Architecting The Unknown

Grady Booch
IBM Fellow
Chief Scientist for Software Engineering
Chief Scientist for Watson/M

Email: gbooch@us.ibm.com
Twitter: @grady_booch
Web: computingthehumanexperience.com
The fundamental premise of science is that the cosmos is understandable.
The fundamental premise of computing is that the cosmos is computable.
Massive Prime
+ CIS Vancouver

Setting the scene for Vantage Point
At a macro level, finding a solution to any complex problems is an iterative and incremental process with periods of punctuated equilibrium.
• Strong vision plus fundamentals in execution
  – Predictability
  – Repeatability
  – Tangible artifacts
  – Measurable results
  – Stable intermediate forms
  – Continuous evolution
  – Continuous refactoring

• Organizational architecture
  – Organized labor
  – Shared stories
  – Tribal memory
  – Rituals
  – Trust
  – Serendipitous connections
The code is the truth but not the whole truth.
All architecture is design but not all design is architecture. A system’s architecture is defined by its significant design decisions (where significant is measured by the cost of change).
Why We Architect

- Vehicle for communication among stakeholders
- Reasoning about an evolving system
- Intentional transformation
- Mechanism for attacking risk
- Accountability
- Preservation of tribal memory
How We Architect

- Crisp abstractions
- Clear separation of concerns
- Balanced distribution of responsibilities
- Simplicity

- Grow a system through the iterative and incremental release of an executable architecture
How I Architect

• Start with use cases
• Shape the most significant design decisions
• Reason about those decisions from the logical, implementation, and deployment views
• Evolve the key abstractions, subsystems, patterns, and services by growing a minimal viable product.
• Repeat
Theft

Classical System

Unprecedented System

Method

Intuition
Engineering An Unprecedented System

- Do the hard part first
- If it works, then it’s useful
- All design domains contain an irreducible kernel of problems that are best addressed through creative and heuristic approaches that combine art and science
- All serious mistakes are made in the first day
- No complex system can be optimum to all parties concerned, nor all functions optimized
How Unprecedented Systems Succeed

• As a rule, software systems do not work well until they have been used, and have failed repeatedly, in real applications [Dave Parnas]
• A complex system that works is invariably found to have evolved from a simple system that works [John Gall]
• Both vertically and horizontally, the most resilient systems tend to exhibit loose coupling and tight cohesion among components [Simon]
• Simplicity is the most important consideration in a design; both implementation and interface must be simple, though it is more important for the implementation to be simple [Gabriel]
• Software which is flexible, simple, sloppy, tolerant and altogether forgiving turns out to be most resilient [Bosworth]
• To an engineer, good enough means perfect [Alexander Calder]
Architecting The Unknown

• Allow yourself to fail
• Fail early
• Fail often
• Fail safely
• Fill your tool bag
• Get out of your comfort zone
• Be inspired by the real world
Software is the hidden writing that whispers the stories of possibility to our hardware.