Release Notes 2.6.0

g9

April 20, 2017

Date:	20-Apr-2017 15:48
URL:	https://esito-conf.inmeta.com/display/g9RN/Release+Notes+2.6.0

Table of Contents

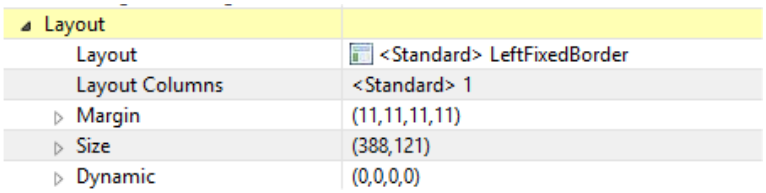
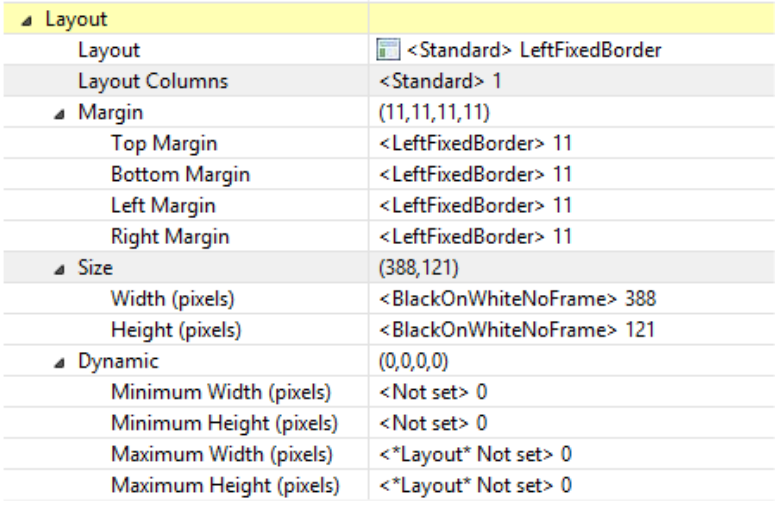
1 News and Changes 4
1.1 Improved Properties view 4
1.2 New g9 project wizard 5
1.3 WSDL/XSD as model source 5
1.4 Multiple model sources 6
1.5 Changes to g9 project properties 6
1.6 React/Redux client generator 7
1.7 Generator for REST wrapping of JGrape services 8
1.8 Perspective changes 8
1.9 Move files from src-gen to src 8
1.10 Type Mappings 9
1.11 No of Characters in Dialog Templates Type Representations 9
1.12 java.time support 9
1.13 Changes in Java 8 date support 9
1.14 BigDecimal 10
1.15 Changes in mapping of Numeric and BigDecimal 10
1.16 Java enums 10
1.17 Support for generating Java Bean Validation annotations 10
1.18 Support for generating Jackson annotations 11
1.19 Support for generating JAXB annotations 11
1.20 Enterprise Architect packages 11
1.21 Xcore feature 11
1.22 Xtext 2.11 12
1.23 Jetty 9.4.x upgrade 12
1.24 Documentation updates 12
2 Generated code and Model Changes 14
2.1 Changes in model files 14
2.2 Changes in generated code 14
3 Dependent versions 15
4 Improvements and corrected Errors 16
4.1 g9 16
4.2 g9 Database Import 23
4.3 Swing generator 23
4.4 ICEfaces generator 24
4.5 Java Domain generator 24
4.6 Java runtime 26
4.7 EA/Xcore/Java synchronization 26
4.8 RESTful services generator 28
4.9 Jasper Print generator 28
4.10 Hibernate generator 29
4.11 Database MySQL generator 29
4.12 Known Issues 29

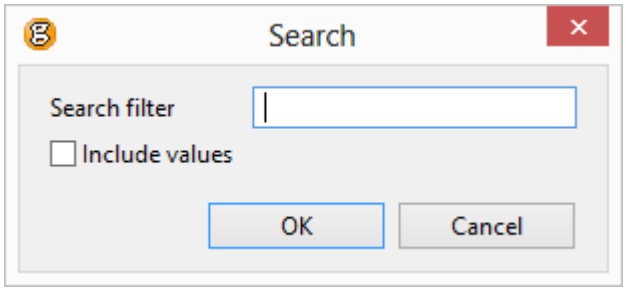
4.12.1 New example projects can get cyclic dependency errors 29
4.12.2 Slow Eclipse workspaces 29
5 Distribution 30
6 Installation 31
6.1 g9 requirements 31
6.2 Esito update-site 31
6.3 Local update-site 31
6.4 Xcore modeling 32
6.5 Writing generators in Xtend 32

1 News and Changes

1.1 Improved Properties view

The g9 Property View has been reorganized. Several properties are grouped and it is added new toolbar buttons on the properties view. See Properties View for more information.

	Toolbar button	Description
	Restore Default Value	Restores the value back to its default value. This is also available as a context menu by right clicking on the Property column
	Toggle Expand State	Some properties are organized with a heading that shows the property values in a compressed fashion. The details are shown when toggling the expand state.
	Expand Off	
	Expand On	
	Filter Used On	Show only the properties edited with a value different from the default value.
	Filter Used Off	Show all properties.

	Toolbar button	Description
	Search Filter On	Show the properties with a name containing the search filter text. Check *Include values* if you want to filter both names and values. 
	Search Filter Off	Remove search filter and show all properties.

New structure on TAB Order, Sort Order, Layout

1.2 New g9 project wizard

The old wizard *g9 UI Modeling Project* has been replaced with the new wizard *g9 Modeling Project*. The new wizard to create g9 projects produces many artifacts needed by g9. More information is available Creating a new g9 Modeling Project in the User Guide.

These artifacts may be created with the new g9 modeling wizard:

- The project name
- Project description
- Domain model source definition with fetching of package structure
- Selection of application frameworks and database system
- Possibility to create target projects, both Java with Maven support and non Java projects
- Adding root package to be used as package prefix
- Setting of various application specific artifacts and packages
- Setup of generators and build for the selected application frameworks

The wizard will create all the defined artifacts and synchronize the domain model into the project.

1.3 WSDL/XSD as model source

WSDL/XSD files are supported as a new model source for g9 projects. See Modeling with WSDL/XSD files as Model Source in the g9 documentation on how to setup and synchronize the models.

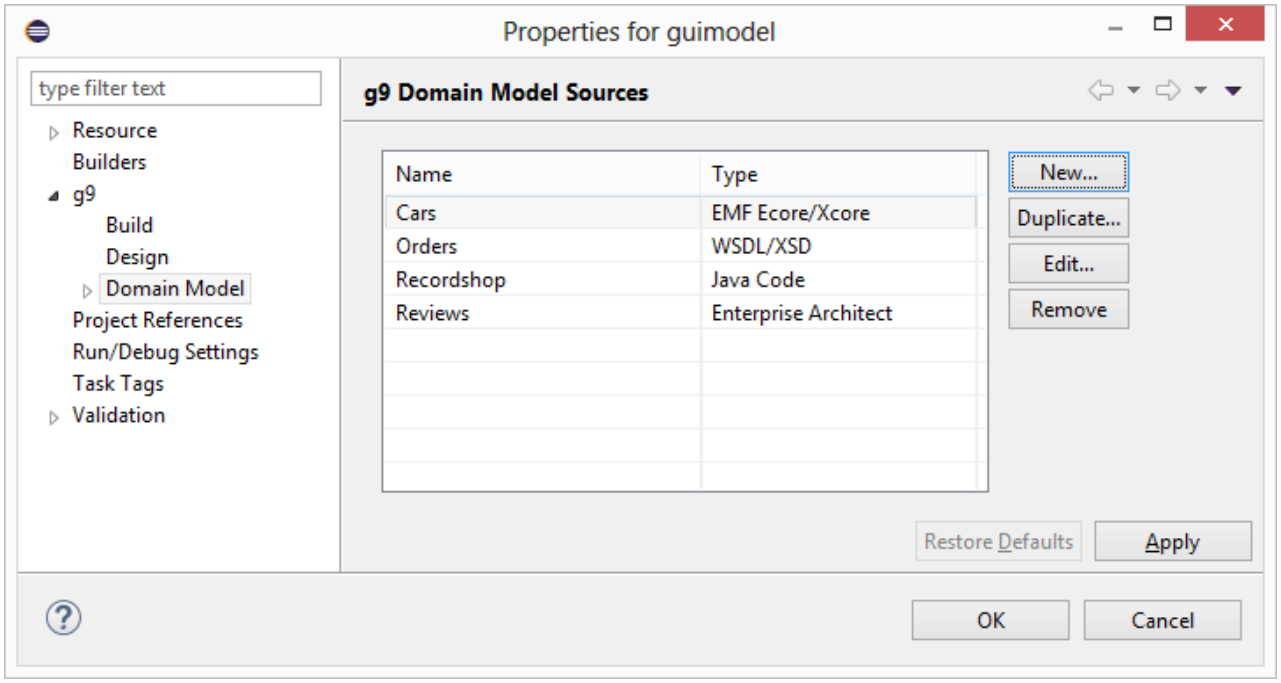
1.4 Multiple model sources

You may combine many model sources in one g9 model. The source types may be of any kind and may be repeated several times. Each class must have a unique combination of domain name and package.

The supported model sources:

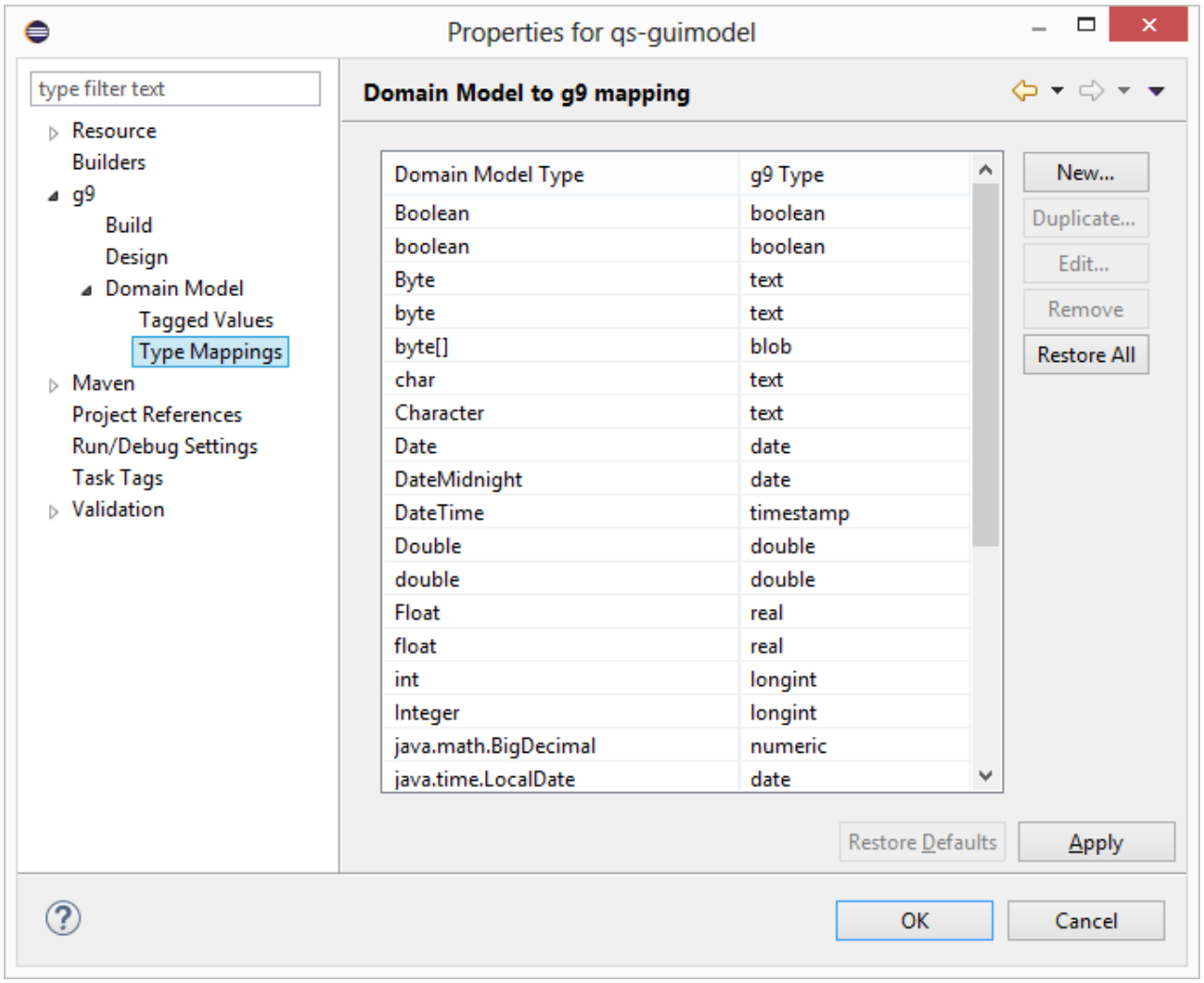
- EMF Ecore/Xcore
- WSDL/XSD
- Java Code
- Enterprise Architect

Sample property settings for several Domain Model sources:



1.5 Changes to g9 project properties

The *Domain Model* property page is changed as explained in the previous paragraph and the *Tagged Values* and *Type Mappings* pages are moved to sub pages as shown below:



In addition is the auto-save / auto-load settings removed and always on.

1.6 React/Redux client generator

The *ReactJS* generator is a Dialog Model generator for the React JavaScript framework, using Redux for component data bindings. The ReactJS generator uses Node.js and webpack to transpile/compile JavaScript code to JavaScript bundles. The npm package manager is used to download project dependencies.

The generated code from the ReactJS generator uses a REST API for data services. This API is generated with the The REST JGrape Wrapper generator.

More information is available, see The ReactJS generator.

The generator is not finished, is in a beta stage and uses elements that will change in coming versions. You may improve the generator to fit your needs.

1.7 Generator for REST wrapping of JGrape services

The REST JGrape Wrapper generator generates a REST layer on top of JGrape service managers. The REST application is based on Spring Boot, and a *SpringBootApplication* class is generated for the Application in the g9 model.

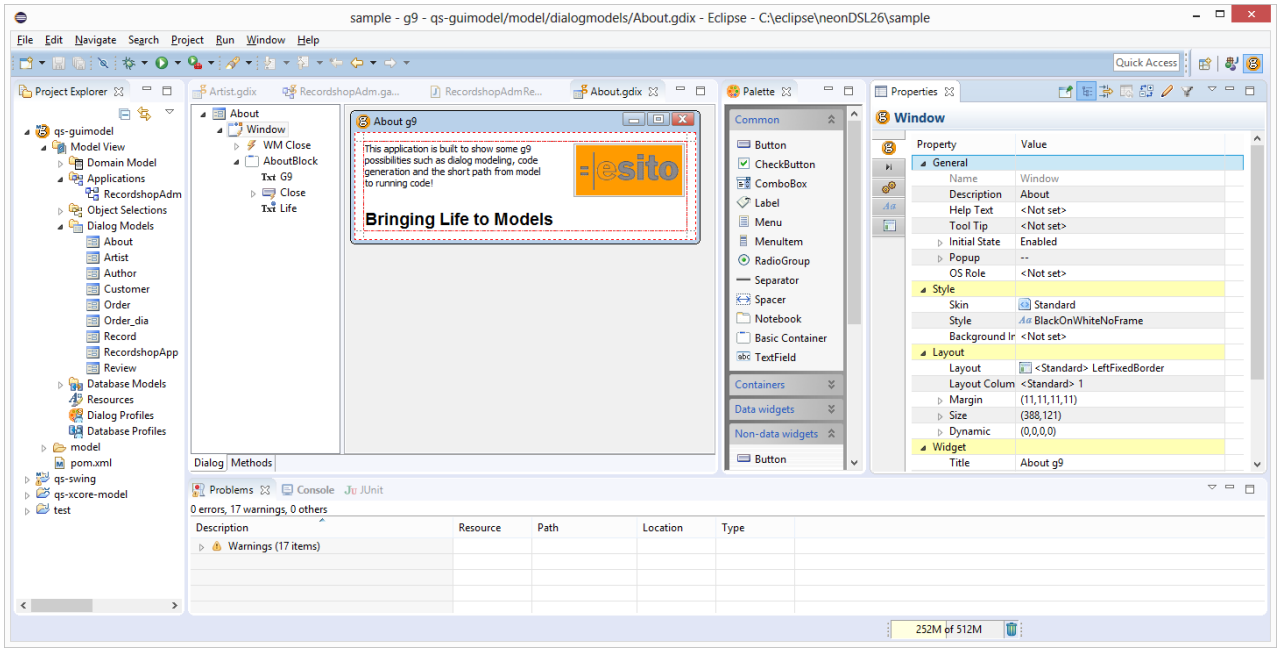
For each Object Selection in the model, a *RestController* is generated with methods for the JGrape actions *Find*, *Find All*, *Insert*, *Update* and *Delete*. One set of action methods is generated for each Role in the Object Selection, possibly limited by the Legal Functions for the role.

Classes for REST exceptions etc. are added to the g9-common runtime library and used by the REST wrapper.

More information is available, see The REST JGrape Wrapper generator.

1.8 Perspective changes

The default g9 perspective is reorganized because most users work on big screens. The Properties View is moved to the right showing more properties than before. You may create your own perspective if other settings are preferred.



1.9 Move files from src-gen to src

If you want to manually maintain a generated file, you can move it from the target directory to the same path in the source directory and it will no longer be overwritten when generated. To assist this process you can now right-click the file and select *g9>Move file to <Source>* and this will move the file and create the required parent folders in the <Source> directory.

1.10 Type Mappings

Some domain model types are extended with full package:

New	Old	g9 Type
java.math.BigDecimal	BigDecimal	numeric
java.time.LocalDate	LocalDate	date
java.time.LocalDateTime	LocalDateTime	timestamp
java.time.LocalTime	LocalTime	time
no.g9.support.Numeric	Numeric	numeric

1.11 No of Characters in Dialog Templates Type Representations

The Type Representation in Dialog Templates has added *Number of Characters* property. This can be used to set a fixed width for ranges. It is especially useful when input types are large and it obviously does not fit in a text-field representation.

For a multi-line field (number of lines > 1) the number of character will be as specified in the dialog templates type representation. For a single line field the number of characters will be set to the number of characters from the type representation only if the field's calculated width is larger the specified number of characters. Otherwise the field's number of character will be set to zero.

See Data Type Representation for details.

1.12 java.time support

The runtime support for the JSR 310 backport (Java Time) is removed, and substituted with support for the `java.time` classes.

1.13 Changes in Java 8 date support

In the general type mapping from model types to g9 types the three default mappings for `LocalDate`, `LocalTime` and `LocalDateTime` are changed and are now specified with their full path: `java.time.LocalDate`, `java.time.LocalTime` and `java.time.LocalDateTime`.

Existing projects will be upgraded with these mappings. However, the old mappings will not be removed in the upgrade process. They must be removed manually if not wanted.

The XSD generator also mapped these three data types and the defaults is changed accordingly.

Note: If an old XSD type mapping does not have a value, i.e. has been using the default value, the old mappings value is set hard to this default value.

1.14 BigDecimal

The Hibernate generator and the g9-jgrape runtime jar now supports attributes of type `BigDecimal`.

The Swing generator and the g9-swing runtime library now supports both the Numeric and `BigDecimal` Java classes for numeric (BCD) attributes.

The ICEfaces and Jouteur generators, and the g9-icefaces2 and g9-jouteur runtime jars, now supports attributes of type `BigDecimal`.

1.15 Changes in mapping of Numeric and BigDecimal

In the general type mapping from model types to g9 types the two default mappings for `Numeric` and `BigDecimal` are changed and are now specified with their full path: `no.g9.support.Numeric` and `java.math.BigDecimal`.

Existing projects will be upgraded with these two mappings. However, the old mappings will not be removed in the upgrade process. They must be removed manually if not wanted.

1.16 Java enums

The Swing generator and the g9 runtime for Swing now supports native Java enums in addition to G9Enumerator based enums. The generator property "Generate Java style enums" in the Java Domain generator is used to select which type of enumerator to generate for the domain model.

The g9 runtime has class `no.esito.util.EnumUtil`, with methods `getEnumValue` and `getEnumLiteral`, which will provide ordinal and text values.

1.17 Support for generating Java Bean Validation annotations

Java Bean Validation annotations can now be generated from the Java Domain generator, by setting the *Generate Bean Validation annotations* generator configuration parameter to *true*. The generator supports the `Valid`, `NotNull`, `Size` and `Digits` annotations.

1.18 Support for generating Jackson annotations

When the *Generate Jackson annotations* generator configuration parameter is *True*, the Java Domain generator will generate some Jackson annotations to the domain classes. If the class has a value in the `JsonFilter` tagged value, a `JsonFilter` annotation will be generated. If the generator configuration parameter *Persistent JSON ignore properties* has a value, a `JsonIgnoreProperties` annotation will be generated for persistent domain classes. If the generator configuration parameter Transient JSON ignore properties_ has a value, a `JsonIgnoreProperties` annotation will be generated for transient domain classes. A `JsonTypeInfo` annotation will be generated to the `lockFlag` property if the type of the property is `java.io.Serializable`, to avoid problems with deserialising the lock flag.

1.19 Support for generating JAXB annotations

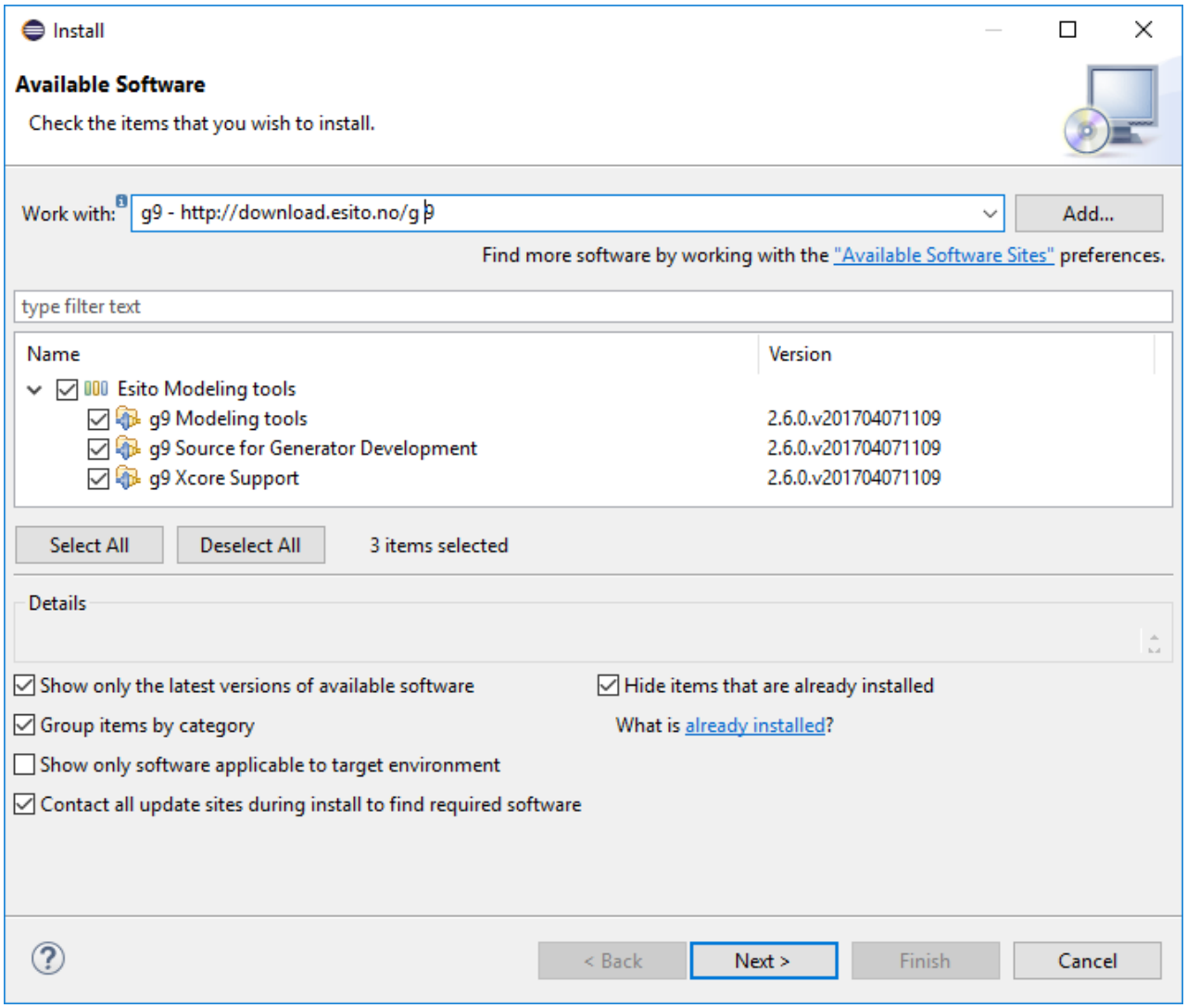
The Java Domain generator now supports generating JAXB annotations for XML marshaling and unmarshaling.

1.20 Enterprise Architect packages

When updating from Enterprise Architect the packages was identified by the Enterprise Architect id. For any other model source the packages are identified by the package path. From this version, this will be the case also for packages specified with Enterprise Architect. This implies that renaming a package in the Enterprise Architect will be the same as adding a new package to the model.

1.21 Xcore feature

To help installation for use with Xcore, the Xcore feature has the necessary dependencies to help installing the necessary Xtext and Xcore plugins from the Eclipse update site.



1.22 Xtext 2.11

Xtext version 2.10 is distributed via the Eclipse update site, but g9 requires version 2.11 for generators. If you have installed Xtext via the Xcore feature or the DSL version you should have the Xtext All-In-One update site available, thus a "Check for updates" will install the 2.11 version.

1.23 Jetty 9.4.x upgrade

The REST server generated web.xml file no longer references the Spring `ContextLoaderListener`.

1.24 Documentation updates

New Getting Started section

- Getting started with g9
  - Getting started with ReactsJS
  - Getting started with ICEfaces
  - Getting started with Swing
- From database to running application
- Getting started with Custom Generators
- Getting started with WSg

The chapters below are moved to the *User Guide > Reference > Domain Model*

- Modeling with Ecore/Xcore
- Modeling with Enterprise Architect
- Modeling with Java code as Model Source
- Modeling with WSDL/XSD files as Model Source

Custom Generator Development is improved with descriptions of generator API and models of all internal g9 models.

- Generator API and models

2 Generated code and Model Changes

2.1 Changes in model files

Use of XML escape sequences in model files

In strings in XML model files CR, LF, TAB and the XMLs own meta characters are all escaped using XMLs escape mechanism. This coding of CR, LF and XML are new to the model files. In addition the coding of the apostrophe character ' was missing. This changes implies an upgrade of all model files.

Database Model sequence numbers

It will be differences between the earlier versions and later versions of the database model file.

Persistent settings for value objects

Changes in the settings need an update from the modeling tool.

Store sequence of attributes and associations for classes in the domain model

The sequence is retrieved from the model when the g9 domain model is updated from modeling tool. As this external model is used by the generators, this change may result in changes in the generated code. These changes should however not be significant for the behavior of the generated code.

2.2 Changes in generated code

Use Java 8 for the g9 runtime

The g9 runtime is now built with Java 8 as target.

Regeneration of code is necessary.

3 Dependent versions

g9 runtime libraries uses 3. party open source libraries. Supported versions in this release are:

Library	Version		Library	Version
Spring	4.3.5.RELEASE		Hibernate	5.1.4.Final
ICEfaces	4.1.1 4.1.0-GA		JSF	2.2.10
log4j	2.8		Jetty	9.4.1.v20170120
Jackson	2.8.6		Spring WS	2.4.0.RELEASE
Castor	1.4.1		JasperReports	6.4.0
Joda Time	2.9.7		Apache POI	3.9
EJB	3.2.0		Guava	21.0

4 Improvements and corrected Errors

If nothing is written in the *Release Comment* section about an error, the error is corrected and a comment is not necessary.

4.1 g9

SUP-1143 - MultiSelect with Move changes selection

SUP-2081 - Illegal representation on attributes does not get Problems error
Release Comment: Custom Widget types that are no longer there now get a problem marker where referenced.

SUP-3599 - Drop on a target with setting of source as a property should change focus to the target object

SUP-3674 - Linked dialog does not display Linked decorator in Project Explorer

SUP-3898 - Redo fails on added elements

SUP-4025 - <Default> values should be a property, not first occurrence
Release Comment: Default to use for design profile, mapping profile and dialog template is now settable in the profiles model. Right-click and select "Use as default". The parent folder also has a property "Use as default" where correct default can be selected. The default object will be annotated with a green check-mark.

SUP-4809 - Drag and drop from palette needs to be more precise

SUP-4820 - Unnecessary loads of models when saving a resource model

SUP-4823 - Drag and drop from the palette does not give insertion marker

SUP-4855 - Paste should not be available on all types of folders in the Model View when the copied object is a file reference

SUP-5045 - Custom Widget Compiler Wizard disables next/finish buttons even if errors are resolved

SUP-5060 - Building a package under the Dialog Models package also build Domain Model elements
Release Comment: Build on a package in "Composite flat" view behaved as if in "Package flat" view. Now it will only build for its respective object type when in Composite view modes.

SUP-5081 - Generalizations/Realizations from external interfaces are not supported in the g9 domain model
Release Comment: Modeling realizations/generalizations from classes and interfaces to interfaces not part of the model, i.e. how to model usage of java.io.Serializable without having this interface as part of the g9 domain model.
In *Enterprise Architect* such dependencies must be modeled as a link (realization or generalization) and both ends must exist in the model. If the interface is inside the domain models logical view, it will be imported as an ordinary interface to g9's domain model. However, if the interface is outside the logical view, the interface will not be incorporated in the model, but realizations and generalizations referencing this interface will be created and the full path of the interface will be stored.
Modeling an interface such as java.io.Serializable in Enterprise Architect is done in the ordinary way by creating the packages java and io and the interface Serializable. Since the java package is outside g9's logical view, g9 will include all packages above java in the package path for the interface. To restrict the path to java.io the package above java must be set as Namespace Root, see Use of property Namespace Root for packages in Modeling with Enterprise Architect.
In *Ecore* such an interface can be modeled by creating an class of type interface and setting the InstanceTypeName property to the full name and path for the interface, i.e. java.io.Serializable.
In *Xcore* modeling such interfaces are done by defining a wrapper for the interface:

```
interface Serializable wraps java.io.Serializable{}
class Answer extends Serializable {}
```

When updating the domain model no interface java.io.Serializable will be created, only a realization with a reference to java.io.Serializable.
The Ecore model generated from such an Xcore model will contain an interface class with InstanceTypeName set.
When using java code as model a class implementing an interface will get a realization in the g9 model. But the interface itself will only be incorporated if the interface is defined in one of the java files included in the model project.

SUP-5222 - Move Tagged Values and Type Mappings to new sub-"Domain Model" pages

SUP-5223 - Removed .gproject auto-save / auto-load settings (always on)

SUP-5225 - Persistent property for classes must be set in model update
Release Comment: In domain models updated from java source the default setting for a class's persistent property is specified by the "Set classes persistent" property in the Java Domain Model Configuration page of the project properties. This default value is used if the class is not a value object or if the java class contains other annotation specifying otherwise. Value object classes are always non-persistent.
An update from domain model is needed on java model projects

SUP-5229 - Dialog Templates Type Representation should add Number of chars
Release Comment: This can be used to set a fixed width for ranges. It is especially useful when input types are large sizes such as 1000 chars where it obviously does not fit in a text-field.
For a multi line field (number of lines > 1) the number of character will be as specified in the dialog templates type representation. For a single line field the number of characters will be set to the number of characters from the type representation only if the field's calculated width is larger the specified number of characters. Otherwise the field's number of character will be set to zero.

SUP-5231 - Widgets added from palette are added to the bottom of the container

SUP-5238 - Wizards with tables resize columns wrong
Release Comment: Problem occurs when resizing a dialog to see more in a table. The table would collapse its columns to a size that often hid content. Now the column resizes with the table. If you manually have dragged the column width it will retain that width when resizing the table.

SUP-5249 - Unnecessary opening of projects from navigator
Release Comment: Upon starting Eclipse all projects would load which is seen by log messages. This was due to validation of project for annotations. Now the projects will only be loaded when expanded or opening it's models in editors.

SUP-5252 - The Ecore model for generators should use the defaultValueLiteral field for EAttribute instead of a g9 annotation
Release Comment: The *initialValue* annotation is no longer exported for attributes, as the *defaultValueLiteral* field of EAttribute has the same value.

SUP-5253 - The Ecore model for generators should not set the inheritAs annotation for classes when the chosen inheritance strategy is the default value
Release Comment: The default inheritance strategy is *Inherited Attributes*, and the *inheritAs* annotation for classes is no longer exported when the default value is chosen.

SUP-5258 - Use of XML escape sequences in model files
Release Comment: In strings in XML model files CR, LF, TAB and the XMLs own meta characters are all escaped using XMLs escape mechanism. This coding of CR, LF and XML are new to the model files. In addition the coding of the apostrophe character ' was missing.
This changes implies an upgrade of all model files.

SUP-5260 - Change of display of multi line properties as tool-tip
Release Comment: The display widget has got some more space. The display also shows tab characters correctly.

SUP-5270 - Width and height for Window derived from Style is reported as derived from Layout

SUP-5279 - Importing an old model project using "Import existing.." without "copy" does not trigger a model upgrade

SUP-5280 - The Standard dialog template is not default
Release Comment: Now there is functionality to select a specific template like "standard" to be the default.

SUP-5281 - The derived Number of characters value is stored for multi line text fields when a dialog is created

SUP-5288 - Don't set Ordered=True for "one" associations and attributes
Release Comment: The default "Ordered" property value for attributes and associations in Xcore is "true", but in the g9 model update from Ecore/Xcore this is now set to false for features with a "one" cardinality.

SUP-5289 - Problem occurred shows if packages to fetch are empty

SUP-5292 - Pasting copied DBMS types in the DB profile is wrong

SUP-5296 - Design Profiles model needs to upgrade old widget names
Release Comment: Design profiles model now performs an upgrade which replaces old widget names such as "SimpleBlock" with the new ones, ie "BasicContainer".

SUP-5297 - Open dialog editor menu-items take focus
Release Comment: Showing a selected "menu" widget in the dialog designer would popup a menu to show it's look, but this could sometimes interfere with other dialogs such as New Wizards.

SUP-5299 - Unable to add certain keys as KeyAccelerator

SUP-5301 - MenuItems steal focus
Release Comment: Menubar will now only open with direct clicks on the menubar in the designer and will not open when its items or menus are just selected.

SUP-5307 - Missing accelerator and standard action documentation

SUP-5310 - Occasional hang after changing project settings
Release Comment: Eclipse would stop responding after changing project settings. The change of settings would be recorded although the Eclipse session would have to be killed.

SUP-5311 - Double-click in dialog editor does not show Properties View

SUP-5312 - Error "Wrong Name" when changing os role names should be auto-fixed

SUP-5320 - Sequence number for columns is not set initially
Release Comment: This gives unnecessary differences between the first version and later versions of the database model file.

SUP-5321 - NPE when moving columns in database model

SUP-5324 - Reorganized Property view

SUP-5326 - Icons for property view

SUP-5330 - Strange Search filter behavior

SUP-5331 - New structure on Resources Layout

SUP-5335 - Wrong path to the g9 EA UML profile

SUP-5339 - Missing the spliced name for value object columns when the attribute has column name
Release Comment: When column names are specified in the domain model these names are used for columns in the database models. If the name of a domain element is used as part of an element in a database model the column name is used rather than the name.
In a database profile case conversion to be used in the database model is specified. This case conversion is only applied to those parts of an element name derived from a name in the domain model. Name parts derived from a column name specification are not affected by this specification.

SUP-5340 - Color inherited is shown as black
Release Comment: New icons to represent colors for Inherited, System and Transparent.

SUP-5342 - Flexibility shows wrong value
Release Comment: A problem with handling of default values for fixed enums. The API bug only affected Flexibility and FrameCode. Whereas propertysheet would show correct value the generators could potentially pickup a different value.

SUP-5347 - Unable to add System image to resources database

SUP-5351 - Changing to Usertype from Role Attribute in dialog model but the role attribute name remains

SUP-5354 - Change MySQL driver to MariaDB
Release Comment: MariaDB JDBC driver has replaced the MySQL GPL-licensed driver.

SUP-5355 - DB Import NPE when previous driver no longer exists
Release Comment: Better error message when a JDBC Driver that was previously selected has been removed.

SUP-5363 - Update from Domain Model should store it's settings on a session basis pr project

SUP-5367 - NPE when picking image for newly added image objects

SUP-5378 - Wrong button order in g9 project properties
Release Comment: The button order was Cancel, OK, It is changed to OK, Cancel

SUP-5380 - Remove GoToDefault - F3 when not an obvious choice

SUP-5381 - Unable to rearrange actions

SUP-5385 - Need to supply widget size from generator render info
Release Comment: new method on `IGeneratorContext` which gets the default widget size.
`GWidgetSize getWidgetSize(GWidget widget)`

SUP-5386 - Horizontal and Vertical Flexibility shows wrong value

SUP-5397 - Invalid thread access exception when upgrading project

SUP-5400 - New domain model problem for associations
Release Comment: Problem: Association cannot be navigable to value object class.

SUP-5403 - Persistent settings for value objects
Release Comment: The persistent property for value object classes is removed from the classes. It is the usage of a value object that decides if the class in each usage context should be regarded as persistent or transient. When updating from the modeling tool any persistent setting for value object classes will be ignored.
In the modelling tool the persistent setting for a value objects' attributes and associations can be either false, true or not set. But when updating from the model the value in the g9 domain model will be either false or not set. I.e. if true the value will be set to *Not Set*. These settings are interpreted as follows: In the case of false, attributes and associations will be regarded as non-persistent in any usage of the value object class. If the case of Not Set, the attribute/association will inherit the persistent setting from the attribute having the value object class as model type.
To clean up existing g9 projects an update from the modeling tool is needed.
These changes will change the generated java code for the domain model. Any value object class was falsely generated with a lock column if the value object class either was marked as persistent or had its persistent setting set to Not Set. Now such classes will never get a lock column field. However this also implies that these classes will be generated without implementing the java.io.Serializable interface. If a value object class need this interface it should be specified in the model.

SUP-5404 - Disabled Checkboxes and Buttons are not possible to select in designer

SUP-5406 - Impossible to select disabled attribute in designer

SUP-5410 - No more widget handles error

SUP-5413 - Wrong window title text in wizard dialogs

SUP-5414 - g9 model files must use UTF-8 to avoid losing characters
Release Comment: All model files used ISO-8859/1 (Latin 1). They are now using UTF-8. Existing models will be converted to UTF-8 the first time they are updated.
This change implies that when editing a model file using a text editor, the editor should understand and use UTF-8. Special care should be taken if a model contains characters from the upper half of 8859 character se such as the Norwegian letters æ,ø, å, Æ, Ø, Å etc.

SUP-5416 - Impossible to switch value on Generator ProblemTypes

SUP-5431 - Add method to map packages to namespace
Release Comment: The IGeneratorConfig Interface is extended with the Method String getNamespace (String pck).

SUP-5435 - Store sequence of attributes and associations for classes in the domain model
Release Comment: The sequence is retrieved from the model when the g9 domain model is updated from modeling tool. This sequence is used when creating the external Ecore model for the domain model. The sorting sequence is used to sort the EStructualFeatures for each EClass in the Ecore model. As this external model is used by the generators, this change may result in changes in the generated code. These changes should however not be significant for the behavior of the generated code.

SUP-5439 - Wrong problem reporting for classes
Release Comment: The problem reported incorrectly was
Class nn has multiple navigable associations with the same role name, ''
with an empty role name.

SUP-5445 - Undo stops working after save

SUP-5448 - Don't overwrite unchanged files

SUP-5456 - Multi-select menu missing entries when selecting two simple blocks in a dialog model

SUP-5482 - Generated generator test won't run

SUP-5499 - Check Button properties are not shown

SUP-5500 - Stepper widget selection marker displays red only, should be dotted red border

SUP-5501 - Checkbutton image properties does not display correctly

SUP-5502 - Stepper widget selection should use the red dotted marker

SUP-5503 - Table widget should not have "Title" and "Frame" property

SUP-5504 - Split pane orientation reads property wrong

SUP-5505 - Window generator properties shows table properties

SUP-5508 - Enumerations don't have property visibility
Release Comment: The visibility property will be read from the model source and set the first time the domain model us updated. The Java Domain generator used to generate all java enums as public but will now set the visibility correctly as specified in the model source.
In Ecore/Xcore models visibility can be set using the g9 annotation visibility.

SUP-5512 - New g9 perspective with palette always visible

SUP-5515 - NPE when changing name on DB model

SUP-5517 - Drag and drop of columns in database model is only working when a tables children are grouped

SUP-5518 - Designer becomes empty after switching from methods tab

SUP-5525 - Refresh linked dialog does not do anything

SUP-5526 - PropertySheet doesn't save changes on focus change sequence

SUP-5528 - Link for runtime gives wrong path separators on linux

SUP-5530 - The 'Linked To' property in dialog elements shows the value as id
Release Comment: The property value is displayed as *dialog.element*.

SUP-5532 - Eclipse freezes when selecting menu title in applications

SUP-5533 - GoTo menu for linked widgets should contain the widget they are linked from

SUP-5534 - MenuItem title change is not refreshed

SUP-5536 - Splitter orientation is shown wrong in design view and in preview
Release Comment: A notebook of type SplitPane was shown with wrong orientation if the Splitter Orientation was explicitly set to either Vertical or Horizontal. If the property was not set the default value was Horizontal and the notebook was displayed correctly both in the design view and in the preview.

SUP-5543 - Swing, React, Ice example wizards missing image

SUP-5551 - The initial width of wizard columns should be calculated to fit the width of the column content and heading

SUP-5556 - There is no default button in the Select Modeling Tool dialog

SUP-5560 - Changes to model in editor is lost
Release Comment: The error occur when having one editor on top, running g9 validate and then bringing another editor on top and editing this by selecting an model item and performing an action that will change the item, i.e. adding a button to a container in a dialog model.

SUP-5563 - No error message when syntactical errors in xml model files
Release Comment: Such errors are now shown as warnings on the error log.

4.2 g9 Database Import

SUP-5278 - DB Import Fetch removes sub packages
Release Comment: It will now remember overrides you made to sub-package as long as the name still matches, regardless of casing. For attributes it will remember if the name still matches also on case since the suggested attribute names are dependent on casing.

SUP-5516 - DB Import Wizard does not show correct help when changing page

SUP-5531 - DB Import Wizard does not refresh project

SUP-5559 - DB Import wizard doesn't pickup re-entered password

4.3 Swing generator

SUP-5242 - Support for model package in the Swing generator
Release Comment: A new generator parameter, *Use model package*, is introduced. If *True*, the Swing generator will append the Dialog Model or Application package to *Client package* for the generated files. The default value is *True*. Set the value to *False* to get the old behavior.

SUP-5303 - Use of enumerations in value objects generate wrong code
Release Comment: The generated code in the <Role>Methods class now uses the correct generated Enum class constructor.

SUP-5306 - NPE when creating value object
Release Comment: In some cases, a new empty domain object was created in the <dialog>Methods class without initializing value object attributes.

SUP-5337 - Way to access Dialog name before the dialog has been instantiated
Release Comment: The dialog name is now generated to the `<dia>Const` interface, with the name `DialogObject._DIALOG_NAME`.

SUP-5394 - Don't set the background color on Table scrollbar buttons
Release Comment: The background color is set by the Swing runtime according to the current theme.

SUP-5398 - Support for Java enums in Swing
Release Comment: The Swing generator and the g9 runtime for Swing now supports native Java enums in addition to G9Enumerator based enums. The generator property "Generate Java style enums" in the Java Domain generator is used to select which type of enum to generate for the domain model.

SUP-5402 - Support for BigDecimal attributes in the Swing generator and g9 runtime
Release Comment: The Swing generator and the g9-swing runtime library now supports both the Numeric and BigDecimal Java classes for numeric (BCD) attributes.

SUP-5434 - Generate @override annotation for the run method in the main application class

SUP-5555 - Generate a Log4j2.xml file when building an Application model

4.4 ICEfaces generator

SUP-5313 - Missing package in ICEfaces if using src-gen

SUP-5486 - Support for BigDecimal attributes in the ICEfaces and Jouteur generators, and g9 runtime
Release Comment: The ICEfaces and Jouteur generators, and the g9-icefaces2 and g9-jouteur runtime jars, now supports attributes of type BigDecimal.

SUP-5521 - Add both src and src-gen to the generated jetty-config.xml
Release Comment: If the generator property value for "Source Directory" is different from "Target Directory", both values are generated to the jetty-config-xml file.

SUP-5555 - Generate a Log4j2.xml file when building an Application model

4.5 Java Domain generator

SUP-5259 - Initial values for attributes are not generated correctly for all java model types
Release Comment: The following data types are corrected: Character and char types are written with surrounding apostrophe '. CR, LF and TAB are coded correctly as \r, \n and \t both in String and in char types.

SUP-5318 - Domain class inheritance needs @MappedSuperclass
Release Comment: When generating java with JPA annotations for domain classes and *Generate single class* is false, the superclass will be generated with the JPA annotation @MappedSuperclass.

SUP-5291 - Use Java 8 for the g9 runtime
Release Comment: The g9 runtime is now built with Java 8 as target. The runtime support for the JSR 310 backport (Java Time) is removed, and substituted with support for the java.time classes.

SUP-5319 - Changes in Java 8 date support
Release Comment: In the general type mapping from model types to g9 types the three default mappings for LocalDate, LocalTime and LocalDateTime are changed and are now specified with their full path: java.time.LocalDate, java.time.LocalTime and java.time.LocalDateTime.
Existing projects will be upgraded with these mappings. However, the old mappings will not be removed in the upgrade process. They must be removed manually if not wanted.
The XSD generator also mapped these three data types and the defaults is changed accordingly.
Note: If an old XSD type mapping does not have a value, i.e. has been using the default value, the old mappings value is set hard to this default value.

SUP-5322 - Initial value for enum attributes doesn't compile
Release Comment: The generated java code for enum initial values now uses the `get(literal)` method.

SUP-5327 - Java generator does not support multiple extensions of interfaces

SUP-5328 - Support for generating Java Bean Validation annotations
Release Comment: Java Bean Validation annotations can now be generated from the Java Domain generator, by setting the *Generate Bean Validation annotations* generator configuration parameter to *true*. The generator supports the `Valid`, `NotNull`, `Size` and `Digits` annotations.

SUP-5329 - Generate JPA with properties from the database profile
Release Comment: Some of the generated JPA annotations uses values from the Database Profile. The default is to use the default Database Profile for the g9 modeling project. By setting the *Database profile for JPA* generator property, the given profile will be used instead. If the given profile is not found in the *Database Profiles* model, a problem is shown in the *Problems View*.

SUP-5346 - Support for filtering the JSON response from REST services
Release Comment: The Java Domain generator now supports adding `@JsonFilter` annotations to domain model classes. This is done by adding a tagged value named *JsonFilter* to the class, with a source http://www.esito.no/g9/json. The value is the filter id as used by Jackson when marshalling JSON from Java objects. The `JsonFilter` annotation is only generated when the *Generate Jackson annotations* generator configuration parameter is *True*.

SUP-5353 - Modeled constructors are generated with a 'return null' statement

SUP-5372 - Support for JPA Index annotations
Release Comment: The Java Domain generator now supports JPA `@Index` annotations. They are generated for groups and attributes with the "Index" DB Property set to `True`, unless the index is also the primary key.

SUP-5427 - Support for generating Jackson annotations
Release Comment: When the *Generate Jackson annotations* generator configuration parameter is *True*, the Java Domain generator will generate some Jackson annotations to the domain classes. If the class has a value in the `JsonFilter` tagged value, a `JsonFilter` annotation will be generated. If the generator configuration parameter *Persistent JSON ignore properties* has a value, a `JsonIgnoreProperties` annotation will be generated for persistent domain classes. If the generator configuration parameter Transient JSON ignore properties_ has a value, a `JsonIgnoreProperties` annotation will be generated for transient domain classes. A `JsonTypeInfo` annotation will be generated to the `lockFlag` property if the type of the property is `java.io.Serializable`, to avoid problems with deserializing the lock flag.

SUP-5429 - Support for generating JAXB annotations
Release Comment: The Java Domain generator now supports generating JAXB annotations for XML marshaling and unmarshaling.

SUP-5527 - Methods support property static

SUP-5562 - Changes in mapping of Numeric and BigDecimal
Release Comment: In the general type mapping from model types to g9 types the two default mappings for Numeric and BigDecimal are changed and are now specified with their full path: no.g9.support.Numeric and java.math.BigDecimal. Existing projects will be upgraded with these two mappings. However, the old mappings will not be removed in the upgrade process. They must be removed manually if not wanted.

4.6 Java runtime

SUP-5396 - The font size for Swing applications should only be scaled on the Windows platform
Release Comment: The font scaling is now only performed on Windows, i.e. no behavior change for Windows.

SUP-5537 - Missing install of g9-common runtime artifacts for Linux and Mac
Release Comment: The g9-common artifacts was missing from the maven-install,sh script.

4.7 EA/Xcore/Java synchronization

SUP-5227 - Check package ID for EcoreExport
Release Comment: When updating from Enterprise Architect the packages was identified by the UML tools own id. For any other model source the packages are identified by the package path. From this version this will be the case also for packages specified with Enterprise Architect. This implies that renaming a package in the UML tool will be the same as adding a new package to the model.

SUP-5230 - StringIndexOutOfBoundsException when doing Domain Model Update
Release Comment: The error occurs when the model contains classes directly under logical view, i.e. without any package.

SUP-5251 - The Java Model import includes the quotes for the initial value for string attributes
Release Comment: The initial value is stored as a string property in the domain model. The value for attributes are set according to type when the initial value consists of a single constant. This is true for all primitive Java types, for their wrapper types, and for String objects. If the initial value of an attribute is java null it will not be set on the attribute in the domin model.
Any other form of initial value will be stored as a string without any form of syntactical analysis.

SUP-5255 - NPE when building after a fetch package structure

SUP-5274 - When Fetch Package Structure fails the real reason is not returned to the user
Release Comment: The underlying error message will be displayed as an error in the Error Log.

SUP-5283 - Checkbox treeview for Java Sync config
Release Comment: The setting of excluded files have been changed to be more user-friendly and intuitive. Now it shows a folder structure from the classpath source folders, where files and folders can be excluded by unchecking in a treeview.

SUP-5360 - Java model update does not remove groups that have been removed from the java source code

SUP-5373 - The JPA annotation @IdClass does not generate a group for the class

SUP-5374 - The JPA annotation @IdClass does not work for associations
Release Comment: Fields interpreted as Associations can be part of an id class and will be added to the group defined on the class containing the JPA @IdClass annotation.

SUP-5417 - Fetching source packages fails in Java Model Configuration dialog
Release Comment: The error occurs when the class path for a java Project References source folders that don't exist in the Project.

SUP-5420 - Cannot open model project after sync

SUP-5432 - Store inner classes, interfaces and enumerators as sub-classes of their surrounding class /interface
Release Comment: When updating the domain model from java or XSD model sources, inner classes, Interfaces and enumerators are supported. In the domain model such elements will be stored as sub elements to their surrounding class/Interface and in the Project browser they will be displayed as such in the same way as attributes and methods are displayed.
For Ecore/Xcore inner classes are supported using the g9 annotation "surroundingClass" The value specifies the surrounding EClass object and should be the name of the class with package path included.
In the generators the IGeneratorContext Interface contains three new methods:

- List<EClass> getClasses(); Returns a list of all EClass elements (classes and Interfaces) in the domain model
- EClass getSurroundingClass(EClassifier); Returns the surrounding class to an inner class, interface or enumerator
- List<EClassifier>getInnerElements(EClass); Returns a list of inner elements, classes, interfaces and enumerators

SUP-5483 - Attributes and parameters of type java.lang.String are not recognized
Release Comment: Field, methods and parameters from java.lang are imported with model type without the path specification java.lang, i.e. String rather than java.lang.String. However types from sub-packages of java.lang is given full path, i.e. java.lang.annotation.IAnnotation.

SUP-5493 - New packages should be selected after a package fetch from model

SUP-5527 - Methods support property static

4.8 RESTful services generator

SUP-4898 - Remove ContextLoaderListener from web.xml in REST server target after Jetty 9.3.x upgrade
Release Comment: The generated web.xml file no longer references the Spring ContextLoaderListener.

SUP-5316 - Missing package when not using generated
Release Comment: When the "generated" package was empty, both the REST client and REST server generators failed.

SUP-5343 - Missing support for the HTTP methods PUT and DELETE
Release Comment: The REST Server generator now support the REST verbs GET, POST, PUT and DELETE. The verb as given in the domain model is no longer case sensitive.

SUP-5344 - Split the Application part from the Spring Boot Controller
Release Comment: When the "Use Spring Boot" generator parameter is *True*, the RestController class no longer is generated with a `@SpringBootApplication` annotation. Instead, a separate application class is generated if the model contains an Application model. Also, the interface implementation is annotated with `@Service`, and the service is autowired in the controller.

SUP-5346 - Support for filtering the JSON response from REST services
Release Comment: The REST Server generator now supports generating controller methods with JSON filtering. Set the *Use JsonFilter* generator property to *True*. Also, filter definitions can be added to interface methods with the tagged value *JsonExcludeFilters*. The format of the tagged value is `filterId1:attr1,...,attrN;filterId2:attr1,...,attrN;...;filterIdN:attr1,...`, where the *filterId* should match one of the *JsonFilter* annotations on the domain classes, and *attrN* is one or more attributes or references in the class with the filterId. For more information, see *Modeling of RESTful web services* in the g9 documentation.

SUP-5399 - The REST Client generator does not generate return statements for POST methods

4.9 Jasper Print generator

SUP-5573 - Update the Jasper Reports dependencies
Release Comment: In the generated POM in the *print* folder, the Jasper Reports version is now 6.4.0, and the Xalan version is 2.7.2. Also, the POM now uses UTF-8 as the source encoding.

SUP-5574 - Jasper prints get wrong page size

4.10 Hibernate generator

SUP-5401 - Support for BigDecimal attributes in the Hibernate generator and g9 runtime
Release Comment: The Hibernate generator and the g9-jgrape runtime jar now supports attributes of type BigDecimal.

4.11 Database MySQL generator

SUP-5304 - Generated MySQL schemas have parameter syntax error

4.12 Known Issues

4.12.1 New example projects can get cyclic dependency errors

This can happen occasionally due to issues with some versions of Xcore/Xtext. New Example projects can get into a cyclic dependency error situation, shown as an exclamation mark on the project node. In this situation "Clean" or reopening the project does not help, however it is usually resolved by restarting Eclipse.

4.12.2 Slow Eclipse workspaces

If you experience that your Eclipse IDE is becoming slow, it is possible to improve. Here are some improvement suggestions:

1. Plugin cache - "eclipse.exe -clean"
   Reloads all plugin cache (safe - you won't lose anything)
2. Local history
   Remove ".metadata/.plugins/org.eclipse.core.resources/.history/"
3. Markers
   In sub-directories of ".metadata/.plugins/org.eclipse.core.resources/.projects/" remove ".markers" and ".markers.snap" files.
   https://codedrawer.wordpress.com/2011/07/25/slow-or-frozen-eclipse-ide-start-after-a-crash/
4. Settings
   When upgrading major version of Eclipse, settings are not cleaned up properly.
   Export your settings, then rename the ".metadata" directory, then import your settings.
   http://blog.pdark.de/2011/09/02/restoring-a-corrupted-workspace-in-eclipse/
5. Create a new Eclipse workspace and import your projects into it.

5 Distribution

The g9-2.6.0.zip distribution consists of the g9 plugin update site and can be downloaded from http://download.esito.no/downloads/tools/g9-2.6.0.zip.

The g9 plugins contain all generators and the Java run-time jars. A list of g9 plugins is displayed using *Help > About Eclipse*, press the g9 icon and the *Plug-in Details* button.

6 Installation

6.1 g9 requirements

The g9 tool-set consists of code that runs on Windows and Linux operating systems. The tool set runs as an Eclipse plug-in and requires Java version 8, and Eclipse Mars or newer. The g9 runtime, however, can be run with Java 7.

6.2 Esito update-site

The g9 Modeling Tools is available from update-site: **http://updatesite.esito.no/g9**.

1. Start Eclipse, go to *Help > Install New Software…*
2. In the *Install dialog*, click the *Add…* button located at the top right of the dialog to add an install site. This brings up the *Add Repository* dialog
3. You can install the most recent version from the remote update site.
    a. In the *Name* field, type **g9 update site**
    b. In the *Location* field, type **http://updatesite.esito.no/g9**
    c. Click OK to return to the Install dialog.
4. Select **g9 update site** in the *Work with* field and select the *Esito Modeling Tools*. Then select the *Next* button.
5. Click *Next* to confirm installation
6. Read and accept the license agreement. To continue installing, select "*I accept the terms of the license agreement*" and click *Finish*.
7. When prompted to restart Eclipse, click *Restart Now* to restart.

6.3 Local update-site

1. Unpack the g9-2.4.1.zip file to a directory of your choice, here called C:\g9
2. Start Eclipse, go to *Help > Install New Software…*
3. In the *Install dialog*, click the *Add…* button located at the top right of the dialog to add an install site. This brings up the *Add Repository* dialog
4. You can install the 2.4.1 version from C:\g9
    a. In the *Name* field, type **g9 local**
    b. Press the "Local…" button and browse to the unpack directory, giving file name C:/g9 in the Location attribute.
    c. Click OK to return to the Install dialog.
5. Select **g9 local** in the *Work with* field and select the *Esito Modeling Tools*. Then select the *Next* button.
6. Click *Next* to confirm installation
7. Read and accept the license agreement. To continue installing, select "*I accept the terms of the license agreement*" and click *Finish*.

8. When prompted to restart Eclipse, click *Restart Now* to restart.

6.4 Xcore modeling

To use Xcore for creating the domain models for g9, the Xcore and Xtext plugins must be installed. Some Eclipse packages already have Xcore and Xtext preinstalled. If not, the recommended update site for Xtext is "Xtext All In One - Releases" http://download.eclipse.org/modeling/tmf/xtext/updates/composite/releases/ . The recommended update site for Xcore is http://download.eclipse.org/modeling/emf/emf/updates/2.10.x/.

For Xcore modeling, the following plugin versions have been tested:

- EMF Xcore version 1.4.0
- Xtext version 2.11.0

6.5 Writing generators in Xtend

To be able to create your own generators in the Xtend language, the Xtend and Xtext plugins must be installed. Some Eclipse packages already have Xtend and Xtext pre-installed. If not, the recommended update site is "Xtext All In One - Releases"

For Xtend support, the following plugin versions have been tested:

- Xtend and Xtext versions: 2.11.0