Article
Andreas Schneider · Jan 2, 2022 3m read

Open Exchange

Tips and tricks of the brand new LOAD DATA command

The last days I've work with the great new feature: LOAD DATA With this post I would like to share my first experiences with you. The following points do not contain any order or other evaluation. These are only things that I noticed when using the **LOAD DATA** command. It should also be noted that these points are based on the IRIS Version 2021.2.0.617 which is a preview release. So it may be that my observations do not apply to newer IRIS versions. But maybe they are helpful for others.

1) The file path is on server side

I have done my first tests via JDBC. The first trap I stumbled into: The file and the filepath must-, *of course ;-)* be on serverside! The JDBC driver handle this **not** on client side. Probably this is obvious, but I had not considered this at first.

2) The file suffix is not relevant

The docs says:

> "*The file name must include a .txt or .csv (comma-separated values) suffix.*"

According to my observation the behavior is not so. The suffix is irrelevant.

3) Read the docs! ... or where are the error rows?

While loading some datafiles, i missed rows. If there is a problem with a line, this line is ignored. This happens silently in the background and the client is not actively notified. After viewing https://https//youtu.be/jm7bDK0FoiI I've noticed that I have to check %SQL_Diag.Result and %SQL_Diag.Message to see the problems in details. I also noticed that this behavior is already described on this page: https://docs.intersystems.com/iris20212/csp/docbook/DocBook.UI.Page.cls?KEY=RSQL_loaddata ... so RTFM ;-)

Some examples about what you can see:

```
SELECT * FROM %SQL_Diag.Result ORDER BY createTime DESC
```

Check the errorCount column of your loading.

You can see (row) details in %SQL_Diag.Message

```
SELECT * FROM %SQL_Diag.Message ORDER BY messageTime DESC
```

You can filter for a specific *diagResult* (%SQL_Diag.Result.ID = %SQL_Diag.Message.diagResult)

```
SELECT * FROM %SQL_Diag.Message
WHERE diagResult=4
ORDER BY messageTime DESC
```

4) LOAD DATA is not supported by $SYSTEM.SQL.Schema.ImportDDL

For my sample app Openflights Dataset I've tried to load all external files with LOAD DATA. The Statements are bundled within a text (sql) file where I also create the tables before.

I've learn you can't do that via $SYSTEM.SQL.Schema.ImportDDL.

By the way the documentation of ImportDDL says that not all SQL Statements are supported. Only a few sql statements are listed on this page.
LOAD DATA unfortunately does not belong to... by the way USE DATABASE unfortunately also not.

5) For unicode handling you have change a setting

To avoid problems with data encoding during loading, please make the following setting on the %Java Server:
-Dfile.encoding=UTF-8
See more details in this post. This problem should be gone in a next IRIS release.

6) Loading stops with error, but data are loaded

Loading data via JDBC stops with an %qparsets error. That looks like this:

```
 Error: [SQLCODE: <-400>:<Fatal error occurred>]
[Error: <<UNDEFINED>zExecute+83^%sqlcq.OPENFLIGHTS.cls10.1 *%qparsets>]
[Location: <ServerLoop>]
```

But don't worry the data was loaded anyway :-) See more details in this post.
This problem should be gone in a next IRIS release.

Andreas

#Best Practices #Contest #CSV #JDBC #SQL #Tips & Tricks #InterSystems IRIS
Check the related application on InterSystems Open Exchange

Source URL:https://community.intersystems.com/post/tips-and-tricks-brand-new-load-data-command