


Software Architects: People & Teams


1



The Need

- The greatest architectures are the product of
 - A single mind or
 - A very small, carefully structured team
 - Rehtin, *Systems Architecting: Creating & Building Complex Systems*, 1991, p21
- Every project should have exactly 1 identifiable architect,
 - For larger projects, principle architect should be backed up by architect team of modest size
 - Booch, *Object Solutions*, 1996

2




Software Architects

- Architect is “jack of all trades”
- Maintainer of system's conceptual integrity
- Part of team
 - Set of people with complementary skills
 - Committed to common
 - Purpose
 - Performance goals
 - Approach
 - Hold each other accountable
- Life of architect is long series of locally suboptimal decisions made partly in the dark
 - Sometimes painful

Philippe Kruchten: *The Software Architect*. WICSA 1999: 565-584


3



What Skills Are Needed To Be An Architect Or Team Member? “Jack of all trades”


<input type="checkbox"/> Software development expertise	<input type="checkbox"/> Leader
<input type="checkbox"/> Domain expertise	<input type="checkbox"/> Technologist
<input type="checkbox"/> Communicator <ul style="list-style-type: none"> – Author – Speaker 	<input type="checkbox"/> Cost estimator
<input type="checkbox"/> Strategist	<input type="checkbox"/> Cheerleader
<input type="checkbox"/> Consultant	<input type="checkbox"/> Politician
	<input type="checkbox"/> Salesperson

4

 Center For Software Engineering

What Skills Are Needed To Be An Architect Or Team Member? (cont.)


- May need different people & skills based on
 - Characteristics of project & domain
 - Lifecycle “phase”
 - Type architecture
 - Enterprise vs. Product-line vs. Product
 - Distinction between junior & senior architects
- Combined into a team
- Complementary strengths

 Center For Software Engineering

What Is Their Job Description?

- Some subset of above skills
- What architects are not usual project
 - Developers
 - Though they may prototype their ideas
 - Managers
 - Except in small organizations


6

 Center For Software Engineering

Architects As Software Development Experts

- Must understand nuances of software development
 - Principles
 - Methods & techniques
 - Methodologies
 - Tools
- Need not be world-class software programmers; but be familiar with methodologies used by developers
- Should understand ramifications of architectural choices
 - Do not live in ivory tower
 - Some architectural choices constrain implementation options
 - Some implementation-level techniques & tools constrain architectural choices

7

 Center For Software Engineering

Architects As Domain Experts

- Software engineering expertise is not enough
- Problem domain nuances
 - Maturity
 - Stability
 - System user profile
- May greatly affect selected & developed architectural solutions
 - Distribution
 - Scale
 - Evolvability
- Requires artifacts that model problem space
 - Not solution space

8

Center For Software Engineering

Team Needs Balance & Shared Vocabulary

- Architecture team Need
 - Software Engineering Expertise (S)
 - Domain Expertise (D)
- Rarely find both in 1 person
- Harder to achieve for large project
 - Usually need more specialties e.g.
 - For SW: OS, DB, Networking, Security

9

Center For Software Engineering

Architects As Communicators

- At least ½ the job
- Must
 - Listen to stakeholder concerns
 - Explain how architecture address problems
 - Negotiate compromises
- Need good communication skills
 - Write
 - Speaking
 - Presenting

10

Center For Software Engineering

Architects As Communicators Must Communicate With:

- Managers
 - Must relay key messages
 - Convince that architecture is useful & important
 - Ensure support throughout project
 - Must listen to concerns
 - Cost
 - Schedule
- Developers
 - Convince that architecture is effective
 - Justify local suboptimal choices
 - Listen to problems
 - Tools
 - Methods
 - Design choices
- Other software architects
 - Ensure conceptual integrity
 - Ensure desired system properties & evolution
- System engineers
 - Coordinate requirements & solutions
 - Explain how architecture addresses
- Customers
 - Determine needs
 - Explain how architecture addresses
- Users
 - Determine needs
 - Explain how architecture addresses
 - Listen to problems
- Marketers
 - Get/help set goals & directions
 - Explain how architecture addresses


11

Center For Software Engineering

Architects As Strategists

- Developing elegant architecture is not enough
 - Technology is only part of picture
 - Architecture must be right for organization
- Must fit organization's
 - Business strategy
 - Rationale behind it
 - Business practices
 - Planning cycles
 - Decision making processes
- Must also be aware of competitors'
 - Products
 - Strategies
 - Processes


12



Architects As Consultants

- Architects must recognize developers are their primary “customer”
- Developers
 - Goals do not match architects
 - Not focused on making architecture successful
 - Focused on
 - Satisfying
 - Functional
 - Quality
 - Scheduling requirements
 - Subsystems for which they are responsible


13



Architects As Consultants (cont.)

- Developers must be convinced to
 - Learn, adhere to, & effectively leverage architecture
 - Architects need to make these tasks reasonably easy
 - Document & report architecturally-relevant modifications
 - Architects need to make clear what’s architecturally-relevant
 - “where are load bearing walls”


14



Architects As Leaders

- Must be technical leader
 - Based on knowledge & achievement
 - Command respect by ideas, expertise, words, & actions
 - Cannot rely on position in org chart
 - Must do so without managerial authority
- Establishes
 - Structure for software
 - Design rules
- Ensures design rules are followed
- To improve productivity & quality, injects:
 - New ideas, solutions, & techniques
 - Mentor newcomers & junior people
- Make decisions & help assure their implementation
 - Enlist help of others in doing so


15



Architects As Technologists

- Understand software development approaches
 - e.g., object-oriented (OO) & component-based
- Understand fundamental technologies
 - Operating system/networking
 - Middleware
 - Security
 - Databases
 - Graphical user interface (GUI) toolkits
- Keep on top of trends
 - E.G., CORBA, COM/DCOM, .NET, JavaBeans, UML, XML, Web Services
- Demonstrated expertise in
 - System modeling
 - Architectural trade-off analysis
 - Tying architectural solutions to system requirements


16



Architects As Cost Estimators

- Must understand financial ramifications of architectural choices
 - Green-field vs. Brown-field development (new or reuse)
 - Cost of COTS adoption
 - Cost of development for reuse
 - Company's financial stability & position in marketplace
- Technologically superior solution is not always most appropriate one
 - Impact on cost & schedule
- Quick, approximate cost estimations are often sufficient
 - Detailed cost estimation techniques can be applied once set of candidate solutions is narrowed down

17



Architects As Cheerleaders

- Especially needed on long, large, complex projects
 - Development teams work in trenches on small subsets of project
 - Managers lose sight of overall project goals
 - Customers get impatient from long wait
- Must
 - Maintain high-level vision with necessary details sprinkled in
 - Convince different stakeholders of architecture's
 - Beauty
 - Utility
 - Adaptability
 - Technological impact
 - Financial impact
 - Keep the troops' morale high


18



Architects As Politicians

- Must get key organization players committed to architecture
- Must do a lot of influencing themselves
 - Find out who the key players are
 - Find out what they want
 - Find out the organization behind the organization
- Architects must continuously
 - Listen
 - Network
 - Articulate
 - Sell vision
 - See problem from multiple viewpoints


19



Architects as Salespeople

- For many of above reasons, architects must sell
 - Overall vision
 - Technological solutions
 - Key properties of architecture
 - Key properties of eventual system that architecture will directly enable
 - Cost/schedule profile
 - Importance of sticking to architecture
 - Penalties of deviating from it


20



Software Architecture Team

- Collection of software architects
- Typically stratified
- Team size fluctuates during life of project
 - 1 architect per 10 developers during project inception
 - 1 architect per 12-15 developers in later stages
- Architects may
 - Become subsystem development leads
 - Maintainers of grand vision on development team
 - Bridges to “central” architecture team for duration of project
 - Be shifted to other projects
 - After project’s architecture is sufficiently stable


21



Role of Architecture Team

- Define software architecture
- Maintain architectural integrity of software
- Assess technical risks associated with design
- Propose order & contents of development iterations
- Coordinate & coexist with other teams
- Assist in project management decisions
- Assist marketing in future product definition


22



Role Of Architecture Team Define Software Architecture

- Define
 - Major design elements
 - Organization/structure
 - Way major elements interact
- Works with system engineers & development teams

23



Role Of Architecture Team Maintain Architectural Integrity Of Software

- Develops & maintains guidelines for
 - Design
 - Programming
- Helps with reviews
 - Major role at end of iteration review
- Approves
 - Changes to interfaces of major components
 - Violations of guidelines
- Final arbiter on aesthetics
- Assists change control board resolve software problems or interfaces

24

Center For Software Engineering

Role Of Architecture Team Assess Technical Risks Associated With Design

- Maintains lists of perceived risks
- May propose exploratory studies or prototypes

25

Center For Software Engineering

Role Of Architecture Team Proposes Order & Contents Of Development Iterations

- Determines order & contents based on
 - Selected scenarios
 - Services to be studied & implemented
- Helps development teams transition from architecture design to more detailed design

26

Center For Software Engineering

Role Of Architecture Team Coordinate & Coexist With Other Teams

- No structural difference between architecture team & other teams
- Just focused on higher level issues

```

    graph TD
      Arch[Architecture team] -- "Architecture design, scenarios, & guidelines" --> DevA[Development Team A]
      Arch -- "Architecture design, scenarios, & guidelines" --> DevB[Development Team B]
      Arch -- "Architecture design, scenarios, & guidelines" --> DevC[Development Team C]
      DevA -- "Modules, subsystems, & tests" --> IntTest[Integration & Test Team]
      DevB -- "Modules, subsystems, & tests" --> IntTest
      DevC -- "Modules, subsystems, & tests" --> IntTest
      IntTest -- "feedback" --> Arch
      SM[Software Management]
  
```


27

Center For Software Engineering

Pitfalls of Software Architect Teams

- Imbalance of skills
 - Lack of software development experience
 - Lack of domain expertise
- Lack of authority
 - Team acts as committee
- Life in ivory tower
- Confusing tools/techniques/methodologies with architectures
- Procrastination

28



Pitfall: Lack Of Authority


□ Problem:

- What incentive for group leaders to
 - Follow recommendations of architecture team
 - Report progress or problems to architecture team
- Architect team
 - Frequently has no explicit authority
 - Architects are not managers
 - Just another team in organization
- Problem compounded when external architect or architecture team is hired

□ Solution:

- Must influence based on skills & experience
- Must communicate

29



Pitfall: Life In Ivory Tower


□ Problem:

- Developers & managers must be aware of architecture team's existence & role

□ Solution:

- Team must continuously communicate with rest of personnel
- Team must be co-located with rest of project personnel
- Do not use team as retirement home for ageing developers
- Architecture team must recognize & adjust to organizational realities
 - Technological base
 - Personnel issues
 - Organizational politics
 - Market pressures

30



Pitfall: Imbalance Of Skills


□ Problem:

- Predominant expertise in one area creates imbalance
 - Database
 - GUI
 - Networking
 - Systems
- Imbalance may affect how architecture is
 - Designed
 - Communicated
 - Evolved

□ Solution:

- Balanced architecture team appropriate to
 - Project type & size
 - Problem domain
 - Personnel

31



Pitfall: Confusing Tools With Architectures


□ Problem:

- Common pitfall
- Usual culprits
 - Databases
 - GUI
 - Case tools
- More recently culprit is middleware
 - "Our architecture is CORBA"
- Tools tend to influence architecture
- Become basis on which architecture is built

□ Solution:

- Balanced architecture team


32



Pitfall: Procrastination

- ❑ Problem:
 - Incomplete & changing information yields indecision
 - Curse of software architects
 - Architects' indecision impacts other teams
 - Domino effect
 - May paralyze entire project


33



Pitfall: Procrastination (cont.)

- ❑ Solution:
 - Often better to make a decision than suspend project
 - Make educated guesses
 - Document rationale for decision
 - Document known consequences
 - Change decision if/when better alternatives present themselves
 - Be decisive
 - Being effective architect demands rapidly making tactical decisions & living with resulting anxiety
 - Suboptimal decisions based on incomplete information

34



Summary

- ❑ Designate architect or assemble architecture team to be creators & proponents of common system goal/vision
- ❑ Architects must be experienced at least in problem domain & software development
- ❑ Software architect is full-time job
- ❑ Charter of software architecture team should
 - Clearly define its roles & responsibilities
 - Clearly specify its authority
- ❑ Do not isolate software architecture team from rest of project personnel
- ❑ Avoid pitfalls

35