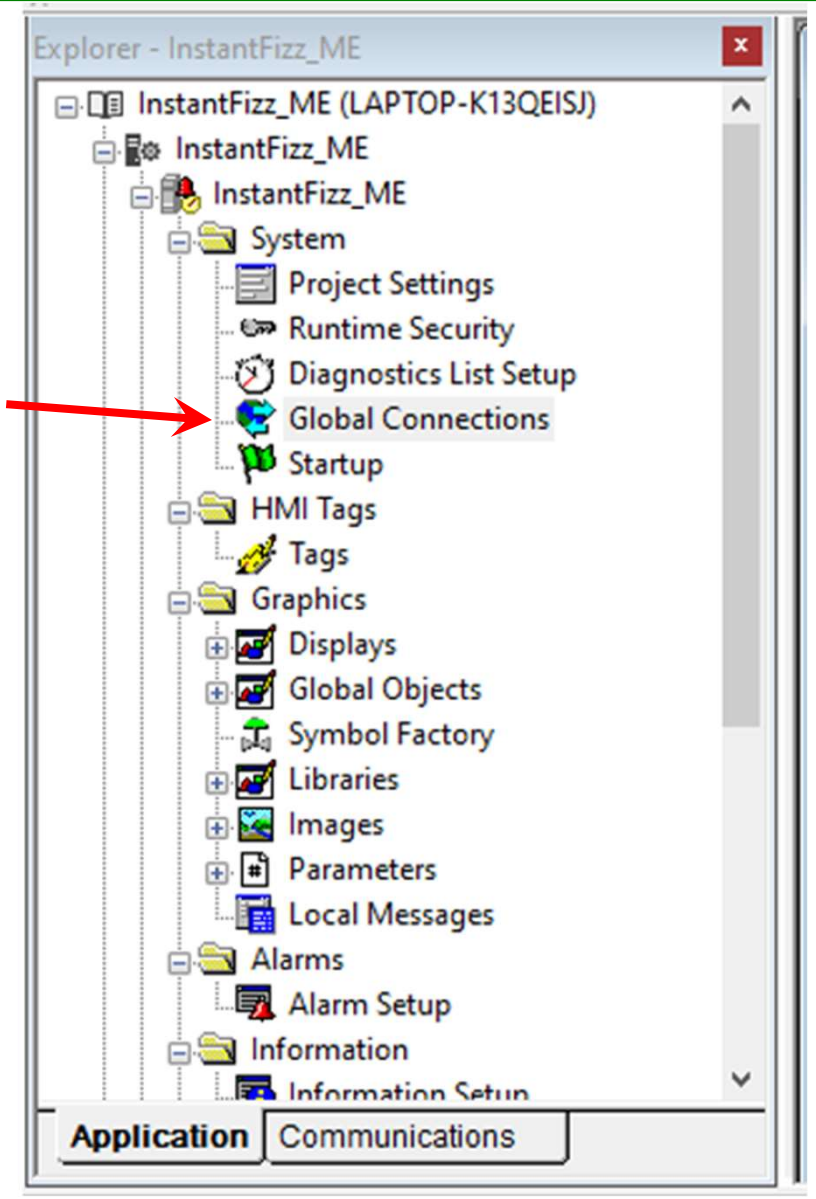
Global Connections & Creating Runtime Applications

Dan Kandray

January 30, 2020

Global Connections

- Global connections are connections that apply to your entire runtime application.
- Global connections allow the data source to control or interact with your application at runtime.



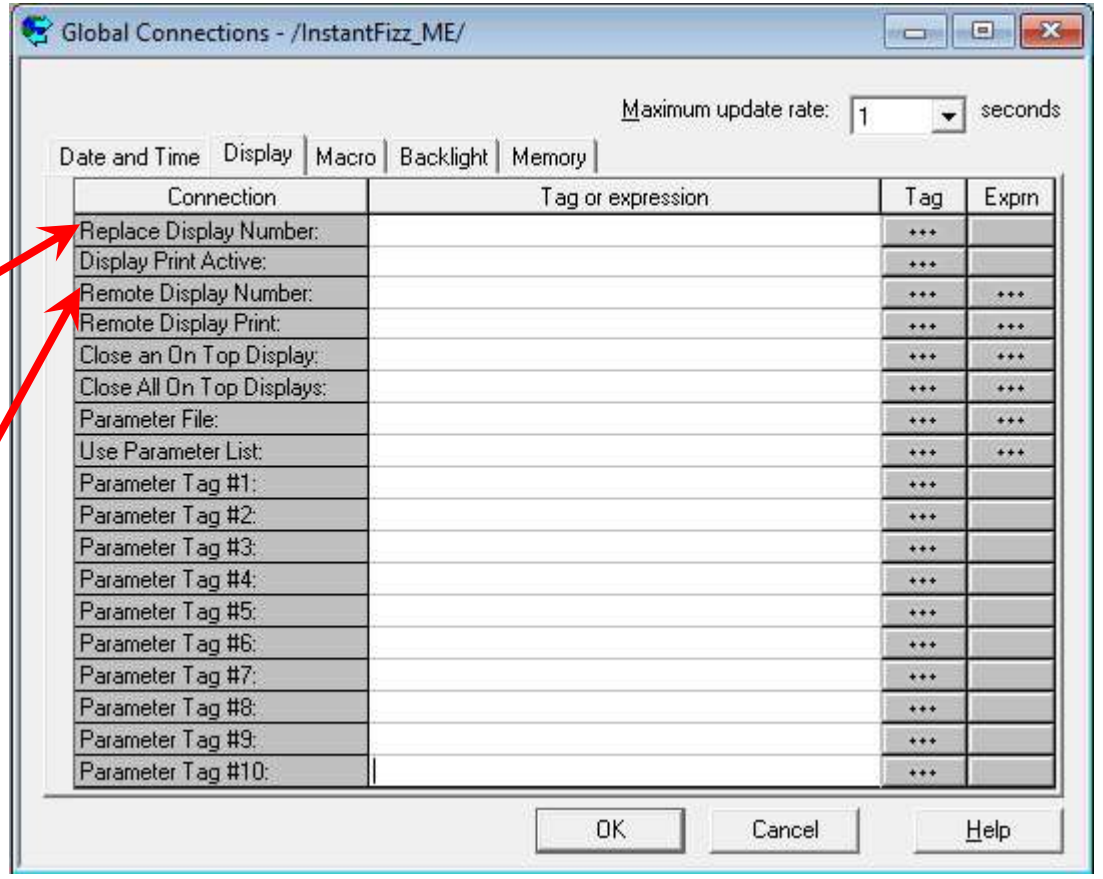
Use global connections to:

Swagelok

- Update the date and time at the data source using the runtime terminal's date and time.
- Update the date and time on the runtime terminal using the data source date and time.
- Notify the data source of the current display number.
- Change the display on the runtime terminal remotely (from the data source).
- Close On Top displays
- Apply parameter files or parameter lists to the tag placeholders in the changed displays.
- Run up to five macros from the data source, when a tag or expression result changes.
- Set the intensity of the backlight on a MobileView, PanelView Plus 7, PanelView Plus 6, PanelView Plus, PanelView Plus Compact, or PanelView Plus CE terminal.
- Monitor runtime RAM usage for troubleshooting purposes.

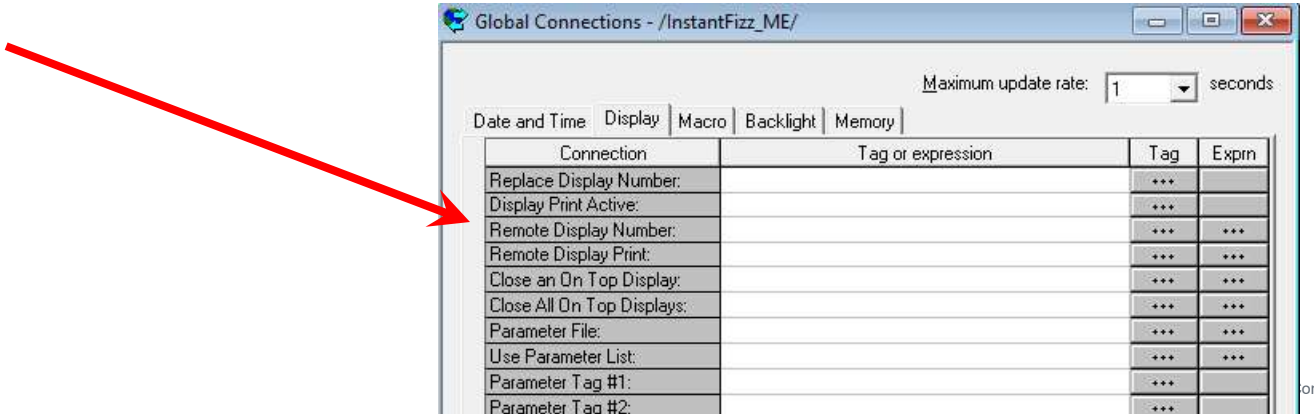
Change displays

- Use these global connections on the Display tab of the Global Connections editor to monitor and control display changes:
  - **Replace Display Number** — notifies the data source of the number of the Replace display that's currently open.
  - **Remote Display Number** — allows the data source to change the display on the runtime computer.



Control display changes remotely

- To control display changes remotely, you can set up the data source to open graphic displays using global connections.
- For example, the Remote Display Number connection is a global connection that you can use to control display changes from the data source (ie the PLC)
- Remote display changes and security
  - If you set up the data source to open graphic displays remotely, remote display changes occur whether the logged-in user has security access to a given display.



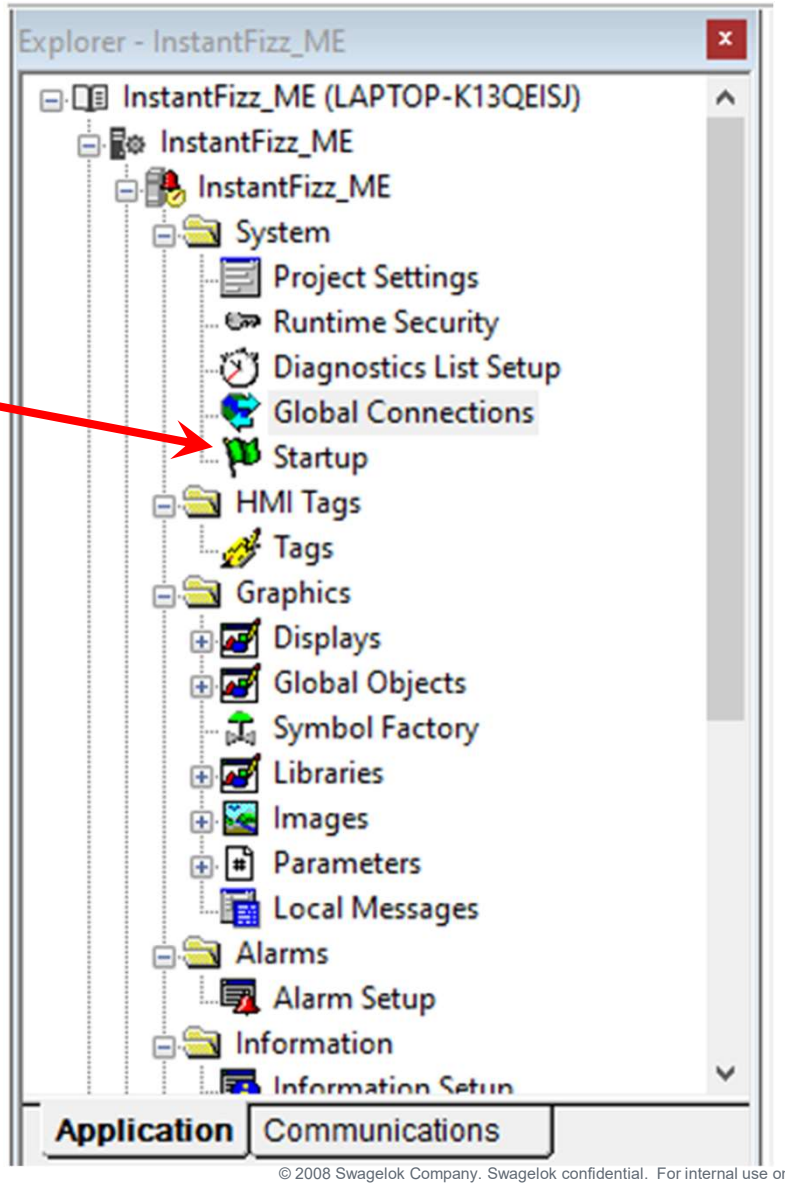
SETTING UP RUNTIME DISPLAYS

Create runtime application files

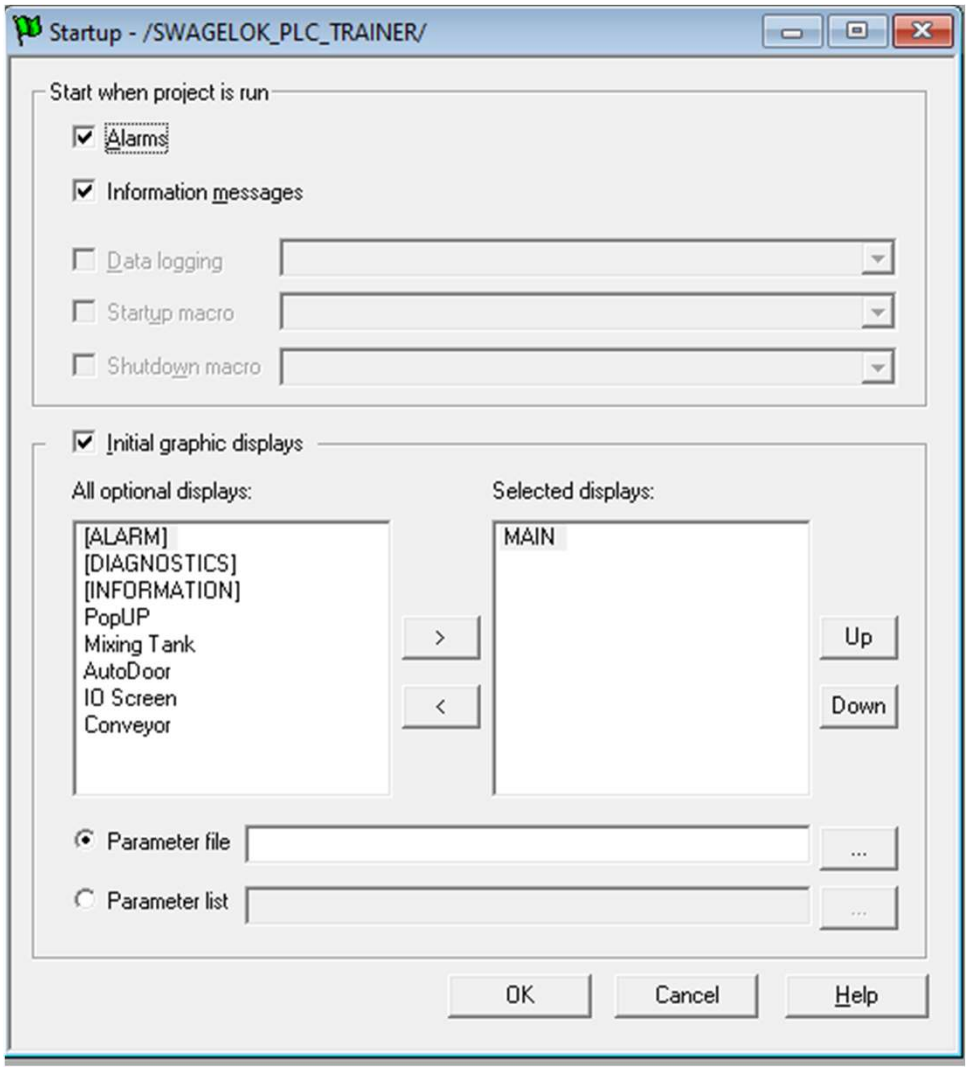
- Before you can run an application, you must create a runtime version.
- When you create the runtime version, FactoryTalk View Studio compiles all of the necessary application information into a single file with the extension .mer.
- This is the file that gets placed on the HMI Terminal and is executed.

Steps to Creating a Runtime Application

1. Specify startup settings
   - Use the Startup editor to specify which application processes and components to start when the application starts at runtime.
   - You can specify startup settings once you have set up all the parts of the application, or you can specify processes and select components in the Startup editor as you create them.



Start up editor

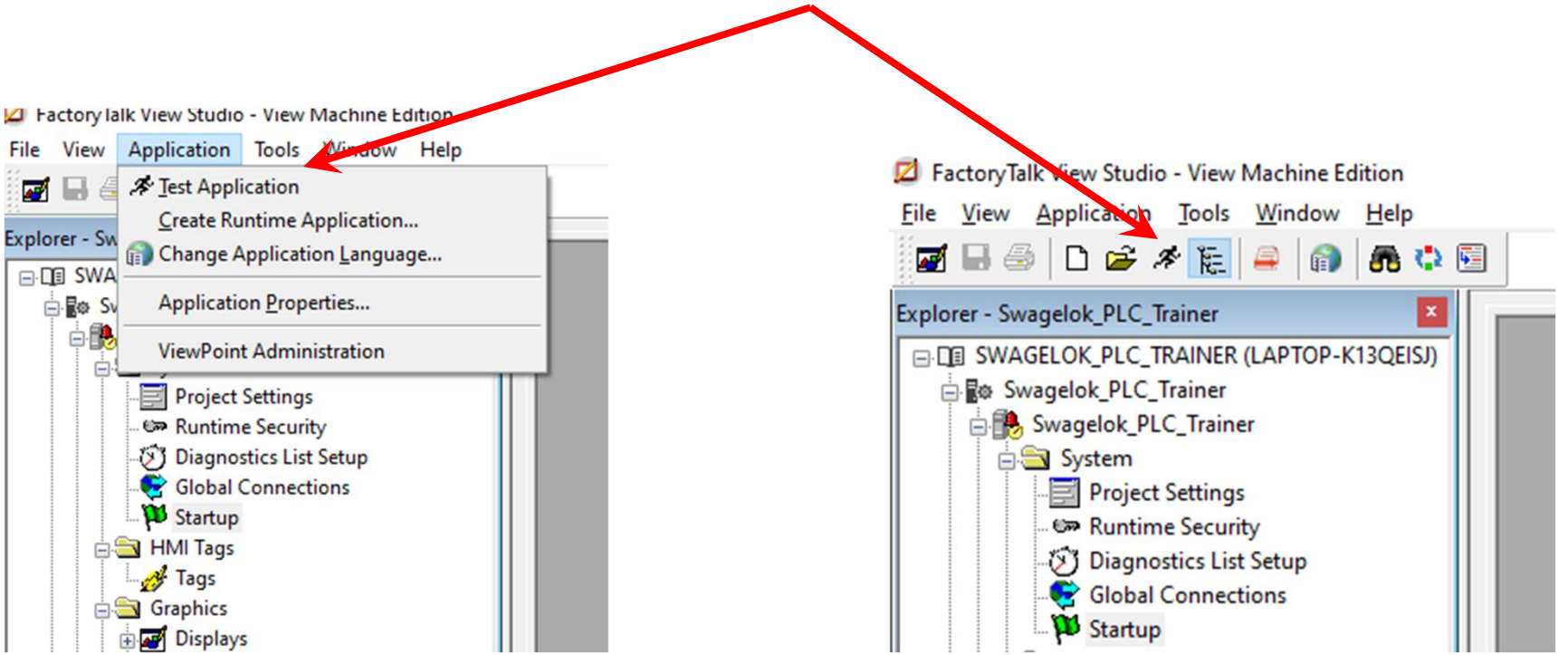


Steps to Creating a Runtime Application

Swagelok

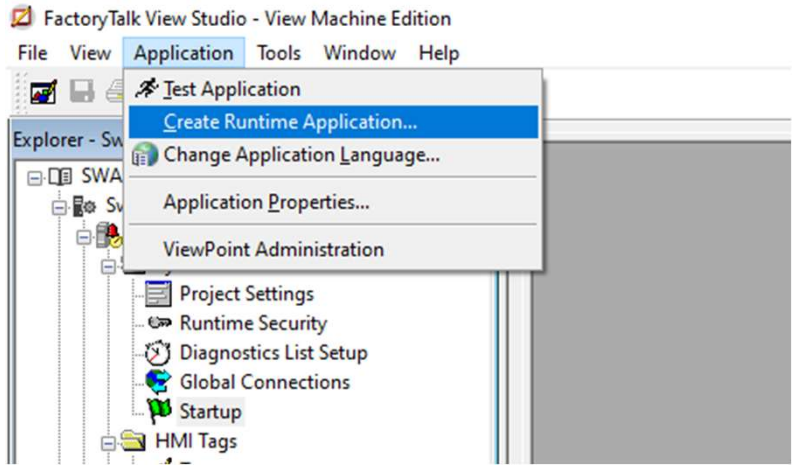
2. Test the application
   - You can test your application in FactoryTalk View Studio at any time during the development process, to make sure that everything works the way you intend.
   - If the development computer is connected to the data source, you can test all functions of the application, including security settings, language switching, communications, and alarm monitoring.
   - A FactoryTalk View ME Station emulator opens on the development computer and runs the application. This runtime version of the application is a temporary version for testing use only. You cannot run it on another computer.
   - There is a two-hour time limit for test running the application in FactoryTalk View Studio.

Testing the application

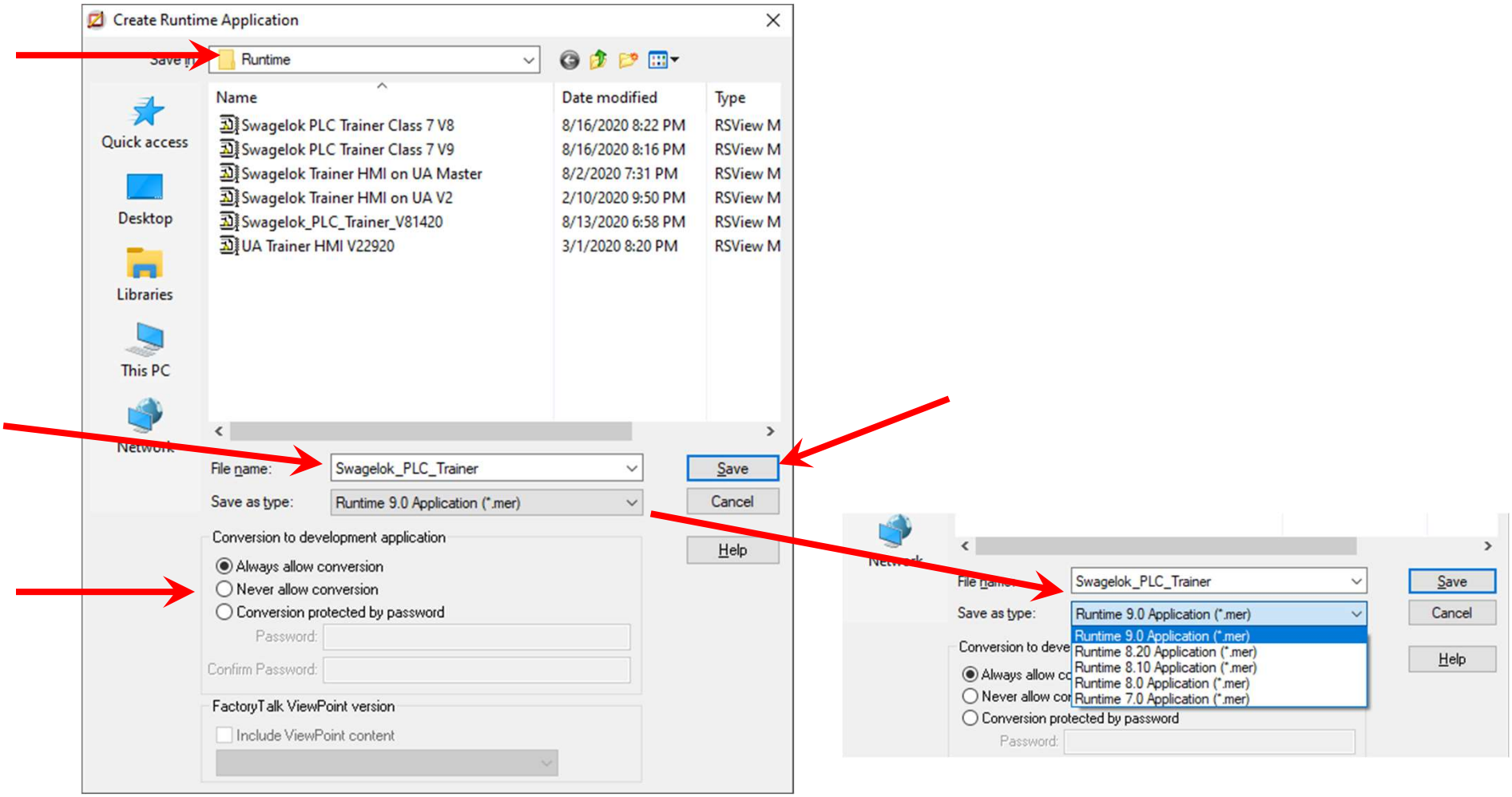


Steps to Creating a Runtime Application

3. Create runtime application files
   - Before you can run an application, you must create a runtime version.
   - When you create the runtime version, FactoryTalk View Studio compiles all of the necessary application information into a single file with the extension .mer.



3. Creating Runtime Application



4. Copy Runtime file to Terminal

- Open Transfer Utility from the Tools pull-down menu