



TIBCO® Order Management

Web Services Guide

Version 6.1.0 | October 2024

Contents

Contents	2
Architecture	8
Services and Operations	8
Integration Channels	14
SOAP Over HTTP	14
SOAP Over JMS	15
REST over HTTP/HTTPS	17
TIBCO Fulfillment Orchestration Suite Overview	26
TIBCO Fulfillment Orchestration Suite Components	26
Fulfillment Management Services	28
Security Header	28
Order Service	28
Activate Order	28
Amend Order	30
Cancel Order	32
Get Order Details	33
Get Order Execution Plan	34
Handle Plan Item In Error	36
Perform Bulk Order Action	38
Submit Order	40
Suspend Order	42
Synchronous Submit Order	44
Withdraw Order	46
Purge Order	48
Automated Order Plan Development Service	50
Generate Plan	50

Generate Partial Plan	52
Order Management Services	53
Data Service	53
Catalog Service	58
TM Forum Adapter Service	59
Common Data Format Specifications	63
ResultStatus	63
ResultFault	64
Notification	65
Order	65
OrderHeader	67
OrderLine	71
ProductModel	75
Execution Plan	76
PlanFragment models	80
Authorization Service API Samples	81
Register tenant	81
Update tenant information	83
Get tenant information	83
Delete tenant	84
Create User	84
Update User	86
Get User	87
Delete User	88
Configuration Service API Samples	90
Actuator APIs	90
Actuator web endpoint 'refresh'	90
Get loggers name	91
Post loggers name	91

Actuator root web endpoint	92
Actuator web endpoint 'loggers'	92
Actuator web endpoint 'health'	92
Actuator web endpoint 'health-path'	93
Application Metadata APIs	93
Application Properties APIs	94
Configuration File APIs	96
Configuration Notification APIs	97
Archival Services API samples	98
Operation Get Order Summary	98
Operation Get Plans By Criteria	98
Operation Get Orders By Criteria	100
Operation Get Plan Item By Criteria	101
Operation Get Pie Chart Details By Criteria	102
Operation Get Order Details	103
Operation Get Plan Details	105
Operation Get Bulk Action	115
Operation Get All Bulk Action	115
Operation Get Audit Trail	115
Operation Submit Audit Trail	116
Operation Purge Order	117
Operation Get Saved Searches	118
Operation Update Saved Searches	118
Operation Saved Searches	119
Operation Delete Saved Searches	120
Orchestrator API samples	121
Feasibility Reply	121
Get Order Details	122
Get Order Status	123
Submit Order	123

Withdraw Order Request	126
Activate Order	127
Cancel Order	127
Suspend Order	128
Bulk Action	128
Plan Item Bulk Error Handler	129
Plan Item Error Handler	130
PlanItem Execute Response	130
Plan Item Suspend Response	131
Milestone Notify Request	132
PreQualification Failed Reply	133
Get Order Messages	133
Operation Get Order State Machine XML	134
Operation Purge Order	134
Operation GetOrderExecutionPlan	135
Operation SubmitOrderExecutionPlan	135
Operation submitPlanErrorNotification	139
Data Services API samples	140
Set PlanItem Request	140
Get PlanItems Details	142
Set Plan Request	143
Get Plan Details	145
Catalog Services API samples	147
Submit Single Action Model	147
Get Bulk Action Model	148
Submit Bulk Action Model	148
Delete Bulk Action Model	149
Get All Action Model	149
Submit Single Product Model	150

Submit Bulk Product Models	156
Get Bulk Product Model	162
Delete Bulk Product Model	162
Get All Product Model	163
Submit Single PlanFragment Model	163
Get Bulk PlanFragment Model	164
Submit Bulk PlanFragment Models	165
Delete Bulk PlanFragment Model	166
Get All PlanFragment Model	167
Process Component API sample	168
Operation Feasibility	168
Operation Plan Item Activate Request	171
Operation Plan Item Error Handler	172
PlanItem Execute Request	173
Operation Milestone Release Request	176
Operation Plan Item Suspend Request	177
Operation PreQualification Failed	178
Operation OpdErrorHandler	182
Jeopardy Services API samples	186
Purge Plans	186
Get Plan in Jeopardy	186
Suspend Rule	187
Resume Rule	187
Update Rule	187
Rule Criteria	187
Add Rule	187
Get Jeopardy Alert Message	188
Get All Rules	188
Delete Rules	188

Migrate Plan Fragments	188
Plan Time Line	189
AOPD API Samples	190
Global Cache Interface	190
Product Hierarchy Interface	190
Synchronous Plan Generation Interface	190
Broker Service API Samples	204
Actuator APIs	204
Post loggers name	204
Actuator root web endpoint	205
Actuator web endpoint 'loggers'	205
Actuator web endpoint 'health'	205
Actuator web endpoint 'health-path'	206
Instance Management APIs	206
Recovery APIs	211
Response Codes and Messages	212
Error Codes	212
TIBCO Documentation and Support Services	214
Legal and Third-Party Notices	216

Architecture

TIBCO Order Management is a metadata-driven order management system, which generates orders based on meta-data specified in product catalogs. Order management and service provisioning are a single-service or product workflow. Order management system, which can adapt to the changes in processes, metadata, and inventory. The traditional OSS/BSS ¹ approaches with the data silos fail to provide a dynamic and agile solution. An end-to-end order management system based on product and service catalogs is a key differentiator of TIBCO Order Management.

TIBCO Order Management is a comprehensive software solution, which designs, deploys, and maintains high-performance scalable enterprise-level business processes for advanced and dynamic order management and fulfillment. TIBCO Order Management enables customers to introduce quickly new product offerings and in most cases requires little or no change for order management and its fulfillment. The product bundles are decomposed into existing products, automatically generating a plan specific to the order.

TIBCO Order Management also enables customers to manage efficiently changes to the existing business process to meet a rapidly changing business environment.

The product model can be defined following SID 9 guidelines by using TIBCO® Product and Service Catalog or any other catalog management system and imported into TIBCO Order Management.

Services and Operations

The interfaces are organized into services that include related operations. The following interfaces are exposed for any client to invoke the services related to the framework. These services can be invoked on REST over HTTP channel.

REST Services endpoint URL has the following format:

`http://localhost:port`

¹Operational/Operations Support System (OSS) software applications support back-office activities operating a company network, provision, and maintain customer services. OSS is traditionally used by network planners, service designers, operations, architects, support, and engineering teams in the service provider.

For Example, `http://localhost:port/v1/order`.

The operations specified in the table follow the naming conventions.

Services and Operations Summary

Module	Service	Operation
Orchestrator	Order	PreQualificationFailedReply
		GetOrderDetails
		SubmitOrder
		WithdrawOrder
		SuspendOrder
		CancelOrder
		BulkAction
		ActivateOrder
		FeasibilityReply
		AmendOrder
		Get Order State Machine XML
		GetOrderMessages
		Purge Order
		GetOrderExecutionPlan
		SubmitOrderExecutionPlan
		OpdErrorHandlerReply
		submitPlanErrorNotification
		GetOrderExecutionPlan
		PlanItemSuspendReply
		MilestoneNotify
		PlanItemExecuteReply
		PlanItemErrorHandlerReply

Module	Service	Operation
Data Service	plan	PlanItemBulkErrorHandlerReply
		getplanfragment
	planItems	getPlan
		setPlan
	planItems	getPlanItems
		setPlanItems

Module	Service	Operation
Catalog Service	Action model	Submit Single Action Model
		Get Bulk Action Model
		Submit Bulk Action Model
		Delete Bulk Action Model
		Get All Action Model
	Plan Fragment model	Submit Single PlanFragment Model
		Get Bulk Plan Fragment Model
		Submit Bulk PlanFragment Model
		Delete Bulk PlanFragment Model
		Get All Plan Fragment Model
	Product model	Submit Single Product Model
		Get Bulk Product Model
		Submit Bulk Product Model
		Delete Bulk Product Model
		Get All Product Model
TM Forum Adapter Service	productOrder	Create, delete, and retrieve a product order

Module	Service	Operation
Authorization Service	User	Create user
		Update user
		Get the user details
		Delete the user
	Tenant	Register tenant
		Update tenant information
		Get tenant information
		Delete tenant

Module	Service	Operation
Archival Service	Order	GetPlansByCriteria
		GetOrderSummary
		GetOrdersByCriteria
		getBulkAction
		getAllBulkAction
		purgeOrder
		GetAuditTrail
		submitAuditTrail
		GetPlanitemByCriteria
		GetPieChartDetailsByCriteria
		GetPlanDetails
		GetOrderDetails
	Searches	GetSavedSearches
		UpdateSavedSearches
		SavedSearches
		DeleteSavedSearches
Jeopardy Service	Plan	Purge Plans This API purges the plans in the final state.
		getPlanInJeopardy

Module	Service	Operation
	Plan Fragment	Plan Fragment Migration
	Plan Timeline	Generate plan completion timeline
Broker Service	Notifications	Re-send pending notifications
		Re-send all pending notifications
		Get pending notifications

i Note: For all the web services to work correctly, the data models must be available to Orchestrator and Automatic Order Plan Development components.

Integration Channels

To provide maximum flexibility for exposing services to the enterprise, service implementation is separated from the interface. By using this, services can be invoked from different transport mechanisms.

The following transports are supported:

- [SOAP over HTTP](#)
- [SOAP over JMS](#)
- [REST over HTTP](#)

SOAP Over HTTP

SOAP over HTTP service calls use the standard synchronous web services message format. The integration pattern is as follows:

SOAP over HTTP Integration Pattern



In the SOAP over HTTP pattern, a service client makes a synchronous service call to the Order Management Services component. The service calls the appropriate operation and performs the required operation before sending a reply to the calling application.

Since TIBCO Order Management is deployed as part of the Order Management Server, the endpoint URLs for SOAP over HTTP have changed. The new endpoint URLs for the respective component is as follows:

- **AOPD Service** in standalone and collocated mode
`http://<host>:<port>/api/aopdService`

SOAP Over JMS

SOAP works over a variety of transports and is not restricted to HTTP web services. JMS provides a highly available transport mechanism that delivers messages over and above what HTTP provides. SOAP over JMS service calls use the TIBCO proposed specification extension for sending SOAP messages over JMS. TIBCO ActiveMatrix BusinessWorks supports calling SOAP over JMS synchronously by using the standard SOAP Request/Reply activity. The non-TIBCO ActiveMatrix BusinessWorks application must be able to call SOAP over JMS services both synchronously and asynchronously by using standard JMS messaging techniques that implement the required elements of the specification extension.

The synchronous integration pattern is as follows:

SOAP over JMS Synchronous Integration Pattern



The asynchronous integration pattern is as follows:

SOAP over JMS Asynchronous Integration Pattern



In the SOAP over JMS pattern, a service client makes a synchronous or asynchronous service call to the Order Management Services component. The service calls the

appropriate operation and performs the required operation before sending a reply back to the calling application.

The JMS destinations where the SOAP message must be sent is specified in the respective web services WSDL document.

REST over HTTP/HTTPS

REST over HTTP service calls use the standard web services message format. The integration pattern is as follows:

REST over HTTP/HTTPS Integration Pattern



In the REST over HTTP/HTTPS pattern, a service client makes a service call to the Order Management Services component. The service calls the appropriate operation and performs the required operation before sending a reply to the calling application.

The endpoint URLs for REST over HTTP/HTTPS have changed. The new endpoint URLs for respective components are as follows:

REST Services

Service	EndPoint	Method	Description
Orchestra tor	<protocol>://<host>:<port>/v1/order/preQualificationFailedReply	POST	PreQualificationFailedReply
	<protocol>://<host>:<port>/v1/order	GET	GetOrderDetails
	<protocol>://<host>:<port>/v1/order	POST	SubmitOrder
	<protocol>://<host>:<port>/v1/order	DELETE	WithdrawOrder
	<protocol>://<host>:<port>/v1/order/suspend	POST	SuspendOrder
	<protocol>://<host>:<port>/v1/order/cancel	POST	CancelOrder
	<protocol>://<host>:<port>/v1/order/bulkaction	POST	BulkAction
	<protocol>://<host>:<port>/v1/order/activate	POST	ActivateOrder
	<protocol>://<host>:<port>/v1/order/feasibilityReply	POST	FeasibilityReply
	<protocol>://<host>:<port>/v1/order/amend	POST	AmendOrder
	<protocol>://<host>:<port>/v1/order/scxml	GET	Get Order State Machine XML
	<protocol>://<host>:<port>/v1/order/messages	GET	GetOrderMessages
	<protocol>://<host>:<port>/v1/order/purgeOrder	DELETE	Purge Order

Service	EndPoint	Method	Description
	<protocol>://<host>:<port>/v1/plan	GET	GetOrderExecutionPlan
	<protocol>://<host>:<port>/v1/plan	POST	SubmitOrderExecutionPlan
	<protocol>://<host>:<port>/v1/plan/opd-error-handler-reply	POST	OpdErrorHandlerReply
	<protocol>://<host>:<port>/v1/plan/error	POST	submitPlanErrorNotification
	<protocol>://<host>:<port>/v1/plan	GET	GetOrderExecutionPlan
	<protocol>://<host>:<port>/v1/planitem/suspendreply	POST	PlanItemSuspendReply
	<protocol>://<host>:<port>/v1/planitem/milestonenotify	POST	MilestoneNotify
	<protocol>://<host>:<port>/v1/planitem/executionreply	POST	PlanItemExecuteReply
	<protocol>://<host>:<port>/v1/planitem/errorHandlerreply	POST	PlanItemErrorHandlerReply
	<protocol>://<host>:<port>/v1/planitem/bulkErrorHandlerreply	POST	PlanItemBulkErrorHandlerReply
	<protocol>://<host>:<port>/v1/planFragment	GET	getplanfragment

Service	EndPoint	Method	Description
Data Service	<protocol>://<host>:<port>/v1/plan	GET	Operation getPlan
	<protocol>://<host>:<port>/v1/plan	POST	Operation setPlan
	<protocol>://<host>:<port>/v1/planitems	POST	Operation setPlanItem
	<protocol>://<host>:<port>/v1/planitems/get	POST	Operation getPlanItems
Catalog Service	<protocol>://<host>:<port>/v1/actionModel	POST	Operation SubmitActionModel
	<protocol>://<host>:<port>/v1/actionModel/all	GET	Operation gets all action models
	<protocol>://<host>:<port>/v1/actionModel/bulk	POST	Operation SubmitActionModels
	<protocol>://<host>:<port>/v1/actionModel/bulk	DELETE	Operation PurgeActionModels
	<protocol>://<host>:<port>/v1/actionModel/bulk	GET	Operation gets bulk action models
	<protocol>://<host>:<port>/v1/planfragmentmodel	POST	Operation SubmitPlanFragmentModel
	<protocol>://<host>:<port>/v1/planfragmentmodel/bulk	DELETE	Operation PurgePlanFragmentModels
	<protocol>://<host>:<port>/v1/planfragmentmodel/all	GET	operation gets all planfragmentmodels

Service	EndPoint	Method	Description
	<protocol>://<host>:<port>/v1/planfragmentmodels/bulk	GET	operation get bulk planfragmentmodels
	<protocol>://<host>:<port>/v1/planfragmentmodels/bulk	POST	Operation SubmitPlanFragmentModels
	<protocol>://<host>:<port>/v1/productmodel	POST	Operation SubmitProductModel
	<protocol>://<host>:<port>/v1/productmodel/bulk	DELETE	Operation PurgeProductModels
	<protocol>://<host>:<port>/v1/productmodel/all	GET	operation gets all productmodels
	<protocol>://<host>:<port>/v1/productmodel/bulk	GET	operation gets bulk productmodels
	<protocol>://<host>:<port>/v1/productmodel/bulk	POST	Operation SubmitProductModels
TM Forum Adapter Service	<protocol>://<host>:<port>/tmf-api/productOrdering/v4/productOrder	POST	Create a product order
	<protocol>://<host>:<port>/tmf-api/productOrdering/v4/productOrder	GET	Retrieve a product order
	<protocol>://<host>:<port>/tmf-api/productOrdering/v4/productOrder	DELETE	Delete a product order

Service	EndPoint	Method	Description
Authorization Service	<protocol>://<host>:<port>/v1/user	GET	Get user
	<protocol>://<host>:<port>/v1/user	POST	Create user
	<protocol>://<host>:<port>/v1/user	PUT	Update user
	<protocol>://<host>:<port>/v1/user	DELETE	Delete user
	<protocol>://<host>:<port>/v1/tenant	GET	Get tenant information
	<protocol>://<host>:<port>/v1/tenant	POST	Register tenant
	<protocol>://<host>:<port>/v1/tenant	PUT	Update tenant information
	<protocol>://<host>:<port>/v1/tenant	DELETE	Delete tenant
Jeopardy Service	<protocol>://<host>:<port>/v1/plan	POST	Purge Plans This API purges the plans in the final state.
	<protocol>://<host>:<port>/v1/plan	GET	getPlanInJeopardy
	<protocol>://<host>:<port>/v1/plan	GET	Enrich plan
	<protocol>://<host>:<port>/v1/plan-fragment	POST	Plan Fragment Migration
	<protocol>://<host>:<port>/v1/planTimeLine	DELETE	Generate plan completion timeline
Archival	<protocol>://<host>:<port>/v2/orders/summar	POST	GetOrderSummary

Service	EndPoint	Method	Description
Service	y		

Service	EndPoint	Method	Description
	<protocol>://<host>:<port>/v2/orders/criteria	POST	GetOrdersByCriteria
	<protocol>://<host>:<port>/v1/orders/bulk-action-job	GET	getBulkAction
	<protocol>://<host>:<port>/v1/orders/bulk-action-job/all	GET	getAllBulkAction
	<protocol>://<host>:<port>/v1/orders/purge	DELETE	purgeOrder
	<protocol>://<host>:<port>/v2/plans/criteria	POST	GetPlanByCriteria
	<protocol>://<host>:<port>/v1/search	GET	GetSavedSearches
	<protocol>://<host>:<port>/v1/search	PUT	UpdateSavedSearches
	<protocol>://<host>:<port>/v1/search	POST	SavedSearches
	<protocol>://<host>:<port>/v1/search	DELETE	DeleteSavedSearches
	<protocol>://<host>:<port>/v1/orders/audit	GET	GetAuditTrail
	<protocol>://<host>:<port>/orders/audit	POST	SubmitAuditTrail
	<protocol>://<host>:<port>/v2/orders/planItems	GET	GetPlanItemByCriteria
	<protocol>://<host>:<port>/v2/orders/pieChartDetails	GET	GetPieChartDetailsByCriteria
	<protocol>://<host>:<port>/v1/plans/detail	GET	GetPlanDetails
	<protocol>://<host>:<port>/v1/orders/detail	GET	GetOrderDetails

Service	EndPoint	Method	Description
Broker Service	<protocol>://<host>:<port>/v1/notifications/re-send	POST	Re-send pending notifications
	<protocol>://<host>:<port>/v1/notifications/re-send/all	POST	Re-trigger all pending notifications
	<protocol>://<host>:<port>/v1/notifications/pending	GET	Get pending notifications

TIBCO Fulfillment Orchestration Suite Overview

New technologies and network architectures enable communications service providers (CSP) to create innovative converged products and services offerings that are introduced faster and have a shorter life cycles than previous service offerings to address a very changing and competitive market.

Because of the rapid pace of change in technology, the industry is evolving to become a contributor and not remain a mere consumer of technology. In this environment, communications service providers face the challenge of defining, managing, and delivering numerous complex products and variations to the market in the most effective way to differentiate themselves. TIBCO has concentrated and structured its services around the following points:

- New product offerings are designed and rolled out in a few weeks including implementation in all of the fulfillment chain.
- Customer orders are instantly fulfilled and provisioned in the network to maximize customer experience.
- Customer orders come from a large variety of order entries such as customer self-care portals, customer sales representative desks, or even network elements detecting service access to leverage fulfillment chain investment and support hardware rationalization.

TIBCO Fulfillment Orchestration Suite Components

The TIBCO Fulfillment Orchestration Suite solution supports end-to-end order fulfillment with order management and catalog capabilities.

The following are the constituents of the TIBCO Fulfillment Orchestration Suite:

- **TIBCO Product and Service Catalog:** TIBCO Product and Service Catalog is a

catalog function that defines and manages the life cycles of commercial and technical offerings.

- **TIBCO Order Management:** TIBCO® Order Management is a metadata-driven order management and fulfillment system, which allows the development of fulfillment plans based on meta-data specified in product catalogs. Order fulfillment and service provisioning are no longer a simple single-service or product workflow. The dynamic bundled offerings along with the explosion of devices, applications, real-time inventory management, and third-party content providers require a complex order fulfillment system, which can adapt to the changes in processes, metadata, and inventory. The Traditional Operation Support System/Business Support System approach with the data silos fails to provide a dynamic and agile solution.
- **TIBCO® Fulfillment Provisioning:** TIBCO Fulfillment Provisioning is a provisioning component that automates the activation of the underlying network services and allocation of all the network resources. This provisioning element implements service order management and resource order management TAM applications, aligns to service and resource activator OSS/J order management components, uses eTOM service configuration and activation, and implements resource provisioning processes and functions.
- **TIBCO® Product and Service Inventory:** TIBCO Product and Service Inventory maintains a current image of customer products at any given point in time that can support fast concurrent read or write access while ensuring data consistency. TIBCO Product and Service Inventory also provides a rich, user-friendly web interface that can be used to explore and modify the contents of the system in a safe and secure manner.

To enable TIBCO Fulfillment Orchestration Suite to provide a truly unified and cohesive solution suite, different components of the suite have been integrated. For instance, the suite provides predefined inter-connectivity between TIBCO Order Management and TIBCO Product and Service Catalog.

Fulfillment Management Services

The fulfillment management services are the set of services that are related to entities associated with the order. You can get information about these entities and perform other operations through Fulfillment Management Services.

Security Header

TIBCO Order Management invokes both RESTful and SOAP-based web services.

- TIBCO Order Management supports various web services. Order services and Automated Order Plan Development services require the WS-Security username Token mechanism. All other service requests require the REST JSON Web Token (JWT) mechanism. It uses TIBCO Administrator™ credentials in a standards-compliant manner.
- For the RESTful service, by default, the Authorization Service provides a description of the username and passwords for the operations on the Order Management System through REST web service requests. Order services in the Order Management Server can be secured by enabling the authentication-based token.

Through the Authorization Service, you can create or modify users. The user and password are validated from the database and created a user if it is not already present.

Order Service

This service contains operations regarding submitting and retrieving orders.

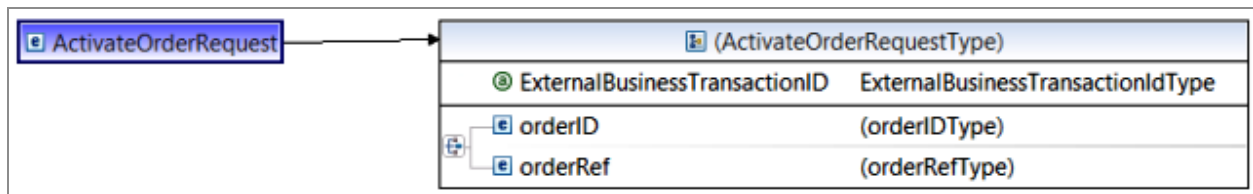
Activate Order

The Activate Order operation is used to activate or resume a suspended order. This operation takes an orderId or orderRef and attempts to activate or resume the order for

the given orderId or orderRef. This operation is asynchronous. Only valid orders and orders that are in SUSPEND state can be activated.

Element name: os:ActivateOrderRequest

The request message format is:

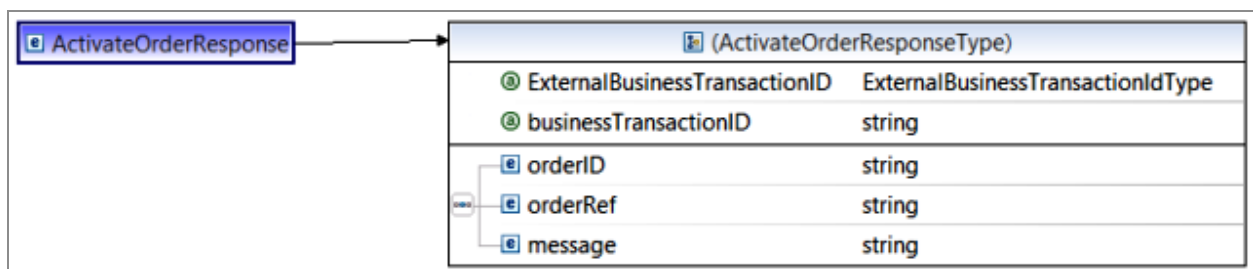


Activate Order Request Data Model

Element Name	Element Type	Description
ExternalBusinessTransactionID	String (Optional)	This is an attribute that can be used by external systems to keep track of requests. Note that the value of this attribute is not stored anywhere in TIBCO Order Management, and it is just meant for use by external systems.
orderId	String (Mandatory, Choice)	The ID of the order generated by Order Management Server. This must be an exact match.
orderRef	String (Mandatory, Choice)	The client order reference ID. This must be an exact match.

Element name: os:ActivateOrderResponse

The reply message format is:



Activate Order Response Data Model

Element Name	Element Type	Description
ExternalBusinessTransactionID	String (Optional)	This is an attribute that can be used by external systems to keep track of requests. Note that the value of this attribute is not stored anywhere in TIBCO Order Management, and it is just meant for use by external systems.
orderId	String (Mandatory, Choice)	Required
orderRef	String (Mandatory, Choice)	Required
message	String (Mandatory, Choice)	Interactive message sent to the client confirming request processing. Required.

Amend Order

An existing order can be modified by using the amend order web service. If the order reference already exists in Order Management Server, then it accepts the amend order request.

The request message format is:

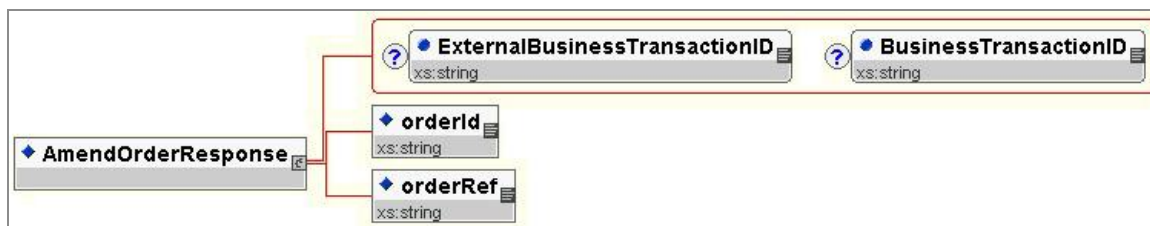


The following table explains elements defined in request schema.

Amend Order Request Data Model

Element Name	Element Type	Description
orderRequest	orderRequestType (Mandatory)	Order data structure. Data model for Order is defined in XSD (refer section Common Data Format Specifications).
ExternalBusinessTransactionID	String	Transaction ID sent by client.

The response message format is:



The following table explains the elements defined in the response schema.

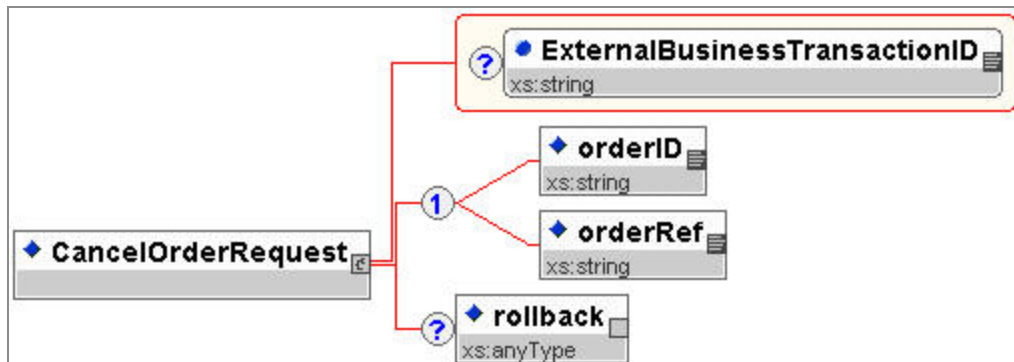
Amend Order Response Data Model

Element Name	Element Type	Description
ExternalBusinessTransactionID	String	Transaction ID sent by client.
BusinessTransactionID	String	Transaction ID generated by Order Management Server and used internally by Order Management Server and Orchestrator.
orderId	String (Optional)	Order ID generated by the order manager (Order Management Server). If the order fails validation, this is omitted.
orderRef	String (Optional)	Client order reference.

Cancel Order

This operation cancels the specified order in the orchestration engine. The order must be previously submitted successfully and exist in the orchestration engine database. This function extracts the original order and resubmits it as an amendment with all order line actions set to CANCEL.

The request message format is:



Cancel Order Request Data Model

Element Name	Element Type	Description
ExternalBusinessTransactionID	String	Transaction ID sent by client.
orderID	String (Mandatory, Choice)	The ID of the Order generated by Order Management Server. This must be an exact match.
orderRef	String (Mandatory, Choice)	The client order reference ID. This must be an exact match.
rollback	Boolean (Optional)	Flag to indicate if the order is to be rolled back.

The reply message format is:

*Cancel Order Response Data Model*

Element Name	Element Type	Description
ExternalBusinessTransactionID	String	Transaction ID sent by client.
BusinessTransactionID	String	Transaction ID generated by Order Management Server, and used internally by Order Management Server and Orchestrator.
orderID	String (Mandatory, Choice)	The ID of the Order generated by Order Management Server. This must be an exact match.
orderRef	String (Mandatory, Choice)	The client order reference ID. This must be an exact match.
message	String	Interactive message sent to client confirming request processing.

Get Order Details

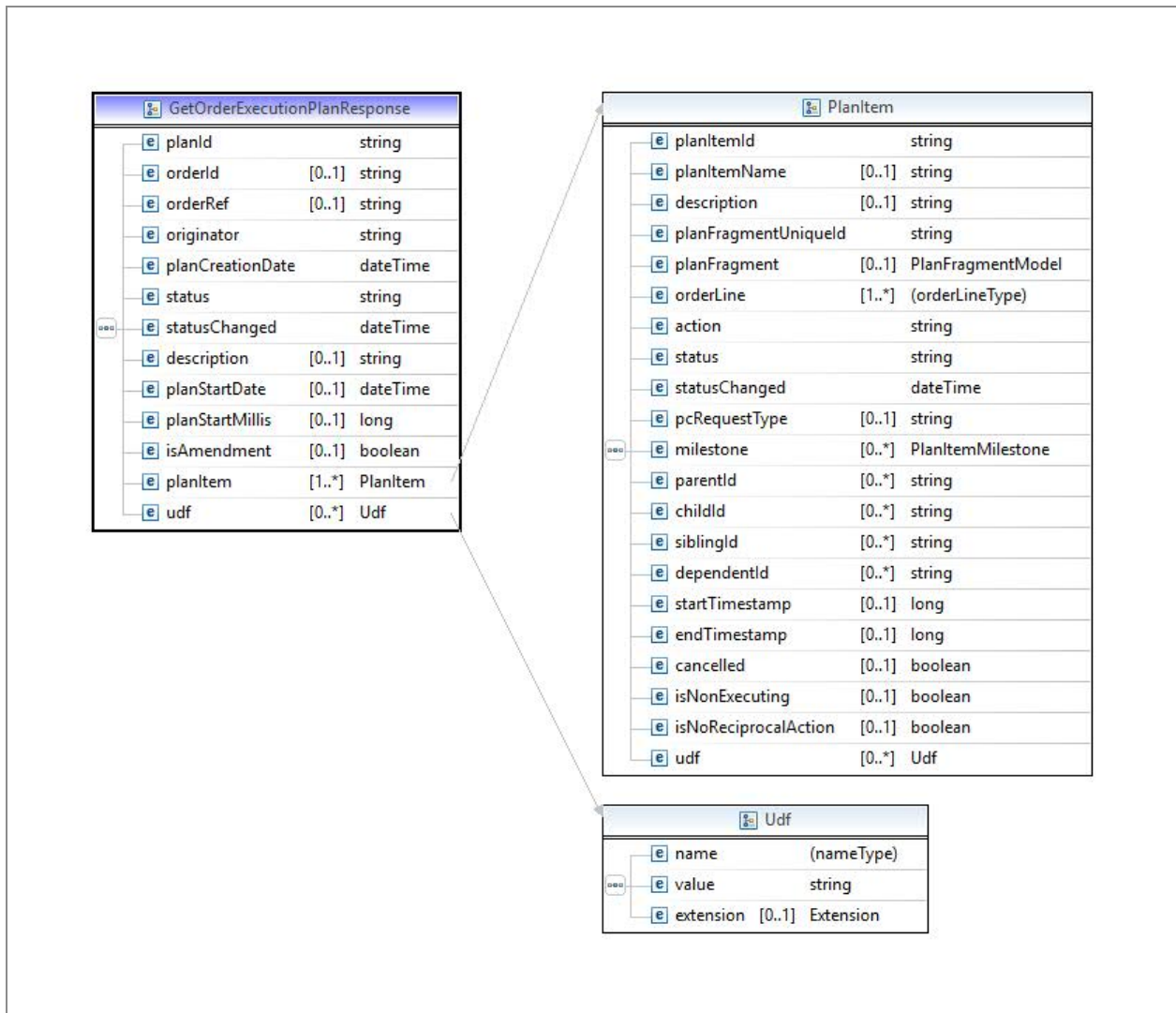
This operation retrieves the order details for a given order ID or order reference and returns to the calling application.

Get Order Execution Plan Request attributes

Element Name	Element Type	Description
externalBusinessTransactionId	String	Transaction ID sent by client.
orderId	String (Mandatory, Choice)	The ID of the Order. This must be an exact match.
planId	String (Choice)	The ID of the plan. This must be an exact match.
orderRef	String (Choice)	The order reference ID. This must be an exact match.

The reply message format is as follows:

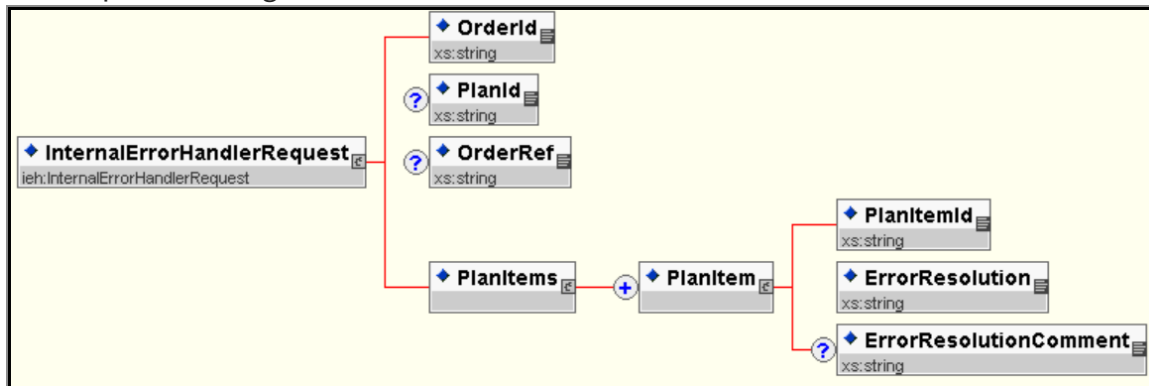
Get Order Execution Plan Reply



Handle Plan Item In Error

The **Handle Plan Item In Error** marks failed plan-items with an ERROR state and allows the user to select an appropriate action for the failed plan-item.

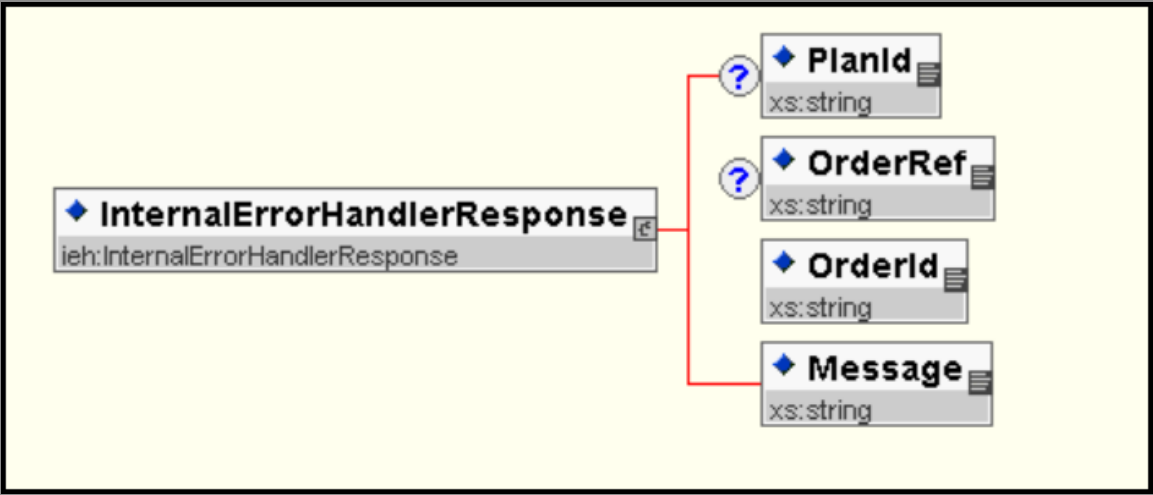
The request message format is:



Handle Plan Item In Error Request Data Model

Element Name	Element Type	Description
orderId	String (Mandatory)	The ID of the order generated by Order Management Server. This must be an exact match.
planID	String (optional)	Plan ID of the plan generated for the order.
orderRef	String (optional)	The client order reference ID.
planItemID	String	Unique Id identifying the plan Item.
errorResolution	String	RETRY RESUME COMPLETE MOPD
errorResolutionComment	String (optional)	Any String based user input.

The reply message format is:



Handle Plan Item In Error Response Data Model

Element Name	Element Type	Description
orderId	String	The ID of the order generated by Order Management Server.
planID	String	Plan ID of the plan generated for the order.
orderRef	String	The client order reference ID.
message	String	Internal error handler service initiated.

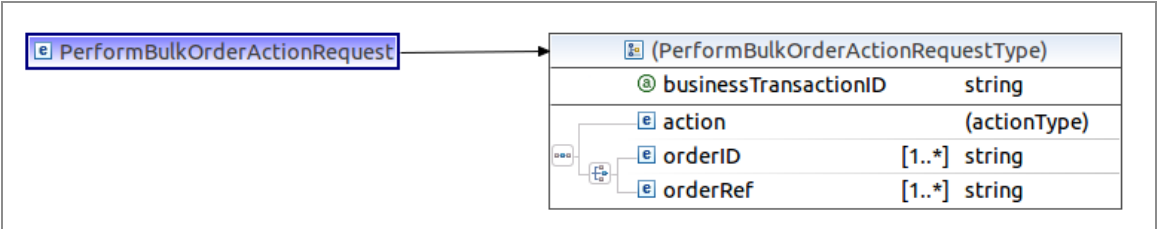
Perform Bulk Order Action

This service can be used to perform the following operations on multiple orders (Order IDs, orderRef):

- CANCEL
- SUSPEND
- RESUME
- WITHDRAW

Only one type of bulk action can be performed on multiple orders in a single request.

The request message format is:

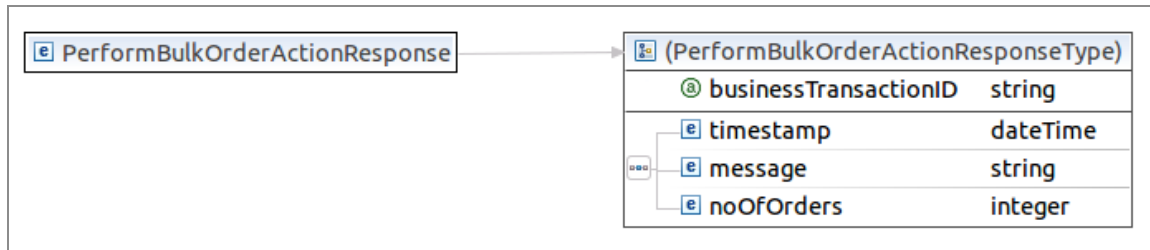


Perform Bulk Order Action Request Data Model

Element Name	Element Type	Description
businessTransactionID	String	Transaction ID sent by client.
action	String	Name of the bulk actions to perform: <ul style="list-style-type: none">• CANCEL• SUSPEND• RESUME• WITHDRAW
orderID	String (Optional)	List of order IDs on which bulk operations are to be performed.
orderRef	String (Optional)	List of order references on which bulk operations are to be performed.

A response in the form of an acknowledgment is returned asynchronously indicating the successful submission of request. The status of the bulk process can be tracked through Order Management Server UI, Order Management Server, and Orchestrator logs.

The response message format is:



Perform Bulk Order Action Response Data Model

Element Name	Element Type	Description
businessTransactionID	String	Transaction ID sent by client.
timestamp	dateTime	Date and time when Order Management Server receives a bulk order request.
message	String (Optional) request	Message indicating successful request submission.
noOfOrders	Integer (Optional)	Number of orders in the request.

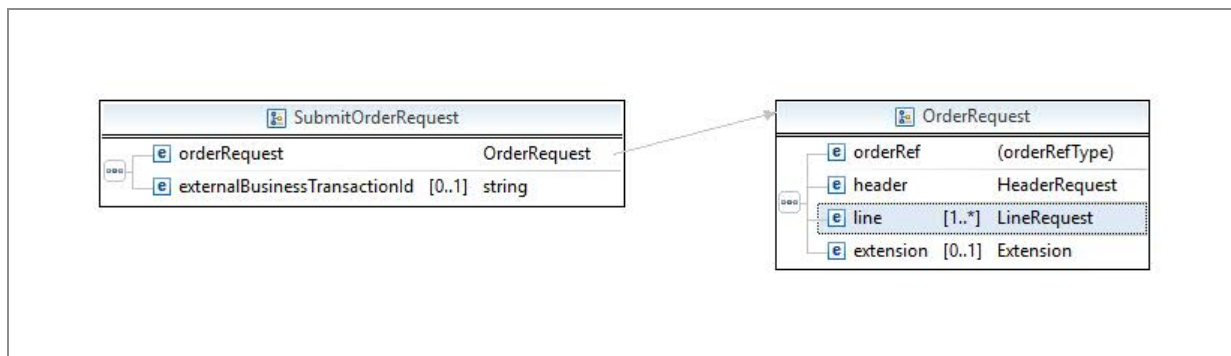
Submit Order

This operation accepts an order for the submission to the order management system and returns the order ID and order reference to the calling application.

- If the order is being submitted for the first time, do not specify the Order ID in the input. A value is generated internally.

The request message format is as follows:

Submit Order Request

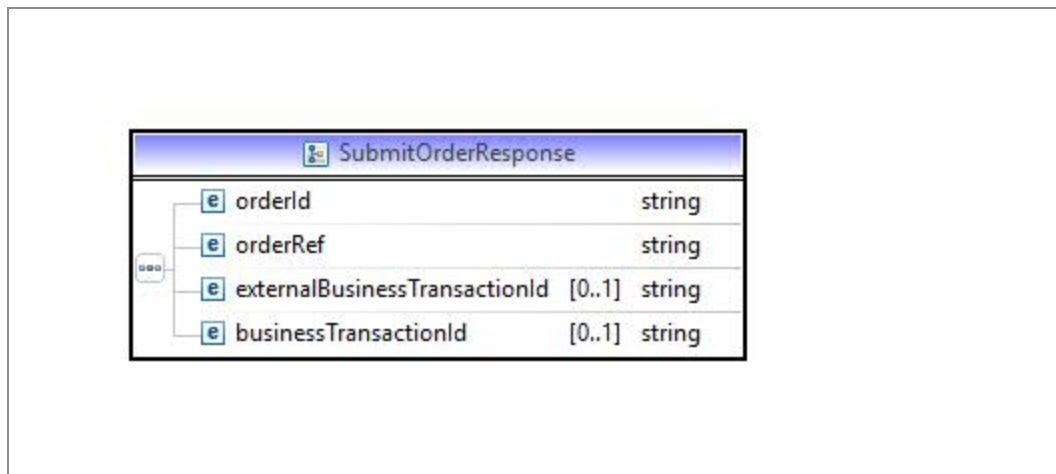


Submit Order Request attributes

Element Name	Element Type	Description
orderRequest	orderRequest (Mandatory)	Order data structure. Data model for Order is defined in (refer section Common Data Format Specifications).

The reply message format is as follows:

Submit Order Response

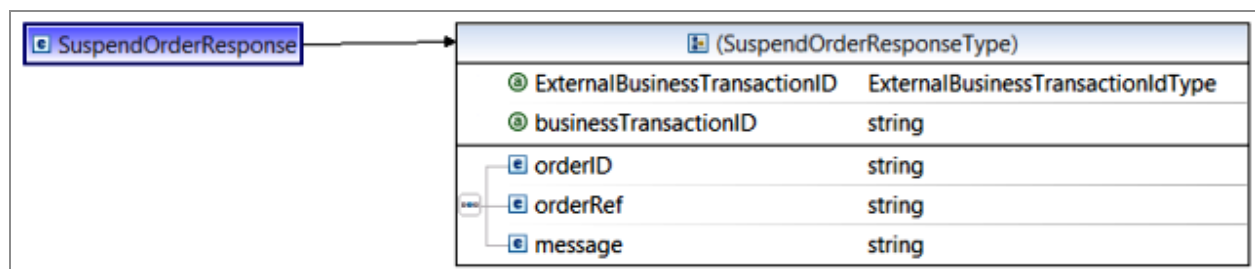


Suspend Order Request Data Model

Element Name	Element Type	Description
ExternalBusinessTransactionID	String (Optional)	This is an attribute, which can be used by external systems to keep track of requests. Note that the value of this attribute is not stored anywhere in TIBCO Order Management, and it is just meant for use by external systems.
orderId	String (Mandatory, Choice)	The ID of the order generated by Order Management Server. This must be an exact match.
orderRef	String (Mandatory, Choice)	The client order reference ID. This must be an exact match.

Element name: `os:SuspendOrderResponse`

The reply message format is:

*Suspend Order Response Data Model*

Element Name	Element Type	Description
ExternalBusinessTransactionID	String (Optional)	This is an attribute, which can be used by external systems to keep track of requests. Note that the value of this attribute is not stored anywhere in TIBCO Order Management, and it is just meant for use by external systems.

Element Name	Element Type	Description
orderId	String (Mandatory, Choice)	The ID of the order generated by Order Management Server. This must be an exact match.
orderRef	String (Mandatory, Choice)	The client order reference ID. This must be an exact match.
message	String	Interactive message sent to the client confirming request processing.

Synchronous Submit Order

This operation accepts an order for submission to the order management system and returns the order ID, order reference, and the plan to the calling application after the order has been completed through the Orchestrator engine.

- If the order is being submitted for the first time, do not specify the Order ID in the input. A value is generated internally.
- If orderRef is not specified, a value is generated and returned to the calling application. This value is always generated whether the order passed validation or not.

i Note: For SOAP over JMS Web services, the length of orderRef must be less than or equal to 100 characters.

An optional validate on submit function is available. If this function is enabled, prior to handing the order to the Order Management component, this service performs a validation on the order to determine if it is valid. The results of this validation are returned. If the order fails validation, it is not submitted to Order Management.

If the function is enabled by a flag, a copy of the order is saved in the offer cache to indicate it has been submitted.

The request message format is:

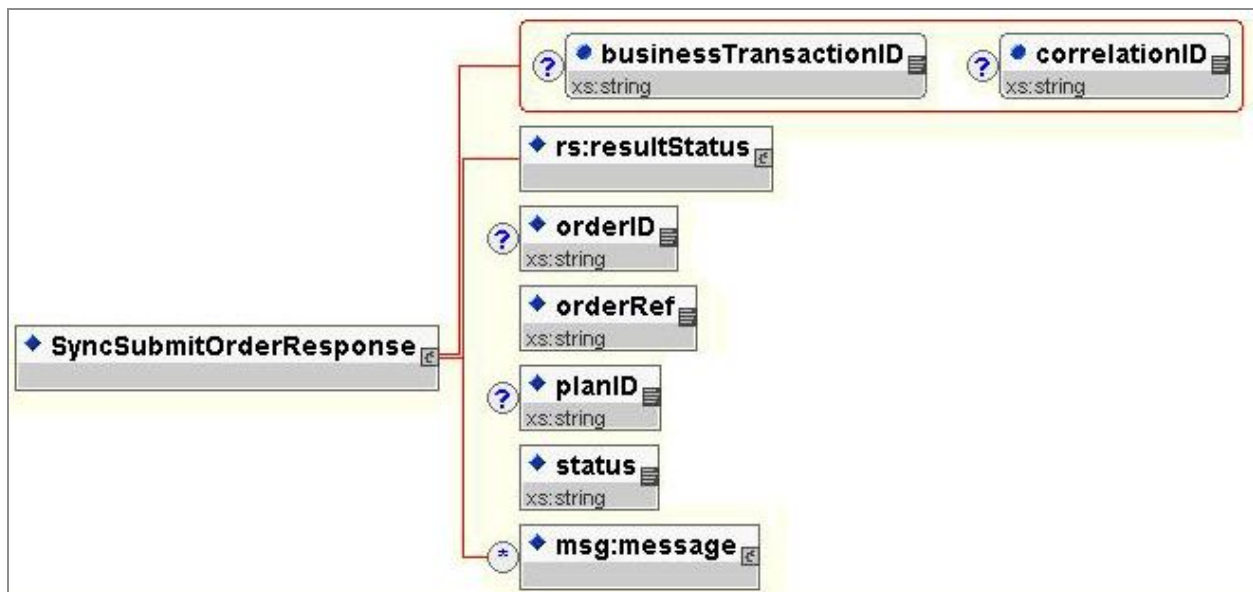
Synchronous Submit Order Request Data Model

*Synchronous Submit Order Request Data Model*

Element Name	Element Type	Description
orderRequest	orderRequestType (Mandatory)	Order data structure. Data model for Order is defined in XSD (refer section Common Data Format Specifications).
ExternalBusinessTransactionID	String	Transaction ID sent by client.

The reply message format is:

Synchronous Submit Order Response Data Model



Synchronous Submit Order Response Data Model Table

Element Name	Element Type	Description
resultStatus	ResultStatus	The result status of the operation.
OrderId	String	OrderID generated by Order Management Server.
OrderRef	String	Client order reference.
planID	String	Plan ID of the plan generated for the order.
status	String	Status of the order. For example, completed, or failure.
ExternalBusinessTransactionID	String	Transaction ID sent by client.
BusinessTransactionID	String	Transaction ID generated by Order Management Server, and used internally by Order Management Server and Orchestrator.

i Note: If the synchronous submitOrder request does not contain either orderID or orderRef element, then Order Management Server generates the orderRef and returns the same in the submitOrder response.

Withdraw Order

This operation withdraws all the submitted orders that are in execution state and do not withdraw orders that have been completed. As an input, this operation accepts the OrderID or OrderRef to withdraw an order and returns the message when sending the order for withdrawal in the Orchestrator component. After receiving an order removal notification from Orchestrator, it removes the requested order from the Order Management System component.

**Note:**

1. You cannot withdraw an order when it is in the COMPLETE state.
2. You can submit an order with the same OrderRef that is already withdrawn.

Withdraw Order Request attributes

Element Name	Element Type	Description
ExternalBusinessTransactionID	String	Transaction ID sent by client.
orderID	String (Mandatory, Choice)	The ID of the order generated by Order Management Server. This must be an exact match.
orderRef	String (Choice)	The client order reference ID. This must be an exact match.

Withdraw Order Response attributes

Element Name	Element Type	Description
orderID	String (Mandatory, Choice)	The ID of the order generated by Order Management Server. This must be an exact match.
orderRef	String (Mandatory, Choice)	The client order reference ID. This must be an exact match.
message	String	Interactive message sent to the client confirming request processing.

Purge Order

Purging of an order can be done in the following ways:

- Purging orders from Database
 - [Purging through Database Scripts](#)
 - [Purging through Partitioning](#)
- [Operation Purge Order \(Orchestrator API\)](#)
- [Operation Purge Order \(Archival API\)](#)
- [Operation Purge Plans \(Jeopardy API\)](#)

Purging through Database Scripts

i Note: Only the orders that are in the final state (for example, CANCELED and COMPLETE) can be purged.

Run the following script to purge orders:

- For order database: `$OM_HOME/db/dbscripts/postgreSQL/order/bin/purge-orders.sh` or `$OM_HOME/db/dbscripts/oracle/order/purge-orders.sh`
- For archival database: `$OM_HOME/db/dbscripts/postgreSQL/archival/bin/purge-orders.sh` or `$OM_HOME/db/dbscripts/oracle/archival/purge-orders.sh`

Usage:

```
purge_orders.sh [ORDER_STATUS] [FROM_DATE] [TO_DATE] [COMMIT_SIZE]
[PURGE_AUDIT]";E.g.: purge_orders.sh COMPLETE 26-04-201616:59:59 29-04-
201617:59:59 10 YES";
```

For more information, see the `purge_orders_ReadMe.txt` file on the following locations:

- PostgreSQL: `$OM_HOME/db/dbscripts/postgreSQL/order/bin` for the order database and `$OM_HOME/db/dbscripts/postgreSQL/archival/bin` for the archival database.
- Oracle: `$OM_HOME/db/dbscripts/oracle/order` for the order database and `$OM_HOME/db/dbscripts/oracle/archival` for the archival database.

The default setting for `enablePurgeCompleteOrder` is true in the Orchestrator service. When this is true, orders are removed from the database once they reach a final state. As a result, APIs in the Orchestrator and Data Service stop working for any orders that have been purged. Therefore, if you want to keep using these APIs for orders after they are completed, you must set `enablePurgeCompleteOrder` to false.

Purging through Partitioning

i Note: Only the orders that are in the final state (for example, CANCELED and COMPLETE) can be purged.

PostgreSQL

Run the following partitioning procedures by using the `select drop_partition_by_name ('<partition_date>');` command:

- For order database: `$OM_HOME/db/dbscripts/postgreSQL/order/scripts/drop_partition_by_name.sql`
- For archival database: `$OM_HOME/db/dbscripts/postgreSQL/archival/scripts/drop_partition_by_name.sql`

For more information, see the `$OM_HOME/db/dbscripts/postgreSQL/order/bin/drop_partition_by_name_Readme.txt` file for the order database and `$OM_HOME/db/dbscripts/postgreSQL/archival/bin/drop_partition_by_name_Readme.txt` file for the archival database.

Oracle

Run the `drop_partition_by_name.sql` script to create `TRUNCATEPARTITION` procedure. To call the 'truncatepartition' procedure, give the following query:

```
<call TRUNCATEPARTITION('partition');>
```

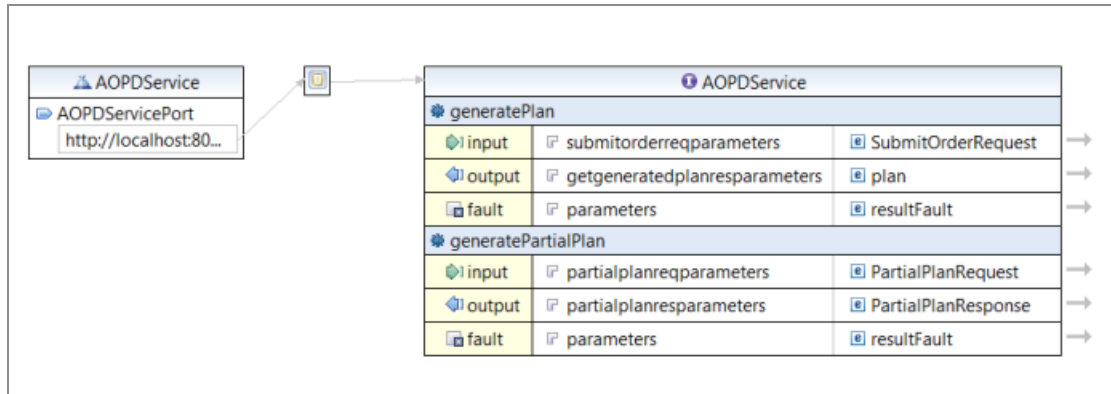
i Note: Input partition must be in `ORD_20220521010000` format.

For more information, see the `$OM_HOME/db/dbscripts/oracle/order/drop_partition_by_name_Readme.txt` file for the order database and `$OM_HOME/db/dbscripts/oracle/archival/drop_partition_by_name_Readme.txt` file for the archival database.

HOME/db/dbscripts/oracle/archival/drop_partition_by_name_Readme.txt file for the archival database.

Automated Order Plan Development Service

The following web services are provided in Automated Order Plan Development:



Generate Plan

The **Generate Plan** web service makes it possible to get a copy of a plan for a particular order prototype, and existing catalog, without submitting the order in the system. The following code sample is an example for the web service request for generatePlan.

Generate Plan Request

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ord="http://www.tibco.com/aff/orderservice"
xmlns:ord1="http://www.tibco.com/aff/order"
xmlns:com="http://www.tibco.com/aff/commontypes">
    <soapenv:Header>
        <wsse:Security soapenv:mustUnderstand="1"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
            <wsse:UsernameToken>
                <wsse:Username>eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX25hbWU
iOiJhZG1pb1biIsIlRFTkFOVElEIjoieVlCQ08iLCJzY29wZSI6WyJyZWFKIiwid3
JpdGUxSWiZXhwIjoxNjEyMjAwODY2LmJhdXRob3JpdGlscyI6WyJST0xFX0FETUOIi
0sImp0aSI6IjA1YjQ0ZDVhLWViYmMtNGFMnNi05ODVLLWZiYWNoNTFfNmRkNiIsI
```

```

mNsaWVudF9pZCI6Im9yZGVyLW1hbmFnZW1lbnQtY2xpZW50In0.
wEbVvMa2hzdWfsNixFU-e0x0UVXtGSiLsMavS4j6tC8</wsse:Username>
    <wsse:Password Type="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-username-token-profile-
1.0#PasswordText"/>
    </wsse:UsernameToken>
  </wsse:Security>
</soapenv:Header>
<soapenv:Body>
  <ord:SubmitOrderRequest ExternalBusinessTransactionID="sdsad">
    <ord:orderRequest>
      <ord1:orderRef>af201</ord1:orderRef>
      <ord1:header>
        <ord1:customerID>CUSTOMER</ord1:customerID>
        <ord1:requiredByDate>2013-11-30T13:20:00-
05:00</ord1:requiredByDate>
      </ord1:header>
      <!--1 or more repetitions:-->
      <ord1:line>
        <ord1:lineNumber>1</ord1:lineNumber>
        <ord1:productID>CFS_TV</ord1:productID>
        <ord1:quantity>1</ord1:quantity>
        <ord1:uom>-</ord1:uom>
        <!--Optional:-->
        <ord1:deliveryAddress>
          <com:line1>Address Line1</com:line1>
          <com:line2>Address Line2</com:line2>
          <com:line3>Pune</com:line3>
          <com:locality>Pune</com:locality>
          <com:region>Maharashtra</com:region>
          <com:country>IN</com:country>
          <com:postCode>411001</com:postCode>
          <com:supplementaryLocation>TIBCO software,
Pune</com:supplementaryLocation>
        </ord1:deliveryAddress>
        <ord1:action>PROVIDE</ord1:action>
        <ord1:requiredByDate>2013-11-30T13:20:00-
05:00</ord1:requiredByDate>
      </ord1:line>
    </ord:orderRequest>
  </ord:SubmitOrderRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Generate Partial Plan

The SOAP web service only provides a preview for order decomposition. The valid plan generation phases are VALIDATION, DECOMPOSING, OPTIMIZING, and PUBLISHING. In all other cases this web service returns a preview of the complete generated plan.

Generate Partial Plan Request

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ord="http://www.tibco.com/aff/orderservice"
xmlns:ord1="http://www.tibco.com/aff/order"
xmlns:com="http://www.tibco.com/aff/commontypes">
  <soapenv:Header>
    <wsse:Security soapenv:mustUnderstand="1"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken>

        <wsse:Username>eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX25hbWUi
        OiJhZG1pbIIsIlRFTkFOVElEIjoieVlCQ08iLCJzY29wZSI6WyJyZWFKIiwid3J
        pdGUixSwiZXhwIjoxNjEyMjAwODY2LCJhdXRob3JpdGlscyI6WyJST0xFX0FETU1
        OIl0sImp0aSI6IjA1YjQ0ZDVhLWVhYmMtNGFmNi05ODVLLWZiYWVhNTFjNmRkN
        iIsImNsaWVudF9pZCI6Im9yZGVyLW1hbmFnZW1lbnQtY2xpZW50In0.
        wEbVkJMa2hzdWfsNixFU-e0x0UVXtGSiLsMavS4j6tC8</wsse:Username>
        <wsse:Password Type="http://docs.oasis-
        open.org/wss/2004/01/oasis-200401-wss-username-token-profile-
        1.0#PasswordText"/>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>
    <ord:PartialPlanRequest ExternalBusinessTransactionID="?">
      <ord:orderRequest>
        <ord1:orderRef>af201</ord1:orderRef>
        <ord1:header>
          <ord1:customerID>CUSTOMER</ord1:customerID>
          <ord1:requiredByDate>2013-11-30T13:20:00-
05:00</ord1:requiredByDate>
        </ord1:header>
        <!--1 or more repetitions:-->
        <ord1:line>
          <ord1:lineNumber>1</ord1:lineNumber>
          <ord1:productID>CFS_TV</ord1:productID>
          <ord1:quantity>1</ord1:quantity>
          <ord1:uom>-</ord1:uom>
          <!--Optional:-->
```

```

        <ord1:deliveryAddress>
          <com:line1>Address Line1</com:line1>
          <com:line2>Address Line2</com:line2>
          <com:line3>Pune</com:line3>
          <com:locality>Pune</com:locality>
          <com:region>Maharashtra</com:region>
          <com:country>IN</com:country>
          <com:postCode>411001</com:postCode>
          <com:supplementaryLocation>TIBCO Software,
Pune</com:supplementaryLocation>
        </ord1:deliveryAddress>
        <ord1:action>PROVIDE</ord1:action>
        <ord1:requiredByDate>2013-11-30T13:20:00-
05:00</ord1:requiredByDate>
      </ord1:line>
    </ord:orderRequest>
    <ord:planGenerationPhase>OPTIMIZING</ord:planGenerationPhase>
  </ord:PartialPlanRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Order Management Services

The location of all Order Management Server web services is `http://<<Orchestrator Host>>:<<port of the orchestrator>>`.

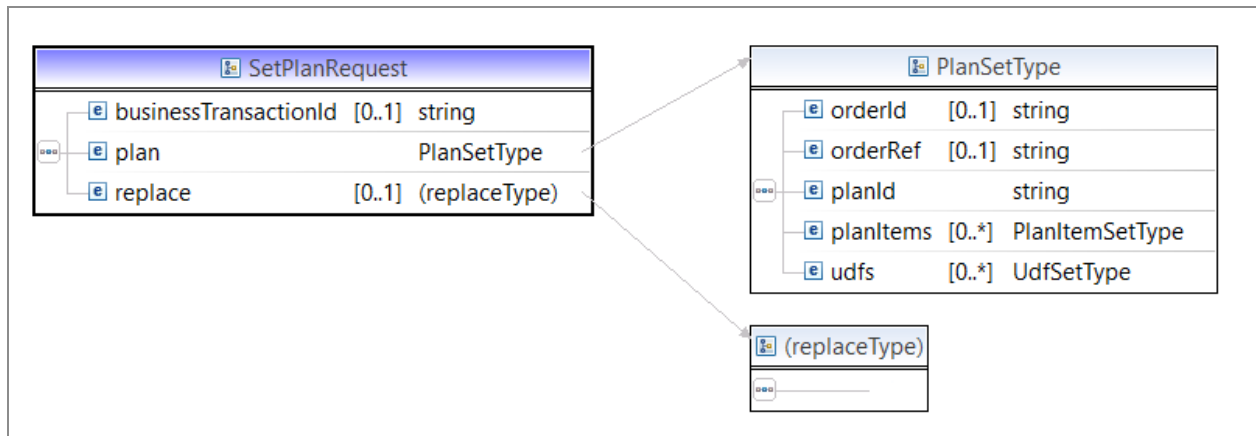
Data Service

Data Service is used to add, update, or modify the User-Defined Fields at the plans and plan items level. There are two types of Data services.

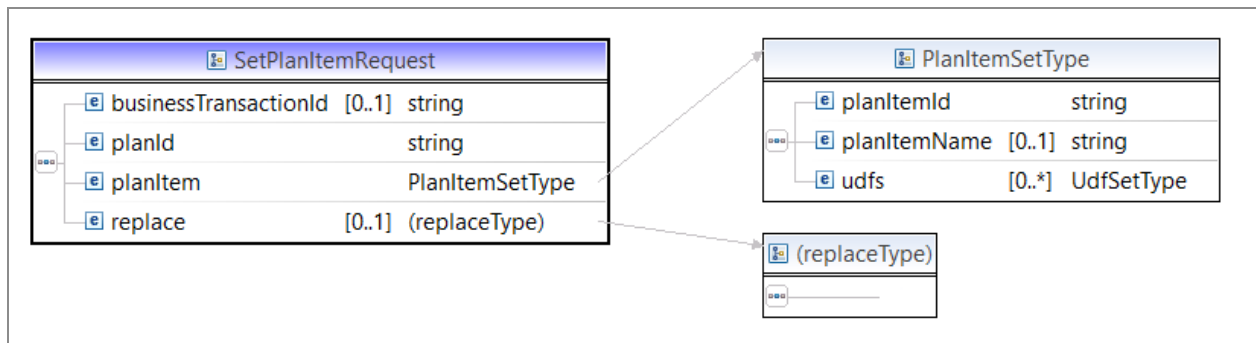
- Set plan or set plan items: This service is used to add, update, or modify the User-Defined Fields at the set plan or set plan items level.
- Get plan or get plan items: Through this service, a user receives the updated User-Defined Fields at the get plan or get plan items level.

The request message formats are as follows:

SetPlanRequest



SetPlanItemRequest



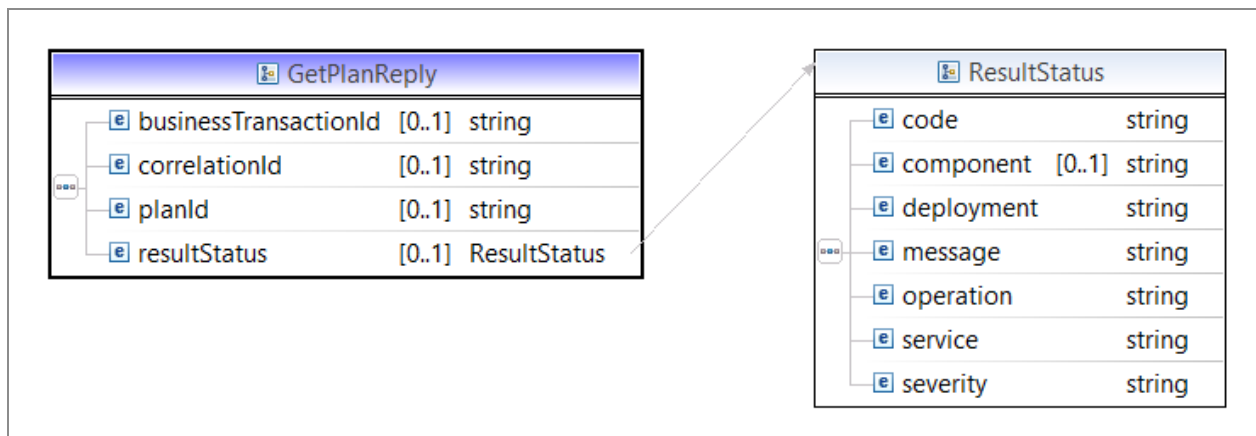
Element Name	Element Type	Description
businessTransactionId	string (Optional)	Transaction Id sent by client.
plan	PlanSetType	Type of plan.
replace	replaceType (Optional)	Type of replace.
orderId	string (Optional)	Id of an Order.
orderRef	string (Optional)	Order reference Id.
planId	string	Unique identifier for the plan.

Element Name	Element Type	Description
planItems	PlanItemSetType	Composite object with information about the plan.
udfs	UdfSetType (Optional)	Composite object with information related to UserDefined Field that is user defined data.
planItemId	string	Unique Id identifying the plan Item.
planItemName	string (Optional)	Name of the plan item.
correlationId	string (Optional)	Unique identifier for the correlation.
resultStatus	ResultStatus (Optional)	Status of the result.
code	string	Code of a plan.
component	string (Optional)	Name of the component.
deployment	string	Deployment of plan.
message	string	A descriptive text string associated with the code.
operation	string	This text field identifies the operation in progress at the point of failure.
service	string	Service of plan.
severity	string	A single character {F E W I S}. F- Fatal, E - Error, W - Warning, I - Information, and S - Success.
description	string (Optional)	Description of plan.

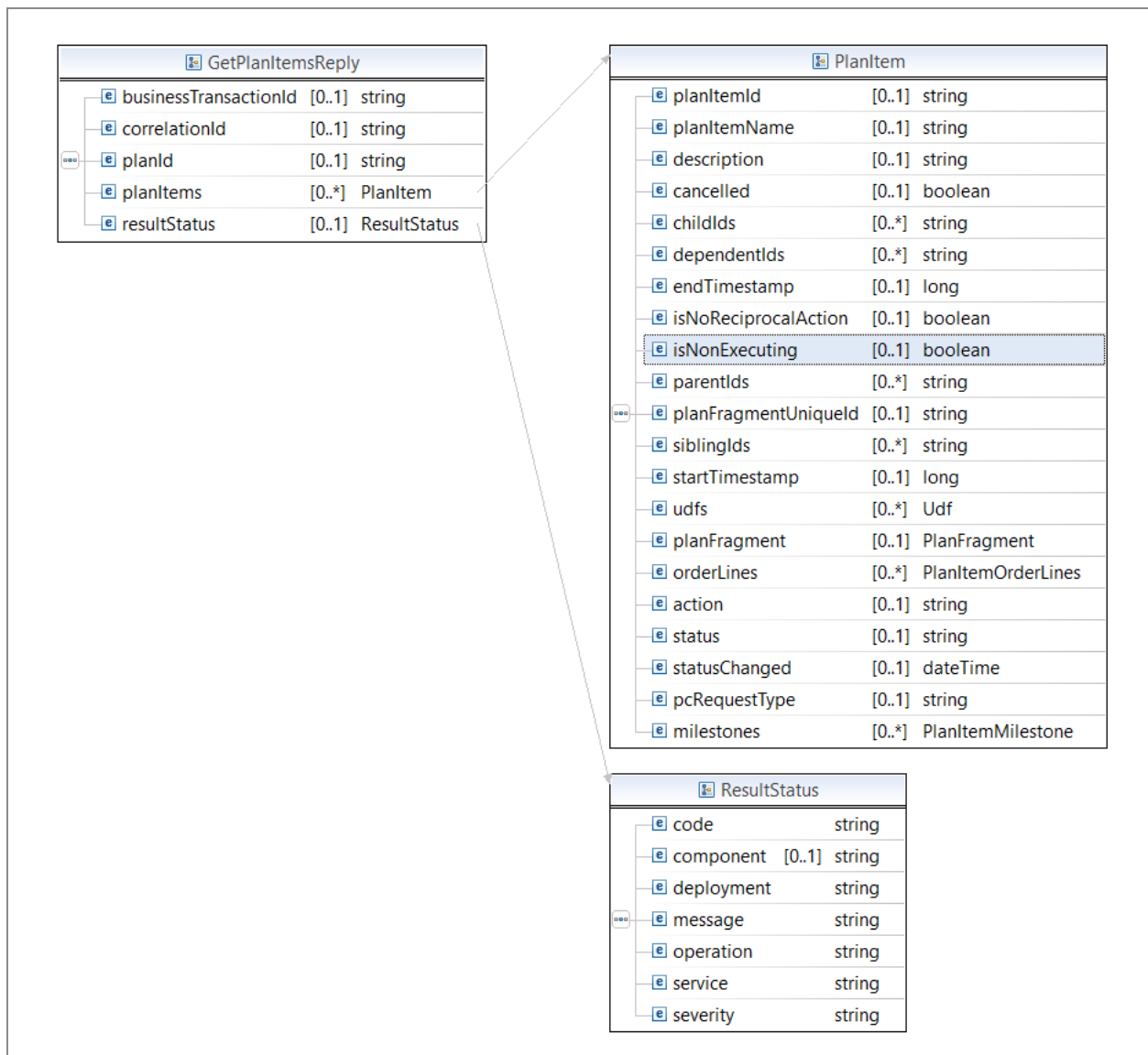
Element Name	Element Type	Description
dependentIds	string (Optional)	Plan Item Ids of dependent products for this plan item.
endTimestamp	long (Optional)	Timestamp that indicates when the plan item completed execution.
startTimestamp	long (Optional)	Timestamp that indicates when the plan item started execution.
parentIds	string (Optional)	Unique parent id.
siblingIds	string (Optional)	Id of the siblings.
planFragment	PlanFragment (Optional)	The process component to invoke when executing the plan item.
action	string (Optional)	Action for the plan Item.
status	string (Optional)	Status of the plan.
statusChanged	dateTime (Optional)	Date Time when the last status was changed.
milestones	PlanItemMilestone (Optional)	Milestone of the plan.

The reply message format are as follows:

GetPlanReply



GetPlanItemsReply



Catalog Service

This REST service is used to load the product model, plan fragment data model, and action model over HTTP.

The endpoints for the Catalog Services are as follows:

- POST <protocol>://<host>:<port>/v1/actionModel

- GET <protocol>://<host>:<port>/v1/actionModel/bulk
- POST <protocol>://<host>:<port>/v1/actionModel/bulk
- DELETE <protocol>://<host>:<port>/v1/actionModel/bulk
- GET <protocol>://<host>:<port>/v1/actionModel/all
- POST <protocol>://<host>:<port>/v1/productmodel
- GET <protocol>://<host>:<port>/v1/productmodel/bulk
- POST <protocol>://<host>:<port>/v1/productmodel/bulk
- DELETE <protocol>://<host>:<port>/v1/productmodel/bulk
- GET <protocol>://<host>:<port>/v1/productmodel/all
- POST <protocol>://<host>:<port>/v1/planfragmentmodel
- GET <protocol>://<host>:<port>/v1/planfragmentmodel/bulk
- POST <protocol>://<host>:<port>/v1/planfragmentmodel/bulk
- DELETE <protocol>://<host>:<port>/v1/planfragmentmodel/bulk
- GET <protocol>://<host>:<port>/v1/planfragmentmodel/all

TM Forum Adapter Service

TM Forum Adapter is an implementation for TMF622 API specification. It acts as a bridge between the TM Forum Specification and the TIBCO Order Management system.

Below mentioned service implementations on TM Forum adapter are mapped to the TIBCO Order Management Service.

Method	TMF622 Service	Order Management Service
POST	/tmf-api/productOrdering/v4/productOrder	/v1/order
DELETE	/tmf-api/productOrdering/v4/productOrder/{productId}	/v1/order
GET	/tmf-api/productOrdering/v4/productOrder/{productId}	/v1/order

The Product Order Resource of TM Forum is mapped with the SubmitOrder of Order Management Service. The following table lists the field to field mapping between the Product Order and SubmitOrder:

TMF622	Order Management Service
id	orderId
externalID	externalBusinessTransactionID, orderRequest/orderRef
priority	orderRequest/header/orderPriority
description	orderRequest/header/description
note	orderRequest/header/notes
requestedCompletionDate	orderRequest/header/requiredOnDate
requestedStartDate	orderRequest/header/requiredByDate
relatedParty/id [where relatedParty/role='customer'] (default value 'dummy customerID')	orderRequest/header/customerID
relatedParty/id [where relatedParty/role='subscriber'] (default value 'dummy subscriberID')	orderRequest/header/subscriberID
baseType	orderRequest/header/udf/name=baseType,value=component json
billingAccount	orderRequest/header/udf/name=billingAccount,value=component json
category	orderRequest/header/udf/name=category,value=component json
channel	orderRequest/header/udf/name=channel,value=component

TMF622	Order Management Service
	json
notificationContact	orderRequest/header/udf/name=notificationContact,value=component json
orderTotalPrice	orderRequest/header/udf/name=orderTotalPrice,value=component json
payment	orderRequest/header/udf/name=payment,value=component json
relatedParty	orderRequest/header/udf/name=relatedParty,value=component json
schemaLocation	orderRequest/header/udf/name=schemaLocation,value=component json
orderItem	orderRequest/line
orderItem/state	orderResult/lines/status
orderItem/id	orderRequest/line/lineNumber
orderItem/productOffering/name	orderRequest/line/productID
orderItem/action/add	orderRequest/line/action/PROVIDE
orderItem/action/modify	orderRequest/line/action/UPDATE
orderItem/action/nochange	orderRequest/line/action/PROVIDE
orderItem/action/delete	orderRequest/line/action/CANCEL
orderItem/quantity	orderRequest/line/quantity
orderItem/product/type	orderRequest/line/UOM

TMF622	Order Management Service
[default '-']	
orderItem/characteristic	orderRequest/line/characteristic
appointment	orderRequest/line/udf/name=appointment,value=component json
billingAccount	orderRequest/line/udf/name=billingAccount,value=component json
itemPrice	orderRequest/line/udf/name=itemPrice,value=component json
itemTerm	orderRequest/line/udf/name=itemTerm,value=component json
itemTotalPrice	orderRequest/line/udf/name=itemTotalPrice,value=component json
orderItemRelationship	orderRequest/line/udf/name=orderItemRelationship,value=component json
payment	orderRequest/line/udf/name=payment,value=component json
product	orderRequest/line/udf/name=product,value=component json
productOffering	orderRequest/line/udf/name=productOffering,value=component json
qualification	orderRequest/line/udf/name=qualification,value=component json
schemaLocation	orderRequest/line/udf/name=schemaLocation,value=component json
type	orderRequest/line/udf/name=type,value=component json

Common Data Format Specifications

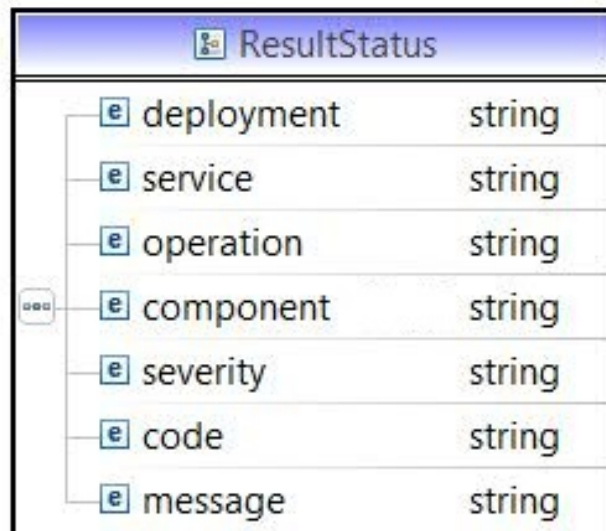
This section describes generic object classes used as input and output objects in the operations and services.

ResultStatus

This class is a result structure common for all operations. For any operation, the reply schema contains this mandatory object. It specifies if the service execution succeeded or failed.

Any client of operation must first check the value of the "severity" field to identify if the execution was successful or not.

The JSON schema structure is as follows:



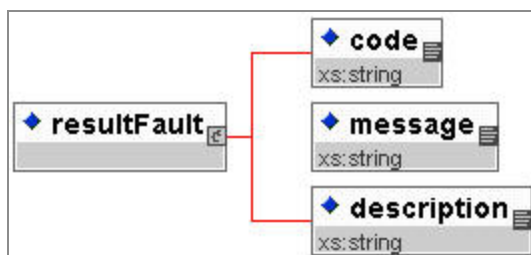
This class contains the following elements:

ResultStatus Element Details

Element name	Description
Component	<p>Name of the component that detected the error.</p> <p>For the TIBCO Order Management component implementation, this field is set to the deployment variable of the TIBCO BusinessEvents in which the error occurred.</p>
Operation	<p>This text field identifies the operation in progress at the point of failure. This is helpful when a request consists of many operations and the same code might be returned in more than one operation.</p> <p>This field displays the name of the operation where the error occurred.</p>
Severity	<p>A single character {F E W I S}.</p> <p>F- Fatal, E - Error, W - Warning, I - Information, and S - Success.</p>
Code	<p>The error code, a string that identifies the error and the TIBCO Order Management component that produced the error. This code is also visible inside component log and audit files.</p> <p>Refer Response Codes and Messages for details of this format.</p>
Message	A descriptive text string associated with the code.

ResultFault

The XML schema structure for the ResultFault is:



ResultFault Element Details

Element name	Description
code	The error code, a string that identifies the error and the TIBCO Order Management component that produced the error. This code is also visible inside the component log and audit files. Refer Response Codes and Messages for details of this format.
message	A descriptive text string associated with a code.
description	The description of the error.

Notification

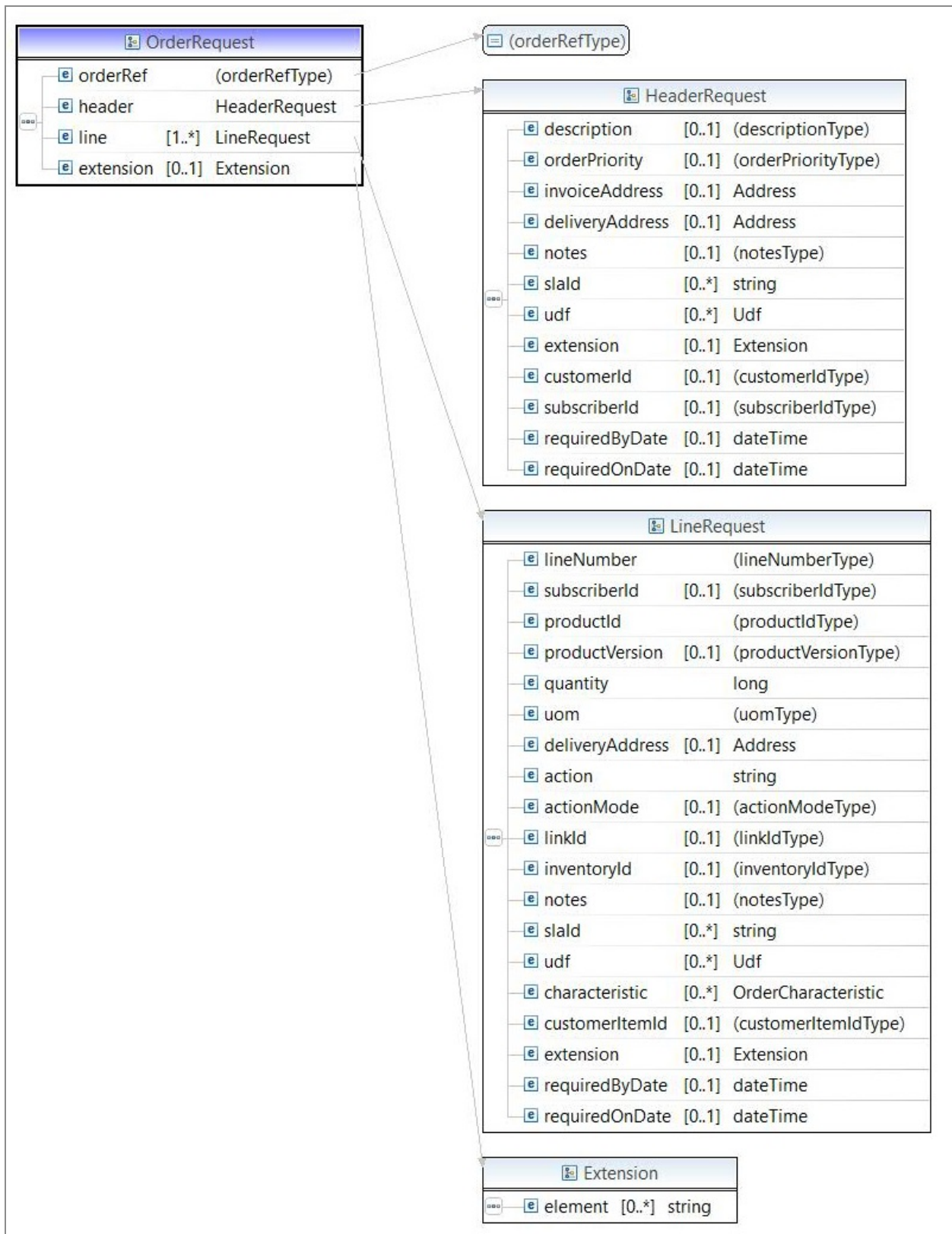
This class is the data model for the *Notification* object.

The \$OM_HOME/samples/Outbound Notification/ folder contains the notification-schema.json file, where an open-API schema file can be viewed via an appropriate tool (such as Swagger UI, Postman). This is just a sample schema and users must not confuse with RESTful API being exposed for these notifications. These JSON notifications are still published via Messaging channel.

The folder also contains sample notifications based on the schema, there are sample notifications for Order, Order Line, Plan Development, Plan Item, and Plan.

Order

This class is the attribute for the *Order* object. It might be used across services and ID defined in JSON format.

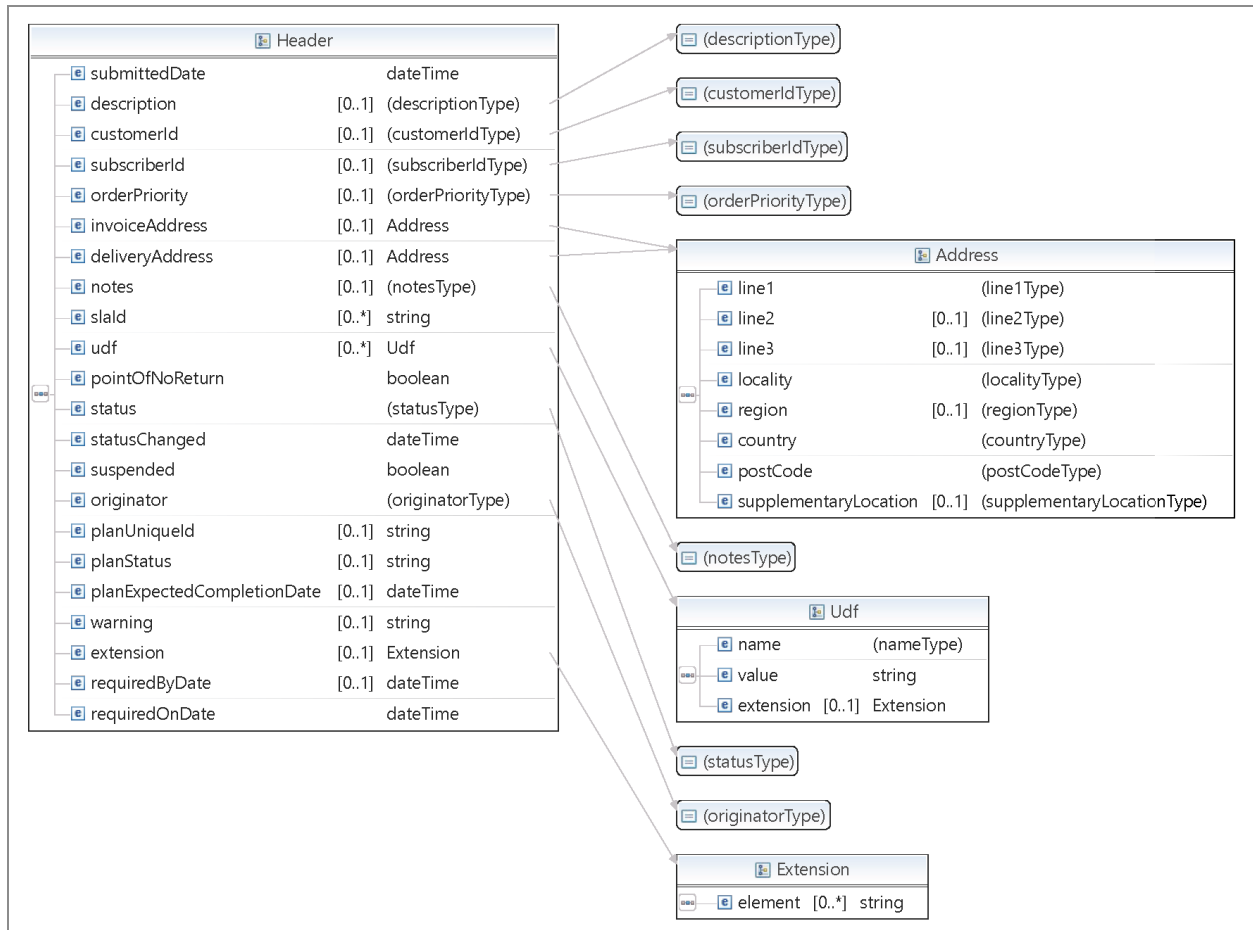


Order Element Details

Element Name	Element Type	Description
orderID	String (Optional)	Order ID. Not required during Submit Order but is returned by Get Order Details and Get Orders.
orderRef	String	Order reference ID is a mandatory field. If the orderRef element is not sent, then the schema validation fails.
header	headerType (Mandatory)	Order Header data structure.
line	lineType	Order Line data structure.

OrderHeader

This class is the attribute for the *header* object. It might be used across services and ID defined in JSON format.



OrderHeader Element Details

Element Name	Element Type	Description
OrderDescription	String (Optional)	Order description.
OrderStatus	String (Optional)	Current order status.
CustomerID	String (Optional)	Customer unique identifier. Omit for non-corporate orders. Either CustomerID or SubscriberID are necessary.
SubscriberID	String	Subscriber unique identifier. Omit for

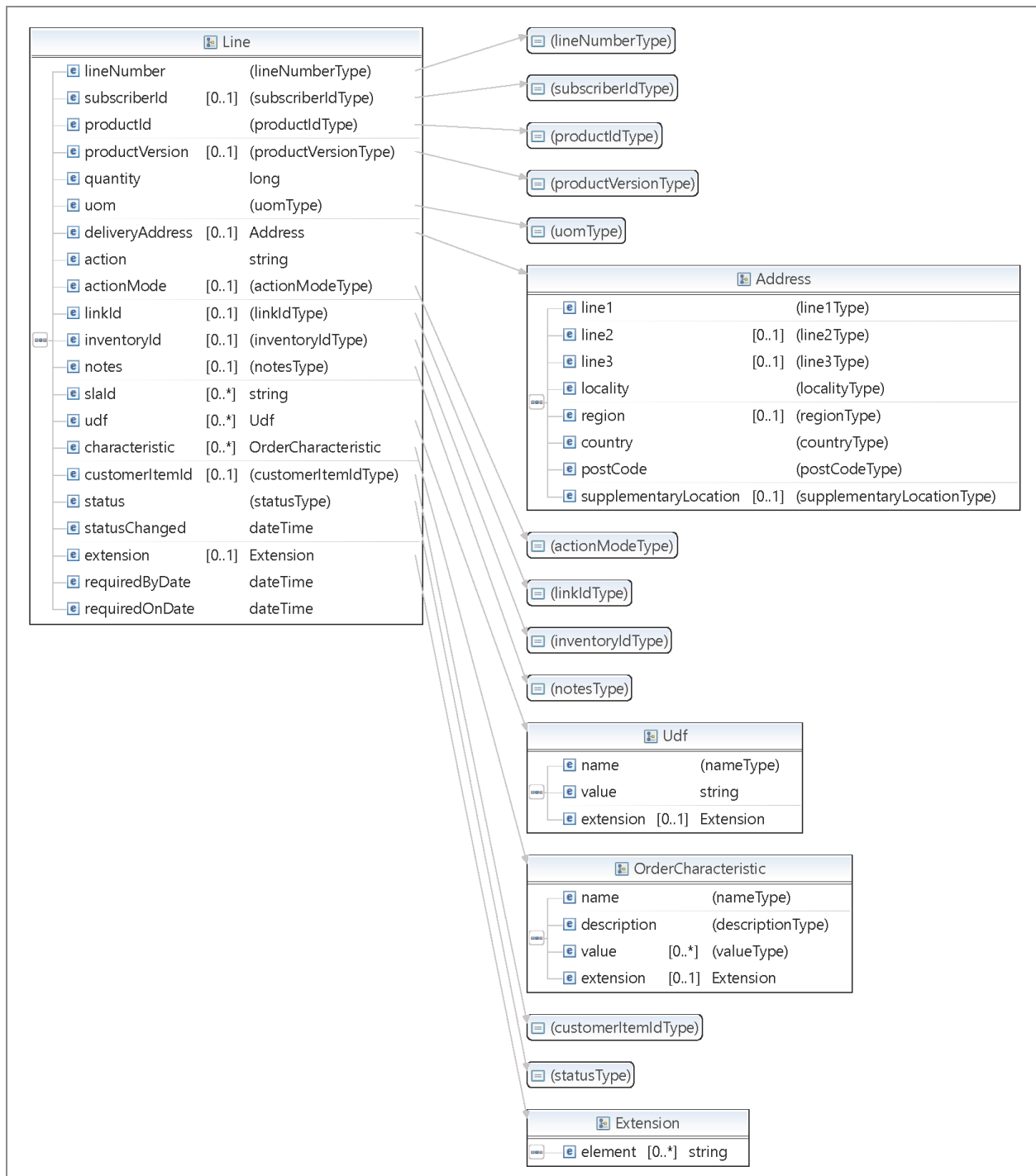
Element Name	Element Type	Description
	(Optional)	corporate orders. Either CustomerID or SubscriberID are necessary.
SubmittedDate	Date (Optional)	Date order was submitted.
SubmittedTime	Time (Optional)	The time order was submitted.
RequiredByDate	DateTime (Optional)	The date and time that a particular order is required to start on.
orderPriority	Integer (Optional)	The priority value assigned to order. The value ranges from 0-9. Note: To use orderPriority, schemas under \$OM_HOME/schemas/schema/oms must only be used.
ExpirationDate	Date (Optional)	The expiration date for the offer. Not relevant for orders.
ExpectedCompletionDate	DateTime (Optional)	The date order is expected to be completed.
InvoiceAddress	Type (Optional)	Invoice address.
InvoiceAddress/AddressLine1	String (Mandatory)	Invoice address line 1.
InvoiceAddress/AddressLine2	String	Invoice address line 2.

Element Name	Element Type	Description
	(Optional)	
InvoiceAddress/Locality	String (Mandatory)	Invoice address locality.
InvoiceAddress/Region	String (Optional)	Invoice address region.
InvoiceAddress/PostCode	String (Mandatory)	Invoice address postcode.
InvoiceAddress/Country	String (Mandatory)	Invoice address country.
DeliveryAddress	Type (Optional)	Delivery address.
DeliveryAddress/AddressLine1	String (Mandatory)	Delivery address line 1.
DeliveryAddress/AddressLine2	String (Optional)	Delivery address line 2.
DeliveryAddress/Locality	String (Mandatory)	Delivery address locality.
DeliveryAddress/Region	String (Optional)	Delivery address region.
DeliveryAddress/PostCode	String (Mandatory)	Delivery address postcode.

Element Name	Element Type	Description
DeliveryAddress/Country	String (Mandatory)	Delivery address country.
Notes	String (Optional)	Order notes.
SLAs	Type (Optional)	List of SLAs passed through to back-end systems. These SLAs do not apply within the order management.
SLAs/SLA	Type (*)	SLA type.
SLAs/SLA/SLAID	String (Mandatory)	SLA ID.
UDF	Type (*)	UDF type
UDF/Name	String (Mandatory)	Name of the attribute.
UDF/Value	String (Mandatory)	Value of the attribute.

OrderLine

This class is the attribute for the *OrderLine* object. It might be used across services and ID defined in JSON format.



The following order line actions are supported:

Provide	Install a new product or service.
---------	-----------------------------------

Update	Update an existing product or service.
Cease	Terminate an existing product or service.
Cancel	Cancel an outstanding order currently in-flight.

Line Element Details

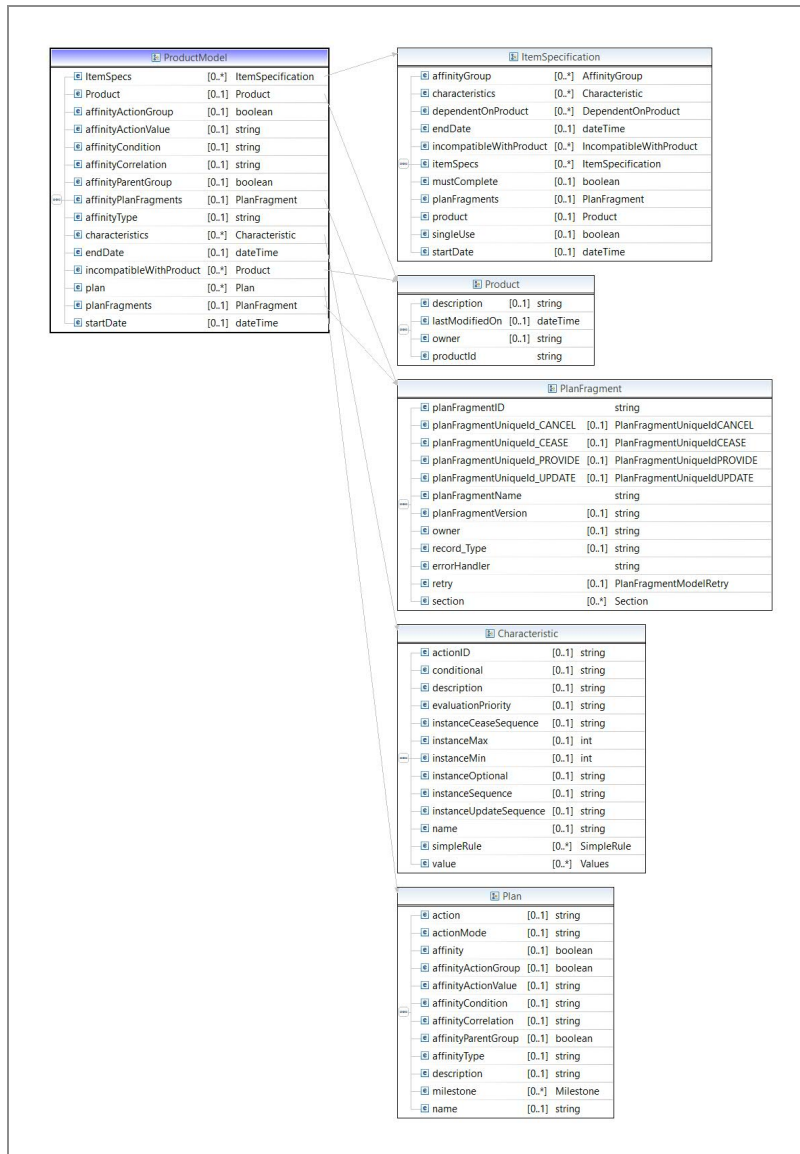
Element Name	Element Type	Description
lineNumber	String (Mandatory)	Order line number.
subscriberID	String (Optional)	Subscriber ID for this order line. Omit for orders for corporate customers not linked to an individual subscriber.
productID	String (Mandatory)	The ID of the product being ordered.
productVersion	String	Version ID of the product being ordered.
quantity	String (Mandatory)	Number of items required for each product in this line.
uom	String (Mandatory)	Unit of measure.
deliveryAddress	addressType (Mandatory)	Delivery address.
action	String (Mandatory)	Action is required. The valid values are: <ul style="list-style-type: none"> • PROVIDE • UPDATE • CEASE • CANCEL

Element Name	Element Type	Description
actionMode	String (Optional)	Action mode. This is optional supplemental information to go with OrderLineAction. Valid value is: <ul style="list-style-type: none"> • MOVE
requiredByDate	DateTime (Mandatory)	The date and time that a particular orderline is required to start on.
linkID	String (Optional)	Correlation ID used by order management to group order lines together for provisioning.
inventoryID	String (Optional)	Inventory ID for this product.
notes	String (Optional)	Order line notes.
slaID	String (Mandatory)	SLA ID. (SLAs passed through to back-end systems. These SLAs do not apply within order management itself.)
udf	String (Mandatory)	Value of the attribute. This must be a minimum of one character in length.
characteristics	Type (Optional)	List of characteristics.
customerItemID	String (Optional)	Customer item ID.
status	String (Optional)	Current order status.
statusChanged	DateTime	Last status update time of order. It is not required on

Element Name	Element Type	Description
	(Optional)	the time of order submission but to get order details it is available.
extension	List of Elements (Optional)	You might enter ANY elements at this point

ProductModel

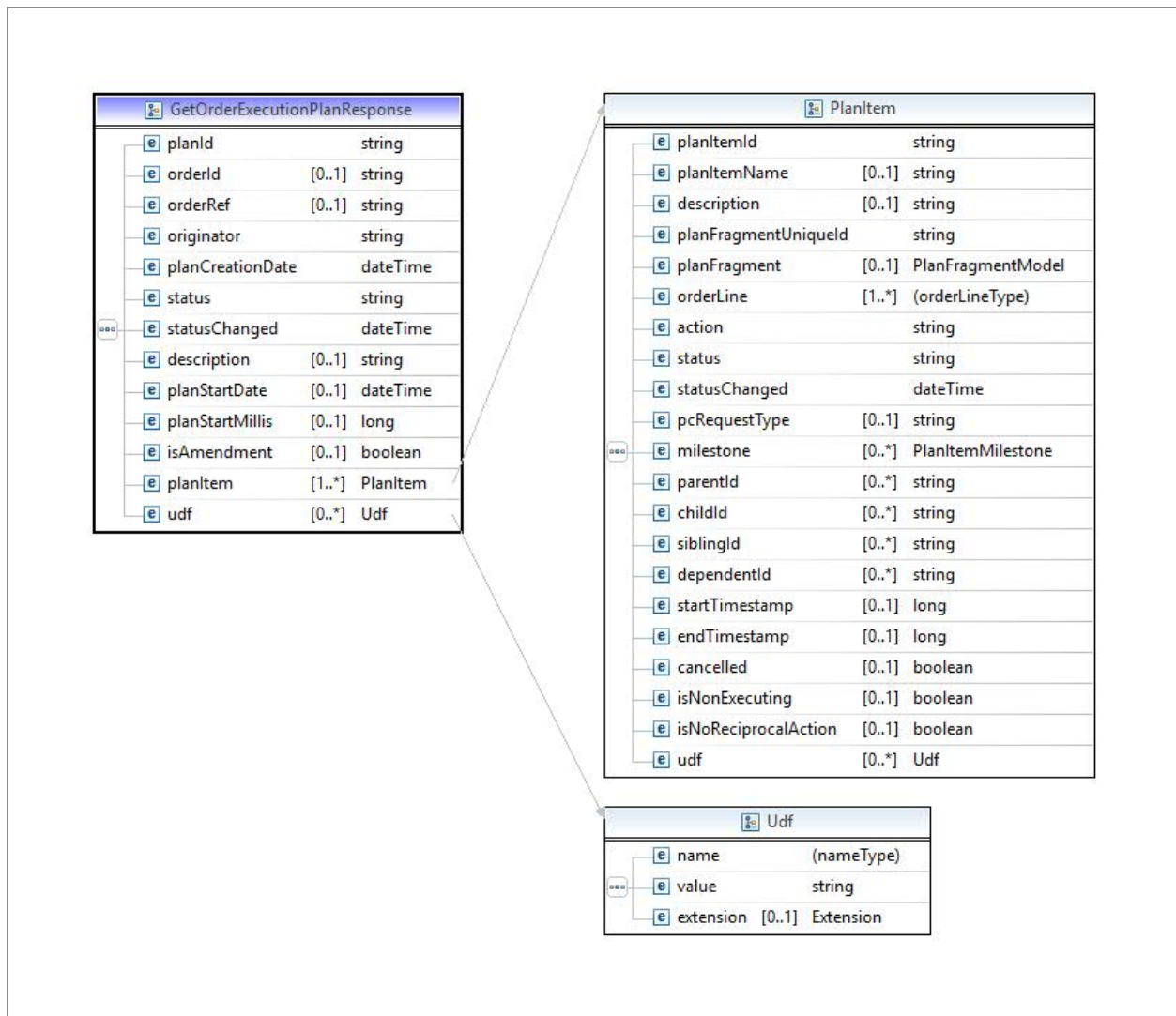
This class is the attribute for the *ProductModel* object. It might be used across services and ID defined in JSON format.



Execution Plan

This class is the attribute for the *plan* object. It might be used across services and ID defined in JSON format.

Here is the data model for the *planItem* object.



Plan and PlanItem Element Details

Element Name	Element Type	Description
planID	String (Mandatory)	Unique identifier for the plan.
orderId	String (Mandatory)	Order ID of the order to which this plan is related.
orderRef	String	Order reference ID.

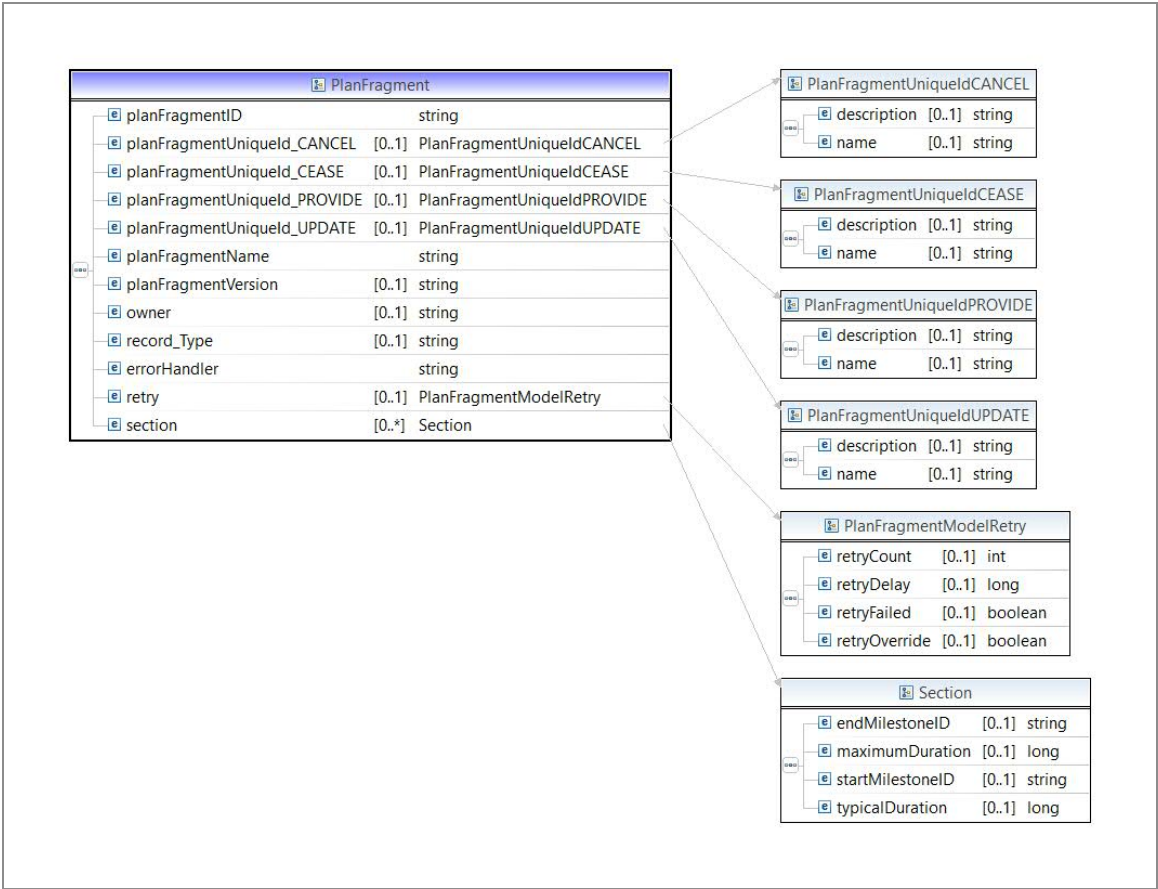
Element Name	Element Type	Description
planCreationDate	DateTime (Mandatory)	The date and time the plan was created.
status	String (Mandatory)	Status of the plan.
statusChanged	dateTime (Mandatory)	Date Time when the last status was changed.
description	String (Mandatory)	Description of the plan.
planStartDate	dateTime (Mandatory)	Start date and time for the plan.
planStartMillis	long	The Timestamp when the plan started executing.
isAmendment	Boolean	A flag identifying if the plan is an amendment plan.
planItem	planItemType	Composite object with information about the plan.
udf	udfSetType	Composite object with information related to User-Defined Field that is user-defined data.
planItemID	String	Unique Id identifying the plan Item.
planItemName	String	Name of the plan item.
planFragmentUniqueID	String	The id of the process component to invoke when executing the plan item.

Element Name	Element Type	Description
planFragment		The process component to invoke when executing the plan item.
orderLine	List	List of order lines for the corresponding plan item.
action	String	Action for the plan Item.
startMilestone		Start of the milestone of a plan.
milestone		Milestone of the plan.
parentID		Plan Item Ids on which this plan item depends.
childID	String	Plan Item Ids that are dependent on this plan item
siblingID	String	Plan Item Ids of sibling products for this plan item.
dependentID	String	Plan Item Ids of dependent products for this plan item.
startTimeStamp	long	Timestamp that indicates when the plan item started execution.
endTimeStamp	long	Timestamp that indicates when the plan item completed execution.
canceled	Boolean	Flag indicating if the plan was canceled.
isNonExecuting	Boolean	The flag indicates that a plan is in "NonExecuting" state, or no plan item is generated for the plan.
isNoReciprocalAction	Boolean	The flag indicates whether the plan is

Element Name	Element Type	Description
		reciprocal, or no plan item is reciprocated for this plan.

PlanFragment models

This class is the attribute for the *planFragment* models object. It might be used across services and ID defined in JSON format.



Authorization Service API Samples

- [Register tenant](#)
- [Update tenant information](#)
- [Get tenant information](#)
- [Delete tenant](#)
- [Create User](#)
- [Update User](#)
- [Get User](#)
- [Delete User](#)

Register tenant

This operation registers tenant information. This API can handle only single tenant registration at a time.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/tenant`

Parameter	Cardinality	Description
X-API-AppId	Mandatory	The application ID is used for getting the user details.
X-API-Key	Mandatory	This key is used for getting the user details.

 **Note:** Note that the `identityProviderType` value is case-sensitive.

The following sample is shown for RelationalSchema (Postgres/Oracle) `identityProviderType`:

```
{
  "tenantId": "TIBCO",
  "clientId": "order",
  "clientSecret": "order",
  "identityProviderType": "POSTGRES",
  "supportAlgorithm": "HS256",
  "signingKey": "100f4c1f-f333-4c25-bd8c-e4809722b6a7",
  "relationalSchema": {
    "dataSourceURL":
"jdbc:postgresql://localhost:5432/userdbll?currentSchema=userschemall",
    "dataSourceUserName": "userll",
    "dataSourcePassword": "userll"
  }
}
```

The following sample is shown for LDAP identityProviderType:

```
{
  "tenantId": "TIBCOLDAP",
  "clientId": "tibco-ldap-client",
  "clientSecret": "tibco-ldap-secret",
  "identityProviderType": "LDAP",
  "supportAlgorithm": "HS256",
  "signingKey": "100f4c1f-f333-4c25-bd8c-e4809722b6a7",
  "ldapSchema": {
    "ldapURLForDirectoryService": "string",
    "directoryServiceDomainName": "string",
    "directoryServiceRootDistinguishedName": "string"
  }
}
```

The following sample is shown for EXTERNAL identityProviderType:

i Note: In case of RS256 supportAlgorithm, jwkSetUrl and issuer are required. They are not needed in case of HS256.

```
{
  "tenantId": "string",
  "clientId": "string",
  "clientSecret": "string",
  "identityProviderType": "EXTERNAL",
  "signingKey": "100f4c1f-f333-4c25-bd8c-e4809722b6a7",
  "supportAlgorithm": "RS256",
  "jwkSetUrl": "string",
```

```
"issuer": "string",
"oidcSchema": {
  "authUrl": "string",
  "accessTokenUrl": "string",
  "scope": "string"
}
}
```

Update tenant information

This operation updates tenant information if the tenant details are already present in database.

Method: HTTP PUT

Endpoint: `http://<host_address>:<port_address>/v1/tenant`

Get User Parameters

Parameter	Cardinality	Description
X-API-AppId	Mandatory	The application ID is used for getting the user details.
X-API-Key	Mandatory	This key is used for getting the user details.

For more information on various identityProviderType scenarios, see the sample from the 'Register tenant' topic in *TIBCO® Order Management Web Services Guide*.

Get tenant information

This operation is used to get the tenant information if it is already present in database.

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/tenant`

Get User Parameters

Parameter	Cardinality	Description
tenantId	Mandatory	This is the TENANT value as stored in the users table in the database.
X-API-AppId	Mandatory	The application ID is used for getting the user details.
X-API-Key	Mandatory	This key is used for getting the user details.

Delete tenant

This operation deletes tenant information if the tenant is already present in database.

Method: HTTP DELETE

Endpoint: http://<host_address>:<port_address>/v1/tenant

Get User Parameters

Parameter	Cardinality	Description
tenantId	Mandatory	This is the TENANT value as stored in the users table in the database.
X-API-AppId	Mandatory	The application ID is used for getting the user details.
X-API-Key	Mandatory	This key is used for getting the user details.

Create User

This request is used to create users.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/user

Create User Parameters

Parameter		Cardinality	Description
tenantId		Mandatory	This is the TENANT value as stored in the users table in the database. If the tenantId is not present in the database, then a new TENANT is created.
X-API-AppId		Mandatory	The application ID is used for getting user details. The default ID is auth.
X-API-Key		Mandatory	This key is used for getting user details. The default ID is auth.
userInfo (Body)	enabled	Mandatory	The value can be true or false. true makes the user accessible through the configurator and Order Management System UI and false makes the user disable.
	password	Mandatory	The password to be used for the user.
	Username	Mandatory	It specifies the user name to be created or modified.
	userRoles	Mandatory	It assigns the role to the user. The default valid role values are ROLE_ADMIN and ROLE_USER. You can override the default roles if required.



Note: If the userName and tenantId provided in the request exist, then the user is modified with the provided values.

Example for the Create User request:

```
{
  "user": [
    {
      "password": "string",
      "userName": "string",
```

```

        "enabled": true,
        "userRoles": [
            "string"
        ]
    }
]
}

```

Update User

This request is used to update an existing user.

Method: HTTP PUT method

Endpoint: `http://<host_address>:<port_address>/v1/user`

Update User Parameters

Parameter		Cardinality	Description
tenantId		Mandatory	This is the TENANT value as stored in the users table in the database. If the tenantId is not present in the database, then a new TENANT is created.
X-API-AppId		Mandatory	The application ID is used for getting user details. The default ID is auth.
X-API-Key		Mandatory	This key is used for getting user details. The default ID is auth.
userInfo (Body)	enabled	Mandatory	The value can be true or false. true makes the user accessible through the configurator and Order Management System UI and false makes the user disable.
	password	Mandatory	The password to be used for the user.
	userName	Mandatory	It specifies the user name to be created or modified.

Parameter	Cardinality	Description
userRoles	Mandatory	It assigns the role to the user. The valid role values are ROLE_ADMIN and ROLE_USER.

i Note: If the userName and tenantId provided in the request exist, then the user is modified with the provided values.

Example for the Update User request:

```
{
  "user": [
    {
      "password": "string",
      "userName": "string",
      "enabled": true,
      "userRoles": [
        "string"
      ]
    }
  ]
}
```

Get User

This request is used to get the details of the existing user.

Method: HTTP GET method

Endpoint: http://<host_address>:<port_address>/v1/user

Get User Parameters

Parameter	Cardinality	Description
X-API-AppId	Mandatory	The application ID is used for getting the user details. The default ID is auth.
X-API-Key	Mandatory	This key is used for getting the user details. The default ID is

Parameter	Cardinality	Description
		auth.
tenantId	Mandatory	This is the TENANT value as stored in the users table in the database.
userId	Mandatory	This is the username value as stored in the users table in the database.

Delete User

This request is used to delete the existing user.

Method: HTTP DELETE method

Endpoint: `http://<host_address>:<port_address>/v1/user`

Delete User Parameters

Parameter	Cardinality	Description
tenantId	Mandatory	This is the tenant value as stored in the users table in the database.
X-API-AppId	Mandatory	The application ID is used for getting user details. The default ID is auth.
X-API-Key	Mandatory	This key is used for getting user details. The default ID is auth.
userInfo (Body)	userName Mandatory	It specifies the user name to be deleted.

Example for Delete User request:

```
[
  {
    "userName": "testuser",
```




```
}  
]
```

Configuration Service API Samples

- [Actuator APIs](#)
- [Application Metadata APIs](#)
- [Application Properties APIs](#)
- [Configuration Files APIs](#)
- [Configuration Notification APIs](#)

Actuator APIs

- [Actuator web endpoint 'refresh'](#)
- [Get 'loggers-name'](#)
- [Post 'loggers-name'](#)
- [Actuator root web endpoint](#)
- [Actuator web endpoint 'loggers'](#)
- [Actuator web endpoint 'health'](#)
- [Actuator web endpoint 'health-path'](#)

 **Note:** Health and Info APIs are not secured so that they can interact with other services easily.

Actuator web endpoint 'refresh'

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/management/refresh`

```
curl -X 'POST' \  
  'http://localhost:9090/management/refresh' \  
  -H 'accept: */*' \  
  -d ''
```

Get loggers name

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management/loggers/{name}

Parameters: name (mandatory)

You can find the possible name parameters in \$OM_HOME/roles/<service-name>/standalone/config/logback_<service-name>.xml file.

Sample:

```
curl -X 'GET' \  
  'http://localhost:9090/management/loggers/com.tibco.fom.bootstrap' \  
  -H 'accept: */*'
```

Post loggers name

Get Configuration

Method: HTTP POST

Endpoint: http://<host_address>:<port_address>/management/loggers/{name}

Parameters: name (mandatory)

You can find the possible name parameters in \$OM_HOME/roles/<service-name>/standalone/config/logback_<service-name>.xml file.

Sample:

```
curl -X 'POST' \
  'http://localhost:9090/management/loggers/com.tibco.fom.bootstrap' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "configuredLevel": "INFO, WARNING, ERROR"
  }
```

Actuator root web endpoint

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management

```
curl -X 'GET' \
  'http://localhost:9090/management' \
  -H 'accept: */*'
```

Actuator web endpoint 'loggers'

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management/loggers

Sample:

```
curl -X 'GET' \
  'http://localhost:9090/management/loggers' \
  -H 'accept: */*'
```

Actuator web endpoint 'health'

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management/health

```
curl -X 'GET' \
  'http://localhost:9090/management/health' \
  -H 'accept: */*'
```

Actuator web endpoint 'health-path'

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management/health/{name}

Parameters: name path parameter can have one of the following values:
readiness or liveness

Example:

```
curl -X 'GET' \
  'http://localhost:9090/management/health/readiness' \
  -H 'accept: */*'
```

Application Metadata APIs

Get supported files metadata

This API returns metadata information such as the list of supported files (both application properties files and configuration files), application Id, and application description for all applications supported by the Configurator.

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/v1/configuration/configFilesMetadata

Save supported files metadata

Method: HTTP POST

Endpoint: http://<host_address>:<port_address>/v1/configuration/configFilesMetadata

Example:

```
[
  {
    "applicationId": "string",
    "applicationDescription": "string",
    "appProperties": "string",
    "configurationFiles": [
      "string"
    ]
  }
]
```

Delete supported files metadata

Method: HTTP DELETE

Endpoint: http://<host_address>:<port_address>/v1/configuration/configFilesMetadata/{applicationId}

Parameters: applicationid (Select value from the dropdown menu)

Application Properties APIs

Get Application Properties by Application ID

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/v1/configuration/{applicationId}

Parameters: applicationId and applicationId (Select these mandatory values from the drop-down menu)

Update Application Properties for Application ID

Method: HTTP PUT

Endpoint: http://<host_address>:<port_address>/v1/configuration/{applicationId}

Parameters: applicationId and applicationId (mandatory values)

Example:

```
[
  {
    "propName": "string",
    "propValue": "string",
    "valueType": "Number, String, Boolean, Password",
    "allowedValues": [
      "string"
    ]
  }
]
```

Save Application Properties for Application ID

Method: HTTP POST

Endpoint: http://<host_address>:<port_address>/v1/configuration/{applicationId}

Parameters: applicationId and applicationId (mandatory values)

Example:

```
{
  "appDescription": "string",
  "categories": [
    {
      "categoryName": "string",
      "properties": [
        {
          "propName": "string",
          "propDescription": "string",
          "propValue": "string",
          "allowedValues": [
            "string"
          ],
          "valueType": "Number",
          "isTenantProperty": true
        }
      ]
    }
  ]
}
```

Delete Application Properties for Application ID

Method: HTTP DELETE

Endpoint: `http://<host_address>:<port_address>/v1/configuration/{applicationId}`

Parameters: applicationId and applicationId (mandatory values)

Example of deleteRequest:

```
[  
  "string"  
]
```

Replicate Tenant Properties

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/configuration/replicateTenantProperties/{sourceTenantId}`

Parameters: sourceTenantId and tenantID (mandatory values)

Get list of applications available with Configurator

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/configuration/availableApplications`

Configuration File APIs

Download Configuration File for Application ID

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/configuration/configFile/{applicationid}`

Parameters: applicationid (Select value from the drop-down menu)

Upload Configuration File for Application ID

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/configuration/configFile/{applicationid}`

Parameters: applicationid (Select value from the drop-down menu)

Request body: On the request body, browse and choose the required file to upload.

Delete Configuration Files for Application ID

Method: HTTP DELETE

Endpoint: `http://<host_address>:<port_address>/v1/configuration/{applicationid}`

Parameters: applicationid (Select value from the drop-down menu)

Example of deleteRequest:

```
[  
  "string"  
]
```

Configuration Notification APIs

GET Configurator notification API

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/configurator/events`

Archival Services API samples

This section provides Archival service API samples of TIBCO Order Management.

Operation Get Order Summary

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v2/orders/summary`

```
{
  "dateRange": {
    "startDate": "2023-01-19T07:11:05.392Z",
    "endDate": "2023-01-19T07:11:05.392Z"
  },
  "groupBy": "HOUR"
}
```

Operation Get Plans By Criteria

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v2/plans/criteria`

Parameter content type: application/json

The plan search criteria are as follows:

- planId
- orderId
- orderRef
- dateRange

- planStatusList
- planItemStatusList
- processComponentIdList
- processComponentNameList

**Note:**

Only the following combinations are allowed and no other combinations of search criteria are allowed:

- dateRange and planStatusList
- dateRange and planItemStatusList
- dateRange and processComponentIdList
- dateRange and processComponentNameList

Example of plansByCriteria:

```
{
  "planId": [
    "string"
  ],
  "orderId": [
    "string"
  ],
  "orderRef": [
    "string"
  ],
  "dateRange": {
    "startDate": "2023-10-09T19:43:19.763Z",
    "endDate": "2023-10-09T19:43:19.763Z"
  },
  "planStatusList": [
    "START, PENDING, EXECUTION, COMPLETE, CANCELLED, SUSPENDING,
    SUSPENDED"
  ],
  "planItemStatusList": [
    "string"
  ],
  "processComponentIdList": [
    "string"
  ],
  "processComponentNameList": [
```

```
    "string"  
  ],  
  "currentPage": 0,  
  "recordsPerPage": 0,  
  "filterOperator": "IN"  
}
```

Operation Get Orders By Criteria

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v2/orders/criteria`

Parameter content type: application/json

The order of search criteria are as follows:

- orderId
- orderRef
- customerId
- subscriberId
- dateRange
- status
- headerUdfs
- orderLineUdfs



Note:

Only the following combinations are allowed and no other combinations of search criteria are allowed:

- dateRange and status
- dateRange and headerUdfs
- dateRange and orderLineUdfs

Example of ordersByCriteria:

```
{
  "orderId": [
    "string"
  ],
  "orderRef": [
    "string"
  ],
  "customerID": [
    "string"
  ],
  "subscriberID": [
    "string"
  ],
  "dateRange": {
    "startDate": "2023-10-09T19:37:50.867Z",
    "endDate": "2023-10-09T19:37:50.867Z"
  },
  "status": [
    "START, BLOCKED, PENDING, OPD, OPDERROR, EXECUTION, COMPLETE,
    CANCELLED, SUSPENDED, SUSPENDING, PREQUALIFICATIONFAILED, WITHDRAWN,
    FEASIBILITY"
  ],
  "headerUdfs": [
    {
      "name": "string",
      "value": "string"
    }
  ],
  "orderLineUdfs": [
    {
      "name": "string",
      "value": "string"
    }
  ],
  "currentPage": 0,
  "recordsPerPage": 0,
  "filterOperator": "IN"
}
```

Operation Get Plan Item By Criteria

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v2/orders/planItems`

The required field parameters are given in the following list:

- orderIds
- pageNumber
- pageSize

i Note: The page size that you provide in the request must be more than 0 and no greater than 20.

Example of GetPlanItemByCriteria:

```
{
  "pageNumber": 1,
  "pageSize": 1,
  "totalPages": 159,
  "totalPlanItems": 159,
  "orders": [
    {
      "orderId": "22aug_02",
      "planId": "b695e962-e02f-497c-b7ed-caf95fef1663",
      "planItems": [
        { "planItemId": "1", "action": "CANCEL", "status": "CANCELLED",
          "planItemName": "NON_EXECUTING" }
      ]
    }
  ]
}
```

Operation Get Pie Chart Details By Criteria

Method: HTTP GET method

Endpoint: http://<host_address>:<port_address>/v2/orders/pieChartDetails

The required field parameters are given in the following list:

- orderIds
- category
The category can be anything from the following list:

- Plan Fragment
- Order Status
- Order Line Status
- Plan Item Status
- Milestone Status
- Delivery Postal Code
- Delivery Locality
- Delivery Region
- Order Line Action

Example of GetPieChartDetailsByCriteria:

```
{
  "orderIds": ["12345", "67890"],
  "category": "Plan Fragment",
  "details": [
    {
      "name": "Fragment A",
      "value": 3
    },
    {
      "name": "Fragment B",
      "value": 2
    }
  ]
}
```

For the category mentioned here, the output includes:

- a name representing the actual plan fragment name
- a value indicating the frequency of that particular plan fragment in the request order list

Operation Get Order Details

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/orders/detail`

Parameters

Name	Type	Value
orderId	string (header)	Order ID value
orderRef	string (header)	Order Ref value

Example:

```
{
  "orderId": "28aug25_10",
  "orderRef": "28aug25_10",
  "header": {
    "submittedDate": 1726029147.763,
    "description": "desc",
    "customerID": "pooja",
    "orderPriority": 4,
    "invoiceAddress": {
      "line1": "1",
      "line2": "2",
      "line3": "3",
      "locality": "PUNE",
      "region": "MH",
      "country": "IN",
      "postCode": "411",
      "supplementaryLocation": "Malegaon"
    },
    "deliveryAddress": {
      "line1": "34",
      "line2": "eff",
      "line3": "wefwe",
      "locality": "effe",
      "region": "gh",
      "country": "1",
      "postCode": "34",
      "supplementaryLocation": "fewf"
    },
    "udf": [
      {
        "name": "country",
        "value": "India"
      }
    ],
    "status": "COMPLETE",
    "statusChanged": 1726029146.89,
```



```

    "requiredByDate": 1685002200
  },
  "lines": [
    {
      "lineNumber": "1",
      "subscriberID": "mona",
      "productID": "PO_TV",
      "productVersion": "1",
      "quantity": 1,
      "uom": "uom",
      "deliveryAddress": {
        "line1": "34",
        "line2": "eff",
        "line3": "wefwfwefe",
        "locality": "ewfewef",
        "region": "ghwcdfe fdew",
        "country": "1",
        "postCode": "34",
        "supplementaryLocation": "fewfmntppplpl"
      },
      "action": "PROVIDE",
      "udf": [
        {
          "name": "daughter",
          "value": "prisha"
        }
      ],
      "status": "COMPLETE",
      "statusChanged": 1726029146.89
    }
  ]
}

```

Operation Get Plan Details

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/plans/detail`

Parameters

Name	Type	Value
orderId	string (header)	Order ID value
orderRef	string (header)	Order Ref value
planId	string (header)	Plan ID value

Example:

```
{
  "planId": "3b7d068c-67a8-4769-ae27-4d014d14b7c9",
  "orderId": "28aug25_10",
  "orderRef": "28aug25_10",
  "originator": "Orchestrator",
  "planCreationDate": 1726029148.132,
  "status": "COMPLETE",
  "statusChanged": 1726029148.124,
  "description": "desc",
  "planStartDate": 1726029148.132,
  "planStartMillis": 1726029148132,
  "isAmendment": false,
  "planItems": [
    {
      "planItemID": "3",
      "planItemName": "EP_PF_CFS_TV_Provide",
      "description": "EP_PF_CFS_TV_Provide",
      "planFragmentUniqueId": "EP_PF_CFS_TV_Provide",
      "parentIds": [
        "2"
      ],
      "isNonExecuting": false,
      "udfs": [
        {
          "flavor": "input",
          "name": "daughter",
          "value": "prisha",
          "originalValue": "prisha"
        },
        {
          "flavor": "config",
          "name": "PRODUCTID",

```

```

        "value": "CFS_TV",
        "originalValue": "CFS_TV"
    },
    {
        "flavor": "config",
        "name": "EOL",
        "value": "N",
        "originalValue": "N"
    },
    {
        "flavor": "config",
        "name": "Action",
        "value": "PROVIDE",
        "originalValue": "PROVIDE"
    },
    {
        "flavor": "config",
        "name": "ORDERLINE",
        "value": "1",
        "originalValue": "1"
    },
    {
        "flavor": "config",
        "name": "M_EP_UDFS",
        "value": "3",
        "originalValue": "3"
    },
    {
        "flavor": "config",
        "name": "itemSpecId",
        "value": "1",
        "originalValue": "1"
    },
    {
        "flavor": "config",
        "name": "TASKID",
        "value": "3",
        "originalValue": "3"
    },
    {
        "flavor": "config",
        "name": "RECORD_TYPE",
        "value": "CFS",
        "originalValue": "CFS"
    },
    {

```

```

        "flavor": "config",
        "name": "PARENT_PRODUCT_ID",
        "value": "Product_TV",
        "originalValue": "Product_TV"
    },
    {
        "flavor": "config",
        "name": "GLOBAL_PRODUCT_NAME",
        "value": "CFS_TV",
        "originalValue": "CFS_TV"
    }
],
"planFragment": {
    "planFragmentID": "EP_PF_CFS_TV_Provide",
    "planFragmentName": "EP_PF_CFS_TV_Provide",
    "planFragmentVersion": "1",
    "record_Type": "Process",
    "errorHandler": "EP_ERROR",
    "retry": {
        "retryOverride": false,
        "retryFailed": false,
        "retryCount": 0,
        "retryDelay": 0
    },
    "section": [
        {
            "startMilestoneID": "START",
            "endMilestoneID": "END",
            "typicalDuration": 60000,
            "maximumDuration": 120000
        }
    ]
},
"orderLines": [
    {
        "orderLineNumber": "1",
        "eol": false
    }
],
"action": "PROVIDE",
"status": "COMPLETE",
"statusChanged": 1726029148.128,
"milestones": [
    {
        "milestoneID": "START",
        "ready": false,

```

```

        "status": "COMPLETE",
        "statusChanged": 1726029148.129
    },
    {
        "milestoneID": "END",
        "ready": false,
        "status": "COMPLETE",
        "statusChanged": 1726029148.129
    }
]
},
{
    "planItemID": "1",
    "planItemName": "EP_PF_PO_TV_Provide",
    "description": "EP_PF_PO_TV_Provide",
    "planFragmentUniqueId": "EP_PF_PO_TV_Provide",
    "childIds": [
        "2"
    ],
    "isNonExecuting": false,
    "udfs": [
        {
            "flavor": "config",
            "name": "TASKID",
            "value": "1",
            "originalValue": "1"
        },
        {
            "flavor": "config",
            "name": "GLOBAL_PRODUCT_NAME",
            "value": "PO_TV",
            "originalValue": "PO_TV"
        },
        {
            "flavor": "config",
            "name": "Action",
            "value": "PROVIDE",
            "originalValue": "PROVIDE"
        },
        {
            "flavor": "config",
            "name": "RECORD_TYPE",
            "value": "PO",
            "originalValue": "PO"
        }
    ]
}

```

```

        "flavor": "config",
        "name": "ORDERLINE",
        "value": "1",
        "originalValue": "1"
    },
    {
        "flavor": "config",
        "name": "PRODUCTID",
        "value": "PO_TV",
        "originalValue": "PO_TV"
    },
    {
        "flavor": "config",
        "name": "M_EP_UDFS",
        "value": "1",
        "originalValue": "1"
    },
    {
        "flavor": "input",
        "name": "daughter",
        "value": "prisha",
        "originalValue": "prisha"
    },
    {
        "flavor": "config",
        "name": "EOL",
        "value": "Y",
        "originalValue": "Y"
    },
    {
        "flavor": "config",
        "name": "itemSpecId",
        "value": "1",
        "originalValue": "1"
    }
],
"planFragment": {
    "planFragmentID": "EP_PF_PO_TV_Provide",
    "planFragmentName": "EP_PF_PO_TV_Provide",
    "planFragmentVersion": "1",
    "record_Type": "Process",
    "errorHandler": "EP_ERROR",
    "retry": {
        "retryOverride": false,
        "retryFailed": false,
        "retryCount": 0,
    }
}

```

```

    "retryDelay": 0
  },
  "section": [
    {
      "startMilestoneID": "START",
      "endMilestoneID": "END",
      "typicalDuration": 60000,
      "maximumDuration": 120000
    }
  ]
},
"orderLines": [
  {
    "orderLineNumber": "1",
    "eol": true
  }
],
"action": "PROVIDE",
"status": "COMPLETE",
"statusChanged": 1726029148.129,
"milestones": [
  {
    "milestoneID": "START",
    "ready": false,
    "status": "COMPLETE",
    "statusChanged": 1726029148.129,
    "dependencies": [
      {
        "dependencyID": "ee80a26a-8e0b-406a-bca0-e85666a8601f",
        "type": "Point",
        "ready": false,
        "status": "COMPLETE",
        "statusChanged": 1726029148.131,
        "timeDelta": 0,
        "point": {
          "planItemID": "2",
          "planID": "3b7d068c-67a8-4769-ae27-4d014d14b7c9",
          "milestoneID": "END",
          "end": "null"
        }
      }
    ]
  }
],
{
  "milestoneID": "END",
  "ready": false,

```

```

        "status": "COMPLETE",
        "statusChanged": 1726029148.129
    }
]
},
{
    "planItemID": "2",
    "planItemName": "EP_PF_Product_TV_Provide",
    "description": "EP_PF_Product_TV_Provide",
    "planFragmentUniqueId": "EP_PF_Product_TV_Provide",
    "parentIds": [
        "1"
    ],
    "childIds": [
        "3"
    ],
    "isNonExecuting": false,
    "udfs": [
        {
            "flavor": "config",
            "name": "Action",
            "value": "PROVIDE",
            "originalValue": "PROVIDE"
        },
        {
            "flavor": "config",
            "name": "EOL",
            "value": "N",
            "originalValue": "N"
        },
        {
            "flavor": "input",
            "name": "daughter",
            "value": "prisha",
            "originalValue": "prisha"
        },
        {
            "flavor": "config",
            "name": "itemSpecId",
            "value": "1",
            "originalValue": "1"
        },
        {
            "flavor": "config",
            "name": "TASKID",
            "value": "2",

```



```

        "originalValue": "2"
    },
    {
        "flavor": "config",
        "name": "PRODUCTID",
        "value": "Product_TV",
        "originalValue": "Product_TV"
    },
    {
        "flavor": "config",
        "name": "GLOBAL_PRODUCT_NAME",
        "value": "Product_TV",
        "originalValue": "Product_TV"
    },
    {
        "flavor": "config",
        "name": "RECORD_TYPE",
        "value": "DEVICE",
        "originalValue": "DEVICE"
    },
    {
        "flavor": "config",
        "name": "ORDERLINE",
        "value": "1",
        "originalValue": "1"
    },
    {
        "flavor": "config",
        "name": "M_EP_UDFS",
        "value": "2",
        "originalValue": "2"
    },
    {
        "flavor": "config",
        "name": "PARENT_PRODUCT_ID",
        "value": "PO_TV",
        "originalValue": "PO_TV"
    }
],
"planFragment": {
    "planFragmentID": "EP_PF_Product_TV_Provide",
    "planFragmentName": "EP_PF_Product_TV_Provide",
    "planFragmentVersion": "1",
    "record_Type": "Process",
    "errorHandler": "EP_ERROR",
    "retry": {

```

```

        "retryOverride": false,
        "retryFailed": false,
        "retryCount": 0,
        "retryDelay": 0
    },
    "section": [
        {
            "startMilestoneID": "START",
            "endMilestoneID": "END",
            "typicalDuration": 60000,
            "maximumDuration": 120000
        }
    ]
},
"orderLines": [
    {
        "orderLineNumber": "1",
        "eol": false
    }
],
"action": "PROVIDE",
"status": "COMPLETE",
"statusChanged": 1726029148.132,
"milestones": [
    {
        "milestoneID": "START",
        "ready": false,
        "status": "COMPLETE",
        "statusChanged": 1726029148.132,
        "dependencies": [
            {
                "dependencyID": "ab969363-848d-450a-be71-993a338be169",
                "type": "Point",
                "ready": false,
                "status": "COMPLETE",
                "statusChanged": 1726029148.132,
                "timeDelta": 0,
                "point": {
                    "planItemID": "3",
                    "planID": "3b7d068c-67a8-4769-ae27-4d014d14b7c9",
                    "milestoneID": "END",
                    "end": "null"
                }
            }
        ]
    }
],
},

```

```

    {
      "milestoneID": "END",
      "ready": false,
      "status": "COMPLETE",
      "statusChanged": 1726029148.132
    }
  ]
},
"udves": [
  {
    "flavor": "input",
    "name": "country",
    "value": "India",
    "originalValue": "India"
  }
]

```

Operation Get Bulk Action

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/orders/bulk-action-job`

Parameters: `jobId` (mandatory)

Operation Get All Bulk Action

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/orders/bulk-action-job/all`

Operation Get Audit Trail

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/orders/audit`

Parameters

Name	Type	Value
orderId (mandatory)	string (header)	Order ID value
orderRef (mandatory)	string (header)	Order Ref value
Note: You must provide at least one of these parameters.		
includePlan (optional)	Boolean (header)	true or false
includePlanItem (optional)	Boolean (header)	true or false
sortByTime	Boolean (header)	true or false

Operation Submit Audit Trail

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/order/audit

Example:

```
{
  "auditTrails": [
    {
      "orderId": "string",
      "orderRef": "string",
      "order": {
        "auditMessage": "string",
        "origin": "string",
        "dateCreated": "2024-07-02T18:49:47.959Z"
      },
      "plan": {
        "planId": "string",
        "auditMessage": "string",
        "origin": "string",
        "dateCreated": "2024-07-02T18:49:47.959Z"
      },
      "planItem": {
```

```

        "planItemId": "string",
        "auditMessage": "string",
        "origin": "string",
        "dateCreated": "2024-07-02T18:49:47.959Z"
    },
    "orderLine": {
        "orderLineNumber": "string",
        "auditMessage": "string",
        "origin": "string",
        "dateCreated": "2024-07-02T18:49:47.959Z"
    },
    "generic": {
        "planId": "string",
        "planItemId": "string",
        "orderLineNumber": "string",
        "auditMessage": "string",
        "origin": "string",
        "dateCreated": "2024-07-02T18:49:47.959Z"
    }
}
]
}

```

Operation Purge Order

The Purge Order API is used to purge orders from both the Archival and the Orchestrator service.

Note: Only the orders that are in the final state (example: CANCELED, COMPLETE) can be purged.

Method: HTTP DELETE method

Endpoint: `http://<host_address>:<port_address>/v1/orders/purge`

```

{
  "orderId": [
    "string"
  ],
  "orderRef": [
    "string"
  ]
}

```

```
],  
"dateRange": {  
  "startDate": "2023-01-19T07:15:13.079Z",  
  "endDate": "2023-01-19T07:15:13.079Z"  
}  
}
```

Operation Get Saved Searches

This API is used to view the existing search criteria those are shared.

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/search`

Parameters

Name	Type	Value
UserName	string (required)	Owner, who creates a search criteria.
tenantID	string (required)	Tenant ID of the search criteria owner.

Operation Update Saved Searches

This API is used to update a search criteria.

Method: HTTP PUT

Endpoint: `http://<host_address>:<port_address>/v1/search`

Parameters

Name	Type	Value
UserName	string (required)	Owner, who creates a search

Name	Type	Value
		criteria.
tenantID	string (required)	Tenant ID of the search criteria owner.
Request body	body	name: Name of the search criteria. visibility: Private or Shared. If it is shared, visible to everyone. criteria: Search string.

Operation Saved Searches

This API is used to create a new search criteria.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/search`

Parameters

Name	Type	Value
UserName	string (required)	Owner, who creates a search criteria.
tenantID	string (required)	Tenant ID of the search criteria owner.
Request body	body	name: Name of the search criteria. visibility: Private or Shared. If set to Shared, it is visible to everyone. criteria: Search string.

Operation Delete Saved Searches

This API is used to delete a search criteria.

Method: HTTP DELETE

Endpoint: `http://<host_address>:<port_address>/v1/search`

Parameters

Name	Type	Value
name	string (required)	Name of the search criteria.
UserName	string (required)	Owner, who creates a search criteria.
tenantID	string (required)	Tenant ID of the search criteria owner.

Orchestrator API samples

This section provides Orchestrator API samples of TIBCO Order Management.

Feasibility Reply

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/order/feasibilityReply`

Parameter content type: application/json

Sample of feasibilityReply value is given below:

```
"orderId": "string",
"orderRef": "string",
"completed": true,
"passed": true,
"message": [
  {
    "lineNumber": "string",
    "type": "string",
    "code": "string",
    "description": "string",
    "udfs": [
      {
        "name": "string",
        "value": "string"
      }
    ]
  }
],
"businessTransactionID": "string",
}
```

Get Order Details

The GetOrderDetails operation of a particular order based on the orderId and optionally include order request and order amendments as part of response.

Method: HTTP GET method

Endpoint: http://<host_address>:<port>/v1/order

Parameters

Name	Type	Description	Value
ExternalBusinessTransactionID (optional)	String (optional)	Transaction ID sent by the client.	Transaction ID sent by the client.
includeAmendment (optional)	Boolean (query)	This flag determines whether or not to include order amendment in response.	true or false
includeOrderRequest (optional)	Boolean (query)	This flag determines whether or not to include the original request in the response.	true or false
orderId (mandatory)	string (query)	Order identification number.	orderid - Order identification number. <div>Note: The value of orderId must not contain ":"</div>
orderRef (optional)	string (query)	Order reference number.	Order reference number.

Name	Type	Description	Value
			Note: The value of orderRef must not contain ":"

Get Order Status

Use this API to view the status of orders.

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/v1/order/status

Input parameters:

orderIdList: <string>

i **Note:** You can add multiple order Ids by clicking the **Add string item** button.

Submit Order

i **Note:** The fulfillment validation is done for the REST API as per the conditions mentioned in 'Order Types' section in the *TIBCO® Order Management User Guide*.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/order

Input submitOrderRequest is mandatory for this operation.

i **Note:** The value of orderId and orderRef must not contain ":"

Parameter content type: application/json

Sample of submitOrderRequest value is given below:

```
{
  "externalBusinessTransactionID": "string",
  "orderRequest": {
    "extension": {
      "any": [
        "string"
      ]
    },
    "header": {
      "customerID": "string",
      "deliveryAddress": {
        "country": "string",
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "postCode": "string",
        "region": "string",
        "supplementaryLocation": "string"
      },
      "description": "string",
      "extension": {
        "any": [
          "string"
        ]
      }
    },
    "invoiceAddress": {
      "country": "string",
      "line1": "string",
      "line2": "string",
      "line3": "string",
      "locality": "string",
      "postCode": "string",
      "region": "string",
      "supplementaryLocation": "string"
    },
    "notes": "string",
    "orderPriority": 0,
    "requiredByDate": "2019-06-19T09:04:59.811Z",
    "requiredOnDate": "2019-06-19T09:04:59.811Z",
    "slaID": [
      "string"
    ],
    "subscriberID": "string",
    "udf": [
```

```

    {
      "extension": {
        "any": [
          "string"
        ]
      },
      "name": "string",
      "value": "string"
    }
  ]
},
"line": [
  {
    "action": "string",
    "actionMode": "string",
    "characteristic": [
      {
        "description": "string",
        "extension": {
          "any": [
            "string"
          ]
        },
        "name": "string",
        "value": [
          {
            "extension": {
              "any": [
                "string"
              ]
            },
            "name": "string",
            "type": "string",
            "value": "string",
            "valueFrom": "string",
            "valueTo": "string"
          }
        ]
      }
    ]
  }
],
"customerItemID": "string",
"deliveryAddress": {
  "country": "string",
  "line1": "string",
  "line2": "string",
  "line3": "string",

```

```

        "locality": "string",
        "postCode": "string",
        "region": "string",
        "supplementaryLocation": "string"
    },
    "extension": {
        "any": [
            "string"
        ]
    },
    "inventoryID": "string",
    "lineNumber": "string",
    "linkID": "string",
    "notes": "string",
    "productID": "string",
    "productVersion": "string",
    "quantity": 0,
    "requiredByDate": "2019-06-19T09:04:59.811Z",
    "requiredOnDate": "2019-06-19T09:04:59.811Z",
    "slaID": [
        "string"
    ],
    "subscriberID": "string",
    "udf": [
        {
            "extension": {
                "any": [
                    "string"
                ]
            },
            "name": "string",
            "value": "string"
        }
    ],
    "uom": "string"
    },
    "orderRef": "string"
}

```

Withdraw Order Request

Method: HTTP DELETE method

Endpoint: http://<host_address>:<port_address>/v1/order

Parameter content type: application/json

Sample of withdrawOrderRequest value is given below:

```
{
  "externalBusinessTransactionID": "string",
  "orderID": "string",
  "orderRef": "string"
}
```

Activate Order

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/order/activate

Parameter content type: application/json

Sample of activateOrderRequest value is given below:

```
{
  "externalBusinessTransactionID": "string",
  "orderID": "string",
  "orderRef": "string"
}
```

Cancel Order

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/order/cancel

Parameter content type: application/json

Sample of cancelOrderRequest value is given below:

```
{
  "externalBusinessTransactionID": "string",
```

```
"orderId": "string",  
"orderRef": "string",  
"rollback": true  
}
```

Suspend Order

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/order/suspend

Parameter content type: application/json

Sample of suspendOrderRequest value is given below:

```
{  
  "externalBusinessTransactionID": "string",  
  "orderId": "string",  
  "orderRef": "string"  
}
```

Bulk Action

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/order/bulkaction

Parameter content type: application/json

Sample of bulkOrderActionRequest value is given below:

```
{  
  "action": "string",  
  "businessTransactionID": "string",  
  "orderRef": [  
    "string"  
  ],  
  "orderId": [  
    "string"  
  ]  
}
```



```
]
}
```

Plan Item Bulk Error Handler

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/planitem/bulkErrorHandlerreply`

Parameter content type: application/json

Sample of `planItemFailedReplyList` value is given below:

```
[
  {
    "orderId": "string",
    "orderRef": "string",
    "planID": "string",
    "planItemID": "string",
    "retry": {},
    "resume": {},
    "complete": {},
    "message": [
      {
        "code": "string",
        "description": "string",
        "lineNumber": "string",
        "type": "string",
        "udfs": [
          {
            "name": "string",
            "value": "string"
          }
        ]
      }
    ],
    "businessTransactionID": "string",
  }
]
```

Plan Item Error Handler

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/planitem/errorHandlerreply

Parameter content type: application/json

Sample of planItemFailedReply value is given below:

```
{
  "orderId": "string",
  "orderRef": "string",
  "completed": true,
  "passed": true,
  "message": [
    {
      "code": "string",
      "description": "string",
      "lineNumber": "string",
      "type": "string",
      "udfs": [
        {
          "name": "string",
          "value": "string"
        }
      ]
    }
  ],
  "businessTransactionID": "string",
}
```

PlanItem Execute Response

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/planitem/executionreply

Parameter content type: application/json

Sample of planItemExecuteReply value is given below:

i Note: Database does not store the number of retries for plan items, so the retry count for plan item is not available.

```
{
  "businessTransactionId": "string",
  "cancelled": true,
  "completed": true,
  "messages": [
    {
      "code": "string",
      "description": "string",
      "lineNumber": "string",
      "type": "string",
      "udfs": [
        {
          "name": "string",
          "value": "string"
        }
      ]
    }
  ],
  "orderId": "string",
  "orderRef": "string",
  "planId": "string",
  "planItemId": "string",
  "success": true,
}
```

Plan Item Suspend Response

Use this API to initiate plan item suspend reply.

Method: HTTP POST

Endpoint: http://<host_address>:<port_address>/v1/planitem/suspendreply

Input parameters:

A sample of a planItemSuspendReply value is given as follows:

```
{
  "resultStatus": {
```

```

    "code": "string",
    "component": "string",
    "deployment": "string",
    "message": "string",
    "operation": "string",
    "service": "string",
    "severity": "string"
  },
  "orderId": "string",
  "orderRef": "string",
  "planID": "string",
  "planItemID": "string",
  "completed": true,
  "success": true,
  "businessTransactionID": "string",
  "correlationID": "string"
}

```



Note: You can provide any one parameter from orderId, orderRef, or planID in the request body.

Milestone Notify Request

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/planitem/milestonenotify

Parameter content type: application/json

Sample of milestoneNotifyRequest value is given below:

```

{
  "orderId": "string",
  "orderRef": "string",
  "planID": "string",
  "planItemID": "string",
  "milestoneID": "string",
  "businessTransactionID": "string",
}

```

i Note: You can provide any one parameter from orderID, orderRef, or planID in the request body.

PreQualification Failed Reply

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/order/preQualificationFailedReply`

Parameter content type: application/json

Sample of preQualificationFailedReply value is given below:

```
{
  "message": [
    {
      "lineNumber": "string",
      "type": "string",
      "code": "string",
      "description": "string",
      "udfs": [
        {
          "name": "string",
          "value": "string"
        }
      ]
    }
  ],
  "orderID": "string",
  "orderRef": "string",
  "action": "RETRYOPD",
  "businessTransactionID": "string",
  "correlationID": "string"
}
```

Get Order Messages

Use this API to view plan item level and order level messages.

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/order/messages`

Input parameters :

orderId: <string>

orderRef: <string>

i **Note:** Any one from orderId or orderRef is mandatory.

Operation Get Order State Machine XML

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/order/scxml`

Parameters

Name	Type	Description
orderRef	string (query)	Order reference number.
orderId	string (query)	Order identification number.

Operation Purge Order

i **Note:** Only the orders that are in the final state (for example, CANCELED, COMPLETE) can be purged.

Method: HTTP DELETE

Endpoint: `http://<host_address>:<port_address>/v1/order/purgeOrder`

orderid is the mandatory parameter.

Operation GetOrderExecutionPlan

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/plan`

Parameters

Name	Type	Description
orderRef	string (query)	Order reference number.
orderId	string (query)	Order identification number.
planId	string (query)	Plan identification number.
ExternalBusinessTransactionID	String (query)	Transaction ID sent by the client.

Operation SubmitOrderExecutionPlan

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/plan`

Example:

```
{
  "planId": "string",
  "orderId": "string",
  "orderRef": "string",
  "originator": "string",
  "planCreationDate": "2023-02-04T18:16:04.927Z",
  "status": "string",
  "statusChanged": "2023-02-04T18:16:04.927Z",
  "description": "string",
  "planStartDate": "2023-02-04T18:16:04.927Z",
  "planStartMillis": 0,
```

```

"isAmendment": true,
"planItems": [
  {
    "planItemMessages": [
      {
        "planItemId": "string",
        "messageType": "string",
        "messageList": [
          {
            "lineNumber": "string",
            "type": "string",
            "code": "string",
            "description": "string",
            "udfs": [
              {
                "name": "string",
                "value": "string"
              }
            ]
          }
        ]
      }
    ]
  },
  "planItemID": "Unique identifier for the plan item within the
plan.",
  "planItemName": "Identifier for the plan item name(Product
Description) within the plan.",
  "description": "Description for the plan item.",
  "planFragmentUniqueId": "The id of the process component to invoke
when executing the plan item.",
  "parentIds": "Plan Item Ids which are dependent on this plan
item",
  "childIds": "Plan Item Ids on which this plan item depends",
  "siblingIds": "Plan Item Ids of sibling products for this plan
item",
  "dependentIds": "Plan Item Ids of dependent products for this plan
item",
  "startTimestamp": 0,
  "endTimestamp": 0,
  "cancelled": true,
  "isNonExecuting": true,
  "isNoReciprocalAction": true,
  "udfs": [
    {
      "evaluationPriority": "string",
      "type": "string",
      "flavor": "string",

```



```

        "name": "string",
        "value": "string",
        "originalValue": "string"
    }
],
"planFragment": {
    "acknowledgeMode": "REST, MESSAGING",
    "planFragmentID": "string",
    "planFragmentName": "string",
    "planFragmentVersion": "string",
    "owner": "string",
    "record_Type": "string",
    "errorHandler": "string",
    "retry": {
        "retryOverride": true,
        "retryFailed": true,
        "retryCount": 0,
        "retryDelay": 0
    },
    "section": [
        {
            "startMilestoneID": "string",
            "endMilestoneID": "string",
            "typicalDuration": 0,
            "maximumDuration": 0
        }
    ]
},
"orderLines": "A list of order lines for this plan item.",
"action": "Action for the plan item.",
"status": "Status for the plan item.",
"statusChanged": "2023-02-04T18:16:04.927Z",
"pcRequestType": "Type of request that was send to the process
component.",
"milestones": [
    {
        "milestoneID": "Unique identifier for this milestone within
the plan item and plan.",
        "description": "Description of the milestone.",
        "ready": false,
        "readyTime": "2023-02-04T18:16:04.927Z",
        "status": "Status for the milestone.",
        "statusChanged": "2023-02-04T18:16:04.927Z",
        "dependencies": [
            {
                "dependencyID": "Unique identifier for this dependency.",
                "type": "Type of dependency.",

```

```

        "ready": false,
        "readyTime": "2023-02-04T18:16:04.927Z",
        "status": "Status of the dependency.",
        "statusChanged": "2023-02-04T18:16:04.927Z",
        "timeDelta": 0,
        "external": {
            "eventName": "Name of the external event that satisfies
this dependency.",
            "eventId": "Unique identifier of the external event that
satisfies this dependency.",
            "eventTime": "2023-02-04T18:16:04.927Z"
        },
        "point": {
            "planItemID": "The plan item Id that uniquely identifies
the
to
            plan item that must reach a milestone for this dependency
be satisfied.",
            "planID": "The plan Id for the plan item Id that
uniquely identifies the plan item that must reach a milestone
for this dependency to be satisfied. If omitted then the
current plan is assumed.",
            "milestoneID": "Warning: CHOICE element",
            "end": "Warning: CHOICE element"
        },
        "time": {
            "absoluteTime": "2023-02-04T18:16:04.927Z"
        }
    }
}
]
}
],
"udves": [
    {
        "evaluationPriority": "string",
        "type": "string",
        "flavor": "string",
        "name": "string",
        "value": "string",
        "originalValue": "string"
    }
]
}

```

Operation submitPlanErrorNotification

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/plan/error`

Example:

```
{
  "orderId": "string",
  "orderRef": "string",
  "errorReason": "string"
}
```

Data Services API samples

This section provides Orchestrator Data service API samples of TIBCO Order Management.

Set PlanItem Request

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/planitems`

Parameters

Name	Type	Value
businessTransactionID (optional)	string (query)	businessTransactionID
CorrectionID (optional)	string (query)	CorrectionID
notificationType	Boolean (query)	true or false
orderId (mandatory)	string (query)	orderid Note: The value of orderId must not contain ":"
orderRef (optional)	string (query)	orderRef Note: You must enter the orderRef value for the older orders where the orderRef was unique and there was no orderId. For the new orders, both the orderId and orderRef are the same, you can enter either one.

Name	Type	Value
Note: The value of orderRef must not contain ":"		
planID (optional)	string (query)	planID
replaceInHeader (optional)	Boolean (query)	true or false <div> Note: When the replace flag is set to true and if the flavor of the udf is input or output, the existing udf for the plan item is replaced. If the flavor is config, then the udf is appended and not replaced even if the replace flag is set to true. </div>
setPlanItemRequest (mandatory)	body	request values

Example of setPlanItemRequest:

```
{
  "businessTransactionID": "string",
  "planID": "string",
  "planItem": {
    "planItemID": "string",
    "planItemName": "string",
    "udf": [
      {
        "evaluationPriority": "string",
        "flavor": "string",
        "name": "string",
        "type": "string",
        "value": "string"
      }
    ]
  },
  "replace": {}
}
```

Get PlanItems Details

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/planitems/get`

Parameters

Name	Type	Value
businessTransactionID (optional)	string (query)	businessTransactionID
CorrectionID (optional)	string (query)	CorrectionID
getPlanItemsRequest (mandatory)	body	request values
idsOnly	Boolean (query)	true or false
includeRelatedPlanItems (optional)	Boolean (query)	true or false
orderId (mandatory)	string (query)	orderid Note: The value of orderId must not contain ":"
orderRef (optional)	string (query)	orderRef Note: You must enter the orderRef value for the older orders where the orderRef was unique and there was no orderId. For the new orders, both the orderId and orderRef are the same, you can enter either one. Note: The value of orderRef must not contain ":"

Name	Type	Value
planID (optional)	string (query)	planID

Example of getPlanItemsRequest:

```
{
  "businessTransactionID": "string",
  "idsOnly": "string",
  "planID": "string",
  "planItem": [
    {
      "planItemID": "string"
    }
  ]
}
```

Set Plan Request

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/plan

Parameters

Name	Type	Value
businessTransactionID (optional)	string (query)	businessTransactionID
CorrectionID (optional)	string (query)	CorrectionID
notificationType	Boolean (query)	true or false
orderId (optional)	string	orderid

Name	Type	Value
	(query)	Note: The value of orderId must not contain ":"
orderRef (optional)	string (query)	orderRef Note: You must enter the orderRef value for the older orders where the orderRef was unique and there was no orderId. For the new orders, both the orderId and orderRef are the same, you can enter either one. Note: The value of orderRef must not contain ":"
planID (optional)	string (query)	planID
replaceInHeader (optional)	Boolean (query)	true or false Note: When the replace flag is set to true and if the flavor of the udf is input or output, the existing udf for the plan item is replaced. If the flavor is config, then the udf is appended and not replaced even if the replace flag is set to true.
setPlanRequest (mandatory)	body	request values

Example of setPlanRequest value:

```
{
  "businessTransactionID": "string",
  "plan": {
    "orderId": "string",
    "orderRef": "string",
    "planID": "string",
    "planItem": [
      {
```



```

    "planItemID": "string",
    "planItemName": "string",
    "udf": [
      {
        "evaluationPriority": "string",
        "flavor": "string",
        "name": "string",
        "type": "string",
        "value": "string"
      }
    ]
  },
  "udf": [
    {
      "evaluationPriority": "string",
      "flavor": "string",
      "name": "string",
      "type": "string",
      "value": "string"
    }
  ]
},
"replace": {}
}

```

Get Plan Details

This operation gives the udf-level plan details.

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/plan/get`

Example:

```

{
  "orderId": "string",
  "orderRef": "string",
  "planID": "string",
  "businessTransactionID": "string",
  "correlationID": "string",
  "idsOnly": true,
  "includeItems": true,

```

```
"matchingUdfNameAllPi": "string",  
"piMatchingUdfNameVal": "string",  
"matchingUdfNameAllPiIgnoreEmpty": true  
}
```

Catalog Services API samples

This section provides catalog service API samples of TIBCO Order Management. The following list shows the different types of API samples:

- [Operation Submit Single Action Model](#): to publish single action model
- [Get Bulk Action Model](#): to get multiple action models
- [Operation Submit Bulk Action Model](#): to publish multiple action models
- [Operation Delete Bulk Action Model](#): to delete multiple action models
- [Operation Get All Action Model](#): to get all action models
- [Operation Submit Single Product Model](#): to publish single product model
- [Operation Submit Bulk Product Model](#): to publish multiple product models
- [Operation Get Bulk Product Model](#): to get multiple product models
- [Operation Delete Bulk Product Model](#): to delete multiple product model
- [Operation Get All Product Model](#): to get all product model
- [Operation Submit Single PlanFragment Model](#): to publish single plan fragment model
- [Operation Get Bulk Planfragment Model](#): to get multiple plan fragment models
- [Operation Submit Bulk PlanFragment Models](#): to publish multiple plan fragment models
- [Operation Delete Bulk PlanFragment Model](#): to delete multiple plan fragment model
- [Operation Get All PlanFragment Model](#): to get all plan fragment model

Submit Single Action Model

This operation is used to publish single action model.

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/actionModel`

Parameter content type: application/xml

Example of Action Model:

```
<?xml version="1.0" encoding="UTF-8"?>
<ActionModel>
  <ActionID>string</ActionID>
  <RecordType>string</RecordType>
  <ShortDesc>string</ShortDesc>
</ActionModel>
```

Get Bulk Action Model

This operation is used to get multiple action models.

Method: HTTP GET method

Endpoint: http://<host_address>:<port_address>/v1/actionModel/bulk

Parameter content type: application/json

Example:

```
{
  "ActionModel": [
    {
      "ActionID": "string",
      "RecordType": "string",
      "ShortDesc": "string"
    }
  ]
}
```

Submit Bulk Action Model

This operation is used to publish multiple action models.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/actionModel/bulk

Parameter content type: application/xml

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<ActionModels>
  <ActionModel>
    <ActionID>string</ActionID>
    <RecordType>string</RecordType>
    <ShortDesc>string</ShortDesc>
  </ActionModel>
</ActionModels>
```

Delete Bulk Action Model

This operation is used to delete multiple action models.

Method: HTTP DELETE method

Endpoint: http://<host_address>:<port_address>/v1/actionModel/bulk

Example:

```
[
  "string"
]
```

Get All Action Model

This operation is used to get all action models.

Method: HTTP GET method

Endpoint: http://<host_address>:<port_address>/v1/actionModel/all

Parameter	Cardinality	Description
page	Mandatory	Page number.
modelsPerPage	Mandatory	Desired number of models per page.

Submit Single Product Model

This operation is used to publish single product model.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/productmodel

Parameter content type: application/xml

Example of Product Model:

```
<?xml version="1.0" encoding="UTF-8"?>
<ProductModel>
  <ItemSpecs>
    <affinityGroup>
      <planFragments>
        <planFragmentID>string</planFragmentID>
        <planFragmentUniqueId_CANCEL>
          <description>string</description>
          <name>string</name>
        </planFragmentUniqueId_CANCEL>
        <planFragmentUniqueId_CEASE>
          <description>string</description>
          <name>string</name>
        </planFragmentUniqueId_CEASE>
        <planFragmentUniqueId_PROVIDE>
          <description>string</description>
          <name>string</name>
        </planFragmentUniqueId_PROVIDE>
        <planFragmentUniqueId_UPDATE>
          <description>string</description>
          <name>string</name>
        </planFragmentUniqueId_UPDATE>
        <planFragmentName>string</planFragmentName>
        <planFragmentVersion>string</planFragmentVersion>
        <owner>string</owner>
        <record_Type>string</record_Type>
        <errorHandler>string</errorHandler>
        <retry>
          <retryCount>0</retryCount>
          <retryDelay>0</retryDelay>
          <retryFailed>true</retryFailed>
          <retryOverride>true</retryOverride>
        </retry>
      </section>
```

```

        <endMilestoneID>string</endMilestoneID>
        <maximumDuration>0</maximumDuration>
        <startMilestoneID>string</startMilestoneID>
        <typicalDuration>0</typicalDuration>
    </section>
</planFragments>
<type>string</type>
</affinityGroup>
<characteristics>
    <actionID>string</actionID>
    <conditional>string</conditional>
    <description>string</description>
    <evaluationPriority>string</evaluationPriority>
    <instanceCeaseSequence>string</instanceCeaseSequence>
    <instanceMax>0</instanceMax>
    <instanceMin>0</instanceMin>
    <instanceOptional>string</instanceOptional>
    <instanceSequence>string</instanceSequence>
    <instanceUpdateSequence>string</instanceUpdateSequence>
    <name>string</name>
    <simpleRule>
        <externalRuleset>string</externalRuleset>
        <name>string</name>
        <ruleSetOutcome>string</ruleSetOutcome>
    </simpleRule>
    <value>
        <discreteValue>string</discreteValue>
        <mandatoryValue>string</mandatoryValue>
        <rangeValue>string</rangeValue>
        <type>string</type>
    </value>
</characteristics>
<DependentOnProduct>
    <Product>
        <description>string</description>
        <lastModifiedOn>2020-09-
17T18:11:46.335Z</lastModifiedOn>
        <owner>string</owner>
        <productId>string</productId>
    </Product>
</DependentOnProduct>
<endDate>2020-09-17T18:11:46.335Z</endDate>
<MustComplete>true</MustComplete>
<planFragments>
    <planFragmentID>string</planFragmentID>
    <planFragmentUniqueId_CANCEL>

```

```

        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_CANCEL>
    <planFragmentUniqueId_CEASE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_CEASE>
    <planFragmentUniqueId_PROVIDE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_PROVIDE>
    <planFragmentUniqueId_UPDATE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_UPDATE>
    <planFragmentName>string</planFragmentName>
    <planFragmentVersion>string</planFragmentVersion>
    <owner>string</owner>
    <record_Type>string</record_Type>
    <errorHandler>string</errorHandler>
    <retry>
        <retryCount>0</retryCount>
        <retryDelay>0</retryDelay>
        <retryFailed>true</retryFailed>
        <retryOverride>true</retryOverride>
    </retry>
    <section>
        <endMilestoneID>string</endMilestoneID>
        <maximumDuration>0</maximumDuration>
        <startMilestoneID>string</startMilestoneID>
        <typicalDuration>0</typicalDuration>
    </section>
</planFragments>
<product>
    <description>string</description>
    <lastModifiedOn>2020-09-17T18:11:46.335Z</lastModifiedOn>
    <owner>string</owner>
    <productId>string</productId>
</product>
<SingleUse>true</SingleUse>
<startDate>2020-09-17T18:11:46.335Z</startDate>
</ItemSpecs>
<Product>
    <description>string</description>
    <lastModifiedOn>2020-09-17T18:11:46.335Z</lastModifiedOn>
    <owner>string</owner>

```



```

    <productId>string</productId>
  </Product>
  <affinityActionGroup>true</affinityActionGroup>
  <affinityActionValue>string</affinityActionValue>
  <affinityCondition>string</affinityCondition>
  <affinityCorrelation>string</affinityCorrelation>
  <affinityParentGroup>true</affinityParentGroup>
  <affinityPlanFragments>
    <planFragmentID>string</planFragmentID>
    <planFragmentUniqueId_CANCEL>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_CANCEL>
    <planFragmentUniqueId_CEASE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_CEASE>
    <planFragmentUniqueId_PROVIDE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_PROVIDE>
    <planFragmentUniqueId_UPDATE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_UPDATE>
    <planFragmentName>string</planFragmentName>
    <planFragmentVersion>string</planFragmentVersion>
    <owner>string</owner>
    <record_Type>string</record_Type>
    <errorHandler>string</errorHandler>
    <retry>
      <retryCount>0</retryCount>
      <retryDelay>0</retryDelay>
      <retryFailed>true</retryFailed>
      <retryOverride>true</retryOverride>
    </retry>
    <section>
      <endMilestoneID>string</endMilestoneID>
      <maximumDuration>0</maximumDuration>
      <startMilestoneID>string</startMilestoneID>
      <typicalDuration>0</typicalDuration>
    </section>
  </affinityPlanFragments>
  <affinityType>string</affinityType>
  <characteristics>
    <actionID>string</actionID>

```

```

    <conditional>string</conditional>
    <description>string</description>
    <evaluationPriority>string</evaluationPriority>
    <instanceCeaseSequence>string</instanceCeaseSequence>
    <instanceMax>0</instanceMax>
    <instanceMin>0</instanceMin>
    <instanceOptional>string</instanceOptional>
    <instanceSequence>string</instanceSequence>
    <instanceUpdateSequence>string</instanceUpdateSequence>
    <name>string</name>
    <simpleRule>
        <externalRuleset>string</externalRuleset>
        <name>string</name>
        <ruleSetOutcome>string</ruleSetOutcome>
    </simpleRule>
    <value>
        <discreteValue>string</discreteValue>
        <mandatoryValue>string</mandatoryValue>
        <rangeValue>string</rangeValue>
        <type>string</type>
    </value>
</characteristics>
<endDate>2020-09-17T18:11:46.336Z</endDate>
<incompatibleWithProduct>
    <description>string</description>
    <lastModifiedOn>2020-09-17T18:11:46.336Z</lastModifiedOn>
    <owner>string</owner>
    <productId>string</productId>
</incompatibleWithProduct>
<plan>
    <action>string</action>
    <actionMode>string</actionMode>
    <affinity>true</affinity>
    <affinityActionGroup>true</affinityActionGroup>
    <affinityActionValue>string</affinityActionValue>
    <affinityCondition>string</affinityCondition>
    <affinityCorrelation>string</affinityCorrelation>
    <affinityParentGroup>true</affinityParentGroup>
    <affinityType>string</affinityType>
    <description>string</description>
    <milestone>
        <dependency>
            <condition>string</condition>
            <milestoneName>string</milestoneName>
            <planName>string</planName>
            <type>string</type>

```

```

        </dependency>
        <name>string</name>
    </milestone>
    <name>string</name>
</plan>
<planFragments>
    <planFragmentID>string</planFragmentID>
    <planFragmentUniqueId_CANCEL>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_CANCEL>
    <planFragmentUniqueId_CEASE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_CEASE>
    <planFragmentUniqueId_PROVIDE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_PROVIDE>
    <planFragmentUniqueId_UPDATE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_UPDATE>
    <planFragmentName>string</planFragmentName>
    <planFragmentVersion>string</planFragmentVersion>
    <owner>string</owner>
    <record_Type>string</record_Type>
    <errorHandler>string</errorHandler>
    <retry>
        <retryCount>0</retryCount>
        <retryDelay>0</retryDelay>
        <retryFailed>true</retryFailed>
        <retryOverride>true</retryOverride>
    </retry>
    <section>
        <endMilestoneID>string</endMilestoneID>
        <maximumDuration>0</maximumDuration>
        <startMilestoneID>string</startMilestoneID>
        <typicalDuration>0</typicalDuration>
    </section>
</planFragments>
    <startDate>2020-09-17T18:11:46.336Z</startDate>
</ProductModel>

```

Submit Bulk Product Models

This operation is used to publish multiple product models.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/productmodel/bulk

Parameter content type: application/xml

Example of Product Models:

```
<?xml version="1.0" encoding="UTF-8"?>
<ProductModels>
  <ProductModel>
    <ItemSpecs>
      <affinityGroup>
        <planFragments>
          <planFragmentID>string</planFragmentID>
          <planFragmentUniqueId_CANCEL>
            <description>string</description>
            <name>string</name>
          </planFragmentUniqueId_CANCEL>
          <planFragmentUniqueId_CEASE>
            <description>string</description>
            <name>string</name>
          </planFragmentUniqueId_CEASE>
          <planFragmentUniqueId_PROVIDE>
            <description>string</description>
            <name>string</name>
          </planFragmentUniqueId_PROVIDE>
          <planFragmentUniqueId_UPDATE>
            <description>string</description>
            <name>string</name>
          </planFragmentUniqueId_UPDATE>
          <planFragmentName>string</planFragmentName>
          <planFragmentVersion>string</planFragmentVersion>
          <owner>string</owner>
          <record_Type>string</record_Type>
          <errorHandler>string</errorHandler>
          <retry>
            <retryCount>0</retryCount>
            <retryDelay>0</retryDelay>
            <retryFailed>true</retryFailed>
            <retryOverride>true</retryOverride>
          </retry>
        </planFragments>
      </affinityGroup>
    </ItemSpecs>
  </ProductModel>
</ProductModels>
```

```

        <section>
            <endMilestoneID>string</endMilestoneID>
            <maximumDuration>0</maximumDuration>
            <startMilestoneID>string</startMilestoneID>
            <typicalDuration>0</typicalDuration>
        </section>
    </planFragments>
    <type>string</type>
</affinityGroup>
<characteristics>
    <actionID>string</actionID>
    <conditional>string</conditional>
    <description>string</description>
    <evaluationPriority>string</evaluationPriority>
    <instanceCeaseSequence>string</instanceCeaseSequence>
    <instanceMax>0</instanceMax>
    <instanceMin>0</instanceMin>
    <instanceOptional>string</instanceOptional>
    <instanceSequence>string</instanceSequence>
    <instanceUpdateSequence>string</instanceUpdateSequence>
    <name>string</name>
    <simpleRule>
        <externalRuleset>string</externalRuleset>
        <name>string</name>
        <ruleSetOutcome>string</ruleSetOutcome>
    </simpleRule>
    <value>
        <discreteValue>string</discreteValue>
        <mandatoryValue>string</mandatoryValue>
        <rangeValue>string</rangeValue>
        <type>string</type>
    </value>
</characteristics>
<DependentOnProduct>
    <Product>
        <description>string</description>
        <lastModifiedOn>2020-09-
17T18:27:06.973Z</lastModifiedOn>
        <owner>string</owner>
        <productId>string</productId>
    </Product>
</DependentOnProduct>
<endDate>2020-09-17T18:27:06.973Z</endDate>
<MustComplete>true</MustComplete>
<planFragments>
    <planFragmentID>string</planFragmentID>

```

```

    <planFragmentUniqueId_CANCEL>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_CANCEL>
    <planFragmentUniqueId_CEASE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_CEASE>
    <planFragmentUniqueId_PROVIDE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_PROVIDE>
    <planFragmentUniqueId_UPDATE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_UPDATE>
    <planFragmentName>string</planFragmentName>
    <planFragmentVersion>string</planFragmentVersion>
    <owner>string</owner>
    <record_Type>string</record_Type>
    <errorHandler>string</errorHandler>
    <retry>
      <retryCount>0</retryCount>
      <retryDelay>0</retryDelay>
      <retryFailed>true</retryFailed>
      <retryOverride>true</retryOverride>
    </retry>
    <section>
      <endMilestoneID>string</endMilestoneID>
      <maximumDuration>0</maximumDuration>
      <startMilestoneID>string</startMilestoneID>
      <typicalDuration>0</typicalDuration>
    </section>
  </planFragments>
</product>
  <description>string</description>
  <lastModifiedOn>2020-09-
17T18:27:06.973Z</lastModifiedOn>
  <owner>string</owner>
  <productId>string</productId>
</product>
<SingleUse>true</SingleUse>
<startDate>2020-09-17T18:27:06.973Z</startDate>
</ItemSpecs>
<Product>
  <description>string</description>

```

```

        <lastModifiedOn>2020-09-17T18:27:06.973Z</lastModifiedOn>
        <owner>string</owner>
        <productId>string</productId>
    </Product>
    <affinityActionGroup>true</affinityActionGroup>
    <affinityActionValue>string</affinityActionValue>
    <affinityCondition>string</affinityCondition>
    <affinityCorrelation>string</affinityCorrelation>
    <affinityParentGroup>true</affinityParentGroup>
    <affinityPlanFragments>
        <planFragmentID>string</planFragmentID>
        <planFragmentUniqueId_CANCEL>
            <description>string</description>
            <name>string</name>
        </planFragmentUniqueId_CANCEL>
        <planFragmentUniqueId_CEASE>
            <description>string</description>
            <name>string</name>
        </planFragmentUniqueId_CEASE>
        <planFragmentUniqueId_PROVIDE>
            <description>string</description>
            <name>string</name>
        </planFragmentUniqueId_PROVIDE>
        <planFragmentUniqueId_UPDATE>
            <description>string</description>
            <name>string</name>
        </planFragmentUniqueId_UPDATE>
        <planFragmentName>string</planFragmentName>
        <planFragmentVersion>string</planFragmentVersion>
        <owner>string</owner>
        <record_Type>string</record_Type>
        <errorHandler>string</errorHandler>
        <retry>
            <retryCount>0</retryCount>
            <retryDelay>0</retryDelay>
            <retryFailed>true</retryFailed>
            <retryOverride>true</retryOverride>
        </retry>
        <section>
            <endMilestoneID>string</endMilestoneID>
            <maximumDuration>0</maximumDuration>
            <startMilestoneID>string</startMilestoneID>
            <typicalDuration>0</typicalDuration>
        </section>
    </affinityPlanFragments>
    <affinityType>string</affinityType>

```

```

<characteristics>
  <actionID>string</actionID>
  <conditional>string</conditional>
  <description>string</description>
  <evaluationPriority>string</evaluationPriority>
  <instanceCeaseSequence>string</instanceCeaseSequence>
  <instanceMax>0</instanceMax>
  <instanceMin>0</instanceMin>
  <instanceOptional>string</instanceOptional>
  <instanceSequence>string</instanceSequence>
  <instanceUpdateSequence>string</instanceUpdateSequence>
  <name>string</name>
  <simpleRule>
    <externalRuleset>string</externalRuleset>
    <name>string</name>
    <ruleSetOutcome>string</ruleSetOutcome>
  </simpleRule>
  <value>
    <discreteValue>string</discreteValue>
    <mandatoryValue>string</mandatoryValue>
    <rangeValue>string</rangeValue>
    <type>string</type>
  </value>
</characteristics>
<endDate>2020-09-17T18:27:06.973Z</endDate>
<incompatibleWithProduct>
  <description>string</description>
  <lastModifiedOn>2020-09-17T18:27:06.973Z</lastModifiedOn>
  <owner>string</owner>
  <productId>string</productId>
</incompatibleWithProduct>
<plan>
  <action>string</action>
  <actionMode>string</actionMode>
  <affinity>true</affinity>
  <affinityActionGroup>true</affinityActionGroup>
  <affinityActionValue>string</affinityActionValue>
  <affinityCondition>string</affinityCondition>
  <affinityCorrelation>string</affinityCorrelation>
  <affinityParentGroup>true</affinityParentGroup>
  <affinityType>string</affinityType>
  <description>string</description>
  <milestone>
    <dependency>
      <condition>string</condition>
      <milestoneName>string</milestoneName>
    </dependency>
  </milestone>
</plan>

```



```

        <planName>string</planName>
        <type>string</type>
    </dependency>
    <name>string</name>
</milestone>
<name>string</name>
</plan>
<planFragments>
    <planFragmentID>string</planFragmentID>
    <planFragmentUniqueId_CANCEL>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_CANCEL>
    <planFragmentUniqueId_CEASE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_CEASE>
    <planFragmentUniqueId_PROVIDE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_PROVIDE>
    <planFragmentUniqueId_UPDATE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_UPDATE>
    <planFragmentName>string</planFragmentName>
    <planFragmentVersion>string</planFragmentVersion>
    <owner>string</owner>
    <record_Type>string</record_Type>
    <errorHandler>string</errorHandler>
    <retry>
        <retryCount>0</retryCount>
        <retryDelay>0</retryDelay>
        <retryFailed>true</retryFailed>
        <retryOverride>true</retryOverride>
    </retry>
    <section>
        <endMilestoneID>string</endMilestoneID>
        <maximumDuration>0</maximumDuration>
        <startMilestoneID>string</startMilestoneID>
        <typicalDuration>0</typicalDuration>
    </section>
</planFragments>
    <startDate>2020-09-17T18:27:06.973Z</startDate>
</ProductModel>
</ProductModels>

```

Get Bulk Product Model

This operation is used to get multiple product models. This is generally used for bundle catalog.

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/productmodel/bulk`

Parameter content type: application/json

Example:

```
{
  "ActionModel": [
    {
      "ActionID": "string",
      "RecordType": "string",
      "ShortDesc": "string"
    }
  ]
}
```

Delete Bulk Product Model

This operation is used to delete multiple product model.

Method: HTTP DELETE method

Endpoint: `http://<host_address>:<port_address>/v1/productmodel/bulk`

Example:

```
[
  {
    "productID": "string",
    "productIDExt": "string"
  }
]
```

Get All Product Model

This operation is used to get all product models.

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/productmodel/all`

Parameter	Cardinality	Description
page	Mandatory	Page number.
modelsPerPage	Mandatory	Desired number of models per page.

Submit Single PlanFragment Model

This operation is used to publish single planfragment model.

Method: HTTP POST method

Endpoint:`http://<host_address>:<port_address>/v1/planfragmentmodel`

Parameter content type: application/xml

Example of PlanFragment:

```
<?xml version="1.0" encoding="UTF-8"?>
<PlanFragment>
  <planFragmentID>string</planFragmentID>
  <planFragmentUniqueId_CANCEL>
    <description>string</description>
    <name>string</name>
  </planFragmentUniqueId_CANCEL>
  <planFragmentUniqueId_CEASE>
    <description>string</description>
    <name>string</name>
  </planFragmentUniqueId_CEASE>
  <planFragmentUniqueId_PROVIDE>
    <description>string</description>
    <name>string</name>
  </planFragmentUniqueId_PROVIDE>
</PlanFragment>
```

```

    </planFragmentUniqueId_PROVIDE>
    <planFragmentUniqueId_UPDATE>
        <description>string</description>
        <name>string</name>
    </planFragmentUniqueId_UPDATE>
    <planFragmentName>string</planFragmentName>
    <planFragmentVersion>string</planFragmentVersion>
    <owner>string</owner>
    <record_Type>string</record_Type>
    <errorHandler>string</errorHandler>
    <retry>
        <retryCount>0</retryCount>
        <retryDelay>0</retryDelay>
        <retryFailed>true</retryFailed>
        <retryOverride>true</retryOverride>
    </retry>
    <section>
        <endMilestoneID>string</endMilestoneID>
        <maximumDuration>0</maximumDuration>
        <startMilestoneID>string</startMilestoneID>
        <typicalDuration>0</typicalDuration>
    </section>
</PlanFragment>

```

Get Bulk PlanFragment Model

This operation is used to get multiple PlanFragment models.

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/v1/Planfragment/bulk`

Parameter content type: application/json

Example:

```

{
  "planFragment": [
    {
      "planFragmentUniqueId_CANCEL": {
        "name": "string",
        "description": "string"
      },
    },
  ],
}

```

```

    "planFragmentUniqueId_PROVIDE": {
      "name": "string",
      "description": "string"
    },
    "planFragmentUniqueId_CEASE": {
      "name": "string",
      "description": "string"
    },
    "planFragmentUniqueId_UPDATE": {
      "name": "string",
      "description": "string"
    }
  }
]
}

```

Submit Bulk PlanFragment Models

This operation is used to publish multiple planfragment models.

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/planfragmentmodel/bulk`

Parameter content type: application/xml

Example of PlanFragment models:

```

<?xml version="1.0" encoding="UTF-8"?>
<PlanFragmentModels>
  <planFragment>
    <planFragmentID>string</planFragmentID>
    <planFragmentUniqueId_CANCEL>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_CANCEL>
    <planFragmentUniqueId_CEASE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_CEASE>
    <planFragmentUniqueId_PROVIDE>
      <description>string</description>
      <name>string</name>
    </planFragmentUniqueId_PROVIDE>
  </planFragment>
</PlanFragmentModels>

```

```

        </planFragmentUniqueId_PROVIDE>
        <planFragmentUniqueId_UPDATE>
            <description>string</description>
            <name>string</name>
        </planFragmentUniqueId_UPDATE>
        <planFragmentName>string</planFragmentName>
        <planFragmentVersion>string</planFragmentVersion>
        <owner>string</owner>
        <record_Type>string</record_Type>
        <errorHandler>string</errorHandler>
        <retry>
            <retryCount>0</retryCount>
            <retryDelay>0</retryDelay>
            <retryFailed>true</retryFailed>
            <retryOverride>true</retryOverride>
        </retry>
        <section>
            <endMilestoneID>string</endMilestoneID>
            <maximumDuration>0</maximumDuration>
            <startMilestoneID>string</startMilestoneID>
            <typicalDuration>0</typicalDuration>
        </section>
    </planFragment>
</PlanFragmentModels>

```

Delete Bulk PlanFragment Model

This operation is used to delete multiple plans in the final state.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/planfragmentmodel/purge`

tenantID is the mandatory header parameter.

Example:

```

{
  "orderPlanRecordList": [
    {
      "planID": "string",
      "orderID": "string"
    }
  ]
}

```

```
]
}
```

Get All PlanFragment Model

This operation is used to get all Plan Fragment models.

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/Planfragmentmodel/all`

Parameter	Cardinality	Description
page	Mandatory	Page number.
modelsPerPage	Mandatory	Desired number of models per page.

Process Component API sample

This section provides process component API samples of TIBCO Order Management.

Operation Feasibility

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/feasibility`

Parameter content type: `application/json`

The headers are mandatory parameters here.

Example of feasibilityRequest:

```
{
  "orderId": "string",
  "orderRequest": {
    "extension": {
      "any": [
        "string"
      ]
    },
    "header": {
      "customerID": "string",
      "deliveryAddress": {
        "country": "string",
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "postCode": "string",
        "region": "string",
        "supplementaryLocation": "string"
      },
      "description": "string",
      "extension": {
        "any": [
          "string"
        ]
      }
    }
  }
}
```



```

    ]
  },
  "invoiceAddress": {
    "country": "string",
    "line1": "string",
    "line2": "string",
    "line3": "string",
    "locality": "string",
    "postCode": "string",
    "region": "string",
    "supplementaryLocation": "string"
  },
  "notes": "string",
  "orderPriority": 0,
  "requiredByDate": "2020-09-18T11:36:00.151Z",
  "requiredOnDate": "2020-09-18T11:36:00.151Z",
  "slaID": [
    "string"
  ],
  "subscriberID": "string",
  "udf": [
    {
      "extension": {
        "any": [
          "string"
        ]
      },
      "name": "string",
      "value": "string"
    }
  ]
},
"line": [
  {
    "action": "string",
    "actionMode": "string",
    "characteristic": [
      {
        "description": "string",
        "extension": {
          "any": [
            "string"
          ]
        },
        "name": "string",
        "value": [
          {

```

```

        "extension": {
            "any": [
                "string"
            ]
        },
        "name": "string",
        "type": "string",
        "value": "string",
        "valueFrom": "string",
        "valueTo": "string"
    }
]
},
"customerItemID": "string",
"deliveryAddress": {
    "country": "string",
    "line1": "string",
    "line2": "string",
    "line3": "string",
    "locality": "string",
    "postCode": "string",
    "region": "string",
    "supplementaryLocation": "string"
},
"extension": {
    "any": [
        "string"
    ]
},
"inventoryID": "string",
"lineNumber": "string",
"linkID": "string",
"notes": "string",
"productID": "string",
"productVersion": "string",
"quantity": 0,
"requiredByDate": "2020-09-18T11:36:00.151Z",
"requiredOnDate": "2020-09-18T11:36:00.151Z",
"slaID": [
    "string"
],
"subscriberID": "string",
"udf": [
    {
        "extension": {
            "any": [

```

```

        "string"
      ]
    },
    "name": "string",
    "value": "string"
  }
],
"uom": "string"
}
],
"orderID": "string",
"orderRef": "string"
},
"businessTransactionID": "string",
"correlationID": "string"
}

```

Operation Plan Item Activate Request

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/planitem/activaterequest

Parameter content type: application/json

The headers are mandatory parameters here.

Example of planItemActivateRequest:

```

{
  "orderID": "string",
  "orderRef": "string",
  "planID": "string",
  "planItem": {
    "planItemID": "string",
    "description": "string",
    "processComponentID": "string",
    "processComponentName": "string",
    "processComponentVersion": "string",
    "processComponentType": "string",
    "processComponentRecordType": "string",
    "orderLine": [
      {
        "orderLineNumber": "string",

```

```

        "productID": "string",
        "productVersion": "string",
        "action": "string",
        "actionMode": "string",
        "quantity": 0,
        "uom": "string",
        "subscriberID": "string",
        "linkID": "string",
        "inventoryID": "string",
        "eol": true
    }
],
    "action": "string",
    "actionMode": "string"
},
"resumeExecution": {},
"cancelAndRollback": {},
"cancelWithNoRollback": {},
"businessTransactionID": "string",
"correlationID": "string"
}

```

Operation Plan Item Error Handler

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/planitem/errorhandlerrequest`

Parameter content type: application/json

The headers are mandatory parameters here.

Example of planItemFailedRequest:

```

{
  "orderId": "string",
  "orderRef": "string",
  "planID": "string",
  "planItem": {
    "planItemID": "string",
    "description": "string",
    "processComponentID": "string",
    "processComponentName": "string",
    "processComponentVersion": "string",

```

```

"processComponentType": "string",
"processComponentRecordType": "string",
"orderLine": [
  {
    "orderLineNumber": "string",
    "productID": "string",
    "productVersion": "string",
    "action": "string",
    "actionMode": "string",
    "quantity": 0,
    "uom": "string",
    "subscriberID": "string",
    "linkID": "string",
    "inventoryID": "string",
    "eol": true
  }
],
"action": "string",
"actionMode": "string"
},
"errorHandler": "string",
"businessTransactionID": "string",
"correlationID": "string",
"message": [
  {
    "code": "string",
    "description": "string",
    "lineNumber": "string",
    "type": "string",
    "udfs": [
      {
        "name": "string",
        "value": "string"
      }
    ]
  }
]
}

```

PlanItem Execute Request

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/planitem/executionrequest

Parameter content type: application/json

Input planItemExecuteRequest is mandatory for this operation.

Example of planItemExecuteRequest:

```
{
  "orderId": "string",
  "orderRef": "string",
  "planId": "string",
  "planItem": {
    "orderLines": [
      {
        "eol": true,
        "orderLineNumber": "string"
      }
    ],
    "planItemID": "string",
    "statusChanged": "2019-06-19T11:52:07.702Z",
    "planItemName": "string",
    "description": "string",
    "cancelled": true,
    "childIds": [
      "string"
    ],
    "dependentIds": [
      "string"
    ],
    "endTimeStamp": 0,
    "isNoReciprocalAction": true,
    "isNonExecuting": true,
    "parentIds": [
      "string"
    ],
    "planFragmentUniqueId": "string",
    "siblingIds": [
      "string"
    ],
    "startTimeStamp": 0,
    "udfs": [
      {
        "evaluationPriority": "string",
        "flavor": "string",
        "name": "string",
        "originalValue": "string",
        "type": "string",
        "value": "string"
      }
    ]
  }
}
```

```

],
"planFragment": {
  "planFragmentID": "string",
  "planFragmentName": "string",
  "planFragmentVersion": "string",
  "owner": "string",
  "record_Type": "string",
  "errorHandler": "string",
  "retry": {
    "retryCount": 0,
    "retryDelay": 0,
    "retryFailed": true,
    "retryOverride": true
  },
  "section": [
    {
      "endMilestoneID": "string",
      "maximumDuration": 0,
      "startMilestoneID": "string",
      "typicalDuration": 0
    }
  ]
},
"action": "string",
"status": "string",
"pcRequestType": "string",
"milestones": [
  {
    "dependencies": [
      {
        "dependencyID": "string",
        "external": {
          "eventId": "string",
          "eventName": "string",
          "eventTime": "2019-06-19T11:52:07.702Z"
        },
        "point": {
          "end": {},
          "milestoneID": "string",
          "planID": "string",
          "planItemID": "string"
        },
        "ready": true,
        "readyTime": "2019-06-19T11:52:07.702Z",
        "status": "string",
        "statusChanged": "2019-06-19T11:52:07.702Z",
        "time": {

```

```

        "absoluteTime": "2019-06-19T11:52:07.702Z"
      },
      "timeDelta": 0,
      "type": "string"
    }
  ],
  "description": "string",
  "milestoneID": "string",
  "ready": true,
  "readyTime": "2019-06-19T11:52:07.702Z",
  "status": "string",
  "statusChanged": "2019-06-19T11:52:07.702Z"
}
]
},
"sla": {
  "maximumDuration": 0,
  "typicalDuration": 0
},
"waitAtMilestoneIds": [
  "string"
],
"notifyAtMilestoneIds": [
  "string"
],
"businessTransactionId": "string",
"correlationId": "string"
}

```

Operation Milestone Release Request

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/planitem/milestonerelease`

Parameter content type: application/json

The headers are mandatory parameters here.

Example of milestoneReleaseRequest:

```

{
  "orderId": "string",
  "orderRef": "string",

```



```

"planID": "string",
"planItem": {
  "planItemID": "string",
  "description": "string",
  "processComponentID": "string",
  "processComponentName": "string",
  "processComponentVersion": "string",
  "processComponentType": "string",
  "processComponentRecordType": "string",
  "orderLine": [
    {
      "orderLineNumber": "string",
      "productID": "string",
      "productVersion": "string",
      "action": "string",
      "actionMode": "string",
      "quantity": 0,
      "uom": "string",
      "subscriberID": "string",
      "linkID": "string",
      "inventoryID": "string",
      "eol": true
    }
  ],
  "action": "string",
  "actionMode": "string"
},
"milestoneID": "string",
"businessTransactionID": "string",
"correlationID": "string"
}

```

Operation Plan Item Suspend Request

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/planitem/suspendrequest`

Parameter content type: application/json

The headers are mandatory parameters here.

Example of planItemSuspendRequest:

```

{
  "orderId": "string",
  "orderRef": "string",
  "planID": "string",
  "planItem": {
    "planItemID": "string",
    "description": "string",
    "processComponentID": "string",
    "processComponentName": "string",
    "processComponentVersion": "string",
    "processComponentType": "string",
    "processComponentRecordType": "string",
    "orderLine": [
      {
        "orderLineNumber": "string",
        "productID": "string",
        "productVersion": "string",
        "action": "string",
        "actionMode": "string",
        "quantity": 0,
        "uom": "string",
        "subscriberID": "string",
        "linkID": "string",
        "inventoryID": "string",
        "eol": true
      }
    ],
    "action": "string",
    "actionMode": "string"
  },
  "businessTransactionID": "string",
  "correlationID": "string"
}

```

Operation PreQualification Failed

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/pqf

Parameter content type: application/json

The headers are mandatory parameters here.

Example of preQualificationFailedRequest:

```

{
  "orderId": "string",
  "orderRequest": {
    "extension": {
      "any": [
        "string"
      ]
    },
    "header": {
      "customerID": "string",
      "deliveryAddress": {
        "country": "string",
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "postCode": "string",
        "region": "string",
        "supplementaryLocation": "string"
      },
      "description": "string",
      "extension": {
        "any": [
          "string"
        ]
      }
    },
    "invoiceAddress": {
      "country": "string",
      "line1": "string",
      "line2": "string",
      "line3": "string",
      "locality": "string",
      "postCode": "string",
      "region": "string",
      "supplementaryLocation": "string"
    },
    "notes": "string",
    "orderPriority": 0,
    "requiredByDate": "2020-09-18T11:40:47.700Z",
    "requiredOnDate": "2020-09-18T11:40:47.700Z",
    "slaID": [
      "string"
    ],
    "subscriberID": "string",
    "udf": [
      {
        "extension": {

```

```

        "any": [
            "string"
        ]
    },
    "name": "string",
    "value": "string"
}
]
},
"line": [
    {
        "action": "string",
        "actionMode": "string",
        "characteristic": [
            {
                "description": "string",
                "extension": {
                    "any": [
                        "string"
                    ]
                },
            },
            {
                "name": "string",
                "value": [
                    {
                        "extension": {
                            "any": [
                                "string"
                            ]
                        },
                    },
                    {
                        "name": "string",
                        "type": "string",
                        "value": "string",
                        "valueFrom": "string",
                        "valueTo": "string"
                    }
                ]
            }
        ]
    }
],
"customerItemID": "string",
"deliveryAddress": {
    "country": "string",
    "line1": "string",
    "line2": "string",
    "line3": "string",
    "locality": "string",
    "postCode": "string",
    "region": "string",

```

```

        "supplementaryLocation": "string"
    },
    "extension": {
        "any": [
            "string"
        ]
    },
    "inventoryID": "string",
    "lineNumber": "string",
    "linkID": "string",
    "notes": "string",
    "productID": "string",
    "productVersion": "string",
    "quantity": 0,
    "requiredByDate": "2020-09-18T11:40:47.700Z",
    "requiredOnDate": "2020-09-18T11:40:47.700Z",
    "slaID": [
        "string"
    ],
    "subscriberID": "string",
    "udf": [
        {
            "extension": {
                "any": [
                    "string"
                ]
            },
            "name": "string",
            "value": "string"
        }
    ],
    "uom": "string"
}
],
"orderID": "string",
"orderRef": "string"
},
"feasibilityFailed": {
    "completed": true
},
"opdFailed": {
    "completed": true,
    "success": true
},
"message": [
    {
        "code": "string",

```

```

    "description": "string",
    "lineNumber": "string",
    "type": "string",
    "udfs": [
      {
        "name": "string",
        "value": "string"
      }
    ]
  },
  "businessTransactionID": "string",
  "correlationID": "string"
}

```

Operation OpdErrorHandler

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/plan/opdErrorHandlerRequest

The headers are mandatory parameters here.

Example:

```

{
  "orderId": "string",
  "orderRef": "string",
  "businessTransactionId": "string",
  "orderRequest": {
    "orderRef": "string",
    "orderId": "string",
    "header": {
      "description": "string",
      "orderPriority": 9,
      "invoiceAddress": {
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "region": "string",
        "country": "string",
        "postCode": "string",
        "supplementaryLocation": "string"
      }
    }
  }
}

```

```

    },
    "deliveryAddress": {
      "line1": "string",
      "line2": "string",
      "line3": "string",
      "locality": "string",
      "region": "string",
      "country": "string",
      "postCode": "string",
      "supplementaryLocation": "string"
    },
    "notes": "string",
    "slaID": [
      "string"
    ],
    "udf": [
      {
        "name": "string",
        "value": "string",
        "extension": {
          "any": [
            "string"
          ]
        }
      }
    ],
    "extension": {
      "any": [
        "string"
      ]
    },
    "customerID": "string",
    "subscriberID": "string",
    "requiredByDate": "2023-02-01T06:38:59.306Z",
    "requiredOnDate": "2023-02-01T06:38:59.306Z"
  },
  "line": [
    {
      "lineNumber": "string",
      "subscriberID": "string",
      "productID": "string",
      "productVersion": "string",
      "quantity": 0,
      "uom": "string",
      "deliveryAddress": {
        "line1": "string",
        "line2": "string",

```

```

        "line3": "string",
        "locality": "string",
        "region": "string",
        "country": "string",
        "postCode": "string",
        "supplementaryLocation": "string"
    },
    "action": "string",
    "actionMode": "string",
    "linkID": "string",
    "inventoryID": "string",
    "notes": "string",
    "slaID": [
        "string"
    ],
    "udf": [
        {
            "name": "string",
            "value": "string",
            "extension": {
                "any": [
                    "string"
                ]
            }
        }
    ],
    "characteristic": [
        {
            "name": "string",
            "description": "string",
            "value": [
                {
                    "name": "string",
                    "type": "string",
                    "value": "string",
                    "valueFrom": "string",
                    "valueTo": "string",
                    "extension": {
                        "any": [
                            "string"
                        ]
                    }
                }
            ]
        }
    ],
    "extension": {
        "any": [
            "string"
        ]
    }
}

```



```
        ]
      }
    }
  ],
  "customerItemID": "string",
  "extension": {
    "any": [
      "string"
    ]
  },
  "requiredByDate": "2023-02-01T06:38:59.306Z",
  "requiredOnDate": "2023-02-01T06:38:59.306Z"
}
],
"extension": {
  "any": [
    "string"
  ]
}
},
"planGenerationFailedMessage": "string",
"amendment": true
}
```

Jeopardy Services API samples

This section provides Jeopardy service API samples of TIBCO Order Management.

Purge Plans

This operation is used to delete multiple plans in the final state.



Note: Only the plans that are in the final state (for example, CANCELED, COMPLETE) can be purged.

Method: HTTP DELETE

Endpoint: `http://<host_address>:<port_address>/plan/purge`

Example:

```
{
  "orderPlanRecordList": [
    {
      "planID": "string",
      "orderID": "string"
    }
  ]
}
```

Get Plan in Jeopardy

This operation is used to get plan in jeopardy.

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/plan/jeopardy`

Suspend Rule

This operation is used to suspend rules.

Method: HTTP PUT method

Endpoint: `http://<host_address>:<port_address>/v1/rule/suspend`

Resume Rule

This operation is used to resume rules.

Method: HTTP PUT method

Endpoint: `http://<host_address>:<port_address>/v1/rule/resume`

Update Rule

This operation is used to update rule.

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/rule/update`

Rule Criteria

This operation is used to update rule criteria.

Method: HTTP POST method

Endpoint: `http://<host_address>:<port_address>/v1/rule/criteria`

Add Rule

This operation is used to add rule.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/rule/add

Get Jeopardy Alert Message

This operation is used to get jeopardy alert message.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/plan/alerts

Get All Rules

This operation is used to get all rules.

Method: HTTP GET method

Endpoint: http://<host_address>:<port_address>/v1/rule

Delete Rules

This operation is used to delete rules.

Method: HTTP DELETE method

Endpoint: http://<host_address>:<port_address>/v1/rule/delete

Migrate Plan Fragments

This operation is used to migrate plan fragments.

Method: HTTP POST method

Endpoint: http://<host_address>:<port_address>/v1/migrate/plan-fragment

Plan Time Line

This operation is used to generate plan time line.

Method: HTTP GET method

Endpoint: `http://<host_address>:<port_address>/planTimeLine`

AOPD API Samples

Global Cache Interface

Fetch Product Model from Global Cache

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/productModels`

Parameters: The `productModelIdList` is mandatory.

Fetch Action Model from Global Cache

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/actionModels`

Parameters: The `actionModelIdList` is mandatory.

Product Hierarchy Interface

Operation `getProductHierarchy`

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/plan/product/hierarchy`

Synchronous Plan Generation Interface

Operation `generatePlan`

Method: HTTP POST

Endpoint: http://<host_address>:<port_address>/plan/sync

Example:

```
{
  "orderRef": "string",
  "orderID": "string",
  "header": {
    "description": "string",
    "orderPriority": 9,
    "invoiceAddress": {
      "line1": "string",
      "line2": "string",
      "line3": "string",
      "locality": "string",
      "region": "string",
      "country": "string",
      "postCode": "string",
      "supplementaryLocation": "string"
    },
    "deliveryAddress": {
      "line1": "string",
      "line2": "string",
      "line3": "string",
      "locality": "string",
      "region": "string",
      "country": "string",
      "postCode": "string",
      "supplementaryLocation": "string"
    },
    "notes": "string",
    "slaID": [
      "string"
    ],
    "udf": [
      {
        "name": "string",
        "value": "string",
        "extension": {
          "any": [
            "string"
          ]
        }
      }
    ],
    "extension": {
      "any": [
```

```

        "string"
    ]
},
"customerID": "string",
"subscriberID": "string",
"requiredByDate": "2023-02-06T04:40:55.860Z",
"requiredOnDate": "2023-02-06T04:40:55.860Z"
},
"line": [
    {
        "lineNumber": "string",
        "subscriberID": "string",
        "productID": "string",
        "productVersion": "string",
        "quantity": 0,
        "uom": "string",
        "deliveryAddress": {
            "line1": "string",
            "line2": "string",
            "line3": "string",
            "locality": "string",
            "region": "string",
            "country": "string",
            "postCode": "string",
            "supplementaryLocation": "string"
        },
        "action": "string",
        "actionMode": "string",
        "linkID": "string",
        "inventoryID": "string",
        "notes": "string",
        "slaID": [
            "string"
        ],
        "udf": [
            {
                "name": "string",
                "value": "string",
                "extension": {
                    "any": [
                        "string"
                    ]
                }
            }
        ]
    }
],

```



```

    "characteristic": [
      {
        "name": "string",
        "description": "string",
        "value": [
          {
            "name": "string",
            "type": "string",
            "value": "string",
            "valueFrom": "string",
            "valueTo": "string",
            "extension": {
              "any": [
                "string"
              ]
            }
          }
        ],
        "extension": {
          "any": [
            "string"
          ]
        }
      },
      {
        "customerItemID": "string",
        "extension": {
          "any": [
            "string"
          ]
        },
        "requiredByDate": "2023-02-06T04:40:55.860Z",
        "requiredOnDate": "2023-02-06T04:40:55.860Z"
      }
    ],
    "extension": {
      "any": [
        "string"
      ]
    }
  }
}

```

Operation generatePartialPlan

Method: HTTP POST

Endpoint: http://<host_address>:<port_address>/plan/partial/sync

Example:

```

{
  "planGenerationPhase": "VALIDATION",
  "orderRequest": {
    "orderRef": "string",
    "orderID": "string",
    "header": {
      "description": "string",
      "orderPriority": 9,
      "invoiceAddress": {
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "region": "string",
        "country": "string",
        "postCode": "string",
        "supplementaryLocation": "string"
      },
      "deliveryAddress": {
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "region": "string",
        "country": "string",
        "postCode": "string",
        "supplementaryLocation": "string"
      },
      "notes": "string",
      "slaID": [
        "string"
      ],
      "udf": [
        {
          "name": "string",
          "value": "string",
          "extension": {
            "any": [
              "string"
            ]
          }
        }
      ]
    },
    "extension": {

```

```

        "any": [
            "string"
        ]
    },
    "customerID": "string",
    "subscriberID": "string",
    "requiredByDate": "2023-02-06T04:41:32.048Z",
    "requiredOnDate": "2023-02-06T04:41:32.048Z"
},
"line": [
    {
        "lineNumber": "string",
        "subscriberID": "string",
        "productID": "string",
        "productVersion": "string",
        "quantity": 0,
        "uom": "string",
        "deliveryAddress": {
            "line1": "string",
            "line2": "string",
            "line3": "string",
            "locality": "string",
            "region": "string",
            "country": "string",
            "postCode": "string",
            "supplementaryLocation": "string"
        },
        "action": "string",
        "actionMode": "string",
        "linkID": "string",
        "inventoryID": "string",
        "notes": "string",
        "slaID": [
            "string"
        ],
        "udf": [
            {
                "name": "string",
                "value": "string",
                "extension": {
                    "any": [
                        "string"
                    ]
                }
            }
        ]
    }
]

```

```

    ],
    "characteristic": [
      {
        "name": "string",
        "description": "string",
        "value": [
          {
            "name": "string",
            "type": "string",
            "value": "string",
            "valueFrom": "string",
            "valueTo": "string",
            "extension": {
              "any": [
                "string"
              ]
            }
          }
        ],
        "extension": {
          "any": [
            "string"
          ]
        }
      }
    ],
    "customerItemID": "string",
    "extension": {
      "any": [
        "string"
      ]
    },
    "requiredByDate": "2023-02-06T04:41:32.048Z",
    "requiredOnDate": "2023-02-06T04:41:32.048Z"
  }
],
"extension": {
  "any": [
    "string"
  ]
}
},
"externalBusinessTransactionID": "string"
}

```

Operation generateAmendOrderPlan

Method: HTTP POST

Endpoint: http://<host_address>:<port_address>/plan/amend/sync

Example:

```
{
  "orderLineStatus": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "orderRequest": {
    "orderRef": "string",
    "orderID": "string",
    "header": {
      "description": "string",
      "orderPriority": 9,
      "invoiceAddress": {
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "region": "string",
        "country": "string",
        "postCode": "string",
        "supplementaryLocation": "string"
      },
      "deliveryAddress": {
        "line1": "string",
        "line2": "string",
        "line3": "string",
        "locality": "string",
        "region": "string",
        "country": "string",
        "postCode": "string",
        "supplementaryLocation": "string"
      },
      "notes": "string",
      "slaID": [
        "string"
      ],
      "udf": [
        {

```

```

        "name": "string",
        "value": "string",
        "extension": {
            "any": [
                "string"
            ]
        }
    },
    ],
    "extension": {
        "any": [
            "string"
        ]
    },
    "customerID": "string",
    "subscriberID": "string",
    "requiredByDate": "2023-02-06T04:42:47.661Z",
    "requiredOnDate": "2023-02-06T04:42:47.661Z"
},
"line": [
    {
        "lineNumber": "string",
        "subscriberID": "string",
        "productID": "string",
        "productVersion": "string",
        "quantity": 0,
        "uom": "string",
        "deliveryAddress": {
            "line1": "string",
            "line2": "string",
            "line3": "string",
            "locality": "string",
            "region": "string",
            "country": "string",
            "postCode": "string",
            "supplementaryLocation": "string"
        },
        "action": "string",
        "actionMode": "string",
        "linkID": "string",
        "inventoryID": "string",
        "notes": "string",
        "slaID": [
            "string"
        ],
    },

```

```

    "udf": [
      {
        "name": "string",
        "value": "string",
        "extension": {
          "any": [
            "string"
          ]
        }
      }
    ],
    "characteristic": [
      {
        "name": "string",
        "description": "string",
        "value": [
          {
            "name": "string",
            "type": "string",
            "value": "string",
            "valueFrom": "string",
            "valueTo": "string",
            "extension": {
              "any": [
                "string"
              ]
            }
          }
        ]
      },
      {
        "extension": {
          "any": [
            "string"
          ]
        }
      }
    ],
    "customerItemID": "string",
    "extension": {
      "any": [
        "string"
      ]
    },
    "requiredByDate": "2023-02-06T04:42:47.661Z",
    "requiredOnDate": "2023-02-06T04:42:47.661Z"
  }

```

```

    ],
    "extension": {
      "any": [
        "string"
      ]
    }
  },
  "plan": {
    "planId": "string",
    "orderId": "string",
    "orderRef": "string",
    "originator": "string",
    "planCreationDate": "2023-02-06T04:42:47.661Z",
    "status": "string",
    "statusChanged": "2023-02-06T04:42:47.661Z",
    "description": "string",
    "planStartDate": "2023-02-06T04:42:47.661Z",
    "planStartMillis": 0,
    "isAmendment": true,
    "planItems": [
      {
        "planItemMessages": [
          {
            "planItemId": "string",
            "messageType": "string",
            "messageList": [
              {
                "lineNumber": "string",
                "type": "string",
                "code": "string",
                "description": "string",
                "udfs": [
                  {
                    "name": "string",
                    "value": "string"
                  }
                ]
              }
            ]
          }
        ]
      }
    ],
    "planItemID": "Unique identifier for the plan item within the plan.",
    "planItemName": "Identifier for the plan item name(Product Description) within the plan.",

```



```

        "description": "Description for the plan item.",
        "planFragmentUniqueId": "The id of the process component to
invoke when      executing the plan item.",
        "parentIds": "Plan Item Ids which are dependent on this plan
item",
        "childIds": "Plan Item Ids on which this plan item depends",
        "siblingIds": "Plan Item Ids of sibling products for this plan
item",
        "dependentIds": "Plan Item Ids of dependent products for this
plan      item",
        "startTimestamp": 0,
        "endTimestamp": 0,
        "cancelled": true,
        "isNonExecuting": true,
        "isNoReciprocalAction": true,
        "udfs": [
            {
                "evaluationPriority": "string",
                "type": "string",
                "flavor": "string",
                "name": "string",
                "value": "string",
                "originalValue": "string"
            }
        ],
        "planFragment": {
            "planFragmentUniqueId_CANCEL": {
                "name": "string",
                "description": "string"
            },
            "planFragmentUniqueId_PROVIDE": {
                "name": "string",
                "description": "string"
            },
            "planFragmentUniqueId_CEASE": {
                "name": "string",
                "description": "string"
            },
            "planFragmentUniqueId_UPDATE": {
                "name": "string",
                "description": "string"
            }
        },
        "orderLines": "A list of order lines for this plan item.",
        "action": "Action for the plan item.",

```

```

    "status": "Status for the plan item.",
    "statusChanged": "2023-02-06T04:42:47.661Z",
    "pcRequestType": "Type of request that was send to the process
component.",
    "milestones": [
      {
        "milestoneID": "Unique identifier for this milestone within
the plan item and plan.",
        "description": "Description of the milestone.",
        "ready": false,
        "readyTime": "2023-02-06T04:42:47.661Z",
        "status": "Status for the milestone.",
        "statusChanged": "2023-02-06T04:42:47.661Z",
        "dependencies": [
          {
            "dependencyID": "Unique identifier for this
dependency.",
            "type": "Type of dependency.",
            "ready": false,
            "readyTime": "2023-02-06T04:42:47.661Z",
            "status": "Status of the dependency.",
            "statusChanged": "2023-02-06T04:42:47.661Z",
            "timeDelta": 0,
            "external": {
              "eventName": "Name of the external event that
satisfies this dependency.",
              "eventId": "Unique identifier of the external event
that satisfies this dependency.",
              "eventTime": "2023-02-06T04:42:47.661Z"
            },
            "point": {
              "planItemID": "The plan item Id that uniquely
identifies the plan item that must reach a milestone for this
dependency to be satisfied.",
              "planID": "The plan Id for the plan item Id that
uniquely identifies the plan item that must reach a milestone
for this dependency to be satisfied. If omitted then the
current plan is assumed.",
              "milestoneID": "Warning: CHOICE element",
              "end": "Warning: CHOICE element"
            },
            "time": {
              "absoluteTime": "2023-02-06T04:42:47.661Z"
            }
          }
        ]
      }
    ]
  }

```

```
    ]
  }
]
},
"udves": [
  {
    "evaluationPriority": "string",
    "type": "string",
    "flavor": "string",
    "name": "string",
    "value": "string",
    "originalValue": "string"
  }
]
},
"amendmentNumber": 0
}
```

Broker Service API Samples

- [Actuator APIs](#)
- [Instance Management APIs](#)
- [Recovery APIs](#)

Actuator APIs

- [Get loggers name](#)
- [Post loggers name](#)
- [Actuator root web endpoint](#)
- [Actuator web endpoint 'loggers'](#)
- [Actuator web endpoint 'health'](#)
- [Actuator web endpoint 'health-path'](#)



Note: Health and Info APIs are not secured so that they can interact with other services easily.

Post loggers name

Get Configuration

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/management/loggers/{name}`

Parameters: name (mandatory)

You can find the possible name parameters in the `$OM_HOME/roles/<service-name>/standalone/config/logback_<service-name>.xml` file.

Sample:

```
curl -X 'POST' \
  'http://localhost:9105/management/loggers/com.tibco.fom.bootstrap' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "configuredLevel": "INFO, WARNING, ERROR"
  }
```

Actuator root web endpoint

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management

```
curl -X 'GET' \
  'http://localhost:9105/management' \
  -H 'accept: */*'
```

Actuator web endpoint 'loggers'

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management/loggers

Sample:

```
curl -X 'GET' \
  'http://localhost:9105/management/loggers' \
  -H 'accept: */*'
```

Actuator web endpoint 'health'

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/management/health

```
curl -X 'GET' \  
  'http://localhost:9105/management/health' \  
  -H 'accept: */*'
```

Actuator web endpoint 'health-path'

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/management/health/{name}`

Parameters: The name path parameter can have one of the following values:
readiness or liveness

Example:

```
curl -X 'GET' \  
  'http://localhost:9105/management/health/readiness' \  
  -H 'accept: */*'
```

Instance Management APIs

Register a new instance to Broker Service

This API accepts requests to register an instance.



Note: Note: The broker service does not verify the health status of the instance during registration. It verifies the health status during the next status verification cycle.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/instance/register`

Example:

```
{
  "instanceId": "string",
  "serviceName": "string",
  "ipAddress": "string",
  "port": 0
}
```

Instance Ping Operation

At regular intervals, the instance computes its status and sends a ping to the Broker service. Based on the status, the broker might initiate the transfer of order messages to another instance.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/instance/ping`

Example:

```
{
  "instanceId": "string",
  "status": "string",
  "inactiveComponentList": [
    "string"
  ]
}
```

Instance Clean Up Operations

This API accepts a list of instance IDs. It then purges instances from the database whose status is down. If an instance is still up while this operation is being performed, the Broker does not purge that instance.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/instance/cleanup`

Example:

```
[
  "string"
]
```

Get Instance Details by Service Name

Given a service name, this operation returns all details of instances associated with that service name.

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/instance`

Parameters

Name	Type	Value
serviceName (mandatory)	string	Name of the service
pageNo (mandatory)	integer	Page number
instancePerPage (mandatory)	integer	Instance per page

Example:

```
{
  "instanceDetailsList": [
    {
      "instanceId": "string",
      "serviceName": "string",
      "ipAddress": "string",
      "port": 0,
      "status": "string"
    }
  ],
  "totalPages": 0,
  "currentPage": 0,
  "instancesPerPage": 0
}
```

Get Instance Details by Instance Id

Given an instance ID, this operation returns the details of that instance.

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/v1/instance/{instanceId}

Parameters

Name	Type	Value
instanceId (mandatory)	string	Id of the instance

Example:

```
{
  "instanceId": "string",
  "serviceName": "string",
  "ipAddress": "string",
  "port": 0,
  "status": "string"
}
```

Get All Available Services

This operation returns all service names available with the Broker Service.

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/v1/instance/services

Example:

```
[
  "string"
]
```

Get All Instance Details

This operation returns all details of instances available with the Broker service.

Method: HTTP GET

Endpoint: http://<host_address>:<port_address>/v1/instance/all

Parameters

Name	Type	Value
pageNo (mandatory)	integer	Page number
instancePerPage (mandatory)	integer	Instance per page

Example:

```
{
  "instanceDetailsList": [
    {
      "instanceId": "string",
      "serviceName": "string",
      "ipAddress": "string",
      "port": 0,
      "status": "string"
    }
  ],
  "totalPages": 0,
  "currentPage": 0,
  "instancesPerPage": 0
}
```

Unregister an instance from Broker Service

This asynchronous operation unregisters the instance from the Broker service.

The unregistration operation does not immediately delete the instance. Instead, it attempts to validate whether the instance is down by verifying its health status.

The operation performs this validation several times. If the instance is not down, it does not get purged. However, if the instance goes down during this process, it is purged from the database.

Method: HTTP DELETE

Endpoint: `http://<host_address>:<port_address>/v1/instance/un-register/{instanceId}`

Parameters

Name	Type	Value
instanceId (mandatory)	string	Id of the instance

Recovery APIs

Re-send pending notifications

This API re-sends pending notifications or request by list.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/notifications/re-send`

Re-send all pending notifications

This API re-triggers all pending notifications or requests.

Method: HTTP POST

Endpoint: `http://<host_address>:<port_address>/v1/notifications/re-send/all`

Get pending notifications

This API gets a list of notifications or requests for which the replies are yet to be received by the Orchestrator.

Method: HTTP GET

Endpoint: `http://<host_address>:<port_address>/v1/notifications/pending`

Response Codes and Messages

This appendix lists the response codes and message details.

Error Codes

The following error codes are currently implemented.

Service	Code	Message
Common HTTP Rest Error Codes	500	Internal Server Error
	401	Full authentication is required to access this resource
	404	Resource Not Found
Submit Order	400	Circular dependency in product model
	400	DUPLICATE_ORDER_LINE
	404	INVALID_PRODUCT_IN_ORDER
	400	PLANFRAGMENT_MAPPING_NOT_FOUND
Withdraw Order Service	404	Order already completed. Cannot withdraw
	404	Invalid OrderID in WithdrawOrderRequest
	404	Invalid OrderRef in WithdrawOrderRequest
Get Order Execution Plan	404	PLAN_NOT_FOUND

Service	Code	Message
Get Order Detail	404	ORDER_NOT_FOUND
AuditTrail	404	
Data Service	FOM-DATA-ACCESS-ORDER-NOT-FOUND-9999	OrderID not found
	FOM-DATA-ACCESS-PLAN-NOT-FOUND-9999	PlanID not found
	FOM-DATA-ACCESS-INVALID-REQUEST-9999	Invalid payload in GetPlanRequestEvent. Both <ALL_PI_SINGLE_UDF_NAME> and <MATCHING_PI_UDF_NAME_VAL> properties are present in the request header.
	FOM-DATA-ACCESS-INVALID-REQUEST-9999	Invalid payload in GetPlanRequestEvent. MATCHING_PI_UDF_NAME_VAL in the request is in incorrect format.
	FOM-DATA-ACCESS-INVALID-INPUT-9999	Invalid value 'string' for input planItemID in SetPlanItemRequestEvent
	FOM-DATA-ACCESS-INVALID-INPUT-9999	PlanItemID not found

TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [Product Documentation website](#), mainly in HTML and PDF formats.

The [Product Documentation website](#) is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The documentation for this product is available on the [TIBCO® Order Management Product Documentation](#) page.

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our [product Support website](#).
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the [product Support website](#). If you do not have a username, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature

requests from within the [TIBCO Ideas Portal](#). For a free registration, go to [TIBCO Community](#).

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, ActiveMatrix BusinessWorks, TIBCO Runtime Agent, TIBCO Administrator, and Enterprise Message Service are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: <https://scripts.sil.org/OFL>

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.cloud.com/legal>.

Copyright © 2010-2024. Cloud Software Group, Inc. All Rights Reserved.