



## Introduction to Data Model Online Training

### Abstract:

This training will introduce you to the DELMIA Apriso data model, it will show you how data is structured in the database and how information can be found on the data model.

### Detailed objectives. After the training you will:

- ▶ Have a general overview of the DELMIA Apriso data model and will be able to access the documentation
- ▶ Understand the data model delivered in the standard product
- ▶ Know how to identify the specifics of the data model for a given implementation on project

### Target audience:

- ▶ New consultants working with DELMIA Apriso

### Requirements:

- ▶ Familiarity with Process Builder user interface
- ▶ Familiarity with MS SQL

### Role and level:

- ▶ DELMIA Apriso users that will be working with more advanced screens



Duration: 60 min





## ▼ Welcome to DAT001

Introduction to Data Model Online  
Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation  
Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ► Chapter 2: Data Model Rules

End of Course

## Chapter 1: Overview

In this chapter you will get the basic overview of what Data Model is in regards to DELMIA Apriso. Later the Documentation Model will be presented and more details about DM will follow.

1. Data Model Documentation
2. Data Model Rules





## ▼ Welcome to DAT001

Introduction to Data Model Online  
Training

## ▼ Chapter 1: Overview

## Overview

Data Model Documentation

Data Model Documentation  
Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ► Chapter 2: Data Model Rules

End of Course

## Overview

The DELMIA Apriso data model is an abstract scheme that organizes all data in DELMIA Apriso and standardizes how they relate to one another and to the elements they represent in the real world.

Usually, the data model is composed of 1200+ tables that serve all Apriso's modules, e.g.:

- Production
- Logistic
- Quality
- Etc.

But it can vary between the modules as well.

It supports the usage of both Microsoft SQL servers and Oracle servers, however MS SQL is more frequently used.



ORACLE®



## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

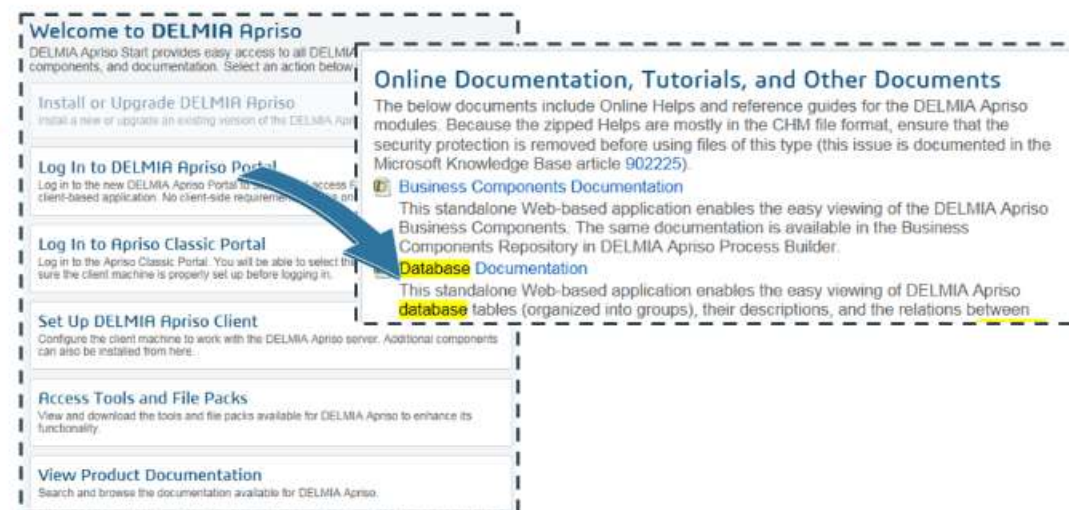
## ▶ Chapter 2: Data Model Rules

End of Course

## Data Model Documentation

The DELMIA Apriso Data Model is well-documented and can be accessed in multiple ways:

- ▶ **Process Builder:**
  - In Process Builder, the documentation is accessible through the Help -> Database Documentation
- ▶ **DELMIA Apriso Portal:**
  - Use <server\_name>/Apriso/Start address and then View Product Documentation -> Database Documentation (link to an online version)



Search...

- ▼ Welcome to DAT001
  - Introduction to Data Model Online Training
- ▼ Chapter 1: Overview
  - Overview
  - Data Model Documentation
  - Data Model Documentation Explained**
  - Table Category Explained, part 1
  - Table Columns Explained, part 2
  - Prime Data, part 1
  - Prime Data, part 2
- ▶ Chapter 2: Data Model Rules
- End of Course

# Data Model Documentation Explained

1

2

3

6

Browser

Tables by Category

Change Management

ECM\_ORDER

ECM\_ORDER\_BOM

ECM\_ORDER\_CHARACTERIST

ECM\_ORDER\_DOCUMENT

ECM\_ORDER\_OPERATION

ECM\_ORDER\_PROCESS

ECM\_ORDER\_PRODUCT

ECM\_ORDER\_WIP\_ORDER

ECM\_ORDER\_WORK\_INSTR\_F

Common

Containment

Framework

Inventory

Machine Integrator

Maintenance

Menus and Authorization

Order Management

Partners

Production

Quality

Resource Management

Time and Labor

Weighing and Dispensing

Tables Inventory

ACTION\_FLAG\_TYPE

ADDRESS

ADDRESS\_CONTACT

ADDRESS\_TRANSLATION

ADDRESS\_TYPE

AGENCY

ALERT

ALERT\_CLASS

Enter search string

Search

Search for: ☒ Methods ☐ Keywords

Browse

WIP\_ORDER

5

Locate in Browser

4

Description

Contain the list of the execution orders, the assignment of an order to a process/receipt/SOP, the assignment to as resource or an order\_header or detail. Numbering can be done by order type, few dates information (use Order Date table if more dates are required), Quantity targeted and completed as well as UOM. Order relation should be use to connect orders together rather than to use the ParentWipOrder field of this table. Contain the tolerance percent for quantity

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
WipOrderNo	NVARCHAR(40)	No		Link to the WIP Order	
WipOrderType	SMALLINT(5,0)	No		Link to the WIP Order type	WIP_ORDER_TYPE
PutAwayLocationID	INT(10,0)	Yes		The default location where inventory should be put away for this Wip Order. For Wip Orders of type Picking (or related types), this is the location where all picked inventory should be dropped for packing or production. For Wip Orders of type Receiving Sc	WAREHOUSE_LOCATION
ProductID	INT(10,0)	Yes		Reference to a product (product number and product version)	PRODUCT
OrderQuantity	DECIMAL	Yes	(0,0)	Order target quantity	
CompletedQuantity	DECIMAL	Yes	(0,0)	Quantity completed of the order. update by business component, not by navigation	
Priority	SMALLINT(5,0)	Yes	(100)	Priority of the order	
ExpectedStartDate	DATETIME	Yes	(getutcdate (0))	Future use	
ReleaseDate	DATETIME	Yes	(getutcdate (0))	Release date of the order	
ScheduledStartDate	DATETIME	Yes	(getutcdate (0))	Future use	
DueDate	DATETIME	Yes	(getutcdate (0))	Due date of the order	
ActualStartDate	DATETIME	Yes	(getutcdate (0))	Actual start date of the WIP order	

- 1 - Browse tables by category
- 2 - Table name
- 3 - Browse tables in alphabetical order
- 4 - Description Panel
- 5 - Switch to see currently displayed table in tree-structure browser (1)
- 6 - Search window



## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ► Chapter 2: Data Model Rules

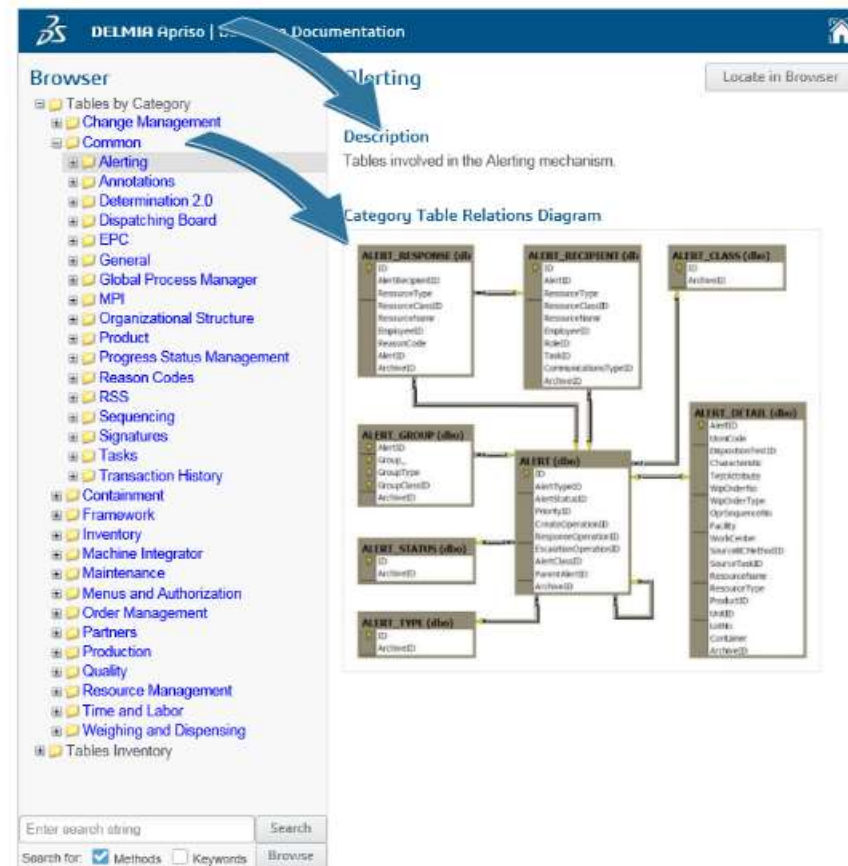
End of Course

## Table Category Explained, part 1

Under the category folder, you can access the Data Model organized by functional elements.

There are descriptions at the folder level. For instance, when selecting category Common/Alerting you will get:

- A short description of the category that displays information about the purpose of the tables included in this particular category
- A table relation diagram overview of the tables involved in this category



## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ► Chapter 2: Data Model Rules

End of Course

## Table Columns Explained, part 2

When selecting a single table you will find:

- A brief description of what the table stores
- Column list:
  - **Key symbol column** – the symbol means the column belongs to table's primary key
  - **Column Name** – the name of the column
  - **Data Type** – column data type (Char, Integer, etc.)
  - **Nullable** – indicates if the column can have a null value
  - **Default** – default column value
  - **Description** – column purpose and possible uses
  - **Links** – if there is a value that should be linked to that column, the table with that value is reflected here and linked (if it is in the documentation)

**DELMIAR Application Database Documentation**

**Browser**

- Tables by Category
  - Change Management
  - Common
  - Alerting
    - ALERT**
    - ALERT\_CLASS
    - ALERT\_DETAIL
    - ALERT\_GROUP
    - ALERT\_RECIPIENT
    - ALERT\_RESPONSE
    - ALERT\_STATUS
    - ALERT\_TYPE
  - Annotations
  - Determination 2.0
  - Dispatching Board
  - EPC
  - General
  - Global Process Manager
  - MPI
  - Organizational Structure
  - Product
  - Progress Status Management
  - Reason Codes
  - RSS
  - Sequencing
  - Signatures
  - Tasks
  - Transaction History
  - Containment
  - Framework
  - Inventory
  - Machine Integrator
  - Maintenance
  - Menus and Authorization
  - Order Management
  - Partners
  - Production

**ALERT**

Description: Stores the various Alerts generated by Alert components.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ID	INT(10,0)	No		Unique identifier of a record (key) in a table.	
AlertTypeID	SMALLINT(5,0)	No		ID that denotes the Alert's type. Linked to the ALERT_TYPE table.	<a href="#">ALERT_TYPE</a>
AlertStatusID	SMALLINT(5,0)	No		ID that denotes the Alert's status. Linked to the ALERT_STATUS table.	<a href="#">ALERT_STATUS</a>
GeneratedOn	DATETIME	No		Date and time that the Alert was created on.	
PriorityID	INT(10,0)	No		ID that denotes the Alert's priority. Linked to the PRIORITY table.	<a href="#">PRIORITY</a>
CurrentEscalationLevel	INT(10,0)	No		<For Future Use>	
MessageID	NVARCHAR(255)	Yes		<For Future Use>	
Message	NVARCHAR(2000)	Yes		Text description of the Alert.	
CreateOperationID	INT(10,0)	Yes		ID of the operation that created the Alert. Linked to the OPERATION table.	<a href="#">OPERATION</a>
ResponseOperationID	INT(10,0)	Yes		Standard Operation to be invoked as the response to the Alert. Linked to the OPERATION table.	<a href="#">OPERATION</a>
EscalationOperationID	INT(10,0)	Yes		<For Future Use>	<a href="#">OPERATION</a>
AlertClassID	INT(10,0)	Yes		ID of the Alert's class. It is used to route the Alert and define its escalation process.	<a href="#">ALERT_CLASS</a>
ExpirationDate	DATETIME	Yes		Expiration date of the Alert. Alerts with expired dates are not displayed on the My Alert List.	
Duration	INT(10,0)	Yes		Duration of time from the GeneratedOn date/time that the Alert is valid. Alerts with expired durations are not displayed on the My Alert List.	
ParentAlertID	INT(10,0)	Yes		<For Future Use>	<a href="#">ALERT</a>
RequiredResponses	INT(10,0)	Yes		Number of responses required by the Alert.	
ResponseCount	INT(10,0)	Yes		Current number of responses against the Alert.	

Enter search string: Search

Search for: ☒ Methods ☐ Keywords Browse

- ▼ Welcome to DAT001
  - Introduction to Data Model Online Training
- ▼ Chapter 1: Overview
  - Overview
  - Data Model Documentation
  - Data Model Documentation Explained
  - Table Category Explained, part 1
  - Table Columns Explained, part 2
  - Prime Data, part 1**
  - Prime Data, part 2
- Chapter 2: Data Model Rules
- End of Course

# Prime Data, part 1

Some of the tables contain Prime Data. Prime Data is a set of certain statuses and pieces of information that have their numeral representation.

This way, the same information can be included in tables that required different data types.

Example: there is a table that contains production orders, including information about order status. Instead of writing New as the first order status, there is a numeral 1.

DELIA Apriso | Database Documentation

Browser

Tables by Category

Change Management

Common

Alerting

Annotations

Determination 2.0

Dispatching Board

EPC

General

Global Process Manager

MPI

Organizational Structure

Product

Progress Status Management

PROGRESS\_STATUS

PROGRESS\_STATUS\_CLASS

PROGRESS\_TRANSITION\_R

Reason Codes

RSS

Sequencing

Signatures

Tasks

Transaction History

Containment

Framework

Inventory

Machine Information

PROGRESS\_STATUS

Locate in Browser

Description

This table contains various prime and custom progress statuses used in different areas: WIP Orders, Work Instructions and Change Management.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ProgressStatus	INT(10,0)	No		Progress status	
TextID	INT(10,0)	Yes		Link to Text table	
Name	NVARCHAR(50)	No		Progress Status unique name	
SequenceNo	INT(10,0)	Yes		Progress Status' order	

Prime Data

1 - New, 2 - Picked, 3 - Packed, 4 - Received, 5 - PGI, 6 - PGR, 7 - Planned, 8 - Planning completed, 9 - Check-in, 10 - Loading start, 11 - Loading end, 12 - Shipment completion, 13 - Shipment start, 14 - Shipment end, 15 - Released, 16 - Picking, 17 - Complete, 18 - Cancelled, 19 - Open, 20 - Closed, 21 - In Progress, 22 - Waiting for Approval, 23 - Waiting for Approval Level 1, 24 - Waiting for Approval Level 2, 25 - Approved, 26 - Approved Level 1, 27 - Approved Level 2, 28 - Rejected, 29 - Paused, 30 - Open, 31 - Corrected, 32 - Closed, 33 - New, 34 - Open, 35 - Hold, 36 - Completed, 37 - New, 38 - Open, 39 - Hold, 40 - Completed, 10000001 - New, 10000002 - Investigation, 10000003 - Implementation, 10000004 - Completed, 10000005 - Cancelled, 10000006 - New, 10000007 - Implementation, 10000008 - Implemented, 10000009 - Completed, 10000010 - New, 10000011 - Investigation, 10000012 - Completed, 10000013 - Cancelled





## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ► Chapter 2: Data Model Rules

End of Course

## Prime Data, part 2

### Reasons for Prime Data:

- Unification of data – there is a standard set of Prime Data for all tables within DELMIA Apriso, marked by the consecutive numbers (1, 2, and so on), as presented on the picture
- If a translation is needed, it is easier to use the numeric code and link it with the correct translation otherwise, each user could write their own type of data
- It is easier to use across the board



### DO NOT TAMPER WITH PRIME DATA!

- It is widely used by Business Components – changing it will result in BCs not working
- If there is a need to add additional entries – use further numbers (e.g. 20000001, as presented on the picture)
- Prime Data is fully controlled by DELMIA Apriso, and it may be removed, changed, or added by the Database Upgrader

DELMIA Apriso | Database Documentation

Browser

- Tables by Category
  - Change Management
  - Common
    - Alerting
    - Annotations
    - Determination 2.0
    - Dispatching Board
    - EPC
    - General
    - Global Process Manager
    - MPI
    - Organizational Structure
    - Product
    - Progress Status Management
      - PROGRESS\_STATUS**
      - PROGRESS\_STATUS\_CLASS
      - PROGRESS\_TRANSITION\_RL
    - Reason Codes
    - RSS
    - Sequencing
    - Signatures
    - Tasks
    - Transaction History
    - Containment
    - Framework
    - Inventory
    - Machine Information

PROGRESS\_STATUS

Locate in Browser

Description

This table contains various prime and custom progress statuses used in different areas: WIP Orders, Work Instructions and Change Management.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ProgressStatus	INT(10,0)	No		Progress status	
TextID	INT(10,0)	Yes		Link to Text table	
Name	NVARCHAR(50)	No		Progress Status unique name	
SequenceNo	INT(10,0)	Yes		Progress Status' order	

Prime Data

1 - New, 2 - Picked, 3 - Packed, 4 - Received, 5 - PGI, 6 - PGR, 7 - Planned, 8 - Planning completed, 9 - Check-in, 10 - Loading start, 11 - Loading end, 12 - Shipment completion, 13 - Shipment start, 14 - Shipment end, 15 - Released, 16 - Picking, 17 - Complete, 18 - Cancelled, 19 - Open, 20 - Closed, 21 - In Progress, 22 - Waiting for Approval, 23 - Waiting for Approval Level 1, 24 - Waiting for Approval Level 2, 25 - Approved, 26 - Approved Level 1, 27 - Approved Level 2, 28 - Rejected, 29 - Paused, 30 - Open, 31 - Corrected, 32 - Closed, 33 - New, 34 - Open, 35 - Hold, 36 - Completed, 37 - New, 38 - Open, 39 - Hold, 40 - Completed, 10000001 - New, 10000002 - Investigation, 10000003 - Implementation, 10000004 - Completed, 10000005 - Cancelled, 10000006 - New, 10000007 - Implementation, 10000008 - Implemented, 10000009 - Completed, 10000010 - New, 10000011 - Investigation, 10000012 - Completed, 10000013 - Cancelled



## ▼ Welcome to DAT001

Introduction to Data Model Online  
Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation  
Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

## Chapter 2: Data Model Rules

In this chapter you will get to know the most common rules and best practices for using the Data Model and the databases.

1. *Data Model Documentation*
2. **Data Model Rules**



## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

## Primary Key, part 1

There are three types of the primary keys in the DELMIA Apriso Data Model

- ▶ An automatically incremented ID of the row

DELMIA Apriso | Database Documentation

INVENTORY2

Description  
This table contains inventory.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ID	INT(10,0)	No		Unique ID of the row	
ProductID	INT(10,0)	No		ID of the Product.	LOT_NO PRODUCT SERIAL_NO
WarehouseLocationID	INT(10,0)	No		ID of the Warehouse Location.	WAREHOUSE_LOCATION
Container	NVARCHAR(40)	Yes		Container number.	CONTAINER
SerialNo	NVARCHAR(40)	Yes		Serial number.	SERIAL_NO
LotNo	NVARCHAR(40)	Yes		Lot number.	LOT_NO
InventoryClassID	INT(10,0)	Yes		ID of the Inventory Class.	INVENTORY_CLASS
InventoryStatus	SMALLINT(5,0)	Yes		Link to the INVENTORY_STATUS table.	INVENTORY_STATUS
PartnerID	INT(10,0)	Yes		Link to the PARTNER table.	PARTNER
GradeID	INT(10,0)	Yes		Link to the GRADE table.	GRADE
ERPMaterialStockID	INT(10,0)	Yes		Link to the ERP_MATERIAL_STOCK table.	ERP_MATERIAL_STOCK
InventoryAllocationID	INT(10,0)	Yes		Link to the INVENTORY2_ALLOCATION table.	INVENTORY2_ALLOCATION
QuantityOnHand	DECIMAL	Yes		Quantity available.	
QuantityAllocated	DECIMAL	Yes		Quantity allocated.	
MinFIFOdate	DATETIME	Yes		Minimum FIFO date.	
MaxFIFOdate	DATETIME	Yes		Maximum FIFO date.	
UnitID	INT(10,0)	Yes		Link to the UNIT table.	UNIT

## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

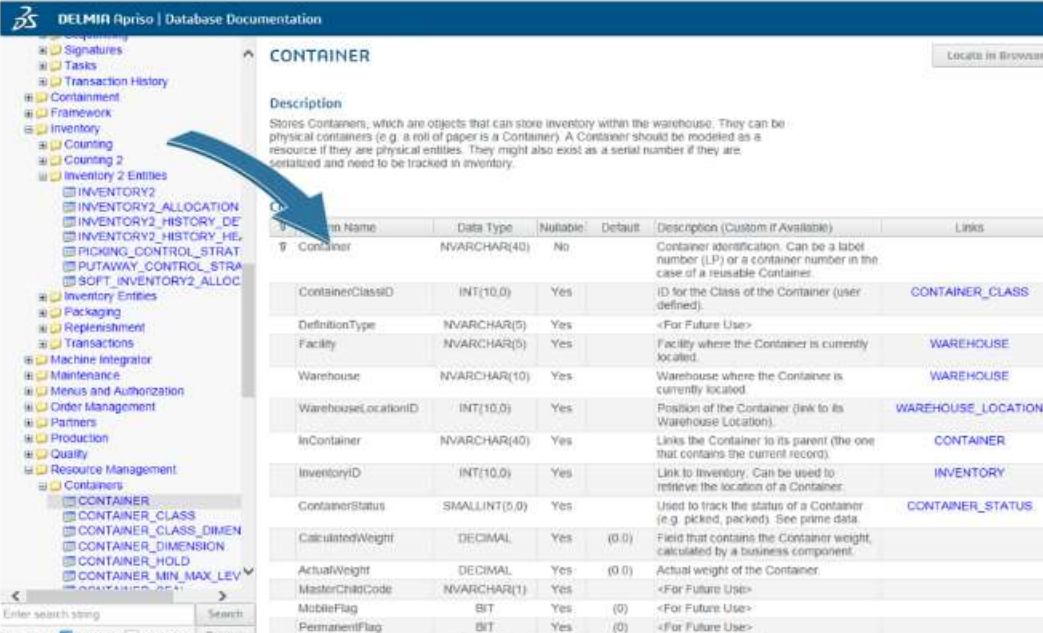
UnitID

Multi Language

## Primary Key, part 2

There are three types of the primary keys in the DELMIA Apriso Data Model

- ▶ An automatically incremented ID of the row
- ▶ A key based on the column which keeps the main information (e.g. a Container field in the CONTAINER table)



**DELMIA Apriso | Database Documentation**

**CONTAINER** [Locate in Browser](#)

**Description**  
Stores Containers, which are objects that can store inventory within the warehouse. They can be physical containers (e.g. a roll of paper is a Container). A Container should be modeled as a resource if they are physical entities. They might also exist as a serial number if they are serialized and need to be tracked in inventory.

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
Container	NVARCHAR(40)	No		Container identification. Can be a label number (LP) or a container number in the case of a reusable Container.	
ContainerClassID	INT(10,0)	Yes		ID for the Class of the Container (user defined).	<a href="#">CONTAINER_CLASS</a>
DefinitionType	NVARCHAR(5)	Yes		<For Future Use>	
Facility	NVARCHAR(5)	Yes		Facility where the Container is currently located.	<a href="#">WAREHOUSE</a>
Warehouse	NVARCHAR(10)	Yes		Warehouse where the Container is currently located.	<a href="#">WAREHOUSE</a>
WarehouseLocationID	INT(10,0)	Yes		Position of the Container (link to its Warehouse Location).	<a href="#">WAREHOUSE_LOCATION</a>
InContainer	NVARCHAR(40)	Yes		Links the Container to its parent (the one that contains the current record).	<a href="#">CONTAINER</a>
InventoryID	INT(10,0)	Yes		Link to Inventory. Can be used to retrieve the location of a Container.	<a href="#">INVENTORY</a>
ContainerStatus	SMALLINT(5,0)	Yes		Used to track the status of a Container (e.g. picked, packed). See prime data.	<a href="#">CONTAINER_STATUS</a>
CalculatedWeight	DECIMAL	Yes	(0.0)	Field that contains the Container weight, calculated by a business component.	
ActualWeight	DECIMAL	Yes	(0.0)	Actual weight of the Container.	
MasterChildCode	NVARCHAR(1)	Yes		<For Future Use>	
MobileFlag	BIT	Yes	(0)	<For Future Use>	
PermanentFlag	BIT	Yes	(0)	<For Future Use>	



## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

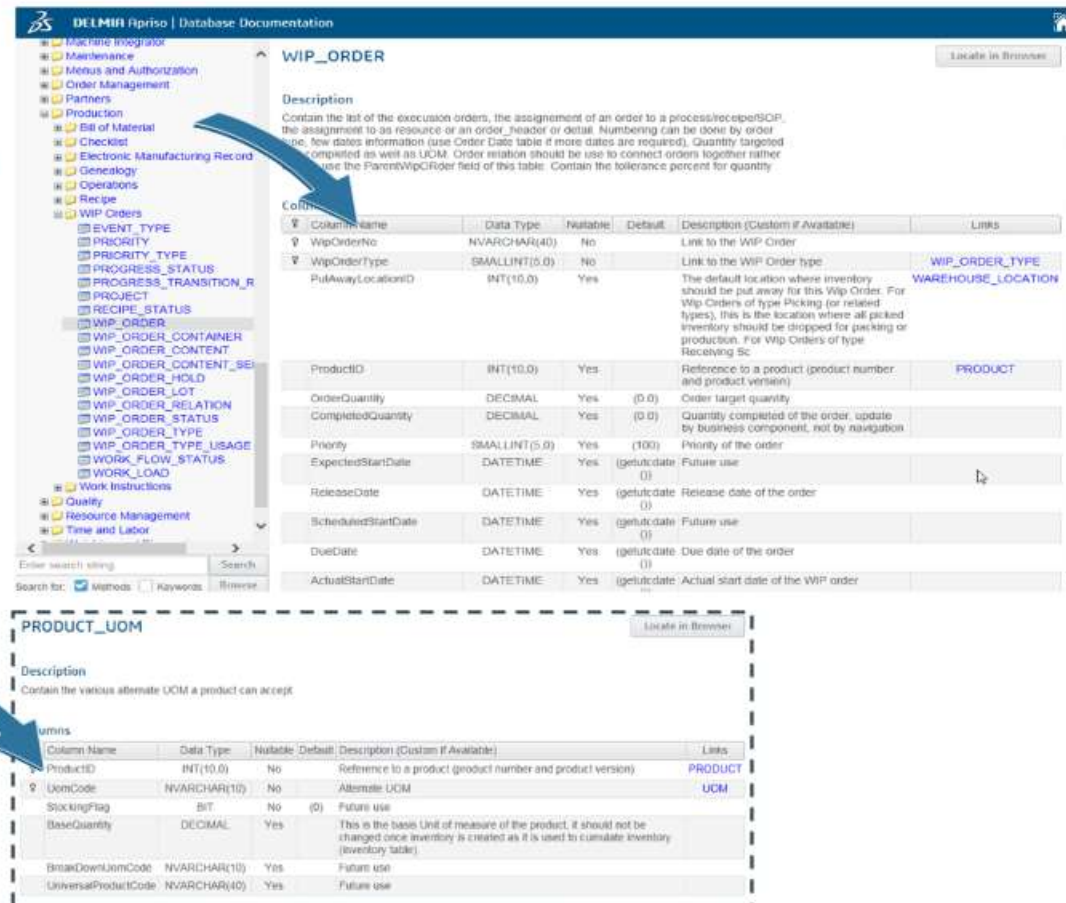
UnitID

Multi Language

## Primary Key, part 3

There are three types of the primary keys in the DELMIA Apriso Data Model

- ▶ An automatically incremented ID of the row
- ▶ A key based on the column which keeps the main information (e.g. a Container field in the CONTAINER table)
- ▶ A key based on a few columns as a main information (e.g. WipOrderNo, WipOrderType from the WIP\_ORDER), or as a property for the primary key from another table (e.g. ProductID, UomCode from the PRODUCT\_UOM)



**WIP\_ORDER**

Description: Contain the list of the execution orders, the assignment of an order to a process/receptor/SOP, the assignment to as resource or an order\_header or detail. Numbering can be done by order type, new dates information (use Order Date table if more dates are required). Quantity targeted to be completed as well as UOM. Order relation should be use to connect orders together rather than use the ParentWipOrder field of this table. Contain the tolerance percent for quantity.

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
WipOrderNo	NVARCHAR(40)	No		Link to the WIP Order	
WipOrderType	SMALLINT(5,0)	No		Link to the WIP Order type	WIP_ORDER_TYPE
PutAwayLocationID	INT(10,0)	Yes		The default location where inventory should be put away for this WIP Order. For Wip Orders of type Picking (or related types), this is the location where all picked inventory should be dropped for picking or production. For Wip Orders of type Receiving Sc	WAREHOUSE_LOCATION
ProductID	INT(10,0)	Yes		Reference to a product (product number and product version)	PRODUCT
OrderQuantity	DECIMAL	Yes	(0,0)	Order target quantity	
CompletedQuantity	DECIMAL	Yes	(0,0)	Quantity completed of the order, update by business component, not by navigation	
Priority	SMALLINT(5,0)	Yes	(100)	Priority of the order	
ExpectedStartDate	DATETIME	Yes	(getutcdate)	Future use	
ReleaseDate	DATETIME	Yes	(getutcdate)	Release date of the order	
ScheduledStartDate	DATETIME	Yes	(getutcdate)	Future use	
DueDate	DATETIME	Yes	(getutcdate)	Due date of the order	
ActualStartDate	DATETIME	Yes	(getutcdate)	Actual start date of the WIP order	

**PRODUCT\_UOM**

Description: Contain the various alternate UOM a product can accept

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ProductID	INT(10,0)	No		Reference to a product (product number and product version)	PRODUCT
UomCode	NVARCHAR(10)	No		Alternate UOM	UOM
StockingFlag	BIT	No	(0)	Future use	
BaseQuantity	DECIMAL	Yes		This is the basis Unit of measure of the product, it should not be changed once inventory is created as it is used to cumulate inventory (inventory table)	
BreakDownUomCode	NVARCHAR(10)	Yes		Future use	
UniversalProductCode	NVARCHAR(40)	Yes		Future use	

## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

## Foreign Key, part 4

A foreign key in one table points to a primary key in another table. The foreign key constraint is used to prevent actions that would destroy links between tables.

The foreign key constraint also prevents invalid data from being inserted into the foreign key column, because it has to be one of the values contained in the table it points to.

Foreign key is a constraint on table level. It is commonly mentioned as a link to another table but is not.

In this illustration, the system generated ID column for Products becomes the foreign key that products are known by in the INVENTORY2 table

The diagram illustrates a foreign key relationship between two tables: PRODUCT and INVENTORY2. A blue arrow points from the ID column in the INVENTORY2 table to the ID column in the PRODUCT table, indicating that the ID in INVENTORY2 is a foreign key that references the primary key ID in PRODUCT.

**PRODUCT Table:**

ID	ProductNo	FUID	Facility	TextID	UnitID	Re
1	100000001	G-CONT-007	7eb0d011-b500-49a-9c99-d06dd411db67	NULL	100522146	NULL
2	100000002	G-ALCO-003	52b51a70-5e3d-40b1-b071a57b5103	NULL	100522147	NULL
3	100000003	X-ERPH-003	71f03e52f1c2-43f0bba0-b86dc47a06	NULL	100522148	NULL
4	100000004	E-PRNT-002	68f567c-967f-4955e311-b0711b0c39c	NULL	100522149	NULL
5	100000005	X-TRTR-001	96dc2b32-05d4-4eeb-b4048-b8f717d	NULL	100522150	NULL
6	100000006	X-WHEL-002	963bdfa5-a2a4-4cxb-b4d8-9e	5592	100522151	1000000610
7	100000007	P-USV-2801	0137bdce-295d-40ee-09dd-de	400	100522152	100000111
8	100000008	A-BLOCK-001	1847e8c1-a90a-4648-aab2-d3e	1	100522153	100000112
9	100000009	A-BLOCK-002	0b602dee-b2d8-4f14-ee9d-451	1	100522154	100000113
10	100000010	A-NOZL-001	4f5a278c-4331-4e1e-9666-6d8	1	100522155	NULL
11	100000011	A-REFL-001				

**INVENTORY2 Table:**

ID	ProductID	WarehouseLocationID	Container	SerialNo	LotNo	InventoryClassID	Inver
1	100000001	100000004	100000057	NULL	PRN2-002	NULL	1
2	100000002	100000004	100000236	NULL	PRN2-001	NULL	1
3	100000360	100000009	100000109	C2214	S3224	NULL	1
4	100000374	100000015	100000058	R5860	NULL	NULL	1
5	100000375	100000015	100000058	C0909	NULL	NULL	1
6	100000409	100000017	100000058	CMT001	NULL	S11078	1
7	100000003	100000019	100000109	NULL	TST1-002	NULL	1
8	100000004	100000019	100000294	NULL	TST1-001	NULL	1
9	100000005	100000020	100000075	NULL	ASY3-002	NULL	1
10	100000006	100000020	100000205	NULL	ASY3-001	NULL	1
11	100000007	100000021	100000224	NULL	FMG1-007	NULL	1

Search...

▼ Welcome to DAT001

Introduction to Data Model Online Training

▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table joining, part 1

Table joining, part 2

Table joining, part 3

Table joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

# Table Joining, part 1

Foreign key data is located in many tables and very frequently there is a need to cross-reference the information from different tables (join tables). You can always use the Data Model Documentation to check which tables can be joined.

```
SELECT *
FROM
    INVENTORY2
    INNER JOIN PRODUCT
        ON INVENTORY2.ProductID = PRODUCT.ID
    INNER JOIN CONTAINER
        ON INVENTORY2.Container = CONTAINER.Container
```

DELMIA Apriso | Database Documentation

GROUP\_TYPE

INCOTERM

INSPECTION\_CHARACTERISTIC

INSPECTION\_CHARACTERISTIC

INSPECTION\_DET\_CHAR\_ATTR

INSPECTION\_DET\_CHARACTERI

INSPECTION\_DET\_CLASS

INSPECTION\_DET\_INSPECTION

INSPECTION\_DET\_SPECIFICATI

INSPECTION\_DETERMINATION

INSPECTION\_LINE

INSPECTION\_PLAN

INSPECTION\_PLAN\_CLASS

INSPECTION\_PLAN\_GROUP

INSPECTION\_PLAN\_RESOURCE

INSPECTION\_PLAN\_SCHEDULE

INSPECTION\_PLAN\_SCHEDULE

INSPECTION\_SEVERITY

INVENTORY

INVENTORY\_CLASS

INVENTORY2

Description

This table contains inventory.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ID	INT(10,0)	No		Unique ID of the row.	
ProductID	INT(10,0)	No		ID of the Product.	<a href="#">LOT_NO</a> <a href="#">PRODUCT</a> <a href="#">SERIAL_NO</a>
WarehouseLocationID	INT(10,0)	No		ID of the Warehouse Location.	<a href="#">WAREHOUSE_LOCATION</a>
Container	NVARCHAR(40)	Yes		Container number.	<a href="#">CONTAINER</a>
SerialNo	NVARCHAR(40)	Yes		Serial number.	<a href="#">SERIAL_NO</a>
LotNo	NVARCHAR(40)	Yes		Lot number.	<a href="#">LOT_NO</a>



## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

## Table Joining, part 2

Please take into account that in the instance in which the primary key in the other table that consists of few columns, you should use all of them in the query.

```
SELECT *
FROM
    INVENTORY2_ALLOCATION
    INNER JOIN WIP_OPERATION
        ON INVENTORY2_ALLOCATION.WipOrderNo = WIP_OPERATION.WipOrderNo
    AND INVENTORY2_ALLOCATION.WipOrderType = WIP_OPERATION.WipOrderType
    AND INVENTORY2_ALLOCATION.OprSequenceNo = WIP_OPERATION.OprSequenceNo
```

**DELMI Rpriso | Database Documentation**

**INVENTORY2\_ALLOCATION**

**Description**  
This table contains the allocation of inventory.

**Columns**

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ID	INT(10,0)	No		Unique ID of the row.	
WipOrderNo	NVARCHAR(40)	Yes		WIP Order Number.	WIP_OPERATION WIP_OPERATION_STEP WIP_ORDER
WipOrderType	SMALLINT(5,0)	Yes		WIP Order Type.	WIP_OPERATION WIP_OPERATION_STEP WIP_ORDER
OprSequenceNo	NVARCHAR(20)	Yes		Operation Sequence Number.	WIP_OPERATION WIP_OPERATION_STEP WIP_ORDER
StepSequenceNo	INT(10,0)	Yes		Step Sequence Number.	WIP_OPERATION_STEP
OrderNo	NVARCHAR(20)	Yes		Order Number.	ORDER_DETAIL ORDER_HEADER
OrderType	SMALLINT(5,0)	Yes		Order Type.	ORDER_DETAIL

**WIP\_OPERATION**

**Description**  
Contain the Wip Operation linked to the various WipOrders. order can be the image a the process used a reference (exp. the ERP (explosion 2)).

**Columns**

Column Name	Data Type	Nullable
WipOrderNo	NVARCHAR(40)	No
WipOrderType	SMALLINT(5,0)	No
OprSequenceNo	NVARCHAR(20)	No
PreviousSequenceNo	NVARCHAR(20)	Yes
NextSequenceNo	NVARCHAR(20)	Yes



## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table joining, part 1

Table joining, part 2

Table joining, part 3

Table joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

## Table Joining, part 3

There are exceptions though:

- ▶ In the RESOURCE\_ table the primary key consists of only one column - ID
- ▶ If you join the EQUIPMENT table with it the ResourceID column is used to join with the 'ID' of the RESOURCE\_ table

```
SELECT *
FROM
    EQUIPMENT
    INNER JOIN RESOURCE_
        ON EQUIPMENT.ResourceID = RESOURCE_.ID
```

EQUIPMENT					
Description					
Stores Equipment within the company. Equipments are types of Resources that need to be maintained in both the "RESOURCE_" table and the "EQUIPMENT" table.					
Columns					
Column Name	Data Type	Nullable	Default	Description (Comments)	
ID	INT(10,0)	No		Unique ID of the EQUIPMENT used to uniquely identify Equipments between servers.	
OwnershipID	INT(10,0)	Yes		<For Future Use>	OWNERSHIP
MaximumLogins	INT(10,0)	Yes	(1)	Maximum number of logins for the Equipment.	
ResourceID	INT(10,0)	No		Resource ID	RESOURCE_
FUID	NVARCHAR(36)	No		Unique ID of the EQUIPMENT used to uniquely identify Equipments between servers.	
EquipmentClassID	INT(10,0)	Yes		The assignment of an Equipment to a Class.	EQUIPMENT_CLASS

RESOURCE_				
Description				
A resource is a generic entity that can be linked to either a container. It has attributes link capacity, mobility information, address...				
Columns				
Column Name	Data Type	Nullable	Default	
ID	INT(10,0)	No		
FUID	NVARCHAR(36)	No		
Name	NVARCHAR(80)	No		
ResourceName	NVARCHAR(80)	No		
ResourceType	SMALLINT(5,0)	No		

## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table joining, part 1

Table joining, part 2

Table joining, part 3

Table joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

## Table Joining

- In some of the other tables i.e. RESOURCE\_LABOR the reference to the RESOURCE\_ table is done by the two columns: ResourceName and ResourceType

```
SELECT *
FROM
    RESOURCE_LABOR RL
    JOIN RESOURCE_ R
        ON RL.ResourceName = R.ResourceName
        AND RL.ResourceType = R.ResourceType
WHERE
    R.WorkCenter = 'ABC'
```

RESOURCE_LABOR					
Description					
The "RESOURCE_LABOR" table stores information about resource labor such as start and end time, started and end employee, resource name and type, status etc. Resource labor can be started for such entities as wip operation, product/lot, span, state and defect. Each type of the entity has its own set of attributes that is stored in this table. The content of this table is shown through Cockpit.					
Columns					
Column Name	Data Type	Nullable	Default	Links	
ID	INT(10,0)	No			
Division	INT(10,0)	Yes	(1)	Future use	
ResourceName	NVARCHAR(80)	No		The name of the resource the resource labor record refers to	RESOURCE_
ResourceType	SMALLINT(5,0)	No		Resource type + resource define uniquely a resource	RESOURCE_

RESOURCE_				
Description				
A resource is a generic entity that can be linked to either a container or a container. It has attributes link capacity, mobility information, address, etc.				
Columns				
Column Name	Data Type	Nullable	Default	
ID	INT(10,0)	No		
FUID	NVARCHAR(36)	No		
Name	NVARCHAR(80)	No		
ResourceName	NVARCHAR(80)	No		
ResourceType	SMALLINT(5,0)	No		

## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table joining, part 1

Table joining, part 2

Table joining, part 3

Table joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

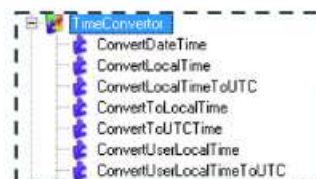
## Data Model Rules

The columns of the datetime type are stored in database in UTC format (unless specified otherwise). When handling the dates, conversions should always be done.

Main rules of conversion apply:

- ▶ Local → UTC: when the user inserts a value in the database
- ▶ UTC → Local: when the user reads a value from database. That should be done in a query by an individual creating a DELMIA Apriso Operation

There are a few BCs that can convert datetime back and forth based on employee, resource and database configuration. Furthermore there are standard SQL functions that also can do that as well.



Columns					
Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
WipOrderNo	NVARCHAR(40)	No		Link to the WIP Order	
WipOrderType	SMALLINT(5,0)	No		Link to the WIP Order type	WIP_ORDER_
PutAwayLocationID	INT(10,0)	Yes		The default location where inventory should be put away for this Wip Order. For Wip Orders of type Picking (or related types), this is the location where all picked inventory should be dropped for packing or production. For Wip Orders of type Receiving Sc	WAREHOUSE_LC
ProductID	INT(10,0)	Yes		Reference to a product (product number and product version)	PRODUCT
OrderQuantity	DECIMAL	Yes	(0.0)	Order target quantity	
CompletedQuantity	DECIMAL	Yes	(0.0)	Quantity completed of the order. update by business component, not by navigation	
Priority	SMALLINT(5,0)	Yes	(100)	Priority of the order	
ExpectedStartDate	DATETIME	Yes	(getutcdate ())	Future use	
ReleaseDate	DATETIME	Yes	(getutcdate ())	Release date of the order	
ScheduledStartDate	DATETIME	Yes	(getutcdate ())	Future use	
DueDate	DATETIME	Yes	(getutcdate ())	Due date of the order	
ActualStartDate	DATETIME	Yes	(getutcdate ())	Actual start date of the WIP order	
ActualCompletionDate	DATETIME	Yes	(getutcdate ())	This is the actual completion date of the WipOperation	
ScheduledDurationSeconds	INT(10,0)	Yes	(0)	Future use	
UomCode	NVARCHAR(10)	Yes		UOM of the quantity ordered	UOM
WorkOrderStatus	SMALLINT(5,0)	Yes		The status of the wip order	WIP_ORDER_S

## ▼ Welcome to DAT001

Introduction to Data Model Online Training

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

## Technical Columns

In the DELMIA Apriso Data Model you will find identical columns at the end of each table. They are called **technical columns** or **WHO columns**.

► They are:

- ReferenceID → Column used to reference another table .
  - LastUpdateOn
  - LastUpdatedBy
  - CreatedOn
  - CreatedBy
  - Active → Active column allow to mark the record as 'deleted' without physically removing it.
  - LastDeleteOn
  - LastDeletedBy
  - LastReactivateOn
  - LastReactivatedBy
  - ArchiveID
  - LastArchiveOn
  - LastArchivedBy
  - LastRestoreOn
  - LastRestoredBy
  - RowVersionStamp → The current version of the record, used for concurrency validation.
- Columns showing who and when a record was created and by whom.
- Columns planned to be used by the archiving tool - not currently used.





## ▼ Welcome to DAT001

[Introduction to Data Model Online Training](#)

## ▼ Chapter 1: Overview

[Overview](#)[Data Model Documentation](#)[Data Model Documentation Explained](#)[Table Category Explained, part 1](#)[Table Columns Explained, part 2](#)[Prime Data, part 1](#)[Prime Data, part 2](#)

## ▼ Chapter 2: Data Model Rules

[Primary Key, part 1](#)[Primary Key, part 2](#)[Primary Key, part 3](#)[Foreign Key, part 4](#)[Table Joining, part 1](#)[Table Joining, part 2](#)[Table Joining, part 3](#)[Table Joining](#)[Data Model Rules](#)[Technical Columns](#)[‘Active’ Column](#)[UnitID](#)[Multi Language](#)

## ‘Active’ Column

**Active columns** let you exclude data from queries you don’t want to see without deleting the record entirely.

In some tables, such as ‘CONTAINER’ or ‘PRODUCT’, the standard course of action is to mark the record as ‘deleted’ without physically removing it.

The good practice is to check them when joining tables and in where clause.

```
SELECT *  
FROM dbo.EMPLOYEE e  
JOIN dbo.EMPLOYEE_ROLE er  
    ON er.EmployeeID = e.ID  
    AND er.Active = 1  
WHERE  
    e.Active = 1
```



▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

Types, Classes, and Groups, part 1

Types, Classes, and Groups, part 2

Types, Classes, and Groups, part 3

# UnitID

When the column is named UnitID, it is automatically treated as linking to the UNIT table.

The UNIT table holds all unitID information across whole database.

It is not possible to have two different entities that have same unitID.

In the UNIT\_CHARACTERISTIC table, an entity is identified only by unitID, not type of entity.

The unitID is used to link to a record in a table like FACILITY, PRODUCT, etc. with a Characteristic (in the UNIT\_CHARACTERISTIC table), Annotation (in the UNIT\_ANNOTATION), etc.

UNIT

Locate in Browser

Description

This table stores Unit Ids that can be assigned to such entities as Facility, Product, Group, Product Group, Order Shipment Stage, Resource Life Cycle, Lot, Warehouse Location, Resource, Container, Work Center, Order, Wip Operation, etc. Once a unit is created, it can be assigned multiple characteristics.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ID	INT(10,0)	No		Unique ID of a record (key) in a table.	

▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

Types, Classes, and Groups, part 1

Types, Classes, and Groups, part 2

Types, Classes, and Groups, part 3

# Multi Language

When the column is named TextID, it is automatically treated as a reference to the TEXT table.

Such columns don't have any foreign keys constraints.

In the TEXT\_TRANSLATION table there are translations of the text for different LanguageIDs.

The LanguageID = 1033 (English USA) is the default language, if there is no translation to a specific language.

TEXT hold texts (similarly like Unit), TEXT\_TRANSLATION is a translation.

TEXT\_TRANSLATION

Locate in Browser

Description

Contains the various text translation of text table. This table can persist multiple texts (short, medium, extended) as well as icons for multiple devices types and URLs

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
TextID	INT(10,0)	No		Link to Text table	TEXT
LanguageID	INT(10,0)	No		Language of the entity	LANGUAGE
Micro	NVARCHAR(1)	Yes		The micro description (1 character)	
Short	NVARCHAR(80)	Yes		Short description (up to 80 characters)	
Medium	NVARCHAR(255)	Yes		Medium description (up to 255 characters)	
Extended	NVARCHAR(2000)	Yes		Extended description (up to 2000 characters)	
Text	NVARCHAR(-1)	Yes		Text information	
MicroAudioURL	VARCHAR	Yes		The universal resource locator (URL) for the audio file representing the micro description	
ShortAudioURL	VARCHAR				
MediumAudioURL	VARCHAR				
LongAudioURL	VARCHAR				
TextAudioURL	VARCHAR				
MicroIconURL	NVARCHAR				
MediumIconURL	NVARCHAR				
LargeIconURL	VARCHAR				
OrgChecksum	INT(10,0)				
InstChecksum	INT(10,0)				

Description

Contains the various text persisted in the system. The text itself is not directly stored in this table. It can be accessed in the Text\_translation table or in some cases, in the Text detail and then Text\_Detail\_translation table

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ID	INT(10,0)	No		Unique identifier of a record (key) in a table	
LiteralID	INT(10,0)	Yes		Keeps unique number of text literal	
Type	NVARCHAR(40)	Yes		Code describing the type of textual description	
FUID	NVARCHAR(36)	Yes		Unique identifier of the entity across multiple FlexNet instances	
LiteralDictionaryID	INT(10,0)	Yes		Reference to the LITERAL_DICTIONARY entry.	LITERAL_DICTIONARY

▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

Types, Classes, and Groups, part 1

Types, Classes, and Groups, part 2

Types, Classes, and Groups, part 3

# Types, Classes, and Groups, part 1

Types, Classes, and Groups are commonly used to classify entities:

- ▶ Type is always an integer data type field
  - The name of the field is usually created by adding 'Type' to the entity name (i.e. ResourceType, WipOrderType, etc.)
  - That field will be a foreign key to a table which keeps the types dictionary. The table has '\_TYPE' added to the main table name (i.e. WIP\_ORDER\_TYPE)
  - The primary key of the table sometimes consists of the entity name, and the type (i.e. WipOrderNo, WipOrderType)

WIP\_ORDER\_TYPE

Locate in Browser

Description

This table contains the various order types. This list can be extended.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
WipOrderType	SMALLINT(5,0)	No		Link to the WIP Order type	
TextID	INT(10,0)	Yes		Link to Text table	
WipOrderTypeUsageID	INT(10,0)	Yes		ID of WIP Order Type Usage	WIP_ORDER_TYPE_USAGE

Prime Data

1 - Production Order, 2 - Maintenance Order, 3 - Inspection Order, 4 - Requisition, 5 - Replenishment Order, 6 - Material Order, 7 - Delivery Order, 8 - Transportation Order, 9 - Campaign, 10 - Return Material Authorization, 11 - Service & Repair Authorization, 12 - Disposal Order, 13 - Customer Order, 14 - Project, 15 - Inventory Count, 16 - Purchase Order, 17 - ASN Inbound Delivery Order, 18 - Customer Return Order, 19 - Packaging Order, 20 - Picking Order, 21 - Putaway Order, 22 - Driven Picking Order, 23 - Master Schedule Order, 24 - Production Picking Order, 25 - Sequenced Picking Order, 26 - Preventive Maintenance Order, 27 - Reactive Maintenance Order, 28 - Tooling Order



- ▼ Chapter 1: Overview
  - Overview
  - Data Model Documentation
  - Data Model Documentation Explained
  - Table Category Explained, part 1
  - Table Columns Explained, part 2
  - Prime Data, part 1
  - Prime Data, part 2
- ▼ Chapter 2: Data Model Rules
  - Primary Key, part 1
  - Primary Key, part 2
  - Primary Key, part 3
  - Foreign Key, part 4
  - Table joining, part 1
  - Table joining, part 2
  - Table joining, part 3
  - Table joining
  - Data Model Rules
  - Technical Columns
  - 'Active' Column
  - UnitID
  - Multi Language
  - Types, Classes, and Groups, part 1
  - Types, Classes, and Groups, part 2**
  - Types, Classes, and Groups, part 3

# Types, Classes, and Groups, part 2

- ▶ An entity class is always referred by a foreign key to the particular class table (i.e. in the CONTAINER table there is a ContainerClassID field which is a foreign key to the CONTAINER\_CLASS table)
- ▶ The class table can have a prime data (standard DELMIA Apriso classes), but you can extend it, by adding more classes

CONTAINER

Location in Browser

Description

Stores Containers, which are objects that can store inventory within the warehouse. They can be physical containers (e.g. a roll of paper is a Container). A Container should be modeled as a resource if they are physical entities. They might also exist as a serial number if they are serialized and need to be tracked in inventory.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
Container	NVARCHAR(40)	No		Container identification. Can be a label number (LP) or a container number in the case of a reusable Container.	
ContainerClassID	INT(10,0)	Yes		ID for the Class of the Container (user defined).	CONTAINER_CLASS
DefinitionType	NVARCHAR(5)	Yes		<For Future Use>	
Facility	NVARCHAR(5)	Yes		Facility where the Container is currently located.	WAREHOUSE
Warehouse	NVARCHAR(10)	Yes		Warehouse where the Container is currently located.	WAREHOUSE
WarehouseLocationID	INT(10,0)	Yes		Position of the Container (link to its Warehouse Location).	WAREHOUSE_LOCATION
InContainer	NVARCHAR(40)	Yes		Link to the Container in the same table that is used to track the container.	CONTAINER
InventoryID	INT(10,0)	Yes		Link to the Inventory in the same table that is used to track the container.	CONTAINER
ContainerStatus	SMALLINT(5,0)	Yes		U	
CalculatedWeight	DECIMAL	Yes	(0,0)		
ActualWeight	DECIMAL	Yes	(0,0)		
MasterChildCode	NVARCHAR(1)	Yes			
MobileFlag	BIT	Yes	(0)		
PermanentFlag	BIT	Yes	(0)		
TextID	INT(10,0)	Yes			
LastMaintenanceDate	DATETIME	Yes	(getutcdate())		
NextMaintenanceDate	DATETIME	Yes	(getutcdate())		
EquipmentID	INT(10,0)	Yes			
MobilityID	INT(10,0)	Yes			
LogisticsID	INT(10,0)	Yes			

CONTAINER\_CLASS

Description

Stores information used to differentiate types of Containers from one another.

Columns

Column Name	Data Type	Nullable	Default	Description (Custom if Available)
ID	INT(10,0)	No		Unique identifier of a record (key) in a table.
Name	NVARCHAR(40)	Yes		Name of the entity.
TextID	INT(10,0)	Yes		Link to the TEXT table.
InventoryAllowed	BIT	Yes	(0)	<For Future Use>
EquipmentAllowed	BIT	Yes	(0)	<For Future Use>
MultipleUseAllowed	BIT	Yes	(0)	<For Future Use>
Shippable	BIT	Yes	(0)	<For Future Use>
MultipleCyclesAllowed	BIT	Yes	(0)	<For Future Use>
Kardex	BIT	Yes	(0)	<For Future Use>
CrossContainerAllowed	BIT	Yes	(0)	<For Future Use>
ParentContainerAllowed	BIT	Yes	(0)	<For Future Use>
TrackedWIP	BIT	Yes	(0)	<For Future Use>
DomainManagerID	INT(10,0)	Yes		ID of the Domain Manager.
PermanentFlag	BIT	Yes	(0)	Specifies if Container of this class is permanent.
MobilityFlag	BIT	Yes	(0)	Specifies if Container of this class is mobile.
UnitID	INT(10,0)	Yes		Link to UNIT table.

Prime Data

1 - PD, 2 - CC, 3 - MC, 4 - DC, 5 - Tracked, 100000001 - Virtual Master Container

## ▼ Chapter 1: Overview

Overview

Data Model Documentation

Data Model Documentation  
Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table joining, part 1

Table joining, part 2

Table joining, part 3

Table joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

Types, Classes, and Groups, part 1

Types, Classes, and Groups, part 2

Types, Classes, and Groups, part 3

## Types, Classes, and Groups, part 3

- ▶ Groups of entities are created in tables with 'GROUP\_' added to the entity name
- ▶ Unlike classes, for groups there is no reference in the entity table to the group table, but the opposite - in the group table there is a foreign key to the entity table (i.e. ProductID field in the PRODUCT\_GROUP table points to a record in PRODUCT table)
- ▶ Also unlike classes, entities can belong to more than one group
- ▶ The GROUP\_ table has a primary key which consists of three fields (Group\_, GroupType, and GroupClassID)
- ▶ Entity group tables typically have a primary key which consists of four fields: foreign key to the entity, and foreign key to the GROUP\_ table

**PRODUCT\_GROUP**

Description  
Contains the assignment of a product to various groups

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ProductID	INT(10,0)	No		Reference to a product (product number and product version)	PRODUCT
Group_	NVARCHAR(40)	No		Defines the assignment of the entity to a group (user defined)	GROUP_
GroupType	SMALLINT(5,0)	No		Type of the group	GROUP_
GroupClassID	INT(10,0)	No		Class of the group (user defined)	GROUP_
DisplayNo	INT(10,0)	Yes		Future use	
UnitID	INT(10,0)	Yes		Link to UnitID (unit characteristics)	UNIT

**PRODUCT**

Description  
The "PRODUCT" table is used to store the list of all valid products for the company. This table stores the master list of product (or materials). One record is created for each product and product revision

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
ID	INT(10,0)	No		Unique identifier of a record (key) in a table	
ProductNo	NVARCHAR(80)	No		The product number	
FLUID	NVARCHAR(36)	Yes		Unique identifier of the entity across multiple FlexNet instances	
Facility					
TextID					
UnitID					
RotationType					
RotationDays					
LotTrackingCode					
SerialTrackingCode					

**GROUP\_**

Description  
Store all Group keys. Any entity within the FlexNet database that can be grouped has a corresponding "GROUP\_" table that is linked to the entity and the "GROUP\_" table that defines the group.

Column Name	Data Type	Nullable	Default	Description (Custom if Available)	Links
Group_	NVARCHAR(40)	No		Group of entities	
GroupType	SMALLINT(5,0)	No		Type of the Group	GROUP_TYPE
GroupClassID	INT(10,0)	No		Class of the Group (user defined)	GROUP_CLASS
ParentGroup	NVARCHAR(40)	Yes		Reference to the Parent Group	GROUP_
ParentGroupType	SMALLINT(5,0)	Yes		Reference to the Parent Group Type	GROUP_
ParentGroupClassID	INT(10,0)	Yes		Reference to the Parent Group Class	GROUP_
TextID	INT(10,0)	Yes		Link to the TEXT table	
UnitID	INT(10,0)	Yes		Link to UnitID (unit characteristics)	UNIT



Overview

Data Model Documentation

Data Model Documentation  
Explained

Table Category Explained, part 1

Table Columns Explained, part 2

Prime Data, part 1

Prime Data, part 2

▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

Types, Classes, and Groups, part 1

Types, Classes, and Groups, part 2

Types, Classes, and Groups, part 3

Other Rules

## Other Rules

Here are few less important - but still vital - rules:

- ▶ ID rows are automatically incremented
  - The ID column is always the first column in the table
  - The initial value is 100000000
- ▶ A column with the GUID will have the name FUID and the type of nvarchar(36), not null
- ▶ The data processing in DELMIA Apriso Business Components is written using the “optimistic concurrency control”. This means that the RowVersionStamp column is checked to determine if the record used in the transaction has been changed. If a change is detected, it returns the “Table XYZ concurrency violation” error code





Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

Types, Classes, and Groups, part 1

Types, Classes, and Groups, part 2

Types, Classes, and Groups, part 3

Other Rules

LAB 1: Querying a Database

LAB 1: Querying a Database

LAB 1: Querying a Database

End of LAB

End of Course

## LAB 1: Querying a Database

### Task:

- ▶ Write a query retrieving data from the database

### What you will learn:

- ▶ How to use the Database Documentation
- ▶ How to join tables according to the DELMIA Apriso Data Model

### Requirements:

- ▶ Use Microsoft SQL Server Management Studio.
  - Login: sa
  - Password: Apriso2017
  - Use the APRISO database
- ▶ In case of any technical problems, please contact [DELMIA.Apriso.training@3ds.com](mailto:DELMIA.Apriso.training@3ds.com)



Remember to use the following to login and name Screens thorough this entire training:

- TRN<yourinitials> if your are an external self-paced learner
- TRN<yourtrigram> if you are a 3DS employee self-paced learner



Duration: 45 min





Prime Data, part 1

Prime Data, part 2

## ▼ Chapter 2: Data Model Rules

Primary Key, part 1

Primary Key, part 2

Primary Key, part 3

Foreign Key, part 4

Table Joining, part 1

Table Joining, part 2

Table Joining, part 3

Table Joining

Data Model Rules

Technical Columns

'Active' Column

UnitID

Multi Language

Types, Classes, and Groups, part 1

Types, Classes, and Groups, part 2

Types, Classes, and Groups, part 3

Other Rules

LAB 1: Querying a Database

LAB 1: Querying a Database

LAB 1: Querying a Database

End of LAB

End of Course

## LAB 1: Querying a Database

- ▶ Retrieve the following data from the database:
  - ProductNo (from the PRODUCT table)
  - SerialNo (from INVENTORY2 table)
  - AvailabilityDate (from SERIAL\_NO table - pay attention to the fact, that there are two column in the primary key in that table)
  - Warehouse (from the WAREHOUSE table)
  - Location (from the WAREHOUSE\_LOCATION table)
  - Extended translations for product and warehouse location (get them from the TEXT\_TRANSLATION table for the 1033 LanguageID - it's English)
  - QuantityOnHand (from the INVENTORY2 table)
  - DefaultUomCode (from the PRODUCT table)
- ▶ Sample script can be found on the training server.  
Go to: Desktop > Training Materials > 1-BPM Utilization Level 1

	ProductNo	SerialNo	AvailabilityDate	Warehouse	Extended	Location	QuantityOnHand	DefaultUomCode
1	X-RING-001	(null)	2016-03-02 22:17:05.000	PRD	Production	PRDPTW16	1.0000000000	EA
2	X-STRP-001	(null)	2016-02-12 14:01:12.000	PRD	Production	PRDPTW16	1.0000000000	EA
3	X-STRP-001	SER0001	2016-02-12 14:01:48.000	PRD	Production	PRDPTW16	1.0000000000	EA
4	X-STRP-001	SER0002	2016-02-12 14:02:05.000	PRD	Production	PRDPTW16	1.0000000000	EA
5	X-RING-001	SERMTA01	2016-03-02 22:18:03.000	PRD	Production	PRDPTW16	1.0000000000	EA
6	X-RING-001	SERMTA02	2016-03-02 22:18:23.000	PRD	Production	PRDPTW16	1.0000000000	EA
7	X-RING-001	SERMTA09	2016-03-02 22:19:31.000	PRD	Production	PRDPTW16	1.0000000000	EA
8	X-RING-001	SERMTA03	2016-03-02 22:18:31.000	PRD	Production	PRDPTW16	1.0000000000	EA
9	X-RING-001	SERMTA04	2016-03-02 22:18:40.000	PRD	Production	PRDPTW16	1.0000000000	EA
10	X-RING-001	SERMTA05	2016-03-02 22:18:56.000	PRD	Production	PRDPTW16	1.0000000000	EA
11	X-RING-001	SERMTA06	2016-03-02 22:19:05.000	PRD	Production	PRDPTW16	1.0000000000	EA
12	X-RING-001	SERMTA07	2016-03-02 22:19:14.000	PRD	Production	PRDPTW16	1.0000000000	EA
13	X-RING-001	SERMTA08	2016-03-02 22:19:22.000	PRD	Production	PRDPTW16	1.0000000000	EA