
Article

[Stefan Cronje](#) · Jan 25, 2023 3m read

[Open Exchange](#)

InterSystems IRIS Persistent Class Audit Package

Hi folks,

I am announcing a new package I have loaded on the OEX, which I am also planning on entering into the contest this month.

In a nutshell, what it offers you are the following.

- Base classes to use on Persistent (table) classes for InterSystems IRIS to keep record history
- These classes enable the historizing of persistent class records into another persistent class when touched.
- This provides for a full history of any record.
- It allows for record rollback to a specific version.
- It can automatically purge old history records.

Do you need it?

Have you ever had the scenario where a data fix has gone terribly wrong, and you need to roll back the update?

Have you ever tried to find out what or who updated or inserted a specific record?

Have you ever needed to find out what has happened to a record over its lifetime?

You can get all this now, by simply extending from two classes.

What this article covers is what it offers. The package contains all the instructions needed, and there are only a few.

The Basics

The table that contains the "current" record have two sets of fields

- Create
 - This contains the details of when the entry was created and is immutable.
- Update
 - This contains the information of the last time the record was updated.

The table that contains the history records have three sets of fields

- Create
 - This is copied as is from the current record when inserted.
- Update
 - This is copied as is from the current record when inserted.
- Historize
 - This contains details on the historization entry on insertion, and is immutable.

Each of the sets above contain the following fields:

Name	Content
DateTimeStamp	The system's date and time
Job	The PID value (\$JOB)
SystemUser	The system user performing the action. \$USERNAME
BusinessHost	The interoperability business host that was involved, if available.
ClientIP	The client's IP address that instructed this, if available.
CSPSessionID	The CSP Session ID that was involved, if available.
Routine	The calling routing. This can really help pinpoint where in the code this originated from.

The "current" record table has a Version property, and this is set when extending from the base class.

The history racoed table has a Version property, which is the version that the "current" record was at that point.

What can it historize?

- Serial Properties
- "Normal" Properties
- Class References
- Arrays of Data types
- Lists of Data types
- Arrays of Objects
- Lists of Objects
- Relationship - where cardinality is "one"
 - Note that in this case, the history table's prpperty must be an object reference and not a relationship.

What can't it historize?

- Relationships where the cardinality is "many"

What about my other triggers?

The triggers activate in order slot number 10. This means you can add your triggers before or after the execution of these triggers.

Additional Functionality

- Record historization can be switched off for a specific table by setting a global. This is to cater for bulk updates where is the history is not required, for example, populating a new field during a deployment.
- An auto purge setting can be set per table in a global. If you have a table that does get a lot of updates, and you only need the previous two records for example, you can set it to keep the last two records in hte

history and remove the older ones. This happens during the historization of the record.

Restoring Previous Versions

The base classes will generate methods in the "current" record class that can be used to restore a specific version, or the previous version, back into the current table.

These can be invoked via SQL too, if you need to restore from a selection of rows.

Note that the restored record will bump the version number on the "current" table and will not have its old version number, which is probably a good thing.

[#ObjectScript](#) [#InterSystems IRIS](#) [#Open Exchange](#)

[Check the related application on InterSystems Open Exchange](#)

Source URL: <https://community.intersystems.com/post/intersystems-iris-persistent-class-audit-package>