

# TIBCO® Order Management - Long Running

## Best Practices

*Version 5.0.1*

*April 2022*



# Contents

---

<b>About this Product</b> .....	<b>4</b>
<b>Introduction to TIBCO Order Management - Long Running Best Practices Guide</b> .....	<b>5</b>
Purpose of the Document .....	5
Scope of the Document .....	5
<b>General Installation Guidelines</b> .....	<b>6</b>
<b>Hardware Configuration Guidelines</b> .....	<b>7</b>
Hardware Selection Guidelines .....	7
Hardware Sizing Guidelines .....	7
Fault Tolerance or Load Balancing Considerations .....	7
<b>Software Configuration Guidelines</b> .....	<b>9</b>
Software Configuration Guidelines for TIBCO Product and Service Catalog .....	9
Software Configuration Guidelines for TIBCO Order Management - Long Running .....	9
<b>Architecture</b> .....	<b>11</b>
TIBCO Product and Service Catalog Strategy .....	11
Other Considerations .....	11
Order Enrichment .....	12
<b>Product Catalog Guidelines</b> .....	<b>13</b>
Best Practices for Catalog Modeling .....	13
Product Modelling Patterns .....	14
Functional Product Initializer .....	14
Helper Product Layer .....	15
Functional Coordination Helper Product .....	16
Use Case Layer Pattern .....	17
<b>Order Fulfillment Guidelines</b> .....	<b>19</b>
Order Management System Guidelines .....	19
Orchestrator Guidelines .....	19
Automatic Order Plan Development Guidelines .....	20
Transient Data Store Guidelines .....	20
<b>Development Guidelines</b> .....	<b>22</b>
Team Organization Guidelines .....	22
Code Organization Guidelines .....	22
Environment Organization Guidelines .....	22
<b>Performance Tuning Guidelines</b> .....	<b>23</b>
Performance Tuning Guidelines for TIBCO Order Management - Long Running .....	23
Performance Tuning Guidelines for Infrastructure .....	23
<b>Guidelines for Operations</b> .....	<b>24</b>

Guidelines for Monitoring .....24

Guidelines for Logging ..... 25

Database Purging or Cleanup Guidelines .....25

**TIBCO Documentation and Support Services ..... 26**

**Legal and Third-Party Notices ..... 27**

## About this Product

---

TIBCO® Order Management is an elastic, catalog-driven order management system for digital service providers. It accepts orders from any customer engagement system and orchestrates the tasks required for fulfilling the orders.

TIBCO Order Management is the next generation of TIBCO® Fulfillment Order Management and partially replaces the old product. To better align TIBCO Fulfillment Order Management with market demand, the product's capabilities have been reorganized into two new products: TIBCO® Order Management and TIBCO® Offer and Price Engine.

TIBCO Order Management is further divided into variant products:

- **TIBCO® Order Management - Low Latency:** Use this new product for scalable processing of low-latency orders
- **TIBCO Order Management - Long Running:** This product continues to support the processing of long-running orders

# Introduction to TIBCO Order Management - Long Running Best Practices Guide

---

The *TIBCO Order Management - Long Running Best Practices* guide provides a variety of best practice guidelines that you can utilize throughout your TIBCO Order Management - Long Running installation. These best practices are based on the experiences gained at large and small customer sites.

The TIBCO Order Management - Long Running best practices described in the guide include agreeing on requirements to ensure successful project implementation and determining hardware and software configuration guidelines.

Architecture best practices and design considerations are also addressed so that the implementation gets started on the right foot.

The guide also addresses configuration management that involves knowing the state of all artifacts that make up your system or project, managing the state of those artifacts, and releasing distinct versions of a system.

Performance testing, usually the last resort to catch application defects, is also covered in the guide. It is labor intensive and usually only catches coding defects. Architecture and design defects might be missed. One method to catch some architectural defects is to simulate load testing on the application before it is deployed and to deal with performance issues before they become problems.

## Purpose of the Document

The best practices described in the *TIBCO Order Management - Long Running Best Practices* guide assists you with many of the configuration and implementation steps, you encounter during your implementation.

The purpose of the document is to assist in the execution of the TIBCO Order Management - Long Running projects, by providing recommendations and guidelines for its implementation. The content is based on the experiences gained in the field.

When the information provided is predominantly general, you can always adapt it to the specific situation at hand. Recommendations might be taken in the context of the business and technical requirements.



- This document is not intended as a replacement for the official TIBCO product documentation provided with your product.
- This document is also not intended as a training reference or tutorial. Basic knowledge of TIBCO® Product and Service Catalog, TIBCO Order Management - Long Running, and its supporting software is assumed.

## Scope of the Document

The *TIBCO Order Management - Long Running Best Practices* guide describes the tips, tricks, and best practices for use with the latest TIBCO Order Management - Long Running software.

The scope of the document includes the following software:

- TIBCO Order Management - Long Running version 5.0
- TIBCO Product and Service Catalog version 5.0

## General Installation Guidelines

---

You might be familiar with which type of product documentation to use when working with *TIBCO Order Management - Long Running*.

TIBCO product documentation might always be the primary reference for installation. However, for tips, tricks, and best practices refer to *TIBCO Order Management - Long Running Best Practices* guide that includes guidelines for installation and configuration for all the required supporting TIBCO software, including TIBCO® Enterprise Message Service.

# Hardware Configuration Guidelines

---

When selecting and configuring your hardware for the TIBCO Order Management - Long Running you have to consider the machine type best suited for your environment and the hardware sizing requirements and load balancing recommendations.

## Hardware Selection Guidelines

You have to select certain core hardware components for TIBCO Order Management - Long Running using the guidelines listed in the specific TIBCO documentation.

Use these guidelines to select key hardware components for TIBCO Order Management - Long Running:

- The Order Management Orchestrator and Automatic Order Plan Development are core components of TIBCO Order Management - Long Running.
- For TIBCO Product and Service Catalog, refer to the TIBCO Master Data Management documentation for guidelines including the *Installation Overview* and *Planning for Installation* sections in the *TIBCO Master Data Management Installation and Configuration* guide.

## Hardware Sizing Guidelines

You must consider certain non-functional requirements when designing infrastructure topography. You also have to follow specific guidelines before making hardware recommendations to a customer.

An example of a non-functional requirement involves putting logging and operational Enterprise Message Service (EMS) traffic on the same Enterprise Message Service instance. However, this leads to delays in processing orders because the orders are blocked by non-business critical logging traffic. A better solution is to create separate Enterprise Message Service instances for logging and core Enterprise Message Service traffic to eliminate this blockage.

Hardware sizing is generally an outcome from preliminary performance testing. All environments are different, and you cannot use a cookbook approach to a particular order management problem. It is a good practice to complete extensive performance testing on lab hardware before recommending particular hardware to a customer. These lab runs must reflect a realistic order size, load mix, and anticipated plan size. The plan must also cover pre-submit and post-submit activities, in particular, validation and eligibility calls to Offer and Price Engine (OPE).

## Fault Tolerance or Load Balancing Considerations

Fault tolerance and load balancing recommendations vary depending on the TIBCO application you are using.

The following recommendations apply for fault tolerance and load balancing. Also, refer to the best practices for Enterprise Message Service.

*Fault Tolerance and Load Balancing Recommendations*

Application	Fault Tolerance	Load Balancing
TIBCO Enterprise Message Service	<p>Dual server deployment with shared database state and active-passive configuration. Managed by Enterprise Message Service, heartbeats are exchanged between the server instances to determine on failover when to alternate the server. Requirement for shared state, either Storage Area Network or database. Generally Storage Area Network is recommended for performance reasons, but this incurs additional hardware and software requirements as outlined in Enterprise Message Service documentation.</p> <p>The alternative is to use OS-level clustering active-passive pair with the clustering software guaranteeing only one Enterprise Message Service instance accesses the shared state at any one time.</p>	<p>Dual server deployment by using active-active configuration, managed by Enterprise Message Service. The requirement for shared state by using Storage Area Network or database, with Storage Area Network preferred.</p> <p>Generally most customer implementations do not require a load balancing configuration.</p>
TIBCO Order Management - Long Running System	Dual server deployment. Third-party load balancer used to send UI client requests to the active server.	Run in active-active configuration and messages load the balance from Java Message Service queues. Do not use SOAP over HTTP interfaces to remove the requirement for a load balancer.
TIBCO Master Data Management TIBCO Product and Service Catalog	Dual server deployment, JBoss cluster managing active-passive pair. Third-party load balancer used to send client requests to the active server. Shared storage required to maintain a state between servers.	This is a design-time component, not a high-load application and it does not require load balancing. However, load balancing could be implemented by using JBoss cluster and active-active pair.



# Software Configuration Guidelines

---

For the best results, you must follow these guidelines to configure your software for TIBCO Product and Service Catalog and TIBCO Order Management - Long Running. Planning and configuring these components correctly enhances system performance.

## Software Configuration Guidelines for TIBCO Product and Service Catalog

It is important to configure your TIBCO Product and Service Catalog correctly to enhance your system performance. It is a good practice to plan and complete the configuration in conjunction with appropriate performance testing and tuning.

Here are some recommended best practices:

- Use Master Data Management Configurator to create the TIBCO Product and Service Catalog enterprise.
- when using the full or partial import, use the template provided by TIBCO Product and Service Catalog. To generate the template, navigate to **Product and Service Catalog Operation>Export FC data->Blank Template**.
- When importing data (generated by another enterprise) into an enterprise, ensure that that drop down values are in sync between those two enterprises. For example, for the PRODUCT repository in the importing enterprise, the RECORD\_TYPE drop-down must have all those values that were present in the exporting enterprise.
- Whenever any repository is modified (either an attribute is added or deleted), the corresponding data source and Input Map must be updated before using the Full or Partial Import feature so that the feature works correctly.

## Software Configuration Guidelines for TIBCO Order Management - Long Running

Configure your TIBCO Order Management - Long Running components correctly to enhance your system performance. Ensure that you plan and complete the configuration in conjunction with appropriate performance testing and tuning.

Configuring components of TIBCO Order Management - Long Running essentially means determining how many instances of each type of engine must be deployed across the number of physical servers available. Also, how much physical RAM must be allocated to each engine. The product documentation provides basic recommendations for a minimal setup. For details about the setup, see "Installation and Deployment Options" in the *TIBCO Order Management - Long Running Installation and Configuration* guide.

Here are some recommended best practices:

- For a typical production setup containing two physical servers, use the basic starting configurations – two instances TIBCO Order Management - Long Running - Orchestrator, Automatic Order Plan Development, and Jeopardy (per server).
- Set identical minimum and maximum heap sizes.
- Set 8 GB of heap size for each TIBCO Order Management - Long Running instance.

Use Configurator to perform TIBCO Order Management - Long Running configurations.

### TIBCO Order Capture System

Regarding system sizing, the number of Order Capture System instances and whether or not a load balancer is needed depends on the number of concurrent users, the desired availability of the system (a single node system can be unavailable in case of crash or maintenance), and the specifications of the machine hosting Order Capture System.

The same considerations apply to Order Capture System database sizing. Order Capture System only uses the database for the temporary storage of the shopping cart.

# Architecture

---

There are several architect strategies to consider when working with TIBCO Order Management - Long Running.

TIBCO Order Management - Long Running architecture involves creating an effective product catalog strategy in addition to making smart validation choices. You might also be aware of the best practices related to enrichment strategy and general recommendations.

## TIBCO Product and Service Catalog Strategy

You can use TIBCO best practices to determine how best to utilize TIBCO Product and Service Catalog in your business environment. This requires a strategy for handling your configuration requirements and decisions when running the TIBCO Product and Service Catalog in either a live production system or in a deployment environment.

You can configure TIBCO Order Management - Long Running so that the product model dynamically publishes through messaging from TIBCO Order Management - Long Running (or potentially another product catalog system). You can also configure TIBCO Order Management - Long Running to read the product catalog from off-line files.

You can make changes to the TIBCO Product and Service Catalog in a live production system; however, you must be aware of the implications. The following implications show the potential danger of a live TIBCO Product and Service Catalog instance dynamically publishing the product model to TIBCO Order Management - Long Running:

- If the catalog is changed incorrectly, the production system is immediately impacted.
- TIBCO Order Management - Long Running requires a live instance of TIBCO Product and Service Catalog.
- New products generally mean new process components and most likely a software release would be required anyway.

It is a good practice to use a live TIBCO Product and Service Catalog instance only in the development environment. All other environments, including test, pre-production, and production can use the off-line files, which are easily version controlled and tagged in conjunction with the developed code.

In summary, carefully plan how to use the TIBCO Product and Service Catalog in your organization in conjunction with your business requirements.

## Other Considerations

When you are designing your system architecture there are several general recommendations and best practices you might consider. These best practices cover end-to-end order tracking and ensure that you do not lose any data.

### OrderRef

OrderRef, an external order ID, must be unique in all TIBCO Order Management - Long Running orders. If you submit an order without a unique OrderRef, the Order Management System does not treat it as a new order and attempts to amend the previous order.

If you are using a non-unique external ID to map to OrderRef or an external ID that you can use with different TIBCO Order Management - Long Running orders, it is a good practice to add the external ID as a header User Defined Format in the order.



User Defined Formats in the TIBCO Order Management - Long Running have limited out-of-the-box searchability.

## General Recommendations

It is important when you are designing your architecture to incorporate procedures that prevent order data loss. It is a good practice to review the following procedures before designing your architecture solution:

- Ensure that your architecture solution includes end-to-end order tracking capability and visibility similar to that used in the order system of the TIBCO Order Management - Long Running.
- Include recovery and repair capabilities in your design. For example, you need a plan for handling a situation when a fulfillment process call fails to connect to a back-end system. Having a plan in place prevents you from having to stop the order.
- Determine the interfaces that the operation needs to complete this work. Resending an order might not be a viable or reasonable option.
- Include target group representatives when planning manual steps and activities.
- Validate your error and exception handling approach with other representatives in the order processing chain from front-end solutions to back-end systems.
- Ensure that your solution architecture accounts for the different capabilities in each system area. For example, a front-end system might not be able to resubmit an order after it is submitted, when a back-end system can receive order data again without considering it an amendment.

## Order Enrichment

When you are fulfilling an order you might have to reference existing data in the network inventory or customer installed base systems. TIBCO offers a sequence of steps you can follow to add this type of information to your order.

### Custom Order Enrichment

It is a good practice to add reference information to the order before it is validated and submitted to TIBCO Order Management - Long Running. As a best practice, you must use a custom order enrichment component to add the necessary data information during the first step.

The custom order enrichment component is based on generic rules and retrieves all of the required validation information, order plan development, and provisioning from the source systems in one step. It then adds the information to the order before it is validated and submitted to TIBCO Order Management - Long Running.

### User-Defined Format

You can add this information through the User Defined Format (UDF) data. This format is a better option because it eliminates the need for process components to complete the retrieval, which might result in less efficient behavior including redundant or fragmented database queries.

## General Recommendations

If you perform validation after enrichment, remove the validation data before submitting the information to TIBCO Order Management - Long Running. This process flow helps eliminate data created in the data interface for each plan item.

It is a good practice to retain any information retrieved to support plan development. This includes information referenced in attribute-based decomposition rules or information required for provisioning.

# Product Catalog Guidelines

---

When creating a product model you might consider your overall product catalog design and how to use the TIBCO Product and Service Catalog in your product modeling. TIBCO provides some example modeling patterns you can use to help manage the product life cycle.

## Best Practices for Catalog Modeling

You must be aware of the best practices to use when creating a product model. You have to consider your product catalog design and how TIBCO Product and Service Catalog can play a role in your product modeling. You also have to consider how to manage the product life cycle.

### Effective Product Modeling

The goal of creating a good product model is to identify correct process components and include the correct number and type of products to maximize reuse. It is a good practice to use smaller catalogs because they improve system performance.

Do the modeling of components as many as are required.

### Product Catalog Design

Designing a product is a delicate balance between a pure product model view versus a model that contains enough technical detail to ensure that you can create correct fulfillment plans for orders.

In a typical design process, a product manager enters an initial view of a product into the catalog and the engineer then adds additional technical artifacts to the model to ensure there is enough information to correctly drive the fulfillment of orders.

Occasionally, you have to add additional products to the catalog that do not map to an actual TIBCO product. TIBCO refers to these products as "helper products," and you can add them to achieve a specific sequence of fulfillment actions or to ensure design principles are met.

### TIBCO Product and Service Catalog

TIBCO Product and Service Catalog can play a role in your product modeling by functioning as:

- A master for product data
- The user of master product data from one or more other systems or catalogs
- TIBCO Master Data Management solution, integrating with both masters and users of catalog data

The complexity of the last two can increase as the number of systems grows.

### Integrating TIBCO Product and Service Catalog with Other Systems

You can integrate TIBCO Product and Service Catalog with other systems; however, you must be aware of the level of automation with other systems and the catalog model capabilities.

There are a number caveats to consider when using other systems with TIBCO Product and Service Catalog. For example, other systems might require data that has no natural representation in TIBCO Product and Service Catalog, including user interface related issues. You also have to consider the sequence in which you want to present the items and the arrangement of the Order. For example, how do you arrange the order if a choice is required that impacts the sequence?

Additionally, other systems might impose restrictions on model structures for performance or management reasons rather than technical or capability reasons.

These issues might require you to define certain patterns and restrictions when you model items in TIBCO Product and Service Catalog regardless of its underlying functionality.

Ensure that the model you are using is compatible with the TIBCO Product and Service Catalog release version.

### Product Life Cycle Management

The following information lists the expected product life cycle management processes you might have to manage:

- Define new products and offerings.
- Validate and or tests new products and offerings.
- Approve new products and offerings.
- Act as a decision maker for retiring or replacing a product.
- Build or connect the required fulfillment elements.
- Validate and or test the fulfillment elements.
- Approve the fulfillment elements.
- Deploy changes and determine how and when changes are deployed.
- Determine the dependencies for life cycle management in other integrated systems.

It is important to understand the consequences of the decisions you make during the life cycle process. The issues you might encounter during the life cycle process range from very fine to very granular and might include the following issues:

- Adapting small, simple processes or process changes before larger changes.
- Filling any architectural gaps for these processes.
- Avoiding business process management modeling.-
- Considering the granularity of products in the catalog.

## Product Modelling Patterns

There are several example modeling patterns that you can apply to your specific business environment.

The following table lists and describes certain terminology used in this section.

Term	Description
Functional Product	A product that is ordered
Technical Product	A product that the functional product decomposes into, through a Product-Comprised-Of relationship (and autoprovision=true). These products are not displayed on the order.

### Functional Product Initializer

There are some business scenarios where you cannot use a standard TIBCO Order Management - Long Running pattern. In these cases, there is a workaround pattern available to address your particular scenario.

#### Pattern Rationale

The standard TIBCO Order Management - Long Running plan development specifies that for any action ordered on a parent product, plan items (such as process components) for children must be executed first, before the parent. This can create a problem in some scenarios because you might have to operate for the parent before performing operations related to the children. For example, initializing an inventory entry for the parent product that the children have to refer to.

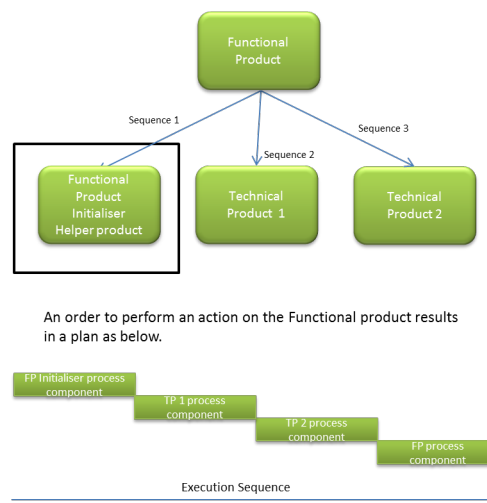
In this case, because the process component for the parent product is always executed last, you cannot use this pattern.

### Pattern Realization

The workaround for this is to create a helper child product (functional product initializer) for the functional product, whose process components for any given action are always executed first in the plan for an order. The process component for the functional product initializer helper could, for example, create an entry for the functional product in an inventory system.

To maximize reusability, this pattern could define two helper products, one for create and one for update, which enables you to reuse the entire product across multiple functional products directly.

Through splitting by action, maximum flexibility in specifying process components is achieved. For example, if you require a create and an update process component in the same action, you could use this pattern.



## Helper Product Layer

In certain situations, you cannot specify certain sequences in the standard TIBCO Order Management - Long Running program, and you have to use multiple back-end calls to perform a single action. To simplify the process TIBCO provides a layer of helper products to use as a workaround in this situation.

### Pattern Rationale

The standard TIBCO Order Management - Long Running behavior specifies that an action ordered on a parent product results in an execution plan containing the process components for the same action for the child products.

In some cases, you might have to use a CEASE action to implement an UPDATE action followed by a PROVIDE action. However, you cannot specify this sequence in standard TIBCO Order Management - Long Running.

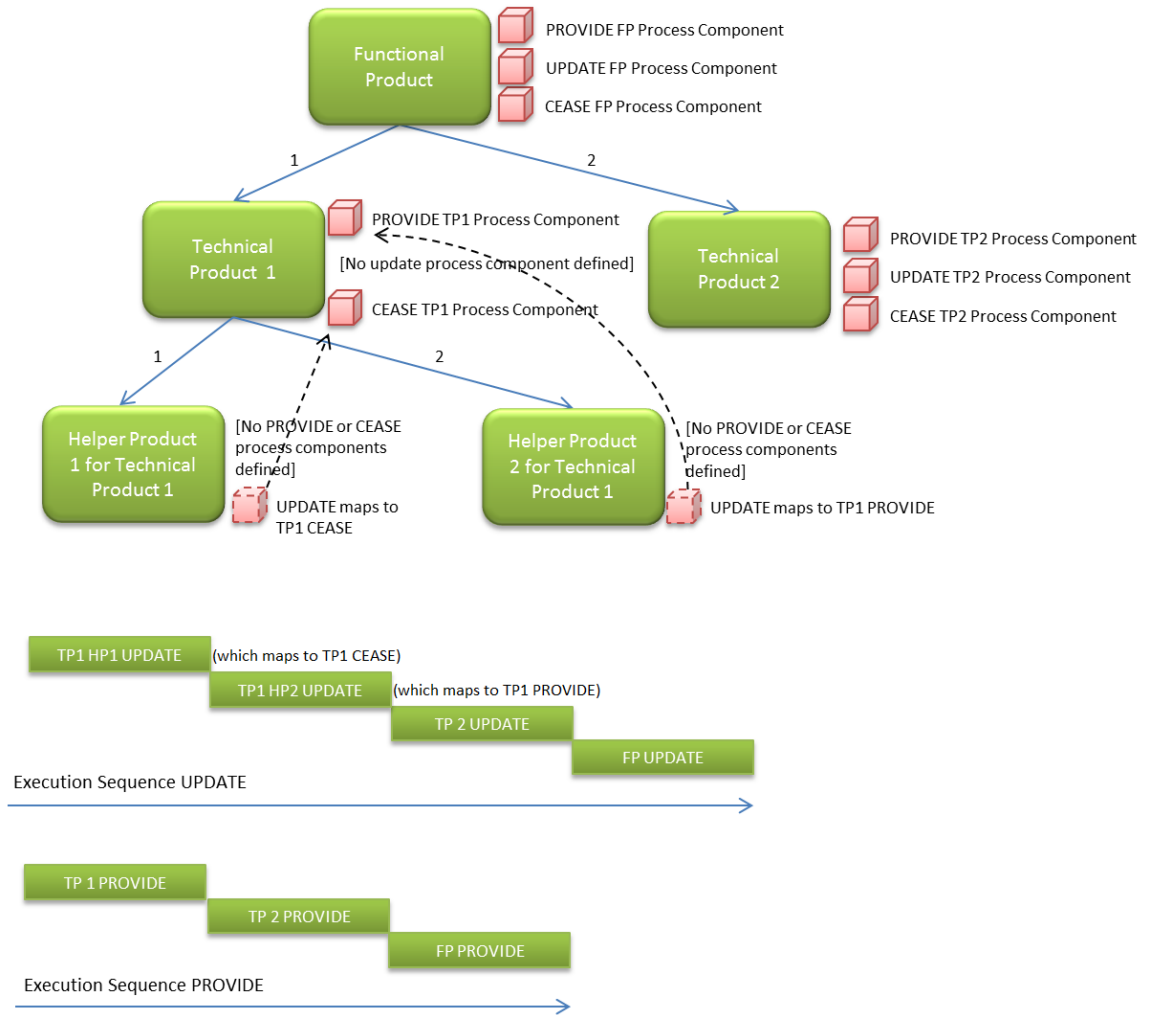
This example requires multiple back-end calls to perform a single action on a product.

### Pattern Realization

The workaround for this situation is to provide a layer of helper products below the technical product layer, one for each of the back-end calls.

In this case, TIBCO Order Management - Long Running handles the orchestration enabling you to keep the process components simple and directly reuse process components. For example, if you have to update a

technical product through a CEASE action followed by a PROVIDE action, those process components can be directly reused by the helper products defined for those actions.



### Functional Coordination Helper Product

Certain situations require an activity for the functional product during the fulfillment of the technical product as compared to the usual scenarios where the activity takes place after the child products are complete. To make the process components to run in parallel, you can create a functional coordinator helper product to perform activities related to the functional product.

#### Pattern Rationale

In an ideal situation, process components for a product action must not assume anything about the parent product that they are comprising. However, some fulfillment scenarios, such as provisioning an access connection, require some activity at the functional product level during the fulfillment of a lengthy technical product.

These activities might include sending an intermediate status message to the submitting party, which can only occur at the functional product level or updating inventory associated with the functional product.

Activities related to the parent product normally occur after the fulfillment activities (process components) for all of the child products are complete. However, in this case, we need an activity for the functional product during the execution of fulfillment of the technical product. This means that the process components must run in parallel.

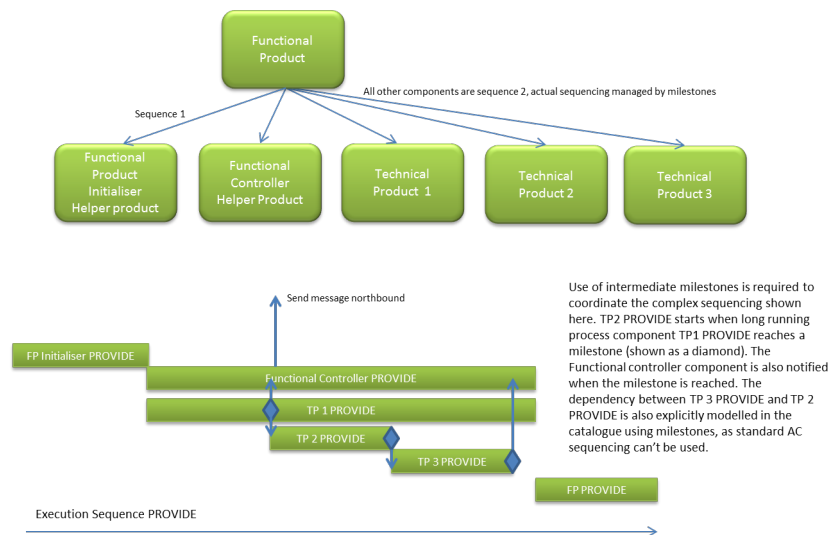


## Pattern Realization

To address this situation, you can create a functional coordinator helper product to perform any activity related to the functional product during the fulfillment of technical products. The sequence number of the functional coordinator must be the same for all technical products so they can run in parallel.

To define relationships between functional and technical product process components, you can use intermediate milestones in the catalog. In this way, communication between functional and technical products is brokered through the TIBCO Order Management - Long Running instead of direct communication.

Technical products simply trigger milestones when defined events occur (for example, receiving a planned message from the south) with no knowledge of any required functional product activity, such as sending a notification to the sending party.



## Use Case Layer Pattern

In the current TIBCO Order Management - Long Running product, you can define only one sequence of activities for a given action. However, you can now use a new layer of helper products in the hierarchy to provide the same flexibility for combining tasks for different actions in the same use case.

### Pattern Rationale

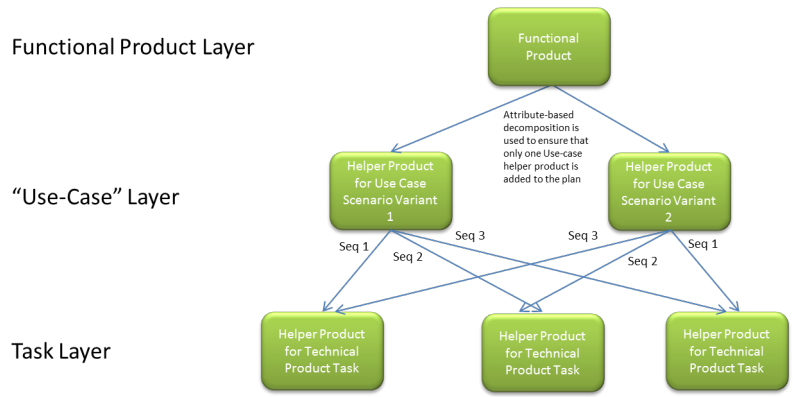
Some use case scenarios require different sequencing for different variants of the same action, for example, an update activity. However, in the current TIBCO Order Management - Long Running product, you can define only one sequence of activities for a given action.

### Pattern Realization

The solution is to use a new layer of helper products in the hierarchy, representing the use case scenario. For a given order, attribute-based decomposition is used to ensure that only one use case helper product is included in the execution plan, making them mutually exclusive.

A layer of tasks is implemented below the layer representing the point of reuse. With the use case layer, the same tasks can be combined in different sequences for the same action. There is no explicit modeling of the technical product, only the tasks that operate on that product.

This pattern is an extension of the helper product layer pattern described previously and provides the same flexibility for combining tasks for different actions in the same use case.



# Order Fulfillment Guidelines

---

TIBCO Order Management - Long Running requires that you follow certain guidelines when viewing and submitting orders. These guidelines and best practices help you achieve optimal performance in your business environment. There are also guidelines and best practices to follow when using the interfaces to develop an automatic order plan.

## Order Management System Guidelines

If orders are to be viewed in Order Management Server, the Order Management Server services must be used to submit orders.

Order Management Server services are slightly different than Orchestrator. Amendments are processed through the AmendOrder and CancelOrder interfaces. Submitting an amendment through SubmitOrder results in the amendment being ignored.

Ensure that the plan fragment model has been synchronized correctly to Order Management Server before viewing the Gantt chart. If the model is not synchronized, it results an error in the web interface.

## Orchestrator Guidelines

If orders are to be viewed in Order Management Server, then the Order Management Server services must be used to submit orders.

Notifications have an impact on overall Orchestrator performance. If notifications are not required, they must be disabled through the use of global variable flags. Turning off notifications in Orchestrator impacts the Activity Log in Order Management Server as well. In all cases, the data element of notifications must be turned off unless necessary.

Orchestrator provides data interfaces for retrieving orders and plans. Invoking these on a live system has a performance impact, so they must only be used when necessary. In most cases, the same information is already available in Order Management Server. Orchestrator provides a retry framework for process components in the event of a failure. This makes use of the cache server and a timer to retry. The system has been designed for this, but it might be more efficient to implement retry handling internally within the process component instead of relying on Orchestrator to do it. This is particularly true in the case of IPE process components.

When implementing error handling for process components, you can configure different error handler names in the plan fragment model. By default, Orchestrator specifies the error handler name as configured in global variables. If one is specified in the plan fragment model for a given process component, then that is specified in the call to the Plan Item Failed Handler. These names have no meaning within Orchestrator, but if necessary might be used within the Plan Item Failed Handler or subsequent error handling framework to route errors.

Scheduler polling interval in global variables is used to configure the cache server scheduler for timed events. This is used for time dependencies, for feasibility, plan development, and process component failures. This must not be set to less than 60000 milliseconds.

There are three methods currently available for cleaning up orders within Orchestrator:

1. Shut down the Orchestrator Cache and Agent engines and purge the backing store database by using SQL scripts. This must be done with the Transient Data Store at the same time. Restart Orchestrator Cache and Agent. This results in an outage of the system when the cleanup occurs.
2. Make use of the CleanupAtEndOfOrder option in Orchestrator that automatically purges the order from the system when it reaches a COMPLETE, CANCELLED, or WITHDRAWN state. The order and plan details are retained within Order Management Server, but no history is available in Orchestrator.
3. Make use of the DeleteOrder/DeletePlan operations in Orchestrator. This requires the use of an external project to determine that orders to purge and then invoke these services with the list of orderIDs. This must be done during a slow period in order processing due to the performance impact.

## Automatic Order Plan Development Guidelines

If orders are to be viewed in Order Management Server, then the Order Management Server services must be used to submit orders.

Plan development performance is related to the size of the catalog and the order being decomposed. If possible reduce the size of both to improve performance.

Do not model plan fragments that do not do anything at execution time. Only plan items that do useful work might go into the plan.

Make use of milestones and overlapping sequencing instead of empty plan items for dependencies.

When making use of Attribute-Based Decomposition, be aware of the potential performance impact of the evaluation of XPath expressions (XSLT processing) in TIBCO BusinessEvents.

## Transient Data Store Guidelines

Transient Data Store data is replicated to the Order Management Server through the use of bridges. If the data in Order Management Server is out of date, verify that the correct bridges are in place for order and plan data.

Transient Data Store stores both order and plan data:

- Order data is read only.
- The Plan data is read-write data.

To make changes to order data in Transient Data Store, the updated order must be submitted as an amendment to Order Management Server and processed accordingly.

The process components must not make use of order data. Instead, they must use plan data.

Order data is stored in large XML CLOBs within the Transient Data Store backing store. Operations are provided to retrieve orders, but extensive use of these causes a performance impact.

Plan data is stored in concepts. Operations are provided to retrieve plans and plan items. This is the preferred way of retrieving data within Process Components.

The Transient Data Store is designed to be a pass-by-reference data cache. This means that duplicate data must not be replicated across plan items. One plan item must be a designated master for any given User Defined Field. If a particular User Defined Field is required by a plan item that doesn't master this data, then the master plan item must be retrieved as well.

Where possible, process components must retrieve only data for its own plan item by using the GetPlanItems interface. If the process component requires data from other plan items, it must retrieve those by using the same interface at the same time if the planItemIDs of those other plan items are known.

It might be necessary for the process component to invoke GetPlanItems multiple times to determine the planItemIDs of dependent plan items. Where it is possible, bundle these into the same GetPlanItems call to reduce the number of invocations.

Retrieving the entire plan from the Transient Data Store by using GetPlan can be less efficient than retrieving plan items by using GetPlanItems. This is particularly the case for larger plans where a process component needs only a limited set of data. However, if it is necessary to do multiple GetPlanItems calls to determine dependent plan items, it might be more efficient to do a single GetPlan instead to get all the information. Make use of the IDs-only option to retrieve a list of IDs in the plan rather than the entire plan data.

The SetPlan operation is provided so that User Defined Fields might be set on the plan header. It is very important not to set data on the plan items by using this interface because it might potentially overwrite other plan item data that might be set by other process components. Setting the plan header User Defined Field data must only be done as a last resort. Use the plan item User Defined Field data instead.

The SetPlanItem operation is provided to set User Defined Field data on an individual plan item. By convention a process component must set plan item User Defined Field data only on its plan item. There is no technical restriction on setting data on other plan items, but this introduces a situation where there might be contention for the data. This is particularly a problem with plans with parallel tracks of execution. The convention of setting User Defined Field data only on its plan item eliminates the possibility of this happening.

Where possible make use of the merge data option in SetPlanItem. Using merge means sending through only the User Defined Fields that have changed rather than the entire set of User Defined Fields. It is only possible to merge data if unique User Defined Field names are implemented and configured in the Transient Data Store global variable flags. If the merge data is not used, then the entire set of User Defined Fields is replaced by the set values. In this case, it is necessary to send through all User Defined Fields, whether or not they have been changed.

1. Shut down the Transient Data Store Cache and Agent engines and purge the backing store database by using SQL scripts. This must be done with Orchestrator at the same time. Restart Transient Data Store Cache and Agent. This results in an outage of the system while the cleanup occurs.
2. Make use of the CleanupAtEndOfOrder option in the Transient Data Store that automatically purges the order from the system when it reaches a COMPLETE or CANCELLED state. The order and plan details are still retained within Order Management Server, but no history is available in Transient Data Store.
3. Make use of the DeleteOrder/DeletePlan operations in the Transient Data Store. This requires the use of an external project to determine that orders to purge and then invoke these services with the list of orderIDs. This must be done during a slow period in order processing due to the performance impact.

# Development Guidelines

---

TIBCO provides development guidelines in the area of the team, code, and environmental organization. These recommendations include things to consider when you first create your environment and set up your servers and tips to avoid unnecessary delays in your environment. Additional information provides guidelines for managing your source control and creating a configuration management strategy.

## Team Organization Guidelines

Consider carefully the management of changes to the product catalog during development. Any TIBCO Order Management - Long Running project has a single product catalog, so changes to it have to be managed properly to avoid unnecessary delays and unintended impacts. Consider maintaining centralized control over changing the product model, rather than all developers on the team to make changes.

Consider also how changes to the catalog are performed. In a development environment, where new entries often have to be added to TIBCO Product and Service Catalog in bulk, it might be worth considering an approach where data related to the product model is prepared outside TIBCO Product and Service Catalog (for example, in a spreadsheet or csv files), and then imported into the catalog periodically before publishing to TIBCO Order Management - Long Running. Care needs to be taken so that sufficient validation of the external data entry is performed. If this approach is adopted, it is worth investing the time to thoroughly understand how TIBCO Master Data Management processes imported data.

## Code Organization Guidelines

Implement a configuration management strategy. This means making use of a dedicated build team who solely be responsible for promoting code between environments. This requires locking down access to environments after development so that only appropriate configuration management staff has access.

Use a source code control system, such as cvs and subversion.

No developer workstation might be master of any code delivered to production.

All deliveries might be based on scripted deployments built directly against a tagged set of artifacts in the source code control system on a machine that is used for that purpose.

Typically configuration files, SQL scripts, and generated wars all are source controlled.

Source-controlling dependent jars might be considered, so that version dependency failures are avoided.

Configuration files for development environments can also usefully be centrally maintained to reduce environment set up overheads.

## Environment Organization Guidelines

In many cases, there is a need for multiple instances of development or test environment. For instance, one for developing the next release, another for bug fixing production, or perhaps other intermediate releases that are being tested in parallel.

When creating environments, it is a good idea to define a system for assigning non-overlapping TCP/IP ports, and never use the default ports provided by the software installer. Multiple instances of TIBCO Order Management - Long Running and TIBCO Product and Service Catalog to resides on the same physical server. Of course, CPU and memory are limiting factors here. If multiple Oracle instances are not possible, then consider a per-environment naming scheme for the schemas used by TIBCO Product and Service Catalog and TIBCO Order Management - Long Running.

# Performance Tuning Guidelines

TIBCO provides performance tuning guidelines and best practices for TIBCO Product and Service Catalog, TIBCO Order Management - Long Running, and the infrastructure.

## Performance Tuning Guidelines for TIBCO Order Management - Long Running

In general, performance tuning is typically an activity that takes place in conjunction with the performance testing.

For Enterprise Message Service tuning, general guidelines can be found in the product documentation, "Tuning" in the *TIBCO Order Management - Long Running Administration guide*. Enterprise Message Service tuning essentially consists of tuning the number of listeners for the input queues for order requests, and processing status notifications from fulfillment components as orders are processed.

For the TIBCO BusinessEvents components, general TIBCO BusinessEvents tuning advice is also relevant. The same applies for the TIBCO BusinessWorks™ components.

The good general advice is that when running performance testing, watch Enterprise Message Service queues throughout the order test to observe a backup of messages. If queues are backing up, then consider what components are processing messages off those queues and increase the consumers.

For Order Management Server, consider the following configuration settings:

```
<ConfValue description="Minimum number of concurrent consumers for listener (default 1)" name="Minimum number of concurrent consumers for listener"
propname="com.tibco.af.oms.webservice.soap.jms.concurrentConsumers" sinceVersion="1.2"
visibility="Advanced">
  <ConfNum default="1" value="1" />
</ConfValue>
<ConfValue description="Maximum number of concurrent consumers for listener (default 1)" name="Maximum number of concurrent consumers for listener"
propname="com.tibco.af.oms.webservice.soap.jms.maxConcurrentConsumers"
sinceVersion="1.2" visibility="Advanced">
  <ConfNum default="5" value="5" />
</ConfValue>
```

## Performance Tuning Guidelines for Infrastructure

### Database

Database tuning is a highly specialized area, and ensure that Database Admin expertise be available when running performance tests and tuning the system.

Use Oracle Enterprise Manager to help tune the database <http://www.oracle.com/technetwork/oem/db-mgmt/db-mgmt-093445.html?ssSourceSiteId=ocomen>. Learning how to create Oracle ADDM reports is a good idea and a good option if OEM is not available.

Adding appropriate indexes (as recommended by these tools) can help improve performance.

### Enterprise Message Service

TIBCO Enterprise Message Service is used as a messaging bus to integrate TIBCO Order Management - Long Running with the external systems, as well for the integration between the TIBCO Order Management - Long Running components. Most of the tuning parameters, such as max connections and memory settings, of TIBCO Enterprise Message Service are available in the `tibemsd.conf` file present in the configuration directory of Enterprise Message Service. Refer to TIBCO Enterprise Message Service documentation for tuning details.

# Guidelines for Operations

---

For best practices, follow these guidelines for monitoring, logging, and database purging.

## Guidelines for Monitoring

TIBCO provides monitoring guidelines and best practices to use on the hardware and software in your business environment.

it is a good practice to monitor the following properties:

### Hardware

- CPU
- Memory
- Disk Space
- Selected critical processes are running

### Enterprise Message Service

- Queue depth and message count
- Consumer sessions
- Presence of listeners on queues and topics

### Administrator

- Process is running
- Memory usage
- Queue length of internal tasks
- Log file contents for warnings and errors

### Databases

- Check database availability through JDBC ping

### TIBCO Order Management - Long Running, TIBCO Fulfillment Orchestration Suite, and Transient Data Store

- Processes are running
- Memory usage
- Check database dependency
- Check Enterprise Message Service dependency
- Check internal errors
- Log file contents for warnings and errors

### TIBCO Product and Service Catalog and TIBCO Master Data Management

- Process is running
- Memory usage
- Check database dependency
- Check Enterprise Message Service dependency
- Check internal errors



- Log file contents for warnings and errors

You can also use the Hawk Accelerator offering from CTS to implement some of these releases more quickly.

## Guidelines for Logging

TIBCO Order Management - Long Running includes a logging framework with all errors and log events logged to a standard Enterprise Message Service destination. Use this in conjunction with logging and event handling system (such as Common Logging and Error handling framework of TIBCO) to ensure TIBCO Order Management - Long Running component logging and errors are handled in a consistent fashion.

Consider to use the same framework for handling logging from custom components (such as process components) as well so that all logging is managed in a central place.

## Database Purging or Cleanup Guidelines

### Order Management System

Order Management System is provided with a purge script that can be used to clean up data about orders that are in a certain state, in a given date range. This can be used to purge order data at regular intervals and can also be run while the system is processing new orders, thus avoiding the need for an outage.

However, purging completely deletes the data about orders and associated objects such as plans, plan items and User Defined Fields. If any of this information needs to be kept, then consider implementing an archiving strategy, where data is not purged, but copied to an archive. This requires some custom development.

Ensure that the business requirements around logging and archiving are properly captured and analyzed.

### Transient Data Store and Orchestrator

Both these components can be configured to self-clean when an order reaches an end-state. Use these features to avoid having to clean up the Transient Data Store and Orchestrator. Order Management System might be seen as the database of record for orders and plans, and the backing stores for Orchestrator and Transient Data Store might just be for dealing with data about orders that are in-flight.

# TIBCO Documentation and Support Services

---

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

## How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website, mainly in HTML and PDF formats.

The TIBCO Product Documentation website is updated frequently and is more current than any other documentation included with the product. To access the latest documentation, visit <https://docs.tibco.com>.

## Product-Specific Documentation

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

- *TIBCO® Order Management - Long Running Release Notes*
- *TIBCO® Order Management - Long Running Installation and Configuration Guide*
- *TIBCO® Order Management - Long Running User's Guide*
- *TIBCO® Order Management - Long Running Administration Guide*
- *TIBCO® Order Management - Long Running Getting Started Guide*
- *TIBCO® Order Management - Long Running Best Practices Guide*
- *TIBCO® Order Management - Long Running Concepts and Architecture Guide*
- *TIBCO® Order Management - Long Running Web Services Guide*

## How to Contact TIBCO Support

You can contact TIBCO Support in the following ways:

- For an overview of TIBCO Support, visit <http://www.tibco.com/services/support>.
- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support portal at <https://support.tibco.com>.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to <https://support.tibco.com>. If you do not have a user name, 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 <https://community.tibco.com>.

## Legal and Third-Party Notices

---

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

ANY SOFTWARE ITEM IDENTIFIED AS THIRD PARTY LIBRARY IS AVAILABLE UNDER SEPARATE SOFTWARE LICENSE TERMS AND IS NOT PART OF A TIBCO PRODUCT. AS SUCH, THESE SOFTWARE ITEMS ARE NOT COVERED BY THE TERMS OF YOUR AGREEMENT WITH TIBCO, INCLUDING ANY TERMS CONCERNING SUPPORT, MAINTENANCE, WARRANTIES, AND INDEMNITIES. DOWNLOAD AND USE OF THESE ITEMS IS SOLELY AT YOUR OWN DISCRETION AND SUBJECT TO THE LICENSE TERMS APPLICABLE TO THEM. BY PROCEEDING TO DOWNLOAD, INSTALL OR USE ANY OF THESE ITEMS, YOU ACKNOWLEDGE THE FOREGOING DISTINCTIONS BETWEEN THESE ITEMS AND TIBCO PRODUCTS.

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 TIBCO Software 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 TIBCO Software Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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.

This 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. Please see the readme.txt file for the availability of this software version 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. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

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 "READ ME" FILES.

This and other products of TIBCO Software Inc. may be covered by registered patents. Please refer to TIBCO's Virtual Patent Marking document (<https://www.tibco.com/patents>) for details.

Copyright © 2010-2022. TIBCO Software Inc. All Rights Reserved.