



Analysis and Advanced Visualization of Software

Pierre Caserta, Olivier Zendra, Damien Bodénès

► To cite this version:

Pierre Caserta, Olivier Zendra, Damien Bodénès. Analysis and Advanced Visualization of Software. ACACES 2011 Seventh International Summer School on Advanced Computer Architecture and Compilation for High-Performance and Embedded Systems, Jul 2011, Fiuggi, Italy. inria-00612601v1

HAL Id: inria-00612601

<https://inria.hal.science/inria-00612601v1>

Submitted on 29 Jul 2011 (v1), last revised 21 Aug 2011 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analysis and Advanced Visualization of Software

Pierre Caserta^{*,1}

** Nancy-University INPL / LORIA, France.*

ABSTRACT

Software is complex and hard to analyze. We aim at collecting information using dynamic and static analysis of the software and use advanced visualization techniques to ease their understanding. Thus the comprehension of the software functioning.

KEYWORDS: Software Analysis; Software Visualization; Software Comprehension

My research consists in apprehending a software in its entirety by visually considering static and dynamic aspects with a particular focus on dynamic analysis. We indeed think that bringing new solutions to analyze and understand software and its runtime will allow faster and better development and maintenance. Performance may also be improved with an improved understanding of what happens at runtime.

Software runtime is indeed complex and hard to analyze. Moreover, a huge quantity of information can be collected during dynamic analysis.

To be able to fully understand the meaning of the huge quantity of data extracted from the dynamic and static analysis of the software, we rely on visualization techniques. Lately, the focus of research in the software visualization domain has been visualizing the static aspects. We thus think that the domain of software visualization of the dynamic aspects still requires substantial research.

Moreover we definitely believe that advanced visualization techniques relying on total immersion can ease the global understanding of data. This kind of technologies are becoming accessible and we think that some of them can be used for software development and maintenance.

Our methodology is as follows.

First we collect a large amount of fine-grained dynamic data on many Java programs. To this end, we designed and developed the VITRAIL Execution Analyzer, that performs a very fine-grained analysis of the runtime of Java software at basic block level and combines it with static information extracted from the source code of the analyzed software. For all we know, no other work has done such a fine-grained kind of profiling before.

Then we interpret the data using VITRAIL Visualizer, an advanced visualization tool we designed and developed. It is designed to visualize dynamic and static data and ease their understanding, by displaying information pertaining to the software according to a well-know metaphor.

¹E-mail: pierre.caserta@loria.fr