


# RPG 3.1

- [Extension ID](#)
- [What's new?](#)
- [Description](#)
- [In what situation should you install this extension?](#)
- [Supported languages](#)
- [Function Point, Quality and Sizing support](#)
- [CAST AIP compatibility](#)
- [Supported DBMS servers](#)
- [Prerequisites](#)
- [Download and installation instructions](#)
- [Prepare and deliver the source code](#)
  - [Source code preparation](#)
  - [What about IBM Db2 for i source code?](#)
  - [Deliver the source code](#)
    - [Using AIP Console](#)
    - [Using the CAST Delivery Manager Tool](#)
  - [Analysis configuration](#)
    - [Using AIP Console](#)
    - [Using CAST Management Studio](#)
  - [Configuring Technical Size measures for RPG in the CAST Health Dashboard](#)
- [What results can you expect?](#)
  - [Links](#)
  - [Objects](#)
    - [CL400](#)
    - [DDS400](#)
    - [RPG 300](#)
    - [RPG 400](#)
  - [Analysis messages](#)
    - [RPG-001](#)
    - [RPG-002](#)
    - [RPG-003](#)
    - [RPG-004](#)
  - [Structural Rules](#)

 **Summary:** This document provides information about the extension providing **RPG** support.

## Extension ID

`com.castsoftware.rpg`

## What's new?

RPG 3.1 is a full rewrite of RPG analyzer. Major changes are:

- Removal of DB400 Universal Importer and DMT extractor; support for the analysis of **IBM Db2 for i** is replaced by the **SQL Analyzer** ( 3.5.4 is required).
- Removal of source code preprocessor, real source code is now visible.
- Removal of MENU type objects.
- Removal of \*.bnd analysis and Bound Service Program type objects.

Please see [RPG 3.1 - Release Notes](#) for more information.

## Description

This extension provides support for applications written using RPG languages and/or CL languages.

## In what situation should you install this extension?

If your application contains source code written using RPG and or CL and you want to view these object types and their links with other objects, then you should install this extension.

## Supported languages

- **RPG III** (also known as GAP 3)
- **RPG IV** (fixed-form and free format)
- **CL**

## Function Point, Quality and Sizing support

This extension provides the following support:

- **Function Points (transactions)**: a green tick indicates that OMG Function Point counting and Transaction Risk Index are supported
- **Quality and Sizing**: a green tick indicates that CAST can measure size and that a minimum set of Quality Rules exist

Function Points (transactions)	✓
Quality and Sizing	✓

## CAST AIP compatibility

This extension is compatible with:

CAST AIP release	Supported
8.3.x	✓

## Supported DBMS servers

This extension is compatible with the following DBMS servers:

DBMS	Supported
CAST Storage Service/PostgreSQL	✓

## Prerequisites

✓	An installation of any compatible release of CAST AIP (see table above)
---	---

## Download and installation instructions

Please see:

- [Download an extension](#)
- [Install an extension](#)

## Prepare and deliver the source code


When the extension is downloaded and installed, you can now package your source code and run an analysis. The process of preparing and delivering your source code is described below:

### Source code preparation

Before the RPG source code can be delivered and then analyzed, it needs to be collected **from the proper iSeries libraries** and transferred to the designated location. During this operation, each artifact (program, CL, DSPF) must be put into a single file only. This will result in a single artifact per file. Furthermore, the type of source code must be expressed using the extension to the file. The appropriate file extensions are listed in the following table.

Source code is kept in a member of a file. The file is kept in a library. Each file can have many members. Each member is the source code for a program. The source code files can have any name but conventionally the names start with Q and end with SRC, for source.


Source code type	iSeries library	Required extension	Used in MetaModel
RPG-III programs	QRPGSRC	*.rpg, *.rpg38	RPG300
ILE RPG Programs	QRPGLESRC	*.rpgle	RPG400
ILE RPG Programs with SQL	QSQLRPGLESRC	*.sqlrpgle	RPG400
Copy source members	QCPYLESRC	*.cpyle	RPG400
CL programs	QCLSRC	*.cl, *.clp, *.cl38, *.clp38	CL400
ILE CL Programs	QCLLESRC	*.clle	CL400
Display Files	QDDSSRC	*.dspf, *.dspf38	DDS400
Printer Files	QDDSSRC	*.prtf, *.prtf38	DDS400
Logical Files	QDBSRC	*.lf, *.lf38	SQL Analyzer/DDS400
Physical Files	QDBSRC	*.pf, *.pf38	SQL Analyzer/DDS400

 The RPG Analyzer is able to autodetect RPG III versus RPG IV, so there is no difference in using \*.rpg or \*.rpgle. What is very important is to distinguish between Display, Printer, Logical and Physical files using the specific file extensions listed in above table.

## What about IBM Db2 for i source code?


In previous releases of the RPG Analyzer extension, when **IBM Db2 for i** source code was delivered, a dedicated extraction process was launched within the RPG Analyzer to transform this source code into a format that could be analyzed. Starting in release **3.1.x**, the RPG Analyzer is no longer responsible for analyzing source code from **IBM DB2 for i**. Instead, this task is undertaken by the **SQL Analyzer** (release **3.5.4** is required).

There is no change to source code delivery process when using AIP Console - the **IBM DB2 for i** source code should be delivered together with the RPG source code and AIP Console will be able to detect this type of source code and create the necessary Analysis Units. For those using legacy CAST Management Studio/Delivery Manager Tool, the source code delivery process has changed and is explained below.

 IBM Db2 for i is also known as DB400.

## Deliver the source code

### Using AIP Console

 RPG is supported in **AIP Console 1.22**.

#### Note about using RPG 3.1.0 with AIP Console 1.22, 1.23, 1.24 and 1.25

**RPG 3.1.0** does not function correctly with **AIP Console 1.22, 1.23, 1.24 and 1.25**, however, a workaround exists. Before delivering your source code, locate the following file on the AIP Node that will be performing the analysis:

```
%PROGRAMDATA%\CAST\AipConsole\AipNode\code-scanner-config.xml
```

Edit the file and remove or comment out the following section:

```
<package packageType="dmtdb400techno.DB400Package"
  extractorType="dmtdb400extractor.DB400Extractor"
  deliveryFile=".lf;.pf;.sqli;.sqlt;.sqlv;.sqlu;.sqlp"
  deliveryData="COE" />
```

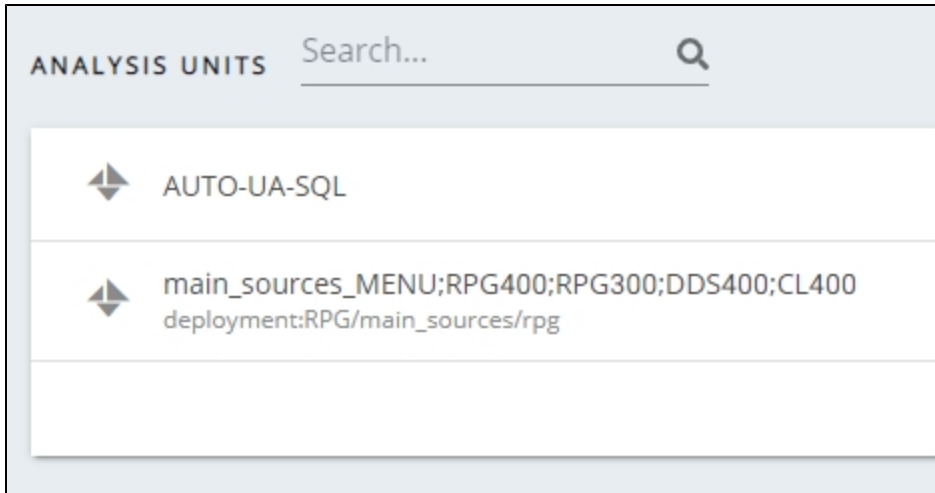
Save the file and restart the AIP Node.

AIP Console expects either a **ZIP/archive file** or **source code located in a folder** configured in AIP Console. You should include in the ZIP/source code folder all **RPG** and **IBM DB2 for i** (if present) source code. CAST highly recommends placing the files in a folder dedicated to RPG. If you are using a ZIP /archive file, zip the folders in the "temp" folder - but do not zip the "temp" folder itself, nor create any intermediary folders:

```
D:\temp
|-----RPG
|-----OtherTechno1
|-----OtherTechno2
```

When the source code has been delivered, AIP Console will create an Analysis Unit for:

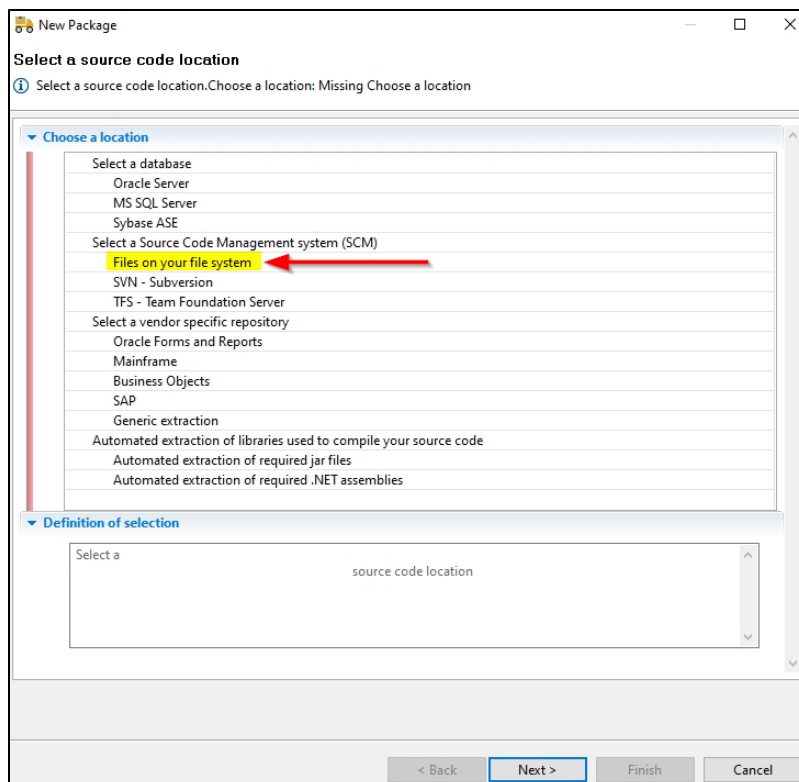
- **IBM Db2 for i** (if present)
- **RPG**



## Using the CAST Delivery Manager Tool

Create a new **Version**, create a new **Package** for your **RPG source code** using the **Files on your file system** option and then choose the location of your source code:

*Click to enlarge*



Run the **Package** action. Before delivering the source code, check the **packaging results**.

## Analysis configuration

### Using AIP Console

Add a dependency between the **RPG Analysis Unit** as **Source** and **SQL Universal Analysis Unit** as the **target**:



The screenshot shows the 'Dependencies' section in the AIP Console. It features a table with columns for 'Origin', 'Source', and 'Target'. A single dependency is listed with 'main\_sources\_MENU;RPG400;RPG300;DDS400;CL400' as the source and 'AUTO-UA-SQL' as the target. A '+ ADD' button is visible in the top right corner of the table area.

Origin ↓	Source	Target
	main_sources_MENU;RPG400;RPG300;DDS400;CL400	AUTO-UA-SQL

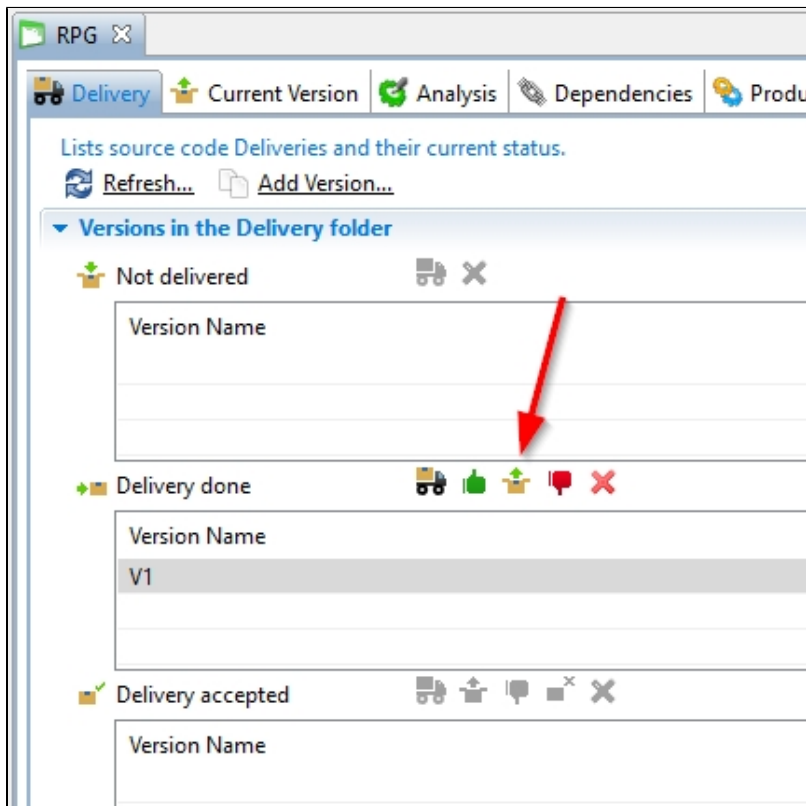
Rows per page: 10 1 - 1 of 1

Starting from **AIP Console 1.26**, this dependency will be automatically added and no manual intervention is required. You should still check that the dependency is present, however.

### Using CAST Management Studio

**Accept** the version and set it as **current**:

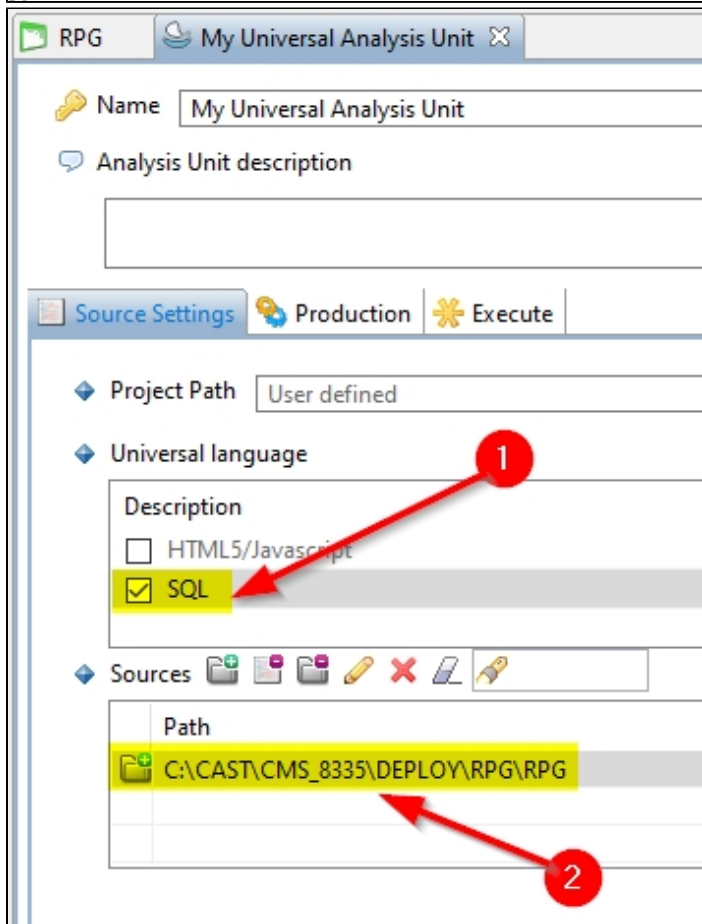
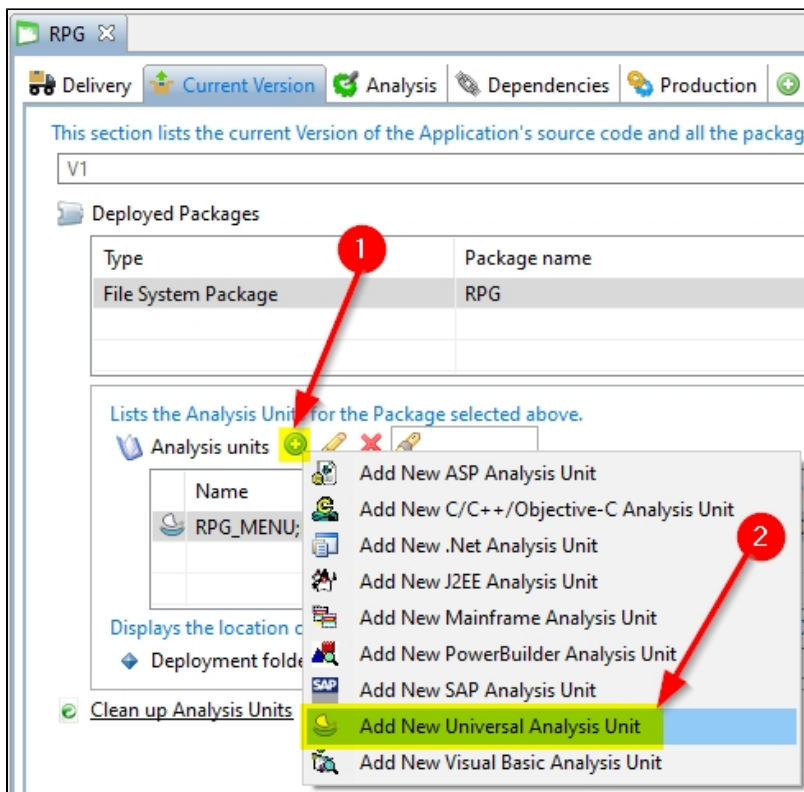
*Click to enlarge*



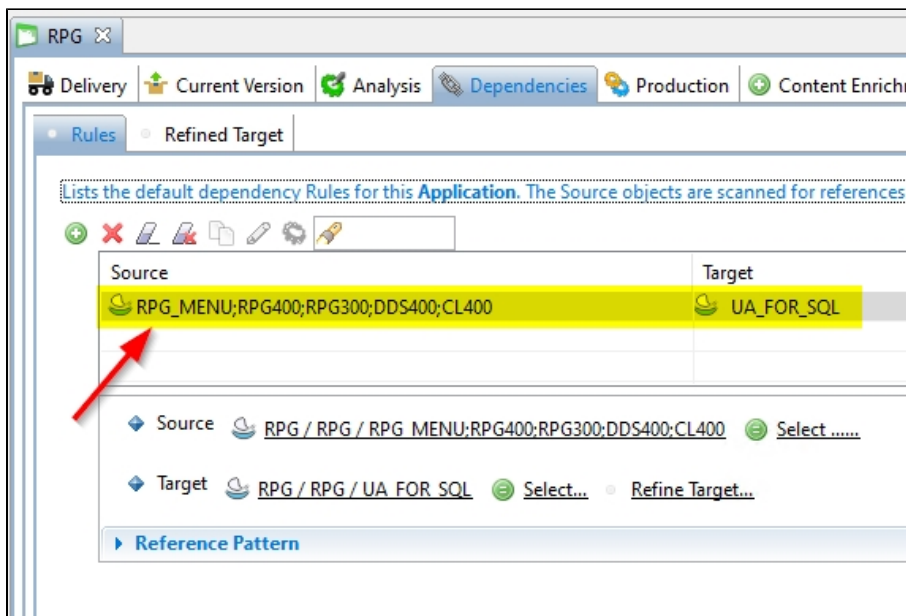
The screenshot shows the 'RPG' project window in CAST Management Studio. The 'Delivery' tab is active, displaying a list of versions in the delivery folder. The 'Delivery done' section is highlighted, and a red arrow points to the 'Delivery done' status icon. The 'V1' version is listed under 'Delivery done'.

Version Name
V1

Now add a **Universal Analysis Unit** pointing to the root deploy folder. This will be used to analyze the **IBM Db2 for i** source code:



Finally, in the **Application editor** > **Dependencies** tab, add a dependency between the **RPG Analysis Unit** as **Source** and **SQL Universal Analysis Unit** as the **target**:



## Configuring Technical Size measures for RPG in the CAST Health Dashboard

Technical Size measures can be displayed for RPG analysis results in the CAST Health Dashboard by manually editing the following file:

```
1.x WAR file: %CATALINA_HOME%\webapps\

```

Add the following entries into the existing section "**TechnicalSizeMeasures**":

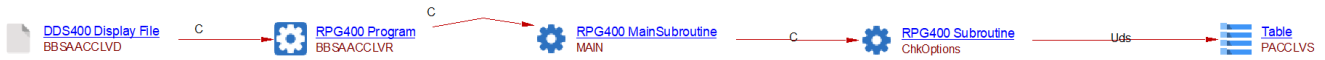
```
"TechnicalSizeMeasures": {
...
  {
    "id": 1008000,
    "label": "Number of RPG400 Program(RPG400)"
  },
  {
    "id": 1008001,
    "label": "Number of RPG400 Subroutine(RPG400)"
  },
  {
    "id": 1008002,
    "label": "Number of RPG400 Procedure(RPG400)"
  },
  {
    "id": 1008003,
    "label": "Number of RPG IV Copy Member(RPG400)"
  },
  {
    "id": 1009000,
    "label": "Number of RPG300 Program(RPG300)"
  },
  {
    "id": 1009001,
    "label": "Number of RPG300 Subroutine(RPG300)"
  },
  {
    "id": 1009003,
    "label": "Number of RPG300 Copy Member(RPG300)"
  }
}
```

Following any changes you make, **save the app.json** file and then **restart** your application server so that the changes are taken into account.

## What results can you expect?

### Links

```
C      SCRLVL      CHAIN      PACCLVS
C      IF          %FOUND
C      DELETE     RACCLV
C      ENDIF
```



```
CALL      PGM(&EXTLIB/&EXTPGM)
```



```
exec sql update vrtfilep set vrupdf = ' ' where vrupdf <> ' ' ;
```





### Objects

#### CL400

Icon	Object Name
	CL Program
	CL Project
	CL Subroutine
	CL Call to Generic Program
	CL EXEC SQL





#### DDS400















Icon	Object Name
	DDS Project
	DDS Section
	DDS Printer File
	DDS Display File




## RPG 300

Icon	Object Name
	RPG III Project
	RPG III Program
	RPG III MainSubroutine
	RPG III Subroutine
	RPG III File Printer
	RPG III File Disk
	RPG III File Workstn
	RPG III File Special
	RPG III File Seq
	RPG III Copy Member
	RPG III SQL Statement/Structure
	RPG III Procedure
	RPG III Call to Generic Program
	RPG III EXEC SQL

	RPG III Missing Copy Member
	RPG III Missing Physical/Logical File
	RPG III Missing Display File
	RPG III Missing Printer File

## RPG 400

Icon	Object Name
	RPG IV Project
	RPG IV Program
	RPG IV MainSubroutine
	RPG IV Subroutine
	RPG IV File Printer
	RPG IV File Disk
	RPG IV File Workstn
	RPG IV File Special
	RPG IV File Seq
	RPG IV Procedure
	RPG IV Procedure Prototype
	RPG IV Copy Member
	RPG IV Call to Generic Program
	RPG IV EXEC SQL

	RPG IV Module
	RPG IV Missing Copy Member
	RPG IV Missing Physical/Logical File
	RPG IV Missing Display File
	RPG IV Missing Printer File
	RPG IV Missing Procedure

## Analysis messages

### RPG-001

<b>Identifier</b>	RPG-001
<b>Message</b>	Cannot resolve copy ...
<b>Severity</b>	Warning
<b>Explanation</b>	Analyser could not find the source file included by a /copy statement. This can lead to various missing results.
<b>User Action</b>	Ensure that all required source code is packaged and delivered.

### RPG-002

<b>Identifier</b>	RPG-002
<b>Message</b>	Cannot resolve DDS file ...
<b>Severity</b>	Warning
<b>Explanation</b>	Analyser could not find a DDS file (pf, lf, dspf, prtfl, ...)
<b>User Action</b>	Ensure that all required source code is packaged and delivered.

### RPG-003

<b>Identifier</b>	RPG-003
<b>Message</b>	Cannot resolve program ...
<b>Severity</b>	Warning
<b>Explanation</b>	Analyser could not find a program.
<b>User Action</b>	Ensure that all required source code is packaged and delivered.  The program can also be in a different technology (COBOL, etc...) in that case it will be linked in a latter analysis step.

### RPG-004

<b>Identifier</b>	RPG-004
<b>Message</b>	Cannot resolve procedure ...
<b>Severity</b>	Warning
<b>Explanation</b>	Analyser could not find a procedure.
<b>User Action</b>	Ensure that all required source code is packaged and delivered.

## Structural Rules

The following structural rules are provided:

<b>3.1.1-funcrel</b>	<a href="https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.1-funcrel">https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.1-funcrel</a>
<b>3.1.0-funcrel</b>	<a href="https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-funcrel">https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-funcrel</a>
<b>3.1.0-beta4</b>	<a href="https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta4">https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta4</a>
<b>3.1.0-beta3</b>	<a href="https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta3">https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta3</a>
<b>3.1.0-beta2</b>	<a href="https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta2">https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta2</a>
<b>3.1.0-beta1</b>	<a href="https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta1">https://technologies.castsoftware.com/rules?sec=srs_rpg&amp;ref= 3.1.0-beta1</a>

## Limitations

- in RPG
  - dynamic calls and dynamic SQL is not supported