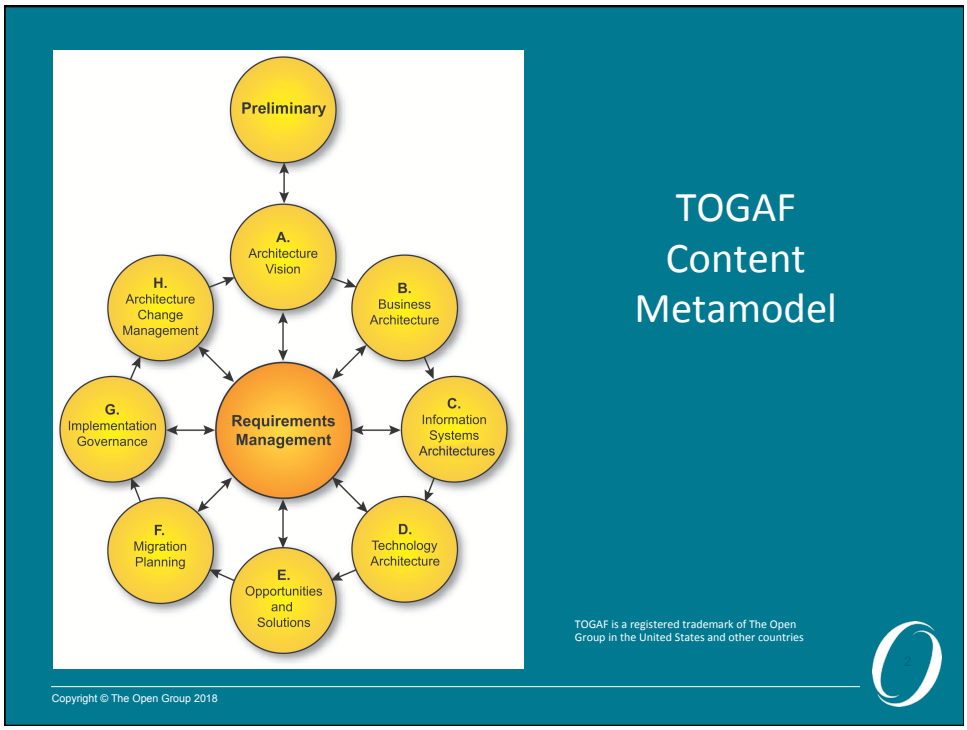


Roadmap

Part I - Introduction
Preface, Executive Overview, Core Concepts, Definitions
Part II – Architecture Development Method
Introduction to ADM
ADM Phase Narratives
Part III – ADM Guidelines and Techniques
Guidelines for Adapting the ADM Process
Techniques for Architecture Development
Part IV – Architecture Content Framework
Content Metamodel
Architectural Artifacts
Architecture Deliverables
Building Blocks
Part V – Enterprise Continuum and Tools
Enterprise Continuum
Architecture Partitioning
Architecture Repository
Tools for Architecture Development
Part VI – Architecture Capability Framework
Architecture Board
Architecture Compliance
Architecture Contracts
Architecture Governance
Architecture Maturity Models
Architecture Skills Framework

- Part IV, Architecture Content Framework, Chapter 30

3

Module Objectives

The objectives of this module are to describe:

- What a *metamodel* is and why it is needed
- Key concepts of the Core Metamodel
- The division of the metamodel into Core and Extensions
- Key concepts of the Core Metamodel Entities
- The components of the TOGAF Content Metamodel

4

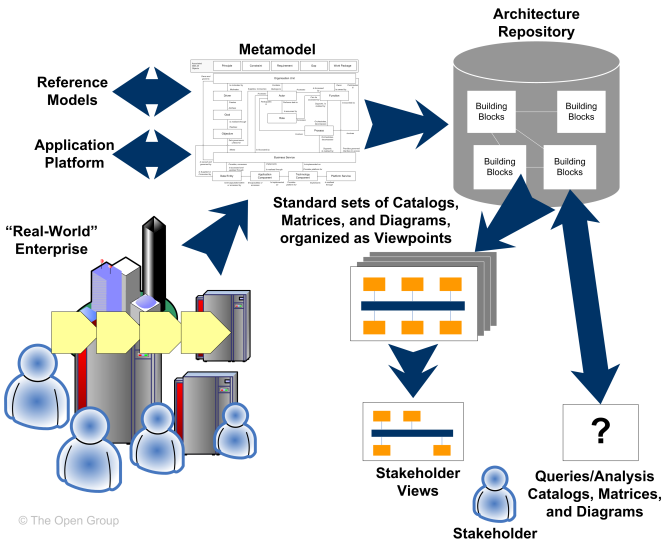
What is a metamodel?

» A metamodel is a precise definition of the constructs and rules needed for creating models
  – Source www.metamodel.com

» A model that describes how and with what the architecture will be described in a structured way.
  – TOGAF Standard, Version 9.2, Chapter 3 *Definitions*

5

Why a metamodel?



6

Benefits of the Metamodel

The content metamodel provides a number of benefits:

- It formalizes the definition of an Enterprise Architecture
- It formalizes the relationship between objects
- It enables an EA tool mapping

7

Formal and Informal Modeling

- When defining architecture content there are choices to be made on the level of structure and formality
- In some cases very formal specific language is needed in order to articulate and govern in a precise or detailed way
- In other cases the use of formal engineering discipline will result in architecture content that is:
  - inappropriate for the audience
  - difficult to communicate

8

Core Content Metamodel Concepts

- A TOGAF architecture is based on
  - Defining architectural building blocks within architecture **catalogs**
  - Specifying the relationships between those building blocks in architecture **matrices**
  - And presenting communication **diagrams** that show in a precise way what the architecture is
- The metamodel is structured into **Core** and **Extension** content
  - Core content is designed not to be altered

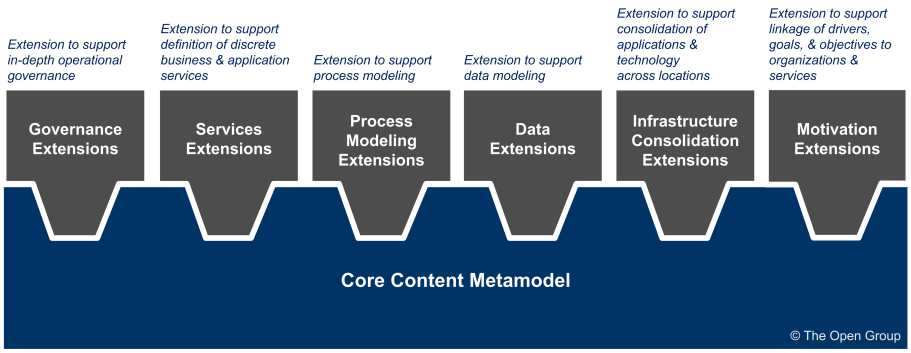
9

Core and Extension Content

- In order to support many scenarios the metamodel has been partitioned into **core** and **extension** content
- The **core** provides a minimum set of architectural content to support traceability across artifacts
- The **extension** content allows for more specific or more in-depth modeling

10

TOGAF Content Metamodel and its Extensions



11

Core Metamodel Entities

- **Actor**: A person, organization, or system that is outside the consideration of the architecture model, but interacts with it.
- **Application Component**: An encapsulation of application functionality that is aligned to implementation structuring.
- **Business Capability:** A particular ability that a business may possess or exchange to achieve a specific purpose
- **Business Service**: Supports business capabilities through an explicitly defined interface and is explicitly governed by an organization.
- **Course of Action**: Direction and focus provided by strategic goals and objectives, often to deliver the value proposition characterized in the business model
- **Data Entity**: An encapsulation of data that is recognized by a business domain expert as a discrete concept. Data entities can be tied to applications, repositories, and services and may be structured according to implementation considerations.
- **Function**: Delivers business capabilities closely aligned to an organization, but not explicitly governed by the organization.

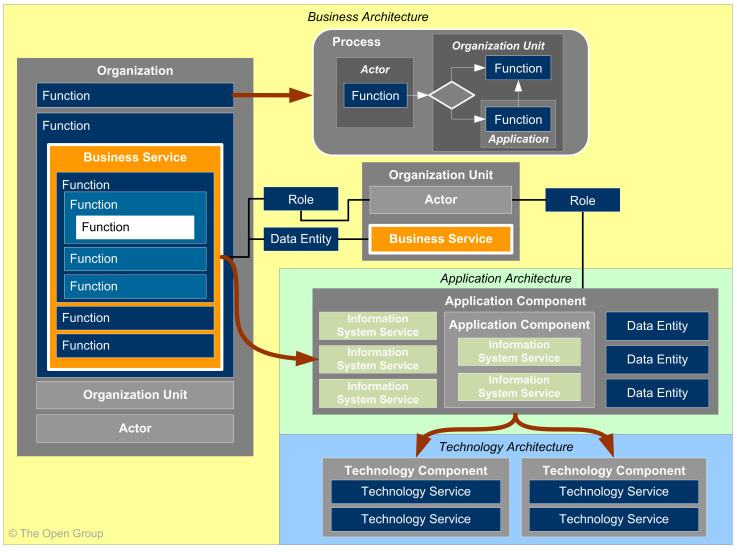
12

Core Metamodel Entities (Cont' d)

- **Information System Service:** The automated elements of a business service. An information system service may deliver or support all of one or more business services.
- **Organization Unit**: A self-contained unit of resources with line management responsibility, goals, objectives, and measures. Organization units may include external parties and business partner organizations.
- **Role**: An actor assumes a role to perform a task.
- **Technology Component**: An encapsulation of technology infrastructure that represents a class of technology product or specific technology product.
- **Technology Service**: A technical capability required to provide enabling infrastructure that supports the delivery of applications.
- **Value Stream**: a representation of an end-to-end collection of value-adding activities that create an overall result for a customer, stakeholder, or end-user

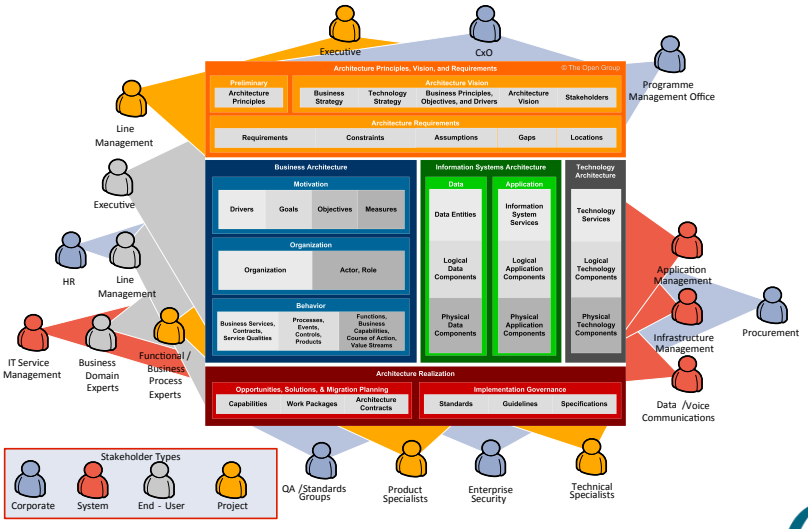
13

Core Entities and their Relationships



14

Stakeholder Needs



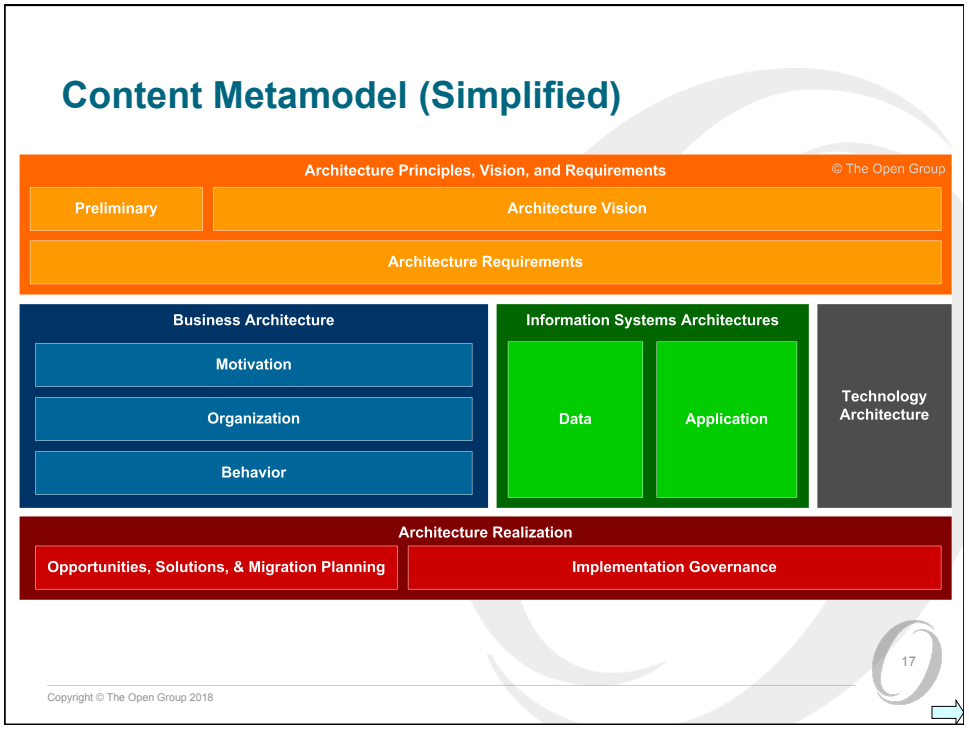
15

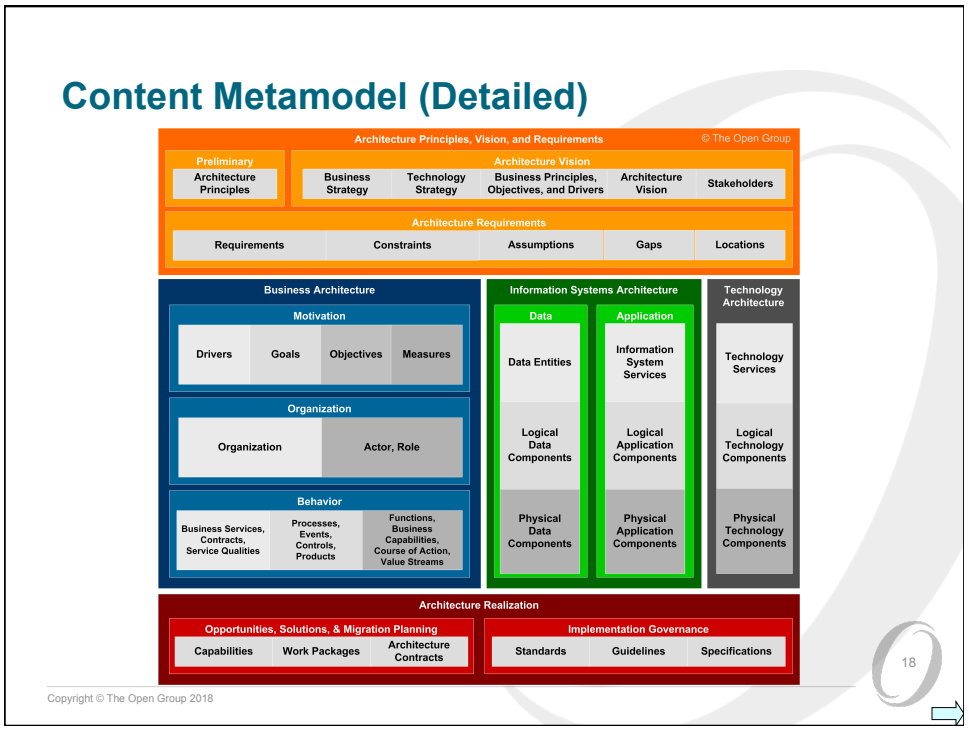
The Content Metamodel

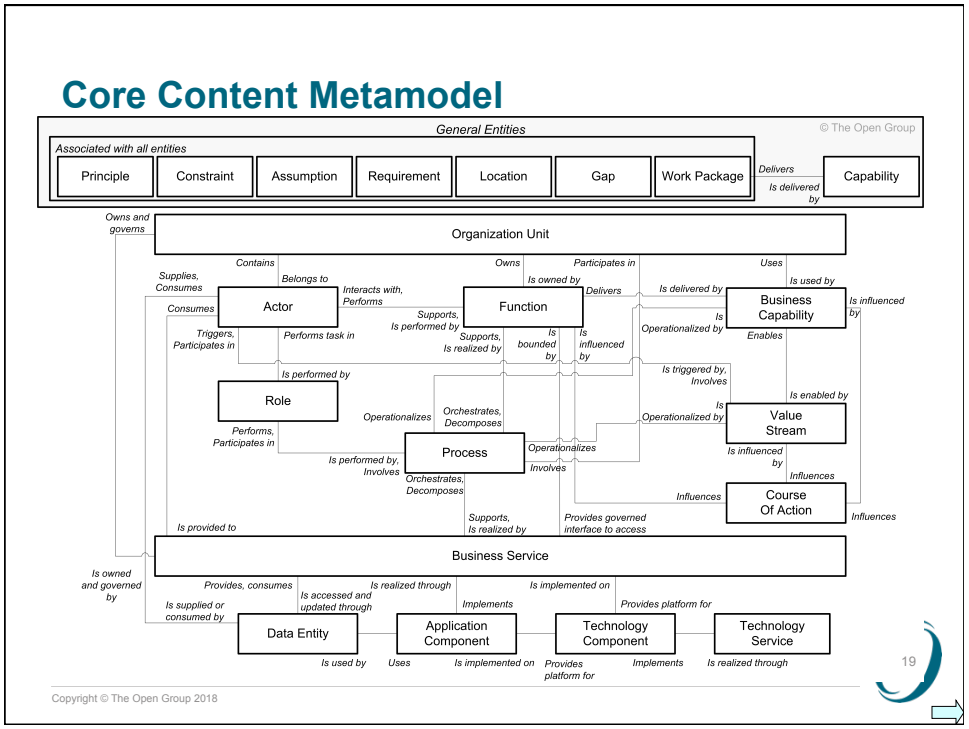
The content metamodel provides definitions of all the types of building blocks that may exist, showing how they can be described and related to one another.

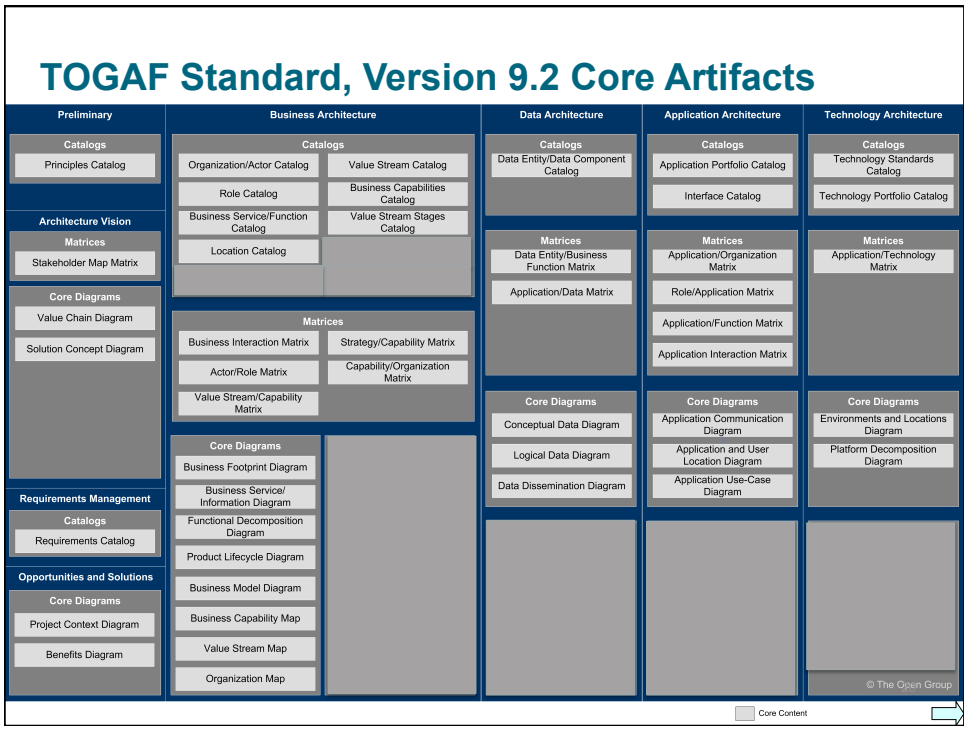
- When creating and managing architectures, it is necessary to consider concerns such as business services, actors, applications, data entities, and technology.
- The metamodel highlights these concerns, shows their relationships and identifies artifacts that can be used to represent them in a consistent way.
- The metamodel can also be used to provide guidance to organizations that wish to implement their architecture using an architecture tool.

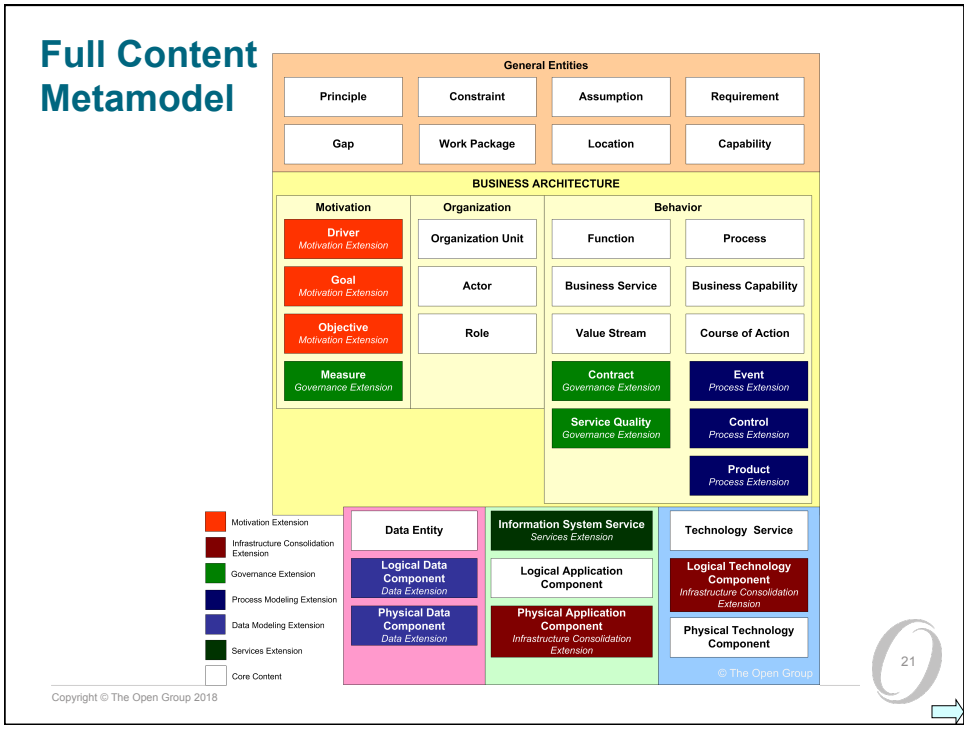
16

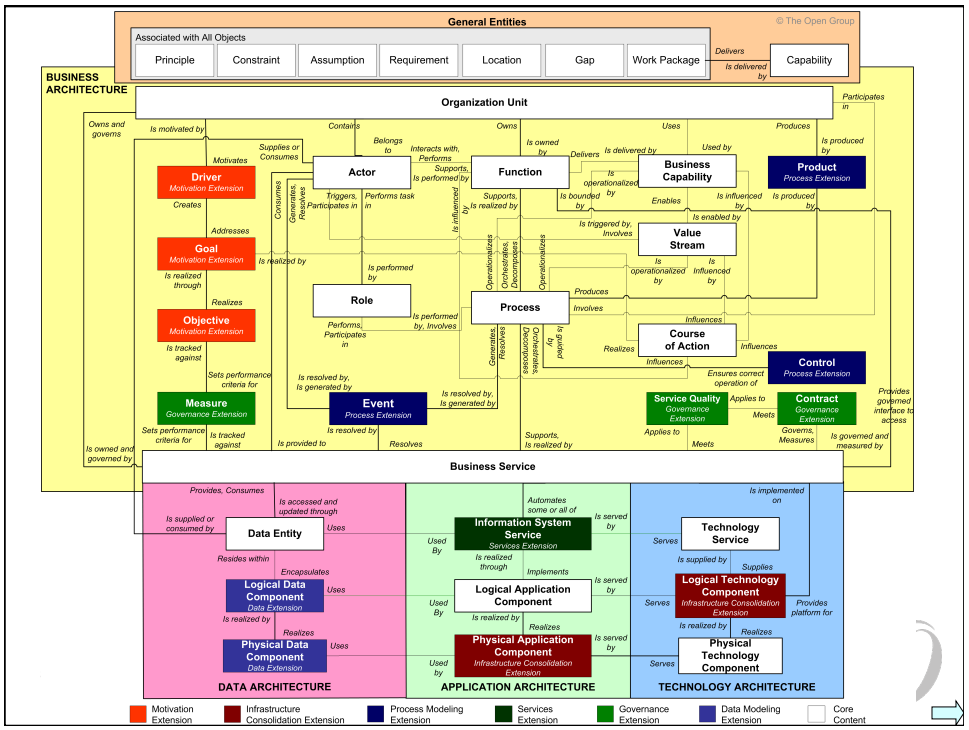


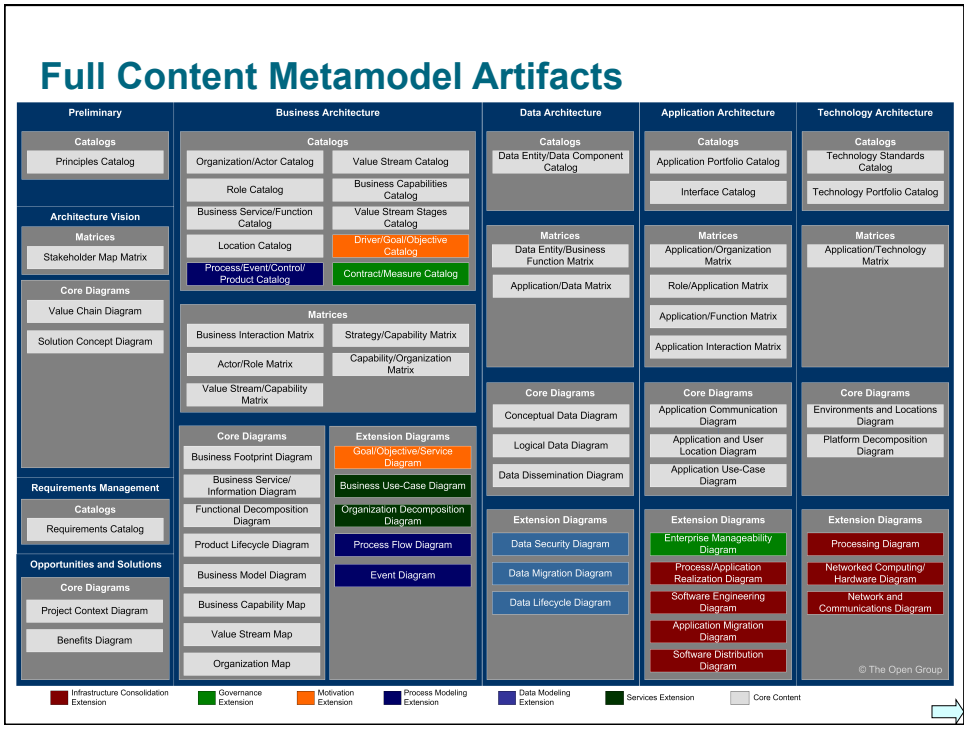


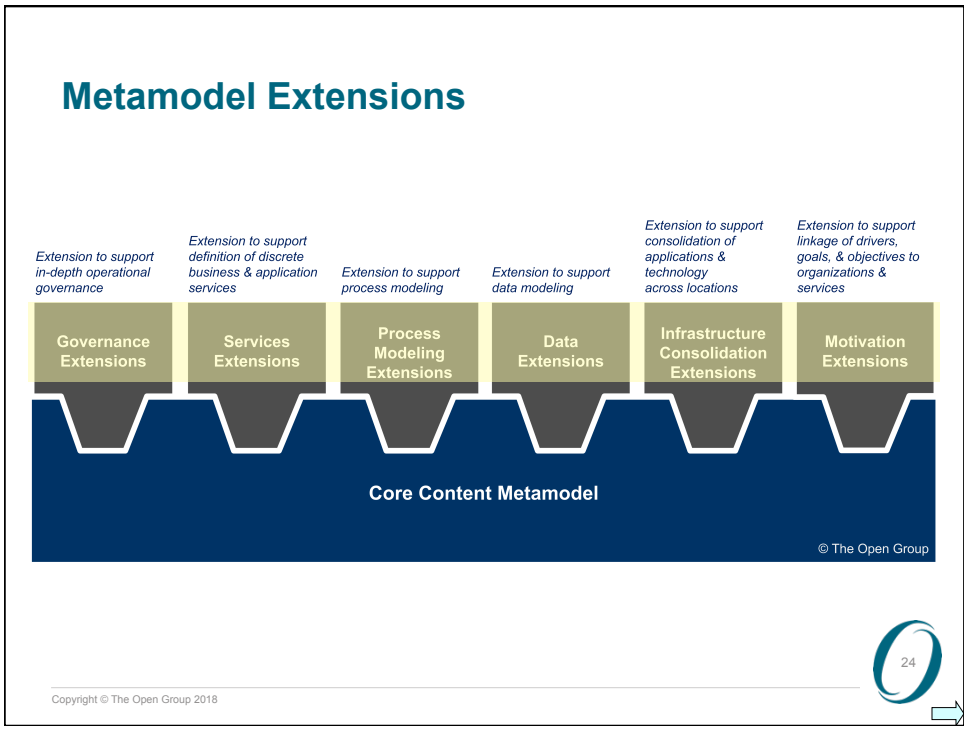


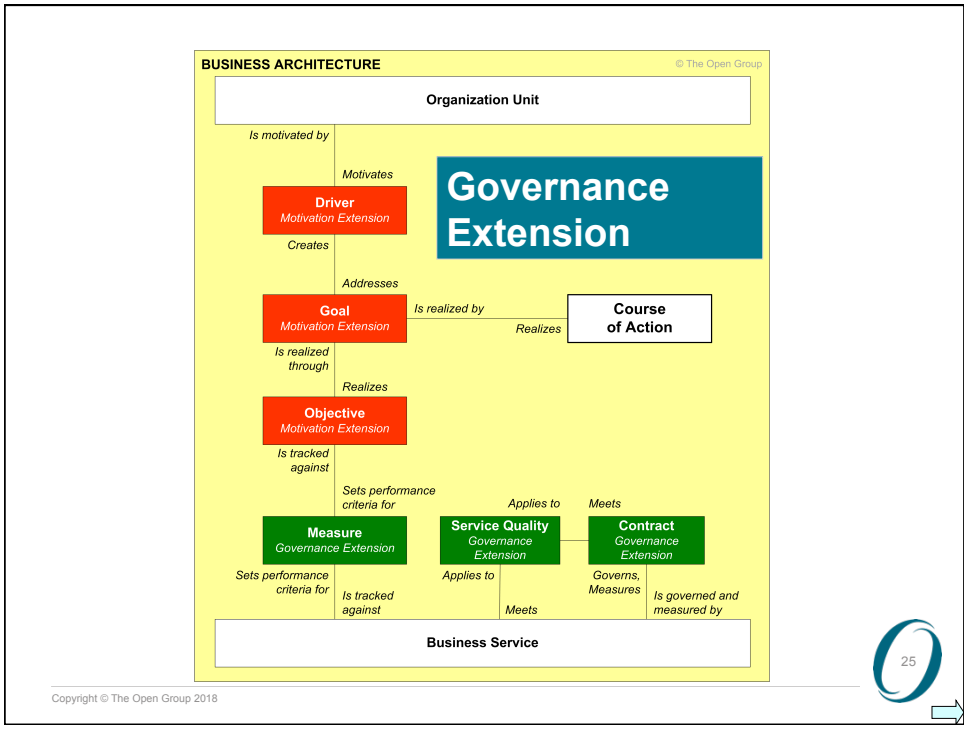




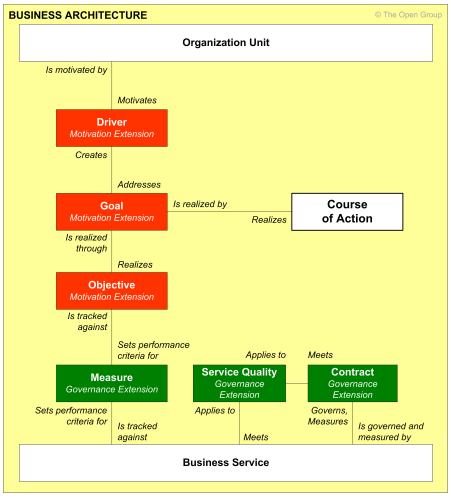






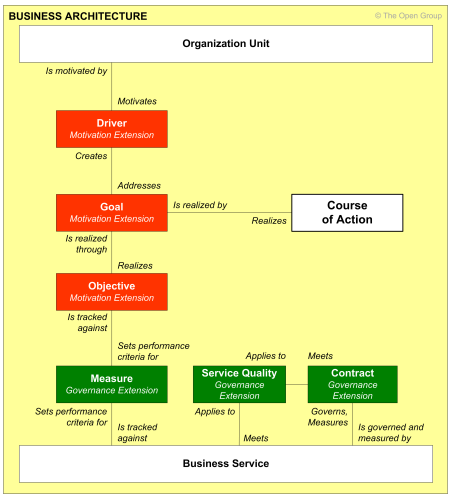


Governance Extension



- Scope:
  - The ability to apply measures to objectives and then link those measures to services
  - The ability to apply contracts to service communication or service interactions with external users and systems
  - The ability to define re-usable service qualities defining a service-level profile that can be used in contracts
  - Creation of additional diagrams to show ownership and management of systems
- Additional diagrams to be created:
  - Enterprise Manageability diagram

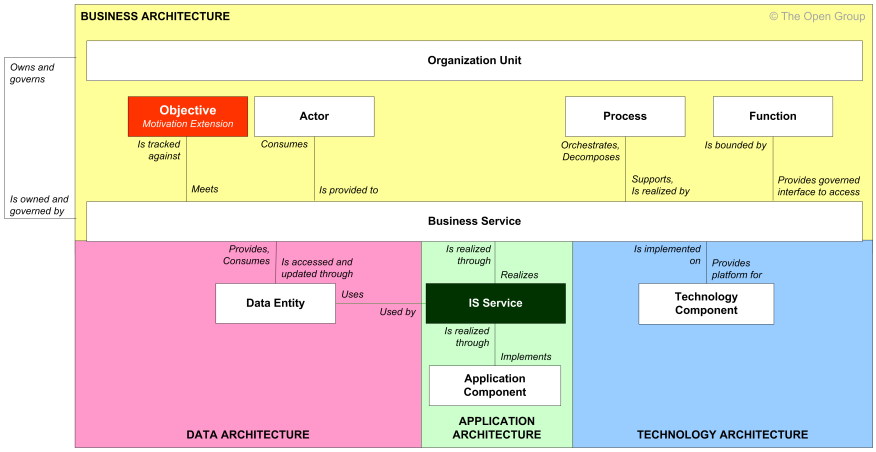
Governance Extension



- This extension should be used in the following situations:
  - When an organization is considering IT change that will result in a significant impact to existing operational governance models
  - When an organization has granular requirements for service levels that differ from service to service
  - When an organization is looking to transform its operational governance practice

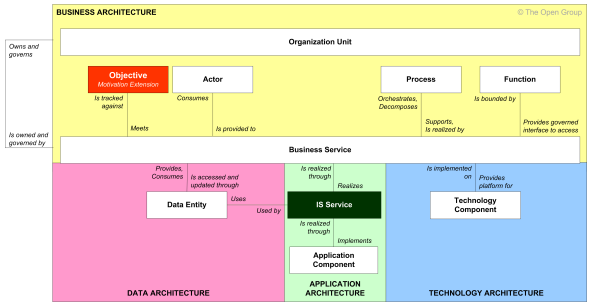
27

Services Extension



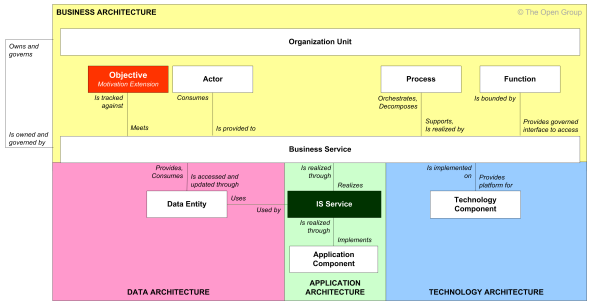
28

Services Extension



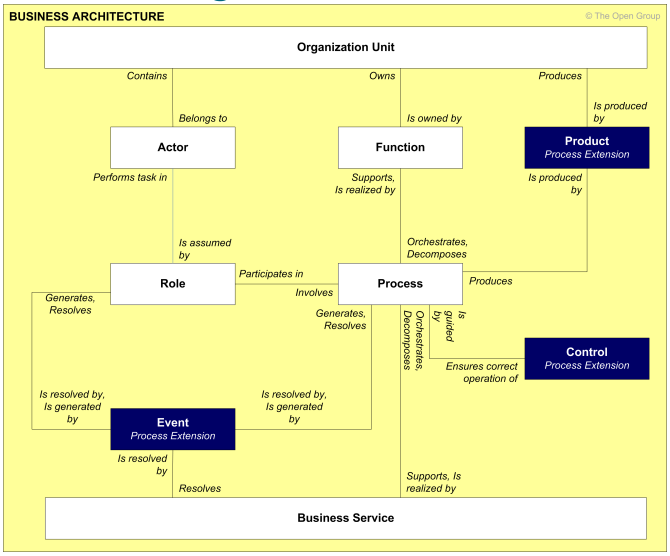
- Scope:
  - Creation of IS services as an extension of business service
- Additional diagrams to be created:
  - Business Use-Case Diagram
  - Organization Decomposition Diagram

Services Extension

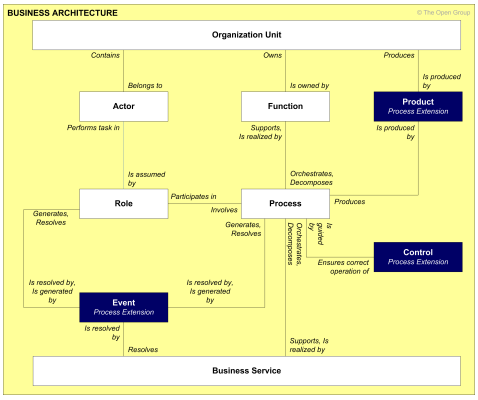


- This extension should be used in the following situations:
  - When the business has a preset definition of its services that does not align well to technical and architectural needs
  - When business and IT use different language to describe similar capabilities
  - Where IT service is misaligned with business need, particularly around the areas of quality of service, visibility of performance, and management granularity
  - Where IT is taking initial steps to engage business in discussions about IT architecture

Process Modeling Extension

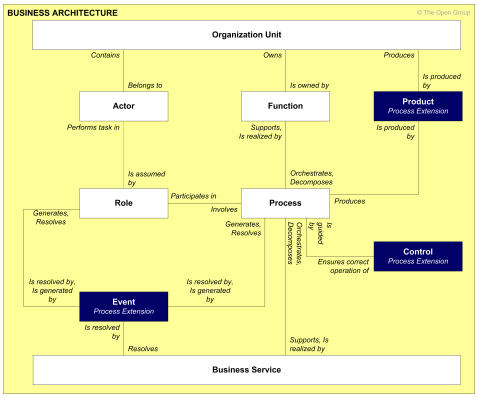


Process Modeling Extension



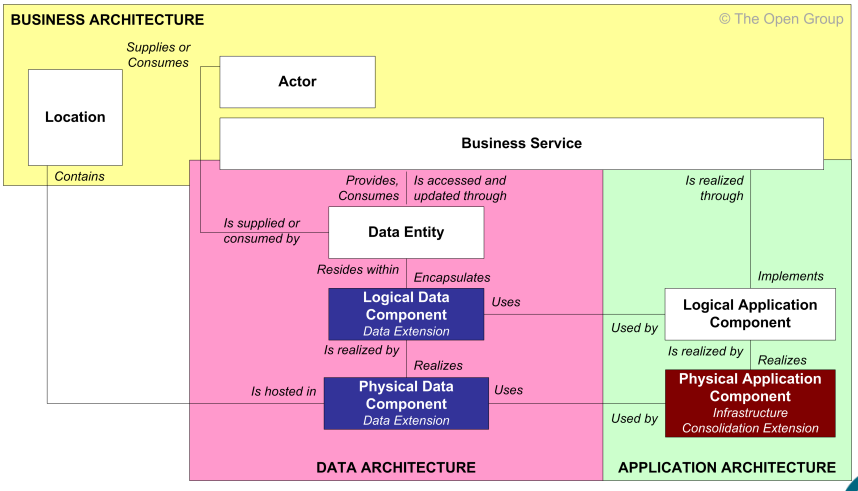
- Scope:
  - Creation of events as triggers for processes
  - Creation of controls that business logic and governance gates for process execution
  - Creation of products to represent the output of a process
  - Creation of event diagrams to track triggers and state changes across the organization
- Additional diagrams to be created:
  - Process Flow diagrams
  - Event diagrams

Process Modeling Extension

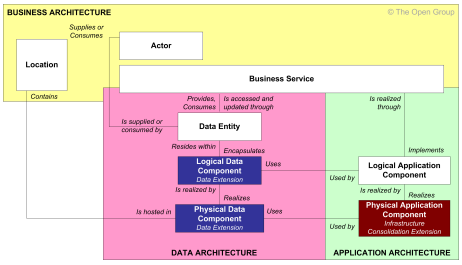


- This extension should be used in the following situations:
  - Where the architecture must pay specific attention to state and events
  - Where the architecture is required to explicitly identify and store process control steps; for example, to support regulatory compliance
  - Where the architecture features critical or elaborate process flows

Data Extension

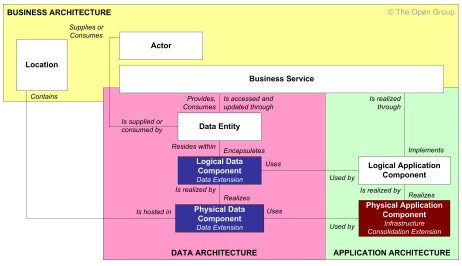


Data Extension



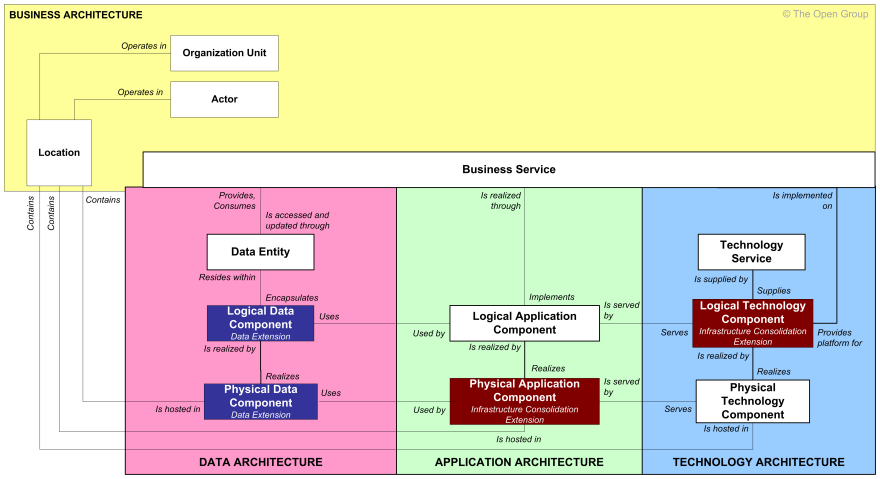
- Scope:
  - Creation of logical data components that group data entities into encapsulated modules for governance, security, and deployment purposes
  - Creation of physical data components that implement logical data components; analogous to databases, registries, repositories, schemas, and other techniques of segmenting data
  - Creation of data lifecycle, data security, and data migration diagrams to show data concerns in more detail
- Additional diagrams to be created: :
  - Data Security diagram
  - Data Migration diagram
  - Data Lifecycle diagram

Data Extension

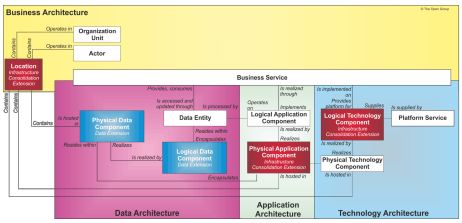


- This extension should be used in the following situations:
  - Where the architecture features significant complexity and risk around the location, encapsulation, and management of or access to data

Infrastructure Consolidation Extension



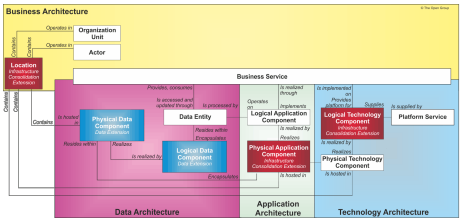
Infrastructure Consolidation Extension



- Additional diagrams to be created:
  - Process/System Realization diagram
  - Software Engineering diagram
  - Application Migration diagram
  - Software Distribution diagram
  - Processing diagram
  - Networked Computing/Hardware diagram
  - Network and Communications diagram

» Scope:
- Creation of logical and physical application components to abstract the capability of an application away from the actual applications in existence
- Creation of logical and physical technology components to abstract product type from the actual technology products in existence
- Creation of additional diagrams focusing on the location of assets, compliance with standards, structure of applications, application migration, and infrastructure configuration

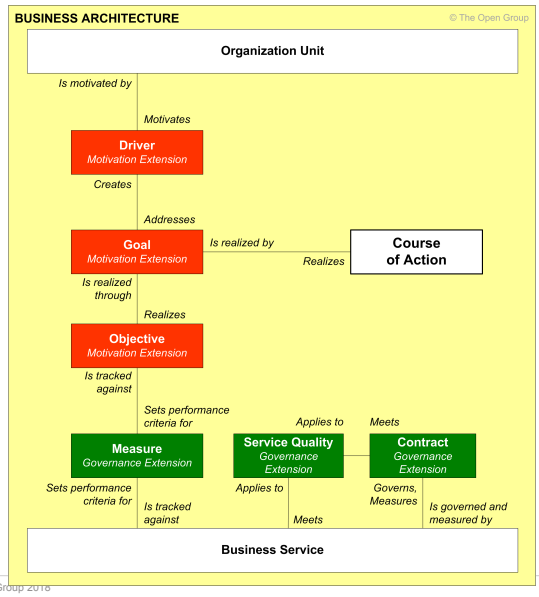
Infrastructure Consolidation Extension



- This extension should be used in the following situations:
  - Where many technology products are in place with duplicate or overlapping capability
  - Where many applications are in place with duplicate or overlapping functionality
  - Where applications are geographically dispersed and the decision logic for determining the location of an application is not well understood
  - When applications are going to be migrated into a consolidated platform
  - When application features are going to be migrated into a consolidated application

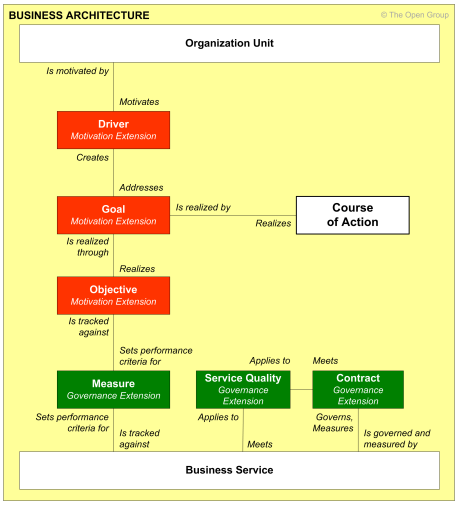
39

Motivation Extension



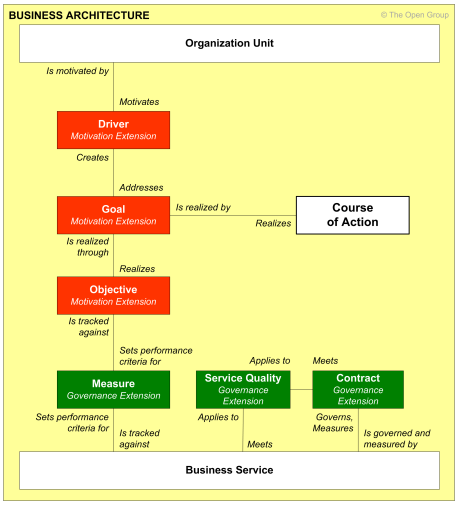
40

Motivation Extension



- The scope of this extension is as follows:
  - Creation of a new metamodel entity for Driver that shows factors generally motivating or constraining an organization
  - Creation of a new metamodel entity for Goal that shows the strategic purpose and mission of an organization
  - Creation of a new metamodel entity for Objective that shows near to mid-term achievements that an organization would like to attain
  - Creation of a Goal/Objective/Service diagram showing the traceability from drivers, goals, and objectives through to services
- Additional diagrams to be created:
  - Goal/Objective/Service diagram

Motivation Extension



- This extension should be used in the following situations:
  - When the architecture needs to understand the motivation of organizations in more detail than the standard business or engagement principles and objectives that are informally modeled within the core content metamodel
  - When organizations have conflicting drivers and objectives and that conflict needs to be understood and addressed in a structured form
  - When service levels are unknown or unclear

Summary

The TOGAF standard provides a rich metamodel
This provides a number of benefits:

- It supports both formal and informal modeling
- It formalizes the definition of an Enterprise Architecture
- It formalizes the relationship between objects
- It enables an EA tool mapping

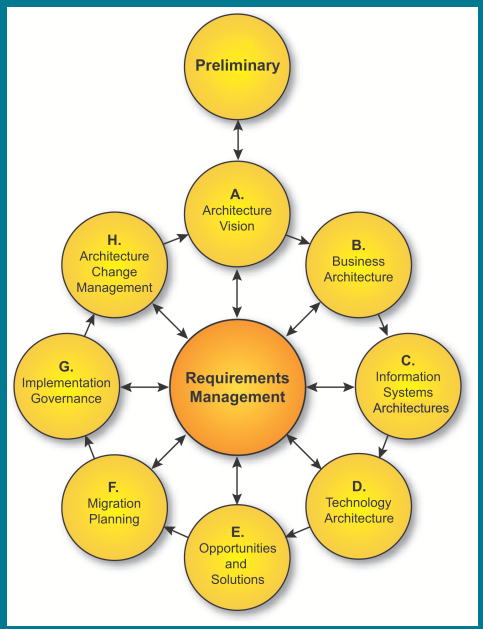
Exercise

Exercise

» Determine which of the Metamodel extensions is most appropriate for the following situations:

1. Where organizations have conflicting objectives
2. Where service levels are unknown
3. Where many applications are in use with overlapping functionality
4. Where management of information is complex
5. Where business process has to support regulatory compliance

45



TOGAF Content Metamodel

46