MID
the modeling company

Innovator 15.1

Migration Manual

innovator

Migration Manual – Innovator 15.1 – Edition September 2021

Contents

Introduction into Migrating Innovator

This chapter informs you about the purpose of the information about the migration and the basic procedure for an upgrade (version change).

Brief Description

Who Should Read This Information?

All those wishing to use models, licenses or user customizations from the previous version in Innovator version 15.1.

Who Does Not Need to Read This Information?

All those wishing to start working with Innovator 15.1 using the new models based on the current model templates and initial settings and who do not wish to use licenses from a previous version's pool.

Anyone who is simply carrying out an update to a new release within the same version (e.g. from Innovator 15.1.1 to Innovator 15.1.2; see Help, chapter "Installing an Update").

Purpose of Information

The information describes the worksteps necessary for migrating models from previous versions to Innovator 15.1. Please note the "Migration of Versions Before Innovator 15.0" section when switching from older versions than Innovator 15.0.

You only need to carry out these steps if you are migrating from Innovator version 15.0 to Innovator 15.1.

The information should protect you against the loss of information during the transformation process and when editing models in Innovator 15.1.

Migration Files

Innovator develops in such a way that changes are made to plug-ins and profiles; these make up the basis of modeling, along with configuration settings and are part of model templates. Plug-ins and changes are made available in migration profiles; you can import these as migration files in the configuration editor. Import only changes the profiles included by MID within scope of delivery for migration profiles. Customization profiles are not modified.

The `Migration*.aob` migration files can be found in the `$INODIR/config/` directory and are language independent. They can be used in both languages to update the profiles included within the scope of delivery of Innovator and used in your models.

Basic Information about Migrating Innovator Models

What is Migration?

Migrating Innovator models means moving models from the previous versions to the current version.

The migration comprises the transformation of the model data using the **Transformer** program or Power-Shell script. Model data can be found within one or more data repositories or in the managed model versions of the project directory. In exceptional cases, preparation and postprocessing are required in the models or their configuration.

Migration can also affect subsequent use of your own templates, initialization files, scripts etc. without concrete models being migrated. You may need to carry out further steps for migrating licenses.

Taking care to read information about migration can avoid loss of information during transformation and can protect Innovator models from being changed.

> **Note**
>
> Ensure that you have taken all measures to protect your models from loss of data:
>
> - You do not receive any notable verification messages in the models to be migrated
> - You have made an external back-up of the complete model data (in particular with regard to necessary transformation preparations in the models of the previous version)

Smooth transfer from Innovator 15.0 to Innovator 15.1 is ensured for all models.

Migration of Versions Before Innovator 15.0

Transformation from versions older than Innovator 15.0 may have to be carried out in several migration steps due to the specific preparations and post-processing that may be necessary for earlier version changes. We're happy to help you with this.

A migration manual always provides a detailed description of the migration from the last version and the version prior to that.

The following table indicates whether you need to carry out specific preparation or Postprocessing for the individual **version changes from the relevant predecessor version** in addition to the work that is normally required and which products are affected. For details, please see the migration manual for the respective Innovator version, which is linked in each case.

You require the **specifications for the version subsequent to your current version** in the following table when migrating to the current version. You are not migrating onto your current version, do not forget.

Examples

- If you are migrating **from Version 12.3** to the current version and use Innovator for Information Architects for this, you must e.g. carry out the relevant postprocessing as of **line 13.0**. This encompasses the two general postprocessings and the special Innovator for Information Architects postprocessing in the Version 13.2 line, the three general postprocessings and special Innovator for Information Architects postprocessing in the line 14.0 line, the general postprocessing in line 14.1, the special postprocessing for Innovator for Information Architects in 14.2 and the three general postprocessings in line 14.3.
- If you are migrating **from Version 14.0** to the current version and use Enterprise Edition, you must e.g. carry out the postprocessing **from line 14.1**.

The appropriate preparations must also be taken into consideration.

Overview of the Preparations and Postprocessing for Transformation from Version 11 R5

Migration to version	Preparation (with product reference)	Postprocessing (with product reference)
12.0		Innovator for Information Architects • Customizing Engineering Actions for Mapping/Renaming for DB2 (Updated description for Version 12.3.; can be carried out from Version 12.0 to Version 15.1.)
12.1	Innovator for Business Analysts • Merging Stereotypes (Can be carried out from Version 11 R5.)	General • Using Execution Rights as Access Rights (Can be carried out up to Version 15.1.)
12.2	General • Adopting Model Language (Can be carried out from Version 11 R5.) • Adoption of Display Languages (Can be carried out from Version 11 R5.)	
12.3		Innovator for Information Architects • Model Stereotype for Conceptual Model (Can be carried out up to Version 13.3 as described.) • Profiles for Database Type Systems (Can be carried out up to Version 15.1.)
13.0		
13.1		Innovator for Enterprise Architects • Migration to ArchiMate® 3 (Can be carried out up to Version 14.0 as described.)

Overview of the Preparations and Postprocessing for Transformation from Version 11 R5

Migration to version	Preparation (with product reference)	Postprocessing (with product reference)
13.2		General • Model Administrator Rules (Can be carried out up to Version 15.1.) • Model Version UUIDs (Can be carried out up to Version 13.3 as described.) Innovator for Information Architects • Your Own DDL Generation (Can be carried out up to Version 15.1.) Innovator for Enterprise Architects • ArchiMate® Data Values (Can be carried out up to Version 15.1.)
13.3		
14.0	General • Stereotypes not visible in Properties • Preparing Menus for Merge (Can be carried out from Version 11 R5.) Innovator for Information Architects • Migration of Database Options (Can be carried out from Version 11 R5.)	General • If applicable, Visibility of Stereotypes in Properties (Can be carried out up to Version 15.1.) • Removing Duplicate Menu Commands in the Configuration (Can be carried out up to Version 15.1.) • If applicable, Visibility of Relationship Properties (Can be carried out up to Version 15.1.) Innovator for Information Architects • If applicable, Enumeration Correction (Can be carried out up to Version 15.1.)
14.1		General • Migrating Configuration from the Details Area (Can be carried out up to Version 15.1.)
14.2	General • Migration: guest logins are prevented as standard (Applies to all transformations from versions 14.1 and older) (Can be carried out up to Version 15.1.)	Innovator for Information Architects • Mapping ER - DB and DB (Can be carried out up to Version 15.1.) Innovator for Business Analysts • BPMN: Assigning Event Definitions to Events (Can be carried out up to Version 15.1.)

Migration to version	Preparation (with product reference)	Postprocessing (with product reference)
14.3	General • Changing to UTF-8 and Adopting Passwords (Applies to all transformations from versions 14.1 and older) (Can be carried out up to Version 15.1.) • Naming Conventions for Model Names (Applies to all transformations from versions 14.1 and older) (Can be carried out up to Version 15.1.)	General • Agent Settings for Managed Models (Can be carried out up to Version 15.1.) • Quickly Creating Impact Analysis Diagrams (Can be carried out up to Version 15.1.) • Concept Relationship in the Whiteboard Diagram (Can be carried out up to Version 15.1.) Innovator for Business Analysts • BPMN: Assigning Boundary Events to a Task (Can be carried out up to Version 15.1.)
15.0		General • Environment Variable for Individual Symbol Path (Can be carried out up to Version 15.1.) Innovator for Enterprise Architects • Models based on "Enterprise Architect for ArchiMate" (Use import option from third-party tools). (Can be carried out up to Version 15.1.) Plug-ins • Custom Plug-ins: Loading Mechanism Changed (Can be carried out up to Version 15.1.)

How to Proceed With the Upgrade

As standard, proceed with the following steps during an upgrade in order to keep the interruption as short as possible for running projects:

> **Note**
>
> Please note that a bus is also required as a central communication component for every license server as of version 14.0. A bus is assigned to precisely one license server.
>
> The bus is implicitly started if a license server is not started as a service. The bus must also be started as a service if the license server is started as a service.

1. Set a new port number area for Innovator 15.1 and deactivate the firewall if necessary.

   (see Installation Manual or Help, chapter "Managing the Architecture", sections "Server and Port Ranges" and "Port Range for Main License Servers and Project License Servers")

2. Install Innovator in a new, empty directory on the computer which the main license server is to run on, preferably in parallel with the existing installation directory.

   (see Installation Manual or Help, chapter "Installation")

3. You determine the computer ID using a command and use an upgrade order to request the license key to activate your licenses for the new Innovator version of MID.

   If you are also carrying out a license extension as part of the upgrade order, you receive a new license repository as well as the license key (`inolic.lr`).

   (see "Determining the Host ID Beforehand with a Command and Sending it Off")

4. Once you have received the license key and, if applicable, an extended license repository, shut down the main license server of the previous Innovator release in order to place your license repository into a consistent state. User logins are not possible, but users can continue to work in the models.

   (see Help, chapter "Starting and Closing License Servers")

   Copy your license repository (`inolic.lr`) from the previous Innovator release to the license directory of the new release. This is normally `$INOLIC`, otherwise it will be `$INODIR`.

5. To retain the data of the central license repository as well as central user management with the login rules, configured projects (internal and external project servers), the license administrator password, and the serial numbers, proceed as follows.

   Start the main license server and the administration program of Innovator 15.1.

   - Change the license key in the administration program to be able to use your previous licenses and other constellations in the new version.

   - If you have received an extended license repository, then load this license file in the administration program so that you can use your extended or modified licenses in the new version.

   > **Attention**
   > ⚠ Never copy this license file directly to the `$INOLIC` directory, since doing so can overwrite your current constellations.

   The license repository is updated to the main license server computer. Innovator is now ready for use with all products you have purchased.

   (see "Activating your Licenses on the Main License Server")

6. If you have not yet set up an internal project license server for the projects of the previous Innovator release, you can do this now in the administration program. Use the port number used up to now and assign the required licenses.

   (see Help, chapter "Structuring Licenses with Internal Projects")

7. You adjust any set up services by switching them to the project license server.

   (see Help, chapter "Executing Servers as Windows Services")

8. Start the project license server. User logins to the project models are then possible again.

   (see Help, chapter "Starting and Closing License Servers")

9. Transform your own model templates so you can use them again.

   (see "Adopting Your Own Base and Add-On Templates")

10. You can create a back-up copy for projects that you wish to continue working with in 15.1 Innovator and then carry out the preparatory work that is required in exceptional cases.

    (see " Transformation Preparations")

11. For projects that you wish to continue working with in Innovator 15.1, shut down the repository's model server and the model versions so that you can transform the data repository and/or model versions.

    (see Help, chapter "Starting and Closing Model Servers")

12. You migrate the projects that you wish to continue to use with the new functions of Innovator 15.1. You can start new projects in Innovator 15.1.

    (see " Migrating Repositories and Model Versions")

13. Ensure that directories used by the new version in the network are updated, e.g. the $INOPRJ project path contains the current model templates in the /config directory.

14. You must start the model server for the migrated projects to be able to edit the models.

    (see Help, chapter "Starting and Closing Model Servers")

15. Load the configuration plug-ins for the migrated models in Innovator 15.1.

    (see "Loading Configuration Extensions")

16. You carry out the postprocessing work that is required in exceptional cases for the transformed models in Innovator 15.1.

    (see "Postprocessing after Transformation")

17. You transfer further data that you want to use in the new version. This can include your own icons, checks, engineering actions, plug-ins and user-specific settings.

    (see "Migration of Further User Data")

> **Note**
>
> If the above recommendations fail to provide the required support or problems occur, please contact the MID hotline.
>
> E-mail: support@mid.de or phone: +49 911 96836-222 (9 a.m. to 5 p.m.).

Check List - Roles and Tasks for Migration

Migration Check Lists

Task	Leading Role	Optional[1]	Done
Install Innovator with new port range on the computer the main license server will be running on and **determine host identification**	Innovator administrator		[ ]
Upgrade order to receive a license key for releasing the licenses in the new version and to receive a new license repository for license modification or extensions, if required.	License manager		[ ]
Receive license key and MID license repository, if necessary	License manager		[ ]
Shut down main license server from previous Innovator release, then **copy** its **license repository** in the new Innovator release's license directory.	Innovator administrator		[ ]
Start main license server and administration program for the new version and **activate licenses on the main license server** to make Innovator ready for use in the new version	Innovator administrator		[ ]
If required, set up internal **project license server for projects from previous Innovator releases** and migrate established services to the project license server	Innovator administrator		[ ]
Start project license server for the projects from previous Innovator releases running there and use them as normal	Innovator administrator		[ ]
Transform your own **model templates**	Innovator administrator or project manager		[ ]
Update the **directories** in the network used by the new version (e.g. the `$INOPRJ` repository path must contain the current model templates in the `/config` directory)	Innovator administrator		[ ]

[1]Optional means that this step can also be carried out for the next migration.

Task	Leading Role	Optional[1]	Done
Migrate projects that should be continued in the new version: • Create a back-up copy of model data before carrying out necessary preparatory work and then carry out the preparations in the previous Innovator version. • Download the data repository's model server and the model versions which you want to transform and then transform the data repositories and model versions. • Start the model server for migrated projects in the new Innovator version • Reload migration files with configuration extensions for new functions • If necessary, carry out necessary post-processing in the new Innovator version • Continue editing	Project manager		[ ]
If necessary, **set-up project license server for projects of the current Innovator releases** and set-up services for the project license server (services do not have to be converted upon upgrade)	Innovator administrator	**X**	[ ]
Add extra user information	Innovator administrator or project manager or user		[ ]

1Optional means that this step can also be carried out for the next migration.

Migration of Your Own Licenses and Templates

This chapter tells you about the main migration steps and the migration of your licenses and templates.

Main Steps

To activate your licenses:

- Install the new Innovator version on the main license server in order to determine the host ID ([see "Preparing for the Activation of Your Licenses"](#))
- Determine the host ID of the main license server and, if you have multiple main license servers, the serial number, too ([see "Determining the Host ID and Serial Number"](#)).
- With your upgrade order, request your license key from MID ([see "Determining the Host ID Beforehand with a Command and Sending it Off"](#))
- Activate your licenses with the license key received from MID GmbH ([see "Activating your Licenses on the Main License Server"](#))

To transfer your own model templates and add-ons:

> **Note**
>
> The creation and maintenance of a separate "master repository" are recommended for the management of your model templates and add-ons. You merely need to transform this master repository in order to be able to use your own model templates and add-ons in the new version as normal after saving them.

- If you do not yet use a master repository, create a master repository prior to the transformation in the predecessor version that contains your self-defined specifications for Innovator models ([see "Adopting Your Own Base and Add-On Templates"](#))
- Transform the Master repository ([see "Transforming with the Transformer"](#))
- Export the model templates and add-ons you defined from your transformed master models into configuration files in Innovator 15.1 ([see "Adopting Your Own Base and Add-On Templates"](#))

Licensing an Innovator Upgrade

What Happens with the Licenses During the Upgrade?

Innovator 15.1's license server reads the license repository of your predecessor version and reuses all of your licenses, your configured internal and external projects, central user management with the login rules and the license administrator password.

Your existing license distributions for older versions' projects are retained when licenses from your predecessor versions are adopted into Innovator 15.1.

Your licenses must be activated for Innovator 15.1 as described below before being reused.

Prerequisites for Activating Your Licenses

The prerequisites for activating your licenses for Innovator 15.1 are as follows:

- You have filled out the order form for an upgrade to Version 15.1, legally signed it and sent it to MID GmbH, preferably by fax (+49 911 96836-100).

- You have sent the order processing (Auftragsbearbeitung) department the host ID of the main license server of Version 15.1, preferably by e-mail. If you use multiple main license servers, you must also specify the serial number in order to enable a unique assignment.

> **Note**
>
> As a result of your upgrade request and to enable the reuse of your previous licenses and constellations, you receive a **license key** by e-mail, so that you can activate your licenses for the new version.
>
> If you are carrying out a license extension together with the upgrade order, you also receive a **license file** by e-mail. You use this to extend or modify the licenses.

Preparing for the Activation of Your Licenses

Context

The following description presumes that you are installing Innovator 15.1 for the first time and have not activated any licenses for this version yet.

The copying of your license repository to the new Innovator version initially enables the serial number to be automatically determined with the host ID.

How to proceed

1. Install Innovator 15.1 on the computer which the main license server should run on, in a new, empty directory, preferably in parallel with the existing installation directory.

   (see Installation Manual or Help, chapter "Installation")

2. To bring your license repository into a consistent state, please shut down the main license server of the previous Innovator version.

   User logins are not possible, but users can continue to work in the models.

3. Please copy the `inolic.lr` license repository from the `$INODIR` directory (Version 15.0) of your Innovator 15.1 installation to the `$INOLIC` directory of Innovator.

4. To enable the normal operation of your previous installation until you receive the activation data from MID, please start up the main license server of the previous Innovator version again.

Determining the Host ID and Serial Number

You cannot immediately use the copied license repository in the new Innovator installation. When you start up an application of Innovator 15.1, you initially receive the message which shows that the License server host not registered in license repository.

To be able to use the licenses, you need to determine the main license server computer's host identification and use this to get a license key or license file to activate the licenses for Innovator 15.1 from the MID order processing department.

There are two ways of determining the host identification and serial number:

A. First of all, determine the host identification via a command. This does not affect work in your previous Innovator version.

B. Determine the host identification using the administration program of Innovator 15.1. This procedure takes longer.

You should always send the information along with the upgrade order to MID order processing (mailto:auftragsbearbeitung@mid.de).

Alternatively, send the information to the address below. If you choose to use the traditional postal service, make sure that the host ID reference can be clearly read.

MID GmbH
Order Processing
Kressengartenstrasse 10
90402 Nuremberg / Germany

MID GmbH will generate a host-specific license key (if you are continuing to use previous licenses) and, if necessary, a host-specific license file (in the case of a changed license set) in accordance with your upgrade order and return them to you by e-mail (normally within one working day). You use them to switch the license of the main license server to the new Innovator version.

Determining the Host ID Beforehand with a Command and Sending it Off

Prerequisites

> ⚠️ **Attention**
>
> You need to determine the host ID in Version 15.1 and on the computer the main license server should run on.
>
> If you do not do this, host identification will be invalid and the license repository will not be accepted for the license server computer.

With the host ID, the serial number can be determined at the same time if your previous license repository is available in the license directory of the new Innovator version.

Context

You can determine the host identification **after** installing Innovator 15.1 without having to close the previous version's main license server and start Innovator 15.1. (You cannot operate both main license servers on one computer at once.) This does not interrupt your work on your projects. We therefore recommend this procedure.

How to proceed

1. Execute the `inolsrv -h` command on the main license server host in the installation directory of Innovator 15.1. (Default: `c:\Program Files\Innovator\15.1\inolsrv -h`)

   The host identification is identified and is ready to be sent by e-mail.

   The serial number is also determined if you have copied your previous license repository to the license directory of the new Innovator version.

2. If the serial number is not determined automatically and you use multiple main license servers, please determine the serial number in your previous Innovator version in the **Help & Support** backstage view in the model editor for unique assignment.

3. Send this information to the MID order processing department ([mailto:auftrags-bearbeitung@mid.de](mailto:auftragsbearbeitung@mid.de)).

   In response, you will receive (normally within one working day) a license key (for the further use of the previous licenses) or, if applicable, a license file (for a modified license set) by e-mail in accordance with your upgrade order.

Determining the Host ID with the Administration Program and Sending it Off

Prerequisites

> ⚠️ **Attention**
>
> You first need to determine the host ID in the version you want to activate the licenses for.
>
> The determination of the host ID must take place on the computer which the main license server should run on.
>
> If you do not do this, host identification will be invalid and the license repository will not be accepted for the license server computer.

With the host ID, the serial number can be determined at the same time if your previous license repository is available in the license directory of the new Innovator version.

Context

You can determine the host ID and the serial number with the **Administration Program** (`InnoAdministration.exe`) of Innovator 15.1. This version's main license server must be running for this. This can only be started if your predecessor version's main license server has already been closed. (Both main license servers cannot be used on one computer at the same time.)

Some fonts contain characters that can hardly be distinguished from each other (e.g., 1, l and I). It is therefore advisable to send the host identification immediately to MID in an e-mail.

E-mail transfer may be supported by an automatic e-mail function in your system. In that case, the only thing you need to do is to select the **E-Mail** command. This way you can easily avoid any kind of misunderstanding.

Read the "Determining Host Identification" Help chapter for this.

How to proceed

1. If you use the same main license server host, stop the main license server of your predecessor version on the main license server computer.

   You do not need to close the project license server for this.

2. Start the main license server (`inolsrv.exe`) of Innovator 15.1 on this computer.

3. Start Innovator 15.1's **Administration Program**.

4. Select **License>Main License Server>Host Identification**.

   The dialog of the same name appears. The dialog displays a unique string as the host ID of your license server computer along with the serial number.

5. To create an e-mail with the displayed information to send to the MID order processing department, click on **E-Mail**.

   An e-mail with the information from the dialog is created with your system's standard e-mail application.

6. Add information to your e-mail, particularly your sender information and send it to the MID order processing (Auftragsbearbeitung) department (mailto:auftragsbearbeitung@mid.de).

   In response, you will receive (normally within one working day) a license key (for the further use of the previous licenses) and, if applicable, a license file (for a modified license set) by e-mail in accordance with your upgrade order.

7. Alternatively, send the information to the address below. If you choose to use the traditional postal service, make sure that the host ID reference can be clearly read.

MID GmbH
Order Processing
Kressengartenstrasse 10
90402 Nuremberg / Germany

MID GmbH will generate a host-specific license key (if you are continuing to use previous licenses) and, if necessary, a host-specific license file (in the case of a changed license set) in accordance with your upgrade order and return them to you by e-mail (normally within one working day).

8. To be able to use the previous Innovator version without restriction while waiting for your activation data, close the **Administration Program** and the license server of Version 15.1 and start the main license server of the previous Innovator version again.

Activating your Licenses on the Main License Server

Context

You can use the activation data which MID sent you to change the main license server's license to the current version.

- Change the license key in the administration program to be able to use your previous licenses and other constellations in the new version.

- If you have received an extended license repository, then load this license file in the administration program so that you can use your extended or modified licenses in the new version.
Never copy this license file directly to the `$INOLIC` directory, since doing so can overwrite your current constellations.

How to proceed

1. On your main license server host, stop the main license server of your predecessor version.

   You do not need to close the project license server for this.

2. Start the main license server (`inolsrv.exe`) of Innovator 15.1 on this computer.

3. Start Innovator 15.1's **Administration Program**.

4. Log-in to the main license server with **License>Main License Server>Login** as license administrator.

   The license conversion commands become active.

5. Select **License>Main License Server>Change License Key**, copy the received license key into the relevant field in the dialog, and confirm with **OK**.

   You then have a run-capable Innovator 15.1 with the same license set.

6. If you have also received a **license file** from MID, select **License>Main License Server>Load License File**, select the received license file in the dialog, and confirm with **OK**.

   You then have a run-capable Innovator 15.1 with a modified license set.

Adopting Your Own Base and Add-On Templates

The following contains important information which you should keep in mind when using your own base and add-on templates in version 15.1.

> **Tip**
>
> As a rule, we recommend that you always maintain your own templates in a separate "master repository" and export them from there to use in your projects. Add changes resulting from projects to your master repository to update them. As of Version 14.0, you use the customization profiles for your adjustments.

> The advantage of using a master repository when upgrading is that you simply have to transform this one repository in order to be able to use your templates in the new version.

If you do not yet use a master repository for your own base and add-on templates (profiles) and want to use these from your predecessor version for models of the current version, then transform the repositories that contain these templates. You can then export the templates from the models transformed to Innovator 15.1 and transfer them to your master repository.

(You can find more information about model templates in the "Preparing and Using Model Templates" help chapter.)

Migration of Further User Data

This chapter describes how you can transfer your own plug-ins and engineering actions, user-specific data and further user data in order to use them in the new Innovator version too.

Specific Data for All Users

What is this Data?

Certain data is the same for all users or for certain user groups e.g. icons, checks or layouts. You can use the `INODIR` environment variable to define, for example, whether this data is to be stored locally (default, possibly also distributed uniformly) or centrally (In a network directory).

- **Central Settings (Parameter Files)**

  Innovator stores settings and certain files (e.g. parameter files, icons and Java files for engineering actions) in a local directory `%PROGRAMDATA%\Innovator\<Version>` as standard.

  If you use a shared workgroup directory in the network, you can synchronize the data stored there with the new version.

  Network type information corresponds to the `INODIR` environment variables.

  As a general rule, you synchronize these user-specific settings with the newly installed data in the `INODIR` directory.

  (Find out about the `INODIR` directory in the installation manual or in the help chapter Installation)

- **Your Own Icons**

  If you have specified a directory as the source for your own icons in the **Options** dialog in the **Resources** tab under **Custom Icons** for previous versions, then you can transfer this icon directory to the new version.

  The icon directory is managed as an `INOICON` environment variable from Version 15.0. Adopt your icon directory from the `DisplayOptions.xml` file into the `INOHOME` directory from the previous versions.

  The directory for your icons can have a total of four subdirectories for the individual image formats:

  - `<Resource>\default`
    You can store icons in BMP format in 16x16 pixel size here

  - `<Resource>\png_default`
    You can store icons in PNG format in 16x16 pixel size here

  - `<Resource>\png_large`
    You can store icons in PNG format in 32x32 pixel size here

  - `<Resource>\xaml`
    Icons are stored in XAML format here

  (For more information, see the help chapter "Using Your Own Icons".)

- **Your Own Verifications**

  If you have compiled your own verifications, these are managed as a part of the user-specific settings. (`inoverify.xml`)

  (For more information, see the help chapter Your Own Verifications.)

- **Your Own DDL Configurations**

  If you have used your own DDL configurations, these are managed as a part of the user-specific settings. (`ddlexport*.xml`)

  (For more information, see the help chapter ["Creating Your Own DDL Configuration"](.).)

- **Your Own Engineering Actions**

  The Java files for use in engineering actions and batch calls are located in the subdirectories of the `$INODIR/java` directory.

  (For more information, see the help chapter ["Managing Your Own Engineering Actions"](.).)

- **Your Own Searches**

  If you have created your own searches, these are managed as a part of the user-specific settings. (`inoref.xml`)

  (For more information, see the help chapter ["Using Your Own Searches"](.).)

- **Your Own Layouts for XML/HTML Documentation**

  If you have made your own layout adjustments, these are managed as a part of the user-specific settings. (`docu.css`)

  (For more information, see the help chapter ["Customizing the Layout of XML/HTML Documentation to Suit Your Corporate Identity"](.).)

- **Your Own Mapping References**

  If you have set-up your own model references for the ER-OO mapping, these are managed as XML files in the `$INODIR/java` directory's subdirectories.

  (For more information, see the help chapter ["Customizing Mapping Between the Conceptual and Object-Oriented Model"](.).)

- **Your Own Configuration of the Details Tool Window**

  If you have created your own configurations for the Details area, these are copied to the new version implicitly with the migration of the corresponding models.

  (For more information, see the help chapter ["Configuring the Details Tool Window"](.).)

- **Your Own Plug-Ins**

  Plug-ins are located in the `<Innovator Installation Directory>/Plugins` directory.

  You cannot use plug-ins from the previous Innovator version in the new version. You require the version of the plug-ins that corresponds to the Innovator version.

  (For more information, see the help chapter ["Creating Plug-Ins for Innovator"](.).)

Reusing Specific Data for All Users

Prerequisites

You have made central user adjustments and created your own data for the use of Innovator.

Context

You want to reuse your central user adjustments in the new Innovator version.

- You transfer your own **checks, DDL configurations, searches, layouts, and mapping references** as part of the user-specific settings in the `$INODIR` directory. Transfer your data or synchronize it with the installed files.

- You transfer your own **icons** to the corresponding directory for the new Innovator version. To enable the use of the icons, this resource path must be entered in the options. You can propagate

the icon directory as `UserIconPath` in the `DisplayOptions.xml` file in the `$INOHOME` directory.

- Your own **Java projects** with the Java files for the use of engineering actions and in batch calls must be compiled against the Innovator SDK of the new version and made available in the subdirectories of the `$INODIR/java` directory.

  If you have received a customer-specific delivery for engineering actions from MID GmbH Support for the earlier version, you need this for the new version, too. Contact MID GmbH Support.

  If you have received a customer-specific delivery from CSI or Consulting from MID GmbH, please contact them.

- **Plug-ins** are located in the `<Innovator Installation Directory>/Plugins` directory.

  If you created the plug-in yourself, set the reference paths to the new version in your plug-in project and recreate your plug-in.

  If you have received an on-demand plug-in from MID GmbH Support, contact MID GmbH Support.

  If you have received a customer-specific plug-in from CSI or Consulting from MID GmbH, please contact them.

  Plug-ins that are included within the scope of delivery for Innovator do not need anything done to them. The new version of the plug-in is installed.

Custom Data of Individual Users

The custom data of individual users generally relates to their specific working environment, so specific options and created perspectives.

Innovator saves these user settings in the `$INOHOME` directory (by default, this is `%APPDATA%/Innovator/<Version>`).

In addition, files containing user-specific, computer-specific, and model-specific settings have arisen in subdirectories.

To ensure that these settings are retained for the user, the content of the `$INOHOME` directory should be transferred from Innovator 15.0 into the `$INOHOME` directory of Innovator 15.1.

The settings must be made for each work station.

Transformation Preparations

You may need to prepare models so that they can be successfully transformed.

> **Attention**
>
> ⚠ You need to create a copy of the project before carrying out transformation preparations to avoid loss of data and to be able to continue using projects in the previous Innovator version in parallel.

General: Type of Primitive Type Modified

What is the Issue?

The following change may affect you if you use slots.

The "Kind of Type" of the following primitive types have been changed from "String" to a concrete type for the "ROOT PROFILE" included within the scope of delivery:

- "Decimal Number" and "Dezimalzahl" use "Decimal Number" with two decimal places
- "Date" and "Kalenderdatum" use "Date"
- "Time" and "Tageszeit" use "Time"
- "Date Time" and "Zeitpunkt" use "Date Time"

Attributes can be typed using primitive types. If these attributes were created as slots within instances of the classifier, then you can assign values according to kind of attribute type.

What Happens During Automatic Transformation?

The values are no longer permissible if the kind of type is changed; these then need to be corrected. To simplify correction, you can change the values in the model to the respective intended formats before transformation. You then only need to open the dialog in the new Innovator version and save the values for the data model.

What You Should Do Before Transforming Affected Models

Context

The previous string needs to be changed to suit the new format. In this case, the values are recognized in the new Innovator version in the dialog for setting the slot value and can be directly saved in the new format.

This favorable case should generally already apply to decimal numbers. The permissible format for decimal numbers consists of numbers with predefined maximum numbers of decimal spaces.

All data and time values should be changed to the form "YYYY-MM-DDThh:mm". Superfluous parts are ignored upon migration and can be filled with information, e.g. "T00:00" for date value and "2000-01-01T" for time value.

- Date: "YYYY-MM-DDThh:mm" -> "YYYY-MM-DD"
- Time: "YYYY-MM-DDThh:mm" -> "hh:mm"
- Date and time: "YYYY-MM-DDThh:mm" -> "YYYY-MM-DDThh:mm"

> **Note**
>
> If extensive customizations need to be made, then MID would be happy to help you migrating models from individually customized plug-ins. The same applies when creating your own plug-ins.

How to proceed

1. Search in the model for elements with any name and element type **Slot**. If the element type is not available, then there are no slots in the model and no editing is required.

2. If slots exist in the model, then search for elements in the model with any name and element type **Instance Specification**.

   The existing instance specifications are displayed in the **Result Region** tool window.

3. Check whether the value or value list in the visible slots displays a data or time in the **Details** tool window for each instance specification in the **Result Region**. You only need to change the format for these slots.

4. Make a selection for each of these slots in the **Details** in the **Set Slot Values...** context menu.

   The dialog of the same name appears.

5. Change all values to the format they will use in the future.

6. Check the values and confirm with **OK**.

   The values are formally adjusted to suit the format that will be recognized in the new Innovator version and that you can confirm.

You can find detailed information about the **Find** and **Set Slot Values** dialogs in the respective Help chapters.

What You Should Do After Transforming Affected Models

The easiest way to find out which slots are affected is to use the The 'Name' value has the 'Name' type but should have the 'Name' type. [VFY284] verify option. This verify option is part of the standard verification routine "Violations of Configuration".

Starting from the attribute types mentioned above, the verification identifies the String format which is now invalid. If these values were formally adapted, then they are shown in the **Set Slot Values** dialog and only need to be confirmed.

If the stored string does not fit the new format type, then it is not possible to show the dialog for setting the slot value of the previously set values. In this case, enter the values in the dialog.

Migrating Repositories and Model Versions

This chapter contains a description of general worksteps necessary for migrating the predecessor version 15.0 to Innovator 15.1.

Transformation Process

The Basics

This information is only relevant for users of Innovator 15.0 who want to upgrade to Innovator 15.1. Migration is performed for those projects (repositories or model versions) which are to be continued in Innovator 15.1 in order to be able to use new functions and configuration options.

You can carry out transformation with the **Transformer** or using commands (transformer or PowerShell). Further steps may be required for migration.

> ⚠️ **Attention**
> - Basically, the models you want to transform must not provide any significant verification messages, especially for the "Configuration violations" and "Unresolved model references" verifications.
> - The model servers of the data repositories and the model versions you want to transform must be shut down, otherwise an error message appears when you try to start the transformation.
> - If your PATH environment variable contains path information to Innovator, then change it to the path of the current version before transforming.

Transformation Process

You need to transform repositories and model versions if you want to use functions which are available in Innovator 15.1 when working on your projects.

If not, continue running the projects in the existing version (see Help, chapter "Structuring Licenses with Internal Projects").

You can carry out the transformation process with the **Transformer** or using commands (transformer or PowerShell).

The installation process of the new Innovator version updates the program components. The transformation process makes extensive changes to model data that are accessed for the transformation if this is required due to further development of Innovator. This means you do not have to carry out any automated customizations.

Any customizations you may have made are created in many situations and automatically converted. However, there are certain customizations that cannot be carried out as standard.

Transforming with the Transformer

Transforming Repositories

Prerequisites

The model servers of the repositories you want to transform are down.

Context

Transformation can occur automatically in consecutive order for various repositories.

> **Note**
>
> You can use PowerShell commands (see "Migration Using Commands") to carry out parallel transformation of repositories.

How to proceed

1. Start the **Transformer** (`inotrans.exe`) in the new Innovator version.
2. Select the **Data Repositories** entry from the **Type** drop-down list.
3. Select the previous version in the **Source** group and use the [...] button to select the source directory if it is different from the directory suggested.
4. If necessary, enter a pattern for constraining repository selection.
5. Select one or more repositories from your previous version from the **Repository** list.

   If you select precisely one repository, you can enter a new name in the **Target** group; if this is not the case, the repositories' names are adopted.
6. Select the target directory (which differs from the source) for the repositories from the **Target** group.
7. If you have selected precisely one source repository, you can enter a new name for 15.1's target repository in the **Repository** field.
8. Finally, click on **Apply** to start the transformer.

   The transformer converts the source repository/repositories for Innovator into the data format used for 15.1.

   You then need to adopt the content of the change set when transforming the data repository. An additional progress bar appears for this.

   Once the transformation has taken place, check whether your model data has been successfully adopted into the new repository.

Transforming Model Versions

Prerequisites

The model servers of the model versions you want to transform are down.

Context

As you would for repositories, you can also use the transformer when transforming model versions of managed models. Select the appropriate **Type**.

All managed models that exist in the project directory are transformed in a single step for the **model versions** type.

> **Note**
>
> The model versions are also automatically transformed when you transform the repository using the PowerShell command; this is because the command refers to the entire project path (INOPRJ)(see "Migration Using Commands").

Migration Using Commands

> ⚠ **Attention**
>
> The model servers of the repositories and model versions you want to transform must be down.

Using the Transformer's Command Line

To use the transformer in the command line, please call the `inotr.exe` program (not `inotrans.exe`) with the following parameters. (Linux: use - not /)

> 🚫 **Restriction**
>
> The model directory's `ivm` source directory name must also be retained if you wish to transform it as well.

- Data repository
  ```
  inotr [/d icw] <source repository> <target repository>
  ```
- Model version directory
  ```
  inotr /d versDir <source directory> <target directory>
  ```
- Display of supported types and their versions
  ```
  inotr /l
  ```
- Copy data repository, if version is up-to-date
  ```
  inotr /c
  ```
- Display of Innovator program version
  ```
  inotr /v
  ```

Using PowerShell Commands

It is also possible to carry out the transformation using a PowerShell command.

This may make migration length considerably shorter when migrating large amounts of repositories thanks to parallel processing (siehe "Automatisierung administrativer Aufgaben").

```
Update-InnoDataDirectory -InoPrjOld <source-InoPrj> -InoPrjNew <target-InoPrj> -ParallelTransformations <number of parallel transformations> -LogDirectory <directory for temporary output>
```

Please take both the number of physical cores and the amount of space available into consideration when establishing parallel transformations (default is 1).

Loading Configuration Extensions

Using New Configuration Options

Version migration initially essentially adopts the previous model configurations.

Configuration modifications may mean that certain customizations need to be carried out upon transformation; these customizations are then automatically made by the transformer.

New configuration options are not automatically entered in the migrated models. If you also want to use these configuration extensions for your migrated models, then you need to import the respective migration files in the configuration ($INODIR\config\Migration *.aob, $INODIR C:\ProgramData\Innovator\15.1\ as default).

Migration Files Are Language-Independent

The migration files are available in a language-independent form. They can be used in both languages to update the profiles included within the scope of delivery of Innovator and used in your models.

Importing Migration File with Configuration Extensions

Prerequisites

You need administrator rights for the model when importing.

Context

All profile elements which correspond to one of the objects in the migration file are replaced by an element with the same name when a migration file is imported.

How to proceed

1. Jump to the model you want to load the configuration modifications for in the **Configuration Editor**

2. Select **Configuration>Import>From Model Fragment...**

   The dialog of the same name appears.

3. Enter the migration file in the **File Name** field.

   It is also possible to use the button to select a migration file using the standard file selection dialog. Select a Migration `*.aob` migration file in the `$INODIR/config/` directory. The file name is adopted into the **Filename** field and the path to this file into the **Path** label by clicking on the **Open** button.

4. To do this, use the following options by default:

   - **Identify elements by namespace** = `no`
   - **Adopt UUID on import** = `yes`
   - **Behavior on Conflicts** = `Abort import`

   The `Abort import` option tells you about possible conflicts without changes having happened already. It tells you which elements in the configuration would be renamed if the import occurred.

   Change the **Behavior on Conflicts** option to `The element already existing in the model is uniquely renamed` to start the import.

5. Confirm with **OK**.

   The migration file is loaded. All existing profile elements that correspond to one of the objects in the migration file are replaced by the elements of the same name from the file.

6. To load the configuration into an open model, select the model root in the model editor and click [F5].

Postprocessing after Transformation

You may have to postprocess models once they have been transformed so that all functions can be used.

Innovator for Business Analysts: Display Options for Diagram Notation in BPMN Diagrams

You can use the "Diagram Notation" display option to control how nodes should look in BPMN diagrams. "Innovator" uses the present display, whereas "Standard" uses the more angular BPMN specification. If no display option is selected, then "Innovator" is used.

The "Diagram Notation" display option included in the "BPMN 2" profile within the scope of delivery has been configured as interchangeable with the "Standard" value in the diagram. This display option causes nodes to be displayed using straight lines in BPMN diagrams, in accordance with BPMN specification. The same applies when loading configuration extensions (migration profiles).

If dotted line notation like the previous Innovator notation should be used in certain BPMN diagrams, then you can change this setting via the display options in the diagram.

If the Innovator notation should be used in all diagrams, then you need to set the diagram notation to "ignore" in the display options in the customization profile. If no display option is set, then the Innovator notation is used.

Customers with their own configuration do not need to reload the migration profiles. You only need to activate them if you want to use the standard notation in the future and not the Innovator notation for nodes as you currently do.

The "Diagram Notation" display option is created in a suitable BPMN diagram profile and set to the "Standard" value. The nodes are displayed using the previous Innovator notation if this display option is not created.

You can find more detailed information about migration files in the "Loading Configuration Extensions" chapter.

Innovator for Business Analysts: Using concepts as business resources

As well as resources, actors and components, you can also use concepts as values for the following properties: "Partner Entity" and "Partner Role" for participants, "Resource Assignment" for activities, "Resources" for Lanes and "Responsible Business Resource" for processes. The configuration must be extended accordingly if you wish to use concepts.

The "Responsible Business Resource" relationship has been made configurable for processes. More element types can now also be used as resources. The applicable configuration needs to be extended to be able to use resources as they were previously used.

The "Responsible Business Resource" relationship is configured for the process in the "BPMN 2" profile included within the scope of delivery. Previously supported element types Business Resource, IT Element, Organizational Unit, Person and Role are permissible.

Migration depends on whether you use the profiles included by MID within the scope of delivery or your own profiles. Existing resources will not be deleted. The resources cannot be exchanged if the configuration has not been adapted.

- You need to reload the migration profile when using the profiles included within the scope of delivery.

  This extends the "Responsible Business Resource" configuration for processes in the "BPMN 2" profile.

  You should then add further resources in the Customization profile.

- You need to configure the "Responsible Business Resource" property accordingly when using your own profiles so that resources can continue to be used for processes.

You can find more detailed information about migration files in the "Loading Configuration Extensions" chapter.

Innovator for Information Architects: Display options for diagram notation in Entity Relationship and Data Vault diagrams

You can use the "Diagram Notation" display option to set diagram notation for Entity Relationship and Data Vault diagrams. You can also use the "SERM Edge Notation" display option for SERM notation to set whether edges should be displayed according to standard SERM notation, classic (as in Innovator Data classiX)) or in James Martin notation.

The notation last used is used for existing diagrams if no display option is set; SERM notation is always used for new diagrams. SERM edge notation is used as standard if no display option is used.

Interface options for diagram notation and SERM edge notation are omitted.

The "Diagram Notation" display option should be set to "null" and "SERM Edge Notation" to the "Standard" value for Entity Relationship profiles and in some "Business Intelligence - *" profiles for Data Vault diagrams in the "Entity/Relationship Modeling" profile included within the scope of delivery. The relevant configurations are added to when the configuration extensions (migration profiles) are reloaded.

The display options replace the appropriate interface options. You cannot access the interface options when carrying out model migration. This means that data cannot be adopted.

If the value is set to "null" for diagram notation, then the notation which was last used in the diagram is evaluated. New diagrams are always created using SERM notation in this case. If another notation should be used in certain ER diagrams, then you can change this display option in the diagram.

If a certain notation should be used in all diagrams, then you need to set the "Diagram Notation" to "ignore" in the customization profile. The display option can then be added with the desired value after this. All existing and new diagrams will then use this notation.

The "SERM Edge Notation" display option is then needed for the "SERM Notation". Set it to another value in the same way or ignore it.

You must set the "Diagram Notation" for new diagrams if you use your own configuration. If you do not, then all new diagrams will be shown in SERM notation. The display option should be added to a suitable profile with the required value. The "null" value should not be used if possible. The "SERM Edge Notation" display option is only required if "SERM" is used as diagram notation and the edges should not be shown in accordance with the standard notation.

You can find more detailed information about migration files in the "Loading Configuration Extensions" chapter.

Innovator for Information Architects: Include columns in the database index

Indexed table columns have the "Include" property. The concept for the enclosed columns are supported by the INCLUDE clause in the CREATE INDEX statement.

If the type system option INCLUDE was used at the index for older Innovator versions, then these need to be adapted.

To do this, add the columns previously used in the INCLUDE type system option to the index. Set the property to "Include". Remove the value for the type system option.

The INCLUDE type system option can then be deleted from the profile once all relevant indexes have been modified.

> ⚠️ **Attention**
> A customer-specific DDL generator also needs to be set to the new property.