GDSN PROVIDER
USER GUIDE

STEP Trailblazer

Table of Contents

Introduction

About This Guide

This guide describes STEP GDSN Provider including how to setup data pools and use data pools to send data to different data recipients.

We recommend that you familiarize yourself with the key concepts of this guide.

The guide assumes that:

- Users have a basic knowledge of STEP.
- Users have a AS2 server setup with an In hotfolder and Out hotfolder to be able to send and receive data.
- Users and trading partners have adopted the GS1 GTIN, GLN, Global Data Dictionary (GDD) and Global Product Classification (GPC) standards.
- Users have a STEP GDSN Provider license or the STEP GDSN Provider license version 2.

If you need information on other STEP components, see the online help or the specific user guides.

GDSN Provider

The Global Data Synchronization Network (GDSN) is an internet-based global network and global registry that enables secure and continuous data synchronization between trading partners. This connection is made via a network of interoperable GDSN-certified data pools.

Within the GDSN network, trade items are identified using target markets and a unique combination of the GS1 Identification Keys called Global Trade Item Numbers (GTIN) and Global Location Numbers (GLN).

GDSN enables trading partners to synchronize data. Any changes to the data pool made by one company are automatically available to all of its trading partners.

GS1 Global Registry®

GS1 describes the GS1 Global Registry as the GDSN information directory that details who has subscribed to trade items or party data, guarantees the uniqueness of the registered items and parties, and ensures that all data pools in the network are complying with a standards-based set of validation rules.

How GDSN Works with Data Pools

GS1-certified data pools are electronic catalogs of standardized item data. They serve as a source and/or a recipient of master data.

The GDSN works together with data pools in the following way:

1. The data provider selects a source data pool and the data recipient selects a recipient data pool as a single point of entry to the GDSN.
2. The data provider registers product and company information in its source data pool. This information is also registered in the GS1 Global Registry.
3. The data provider agrees with a data recipient to synchronize data from the provider to the recipient. The provider then makes a publish request to the data pool, so that relevant registered items are sent to the recipient.
4. The data recipient makes a subscription request (Catalog Item Subscription -CIS).
5. If the subscription criteria match items that are registered in the GS1 Global Registry, the recipient's data pool is notified using a Catalog Item Notification (CIN) message, and then the synchronization takes place. Data is published from the data provider's data pool to the recipient's data pool.
6. After receiving the data, a Catalog Item Confirmation message (CIC) is sent from the data recipient to the data provider.

Common Standards

Standard	Description
GTIN	A Global Trade Item Number is a global identification number that can be used by a company to uniquely identify trade items. Trade items are defined as products or services.
GLN	The Global Location Number is a unique 13-digit identification number. The GLN can be used to identify a company's physical location and to identify corporate entities as well as a company's legal and functional entities. Each data provider and each data recipient has their own unique GLN that is used when publishing and subscribing for data.
GPC	To ensure that products are classified correctly and uniformly, GDSN uses GS1 Global Product Classification (GPC), a system that gives data providers and data recipients a common language for grouping products in the same way everywhere in the world.

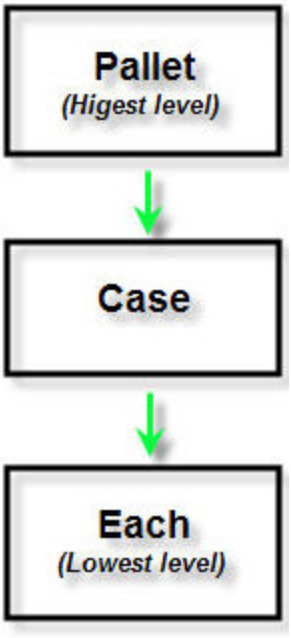
Key Terminology

Term	Definition
CIS	Catalog Item Subscription. Subscription message sent by a data recipient to establish a request for data.
CIN	Catalog Item Notification. Notification message used to transmit new or updated item information from the data provider to a data recipient. The CIN includes the requested product data.
CIC	Catalog Item Confirmation. Confirmation message sent to the data provider by a data recipient.
Synclist	Synchronization List. List that includes all synchronized catalog items (GTIN, GLN, TM). Keeps track of where data has been notified.

About Trade Item Hierarchies

The trade item hierarchy - or the packaging hierarchy - describes the relationship between trade items that contain other trade items, and it describes on which level in the hierarchy each item fits in. A trade item could, for example, belong to one of the following levels: base unit (Each), case, and pallet. Regardless of how many levels are in a hierarchy, the final level must be a base unit.

A parent item is an item that contains lower level trade items (children) in a packaging hierarchy. A child item is an item with a higher level trade item (parent) in a packaging hierarchy. A child item can have multiple parents, and it can therefore be included in many packaging hierarchies.



For detailed information about GSDN, see www.gs1.org.

GDSN Component Model Overview

The GDSN Provider uses a component model to define the objects, references and attributes of the component. These elements define the configuration of the GDSN Provider component, specify how STEP communicates with GDSN, and store the status of the products within GDSN.

You have two options for configuring the GDSN component model. You can either configure the component model manually or use the **Setup GDSN Publisher Data Model** dialog. We recommend that you use the dialog to configure the component model because most settings are related to the internal workings of the GDSN solution and can be set automatically. For more information, see Setting Up the GDSN Component Model.

Main Elements of the GDSN Component Model

The following section describes the main elements of the component model. For a detailed list of all elements of the component model see GDSN Provider Component Model Elements at the end of this topic.

GDSN Product

GDSN Products define the object types of the products and packaging objects that can be registered and published in GDSN. GDSN products must have a GTIN attribute.

Target Market Object Type

A GDSN product is registered to a target market. The target market defines which STEP contexts to use when product data are extracted for that particular market. The context is determined by a reference from the target market to the context. The target market has an attribute that contains the code of the target market. This code must follow the naming scheme for target markets defined by the GDSN data pool. Target markets are defined per data pool and all target markets for a data pool have the same parent object in STEP.

Recipient

A GDSN product is published to a data recipient. The recipient has a GLN attribute for the GLN of the data recipient. Recipients are defined per data pool and all recipients of a data pool have the same parent object in STEP.

Data pool

The Data pool object represents the source GDSN data pool where the GDSN products are registered. The data pool has a GLN attribute that holds the GLN of the data pool. The data pool defines the format through a reference to a format object. The data pool has an attribute for the GDSN data pool GLN and for the GDSN data pool user name.

Note: The GDSN data pool user name is not a STEP user name.

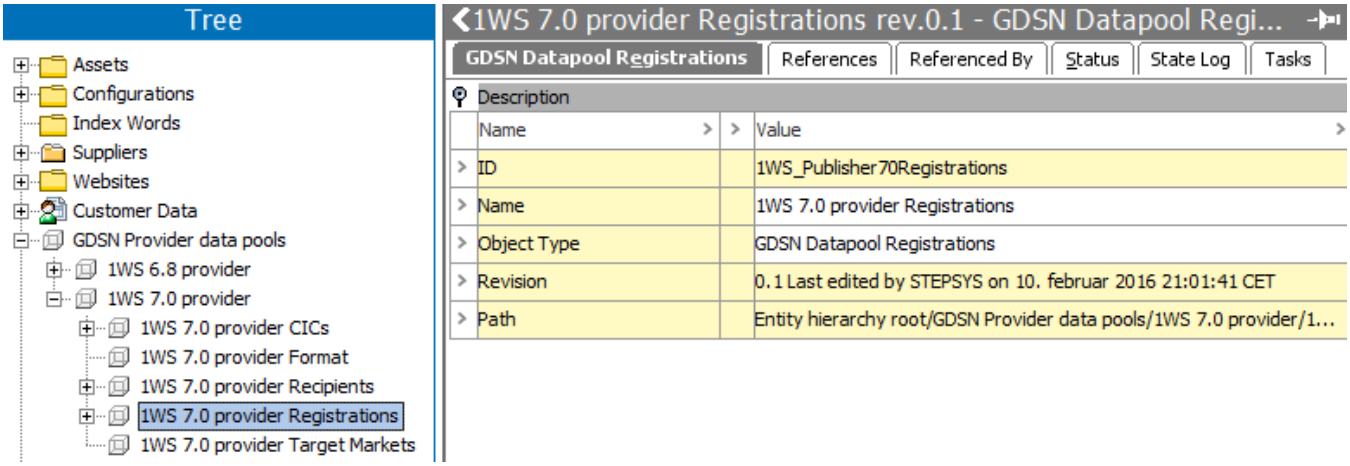
Communication between STEP and the GDSN data pool takes place through hotfolders. You configure the in and out hotfolders on the data pool in two attributes: GDSN AS2 hotfolder In and GDSN AS2 hotfolder Out.

All target markets, recipients, registrations and CIC objects that are valid for a data pool are children of the data pool object in STEP. The data pool assumes that all valid target markets are children of an object with the ID '<Datapool ID>TMs', the recipients are children of an object with ID '<Datapool ID>RCs', and so on.

The following table shows the complete object ID list.

Component model object	Expected ID
Target markets	<Datapool-ID>TMs
Recipients	<Datapool-ID>RCs
Registrations	<Datapool-ID>REGs
CIC objects	<Datapool-ID>CICs

The following screenshot shows the expected children of a data pool entity node. The GDSN Provider component will not function correctly if these objects do not exist. The default configuration setup creates these entities as children of the data pool.



Format

This is the data pool format. The GDSN Provider is format independent, which means that all GDSN data pool specific formats are configured individually. The GDSN data pool configurations are stored in a Format object. The Format object can contain format specifications, for example, for 1 World Sync. You can create any number of Format objects.

The format defines how the exported STEPXML of a GDSN product is translated to the XML format expected by the relevant GDSN data pool. The format also defines how the XML messages received from the GDSN data pool

are translated into messages that the GDSN Provider understands. The format supplies an XML schema for the XML format the GDSN data pool understands. The schema is stored in an asset that is linked to the format through a reference.

CIC (Catalog Item Confirmation)

The CIC object type is used to create objects that can run in a workflow that handles CIC messages. The CIC object contains a reference to the recipient that returns the CIC message and to the registration. You can place part of the CIC message in attributes on the CIC object. CIC objects are specified per data pool and all CIC objects have the same parent object in STEP.

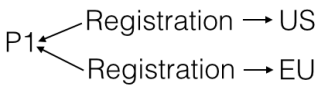
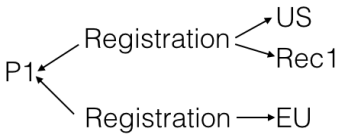
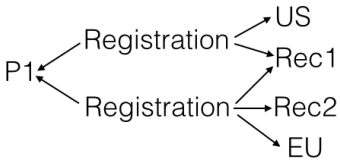
The CIC object stores the CIC status that is returned to a recipient when a product needs to be corrected. When data needs to be corrected, a CIC workflow is started that handles the correction. A CIC object is only created when a CIC workflow is started. If a returned CIC message does not require any data provider actions, no CIC object is created.

Status

The component model stores the registration and publishing status of the products in GDSN. Once a product is registered for a given target market, a registration entity is created with a reference to the product and the target market. The registration entity holds the status of the product registration of the given target market.

If the product is later registered to another target market, a new registration entity is created that refers to the product and the new target market. Two registration entity objects then exist for the product: one for the first target market and one for the second target market.

If a product is later published to one or more recipients in a target market, a reference is created from the registration entity that references the target market to each of the recipient entities. The status of the product published to the recipient is stored on the reference from the registration to the recipient. This makes it possible to record, for example, that one recipient has accepted the published product while another recipient has sent a CIC message requesting a correction of an attribute.

Product registered in two markets	Published to recipient Rec1 in US market	Published to recipients Rec1 and Rec2 in EU market
		

Package hierarchy

The component model uses product to product references to model package hierarchy relationships. Parent items have a product to product reference to the children in the package hierarchy. Each level in the package hierarchy, for example, a Pallet to Case or a Case to an Each, can use the same or different product to product reference

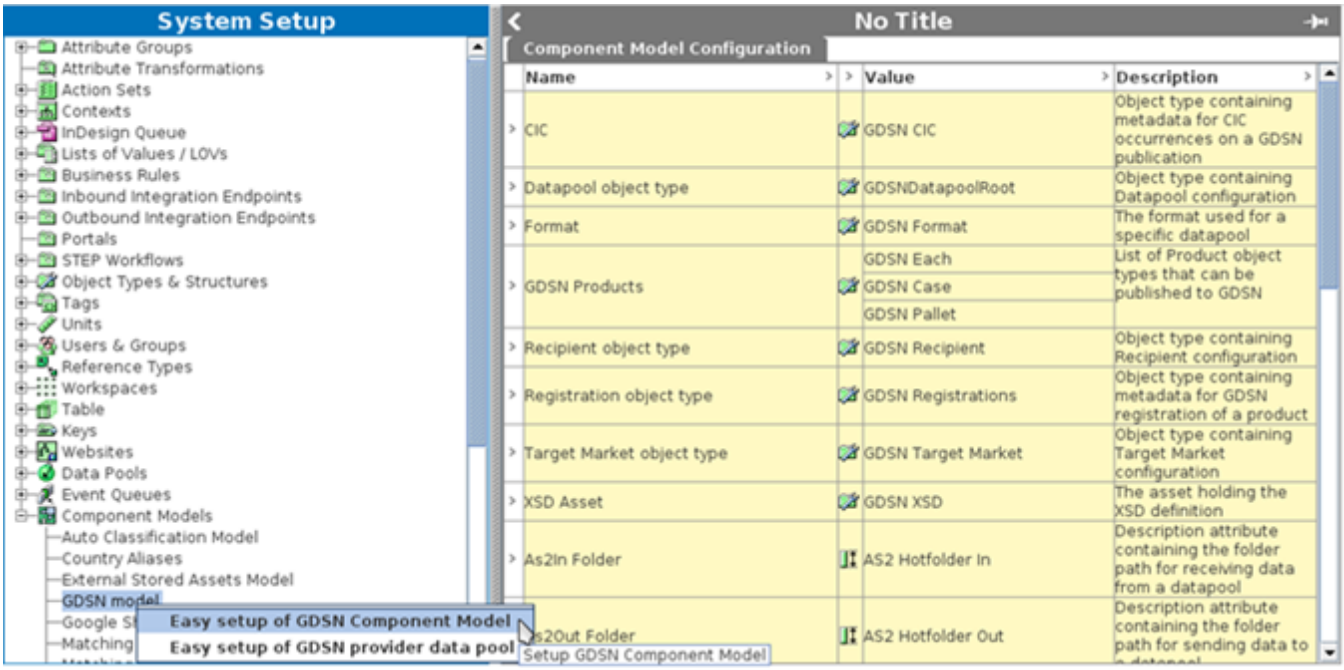
types. You can therefore configure the component model to use one product to product reference type for the Pallet to Case relationship and another product to product reference type for the Case to Each relationship. An attribute on a package hierarchy reference type specifies the quantity of the next level in the package hierarchy. Another attribute on the reference holds the registration status.

Setting Up the GDSN Component Model

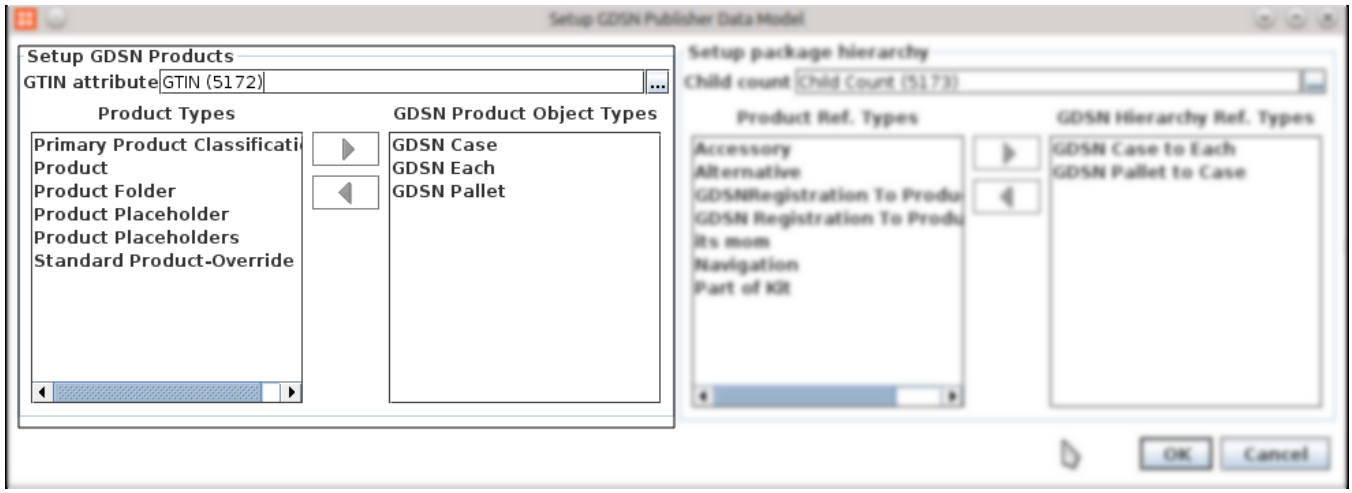
The GDSN component model currently consists of 33 different aspects, and they all have to be configured. Many of the aspects are dependent on each other and related to the internal workings of the GDSN solution. Most of them, however, can be configured automatically.

We therefore recommend that you use the **Setup GDSN Publisher Data Model** dialog to configure the component model. When you use the dialog you specify settings for 4 aspects and the remaining aspects are then specified automatically.

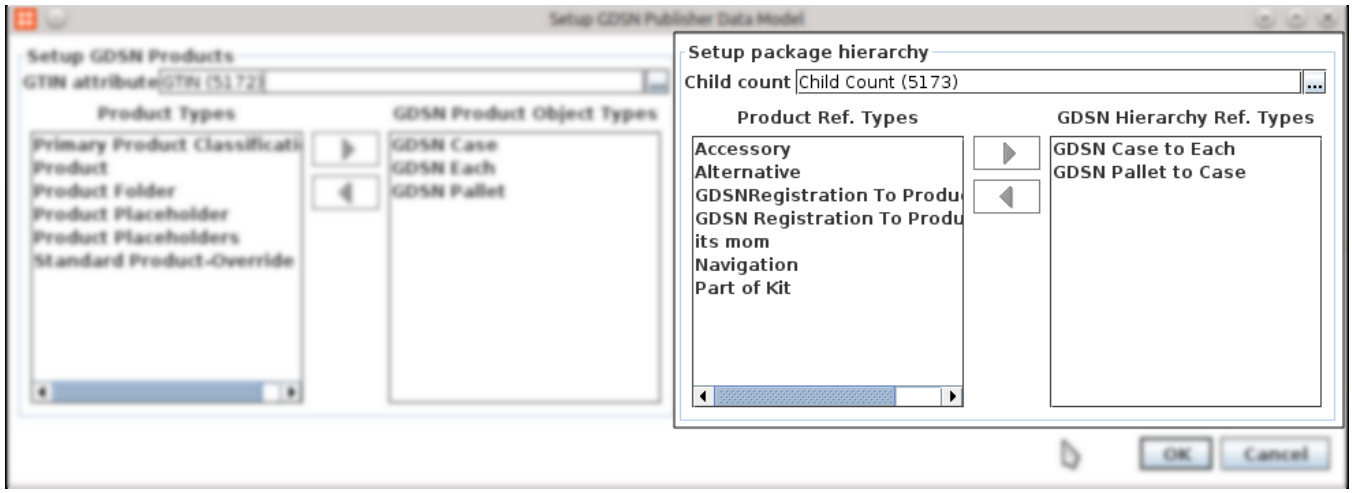
1. In **System Setup**, expand **Component Models**.
2. Right-click **GDSN Model**, and then choose **Easy setup of GDSN Component Model**.



3. In the **Setup GDSN Products** area, specify the following:
   - **GTIN attribute**: Select the attribute that you want to hold the Global Trade Item Number. Click the ellipsis button **(…)** to search or browse for the relevant attribute.
   - **GDSN Product Object Types**: Select the product object types that can be published to the GDSN network.



4. In the **Setup Package Hierarchy** area, specify the following:
   - **Child Count**: Select the attribute that you want to hold the quantity of the next lower level trade item. This attribute applies to the package hierarchy link from a GDSN product to a child product in the package hierarchy.
   - **GDSN Hierarchy Ref. types**: Select the product reference types to be used when the GDSN package hierarchy is created.



The configuration of GDSN Product Object Types is mandatory, whereas the manual configuration of the other aspects described in the previous steps is optional. If you do not specify the settings manually, they are created automatically. If you choose an existing GTIN attribute, it is automatically made valid for all the selected GDSN product object types. Likewise, the package hierarchy child count is automatically made valid for all the package hierarchy link types.

The GDSN product object types and the package hierarchy links must fit together. A package hierarchy link must have a source or target type that is of one of the selected GDSN product object types. If the object types and links do not fit together, a message is displayed saying that the selected source type is not a GDSN product, and the OK button is disabled.

If you do not select a package hierarchy, a single package hierarchy link is created automatically. This reference type will then have all the selected GDSN product object types as valid sources and targets.

GDSN Provider Component Model Elements

The following table describes the elements of the Provider component model.

Name	Type	Description	How it is used
CIC	Object type	Object type that contains meta data for CIC occurrences on a GDSN publication.	When a recipient sends a CIC message, a new entity object of this type is created.
Datapool object type	Object type	Object type that contains data pool configuration.	Objects of this type are created automatically. Specify an ID and name.
Format	Object type	The format used for a specific data pool.	Objects of this type are created automatically. ID and name are based on the data pool. Format objects are special because they include tabs for inbound and outbound configurations.
GDSN Products	Object type	List of product object types that can be published to GDSN	These are the products and packaging objects. Must be pre-configured and then selected in the **Setup GDSN Publisher Data Model** dialog.
Recipient object type	Object types	Object type that contains recipient configuration	This type is created automatically. You must create the entity objects when the component model has been setup. The recipient objects must be created in an entity folder as a child of the data pool entity object.
Registration object type	Object type	Object type that contains meta data for GDSN registration of a product	This type is created automatically. Objects of this type are created by the GDSN Provider component when a product is registered in GDSN.

Name	Type	Description	How it is used
Target Market object type	Object type	Object type that contains target market configuration	This type is created automatically. You create entity objects of this type to represent the target markets that a product can be registered to. Create the target markets object in an entity folder as a child of the data pool entity object.
XSD Asset	Object type	The asset that holds the XML schema definitions	This object type is used to store the schema that the generated XML is validated against before the XML is sent to the data pool.
As2In Folder	Attribute	Description attribute that contains the folder path for receiving data from data pool	The attribute points to a hotfolder where files from the GDSN network are picked up by the inbound integration endpoint.
AS2Out Folder	Attribute	Description attribute that contains the folder path for sending data to a data pool.	This attribute points to a hotfolder where XML files for the GDSN network are located. It is assumed that the files are picked up by the GDSN network using an AS2 server.
CIC status	Attribute	Description attribute that contains the CIC status on the reference that relates the registration to the recipient.	Contains the CIC status for a publication of a product to a given recipient.
Datapool GLN	Attribute	Description attribute that contains the GLN of the data pool.	The data pool itself also has a GLN. So, for example, the 1WorldSync data pool has a GLN. The data pool's GLN is part of the protocol within GDSN.
Datapool Username	Attribute	Description attribute that contains the user name of the provider to the data pool.	The user name that is used when sending and receiving data from the data pool. This is not a STEP user name.
Format Inbound Configuration	Attribute	The inbound integration configuration for this format.	This attribute contains the inbound configuration of the format. The inbound configuration is viewable in the **Inbound** tab on the format editor.

Name	Type	Description	How it is used
Format Outbound Configuration	Attribute	The outbound integration configuration for this format.	The attribute contains the outbound configuration of the format. The outbound configuration is viewable in the Outbound tab on the format editor.
GTIN	Attribute	Specification attribute that contains the GTIN	This attribute is the GTIN of the product or the packaging objects that are part of a package hierarchy.
My GLN	Attribute	Description attribute that contains the GLN for a registration.	Attribute that identifies a company's registering and publishing items (the provider GLN).
Package Hierarchy Reference Status	Attribute	Description attribute that contains the status of a packaging object's registration.	Meta data on the reference from a packaging object to the lower level item that shows the status of the registration.
Pending command	Attribute	Description attribute containing the pending command for the registration if another command needs to be run before the current command	This attribute must be valid for the registration object type. It is used to store the pending command used by the Business Actions 'Set PendingCommand' and 'Execute Pending Command'
Publish status	Attribute	Description attribute that contains the status of a registration to recipient reference.	Metadata on the reference from a registration to a recipient. The attribute contains the status of the publication of a product or a packaging object.
Quantity of next lower level package	Attribute	Description attribute that contains the quantity of the next lower level packages in this package.	This attribute specifies the number of lower level objects a given packaging object can contain. The lower level packaging object is given as the source of the packaging hierarchy reference.
Recipient GLN	Attribute	Description attribute that contains the GLN of a recipient	The GLN of the recipient of a publication.

Name	Type	Description	How it is used
Registration Status	Attribute	Description attribute that contains the status of a registration.	Meta-data on the Registration object that contains the status of the registration to a given target market.
Target Market Name	Attribute	Description attribute that contains the target market.	The name of the target market as defined by the data pool. If this attribute is not set, or if it uses name other than the one defined by the data pool, the registration to the target market will fail.
CIC to Recipient reference	Reference type	Reference type that relates the CIC to the recipient.	The reference is created when a CIC message is received from a recipient, and links the recipient to the CIC message.
CIC to Registration reference	Reference type	Reference type that relates the CIC to the registration.	The reference is created when a CIC message is received from a recipient, and links the CIC message to a registration and thereby to a product or packaging object and a target market.
Datapool to format reference	Reference type	Reference type that relates the data pool to its format definition.	The system will use this reference to find the format node that holds the data pool configuration.
Format to asset reference	Reference type	Reference type that relates the format to the asset that contains the XML schema asset.	The XML schema is used to validate the generated XML before the XML is sent to the data pool.
Package Hierarchy References	Reference type	The reference type that relates a packaging object to its child packaging object.	This reference type must be created manually before using the setup dialog. The references are used to build the packaging hierarchy.
Registration to Product Reference	Reference type	The reference type that relates a registration object to a packaging object.	The reference is created when a product or packaging object is registered in the GDSN network.
Registration to Recipient Reference	Reference type	The reference type that relates a registration object to a recipient.	The reference type is created when a product or packaging object is published to the GDSN network.

Name	Type	Description	How it is used
Registration to Target Market Reference	Reference type	The reference type that relates a registration object to a target market.	The reference is created when a product or a packaging object is registered in the GDSN network and shows the target market for a registration.
Target Market to Context Reference	Reference type	The reference type that relates a target market to a context.	The referenced context is used when data is exported for a GDSN registration. Values for attributes and so on are defined by the language settings of the context. Must be created when a new target market is created.

Setting Up GDSN Provider Data Pools

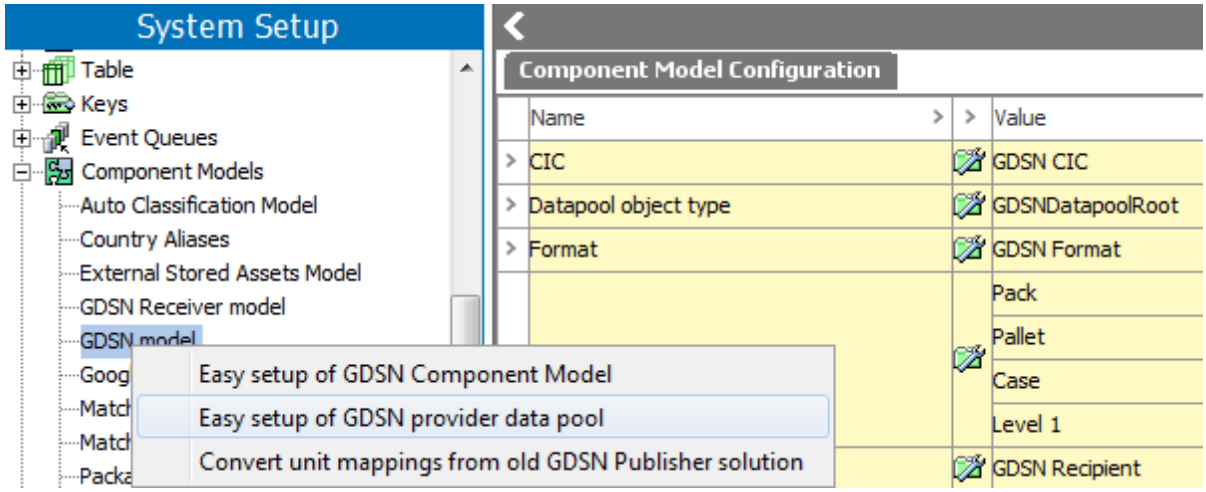
A STEP provider data pool handles the communication between the STEP system and the external data pool. The following requirements must be met for the STEP provider data pool to function correctly.

- An outbound integration endpoint to dispatch messages to a hot-folder.
- An inbound integration endpoint to pick up the incoming message from a hot-folder. Both endpoints will be configured with the data pool as receiver or delivery plug-in.
- The setup of an AS2 server is required to take the messages from the out-folder and transfer them to the external data pool and to deliver the external messages from the data pools in the in-folder. Setting up the AS2 server is beyond the scope of this documentation.

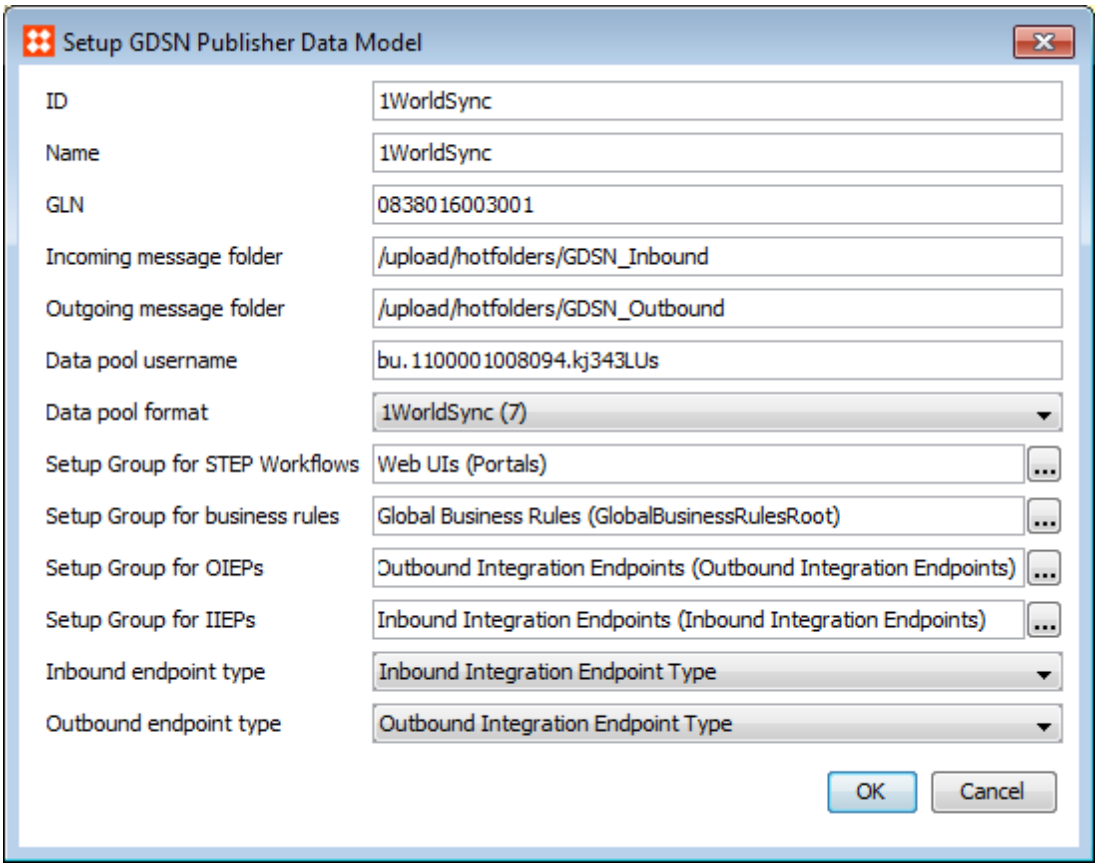
You can use the **Setup GDSN Publisher Data Model** dialog to set up the STEP data pool and the integration endpoints. The dialog contains information on how to communicate with the external data pool and is also used to group and store information related to the communication with the external data pool.

To Set Up The GDSN Data Pool

1. In **System Setup**, expand **Component Models**.
2. Right-click **GDSN Model**, and then choose **Easy setup of GDSN Provider Data Pool**.



3. In the **Setup GDSN Publisher Data Model** dialog, enter information for your desired data pool:



Field	Description
ID	The STEP ID of the STEP data pool object. This ID is also used as ID prefix for other STEP objects created under the data pool such as IIEPs, OIEPs, and grouping entities.
Name	The STEP name of the data pool
GLN	The Global Localization Number (GLN) of the data pool. For example, 838016003003 for 1WorldSync.
Incoming message folder	The folder messages from the external data pool are delivered to.
Outgoing message folder	The folder messages to be transferred to the external data pool are taken from.

Field	Description
Data pool user name	The user name / password to be included in the messages to the external data pool as part of the identification and authorization of the messages.
Data pool format	The format helps determine which templates to use for the messages sent to the external data pool. It can also contain information about the structure of the messages that the data pool can receive. This way, messages can be validated before being dispatched to the external data pool.
Setup Group for STEP Workflows	Depending on the selected data pool format, a STEP workflow that handles CICs (Catalog Item Confirmation) is created. The workflow is placed in the selected setup group. This field is automatically populated with any setup group that allows the creation of STEP workflows. If no setup group is found, a warning message is displayed before the setup dialog is displayed.
Setup Group for business rules	Depending on the selected data pool format, various business actions are created. (The business rules are used, for example, to change the status of an issue when an acknowledge message for a registration message is received). These business rules are placed in the selected setup group. This field is automatically populated with a random setup group that allows the creation of business rules. If no setup group is found, a warning message is displayed before the setup dialog is displayed.
Setup Group for the OIEPs	Setup group for the outbound integration endpoint used to generate messages to the external data pool. This field is automatically populated with a random setup group that allows the creation of outbound integration endpoints. If no setup group is found, a warning message is displayed before the setup dialog is displayed.
Setup Group for the IIEPs	Setup group for the inbound integration endpoint is used to pick up and process messages from the external data pool. This field is automatically populated with a random setup group that allows the creation of inbound integration endpoints. If no setup group is found, a warning message is displayed before the setup dialog is displayed.
Inbound endpoint type	Select the type of inbound integration endpoint (IIEP) you want to create. This field is automatically populated with a random outbound endpoint type that is valid for the inbound integration endpoints type.
Outbound endpoint type	Select the type of outbound integration endpoint (OIEP) you want to create. This field is automatically populated with a random outbound endpoint type that is valid for the outbound integration endpoints type.

After setting up the data pool object, configure the target markets and recipients, and complete the format of the outbound message template. For more information, see Data Pool Configuration.

Data Pool Configuration

When you have setup the data pool object, you need to configure target markets and recipients and complete the format of the outbound message template.

A data pool has 5 grouping entities:

- **Target Markets Group:** Contains information about all the target markets used by this data pool.
- **Recipients Group:** Contains information about all the recipients used by this data pool
- **Registrations Group:** Contains information about which products are registered with the data pool - and for which target markets.
- **CIC Group:** Contains information about the CICs (Catalog Information Confirmations) the recipients have sent back.
- **Format**: Contains the configuration of the format of the outbound messages.

To Create Target Markets

1. In the **Tree**, expand the relevant data pool, and then select the target market entity.

   You must create all target markets for the data pool in this folder, and the object type must be the target market type configured in the component model. This is the only object type that is valid below the target market folder.

2. Right-click the **Target Markets** entity, and then select **New Entity Note**.
3. Enter an **ID** and a **Name**, and then click **Create**.
4. In the **Tree** select the entity you just created, and then in the **GDSN Attr Target Market Name** field, enter the GDSN name of the target market.
5. On the **Reference** tab, in the **GDSN Target Market to Context** field, link to the contexts to be used when messages are sent out in this target market.

To Create Recipients

1. In the **Tree**, expand the relevant data pool, and then expand the recipients entity.

   You must create all recipients for the data pool in this folder, and the object type must be the recipient type configured in the component model. This is the only object type that is valid below the recipient folder.

2. Right-click the **Recipients** entity, and then select **New Entity Note**.
3. Enter an **ID** and a **Name**, and then click **Create**.
4. In the **Tree** select the entity you just created, and then in the **GDSN Attr Recipient GLN** field, enter the GLN of the recipient.

To Complete the Format Templates

Before you can register products in the data pool, you must configure the format templates. First, complete the generic XML template with all the XML structures for the attributes to registered. Next, complete the mappings in the Export Manager. You have to map the unmapped targets of mandatory data in the register template. For more

information, see Configuring the Outbound Message Format and Data Export Wizard in the Export Manager documentation.

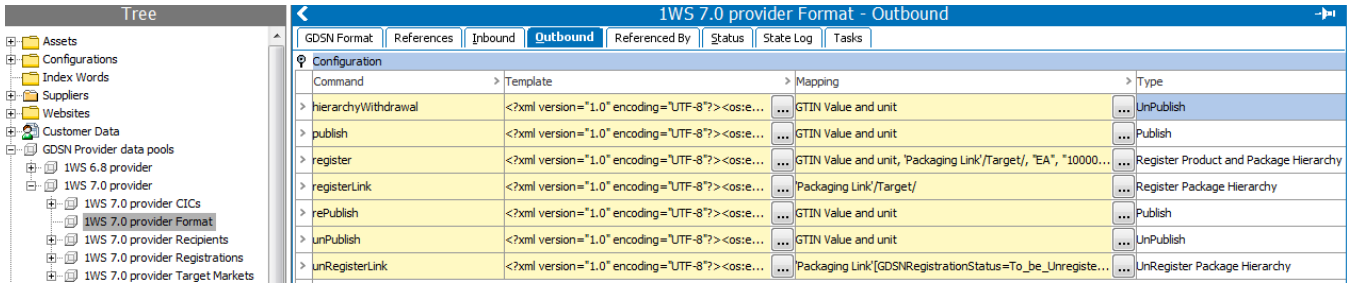
Configuring the Outbound Message Format

Before you can register products in the data pool, you must configure the format templates. The outbound format configuration defines the format that is used when messages are generated for a GDSN data pool. The configuration is a list of the message types that can be generated. Each message contains four components:

- A command
- A generic XML template
- Export mappings
- The message type

To Configure Format Templates

1. In the **Tree**, expand the relevant data pool, and then click the **Format** entity.
2. Click the **Outbound** tab. The tab has the following columns:



- **Command**

  The command is a key that identifies the type of the message. The command key is stored in the configuration of the business actions that generate messages for GDSN, and it is used in the configuration of the Web UI action buttons. See the **GDSN Web UI Buttons** section of the **GDSN Provider** documentation.

- **Template**

  The generic XML follows the syntax of the generic XML export format (see Generic XML). The GDSN Provider component defines some additional generic XML parameter tags:

  To edit the XML template, click the ellipsis button **(…)**. The **Create new template** dialog appears. The following parameters are available:

Parameter Tag	Description
<?Parameter	The GLN the product is registered from. Configured in either the business action starting

Parameter Tag	Description
MyGLN?>	the endpoint or in the configuration of the register / publish buttons in the Web UI.
<?Parameter Username?>	The data pool user-name. Is configured directly on the data pool.
<?Parameter DataPoolGLN?>	The data pool GLN. Is configured directly on the data pool.
<?Parameter TargetMarket?>	The target market attribute value from the selected target market. Is configured on the target market entity.
<?Parameter RecipientGLN?>	The GLN of the selected recipient. Is configured on the recipient.
<?Parameter TimeStamp?>	A time stamp in the format: yyyy-MM-dd'T'HH:mm:ss
<?Parameter MessageID?>	A string generated from the message type name, a time stamp, and a random number, to be used as a message ID.
<?Parameter TargetMarket|{att_id}?>	The value of the Target Market attribute with ID att_id.
<?Parameter Recipient|{att_id}?>	The value of the Recipient attribute with ID att_id.

- **Mapping**

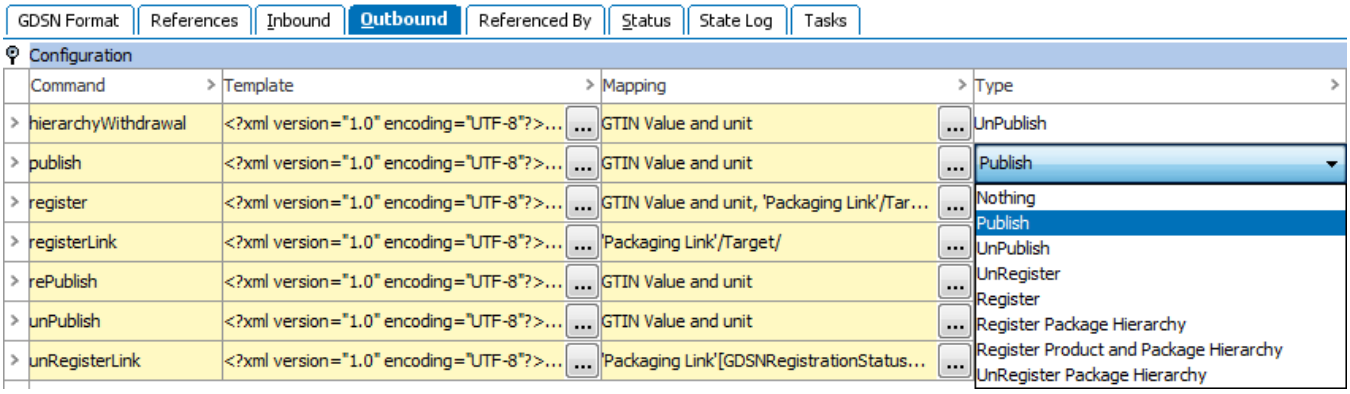
The export mappings define the mapping between the STEP data model and the targets defined by the generic XML template.

In the **Mapping** column, click the ellipsis button **(…)** , next to the relevant mapping.

For information about the Mappings dialog, see the Export Manager documentation.

- **Type**

The type of a message is used to update the status of the products sent to GDSN. If the message type is 'register', for example, you can update the status of the Registration to 'Registration Pending'. If 'nothing' is selected, the status is not changed.



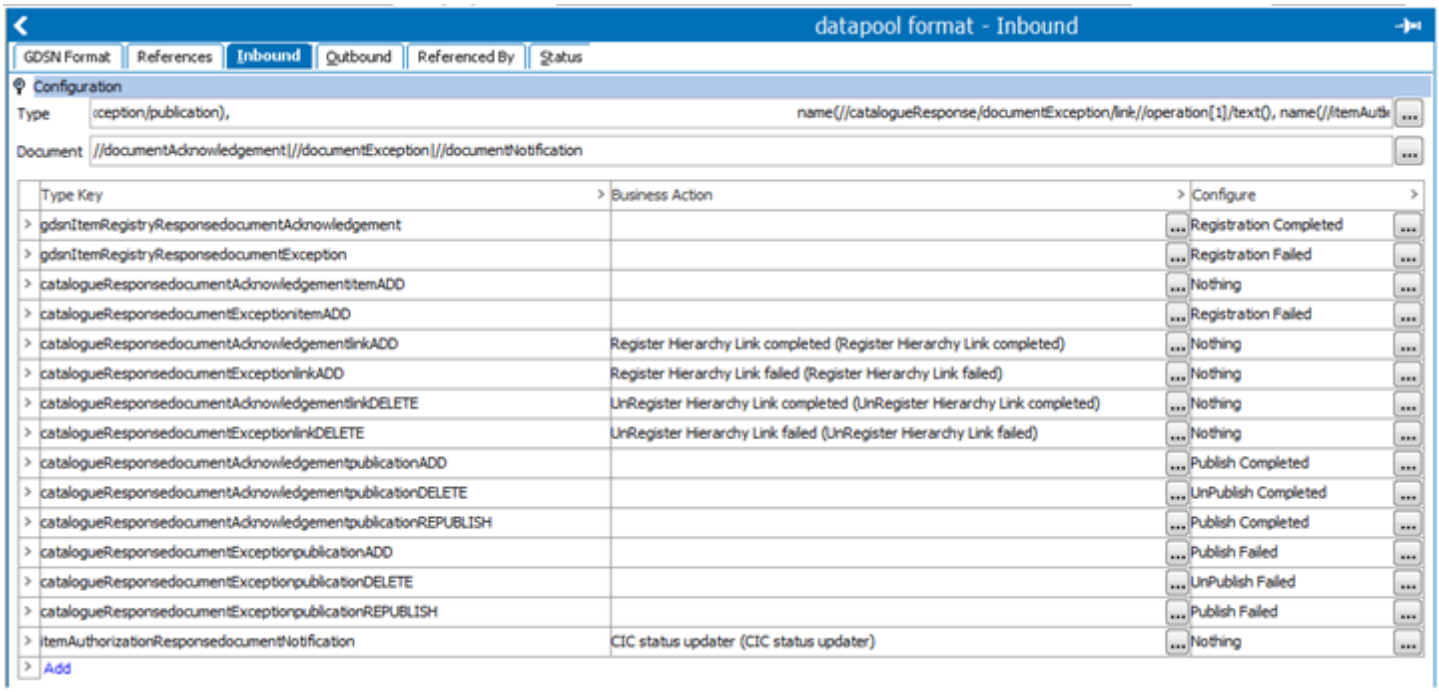
The following message types are available:

Status	Description
Nothing	Used when no status updates are required for the outbound message
Publish	Sets the publish status to publish_pending
UnPublish	Sets the publish status for the product to unpublish_pending
UnRegister	Sets the registration status to Unregister_pending
Register	Sets the registration status to register pending
Register Package Hierarchy	Sets the registration status for all objects in the hierarchy to register_pending; it is assumed that the products are already registered
Register Product and Package Hierarchy	Sets the registration status on all products in the hierarchy to register_pending and the same for the objects.
Unregister Package Hierarchy	Sets the registration status of the hierarchy to Unregister_pending

Configuring the Inbound Message Format

The inbound configuration defines the way the data pool receives messages from the GDSN. The inbound tab has 3 areas:

- **Type** field: XPath to find the type of the incoming message
- **Document** field: XPath to split messages into documents
- Configuration table: table to configure message types



Type field

The type field contains an XPath to determine the type of the incoming message. It evaluates the XPath on the message and uses the result as a key to finding the message type.

The following shows an XML sample:

```
<note>
<to>Bob</to>
<from>Frank</from>
<heading>Reminder</heading>
<body>Updates due on Tuesday.</body>
</note>
```

To find out whether this is a note or a letter message type, apply the following XPath:

```
concat(name(//note),name(//letter))
```

In this example, the note tag is returned because note is present in the XML.

Document

Sometimes there are multiple actions in a GDSN message, for example, when more products are registered at the same time. In this case several documents are returned for parsing. The document XPath is used to specify which documents are to be handled separately. In the XML example above, the following XPath returns 2 parts of the XML.

```
//body|//heading
```

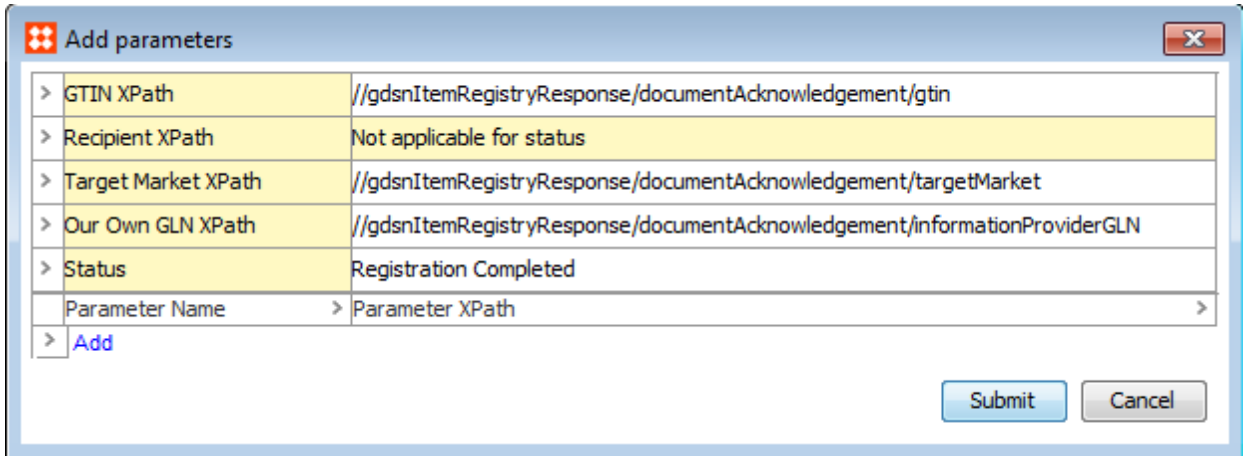
These XML parts are removed from the XML and inserted one at a time. The result messages are evaluated to determine the message type via the type XPath.

Configuration table of message types

The key that is found using the type from the previous example is used to determine what to do with the incoming message.

For each type key in the **Type Key** column, the related business rule is displayed when applicable in the **Business Action** column, and the related configuration is displayed in the **Configure** column.

1. To setup the configuration of a specific document type, click the ellipsis button **(...)** in the **Configure** column. The **Add Parameters** dialog is displayed.



2. Specify the XPaths that are applicable for the document type to identify important attributes in the document. In some cases the Recipient XPath is not applicable.
3. Specify the status that you want this action to set on the targeted item. If no status is applicable, choose None. If you chose none, verify that the status of your item is correct after this is run. You can for example use business action to verify this.

4. Optionally, add a parameter to the action. This requires a name and an XPath. The XPath is used to find the information needed in the document, and the name is used to identify this information later. The mapped information is accessed is by using the business rule that is associated with the current action. In the business action, use the JavaScript Plug-in and bind in the GDSN Data Map. The data can now be found using the parameter name.

Using Business Rules with GDSN

Business rules can be used in a variety of ways in combination with the GDSN component. They can, for example, be used to send out messages, update status, and start workflows. The following describes how to use GDSN specific business actions and how to run business actions when receiving messages from GDSN.

Business action on inbound messages

In the inbound format configuration you can configure which business actions should be run when different message types are received. You can use this to update status, transition workflows or send out notification e-mails.

In the inbound format configuration, you can configure a map from strings to XPaths. When a message is received, the XPaths are evaluated and a map is created from the string keys to the evaluated values. This map can be bound into a JavaScript business action so that the business action can access parts of the incoming message.

For general information about how to set up business actions, see Business Actions Overview in the Business Rule Documentation.

Business Actions for GDSN

1. In **System Setup**, expand **Business Rules**, and then select the relevant action.
2. On the **Business Rule** tab, in the lower left corner, click **Add New Business Action**.
3. Click the **Edit Operation**  icon.
4. In the **Edit Operation** dialog, from the drop-down list, select GDSN.

The following actions are available:

Execute command

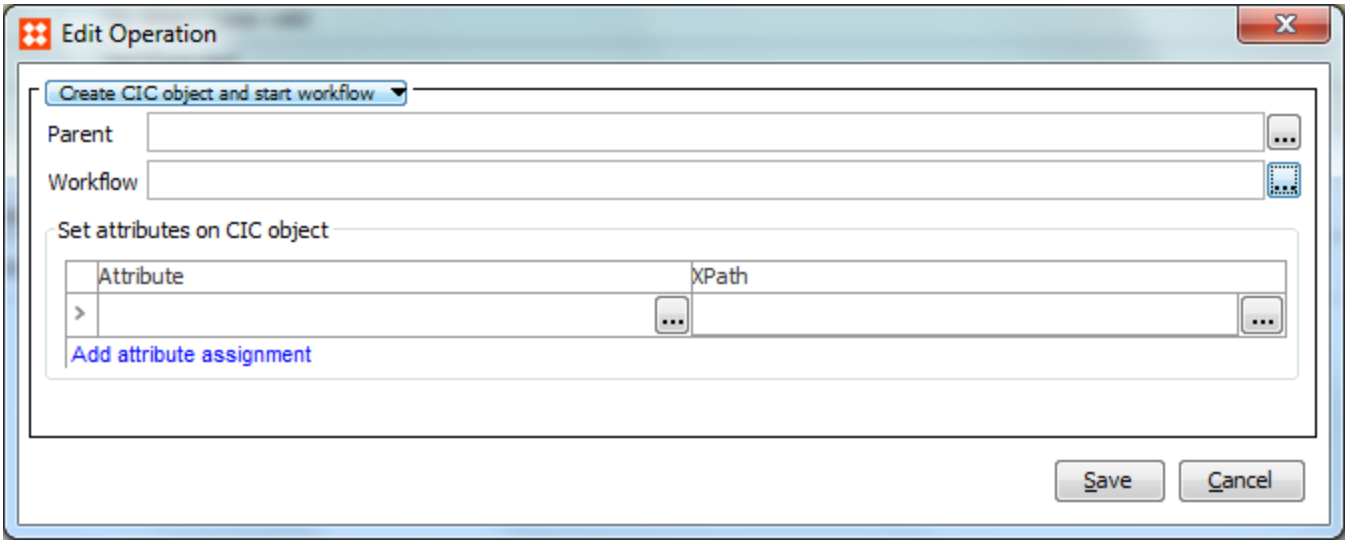
This action sends out a message to a data pool.

Field	Description
Command	Enter the command parameter that matches the command from the outbound configuration of the message type to be sent out.
Endpoint	Specify the outbound endpoint to be used to send out the message. A list of available endpoints displays when you click the ellipsis button **(...)**.

Field	Description
Expand package hierarchy	When checked, selecting a hierarchy top node exports the entire hierarchy below the node.
GLN	Specify the GLN the products must be sent from. This is the GLN the MyGLN parameter tag resolves to.
Recipient	Optional For a publish template, specify the recipient you want to publish the products to. The Recipient parameter tag will contain the GLN of the recipient.
Target Market	Specify the target market the message should be sent in. The target market parameter tag will contain the target market attribute value of the selected target market.

Create CIC Object and Start Workflow

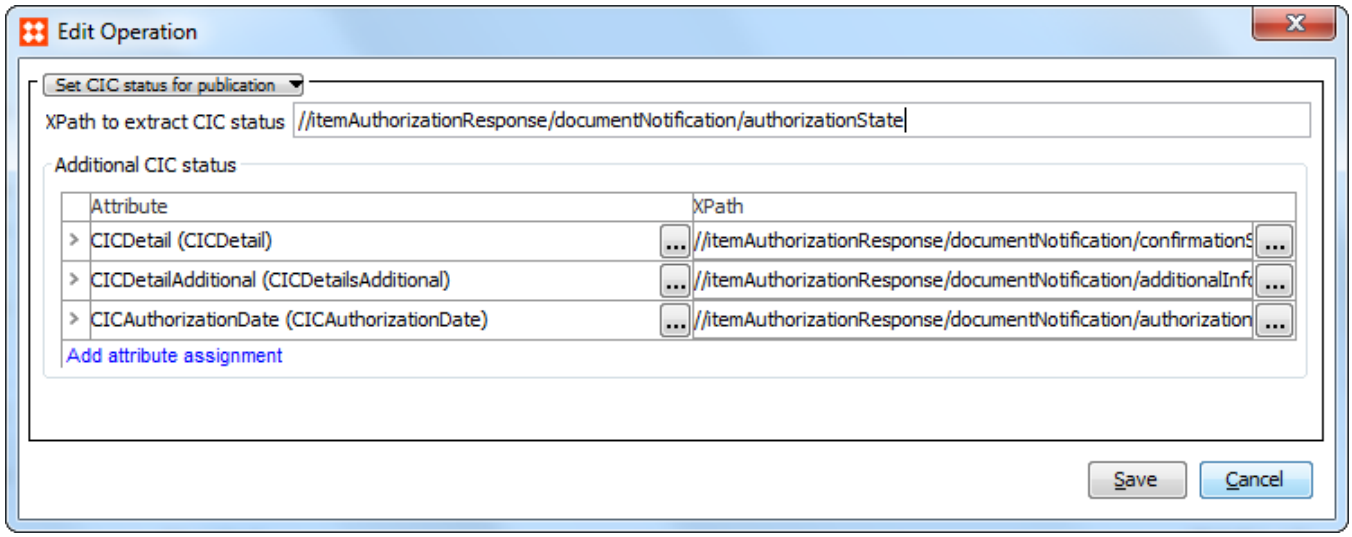
This plug-in creates a CIC entity and starts it in a workflow. It is connected with the product the CIC applies to and with the recipient that sends the CIC. Furthermore, using XPaths, it is possible to place parts of the CIC message in an attribute on the CIC object.



Field	Description
Parent	The CIC entity is created below the entity specified.
Workflow	The CIC entity is started in the workflow specified.
Set attributes on CIC object	Click the **Add attribute assignment** link. Select the attributes to be set on the CIC object and the XPaths to be evaluated to find the values from the inbound message.

Set CIC Status for Publication

Sets the CIC status for a publication to a recipient.



Field	Description
XPath to extract CIC status	Mandatory Specify an XPath that points out the CIC status in the inbound message.
Additional CIC status	Optional Specify an XPath to additional CIC information and point the attribute that should be used to store this information. For example, the additional CIC information could be text values that go along with the CIC999 status or it could be the date CIC status date.

Update Package Hierarchy Status

Sets the registration status of a package hierarchy link.

In the **Child GTIN XPath** field, specify an XPath that points out the GTIN of the child product in the inbound message.

From the **Status after Updating** list, select the status that you want the link to have after the action has been run.

'Evaluate JavaScript' with Binding to 'GDSN Publisher Product Validation'

The 'Evaluate JavaScript' plugin in connection with the 'GDSN Publisher Product Validation' binding provides the option to write JavaScript business rules that can access information about a product instance that is registered for a given data pool.

For example, this could be used to write a script that can validate if all required attributes for a given data receiver have been populated on a given product. For more information about the Validation Condition option, see GDSN Publish Action Button or GDSN Register Action Button.

Set Pending File

This Business Action is used for the 1WorldSync data pool in conjunction with using the 'Pre command' option in the Web UI (GDSN Web UI Buttons). When a pre-command is executed, the system will await a response from the data pool and GDSN before the real command is executed.

For the 1WorldSync data pool, these responses are not combined in one file and will thus not be received at the same time. Instead the responses are received asynchronously, one response message will be received from the 1WorldSync data pool and another response message will be received from the GDSN. Only when both responses have been received, is the real command (the pending command) executed.

The 'Set pending command' operation for this business action tracks which command should be executed when the response messages have been received for the pre-command, the pending command will be stored in a description attribute that is valid for the registration object (see GDSN Provider Component Model Elements).

The Set Pending File business action is triggered via the Inbound format configuration with the Type Key 'catalogueResponsedocumentAcknowledgementitemADD'.

Originating Message ID: Specify the XPath for the Originating Message ID. The default value is //originatingMessageId/text().

Execute Pending File

This Business Action will execute the pending command that is stored in a description attribute on the registration object. For further information, see Set Pending File above.

The Execute Pending File business action is triggered via the Inbound format configuration with the Type Key 'gdsnItemRegistryResponsedocumentAcknowledgement'.

GDSN in Web UI Overview

Handling GDSN in Web UI starts with creating a Web UI that shows information about GDSN registrations, publications, and the package hierarchy. For details, see **Create a GDSN Web UI**.

In addition to typical Web UI elements, additional tabs and buttons are available to perform GDSN functions.

Details about configuring the tabs that are available for Product Details are included in these sections:

- GDSN Registration Tab
- GDSN Hierarchy Tab

An overview of the buttons that are available to handle required GDSN actions can be found in the GDSN Web UI Buttons section.

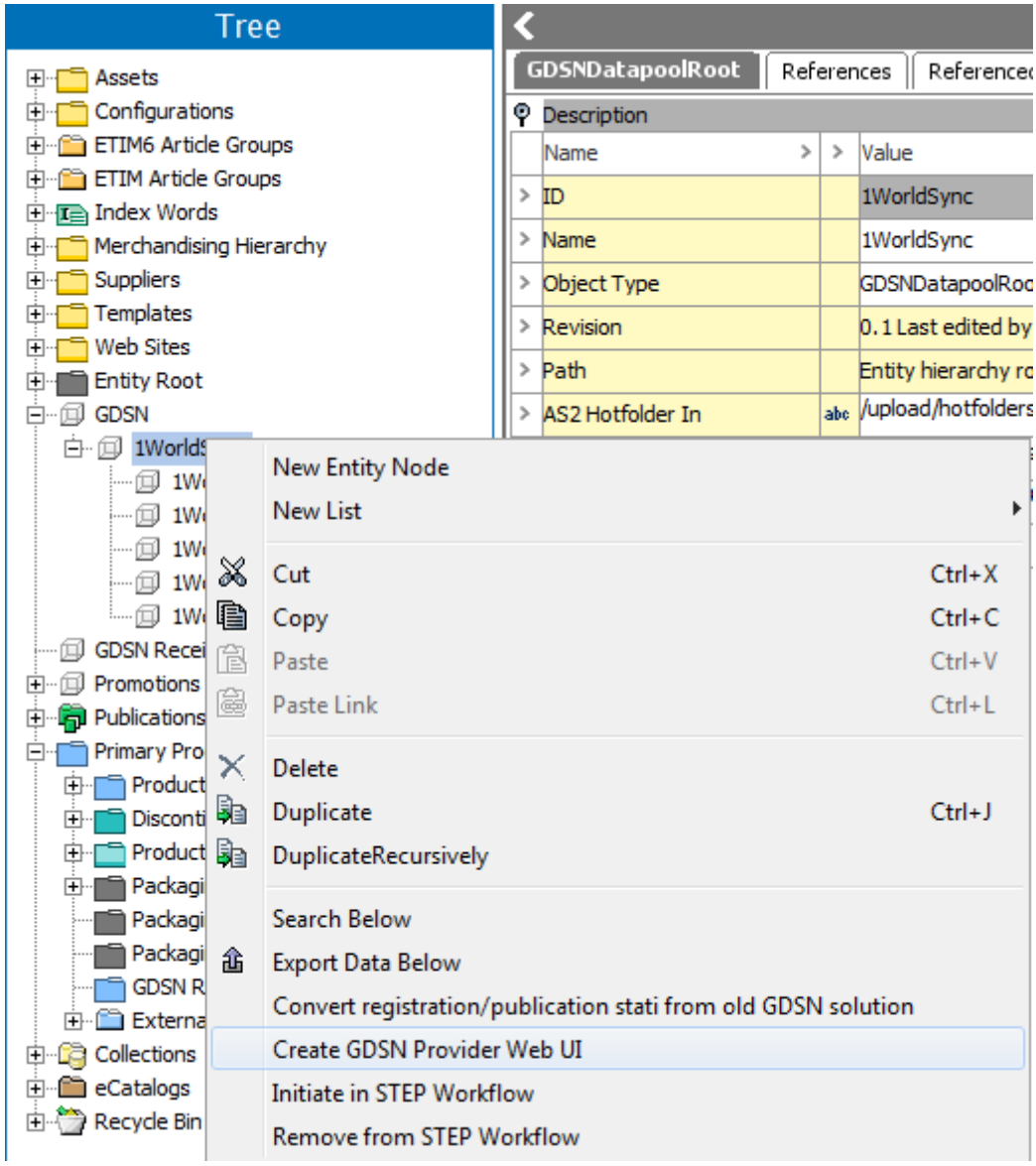
Details about configuring the GDSN buttons are included in these sections:

- GDSN Register Action Button
- GDSN Publish Action Button
- GDSN Hierarchy Withdrawal Action Button

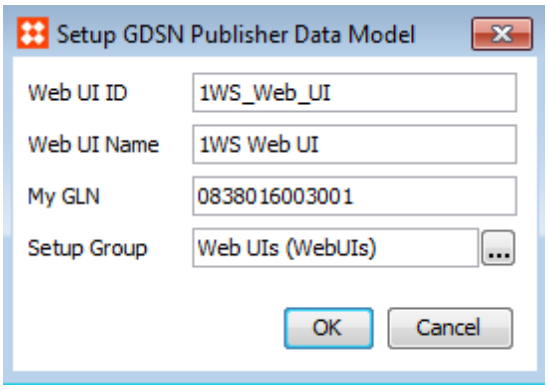
Create a GDSN Web UI

Use STEP Workbench to create a Web UI that is based on the 'User Web UI'. The Web UI illustrates how you can create a GDSN Web UI, but it is not production-ready. Additional customization is required to address your company's needs.

1. In the Tree expand GDSN and right-click the relevant data pool entity.
2. Click **Create GDSN Provider Web UI**.



3. In the **Setup GDSN Publisher Data Model** dialog, specify the following:



- **Web UI ID:** The ID of the Web UI. Is also used as part of the URL for the Web UI. The Web UI ID must not contain spaces.
- **Web UI Name:** The name of the Web UI.
- **My GLN:** The Global Location Number of your company or the company that is publishing.
- **Setup Group:** Select the setup group where the Web UI configuration is stored. The setup group field is populated with any setup group that allows Web UIs. If there are no setup groups that allow the creation of a Web UI, a warning is displayed before the dialog is displayed.

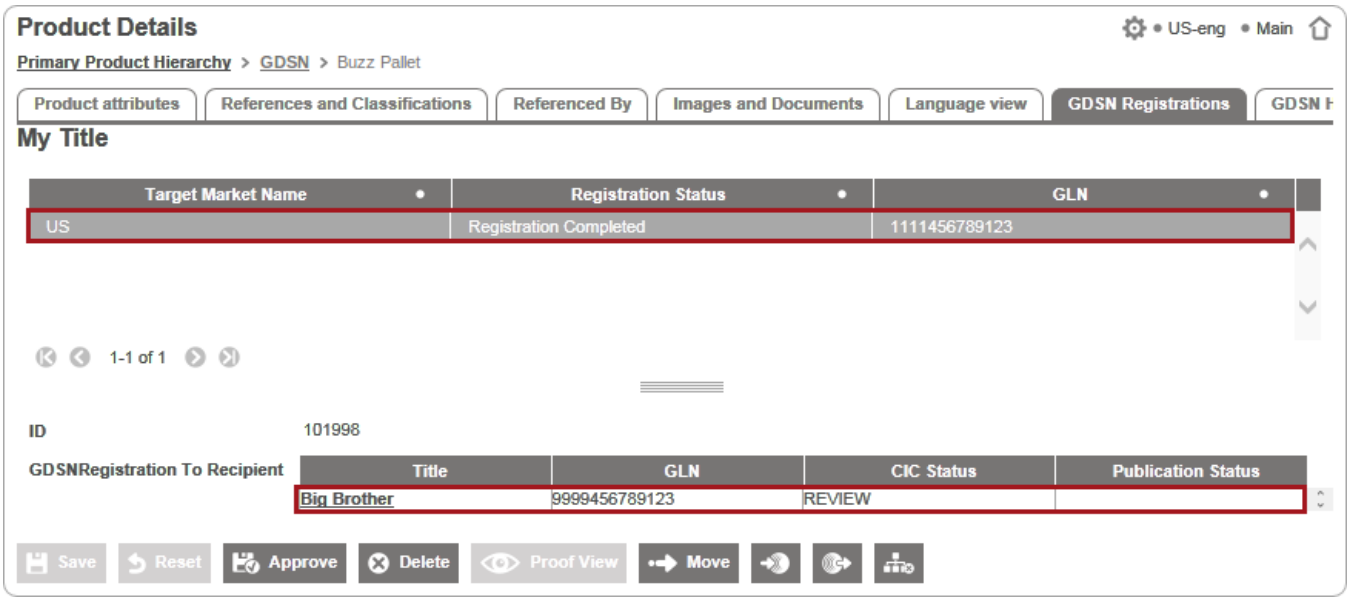
GDSN Registration Tab

The GDSN registration tab contains a master table at the top that shows information about registrations. The table lists all the GDSN registrations made for the selected product.

For each registration the following information is displayed:

- **Target Market:** Lists the target market the registration is made for.
- **Registration Status:** Lists the status of the registration. For example, Registration Completed.
- **GLN:** Lists the Global Location Number of your company or the company you are publishing for.

Once a product is registered with the GDSN it can be published to multiple GDSN data pool recipients.



Select a registration in the master table, and the details area shows a list of publications registered in the target market of the selected registration.

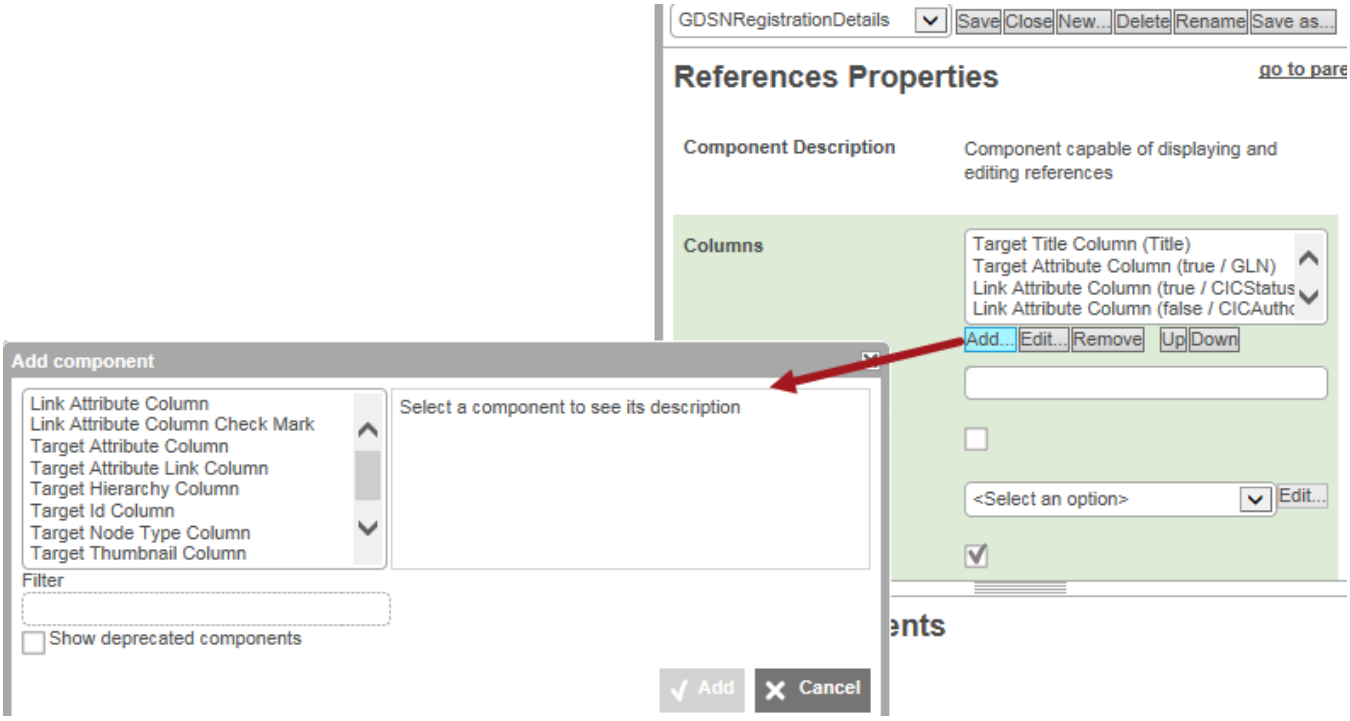
For each publication the following information is displayed:

- **Title:** Lists the name of the recipient.
- **GLN:** Lists the Global Location Number of the recipient.
- **GDSN CIC Status:** Lists information about whether the recipient has accepted the published data.
- **GDSN publication status:** Lists whether the recipient has received the product.

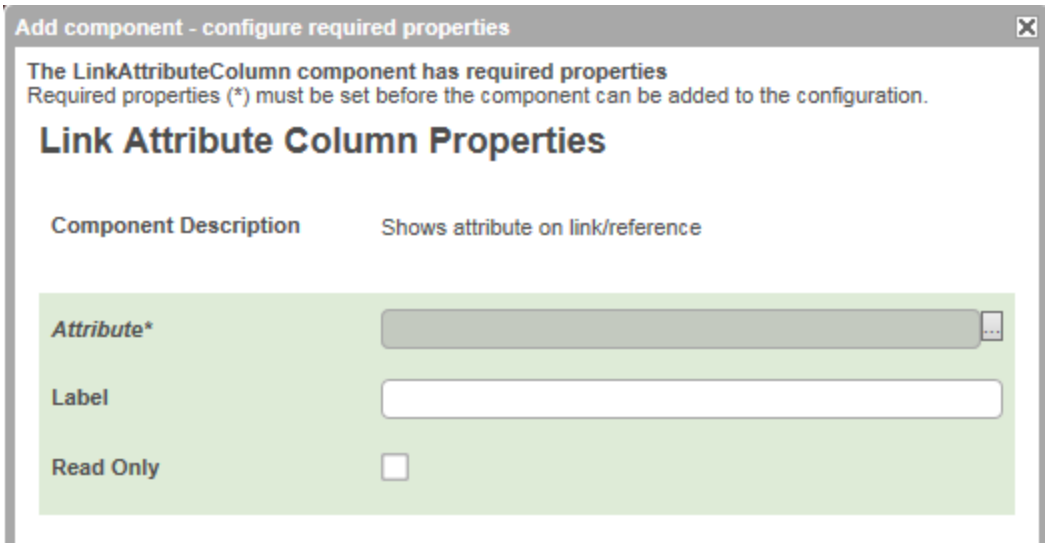
Display an attribute value in the publication table

Additional columns can be configured to both the master table (registrations) and the details table (publications). This could be a column for the publication table that displays additional comments for the CIC999 code.

1. In the Web UI, enter Design mode.
2. Select the screen **GDSNRegistrationDetails**.
3. In the Designer, navigate to the **References** component placed within the Node Editor.
4. Click the Add button for the Columns property and the **Add component** dialog displays.



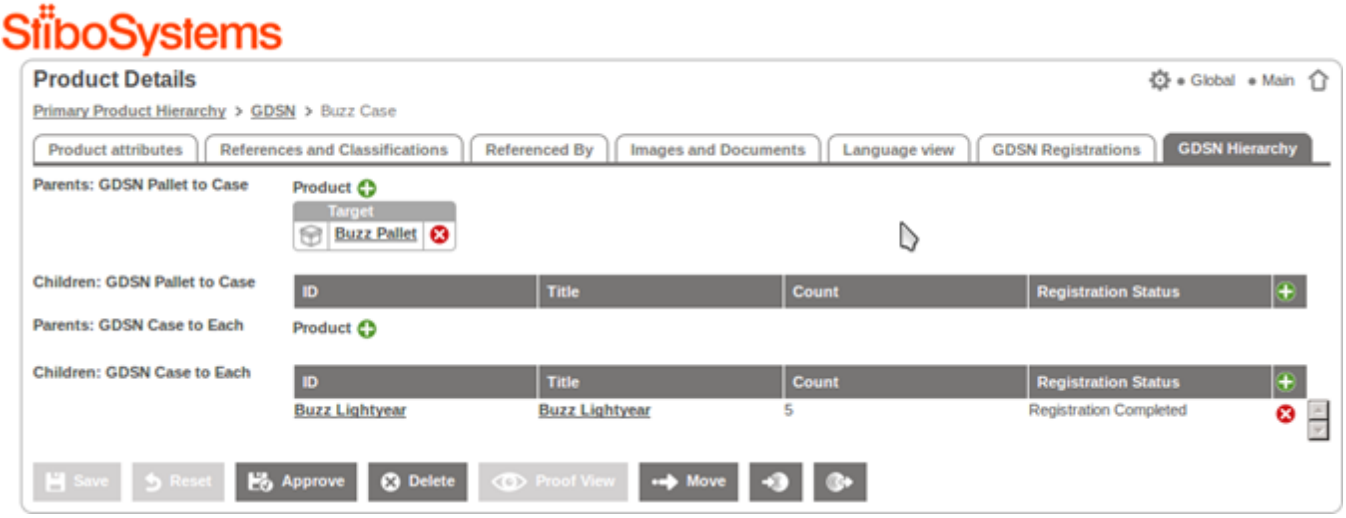
5. Select the **Link Attribute Column** component in the list, click the **Add** button, and a configuration dialog appears.



6. Specify the attribute for the new column and click the **Add** button. A label is optional. If not specified, the column title will be the same as the attribute's name.
7. In the **Add component** dialog, click the **Add** button.
8. Click **Save** for the configuration.
9. Click **Close** for the Designer.

GDSN Hierarchy Tab

The GDSN Hierarchy tab displays information about the package hierarchy. It shows two tables for each GDSN package hierarchy link type defined for the GDSN model. The first table shows the relation to the parent product in the GDSN package hierarchy. The second table shows information about the children in the GDSN package hierarchy.

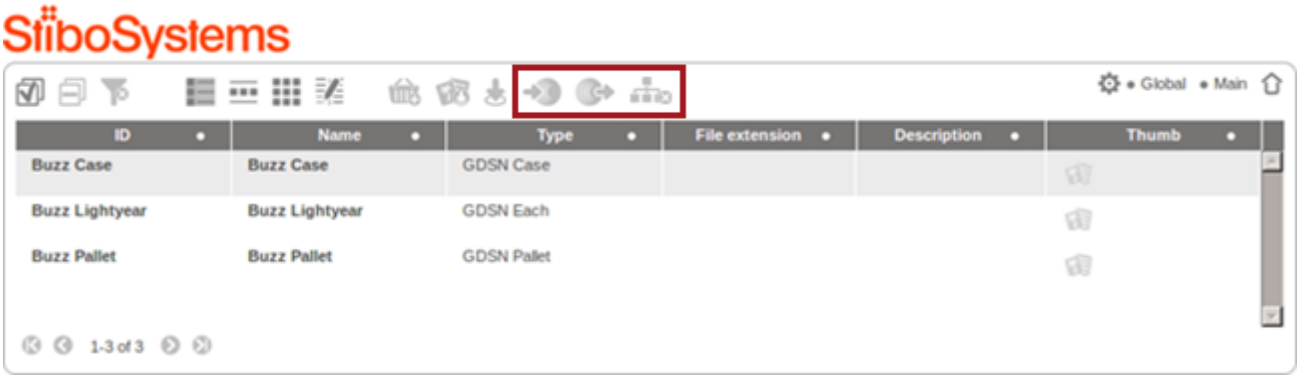


For each link to a child GDSN product, the following information is displayed:

- **ID:** Lists the ID of the child GDSN product.
- **Title:** Lists the title of the child GDSN products
- **Count:** Lists the amount of child products that this product contains.
- **Registration Status:** lists whether the package hierarchy link has been registered with the data pool or not.

GDSN Web UI Buttons

The following button components are available for use in a GDSN Web UI:



- The Register button, , button registers the selected products with a GDSN datapool. For configuration details, see the **GDSN Register Action Button** section of the **GDSN Provider** documentation.
- The Publish button , , publishes the selected products to a GDSN recipient. For configuration details, see the **GDSN Publish Action Button** section of the **GDSN Provider** documentation.
- The Hierarchy Withdrawal button, , sends a hierarchy withdrawal command for the selected packaging hierarchy. For configuration details, see the **GDSN Hierarchy Withdrawal Action Button** section of the **GDSN Provider** documentation.

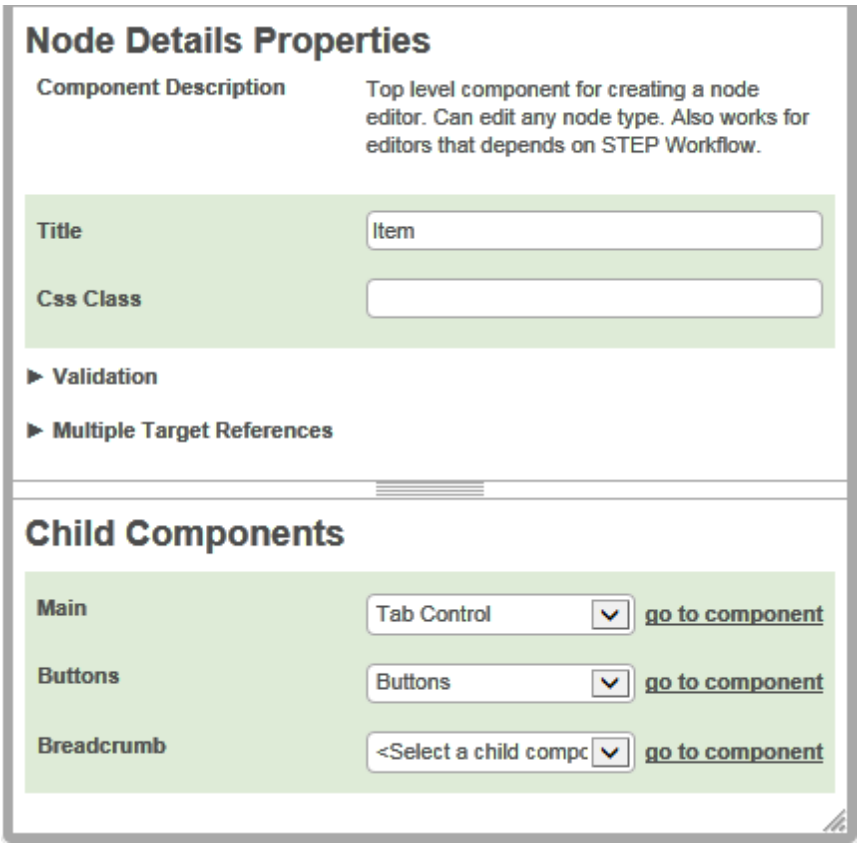
The GDSN Action buttons can be used on both NodeDetail and NodeList screens in either the Toolbar or ButtonPanel. The same set of configuration properties exist regardless of the location of the buttons.

The buttons are pre-configured in the following locations:

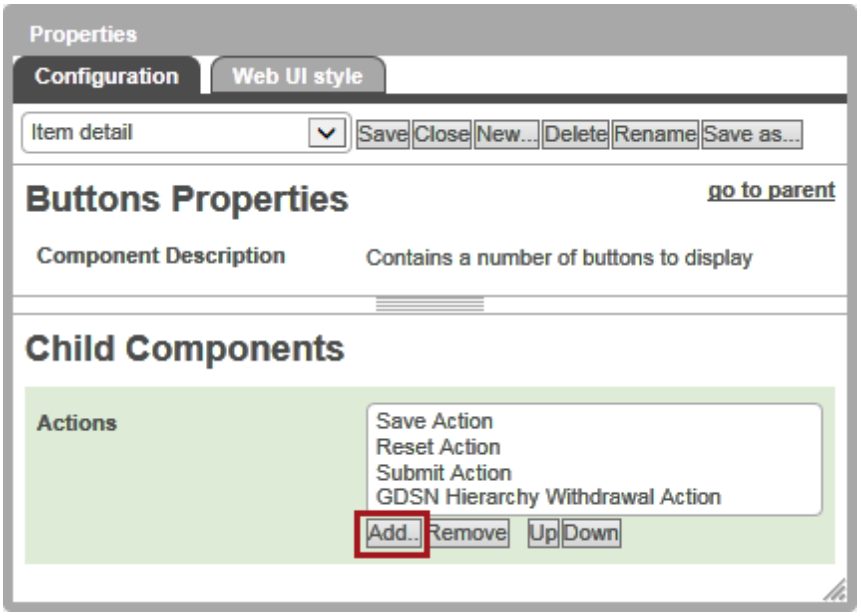
- Generic Search result screen
- Power Search result screen
- Product Details screen

Adding a GDSN button to a Node Details page

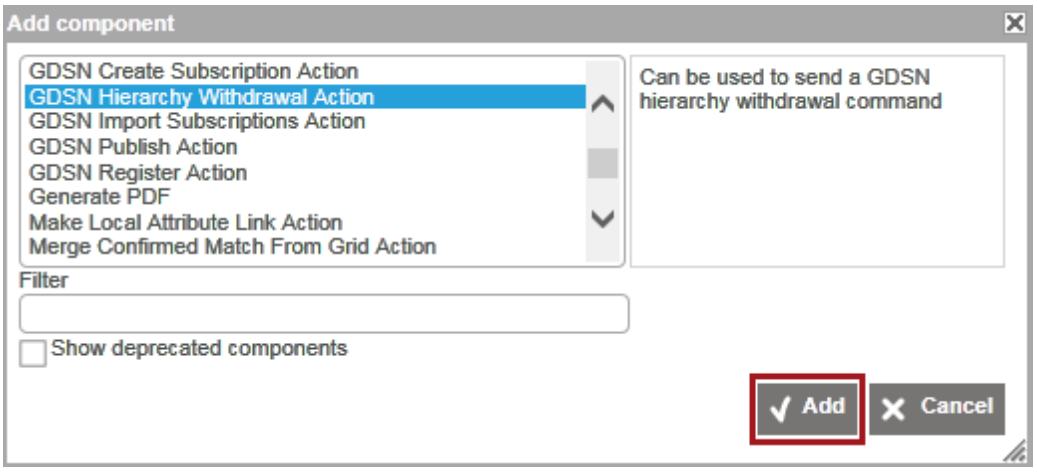
1. Open Designer for a Node Details page. The Node Details Properties configuration dialog is displayed.
2. In the Child Components section, click the **go to component** link for the Buttons field.



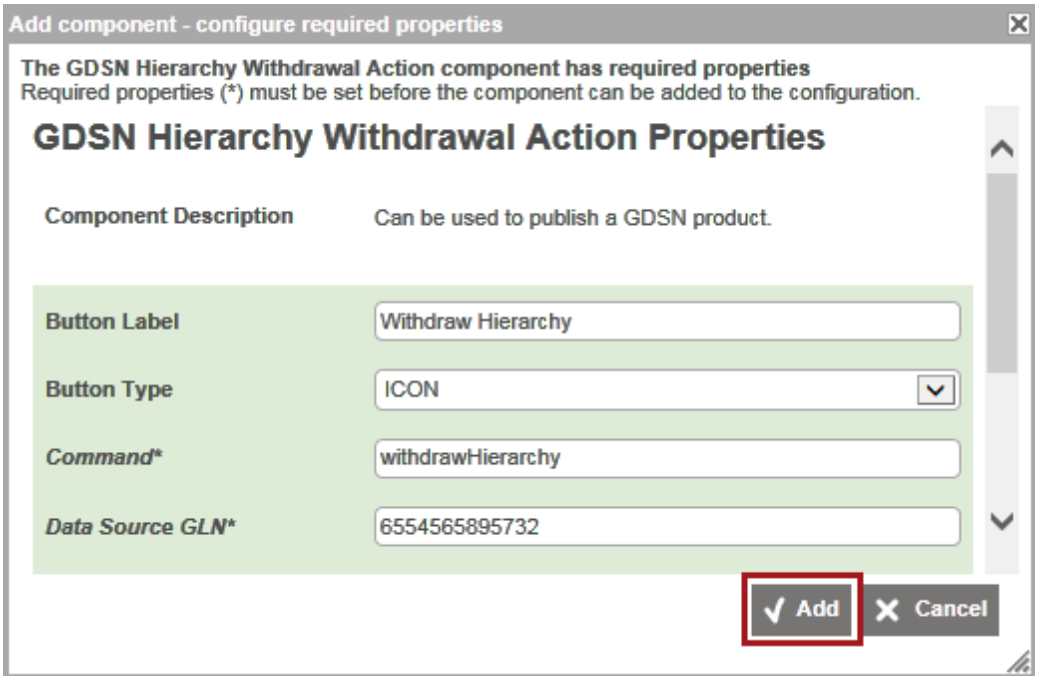
3. On the Buttons Properties configuration dialog, in the Child Components section, click the **Add** button for the Actions field.



4. On the Add component dialog, select a GDSN component and click the **Add** button.



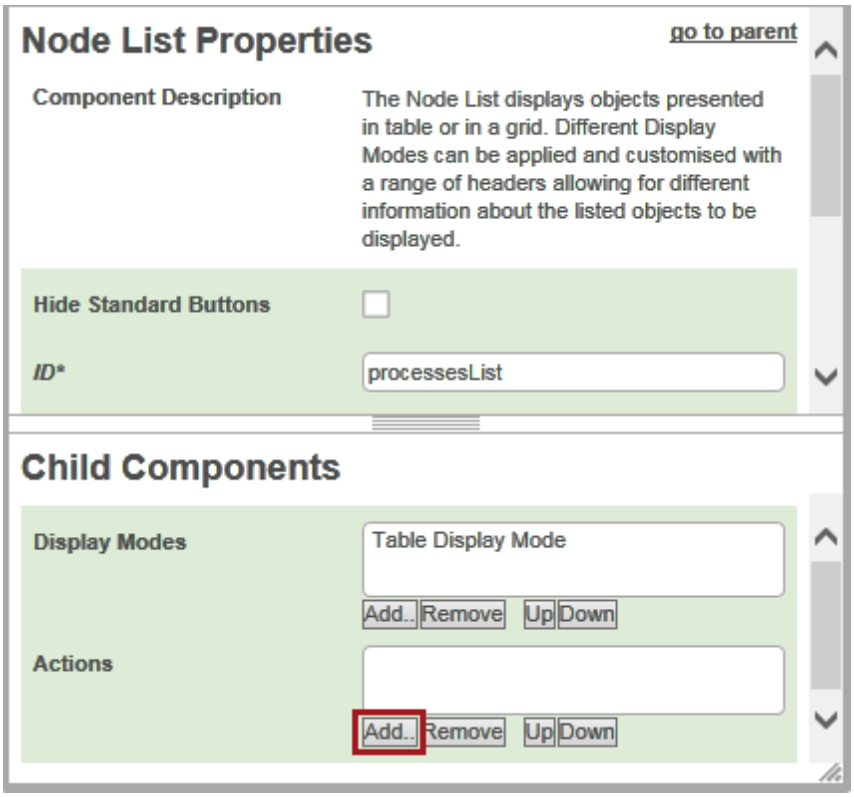
5. On the Add component - configure required properties dialog, add all necessary data and click the **Add** button.



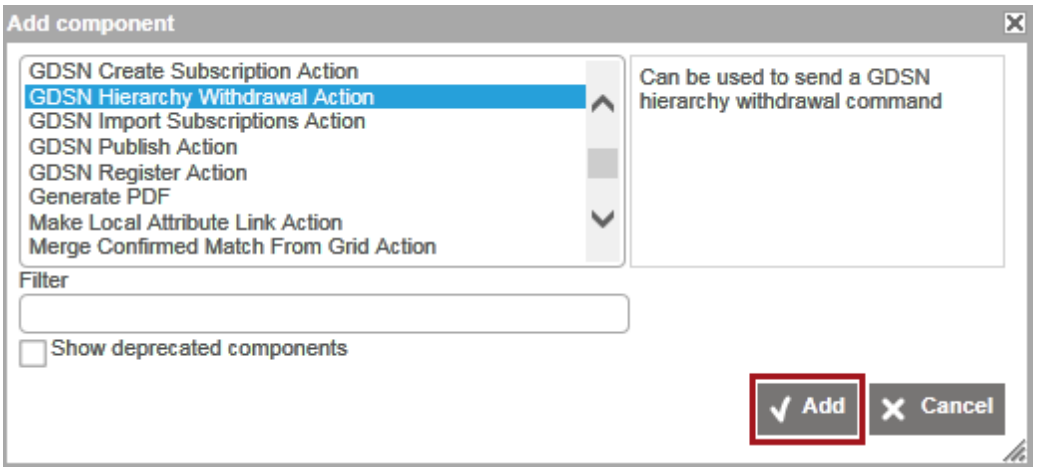
6. The button displays on the Node Details page Button Panel.

Adding a GDSN button to a Node List page

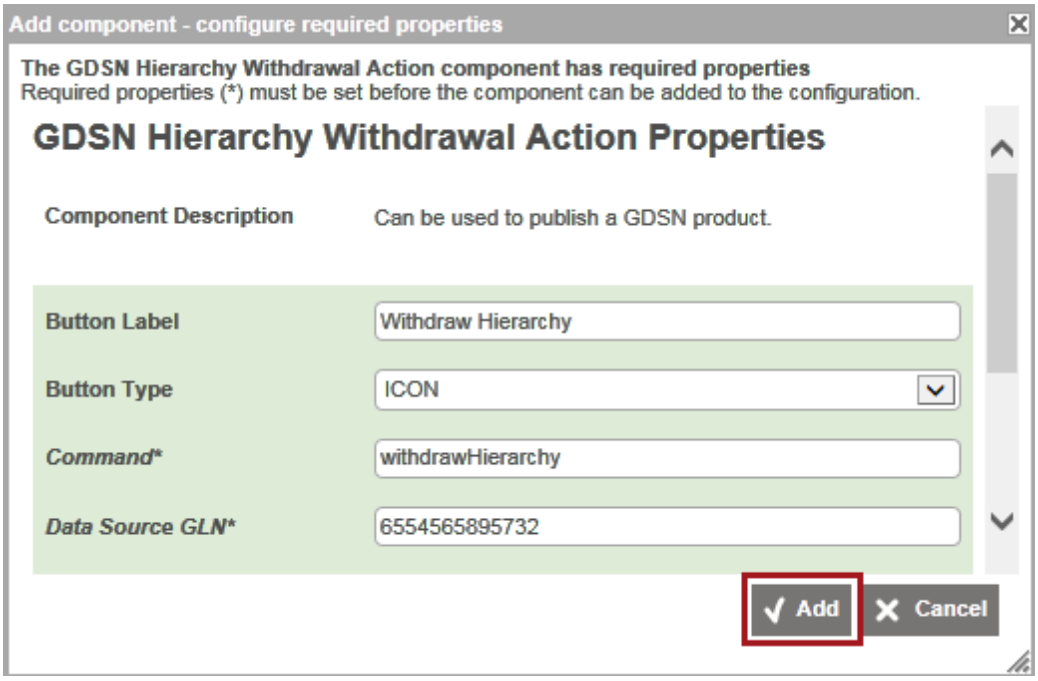
1. Open Designer for a Node List page. The Node List Properties configuration dialog is displayed.
2. In the Child Components section, click the **Add** button for the Actions field.



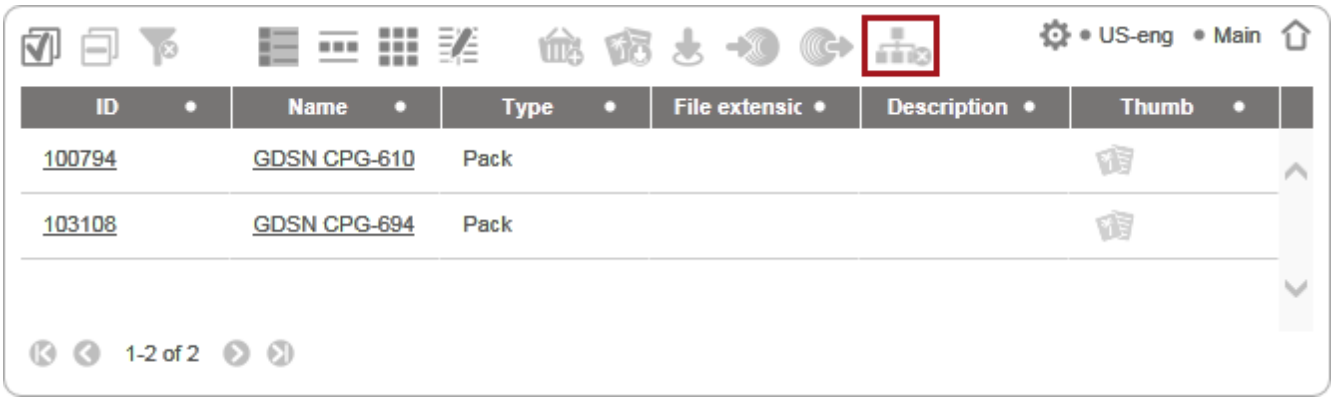
3. In the Add component dialog, select a GDSN component and click the **Add** button.



4. In the Add component - configure required properties dialog, add all necessary data and click the **Add** button.

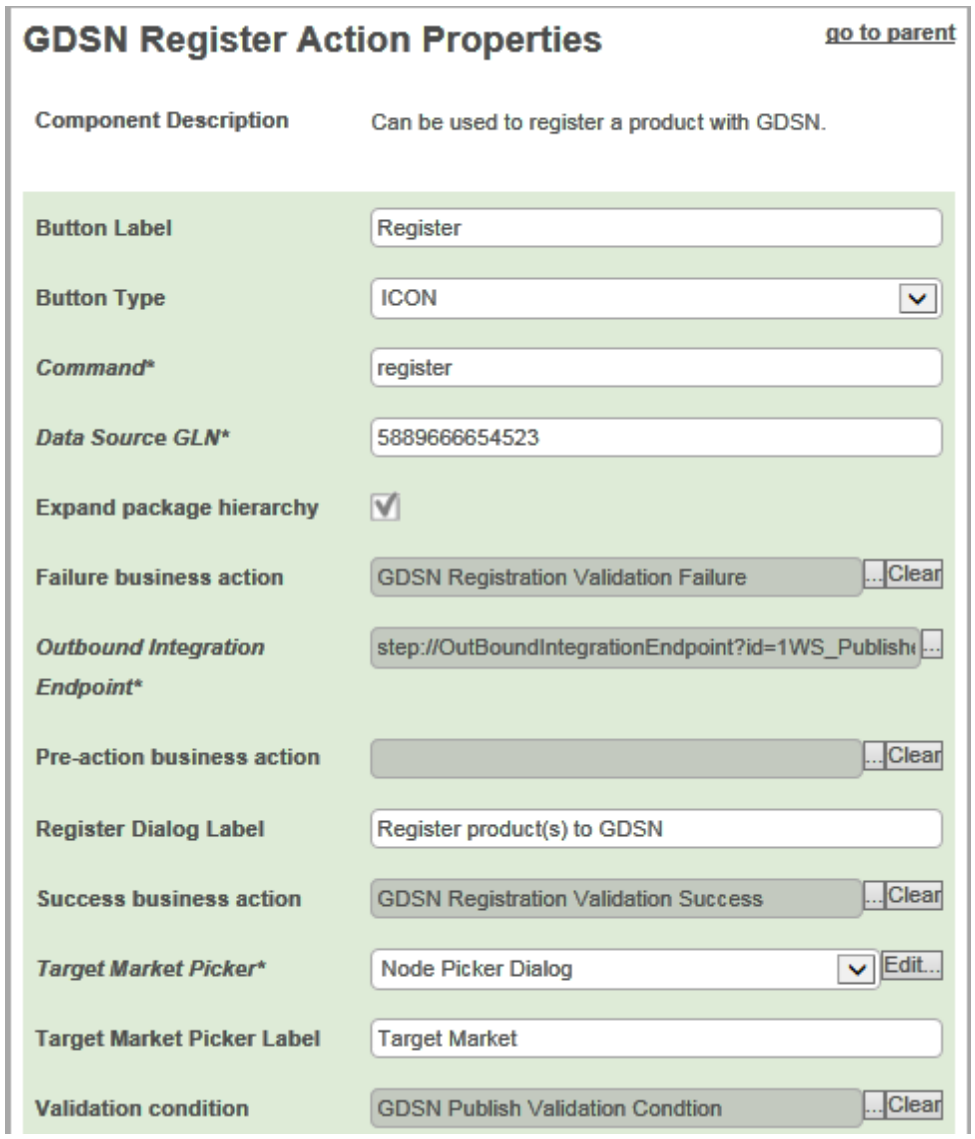


5. The button displays on the Node List page Toolbar.


US-eng
Main
ID
Name
Type
File extensic
Description
Thumb
100794
GDSN CPG-610
Pack
103108
GDSN CPG-694
Pack
1-2 of 2

GDSN Register Action Button

In the Designer, the Web UI component for the Register button is called 'GDSN Register Action'. Each parameter is described in the following table.



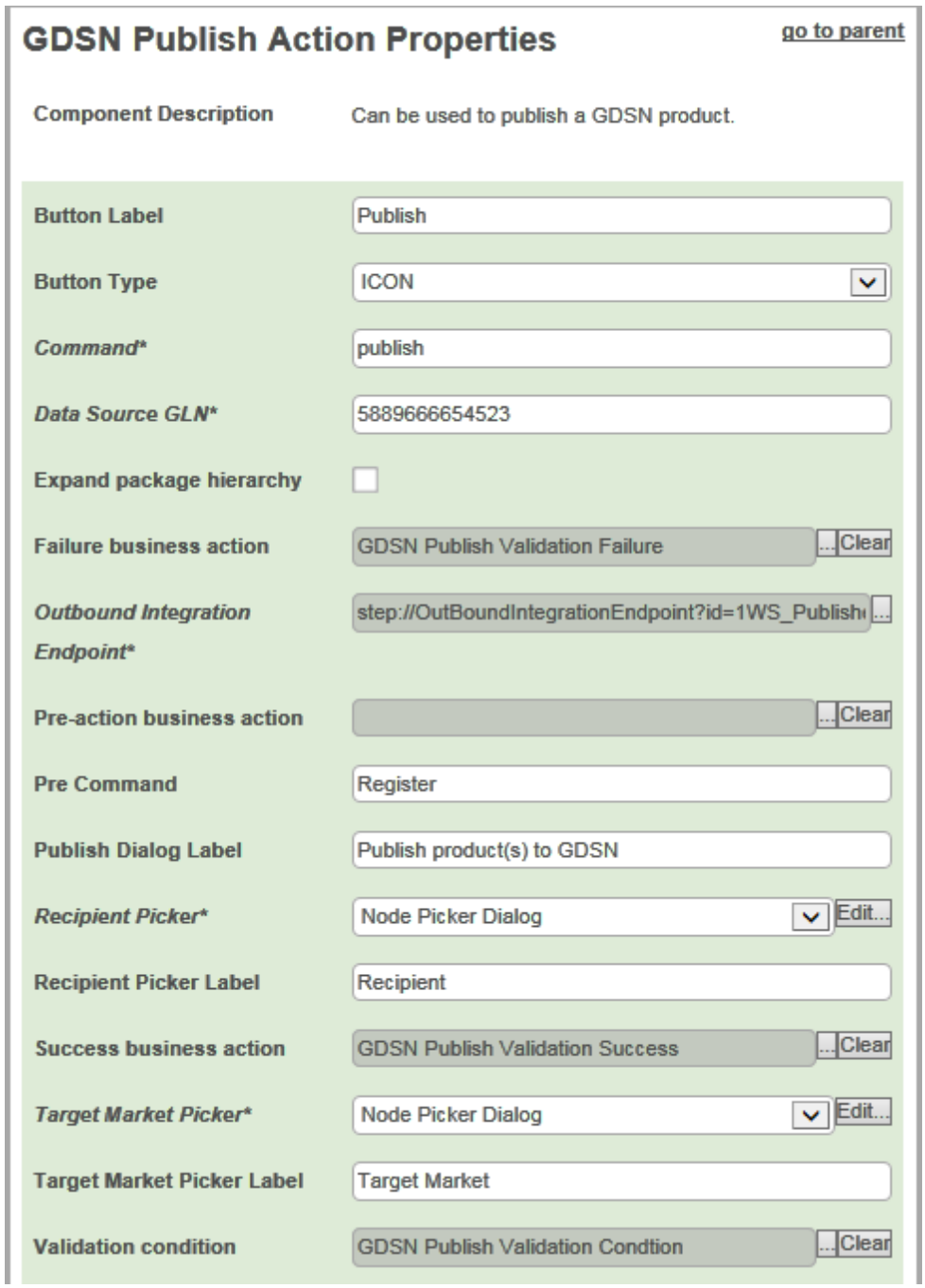
Parameter	Description
Button	Optional

Parameter	Description
Label	Add a label for the Register button, displayed when the Button Type is Text or Both
Button Type	Optional Specify if the button should appear with an icon, text (the button label) or both.
Command	Mandatory Specify which GDSN command should be executed when a user clicks the Register button. In the default GDSN configuration, the command is 'register'.
Data Source GLN	Mandatory Enter your Global Location Number (GLN). This is the GLN that the registrations are sent from.
Expand package hierarchy	When checked, the registration also includes lower level objects in the packaging hierarchy. When unchecked, the registration will only include the selected product.
Failure business action	Optional Specify which business action to run if the Validation condition fails.
Outbound Integration Endpoint	Mandatory Specify the Outbound Integration Endpoint that should be used to send the command to the data pool.
Pre-action business action	Optional Specify the Business Action that runs first (before any other Business Rules) when clicking the Register button. This could be used to determine which GDSN command should be sent (Change_By_Refresh or Modify).
Register Dialog Label	Optional Apply the label of the Register dialog that appears when clicking the Register button.
Success business action	Optional Specify which business action to run if the Validation condition is successful.

Parameter	Description
Target Market Picker	Mandatory Configure the pop-up dialog that enables the end user to pick the target markets for the registration.
Target Market Picker Label	Mandatory Apply the label of the pop-up dialog used by end users to pick the target markets for the registration.
Validation condition	Optional Specify which business condition should be validated before the product is processed further. Depending on the outcome of the validation, the 'Success business action' or the 'Failure business action' is called. If no condition is configured, no validation is executed and the Success / Failure business actions are not used. If a condition is configured, a Preflight option is presented in the Register dialog, enabling the user to run the Validation conditions and the Success / Failure business actions without actually registering to the GDSN.

GDSN Publish Action Button

In the Designer, the Web UI component for the Register button is called 'GDSN Register Action'. Each parameter is described in the following table.

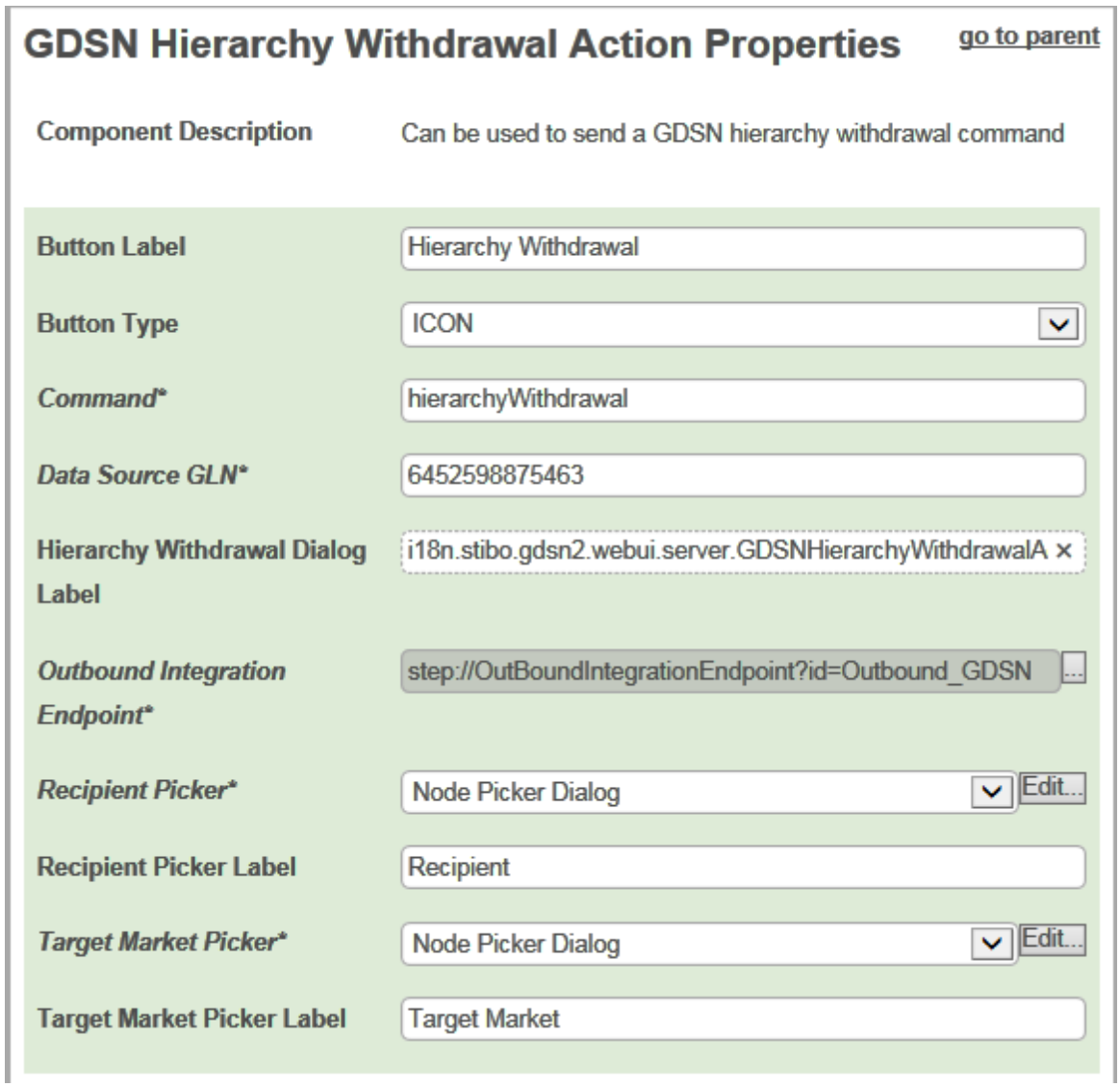


Parameter	Description
Button Label	Optional Add a label for the Publish button, displayed when the Button Type is Text or Both
Button Type	Optional Specify if the button should appear with an icon, text (the button label) or both.
Command	Mandatory Specify which GDSN command that should be executed when a user clicks the Publish button. In the default GDSN configuration, the command is 'publish'.
Data Source GLN	Mandatory Enter your Global Location Number (GLN). This is the GLN that the publications are sent from.
Expand package hierarchy	When checked, the publication also includes lower level objects in the packaging hierarchy. When unchecked, the publication will only include the selected product.
Failure business action	Optional Specify which business action to run if the Validation condition fails.
Outbound Integration Endpoint	Mandatory Specify the Outbound Integration Endpoint that should be used to send the command to the data pool.
Pre-action business action	Optional Specify the Business Action that runs first (before any other Business Rules) when clicking the Publish button. This could be used to determine which GDSN command should be sent (Change_By_Refresh or Modify).
Pre Command	Optional Specify the GDSN command that runs prior to the publishing of the product. This could be a register command that is configured to send an update document type to GDSN in order to ensure that the current attribute values in STEP are synched with the values that are stored in GDSN.

Parameter	Description
Publish Dialog Label	Optional Apply the label of the publish dialog that appears when clicking the Publish button.
Recipient Picker	Mandatory Configure the pop-up dialog that enables the end user to pick recipients for the publication.
Recipient Picker Label	Mandatory Apply the label of the pop-up dialog used by end users to pick recipients for the publication.
Success business action	Optional Specify which business action to run if the Validation condition is successful.
Target Market Picker	Mandatory Configure the pop-up dialog that enables the end user to pick the target markets for the publication.
Target Market Picker Label	Mandatory Apply the label of the pop-up dialog used by end users to pick the target markets for the publication.
Validation condition	Optional Specify which business condition is validated before the product is processed further. Depending on the outcome of the validation, the 'Success business action' or the 'Failure business action' is called. If no condition is configured, no validation is executed and the Success / Failure business actions are not used. If a condition is configured, a Preflight option is presented in the Publish dialog, enabling the user to run the Validation conditions and the Success / Failure business actions without actually publishing to the GDSN.

GDSN Hierarchy Withdrawal Action Button

In the Designer, the Web UI component for the Hierarchy Withdrawal button is called GDSN Hierarchy Withdrawal Action. Each parameter is described in the following table.



Parameter	Description
Button Label	Optional Add a label for the Hierarchy Withdrawal button, displayed when the Button Type is Text or

Parameter	Description
	Both
Button Type	Optional Specify if the button should appear with an icon, text (the button label) or both.
Command	Mandatory Specify which GDSN command that should be executed when a user clicks the Hierarchy Withdrawal button. In the default GDSN configuration, the command is 'hierarchyWithdrawal'.
Data Source GLN	Mandatory Enter your Global Location Number (GLN). This is the GLN that the publications are sent from.
Outbound Integration Endpoint	Mandatory Specify the Outbound Integration Endpoint that should be used to send the command to the data pool.
Hierarchy Withdrawal Dialog Label	Optional Add the Hierarchy Withdrawal Dialog Label to the dialog that appears when clicking the Hierarchy Withdrawal button.
Recipient Picker	Mandatory Configure the pop-up dialog that enables the end user to pick recipients for the hierarchy withdrawal.
Recipient Picker Label	Mandatory Apply the label of the pop-up dialog used by end users to pick recipients for the hierarchy withdrawal.
Target Market Picker	Mandatory Configure the pop-up dialog that enables the end user to pick target markets for the hierarchy withdrawal.

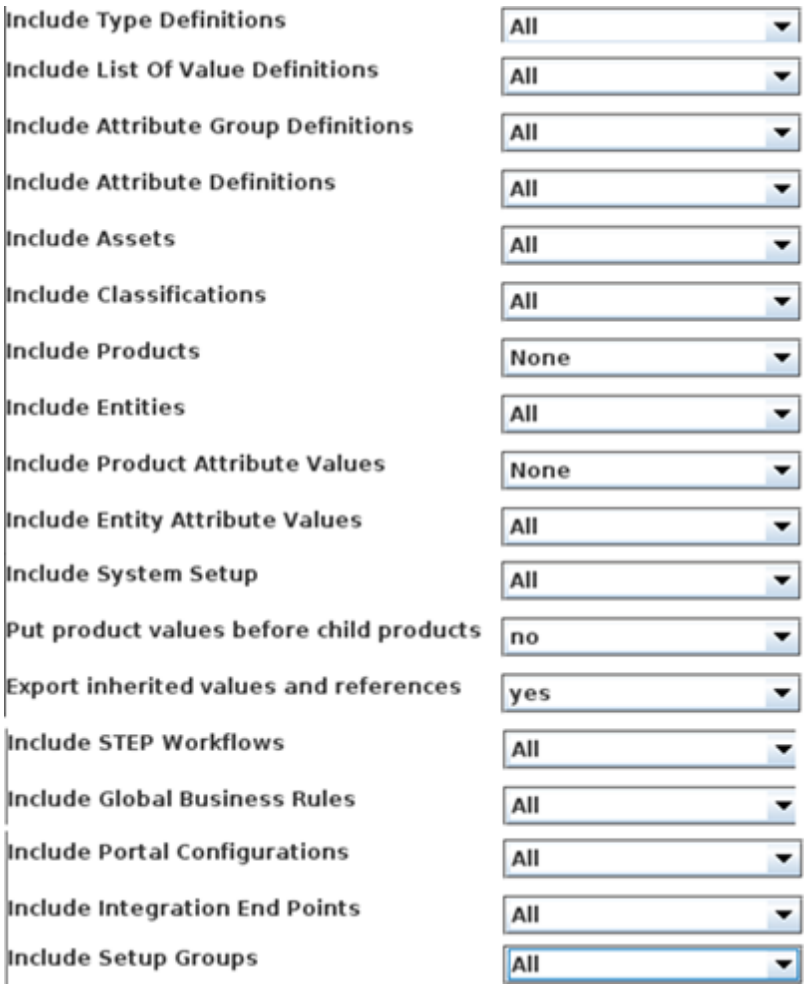
Parameter	Description
Target Market Picker Label	Mandatory Apply the label of the pop-up dialog used by end users to pick target markets for the hierarchy withdrawal.

Moving a GDSN Setup to a New System

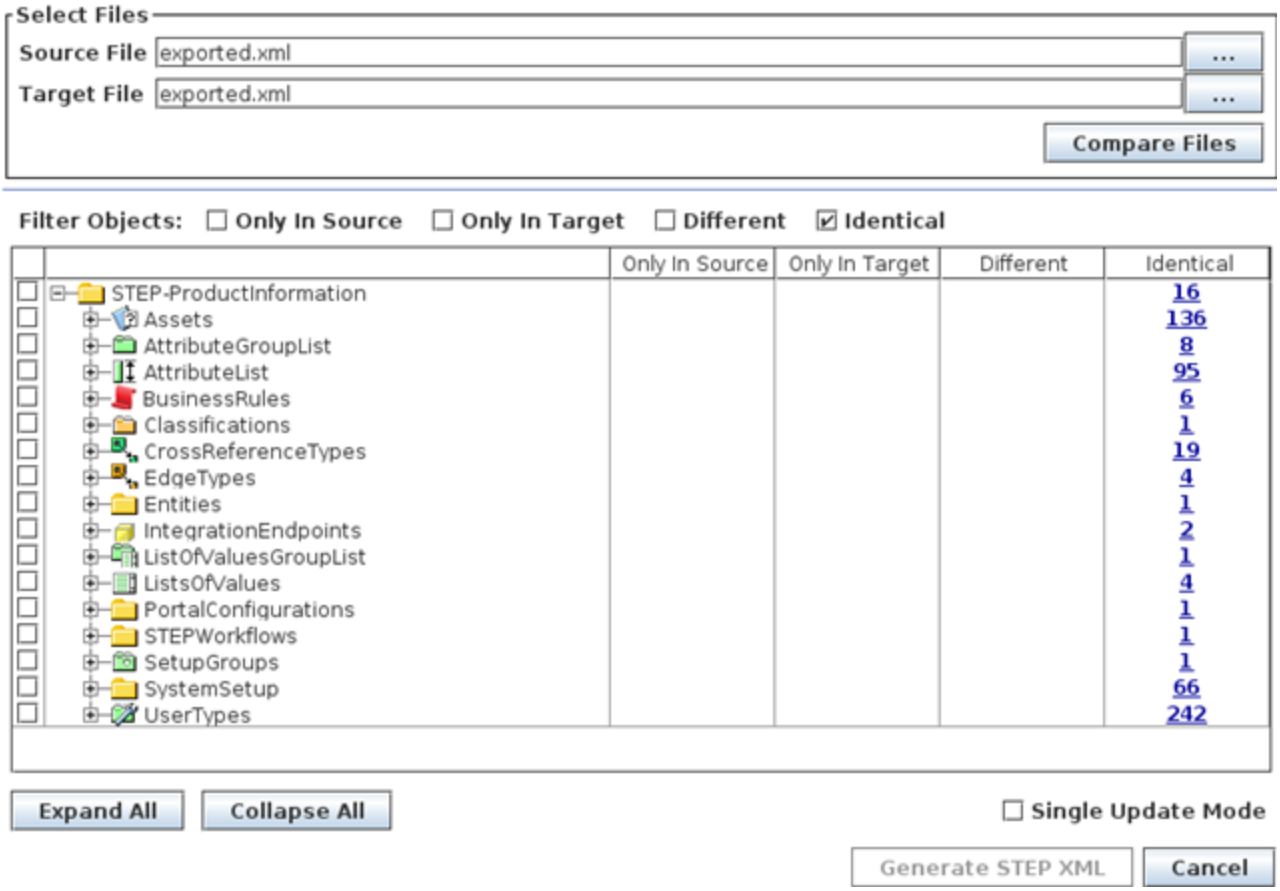
For both **GDSN Receiver** and **GDSN Provider**, the following describes how to move a system that has been set up with the easy setup wizard. Although this is the DTAP-recommended approach (Development-Test-Acceptance-Production), the scenario should be modified to meet your company requirements.

To move a GDSN setup from one system to another, use the Export Manager's STEPXML format and the Export Comparison tool. For information about exporting assets and data, see Exporting Images and Documents and Exporting Data and Images.

1. In the **Tree**, locate the XSD asset that has been uploaded for GDSN.
2. Right-click the XSD asset and click **Export Images & Documents**.
3. In the Tree, right-click the relevant data pool and click **Export data below** to export the remaining system setup. The export must not include the component model. This image shows the relevant STEPXML options.



4. From the **File** menu, point to **Export** and click **Compare System Setup Exports**. Remove any exported elements that should not be imported into the new system.



5. For **Source File** and **Target File**, locate the XML file exported and check the Filter Objects **Identical** option.
6. Select the elements needed for the import and click **Generate STEP XML**. A new export file is created that can be imported.
7. Import the export file into the new system.

Note: For **GDSN Receiver** solutions only, when the file is imported, three errors are generated in the BGP Execution Report since references point to objects that do not yet exist. The Data Pool Entity references the product folder 'GDSN IMPORT ROOT' and the assets 'GDSN Receiver DatapoolCINSample' and 'GDSN Receiver DatapoolXSD'. This is expected and does not affect the system import.

---

8. Import the XSD asset on top of the one that is already in the system to get the content. If the object type of the XSD asset is Zip file, change it to the correct file type as specified in the component model.
9. Manually set the references on the **Data Pool Format Entity** references tab.
10. Create a Import Root product folder on the target system.
11. Set the reference to the Import Root product folder on the **Data Pool Entity** references tab.
12. Locate the component model, and specify the correct object types, attributes and references.

GDSN Conversion

If you are using the GDSN solution available prior to STEP Trailblazer 7.1 and want to upgrade to the current STEP GDSN solution, you can use the GDSN conversion script. Before you run the script, do the following:

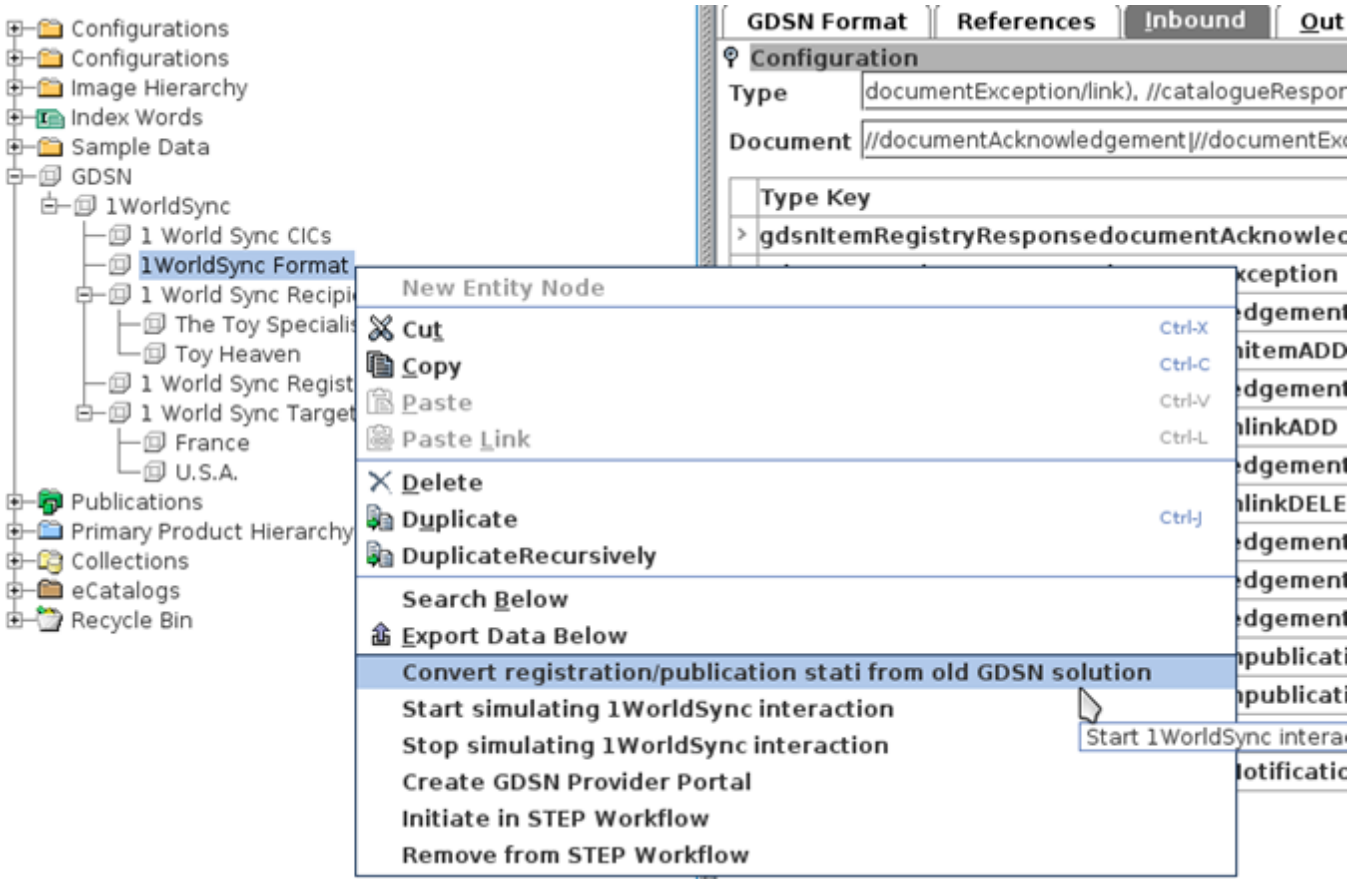
1. Set up the new GDSN Component model. For more information, Setting Up the GDSN Component Model.
2. Set up provider data pools that match the provider data pools in the old GDSN solution.
3. For the new data pools, create recipients that match the old GDSN recipients.
4. Create Target Markets for the provider data pools.

The conversion script only needs to read the GLN of the provider data pool to find the corresponding data pool in the old solution. The data pool does not need to be fully configured before you run the conversion script. The GDSN Format Inbound and Outbound mappings, for example, do not need to be configured in order for the conversion script to work.

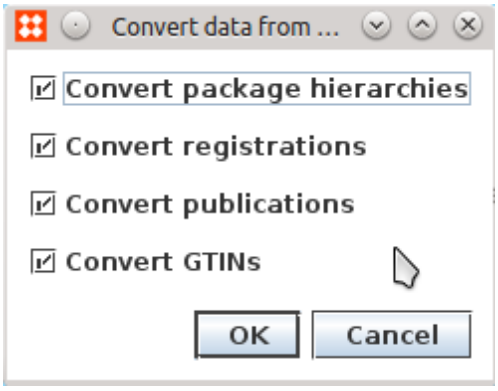
However, you have to create the target markets before you run the conversion script because the new registrations must link to these registrations. You also have to create the new recipients before you run the conversion script because the publication in the new model needs to create a link to the recipients.

To start the GDSN conversion script

1. In the **Tree**, expand **GDSN**, right-click the relevant GDSN data pool and select **Convert registration/publication stati from old GDSN solution**.



2. In the **Convert data from** dialog, specify which data you want to convert from the old GDSN solution. You have the following options:
   - **Convert package hierarchies:** Convert the old GDSN hierarchies into package hierarchies using the new GDSN model.
   - **Convert registrations** Convert the status of old GDSN registrations to match the new GDSN solution.
   - **Convert publications:** Convert the status of old GDSN publications to match the new GDSN solution.
   - **Convert GTINs:** Convert the GTIN of the old GDSN products to the GTIN attribute of the new GDSN Component model.

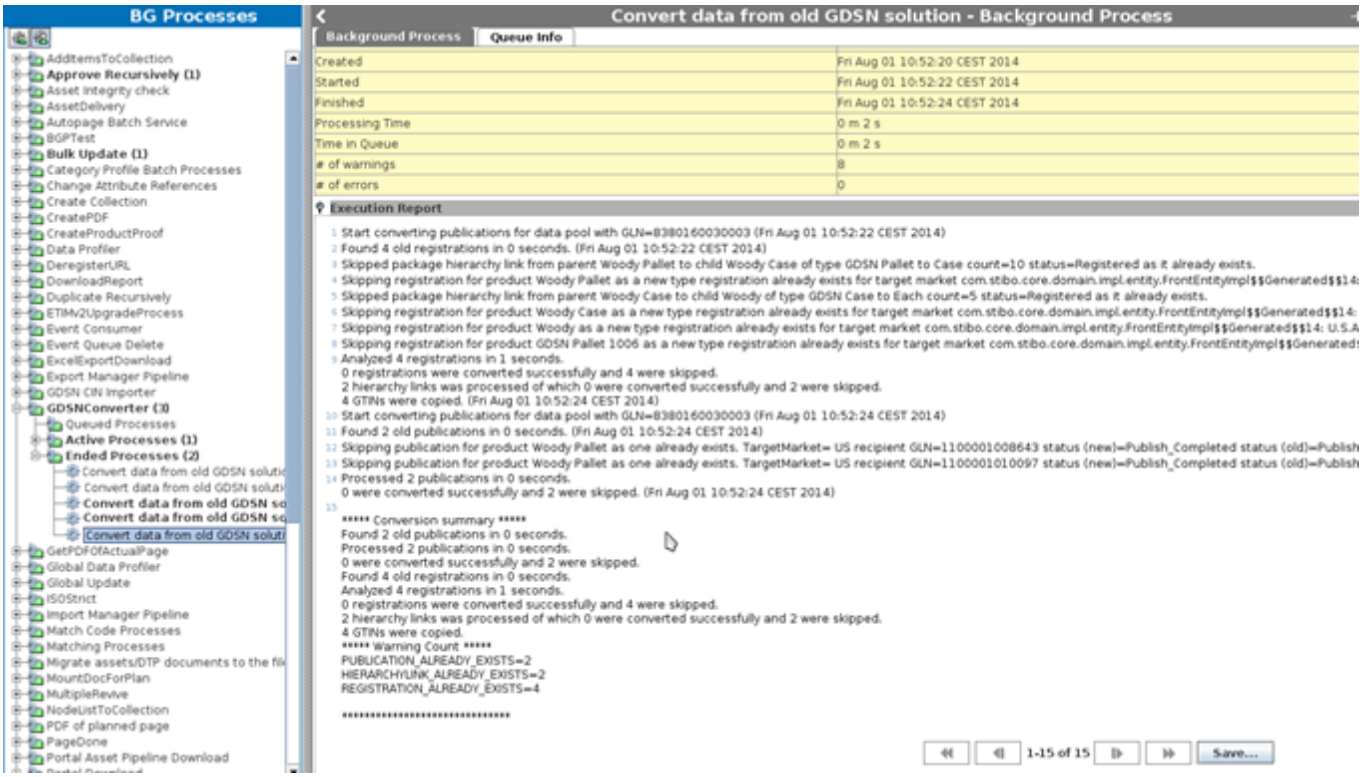


The GDSN conversion script uses the selected data pool to find the corresponding data pool in the old GDSN model. The script finds all old registrations, package hierarchies, and publications for the selected data pool and starts converting them into the new GDSN model.

In the old GDSN solution, the GTIN that a product is registered with is written in a hidden attribute. When you select **Convert GTINs**, this value is copied to the GTIN attribute you have selected for the new GDSN component model. This option is best used when the GTIN attribute used by the new GDSN Component model is not the same as the one that was used when the products were registered. You can also select this option to ensure that the GTIN that is used is the GTIN the product was registered with in case it has been changed by accident.

The conversion is runs as a background process because the conversion takes time if there are many GDSN products for the selected data pool. The conversion script continues to run even if it encounters errors. Information about errors is shown in the execution report of the background process. The execution report also provides information about the progress of the conversion script.

The conversion script is able to handle that data has already been converted. So, if you receive errors, for example, because the GDSN object type hierarchy is not correctly configured, you can restart the conversion script, and then only check the option to convert package hierarchies. When the conversion script encounters data that has already been converted, it skips the conversion of that particular registration, publication, or hierarchy and describes in the execution report that it was skipped.



The same type of error or warning is only output a maximum of 100 times in the running feedback. Apart from the running feedback, there is a Conversion summary at the end of the execution report. The summary details how many registrations and publication that were converted and how long it took. It also provides an overview of the number of errors and warnings that were encountered during the conversion process.