AGREE tutorial

JULIEN DELANGE
Software Engineering Institute
jdelange@sei.cmu.edu

November 6, 2014

1 Introduction

This document is a tutorial to learn to use the AGREE language and its associated toolset. This is not a user-manual that covers all aspect of the language features, all these aspects are described in the AGREE user-manual[1]. This document is a way to learn how to use the language and its associated tools through several case studies.

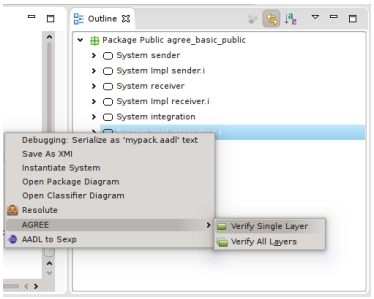

Figure 1: AGREE menu in OSATE outline

1.1 Use Analysis Tools

To use AGREE, model components must be annotated with AGREE annex subclauses. Then, invoking AGREE can be done by selecting the top-level system instance and make a right-click and select two options:

1. **Verify Single Layer**: analyze and verify only one depth of the component hierarchy.
2. **Verify All Layers**: analyze the complete components hierarchy.

[1] https://github.com/smaccm/smaccm/tree/master/documentation/agree

1.2 Limitations

When using AGREE, your models must enforce some constraints. There is the list of the constraints your model has to enforce:

- **Execution Order**: the execution order of the model is done in the order of the declaration of the subcomponents.
- **Multiple fanin** are **not** supported. In other words, an incoming feature can have only one incoming connection.
- **Top-level component** must have an AGREE subclause, even if you do not want to verify anything and want to validate the subcomponent. Hopefully, you can insert a dummy subclause like the one shown below.

```
system mysystem
annex agree {**
   guarantee "dummy" : true;
**};
end mysystem;
```

1.3 Understanding Analysis Results

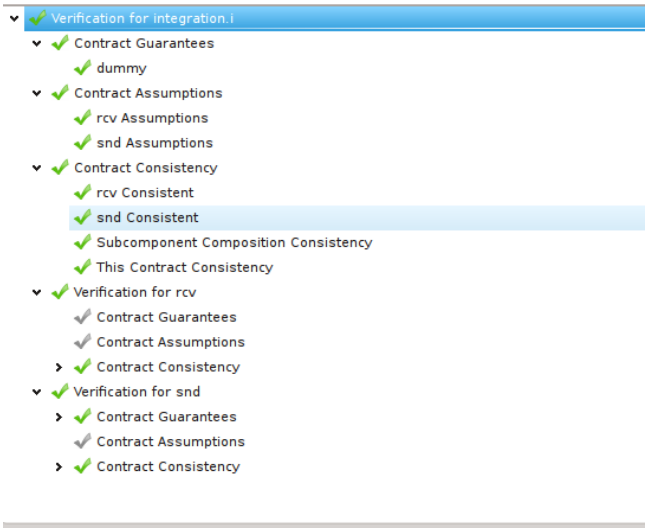

Figure 2: AGREE Results View in OSATE