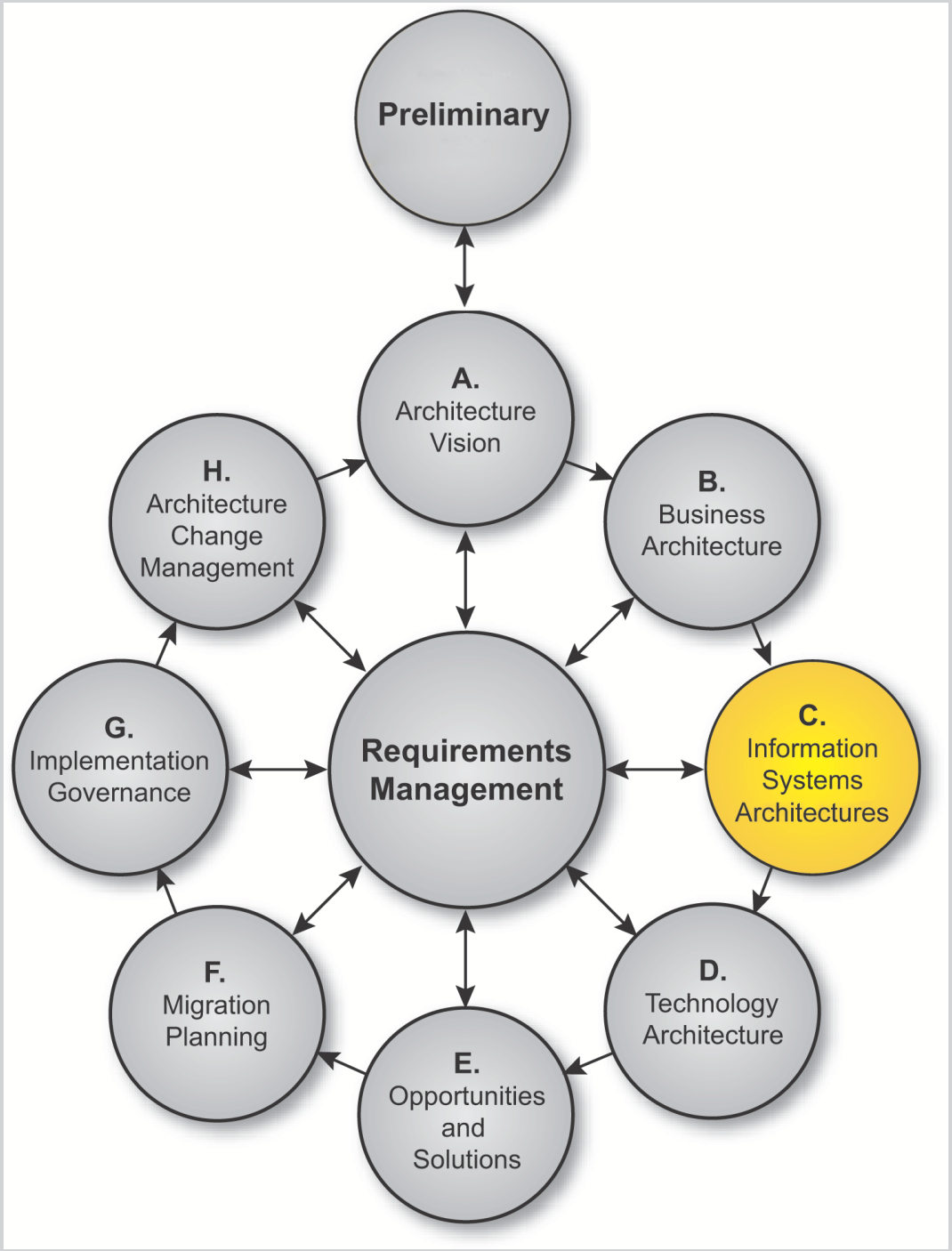
TOGAF

Version 9 Enterprise Edition

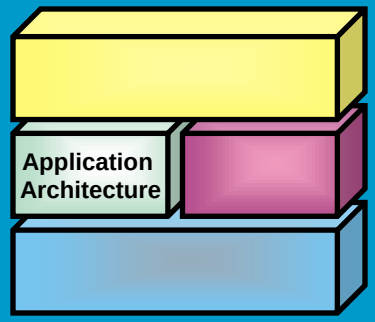
# Module 20A Phase C Application Architecture – Catalogs, Matrices and Diagrams



Published by The Open Group, January 2009




# Phase C: Application Architecture – Catalogs, Matrices and Diagrams






TOGAF™

# Module Objectives

The objectives of this module are to understand:

- The Catalogs, Matrices and Diagrams of Phase C, Application Architecture
- What they consist of
- How they are used

# TOGAF 9 Viewpoints

<table>
<tr>
<td>Preliminary Phase<br>• Principles catalog<br><br>Phase A, Architecture Vision<br>• Stakeholder Map matrix<br>• Value Chain diagram<br>• Solution Concept diagram</td>
<td>Phase B, Business Architecture<br>• Organization/Actor catalog<br>• Driver/Goal/Objective catalog<br>• Role catalog<br>• Business Service/Function catalog<br>• Location catalog<br>• Process/Event/Control/Product catalog<br>• Contract/Measure catalog<br>• Business Interaction matrix<br>• Actor/Role matrix<br>• Business Footprint diagram<br>• Business Service/Information diagram<br>• Functional Decomposition diagram<br>• Product Lifecycle diagram<br>• Goal/Objective/Service diagram<br>• Use-Case diagram<br>• Organization Decomposition diagram<br>• Process Flow diagram<br>• Event diagram</td>
<td>Phase C, Data Architecture<br>• Data Entity/Data Component catalog<br>• Data Entity/Business Function matrix<br>• System/Data matrix<br>• Class diagram<br>• Data Dissemination diagram<br>• Data Security diagram<br>• Class Hierarchy diagram<br>• Data Migration diagram<br>• Data Lifecycle diagram</td>
<td>Phase C, Application Architecture<br>• Application Portfolio catalog<br>• Interface catalog<br>• System/Organization matrix<br>• Role/System matrix<br>• System/Function matrix<br>• Application Interaction matrix<br>• Application Communication diagram<br>• Application and User Location diagram<br>• System Use-Case diagram<br>• Enterprise Manageability diagram<br>• Process/System Realization diagram<br>• Software Engineering diagram<br>• Application Migration diagram<br>• Software Distribution diagram</td>
</tr>
<tr>
<td colspan="2">Phase D, Technology Architecture<br>• Technology Standards catalog<br>• Technology Portfolio catalog<br>• System/Technology matrix<br>• Environments and Locations diagram<br>• Platform Decomposition diagram<br>• Processing diagram<br>• Networked Computing/Hardware diagram<br>• Communications Engineering diagram</td>
<td>Phase E. Opportunities & Solutions<br>• Project Context diagram<br>• Benefits diagram</td>
<td>Requirements Management<br>• Requirements catalog</td>
</tr>
</table>

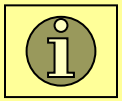


TOGAF™

# Catalogs, Matrices and Diagrams

**Catalogs**
- Application Portfolio catalog
- Interface catalog

**Matrices**
- System/Organization matrix
- Role/System matrix
- System/Function matrix
- Application Interaction matrix



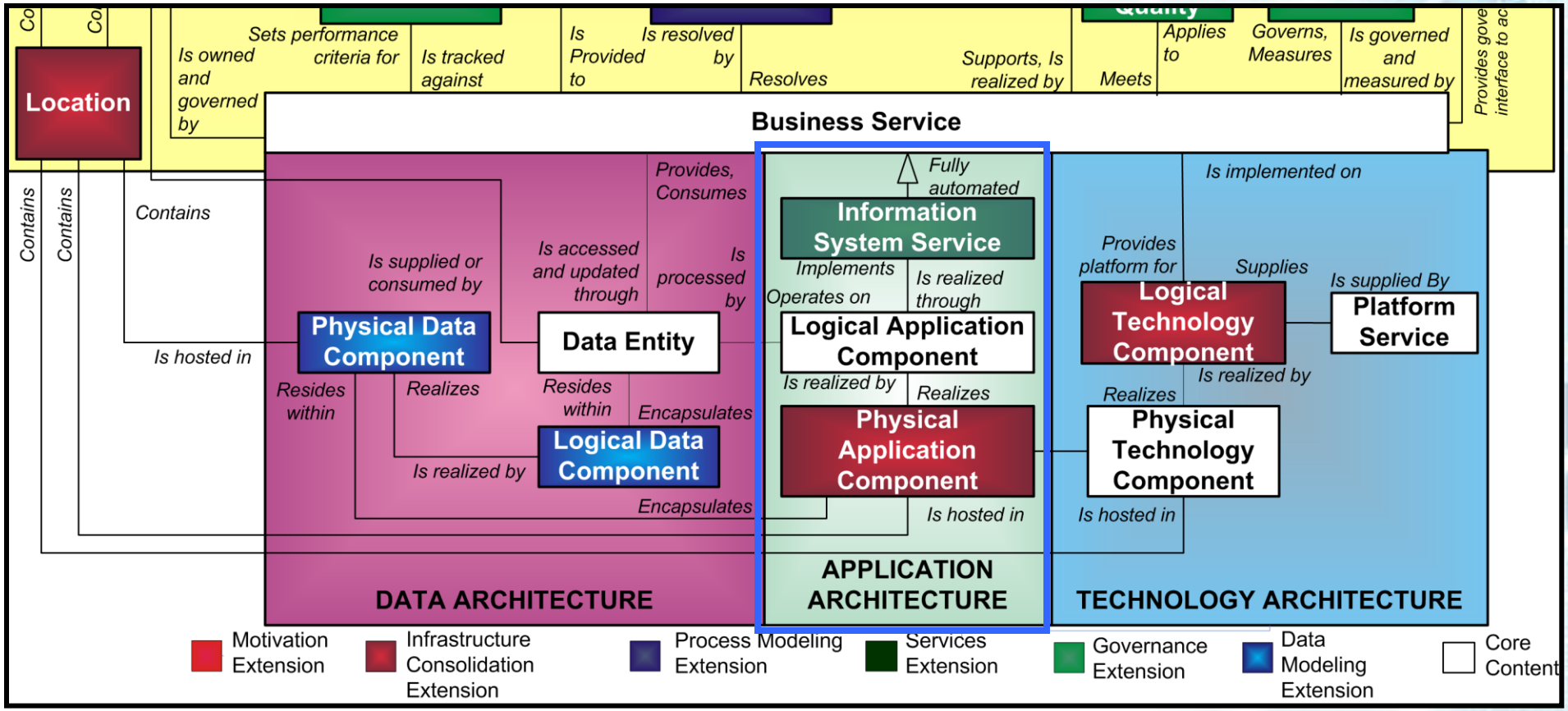
**The exact format of the catalogs, matrices and diagrams will depend on the tools used**

**Diagrams**
- Application Communication diagram
- Application and User Location diagram
- System Use-Case diagram
- Enterprise Manageability diagram
- Process/System Realization diagram
- Software Engineering diagram
- Application Migration diagram
- Software Distribution diagram

# Catalogs

| Catalog | Purpose |
|---|---|
| **Application Portfolio Catalog** | To identify and maintain a list of all the applications in the enterprise. This list helps to define the horizontal scope of change initiatives that may impact particular kinds of applications. An agreed Application Portfolio allows a standard set of applications to be defined and governed.<br><br>It contains the following metamodel entities:<br>•Information System Service<br>•Logical Application Component<br>•Physical Application Component |
| **Interface Catalog** | The purpose of the Interface catalog is to scope and document the interfaces between applications to enable the overall dependencies between applications to be scoped as early as possible.<br><br>It contains the following metamodel entities:<br>•Logical Application Component<br>•Physical Application Component<br>•Application *communicates with* application relationship |

# Exercise






TOGAF™

# Matrices

- System/Organization matrix
- Role/System matrix
- System/Function matrix
- Application Interaction matrix

# System/Organization Matrix

- The purpose of this matrix is to depict the relationship between systems (i.e., application components) and organizational units within the enterprise.
- The mapping of the Application Component-Organization Unit relationship is an important step as it enables the following to take place:
  - Assign usage of applications to the organization units that perform business functions
  - Understand the application support requirements of the business services and processes carried out by an organization unit
  - Support the gap analysis and determine whether any of the applications are missing and as a result need to be created
  - Define the application set used by a particular organization unit

# Example System/Organization Matrix

| APPLICATION (Y-AXIS) AND ORGANISATION UNIT (X-AXIS) | CUSTOMER SERVICES | PROCUREMENT AND WAREHOUSING | HR | CORPORATE FINANCE |
|---|---|---|---|---|
| SAP HR | X | X | X | |
| SIEBEL | X | X | | |
| SAP FINANCIALS | X | X | | X |
| PROCURESOFT | X | X | | |

# Role/System Matrix

- The purpose of the Role/System matrix is to depict the relationship between systems (i.e., application components) and the business roles that use them within the enterprise.
- The mapping of the Application Component-Role relationship is an important step as it enables the following to take place:
  - Assign usage of applications to the specific roles in the organization
  - Understand the application security requirements of the business services and processes supporting the function, and check these are in line with current policy
  - Support the gap analysis and determine whether any of the applications are missing and as a result need to be created
  - Define the application set used by a particular business role; essential in any move to role-based computing

# Example Role/System Matrix

| APPLICATION (Y-AXIS) AND FUNCTION (X-AXIS) | CALL CENTRE OPERATOR | CALL CENTRE MANAGER | FINANCE ANALYST | CHIEF ACCOUNTANT |
|---|---|---|---|---|
| SAP HR | X | X | X | X |
| SIEBEL | X | X | | |
| SAP FINANCIALS | X | X | X | X |
| PROCURESOFT | X | X | | |

# System/Function Matrix

- The purpose of the System/Function matrix is to depict the relationship between systems (i.e., application components) and business functions within the enterprise.
- The mapping of the Application Component-Function relationship is an important step as it enables the following to take place:
  - Assign usage of applications to the business functions that are supported by them
  - Understand the application support requirements of the business services and processes carried out
  - Support the gap analysis and determine whether any of the applications are missing and as a result need to be created
  - Define the application set used by a particular business function

# Example System/Function Matrix

| APPLICATION (Y-AXIS) AND FUNCTION (X-AXIS) | CALL CENTRE 1ST LINE | WAREHOUSE CONTROL | VACANCY FILLING | GENERAL LEDGER MAINTENANCE |
|---|---|---|---|---|
| SAP HR | X | X | X | X |
| SIEBEL | X | X | | |
| SAP FINANCIALS | X | X | | X |
| PROCURESOFT | X | X | | |

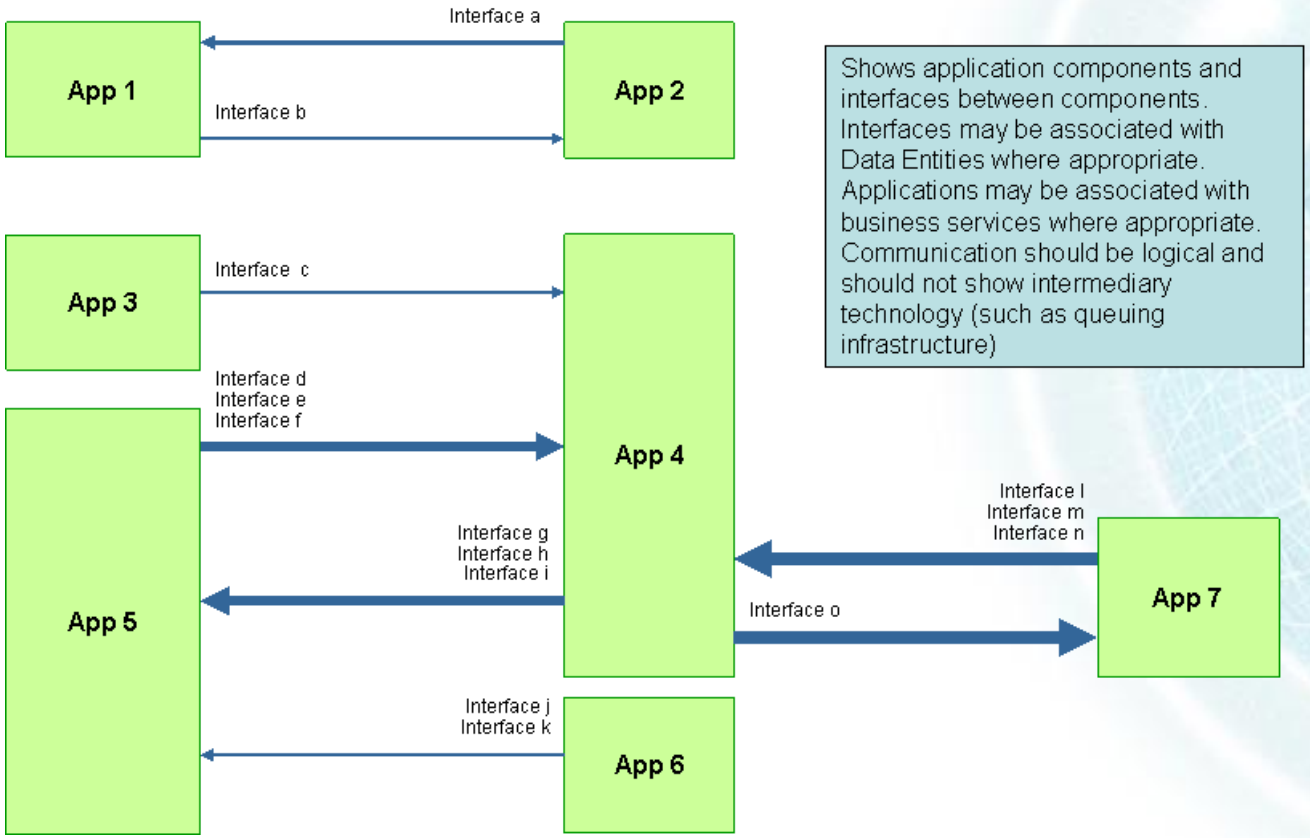
# Diagrams

- Application Communication diagram
- N2 model or Node Connectivity diagram
- Application and User Location diagram
- System Use-Case diagram
- Enterprise Manageability diagram
- Process/System Realization diagram
- Software Engineering diagram
- Application Migration diagram
- Software Distribution diagram

# Application Communication Diagram

- The purpose of the Application Communication diagram is to depict all models and mappings related to communication between applications in the metamodel entity.
- It shows application components and interfaces between components.
- Communication should be logical and should only show intermediary technology where it is architecturally relevant.

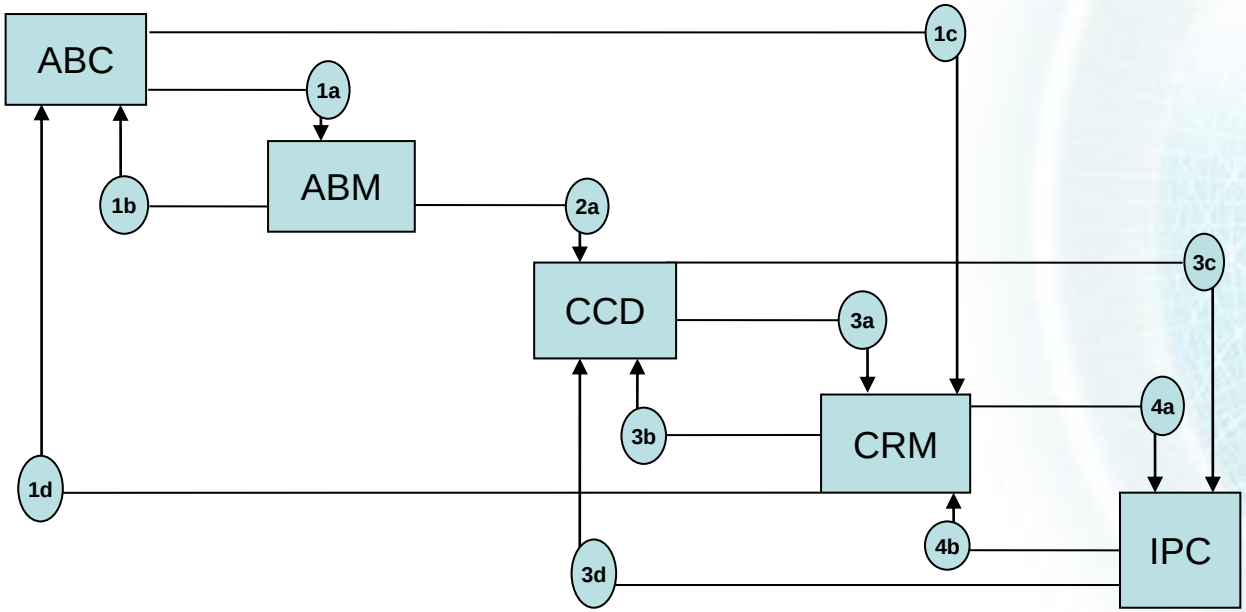
# Application Communication Diagram






TOGAF™

# N2 Model






TOGAF™

# Information Exchange Matrix

| LABEL | SOURCE | DESTINATION | DATA ENTITY | EVENT TRIGGERED |
|---|---|---|---|---|
| 1a | ▪ ABC | ▪ ABM | ▪ Sales order (create request) | ▪ New sales order from front end |
| 1b | ▪ ABM | ▪ ABC | ▪ Sales order (confirm create) | ▪ Order created in the backend ERP system |
| 2a | ▪ ABM | ▪ CCD | ▪ Product catalog | ▪ Subscribe/Publish timer |

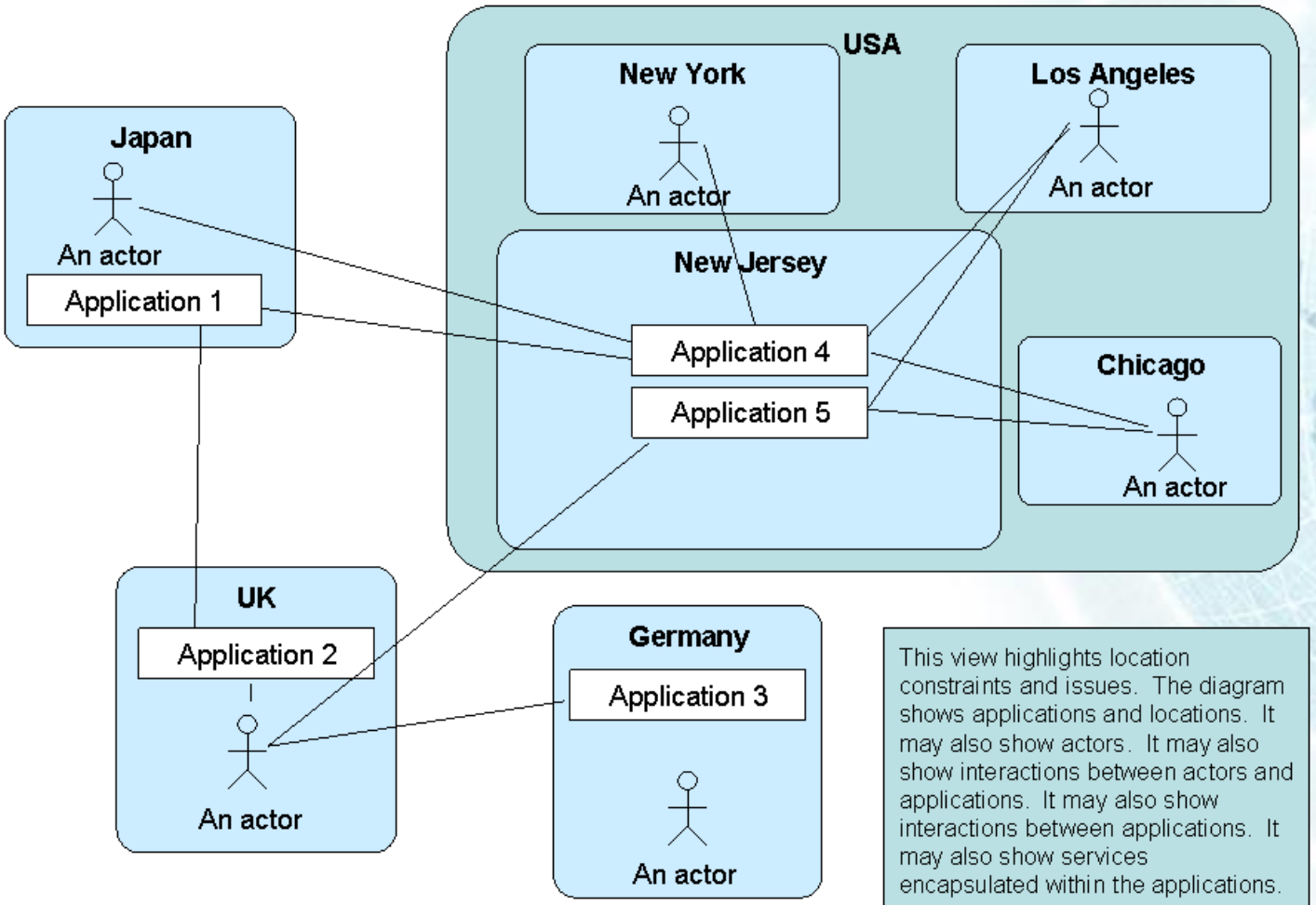
# Application & User Location Diagram

- The purpose of this diagram is to clearly depict the business locations from which business users typically interact with the applications, but also the hosting location of the application infrastructure.
- The diagram enables:
  - Identification of the number of package instances needed
  - Estimation of the number and the type of user licenses
  - Estimation of the level of support needed
  - Selection of system management tools, structure, and management system
  - Appropriate planning for the technological components of the business
  - Performance considerations while implementing solutions

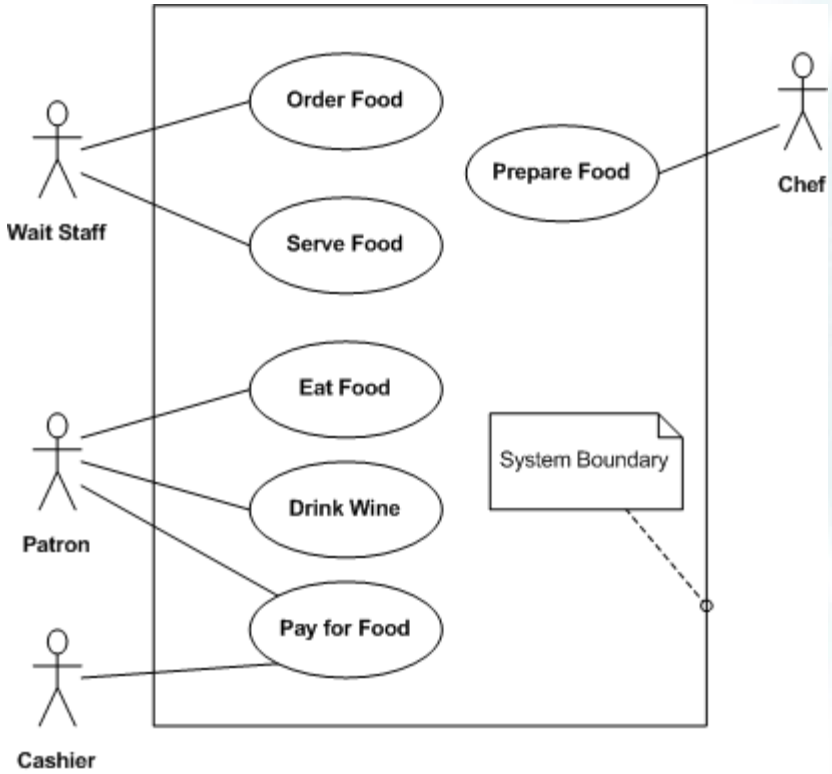
# Example Application & User Location Diagram (part 1)

| APPLICATION | USER TYPE | INTERNAL, CUSTOMER OR PARTNER | USER BUSINESS LOCATION | LOCATION ADDRESS | ORG UNIT (USER BELONGS TO) |
|---|---|---|---|---|---|
| CRM | Developer<br>Super User<br>Administrator | Internal | NA Western Region<br><br>EMEA Headquarters, UK | Chicago Sears tower office Chicago<br><br>Downtown office Middlesex, London | NA Sales & Marketing<br><br>EMEA Sales |
| SAP R/3 | Test Engineers<br>Mechanical Engineers<br>Procurement managers | Internal | Beijing Manufacturing Plant | | Manufacturing & logistics |

# Example Application & User Location Diagram (part 2)



# System Use Case Diagram




***Source: wikipedia.org***

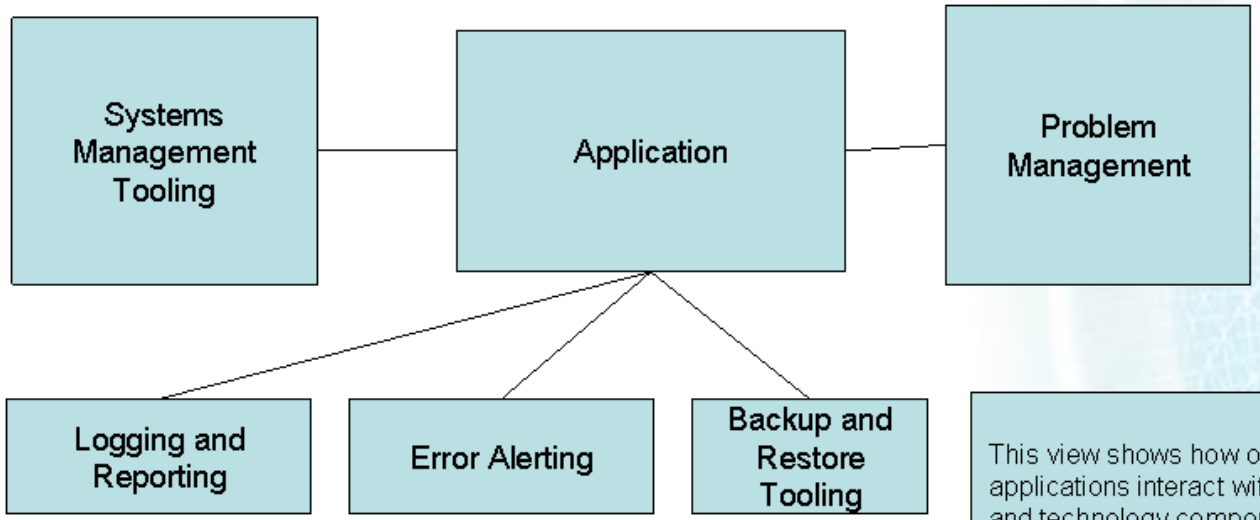


TOGAF™

# Enterprise Manageability Diagram

- The Enterprise Manageability diagram shows how one or more applications interact with application and technology components that support operational management of a solution.
- Analysis can reveal duplication and gaps, and opportunities in the IT service management operation of an organization.

# Example Enterprise Manageability Diagram



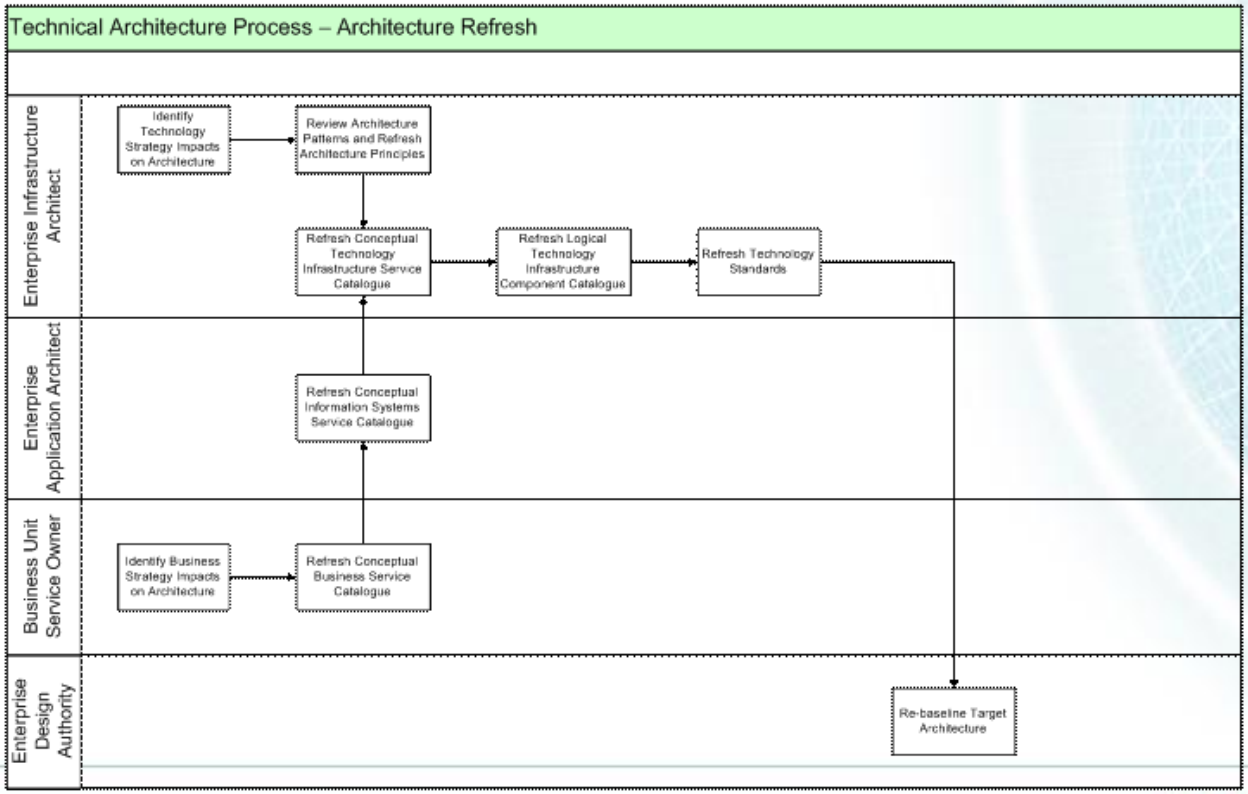

This view shows how one or more applications interact with application and technology components that support operational management of a solution. This view is really a filter on the application-application communication view. This diagram may also show support personnel access to administrative functions of the system. This diagram may also address environment migration, configuration management, etc.

# Process/System Realization Diagram

- The purpose of the Process/System Realization diagram is to clearly depict the sequence of events when multiple applications are involved in executing a business process.
- It enhances the Application Communication diagram by augmenting it with any sequencing constraints, and hand-off points between batch and real-time processing.

# Example Process/System Realization Diagram

UML sequence (most detail) and activity diagrams (less detail) can be used, or a less formal swimlaned flowchart (least detail). BPMN is also an option. The decision on diagram form will depend on the level of detail and formality. Generally, the non-formal view is best suited to stakeholders, but specific areas of architecture risk need to be addressed in more detail. The diagram can show organisations, actors, application components, data entities and architecturally significant technology components.



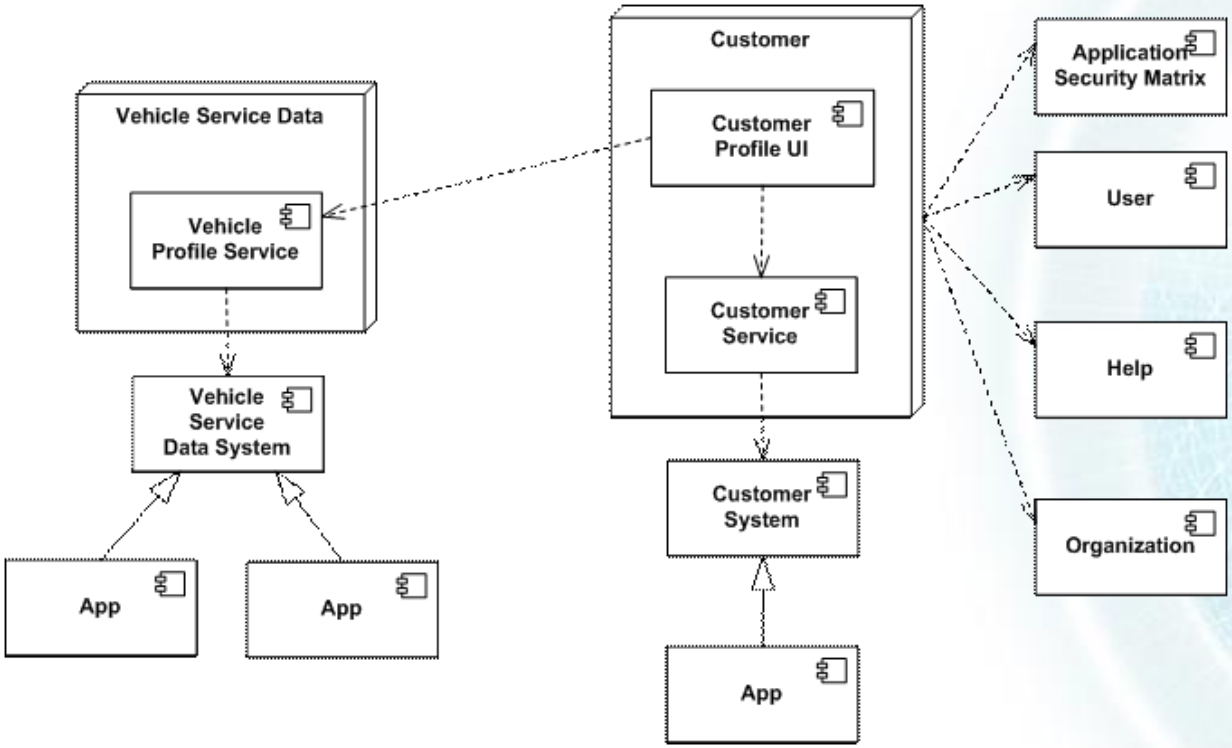



TOGAF™

# Software Engineering Diagram

- The Software Engineering diagram breaks applications into packages, modules, services, and operations from a development perspective.
- It enables more detailed impact analysis when planning migration stages, and analyzing opportunities and solutions.
- It is ideal for application development teams and application management teams when managing complex development environments.

# Example Software Engineering Diagram

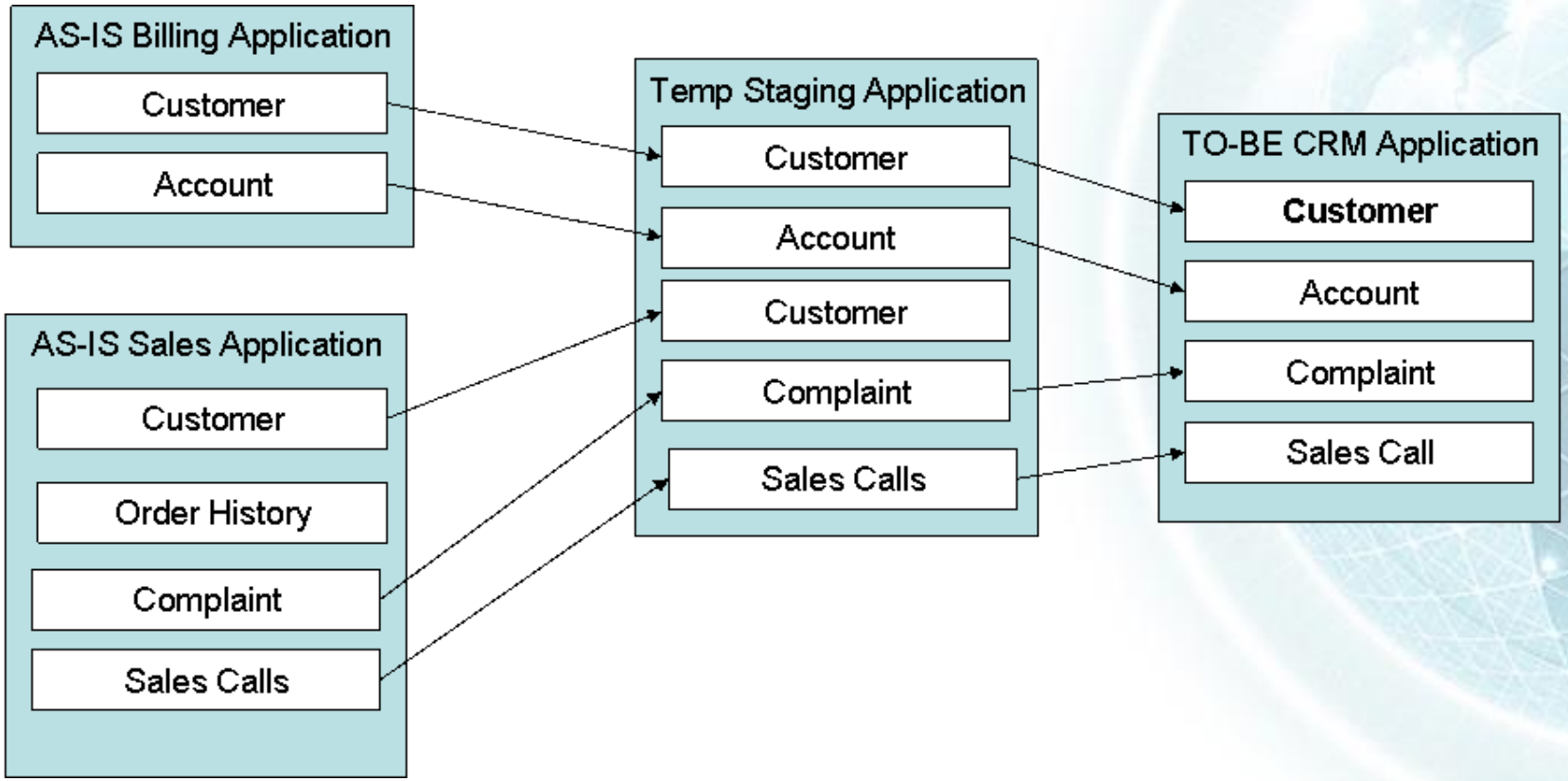



Breaks applications into packages, modules, services and operations from a development perspective. May show dependencies between functional components

# Application/Migration Diagram

- The Application Migration diagram identifies application migration from baseline to target application components.
- It enables a more accurate estimation of migration costs
- It should be used to identify temporary applications, staging areas, and the infrastructure required to support migrations

# Example Application/Migration Diagram

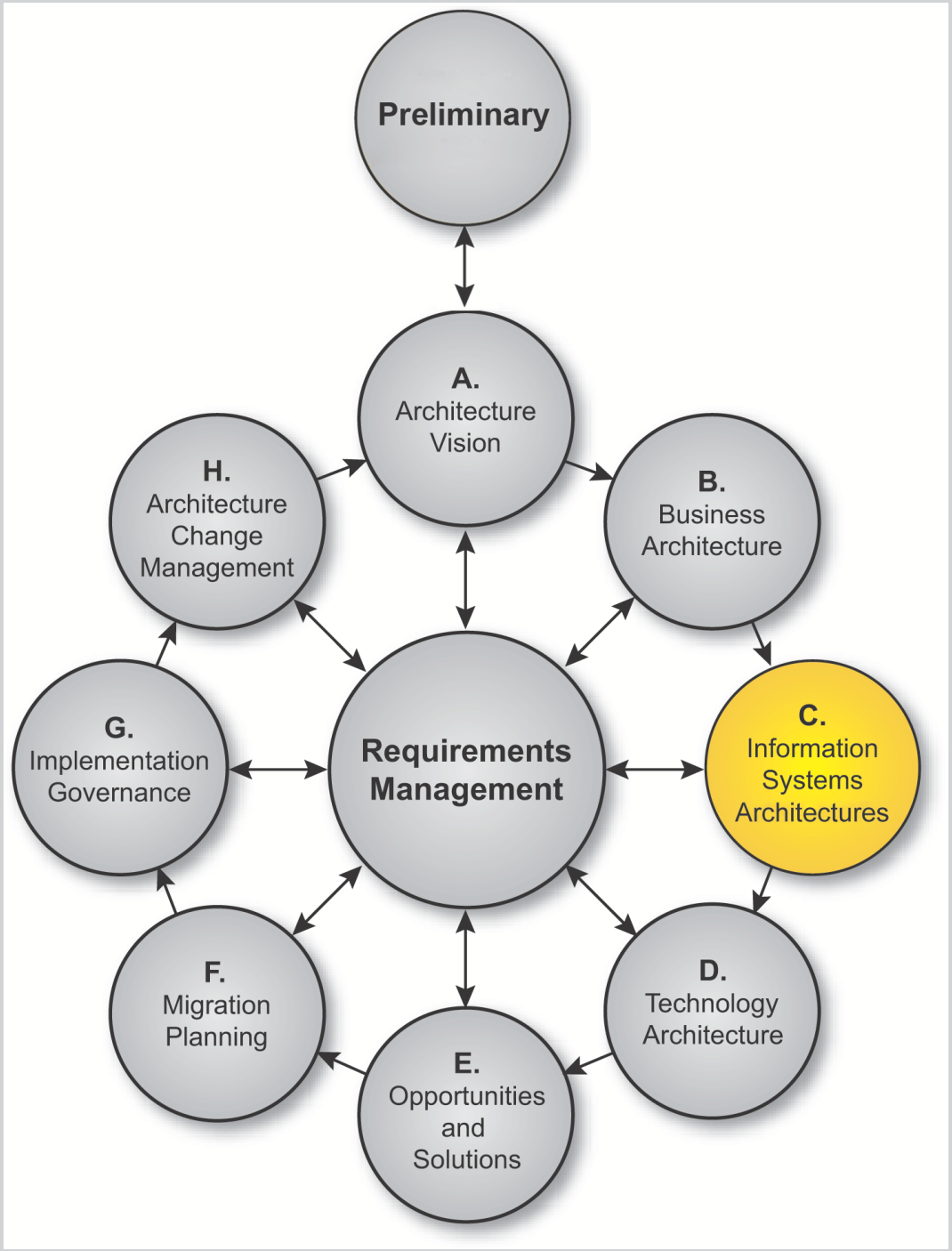




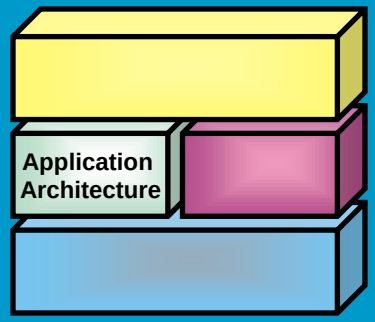

TOGAF™

# Software Distribution Diagram

- This diagram is a composite of the Software Engineering diagram and the Application-User Location diagram.
- Depending on the circumstances, this diagram alone may be sufficient, or may not be needed.




# Phase C: Application Architecture – Catalogs, Matrices and Diagrams






TOGAF™