What do we mean by Architecture?

Mark W. Maier, Ph.D.
The Aerospace Corporation
15049 Conference Center Dr.
Chantilly, VA 20121 USA
mark.w.maier@aero.org

Outline of Talk

- Setting the stage: Criteria for architecture
  - » References and background
    - – The Art of Systems Architecting, Second Edition, Maier and Rechtin, CRC Press, 2000
    - – IEEE 1471 Recommended Practice for Architectural Description, published standard and associated record of deliberations
- Civil architecture as a metaphor for systems practice
  - » What does the metaphor really say?
- Systems Architecting ala Rechtin and Maier
  - » Importance
  - » Attributes and scope
  - » Examples of methods

Criteria for an Answer

- The groups pursuing “Software Architecture” or “Systems Architecture” or “Enterprise Architecture” don’t agree what architecture is
- Does this matter?
  - » It is certainly an impediment to mutual understanding, and possibly an impediment to developing good practices
- What are criteria for a practical and useful definition of the practice of architecture on complex systems?
  - » It should be reasonably clear what things are “architectural” and what things are not
  - » Doing the things that are architectural should be important (strongly correlated with success/failure)
  - » Good architectural practices should be different from good XXX practices (XXX = engineering, project management, etc.)
  - » It should related to the established practice of architecture (civil)

What "Architecture" Evokes

- Architect, Architecture, and Architecting come with baggage
- Architecture suggests overall structure, general characteristics of design, attention to aesthetics
  - These are useful ideas, although the notion of aesthetics must be taken generally, to mean attention to soft criteria
  - Consider, can reliability or security be a "soft" criteria?
- If we are going to say "architecture," lets mean something by it

Architecture as a Metaphor

- The architecture metaphor is powerful in understanding good practices
- The architect works for the client and with the builder
  - The architect is compensated separately from the building. Building yields no profit to the architect
- You expect the architect to help you develop requirements
  - An architect who needs complete and consistent requirements to begin gets fired
  - Your builder requirements are an outgrowth of joint exploration of both problem (what you want) and solution (what you can have)
- The architects products are abstracted designs
  - Floorplans, elevations, cost estimates, etc. are not complete building plans, but they are on the path to complete building plans
  - Building plans come after the architecture description is settled

Architecture as Metaphor - Continued

- Architecture descriptions and the architecture are different
  - The floorplan is not *the architecture*, nor is the elevation, nor is the cost estimate
  - They represent different views of the architecture, and abstract object
  - Each contains different information, but also overlaps with the other views
- A good architecture representation is not just physical structure
  - Would you start building without a cost estimate? Would you even hire a general contractor?
- Architectures and architecture descriptions have very different standards
  - Blueprint standards vs local architecture regulations
- Architectures are cheap, at least compared to buildings
  - Explore several to lower risk, architects show alternative concepts

Three Systems Paradigms

Characteristic	Architecting	A & E	Engineering
Situation/Goals	Ill-Structured	Constrained	Understood
	Satisfaction	Compliance	Optimization
Methods	Heuristics	⟷	Equations
	Synthesis	⟷	Analysis
	Art and Science	Art **and** Science	**Science** and Art
Interfaces	Focus on "Mis-Fits"	Critical	Completeness
System Integrity Maintained Through	"Single Mind"	Clear Objectives	Disciplined Methodology and Process
Management Issues	Working for Client	Working with Client	Working for Builder
	Conceptualization and Certification	Whole Waterfall	Meeting Project Requirements
	Confidentiality	Conflict of Interest	Profit versus Cost

What is Architecture?

Architecture: A fundamental or unifying structure of a thing (Dictionary)
Note: 5000 years of practice and it still isn't completely clear
Note: Architecture as property of a thing, not a thing itself

Systems architecting is that part of systems engineering most concerned with purpose determination, concept formulation, structuring, and certification for use

- Purpose: Why does somebody want the system?
- Concept and Structure: The overall structural features, in any set of views
- Certification: Determination of fitness for use

Formal Definitions

- Architecture, of a system
  - The fundamental and unifying system structure defined in terms of system elements, interfaces, processes, constraints, and behaviors (INCOSE SAWG)
  - The fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution (IEEE P1471 Proposed)
  - The structure of components, their relationships, and the principles and guidelines governing their design and evolution over time (Perry-Garlan, NOT IEEE 610.12)
  - A set of information that defines a systems value, cost, and risk sufficiently for the purposes of the systems sponsor (Maier's rule of thumb)

Thought Experiment

What is the architecture of this building?

What is the architecture of the Internet?

Rechtin's Manifesto: Importance

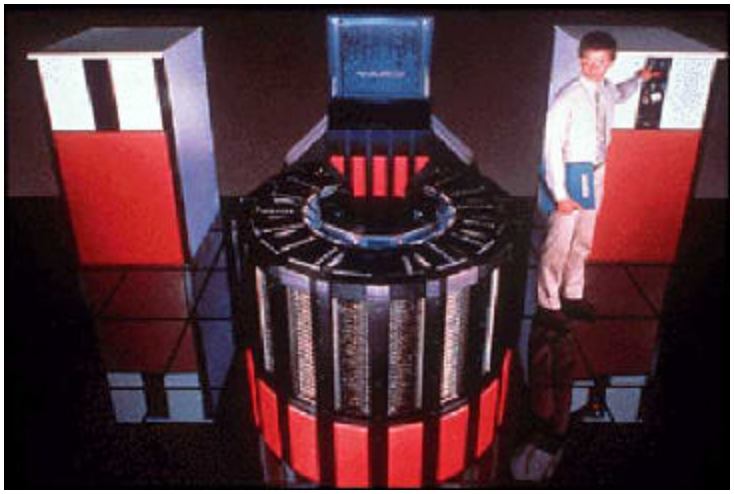
- Successful, unprecedented systems always have an identifiable architect (a small number of people)
- This architect:
  - » Synthesizes a multi-view (requirements and design) system concept which is both satisfactory and feasible
  - » The architect's involvement is strongest at initial concept formation, but continues to maintain conceptual integrity and to certify for client use
  - » The methods of the architect are a fusion of art and science, with heuristics and modeling forming the primary synthesis tools
- Also, the distinction is best established in the most mature disciplines
- Challenge: Name a successful, unprecedented system that does not have an architect

Examples

DC-3
SR-71
Ford Mass Production
Saturn/Apollo
Nuclear Submarine
Polaris
Deep Space Network
IBM 360 OS
Cray Supercomputers
ARPAnet/Internet
Ethernet

Cray SuperComputers

- Seymour Cray was responsible for the fastest computer in the world for roughly 20 years
- The architecture was the vector register supercomputer
- Cray had close client relationships with National Labs, who acted as committed customers for innovations
- He was well-known for taking for moving among problems of highly variable technical depth and producing designs with well balanced performance on real problems



Other good case studies include the Ford mass production system (of the early 20th century), the DC-3, the SR-71, and various standards

Attributes of Effective Architects, and What They Are Not

- Holistic View
  - » Interactive consideration of form and function
  - » Variable technical depth
- Lifecycle Roles
  - » Works from concept, may oversee development, "certifies" for use
  - » Chooses stable intermediate forms in evolutionary, incremental development
- Relationships
  - » Often works directly for the system client, but other relationships are also used
  - » "Third-party-like" relationship is fundamental
- Key Description: Architects develop satisfactory and feasible system concepts, maintain the integrity of those concepts through development, and certify built systems for client use
- Project/Program Managers
  - » The architects emphasis is technical
  - » But, a good duo is much better than either alone
- Lone Inventors or Scientists
  - » They spend a very large amount of time communicating; architectures must be sold and explained and their integrity maintained
  - » Architecting is not a science, but depends on science
- "Top Level Only" Designers
  - » Details are often critical to feasibility and/or risk
- Collaborations
  - » Experience shows the value of a coherent vision

Understanding Current Challenges

Architecting Actual Things

- Collections of loosely interacting systems with distributed authority
  - » Enterprise architecting, systems-of-systems architecting
  - » Metaphor: Urban Planning
- Systems developed and delivered in a discrete project
  - » "One-shot" versus incremental delivery
  - » Low versus high technology
  - » Software versus hardware versus IT integration versus ???
  - » Metaphor: Civil Architecture

Architecture-related Standards

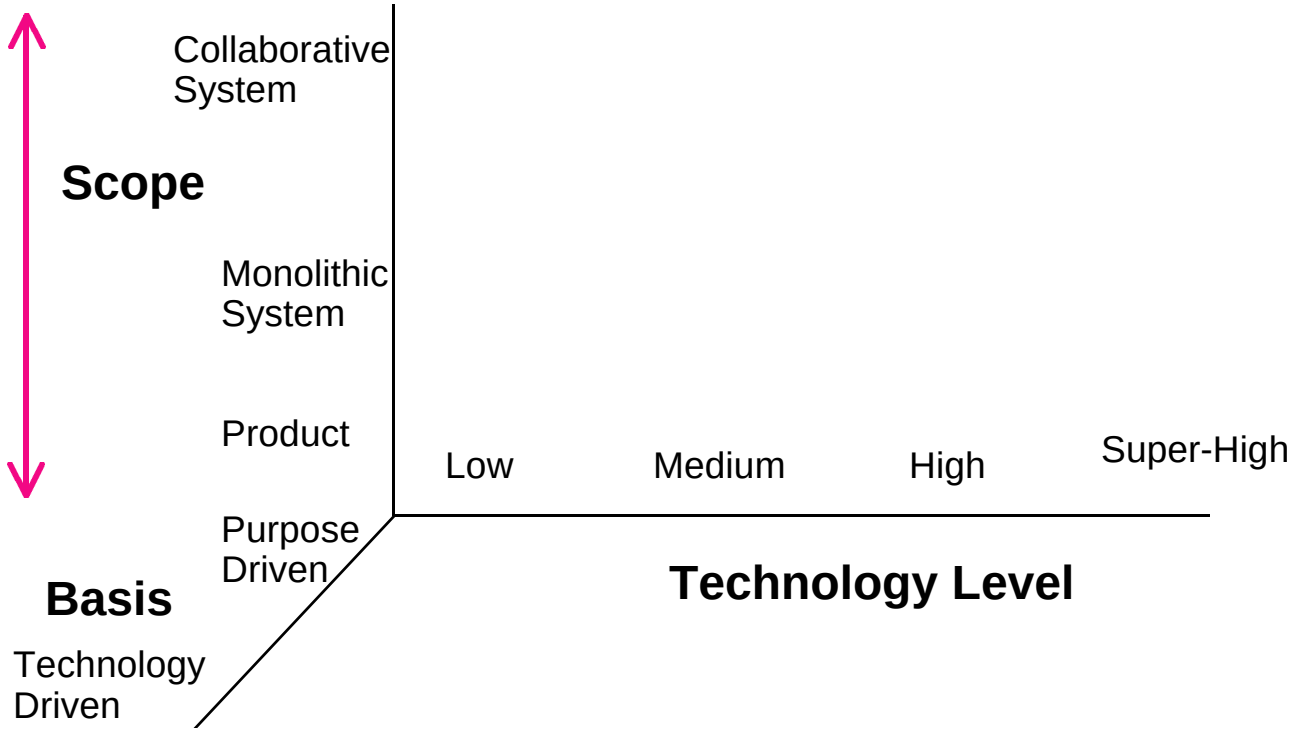
- Standards for describing architectures
  - » Enterprise level versus system/project level
  - » Normative frameworks (and the world of languages and views)
  - » "Higher-level" organizations
- Standards/practices for doing architecting
  - » Processes integrated with modeling languages and heuristics
  - » Heuristics and patterns
- Standard architectures
  - » Normative components, structures, or properties

How are Systems Different?

- If we are going to find good practices we'll have to find good classifications
- A useful system classification:
  - » Distinguishes systems with different good development practices
  - » Clusters systems with the same good development practices
- Other taxonomies may be interesting, but they aren't useful
- Some classifications that have been found to be useful:
  - » Technology level: Low => Super-High
  - » Scope: Product => System-of-systems or Enterprise
  - » Basis: Purpose-driven versus Technology-driven
- Many good practice distinctions are clear across the range, but others are not

Example: Do Architectures get Bigger or Smaller?

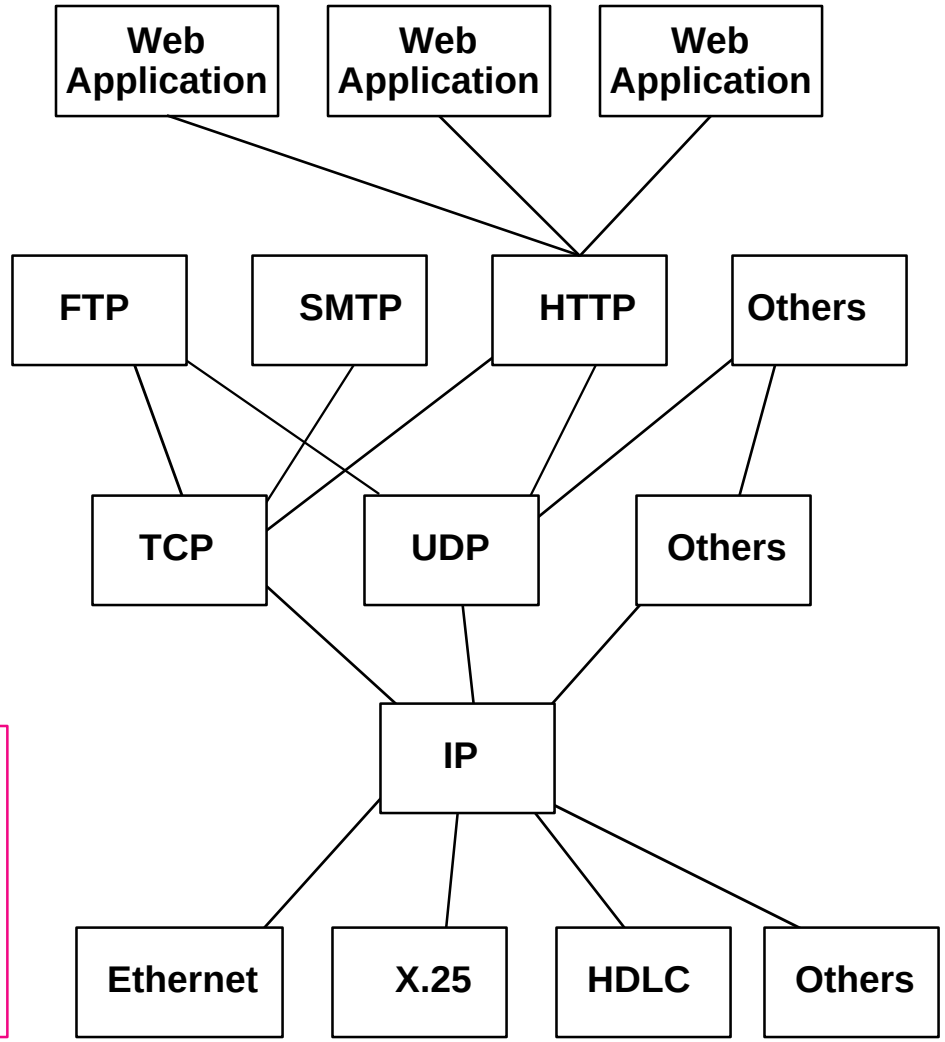
As the "system" gets bigger does the architecture get bigger?



- Collaborative System: A collection of systems which:
  - » Provides functions not provided by any of the component systems alone
  - » If partially or wholly disassembled the components continue to fulfill useful purposes in their own right, and
  - » The constituent systems are managed, at least in part, for their own purposes rather than the purposes of the collected system-of-systems.

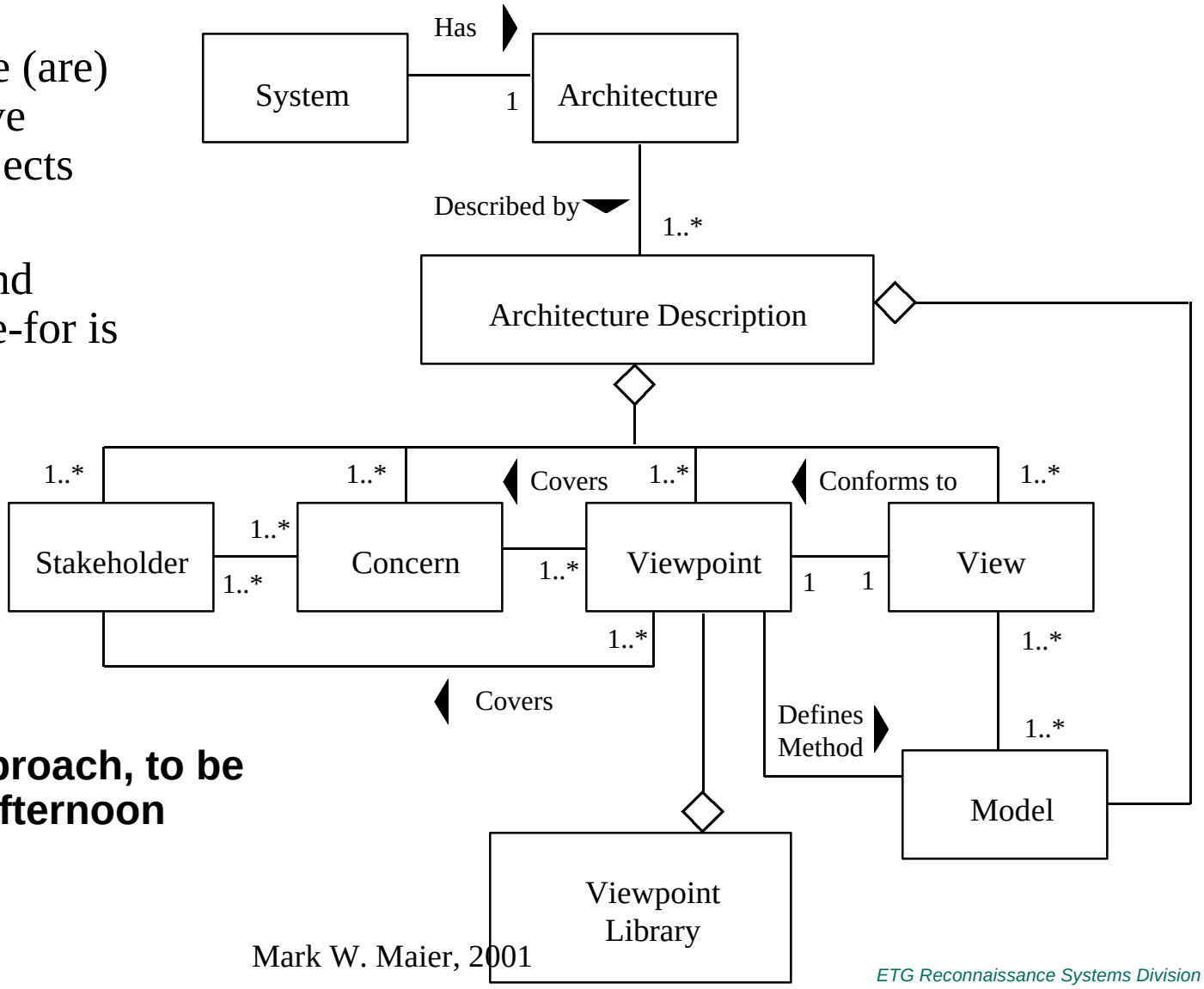
Answer: Internet Architecture

- The architecture of the Internet gets smaller as you aggregate
- The "organizing structure" is IP
  - » The architecture is no less real for not being physical
  - » This is not uncommon today as we architect *collaborative systems*
- Collaborative bodies control architecture evolution
  - » IETF, etc.
- Structure repeats in the web
- Lesson: Avoid the sin of architecting too much
  - » Also see example of MPEG and similar standards



“Describing” Standards: IEEE 1471

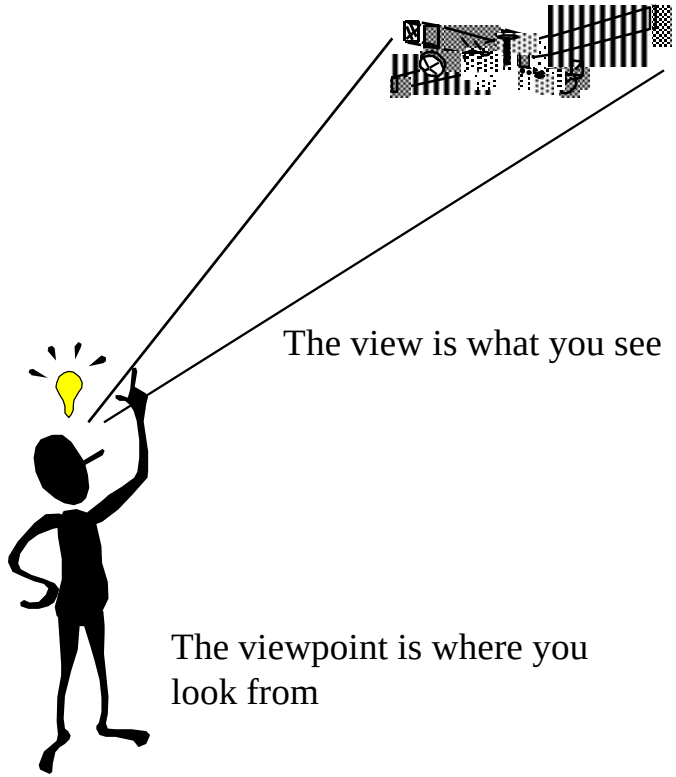
- There could-be (are) standards above individual projects
- Architecture “blueprints” and standards there-for is a hot topic



•1471 has an approach, to be discussed this afternoon

Organization by Views

- Most frameworks are organized by “view”
- Definitions:
  - » A view is a description of the entire system from the perspective of a set of related concerns. A view is composed of one or more models.
  - » A viewpoint is a standard or template for constructing a view
  - » A model is an abstraction or representation of some aspect of a thing (here of a system)
- An “architecture” and an “architecture description” are often used interchangeably



Conclusions

- If one is going to be an architect of complex systems, one should not use the metaphor lightly
  - » It tells us that concept formulation, initial structuring, choice of invariant forms, and certification for use are different
  - » It illustrates the importance of distinctions in scope and technology of systems to methods (a house versus an urban plan versus “The Big-Dig”
  - » It shows that flexibility of thought (transdisciplinary methods, heuristics, soft-hard methods) are not incompatible with research and education
  - » It also shows the importance of “third-party-like” relationships in effective development
- Architecting happens, whether planned and deliberate or not
  - » Bad architectures, in the sense of bad initial concepts, are the sure road to failure
  - » Today’s architectures are tomorrows legacy

For More Information

- Some of my books/papers
  - The Art of Systems Architecting, Second Edition, Maier and Rechtin, CRC Press, 2000
  - Architecting Principles for Systems of Systems, Systems Engineering, Journal of INCOSE, 1:4, 1998
- IEEE Architecture Working Group websites:
  - http://www.pithecanthropus.com/~awg
- IEEE 1471 Interest Group mailing list

  To subscribe, send an email message to majordomo@davebert.mitre.org, with the line:

  subscribe IEEE-1471-interest <<your Name>>

  (plain text only, no HTML markup or .signatures, please)