# OSSS+R: Simulation of Reconfigurable Architectures

Dipl. Inform. Andreas Schallenberg, Dipl. Inform. Frank Oppenheimer and
Prof. Dr.-Ing. Wolfgang Nebel

Carl v. Ossietzky University Oldenburg, Germany

Runtime reconfigurable systems are a promising way to solve problems when designing embedded systems since they combine software-like flexibility with hardware parallelism and performance. Today there are prototyping platforms available on the market and intensive scientific research on how to exploit this opportunity is being done. Designing complex systems requires an adequate way of modeling. Unfortunately the common description languages suffer from two major drawbacks: They provide a low level of abstraction and focus on static ASIC-style hardware. The former problem is being addressed by hardware description languages like SystemC[1] or OSSS[2], which is based on SystemC.

The approach presented here named OSSS+R extends OSSS with capabilities to model, simulate and synthesise reconfigurable architectures. The simulation capabilites are demonstrated at the booth.

Since OSSS is an object oriented language the reconfigurable components are also modeled as objects in OSSS+R. The key idea is the use of polymorphism. Reconfigurable areas on a FPGA and polymorphic objects in the modeling world share important properties: Reconfiguring a part of a FPGA during runtime requires that the non-changing part of the FPGA can still communicate with the modified part. Therefore the interface between them should be static although the implementation of the reconfigured part has been modified and now provides new functionalities. On the other hand using polymorphism means having a typed reference to an object which type is not necessarily exactly known. The object instance and object type may be exchanged during runtime but the type of the reference does never change. The reference is the interface to the objects contents, just like a counterpart to the interface of the reconfigured area which provides access to the different configurations. Figure 1 illustrates the similarities.

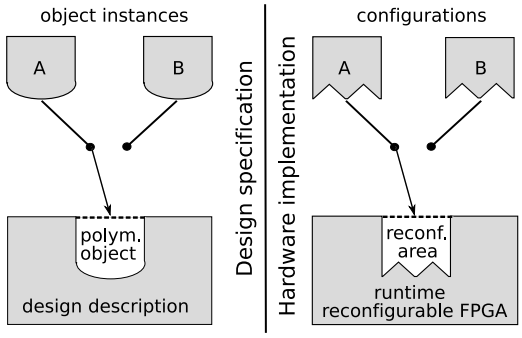


Figure 1: Polymorphism and configurations

Implementing a dynamic design requires some mechanism to remember the current configuration, detect necessary changes and resolve conflicts due to parallel requests to exclusive ressources. This infrastructure is complex and its manual design is an error-prone and time consuming task. Fortunately the infrastructure can be automatically generated for OSSS+R designs when some guiding information is provided by the designer. This guiding information contains a selection of arbitration algorithms, structural information (like FPGA types) and timing information (e.g. expected reconfiguration times). The generated infrastructure consists of a set of hierarchically organised controllers. A set of distributed controllers for each reconfigurable area handles access requests by the static design parts. Requests are delayed when the access requires a reconfiguration or other accesses are currently on the fly. The controller uses the central unit to accomplish reconfigurations. That unit resolves conflicts between different distributed controllers and provides an interface to the FPGAs configuration port.
The tool which is shown on the booth already allows a cycle accurate simulation of OSSS+R models. The simulation environment will be publically available just like the OSSS simulation environment already is.
The project is partially funded by the DFG under grant No. NE 629/7.

[1] http://www.systemc.org
[2] http://odette.offi s.de