



# Trace-based Debugger for Dynamically Composed Applications

Master thesis for: Philipp Lengauer

Matr. No.: 0656104

Email:Lengauer (at) ase.jku.at

The plug-in framework Plux allows developers to build components. At runtime Plux can compose such components to an application. This is similar to other plug-in frameworks such as Eclipse, OSGi or NetBeans. However, in contrast to other frameworks, Plux allows end users to add and remove components from application at run time without restarting it. To enable this, Plux imposes high demands on the components, i.e., each component must be able to cope with the addition and removal of every other component in any possible order. When a component doesn't meet this requirement it will crash or yield an unexpected behavior. In this master thesis, a tool should be developed to help the developer of a component to figure out why the component failed, and what part of the component failed.

The tool must be able to:

- 1) Log all Plux composition events at run time.
- 2) Visualize the created log file.
- 3) Search for errors in the execution automatic and user guided.
- 4) Compare log files and highlight the differences. Different comparison strategies must be supported, e.g.: a simple difference similar to text diff utilities, or a search for missing composition events.
- 5) Filter log files and highlight relevant or hide irrelevant parts, e.g., Filter by the time or the order of the events.

The tool must be created in C# for the plug-in framework Plux. The work and the code must be discussed on a regular basis with the supervisor.

Supervisor: DI Markus Löberbauer

Start: Februar 2011