Improving Software Architecture Competence

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Paul Clements
16 May 2007 / SATURN 2007

Software Engineering Institute | Carnegie Mellon

Improving Software Architecture Competence

Most of the work in architecture to date has been technical

- Design and creation
- Evaluation and analysis of architectures
- Styles and patterns
- Architectural reuse and software product lines
- Architectures for particular domains
- Architectural re-engineering and recovery

But architectures are created by *architects*…

- How can we help them do their best work?
- What does it mean for an architect to be competent?
- How can an architect improve his/her competence?

…working in *organizations*.

- How can we help an organization help their architects do their best work?
- What does it mean for an organization that produces architectures to be competent?
- How can an organization improve its competence in architecture?

Competence

Competent: Capable of performing an allotted or required function.

- Source: The American Heritage® Stedman's Medical Dictionary, Published by Houghton Mifflin Company, 2002.

Proposal:

A competent architect (architecting organization) is one that carries out his/her (its) architecture-related duties competently.

Performance of duties may be hindered by an incompetent organization, but this gives us a way to evaluate architects and organizations by looking at

- Past performance
- Present performance

Models of competence

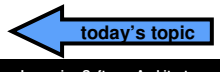
We are currently working with four models of competence

- "DSK model" -- a model based on the duties, skills, and knowledge of software architects
- "Human performance" model of competence – based on the teleonomic model of human competence of Thomas Gilbert
- "Organizational coordination" model of competence – based upon how organizational entities interact and coordinate among themselves
- "Organizational learning" model of competence – based on how organizations acquire, internalize, and utilize information

Prediction: Our ultimate model of architecture competence will take the best from all four of these.

Current status:

- #3 and #4 are in the learning stages
- Building a model based on #2, informed by #1



Duties/Skills/Knowledge Model

To measure how competent an architect is, we should be able to measure how well he/she

- performs architectural duties
- masters architectural skills
- possesses needed architectural knowledge

First step: Find out what those are!

- What are their duties?
- What skills and knowledge made them "capable of performing their allotted or required function?"

How can we find this out?

We can survey the "community"

Three broad sources of information (with count so far)

- "Broadcast" sources: Information written by self-styled experts for mass anonymous consumptions
  - Web sites: e.g., Bredemeyer, SEI, HP, IBM (16*)
  - Blogs and essays (16*)
  - "Duties" list on SEI web site
  - Books on software architecture (25 top-sellers)
- Education and training sources:
  - University courses in software architecture (29*)
  - Industrial/non-university public courses (22*)
  - Certificate and certification programs in architecture; e.g., SEI, Open Group, Microsoft (7*)
- "Architecture for a living" sources
  - Position descriptions for software architects (60)
  - Résumés of software architects (12)
  - Questionnaires from practicing architects (30+, not yet processed)

* Exhaustive or near-exhaustive web search

Survey results to date

To date, we have surveyed over 200 sources. A questionnaire campaign aimed at practicing software architects is underway.

We have cataloged

- 201 duties
- 85 skills
- 96 knowledge areas

We have grouped the data into clusters using an affinity exercise.

Architectural duties, after affinity exercise

Architecting	• Overall • Creating the architecture • Architecture evaluation and analysis • Documentation • Existing system and transformation
Life cycle phases other than architecture	• Requirements • Testing • Coding and development
Technology related	• Future technologies • Tools and technology selection
Interacting with stakeholders	• Overall • Clients • Developers
Management	• Project management • People management • Support for project management
Organization and business related	• Organization • Business
Leadership and team building	• Technical Leadership • Team Building

Architectural skills, after affinity exercise

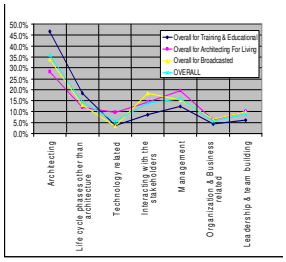
Communication skills	Out
	Both (i.e., two-way)
	In
Inter-personal skills	Within team
	With other people
	Leadership skills
Work skills	Effectively managing high workload
	Skills to excel in a corporate environment
	Skills for handling large amounts of information
Personal skills	Personal qualities
	Skills for handling unknown
	Skills for handling unexpected
	Learning

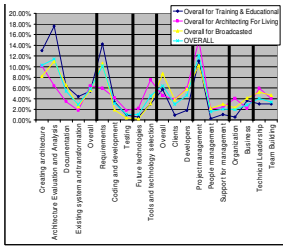
Architectural knowledge, after affinity exercise

Computer science knowledge	Knowledge of architecture concepts
	Knowledge of software engineering
	Design knowledge
	Programming knowledge
Knowledge of technologies and platforms	Specific
	Platforms
	General
Knowledge about organizational context and management	Domain
	Industry
	Enterprise knowledge
	Leadership and management

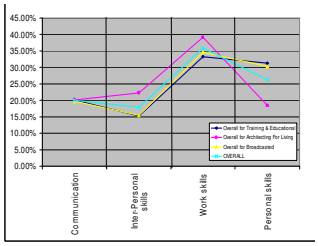
Graphs showing relative frequency of mention

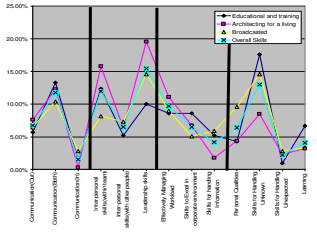
Duties



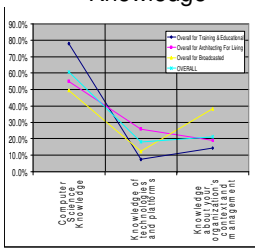


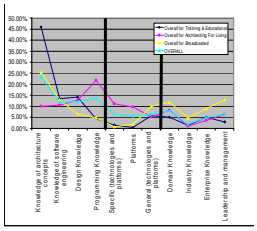
Skills





Knowledge





Software Engineering Institute | Carnegie Mellon

Improving Software Architecture Competence
Paul Clements, 29 March 2007

11

Duties/Skills/Knowledge

Advantages

- It applies equally well to individuals, teams, and organizations.
- It straightforwardly suggests an assessment instrument.
- It straightforwardly suggests an improvement strategy
  - Improve your duties
  - Improve your skills
  - Improve your knowledge

Software Engineering Institute | Carnegie Mellon

Improving Software Architecture Competence
Paul Clements, 29 March 2007

12

What are an *organization's* duties, skills, and knowledge?

List may include:

- Hire talented architects
- Establish a career track for software architects
- Make the position of architect highly regarded through visibility, reward, and prestige
- Establish a clear statement of duties, responsibilities, and authority for software architects
- Establish a mentoring program for architects
- Start an architecture training and education program
- Track how architects spend their time
- Establish an architect certification program
- Measure architects' performance
- Provide a forum for architects to communicate, and share information and experience
- Put in a place organization-wide development practices centered around architecture
- Establish and empower an architecture review board
- Measure quality of architectures produced
- Initiate software process improvement or software quality improvement practices

Gilbert's "Human Competence" work

Thomas Gilbert (1927-1995) is regarded as the "father of human performance" work

- Thomas F. Gilbert, *Human Competence – Engineering Worthy Performance.* HRD Press, Inc., 1996 "Tribute Edition." Book originally published 1978.

Gilbert strongly advocates measuring performance, not knowledge or behavior or motivation or skills or....

- "If I want to know if people are competent, I have to observe how they behave, don't I? My answer to such questions is a firm 'No!'"
- Worth = Value of result / Cost to achieve it. W = V / C
- Egyptian pyramids are "monuments to useless knowledge"
- Arabic alphabet was a much more "worthy" achievement

Measuring Worthy Performance: W = V / C

Performance (or the *worth* of the result) has the following dimensions or "requirements":

Quality

- **Accuracy**: Degree to which accomplishment matches a model, without errors of omission or commission.
- **Class**: Comparative superiority of an accomplishment beyond mere accuracy. Possible measures include market value, judgment points (as for show dogs), physical measures (such as number of mfg. flaws), opinion ratings (Oscars, "MVP")
- **Novelty**: An engine that gets 100mpg is novel. For artistic novelty we probably resort to judgmental points or opinion rating.

Quantity (or Productivity)

- **Rate**: Applies when bulk is time-sensitive; pieces produced per hour; time to completion
- **Timeliness**: Time, not bulk, is key: letter mailed by sundown, Cinderella home by midnight
- **Volume**: Bulk is important, but not time-sensitive. "How many fish did you catch?"

Cost

- **Labor** (behavior repositories): Includes direct overhead, benefits, wages, insurance, taxes
- **Material** (environmental support): Includes supplies, tools, space, energy
- **Management**: Supervision, its supports, public taxes, internal allocations of admin costs.

An assessment instrument: Gilbert's Performance Audit

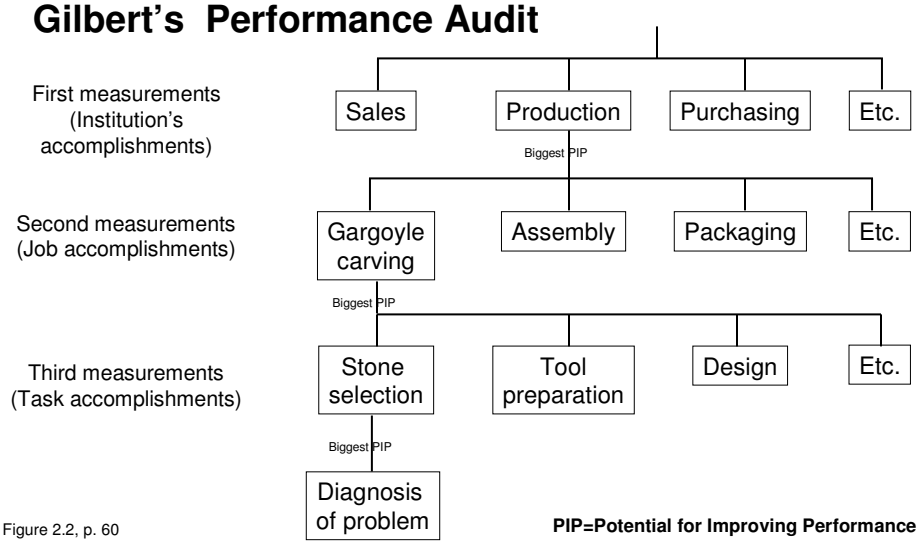
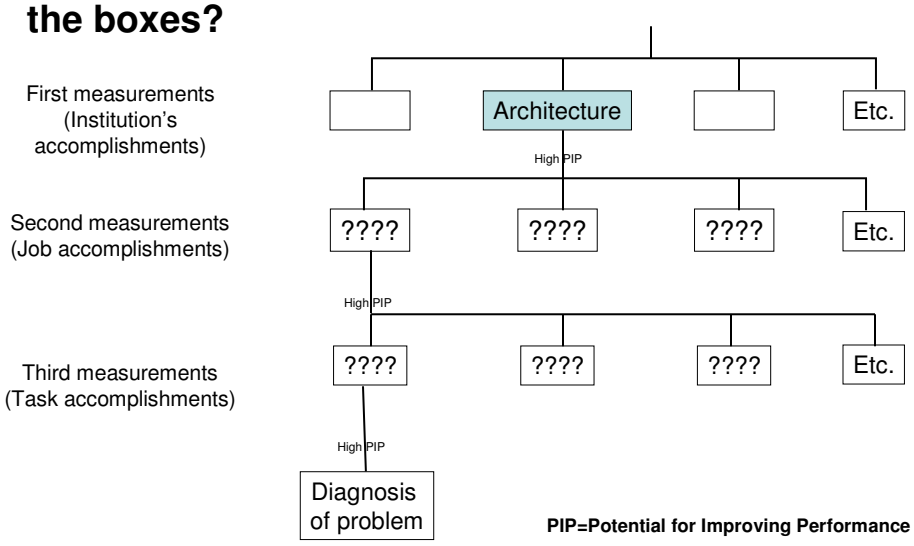


Figure 2.2, p. 60

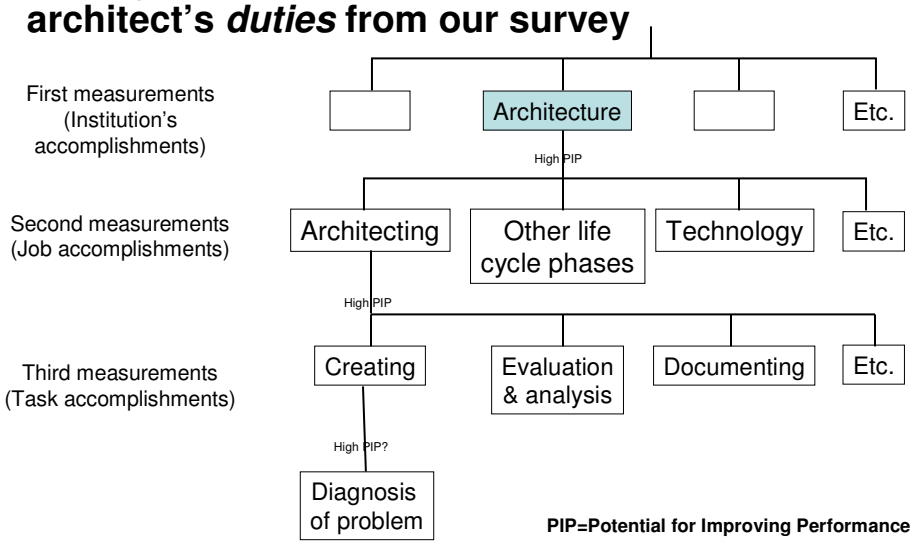
PIP=Potential for Improving Performance

How to apply this to architecture? What goes in the boxes?



PIP=Potential for Improving Performance

What goes in the boxes? Answer: The architect's *duties* from our survey



PIP=Potential for Improving Performance

Game plan to build competence model based on human performance

1. Identify what "worthy performance" means for each task involved in architecture.
2. Identify what costs are involved for each task involved in architecture.
3. Identify performance-related measures of each
4. Identify an exemplary measure – the best we could hope for – of each
5. Build an assessment instrument that will gather measurements in an organization, compare them to exemplar in each category, and identify best potential areas for performance improvement.
6. Suggest specific improvement strategies.

Example of applying Steps 1, 2, and 3

TASK: CREATING THE ARCHITECTURE

Quality

- Accuracy: Is the architecture the right one for the task at hand?
  Measure: Total cost of changes (= revisiting decisions) to the architecture during development [accounts for lots of small changes as well as number of big ones]. Cost means cost of making change in the architecture AND cost of downstream resulting changes. Measure as % of total cost of system, to (a) find exemplar; and (b) compare systems.
  Comments: Doesn't help for changes that were too expensive to address. Alternative measure is to capture satisfaction of important requirements (e.g., QA scenarios) and test fulfillment (e.g., ATAM-style walkthroughs).
- Class: How many architectures were influenced by this one? Whole thing? Pieces? Ideas?
- Novelty: N/A

Quantity (or Productivity)

- Rate: Time to completion.
- Timeliness: Deadlines met.
- Volume: Size of system.

Cost

- Labor (behavior repositories): Count staff hours for architects
- Material (environmental support): Staff hours for consultants; costs of tools used by architect. Travel costs. Communication costs.
- Management: Count staff hours for managers

Step 6: Gilbert's Behavior Model Showing Improvement Opportunities

Generalized description of Behavior Engineering Model, showing the things we can do to increase competence through greater behavior efficiency:

-----E: Environment Supports ----------		
Data	*Instruments*	*Incentives*
Relevant/frequent feedback about performance adequacy; Descriptions of expected performance. Clear and relevant guides to adequate performance.	Tools and materials designed to match human factors.	Adequate financial incentives contingent upon performance; Non-monetary incentives;
------P: Person's Repertory of Behavior ----------		
Knowledge	*Capacity*	*Motives*
Training matching exemplary performance; Placement	Flexible scheduling to match peak capacity; Prosthesis; Shaping; Adaptation; Selection.	Assessment of motives; Recruitment of people to match realities of situation.

Software Engineering Institute | Carnegie Mellon
Improving Software Architecture Competence
Paul Clements, 29 March 2007

21

Summary

We are just beginning to construct the Human Performance model

You can help!

The goal of the Architecture Competence working session is to craft descriptions and measures for as many architecture tasks as we can.

Software Engineering Institute | Carnegie Mellon
Improving Software Architecture Competence
Paul Clements, 29 March 2007

22

Acknowledgments

Members of the SEI Architecture Competence Project

- Len Bass
- Paul Clements
- Rick Kazman
- Mark Klein

The duties/skills/knowledge survey was carried out by

- Prageti Verma, Symbosis Centre for Information Technology, Pune
- Shivani K. Reddy, Symbiosis Centre for Information Technology, Pune
- Divya Devesh, Indian Institute of Technology Guwahati

Read more at www.sei.cmu.edu/architecture