Genesys Interactive Insights User's Guide

Troubleshooting Incompatibility

12/15/2025

Contents

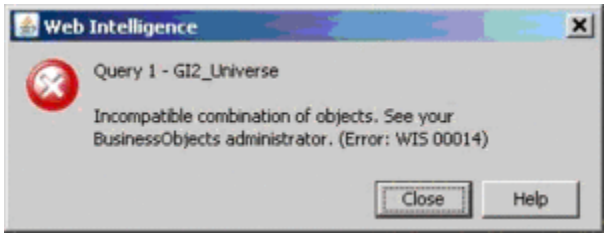
Troubleshooting Incompatibility

It is possible for your custom reports to generate results that are difficult to interpret, to generate errors, or to require excessively long query-processing times when certain combinations of GI2 measures and dimensions are included in the report. These conditions can occur under several circumstances, including:

- Improperly combining incompatible dimensions—such as the `Queue` and `State Name` dimensions—in the same report.
- Improperly combining disposition and interval measures in the same report.

For this reason, Genesys recommends that when you create or customize new reports, you try to employ measures and dimensions that belong to the same class. Additionally, you should select one or more dimensions from the `Time` class in every report. Observing this simple rule will minimize errors and confusion among your report users.

Incompatible Objects

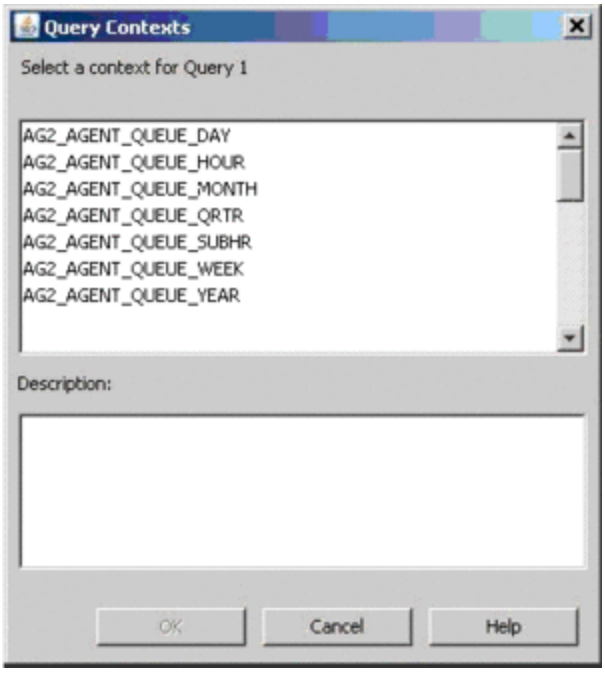


Incompatible Objects Error Message

Incompatibilities can result if you add to your custom reports measures and dimensions from different classes—even if you mix objects from a parent and its subclasses. Web Intelligence displays an error, shown in the figure to the right, when it encounters an incompatibility.

To resolve this problem, you can modify the existing contexts to add joins between tables—where they can be joined—or you can create new contexts defining these relationships. In scenarios where two tables cannot be joined, avoid mixing measures and dimensions within the same report.

Ambiguous Queries



The Query Contexts Dialog Box
(Appears When the Report Query is Ambiguous)

If you build custom reports—even when you select objects from the same class—the **Query Contexts** dialog box can appear, as shown in the figure *The Query Contexts Dialog Box*. Web Intelligence displays this dialog box when the resulting query of your custom report is ambiguous—that is, when the query does not uniquely identify the table from which data should be retrieved. On the contrary, an ambiguous query can be executed against more than one table in the database.

For example, if you fail to include in your report a time-related dimension, then Web Intelligence displays the **Query Contexts** dialog box before the query is executed—because like results are stored in all of the _SUBHR, _HOUR, _DAY, _WEEK, _MONTH, _QRTR, and _YEAR aggregation tables and views. Only after you have specified the appropriate context (the appropriate time dimension in this case) can Web Intelligence display the report's results.

As another example, if you attempt to run a custom report in which you added only the **Queue** and **Queue Group** dimensions from the **Queue** class (from GI2_Universe) to the query definition and nothing else, a message similar to that shown in the *The Query Contexts Dialog Box* figure appears. Queue-related data from this class can be retrieved from any of the following aggregate tables:

- AG2_AGENT_QUEUE_*
- AG2_QUEUE_*
- AG2_QUEUE_ABN_*
- AG2_QUEUE_ACC_AGENT_*
- AG2_QUEUE_GRP_*

Such a query, without any measures or a time-related dimension, provides insufficient information for Web Intelligence to determine the table (or view) from which it should retrieve the desired data.

To supress the message, so that users of your report do not see it, add the appropriate universe elements that satisfy Web Intelligence's request for additional information. Even if you choose not to display these elements in the report, make them part of the underlying query. Optionally, you can preselect the appropriate context, so that the message will not be displayed to users. Refer to BO/BI documentation for information on how to create contexts.

Finally, as is the case with any report design, study the results of your generated custom report to ensure that they make sense.

GI2 releases that are incompatible with BI 4.2

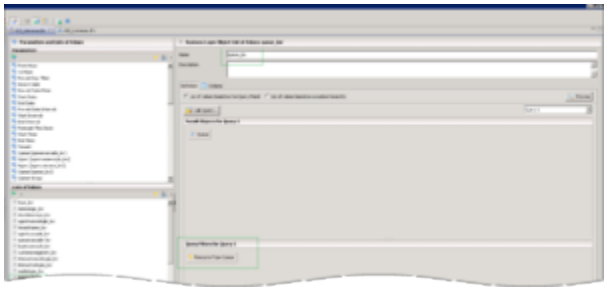
In scenarios where you have installed SAP BI 4.2 with unsupported GI2 releases (GI2 8.5.000), drilling up from Agent to Agent Group in customized reports can fail, producing an error message such as:

Modify joins for these Queue dimensions

Modify joins for these Queue Group dimensions

Modify joins for these Agent Name dimensions

Modify joins for these Agent Group dimensions



Modify LOVs

```
The following database error occured: The multi-part identifier 'GROUP_GROUP_TYPE_CODE' could
not be bound...
```

.

In these scenarios, Genesys recommends that you upgrade to GI2 release 8.5.001.02 or later, which interoperates correctly with BI 4.2. Alternatively, you can customize GI2 to work correctly with BI 4.2 by using the instructions in the following procedure.

> ## Important
>
> Genesys recommends that you install GI2 with the recommended releases of supporting software, including RAA, Gensys Info Mart, and BI 4.x software. Compatible releases are listed in the *8.5 Product Alerts*, available on the Genesys Interactive Insights page.

Editing the universe so that older versions of GI2 can work with BI 4.2

To resolve problems with drill operations (Agent up to Agent Group) in scenarios where you have installed BI 4.2 with releases of GI2 that do not support it, make the following changes to the universe:

1. In the class Service Objects, add the following new conditons:
   - Resource Type Agent: `RESOURCE_GI2.RESOURCE_TYPE_CODE = 'AGENT' and RESOURCE_GI2.RESOURCE_TYPE = 'Agent'`
   - Resource Type Queue: `RESOURCE_Q.RESOURCE_TYPE_CODE = 'QUEUE'`
   - Resource Type Queue and None: `RESOURCE_Q.RESOURCE_TYPE_CODE in ('QUEUE','NONE', 'UNKNOWN')`
   - Group Type Queue: `GROUP_Q.GROUP_TYPE_CODE in ( 'QUEUE','UNKNOWN','NO_VALUE')`
   - Group Type Agent: `GROUP_.GROUP_TYPE_CODE in ('AGENT', 'UNKNOWN', 'NO_VALUE')`
2. For each of the following dimensions, in all classes except the details classes, remove the **Where** clause:
   - Agent Name
   - Agent Group
   - Queue
   - Queue Group
3. Modify the following joins to support deleted **Where** expressions:
   - For Agent Name dimensions, modify the joins for the dimensions shown in the figure **Modify joins for these Agent Name dimensions**.
   - For Agent Group dimensions, modify the joins for the dimensions shown in the figure **Modify joins for these Agent Group dimensions**.
   - For Queue dimensions, modify the joins for the dimensions shown in the figure **Modify joins for these Queue dimensions**.
   - For Queue Group dimensions, modify the joins for the dimensions shown in the figure **Modify joins for these Queue Group dimensions**.
4. Modify the definitions of the following LOVs, as shown in the figure **Modify LOVs**, to reflect the corresponding condition from the Service Objects class: queue_lov, queuegroup_lov, queueandnone_lov, agentname_lov, agentgroup_lov