

RPG 2.0

- [Extension ID](#)
- [What's new?](#)
- [Description](#)
- [In what situation should you install this extension?](#)
- [Supported Versions of RPG](#)
- [Embedded SQL support](#)
- [Function Point, Quality and Sizing support](#)
- [CAST AIP compatibility](#)
- [Supported DBMS servers](#)
- [Prerequisites](#)
- [Download and installation instructions](#)
 - [CAST Transaction Configuration Center \(TCC\) configuration](#)
 - [Manual import action for CAST AIP 8.2.x](#)
- [Prepare and deliver the source code](#)
 - [Source code preparation](#)
 - [Source code preprocessing](#)
 - [Deliver the source code](#)
 - [Using AIP Console](#)
 - [Using the CAST Delivery Manager Tool](#)
 - [Dependency configuration in the CAST Management Studio](#)
 - [Configuring Technical Size measures for RPG in the CAST Health Dashboard](#)
 - [Logging mechanism](#)
- [What results can you expect?](#)
 - [RPG MetaModel](#)
 - [Objects](#)
 - [MENU](#)
 - [CL400](#)
 - [DB400](#)
 - [DDS400](#)
 - [RPG 300](#)
 - [RPG 400](#)
 - [Structural Rules](#)
- [Limitations](#)
 - [False violations reported for the Quality Rule 7388: Avoid artifacts having recursive calls](#)



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

Extension ID

`com.castsoftware.rpg`

What's new?

Please see [RPG 2.0 - What's new](#) for more information.

Description

This extension provides support for applications written using RPG languages.



Although this extension is officially supported by CAST, please note that it has been developed within the technical constraints of the CAST Universal Analyzer technology and to some extent adapted to meet specific customer needs. Therefore the extension may not address all of the coding techniques and patterns that exist for the target technology and may not produce the same level of analysis and precision regarding e.g. quality measurement and/or function point counts that are typically produced by other CAST AIP analyzers.

In what situation should you install this extension?

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

Supported Versions of RPG

- **RPG III** (also known as GAP 3)
- **RPG IV** with traditional operation codes
- **RPG IV** with alternate Calc spec and Extended Factor 2
- **RPG IV** with /FREE directive

Embedded SQL support

Embedded SQL is supported, primarily to obtain Usiud links from RPG400 and CL400 objects to SQL Tables and SQL Views. The resolution of these link types requires that the database (DB2 UDB) is delivered and analyzed, however, the DB400 extractor can be used to simulate this database.

- The following source code is considered as embedded SQL:
 - For **RPG400**: C/EXEC SQL ... C/END-EXEC
 - For **RPG Free Format**: EXEC SQL ... ;
 - For **CL**: RUNSQL(...)
- Dynamic execution is not supported.

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
CSS/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](#)

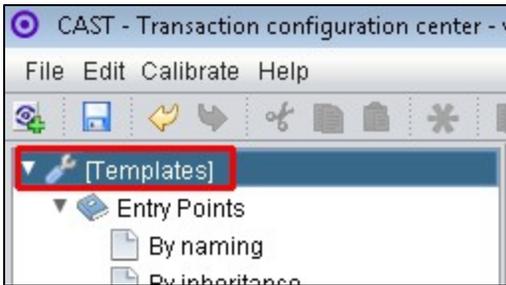
CAST Transaction Configuration Center (TCC) configuration

A set of RPG Transaction Entry/End Points, Data Entities and Excluded Items for use in the CAST Transaction Configuration Center is delivered in the extension via a .TCCSetup file. Therefore if you are using:

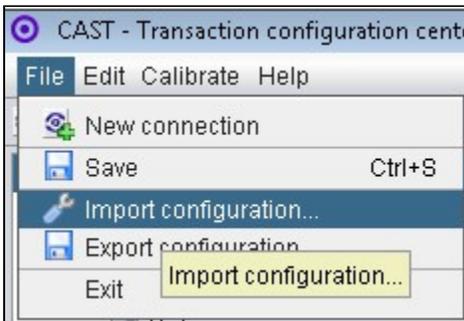
- **CAST AIP 8.3.x**, there is nothing for you to do: this configuration will be automatically imported during the extension installation and will be available in the CAST Transaction Configuration Center
- **CAST AIP 8.2.x**, you can manually import the file %PROGRAMDATA%\CAST\CAST\Extensions\com.castsoftware.rpg.<version>\RPG.TCCSetup to obtain configuration (see instructions below).

Manual import action for CAST AIP 8.2.x

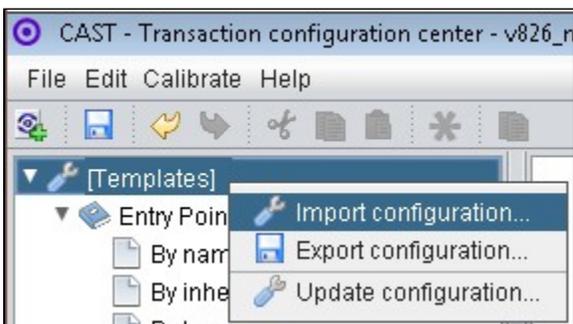
- Locate the .TCCSetup file in the extension folder: %PROGRAMDATA%\CAST\CAST\Extensions\com.castsoftware.rpg.<version>\RPG.TCCSetup
- In the CAST Transaction Configuration Center, ensure you have selected the **Templates** node:



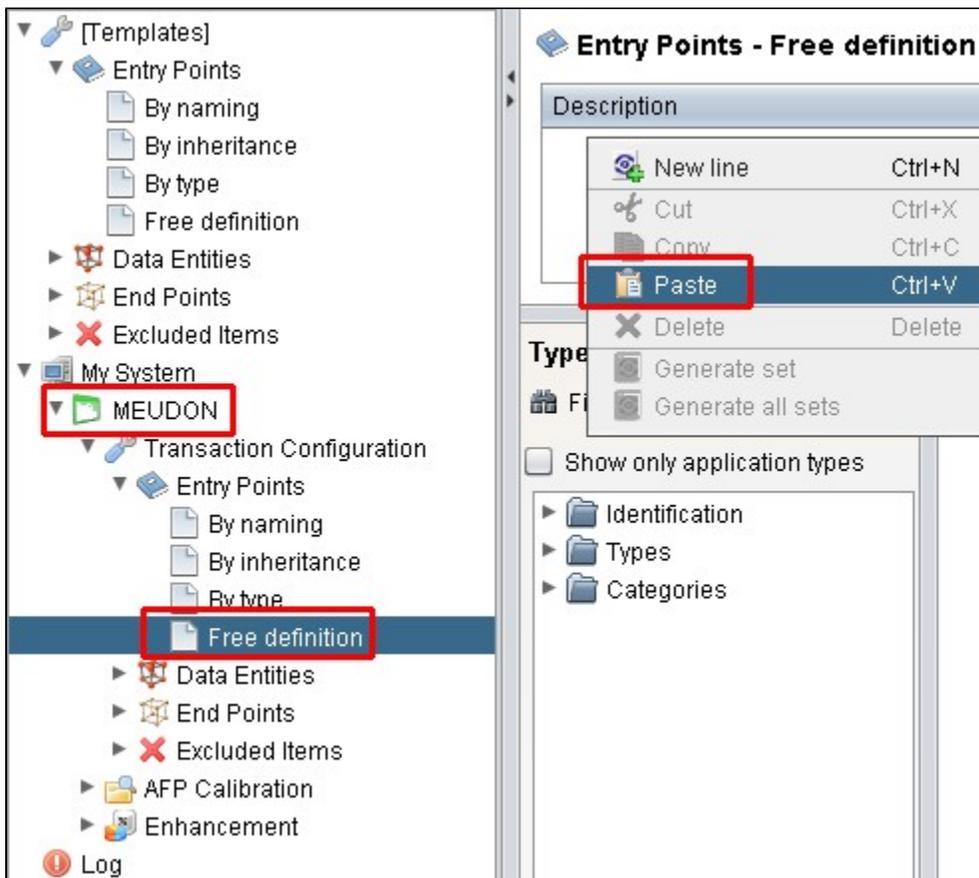
- This .TCCSetup file is to be imported into the CAST Transaction Calibration Center using either the:
 - **File > Import Configuration** menu option:



- Or right clicking on the **Template node** and selecting **Import Configuration**:



- The import of the "RPG.TCCSetup" file will provide you with a sample configuration under Templates (click to enlarge)
- Now right click any of the imported items and select copy
- Paste the item into the **equivalent node** under the **Application**, for example, below we have copied it into the **Application MEUDON**:



- Repeat for any additional items or generic sets that have been imported from the .TCCSetup file.

Prepare and deliver the source code

Once 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 files 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	QSQRPGLESRC	*.sqlrpgle	RPG400
Copy source members	QCPYLESRC	*.cpyle	RPG400
CL programs	QCLSRC	*.cl, *.clp, *.cl38, *.clp38	CL400
ILE CL Programs	QCLLESRC	*.cle	CL400
Display Files	QDDSSRC	*.dspf, *.dspf38	DDS400
Printer Files	QDDSSRC	*.prt, *.prt38	DDS400
Logical Files	QDBSRC	*.lf, *.lf38	DB400/DDS400
Physical Files	QDBSRC	*.pf, *.pf38	DB400/DDS400

Bound service programs	QBNDPGM	*.bnd	RPG400
Menu	QMNUSRC orQMENUSRC	*.menu	MENU
SQL Files		*.SQL	DB400

Source code preprocessing

RPG source code needs to be preprocessed so that CAST can understand it and analyze it correctly. In previous releases of the extension, this preprocessing was a **manual action** that needed to be completed **before** the code was analyzed. However, in this release and all future releases, the code preprocessing is **actioned automatically** when an analysis is launched or a snapshot is generated (the code is preprocessed before the analysis starts). In other words you only need to package, deliver and launch an analysis/generate a snapshot for the preprocessing to be completed.

RPG Analyzer provides specific pre-processors for each language embedded in this extension: the preprocessor per language only considers the files with the expected extensions.

 Note that the CAST Management Studio will use the LISA folder to analyze the preprocessed files.

Deliver the source code

Using AIP Console

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

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 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
```

The following file extensions will trigger the extractor to process the **IBM Db2 for i related files** correctly - so ensure that at least one of these is delivered if you have **IBM Db2 for i definitions (DDS and/or DDL)**:

- .lf
- .pf
- .sql
- .sqli
- .sqlt
- .sqlv
- .sqlu
- .sqlp

When the source code has been delivered, AIP Console will create:

- **One RPG package** with the provided source code. The associated Analysis Unit will end with a suffix as shown below:

_MENU;RPG400;RPG300;DDS400;CL400

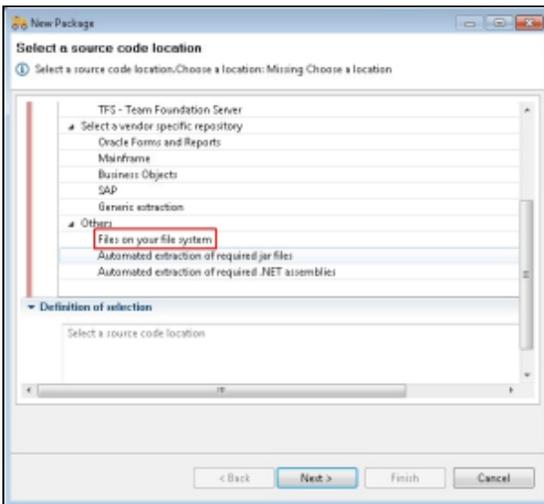
- **One RPG DB400 package** containing source code with the **specific IBM Db2 for i file extensions** listed above. The associated Analysis Unit will have a name depending on what has been discovered and will be extracted into a subfolder named as below:

/RPG400-Package

Using the CAST Delivery Manager Tool

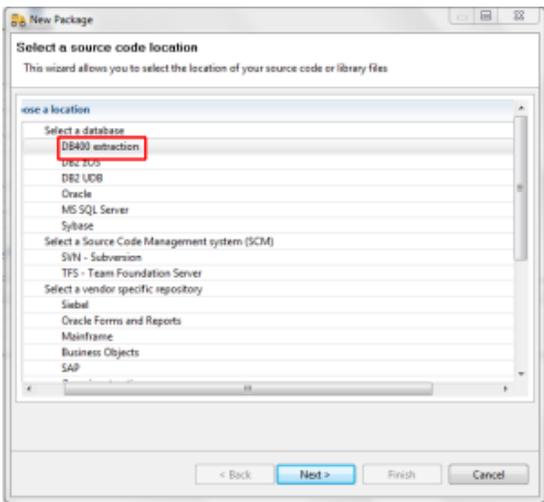
- create a new **Version**
- create a new **Package** for your **RPG source code** using the **Files on your file system** option and choose the location of your source code:

Click to enlarge



- create an additional new package for your Tables and Views using the **DB400 extraction** option:

Click to enlarge



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

Dependency configuration in the CAST Management Studio

In the **Application editor** > **Dependencies** tab, add a dependency between the **Universal Analysis Unit** as **Source** and **LIB** (UDB DB2 schema) 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 RPG400 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.

Logging mechanism

During the analysis, the Universal Analyzer may throw errors or warnings. The table below lists the most significant errors/warnings and lists a suggested remediation action:

Tools	Error/Warning Message	Action
Preprocessor	File 'FileName' is empty	This is caused by source code that is referring file is empty. You can safely ignore this warning.
Preprocessor	Unable to open file 'FullFileName'	This is caused by source code that is referring file that cannot be found in the preprocessed code. You can safely ignore this warning.
Preprocessor	PF corresponding to the table 'tableName' was not found	This is caused by source code that is referring the Table that cannot be found in the preprocessed code. You can safely ignore this warning.
Preprocessor	File Disk 'FDescName' not found : missing record	The record will not be created. This is caused by source code that is referring the FDescName that cannot be found in the preprocessed code. You can safely ignore this warning.
Preprocessor	File Disk 'FDescName' not found : renamed record 'RecordName'	The record will not be created. This is caused by source code that is referring the FDescName that cannot be found in the preprocessed code. You can safely ignore this warning.
Preprocessor	File Disk 'FDescName: renamed record 'RecordName' doesn't match the DDS record 'RecordName'	This is caused by source code that is referring the FDescName that cannot be found in the preprocessed code
Preprocessor	Error NOMAIN in a PROGRAM	The program will not be preprocessed. Program will not be created
Preprocessor	Copy not found 'CopyName'	This is caused by source code that is referring the called Copy that cannot be found in the preprocessed code. You can safely ignore this warning.
Preprocessor	Missing called program='PgmName'	This is caused by source code that is referring the called program that cannot be found in the preprocessed code. You can safely ignore this warning.
Preprocessor	Error on match if/endif at line 'LineNumber'	Object will not be created.

Preprocessor	File does not match RPG III format (RPG IV)	This means file is <i>not</i> RPG III and probably is RPG IV.
Preprocessor	File does not match RPG IV format (RPG III)	This means file is <i>not</i> RPG IV and probably is RPG III.
Analyzer	Unable to find the object end for type 'RPG400Subroutine'	This is caused by source code that is referring to a SubRoutine that cannot be found. You can safely ignore this warning.
Analyzer	Unable to find the regexp for the link 'accessLink'	The link will not be created -
Analyzer	An unnamed object of type 'RPG400SQLStructure' has been detected	This is because RPG400SQLStructures have no specific name, as such, the CAST framework will give them the name "unnamed". You can safely ignore this warning.
Analyzer	Duplicate object of type 'RPG400SQLStructure' has been detected: '<unnamed>'	As mentioned above, structures have no specific name and are automatically given the name "unnamed" by CAST - as such multiple objects with the name "unnamed" will exist and will cause this error. You can safely ignore this warning.
Analyzer	Unable to find the object end for type 'RPG400Rule'	This is caused by source code that is referring to a Rule cannot be found. You can safely ignore this warning.
Analyzer	Duplicate object of type 'MENUItem' has been detected: 'XXXXXX'	This is due to a duplicate Menu name in the original Menu file. You can safely ignore this warning.
Analyzer	Warning after preprocessing the source code- Unknown format for ' A XA3439 R REFFLD(DATM FRFGFDI4)'	<p>Follow Sample Reference Function Specifications</p> <pre> 1.....+.....2.....+.....3.....+.....4.....+.....5.....+.....6.....+..... 7.....+.....8 00010A REF(FILE1) (1) 00020A R RECORD1 00030A FIELD1 R (1) 00040A FIELD2 R (1) 00050A FIELD3 R REFFLD(FLD3) (2) 00060A FIELD4 R REFFLD(FLD4 FILE2) (3) 00070A FIELD5 R REFFLD(FLD5 LIB1/FILE3) (4) 00080A FIELD6 R REFFLD(RECORDB/FLD6 LIB1 /FILE4) (5) 00090A FIELD7 R REFFLD(FIELD6 *SRC) (6) 00100A FIELD8 R REFFLD(FLD6) (7) 00110A R RECORD2 00120A FIELD1 20 (8) 00130A 00140A R RECORD3 00150A FIELD1 R REFFLD(RECORD2/FIELD1 *SRC) (9) 00160A 00170A R RECORD4 00180A FIELD1 R REFFLD(FIELD1 *SRC) (10) A </pre> <p>Note:For line 00010, you can also specify library name and record format name. See the REF keyword example/</p> <p>You must specify R in position 29 for each field that refers to another field that was previously defined.</p> <p>RPG preprocessor supports only the cases (2) and (7) : Only one parameter for the REFFLD function.</p>

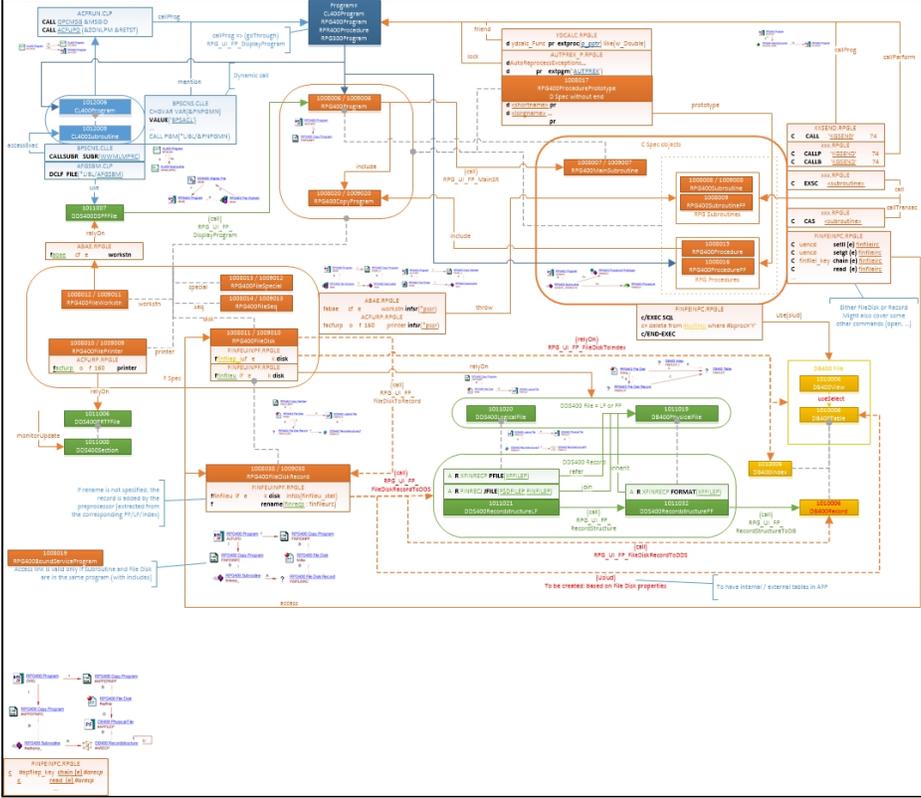
For XXL applications, if any performance issue is experienced during the "Run Extensions at application level" step of the analysis, please contact CAST Support.

Task	Duration	Progress	Log	Results
Application "RPG_CYCLO_TEST"	1m02s			
Run analysis for 'RPG_CYCLO_TEST'	41s			
Run Extensions at application level for RPG_CYCLO_TEST	18s		Yes	
Run Copy Paste metrics calculation for RPG_CYCLO_TEST	1s			
Run Dynamic Link Manager	1s		Yes	
Update SQL XXL Table Size for "RPG_CYCLO_TEST"	1s			skipped : no Table Size data
Run Data Flow Security Analysis on "RPG_CYCLO_TEST"	1s			skipped : User input security...
Prepare snapshot in "rpg12_mardi15_cast81_local"	4s			
Generate Modules in "rpg12_mardi15_cast81_local"	1s			
Run CSV generation	4s		Yes	
Run Data Flow Security Analysis on "rpg12_mardi15_cast81_local"	1s			

What results can you expect?

RPG MetaModel

Below you can find a description of the RPG MetaModel, please click the image to download a full scale PDF version:



Objects

The following objects are displayed in CAST Enlighten:

MENU

Icon	Object Name
1	MENU Bar

2		MENU Item
3		MENU Project

CL400

	Icon	Object Name
1		CL400 Program
2		CL400 Project
3		CL400 Subroutine

DB400

	Icon	Object Name
1		DB400 Project
2		DB400 Table
3		DB400 View
4		DB400 Library
5		DB400 Index
6		DB400 Record

DDS400

	Icon	Object Name
1		DDS400 Project
2		DDS400 Section
3		DDS400 PRTF File
4		DDS400 DSPF File

5		DDS400 Physical File
6		DDS400 Logical File
7		DDS400 RecordstructureLF
8		DDS400 RecordstructurePF
9		DDS400 Joinstructure

RPG 300

	Icon	Object Name
1		RPG300 Project
2		RPG300 Program
3		RPG300 MainSubroutine
4		RPG300 Subroutine
5		RPG300 File Printer
6		RPG300 File Disk
7		RPG300 File Workstn
8		RPG300 File Special
9		RPG300 File Seq
10		RPG300 Rule
11		RPG300 Copy Member
12		RPG300 File Disk Record
13		RPG300 SQL Statement/Structure

14		RPG300 Data Structure
15		RPG300 Procedure
16		RPG300 Local Record Structure

RPG 400

	Icon	Object Name
1		RPG400 Project
2		RPG400 Program
3		RPG400 MainSubroutine
4		RPG400 Subroutine
5		RPG400 Subroutine FreeFormat
6		RPG400 File Printer
7		RPG400 File Disk
8		RPG400 File Workstn
9		RPG400 File Special
10		RPG400 File Seq
11		RPG400 Procedure
12		RPG400 Procedure FreeFormat
13		RPG400 Procedure Prototype
14		RPG400 Bound Service Program
15		RPG400 Copy Member

16		RPG400 Data Structure
17		RPG400 File Disk Record

Structural Rules

The following structural rules are provided:

2.0.10	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.10
2.0.9	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.9
2.0.8	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.8
2.0.7	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.7
2.0.6	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.6
2.0.5	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.5
2.0.4	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.4
2.0.3	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.3
2.0.2	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.2
2.0.1	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.1
2.0.0	https://technologies.castsoftware.com/rules?sec=srs_rpg&ref= 2.0.0

Limitations

False violations reported for the Quality Rule 7388: Avoid artifacts having recursive calls

To cover the dynamic call of programs in CL, for any program name referenced in a variable, a link is created. It also generates a link with the current CL program when the name of the CL program is used in a variable which is used as a parameter in a program call. These types of links are required to reduce the amount of dead code reported by the analyzer and for improved transactions (the recursive call does not affect the transaction graph).

However, these links can cause false violations of the Quality Rule "Avoid artifacts having recursive calls - 7388".



This limitation has been removed in RPG 2.0.3