Polarion 2015 How-To

Variant Management Project Setup

1 Overview

This document describes how to configure existing projects, created with Polarion version prior to version 2015, for Variant Management usage. The configuration provided below is based on Polarion best practices for a product line engineering approach as defined in "Specification Project with Variant Management" template.

2 System Requirements

- You must run Polarion version 2015 (or later).
- A valid license for the Polarion VARIANTS Add-on product must be installed on the Polarion server
- The Polarion VARIANTS Server Add-on product must be obtained from the Polarion Software web site and installed according to the included instructions. http://www.polarion.com/downloads/index.php

3 Project Configuration

Configuring an existing project for Variants Management requires several simple steps that are described below:

3.1 Configure "Variant" Work Item type

1. Create a new Work Item type that represents a Variant item (Project Administration > Work Items > Types)
   The type ID should be "variant" or you can set it by system property: com.polarion.variants.variantType

   ![Variants Work Item Type](image)

2. Setup a workflow for the new type (Project Administration > Work Items > Workflow)
   The recommended status enumeration can be fairly simple, something like: Draft, In Review, Defined and Rejected
3. Define following custom fields for the new type (Project Administration > Work Items > Custom Fields):
   Add a custom field Feature Selection, with ID: featureSelection and type: Feature Selection

4. Update the Work Item form layout (Project Administration > Work Items > Form Configuration > Form Layouts)
   Add Variant Documents and Feature Selection as separate Sections. Exclude the Feature Selection field from the custom fields panel:
   ```xml
   ...<field id="description"/>
   <field id="featureSelection"/>
   <field id="variantDocuments"/>
   ...<panel description="Custom Fields">
   <field id="@allCustomFields,-featureSelection"/>
   </panel>
   ```
   Hint: You can also create a new project using the "Specification Project with Variant Management" project template and reuse the configuration files by copying them to your existing projects:
   - polarion/tracker/fields/variant-custom-fields.xml
   - polarion/tracker/workflow/variant-workflow.xml
   - polarion/tracker/fields/variant-status-enum.xml
   - polarion/hats/_default/tracker/variant-form-layout.xml

3.2 Configure "Feature" Work Item type

1. Create a new Work Item type that represents a Feature item (Project Administration > Work Items > Types)
   The type ID should be "feature" or you can set it by system property: com.polarion.variants.featureType

2. Setup a workflow for this type (Project Administration > Work Items > Workflow)
   The recommended status enumeration can be pretty simple. For example: Active and Obsolete
3. Define a new custom field for this type (Project Administration > Work Items > Custom Fields):
   Add custom field Variation Type with ID: variationType and type: Enum: Variation Type

4. Configure additional link roles between Feature type Work Items (Work Items > Enumerations >
   workitem-link-role-enum.xml)
   ID: pvRequires ; Name: requires
   ID: pvConflicts ; Name: conflicts with

   **Note:** Define a link rule for both "requires" and "conflicts with" roles, so that such links are used only between one Feature and another Feature.

   **Rules for link role: requires**

   **Hint:** You can also create a new project using the "Specification Project with Variant Management" project template and reuse the configuration files by copying them to your existing projects:

   - .polarion/tracker/fields/feature-custom-fields.xml
   - .polarion/tracker/workflow/feature-workflow.xml
   - .polarion/tracker/fields/feature-status-enum.xml

3.3 Configure Specification Work Item types

   **Note:** Specification Work Item types can be any Work Item type which will be part of your Product Line and will be included in Master Specification Documents (e.g. Requirement, Test Case, SW Requirement, etc.)

   1. Add a custom field Restriction to each Work Item type which will be used as a specification item
      ID: pvRestriction and type: pvSCL
3.4 Separating Common Specifications and Variant Development Projects

It is possible to manage variants across multiple projects. For example, you might set up one project for specifications and another for variants. One project, called the Definition Project must contain the Feature Model and Master Specification documents.

Consider an example where the master requirements and test case specifications are defined and managed in one project, and variant specifications and the actual variant development is managed in another project.

**Definition Project**

One project, called the Definition Project contains the Feature Model and the Master Specifications. The Feature Model Document contains the Feature items and defines the variability model. Master Specification Documents contains common, shared and variant requirements (or test cases, or other asset types), restricted to features by the Restriction field.

**Variant Development Projects**

The development of each variant is separated into a dedicated project. This project contains the Variant item (variant definition), and the generated specification Documents.

For these projects, an administrator needs to define a custom field for the Variant type enabling project users to subsequently select a project as the Definition Project for each Variant type Work Item. When generating Variant Specifications, the Definition Project will be used as the source for Master Specifications.

The following section describes the configuration steps for your administrator.

**3.4.1 Configure Definition Project Custom Field**

To configure a project to support cross-project Variant Management:

1. Open Custom Field administration (Project Administration > Work Items > Custom Fields):
2. Add a new custom field for type Variant with ID:definitionProject and type:Enum Project