SIEMENS

Factory 2020

Release Notes

01-022495-20200-E

Proprietary and restricted rights notice; Trademarks

Contents

Chapter 1: Common program information for Factory release 2020

Supported platforms

Microsoft Windows operating systems 64 bit versions for Factory 2020

- Windows 10 (64-bit only) (version 1803 or higher)
- Windows 8.1 with Update KB2919355 (64-bit only)
- Windows 7 SP1 Update KB4019990 (64-bit only)

> **Note**
>
> Release 2020 does not support 32-bit operating systems.

Autodesk packages, as supported by Autodesk on the underlying operating system

- AutoCAD 2019 and 2020
- AutoCAD Architecture 2019 and 2020
- AutoCAD MEP 2019 and 2020

If you run the software on an unsupported platform and encounter a problem, and the problem can be duplicated on a supported platform, we will work to fix the problem on the supported platform. A fix developed for a supported platform might resolve a problem on an unsupported platform. However, we may not work on resolving problems that do not occur on supported platforms.

Software licensing

The Siemens PLM Software applications FactoryCAD and FactoryFLOW require a license file for operation. Siemens PLM Software provides a license file when the software is purchased.

To enable sharing of a license among multiple machines, a license can be served over a network by Siemens PLM license server software. Siemens PLM license server software for Microsoft Windows platforms is included on the product DVD or available for download from the Customer Support (GTAC) Web site: http://www.siemens.com/gtac.

Online User Guide (Help)

Online User Guide (Help) and tutorial files are provided in Microsoft compressed HTML (CHM) help format. The CHM files are installed with the application software.

If the application software is installed somewhere other than the local file system, the CHM files may not be viewable. Microsoft security measures usually disable the viewing of CHM files over a network. Workarounds to enable full CHM functionality over a network are described in the Microsoft knowledge base article posted at http://support.microsoft.com/kb/896358/.

Share drawings containing Smart Factory Objects

AutoCAD drawings created with FactoryCAD or FactoryFLOW programs may include three-dimensional models of factory equipment called Smart Factory Objects.

The term Smart Factory Objects encompasses objects developed for FactoryCAD and FactoryFLOW using the following means:

- Autodesk ARX API
- FactoryCAD XML Object Toolkit
- the FactoryCAD generic tool object, which can incorporate multiple blocks for 3D and orthogonal views, including blocks generated from JT files
- blocks registered in a FactoryCAD library

Here are three methods, in order of preference, for viewing AutoCAD drawings containing Smart Factory Objects on machines where neither FactoryCAD nor FactoryFLOW are installed.

Method 1: Install free FactoryCAD Object Enablers (preferred method)

The FactoryCAD Object Enablers software enables display in AutoCAD of Smart Factory Objects, even when neither FactoryCAD nor FactoryFLOW is installed. The enablers software is free, does not require a license file, and does not require loading a Factory menu or starting a Factory program.

For instructions on installing the enablers, see **Install Guide Enablers<release>.pdf** on the Factory installation disk or software download site.

With the free FactoryCAD Object Enabler, drawings containing Smart Factory Objects can be opened on machines with AutoCAD but not FactoryCAD and show full Smart Factory Objects in plan and isometric views. In the absence of FactoryCAD, the Smart Factory Objects cannot be edited, but regular AutoCAD functionality such as copy, erase, move, osnaps, layers, wblock, and so on can be applied.

Method 2: Save drawings with detailed proxy graphics (does not require installing enablers)

This method creates a file that includes not only the object information, but also a detailed proxy graphic so that object geometry is available on machines that do not have Factory software, but do have software that understands AutoCAD proxy graphics. Proxy graphics increase the size of the drawing depending on the amount of Smart Factory Objects contained in the drawing and depending on whether the proxy graphics are 2D or 3D. This method is especially useful for sending out drawings that contain graphical information for additions by outside drafters.

Once a drawing with additions is returned and opened in FactoryCAD, the proxy graphics are replaced with the original Smart Factory Object geometry.

1. Start FactoryCAD.
2. Click **FCAD Tools** tab→**Import/Export** panel→**Save With Detailed Proxy Graphics ON**.

   The **Save With Detailed Proxy Graphics** setting lasts for the current AutoCAD session only. Each time you start AutoCAD, the setting value returns to the default value, **No**.
3. Set the current view according to whether you want to generate 2D or 3D proxy graphics.

To generate this kind of proxy graphics	Set the current view to
2D	Plan view
3D	An isometric view

4. Make some change to the drawing so that the drawing database has been changed to the current view.

5. Save the drawing.

Method 3: Explode Smart Factory Objects before distributing the drawing (not recommended)

You can explode Smart Factory Objects to replace them with AutoCAD primitives that can be viewed, snapped to, and manipulated in basic AutoCAD. These drawings will be larger than those with proxy graphics and do not contain object intelligence. This method does not allow for retention of Smart Factory Objects information, and thus the ability to return to smaller file size.

1. Create a copy of the drawing that contains the Smart Factory Objects.

2. Open the copy in AutoCAD and zoom to a view that contains the Smart Factory Objects you want to explode.

3. Start FactoryCAD.

4. Select the objects to explode (objects other than Smart Factory Objects will be filtered out) and then choose one of the explode commands:

   - Click **FCAD Tools** tab→**Import/Export** panel→**Explode all FactoryCAD Objects to 2D**.

   - Click **FCAD Tools** tab→**Import/Export** panel→**Explode all FactoryCAD Objects to 3D**.

5. Save the drawing.

Chapter 2: Changes from Factory 2019 to 2020

Common changes

- Added support for AutoCAD 2020, AutoCAD MEP 2020, and AutoCAD Architecture 2020.
- Dropped support for AutoCAD 2018, AutoCAD MEP 2018, and AutoCAD Architecture 2018.

JT Enhancements

- JT File Version Support.
  - Read up to JT 10.3 format.
  - Export up to JT 10.2 format.
  - Default value is set to JT 10.1 to ensure compatibility with other applications.
- Performance
  - Improved performance on JT File Import when using Decimation.

Language support

- FACTORY command is now available for any language.
- Command automatically load menus/ribbons and set trusted paths. Previously this was installed in a language specific location.
- FACTORY command can now be run for AutoCAD installed in any language.

New settings in FactoryCAD configuration

- Block Library Path:

  The Block Library Path now can be specified in the **FactoryConfig.xml** file. Add the following setting:

  <Setting name="Block Library Path" value=""/> (where value is the location of the CIMFC_FI file containing the locations of the Block Libraries).

  An empty value will use the default path **C:\ProgramData\Siemens PLM Software\Factory Programs [release number]\Support**

- Global Parameter Set Path:

  The Global Parameter Set Path can now be specified in the **FactoryConfig.xml** file.

  - Supports a common location that can be shared with multiple users.
  - Manages **.pst** and **GlobalSets<>.xml** at enterprise level.

Add the following setting:

<Setting name="Global Parameter Sets Path" value=""/> (where value is the location of the pst files and the Metric and Imperial folders containing the **GlobalSets<>.xml**)

An empty value will use the default path **C:\ProgramData\Siemens PLM Software\Factory Programs [release number]\Support**

Support for the latest Python version Python 3.7.1

Changes for FactoryFLOW

Feature area	Change
Reports	Since Factory V2019, all standard reports are available in Excel format, legacy Crystal Report format or CSV format. The new Excel format uses pivot tables which supports a tree structure allowing users to expand and collapse root nodes and modify data after the report is generated. Option to generate Excel or Crystal Report format is available in the Settings under the General Settings tab. Customized Excel reports can be added through Reports–>Customize Menu.

Chapter 3: Changes from Factory 2018 to 2019

Common changes

- Added support for AutoCAD 2019, AutoCAD MEP 2019, and AutoCAD Architecture 2019
- Dropped support for AutoCAD 2017, AutoCAD MEP 2017, and AutoCAD Architecture 2017.

Changes for FactoryCAD

Feature area	Change
Units	Object Creation Units and Dialog Display Units are now saved with the DWG file instead of in their previous locations in the registry. These units are now drawing specific allowing users to easily work with different units in the same session. For new files and legacy files, settings have been added to the FactoryConfig.xml file that define the default behavior.
JT Import	A status indicator has been added for JT import (block and XML). The progress meter is displayed in the AutoCAD status bar during the import process. 
Report Toolkit	XML Object Elevation has been added to the list of 'Available Object Controls' in the Report Toolkit.
Robots	Several new robots have been included with this release. KR150 R3100 Prime KR180 R2900 Prime KR210 R2700 Prime KR210 R3100 Ultra KR240 R2900 Ultra KR270 R2700 Ultra KR300 R2500 Ultra

Changes for FactoryFLOW

Feature area	Change
Find in Drawing	The menu item 'Find in Drawing' has been added to the Activity Point context sensitive menu (RMB).
Model Space	Find in Drawing now automatically switches to model space if a paper space tab is active.

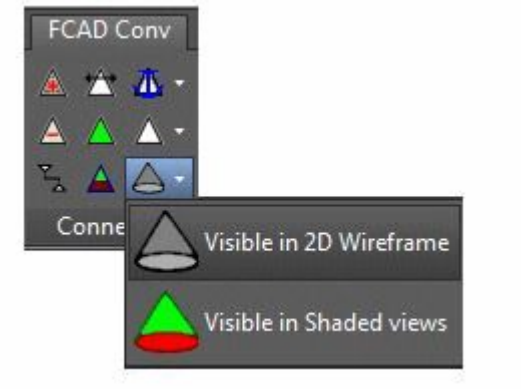
Feature area	Change
Unit Validation	FactoryFLOW requires that the Drawing Units and Object Creation Units are set to the same values. If the units are mismatched, a message will be displayed notifying the user that the units need to be properly set.
Reports	All standard reports are now available in Excel format, legacy Crystal Report format or CSV format. The new Excel format uses pivot tables which supports a tree structure allowing users to expand and collapse root nodes and modify data after the report is generated. Option to generate Excel or Crystal Report format is available in the Settings under the General Settings tab. Customized Excel reports can be added through Reports>Customize Menu.

Chapter 4: Changes from Factory 2017 to 2018

Common changes

- Added support for AutoCAD 2018, AutoCAD MEP 2018, and AutoCAD Architecture 2018
- Dropped support for AutoCAD 2016, AutoCAD MEP 2016, and AutoCAD Architecture 2016.

Changes to FactoryCAD

Feature area	Change
Connectors	Added capability to display connectors using visual styles other than 2D Wireframe. A pulldown button has been added in the ribbon panel that is used to control the visibility of connectors in Shaded Views. Connectors will continue to be visible in 2D Wireframe. 

Changes for FactoryFLOW

Feature area	Change
Reports	Complete Flow Results are now available in Excel format. User should pick Reports>Standard Reports>Flow Results>Complete Flow Results>Complete Results Excel. This functionality has also been extended to allow users to create customized Excel reports. These can be added through Reports>Customize Menu.

Chapter 5: Changes from Factory 2016 to 2017

Common changes

- Added support for AutoCAD 2017, AutoCAD MEP 2017, and AutoCAD Architecture 2017
- Dropped support for AutoCAD 2015, AutoCAD MEP 2015, and AutoCAD Architecture 2015.
- Officially certified Office 2013 and 2016.
- There is now a link to the Tecnomatix Community Support site added to the Help menu.

  Click **Factory**→ **Help** group→ **Factory Layout Software Help**→ **Tecnomatix Community**

Changes for FactoryCAD

Feature area	Change
File load performance	Enhanced menu loading to improve performance and prevent unloading and reloading of menus when switching between files. Once the Factory menus are loaded in the session they will remain loaded until AutoCAD is closed.
Tutorials	Added link to the FactoryCAD Tutorial from the Help menu. To access the Tutorials, do the following: Click **Factory** tab→**Help** group→**Factory Layout Software**→**Help**→**FactoryCAD Tutorials**.
Backward compatibility	Factory drawing version has been added to the drawing dictionary. As long as the Factory drawing version is not updated between releases, drawings created in later versions of the product can now automatically be opened without requiring a patch. The drawing version can be found in the Factory settings. The drawing version can also be displayed using the **CIMFDWGVERSION** command.
Direct Model (JT) export	Translation of AutoCAD mtext is now supported for all font types.
Block manager	All block libraries included with FactoryCAD were migrated to the Library Manager and Factory Explorer. These objects can now be inserted directly from the Factory Explorer. Documentation to **Import items to a library** was updated to include creation of bitmaps to support display of preview images in the Library Object view panel in the Factory Explorer. For this release both the Block Manager and Library Manager functionality will be supported.

Feature area	Change
Enablers	Enhanced enabler setup to prevent license errors if FactoryCAD was previously installed on the machine. A new key was added in HKEY_LOCAL_MACHINE\SOFTWARE\Siemens PLM Software\Factory Programs\[version] **FactoryEnabler→DWORD→1**

Changes for FactoryFLOW

Feature area	Change
File load performance	Enhanced menu loading to improve performance and prevent unloading and reloading of menus when switching between files. Once the Factory menus are loaded in the session they will remain loaded until AutoCAD is closed.
Tutorials	Added link to the FactoryCAD Tutorial from the Help menu. To access the Tutorials, do the following: Click **Factory** tab→**Help** group→**Factory Layout Software→Help→FactoryFLOW Tutorials**. In the **FactoryFLOW** window, choose **Help→FactoryFLOW** Tutorials.
Component Tree	Find function enhanced to search for parts, assemblies, activity points, and material handling equipment in the FactoryFLOW window data tree. The assembly tree will expand and scroll as needed to show the selected items.
Part Routing	Enhanced Part Routings to automatically display in sequential order and added the capability to sort the moves by clicking on the column headers. Any time the Part Routing is loaded/reloaded, it will initially appear in sequential order..
JRE Version	Java SE 7 and higher versions of JRE are now supported for conversion of legacy **.flo** project files.

Chapter 6: Changes from Factory 2015 to 2016

Common changes

- Added support for AutoCAD 2016, AutoCAD MEP 2016, and AutoCAD Architecture 2016.
- Dropped support for AutoCAD 2014, AutoCAD MEP 2014, and AutoCAD Architecture 2014.
- Enhanced uninstall routine to remove common files when both FactoryCAD/FactoryFLOW and In Context Editor are uninstalled.
- Moved product version and other information to the Product Details dialog box, and added other information. To display the dialog box, do the following:

  Click **Factory** tab→**Help** group→**Factory Layout Software About** and then in the **About Factory Layout Software** dialog box click **More Information**.

Changes for FactoryCAD

<table>
<tr><th>Feature area</th><th>Change</th></tr>
<tr><td>SDX Layout Parameters</td><td>• Added capability to purge unused, or all, SDX layout parameters from a drawing.

• Modified WBLOCK command behavior so that layout parameters that are used for objects in the selection set are copied to the new drawing.</td></tr>
<tr><td>Toolbars</td><td>FactoryCAD toolbars are all hidden when the software is installed. To turn on display of individual toolbars, start FactoryCAD and then click View tab→Windows group→Toolbars→CIMFCAD and then choose the name of the toolbar that you want to display..</td></tr>
<tr><td>File load performance</td><td>Added menu cleaning to avoid proliferation of menu references that was causing a slowdown in file loading performance.</td></tr>
<tr><td>Support for shared location of Factory Settings and LibraryManager.xml</td><td>Added support for using Factory settings files stored on a network.

Added support for alternative location of the LibraryManager.xml file, including on a network. Location is specified by a setting in the FactoryConfig.xml file.</td></tr>
<tr><td>Direct Model (JT) export versions</td><td>Added support for exporting to JT versions 9.0, 9.5, and 10.1.</td></tr>
<tr><td>Create XML objects from .prt or Parasolid files</td><td>Added support on 64-bit platform for conversion of the latest version of .prt and Solid Edge Parasolid files.
<table>
<tr><th rowspan="2">Installation platform</th><th colspan="2">Supported versions</th></tr>
<tr><th>NX files (.prt)</th><th>Solid Edge files (.par, .psm, .pwd, .asm)</th></tr>
<tr><td>32-bit</td><td>through NX 9</td><td>through ST6</td></tr>
<tr><td>64-bit</td><td>through NX 10</td><td>through ST7</td></tr>
</table>
</td></tr>
</table>

Changes for FactoryFLOW

Feature area	Change
Excel import performance	Improved performance when importing routing data from Excel, especially from large data sets. Added check and warning when old format Excel files contain over 65,000 rows.

Chapter 7: Changes from Factory 2014 to 2015

Common changes

- Added support for AutoCAD 2015, AutoCAD MEP 2015, and AutoCAD Architecture 2015.
- Dropped support for AutoCAD 2013, AutoCAD MEP 2013, and AutoCAD Architecture 2013.

Changes for FactoryCAD

Feature area	Change
Custom object dialog toolkit	Enhancements to the dialog toolkit enable users to more easily position controls. • Controls that are a member of a group now move with the group, so adjusting the position of a group of controls is much quicker. • Identifying controls as members of a group (or not) is certain because the group name is automatically added to the control name. • Multi-tab design is less prone to error because the active tab is constantly identified. • New **Reference Objects** field in custom dialog allows editing of properties of objects incorporated in an XML object by reference.
Factory Explorer	Added new Library Object view panel. The panel displays object preview images or generic icons for items contained in the folder level selected in the tree. When a single item is highlighted in the tree or view panel, a list of its registered item properties is displayed in a portion of the preview panel. Object preview image files can be specified in the registered item properties for each item. Images must be in .bmp format.
XML Object Layers	Added utility to reset layers of selected XML object according to the current layer standards file. **FCAD Tools** tab→**Managers** group→**Drawing Conversion Utilities→Reset Layers**. The new utility also is available in the custom object API, command name **CIMFRESETFACTORYLAYER**.

Changes for FactoryFLOW

Feature area	Change
Reports	Added a new standard report of flow and utilization results. **Reports→Standard Reports→Flow and Utilization Resuts→.CSV Text File**

Notes:

Chapter 8: Changes from Factory 2013 to 2014

Common changes

- Added support for AutoCAD 2014, AutoCAD MEP 2014, and AutoCAD Architecture 2014.
- Dropped support for AutoCAD 2012, AutoCAD MEP 2012, and AutoCAD Architecture 2012.

Changes for FactoryCAD

Feature area	Change
Mirroring objects	• Added new capability to set the **Allow Mirroring** parameter when creating an XML object by converting geometry via the following commands in the **FactoryCAD** tab→**XML Objects** group→**Create from** drop-down: o **Create from JT**  o **Create from Parasolid** 
Custom object API	**CimfSetControl** – Enhanced function to recalculate all control equations in selected object instances, rather than only the specified control equations.
Library Manager	• Enhanced library export function. All files referenced by registered items are now exported and a relative path is used for files such as animation files, dialog pictures, and help files. If referenced files are not under a library folder, those files are exported to a new **Relocated** folder and references to them are updated in the exported library. The original library is not changed. • Added button to switch Library Manager between two dialog box sizes.
Object preview independent window	Added switch preview mode 2D/3D capability.
Create XML objects from JT models	Added option to create **Allow Mirroring** property and set it to true during object creation.

Changes for FactoryFLOW

Feature area	Change
Default Attributes	Added capability to define default attributes for activity points and material handling devices.
Operator Walk Path	Added report for Operator Walk Times.

Notes:

Chapter 9: Changes from Factory 16 to 2013

Common changes

- Added support for AutoCAD 2013, AutoCAD MEP 2013, and AutoCAD Architecture 2013.
- Dropped support for AutoCAD 2011, AutoCAD MEP 2011, and AutoCAD Architecture 2011.
- Added new tabs for FactoryCAD and FactoryFLOW commands to the AutoCAD ribbon. All of the commands available in toolbars and menus are now available in the ribbon.
- Moved to a new version of Siemens PLM licensing. If the software license is acquired at run time from a license server, rather than from a local license file, the server must be running Siemens PLM License Sever version 5.3.1 or newer.

Changes for FactoryCAD

Feature area	Change
Create XML objects from Parasolid models	Added new capability to add an **Allow Mirroring** parameter when converting Parasolid PRT models to XML objects.
Import JT to AutoCAD	• Added new **Visibility Sphere** simplification option. • Added documentation describing differences between EAIIN (block creation) and JT2XML (XML object creation) when translating JT files. • Improved EAIIN performance. • Updated documentation explaining the Decimate and Level of Detail (LOD) parameters.

Changes for Smart Factory Object Enablers

- Revised installation to install enablers for all machine users without manual registry modification.

Changes for FactoryFLOW

Feature area	Change
FactoryFLOW editor flow diagrams and assembly objects	Added capability to import/export and modify generic attributes of type string on assembly (product/assembly/part) and move arrow objects.

Feature area	Change
FactoryFLOW database engine	• Changed the project database engine to SQLite. This change applies to all installations of FactoryFLOW release 2013. Using SQLite for the database engine allows operation of FactoryFLOW on 32-bit and 64-bit operating systems regardless of the presence of any Microsoft Office version. • Migrated the project file format to a format compatible with the database engine SQLite. The new project filename extension is **.qflo**. FactoryFLOW can convert **.flo** project files from earlier releases to the new format. The conversion requires the presence of the Java SE 7 runtime extension. You can download Java SE 7 JRE from Oracle at http://www.oracle.com/technetwork/java/javase/downloads Multiple versions of Java can coexist on a machine.
FactoryFLOW project file	• Added option to compact and repair the FactoryFLOW project database upon closing a FactoryFLOW project. The option settings appear in the **FactoryFLOW Settings** dialog box on the **General Settings** tab.

Chapter 10: Converting Access code to SQLite

The following sections provide a reference for issues related to converting legacy Microsoft Access code related to FactoryFLOW for use with release 2013 and later versions.

Update query using table shortcut issue

Original query	update TEMPLATE_TABLE T1, TIME_TABLE T2 set T1.Unit_Act_Time = (T2.Const_0 + T1.Var_1 * T2.Const_1) * T1.Multiplier where T1.Act_Name = T2.Act_Name AND T1.Var_1 IS NOT NULL AND T1.Var_2 IS NULL
Query initially adapted for SQLite	update TEMPLATE_TABLE set Unit_Act_Time = (select (T2.Const_0 + TEMPLATE_TABLE.Var_1 * T2.Const_1) * TEMPLATE_TABLE.Multiplier FROM TIME_TABLET2 where TEMPLATE_TABLE.Act_Name = T2.Act_Name AND TEMPLATE_TABLE.Var_1 IS NOT NULL AND TEMPLATE_TABLE.Var_2 IS NULL)

From above query it is expected that update `Unit_Act_Time` for only those rows for which `TEMPLATE_TABLE.Var_1 IS NOT NULL AND TEMPLATE_TABLE.Var_2 IS NULL`.

But actually there is no such row, so select statement returns nothing and NULL get updated to each row (Unit_Act_Time sets NULL).

The following correction fixes the issue. While setting values, select the required value (use only required where conditions) and keep other where condition outside of select.

```
update TEMPLATE_TABLE set Unit_Act_Time = (select (T2.Const_0 +
TEMPLATE_TABLE.Var_1 * T2.Const_1) * TEMPLATE_TABLE.Multiplier
from TIME_TABLET2 where TEMPLATE_TABLE.Act_Name = T2.Act_Name)
where
TEMPLATE_TABLE.Var_1 IS NOT NULL AND TEMPLATE_TABLE.Var_2 IS NULL
```

Insert primary key auto increment

In SQLite while inserting data if we want to insert filed to Auto increment then just have to pass NULL to it.

Select Into not working

SQLite does not support the following syntax:

```
insert into {newly created table} (field1, field2)
```

```
'dbCmd = New SQLite  Command("SELECT * INTO FLOW_TBL_ACT_STD_TIME_TEMP
FROM FLOW_TBL_ACT_STD_TIME", g_dbFlow)
```

```
dbCmd = New SQLite  Command("INSERT INTO FLOW_TBL_ACT_STD_TIME_TEMP SELECT *
FROM FLOW_TBL_ACT_STD_TIME", g_dbFlow)
```

Database opened in VB and calling C++ function

If database is already opened in VB.Net file and without closing it you try to call c++ function, and in C++ if some database updating is performed in C++, it will give error “database is locked”.

Such changes are accomplished by first getting the data in arrays and later calling C++ function on the data.

```
Dim arrPartIndex As New ArrayList
Dim arrUseScrapRate As New ArrayList
Dim arrRouteTable As New ArrayList
Dim arrRoundMoves As New ArrayList
Dim arrRoundingThreshold As New ArrayList
'-------------------------------------------------
Do While dbRdr.Read()
    //-Some code
    //-Some code
    //-Some code


//Populate data structure

arrPartIndex.Add(CInt(Val("" & dbRdr("Part_Index"))))
arrUseScrapRate.Add(lUseScrapRate)
arrRouteTable.Add(strRouteTable)
arrRoundMoves.Add(lRoundMoves)
arrRoundingThreshold.Add(dRoundingThreshold)

    //-Some code
    //-Some code
    //-Some code
dbRdr.Close()

Dim cnt As Integer
Dim counter As Integer
counter = 0
For Each cnt In arrPartIndex
    lCalResult = g_objFlowWrapper.Cimf_FlowMovesCalc(g_strDBName, CInt
(arrPartIndex.Item(counter)), CInt(arrUseScrapRate.Item(counter)),
 arrRouteTable.Item(counter).ToString(), CInt(arrRoundMoves.Item(counter)),
CDbl(arrRoundingThreshold.Item(counter)))
    counter = counter + 1
    If lCalResult = 0 Then Return False
Next
```

Executescalar issue

If executescalar returns null the it crashes for following lines, such as:

```
newUnitFreq = dbCmd2.ExecuteScalar
```

Change the code to:

```
Val("" & dbCmd2.ExecuteScalar)
```

Delete with * for deleting all element

SQLite does not allow * in delete query.

Original query	'Dim dbCmd As New SQLite Command("Delete * From PARTS_TABLE Where Part_Index Not In (Select Part_Index From Flow_Tbl_Data_Tree)", g_dbFlow)
Query adapted for SQLite	Dim dbCmd As New SQLite Command("Delete From PARTS_TABLE Where Part_Index Not In (Select Part_Index From Flow_Tbl_Data_Tree)", g_dbFlow)

Delete query with table.*

As update command delete query also has issue for using table shortcuts.

Original command not working in SQLite	TransactionExecute("delete N.* from NODES_TABLE N, Flow_Tbl_Data_Tree DT WHERE N.Part_Index = DT.Part_Index AND DT.Locked = 'N'")
Resolution	TransactionExecute("delete from NODES_TABLE WHERE Part_Index In (select Part_Index from Flow_Tbl_Data_Tree WHERE Locked = 'N')")

Using db.datasource not get correct file name

After SQLite connection, the following should provide the actual file name:

```
g_dbFlow.DataSource
```

Alter table alter table column , delete column is not supported in SQLite

SQLite has limited ALTER TABLE support that you can use to add a column to the end of a table or to change the name of a table. If you want to make more complex changes in the structure of a table, you will have to recreate the table. You can save existing data to a temporary table, drop the old table, create the new table, then copy the data back in from the temporary table.

Example

Suppose you have a table named "t1" with columns names "a", "b", and "c" and that you want to delete column "c" from this table. The following steps illustrate how this could be done:

```
BEGIN TRANSACTION;
CREATE TEMPORARY TABLE t1_backup(a,b);
INSERT INTO t1_backup SELECT a,b FROM t1;
DROP TABLE t1;
CREATE TABLE t1(a,b);
INSERT INTO t1 SELECT a,b FROM t1_backup;
DROP TABLE t1_backup;
COMMIT;
```

Original code	'dbCmd = New SQLite Command("ALTER TABLE RESULT_TABLE DROP COLUMN RECALC_FLAG", g_dbFlow) 'dbCmd.ExecuteNonQuery() 'dbCmd = New SQLite Command("ALTER TABLE RESULT_TABLE ADD COLUMN UNIT_LOAD_ID LONG", g_dbFlow) 'dbCmd.ExecuteNonQuery() 'dbCmd = New SQLite Command("ALTER TABLE RESULT_TABLE ADD COLUMN TRANSPORT_STACK_HT LONG", g_dbFlow) 'dbCmd.ExecuteNonQuery() 'dbCmd = New SQLite Command("ALTER TABLE RESULT_TABLE ADD COLUMN CONT_HT DOUBLE", g_dbFlow) 'dbCmd.ExecuteNonQuery() 'dbCmd = New SQLite Command("ALTER TABLE RESULT_TABLE ADD COLUMN RECALC_FLAG TEXT(1)", g_dbFlow) 'dbCmd.ExecuteNonQuery()
Code revised for SQLite	'Following column added 'UNIT_LOAD_ID LONG, TRANSPORT_STACK_HT LONG, CONT_HT DOUBLE, RECALC_FLAG TEXT(1)) strSql = "CREATE TABLE RESULT_TABLE_TEMPTBL (PART_INDEX LONG NOT NULL, FROM_NODE LONG NOT NULL, TO_NODE LONG NOT NULL, PROD_INDEX LONG, [PERCENT] DOUBLE, " strSql = strSql & "PROD_NAME TEXT(255), PROD_COLOR TEXT(15), PART_NAME TEXT(255), FROM_ACT_PNT_NAME TEXT(255), TO_ACT_PNT_NAME TEXT(255), " strSql = strSql & "FROM_ACT_PNT_TYPE TEXT(255), TO_ACT_PNT_TYPE TEXT(255), MH_EQUIP_NAME TEXT(255), MH_EQUIP_TYPE TEXT(30), " strSql = strSql & "MH_EQUIP_SPEED DOUBLE, MH_EQUIP_EFFECTIVENESS DOUBLE, MH_EQUIP_LAYER TEXT(255), MH_EQUIP_COLOR TEXT(15), " strSql = strSql & "FLOW_PATH_COLOR TEXT(15), CONT_NAME TEXT(255), CONT_TYPE TEXT(255), CONT_PER_TRIP DOUBLE, PARTS_PER_CONT DOUBLE, " strSql = strSql & "CONT_STORG_QTY DOUBLE, PARTS_MOVED DOUBLE, CONT_MOVED DOUBLE, MOVE_FREQ DOUBLE, CAL_DIST_FLAG TEXT(1), " strSql = strSql & "LOAD_ACT_FLAG TEXT(15), LOAD_ACT_TEMPLATE_NAME TEXT(255), UNLOAD_ACT_FLAG TEXT(15), UNLOAD_ACT_TEMPLATE_NAME TEXT(255), " strSql = strSql & "USE_MHE_LOAD_TEMPLATE_FLAG TEXT(1), USE_MHE_UNLOAD_TEMPLATE_FLAG TEXT(1), SGL_LOAD_TIME DOUBLE, SGL_UNLOAD_TIME DOUBLE, " strSql = strSql & "SGL_LOAD_UNLOAD_TIME DOUBLE, SGL_PATH_DIST DOUBLE, SGL_PATH_TIME

```
DOUBLE, ADJUST_PATH_DIST DOUBLE, "
strSql = strSql & "ADJUST_PATH_TIME DOUBLE, TOTAL_DIST DOUBLE, TOTAL_TIME DOUBLE,
VAR_COST DOUBLE, FIXED_COST DOUBLE, TOTAL_COST DOUBLE, "
strSql = strSql & "TO_STORG_TYPE TEXT(25), FLOOR_STACK_HT DOUBLE, FLOOR_FOOTPRINT
DOUBLE, ACTUAL_FLOOR_SPACE DOUBLE, "
strSql = strSql & "ADJUST_FLOOR_SPACE DOUBLE, SHELVE_FOOTPRINT DOUBLE,
ACTUAL_SHELVE_SPACE DOUBLE, ADJUST_SHELVE_SPACE DOUBLE, "
strSql = strSql & "LOADS_PER_DAY_RCV DOUBLE, LOADS_PER_DAY_SHIP DOUBLE,
FULL_EMPTY_FLAG TEXT(1), PATH_OBJ_ID TEXT(20), "
strSql = strSql & "EXIT_ACT_FLAG TEXT(1), DI_OBJ_ID TEXT(20), UNIT_LOAD_ID LONG,
TRANSPORT_STACK_HT LONG, CONT_HT DOUBLE, RECALC_FLAG TEXT(1))"
dbCmd = New SQLite Command(strSql, g_dbFlow)
dbCmd.ExecuteNonQuery()
'coping the data into temp table
strSql = "INSERT into RESULT_TABLE_TEMPTBL "
strSql = strSql & "(PART_INDEX , FROM_NODE , TO_NODE , PROD_INDEX , PERCENT , "
strSql = strSql & "PROD_NAME , PROD_COLOR , PART_NAME , FROM_ACT_PNT_NAME ,
TO_ACT_PNT_NAME , "
strSql = strSql & "FROM_ACT_PNT_TYPE , TO_ACT_PNT_TYPE , MH_EQUIP_NAME ,
MH_EQUIP_TYPE , "
strSql = strSql & "MH_EQUIP_SPEED , MH_EQUIP_EFFECTIVENESS , MH_EQUIP_LAYER ,
MH_EQUIP_COLOR , "
strSql = strSql & "FLOW_PATH_COLOR , CONT_NAME , CONT_TYPE , CONT_PER_TRIP ,
PARTS_PER_CONT , "
strSql = strSql & "CONT_STORG_QTY , PARTS_MOVED , CONT_MOVED , MOVE_FREQ ,
CAL_DIST_FLAG , "
strSql = strSql & "LOAD_ACT_FLAG , LOAD_ACT_TEMPLATE_NAME , UNLOAD_ACT_FLAG ,
UNLOAD_ACT_TEMPLATE_NAME , "
strSql = strSql & "USE_MHE_LOAD_TEMPLATE_FLAG , USE_MHE_UNLOAD_TEMPLATE_FLAG ,
SGL_LOAD_TIME , SGL_UNLOAD_TIME , "
strSql = strSql & "SGL_LOAD_UNLOAD_TIME , SGL_PATH_DIST , SGL_PATH_TIME ,
ADJUST_PATH_DIST , "
strSql = strSql & "ADJUST_PATH_TIME , TOTAL_DIST , TOTAL_TIME , VAR_COST , FIXED_COST ,
TOTAL_COST , "
strSql = strSql & "TO_STORG_TYPE , FLOOR_STACK_HT , FLOOR_FOOTPRINT ,
ACTUAL_FLOOR_SPACE , "
strSql = strSql & "ADJUST_FLOOR_SPACE , SHELVE_FOOTPRINT , ACTUAL_SHELVE_SPACE ,
ADJUST_SHELVE_SPACE , "
strSql = strSql & "LOADS_PER_DAY_RCV, LOADS_PER_DAY_SHIP, FULL_EMPTY_FLAG,
PATH_OBJ_ID, EXIT_ACT_FLAG, RECALC_FLAG"
strSql = strSql & ") select PART_INDEX , FROM_NODE , TO_NODE , PROD_INDEX , PERCENT , "
strSql = strSql & "PROD_NAME , PROD_COLOR , PART_NAME , FROM_ACT_PNT_NAME ,
TO_ACT_PNT_NAME , "
strSql = strSql & "FROM_ACT_PNT_TYPE , TO_ACT_PNT_TYPE , MH_EQUIP_NAME ,
MH_EQUIP_TYPE , "
strSql = strSql & "MH_EQUIP_SPEED , MH_EQUIP_EFFECTIVENESS , MH_EQUIP_LAYER ,
MH_EQUIP_COLOR , "
strSql = strSql & "FLOW_PATH_COLOR , CONT_NAME , CONT_TYPE , CONT_PER_TRIP ,
PARTS_PER_CONT , "
```

```
strSql = strSql & "CONT_STORG_QTY , PARTS_MOVED , CONT_MOVED , MOVE_FREQ ,
CAL_DIST_FLAG , "
strSql = strSql & "LOAD_ACT_FLAG , LOAD_ACT_TEMPLATE_NAME , UNLOAD_ACT_FLAG ,
UNLOAD_ACT_TEMPLATE_NAME , "
strSql = strSql & "USE_MHE_LOAD_TEMPLATE_FLAG , USE_MHE_UNLOAD_TEMPLATE_FLAG ,
SGL_LOAD_TIME , SGL_UNLOAD_TIME , "
strSql = strSql & "SGL_LOAD_UNLOAD_TIME , SGL_PATH_DIST , SGL_PATH_TIME ,
ADJUST_PATH_DIST , "
strSql = strSql & "ADJUST_PATH_TIME , TOTAL_DIST , TOTAL_TIME , VAR_COST , FIXED_COST ,
TOTAL_COST , "
strSql = strSql & "TO_STORG_TYPE , FLOOR_STACK_HT , FLOOR_FOOTPRINT ,
ACTUAL_FLOOR_SPACE , "
strSql = strSql & "ADJUST_FLOOR_SPACE , SHELVE_FOOTPRINT , ACTUAL_SHELVE_SPACE ,
ADJUST_SHELVE_SPACE , "
strSql = strSql & "LOADS_PER_DAY_RCV, LOADS_PER_DAY_SHIP, FULL_EMPTY_FLAG,
PATH_OBJ_ID, EXIT_ACT_FLAG, RECALC_FLAG"
strSql = strSql & " from(RESULT_TABLE)" dbCmd = New SQLite Command(strSql, g_dbFlow)
dbCmd.ExecuteNonQuery()
'drop old table,
dbCmd = New SQLite Command("drop table RESULT_TABLE", g_dbFlow)
dbCmd.ExecuteNonQuery()
'rename the new one.
dbCmd = New SQLite Command("ALTER TABLE RESULT_TABLE_TEMPTBL RENAME TO
RESULT_TABLE", g_dbFlow)
dbCmd.ExecuteNonQuery()
```

Replacement for CompactAndRepair

Instead of compact and repair we can just use VACUUM command in SQLite. Vacuum command works similar to compact and repair.

The VACUUM command rebuilds the entire database. There are several reasons an application might do this:

Unless SQLite is running in "auto_vacuum=FULL" mode, when a large amount of data is deleted from the database file it leaves behind empty space, or "free" database pages. This means the database file might be larger than strictly necessary. Running VACUUM to rebuild the database reclaims this space and reduces the size of the database file.

Frequent inserts, updates, and deletes can cause the database file to become fragmented - where data for a single table or index is scattered around the database file. Running VACUUM ensures that each table and index is largely stored contiguously within the database file. In some cases, VACUUM may also reduce the number of partially filled pages in the database, reducing the size of the database file further.

Normally, the database page_size and whether or not the database supports auto_vacuum must be configured before the database file is actually created. However, when not in write-ahead logmode, the page_size and/or auto_vacuum properties of an existing database may be changed by using the page_size and/or pragma auto_vacuum pragmas and then immediately VACUUMing the database. When in write-ahead log mode, only the auto_vacuum support property can be changed using VACUUM.

More information can found on http://www.SQLite .org/lang_vacuum.html

```
Dim dbCmd As New SQLite  Command("VACUUM", g_dbFlow)

                    Try
                        dbCmd.ExecuteNonQuery()
                        g_dbFlow.Close()
                    Catch ex As Exception
                        MessageBox.Show(ex.Message)
                    End Try
```

SQLITE date time format

While inserting elements in DATE type of column make sure that date is only this format:

YYYY-MM-DD HH:MM

No seconds and no AM or PM.

Getting data from date field from .NET

```
Dim dateTimeFormat As DateTime
  dateTimeFormat = dbRdr("Start_Window")
  strr = dateTimeFormat.ToString()
```

SQLite case sensitivity

To make case in-sensitive column, make sure that while creating text column it should be of type "TEXT COLLATE NOCASE".

Example

```
CREATE TABLE TBL_TMP_DLG_SETTING (DLG_NAME TEXT(50) NOT NULL COLLATE NOCASE, ...
```

It can be used while sorting:

```
/* Sorting of column c is performed using the NOCASE collating sequence. */
SELECT x FROM t1 ORDER BY c COLLATE NOCASE, x;
```

Notes:

Chapter 11: Global technical access center (GTAC)

To report any serious problems regarding the software, please contact Global Technical Access Center.

Phone:

- USA and Canada: 800-955-0000 or 714-952-5444
- Outside the United States and Canada: Contact your local support office.

Website:

You can log incident reports and view any existing resolutions for incident reports on the Web at https://support.industrysoftware.automation.siemens.com/gtac.shtml.

Notes:

Chapter 12: Siemens PLM Community

The Siemens PLM Community for FactoryCAD, FactoryFLOW, and In Context Editor are available at http://community.plm.automation.siemens.com/t5/General-Tecnomatix-Forum/bd-p/tecnomatix-forum . The Tecnomatix Community (Siemens PLM Community for FactoryCAD, FactoryFLOW, and In Context Editor) is also available through the **Factory Layout Software Help**.

> **Note**
>
> A webkey and password are required to log on to the forum. Licensed users can get WebKey support at http:/www.siemens.com/plm/support.

Siemens Industry Software

Headquarters
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

Europe
Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific
Suites 4301-4302, 43/F
AIA Kowloon Tower, Landmark East
100 How Ming Street
Kwun Tong, Kowloon
Hong Kong
+852 2230 3308

About Siemens PLM Software

Siemens PLM Software, a business unit of the Siemens Industry Automation Division, is a leading global provider of product lifecycle management (PLM) software and services with 7 million licensed seats and 71,000 customers worldwide. Headquartered in Plano, Texas, Siemens PLM Software works collaboratively with companies to deliver open solutions that help them turn more ideas into successful products. For more information on Siemens PLM Software products and services, visit www.siemens.com/plm.