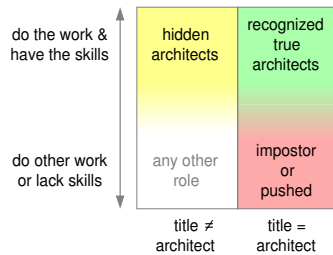


Systems Titles and Roles



Gerrit Muller

Buskerud University College

Frogs vei 41 P.O. Box 235, NO-3603 Kongsberg Norway

gaudisite@gmail.com

This paper has been integrated in the book "Systems Architecting: A Business Perspective", <http://www.gaudisite.nl/SABP.html>, published by CRC Press in 2011.

Abstract

Many different titles are used for systems related roles. Examples are systems architect, engineer, designer, tester, or manager. We discuss in this paper these roles and titles. We also explain the context sensitivity of these titles.

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

All Gaudí documents are available at:
<http://www.gaudisite.nl/>

version: 0

status: draft

October 20, 2017

1 Introduction

The following questions are asked frequently during and after the courses:

- What is the difference between *systems engineers* and *systems architects*?
- Why do all these people have the title systems architect, while they actually don't do the work?

The first questions is also posed in other variants, using titles such as *system designer* or *systems manager*. To complicate matters more there are people who do part of the systems level work, for example *requirements analyst*, *systems analyst*, *system integrator* or *system tester*, complementing the systems architect.

2 Cultural differences in terms

Exactly the same titles are used differently in different companies (or even divisions or product groups within one company), in different domains (e.g. defense, automotive, consumer electronics, IT), and geographic regions. No single unified standardized definition is used across companies, domains, and geographies. We do recommend to *calibrate* terminology when entering new territory, and to be continuously alert for differences in interpretation even after calibration.

Throughout this book we use the term architecture for the combination of two crucial aspects:

Depth understanding of the system-of-interest including product specification, decomposition in subsystems and components, interface management, and function and resource allocation, to create a sound and fitting system that fulfills all qualities (e.g. safety, reliability, performance).

Breadth understanding of the context including the customer context and the stakeholders in the value chain, and the life cycle context from conception to decommissioning and all related business aspects.

Be aware that the term architect is used often for the *system-of-interest* part only. We use the term *system design* for this subset of architecting work.

A major professional society in the *systems* world is INCOSE, the International Council of Systems Engineering. The *Systems Engineer* as depicted by most of INCOSE documents is a very broad function, including work of the project leader, the requirements analyst, the systems architect, the configuration manager, and quality assurance.

Another extreme for the definition of systems engineer was in the medical domain, where this job of the systems engineer was solely the electro-mechanical design of cables and cabinets.

3 Title versus skills and actual job

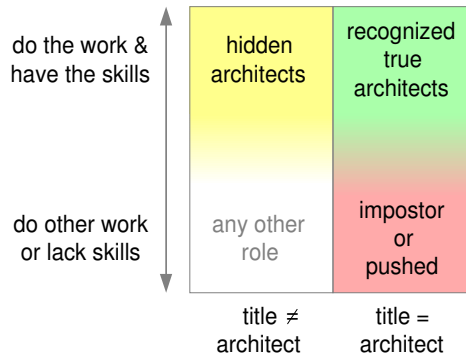


Figure 1: Four quadrants to classify architect and titles

First of all we have to distinguish what role or function someone performs and the title that is being used by the people in the context. Figure 1 shows the four quadrants that you get by using title as horizontal axis and competence level as vertical axis. Note that the title axis is discrete, while the competence level is continuous. The figure shows four quadrants:

Any other role (bottom left) for these persons that don't do the job of an architect and do not have the title.

Imposters or pushed persons (bottom right) are people lacking the skills or actually not doing the work of an architect that nevertheless have the title architect. Note that impostors are these people that actually pursued the title, for example because of status or payment. Another category are those people that are pushed by their management into this job, but lack the capability to do it. People do not become true architects by declaration.

Hidden architects (top left) can be found in many organizations. These organizations might use different titles or they might not be aware of the *systems* discipline.

Recognized true architects (top right) are these architects that actually do the architecting job skillfully and got the title in recognition.

4 Systems roles and titles

In this section we provide a set of roles and relate these to the development life cycles. As explained in the previous sections these roles can be allocated in different

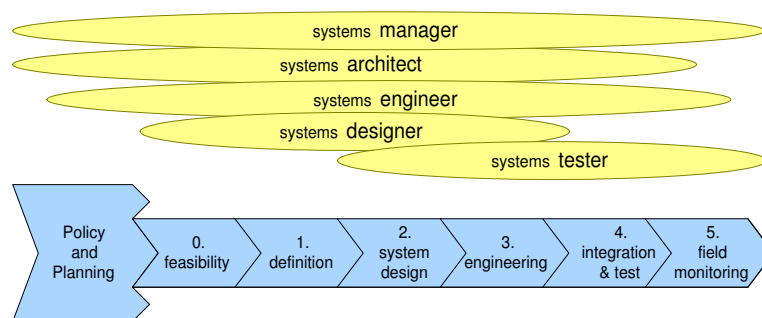


Figure 2: System Roles mapped on the development life cycle

ways and different terms can be used than shown here. However, the conceptual roles as shown here are quite universal.

Figure 2 shows the following roles:

Systems manager is the overall responsible for all systems aspects, ranging from strategically positioning in the portfolio and the time to final operational performance in the field. Note that such broad definition does not leave much room for in-depth understanding. An alternate term for this role can be *program manager*.

Systems architect who combines understanding of the context with in depth understanding of the solution to create an appropriate system. Note that the architect role combines some perceptive and creative modes of operation with more analytical modes. This mixture limits how far the architect can go in the real engineering.

Systems engineer is very close to the systems architect, but the emphasis shifts from perceptive and creative more to engineering. With engineering we mean the capability to finalize and document all details required for the later processes such as logistics, manufacturing, sales, and customer support. Note that also the systems engineer has limits and will depend on specialized engineers (e.g. mechanical, electrical, or software) to finish the last details of the technical product documentation.

Systems designers take the product specification as starting point and work on (potential) solutions. System designers are “inward” focused, where system architects connect the *outward* and *inward* perspectives.

System testers verify that the solution performs as specified. In practice system testers need also trouble shooting capabilities to diagnose the cause of lacking performance.

References

- [1] Gerrit Muller. The system architecture homepage. <http://www.gaudisite.nl/index.html>, 1999.

History

Version: 0, date: June 25, 2010 changed by: Gerrit Muller

- Created, no changelog yet