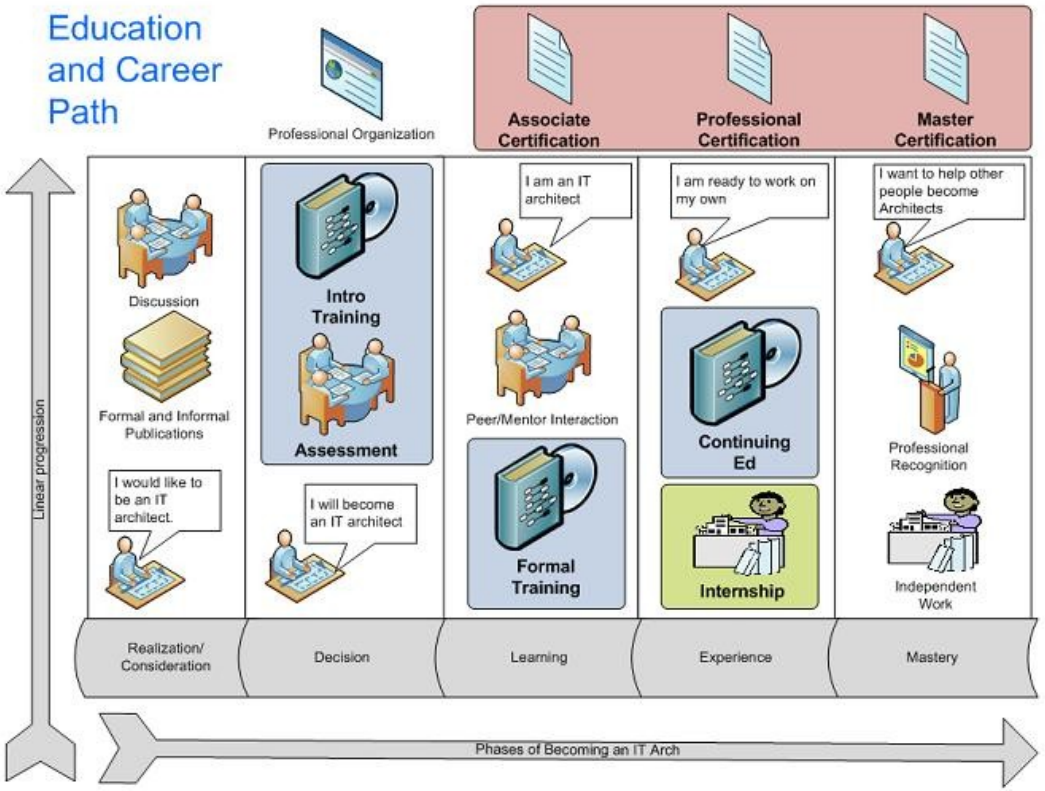
Improving Architecture Competence

Jason Baragry
Software Architect
Sun Microsystems
(a wholly-owned subsidiary of Oracle)

IASA Education Initiative



- Business Technology Strategy
- Design for Architects
- Understanding the IT Environment
- Human Dynamics
- Quality Attributes
- Software Architecture
- Infrastructure Architecture

http://www.iasahome.org/web/home/education

Waldo - “On System Design”

“I am beginning to believe that the art and craft of system design is in danger of being lost...

Carefully designed systems, in which the right abstractions are combined in just the right way to produce a system ... are unlikely to happen using the kind of design techniques that are popular today...

We are unable to train engineers and scientists adequately in system design. The economics of the industry push us in directions that don’t favor design."

Waldo - “On System Design”

“...good design is a capability that some people have, and others simply do not...

Whether this is an innate skill that people are born with, or one that is cultivated over time in ways that we don’t understand, is a question far too deep for me to address here. I neither know nor care...

But by the time someone is designing a computer system, whatever it takes to be a good designer is either there or it is not. When it is there, it can be developed and honed. It can also be degraded or warped. But when it is not there, there is no technique or process that can make up the deficit...”

Waldo - “On System Design”

“My own conclusion is that system design is really a matter of technique, a way of thinking rather than a subject that can be taught in a particular course.

It might be possible to build a program that teaches system design by putting students through a series of courses that hone their system design skills as they move through the subject matter of the courses.

Such a series of courses would, in effect, be a formalized version of the apprenticeship that is now the way people acquire their system design technique.”

Hope for passing on design skills

- Agile Methods
  - TDD requires thought about abstractions and interfaces
  - time to think and share, mentor and learn.
  - Criticism is built into the process.
  - Work occurs in small groups
- Open Source projects
  - It has a built-in meritocracy, with masters at the top
  - New developers learn from example
  - Developers face real criticism and have the time and opportunity to learn and improve.
  - The system is an open, growing example.

SEI: workshop on Arch Competence

- understand the current trends, methods, purposes, outcomes, and experiences that pertain to helping architects and architecture organizations measure and improve their architecture competence.
- Participants:
  - oversee architecture competence improvement programs in their respective companies
  - consult with companies who wish to increase their architecture competence
  - have related academia-based efforts
- Relate it to the SEI Arch Competence Framework

Ratheyon

- the Architecture Review Board (ARB)
- RCAP—a formally defined architect certification program
- REAP—a standards-based architecture process
- reference architectures
- an architecture repository
- architecture tools that employ various vendor alternatives
- a cost estimation tool for developing an architecture
- collaboration opportunities across community of architects
- architecture training, with multiple levels of detail
- architecture standards—government and industry

Bredemeyer Consulting

- is architecture a strategic concern in the org.
  - is the strategy firmly footed so that the architect can be successful?
  - Is the architect allowed to be a strategic contributor to the formation of the strategy?
  - Who is the lowest level manager in the organization whose purview includes architecture?
  - Are architects present at all meetings where they could make a significant contribution?
  - How important are the economic decisions being made by architects?

Bredemeyer

"there is a significant shift from tech lead to architect, and again from architect to product line/family architect and again to solution/portfolio architect, and chief architect and enterprise architect.

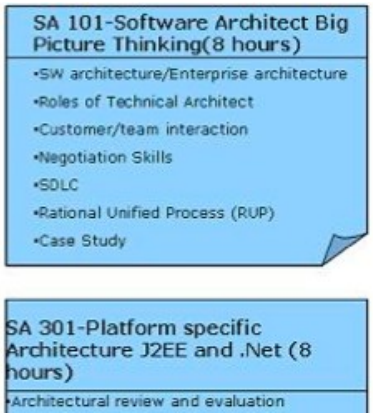
... we have to become more self-conscious about the fact that we have quite distinct pools of competencies ... we need to draw the talent that seeds the next higher layer (in terms of breadth of scope and strategic contribution).

It is useful to be explicit about nurturing the architect tree"

	What you KNOW	What You DO	What You ARE
Technology	In-depth understanding of the domain and pertinent technologies Understand what technical issues are key to success Development methods and modeling techniques	Modeling Tradeoff analysis Prototype/experiment/simulate Prepare architectural documents and presentations Technology trend analysis/roadmaps Take a system viewpoint	Creative Investigative Practical/pragmatic Insightful Tolerant of ambiguity, willing to back-track, seek multiple solutions Good at working at an abstract level
Consulting	Elicitation techniques Consulting frameworks	Build "trusted advisor" relationships Understand what the developers want and need from the architecture Help developers see the value of the architecture and understand how to use it successfully Mentor junior architects	Committed to others' success Empathetic, approachable An effective change agent, process savvy A good mentor, teacher
Strategy	Your organization's business strategy and rationale Your competition (products, strategies and processes) Your company's business practices	Influence business strategy Translate business strategy into technical vision and strategy Understand customer and market trends Capture customer, organizational and business requirements on the architecture	Visionary Entrepreneurial
Organizational Politics	Who the key players are in the organization What they want, both business and personal	Communicate, communicate, communicate! Listen, network, influence Sell the vision, keep the vision alive Take and retake the pulse of all critical influencers of the architecture project	Able to see from and sell to multiple viewpoints Confident and articulate Ambitious and driven Patient and not Resilient Sensitive to where the power is and how it flows in your organization
Leadership	Yourself	Set team context (vision) Make decisions (stick) Build teams Motivate	You and others see you as a leader Charismatic and credible You believe it can and should be done, and that you can lead the effort You are committed, dedicated, passionate You see the entire effort in a broader business and personal context

Satyam Consulting

- sw eng -> designer -> bus analyst -> bus consultant or tech architect -> practice head and solution head. finally consulting head



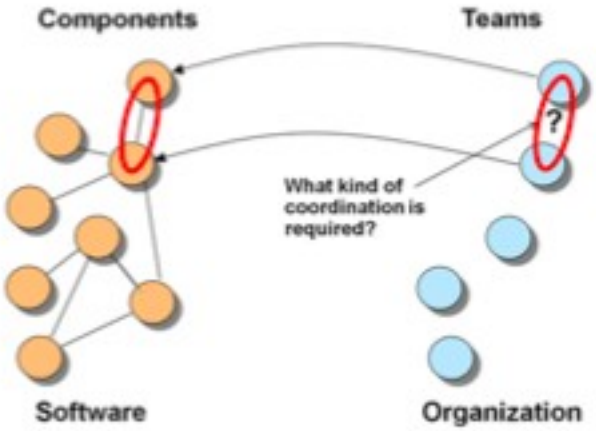
- Measuring competence:
  - faster: the number of arch engagements completed successfully
  - better: number of messages of appreciation received
  - cheaper, how much the final project comes in under budget
  - larger, size and complexity of the completed engagements
  - steadier, measured by the consistency across business units

Boeing

- internal certification process.
- domains of expertise:
  - small real-time embedded systems
  - missions systems
  - satellite systems including flight controls and payload systems
  - enterprise systems—for example, airplane parts planning
- internal sw arch conference for sharing ideas

Hersleb – CMU research

- What does the interaction between components imply about the needed interaction between the teams that produce them?
  - > Following Conway's law
- architectural decisions create a “coordination landscape,” architecture and organizational structures are strongly related, and congruence is necessary for project success



SEI Arch Competency Framework

- Cluster items into 3 broad areas:
  - > Engineering Competence Area category
  - > Project Planning and Execution
  - > Organizational Ecosystem
    - > decomposed into a number of activities, then competence areas and practices. E.g., Project Management activity:

Sample Practices

- making architects members of the product/system planning team
- giving architects access to the organization's market and customer data and business case
- having architects participate in the creation of a project's vision statement

Commom Failure Outcomes

- Outsourced much of the architecture function
- Outsourced development.
- A single hero architect.
- The organization is geographically dispersed... divergence in the understanding of the architecture.
- The architecture team is detached from the reality of the project/business
- The business people are detached from the reality of the architecture and technology

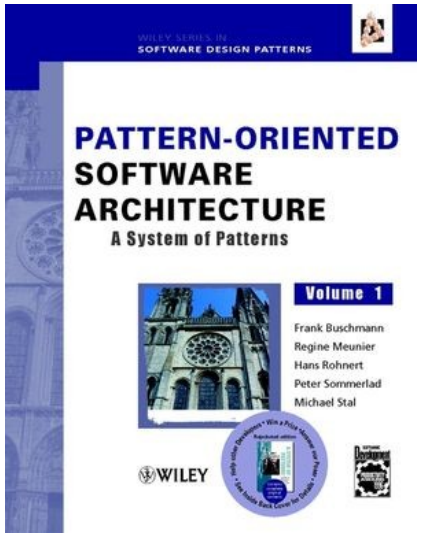
SEI: Improvement Mechanisms

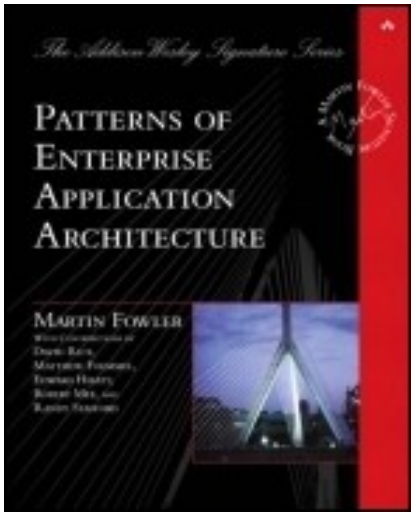
- leadership development program:
  - junior architects meet with each other and senior leadership figures.
- Online message boards:
  - internal. observed to foster desirable levels of information exchange and networking
- Structured analysis sessions:
  - have been observed to foster effective and focused improvement activities.

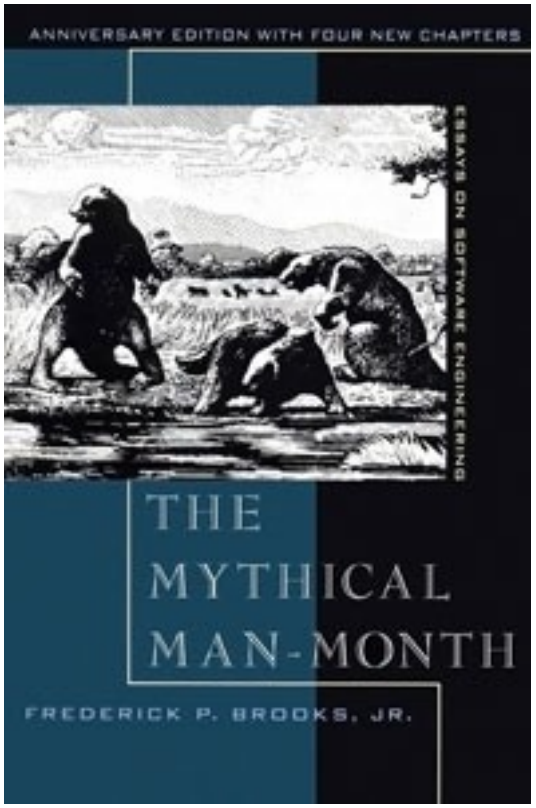
Where should I start?

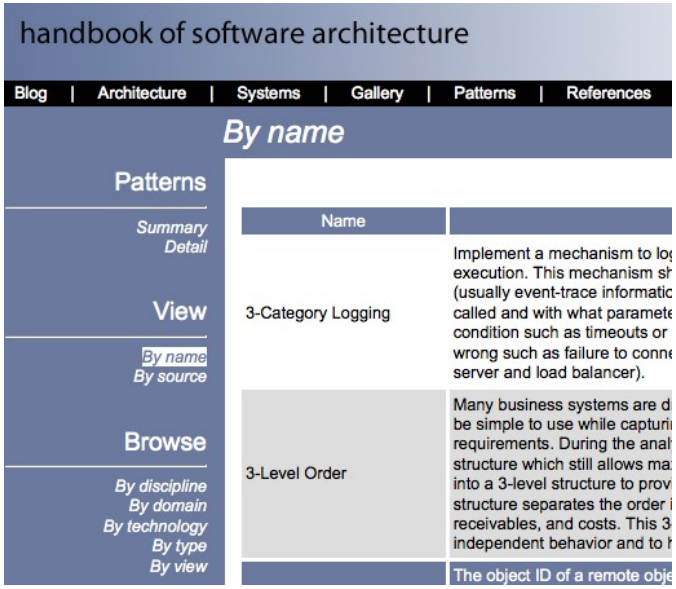
work with a good sw architect

Where should I start?





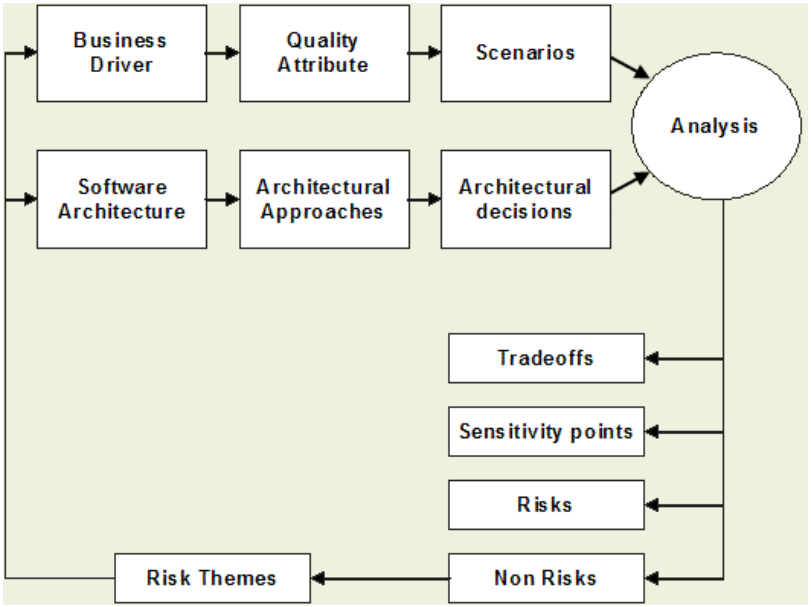


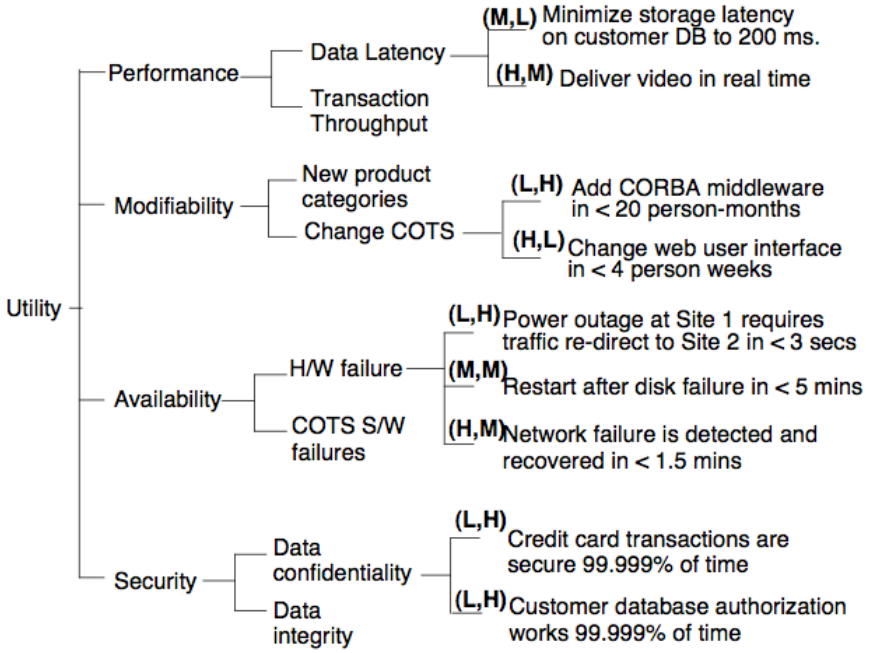


- Classic Articles
  - No Silver Bullet
  - Programming in the Large vs Programming in the Small
  - On the Criteria To Be Used in Decomposing Systems into Modules

Where should I start?

- learn an analysis method
  - > E.g., SEI's ATAM





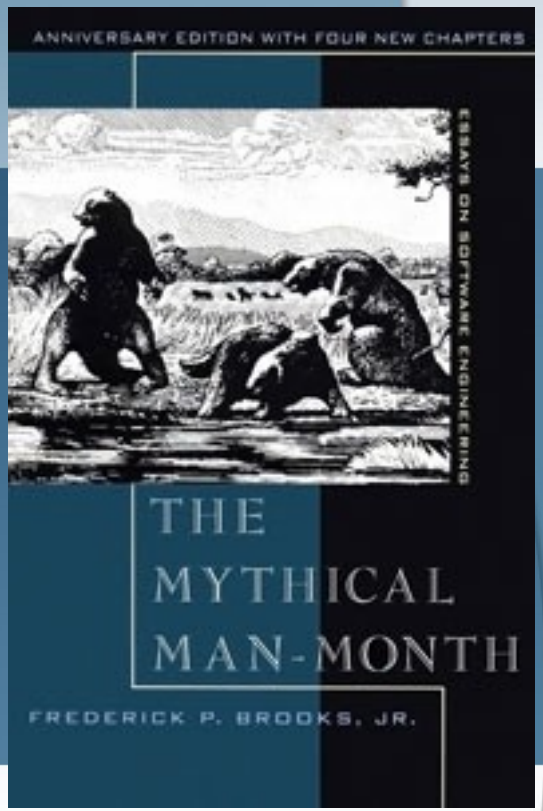
Where should I start?

- “laws” of software development
  - e.g., Gall's, Brooks', Conway's

What should IASA.no do?

Thank You

Jason Baragry
jason.baragry@sun.com
blogs.sun.com/jason
twitter:@jbaragry



"The programmer … works only slightly removed from pure thought-stuff. He builds castles in the air, from air, creating by exertion of the imagination. Few media of creation are so flexible, so easy to polish and rework, so readily capable of realizing grand conceptual structures."

"I postulate that the difficulties with building software can be divided ... into the essence, which is the conceptual structure of the software itself, quite apart from any realization, and the accidents, … the process of realizing the conceptual structure in executable form.

- Brooks, *No Silver Bullet Retrospective*

“I believe the hard part of building software to be the specification, design, and testing of this conceptual construct, not the labor or representing it and testing the fidelity of the representation. We still make syntax errors, to be sure; but they are fuzz compared with the conceptual errors in most systems.”

- Brooks, *No Silver Bullet*

Its hard to develop solution models

A blind man, a Lesbian, and a frog walk into a bar. The barkeep looks at them and says,

"What is this – a joke?"

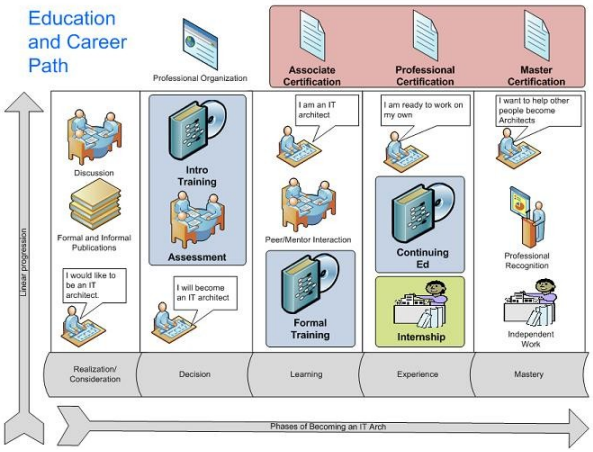
- The only solution is to Iterate

  We produce the obvious solution and then criticize them, in order to find out why they do not work. In this way, we become acquainted with the problem, and may proceed from bad solutions to better ones – provided always that we have the creative ability to produce new guesses, and more new guesses. …

  > Popper, *Conjectures and Refutations*


Sun
microsystems
Improving Architecture Competence
Jason Baragry
Software Architect
Sun Microsystems
(a wholly-owned subsidiary of Oracle)

IASA Education Initiative



- Business Technology Strategy
- Design for Architects
- Understanding the IT Environment
- Human Dynamics
- Quality Attributes
- Software Architecture
- Infrastructure Architecture

http://www.iasahome.org/web/home/education

2

*** sounds easy, but not much material.**
*** lots on what should an architect be, lots on specific skills (see IASA training),**
*** but not much on how to improve architects or architect communities.**

Waldo - “On System Design”

“I am beginning to believe that the art and craft of system design is in danger of being lost...

Carefully designed systems, in which the right abstractions are combined in just the right way to produce a system ... are unlikely to happen using the kind of design techniques that are popular today...

We are unable to train engineers and scientists adequately in system design. The economics of the industry push us in directions that don’t favor design."

3

http://research.sun.com/techrep/Perspectives/PS-2006-6.pdf

presented at OOPSLA 2006 Essays

Waldo - “On System Design”

“...good design is a capability that some people have, and others simply do not...

Whether this is an innate skill that people are born with, or one that is cultivated over time in ways that we don’t understand, is a question far too deep for me to address here. I neither know nor care...

But by the time someone is designing a computer system, whatever it takes to be a good designer is either there or it is not. When it is there, it can be developed and honed. It can also be degraded or warped. But when it is not there, there is no technique or process that can make up the deficit...”

4

* **Brooks (turing award lecture) talked of the work he had been doing to try to find the underlying common feature of good design, not just in computer hardware and software but also in such endeavors as architecture, graphics, and the fine arts. The only thing that he could find that good designs had in common was that they were produced by good designers**

Waldo - “On System Design”

“My own conclusion is that system design is really a matter of technique, a way of thinking rather than a subject that can be taught in a particular course.

It might be possible to build a program that teaches system design by putting students through a series of courses that hone their system design skills as they move through the subject matter of the courses.

Such a series of courses would, in effect, be a formalized version of the apprenticeship that is now the way people acquire their system design technique.”

5

*** Gilbert Ryle: distinction between knowing how and knowing that**

Hope for passing on design skills

- Agile Methods
  - > TDD requires thought about abstractions and interfaces
  - > time to think and share, mentor and learn.
  - > Criticism is built into the process.
  - > Work occurs in small groups
- Open Source projects
  - > It has a built-in meritocracy, with masters at the top
  - > New developers learn from example
  - > Developers face real criticism and have the time and opportunity to learn and improve.
  - > The system is an open, growing example.

6

*** Agile and FOSS and also lead to crap designs.**
*** Communication and feedback loops**
**** Whether it be to a group of peers or a master, the real point is that the design needs to be expressed to someone else.**
*** Not against big design up-front:**
**** The very act of writing the design document helps to clarify the design itself. In the same way, having to communicate the design during group programming helps to clarify and simplify the design**
*** Should add design patterns**
**** personal communication: design patterns worried about: "religious fervour", "abstractions of programming practices".**
*** Bonus:**
**** This ability to learn, to teach, and to tackle hard technical problems without the oversight or interference of management is also, I believe, one of the prime reasons for the popularity of open source projects among engineers. Such projects are places where technical decisions can be made on technical grounds, and where the decision making powers are given to those who have shown technical ability in the past.**

SEI: workshop on Arch Competence

- understand the current trends, methods, purposes, outcomes, and experiences that pertain to helping architects and architecture organizations measure and improve their architecture competence.
- Participants:
  - > oversee architecture competence improvement programs in their respective companies
  - > consult with companies who wish to increase their architecture competence
  - > have related academia-based efforts
- Relate it to the SEI Arch Competence Framework

7

http://www.sei.cmu.edu/library/abstracts/reports/09tn005.cfm
• oversee architecture competence improvement programs in their respective companies
• consult with companies who wish to increase their architecture competence
• have academia-based efforts of their own through which they are trying to build a competence assessment instrument or investigate some aspect of architecture competence

·Experience Reports:
- **2 large DoD contractors: Ratheyon, Boeing**
- **1 Indian consulting house: Satyam**
- **1 experienced SW Arch competence consultant: Bredemeyer**
- **1 academic on Organisation coordination theory and SW Arch: Herbsleb (CMU)**

Ratheyon

- the Architecture Review Board (ARB)
- RCAP—a formally defined architect certification program
- REAP—a standards-based architecture process
- reference architectures
- an architecture repository
- architecture tools that employ various vendor alternatives
- a cost estimation tool for developing an architecture
- collaboration opportunities across community of architects
- architecture training, with multiple levels of detail
- architecture standards—government and industry

8

*** Identified overall drivers for their architecture**
**** GIG (global information grid)**
**** Increasing complexity of the solutions they produce.**
*** Mandates**
**** requirement to use DoDAF**
**** The second mandate requires that every federal agency have a "strategic plan [that] will shape the redesign of work processes and guide the development and maintenance of an Enterprise Architecture and a capital planning and investment control process..**

*** Both make it clear that architecture competence is a definite requirement for companies wishing to do business with the U.S. governmen**

*** four areas of focus: people, process, governance, and resources**
Initiatives:
- **the Architecture Review Board (ARB)—a company-wide governance body that provides oversight and guidance on architecture policy, process, and certification, and so forth. The ARB also performs independent reviews of architectures.**
- **the Raytheon Certified Architect Program (RCAP)—a formally defined architect certification program**
- **the Raytheon Enterprise Architecture Process (REAP)—a standards-based architecture process**
- **reference architectures—a collection of domain-specific, partially populated templates for architecture efforts**
- **an architecture repository—a secure, metadata-enabled data store**
- **architecture tools that employ various vendor alternatives**
- **a cost estimation tool that helps estimate the effort and cost of developing an architecture**
- **collaboration opportunities—various means for sharing knowledge across Raytheon's com- munity of architects**
- **architecture training, with multiple levels of detail based on the students' level of experience**
- **architecture standards—government and industry resources that can support the items listed above**

Bredemeyer Consulting

- is architecture a strategic concern in the org.
  - > is the strategy firmly footed so that the architect can be successful?
  - > Is the architect allowed to be a strategic contributor to the formation of the strategy?
  - > Who is the lowest level manager in the organization whose purview includes architecture?
  - > Are architects present at all meetings where they could make a significant contribution?
  - > How important are the economic decisions being made by architects?

9

*** Bredemeyer consulting**

*** Architecture, he said, is the translation from strategy to technology.**
*** investigate whether architecture is a strategic concern in the organization.**
o is the strategy firmly footed so that the architect can be successful?
o Is the architect allowed to be a strategic contributor to the formation of the strategy?
o Who is the lowest level manager in the organization whose purview includes architecture? If it's the CEO, do architects and the CEO talk readily with each other?
o Are architects present at all meetings where they could make a significant contribution? • How important are the economic decisions being made by architects?

*** has his own competency framework**
o The five areas are technology, consulting, strategy, organizational politics ("like gravity, the weak force that holds everything together"), and leadership.

o

Bredemeyer

"there is a significant shift from tech lead to architect, and again from architect to product line/family architect and again to solution/portfolio architect, and chief architect and enterprise architect.

... we have to become more self-conscious about the fact that we have quite distinct pools of competencies ... we need to draw the talent that seeds the next higher layer (in terms of breadth of scope and strategic contribution).

It is useful to be explicit about nurturing the architect tree"

	What you KNOW	What You DO	What You ARE
Technology	In-depth understanding of the domain and pertinent technologies Understand what technical issues are key to success Development methods and modeling techniques	Modeling Tradeoff analysis Prototype/experiment/simulate Prepare architectural documents and presentations Technology trend analysis/roadmaps Take a system viewpoint	Creative Investigative Practical/pragmatic Insightful Tolerant of ambiguity, willing to back-track, seek multiple solutions Good at working at an abstract level
Consulting	Elicitation techniques Consulting frameworks	Build "trusted advisor" relationships Understand what the developers want and need from the architecture Help developers see the value of the architecture and understand how to use it successfully Mentor junior architects	Committed to others' success Empathetic, approachable An effective change agent, process savvy A good mentor, teacher
Strategy	Your organization's business strategy and rationale Your competition (products, strategies and processes) Your company's business practices	Influence business strategy Translate business strategy into technical vision and strategy Understand customer and market trends Capture customer, organizational and business requirements on the architecture	Visionary Entrepreneurial
Organizational Politics	Who the key players are in the organization What they want, both business and personal	Communicate, communicate, communicate! Listen, network, influence Sell the vision, keep the vision alive Take and retake the pulse of all critical influencers of the architecture project	Able to see from and sell to multiple viewpoints Confident and articulate Ambitious and driven Patient and not Resilient Sensitive to where the power is and how it flows in your organization
Leadership	Yourself	Set team context (vision) Make decisions (stick) Build teams Motivate	You and others see you as a leader Charismatic and credible You believe it can and should be done, and that you can lead the effort You are committed, dedicated, passionate You see the entire effort in a broader business and personal context

http://www.bredemeyer.com/architect.htm

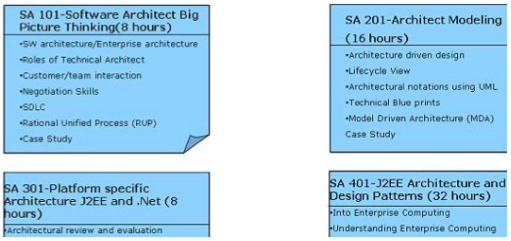
umbrella for architecture at various levels:

+ The critical insight here, is that there is a significant shift from tech lead to architect, and again from architect to product line/family architect and again to solution/portfolio archi- tect, and chief architect and enterprise architect. As this field matures, we have to become more self-conscious about the fact that we have quite distinct pools of competencies that we need to develop, and that from these pools, we need to draw the talent that seeds the next higher layer (in terms of breadth of scope and strategic contribution). It is useful to be expli- cit about nurturing the architect tree

*** Dana counseled us not to be too hard on ourselves because we don't do a good job of quantifying the benefit of architects and architectural practices. He reminded us that much larger decisions are often made on golf courses on the basis of personal relationships, trust, and the ad- vice of experts.**

Satyam Consulting

- sw eng -> designer -> bus analyst -> bus consultant or tech architect -> practice head and solution head. finally consulting head



- Measuring competence:
  - > faster: the number of arch engagements completed successfully
  - > better: number of messages of appreciation received
  - > cheaper, how much the final project comes in under budget
  - > larger, size and complexity of the completed engagements
  - > steadier, measured by the consistency across business units

11

The enterprise architect program consists of a course that covers the following topics:
*** The Zachman Framework**
*** The Open Group Architecture Framework (TOGAF) – the Core aspects**
*** The TOGAF Framework – Life Cycle and Governance aspects**
*** The Reference Model of Open Distributed Processing (RMODP) Framework – Engineering, Technology and Other Components**
*** Business Architecture to Application Architecture**
*** Focus on Business Analysis to Application Abstraction**
*** Application Architecture to Solution Architecture**
*** Focus on Nonfunctional Requirements**
*** SOA : What and Why**
*** SOA : Web Services and Beyond**
*** J2EE : Features and Emerging Trends**
*** .NET : CLR, COM+, WPF, WCF, WF**

Boeing

- internal certification process.
- domains of expertise:
  - > small real-time embedded systems
  - > missions systems
  - > satellite systems including flight controls and payload systems
  - > enterprise systems—for example, airplane parts planning
- internal sw arch conference for sharing ideas

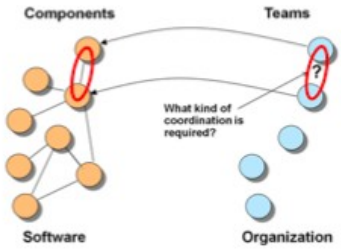
12

*** Their duties include**
****• software and system architecture requirements, analysis, and tradeoffs**
****• software and system architecture development and evaluation**
****• technical team leadership, project technical leadership, team building, and planning**
****• computing resources, networks, storage devices, busses, sensors, and communications hard-**
ware architecture and high-level system design
****• high-level software design**

*** domains of expertise:**
*** small real-time embedded systems—for example, flight controls inside a smart bomb, avionics flight controls, avionics software to which a pilot interfaces, or unmanned vehicle control systems**
*** missions systems—for example, airborne and ground-based Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) missions systems or airplane cabin systems**
*** satellite systems including flight controls and payload systems**
*** enterprise systems—for example, airplane parts planning and distributions systems for dozens of airlines**

Hersleb – CMU research

- What does the interaction between components imply about the needed interaction between the teams that produce them?
  - > Following Conway's law
- architectural decisions create a “coordination landscape,” architecture and organizational structures are strongly related, and congruence is necessary for project success



SEI Arch Competency Framework

- Cluster items into 3 broad areas:
  - > Engineering Competence Area category
  - > Project Planning and Execution
  - > Organizational Ecosystem
    - > decomposed into a number of activities, then competence areas and practices. E.g., Project Management activity:

Sample Practices

- making architects members of the product/system planning team
- giving architects access to the organization's market and customer data and business case
- having architects participate in the creation of a project's vision statement

Commom Failure Outcomes

- Outsourced much of the architecture function
- Outsourced development.
- A single hero architect.
- The organization is geographically dispersed... divergence in the understanding of the architecture.
- The architecture team is detached from the reality of the project/business
- The business people are detached from the reality of the architecture and technology

15

*** The organization has outsourced much of the architecture function and is critically dependent on a labor pool outside of the organization's control.**

- **The organization has outsourced development, so there is no internal source for developing architects.**
- **The organization depends on a single hero architect.**
- **The organization is geographically dispersed, work is distributed across teams, and problems are not discovered until integration time and are therefore costly to resolve. The root cause is divergence in the employees' understanding of the architecture.**
- **The architecture team is detached from the reality of the project/business, and the architecture is likely to be late.**
- **The business people are detached from the reality of the architecture and technology, and the project is likely to fail.**

*** other failaure themes were noted previously:** (e.g., failure to control the scope or failure to coordinate with important stakeholders

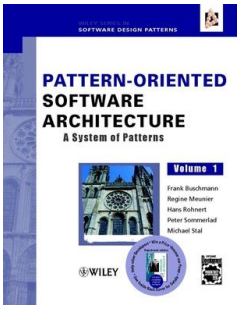
SEI: Improvement Mechanisms

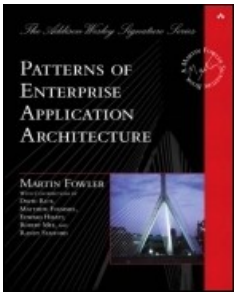
- leadership development program:
  - > junior architects meet with each other and senior leadership figures.
- Online message boards:
  - > internal. observed to foster desirable levels of information exchange and networking
- Structured analysis sessions:
  - > have been observed to foster effective and focused improvement activities.

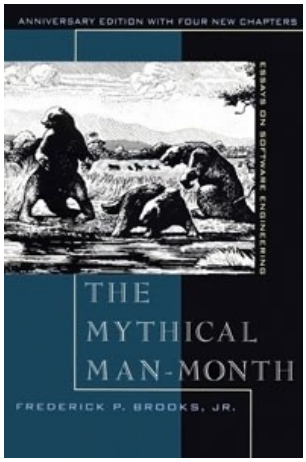
Where should I start?

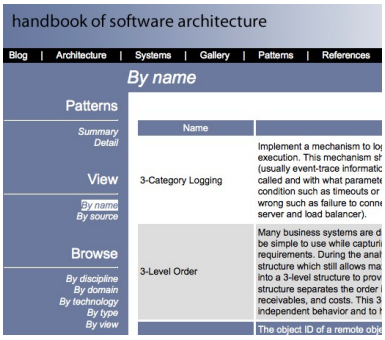
work with a good sw architect

Where should I start?









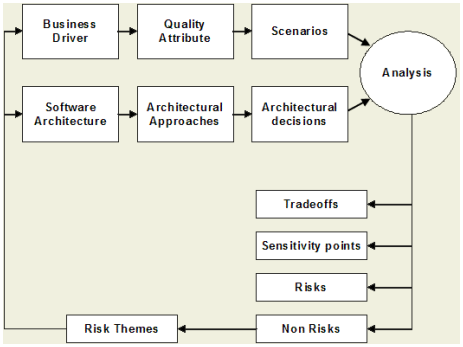
- Classic Articles
  - > No Silver Bullet
  - > Programming in the Large vs Programming in the Small
  - > On the Criteria To Be Used in Decomposing Systems into Modules

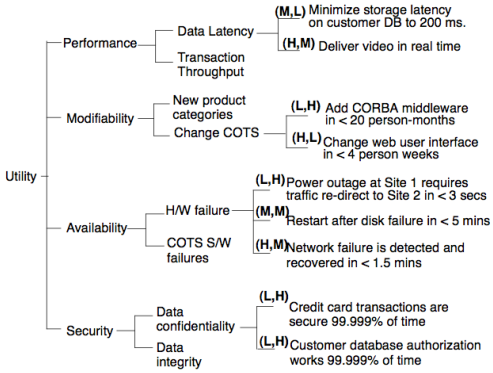
18

http://www.handbookofsoftwarearchitecture.com/index.jsp?page=Main

Where should I start?

- learn an analysis method
  - > E.g., SEI's ATAM





http://www.sei.cmu.edu/library/abstracts/reports/00tr004.cfm

Where should I start?

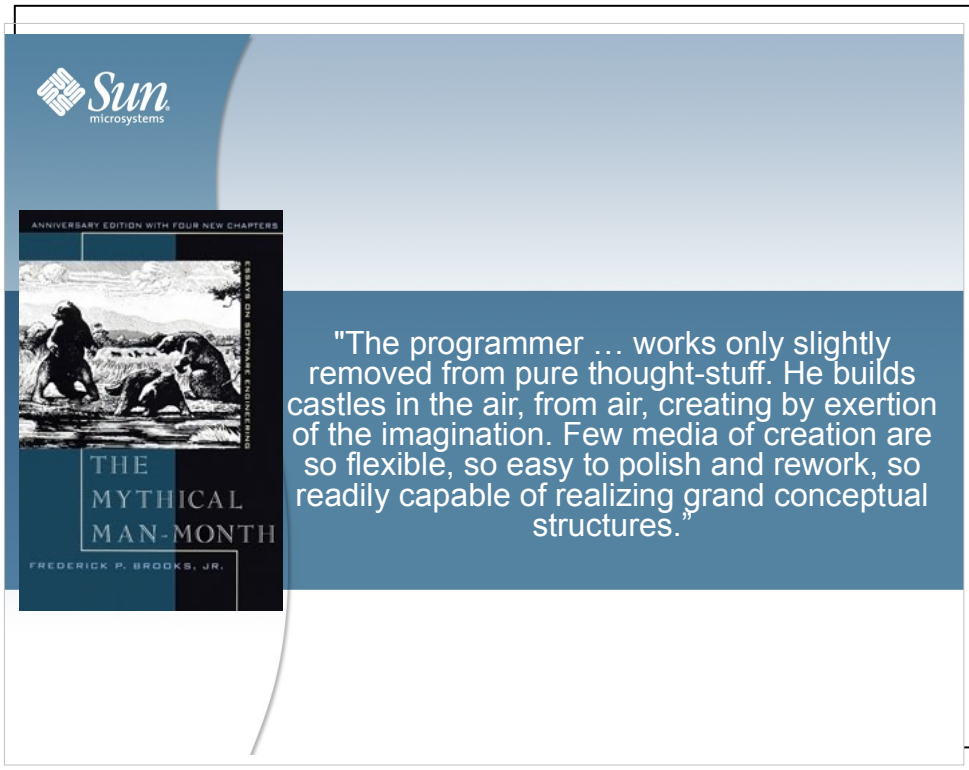
- "laws" of software development
  - > e.g., Gall's, Brooks', Conway's

* http://www.natpryce.com/articles/000749.html
* http://www.globalnerdy.com/2007/07/18/laws-of-software-development/

What should IASA.no do?

- Click to add an outline


Sun
microsystems
Thank You
Jason Baragry
jason.baragry@sun.com
blogs.sun.com/jason
twitter:@jbaragry



- our job is to model solutions in our minds and then get them realised using the technology of our discipline.

- regardless of whether you are an enterprise architect, BU architecture, or enterprise architect

Rest of the quote:

“Yet the program construct, unlike the poet's words, is real in the sense that it moves and works, producing visible outputs separate from the construct itself. It prints results, draws pictures, produces sounds, moves arms. The magic of myth and legend has come true in our time. One types the correct incantation on a keyboard, and a display screen comes to life, showing things that never were nor could be.

Programming then is fun because it gratifies creative longings built deep within us and delights sensibilities we have in common with all men."

"I postulate that the difficulties with building software can be divided ... into the essence, which is the conceptual structure of the software itself, quite apart from any realization, and the accidents, … the process of realizing the conceptual structure in executable form.

- Brooks, *No Silver Bullet Retrospective*

- At the heart of the argument is the distinction between accidental complexity and essential complexity. Accidental complexity relates to problems that we create on our own and can be fixed—for example, the details of writing and optimizing assembly code or the delays caused by batch processing. Essential complexity is caused by the problem to be solved, and nothing can remove it—if users want a program to do 30 different things, then those 30 things are essential and the program must do those 30 different things. (silver bullet on wikipedia)

- its the essential complexity that interests the solution architect. Getting the concept right

- No Silver Bullet Retrospective. OOPSLA 2007

- http://www.infoq.com/articles/No-Silver-Bullet-Summary

- See Fowler in the Werewolf suit:

- http://www.youtube.com/watch?gl=AU&hl=en-GB&v=Z-1X3duvryA

"I believe the hard part of building software to be the specification, design, and testing of this conceptual construct, not the labor or representing it and testing the fidelity of the representation. We still make syntax errors, to be sure; but they are fuzz compared with the conceptual errors in most systems."

- Brooks, *No Silver Bullet*

- For me, this is the main responsibility of the SW Architect – getting the Essential Complexity correct – the Conceptual Construct

- there are other responsibilities to be sure, but the consequences of getting them wrong are insignificant compared to getting this one wrong.

Its hard to develop solution models

A blind man, a Lesbian, and a frog walk into a bar. The barkeep looks at them and says,

"What is this – a joke?"

- The only solution is to Iterate

  We produce the obvious solution and then criticize them, in order to find out why they do not work. In this way, we become acquainted with the problem, and may proceed from bad solutions to better ones – provided always that we have the creative ability to produce new guesses, and more new guesses. …

  > Popper, *Conjectures and Refutations*

- Many different concepts exist at different levels of granularity and different levels of 'fuzzyness'.

- impossible to say which is the "correct" model. Can only falsify it to check if it is useful.

.

"To understand a problem means to understand its difficulties; and to understand its difficulties means to understand why it is not easily soluble – why the more obvious solutions do not work. We produce the obvious solution and then criticize them, in order to find out why they do not work. In this way, we become acquainted with the problem, and may proceed from bad solutions to better ones – provided always that we have the creative ability to produce new guesses, and more new guesses. … If we have been working on a problem long enough, and intensively enough, we begin to know it, to understand it, in the sense that we know what kind of guess or conjecture or hypothesis will not do at all, because it simply misses the point of the problem, and what kind of requirements would have to be met by any serious attempt to solve it. We begin to see the ramifications of the problem, its subproblems, and its connections with other problems."

.

- Popper, *Objective Knowledge*