Responsibilities of a Software Architect

This article describes the myriad responsibilities of a software architect, and attempts to identify human personality traits that naturally aid a person in such position.

Introduction

A software architect is responsible for creating or selecting the most appropriate architecture for a system (or systems), such that it suits the business needs, satisfies user requirements, and achieves the desired results under given constraints. This article describes the myriad responsibilities of a software architect, and attempts to identify human personality traits that naturally aid a person in such position.

Architect's Responsibilities

An architect abstracts the complexity of a system into a manageable model that describes the essence of a system by exposing important details and significant constraints.

An architect maintains control over the architecture lifecycle parallel to the project's software development lifecycle. Although an architect may be most visible during the requirements and design stages of a project lifecycle, he or she must proactively monitor the adherence of the implementation to the chosen architecture during all iterations. Architecture on paper is fruitless unless implemented proficiently.

An architect stays on course in line with the long term vision when projects' scope creep attempts to manipulate software architecture in a certain way in order to satisfy the desires of myriad stakeholders. An architect must focus on actions that produce results early while staying on course for the long term. When project variables outside of one's control change the architect must adjust the strategy given the resource available while maintaining the long term goal.

An architect progressively makes critical decisions that define a specific direction for a system in terms of implementation, operations, and maintenance. The critical decisions must be faithfully made and backed up by understanding and evaluation of alternative options. These decisions usually result in tradeoffs that principally define characteristics of a system. Additionally these decisions must be well documented in a manner understood by others.

An architect sets quantifiable objectives that encapsulate quality attributes of a system. The fitness of the architecture is measured against set marks.

An architect works closely with executives to explain the benefits and justify the investment in software architectures. This may be done by participating in business process re-engineering activities, by using Cost Benefit Analysis Method, or by measuring the level of component / architecture re-use between projects with the help from the software process improvement team. Software architect must be effective in order to deliver results that are meaningful to the projects that have an impact on the bottom line that result in greater profits.

An architect inspires, mentors, and encourages colleagues to apply intelligently customized industry's best practices. Educating the recipients and participants of system architecture is essential to successfully selling the chosen architectural path. Specifically the stakeholders must be able to understand, evaluate, and reason about software architecture. If an architect is the only one who can read and understand documented system architecture, then he has failed to integrate his best practices into the organizational culture.

An architect fights entropy that threatens architect's structural approach to problem solving. It's an architect's job to keep the inertia going once the project is in progress. He or she must convince all relevant stakeholders that the chosen approach is sound – moreover the chosen architectural solution must be well explained and justified. The benefits of implementing a system in a particular way must be explained not only in terms of “that’s the right pattern for this problem,” but also to demonstrate the measurable benefits - such as easier integration. For example, in a product line approach an architect must be able to demonstrate how the subsequent projects will be easier to implement due to the presence of a common base from which subsequent work can be done.

An architect creates and distributes tailored views of software architectures to appropriate stakeholders at appropriate intervals. For example, a customer may demand to become more involved with a project and they may need to know an abstract view of a system on the level understood by them. A government customer may require an architect to demonstrate early in the project how a given system meets High Level Architecture requirements for a specific framework. It’s the architect’s responsibility to identify and present a sufficient level of information that a customer needs.

An architect acts as an agent of change in organizations where process maturity is not sufficient for creating and maintaining architecture centric development. If the concept of software architecture is not well recognized in an organization it may be a “tough” sell to formally recognize the role of software architecture in a SDLC. Without senior management commitment and without mature software development process, architecture of the system on paper may not reflect the actual architecture of a system.

Architect’s Personality and Other Traits

No empirical studies have been done to determine the best character traits that define a successful architect. But it’s reasonable to derive the following traits based on the duties of an architect.

An architect is a human filter that process complexities and outputs an abstract high level model of a system. Conveying the output to the stakeholders requires excellent communication skills – written, verbal, and presentational.

An architect is a negotiator. The method of principled negotiation should be the tactic of choice for an architect. This method is most suitable in contrast to soft or hard negotiation method, because it seeks mutual cooperation between an architect and project stakeholders. An architect will be expected to deliver better, faster, and cheaper, but since only two-way combo can be selected an architect must negotiate to decide which aspects of a system will be considered first and under what conditions.

An architect must convey a sense of credibility and trust; an architect must be perceived as successful. An architect can attain such status with his prior successful experience, formal training in the field (certifications in the future), and by his or her ability to deliver successful and relevant architectural artifacts through every stage of the SDLC.

An architect believes in his ability to perform well. In a leadership position attitude is everything – if the passion for success is absent, then an architect must step down from the leadership pedestal.

An architect must be patient and resilient, as the only thing constant is the change itself. Since software architecture has direct influence on the quality characteristics of a system, an architect will interact with a great number of people with a full spectrum of personalities. He or she must quickly adapt to the way stakeholders operate, as it’s not possible or feasible to expect them to speak the language of an architect.

Afterthought

In order to be effective, an architect must be familiar with the business domain at hand so that solutions crafted are practical and less academic. At the same time an architect must stay in touch with the rapid evolution of the field as the discipline grows towards becoming a true engineering discipline. New methodologies, practices, and vendor tools are re-defining, again and again, the responsibilities and duties of an architect. Proactive participation and involvement in the software architecture community in is a duty of every architect.

Links

What are the Duties of a Chief Software Architect? http://www.sei.cmu.edu/ata/arch_duties.html

Human Resources' view of this role (Monster).
http://chief.architect.jobs.monster.com/

Odd Twist on the Title
http://www.computerbits.com/archive/2000/0300/whatname.html