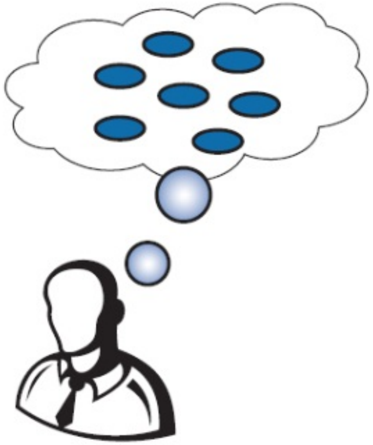
Kai Schwidder

Distinguished IT Architect

Enterprise IT Architectures

Solution IT Architectures – Primer



Learning objectives of this lecture

- The skills you need to be an IT Architect
  - **Judge**
  - **Visualize**
  - **Communicate**
- This will be illustrated through the three “Cs”:
  - **Context**
  - **Common Sense**
  - **Communication**

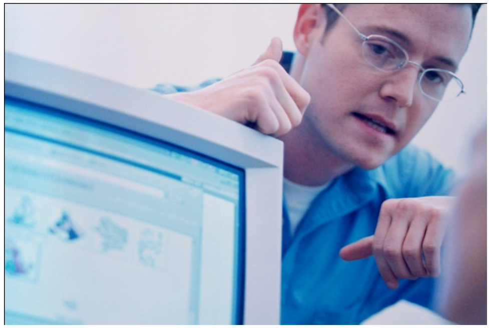
Characteristics of an IT Architect

❖ Creates workable concepts

- ○ Accountable for the integrity of the customer solution
- ○ Recognised lead technical authority
- ○ Able to architect full solution although focus will vary due to differing background and experience
- ○ Capable of delving into detail when required
- ○ Full life cycle, i.e. from concept, through development & roll-out to support/managed service
- ○ Strong leadership, technical & communication skills
- ○ Ability and willingness to mentor and coach

❖ Overall, the architect is reducing:

- ○ Complexity
- ○ Uncertainty
- ○ Ambiguity



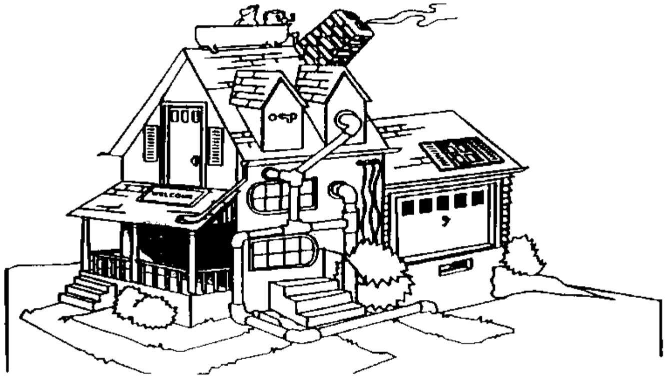
The fundamental skill of an IT Architect is to understand the context they are working in

❖Putting IT architecture in context itself – why do we need it?

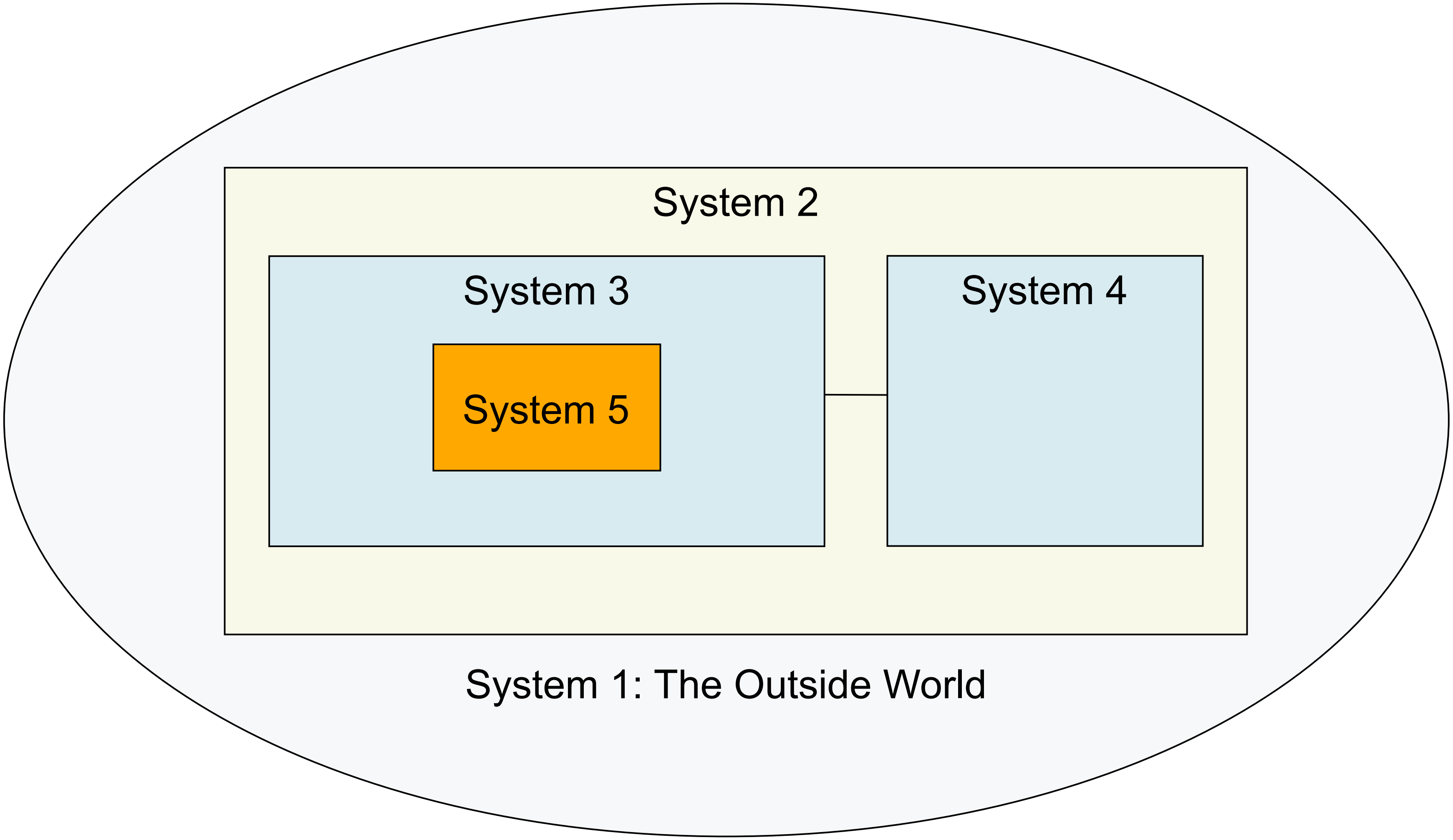
○ System context and boundaries

○ Architecture context – aspects of arch

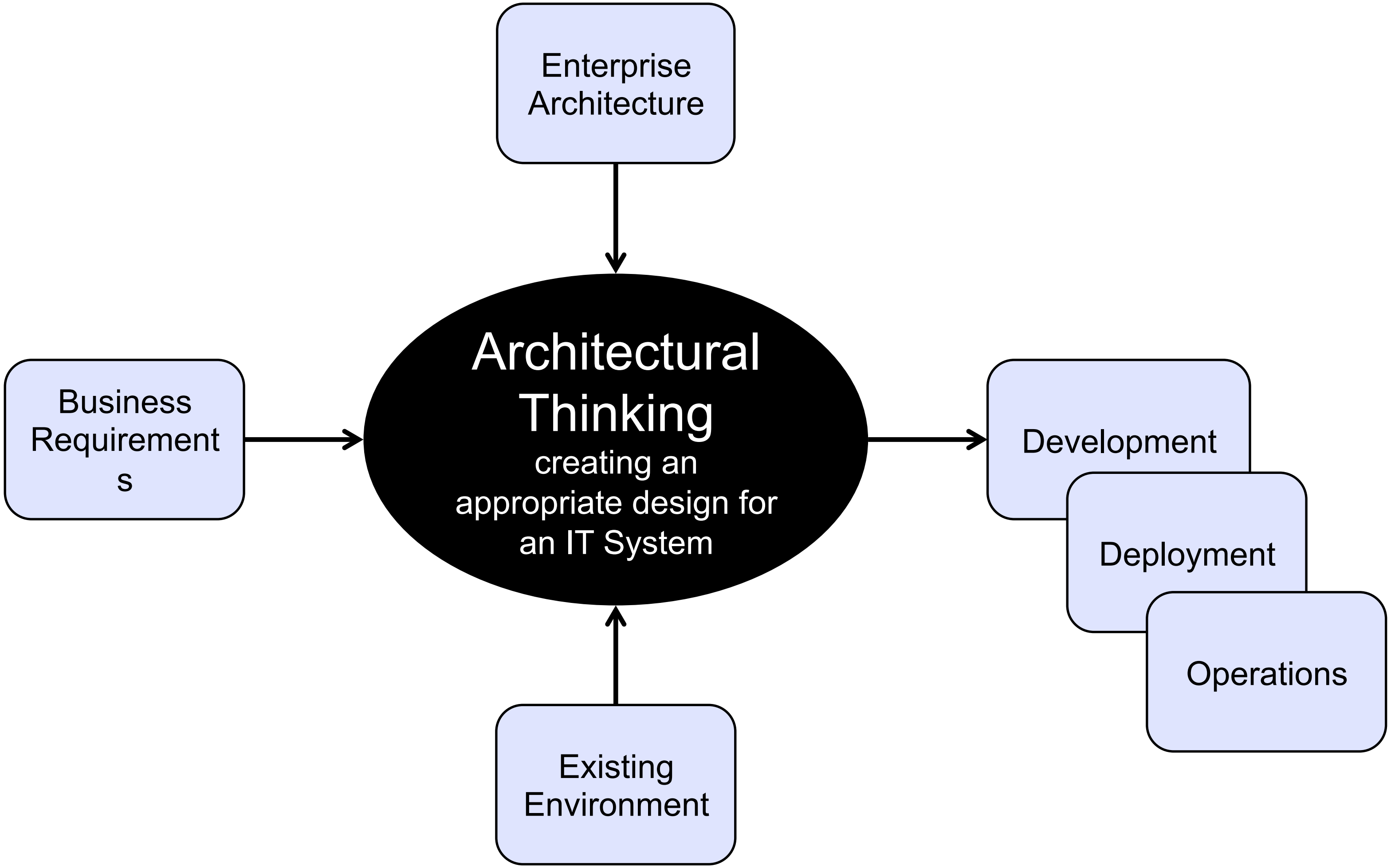
○ Project context



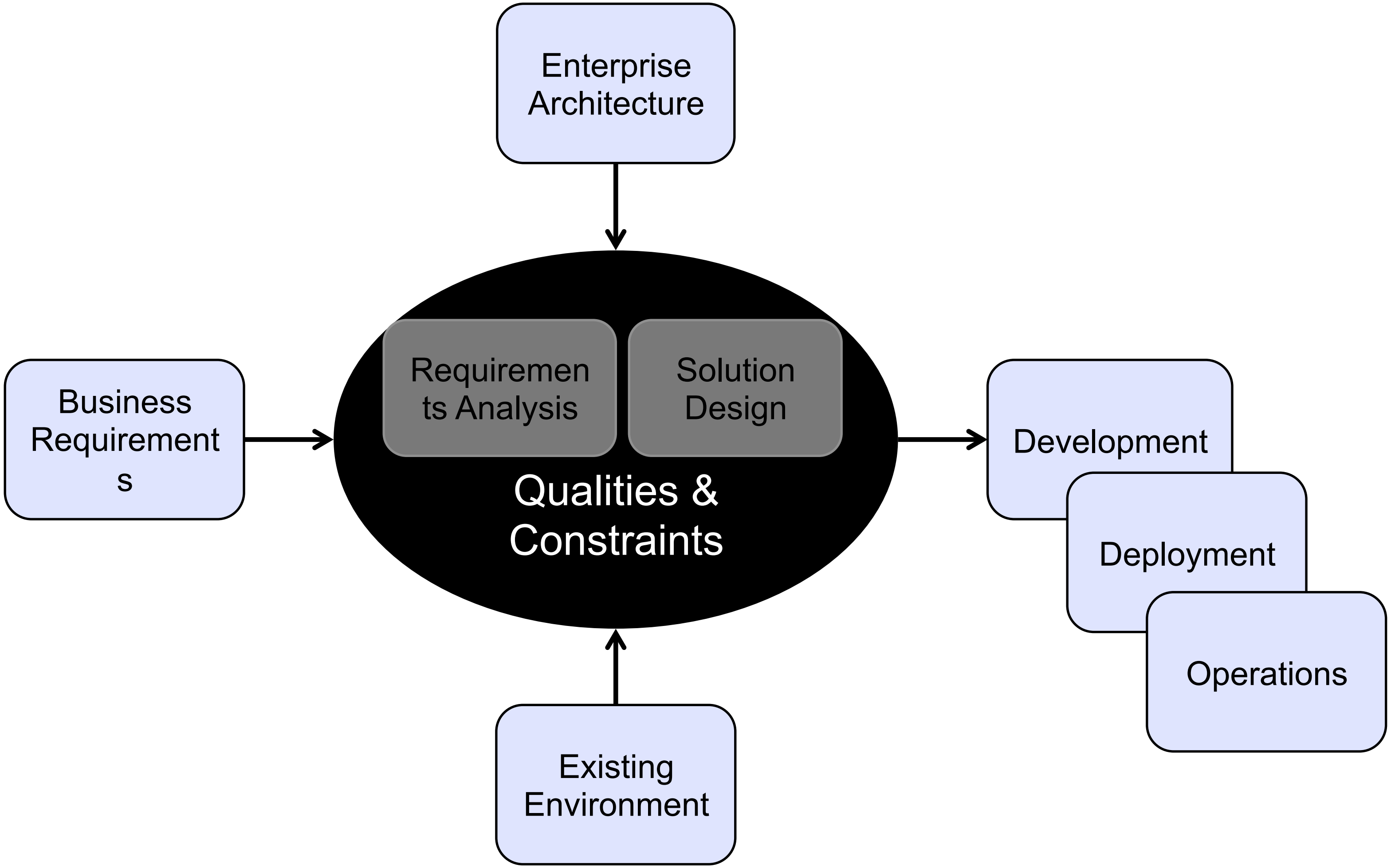
Finding the system boundaries



Architectural Thinking : Balancing many forces



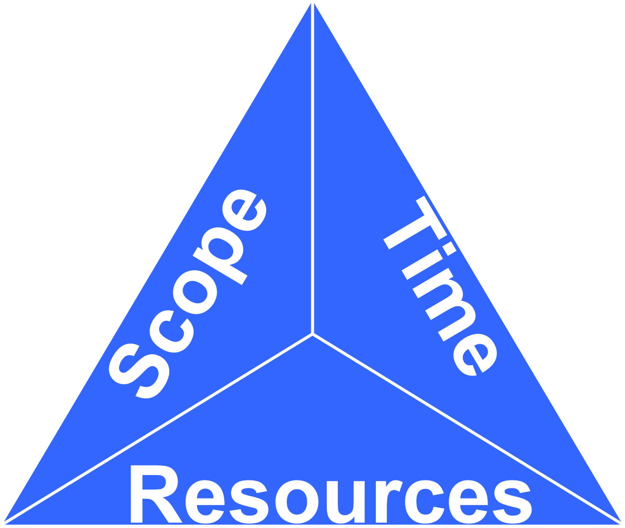
The IT Architect focuses on the following aspects



Architectural decisions are made in context of the overall management of the project

- **You need to understand your project's context:**
  - ○ Be the project manager's best friend
  - ○ Consider cost and budget implications

- **How much will it cost?**
  - ○ Is it built to order?
  - ○ Is it built to cost?

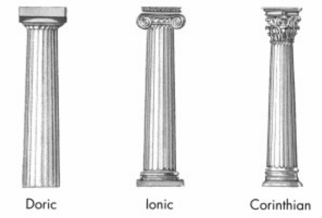


Changing one side will always have an affect on the other two sides.

The Process of Architecting

The Normative way

- Schools of architecture
- Dogma



The Rational way

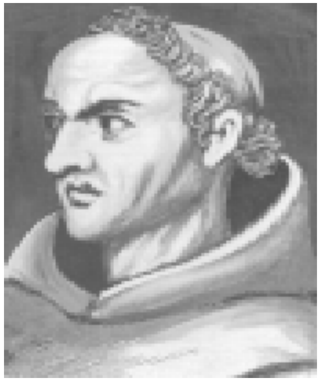
- Procedure driven
- Logical

The Argumentative way

- Mechanistic
- Brainstorming

Just use common sense (heuristic reasoning)

- The knowledge of what is reasonable within a given context.
  - **Includes insights, lessons learned and rules of thumb**
- Heuristics can be prescriptive…
  - **Keep It Simple, Stupid**
- …or descriptive
  - **If anything can go wrong, it will.**



➢ ***The simplest solution is usually the correct one.***

And be aware of the following …

❖Requirements

- Don’t assume the original statement of the problem is necessarily the best, or even the right, one.
- Success is defined by the user, not the architect.

❖Design

- You can’t avoid redesign, or if first you don’t succeed…
- No system can be optimum for all users (or all database accesses!)

❖ Development

- Quality cannot be tested in, it has to be built in.
- Something good enough in a small system is unlikely to be good enough in a complex one.

❖ Test

- Testing is a system in itself.
- Regardless of everything, the acceptance criteria determine what gets built.
- The sooner you find the problem, the cheaper it is to fix it.

Why do we need a Method to communicate ?

- ○ Provides a mechanism to enable a common language among all practitioners delivering business solutions

- ○ Fundamental component to accelerating the shift to asset based services, providing a mechanism for practitioners to reuse knowledge and assets using a consistent, integrated approach

- ○ Shifting from labour based to asset based services to compete more effectively in the marketplace by increasing productivity and minimizing cost, risk, and time to market

The weapons of the IT Architect…

The Box

The Line

As with all weapons, they need to be used carefully to stop you getting hurt.

Communication skills are important for a successful IT Architect

❖Spoken Communication

- ○ The IT Architect is often asked to present their solution
- ○ Different audiences require different approaches
  Business sponsor, project manager, developer…

❖Written Communication

- ○ An undocumented architecture is difficult to build and maintain; it leads to a lot of repeat discussions.
- ○ Architectural documentation can be a time-saver, not a time-waster.
- ○ Documentation is extracted from work products, but focuses on key messages and stakeholder needs

Do not assume ! Ask and question whenever possible to reduce your risk

WHY vs. RISK

Learning Points

○ Context, Common Sense and Communication

○ Remember there are different ways of producing an IT Architecture and learn when to use the appropriate technique.

○ Use methods, modelling languages and design tools to enable re-use of IT Architecture assets.

○ Patterns are important. Don’t assume that your problem is “first-of-a-kind”. Use patterns to break down the complexity of your problem.

Questions