeitsplatzname	Verbuchung aktiv	ERP-Upload aktiv	Zugew
EduDemo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
EduDemo1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
M_Line_1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
M_WPL_13	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
M_WPL_14	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
M_WPL_15	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
M_WPL_16	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Bild 15: ERP-Upload Aktivierung pro Arbeitsplatz

8.7.2 ERP Zykluszeit

Pfad: Stammdaten > Arbeitsplatz > Templatekonfiguration > Selektionsbaum Templates > Arbeitsplatz > Buchungslogik

Das Zeitintervall, in dem ERP-Uploads bzgl. Mengen stattfinden, kann in der Workbench eingestellt werden. Eine Erhöhung des Zeitintervalls ist beispielsweise sinnvoll, wenn die Systembelastung durch die laufenden Datenübertragungen zu hoch ist.

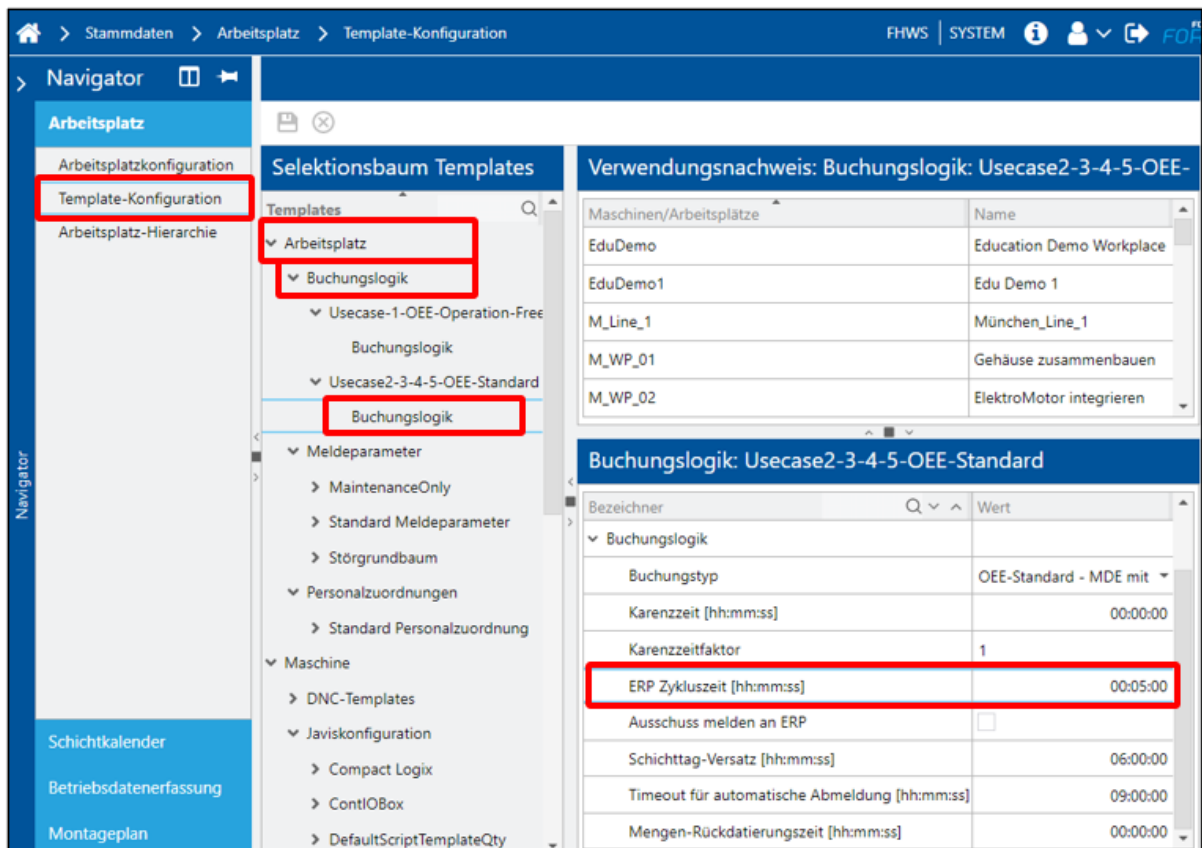


Bild 16: Einstellung der ERP Zykluszeit

8.7.3 Ausschuss melden an ERP

Pfad: Stammdaten > Arbeitsplatz > Templatekonfiguration > Selektionsbaum Templates > Arbeitsplatz > Buchungslogik

Im Falle der Mengenbuchung von Ausschuss kann die Menge zusammen mit dem Ausschussgrund an das ERP-System gemeldet werden. Dies ist ebenfalls in der Workbench aktivierbar bzw. deaktivierbar.

Stammdaten > Arbeitsplatz > Template-Konfiguration

FHWS | SYSTEM

Navigator

- Arbeitsplatz
 - Arbeitsplatzkonfiguration
 - Template-Konfiguration
 - Arbeitsplatz-Hierarchie
- Schichtkalender
- Betriebsdatenerfassung
- Montageplan

Selektionsbaum Templates

Templates

- Arbeitsplatz
 - Buchungslogik
 - Usecase-1-OEE-Operation-Free
 - Buchungslogik
 - Usecase2-3-4-5-OEE-Standard
 - Buchungslogik
 - Meldeparameter
 - MaintenanceOnly
 - Standard Meldeparameter
 - Störgrundbaum
 - Personalzuordnungen
 - Standard Personalzuordnung
 - Maschine
 - DNC-Templates
 - Javiskonfiguration
 - Compact Logix
 - ContIOBox
 - DefaultScriptTemplateQty

Verwendungsnachweis: Buchungslogik: Usecase2-3-4-5-OEE-

Maschinen/Arbeitsplätze	Name
EduDemo	Education Demo Workplace
EduDemo1	Edu Demo 1
M_Line_1	München_Line_1
M_WP_01	Gehäuse zusammenbauen
M_WP_02	ElektroMotor integrieren

Buchungslogik: Usecase2-3-4-5-OEE-Standard

Bezeichner	Wert
Buchungstyp	OEE-Standard - MDE mit
Karenzzeit [hh:mm:ss]	00:00:00
Karenzzeitfaktor	1
ERP Zykluszeit [hh:mm:ss]	00:05:00
Ausschuss melden an ERP	<input type="checkbox"/>
Schichttag-Versatz [hh:mm:ss]	06:00:00
Timeout für automatische Abmeldung [hh:mm:ss]	09:00:00
Mengen-Rückdatierungszeit [hh:mm:ss]	00:00:00

Bild 17: Ausschuss melden an ERP

Der Ausschussgrund muss mit den Ausschussgründen im ERP-System übereinstimmen. ERP-Ausschussgründe sind in den Stammdaten der Workbench zu pflegen.

Pfad: Stammdaten > Betriebsdatenerfassung > Qualitätsmerkmale

Stammdaten > Betriebsdatenerfassung > Qualitätsmerkmale

Navigator

- Arbeitsplatz
- Schichtkalender
- Betriebsdatenerfassung**
- Phasen/Status-Definitionen
- Statusdetails
- Zeitbasen
- Qualitätsmerkmale**
- Betriebszustandsklassen
- Personaldaten
 - Personal
 - Qualifikation

Qualitätsmerkmale

Kurzbeschreibung	Beschreibung	Farbe	Code	Standard Qualitätsdetail	Sortierfolge
AUS	Ausschuss	#C60A44	2	S1	
GUT	Gut	#00CC84	1	Y	
NACH	Nacharbeit	#FFB500	3	R1	

Qualitätsdetails

Kürzel	Kurzbeschreibung	Beschreibung	Farbe	Qualitätsmerkmal	ERP Code	Sortierfolge
R1	R1	Oberfläche	#FFB500	NACH		
R2	R2	Geometrie	#FFDD00	NACH		
S1	S1	Oberfläche	#C60A44	AUS		
S2	S2	Geometrie	#660033	AUS		
Y	GUT	Gutmenge	#00CC84	GUT		
Y2	Y2	Sehr Gut	#ccff00	GUT		

Bild 18: Ausschussgründe verwalten

9 Datenstrukturen

9.1 XSL Basis-Datentypen

Die Transformation der übermittelten XML-Daten wird über XSDs (XML Schema Definitions) geregelt, die fest in MES FLEX eingebunden sind. XSDs beinhalten Informationen über den Aufbau der verwendeten Datenstrukturen. Dabei werden Datenelemente bzw. Attribute auf Basis von standardmäßig in XSL vorhandenen Datentypen spezifiziert. Dazu gehören folgende Datentypen:

Datentyp	Anwendung
xs:string	Zeichenfolge String
xs:boolean	Flag
xs:long	Ganzzahl
xs:double	Gleitkommazahl
xs:date	Terminangabe im Format: YYYY-MM-DD Y = Jahr M = Monat D = Tag Beispiel: 2021-01-30
xs:dateTime	Terminangabe im Format: YYYY-MM-DDTHH:mm:SS Y = Jahr M = Monat D = Tag H = Stunde m = Minute S = Sekunde Beispiel: 2021-01-30T09:10:20
xs:duration	Zeitintervall im Format: PyYmMdDThHmMsS y = Jahre m = Monate d = Tage h = Stunden m = Minuten s = Sekunden Beispiele: P1Y11M30D P30D23H59M45S

Tabelle 19: Angewendete XSL Basis-Datentypen

9.2 XSL Komplexe Datentypen

Aus den XSL Basis-Datentypen können über die XSL komplexe Datentypen gebildet werden. Dazu vereint man mehrere Datenelemente bzw. Attribute in einer hierarchischen Struktur. Auch diese gebildeten komplexen Datentypen werden wiederum für die Spezifikation anwendungsspezifischer Datenstrukturen genutzt. Beispiele anwendungsspezifischer Datenstrukturen in MES FLEX sind Fertigungsaufträge, Personalstämme oder Schichten.

Eine XSD für ERP-Schlüssel (*ERPKeyType*) und einen darauf aufbauenden Datentyp für Aufträge (*OrderType*) sieht beispielsweise folgendermaßen aus:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="ERPKeyType">
    <xs:sequence>
      <xs:element name="Client">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="1"/> <xs:maxLength value="32"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="CompanyCode">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="1"/> <xs:maxLength value="32"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="Plant">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="1"/> <xs:maxLength value="32"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="OrderType">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="OrderERPKey" type="ct:ERPKeyType">
        </xs:element>
        <xs:element name="OrderNumber">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:minLength value="1"/> <xs:maxLength value="32"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="TargetStartDateTime" type="xs:dateTime" minOccurs="0">
        </xs:element>
        <xs:element name="TargetQuantity" minOccurs="0">
          <xs:simpleType>
            <xs:restriction base="xs:double">
              <xs:minExclusive value="0"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Bild 19: Beispielhafte XSD für die Datentypen *ERPKeyType* und *OrderType*

Nach diesem Schema erzeugte XML-Datensätze eines Auftrags (vom Datentyp *OrderType*) können dann folgendermaßen aussehen:

```
<?xml version="1.0" encoding="UTF-8"?>
<ord:OrderType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <ord:OrderERPKey>
    <Client>123</Client>
    <CompanyCode>4900</CompanyCode>
    <Plant>RBURG</Plant>
  </ord:OrderERPKey>
  <ord:OrderNumber>MMDD5929</ord:OrderNumber>
  <ord:TargetStartDateTime>2020-08-01T00:00:00</ord:TargetStartDateTime>
  <ord:TargetQuantity>1</ord:TargetQuantity>
</ord:OrderType>
```

Bild 20: Beispiel für XML-Datensatz eines Fertigungsauftrags

Eine XSD stellt also die Informationen bereit für den Aufbau und die Anordnung der Elemente bzw. Attribute eines Datensatzes im XML-Format.

Dies umfasst folgende Informationen:

- Anzahl und Reihenfolge der Elemente und Kind-Elemente
- Datentypen der Elemente
- Voreingestellte und konstante Werte für die Elemente

In den nachfolgenden Kapiteln wird konkreter auf die in MES FLEX eingesetzten Datenstrukturen bzw. XSDs eingegangen. Unterschieden wird dabei nach Download- und Upload-Datendienst.

10 Download

Über den Download-Datendienst wird MES FLEX mit folgenden für die Produktion relevanten Datensätzen versorgt:

- Aufträge
 - Auftragskopf
 - Arbeitsvorgänge
 - Komponenten
 - Fertigungshilfsmittel
 - Sachmerkmale des Materials
- Schichtdaten
- Personaldaten
- Gemeinkosten
- Lagerdaten aus Lagerverwaltungssystem
- Antwort auf Mengenanfragen

Diese Datensätze werden als Sequenz von Attributen übertragen, deren Struktur über entsprechende XSDs genau spezifiziert ist.

- ❗ Nur die Auftragskopfdaten und die zugehörigen Vorgänge werden zwingend benötigt. Sämtliche andere Datensätze werden nur zu Anzeigezwecken auf der MES FLEX Benutzeroberfläche oder für funktionale Erweiterungen wie das Werkzeugdatenmanagement (TDM), Dokumentenverwaltung sowie das Behälter-Trace benötigt.
- ❗ Geschieht ein Fehler in der Datenverarbeitung, werden die Daten zurückgerollt, d. h. die Verarbeitung der Nachricht wird verworfen.

Einige Attribute sind obligatorisch anzugeben. Dies hängt davon ab, ob es sich bei der Datensatz-Aktion um einen Neueintrag bzw. Änderung oder ein Löschen handelt. Dies ist in den nachfolgenden Tabellen (wo relevant) in den rechten Spalten ersichtlich (Pflichtfeld für Aktion).

10.1 Download Basis-Datentypen

MES FLEX verwendet neben den standardmäßig in der XSL vorhandenen Datentypen spezielle Datentypen, die ebenfalls für die Spezifikation der Download-Datensätze eingesetzt werden.

10.1.1 Zeitfelder

Neben dem in XSL bereits vorhandenen Datentyp *xs:dateTime* (siehe Tabelle 19: Angewendete XSL Basis-DatentypenTabelle 19) werden Zeitangaben auch in Form einer Sequenz von Zeitfeldern übertragen. Dafür gibt es in MES FLEX eine XSD für den komplexen Datentyp *ct:timeUnitType*.

Zur Definition eines Zeitfeldes gehört immer die Definition eines Zeitwerts und einer zugehörigen Zeiteinheit.

Basierend auf dieser Einheit wird der übermittelte Wert des Zeitfelds auf einen MES FLEX internen einheitlichen Zeitbasiswert umgerechnet. Ohne eine gültige Einheit kann nicht sichergestellt werden, dass der Zeitwert richtig umgerechnet und interpretiert wird.

Folgende Einheiten bei Zeitwerten werden für den ERP-Download unterstützt:

Zeiteinheit	Beschreibung
MS	Millisekunde
S	Sekunde
MIN	Minute
STD	Stunde
H	Stunde
M90	Minute / 90
TMU	Technische Minute

Tabelle 20: Unterstützte Zeiteinheiten bei Zeitfeldern

Eine XML-Nachricht mit einer Zeitangabe, die auf Basis von Zeitfeldern erstellt wurde, kann so aussehen:

```
<ord:StandardValue1>20</ord:StandardValue1>
<ord:StandardUnit1>MIN</ord:StandardUnit1>
<ord:StandardValue2>28</ord:StandardValue2>
<ord:StandardUnit2>S</ord:StandardUnit2>
```

Bild 21: Beispiel einer Zeitangabe in Form von Zeitfeldern

10.1.2 Multilinguale Texte

Texte können allgemein als multilinguale Texte übertragen werden. MES FLEX unterstützt auf der graphischen Benutzeroberfläche im Moment folgende Sprachen:

- Deutsch
- Englisch (US)
- Englisch (GB)
- Französisch
- Chinesisch

Zur Übertragung gibt es hierfür in MES FLEX XSDs für folgende komplexe Datentypen:

Datentyp	Anwendung
ct:SupportedLanguagesType	Unterstützte Sprache mit folgenden möglichen Werten (Codes) nach ISO 639: de-DE en-US en-GB fr-FR zh-CN
ct:MultilanguageTextType	Multilingualer Text als Sequenz von Sprach-Codes und zugehörigen Texten.

Tabelle 21: Datentypen für multilinguale Texte

Eine XML-Nachricht mit einem multilingualen Text, die auf Basis dieser Datentypen erstellt wurde, kann so aussehen:

```
<ord:Description>
  <Translation>
    <Language>de-DE</Language>
    <Text>FORCE ERP Verbindung</Text>
  </Translation>
  <Translation>
    <Language>en-US</Language>
    <Text>FORCE ERP interface</Text>
  </Translation>
</ord:Description>
```

Bild 22: Beispiel eines multilingualen Textes

Falls eine Text-Übersetzung fehlt, wird der Text der im System festgelegten Default-Sprache übernommen.

Fehlt auch dieser Text, werden die Sprachcodes absteigend sortiert und die Übersetzung des ersten Sprachcodes übernommen.

Konfigurierte Sprachen	Default	Übertragene Übersetzungen	Übernommene Übersetzungen
de-DE	x	n/a	Englisch (GB)
en-GB		Englisch (GB)	Englisch (GB)
en-US		Englisch (US)	Englisch (US)
fr-FR		n/a	Englisch (GB)

Tabelle 22: Beispiel einer Textersetzung bei nicht vorhandener Übersetzung in der Default-Sprache

10.1.3 Datensatz-Aktion

Ob der übertragene Datensatz einem Neueintrag, einer Änderung oder dem Löschen des entsprechenden Datensatzes in MES FLEX dienen soll, wird über ein Attribut des komplexen Datentyps *ct:actionType* geregelt.

Aktion	Beschreibung
ADD	Datensatz anlegen/ändern
DELETE	Datensatz löschen
NOP	Datensatz nicht übernehmen Dies gilt nicht für Kind-Elemente. So können gezielt untergeordnete Attribute zur Verarbeitung übertragen werden. Übergeordnete Attribute werden nicht übernommen.

Tabelle 23: Mögliche Aktionen bei der Übertragung der Daten

10.1.4 ERP-Schlüssel

Ein ERP-Schlüssel ist ein individueller Schlüssel für die Zuweisung von Aufträgen, Maschinen, Personal oder Schichten zu einem Mandanten, Buchungskreis oder Betriebsgebäude im ERP-System. ERP-Schlüssel können in der Workbench konfiguriert werden, damit sie mit dem ERP-System übereinstimmen. Zur Übertragung eines ERP-Schlüssels dient der komplexe Datentyp *ct:ERPKeyType* mit folgenden Attributen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
Client	Mandant	xs:string(32)	x	x
CompanyCode	Buchungskreis	xs:string(32)	x	x
Plant	Werk	xs:string(32)	x	x
SystemId	System-ID	xs:string(32)	x	

Tabelle 24: Attribute im ERP-Schlüssel (siehe auch ERPDownloadOrder.XSD)

Der Buchungskreis kann als finanzbuchhalterische Einheit der Firma genutzt werden, um die Geschäftsabwicklung zu unterscheiden.

Die System-ID dient der eindeutigen Identifikation des ERP-Schlüssels im System. Das ist wichtig, falls die Attribute Mandant, Buchungskreis und Werk nicht eindeutig sind. Die System-ID kann so z. B. zur Unterscheidung von Anbindungen an das Test- oder Produktiv-System genutzt werden.

Wo überall der ERP-Schlüssel zur Anwendung kommt, wird in den nachfolgenden Kapiteln genauer beschrieben.

10.1.5 Material

Zur Definition und Übertragung eines Materials steht der komplexe Datentyp *ct:MaterialType* mit folgenden Attributen zur Verfügung:

Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
Description	Beschreibung	ct:MultilanguageTextType	x	
Number	Materialnummer	xs:string(255)	x	
Type	Materialtyp	xs:string(255)		

Tabelle 25: Attribute im Datensatz Material

Die Materialnummer, auch Teilenummer genannt, dient primär zur Identifikation des Materials. Die Angabe des Materialtyps ist optional.

Der Begriff Material dient in MES FLEX als allgemeiner Begriff der verarbeiteten Teile in der Fertigung. Material wird an einem Arbeitsplatz verarbeitet (z. B. bearbeitet oder verbaut) und daraus kann (aber muss nicht) neues Material entstehen. Die in diesen Verarbeitungsprozess am Arbeitsplatz eingehenden Materialien werden als Komponenten bezeichnet, wenn dabei ein anderes bzw. neues Material entsteht. Wenn ein Material am Arbeitsplatz nur bearbeitet wird, z. B. gefräst oder geformt wird, so spricht man weiterhin nur von Material.

Es gibt also Material, das aus mehreren Komponenten zusammengesetzt wird. Das zu fertigende Material benötigt dabei eine Mindestanzahl dieser Komponenten, um korrekt verbaut zu werden und in dem gewünschten Produkt zu resultieren. Daraus bildet sich die Komponentenliste, welche mit den Vorgängen übermittelt werden kann (siehe Kapitel 0 bzw. 0).

Zum Beispiel: Das gewünschte Endmaterial ist ein Zylindermotor. Er besteht, vereinfacht ausgedrückt, aus 4 Zylindern, 4 Schrauben und 1 Kurbelwelle. Diese Teile bilden dann die Komponenten des Materials Zylindermotor. Man kennt diese auch in Form einer sogenannten Stückliste.

Wird ein Material erstmalig an MES FLEX übermittelt, wird es in einer MES FLEX -internen Materialliste hinterlegt und steht fortan für die Suche (z. B. in der Leistungsanalyse des Office-Client) zur Verfügung. Aus diesem Grund ist es sinnvoll, nicht nur die Materialnummer zu übertragen, sondern auch einen Materialtyp und eine sinnvolle Beschreibung. Außerdem können für eine noch detailgenauere Spezifikation die Sachmerkmale laut Kapitel 0 genutzt werden.

10.1.6 Benutzerdatenfelder

Für die Übertragung von benutzereigenen Daten und deren Anzeige auf der Benutzeroberfläche von MES FLEX steht der komplexe Datentyp *ct:UserDataType* zur Verfügung mit folgenden frei verwendbaren Attributen:

Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/ Ändern	Löschen
UF1	Benutzerfeld 1	xs:string(255)		
UF2	Benutzerfeld 2	xs:string(255)		
...				
UF50	Benutzerfeld 50	xs:string(255)		


Tabelle 26: Attribute im Datensatz Benutzerdatenfelder

10.2 Auftragsdaten

Für die Auftragsversorgung aus dem ERP-System kommt eine XSD für Aufträge zum Einsatz. Diese XSD regelt die Übertragung der Auftragskopfdaten und der folgenden untergeordneten Objekte:

- Arbeitsvorgänge (AVOs) des Auftrags
- Sachmerkmale des eingesetzten Materials
- Verwendete Komponenten
- Verwendete Fertigungshilfsmittel

Für die Auftragskopfdaten und untergeordneten Objekte wird jeweils ein eigener XML-Datensatz erzeugt.

-  Komponenten sind nur notwendig, wenn die zu verbauenden Komponenten in MES FLEX angezeigt oder an das ERP-System zurückgemeldet werden sollen.

Fertigungshilfsmittel sind nur notwendig, wenn sie zur Anzeige in MES FLEX benötigt werden oder falls das Werkzeugdatenmanagement (TDM) eingesetzt wird.

10.2.1 Auftragskopf

Die wichtigsten Bestandteile im Datensatz Auftragskopf sind:

- der *ERP-Schlüssel* und die *Auftragsnummer* zur Identifikation des Auftrags
- der *Soll-Starttermin* und *Soll-Endtermin* als Zeitvorgaben für die Fertigung
- das *produzierte Material* (Endprodukt) des Auftrags
- die *Sollmenge mit Mengeneinheit* als Mengenvorgabe des produzierten Materials

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/ Ändern	Löschen
action	Datensatz-Aktion	ct:actionType	x	x
ERPKey	ERP-Schlüssel	ct:ERPKeyType	x	x
OrderNumber	Auftragsnummer	xs:string(32)	x	x
OrderSplit	Auftragssplit	xs:string(8)	x	x
ERPStatusIds	Status-IDs aus ERP-System (mit Leerzeichen als Trennungszeichen)	xs:string(255)		x
ERPStatusCodes	Zugehörige Status-Codes aus ERP-System (mit Leerzeichen als Trennungszeichen)	xs:string(255)		
ERPStatusCodesTranslation	Übersetzung für Status-Codes (mit Leerzeichen als Trennungszeichen)	xs:string(255)		
OrderPriority	Priorität	xs:string(255)		

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/ Ändern	Löschen
ReleaseDateTime	Freigabetermin	xs:dateTime		
TargetStartDateTime	Soll-Starttermin	xs:dateTime	x	
TargetEndTime	Soll-Endtermin	xs:dateTime	x	
TargetQuantity	Sollmenge	xs:double	x	
DisplayQuantityUnit	Mengeneinheit	xs:string(255)	x	
ProductionVersion	Fertigungsversion	xs:string(255)		
PlannerGroupInCharge	Verantwortliche Planergruppe/Abteilung	xs:string(255)		
Description	Beschreibung	ct:MultilanguageTextType		
ScheduledStartDate	Terminierter Starttermin	xs:dateTime		
ScheduledEndDate	Terminierter Endtermin	xs:dateTime		
MaterialCharacteristics	Sequenz von Sachmerkmalen des produzierten Materials	sequence of ord:MaterialCharacteristicType		
UserDataFields	Benutzer-Datenfelder	ct:UserDataTypes		
Operations	Sequenz von Arbeitsvorgängen	sequence of ord:OperationType	x	
Material	Produziertes Material	ct:MaterialType	x	

Tabelle 27: Attribute im Datensatz-Auftragskopf

10.2.2 Sachmerkmale des Materials

Für die Definition und Übertragung des Datensatzes „Sachmerkmale des Materials“ steht der komplexe Datentyp *ord:MaterialCharacteristicType* zur Verfügung. Dieser bietet folgende Attribute:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
orderItem	Auftragsposition	xs:string(255)	x	
classType	Bezeichnung Klassenart	xs:string(255)	x	
characteristicName	Merkmalname	xs:string(255)	x	
characteristicValue1	Merkmalwert 1	xs:string(255)		
characteristicValue2	Merkmalwert 2	xs:string(255)		
characteristicValue3	Merkmalwert 3	xs:string(255)		
characteristicValue4	Merkmalwert 4	xs:string(255)		
characteristicValue5	Merkmalwert 5	xs:string(255)		
characteristicValue6	Merkmalwert 6	xs:string(255)		
characteristicValue7	Merkmalwert 7	xs:string(255)		
characteristicValue8	Merkmalwert 8	xs:string(255)		
characteristicValue9	Merkmalwert 9	xs:string(255)		
characteristicValue10	Merkmalwert 10	xs:string(255)		
characteristicValue11	Merkmalwert 11	xs:string(255)		
characteristicValue12	Merkmalwert 12	xs:string(255)		
characteristicValue13	Merkmalwert 13	xs:string(255)		

Tabelle 28: Attribute im Datensatz Sachmerkmale des Materials

10.2.3 Arbeitsvorgänge

Die zum Auftrag gehörenden Arbeitsvorgänge (AVOs) werden als Sequenz einzelner Arbeitsvorgänge übertragen.

Die wichtigsten Bestandteile eines AVOs sind:

- *ERP-Schlüssel, Vorgangsnummer und Langtext* zur Identifikation des AVOs
- *Arbeitsplatz oder Arbeitsplatzgruppe* für die Fertigung
- *Vorgangssplit* zur Definition der Zuweisung auf einen Arbeitsplatz der Arbeitsplatzgruppe (Kapazitätsgruppe in der ERP-Hierarchie) beim Auto-Split-Verfahren
- *Rückmeldenummer* als Identifier für das ERP-System, um eine spätere Rückmeldung wieder dem Auftrag bzw. AVO zuzuordnen
- die *Sollmenge* mit Mengeneinheit als Mengenvorgabe des zu produzierenden Materials
- Geplanter *Starttermin* und *Endtermin* als Zeitvorgaben für die Fertigung des Materials
- *Rüstzeit* und *Stückzeit* mit Zeiteinheit als weitere Zeitvorgaben für die Fertigung des Materials
- *Stückzeitfaktor* zur Berechnung einer *Stückzeit pro Mengeneinheit*
- *Komponentenliste* für die Definition der benötigten Komponenten
- *Fertigungshilfsmittel* für die Definition der benötigten Hilfsmittel (z. B. Werkzeuge)

Über den Stückzeitfaktor kann die Stückzeit pro Mengeneinheit der Maschinenteknologie angepasst werden:

Stückzeit pro Mengeneinheit = Stückzeit / Stückzeitfaktor

Für die Definition und Übertragung eines AVOs steht der komplexe Datentyp *ord:OperationType* zur Verfügung. Dieser bietet folgende Attribute:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
action	Datensatz-Aktion	ct:actionType	x	x
OrderERPKey	ERP-Schlüssel	ct:ERPKeyType	x	x
OperationNumber	Vorgangsnummer	xs:string(32)	x	x
OperationSplit	Vorgangssplit <ul style="list-style-type: none"> — 0: ohne Vorgangssplit (Pull-Verfahren) — 1: Zuweisung zum 1. Arbeitsplatz der Arbeitsplatzgruppe/Kapazitätsgruppe — 2: Zuweisung zum 2. Arbeitsplatz der Arbeitsplatzgruppe — 3: usw. 	xs:string(8)	x	x

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
ConfirmationNumber	Rückmeldenummer	xs:string(255)	x	x
ERPYieldQuantity	Gemeldete Gutmenge	xs:double		
ERPReworkQuantity	Gemeldete Nacharbeitsmenge	xs:double		
ERPScrapQuantity	Gemeldete Ausschussmenge	xs:double		
ERPPlannedScrapQuantity	Geplante Ausschussmenge	xs:double		
ERPStatusIds	Status-IDs im ERP-System (mit Leerzeichen als Trennungszeichen)	xs:string(255)		x
ERPStatusCodes	Status-Codes im ERP-System (mit Leerzeichen als Trennungszeichen)	xs:string(255)		
ERPStatusCodesTranslation	Übersetzung für Status-Codes	xs:string(255)		
Description	Beschreibung	ct:MultilanguageTextType		
OperationText	Langtext	ct:MultilanguageTextType	x	
ControlKey	Steuerschlüssel	xs:string(255)		
OrderType	Auftragsart	xs:string(255)		
FunctionType	Funktion des AVOs <ul style="list-style-type: none"> — 1: Normal — 2: Training — 3: Wartung 	xs:long		
LeadingOperation	Führungsvorgang innerhalb Vorgangsblock (ja/nein) <ul style="list-style-type: none"> — true — false 	xs:boolean		
BlockNumber	Vorgangsblock	xs:string(255)		
BlockDescription	Beschreibung Vorgangsblock	ct:MultilanguageTextType		
OperationSequence	Auftragsreihenfolge	xs:string(255)		
AutoStartId	ID, die von der Maschine kommt, um den AVO automatisch zu starten	xs:string		
TargetWorkplace	Arbeitsplatz für die Fertigung	xs:string(255)	x	

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
TargetWorkGroup	Arbeitsplatzgruppen-ID des Arbeitsplatzes	xs:string		
WorkplaceGroup	Arbeitsplatzgruppe des Arbeitsplatzes	xs:string(255)		
TargetStartDateTime	Geplanter Starttermin	xs:dateTime	x	
TargetEndDateTime	Geplanter Endtermin	xs:dateTime	x	
DefaultStrokeFactor	Hubfaktor (Produzierte Menge pro Hub bzw. pro Takt/Produktionszyklus)	xs:double		
DefaultPieceTimeFactor	Stückzeitfaktor (Dient zur Berechnung der Stückzeit pro Mengeneinheit)	xs:double	x	
DisplayQuantityUnit	Mengeneinheit	xs:string(255)	x	
TargetQuantity	Sollmenge	xs:double	x	
StandardValue1	Soll-Rüstzeit	xs:string(255)	x	
StandardUnit1	Soll-Rüstzeit Einheit	ct:timeUnitType	x	
StandardValue2	Stückzeit	xs:string(255)	x	
StandardUnit2	Stückzeit Einheit	ct:timeUnitType	x	
StandardValue3	Standardwert 3	xs:string(255)		
StandardUnit3	Standardwert 3 Einheit	ct:timeUnitType		
StandardValue4	Standardwert 4	xs:string(255)		
StandardUnit4	Standardwert 4 Einheit	ct:timeUnitType		
StandardValue5	Standardwert 5	xs:string(255)		
StandardUnit5	Standardwert 5 Einheit	ct:timeUnitType		
StandardValue6	Standardwert 6	xs:string(255)		
StandardUnit6	Standardwert 6 Einheit	ct:timeUnitType		
AlternateOperation Number	Alternative Vorgangsnummer	xs:string(255)		
ProductionVersion	Fertigungsversion	xs:string(255)		

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
CounterNumber	Zählernummer (Presstationszähler)	xs:long		
UserStatus	Anwenderstatus im ERP-System	xs:string(255)		
OverdeliveryCheck	Prüfung auf Einhaltung der Überlieferungstoleranz	xs:boolean		
OverdeliveryQuantity	Überlieferungstoleranz (Beschränkung in Prozent für die gebuchte Menge oberhalb der Sollmenge)	xs:double		
UnderdeliveryCheck	Prüfung auf Einhaltung der Unterlieferungstoleranz	xs:boolean		
UnderdeliveryQuantity	Unterlieferungstoleranz (Beschränkung in Prozent für die gebuchte Menge unterhalb der Sollmenge)	xs:double		
DefaultTransportQuantity	Voreingestellte Menge für Materialbewegungen in Track & Trace	xs:double		
QualityCheckCycle	Produzierte Menge, nach der eine Qualitätsprüfung durchgeführt werden soll	xs:double		
StandardPackSize	Maximale Menge, die in den Ausgangsbehälter produziert werden kann (Track & Trace)	xs:double		
DefaultComponentConsumptionFactor	Vordefinierte Verbrauchsmenge einer Komponente pro produzierter Materialeinheit (falls diese Komponente nicht in der Komponentenliste enthalten ist)	xs:double		
EarliestStartDate	Frühester Starttermin	xs:dateTime		
LatestStartDate	Spätester Starttermin	xs:dateTime		
EarliestEndDate	Frühester Endtermin	xs:dateTime		
LatestEndDate	Spätester Endtermin	xs:dateTime		
ScheduledStartDate	Terminierter Starttermin	xs:dateTime		
ScheduledEndDate	Terminierte Endtermin	xs:dateTime		
TargetQueueTime	Soll-Wartezeit (Vorgabedauer der Wartezeit in ms)	xs:long		
TargetQueueTimeUnit	Soll-Wartezeit Einheit	ct:timeUnitType		

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
TargetProcessingTime	Soll-Bearbeitungszeit (Vorgabedauer der Bearbeitungszeit in ms, wenn Bearbeitungszeit nicht von der Stückzahl abhängt) Bsp: 1 Brötchen zu backen dauert 15 Minuten. Wenn man 3 Brötchen gleichzeitig backt, dauert es nicht länger.	xs:long		
TargetProcessingTimeUnit	Soll-Bearbeitungszeit Einheit	ct:timeUnitType		
TargetTeardownTime	Soll-Abrüstzeit (Vorgabedauer der Abrüstzeit in ms)	xs:long		
TargetTeardownTimeUnit	Soll-Abrüstzeit Einheit	ct:timeUnitType		
TargetWaitTime	Soll-Liegezeit (Vorgabedauer der Liegezeit)	xs:long		
TargetWaitTimeUnit	Soll-Liegezeit Einheit	ct:timeUnitType		
TargetMoveTime	Soll-Transportzeit	xs:long		
TargetMoveTimeUnit	Soll-Transportzeit Einheit	ct:timeUnitType		
MinimumSendAheadQuantity	Mindestweitergabemenge (Nach Produktion dieser Menge darf Nachfolgevorgang bereits starten)	xs:double		
MinimumOverlapTime	Mindestüberlappungszeit (bei Sollüberlappungen) Der Nachfolgevorgang soll bereits vor Ende des Vorgängers beginnen. Die Überlappung muss mindestens diese vorgegebene Dauer haben.	xs:long		
MinimumOverlapTimeUnit	Mindestüberlappungszeit Einheit	ct:timeUnitType		
UserFields	Benutzer-Datenfelder	ct:UserDataTypes		
OperationComponents	Verwendete Komponenten bzw. Komponentenliste (siehe Kapitel 0)	sequence of ord:OperationComponentType		
ProductionResourceTools	Verwendete Fertigungshilfsmittel (siehe Kapitel 0)	sequence of ord:ProductionResourceToolType		

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
Material	Produziertes Material	ct:MaterialType	x	
CustomerEnhancements	Benutzerfelder	sequence of ord:PartialOrderQuantityDataSection (*)		

Tabelle 29: Attribute im Datensatz Arbeitsvorgänge

(*) Attribute im Datentyp *ord:PartialOrderQuantityDataSection*:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
PartialOrderQuantityDataSection	Benutzerfelder	sequence of ord:PartialOrderQuantityDataType (**)		

Tabelle 30: Attribute im Datentyp *ord:PartialOrderQuantityDataSection*(**) Attribute im Datentyp *ord:PartialOrderQuantityDataType*:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
UF1	Benutzerfeld	xs:string(255)		
...				
UTF50	Benutzerfeld	xs:string(255)		

Tabelle 31: Attribute im Datentyp *ord:PartialOrderQuantityDataType*

10.2.4 Komponenten

Die für einen AVO benötigten Komponenten werden als Sequenz einzelner Komponenten übermittelt und bildet damit die Komponentenliste (auch Stückliste genannt).

Zur Identifikation einer Komponente dient die *Komponentennummer* und *Positionsnummer*. Die Positionsnummer dient der Reservierung oder Festlegung des Sekundärbedarfs.

Für die Definition und Übertragung einer Komponente steht der komplexe Datentyp *ord:OperationComponentType* zur Verfügung. Dieser bietet folgende Attribute:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
action	Datensatz-Aktion	ct:actionType	x	x
ComponentNumber	Komponentennummer	xs:string(255)	x	x
PositionNumber	Positionsnummer der Reservierung / des Sekundärbedarfs	xs:string(255)	x	x
BaseUnitOfMeasure	Basismengeneinheit	xs:string(255)		
BatchNumber	Chargennummer	xs:string(255)		
ConfirmationNumber	Rückmeldenummer des AVOs	xs:string(255)		
Description	Beschreibung	ct:MultilanguageTextType		
MaterialGroup	Warengruppe	xs:string(255)		
MaterialType	Materialtyp	xs:string(255)		
RequirementDate	Bedarfstermin	xs:dateTime		
RequirementQuantity	Bedarfsmenge	xs:double		
ReservationNumber	Nummer der Reservierung bzw. des Sekundärbedarfs	xs:string(255)		
StorageBin	Lagerplatz	xs:string(255)		
StorageLocation	Lagerort	xs:string(255)		
TraceNeeded	Legt fest, ob für die Entnahme dieser Komponente die Chargennummer vom Benutzer des SFTs eingegeben werden muss.	xs:boolean		
UserFields	Benutzer-Datenfelder	ct:UserDataTypes		

Tabelle 32: Attribute im Datensatz Komponente

10.2.5 Fertigungshilfsmittel

Die für einen AVO benötigten Fertigungshilfsmittel werden als Sequenz einzelner Fertigungshilfsmittel definiert.

Zur Identifikation eines einzelnen Fertigungshilfsmittels dient eine Nummer, Typkennung und optional eine Gruppe.

Für die Definition und Übertragung eines einzelnen Fertigungshilfsmittels steht der komplexe Datentyp *ord:ProductionResourceToolType* zur Verfügung. Dieser bietet folgende Attribute:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
Number	Nummer	xs:string(255)	x	x
Type	Typkennung	xs:string(255)	x	x
Sequence	Laufende Nummer	xs:string(255)		
Quantity	Menge	xs:double		
Unit	Mengeneinheit	xs:string(255)		
Group	Gruppe	xs:string(255)		
UserFields	Benutzer-Datenfelder	ct:UserDataTypes		
Description	Beschreibung	ct:MultilanguageTextType		

Tabelle 33: Attribute im Datensatz Fertigungshilfsmittel

10.3 Schichtdaten

Schichtdaten werden immer als Gesamtversorgung übertragen. Das bedeutet die Daten aller Schichten werden zusammen in einem Datensatz übermittelt. Alle vorhandenen Schichten werden gelöscht bis auf die aktuelle Schicht. Diese wird nicht verändert.

Schichten können auch komplett in MES FLEX gepflegt werden.

Jedes übermittelbare Attribut hat einen entsprechenden Eintrag in der eingesetzten Datenbank in MES FLEX. Einige Attribute sind verpflichtend anzugeben, je nachdem ob es sich bei der Datensatz-Aktion um einen Neueintrag, eine Änderung oder ein Löschen handelt. Dies ist in den nachfolgenden Tabellen in den Spalten rechts ersichtlich (Pflichtfeld für Aktion).

Die Schichten werden als Sequenz einzelner Schicht-Definitionen zusammengefasst und übertragen.

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
action	Datensatz-Aktion	ct:actionType	x	x
ShiftsType	Schichten	sequence of shift:ShiftType	x	x

Tabelle 34: Attribute im Datensatz Schichten

Die wichtigsten Bestandteile einer Schicht-Definition sind:

- der *ERP-Schlüssel* zur Identifikation
- der *Arbeitsplatz*, für den die Schicht-Definition gelten soll
- der *Schichttyp* zur Festlegung ob Frei-, Früh-, Spät-, Nachschicht oder manuelle Schicht
- der *Schichttag* zur Festlegung des Wochentages
- der *Starttermin* und *Endtermin* zu Festlegung des Zeitintervalls (Uhrzeiten) der Schicht

Manuelle Schichten können im Gegensatz zu den anderen Schichttypen über einen speziellen Dialog im Shopfloor Terminal (SFT) angelegt werden.

Über diesen Schichttyp können z. B. Werker vor Ort kurzfristig ungeplante Tätigkeiten bzw. AVOs einer Schicht zuordnen. Dies erspart eine spätere Schicht-Korrektur.

Zur Definition einer einzelnen Schicht dient der komplexe Datentyp *shift:ShiftType* mit folgenden Attributen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
ERPKey	ERP-Schlüssel	ct:ERPKeyType	x	x
Workplace	Arbeitsplatz	xs:string(255)	x	x
ShiftTypeCode	Schichttyp 1-10: Freischicht 11: Frühschicht 12: Spätschicht 13: Nachtschicht 14-15: Manuell	xs:long	x	x
ShiftDate	Schichttag	xs:date	x	x
StartDateTime	Starttermin	xs:dateTime	x	x
EndDateTime	Endtermin	xs:dateTime	x	x
Breaks	Schichtpausen	sequence of shift:ShiftBreak Type (*)		

Tabelle 35: Attribute im komplexen Datentyp *shift:ShiftType*


Über den komplexen Datentyp *shift:ShiftBreakType* können zusätzlich eine oder mehrere Zeitintervalle für Pausen innerhalb der Schicht definiert werden.

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
StartDateTime	Starttermin	xs:dateTime	x	x
EndDateTime	Endtermin	xs:dateTime	x	x

Tabelle 36: Attribute im komplexen Datentyp *shift:ShiftBreakType*

10.4 Personaldaten

Personaldaten werden immer als Gesamtversorgung übermittelt. Das bedeutet die Daten des kompletten an der Produktion beteiligten Personalstamms werden übermittelt. So werden auch Mitarbeiter gelöscht, die nicht mehr in der Gesamtversorgung beinhaltet sind.

-  Der Personalstamm kann auch komplett in MES FLEX gepflegt werden. Dabei können auch abweichende bzw. abstrakte Personennamen oder Rollen (z. B. „Bediener01“) festgelegt werden. Durch Setzen eines entsprechenden Attributs **Nicht löschen bei Neuversorgung** in MES FLEX kann ein Überschreiben bei einer Neuversorgung im Download verhindert werden.

Die Personaldaten werden als Sequenz aller Mitarbeiter des Personalstamms zusammengefasst und übertragen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
action	Datensatz-Aktion	ct:actionType	x	x
PersonsType	Personen	sequence of pers:PersonType	x	x

Tabelle 37: Attribute im Datensatz Personaldaten

Die wichtigsten Daten eines Mitarbeiters sind:

- *ERP-Schlüssel, Personalnummer, Ausweisnummer* und *Name* zur Identifikation des Mitarbeiters
- *Arbeitsplatz*, für den die Schicht-Definition gelten soll
- *Schichttyp* zur Festlegung ob Frei-, Früh-, Spät-, Nachschicht oder manuelle Schicht
- *Schichttag* zur Festlegung des Wochentages

Zur Definition eines Mitarbeiters dient der komplexe Datentyp *pers:PersonType* mit folgenden Attributen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
PersonERPKey	ERP-Schlüssel	pers:PersonERPKeyType		
PersonnelNumber	Personalnummer	xs:string(255)	x	x
IdentificationNumber	Ausweisnummer	xs:string(255)	x	x
FirstName	Vorname	xs:string(255)	x	x
LastName	Nachname	xs:string(255)	x	x
CostCenter	Kostenstelle	xs:string(255)		

Tabelle 38: Attribute im komplexen Datentyp *pers:PersonType*

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
Client	Mandant	xs:string(32)	x	x
CompanyCode	Buchungskreis	xs:string(32)	x	x
PersonnelArea	Personalbereich	xs:string(255)	x	x
SystemId	System-ID	xs:string(32)		

Tabelle 39: Attribute im komplexen Datentyp *pers:PersonERPKeyType*

Die *System-ID* dient der eindeutigen Identifikation des ERP-Schlüssels im System. Das ist wichtig, falls die Attribute Mandant, Buchungskreis und Personalbereich nicht eindeutig sind.

10.5 Daten zu Gemeinkosten

MES FLEX bietet die Funktion, bei der Bearbeitung eines Auftrags anfallende Gemeinkosten über sogenannte Gemeinkostensammler zu akkumulieren und über den Upload-Datendienst an das ERP-System rückzumelden. Eine Rückmeldung ist bisher allerdings nur für SAP ERP-Systeme bzw. im IDoc-Format möglich.

Die Definitionen der Gemeinkostensammler bzw. die Gemeinkostensammler-Stammdaten können automatisiert als Gesamtversorgung vom ERP-System an MES FLEX übertragen werden.

Für die Zuordnung eines Gemeinkostensammlers zu einem Kostenrechnungskreis dient der komplexe Datentyp *oc:OverheadCostERPKeyType* als eigener ERP Schlüssel mit folgenden Attributen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
Client	Mandant	xs:string	x	x
CompanyCode	Buchungskreis	xs:string	x	x
ControllingArea	Kostenrechnungsbereich	xs:string	x	x
SystemId	System-ID	xs:string	x	x

Tabelle 40: Attribute im komplexen Datentyp *oc:OverheadCostERPKeyType*

Der Datensatz zur Definition der eingesetzten Gemeinkostensammler wird als Sequenz einzelner Gemeinkostensammler-Definitionen zusammengefasst und vom ERP-System an MES FLEX übertragen.

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
action	Datensatz-Aktion	ct:actionType	x	x
OverheadCost	Gemeinkosten	sequence of oc:OverheadCostsCollectorType	x	x

Tabelle 41: Attribute im Datensatz Gemeinkosten

Zur Definition eines einzelnen Gemeinkostensammlers dient der komplexe Datentyp *oc:OverheadCostsCollectorType* mit folgenden Attributen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
OverheadCostsERPKey	ERP-Schlüssel	oc:OverheadCostERPKeyType	x	x
Plant	Werk	xs:string		
OverheadCostsCollectorType	Kostensammler-Typ: OVERHEAD_COST_CENTER (für Gemeinkostensammler) INTERNAL_ORDER (für Innenaufträge)	xs:string	x	x
CostUnit	Kostenstelle (bei Gemeinkostensammler)	xs:string		

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/ Ändern	Löschen
OrderNumber	Auftragsnummer	xs:string		
OrderType	Auftragstyp	xs:string		
Abbreviation	Abkürzung	ct:MultilanguageTextType		
Description	Beschreibung	ct:MultilanguageTextType		
ResponsiblePerson	Verantwortliche Person	xs:string		
Color	Farbcode für Visualisierung (RGB-Wert)	xs:string		
ValidFrom	Startdatum der Gültigkeit	xs:dateTime		
ValidTo	Enddatum der Gültigkeit	xs:dateTime		
ExecutionCostUnitChangeable	Bezieht sich auf die Senderkostenstelle (zu entlastende Kostenstelle) Konfiguration für die Eingabe/Änderung Definiert, ob die Senderkostenstelle im SFT eingegeben und verändert werden darf	xs:boolean		
ReceiverCostUnitChangeable	Bezieht sich auf die Verursacherkostenstelle (zu belastende Kostenstelle) Direkt für die innerbetriebliche Leistungsverbuchung Definiert, ob die Verursacherkostenstelle im SFT eingegeben und verändert werden darf	xs:boolean		
ERPAutomaticBooking	Verbuchungstyp der rückgemeldeten Gemeinkosten im ERP-System: AUTO: Die Zuweisung bzw. Verbuchung der Kosten im ERP-System erfolgt automatisiert. MANUAL: Die Kosten werden zunächst in eine eigene Tabelle geschrieben. Die	xs:string		

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
	Zuweisung bzw. Verbuchung der Kosten im ERP-System erfolgt später manuell.			
ERPReportingEnabled	Aktivierung der Rückmeldung angefallener Gemeinkosten	xs:boolean		
UserStatus	Benutzerstatus	xs:string		

Tabelle 42: Attribute im komplexen Datentyp *oc:OverheadCostsCollectorType*

10.6 Lagerdaten aus Lagerverwaltungssystem

ERP-Systeme mit einer vorhandenen Lagerverwaltung können zusätzlich Daten zum Lagerbestand von Materialien übermitteln.

Die kompletten Lagerdaten werden dann als Sequenz von Lagerständen einzelner Materialien zusammengefasst und übertragen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
StocksType	Lagerdaten	sequence of sto:StockType	x	x

Tabelle 43: Attribute im Datensatz Lagerdaten

Zur Definition eines Lagerstands zu einem bestimmten Material dient der komplexe Datentyp *sto:StockType* mit folgenden Attributen:

XML-Attribut			Pflichtfeld für Aktion	
Name	Beschreibung	Datentyp (Feldlänge)	Einfügen/Ändern	Löschen
Material	Material	ct:MaterialType	x	x
Quantity	Menge	xs:double	x	x
QuantityUnit	Mengeneinheit	xs:string(3)	x	x
StockLocation	Lagerort	xs:string(10)	x	x

Tabelle 44: Attribute im komplexen Datentyp *sto:StockType*

10.7 Antwort auf Mengenanfragen

Zusätzlich zu den genannten Datensätzen gibt es noch Datensätze mit Mengenangaben, die als Antwort dienen zu Anfragen von MES FLEX an das ERP-System bzgl. folgender Inhalte:

- Anfrage zu Materialeinheiten
- Anfrage zu Verpackungseinheiten
- Anfrage zu Restmengen im Lagerverwaltungssystem (nach Abschluss von AVOs)

11 Upload

Über den Upload-Datendienst werden Ereignisse bzw. Ereignissequenzen und zugehörige Datensätze in Form von Upload-Nachrichten von MES FLEX an das ERP übermittelt.

In MES FLEX selbst wird durch Trigger-Ereignisse wie z. B. ein Wechsel der Arbeitsvorgangs-Phasen (z. B. *Rüsten*, *Produktion* oder *Unterbrochen*) oder Mengenbuchungen eine Nachricht im Event Message Stream erzeugt und mit entsprechenden Ereignisdaten angereichert. Über Anpassungen in der Rule Engine können Ereignisse mit kundenspezifischen Berechnungsdaten („Kennwerte“) angereichert werden.

Folgende Ereignisse werden über den Upload-Datendienst an das ERP übermittelt:

- Wechsel einer Arbeitsvorgangsphase (AVO-Phase)
- Meldungen über AVO-Dauer bzw. Arbeitszeit
 - Werden in MES FLEX nur bei Fertigstellung eines AVOs oder bei Schichtende (AVO läuft Schicht-übergreifend) generiert
 - Pro Leistungsart: Maschinenproduktionszeit, Maschinenrüstzeit
 - Bei Personalzeit: Pro Person und Leistungsart
- Mengenmeldungen
 - Asynchron oder Einzel & || gesteuert nach Intervallen
 - Pro Mengentyp eine separate Meldung
 - Gutmenge
 - Ausschussmenge
 - Nacharbeitsmenge
- Korrekturen
 - Delete / Insert
 - Restriktionen sind im Detail zu spezifizieren und müssen durch das ERP unterstützt werden
 - Nur möglich mit eindeutiger Meldungsnummer
- Kennzahlen
 - Sind nach Schnittstellenausprägung ERP zu definieren
 - Z. B. Ist-Stückzeit Berechnung → Rule Engine Anwendung
- Ad-hoc-Wartung über das Shopfloor Terminal (SFT)
 - Ad-hoc-Wartung anfordern (Request Maintenance)
 - Ad-hoc-Wartung-Start (Start Maintenance)
 - Ad-hoc-Wartung beendet (Finish Maintenance)

Die Datenstrukturen dieser Ereignisse werden als DTOs (Data Transfer Objects) angelegt und nach dem objektorientierten Vererbungsprinzip aufgebaut. Jedes Ereignis enthält einen generischen Standardinhalt der Basisdatenstruktur und entsprechend weitere Vererbungen sowie einen eigenen für das Ereignis spezifischen Inhalt.

Die DTOs werden dann von MES FLEX in übertragbare XML-Dokumente umgewandelt.

11.1 Upload Basis-Datentypen

Für die zu übermittelnden Nachrichten stehen zwei Basis-Datentypen zur Verfügung.

11.1.1 ERP-Schlüssel

Zur Identifikation von ERP-Ressourcen wie Arbeitsplätze, Vorgänge oder Personal steht der Datensatz *ERPContext* vom Datentyp *ERPContextType* zur Verfügung.

Dieser bietet folgende Attribute:

ERPContextType			
Attribut	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
UUID	Global eindeutige ID der ERP-Ressource	xs:string(36)	x
Identifizier1	Mandant	xs:string(255)	x
Identifizier2	Buchungskreis	xs:string(255)	x
Identifizier3	Werk	xs:string(255)	x
SystemNumber	System-ID	xs:string(255)	

Tabelle 45: Attribute im Datensatz ERP-Schlüssel

11.1.2 Nachrichten an ERP-System

Für Nachrichten an das ERP-System wird allgemein der komplexe Datentyp *ERPUploadMessage* verwendet. Dieser bietet folgende Attribute:

ERPUploadMessage			
Attribut	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
ConfirmationNumber	Rückmeldenummer	DefaultStringType	x
MessageIdentifier	Eindeutige Nachrichten-ID	xs:string(36)	x
MessageType	Nachrichtentyp: — REGULAR (regulär) — CORRECTION (Korrektur)	xs:string	x
MessageTimestamp	Zeitstempel	xs:dateTime	x
ActionTimestamp	Zeitstempel (bei Korrekturen der Zeitpunkt der Korrektur)	xs:dateTime	x
CustomMessageData	Sequenz von kundenspezifischen Daten (verkettete Paare aus Key mit Value)	sequence of xs:string	

ERPUploadMessage			
Attribut	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OperationMessage	Nachricht bzgl. Vorgänge	OperationMessageType (siehe Kapitel 11.2)	
OperationSchedule	Nachricht bzgl. AVO-Zeitplanung	OperationScheduleType (siehe Kapitel 11.3)	
WarehouseMessage	Nachricht bzgl. Lagerverwaltungssystem	WarehouseMessageType (siehe Kapitel 11.4)	

Tabelle 46: Attribute im Datensatz Nachricht an ERP-System

Je nach Nachrichtentyp kommt entweder die Datenstruktur *OperationMessage*, *OperationSchedule* oder *WarehouseMessage* zum Einsatz.

In den nachfolgenden Kapiteln wird der Einsatz dieser Datenstrukturen näher beschrieben.

11.2 Nachrichten zu Arbeitsvorgängen (AVOs)

In diesem Kapitel werden die Nachrichten zu AVO-Ereignissen erläutert. Diese können alternativ auch als Callback über die FORCE Bridge API (siehe Kapitel 5 „Callbacks und Ereignisse“) versendet werden. Nähere Informationen zum ERP-Callback finden Sie in der Onlinedokumentation in der Swagger UI. Für Nachrichten zu AVO-Ereignissen steht der komplexe Datentyp *OperationMessageType* zur Verfügung. Dieser bietet folgende Attribute:

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
WorkplaceIdentification	UUID	Global eindeutige ID	xs:string(36)	x
	ERPContext	ERP-Schlüssel	ERPContextType	x
	ERPWorkplaceNumber	Arbeitsplatznummer im ERP	xs:string(255)	x
	WorkplaceNumber	Arbeitsplatz	xs:string(255)	x
OperationIdentification	UUID	Global eindeutige ID	xs:string(36)	x
	ERPContext	ERP-Schlüssel	ERPContextType	x
	OrderNumber	Auftragsnummer	xs:string(255)	x
	OperationNumber	Vorgangsnummer	xs:string(255)	x
	OrderSplit	Auftragssplit	xs:string(255)	x
	OperationSplit	Vorgangssplit	xs:string(255)	x
	OrderProductionVersion	Fertigungsversion	xs:string(255)	
PersonnelIdentification	OperationSequence	Reihenfolge des AVOs im Auftrag	xs:string(255)	
	UUID	Global eindeutige ID	xs:string(36)	x
	ERPContext	ERP-Schlüssel	ERPContextType	x
ShiftIdentification	PersonnelNumber	Personalnummer	xs:string(255)	x ³
	UUID	Global eindeutige ID	xs:string(36)	x
	ShiftDay	Datum des Schicht-Ereignisses (in UTC)	xs:date	x

³ kein Pflichtfeld bei Phasenwechsel in *Rüsten*, *Produktion* oder *Unterbrochen*

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
	ShiftCode	Schichtkennung	xs:string(255)	x

Tabelle 47: Attribute im Datensatz AVO-Ereignis

Weitere Attribute im Datentyps *OperationMessageType* stehen je nach Art der AVO-Meldung zu Verfügung.

11.2.1 Wechsel einer AVO-Phase

Für Nachrichten zu AVO-Ereignissen in Form eines Wechsels der AVO-Phase stehen folgende weitere Attribute im Datentyp *OperationMessageType* zur Verfügung:

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OperationPhase		Wechsel der AVO-Phase in: <ul style="list-style-type: none"> — SETUP (Rüsten) — PROCESSING (Produktion) — INTERRUPTED (Unterbrochen) — COMPLETED (Fertiggestellt) — CLOSED (Abgeschlossen) 	xs:string	x

Tabelle 48: Attribute im Datensatz Wechsel AVO-Phase

11.2.2 AVO -Mengenbuchungen und Begründungen

Für Nachrichten zu AVO-Meldungen in Form von Mengenbuchungen und Begründungen stehen im Datentyp *OperationMessageType* folgende weitere Attribute zur Verfügung:

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OperationQuantity	MaterialNumber	Nummer des im AVO produzierten Materials laut ERP-System	xs:string(255)	x
	RelativeQuantity	Durchgeführte Begründung als Sequenz aus: <ul style="list-style-type: none"> — Qualitätsdetail — Qualitätsdetail laut ERP (optional) 	sequence of <ul style="list-style-type: none"> — xs:string(255) — xs:string(255) 	x

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
	AbsoluteQuantity	<p>Durchgeführte Mengenbuchung als Sequenz aus</p> <ul style="list-style-type: none"> — Qualität <ul style="list-style-type: none"> ○ YIELD (Gutmenge) ○ SCRAP (Ausschuss) (optional) ○ REWORK (Nacharbeit) (optional) — Menge — Mengeneinheit <p>Jeder Qualitäts-Typ darf in dieser Nachricht nur einmalig gebucht werden.</p>	<p>sequence of</p> <ul style="list-style-type: none"> — xs:string(255) — xs:double — xs:string(255) 	x

Tabelle 49: Attribute im Datensatz Mengenbuchungen und Begründungen

11.2.3 AVO-Dauer

Für Nachrichten zu AVO-Meldungen in Form von AVO-Dauern stehen im Datentyp *OperationMessageType* folgende weitere Attribute zur Verfügung:

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OperationDuration	ActivityType	Zu buchende Leistungsart (z. B. Maschinenzeit)	xs:string(255)	x
	ActivityRelativeDuration	Zu buchende Dauer (relative Zeit)	xs:duration	x
	ActivityAbsoluteDuration	Zu buchende Dauer (absolute Zeit)	xs:duration	x

Tabelle 50: Attribute im Datensatz AVO-Dauer

11.2.4 Komponenten Mengen

Für Nachrichten zu AVO-Meldungen in Form von Mengenbuchungen zu Komponenten stehen im Datentyp *OperationMessageType* folgende weitere Attribute zur Verfügung:

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OperationComponent	ComponentNumber	Komponentennummer	xs:string(255)	x
	RelativeQuantity	<p>Durchgeführte Begründung als Sequenz aus</p> <ul style="list-style-type: none"> — Qualitätsdetail 	<p>sequence of</p> <ul style="list-style-type: none"> — xs:string(255) 	x

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
		<ul style="list-style-type: none"> Qualitätsdetail laut ERP (optional) 	<ul style="list-style-type: none"> xs:string(255) 	
	AbsoluteQuantity	<p>Durchgeführte Mengenbuchung als Sequenz aus</p> <ul style="list-style-type: none"> Qualität <ul style="list-style-type: none"> YIELD (Gutmenge) SCRAP (Ausschuss) (optional) REWORK (Nacharbeit) (optional) Menge Mengeneinheit <p>Jeder Qualitäts-Typ darf in dieser Nachricht nur einmalig gebucht werden.</p>	<p>sequence of</p> <ul style="list-style-type: none"> xs:string(255) xs:double xs:string(255) 	x

Tabelle 51: Attribute im Datensatz Komponenten Mengen

11.2.5 Zusammenbau von Komponenten

Für Nachrichten zu AVO-Ereignissen beim Zusammenbau von Komponenten stehen im Datentyp *OperationMessageType* folgende weitere Attribute zur Verfügung:

OperationMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OperationAsBuilt	ComponentNumber	Komponentennummer	xs:string(255)	x
	ComponentPosition	Positionsnummer	xs:string(255)	x
	SerialNumberOutput	Seriennummer des produzierten Materials	xs:string(255)	x
	ContainerNumberOutput	Ausgangsbehälter	xs:string(255)	
	AssemblyType	<p>Assembly-Typ:</p> <ul style="list-style-type: none"> SERIAL BATCH (Charge) 	xs:string	x
	SerializationType	<p>Serialisierungstyp:</p> <ul style="list-style-type: none"> INPUT (assemble) DELETE (disassemble) UPDATE (exchange) 	xs:string	x

Tabelle 52: Attribute im Datensatz Zusammenbau von Komponenten

11.3 AVO-Zeitplanung

Für die Nachrichten bezüglich einer geänderten Zeitplanung (Neuterminierung) eines AVOs stehen folgende weitere Attribute zur Verfügung:

OperationScheduleType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OperationIdentification	UUID	Global eindeutige ID	xs:string(36)	x
	ERPContext	ERP-Schlüssel	ERPContextType	x
	OrderNumber	Auftragsnummer	xs:string(255)	x
	OrderSplit	Auftragssplit	xs:string(255)	x
	OperationSplit	Vorgangssplit	xs:string(255)	x
	OrderProductionVersion	Fertigungsversion	xs:string(255)	
	OperationSequence	Reihenfolge des AVOs im Auftrag	xs:string(255)	
TargetResource	UUID	Global eindeutige ID	xs:string(36)	x
	ERPContext	ERP-Schlüssel	ERPContextType	x
	ERPWorkplaceNumber	Arbeitsplatznummer im ERP	xs:string(255)	x
	WorkplaceNumber	Arbeitsplatz	xs:string(255)	
	WorkplaceType	Arbeitsplatz-Typ: SINGLE (Einzelarbeitsplatz) GROUP (Gruppenarbeitsplatz)	xs:string	x
TargetStart		Planstart des AVOs	xs:dateTime	x
TargetEnd		Planende des AVOs	xs:dateTime	x

Tabelle 53: Attribute im Datensatz AVO-Zeitplanung

11.4 Gemeinkosten-Meldungen

MES FLEX bietet die Funktion, bei der Bearbeitung eines Auftrags anfallende Gemeinkosten über sogenannte Gemeinkostensammler zu akkumulieren und über den Upload-Datendienst an das ERP-System rückzumelden.

Eine Rückmeldung ist bisher allerdings nur für SAP ERP-Systeme über die XSLT im IDoc-Format möglich. Weitere Informationen dazu im Handbuch zu **ERP Interface & Daten und Events via SAP IDoc**.

11.5 Lagerdaten an Lagerverwaltungssystem

Über den Upload-Datendienst können Lagerstände, die sich durch die laufende Produktion verändern, an das Lagerverwaltungssystem des ERP-Systems rückgemeldet werden. Lagerstände werden für Track & Trace Anwendungen (z. B. für Behältermengen) verwendet.

Für diese Art Nachrichten über veränderte Lagerstände bestimmter Materialien steht der komplexe Datentyp *WarehouseMessageType* zur Verfügung. Dieser bietet folgende Attribute:

WarehouseMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
OrderIdentification	UUID	Global eindeutige ID	xs:string(36)	x
	ERPContext	ERP-Schlüssel	ERPContextType	x
	OrderNumber	Auftragsnummer	xs:string(255)	x
	OrderSplit	Auftragssplit	xs:string(255)	x
	OrderProduction Version	Fertigungsversion	xs:string(255)	
OperationIdentification	UUID	Global eindeutige ID	xs:string(36)	x
	ERPContext	ERP-Schlüssel	ERPContextType	x
	OrderNumber	Auftragsnummer	xs:string(255)	x
	OrderSplit	Auftragssplit	xs:string(255)	x
	OperationSplit	Vorgangssplit	xs:string(255)	x
	OrderProduction Version	Fertigungsversion	xs:string(255)	
	OperationSequence	Reihenfolge des AVOs im Auftrag	xs:string(255)	
WarehouseMessageType		Lager-Typ im Lagerverwaltungssystem, auf den sich die Nachricht bezieht: <ul style="list-style-type: none"> — SERIAL (Rückmeldung des Lagerstands zu serialisiertem Einzelteil) — PRODUCTION_CONTAINER (Rückmeldung des Lagerstands eines Produktionsbehälters, der dem Lagerverwaltungssystem unbekannt ist) — INVENTORY_CONTAINER (Rückmeldung des Lagerstands zu einem bekannten Lagerbehälter) 	xs:string	
WarehouseEntity	ReferenceValue	Referenzwert (Tracking-Nummer eines Behälters oder führende Seriennummer eines Einzelteils)	xs:string(255)	
	MaterialNumber	Materialnummer	xs:string(255)	
	AbsoluteQuantity	Sequenz aus <ul style="list-style-type: none"> — Qualität <ul style="list-style-type: none"> ○ YIELD (Gutmenge) ○ SCRAP (Ausschuss) (optional) ○ REWORK (Nacharbeit) (optional) — Menge (Lagerstand) — Mengeneinheit 	sequence of <ul style="list-style-type: none"> — xs:string(255) — xs:double — xs:string(255) 	

WarehouseMessageType				
Attribut	Sub-Attribute	Beschreibung	Datentyp (Feldlänge)	Pflichtfeld
		Jeder Qualitäts-Typ darf in dieser Nachricht nur einmalig gebucht werden.		

Tabelle 54: Attribute im Datensatz Lagerdaten

Je nach *WarehouseMessageType* kommt entweder *OrderIdentification* (bei *PRODUCTION_CONTAINER*) oder *OperationIdentification* (bei *SERIAL*) zum Einsatz.

11.6 Mengenanfragen

Zusätzlich zu den genannten Datensätzen gibt es noch Datensätze bzgl. Mengenanfragen an das ERP-System bzgl. folgender Inhalte:

- Anfrage zu Materialeinheiten
- Anfrage zu Verpackungseinheiten
- Anfrage zu Restmengen im Lagerverwaltungssystem (nach Abschluss von AVOs)

11.7 Serialisierung und Korrekturfähigkeit

Eine eindeutige Serialisierung der übermittelten Daten ist notwendig, um Sequenzreihenfolgen und eindeutige Zuordnungen von Meldungen zu gewährleisten.

Korrekturfähigkeit kann nur sichergestellt werden, wenn jede einzelne Meldung mit einer überall eindeutigen Nachrichtennummer versehen ist und das übergeordnete ERP ebenfalls Stornierungsprozesse (Delete / Insert von Nachrichten) zulässt. Ein Korrekturprozess muss für jedes ERP vollständig spezifiziert werden.

12 Anhang

12.1 Abkürzungen und Begriffe

Abkürzung	Erklärung
OAuth	
API	Application Programming Interface – Programmierschnittstelle
AVO	
FORCE Bridge API	RESTful API zu MES FLEX, die es erlaubt, Daten REST-konform sowohl abzurufen als auch zu schreiben. Die FORCE Bridge API MDE (Maschinendatenerfassung) der MES FLEX kann mengenbasierte Stückzähler-Informationen von der Maschine sammeln, um die Berechnung der tatsächlich produzierten Menge zu berechnen.
CAS	
CAQ	Computer-Aided Quality – Computergestützte Qualitätssicherung
DTO	Data Transfer Objects – Datentransferobjekt
ERP	Enterprise Resource Planning – Warenwirtschaftssystem Beschreibt eine Softwarelösung zur Ressourcenplanung in einem Unternehmen. Ein ERP-System kann eine Datenquelle sein.
HAL	
Infor ION	Infor Intelligent Open Network. Infor ION ist eine Middleware-Cloud-Plattform, die den standardisierten Datenaustausch im XML-Format zwischen ERP-Systemen und MES FLEX ermöglicht.
IoT / IIoT	Internet of Things / Industrial Internet of Things
JSON	
KPI	Key Performance Indicator – Schlüsselkennzahl
MDE	Maschinendatenerfassung
RESTful	Representational State Transfer – Ein Programmierparadigma für verteilte Systeme. Die Idee dahinter ist, serverseitige Datenbankeinträge als Ressourcen anzubieten, die über einheitliche Befehlssätze erstellt, abgerufen, geändert oder gelöscht werden können.
RFC	Remote Function Call – Standardisiertes Interface bzw. Verfahren für die Kommunikation mit SAP-Systemen.
SFT	Shopfloor Terminal
Swagger UI	Ein Werkzeug zur Visualisierung und Interaktion mit Open-API-Ressourcen.
TDM	Tool Data Management – Werkzeugdatenmanagement
URI	Uniform Resource Identifier – einheitlicher Bezeichner für Ressourcen

	Ein Bezeichner bestehend aus einer Zeichenkette, die zur Identifizierung einer abstrakten oder physischen Ressource verwendet wird. RFC 3986
UUID	Universally-Unique-Identifier – Global eindeutige ID. Ist ein Standard für Bezeichner, die in der Softwareentwicklung verwendet werden.
XML	Extensible Markup Language – Auszeichnungssprache zur Darstellung hierarchisch strukturierter Daten
XSL	Extensible Stylesheet Language – Transformationssprachen zur Definition von Layouts für XML-Dokumente
XSD	XML Schema Definition – Standard zum Definieren von Datenstrukturen für XML-Dokumente auf Basis einer großen Anzahl von unterstützten Datentypen. Die Definition erfolgt über eine XSD-Datei, die in Form eines XML-Dokuments gehalten ist.
XSLT	XSL Transformation – Bestandteil der XSL. Dient der Umwandlung von XML-Dokumenten zwischen verschiedenen Systemen bzw. Standards. MES FLEX verwendet die XSLT zur Umwandlung zwischen SAP IDoc-Dokumenten und XML-Dokumente, die MES FLEX verarbeiten kann.