Program Description
ISVis is an architectural visualization and extraction tool for analyzing legacy software in order to:

- Acquire behavioral understanding
- Recover original design rationale
- Locate change sites
- Validate design and implementation

The ISVis main screen allows the analyst to define architectural models composed from combinations of files, classes and functions. This helps build higher level abstractions to represent major elements of a software architecture. The scenario view allows the grouping of interactions into scenarios to provide higher level behavioral abstractions of dynamic program execution.

Program Features
ISVis has a wide range of features including:

- Analysis of program event traces numbering over 1,000,000 events
- Simultaneous analysis of multiple traces for the same program
- Multiple views including actor and interaction lists and relationships, scenarios, and source code (via XEmacs)
- Informational Mural visualization techniques to portray global overviews of scenarios
- Abstraction of actors through containment hierarchies and analyst-defined components
- Selective filtering of individual or multiple occurrences of a particular interaction
- Definition of higher-level scenarios comprising repeated sub-scenes
- Analyst-specified interaction patterns, including regular expression wildcards for actors
- Use of patterns for locating the same or similar scenarios from interaction lists
- Saving and restoring of analysis sessions
- Connection to an ACMEServer and export of recovered descriptions the ACME Architectural Description Language.

MORALE
ISVis supports the Mission Oriented Architectural Legacy Evolution (MORALE) reengineering process. The goal of the MORALE project is to facilitate the evolution of legacy software systems. Facilitation takes the form of improved quality by requirements validation, reduced risk via architectural evaluation and assessment, and increased productivity from maintenance and access to design rationale and from high level reuse of architectural components.
ISVis Sample Screens

The screen shots below show the main ISVis screen in the background and the scenario view in the foreground. This screen display was captured during the analysis of the Mosaic web browser and displays several of the key features of ISVis. In the Scenario View, the yellow line labeled *image resolve* is a user-defined scenario which provides an abstraction of a grouping of program events (represented by the gray lines in the display). The mural provides a visualization of the scenarios by showing location and frequency information of the scenario occurrences. In this case, the thick band of yellow in the center of the mural depicts a high concentration of this particular scenario that occurred roughly in the middle of the Mosaic program execution. The small vertical bars in the main view also provide mural visualization for all the items in the various list boxes. The red box around the HTMLParse actor shows it has the primary focus. Secondary items of interest are outlined in orange as can be seen in the main view. These features work together to provide powerful tools for program understanding.

Availability

ISVis version 1.1. The ISVis distribution includes an online user's manual, tutorial, and source code with makefiles. It can be downloaded from the MORALE tools page at http://www.cc.gatech.edu/morale/tools/ or from the EDCS Tool Page. ISVis currently analyzes legacy code written in either C or C++ and compiled with Solaris debugging information.

Contact Information

morale-support@cc.gatech.edu
http://www.cc.gatech.edu/morale

System Requirements

- SUN Solaris 2.5 (and its compiler)
- RogueWave tools.h++ (comes with Solaris WorkBench)
- Motif 1.2
- X11R5
- Xemacs
- ACMEServer (if ACME export services desired)

Sponsorship