

Project A6-9805, A tool environment for the development of embedded systems

Most of the research performed within the ARTES programme is related to temporal analysis and especially scheduleability and execution time analysis that are important. However, very little attention has been made to software architectural issues, such as reuseability, maintainability, and flexibility, of software constructions. Moreover, the success of a software system throughout its complete lifecycle depends heavily on experienced individuals (oracles) that make the correct design decisions. Since current architecture design is based on oracles, we support all activities that try to establish design of real-time systems as engineering discipline. Applying formal methods on the architectural level of real-time software constructions is interesting. Moreover, the approach proposed in the licentiate thesis makes modelling of hybrid systems possible. Real-time systems typically interacts with- and reacts on its environment. Thus, a real-time system consists of both software and the physical hardware that is controlled. Consequently, making models of both hardware and software in order to investigate and verify the complete systems is very important.

Systemite supports activities that may result in methods and tools that can be used to increase efficiency of development teams and potentially be a part of a total SystemWeaver concept.

Systemite develops methods and technical solutions to enhance the efficiency in the development process for complex systems in distributed organisations.

Systemites mission is to increase the customer's efficiency and resource usage in the development process by introducing component based systems development methodologies and relevant tool support.

In the area of "relevant tool support", Systemite offers the SystemWeaver systems development concept, which includes software products as well as consulting services.

The SystemWeaver concept is addressing the three C's: Consistency, Correctness and Completeness (CCC), in all levels of abstraction of a system. We see this project as one important step towards formalized architectures and automated CCC-verification.