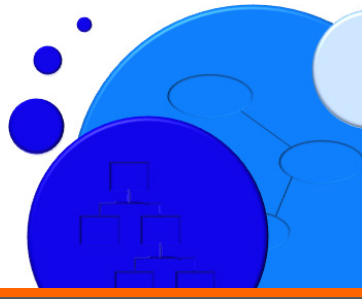
Cradle-7
From concept to creation...

3SL

Reusing Information with Adaptations in Cradle

RA004/02 July 2016



http://www.threesl.com
salesdetails@threesl.com
support@threesl.com

Structured Software Systems Ltd (3SL)
Suite 2, 22a Duke Street
Barrow-in-Furness
Cumbria LA14 1HH, UK
Tel: +44 (0) 1229 838867
Fax: +44 (0) 1229 870096
Regd: 2153654 VAT: GB 473 2757 28

Contents

Introduction 1
Subject 1
Adaptations 1
1 Adapting One Type of Information 2
1.1 Typical Data 2
1.2 Schema Setup 2
1.3 Creating New Items 3
1.4 Use a Library Item 3
1.5 Adapt a Library Item 3
1.6 Browsing 4
2 Adaptations for Two Types of Item 4
2.1 Typical Data 4
2.2 Schema Setup 5
2.3 Alternative Schemas 6
2.4 Creating New Items 6
2.5 Use a Library Item 6
2.6 Adapt a Library Item 7
3 Summary 7

List of Tables

Table 1:Link Rules for Adapting One Item Type 3
Table 2:Link Rules for Adapting Two Item Types 5

Introduction

Most organisations have ***products*** that are created and evolved in ***projects***. Product information is typically managed as a ***library*** of standard items and is often baselined. Each project will add new information and change existing information. In general, such changes should not affect items in the library or the cross references between them to ensure other uses of the library do not become invalid.

Subject

This paper describes how to use Cradle's ***adaptation*** mechanism to implement a library and projects that reference that library in a ***single database***.

It considers two uses of adaptations that meet all the needs of any project:

1. Adaptations of one information type. This arises when a library and project have the same type of information. The project's items are specific to the project, referenced from the library, or are adaptations of the items in the library.
2. Adaptations between two types of information. This case occurs when a project has a set of items that are to be linked to library information of a different type, and may then create adaptations of that library information for its own use.

The paper assumes a basic familiarity with Cradle.

Adaptations

An ***adaptation*** is a copy of an item that also inherits the item's related information. The adaptation is linked to the original item and the inherited items by user-defined types of cross reference. Adaptations are highlighted by user-defined marks in their names and, as are references, by being shown in the UI and reports highlighted in the colour defined for the adaptation and reference link types.

An adaptation is the means to create a project's copy of an item in a library, so the project can change the item for its own purposes.

Adapting One Type of Information 1

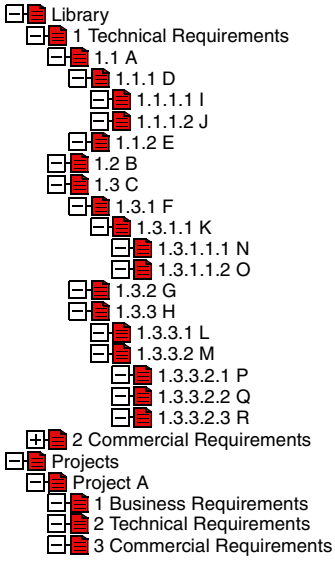
This is the simplest situation. A hierarchy of items exists in the library and the project will have its own hierarchy of the same type of item. The project will:

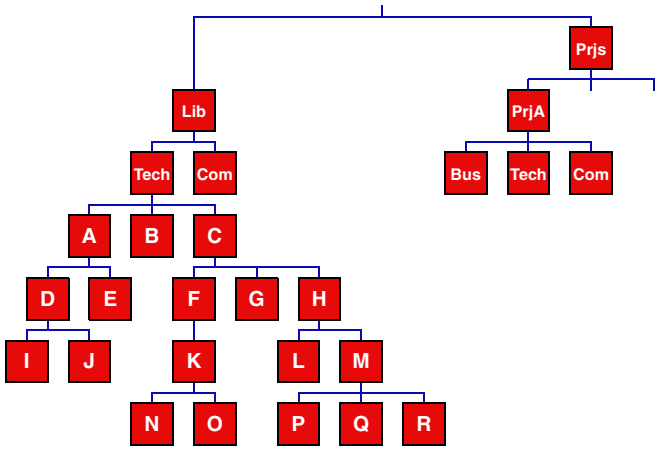
- Create its own items
- Reference items from the library
- Create its own adaptations of library items

The important characteristics of this case is that all items are of the same type.

Typical Data 1.1

A library has requirements as **REQ** items. A project has requirements, also as **REQ** items. The project may reference library requirements or adapt them. All other project information will link to requirements *in the project's own hierarchy*. A typical set of requirements could be:



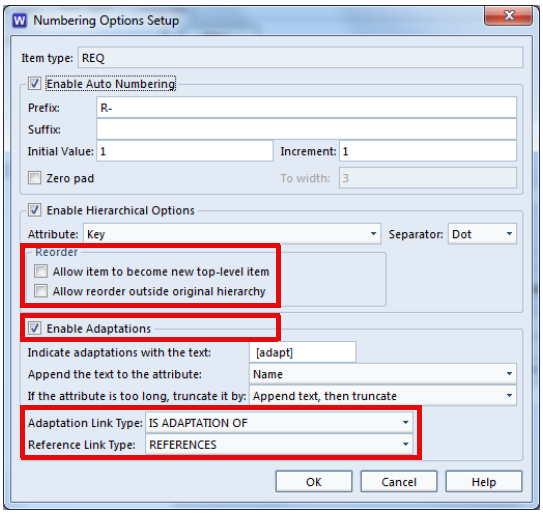


Schema Setup 1.2

Define three link types with names such as:

- **HAS CHILD**, for the hierarchical parent-child cross references
- **REFERENCES**, for references from projects into the library
- **IS ADAPTATION OF**, for adaptation links

and set colours for the references and adaptations link types that will be used to colour items in the UI.



Enable hierarchical options and do not allow reordering:

- To become top-level items
- Outside an item's own hierarchy

Enable adaptations for the item type **REQ** and specify the adaptations and references link types.

Define the link rules for these three link types for the item to be:

Table 1: Link Rules for Adapting One Item Type			
From Item	**To Item**	**Link Type**	**Default**
REQ	**REQ**	**HAS CHILD**	Y
REQ	**REQ** or **<any>**	**REFERENCES**	
REQ	**REQ**	**IS ADAPTATION OF**	

Cradle will create these rules automatically for you.

Creating New Items 1.3

The project will create new items by selecting existing **REQ** items and using the Cradle operations:

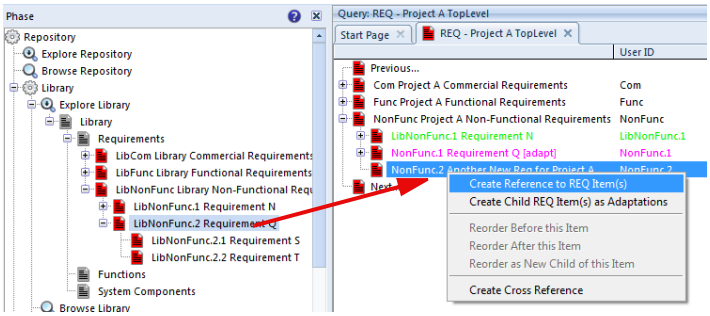
- **New** → **Child**, to create a child of an existing **REQ** item
- **New** → **Sibling**, to create a sibling of an existing **REQ** item
- **New** → **Hierarchy**, to create a hierarchy of new **REQ** items

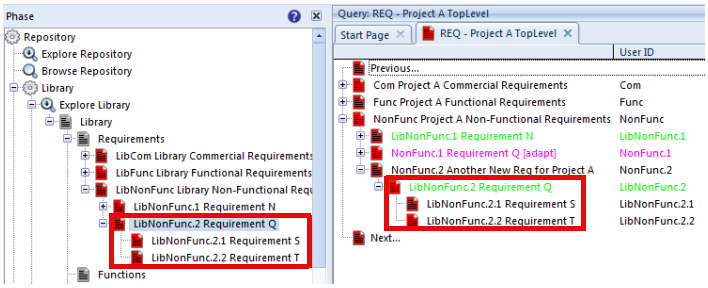
All new items will be linked to the existing items by **HAS CHILD** links, as this is the default link type for cross references between **REQ** and **REQ** items.

Use a Library Item 1.4

To use a library requirement inside the project's hierarchy:

1. Open the library hierarchy and find the item to be referenced
2. Open the project hierarchy and find where the referenced item should be placed
3. Drag and drop the library item onto the project item and choose **Create Reference...**
4. Click **OK** in the next dialog





A cross reference is created from the project item library item. The library item is shown in green to emphasise that it is a reference.

The expansion of the library's requirement is now part of the project's requirement hierarchy, so requirements **S** and **T** in the example can be seen in the project's own requirement hierarchy.

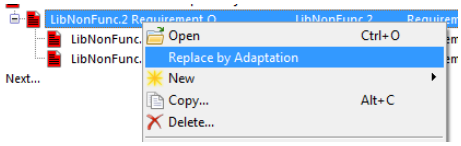
In most cases, the referenced requirement and its expansion would be read-only to the user who has referenced them.

Adapt a Library Item 1.5

If the project wants to change an item referenced from the library, to:

- Change the requirement itself
- Change the items that are linked to the requirement
- Change any of the linked requirements

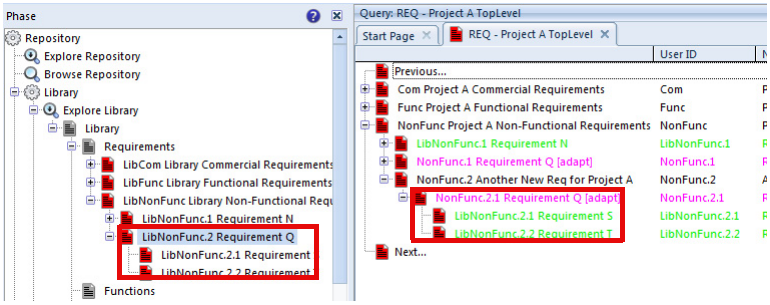
it cannot change the library item because it may be used elsewhere, and it should be read-only to the project. So the project must create an adaptation of the library requirement, by selecting it and:



- Right click and **Replace by Adaptation**, or
- Choose the **Item** tab and select **Replace by Adaptation**

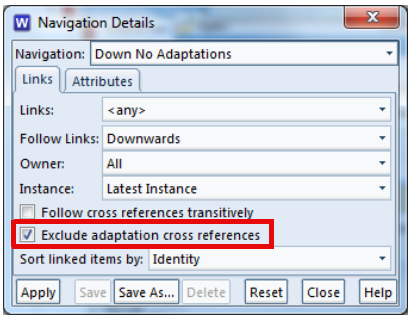
A dialog will be shown that summarises what links will be created. Select **OK** in this dialog and:

- The requirement will be replaced by an adaptation, a copy of the requirement that is owned by the user
- The item is named and coloured to show that it is an adaptation
- The item is linked into the project's requirement hierarchy
- The item is linked by references to the same items to which the library requirement is linked



The original library requirement **Q** has been replaced by an adaptation of **Q** that the project can modify as needed, and the expansion of the library requirement **Q** is now the expansion of the project's own requirement **Q**. So requirements **S** and **T** in the example can be seen in the project's own requirement hierarchy as references.

Browsing 1.6



When a database uses adaptations there can be many types of links between items. When browsing such databases, it is helpful to not follow adaptations' cross references except when you know you want to use them, such as to find library items that have been adapted by projects.

To avoid following adaptations' cross references, always use a navigation which will exclude adaptation cross references, such as the **Down No Adaptations** navigation provided with all Cradle systems. You can exclude adaptation cross references by simply selecting a checbox in the setup of the navigation, as shown.

Adaptations for Two Types of Item 2

This is the most common situation where adaptations are helpful. A hierarchy of items exists in the library and the project has a hierarchy of a different type of item. The project will:

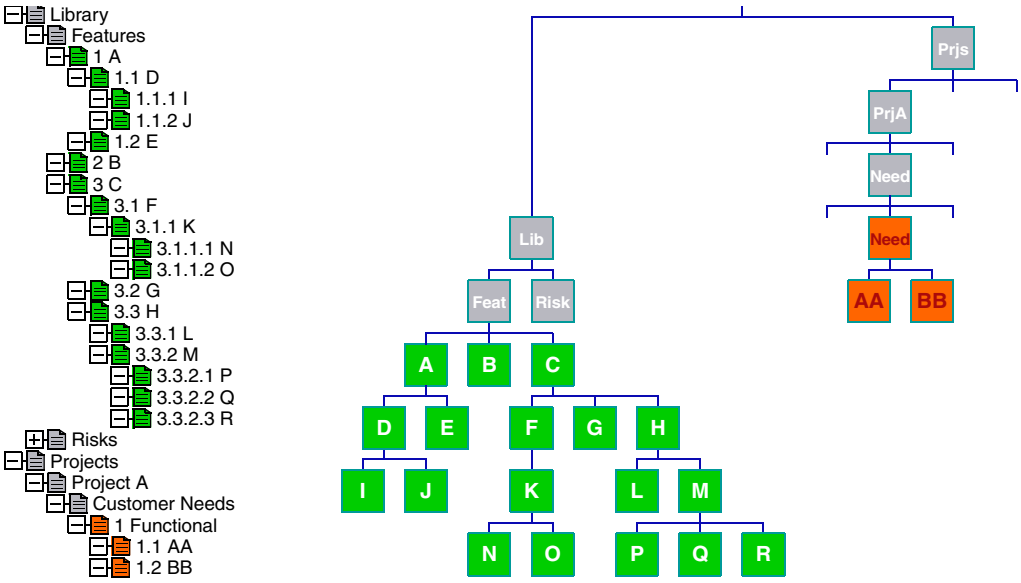
- Create its own items
- Reference items from the library
- Create its own adaptations of library items

The important characteristics of this case are that two item types are involved, and that any other project information is to only link to the project's information, and not information in the library.

Typical Data 2.1

A project has customer needs as **NEED** items and creates features as **FEATURE** items to satisfy these needs. The features may be derived from a set of standard features held in the library. All other project information will only link to features ***in the project's own hierarchy.***

A typical set of information could be:

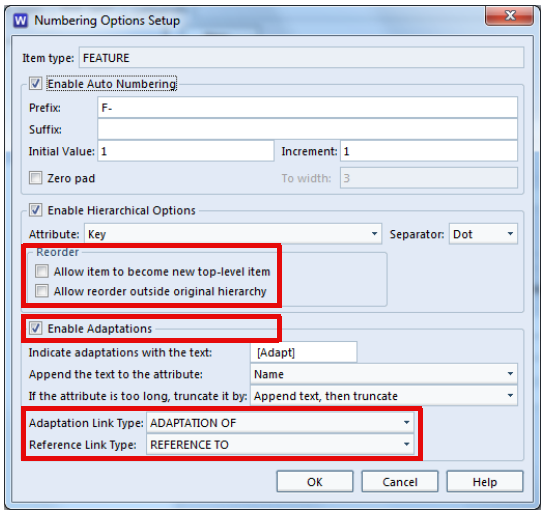


Schema Setup 2.2

Define four link types with names such as:

- **HAS CHILD**, for the hierarchical parent-child links
- **REFERENCE TO**, for references from projects into the library
- **ADAPTATION OF**, for adaptation links
- **SATISFIED BY**, for links from needs to the features

and set colours for the references and adaptations link types that will be used to colour items in the UI.



Enable hierarchical options and do not allow reordering:

- To become top-level items
- Outside an item's own hierarchy

Enable adaptations for **FEATURE**s and **NEED**s and specify the adaptations and references link types for both of the item types. The figure to the left shows the setup for **FEATURE** items. The setup for **NEED**s is the same.

Define the link rules for these items and link types to be:

Table 2: Link Rules for Adapting Two Item Types

From Item	To Item	Link Type	Default
FEATURE	**FEATURE**	**HAS CHILD**	Y
FEATURE	**FEATURE** or <any>	**REFERENCE TO**	
FEATURE	**FEATURE**	**ADAPTATION OF**	
NEED	**NEED**	**HAS CHILD**	Y
NEED	**NEED** or <any>	**REFERENCE TO**	
NEED	**NEED**	**ADAPTATION OF**	
NEED	**FEATURE**	**SATISFIED BY**	

Cradle will create these rules automatically for you.

Alternative Schemas 2.3

In general, when items of type **A** are linked to items of type **B**:

- To reference from **A**s to **B**s, enable adaptations for the **A**s
- To create adaptations of **B**s from **A**s, enable adaptations for the **B**s

So if adaptations are enabled for **NEED**s, but not **FEATURE**s, you can create a reference from a **NEED** to a **FEATURE**.

If adaptations are enabled for **FEATURE**s and not **NEED**s, you can create a link from a **NEED** to an adaptation of a **FEATURE**.

If adaptations are enabled for **FEATURE**s and **NEED**s, you can:

- Create a reference from a **NEED** to a **FEATURE**
- Replace a referenced **FEATURE** by an adaptation
- Create a link from a **NEED** to an adaptation of a **FEATURE**, doing the first two of these operations in sequence

You can also create any cross references allowed by the schema's link rules, such as **SATISFIED BY** links from **NEED**s to **FEATURE**s.

Creating New Items 2.4

The project will create new items by selecting existing **NEED** items and using the Cradle operations:

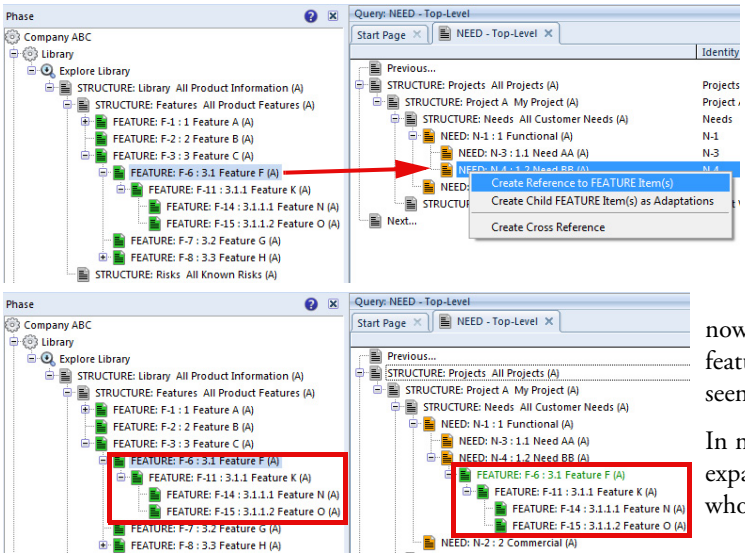
- **New** → **Child**, to create a child of an existing **NEED** item
- **New** → **Sibling**, to create a sibling of an existing **NEED** item
- **New** → **Hierarchy**, to create a hierarchy of new **NEED** items

New items will be linked to the existing items by **HAS CHILD** links, the default link type for cross references between **NEED** items.

Use a Library Item 2.5

To reference a library feature from the project's needs hierarchy:

1. Open the library hierarchy and find the item to be referenced
2. Open the project hierarchy and find where the referenced item should be placed



3. Drag and drop the library item onto the project item and choose **Create Reference...**
4. Click **OK** in the next dialog

A cross reference is created from the project item library item. The library item is shown in green to emphasise that it is a reference.

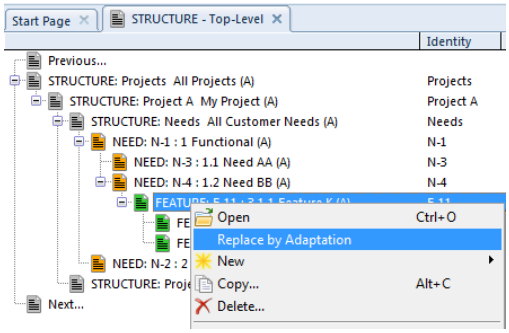
The expansion of the library's feature is now part of the project's item hierarchy, so features **K**, **N** and **O** in the example can be seen in the project's own hierarchy.

In most cases, the referenced feature and its expansion would be read-only to the user who has referenced them.

Adapt a Library Item 2.6

If the project wants to change an item referenced from the library, to:

- Change the item itself, the **FEATURE** in this case
- Change the items that are linked to the item
- Change any of the linked items

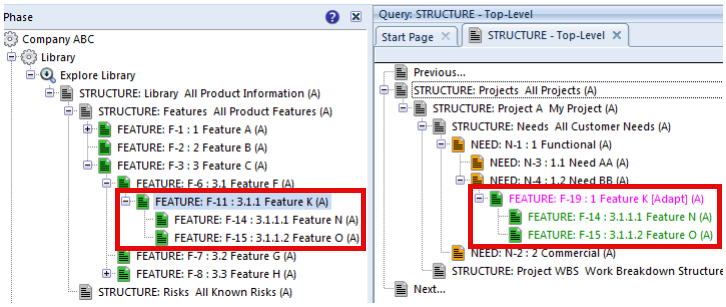


it cannot change the library item because it may be used elsewhere, and it should be read-only to the project. So the project must create an adaptation of the library item, by selecting it and:

- Right click and **Replace by Adaptation**, or
- Choose the **Item** tab and select **Replace by Adaptation**

A dialog will be shown that summarises what links will be created. Select **OK** in this dialog and:

- The feature will be replaced by an adaptation, a copy of the feature that is owned by the user
- The item is named and coloured to show that it is an adaptation
- The item is linked into the project's needs hierarchy
- The item is linked by references to the same items to which the library feature is linked



So the original library feature **K** has been replaced by an adaptation of **K** that the project can modify as needed, and the expansion of the library feature **K** is now the expansion of the project's feature **K**. So features **N** and **O** in the example can be seen in the project's own hierarchy:

Summary 3

An organisation can reuse information between projects by simply having a set of items in a library and then creating cross references from these into the projects that use them.

However, this creates a problem that if a project wishes to change a library item that it is using, it cannot change that library item:

- Because other projects may be using it
- Because the project should have read-only access to the library

The adaptations mechanism overcomes these problems, making it easy to reference library information and to make project-specific adaptations of this information.

The adaptations mechanism is part of the **Cradle-PDM** (project data management) module, so it is free to use, and is available in all Cradle systems.