An IT Architecture Executive Overview

Share in Long Beach
Session: 1514
February 25-March 2, 2001

George E. Kurtz III, President
GEK IT Management Consulting, Inc.
Phone 407/482-8812
Email:gekurtz@aol.com

Agenda

- Who are we?

- What are the IT Infrastructure Management Challenges?

- What kind of IT Infrastructure do we recommend?

- What are the Benefits?

Who Are We?

Beyond Integration...
Everything must work together
As a cohesive whole

- To achieve that, we combine the best:
  - Technology
  - Methodology
  - Infrastructure

What are the IT Infrastructure Management Challenges?

IT Infrastructure Management Challenges

Skill shortage

How do we attract and retain talent? Avoid losing corporate knowledge when people leave? Gain access to and capture world-class knowledge?

Communication

How do we promote communication across IT functions? Gain consensus on architecture and tool requirements?

Integration and manageability

How do we integrate the traditionally disparate disciplines of application development, deployment, systems management and data management?

Transition

How do we migrate from a product-driven to process-driven ESM approach? Migrate from a procedure-oriented to CBD approach? Incrementally migrate to a complete and persistent data architecture and management process?

What kind of IT Infrastructure do we recommend?

A Process-Driven Architecture

- **A Fully Integrated**

- **A Systematic Approach**

- **Use a common language and methodology**

- **Ensure there are fully-integrated processes for:**
  - Application development
  - Application deployment
  - Data Management
  - ESM

Infrastructure Objectives

- **Establish world-class IT organization**
  - Get quality products to market faster
  - Low cost provider
  - Assimilate new acquisitions
  - Expand international presence

- **Capture "intellectual capital" in re-usable form**

- **Position IT as a profit center**

- **30-day development cycles or less**

- **Portable applications** (platform, data, network and user interface)

- **Automatic systems management**

- **A common model across the enterprise**

- **Reduce costs and improve ROI**

Our IT Architecture

Business Model

Rapid Applications Development Environment

Automatic Systems Management

Vendor Partnerships
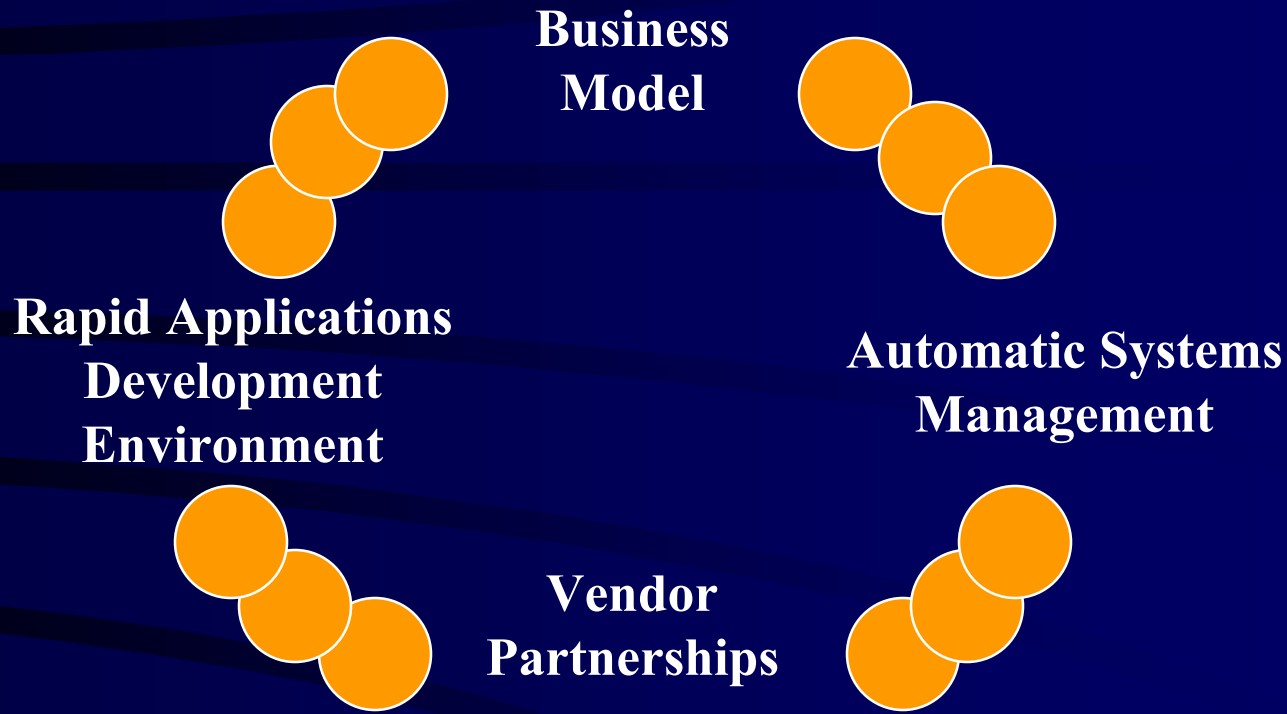
A Well-Defined Infrastructure		
Policies & Methodology	Organization & People	Tools & Automation
• Standard	• Ownership/Accountability	• Functionality
• Processes/Procedures	• Skills	• Integration
• Enforcement	• Training	
• Documentation		

Technical Architecture
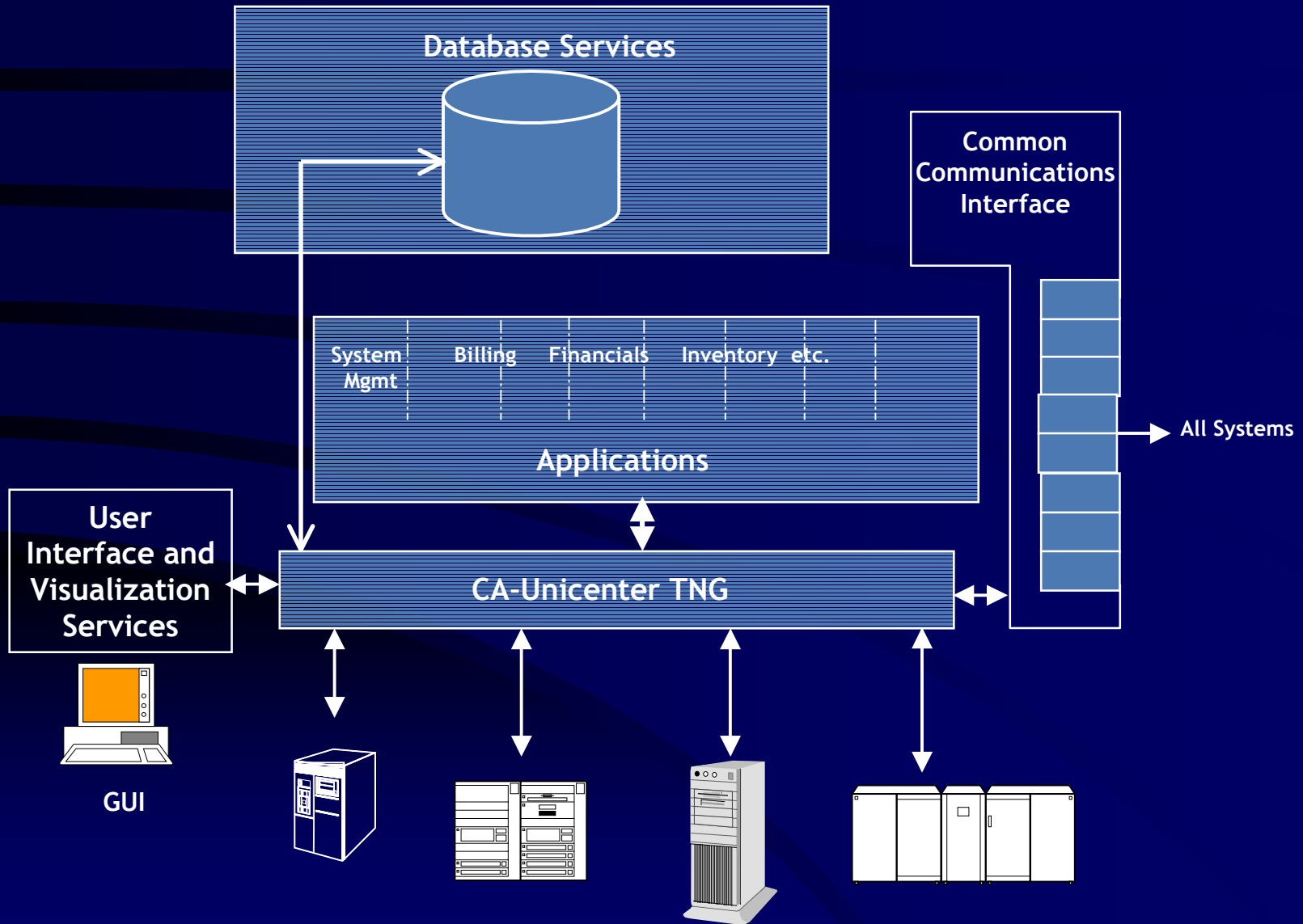
Strategy

- Develop a fully integrated processes for:
  - Application development
  - Application deployment
  - ESM
  - Data Management
  - Use Catalysis to provide the common language and methodology to help integrate the above functions

About *Catalysis*™

A *next-generation* *standards-aligned* method

    For *open distributed component systems*

        from *components* and *frameworks*

        that reflect and support *an adaptive enterprise*

Catalysis has been in development and use since 1992
Supports components, OO, legacy, heterogeneous systems
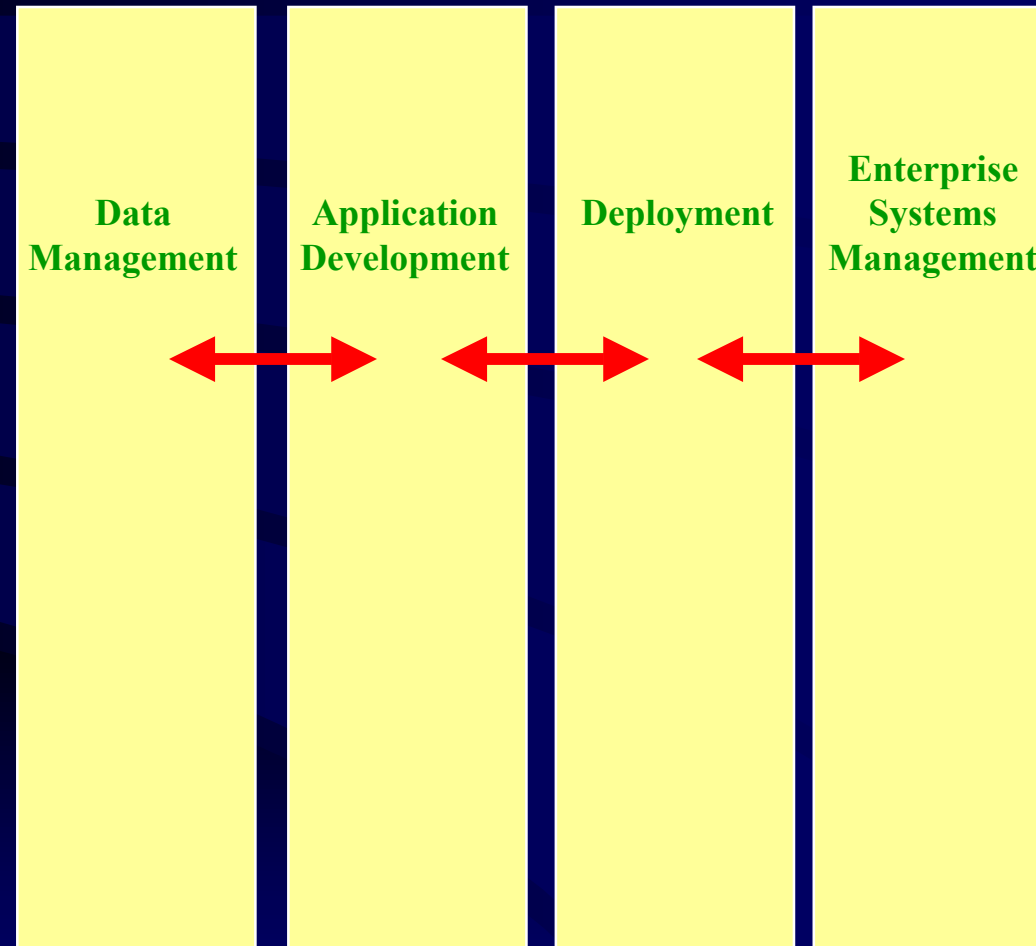Addison Wesley, *"Objects, Components, Frameworks…"* 1998,
D'Souza & Wills

OBJECTS, COMPONENTS, AND FRAMEWORKS WITH UML
THE CATALYSIS APPROACH
DESMOND F. D'SOUZA
ALAN C. WILLS
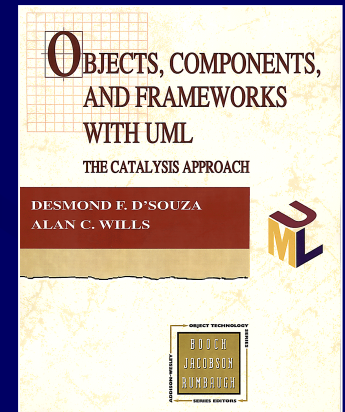
Catalysis - High Level View
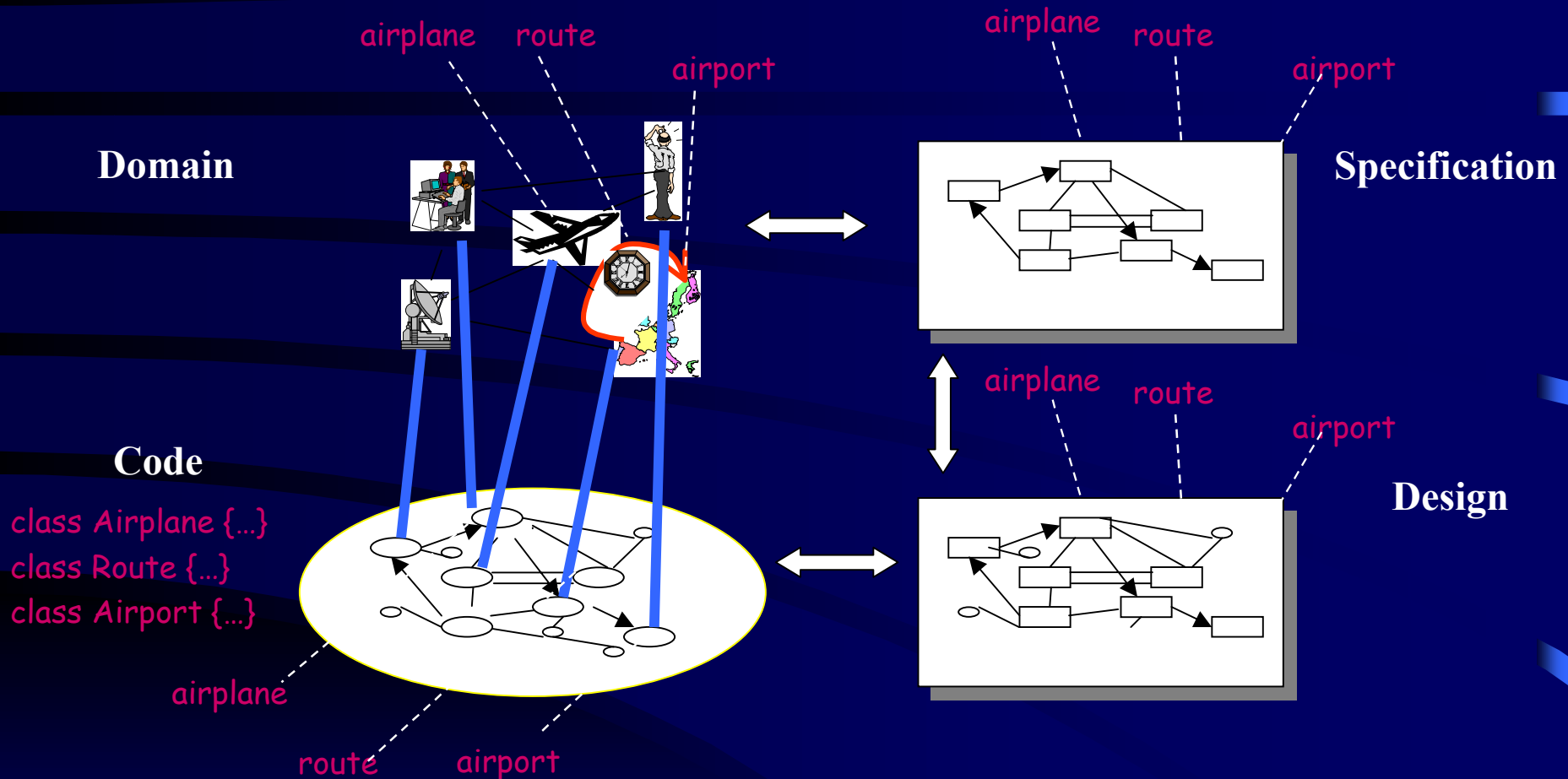
- – Similar <u>domain terms</u> appear at all levels from requirements to code
- – Similar <u>modeling constructs</u> are used to describe them at all levels
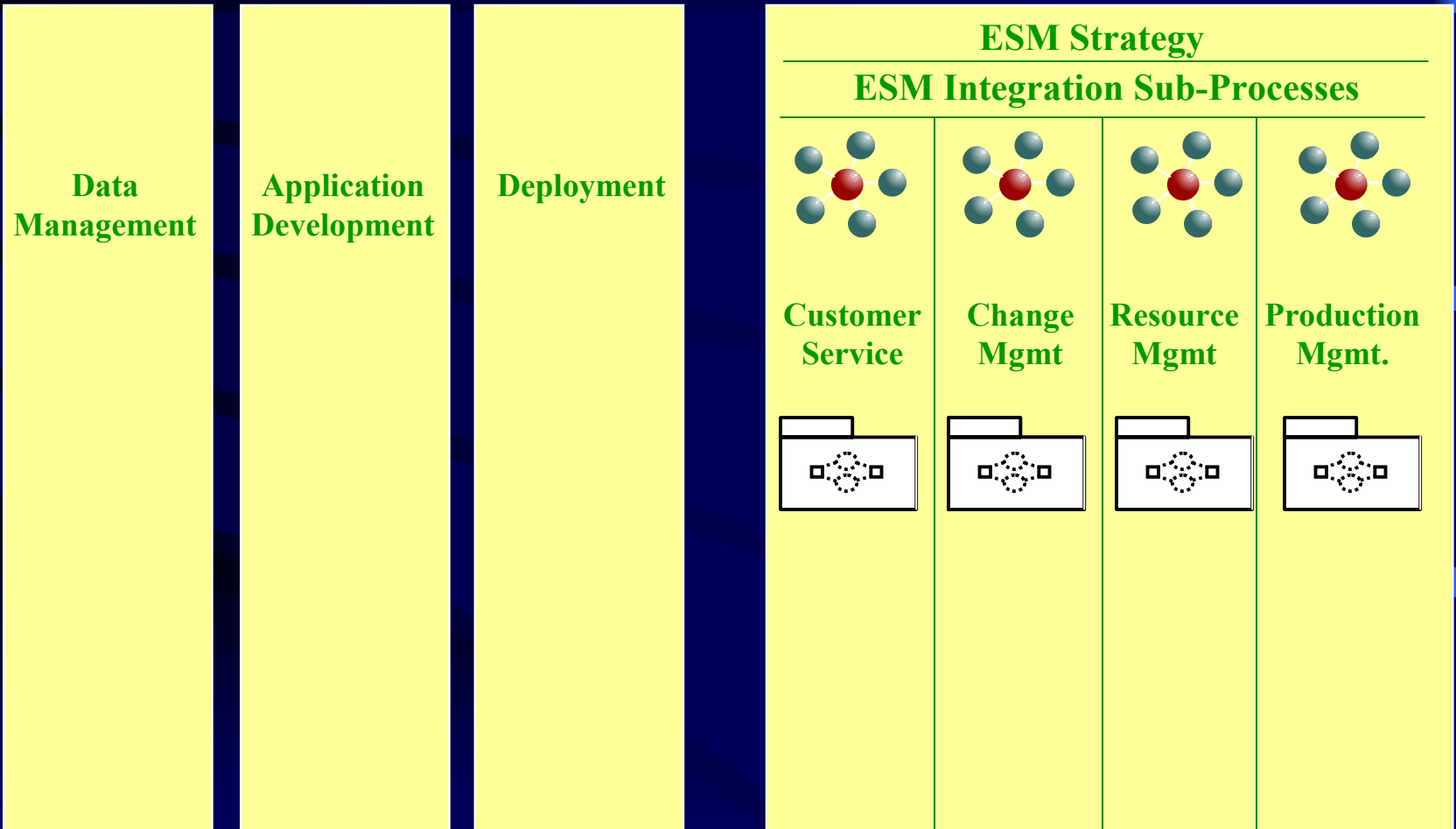
14

UML Development Process with Catalysis

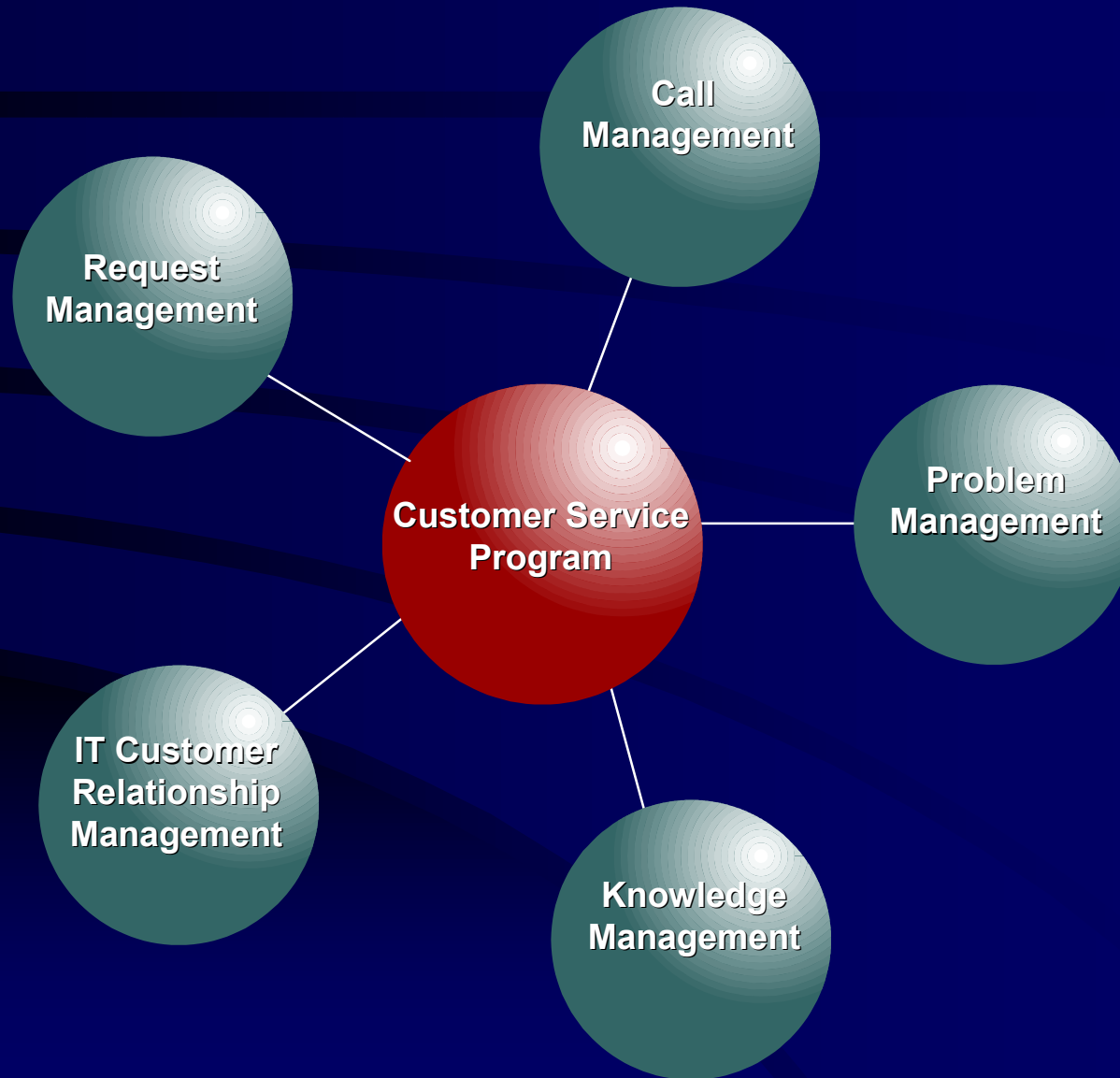UML = Unified Modeling Language, standard notation for OO design

Domain/Business

Component Spec

Internal Design

Requirements

Understand the business problem, system context and requirements.

Domain/Business Models

System Context

System/Component Spec

Describe external behavior of target system using problem domain model

Scenarios

Type Model and Op Specs

Internal Architectural Design

Partition technical and application architecture components and their connectors to meet design goals

Platform, Physical Architecture

Logical Application Architecture

Internal Component Design

Design interfaces and classes for each component; build and test

Interface and Class Specs

Implementation and Test

Dictionary

UI Design

dialog flow, prototype, usability

DB Design

class mapping, transactions, etc.

Enterprise Infrastructure Management Processes

Data Management	Application Development	Deployment	Enterprise Systems Management

Enterprise Systems Management
Sub-Process Work Breakdown

Data Management	Application Development	Deployment	ESM Strategy

ESM Strategy

ESM Integration Sub-Processes

Customer Service	Change Mgmt	Resource Mgmt	Production Mgmt.

Strategist's View of Customer Service Domain

Modeler's View

Customer Service Domain Models

Essential Model

- Defines problem space
- Captures business process knowledge

refinement

refinement

As-Is Model

To-Be Model

- Current process
- Current roles & responsibilities
- Current tool capabilities
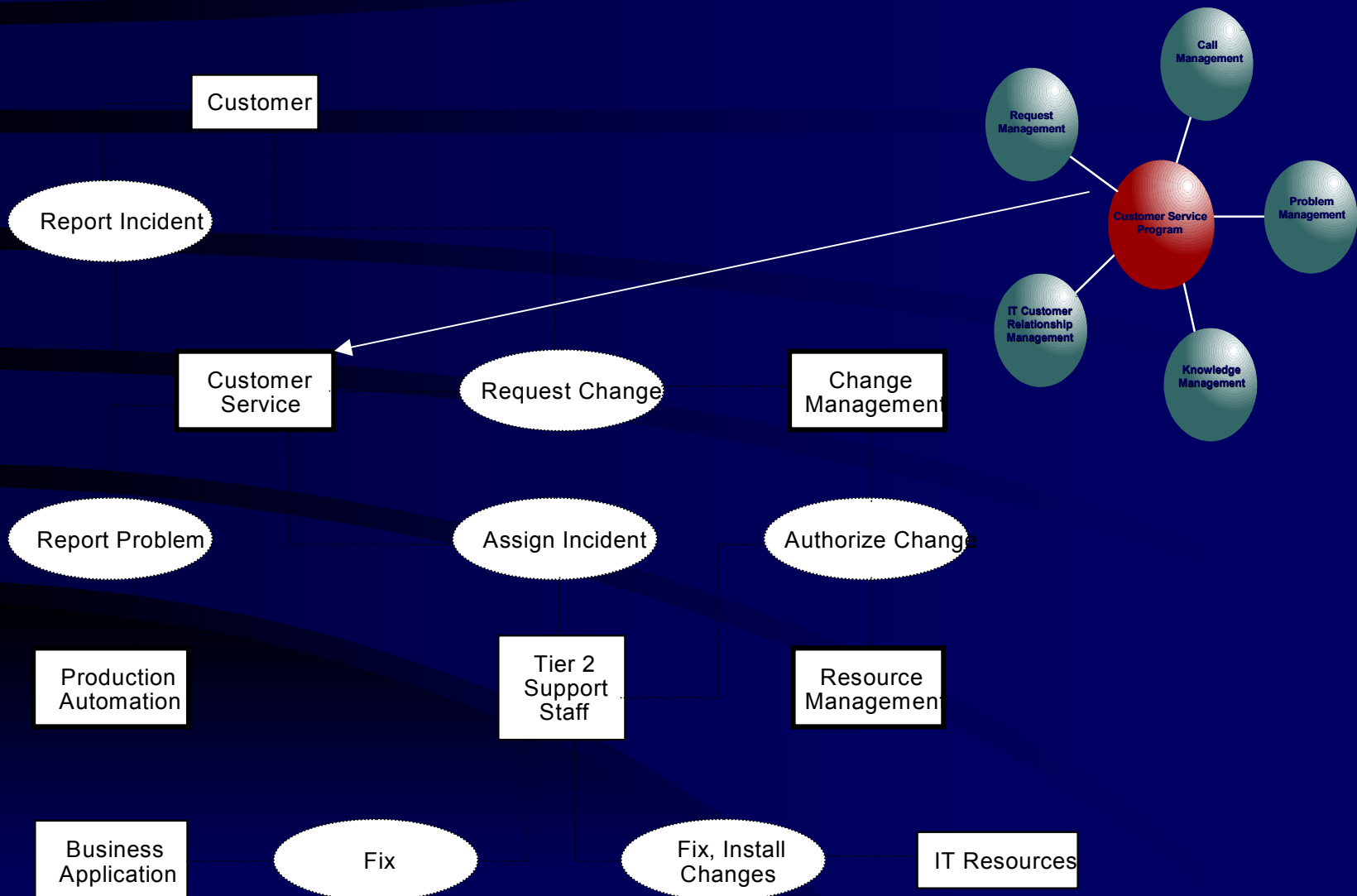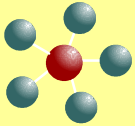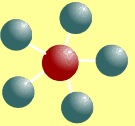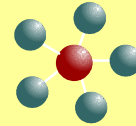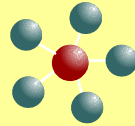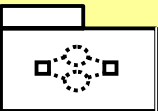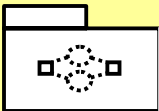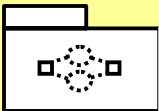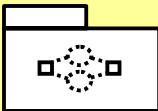
- Future process
- Future roles & responsibilities
- Tool requirements
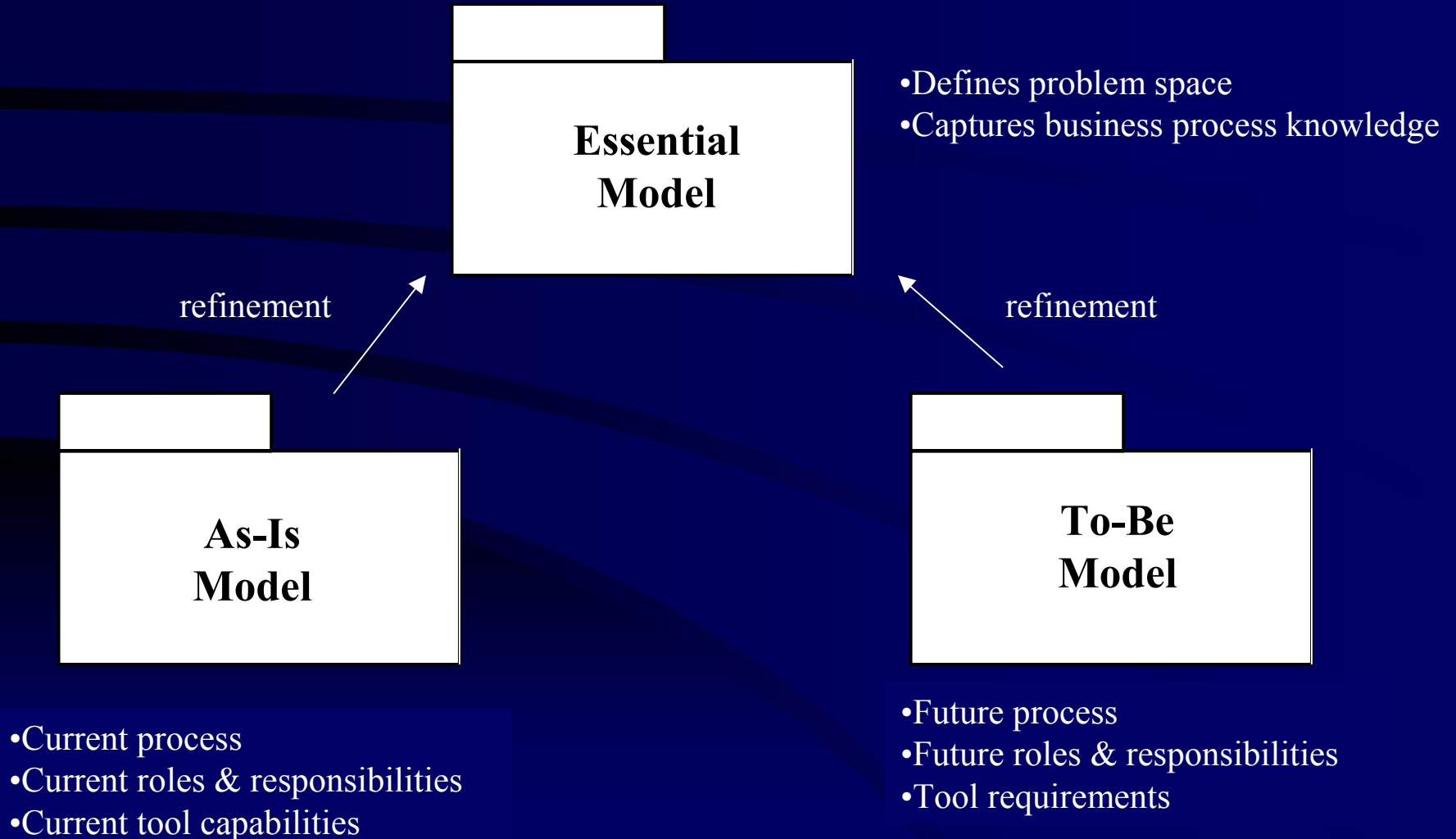
Application Development Sub-Process Work Breakdown

Data Management

Component Based Development Strategy

Best Practices	Architecture	Software Manufacturing Setup	Reuse Program

Deployment

Enterprise Systems Management
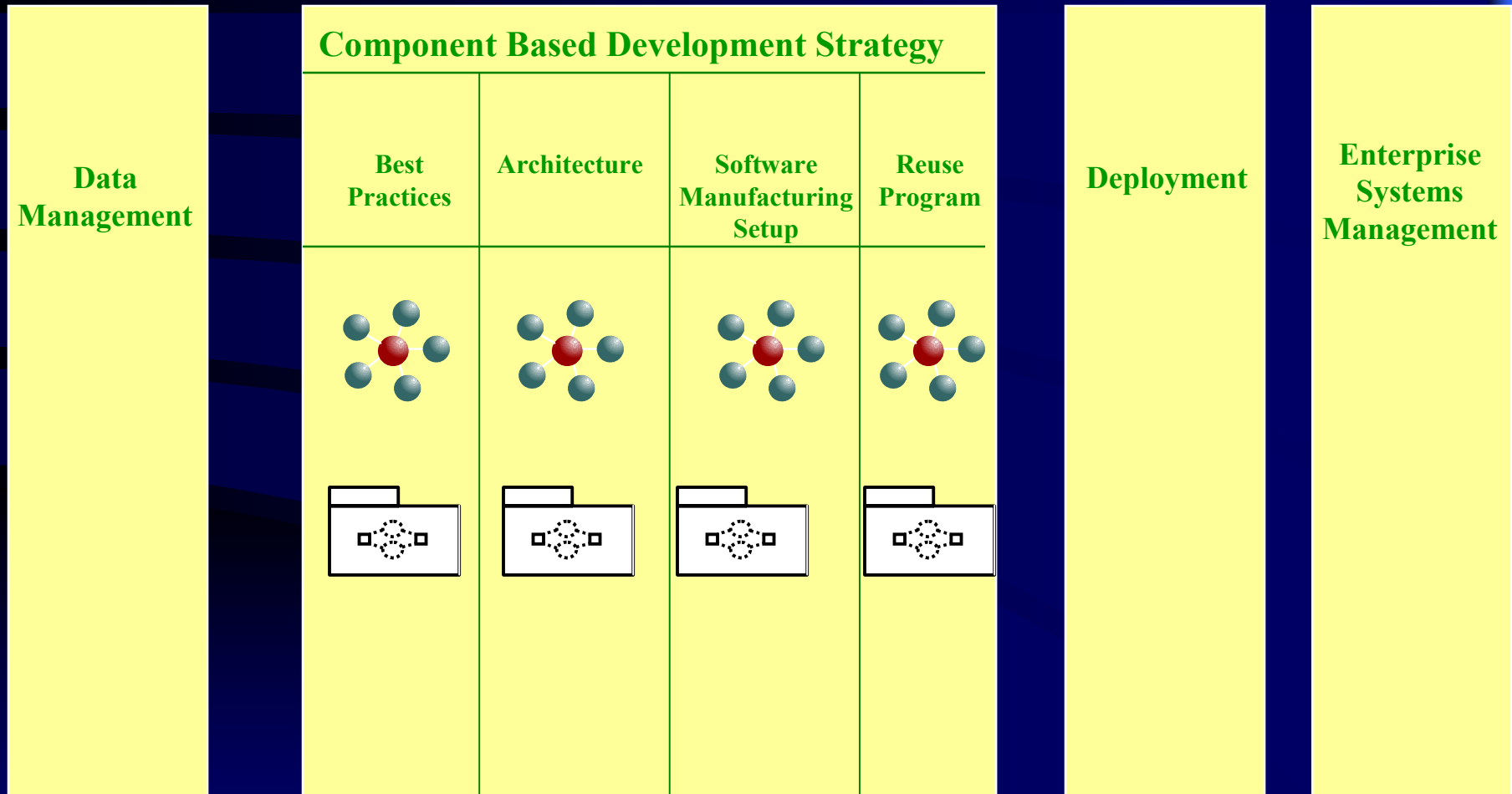
Deployment Sub-Process Work Breakdown

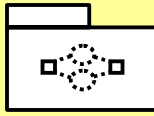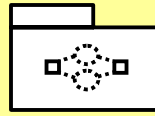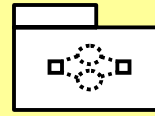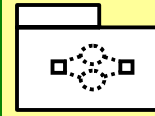| Deployment Strategy			
Deployment Planning	**Rollout Management**	**Package Distribution**	**Data Migration**
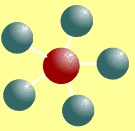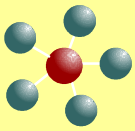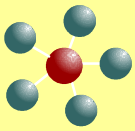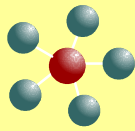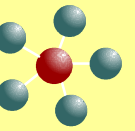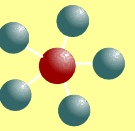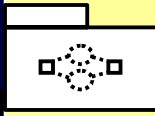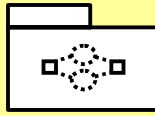
Application Development

Data Management

Enterprise Systems Management

Data Management Sub-Process Work Breakdown

Data Management Strategy						Application Development	Deployment	Enterprise Systems Management
Meta Data Track	Data Architecture	Enterprise Data Mode	Enterprise Reference Files	Data Warehouse/ Business Intelligence	Legacy Integration			

Benefits

- Systematic approach to <u>process</u> definition and <u>integration</u>

- Knowledge capture - insurance against people leaving

- Thoughtful transition plan

- IT-wide consensus, buy-in and common language

- Rigorous tool requirements