Principal Investigators

Spencer Rugaber (spencer@cc.gatech.edu) - Point of Contact
Gregory Abowd (Gregory.Abowd@cc.gatech.edu)
Ashok Goel (ashok.goel@cc.gatech.edu)
Michael McCracken (mike@cc.gatech.edu)
Melody Moore (melody@cc.gatech.edu)
Colin Potts (potts@cc.gatech.edu)
Linda Wills (linda.wills@ee.gatech.edu)

also used to detect and model components of the user interface in order to support evolution across environments. This aspect of MORALE extends previous work on cliche recognition (detection of commonly occurring patterns in code) and interleaving detection (determination of situations in which code implements more than one design goal or requirement).

The fourth aspect is the generative component which utilizes the results of the three analytical components to produce an abstract architectural design for the new system. The paradigm for this component is that the system is viewed as an abstract "device" and thus the development of the new architure is viewed as a redesign process. Consequently this work is able to draw on a variety of past work on model-based redesign.

Research Products

Method descriptions

Tutorial and reference descriptions of the constituent MORALE method are currently being created as a set of guidebooks. Included are the following topics:

- Inquiry Cycle - support for converting a mission statement into a network of goals and for elaborating and refining these goals into operational requirements that describe the expected behavior of the enhanced version of a software system.
- Software Architecture Analysis Method (SAAM) - method for assessing impact on software quality of a set of scenarios with respect to an architecture
- Synchronized Refinement - method for systematic program analysis and application model construction.
- SBF - tutorial and reference manual for Structure / Behavior / Function system representation notation.
- MORPH - a technique for reengineering user interfaces (UI), including a rule set for detecting UI constructs in source code.

Tools

Prototype development of the following tools is underway:

- ISVIS (Interaction Scenario Visualizer) - graphical tool that supports browsing and analysis of execution event traces
- Scenic - Hierarchical goal representation editor/viewer for scenarios.
- SAAM Tool - Graphical tool to display and analyze software architectures.
- Sirrine - Intelligent redesign tool that uses SBF models to support architectural evolution.
- MORPH Toolset - Automated support for detection, representation, and transformation of user interface components.

Case Study

We plan to report on the results of applying the MORALE methodology to an existing legacy system. Included in this study are the following efforts:

- Case Study Schema: preconditions, structure, procedure, product definition, and evaluation metrics for case studies of legacy system evolution.
- Assessment of SBF as an architectural description language for MORALE.
- Integration of individual tools to form composite MORALE toolset.

MORALE
Mission Oriented Architectural Legacy Evolution

Georgia Institute of Technology
[ARPA Order No. D882]

Program Manager: John Salasin
Research Area: Evolutionary Design of Complex Software (EDCS)

Objective

The Mission Oriented Architectural Legacy Evolution (MORALE)[1] project supports evolution of software that is driven by a change in its purpose or mission. The key to the approach is exploiting the synergy between requirements analysis, architecture, and reverse engineering. Effective system evolution requires understanding both the way that an existing system accomplishes its tasks and the mission-oriented rationale for any changes that feed its evolution. Understanding the architecture of an existing system aids in predicting the impact of changes mandated by new mission-oriented requirements. Requirements analysis techniques, in turn, suggest what concepts are most useful in understanding how an existing system works and how it should evolve. The MORALE suite of tools and techniques leverages this synergy by growing a common model of the architecture for multiple versions of a system. The common model is a basis for assessing the effects of proposed changes and the extent to which legacy code can be reused.

Approach

Mission Oriented Architectural Legacy Evolution (MORALE) synthesizes, refines, and applies several existing software engineering methods. The first is a mission-directed requirements determination process that, through a systematic use of inquiry and refinement, turns mission-oriented goals into specifications of the desired behaviors of architectural components. This aspect of MORALE extends previous work on Inquiry-based Requirements Analysis and the use of schematic scenarios to elicit and formulate requirements.

The second aspect of MORALE is an architectural assessment process that ascertains the impact of new requirements on an existing system's architecture. It analyzes architectural descriptions in order to determine how well they satisfy a large class of quality criteria. The work extends previous work on the Software Architecture Analysis Method to perform this analysis.

The third aspect is a reverse engineering process for extracting architectural information from an existing system. Beyond the traditional structural analyses provided by existing commercial and research tools, the reverse engineering process extract behavioral information from dynamic analyses of the message flow among architectural components. Static and dynamic analysis are

1. Effort sponsored by the Defense Advanced Research Projects Agency, and Rome Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-96-2-0229. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency, Rome Laboratory, or the U.S. Government.