Planning for Component-based Configurations*

Gail Haddock

University of Texas - Arlington
Department of Computer Science and Engineering
416 Yates, Arlington, Texas 76019
haddock@csr.uta.edu

Abstract

The Scenario-based Engineering Process (SEP) is a novel approach to developing complex systems (Haddock & Harbison 1994). SEP builds new application systems through a selection process that groups primitive components into application specific components. The selection of primitive components and the construction of interfaces among components in an application system is currently a tedious manual undertaking. The automation of this process will require a configuration system that can support the complex interactions of the components, the dynamic requirements of users, and the capabilities of providing multiple viewpoints and managing extensive domains.

The University of Michigan Procedural Reasoning System, UM-PRS (Lee et al. 1993), is a reactive reasoning and planning system based on PRS (Georgeff & Lansky, 1990). UM-PRS is currently being used in the autonomous vehicle domain. Its ability to continually consider the real-time dynamic environment and access plans accordingly fits well into military applications, where plans have already been generated in the form of standing operating procedures and reactions to the quickly changing environment are paramount. Our SEP domain does not require the hard real-time speed of a reactionary system. However, much of the UM-PRS architecture maps readily to the configuration problem in the SEP domain.

The Scenario-based Engineering Procedural Reasoning System, SEPRS, will use the architecture of UM-PRS to implement a configuration system for SEP. Primitive components will take the place of plans and will be selected according to the application requirements and the application architecture in progress.

The interpreter will use the application requirements as goals to satisfy by accessing the primitive components. Components previously selected for an application architecture will be in the in-process area. They are accessed by the interpreter to determine which goals are not yet satisfied. The interpreter will activate relevant primitive components that are maintained by the intention structure. The intention structure will release the chosen primitive component to the component integrator. The component integrator will employ Adaptive Semantic Language techniques (Hannon 1994) to build the interfaces and messages necessary for adding the primitive component to the application architecture. The grouping of primitive components into components remains a manual task, as this grouping can be done from a variety of viewpoints. For example, some groupings may be done solely for marketing purposes. The environment area of UM-PRS then becomes our system engineer. The system engineer's modifications are added back to SEPRS through a component monitor, who sends the component determinations to the in-progress area, thus completing the cycle.

Since requirements are continually accessed by the interpreter, user modifications can be interjected at any point in the architecture creation cycle. These modifications may immediately cause primitive components to be deselected and their interfaces disconnected, which may then require an extensive reconfiguration of the architecture. SEPRS also supports the expansion of primitive components. As new technologies are invented that result in new components, those components can be added to the system.

We are building a configuration system, SEPRS, for component-based architecture methodologies by adapting the UM-PRS reactive planning system. We expect it to fit well in our system engineering environment that includes scenario modeling, object-oriented analysis and design, and simulation systems.

*This research has been partially supported by the National Science Foundation, the National Center for Manufacturing Sciences, the Advanced Research Projects Agency and the State of Texas.

References

Georgeff, M. and Lansky, A. 1990. Reactive Reasoning and Planning. Allen, J., Hendler, J., and Tate, A., eds., *Readings in Planning*, San Mateo, Ca: Morgan Kaufmann.

Haddock, G., and Harbison, K. 1994. From Scenarios to Domain Models; Processes and Representations. In Proceeding of Knowledge-based Artificial Intelligence Systems in Aerospace and Industry. Bellingham, Wa: International Society for Optical Engineering.

Hannon, C. 1994. An n-Towers Model for the Knowledge Representation of an Adaptive Semantic Language. Master's thesis, Dept. of Computer Science Engineering, University of Texas at Arlington.

Lee, J., Huber, M., Durfee, E., Kenny, P. 1993. UM-PRS: An Implementation of the Procedural Reasoning System. Technical Report, University of Michigan.