Design Tool Encapsulation in CAD Frameworks
- Dissertation Thesis -

Olav Schettler

Abstract

Electronic design automation (EDA) environments aid in the design of large electronic systems by maintaining a wealth of *structural information* about design objects. Design objects reference the actual *representation information* and are associated with domain, level of abstraction, version, and status. Relationships between design objects describe hierarchical composition and functional equivalence.

Open EDA environments are constructed from framework components which provide *design information management* and other services and the design tools which perform the actual design steps. Roughly, design tools fall in two groups with respect to their incorporation into a framework-based EDA environment. Tightly integrated tools access framework services directly through application programming interfaces. Encapsulated tools interface to framework services by means of wrappers. Unfortunately, as long as generally accepted standard framework interfaces are not available, many relevant third-party design tools can only be encapsulated. With these tools, automatic maintenance of consistency between structural and representation information is particularly hard because important information is hidden in design files and is not readily available to design information management by an EDA environment.

In this thesis, a new framework service is described that facilitates mapping between structural and representation information managed by an EDA environment and syntactically correct and complete design files manipulated by encapsulated design tools. With this new service, tool integrators can build advanced tool wrappers that analyse files resulting from a design step and update the structural information of affected design objects accordingly.

A conceptual schema for structural information and a generator for design file processors are described. Advanced tool wrappers are scripted in a framework extension language that allows to manipulate structural information. Design file processors are used in these wrappers to extract relevant structural information and to construct valid design files from representation information associated with design objects.

A design environment for high-level synthesis is reported that comprises an implementation of the conceptual schema, an extension language, and generated VHDL language processors. Some commercial design tools are encapsulated into it. In this environment, structural information and its automatic management is available to the designer regardless of the way a particular design tool is incorporated into the environment.