19.0 SP1 Version News

Alf Plugin

Released on: November 12, 2018

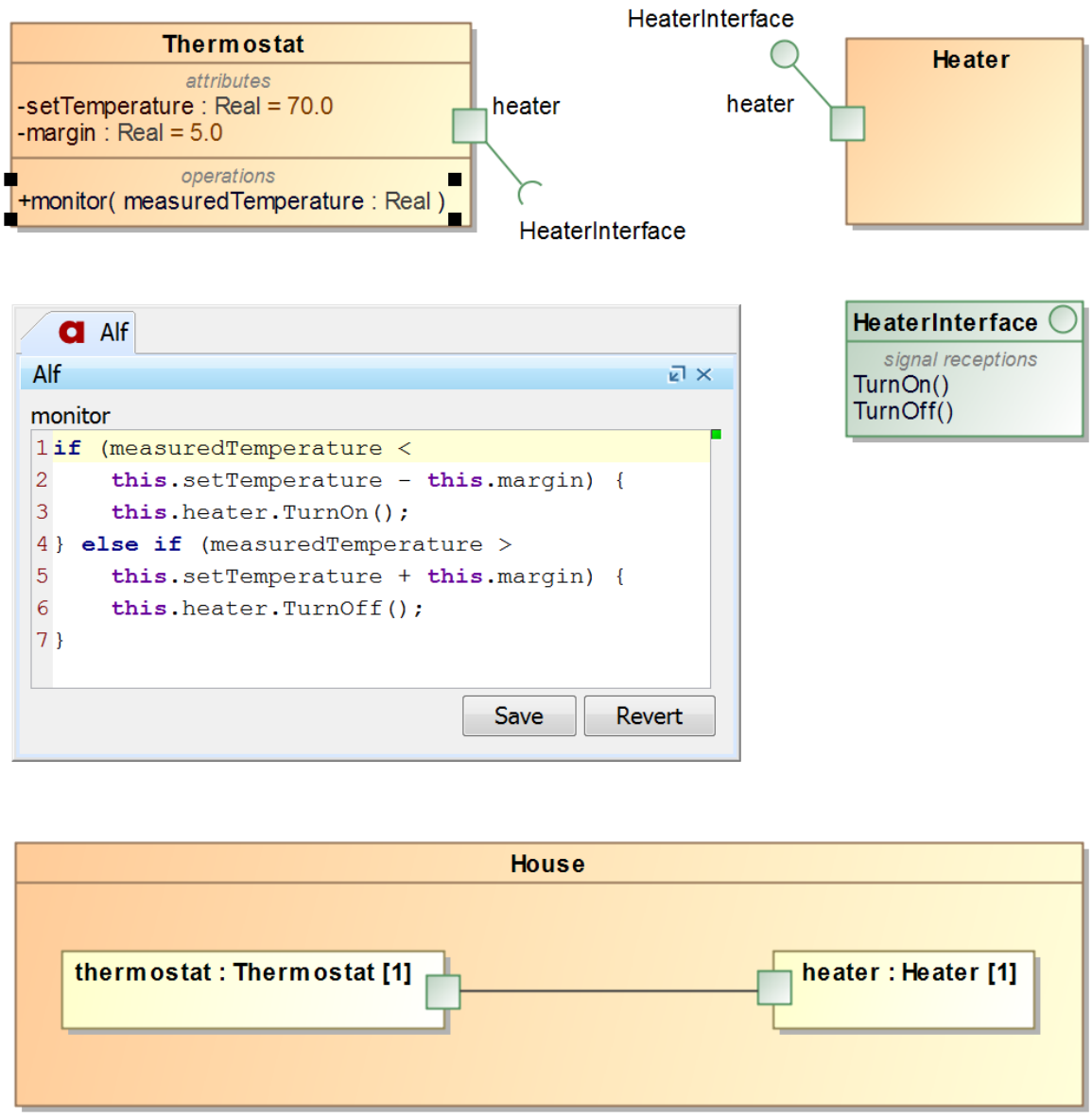
The Alf Plugin version 19.0 SP1 adds new features for making invocations through ports using Alf, viewing the Alf bodies of read-only elements and resetting a project through the Open API. It also provides bug fixes.

What you get:

Making invocations Through Ports

You can access a Port using the usual Alf notation for a Property. Sending a Signal or calling an Operation on the Port (consistent with the Port's type) then makes an invocation through the Port and across any Connector attached to the Port. That is, the Alf invocation is mapped into an invocation action whose **onPort** property is set to the target Port.

The Alf language requires that the type a property that is the target of a Signal send have a Signal Reception for the Signal being sent. Therefore, in order to send a Signal through a Port, the type of the Port *must* have a Signal Reception for the Signal.

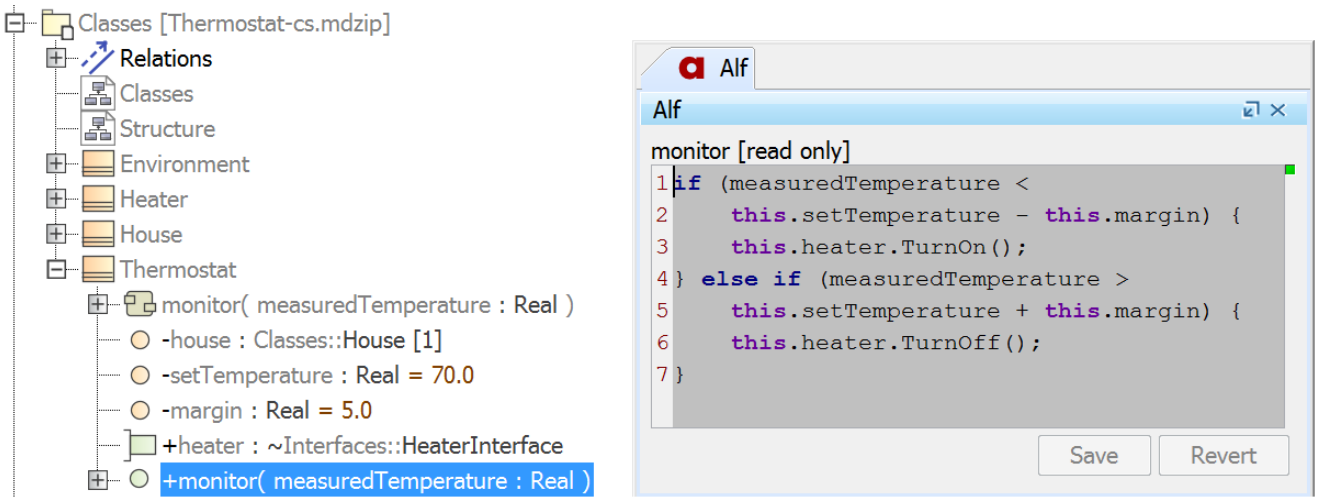


Using Alf to send Signals through a Port

Back to top

Viewing Read-Only Alf Bodies

Sometimes an Alf body will be attached to an element that is editable, such as when it is in a read-only used project or a part of a Teamwork Cloud project that is not locked for edit. In this case, the Alf Editor will still show the Alf code for the element, but on a gray background to indicate that the code is not editable. If the element becomes editable (for instance, by locking it for edit in Teamwork Cloud), then the Alf Editor will allow the code to be edited.



Viewing the Alf body of a read-only element

Back to top

Open API to Reset a Project

Once the *AlfImporter* API is used on a project, it is no longer consistent to use the *AlfCompiler* API with that project. To be able to use the *AlfCompiler* API again, you previously had to save, close and re-open the project. Now, you can instead simply use the new *AlfActionUtil,.resetActiveProject* method to return the currently active project to a state in which the *AlfCompiler* can be used, or use to *AlfActionUtil.resetProject* method to similarly reset a specific project.

```
AlfImporter importer = new AlfImporter(modelDirectory, progressStatus);
Path path = Paths.get(modelDirectory, modelFileName);
if (importer.parse(path)) {
    AlfActionUtil.executeSession("Import Alf", new Runnable() {
        public void run() {
            importer.compile(path);
        }
    });
}
// It is unsafe to use the AlfCompiler API at this point.

AlfActionUtil.resetActiveProject();

// It is safe to use the AlfCompiler API from here on...
```

Back to top

Bug Fixes

The following bugs have been fixed in the Alf Plugin 19.0 SP1:

- When a project is opened, the Alf Plugin searched the entire project for Alf code, even if the project was not an Alf project.
- In certain cases, a constraint violation was being reported on an Activity without an Alf body.
- The SysML *Real* primitive value type was not being properly recognized as being equivalent to the UML *Real* primitive type for the purposes of Alf type checking.

In addition, the Alf compiler was updated to Alf Reference Implementation v1.1.0e, which fixed a few minor compiler bugs.

Back to top