

Managing ID ranges for custom extensions

- [Introduction](#)
 - [IDs for Objects](#)
 - [IDs for Quality Rules](#)
 - [IDs for Architecture Checker models](#)
- [ID ranges available for use](#)
- [How are ID ranges allocated?](#)
 - [What happens if you have already defined the IDs in your extension?](#)
- [Reserving ID ranges](#)
- [Using the IDs in your extension](#)
 - [file_level and file_no](#)



Summary: this section provides information about the range of IDs that are reserved for the following items for use with custom extensions::

- Extensions
- Objects
- Quality Rules

Introduction

When:

- creating custom extensions that create objects to be saved in the CAST Analysis Service schema or use custom Quality Rules
- or when defining Architecture Checker models

you will need to manage the range of IDs that you assign to these items. For example:

IDs for Objects

For objects, you will have configured items in the **XXXMetaModel.xml** file as follows:

```
<type name="CAST_AngularJS_ModelVariable" id="1020010">
  <description>AngularJS Model variable</description>
  <inheritedCategory name="CAST_AngularJS_Scope" />
  <inheritedCategory name="UAObject" />
  <inheritedCategory name="METRICABLE" />
  <inheritedCategory name="HTML5" />
  <attribute name="AUTO_FOLDER_THRESHOLD" intValue="1"/>
  <tree parent="CAST_HTML5_Enlighten" category="EnlightenTree" />
</type>
<type name="CAST_AngularJS_ModelFunction" id="1020011">
  <description>AngularJS Model function</description>
  <inheritedCategory name="CAST_AngularJS_Scope" />
  <inheritedCategory name="UAObject" />
  <inheritedCategory name="METRICABLE" />
  <inheritedCategory name="HTML5" />
  <attribute name="AUTO_FOLDER_THRESHOLD" intValue="1"/>
  <tree parent="CAST_HTML5_Enlighten" category="EnlightenTree" />
</type>
```

IDs for Quality Rules

For objects, you will have configured items as follows:

```
<?xml version='1.0' encoding='utf-8'?>
<root>
  <metric id='1020500' type='quality-rule' originalName='Avoid the use of the default JavaScript implementation
  [].forEach in AngularJS web app' active='true' detached='false' ignored='false' />
  <metric id='1020502' type='quality-rule' originalName='Avoid the use of JSON.parse and JSON.stringify in
  AngularJS web app' active='true' detached='false' ignored='false' />
  <metric id='1020504' type='quality-rule' originalName='Avoid using the call of web service with AngularJS
  $http inside a loop' active='true' detached='false' ignored='false' />
</root>
```

IDs for Architecture Checker models

Starting from CAST AIP 8.3.2, a unique identifier for your Architecture Model must now be defined in the CAST Architecture Checker GUI for all models assigned to Applications. This ID is used to identify the Architecture Model throughout the CAST Management Studio and the CAST dashboards (the ID will be saved to a table in the CAST Management Service schema).

A positive and even number above **2,000,000** can be assigned as an ID for an Architecture Checker model and you should reserve an ID using the **Quality Range Reservation** option.

ID ranges available for use

The following ID ranges are available for use. As a very general rule you will need to configure IDs above 1,500,000:

ID	Target clients	xxxmetamodel.xml file_level attribute
ID above 2,000,000	Reserved for CAST User Community and CAST Labs extensions	client
ID above 1,500,000 and below 2,000,000	reserved for field consultants (legacy use only)	prepackaged
ID above 1,000,000 and below 1,500,000	reserved for built-in packages created by CAST (official CAST AIP extensions)	builtin
ID above 0 and below 1,000,000	reserved for CAST R&D and products (core CAST AIP range)	core

How are ID ranges allocated?

- ID ranges are automatically allocated on a **first come, first served** basis.
- IDs are allocated in groups of 1000, up to a total of 10,000 per extension.
- IDs can be reserved for Objects or Quality Rules, or you can reserve ranges of both types.

What happens if you have already defined the IDs in your extension?

It is not possible to reserve a specific range of IDs if you have already [created](#) an extension and defined the range of IDs for objects and/or Quality Rules in that extension - as stated above they are allocated on a first come, first served basis. However, if the range you have defined in the extension has not already been allocated ([sign in](#) to the CAST Extend website and click the **ID Ranges** option in the menu bar to view reserved IDs) it is possible to [upload](#) your extension and the ranges you are using will be automatically reserved for you.



Note that if you [upload](#) an extension without pre-reserving the ID ranges you have defined and the ranges have already been allocated to another extension, the extension will be rejected at the end of the upload process. In this situation, you will need to reserve a new range of IDs, modify your extension to use this new range, then upload the extension again.

Reserving ID ranges

Before [uploading](#) your custom extension to share with the wider community, you must ensure that you reserve a range of IDs for use in your extension. To do so:

- [Sign in](#) to the CAST Extend website.
- Click the **ID Ranges** option in the menu bar:



- A new page will be displayed containing two sections:
 - One to reserve **Object ID** ranges
 - One to reserve **Quality Rule ID** ranges (to be used for Architecture Checker models also)

The screenshot shows the top navigation area of the 'ID RANGES' page. It features two tabs: 'Object Range Reservation' and 'Quality Range Reservation'. The 'Object Range Reservation' tab is currently selected and highlighted with a blue underline. A red box highlights both tabs.

- To reserve an ID range click either **Reserve Object ID Range** or **Reserve Rule ID Range**, depending on the IDs you require:

The screenshot shows a table titled 'Object ID Ranges [60]'. In the top right corner, there is a button labeled 'RESERVE OBJECT ID RANGE' with a lock icon, which is highlighted with a red box. The table has three columns: 'Extension ID', 'Start Range', and 'End Range'. It contains two rows of data.

Extension ID	Start Range	End Range
com.castsoftware.uc.vms_basic	-1426385216	-1426384217
com.castsoftware.wbslinker	20000	20999

- The following window will be displayed:

Reserve Object ID range ✕

Enter your extension ID to reserve object ID range.
The range will be automatically generated per specified amount.

Extension ID

Range amount

CANCEL
RESERVE

- Enter the required information as follows:

Extension ID	<p>if you are reserving an ID range for an extension:</p> <ul style="list-style-type: none"> • Enter the ID of your extension in the form com.castsoftware.uc.<name> / com.castsoftware.labs.<name> depending on the type of extension you are uploading. When you upload your extension, you need to ensure that the Extension ID you are using is identical to the Extension ID you enter here. <p>If you are reserving an ID range for an Architecture Checker model:</p> <ul style="list-style-type: none"> • Enter the ID of your extension in the form com.castsoftware.uc.<free-form text>. Where <free-form text> can be any string that describes your initiative or your project.
Range amount	Choose a number between 1 and 10 . 1 corresponds to a range of 1000 IDs, therefore if you enter 3, you will be granted 3000 IDs.

- Click **Reserve** to reserve the selected range of IDs. They will now be shown in the relevant list:

com.castsoftware.labs.test6	2005000	2005999
com.castsoftware.uc.bpel.1...	2006000	2006999
com.castsoftware.uc.test	2007000	2007999

Using the IDs in your extension

Once your ID range has been reserved, you can then use the allocated IDs when [creating](#) your extension. In essence this means assigning them in your `xxMetaModel.xml` file. Take the example above where we have been allocated a range of 1,000 IDs between 2,007,000 and 2,007,999:

file_level and file_no

You need to declare the **file_level** and **file_no** attributes at the start of your [xxxMetaModel.xml file](#) (after the XML declaration):

```
<metaModel file_level="client" file_no="XXX">
```

They should always be set as follows:

- **file_level="client"** > for customers creating **CAST User Community** and **CAST Labs** extensions
- **file_no** > for a range of 1,000 IDs between 2,007,000 and 2,007,999, **file_no** should be set as follows (with **file_level** set to **client** (above 2,000,000)):

```
<metaModel file_level="client" file_no="7">
```