

On the Meaning of Software Architecture

Interview Questionnaire

July, 2004

I. Goals

- To develop a deep understanding of what architects do
 - i. What they think architecture is
 - ii. How they think about it
 - iii. How they relate it to requirements, detailed design and implementation
 - iv. How they go about creating and evolving it
- To capture the meaning of software architecture in practice

II. Privacy issues

- Anonymity (of both architects and company) is guaranteed, unless explicit permission is provided by the interviewee and/or his/her company
- Company confidentiality will be maintained on architecting process issues if requested.
- Company confidentiality will be maintained on IP issues, unless explicit permission is provided by the interviewee and/or his/her company

III. Questionnaire

- Note that the questions below may have overlaps, please feel free to repeat your answer if appropriate.
- The order of the questions can be altered according to the architect's preference.
- Questions may also be skipped.

A. Problem Domain

A1. Describe the problem domain(s) that you have been working in and most familiar with.

A2. What are the characteristics in each domain?

- Is there a well defined and established business process behind the problem for which the software is to provide automation?
- Is this a new business opportunity where the software is to be used by a group of potential users?
- Are the requirements well defined/documented and well known, or are they hypothetical and uncertain?

B. Overview of Software Architecture

This section is used to capture the architect's understanding of software architecture and its meaning in the development cycle, to the project, and during (or perhaps after) the lifetime of the system.

B1. What do you consider to be the salient characteristics of software architecture?

- How detailed should it be?
- Should it be prescriptive or descriptive?

B2. What drives the architecture?

- Why do you need an architecture?
- What are the driving forces in its creation?
 - Is it the non-functional requirements,
 - the functional requirements,
 - the particular blend/mix of both,
 - the organization's culture,
 - or business needs?

B3. What is the meaning of architecture in your opinion?

Note this is not the same as the definition of architecture. The meaning is more from a philosophical standpoint.

- Can you elaborate the meaning in the context of software construction?
 - in relation to requirements (functional and non-functional), and
 - with respect to the impact to the project and the system in its lifetime.

B4. Given all of above discussions, how do you view software architecture?

- Is an architecture a technical need,
- or a business need?

C. About Requirements

This section helps us to understand the architect's opinion in the relation of the requirements and the architecture.

C1. How do you view requirements?

- How do you distinguish functional and non-functional requirements?
- How do you deal with inconsistent requirements?

C2. What do you think of “architecting requirements”?

In the traditional Waterfall model, architecture is part of the Design Phase. “Architecting requirements” means to do the architectural work in the requirement (problem) space?

- Is it sensible?
- What do you think that means?
- Do you already do some of that? What are they?
- What would you need before you can move towards that direction in terms of methods and tools?

C3. How do you identify requirement variance?

- What environment factors help you to determine the variance?
- How do you decide whether the supplied requirements are the “true” requirements or not?
 - Whether it is too narrowly defined,
 - too open and unspecific,
 - with hidden assumptions,

C4. How do you transform the functional requirements into an architecture?

- Do non-functional requirements play a role in this transformation?
- Any examples?

C5. How do you handle the non-functional requirements?

- Do you integrate these with the functional ones initially or after you have considered the functional ones and built a skeleton architecture?
- How do the functional and non-functional aspects interplay in the design of an architecture? Is there an ordering or a set of priorities for non-functional requirements?

C6. Do you have domain-specific standard or generic styles, structures, patterns, transformations, or techniques for non-functional requirements?

- Do you have preferred ones and why? Examples?
- For example, there are standard techniques used in telephone switches - eg, watch dog timers which are used for certain kinds of reliability and fault diagnosis techniques.

C7. While designing the architecture, what do you look for in the requirements?

- Do you try to reflect the unique characteristics of the problem in the architecture?
- What are some examples?

C8. In designing an architecture, what do you do with requirements that you know will entail an overly complex or costly system?

C9. How do you evaluate the architectures?

- Is there a formal evaluation process? How does it work?

D. Relating to Evolution

D1. How do new requirements affect the architecture after the system is built?

D2. How do you handle continuous requirements change?

D3. How does the architecture evolve during the system's lifetime in response to changing requirements?

- How do you deal with architectural drift or erosion?

D4. What measures do you take while designing the architecture to minimize the impact of change?

- How do you do to identify and understand the various effects of requirements changes?

D5. How do you reuse an architecture?

- How do you make an architecture reusable?
- Are you usually concerned with reusability while designing?
- Do you make use of product line architectures?
- Do find common parts that you can reuse?

E. Professional Background

To gain a basic understanding of the architect's professional experience in architectural design and software development.

E1. Describe your overall professional architecting experience.

- Duration, number of projects, size of projects, budget, success/failure etc.

E2. Describe the architectural (and/or requirements) aspects of one successful project that has left a profound impression on you.

- What played a critical role in success/failure?

E3. Repeat A.2 for a failed or not so successful project.

E4. Have you had a successful project where you felt the architecture was not a good one?

E5. What about the reverse: unsuccessful with a good architecture?

F. Comments

F1. What do you think of the questions?

- Are they relevant?
- Do they help you to think differently of architecture compare to before?
- Do you find this interview a useful exercise? Why and how?

F2. Do you have any recommended architects who might be interested in the interview?

- Names and contact info.