# Release notes

## IBM Industry Models Utilities

## For models version 8.8.0.0

28th April 2016

# Contents

About these release notes ..... 1
What's in the box ..... 1
Fixes since the previous release ..... 2
Known issues and limitations ..... 3

# About these release notes

These release notes address known product issues and related information about the IBM Industry Models Utilities which are a component of an IBM Industry Models product.

For information about by-design changes, new and enhanced features in the current release, see the “What’s new” topic in the relevant Industry Models Knowledge Center, which are available at:

http://www.ibm.com/support/knowledgecenter

# What’s in the box

Industry Models Utilities is an Eclipse repository containing extensions to Rational Software Architect (RSA), InfoSphere Data Architect (IDA) and Information Governance Catalog (IGC) for Eclipse. The extensions provide additional functionality that can be used with Industry Models in these environments.

The Industry Models Utilities Eclipse repository includes features that contain the extensions to IDA, RSA and IGC for Eclipse. The features are organized into categories that each target a particular environment. The categories help to simplify the installation of the Industry Models Utilities. At installation time a category is chosen that best matches your environment and the relevant features are installed.

The product includes:

- Industry Models RSA Utilities for use with the Banking Process and Service Models and Insurance Process and Service Models
- Industry Models IDA Utilities for use with Banking and Financial Markets Data Warehouse and Insurance Information Warehouse
- Industry Models IGC Utilities for use with all IBM Industry Models products

# Fixes since the previous release

| Reference | Description |
|---|---|
| MD00105 | **Attributes created by the Transfer Object Generation utility are created in random order**<br><br>Previously Transfer Object attributes created by the Transfer Object Generation utility were created in random order.<br><br>Now Transfer Object attributes are created in this order:<br>• Attributes from type (as per the storage order of the super-type)<br>• Attributes from Type Interfaces realized by the type (in the parameter order)<br><br>starting with top-level super-type, then next super-type, repeating until source class has been processed |
| MD00106 | **IDA Model Publishing utility fails when non-model files are in the project**<br><br>Previously the Prepare IDA models for Web Publishing utility failed if any file other than a data model file existed in the IDA project. |
| 27380 | **WSDM ServiceInterface operation validation constraint incorrectly warns for valid isException parameters**<br><br>Previously the "WSDM <<ServiceInterface>> Operations should have Document Centered Style parameters" validation constraint incorrectly issued warnings for valid isException parameters.<br><br>Now the validation constraint just checks that isException parameters:<br>• are of type DataType, Class or Signal [the type can be defined in any model]<br>• have the <<MessageType>> stereotype applied |
| 29857 | **Transfer Object wizard does not appear for a Utility class with no Component Realization**<br><br>Previously for Classes in the Utility Components package in the IDM that had not yet been assigned to a Component, running the Transfer Object Generation utility failed immediately and the wizard was not displayed. |
| 31188 | **Transfer Object Generation utility warning when creating new Transfer Object with a self-referencing attribute**<br><br>Previously the Transfer Object Generation utility issued warnings and did not create Transfer Object attributes for IDM classes that contained self-referencing attributes. |
| 31662 | **RSA Publishing creates invalid links from Transfer Object attributes to parameters in IDM Type Interface Operations**<br><br>Previously RSA Publishing created invalid hyperlinks in the generated HTML files for Transfer Object attribute to IDM Type Interface Operation parameter <<derive>> relationships |

# Known issues and limitations

| Reference | Description |
|---|---|
| 23591 | **Industry Models SOA validation constraints must allow for the use of subtypes of RequestHeader and ResponseHeader.**<br><br>WSDM service parameter <<MessageType>> data types must have a request or response attribute of the RequestHeader or ResponseHeader type.<br><br>Currently, if you create subtypes of RequestHeader or ResponseHeader and use these as the types of the WSDM service parameter <<MessageType>> data type attributes, when you validate WSDM, warnings state that these attributes must be of the RequestHeader or ResponseHeader type.<br><br>*Workaround:* disable the following Industry Models validation constraints in Preferences in RSA:<br><br>• Request WSDM Service Parameter MessageType should contain an attribute of type RequestHeader<br><br>• Response WSDM Service Parameter MessageType should contain an attribute of type ResponseHeader |
| 23602 | **Add Simple Object Access Protocol (SOAP) fault exception parameters to WSDM ServiceInterface operations.**<br><br>WSDM ServiceInterface operations can contain three parameters:<br><br>• Request “in”<br><br>• Response “out”<br><br>• SOAP fault exception<br><br>SOAP fault exception parameters are created as SOAP faults in the generated WSDL files, and enable a better error handling strategy for your SOAP services:<br><br>• Information about minor issues or warnings can be returned in the response header in the body of the response<br><br>• Serious issues can be returned as SOAP faults<br><br>*Workaround:* to manually create a SOAP fault exception parameter:<br><br>1. Either create a data type in the WSDM definitions package for SOAP fault information, or customize the ErrorInfo IDM data type.<br><br>2. Create a parameter on a service interface operation with the direction set to Out.<br><br>3. Set the parameter type to the SOAP fault data type you created in the WSDM definitions package or to the ErrorInfo IDM data type.<br><br>4. Set the parameter as an exception parameter:<br><br>• In the Properties > Service view, select Exception, or<br><br>• In the Properties > Advanced view, set Is Exception to True<br><br>The exception parameter you created can be copied and pasted into other service interface operations as required. |

| Reference | Description |
|---|---|
| 31089 | **Transfer Object Generation utility does not support IDM class operation parameters**<br><br>The Transfer Object Generation utility allows you to create Transfer Object attributes from IDM class attributes, associations and type interface operation parameters. However, currently it does not support the creation of Transfer Object attributes from IDM class operation parameters.<br><br>*Workaround:* Manually create the required Transfer Object attributes. Alternatively, add an equivalent operation to a Type Interface that the IDM class realizes |
| 31790 | **JSDM Service Generation utility fails for source BOM Capability Operation with an unnamed or un-typed return parameter**<br><br>The JSDM Service Generation utility fails with the following error when the source BOM Capability Operation has an unnamed return parameter:<br><br>• ERROR : Exception caught during creation of JDM Interface Operation operation2: Cannot create attribute without a name: <Response DataType name><br><br>The JSDM Service Generation utility fails with the following errors when the source BOM Capability Operation has a return parameter with no type:<br><br>• ERROR : Cannot get design model type for BOM source parameter "null". BOM source parameter type is null.<br><br>• FATAL : Modification failed: "Create JDM Interface Operations based on BOM Capability Operations" - null<br><br>*Workaround:* ensure that BOM Capability Operation return parameters have a valid type and a name. |
| 32192 | **BOM to IDM Element Generation utility fails creating IDM component class for source BOM class with attributes that have no documentation**<br><br>The BOM to IDM Element Generation utility fails with the following error when the *Create a Component Class* option is enabled and the source BOM class contains attributes with no documentation:<br><br>• FATAL : Modification failed: "Create IDM element from BOM Element" - null<br><br>*Workaround:* ensure that BOM attributes have documentation. |
| n/a | **Industry Models validation incorrectly reports problems when UML models are not open.**<br><br>When you validate BPS/IPS for Standard Tooling models or projects and all UML models are not open, validation errors or warnings can display to indicate that intermodel mappings are missing.<br><br>*Workaround:* ensure that all service models are open in the BPS/IPS for Standard Tooling project before you run Industry Models validation. |