

Analysis and Advanced Visualization of Software

The Problem:

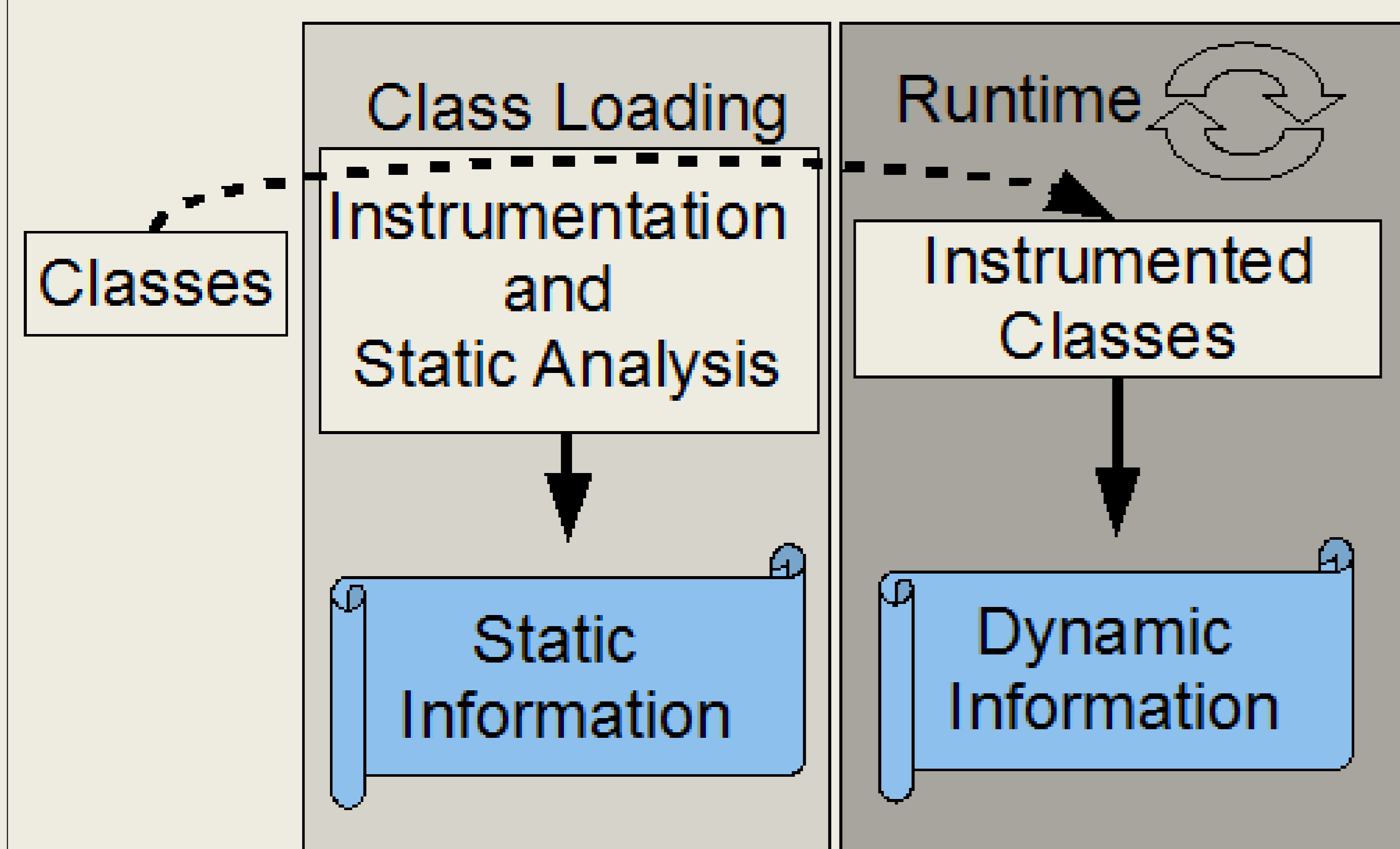
Software is complex and contains large amounts of information. This makes understanding its design, functionality and behavior a difficult and time-consuming task.

A Solution:

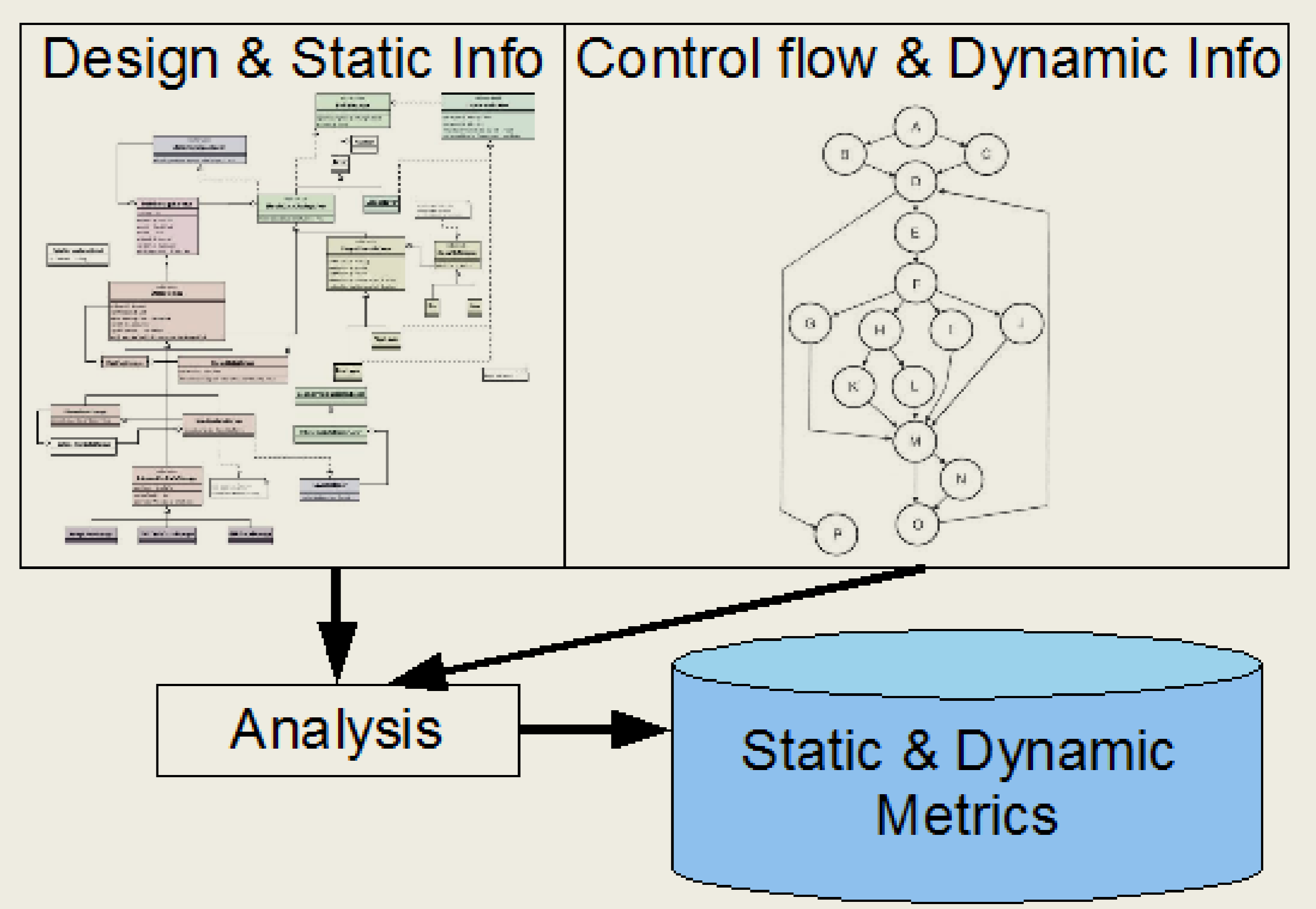
Use analysis techniques to extract and compute useful metrics on the software system. Then represent these metrics through suitable abstractions and metaphors to ease the understanding of the software.

Tracing Software with VITRIL JBinTrace:

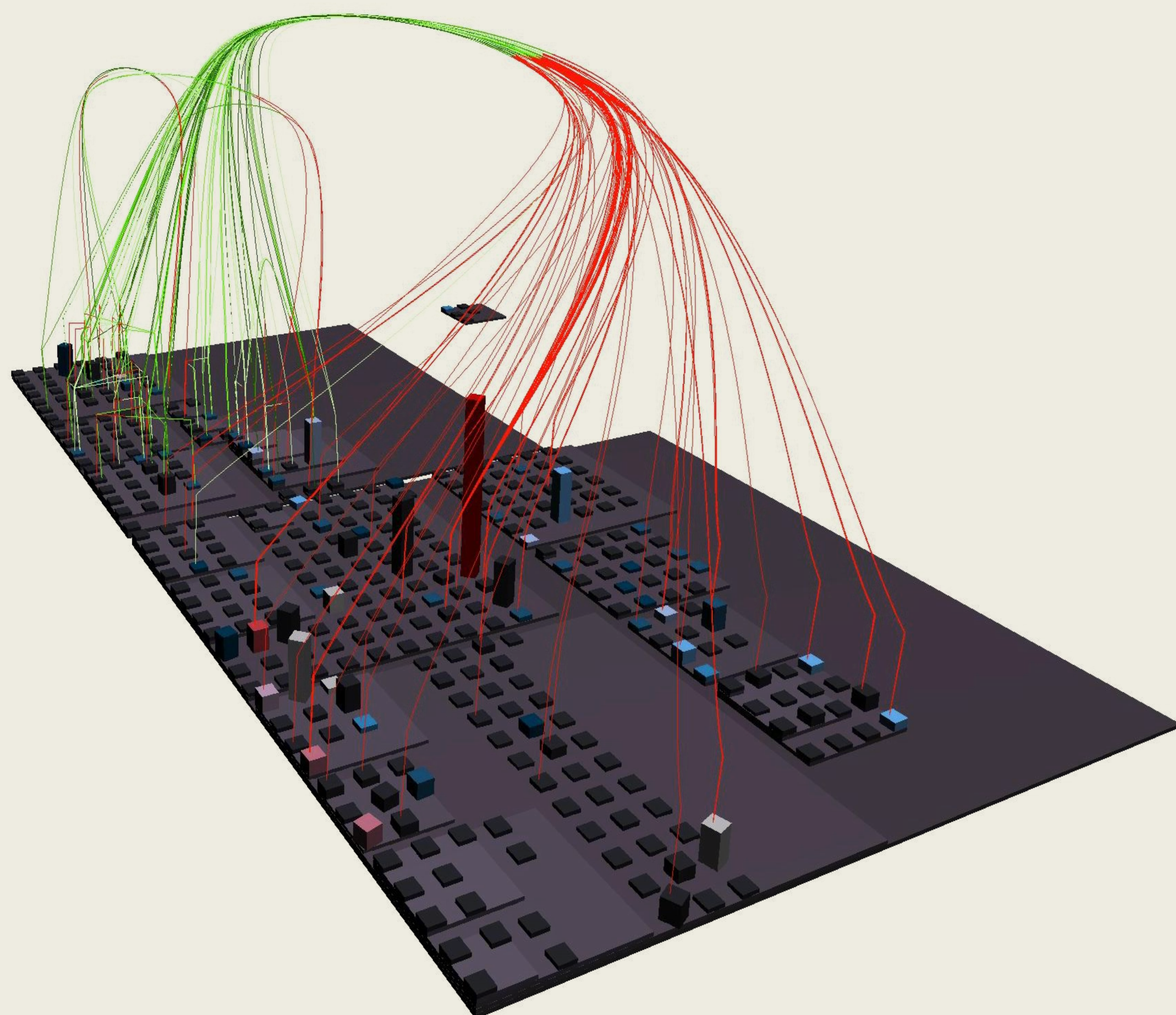
Instrument bytecode to collect static as well as dynamic information on the software.



Analyzing Software with VITRIL Analyzer:



Visualizing Software with VITRIL Visualizer:



Understanding Software Functioning:

- Static relations between software components.
- Dynamic relations between software components.
- Evolution of these dynamic relations during runtime.
- Impact of each component on the whole system.

Purpose :

- Design and Source code optimizations.
- Improve maintenance process.
- Study the habits of object-oriented developers.

Current Work:

- We perform tracing with VITRIL JBinTrace and analysis with VITRIL Analyzer on many pieces of software to get useful static and dynamic information.
- We study how software was designed and its behavior at runtime using VITRIL Visualizer.
- We use new interaction and display technologies to ease manipulation and understanding of the visualization.