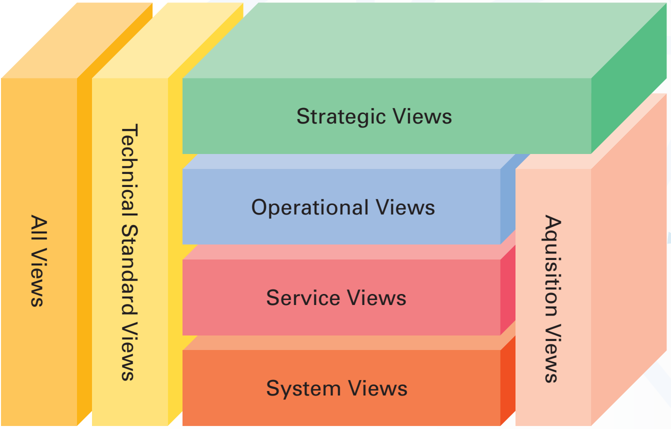
Architecture Frameworks

In recent years there have been several significant attempts to provide assistance to architects, particularly in relation to what kind of system descriptions might be relevant for system architecture. This assistance is usually expressed through Architecture Frameworks, which serve to structure and organise architectural descriptions. (Other techniques and tools are relevant in the 'implementation domain' but cannot be covered here.)

Architecture Frameworks define a set of viewpoints on to an underlying architecture and they may be supported by a metamodel (a precise definition of what can be described in the architecture through each of the viewpoints). The emerging international standard ISO/IEC 42010 addresses architecture descriptions and architecture frameworks.

Examples of widely used architecture frameworks include: Zachman, MODAF, DoDAF and TOGAF (see Page 6). Note that some Architecture Frameworks focus more on the architecting process and others focus more on architecture products.

A visual representation of types of views in MODAF is shown below:



Further information

There are many resources covering topics in architecture. Here are a few you might find useful:

- The INCOSE UK Chapter's Architecture Working Group wiki pages: www.ukawg.org
- The UK Ministry of Defence's Architecture Framework (MODAF): www.mod.uk/modaf
- The Open Group's Architecture Framework (TOGAF): www.opengroup.org/togaf
- The ISO/IEC standard on Architecture Descriptions (ISO/IEC 42010): www.iso.org
- The UK Department for Transport's Architecture Framework (TRAK), originally created by London Underground: http://trak-community.org
- The US Department of Defense's Architecture Framework (DoDAF): http://cio-nii.defense.gov/sites/dodaf20
- The original IT-inspired Architecture Framework from John Zachman: www.zifa.com
- The book by Maier and Rechtin gives a particular perspective on architecture: M. W. Maier and E. Rechtin. 2009. The Art of Systems Architecting. ISBN:978-1420079135

This leaflet

This leaflet is intended to be a simple introductory guide to system architecture and architecting, particularly as practised in the UK. It is one of a series of introductory guides produced by the INCOSE UK Chapter.

For further information, advice and links to helpful websites go to: www.incoseonline.org.uk
Download copies of this leaflet and other Systems Engineering resources online at: www.incoseonline.org.uk
For more information about the worldwide Systems Engineering professional community, go to www.incose.org

Series editor: hazel.woodcock@uk.ibm.com
Z8 lead author: Mike Wilkinson (Atkins)

INCOSEUK @incoseuk

Z8 Issue 1.0 October 2010

Z8
Issue 1.0
October 2010

System Architecture

What is system architecture?

Architecture is a popular and evidently useful concept, with many practical benefits (see Page 4) - unfortunately for the novice and the unwary there are many different interpretations in widespread use.

Drawing on a variety of such interpretations, the following is our summary definition that captures the majority of the common ground:

"The architecture of a system is its *fundamental structure* – which may include principles applying to the structure as well as specific structures."

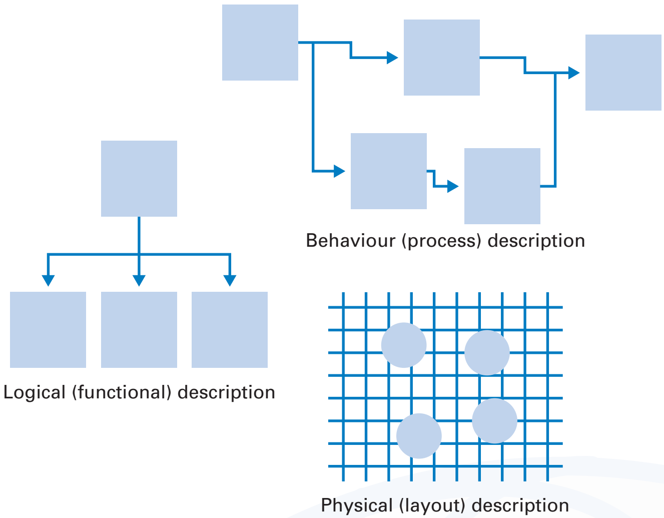
Some authors broaden the definition of architecture to include, for example, principles associated with the realisation of the system (how it is implemented) or governing its evolution over time.

What fundamental means in practice is found to be context dependent. The identification of fundamental types of structure usually depends on the nature of the system as well as on the purpose of the architecture. Structure that is judged not to be fundamental should be excluded from the architecture.

The architecture of a system is its *fundamental structure*

What is architectural structure?

A typical system architecture will not usually be described by a single type of structure – it is likely to include logical (functional) structure, behaviour (process) structure, physical structure and potentially other types of structure (eg financial or commercial). What needs to be included depends on what is judged to be fundamental, as discussed earlier.


Behaviour (process) description

Logical (functional) description

Physical (layout) description

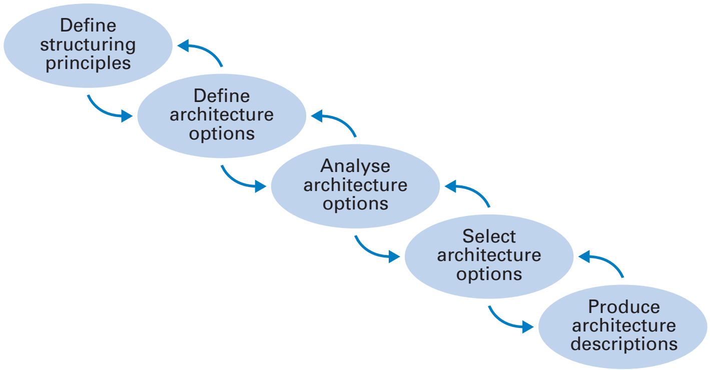
Illustrative architecture elements

What is system architecting?

System architecting is simply the process of creating (and describing) a system architecture, which we regard as a process within Systems Engineering (see Page 4). It may be motivated by a variety of factors; for example, ‘forward architecting’ aims to provide a basis for more detailed design, whereas ‘reverse architecting’ captures an existing system architecture for analysis.

Architecting can be more or less systematic but typically involves: understanding context; exploration of alternatives; understanding trades; supporting decision making; and so on. An illustrative architecting process is shown on Page 3 – in practice the precise form of the architecting process will depend on the purpose and context of the architecture.

There are many good guiding principles for architecting, including modularity, high cohesion, loose coupling, etc. As an overall principle, we paraphrase Saint-Exupéry: *“An architect knows he has achieved perfection not when there is nothing left to add, but when there is nothing left to take away.”*


Define structuring principles
Define architecture options
Analyse architecture options
Select architecture options
Produce architecture descriptions

Illustrative architecting process

System architecting is the process of creating a system architecture

What are the different types of architecture?

There are many types of architecture in use, each of which may be focussed on a particular topic of interest, or on a specific purpose, or on a specific set of systems.

Some examples of architectures with a focus on specific topics are: Operational; Programme; Security; Information.

Some examples of architectures with specific purposes are: Integration; Problem Domain Definition; Design-Controlling.

Some examples of architectures addressing a specific set of systems are: System of Systems; Product Family; Enterprise.

Some practitioners regard each type of architecture as a viewpoint on to a single underlying architecture (ie: a single system could have a security architecture and an information architecture, etc) – more on this in the section on Architecture Frameworks (see Page 5).

What is the role of architecture?

In use, architectural descriptions will usually have a primary role (or purpose) and a multitude of secondary ones. Some examples are:

- Acting as a vehicle for communications between stakeholders;
- Establishing context;
- Capturing key business and technical concepts;
- Providing a decision/trade framework;
- Providing guidance to the creation of detailed designs;
- Handling complexity and uncertainty;
- Supporting re-use of system elements;
- Supporting extension, enhancement or scaling;
- Dealing with transformation or evolution of systems over time.

A well crafted architecture should deliver the desirable outcomes (benefits) associated with each of its primary and secondary roles.

How is architecting related to Systems Engineering?

There are many and diverse beliefs about the answer to this question.

Some system architects regard Architecting as a subset of Systems Engineering, whereas others – particularly those from the IT world and those involved in applying systems thinking at the business level – tend to regard Architecting as providing something missing from Systems Engineering, which they see as applying only at the physical level.

The answer to the question clearly depends on how one defines Systems Engineering. In the UK, Systems Engineering has historically been considered broadly, applying at all levels and embracing both synthetic and analytical methods. Hence, we advocate the view that Architecting is best regarded as a subset of a broadly drawn Systems Engineering.