Information Systems Architecture and Enterprise Modeling

Prof. Dr. Knut Hinkelmann
Chapter 1: Introduction to Enterprise Architecture

- Motivation: Business IT Alignment
  - Challenge: Agility

- Approach Enterprise Architecture
  - Transparency

- Enterprise Architecture Frameworks
A Common Situation

Heterogenous and complex IT landscapes: patchwork of systems, processes, technologies etc.

(Hanschke 2010, p. 1f)
Problem: Alignment of Business and Information Technology (IT)

- Almost all processes have become IT reliant, if not fully automated

- The alignment of business and IT has to deal with problems like the following:
  - What happens to IT if the company has to react on market requirements?
  - What IT innovations are needed to remain competitive?
  - How do changes in the IT affect the business?
Alignment of Business and IT

- Mutual alignment
  - Business organization is able to use information technology effectively to achieve business objectives
  - Information Technology is enabler for (new) business models

- Business Perspective
- Information Technology Perspective

requirements

enabler
Drivers for Business and IT Alignment

Alignment of Business and IT can have internal and external reasons, e.g.

- Internal Drivers
  - Business Process Management / Optimisation
  - Reorganisation
  - Migration of Information Systems
  - Changes in IT infrastructure

- External Drivers
  - Pressures from
    - customers (new integrated services, individual products, …)
    - suppliers and other business partners
    - regulatory bodies (e.g. SOX, Basel II, and laws in general)
  - Market Opportunities, new business models
  - Innovations
Challenges confronting an Enterprise

Mutual Dependencies between Business and IT

- Business and IT alignment can start
  - top-down: Business defines requirements for IT
  - bottom-up: Changes in IT effect the way of making business

- Change in the enterprise is usually a compromise, for example
  - business requirements cannot be fully satisfied, because
    - there are already systems available that cannot be replaced (reasons can be costs or other dependencies)
    - standards set by IT strategy avoid unmanageable varieties and ensure reliability
    - centralisation reduces costs at the expense of specialisation
  - Chances of IT innovations cannot be implemented, because of
    - missing skills of employees
    - business processes or organisation are not appropriate
    - incompatibility with business strategy
Strategic Alignment Model of Henderson and Venkatraman (1993)

- Four dominant perspectives to tackle alignment (see figure)
- Two dimensions
  - Functional Integration: Aligning business and IT
  - Strategic Fit: Aligning internal and external drivers
- Two principle approaches for alignment:
  - top-down: take the business strategy as the starting point and derive the IT infrastructure
  - bottom-up: focus on IT as an enabler: start from IT strategy deriving organisational infrastructure

(Lankhorst et al. 2005, p. 6f)
Challenge: Agility

- Increasingly dynamic environment because of mergers, acquisitions, innovations, new regulations etc.

- To improve chances of survival, enterprises need to be agile

- Agility is the ability to
  - quickly adapt themselves to changes in their environment and
  - seize opportunities as they avail themselves

- Agility has become a business requirement in many lines of business, e.g.
  - car industry (new model within few months instead of 6 years)
  - banking industry (time to market for new product in few weeks instead of 9-12 months)
Agility is not a Reality

In practice, enterprises see themselves hampered in their ability to change in several ways:

- being uninformed about their own products, services, capabilities, internal structures
- traditionally organisations were designed with efficiency and effectiveness in mind rather than agility
- no common understanding and governance of key data resources
- a plethora of legacy applications and infrastructures
- duplicated functionality in terms of people and/or technology
- interwoven and unclear responsibilities
- organisational silos, self-contained business units who operate on their own, with no sharing of data
- silo applications, i.e. self-contained and isolated applications, which only provide functionality to a specific business process
- old generation ERP systems embedded in the organisation's package based silos

Extended Virtual Enterprise

- Agile enterprises co-operate with a large number of suppliers, partners, and sub-contractors, e.g.
  - components are manufactured outside
  - detailed design tasks may be subcontracted
  - after sales service may be provided by third party
  - a close cooperation with partners in a supply network
  - strategic relations with some suppliers

- When considering business processes of an enterprise, the scope must include all value-adding activities – internal and external

(Bernus et al. 2003, p. 10f)
Case: Southwest Airlines

- Read the Southwest Airline case description
- Sit together in groups and discuss the following questions
Enterprise Architecture: Achieving Transparency

- Any organisation benefits from having a clear understanding of its:
  - structure, products, operations, technology etc.
  - the relations tying these together and
  - relations connecting the organisation to its surroundings

- Transparency is a key input for strategic IT control:
  - Clarity on the interdependencies that exist in the landscape
  - A clear statement of progress made toward goals
  - The extent to which planning and business requirements have been enacted

(Lankhorst et al. 2005, p. 6)
(Hanschke 2010, p. 3)
Definitions

- **Architecture** is a fundamental organisation of a system embodied in its components, their relationships to each other, and to the environment, and the principle guiding its design and evolution.

- **Enterprise**: any collection of organisations that has a common set of goals and/or a single bottom line

- **Enterprise Architecture**: a coherent whole of principles, methods, and models that are used in the design and realisation of an enterprise's organisational structure, business processes, information systems, and infrastructure

- **Informations Systems Architecture**: synonym for Enterprise Architecture

(Lankhorst et al. 2005, pp. 2f)
Enterprise Architecture

- An Enterprise Architecture contains all *relevant*
  - Business structures
  - IT structures
  - and their relationships

- Enterprise Architecture gives an overall view on the enterprise
  - merge distributed information from various organisational entities and projects into a whole
  - show the interconnectedness and dependencies between these information

- Show which information systems contribute to which business processes.
Objective of Enterprise Architecture

- Ensuring alignment of business strategy and IT investments
- Describing the interaction between business and information technology
- Making dependencies and implications of changes in business and IT transparent
- Supports communication between different stakeholders by appropriate models
Enterprise Architecture Frameworks

- Zachman Framework
  - Origin and basis for many other approaches

- The Open Group Architecture Framework (TOGAF)
  - TOGAF is based on the „Technical Architecture Framework for Information Management“ (TAFIM) of the US Department of Defense (DoD)

- US Federal Enterprise Architecture Framework (FEAF)
  - Structure for enterprise architectures of US Administrations
  - Supports the development of standardized processes

- Department of Defense Architecture Framework (DoDAF)
  - Used for enterprise architectures in the US military sector
  - Especially suited for large systems with complex integration and communication tasks

- Extended Enterprise Architecture Framework (E2AF)
  - Based on existing frameworks like FEAF and TOGAF

Vgl. (Hanschke 2009)
The Zachman framework is regarded the origin of enterprise architecture frameworks (although originally called "Framework for Information Systems Architecture")

John A. Zachman published the first version in 1987, which he extended in 1992 together with John F. Sowa

The Framework is often referenced as a standard approach for expressing the basic elements of enterprise architecture

The framework is a logical structure for classifying and organising the descriptive representations of an enterprise that are significant to

- the management of the enterprise
- the development of the enterprise's systems  

(Lankhorst et al. 2005, p. 24)
Dimension 1 – Perspectives

Zachman uses the analogy of classical architecture

For the different stakeholders different aspects of a building are relevant - models of the building from different perspectives

**Bubble charts:** conceptual representation delivered by the architect

**Architect's drawing:** transcription of the owner's perceptual requirements – owner's perspective

**Architect's plans:** translation of the owner's requirements into a product – designer's perspective

**Contractor's plans:** phases of operation, architect's plans constrained by nature and technology – builder's perspective

**Shop plans:** parts/sections/components of building details (out-of-context specification) – subcontractor's perspective

**The building:** physical building itself

(Zachman 1987)
Bubble Chart

Architect’s Plan

(Zachman 1987)
Dimension 1: Architectural Representations with analogs in Building and Information Systems

<table>
<thead>
<tr>
<th>Generic</th>
<th>Buildings</th>
<th>Information Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ballpark</td>
<td>Bubble charts</td>
<td>Scope/objectives</td>
</tr>
<tr>
<td>Owner's representation</td>
<td>Architect's drawings</td>
<td>Model of the business (or business description)</td>
</tr>
<tr>
<td>Designer's representation</td>
<td>Architect's plans</td>
<td>Model of the information system (or information system description)</td>
</tr>
<tr>
<td>Builder's representation</td>
<td>Contractor's plans</td>
<td>Technology model (or technology-constrained description)</td>
</tr>
<tr>
<td>Out-of-context representation</td>
<td>Shop plans</td>
<td>Detailed description</td>
</tr>
<tr>
<td>Machine language representation</td>
<td>—</td>
<td>Machine language description (or object code)</td>
</tr>
<tr>
<td>Product</td>
<td>Building</td>
<td>Information system</td>
</tr>
</tbody>
</table>

(Zachman 1987)
### Dimension 1 - Perspectives

<table>
<thead>
<tr>
<th>Role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planner</td>
<td>The content of these cells defines the scope of the enterprise, identifying what should possibly be modeled.</td>
</tr>
<tr>
<td>Owner</td>
<td>These cell models comprise the Business Model - the Owner's expectations from a business perspective for the operating enterprise.</td>
</tr>
<tr>
<td>Designer</td>
<td>These cell models comprise the technology neutral System Model - the Designer's plan for enabling the Business Model.</td>
</tr>
<tr>
<td>Builder</td>
<td>These cell models comprise the Technology Model - the Builder's plan for applying technology to the System Model.</td>
</tr>
<tr>
<td>Sub-contractor</td>
<td>These cells are listings, identifying the actual solutions that have been implemented.</td>
</tr>
<tr>
<td>Functioning Enterprise</td>
<td>The functioning enterprise.</td>
</tr>
</tbody>
</table>

- Each representation is different nature, in content, in semantics from the others – representing different perspectives.
- Representations do not correspond to different levels of details – level of detail is an independent variable, varying within one representation.
Dimension 2: Aspects of an Architecture

- There exist different types of descriptions oriented to different aspects
- Zachman associates each aspect with question word
  
  WHAT  material description
  HOW  functional description
  WHERE  location description
  WHO  organisational description
  WHEN  temporal description
  WHY  motivational description

(Zachman 1987)
Combination of the two ideas

For each different type of description there are different perspectives:

<table>
<thead>
<tr>
<th>Planner</th>
<th>What</th>
<th>How</th>
<th>Where</th>
<th>Who</th>
<th>When</th>
<th>Why</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The content of these cells defines the scope of the enterprise, identifying what should possibly be modeled.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Owner</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>These cell models comprise the Business Model - the Owner's expectations from a business perspective for the operating enterprise.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Designer</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>These cell models comprise the technology neutral System Model - the Designer's plan for enabling the Business Model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Builder</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>These cell models comprise the Technology Model - the Builder's plan for applying technology to the System Model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sub-contractor</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>These cells are listings, identifying the actual solutions that have been implemented.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Functioning Enterprise</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The functioning enterprise.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Zachman Framework – each cell contains models

<table>
<thead>
<tr>
<th>Role</th>
<th>Scope (contextual)</th>
<th>Planner</th>
<th>Enterprise Model (conceptual)</th>
<th>Business Owner</th>
<th>System Model (logical)</th>
<th>Designer</th>
<th>Technology Model (physical)</th>
<th>Implementer</th>
<th>Detailed Representation (out-of-context)</th>
<th>Subcontractor</th>
<th>Functioning System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope (contextual)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Planner</td>
<td>List of things important to the business</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enterprise Model (conceptual)</td>
<td>e.g. Semantic Model</td>
<td>e.g. Business Process Model</td>
<td>e.g. Business Logistics System</td>
<td></td>
<td>e.g. Workflow Model</td>
<td>e.g. Master Schedule</td>
<td>e.g. Business Plan</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Business Owner</td>
<td>e.g. Logical Data Model</td>
<td>e.g. Application Architecture</td>
<td>e.g. Distributed System Architecture</td>
<td>e.g. Human Interface Architecture</td>
<td>e.g. Process Structure</td>
<td>e.g. Business Rule Model</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Model (logical)</td>
<td>e.g. Logical Data Model</td>
<td>e.g. Application Architecture</td>
<td>e.g. Distributed System Architecture</td>
<td>e.g. Human Interface Architecture</td>
<td>e.g. Process Structure</td>
<td>e.g. Business Rule Model</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Designer</td>
<td>e.g. Logical Data Model</td>
<td>e.g. Application Architecture</td>
<td>e.g. Distributed System Architecture</td>
<td>e.g. Human Interface Architecture</td>
<td>e.g. Process Structure</td>
<td>e.g. Business Rule Model</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Technology Model (physical)</td>
<td>e.g. Physical Data Model</td>
<td>e.g. System Design</td>
<td>e.g. Technology Architecture</td>
<td>e.g. Presentation Architecture</td>
<td>e.g. Control Structure</td>
<td>e.g. Rule Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Implementer</td>
<td>e.g. Physical Data Model</td>
<td>e.g. System Design</td>
<td>e.g. Technology Architecture</td>
<td>e.g. Presentation Architecture</td>
<td>e.g. Control Structure</td>
<td>e.g. Rule Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailed Representation</td>
<td>e.g. Data Definition</td>
<td>e.g. Program</td>
<td>e.g. Network Architecture</td>
<td>e.g. Security Architecture</td>
<td>e.g. Timing Definition</td>
<td>e.g. Rule Definition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role (out-of-context)</td>
<td>e.g. Data Definition</td>
<td>e.g. Program</td>
<td>e.g. Network Architecture</td>
<td>e.g. Security Architecture</td>
<td>e.g. Timing Definition</td>
<td>e.g. Rule Definition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subcontractor</td>
<td>e.g. Data Definition</td>
<td>e.g. Program</td>
<td>e.g. Network Architecture</td>
<td>e.g. Security Architecture</td>
<td>e.g. Timing Definition</td>
<td>e.g. Rule Definition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Functioning System</td>
<td>e.g. Data Definition</td>
<td>e.g. Program</td>
<td>e.g. Network Architecture</td>
<td>e.g. Security Architecture</td>
<td>e.g. Timing Definition</td>
<td>e.g. Rule Definition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Case: Southwest Airlines (revisited)

- Could an Enterprise Architecture have helped to avoid the problems before the Southwest Transformation?
- How could Southwest exploit Enterprise Architecture?
Alternative Frameworks

- Die distinction between perspectives and aspects can be found in various other frameworks, e.g.
  - ARIS – Architecture of integrated Information Systems
  - BPMS - Business Process Management Systems
  - Best Practice Enterprise Architecture
  - PlugIT Modelling Framework
  - MDA - Model-driven Architecture
  - TOGAF - The Open Group Architecture Framework

- They vary in the number and concrete definition of perspectives and aspects
ARIS –
Architecture for integrated Information Systems

Views correspond to aspects
Levels correspond to perspectives
Aspects und Perspectives fo the BPMS Paradigm (Business Process Management Systems)

Aspects:

- Products
- Business Processes
- Organisational Structure
- Information Technology

Perspectives:

- Strategic Decision Process
- Re-Enginnering Process
- Resource Allocation Process
- Workflow Process
- Performance Evaluation Process

http://www.boc-eu.com
Best Practice Enterprise Architecture
OMG’s Model-Driven Architecture MDA

MDA comprises three levels of abstraction with mappings between them

CIM Computation-Independent Model
♦ modelling the requirements for the system describing the situation in which the system will be used
♦ hiding much or all information about the use of IT systems

PIM Platform-Independent Model
♦ describing operations of the system while hiding details for a particular platform
♦ describing those parts of the system specification that do not change from one platform to another

PSM Platform-Specific Model
♦ Combines specifications of PIM with details about a particular type of platform

The levels correspond to perspectives
OMG's Model-Driven Architecture

- MDA is provided by Object Management Group OMG
- Aims to provide an open, vendor-neutral approach to interoperability
- Builds upon OMG's modelling standards
  - Unified Modelling Language UML
  - Meta Object Facility MOF
- MDA wants to raise the level of abstraction at which software solutions are specified
  - generate code from models and views
  - Example: specify software in UML instead of programming it in Java
- Recently, OMG has extended the focus of MDA to cover business aspects of a company, e.g.
  - Business process modelling notation BPMN
  - Business motivation model BMM
  - Semantics for Business Vocabulary and Rules SBVR

(Lankhorst et al. 2005, p. 25f)
Partial Architectures of the Best Practice Architecture

- **Business Architecture**
  - Describing main entities that determine the business: business processes, functions, products, business units and business objects.

- **Application Architecture**
  - Documentation of the information systems landscape, i.e. information systems, their data and interfaces and the information flow
  - Bridge between business architecture and the architectures of technology and infrastructure

- **Technology Architecture**
  - Determination of enterprise-specific technical standards for information systems, interfaces and infrastructure

- **Infrastructure Architecture**
  - Entities of the infrastructure, on which the information systems are running
TOGAF - The Open Group Architecture Framework

- Building Blocks and method to develop enterprise architectures
- Architecture Development Method (ADM)
  - generic method for the development of an enterprise architecture
  - Objectives, approaches, needed input, activities and results for each phase of the life cycle
- Enterprise Continuum
  - reference description in form of graphical models and text documents
  - Foundation Architecture
    - Building Blocks: Services and functions an architecture has to support:
    - Technical Reference Modell (TRM): defines the regulation framework
    - Standard Information Base (SIB): technical building blocks standardized for the enterprise
  - Integrated Information Infrastructure Reference Model
    - Description of reference architecture for the integration of Information systems
- Resource Base
  - templates, case studies etc
TOGAF Technical Reference Model
The model of an enterprise architecture described in TOGAF distinguishes four partial architectures:

- **Business Architecture**
  - Strategies, governance, organisation and business processes of the enterprise

- **Data Architecture**
  - Data and their relations as well as principles for the organisation and the management of resources

- **Application Architecture**
  - Information systems and their relations to business processes

- **Technology Architecture**
  - Current technical realisation and future enterprise-specific standards like operating system, middleware and infrastructure

Data Architecture and Application Architecture together are the **Information System Architecture**
TOGAF Architecture Development Cycle

(based on The Open Group 2002)
Perpectives and Aspects in the Project plugIT

plugIT is a project funded by the EU http://plug-it-project.eu/
Relations between Models and Model Elements
Relations between Models and Model Elements

- The models of the Framework are not isolated
- There are relations between (elements of) the models
  - **Horizontal Relations**: In same perspective, e.g.
    - Data used in a process
    - Application implementing a process activity
  - **Vertical relations**: Between different perspectives
    - Implementation of an application
    - Database model for an entity relationship model
The unambiguous specification and description of components and especially their relationships in an architecture requires a coherent architecture modelling language – or modelling languages.

Requirements for modelling languages:
- enable integrated modelling of architectural domains
- should be understandable by both people from IT and people with a business background
- allow transition from "as is" to "to be": provide analysis methods for quantitative and qualitative impact of changes

There are no languages specifically designed for describing enterprise architectures. However, there are languages for subdomains:
- Business Process Modelling
- Software Modelling
Architecture Languages

- Software Modelling
  - For software modelling, UML is the dominant language

- Business Modelling
  - For business process modelling there are a multitude of languages, e.g.
    - Business Process Management Notation BPMN
    - Event-driven Process Chains EPC
    - Flow Diagrams
    - Petri Nets
    - IDEF
    - and a lot of vendor-specific variants
  - For other aspects there are emerging languages and standards, e.g. Business rules, Business motivation,
*Perspectives, Aspects, and Frameworks*

Perspectives

Data/Process/People/Applications/Products/Motivation

Aspects

Language families

BPMS/OMG...
Summary: Enterprise Architecture, Alignment and Agility

■ Use of the EA models
  ♦ Designing a new business/company (analogy: building a new house)
  ♦ Reorganisation of the enterprise
    ● Business Process Re-Engineering
    ● migration of an IT infrastructure
    ● exchanging/upgrading an information system
      (analogy: reconstructing a building)

■ Any re-organisation must ensure alignment of Business and IT

■ Enterprise Architecture support agility by
  ♦ providing transparency of context in case of business IT alignment
    ● requirements of business for IT
    ● influences of IT changes on business

■ On the other hand, any re-organisation project leads to changes of the Enterprise Architecture
Enterprise Architecture vs. Business Process Management

- From the *business process perspective*, enterprise architecture achieves enterprise integration through
  - capturing and describing processes, strategies, organisation structures, information and material flow, resources etc.
  - concentration on how to perform core business processes in an organisation
  - considering the information and material flow in the entire process

- In this sense, business process management (BPM) and business process re-engineering (BPR) rely on enterprise architecture

- *Tools for BPM and BPR are part of the toolset of enterprise architecture*
Communicating about Architecture

- Different types of stakeholders have their own viewpoints on the architecture
- Architectures are subject to change; methods to analyse the effects of changes are necessary
- An integrated set of methods for specification, analysis and communication of architectures is needed that fulfils the needs of different types of stakeholders

(Lankhorst et al. 2005, p. 4)