



Innovator 14.1

# ReleaseInfo

Innovator 14.1.1.21128



Contents

- Innovator 14.1 (Innovator 14.1.1.21128) ..... 1
  -  Version Consistency ..... 1
    - Overview of Version Numbers (Builds) ..... 1
    - About the API Documentation ..... 1
  - Administration ..... 1
    - User Management: Filtering by "User Name in Model" for Member Assignment..... 1
  - Model Editor ..... 1
    - DDL Generation: DropTable Before Each CREATE TABLE ..... 1
    - ER-DB Mapping: Short Name Constraints Not Applied To Empty Names ..... 1
    - ER-DB: Denormalization of Sibling Tables ..... 2
    - Search: Performance Improved for Decision Tables ..... 2
    - Change Logging when Copying Diagrams ..... 2
    - Diagram Copy: Handling of Constraints Corrected..... 2
    - Excel Import of Multi-Line Element Names..... 2
    - Excel Import: Enumerated Values Correctly Imported..... 2
    - Import of a Model Fragment with Model Root..... 2
    - Stereotype Display in Table Editor for Content..... 2
  - Configuration ..... 2
    - Documentation: Additional Details for Operations ..... 2
- Innovator 14.1 (Innovator 14.1.1.20822) ..... 3
  - Administration ..... 3
    - classIX Licenses for Project License Server Version 11.5 ..... 3
    - LDAP Synchronisation: Independent of User's OU ..... 3
    - Version Management: Proposed Names for Backups ..... 3
  - Model Editor ..... 3
    - Beamer in Extras ..... 3
    - Change Logging for View Expressions ..... 3
    - DDL Export: Existence of Output File ..... 3
    - DDL Export: Formatting ..... 3
    - DDL Export: Referencing of SubSelect View ..... 3
    - DDL Generation: "WHERE " at the Beginning of a View Expression ..... 3
    - DMN: Handling of Large Decision Tables ..... 4
    - BPMN Import Supports Process with Process Definition Diagram ..... 4
    - BPMN: Deleting a Task with Loop Characteristics ..... 4
    - DMN: Authority Requirements of a Knowledge Source are Imported ..... 4
    - DMN: Default Values in Simulation ..... 4
    - DMN: Simulation with Built-in Functions today() and now() ..... 4
    - Documentation: Large Diagrams ..... 4
    - ER Diagram: Role (in OLAP Hierarchy) ..... 4
    - ER-DB Mapping of View Expressions ..... 5
    - ER-DB Mapping Uses Data Type and Default Value from the Correct Entity ..... 5
    - Exception Handler in the Activity Diagram ..... 5
    - Importing Model Fragment: Import from Managed Model ..... 5
    - Importing Model Fragment: Import into Managed Model ..... 5
    - Unique Names when Changing the Element Type of Concepts ..... 5
  - Configuration ..... 5
    - Profile "J2SE": Stereotype Properties "Java modifier" Moved ..... 5
    - Importing Model Fragment: Different Roles of Execution Rights ..... 5
- Innovator 14.1 (Innovator 14.1.1.20510) ..... 6
  - Administration ..... 6
    - Version Management: Change Log is Copied ..... 6
    - Version Management: Place Large Models under Version Management ..... 6

Model Editor .....	6
<input type="checkbox"/> Automatic WHERE Addition in View Expressions Omitted.....	6
<input type="checkbox"/> Concept Diagram: Connecting to Multiple Representations of a Concept.....	6
<input type="checkbox"/> Copying of Entities Including Referencing Relationships.....	6
<input type="checkbox"/> Initial Display of Keys in Entity Nodes .....	6
<input checked="" type="checkbox"/> ArchiMate Export with Local Validation .....	6
<input checked="" type="checkbox"/> DMN: Export and Import of Lists.....	6
<input checked="" type="checkbox"/> DMN: Modifying the Properties When Changing a Boxed Expression .....	6
<input checked="" type="checkbox"/> Excel Import with UUID Determination of the Stereotype.....	7
<input checked="" type="checkbox"/> Root Elements in Process Hierarchy.....	7
<input checked="" type="checkbox"/> Synchronize Model Elements: Change Sets.....	7
Innovator 14.1 (Innovator 14.1.1.20401).....	8
General Functions .....	8
<input checked="" type="checkbox"/> Architecture: Component Bus Required .....	8
<input checked="" type="checkbox"/> Checklists Supplement Help .....	8
<input checked="" type="checkbox"/> Transformator: Transformation for Version Directories .....	8
<input checked="" type="checkbox"/> Verify Option for the Use of Interfaces .....	8
<input checked="" type="checkbox"/> Version Management: Symbolic Names for Innovator Links .....	8
<input type="checkbox"/> ArchiMate: Support for Version 3.1 .....	8
<input type="checkbox"/> Directories of Log Files .....	8
<input type="checkbox"/> DMN: Full Support for Version 1.1 .....	9
<input type="checkbox"/> DMN: Value Range of Input Changed.....	9
<input type="checkbox"/> Java VM Settings Support Environment Variables .....	9
<input type="checkbox"/> Migration of Join Options at From Clauses .....	10
<input type="checkbox"/> Version Management: Improved Operational Stability .....	10
<input checked="" type="checkbox"/> Migration: Cleanup of the File inofont.ini .....	10
Administration .....	10
<input checked="" type="checkbox"/> Model Server: Shut Down Automatically When Inactive .....	10
<input checked="" type="checkbox"/> User Configuration: Right for Environment Administrators .....	10
<input checked="" type="checkbox"/> Version Management: Authorizations for Model Versions.....	10
<input checked="" type="checkbox"/> Version Management: Creating New Managed Models .....	10
<input checked="" type="checkbox"/> Version Management: Defining Resource Restrictions.....	11
<input checked="" type="checkbox"/> Version Management: Model Recovery via the Administration Program.....	11
<input checked="" type="checkbox"/> Version Management: Restore Backup.....	11
<input type="checkbox"/> Login not Needed for Single Sign-on .....	11
<input type="checkbox"/> Performance: Faster Listing of Models in the Administration Program.....	11
<input type="checkbox"/> User Configuration: Rules for System Accounts (Windows).....	11
<input type="checkbox"/> Using Group Managed Windows Service Accounts.....	11
<input type="checkbox"/> Version Management: Place Models Under Version Control.....	11
<input checked="" type="checkbox"/> Determination of the Domain with LDAP Synchronization .....	12
<input checked="" type="checkbox"/> Export of Large Models Possible .....	12
<input checked="" type="checkbox"/> Project License Servers: Starting Previous Versions.....	12
<input checked="" type="checkbox"/> Server Installation Allows Creating New Models in the Administration Program.....	12
<input checked="" type="checkbox"/> Set Up Model Server as Service with Absolute Path via Command Line.....	12
<input checked="" type="checkbox"/> Version Management: Java Compatibility of Agents (inoagent) .....	12
Model Editor .....	12
<input checked="" type="checkbox"/> ArchiMate: Import and Export of the Enterprise Architecture Model .....	12
<input checked="" type="checkbox"/> Beamer: Using Different Sorting Methods .....	12
<input checked="" type="checkbox"/> BI: Manual Assignment of DB Views to Hierarchy Levels .....	12
<input checked="" type="checkbox"/> BI: Variants of the Measures Dimensions Diagram .....	13
<input checked="" type="checkbox"/> DMN: Simulation of Decisions.....	13
<input checked="" type="checkbox"/> Excel: Import and Export of Model Elements and their Properties.....	13
<input checked="" type="checkbox"/> Sorting Options in the Structure Content.....	13
<input checked="" type="checkbox"/> Synchronize Model Elements: Automatically Select the Determination Method of the Reference Time .....	13
<input checked="" type="checkbox"/> SysML Model Templates and Add-ons .....	14

- Version Management: Creating and Managing of Model Versions ..... 14
- Whiteboard Diagram: Further Reference Edges ..... 14
- BI: Create Dimension Attribute Directly for Matching Attribute..... 14
- BI: Dimension Export and Dimension Maintenance with Formatting and Without Verifications ..... 14
- BPMN Import of Diagrams from Process Note..... 14
- Change Sets: Log Size Not Used as Deletion Criterion ..... 14
- Composite Structure Diagram: Creating Connectors ..... 14
- Concept Diagram: Concept Connections Permitted as Target of Concept Connections ..... 15
- Concept Diagram: Separate Directional Markers of Concept Connection ..... 15
- DB Import: Importing or Exporting Connections..... 15
- DB Mapping Without Error Message When Missing Access Right ..... 15
- Display of "is Protected" Changed..... 15
- Improved Creation of the Uniqueness of Short Names ..... 15
- Model Selection on Opening and Logging In..... 15
- Modeling Steps are Removed ..... 15
- Performance: Improved Start Time and Main Memory Consumption of the Client ..... 16
- Search with Operator 'contains'..... 16
- Show Name of Connected Elements in Properties..... 16
- Specification Texts: Cancel Possible ..... 16
- Standard Layout of Dependencies Between Contents of a Structure Definition or Classifier Node Improved..... 16
- Structure Diagram: Moving Elements ..... 16
- Table Editor "Contents as Table": Register Labeling ..... 16
- Verification of Access Rights during Renaming ..... 16
- ☑ "Change Element Type" Wizard Suppresses Card Switching When Errors Occur ..... 16
- ☑ Activity Diagram: Object Flow Does Not End Server On Undo..... 16
- ☑ Attachments in Copied Models of the Same Repository..... 17
- ☑ BPMN Import Can Import Into Components..... 17
- ☑ BPMN: Drag and Drop a Component onto a Data Object ..... 17
- ☑ BPMN: SVG Export of Data Objects in Process Diagrams..... 17
- ☑ Concept Diagram: Split Concept Connections for Nested Concept Nodes..... 17
- ☑ Convert to Model File..... 17
- ☑ DB Element Selection When Scrolling ..... 17
- ☑ DB Mapping a Foreign Key in Multiple Merged Tables ..... 17
- ☑ DB Mapping Adjusts Null Property..... 17
- ☑ DB Mapping Merges Local Table Columns ..... 17
- ☑ DB Mapping: Configuration Loader Creates Engineering Action..... 18
- ☑ DB Mapping: Configuration Loader Creates Engineering Action..... 18
- ☑ DB Mapping: Cross-model Mapping Between Conceptual and Object-Oriented Models Improved ..... 18
- ☑ DB Mapping: Maintenance of Null Property of External Attributes when Changing a Relationship..... 18
- ☑ DDL Export for Microsoft SQL Server Places COLLATE Correctly..... 18
- ☑ DDL Export for Microsoft SQL Server: ForeignKey/Trigger Statements ..... 18
- ☑ DDL Export: From Clause Statements in Union Views Create Namespaces ..... 18
- ☑ DDL Import Creates View Columns for Union View ..... 18
- ☑ Duplicate Descriptions After Changing the Element Type Are Prevented..... 18
- ☑ Import of Model Fragments ..... 19
- ☑ Instances of Associations Can Be Copied Only to a Limited Extent ..... 19
- ☑ Merging of Aggregations Corrected ..... 19
- ☑ Order When Moving Entries in Compartments..... 19
- ☑ Specification Texts: Treatment of Text Indentations ..... 19
- ☑ Structure Elements Show Properties for "Used Features" ..... 19
- ☑ Synchronize Model Elements: Algorithm for Determining the Comparison Pairs ..... 19
- ☑ Synchronize Model Elements: Merging Views on Processes Without Definition Rejected..... 19
- ☑ Synchronizing Model Elements: Displaying of Edges when Merging Diagrams ..... 19
- ☑ Undo after Creating a New Process View..... 20
- ☑ Visibility of Stereotype Properties..... 20
- Configuration ..... 20
  - Customize Display Options in Extension Profiles ..... 20
  - Documentation: Additional Information Chapters..... 20

- Information Structure of Details Tool Window is extendable ..... 20
- Display Options of Diagrams Cleared ..... 20
- Documentation: Handling Tables ..... 21
- JavaPackage Allows Additional Characters in Name ..... 21
- Properties Combined in Documentation Structures ..... 21
- ☑ Documentation: Postprocessing Command Supports Spaces in Path..... 21

- New Function
- Changed
- Fixed

## Innovator 14.1 (Innovator 14.1.1.21128)

### Version Consistency

Please note when updating your Innovator installations:

Identity of model server version and client version is highly recommended for safe use of Innovator. It is mandatory that model server and clients have identical release numbers (XX.X.X).

The hotfix is fully interface-compatible with Innovator 14.1.1.20822 and limited interface-compatible with Innovator 14.1.1.20510 and 14.1.1.20401. You can use the server components of this release with older client components of version 14.1.1. However, clients of version 14.1.1.21128 require servers of the same version.

You can use your own plug-ins without creating them again.

Repositories updated by transformation cannot be used in older versions.

### Overview of Version Numbers (Builds)

Product Description	Version Number (Build)
Innovator 14.1.1 Hotfix	Version 14.1.1.21128
Innovator 14.1.1 Hotfix	Version 14.1.1.20822
Innovator 14.1.1 Hotfix	Version 14.1.1.20510
Innovator 14.1.1	Version 14.1.1.20401

### About the API Documentation

The API changes are continuously documented in the online help (see "User Customizations> Innovator SDK Documentation Library>Change Log").

## Administration

- User Management: Filtering by "User Name in Model" for Member Assignment**

In the dialog for assigning members to a group, the filter also affects "User Name in Model".

## Model Editor

- DDL Generation: DropTable Before Each CREATE TABLE**

In the DDL configurations, the option "DROP statement before the CREATE statement for tables" in the group "Formatting" effects the output of a DROP TABLE statement before each CREATE TABLE statement.

- ER-DB Mapping: Short Name Constraints Not Applied To Empty Names**

Name constraints for short names apply only to assigned names. Multiple elements with empty short names are permitted.

#### ER-DB: Denormalization of Sibling Tables

When merging tables using foreign keys that are linked with "Constraint (of disjoint DB Tables)", this link is transferred to the target table without checking the configuration.

#### Search: Performance Improved for Decision Tables

In large decision tables, the processing of the search by name takes significantly less time.

#### Change Logging when Copying Diagrams

For annotations in diagrams, the associated graphical elements are logged using the associated logical information. When the diagram is copied, the logical information may not be available for the change log of the affected elements. In this case, the change logging for the affected elements does not take place and thus does not lead to the shutdown of the model server.

#### Diagram Copy: Handling of Constraints Corrected

If you copy a diagram, then all graphic contents are copied with the connections to the logic. If the copy set also contains a part of this logic, the connections will be redirected to the copied logic. Constraint connection edges, which are connected to the wrong graphic by the different adoption of the logic, are automatically removed from the copied diagram. The copied diagram ensures that all constraint nodes are fully displayed. If necessary, connection edges to existing graphic contents are automatically added. The constraint node is removed from the diagram if not all connection edges can be added.

#### Excel Import of Multi-Line Element Names

Innovator does not support multiline element names, so line breaks are removed when importing element names.

#### Excel Import: Enumerated Values Correctly Imported

The Excel import imports an enumerated value with the properties "Text of Specification Expression" and "Enumeration Value".

#### Import of a Model Fragment with Model Root

When importing a model fragment that includes the root node of the Innovator model, all existing application configurations and database connections in the target model are retained. Any display languages defined in the target model that are neither contained in the model fragment nor used by a translation value in the target model will be deleted.

#### Stereotype Display in Table Editor for Content

Stereotypes without localized names are displayed in the table editor for content as a combination of logical stereotype name and generalization name.

## Configuration

#### Documentation: Additional Details for Operations

For operations the properties "Realizes" and "is Realized by" are available for the details tree and documentation.

## Innovator 14.1 (Innovator 14.1.1.20822)

### Administration

**classiX Licenses for Project License Server Version 11.5**

A running project license server of version 11.5 can be assigned available classiX licenses.

**LDAP Synchronisation: Independent of User's OU**

Automatic LDAP synchronization also works for users who are in a different organizational unit (OU) than the groups.

In addition to the group data, the relevant user data is also stored in the connection data.

**Version Management: Proposed Names for Backups**

The proposal for the name of a backup is designed in such a way that backups are sorted by creation time in ascending order.

### Model Editor

**Beamer in Extras**

The "Beamer" tool window for role-based mapping themes for the focused derivation of model elements is opened in the "Extras" ribbon tab.

**Change Logging for View Expressions**

The expressions of view elements are added to the change log as unformatted text.

**DDL Export: Existence of Output File**

The existence of an output file is not a check criterion for a successful DDL export.

**DDL Export: Formatting**

In the dialog "Export DDL file" the checkbox "Generate database object names quoted" in the area "Options" has been replaced with the list option "Quotes/brackets for element names". The list option "Character for the end of the command" has been added.

**DDL Export: Referencing of SubSelect View**

The reference to a SubSelect view within a From clause has been replaced in the DDL export by a SELECT statement.

**DDL Generation: "WHERE " at the Beginning of a View Expression**

The \$INODIR\java\InoIMInOut\configuration directory contains a viewwhere.xml file that contains all strings for which "WHERE " is not inserted at the beginning of a view expression.

#### **DMN: Handling of Large Decision Tables**

If the number of cells in a decision table exceeds the threshold value of 500, the cell contents are displayed as static text. The inline editor is not used. This means:

- no direct text input in the cell
- no syntax highlighting
- no jumping to elements via hyperlinks

The cell content can be modified with the context menu command "Modify" in a dialog. It can also be opened by double-clicking on a cell.

If the number of cells in a decision table exceeds the threshold value of 1200, the rules are output page by page. The pages can be switched in the upper left corner of the decision table. Alternatively, [Ctrl]+[Page up] and [Ctrl]+[Page down] can be used for scrolling. A rule or cell must be selected for this.

Entries can be dragged and dropped within a page as usual. For a cross-page move, the context menu command "Move Rule..." is available for the selected rule.

#### **BPMN Import Supports Process with Process Definition Diagram**

The BPMN import can import a process with a process definition diagram into a model where the model configuration does not support process definition diagrams. In this case, the process definition diagram is imported as the definition diagram of the process.

#### **BPMN: Deleting a Task with Loop Characteristics**

Loop characteristics are stored in a separate element below a task. After deleting a task with loop characteristics, the elements are restored correctly when running undo and do not trigger the error message "Root element not found".

#### **DMN: Authority Requirements of a Knowledge Source are Imported**

Authority requirements of a knowledge source (decision, input data or knowledge source) are adopted upon DMN import.

#### **DMN: Default Values in Simulation**

Standard values within the generated test data are correctly evaluated by the simulation, e.g. time data with time zone.

#### **DMN: Simulation with Built-in Functions today() and now()**

In DMN simulation, the built-in functions today() and now() can be used in literal expressions to use current time information.

#### **Documentation: Large Diagrams**

When generating HTML documentation, even very large diagrams are displayed completely.

#### **ER Diagram: Role (in OLAP Hierarchy)**

In an entity relationship diagram, relationship edges can be added to entities while the connected entity is automatically included in the diagram. If the connected entity is added manually to the diagram, the relationship edges are automatically included. In both cases, relationships of type "Role (in OLAP Hierarchy)" are supported.

**ER-DB Mapping of View Expressions**

When mapping view expressions, the number and position of blanks and line breaks are copied.

**ER-DB Mapping Uses Data Type and Default Value from the Correct Entity**

Mapping from the conceptual model to the database schema does not map the default value and data type of an entity attribute to the table column if the entity attribute is in an entity whose database table was merged into the primary key.

**Exception Handler in the Activity Diagram**

An edge "Exception handler" can be drawn from the actions "Accept event" and "Send signal" if this is configured.

**Importing Model Fragment: Import from Managed Model**

When importing a model fragment from a managed model, no version information is taken from the fragment even if the model is part of the fragment.

**Importing Model Fragment: Import into Managed Model**

When importing a model fragment into a managed model, the UUID of the model is not taken from the fragment even if the model is part of the fragment. The managed model keeps its UUID and is not moved.

**Unique Names when Changing the Element Type of Concepts**

For concepts, the name must always be unique with regard to the namespace and stereotype to avoid inconsistencies. This unique name is also ensured for the "Refactor>Change Element Type" command.

## Configuration

**Profile "J2SE": Stereotype Properties "Java modifier" Moved**

In the profile "J2SE" there are the stereotypes "JavaProperty", "JavaAssociationEnd" and "JavaField" for the element type "Attribute", each with the three stereotype properties "Java modifier: volatile", "Java modifier: final" and "Java modifier: transient". These three stereotype properties have been moved to the common super-stereotype "codeable".

**Importing Model Fragment: Different Roles of Execution Rights**

Upon importing profiles as model fragments (e.g. migration profiles), diverging roles of execution rights are considered and included in the model. The import is not rejected with an error message about incompatible roles of execution rights.

## Innovator 14.1 (Innovator 14.1.1.20510)

### Administration

**Version Management: Change Log is Copied**

When a model is placed under version management or a new version of a managed model is created, the change log of the source model is copied to the newly created model.

**Version Management: Place Large Models under Version Management**

A model can also be placed under version management if its size exceeds 2GB.

### Model Editor

**Automatic WHERE Addition in View Expressions Omitted**

In non-empty view expressions that do not begin with CREATE VIEW or WHERE, a WHERE is not automatically added.

**Concept Diagram: Connecting to Multiple Representations of a Concept**

Concepts can be displayed as nodes multiple times in a concept diagram. Concept links can also be linked to a different node even if this is another representation of the same concept.

**Copying of Entities Including Referencing Relationships**

When copying an entity in the model structure, its relationships for which the entity is on the foreign key side are also copied.

**Initial Display of Keys in Entity Nodes**

When using the current MID profile "Entity/Relationship Modeling", the section contents "Key" and "Foreign Key" are displayed open by default in an entity node. This can be changed for each diagram via the context menu "Display Options".

**ArchiMate Export with Local Validation**

The export to ArchiMate does not require an Internet connection to validate the exported file.

**DMN: Export and Import of Lists**

DMN lists are correctly exported and imported.

**DMN: Modifying the Properties When Changing a Boxed Expression**

If a variable name is changed in a Boxed Expression, the property "Variable name" is immediately modified in the "Properties" tool window.

**Excel Import with UUID Determination of the Stereotype**

When importing Excel files, it is not necessary to specify the property "Stereotype" if the property "UUID" is specified. In this case, the stereotype of the existing model element is not changed.

If the property "Stereotype" is specified, it is only changed if it is unique.

**Root Elements in Process Hierarchy**

The Process Hierarchy view displays the call hierarchy of the processes. The processes that are either not called at all, called multiple times, or called by a process from another package are displayed as root. If several processes call each other in a ring and each process is called only once, all processes involved are displayed as root. This is especially true if the process is called within the same process.

**Synchronize Model Elements: Change Sets**

The selection of change sets when synchronizing model elements requires the privilege "Viewing Changesets".

## Innovator 14.1 (Innovator 14.1.1.20401)

### General Functions

#### ■ Architecture: Component Bus Required

For the extended control options and the necessary communication between Innovator components, a bus (inobus) must be active in addition to each license server (main or project license server). As services, the license server and the bus must always be installed in parallel. When a license server is started manually, however, the bus is automatically started and terminated with it.

#### ■ Checklists Supplement Help

In order to support and simplify the commissioning and control of processes in connection with Innovator for those responsible, a checklist for migration is available.

#### ■ Transformator: Transformation for Version Directories

The transformator provides the possibility to transform all models under version version in the project directory in one single step. Select the type "Model Versions" in the transformator.

#### ■ Verify Option for the Use of Interfaces

In addition to the check option "Interface realization missing [VFY298]", there is also the check option "Interface usage missing [VFY554]" for interfaces. A message is delivered if there is neither a specialization nor a realization or use of the interface. If a port is typed directly with the interface, this is treated as realization (non-conjugated) or usage (conjugated) depending on the property "is Conjugated".

#### ■ Version Management: Symbolic Names for Innovator Links

For models under version management, multiple symbolic names can be assigned, under which the respective model version can be addressed via the Innovator link protocol (inno:). Symbolic names can be specified for model versions in the administration program. The name marked as "Standard" is used by the model editor when creating documentation or links (e.g. by copying). If a standalone model is set under version control, then the name under which the model was previously available is automatically set as the symbolic name.

#### ArchiMate: Support for Version 3.1

Innovator supports the modeling of enterprise architectures based on the current ArchiMate 3.1 standard.

#### Directories of Log Files

Innovator logs all its servers and services. For the storage of the log files, information of the environment variables is used.

The following placeholders are used for the directories listed below.

<INOHOST>: the set INOHOST

<INOLIC>: directory of the license file

<VERS>: current version number

<LICPORT>: INOHOST port

<BUSPORT>: INOHOST port+1 (fix)

<HUBPORT>: INOHOST port+2 (fixed)

<AGENTPORT>: INOHOST port+3 (can be set)

Directories and file names of the log files:

- License server
  - Protocol path: <INOLIC>\logs\inolsrv.<LICPORT>
  - Log file: inolsrv.log
- Bus
  - Protocol path: <INOLIC>\logs\inobus.<BUSPORT>
  - Log files: bus.log, bus.pid
- Hub
  - Protocol path: <INOLIC>\logs\inohub.<HUBPORT>
  - Log files: hub.log, hub.pid
- Agent
  - Protocol path: <INOPRJ>\ivm\logs\inoagent.<hostname>.<AGENTPORT>
  - Log files: agent.sqlite, agent.log, stdout.txt, stderr.txt
- Model server
  - Protocol path: <INOPRJ>\<Repository name>
  - Log file: srv.log
- Service log files for error messages when starting a service
  - C:\Windows\logs\innovator\<VERS>\<Service name>.log

Attention:

- If INOPRJ is changed after installing an agent service in the system environment, then the agent service must be uninstalled and reinstalled, because INOPRJ is set for the agent during the service installation (Inoagent does not read the registry).
- If INOLIC (or INODIR, if INOLIC is not set) is changed after installing an bus service in the system environment, then the bus service must be uninstalled and reinstalled because the log directory for the bus is specified during service installation (Inobus does not read the registry).

#### **DMN: Full Support for Version 1.1**

Innovator fully supports decision modeling at conformance level 3 based on the current DMN 1.1 standard. New graphical editors allow all structural elements and function expressions to be edited in FEEL notation. Import and export of standard-compliant XML files are also possible.

#### **DMN: Value Range of Input Changed**

In decision tables, the range of values supplied for selection for input columns depends among other things on the referenced type. If it is a primitive type of the Boolean Term type, the value range returns the logical values "true" and "false".

During migration, the previous value "yes" is converted to "true" and "no" to "false" for these cases.

#### **Java VM Settings Support Environment Variables**

You can also use environment variables in the inojvm.ini file to specify the Java installation to be used. Double quotation marks in the path specification are ignored.

Examples:

JRE Home=\$(ProgramFiles)\Java\jre<Java version>

JRE Home=\$(INODIR)\java\jre<Java version>

#### Migration of Join Options at From Clauses

The transformation step from 13.1 to 13.2 transfers the database options 'INNER JOIN', 'LEFT OUTER JOIN', 'RIGHT OUTER JOIN', and 'FULL OUTER JOIN' from From clauses to the property 'Join Kind'.

#### Version Management: Improved Operational Stability

The provision of models under version management has been made more tolerant and stable to influences from IT operations. If a computer with running model versions is shut down, the agent shuts down the Innovator environment in a controlled manner. When the computer is restarted, the environment is restored to its previous state. A user with the "Manage Environment" permission can also directly shut down all running models under version management on a computer using a single command.

#### Migration: Cleanup of the File inofont.ini

In the shipped inofont.ini (INODIR) there are no more entries for Motif or PostScript because they are not used.

## Administration

#### ■ Model Server: Shut Down Automatically When Inactive

In the administration program, you can specify the duration of inactivity after which a (standalone or managed) model server shuts down automatically.

#### ■ User Configuration: Right for Environment Administrators

In the central user administration, groups and users can be assigned the "Manage Environment" right in the properties in order to perform special administration tasks. This includes recovering defective models, restoring backups and any physical deletion of model versions, backups and repair data. The environment right also includes version management rights for all models.

#### ■ Version Management: Authorizations for Model Versions

The rights to use model versions are managed in the central user administration. When creating login or model administrator rules, a distinction is made between models under version management (managed models) and models on standalone model servers. The rules are displayed separately and do not overlap. For managed models, the separate rule type version rule replaces the model server rule (with '#'). A version rule allows, among other things, the creation and start of model versions or the editing of their properties.

#### ■ Version Management: Creating New Managed Models

In the "Version" menu ribbon, the "New Model" command can be used to create a model in its first version directly in version management. Some additional functions are available in the wizard for new models. This option is also available in the backstage view "New" in the model editor.

### ■ **Version Management: Defining Resource Restrictions**

For each agent, in the View menu ribbon you can set a property how much free main memory (RAM) must be available on the agent's machine in order to start another model version. This is to ensure that the machine's resources are not exceeded by too many active managed model servers.

Editing the properties of an agent requires the Manage Environment right.

### ■ **Version Management: Model Recovery via the Administration Program**

If a model version is terminated unexpectedly, a recovery can be performed using the administration program. This command requires the "Manage Environment" right. If a recovery is not successful, the model data can be downloaded and later replaced by corresponding repair data from MID Support.

### ■ **Version Management: Restore Backup**

The "Environment>Backup>Replace Model Version" command replaces the current model version with the selected backup. This resets the managed model to the status of the selected backup. The model version and backup must not be active. The command requires the "Manage Environment" right.

### **Login not Needed for Single Sign-on**

In the standard single sign-on mode, no explicit login to model servers or model versions is required. The rights required for functions and commands are automatically synchronized with the user administration during execution. The tooltips of the commands indicate in detail which rights are required for execution.

### **Performance: Faster Listing of Models in the Administration Program**

The administration program displays all running models to which you can log on, instead of only those that you can administrate. The eliminated check leads to a significant performance gain when a large number of repositories is running.

### **User Configuration: Rules for System Accounts (Windows)**

In the user administration, the rules for the local system account (nt authority, system) that apply equally to all computers must be replaced by rules for individual computer accounts if corresponding services are used. The account to be used must match the computer name followed by "\$". The domain must be the same as that used by Innovator users.

### **Using Group Managed Windows Service Accounts**

In the administration program, services under Windows can optionally be entered using "Group Managed Service Accounts". Under Windows, a group managed service account allows a better assignment of access rights for individual programs, e.g. to network shares and other resources. Group managed service accounts are available from Microsoft Windows Server 2008R2.

### **Version Management: Place Models Under Version Control**

In the administration program, standalone models can be versioned with the command "Model>Place Under Version Control". An existing model of a standalone model server can either be created as the first model version of a new model or be added to the hierarchy of an existing managed model as an additional model version. Existing external references (Innovator links) to the model remain valid.

**Determination of the Domain with LDAP Synchronization**

When synchronizing the user configuration with the LDAP server, the domain determination has been improved to avoid the "A referral was returned from the server" error.

**Export of Large Models Possible**

In the administration program it is possible to export models over 2 GB size to a repository.

**Project License Servers: Starting Previous Versions**

Project license servers of versions prior to 13.2 can be started without problems even if the LDAP filters are more than 256 characters long.

**Server Installation Allows Creating New Models in the Administration Program**

The Server installation type allows new models to be created in the Administration program.

**Set Up Model Server as Service with Absolute Path via Command Line**

A model server can be installed as a service in two ways:

1. via the interface of the administration program
2. via command line (inosrv /bi <repository name>)

The absolute path can also be specified in the repository name when installing from the command line.

**Version Management: Java Compatibility of Agents (inoagent)**

Agents (inoagent) are compatible with Java versions also higher than Java 8.

## Model Editor

■ **ArchiMate: Import and Export of the Enterprise Architecture Model**

The enterprise architecture model can be exported to or imported from other tools in the ArchiMate Model Exchange file format using commands in the Extras ribbon.

This requires the use of the newest model template "Enterprise Architect for ArchiMate".

■ **Beamer: Using Different Sorting Methods**

In the "Beamer" tool window the drop-down list "Sorting" in the upper group provides the possibility to display the trees as in the model structure or alphabetically sorted. The setting is saved for the user.

■ **BI: Manual Assignment of DB Views to Hierarchy Levels**

If an entity or a view is assigned to a hierarchy level, it has a property 'DB elements'. In the corresponding dialog, you can choose which view or table is used to convert the hierarchy in the respective DB models. When you remove the assignment of an entity or a view to a dimension level, the corresponding obsolete mapping dependencies are deleted.

## ■ BI: Variants of the Measures Dimensions Diagram

A measure dimension diagram with many hierarchies used by the cube can be displayed more compactly in two ways.

- Dimensions and hierarchies can be converted from nodes with node contents to nodes with compartments and vice versa.  
To do this, use the commands in the context menu "Convert".
- The use of a hierarchy can be displayed directly as an edge between the cube and the hierarchy without a dimension level. The prerequisite is that the hierarchy has only one occurrence in the diagram and the associated cube dimension none at all.  
To do this, remove the node of the dimension level. You can then make the edge visible.

## ■ DMN: Simulation of Decisions

To have the modeled DMN decision graphs checked for logical and syntactical correctness, you can use the Simulation feature. You use test cases and test data to simulate decisions. To create the underlying test data records, select the information (DMN: InputData) in the structure content or the diagram and call the command "Create template for DMN test data" in the context menu. In an external editor, a template for a test data record is offered, whereby each test data record in this file should have a unique name.

With the command "Design>Validation>Simulate" in the dialog "Simulate" you can create different test cases from these test data records by combining them, which form the basis of a simulation of the decision. The simulation can be started with a test case in the same dialog. Alternatively, an already created test case can be called via the drop-down list of the command. The simulation results are displayed in a dialog box.

## ■ Excel: Import and Export of Model Elements and their Properties

You can export any model elements and their visible properties to Microsoft Excel. All element types that can be displayed in Excel are supported during import, such as data structures, requirements, concepts, but no diagrams. You can change and create model elements by importing a suitable Excel file.

## ■ Sorting Options in the Structure Content

The structure content in the Model Tree tool window view can be displayed alphabetically, alphabetically descending, and according to the user-defined sort order.

## ■ Synchronize Model Elements: Automatically Select the Determination Method of the Reference Time

The reference time for the merge comparison can be determined differently according to the comparison objects. The selection can be made automatically, too.

Innovator then automatically activates the radio button "Determined from change sets" when adding change sets to the comparison set, for example.

The priorities of the automatic selection are:

1. "Determined from change sets"
2. "Version creation time"
3. "Determined from root elements"

If no automatic selection is made, the user's selection is retained.

### ■ SysML Model Templates and Add-ons

Innovator for Software Architects provides model templates, add-ons and a demo model for the Systems Modeling Language (OMG SysML).

### ■ Version Management: Creating and Managing of Model Versions

In the model editor, there is a backstage view "Version Management" for managed models. It allows the basic management of all versions of the model, i.e. the creation of new versions and backups as well as the starting and stopping of existing model versions. The user needs a corresponding version rule.

### ■ Whiteboard Diagram: Further Reference Edges

In the whiteboard diagram, additional reference edges between nodes in different diagrams can be displayed. For a behavior, the reference edge can be mapped to the behavior-specific classifier or to the specification. For the lifeline in the sequence diagram, the reference edge for the represented attribute can be displayed.

### BI: Create Dimension Attribute Directly for Matching Attribute

If a dimension layer is linked to an entity or view, the dialog for linking to an attribute is opened immediately in the diagram when a dimension attribute is created.

### BI: Dimension Export and Dimension Maintenance with Formatting and Without Verifications

When exporting dimensions, it is possible to define formatting per type system and per type class. The values of the dimension elements are then formatted in the SQL file according to the data type of the type class.

Dimension export and dimension maintenance do not require verifications. The affected verifications include the verification of the short name of the entity and the entity attributes, the stereotype of the entity, and the foreign key integrity.

Both can be done using the DimensionSettings.xml file in the INOHOME or INODIR directory. After the dimension export has been invoked, the file with default values exists in the INOHOME directory. The entries in the file can be changed. The file can be made available to all users in the common INODIR working group directory, as long as no file with the same name exists in the INOHOME directory.

### BPMN Import of Diagrams from Process Note

The BPMN import supports only import from files. To import diagrams from Process Note into Innovator, use the functionality in previous Innovator versions.

### Change Sets: Log Size Not Used as Deletion Criterion

In the properties of change logging, the log size is no longer used as a deletion condition. It is deleted during the transformation.

### Composite Structure Diagram: Creating Connectors

A communication connection can be established from a port on a part to a port on the surrounding part.

#### **Concept Diagram: Concept Connections Permitted as Target of Concept Connections**

In concept diagrams, concept links can be created between two concepts or from a concept to an existing concept link. The prerequisite for this is that the concept link is allowed as a target in the configuration. A concept link can only be used as a target if it runs between two concepts. The resulting edge is always connected directly.

The included configuration allows associations, compositions and aggregations according to the ArchiMate specification.

#### **Concept Diagram: Separate Directional Markers of Concept Connection**

For concept connections, the properties "Direction Indicator" and "Direction Indicator Style" are replaced by the properties "Arrow Type (Source)" and "ArrowType (Target)", which also allow the value "None". This allows the directional marking to be set separately for both sides.

In the context of migration, the values of "Direction Indicator" and "Direction Indicator Style" are being transferred to the new properties.

For the property "Edge Style" the value "Standard" has been renamed to "Solid".

#### **DB Import: Importing or Exporting Connections**

The settings of one or more selected database connections can be saved to a file or read from a file via the menu in the dialog or via the context menu.

#### **DB Mapping Without Error Message When Missing Access Right**

The mapping does not, in accordance with its other behavior, return an error on attempts to change a model element or to create a mapping dependency when an access right is missing.

The model element remains unchanged and the mapping dependency is not created.

#### **Display of "is Protected" Changed**

The package property "is Protected" can be activated by any user with access rights, but can then only be viewed and changed by users with configuration rights.

#### **Improved Creation of the Uniqueness of Short Names**

In the configuration, you can assign short name restrictions for various element types (such as entity attributes). The restriction of uniqueness is evaluated when elements of the affected element type are created and copied. A short name is never created from an empty short name. The system therefore only evaluates if the creation template or copied element already has a short name.

#### **Model Selection on Opening and Logging In**

"Standalone Models" and "Managed Models" (with version control) are displayed in a mutual tree both on the backstage "Open & Login" under "Model Servers in the Network" and in the dialog "Change model" of the editor "Synchronize Model Elements".

#### **Modeling Steps are Removed**

The change propagation feature including the modeling steps has been removed from the "Change Set" document window, because it has become both superfluous and incompatible with the merge and version management.

**Performance: Improved Start Time and Main Memory Consumption of the Client**

During the installation of the Innovator clients, native images are now created. The installation takes significantly longer, but results in a much faster application startup time and lower memory requirements.

 **Search with Operator 'contains'**

The search for properties, labels and type system options with the operator 'contains' treats '\*' as content of the search value (not as wildcard).

 **Show Name of Connected Elements in Properties**

If exactly one element is assigned to a property in the "Properties" tool window that can have an element set as value, its name is displayed as value.

 **Specification Texts: Cancel Possible**

In the dialog box asking whether textual changes are to be saved when closing a specification text, there is an additional option to cancel the closing process. The same applies to I/O specifications in Business Intelligence area.

 **Standard Layout of Dependencies Between Contents of a Structure Definition or Classifier Node Improved**

The standard layout draws dependencies between contents of the same structure definition or classifier node on the right outside instead of inside the node and thus makes a selection possible. Both creating and reconnecting dependencies is therefore possible in the diagram.

 **Structure Diagram: Moving Elements**

In the structure diagram, structure groups and structure elements can be moved within a structure definition as well as to another structure definition.

 **Table Editor "Contents as Table": Register Labeling**

Each tab of the Table Editor "Contents as Table" is labeled with the type of content and the name of the selected table view.

 **Verification of Access Rights during Renaming**

To rename an element, the access rights to this element must exist and the element must not be locked by any other user. Access rights to all diagrams with an occurrence of this element are not required.

 **"Change Element Type" Wizard Suppresses Card Switching When Errors Occur**

If an error occurs in the "Change Element Type" wizard when changing the card to the next step, the card is not changed.

 **Activity Diagram: Object Flow Does Not End Server On Undo**

In activity diagrams, object flows can be linked with pins of structured activity nodes. The direction of the object flow and the position of the involved action decide which pins are valid source and target for the object flow. Subsequently changing the structuring may cause such an object flow to be incorrectly modeled, which

then will be reported by the corresponding verification option [VFY305]. If you delete an incorrect object flow or the involved structured activity node, the undo will be rejected with an error message.

**Attachments in Copied Models of the Same Repository**

File attachments from models copied within a repository can be edited in parallel.

**BPMN Import Can Import Into Components**

When importing BPMN, a model element of the element type "Component" can also be used as a structural element that serves as storage location in the model.

**BPMN: Drag and Drop a Component onto a Data Object**

As with classes, components can also be dragged and dropped onto a data object. This will assign a business object to the data object, for which the component is the data structure.

**BPMN: SVG Export of Data Objects in Process Diagrams**

When exporting BPMN diagrams to SVG, data objects are displayed correctly.

**Concept Diagram: Split Concept Connections for Nested Concept Nodes**

The connector nodes of split concept connection edges in the concept diagram are positioned in the container of the connected concept node. Even with deeper nesting, opening the diagram does not terminate the server.

**Convert to Model File**

With the command "Convert to model file..." all contents of the model are exported into a new model file in such a way that the model can be started even when redefined stereotypes are used.

**DB Element Selection When Scrolling**

If the lists in the DB Manager contain a large number of elements, the selection of the elements is not changed by scrolling.

**DB Mapping a Foreign Key in Multiple Merged Tables**

Mapping from the conceptual model to the database schema maps a foreign key correctly, even if the foreign key columns are located in a database table that has been multiply merged to the unique key. In this case, a foreign key will be generated for each instance of the merge, and the table columns of the instance will be included in the foreign key of the corresponding instance.

**DB Mapping Adjusts Null Property**

When mapping from the conceptual model to the database schema, the null property of the corresponding table column is set to "null" for an entity attribute with the Null property "null", as long as the database table of the entity has not been merged to the primary key or the table column points to a merged table.

**DB Mapping Merges Local Table Columns**

Manual mapping from the conceptual model to the database schema merges local table columns whose entity attributes have been merged. The prerequisite is that all merged table columns are local, that is, they are not used in any key.

**DB Mapping: Configuration Loader Creates Engineering Action**

The configuration loader creates the engineering action for the configuration of the mapping between the object-oriented and the conceptual model with the correct location ("de.mid.innovator.m2msdk.application.ConfigurationApplication").

**DB Mapping: Configuration Loader Creates Engineering Action**

The configuration loader creates the engineering action for the configuration of the mapping between the object-oriented and the conceptual model with the correct location ("de.mid.innovator.m2msdk.application.ConfigurationApplication").

**DB Mapping: Cross-model Mapping Between Conceptual and Object-Oriented Models Improved**

Cross-model mapping between a conceptual and an object-oriented model is also possible between models running on the same model server when no "User Name in Model" is specified in the user administration.

**DB Mapping: Maintenance of Null Property of External Attributes when Changing a Relationship**

If the 'Optional or Mandatory' property of a relationship changes from 'Mandatory' to 'Optional', then the Null property of the attributes of the associated foreign key is set to 'null'. Exceptions are the attributes for which 'not null' is required because of the participation in another foreign key.

**DDL Export for Microsoft SQL Server Places COLLATE Correctly**

When exporting as a DDL file, COLLATE is placed correctly and precedes (NOT) NULL for table columns.

**DDL Export for Microsoft SQL Server: ForeignKey/Trigger Statements**

With the argument "foreignkeyandTriggersPositioning=EachTableEnd" the ForeignKey/Trigger statements can be output per table at the end of the statements rather than in a separate file (default "foreignkeyandTriggersPositioning=AfterAllTables").

**DDL Export: From Clause Statements in Union Views Create Namespaces**

The options "Generate Catalog or Schema Name in From Clauses of Views" in the "Namespace" group of DDL configurations also generate namespaces for From Clause statements of Union views.

**DDL Import Creates View Columns for Union View**

Importing a DDL file creates its own view columns on a Union View, so that it is not necessary to edit the database view afterwards with "Design>Adjust>Adjust Select Statements".

**Duplicate Descriptions After Changing the Element Type Are Prevented**

You can change some element types. To do this, you must select a creation template. This creation template is used to create the new element and creates all the content defined in it (for example, labels or specifications). All connections from the converted element are then reassigned to the new element. For labels, specifications, and stereotype properties, the values are only copied if they have not already been created using the creation template.

#### **Import of Model Fragments**

When importing a model fragment, elements can be found via UUID and thus change their owner. The found element can have an owner before the import that is not contained in the model fragment. Sub-elements of these elements are deleted during import if they are not contained in the model fragment.

#### **Instances of Associations Can Be Copied Only to a Limited Extent**

When copying elements, references to not selected elements are adopted in addition to the explicitly selected elements. When pasting, these references are to be mapped to the existing elements. Conformity of the existing elements with the originally referenced elements is ensured. In the case of associations, e.g. the types of the association roles must match. Similarly, the instances of types must match for instances of associations. Otherwise the association or instance will be rejected as incompatible.

#### **Merging of Aggregations Corrected**

When aggregations are merged, the sequence of the involved roles is, even in unfavorable cases, not changed; a resulting consistency error is avoided.

#### **Order When Moving Entries in Compartments**

If several entries in compartments are moved at the same time, the order of the entries remains the same.

#### **Specification Texts: Treatment of Text Indentations**

In specification texts, text indentations with non-integer pixel specifications, which can be created by copying and pasting from other applications, are rounded to integers to ensure correct rendering.

#### **Structure Elements Show Properties for "Used Features"**

For structure definition, structure group, and structure entry, the Used Features property is visible in the Properties tool window.

#### **Synchronize Model Elements: Algorithm for Determining the Comparison Pairs**

When comparing and merging, the system automatically determines which elements from the left and right models are combined as comparison pairs. The corresponding algorithm has been adapted so that even if an element is renamed and a new element is created with the old name of the renamed element, the correct elements are recognized as counterparts.

#### **Synchronize Model Elements: Merging Views on Processes Without Definition Rejected**

Each process can be displayed in any number of views. One of these views is marked as a definition. When you merge views on a process, the system ensures that the process in the target model either already has a definition or that the definition is copied during the merge. Otherwise, all views of the affected process are rejected. If this would result in an empty process definition diagram, this is also rejected with an error message.

#### **Synchronizing Model Elements: Displaying of Edges when Merging Diagrams**

When merging diagrams, the edge support points are adopted in correct order. An incorrect order may have caused a diagram to be interpreted as identical, but the display of the edges could have shown clearly visible differences.

### Undo after Creating a New Process View

When you create a process, a separate process diagram with a view of this process usually results. If there is no other view of the process, the view in the process diagram is automatically marked as process definition. If you drag this process into another BPMN diagram, a new view will be created there. If there is no view apart from the view in the process diagram, the new view is marked as process definition and the view in the process diagram becomes a maintained copy of this process definition. The Undo function deletes the newly created view and marks the view in the process diagram as process definition again.

### Visibility of Stereotype Properties

Only the stereotype properties marked as visible are displayed in the Properties tool window. Values of non-visible stereotype properties that have been set by plugins are not displayed.

## Configuration

### ■ Customize Display Options in Extension Profiles

In the Display Options dialog, the settings from imported profiles are also displayed. The imported rules are displayed in the tree with the name of the profile. The imported settings can be marked as ignored. The settings marked in this way are then treated as if the setting had not been created. In the tree, the ignored rules are marked with the icon "no entry". If a rule is marked as ignored, then all nested rules and all display options contained in them are automatically marked as ignored.

Rules can be copied inside the dialog.

### ■ Documentation: Additional Information Chapters

For all elements with lower and upper limits there are the additional information chapters "is optional" and "is more than one possible" in the configuration of the documentation.

### ■ Information Structure of Details Tool Window is extendable

The configuration of the information structure in the tool window "Details" is no longer part of the model data, but can be managed in profiles. This allows these detail area configurations to be easily extended and distributed. Existing detail area configurations can be used in enhancement profiles using the familiar profile import mechanism. Here it is also possible to ignore imported detail configurations or to enhance them with your own configuration settings. The display in the tool window "Details" is calculated from the profiles imported at this point.

Further important information for users of earlier Innovator versions can be found in the Migration Manual (see Help "Browse" or PDF in the Customer Area).

### Display Options of Diagrams Cleared

For all diagrams, display options can be defined in the configuration for the diagram itself as well as for all nodes and edges and their nested contents. The list for the diagram itself is limited to several font and color settings that are used by all elements of the diagram. Additional display options are not supported for the diagram and are removed during migration.

#### Documentation: Handling Tables

If a new subchapter is created for a table in the documentation configuration, this chapter is automatically assigned to the table as a new table column. Hierarchical chapters and images are not allowed as table columns. Therefore, these chapters are not available for creating tables. The assignment as a table column also takes place if the subchapters are created by copying and pasting.

When you create a table, a percentage column width is assigned for all table columns. For all added columns, the column width is calculated according to the same procedure as for creating the table. All new columns have this calculated width. The width of the existing columns is converted to the new width according to the previous percentage ratio.

If a subchapter used as a table column is marked as hidden, the corresponding table column is not output when the documentation is generated.

#### JavaPackage Allows Additional Characters in Name

The names of Java packages allow additionally '\_' and '\$', even at the beginning of the name.

#### Properties Combined in Documentation Structures

In the "Documentation Structures" view of the configuration editor, the chapter properties of the "Paragraph Format" and "Graphic Properties" tabs have been moved to the "Properties" tab. The list of available properties depends on the chapter and the intended use. For example, neither paragraph format nor numbering are offered for chapters within the "Details" configuration.

#### Documentation: Postprocessing Command Supports Spaces in Path

Documentation generation supports postprocessing commands (e.g. inojava), even if the executable programs are located in a directory with a blank character in the name.

---

Copyright © 2019 MID GmbH

For questions, please contact our hotline. Tel.: +49 911 96836-222, E-mail: support@mid.de.

You can also find this document along with others in the exclusive log-in download area for Innovator service customer under <http://www.mid.de/support/>.

Your MID team

MID GmbH, Kressengartenstrasse 10, 90402 Nuremberg, Germany

Tel.: +49 911 96836-0, Fax: +49 911 96836-100, E-mail: info@mid.de, Internet: <http://www.mid.de>