

## GOOP<sup>©</sup> Development Suite – News in version 3.1

This release includes installers for LabVIEW 8.6 and 8.5. For LabVIEW 8.2.1 we refer to GOOP Development Suite v2.5. For earlier LabVIEW versions: use the GOOP Inheritance Toolkit and UML Modeller v1.

### New Features:

- Clone VI is improved – you can now clone a method VI to multiple classes!
- Updated GOOP 3 class templates (v1.1.1.0) and GOOP 3 kernel VIs (v3.1.2)
  - Added user reference counter for named objects. You can track how many clients are using each object.
  - Improved performance in Create and Destroy
  - Cleaner method template for Create. Cleaned up so the ObjectReference.ctl don't need to be used in Create.
  - Bux fix for object status.
- Updated the “Update to new class template” feature for the new class templates.

#### Q1. How do I use the new class template and kernel?

A. They are automatically used for all classes created with the new version of the tool. The GOOP kernel (run-time VIs used by all GOOP3 classes) is backward compatible. Existing classes will automatically use the new GOOP kernel.

#### Q2. How do I update existing classes to the new template?

A. Select “Tools->GOOP->Class Template Update...”. This will open a UI where you can select individual classes or update all classes in your project. The update will remove unnecessary wires etc. and keep your code executable (backup first). NOTE. This feature is not available for Community Edition users, it is only for purchased licenses of the tool.

### Fixed issues:

- Check if file is read-only before saving.
- Open VI function in State Diagram open the VI in wrong application instance.
- If file UMLModeller\_WhereIsIniFileStored.ini points to an unexisting location the UML tool won't start.
- Changed the class template of GetObjectStatus to re-entrant.
- Rename class using a prefix and an underscore results in incorrect name.

- After a Clone Method function has been performed and it overwrites the target file, then the all Source Code is selected on the block diagram.
- Error in labeling an transition in the state diagram – UML tool crashes.
- Tab order in Add Method dialog improved.
- New Feature Added: Search function for finding objects on diagram.
- Edit Icon function don't save the VI after updating.
- Can't debug classes located under folders inside lvlibs.
- Icon Body Text incorrect for certain names.

## **Did you know that...**

The tool is packed with cool features and you probably haven't seen them all. Here are some features you may not have noticed:

- You can launch the VI Icon editor on any VI (class VI or standard VI). Right click on the VI and select "GOOP->Edit Icon...".
- You can select one class to reverse engineer, it will show that class and the classes it use. Right click on the class and select "UML->"View class in class diagram".
- You can add an Active Object design pattern to an existing class. This feature adds a process VI and an internal job queue for the class. Right click on a class (native or GOOP3) and select "GOOP->Add Design Pattern".
- You can reverse engineer state machine VIs into a UML state diagram. This feature works for enum and string based state machines.
- When you are bored: try the random color generation in the Class Icon UI or, in the UML tool, try the "Tools->Auto Place..." features. These will cheer you up.