

Changes for Procead Version 2.0

Changes based on the predecessor version 1.5, revision 896

German or English user interface

- Change to your preferred language at run time

Source Code Linking and Navigating

- New Code Linking feature introduces symbolic linking of source code with EA elements. This closes the gap to 100% requirements traceability into the source code .

Rule Compliance

- Added increments of rule compliance:
 - Sloppy* - Rule breaches are acceptable by user confirmation
 - Override* - Rule breaches are acceptable when providing a reason statement
 - Strict* - Rule compliance enforced (no exceptions)

Procead Version Warning

- Procead spots a warning , if the project model has been processed by a newer version

Fixed Issues

- added model branch version increment after degrading the model branch root package
- instead of using the target version, Procead now asks a branch version increment if an invalid version is detected on a version root package

Changes for Procead Version 1.5

Changes based on the predecessor version 1.4, revision 797

Product Configuration Saving

- Obligatory saving of work products to a named baseline is now adjustable between "don't save" and "save to model" at each decision gate.

Product Configuration Locking

- Extended decision gate definition to lock work products when a project progress decision is made.

Qualification relevance of Action Effects

- Added possibility to adjust qualification relevance of Action Effects.

Status related Locking

- Extended product state definition to prevent elements from further modification. Element locks are now related to its status.

Release Notes

Auto Unlocking

- Procead now asks for permission to remove locks when applying quality status rules (instead of posting an error)

Free Trial

- Procead now shipped with an immanent 30 days trial, making formal request for trial license obsolete.

Fixed Issues

- Elements not ruled by Procead won't affect version numbers of owning elements any more
- Deleting of essential definition objects (causing an unhandled exception) now prohibited

Changes for Procead Version 1.4

Changes based on the predecessor version 1.3, revision 582

Qualification relevance of Test-Classes

- Added possibility to adjust qualification relevance on Test-Classes. This enables process models, where element tests are specified as a result of an element test, instead of being part of the element specification itself. This is commonly applied to Inspections/Reviews.

Incrementing a model branch version

- When sweeping through the product qualification of a model branch its now possible to rule version increments. This applies only to projects having version control on model branches enabled.

Fixed Issues

- Wrongly discovered change event while adding elements fixed

Changes for Procead Version 1.3

Changes based on the predecessor version 1.2, revision 562

Version Control on Model Branches

- Procead allows *Packages* to be configured as origin of a target version for a model branch. This enables e.g. master documents to have a disconnected version number besides the project target version.

Fixed Issues

- Compatibility with RDBMS MySQL 5.7 improved (ISS0049, ISS0051)

Changes for Procead Version 1.2

Changes based on the predecessor version 1.1, revision 470

Release Notes

Software Upgrading

- Enabled upgrading of software product with downward compatibility to existing user definitions

Synchronisation of Settings

- A running Procead instance applies settings from the model immediately on invocation of the settings dialog

Defining Project Target Version manually

- The target version increment (*TargetIncrement*) of a "proceed"- transition may be defined as "custom"
- If the *TargetIncrement* of a "proceed"- transition is set to "custom", the project target version will not be incremented by Procead, but must be entered by hand during project progress decision making
- Procead alerts, when entering a lower version than the current project target version

New Dialog to support setting of Version Increments

- A dialog to set the *TargetIncrement* of a "proceed" - transition appears on creation or selection

New Transition Stereotype

- A new stereotype "qualify" for transitions between product states in a product qualification model is introduced to distinguish qualification and proceeding
- The formerly sole "proceed"-stereotyped transition is deprecated and should be replaced in all product qualification models

Project Target Version instantly effective on change

- Changes on the project target version are visible to the team (effective on all running instances of Procead) immediately after its change

Element Version set on creation

- Procead sets the element version to the current *project target version*, in case the element is created and the status is a qualified product state

Don't initialize Elements' Status

- Procead doesn't initialize the status of elements or packages any more. This is a function of EA in conjunction with a project template package definition

Role based Procead Settings

- The Procead Settings may be write protected with ordinary locks (with user security enabled in EA)

Modifying linked documents

- Linked documents are qualification relevant and optionally followed

Procead Version 1.1

Pilot Release, Version 1.1, Revision 470

Anforderungen zur Definition und Umsetzung des Inkrements hinzufügen

Release Notes

1. Fehlenden Trace zum UseCase hinzufügen

2. Anforderungen in SW-Spezifikation Procead hinzufügen

Die Zielversion soll immer beim Start einer neuen Iteration in der Projektdurchführung durch Procead automatisch inkrementiert werden.

Vorschlag: Jeder EP erhält ein tagged value, aus dem das Inkrement hervorgeht.

z.B.: TargetVersionIncrement = none|major|minor, default=none

Procead inkrementiert bei Erreichen des EP bei TargetVersionIncrement

- none um 0

- major um 1.0

- minor um 0.1

Enabled Features of Pilot

- Work- and Qualification Flows definable using UML
- Version and Status Control on Elements and Packages
- Configuration Management on Deliverables and Decision Gates
- Quality Management on Elements and Packages
- Visibility of overall Project Progress and Qualification Level
- Continuous Documentation of Decisions and Decision Criteria
- Alignment of Version and Status with subordinated Elements or Packages
- Draws Baselines on Project Progress Decisions