Sage Evolution

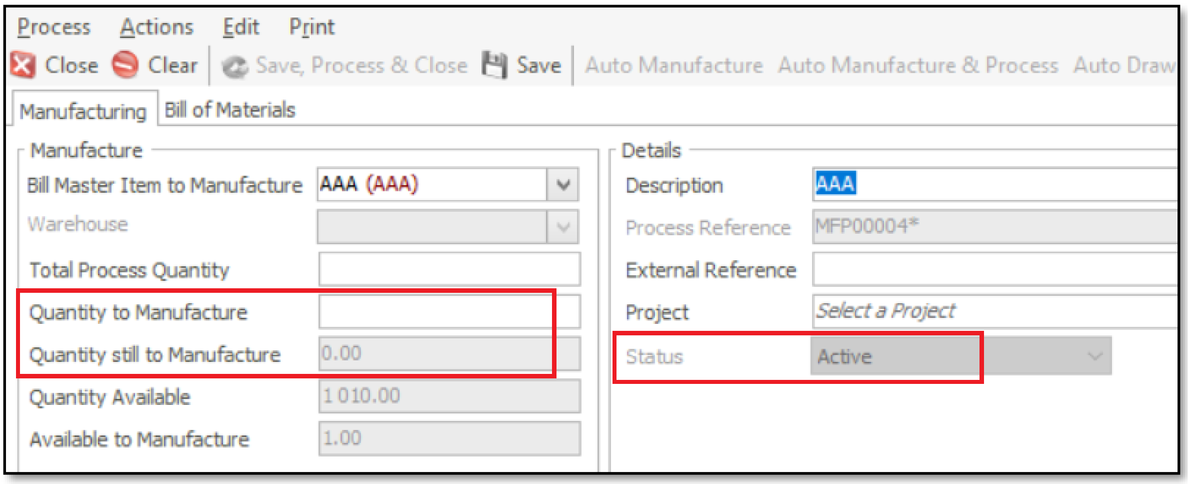
Knowledgebase Article

Article Outline	Manufacturing process doesn't have Completed status after all units have been manufactured, shows Active status instead.
Article Date	2020/05/11
Knowledge Type	Solving an unknown issue
Knowledge Activity	Processing
Primary Module	Manufacturing/BOM
Secondary Module	Manufacturing/BOM
Knowledge Source	Incoming Customer Query
Incident Reference Nr	HD12617825

Description of the Issue

You may observe that a Manufacturing (MF) process does **not** have a status of **Completed** after **all** units to be manufactured, have indeed been manufactured.

Instead the MF process may still have a status of **Active** as can be seen below.



Bill Master Item to Manufacture: AAA (AAA)
Warehouse:
Total Process Quantity:
Quantity to Manufacture:
Quantity still to Manufacture: 0.00
Quantity Available: 1 010.00
Available to Manufacture: 1.00

Description: AAA
Process Reference: MFP00004*
External Reference:
Project: Select a Project
Status: Active

Solution / Details

Please first note:

It's recommended that this query should preferably be fixed by a professional Evolution support consultant as it may include complex, database analysis skills and understanding how to compile SQL queries, to correct the above problem.

Therefore, please contact either your Evolution business partner (BP) or Evolution Support to get this issue resolved.

This issue may be due to having improperly defined foreign objects such as custom triggers, -stored procedures, -tables and/or -views that are part of the company database.

In principle, there is nothing wrong with having foreign objects to perform certain, extended functions and/or to assist in integrating the company database with any kind of external application.

However, when these kinds of foreign objects are not properly compiled/setup/configured they may then conflict with Evolution's standard objects and features/functions. As a result, the above issue (as an example) may then be observed.

To inspect the existence of any foreign objects on your company database, do the following:

Phase 1: Create Demo company

Create a new demo company called e.g. DEMO in the **same** Evolution version as your live company database.

Phase 2: Run SQL scripts to identify foreign objects

2.1 Run the following SQL scripts on the **live** company database (DB).

a) Check for Foreign Tables

```
SELECT * FROM   SYSOBJECTS
WHERE xtype = 'U';
GO
```

b) Check for Foreign Triggers

```
SELECT name, is_instead_of_trigger
FROM sys.triggers
WHERE type = 'TR';
```

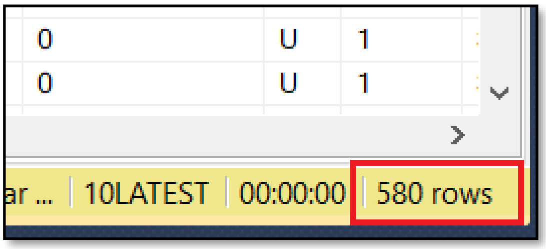
c) Check for Foreign Stored Procedures

```
SELECT name, create_date, modify_date
FROM sys.procedures
```

d) Check for Foreign VIEWS

```
SELECT OBJECT_SCHEMA_NAME(v.object_id) schema_name, v.name
FROM sys.views as v
```

In each case above, check the number of records found at the bottom of the results pane



```
0        U    1
0        U    1
ar ... | 10LATEST | 00:00:00 | 580 rows
```

2.2 Run the same SQL scripts above on the **DEMO** company database (DB).

In each case above, check the number of records found at the bottom of the results panel.

Phase 3: Comparison between Live and DEMO company

1. If there are any differences in the number of records between the Live and Demo company DB's it means foreign objects may have been added on the live company DB.

   Foreign objects are typically added to add additional, custom added features in the Evolution company and/or to integrate the Evolution company with an external application.

2. In this case (having a difference in the number of tables), consult with the person/consultant who added the additional tables on the Live company DB, on point 3 below.

3. As a proposed methodology going forward:

   1. Backup the live company DB.
   2. Restore the backup as a new DB called e.g. **LIVE-TEST**
   3. Identify and drop all **FOREIGN** objects on the **LIVE-TEST** DB
   4. Capture and process a dummy Manufacturing process in the LIVE-TEST DB in the usual way as done in the actual Live DB.
   5. In here you can now test if the newly completed Manufacture Process has a **Completed** Status or not.
   6. If the Status = **Completed**: That means that one or more of the foreign objects that were dropped may be responsible for the above Manufacturing query: Manufacturing processes **not having** a Completed status.
   7. This means that the foreign object developer should identify and correct the relevant foreign object/s in the **Live** company DB that may be causing the Manufacturing query in the first place.

Disclaimer: *These articles refer to possible solutions and a platform to share information. Each article describes a method that solved a query (knowledge gathered from previous sites) and how Sage Evolution should operate. These articles make reference to a specific Sage Evolution version, however the thought process can be generalised. Please note the information contained in these articles should be treated as guidelines and adapted to accommodate differences in business processes and IT environments. Articles may not be applicable to all environments. If this article did not resolve your query please contact the Sage Evolution Support Department directly on: +27 (0) 86 112 6837.*