Research Review 2017

Guided Architecture Trade Space Exploration of Safety Critical Software Systems

Sam Procter, Architecture Researcher

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

DM17-0784

Research Overview

Engineering critical systems is difficult because it is impossible to fully evaluate all possible options.

Individual design choices often have *far reaching* impacts across the system.

As systems become increasingly complex, understanding these impacts becomes both more difficult and more important.

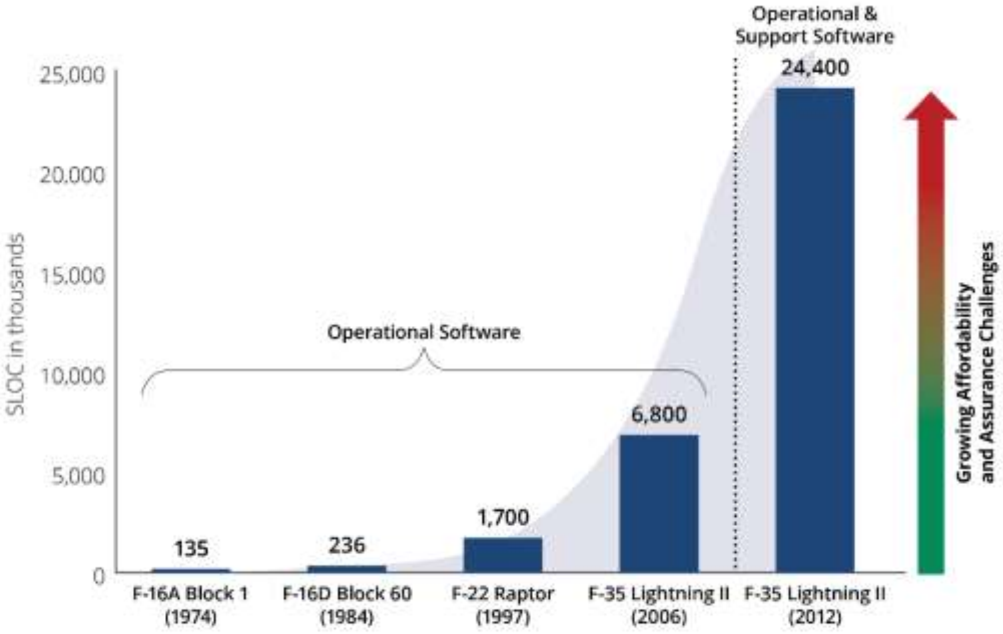
We build on past SEI successful architecture modeling research to partially automate exploration of a system's design trade space.

This automation doesn't replace the system designer's expertise, rather it *augments* it by generating a huge number of options and analyzing them for what the designer cares about.

System designers are able to *guide* the exploration using a visual steering tool.

This project's approach is to integrate SEI's architecture modeling language and tools with an existing trade space exploration tool.

Why do We Need Something Different?



Software as % of total system cost

1997: 45% → 2010: 66% → 2024: 88%

Graphic: Hagan/Sorenson, "Delivering Military Software Affordably," *Defense AT&L*, Mar-Apr 2013

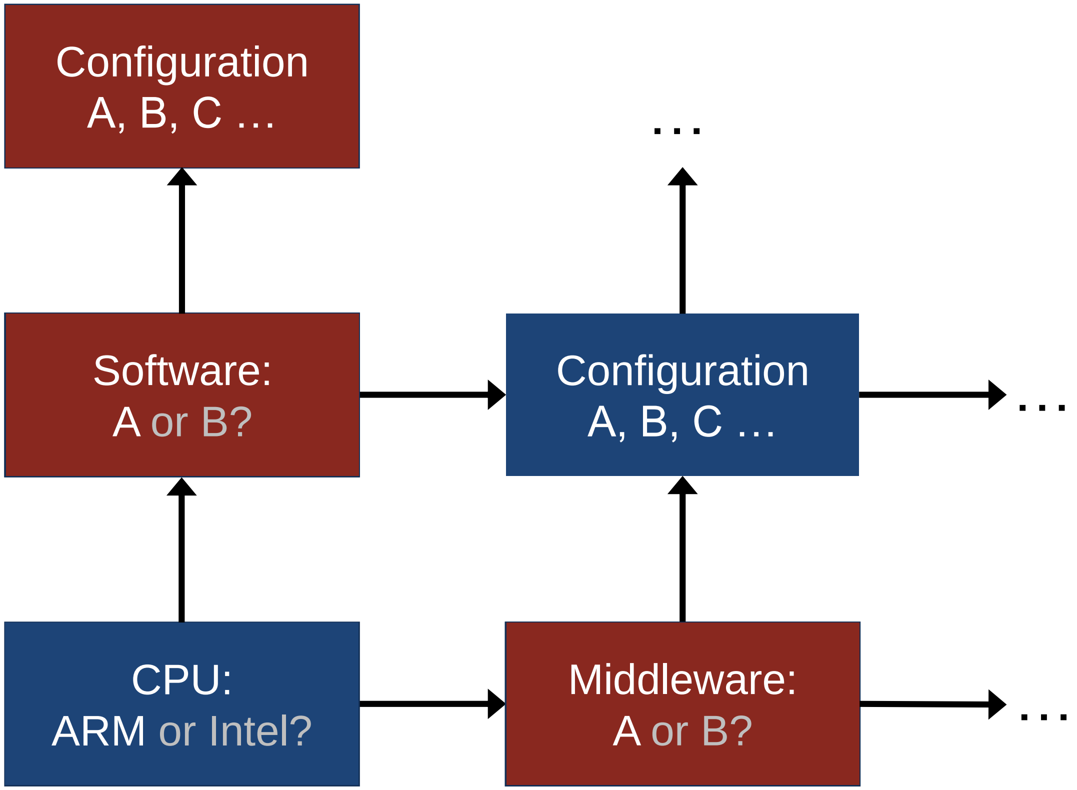
The cost of developing software-driven systems is rising rapidly.

Existing SEI work includes the *Architecture Analysis and Design Language* (AADL)

- Allows designers to build high-fidelity system models
- Then analyze them for various quality attributes using tooling (OSATE)

This work is an enabling technology for a system design paradigm shift to *design-by-shopping*

An Abstract View of System Design



Broadly speaking, there are two considerations in system design:

- Ensuring the system is buildable (i.e., no conflicts)
- Ensuring necessary *quality attributes* are met
  - Cost
  - Power Consumption
  - Etc.

Component *interactions* make design challenging

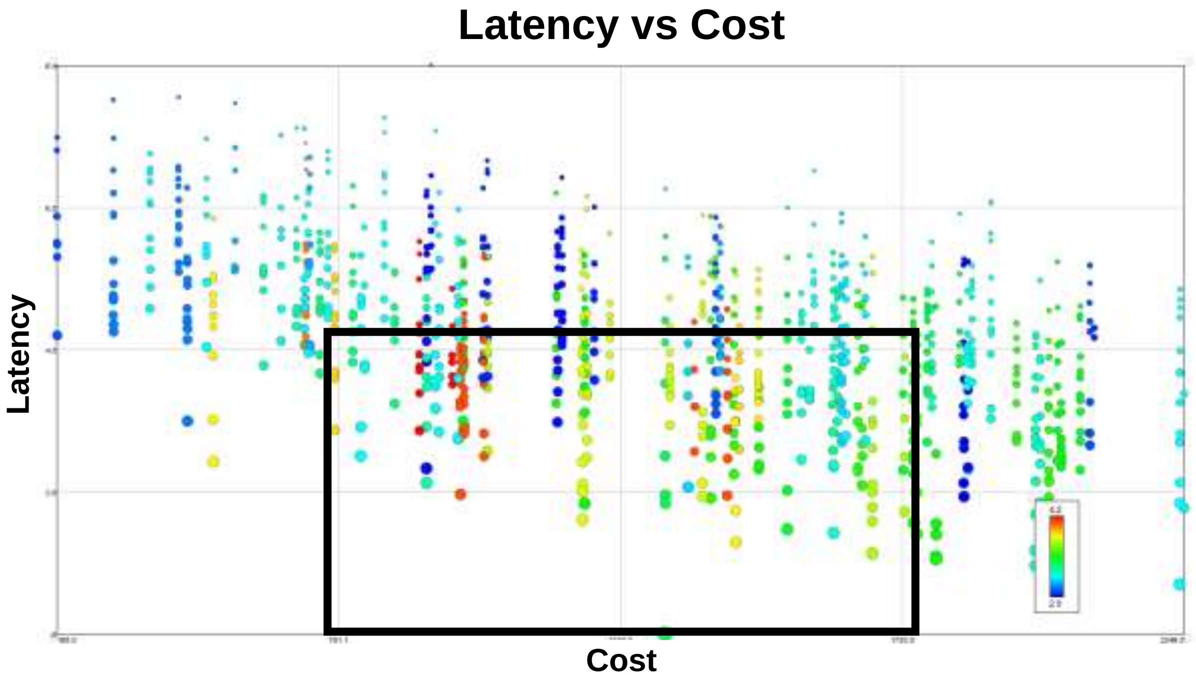
GATSE Project Tasks

1. Extend existing architecture modeling language (SEI’s AADL) to encode component choices and their interactions

2. Extend existing architecture modeling tooling (SEI’s OSATE) to automatically analyze the resulting system for cost, weight, performance, etc.

3. Enable trade space visualizer (Penn State’s ATSV) to automatically select valid components and configurations, visually display analysis results, and enable analyst shopping

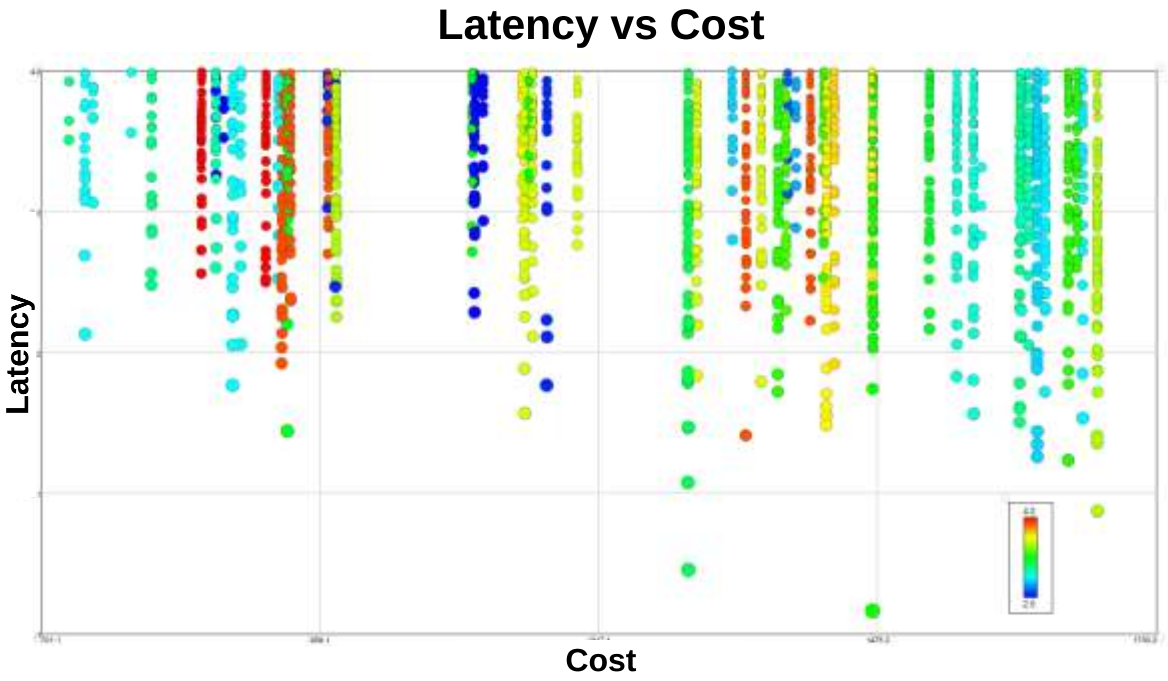
Design by Shopping in GATSE



At the outset, a system’s design space might be essentially a spread out “cloud” of points – each representing a possible system architecture

- Designers can “focus” on specific areas – this restricts the parameters ATSV will send to OSATE

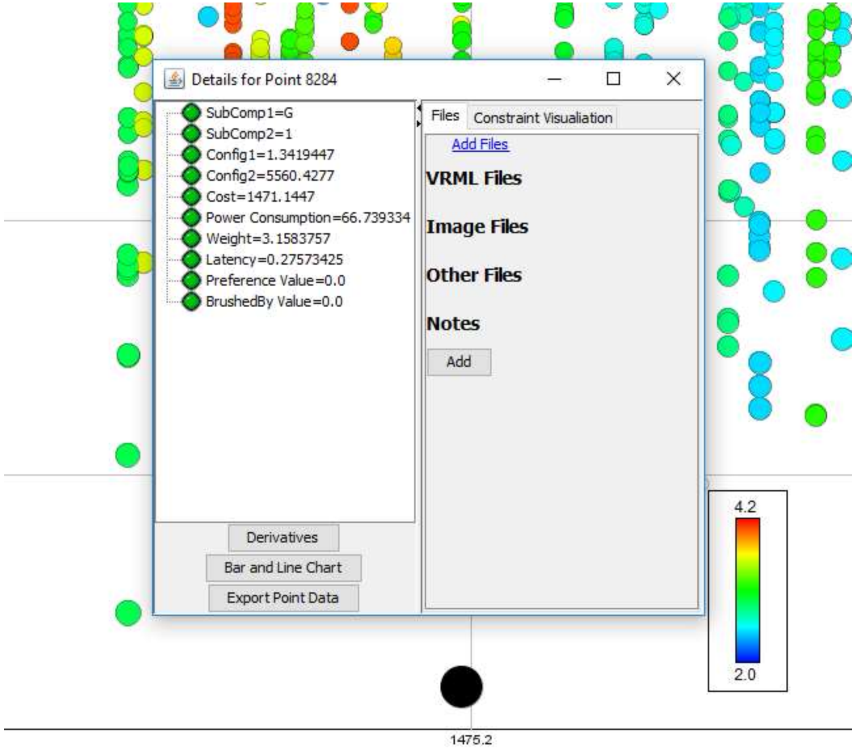
Design by Shopping in GATSE



At the outset, a system’s design space might be essentially a spread out “cloud” of points – each representing a possible system architecture

- Designers can “focus” on specific areas – this restricts the parameters ATSV will send to OSATE

Design by Shopping in GATSE



Details for Point 8284

SubComp1=G
SubComp2=1
Config1=1.3419447
Config2=5560.4277
Cost=1471.1447
Power Consumption=66.739334
Weight=3.1583757
Latency=0.27573425
Preference Value=0.0
BrushedBy Value=0.0

At the outset, a system’s design space might be essentially a spread out “cloud” of points – each representing a possible system architecture

- Designers can “focus” on specific areas – this restricts the parameters ATSV will send to OSATE
- Once a suitable architecture is found, the exact configuration is shown.

Artifact Availability

No publications (yet!) – pending more complete experimental analysis.

Code and user documentation are available on GitHub:

- https://github.com/osate/osate2-gtse

Tooling is also directly installable into OSATE via experimental update site:

- http://aadl.info/aadl/osate/experimental/

Future Work

Bottom Line: This project connects a number of existing technologies to enable designers to visually explore a system's trade space.

Future Work: As new analyses are added, they will continue to be integrated and automated.

Long Term: Since we can use *any* quantifiable analysis, advancing the state-of-the-art will involve quantifying traditionally qualitative measures, like safety and security.

Contact Information

Point of Contact

Sam Procter

Architecture Researcher

sprocter@sei.cmu.edu

Contributors

Lutz Wrage

Peter Feiler