ISR Institute for Software Research
University of California, Irvine

Reference Architectures: A Research Perspective

Eric Dashofy – John Georgas – Scott Hendrickson
Institute for Software Research
University of California, Irvine

http://www.isr.uci.edu/

Obligatory Definitions Slide

A software system's ***architecture*** is the set of principal design decisions about the system.

- Examples:
  - Structural decisions
  - Concurrency decisions
  - Management decisions

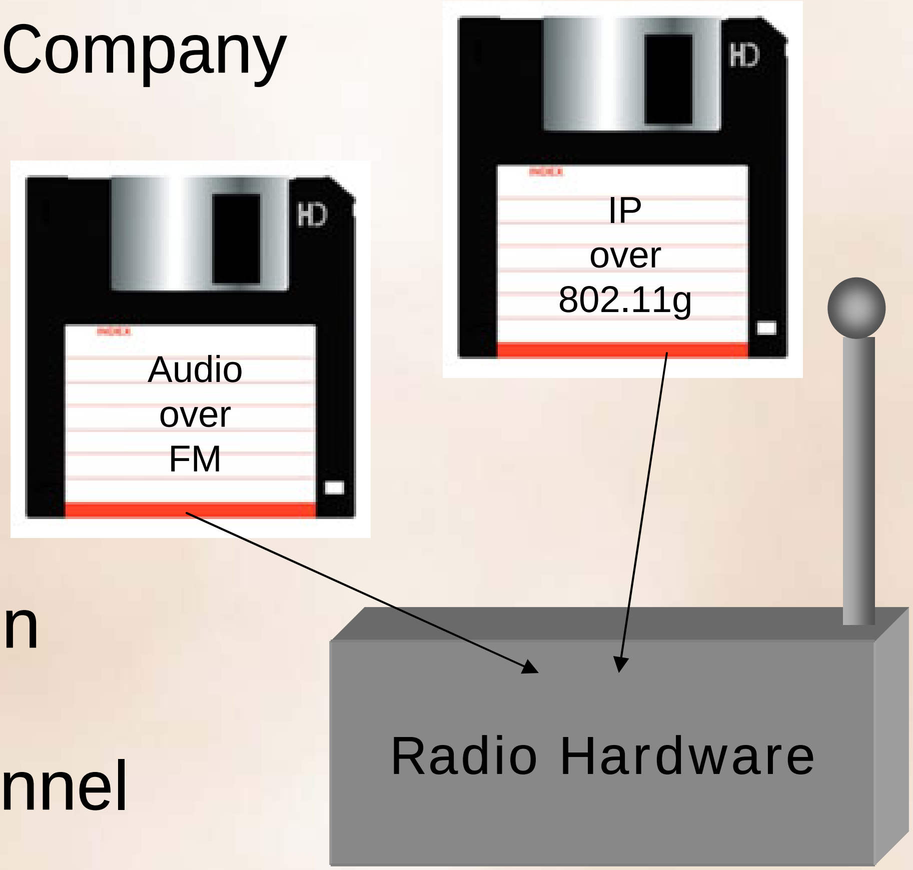
A ***reference architecture*** is the set of principal design decisions that are simultaneously applicable to multiple related systems, typically within an application domain, with explicitly defined points of variation.

Empirical Data on the use of Reference Architectures

- Some successes...
  - *Koala* Component Model
    - Increase design understandability, architectural longevity, code quality.
    - Success in finding implementation violations of architectural principles and increase in reuse.
  - *tss*
    - Reusable building blocks resulting in 24% to 80% reuse across related products.
  - OMG/CCSDS (Consultative Committee for Space Data Systems)
    - Developing reference architecture for control, services, etc.
    - Benefits for (OTS) reuse, simulation/analysis, interoperability
- But, do it with care...
  - SEI study on deficient reference architectures.

Domain: Software Defined Radio (SDR)

- Work with The Boeing Company
- Increase capabilities of radios
- General-purpose hardware (DSP, GPP)
- Special purpose software per-application
  - Called “waveforms”
- One waveform per channel



Audio over FM

IP over 802.11g

Radio Hardware

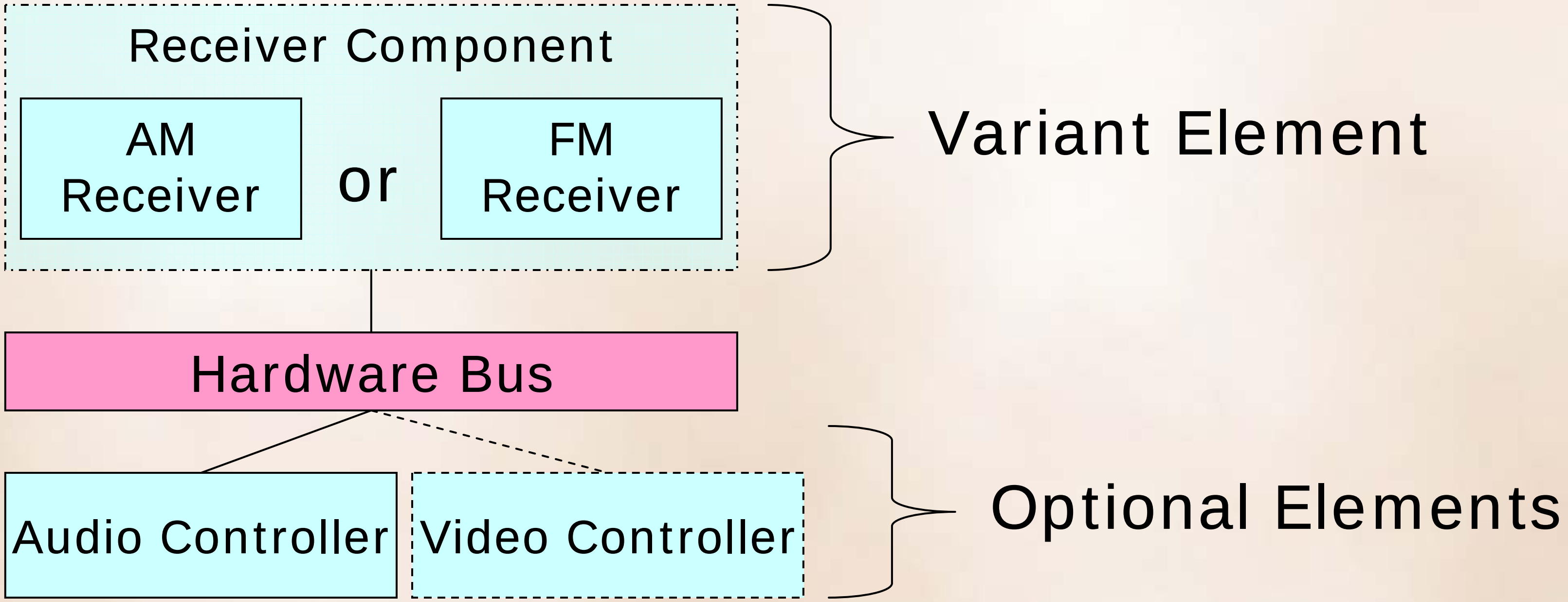
The Official Reference Architecture

- The “software communications architecture” (SCA)
- Developed by JTRS program
- CORBA-based
  - Mandated use of a CORBA ORB for component communication
  - Primarily a detailed set of CORBA interfaces for components that might be part of an SDR
- Declared goals
  - Component portability
  - Component reuse
  - Reduced cost via leveraging commercial standards

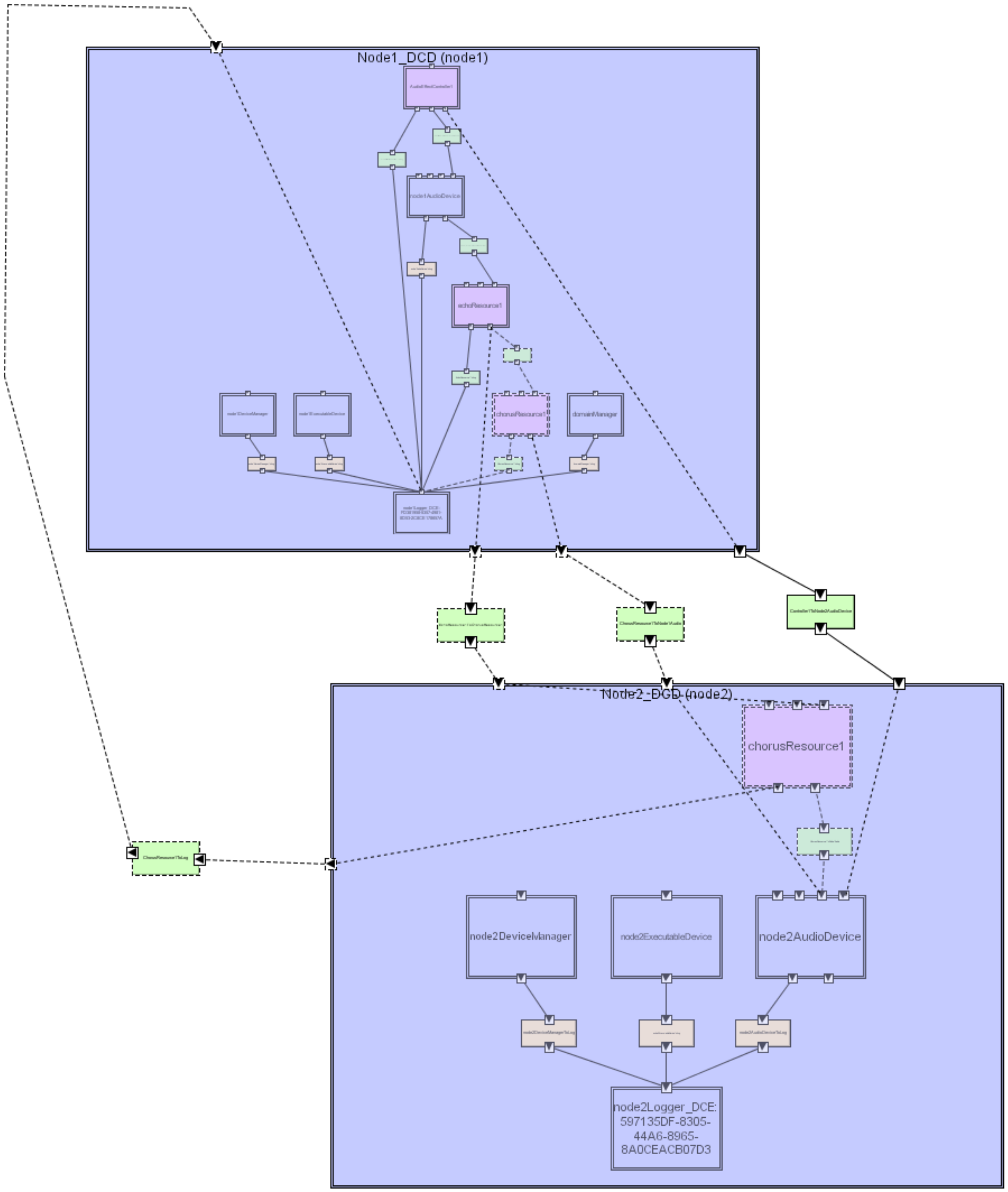
The Adequacy of the SCA

- Good for portability and reusability
- Questionable for many other important qualities
    - "Doability"—what guidance is provided on how to actually build a working radio?
        - What proof is there that this specification can even be used to create a working radio?
    - Efficiency—is CORBA really the right choice here?
    - Correctness—what kind of analyses are possible?
    - Deployability, scalabilty, compatibility...

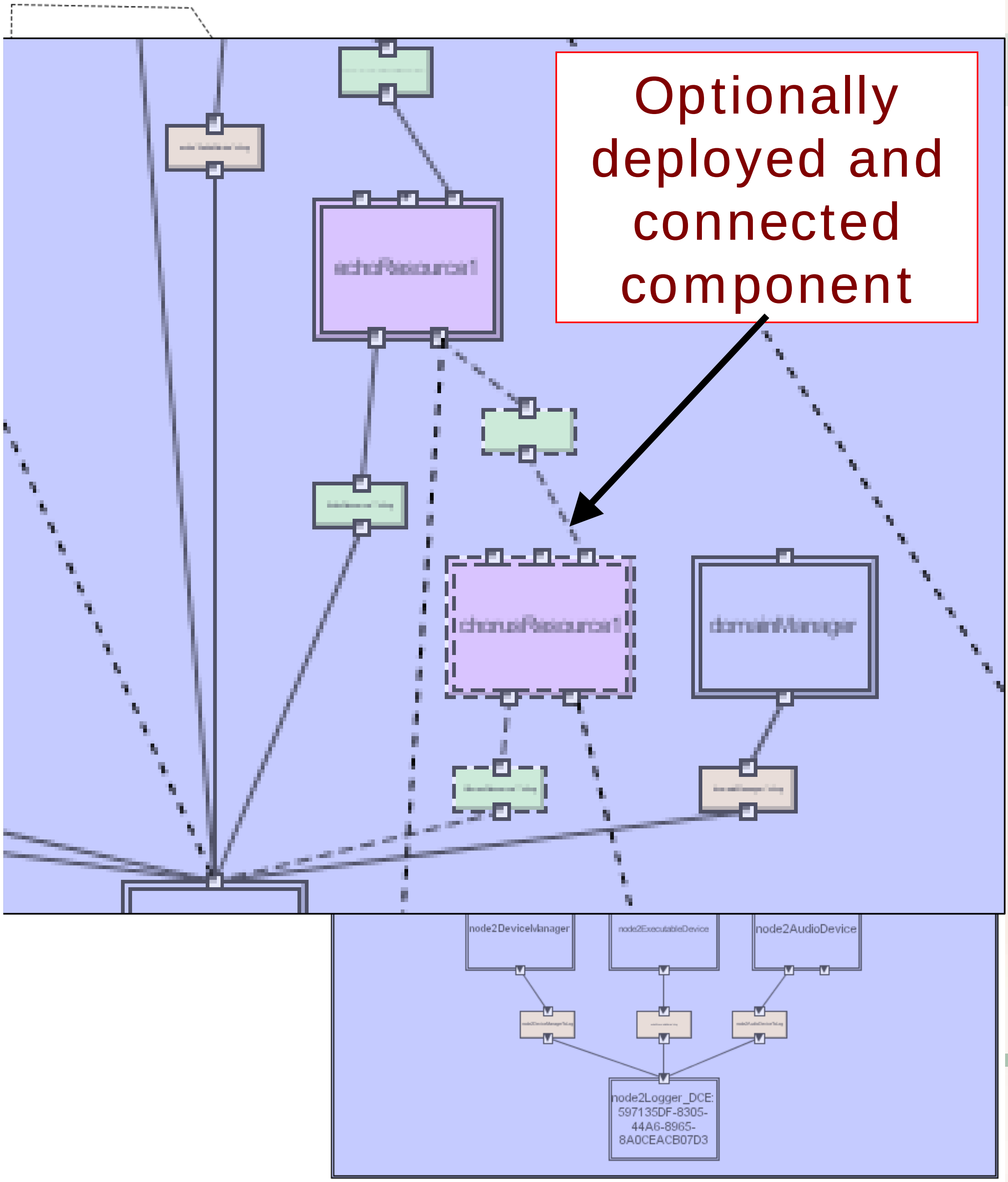
More Power from the Research Community: Product-line architecture modeling



- Variant elements are always included but their type varies
- Optional elements may or may not be included
- Elements may be both optional and variant



- Modeling multiple deployments
  - Graphically
  - Simultaneously
- Identifying latent deficiencies
- Providing stronger artifacts for review & communication



Optionally deployed and connected component

Benefits:

- Explicit modeling using a real ADL gives users a concrete picture of their software
- Product-line facilities let users visually see and manipulate variations

Lessons from the Acquisition Perspective

- Carefully understand what your reference architecture is good for (and not good for)
- Optimize for the most important qualities
  - Portability doesn’t matter so much if you can’t make it work right
- Derive the RA from existing working systems if possible
  - If not, then lengthen your cycle time to validate the RA and permit for serious revisions & extensions
- Keep realistic expectations about commercial technologies
  - What are you really buying? At what cost?
  - Are there research technologies that can be developed?
- Give implementers/developers a serious stake in the future of the architecture
- Consider investing in architecture-centric tools

References

- OMG Space Domain Task Force
  - http://www.omg.org/space/
- Rob van Ommering. "Software Reuse in Product Populations".
- A.V. Fioukov, E.M. Eskenazi, D.K. Hammer, M.R.V. Chaudron. "Evaluation of Static Properties for Component-Based Architectures"
- Frank van der Linden and Jürgen K. Müller: "Creating Architectures with Building Blocks".
- Brian P. Gallagher. "Using the Architecture Tradeoff Analysis Method to Evaluate a Reference Architecture: A Case Study."
  - http://www.sei.cmu.edu/publications/documents/00.reports/00tn007.html
- ArchStudio 3 Website:
  - http://www.isr.uci.edu/projects/archstudio/
- xADL 2.0 Website
  - http://www.isr.uci.edu/projects/xarchuci/