

Model Management for the Composer Encyclopedias

Session 130

Adel Harris
Texas Instruments

© Texas Instruments 1996

1



Model Management Topics

- Definitions and objectives
- Composer Encyclopedia architecture and Version Control functionality
- Factors and complexities
- Model types and strategy examples
- Practices to ensure success of encyclopedia usage and activities

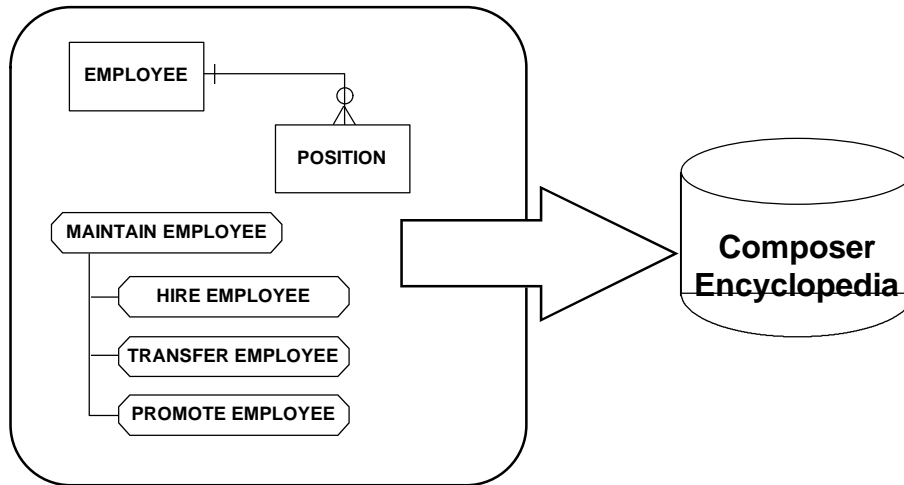
© Texas Instruments 1996

2



What is a Model?

Composer Model



What is a Model Management Strategy?

- A chosen method of organizing the Composer Encyclopedia(s) in terms of models and their interrelationships
- Defined plan for the initiation and maintenance of models, and the propagation of change between models using the encyclopedia(s)

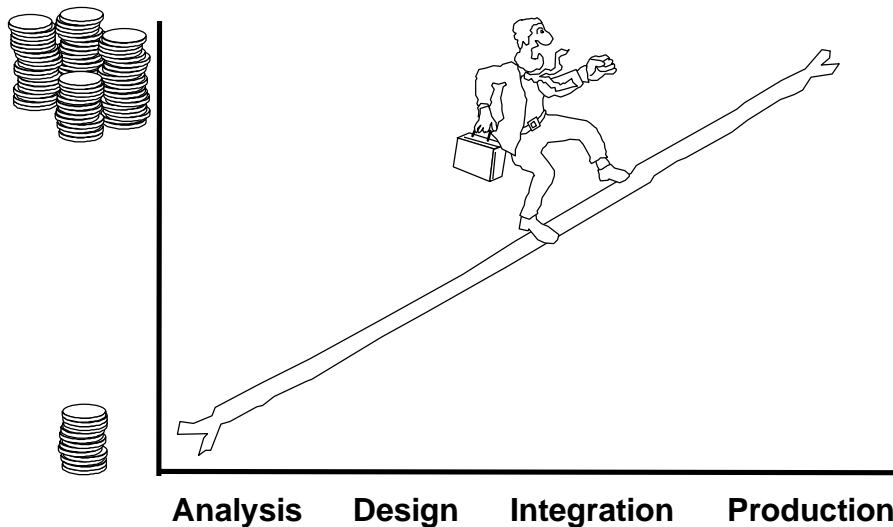


Why a Model Management Strategy?

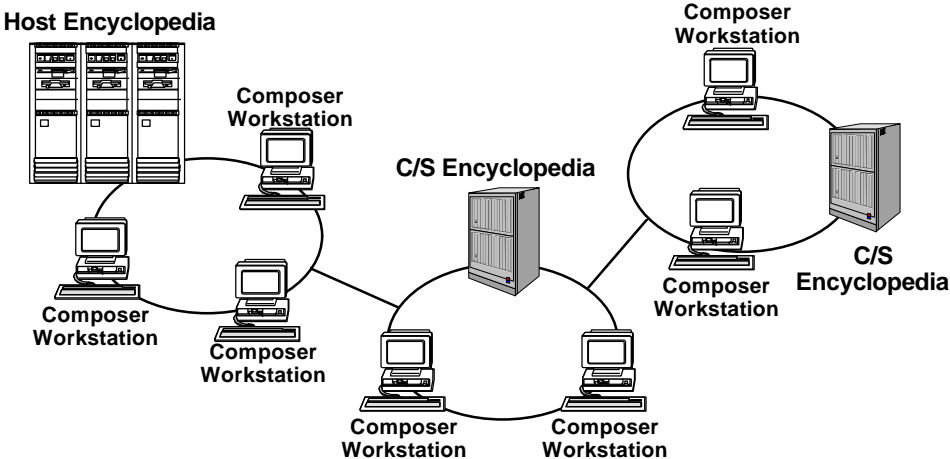
- For model-based development, it provides the structure and mechanism to:
 - Attain desired level of consistency and integration across development projects and operational systems
 - Initiate and maintain concurrent, interrelated development and maintenance projects
 - Implement requirements for change, configuration, and release management for Composer model objects



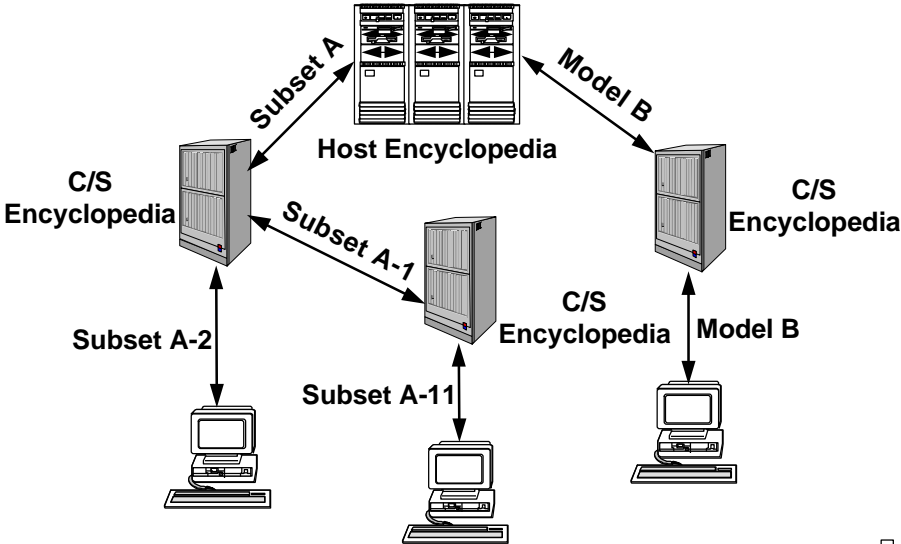
Cost of Implementing a Strategy



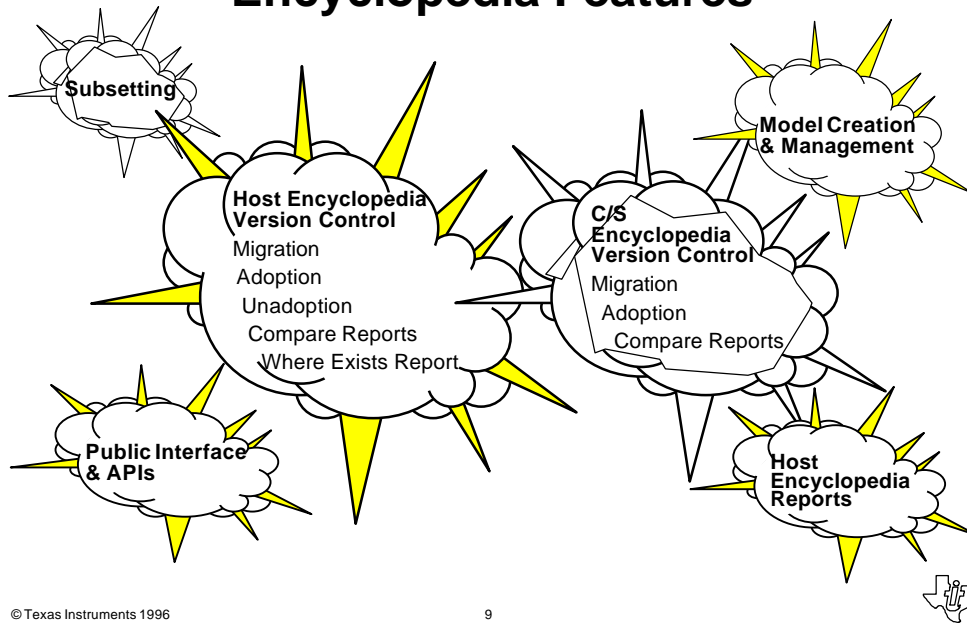
Encyclopedia Architecture



Movement Between Encyclopedias



Encyclopedia Features

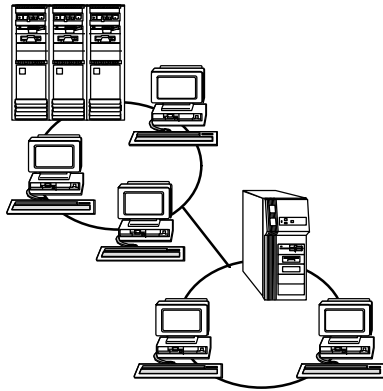


Version Control Differences

Basis for Common Ancestry	Migration and Model Family
Adoption Report Options and Terminology	Compare Report Options and Level of Granularity



Composer Encyclopedia Advantages



- Integrated set of repository products
- Scalable and flexible development architecture
- Multiple team development environment
- Ability to distribute encyclopedia functionality and decentralize control

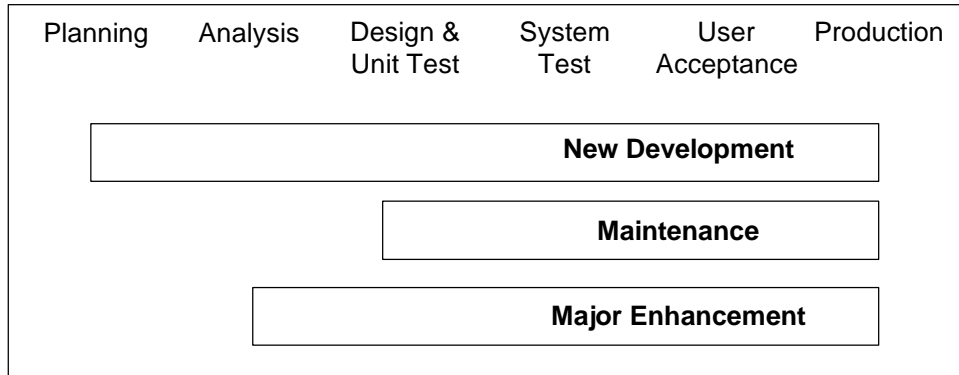


Model Management Factors

- Number of encyclopedia models
 - Diversity
 - Complexity
 - Evolution
 - Technical
- Interrelationships between models
 - Consistency and integration targets
 - Organizational Structures
 - Concurrent development and maintenance projects



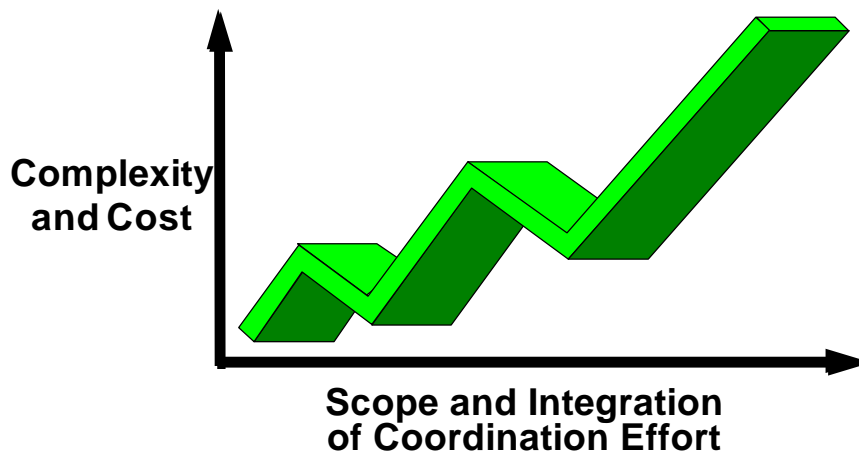
Concurrent Projects



- Active projects may exist in more than one phase
- More than one active project may exist in the same phase



Model Management Continuum



Model Types

- Centralized or Support Models
 - Manage reusable and common objects
 - Integrate highly related projects for a given release
 - Build and control physical data structure
 - Source of Composer objects for test and production applications
 - Responsibility of centralized support group or task force

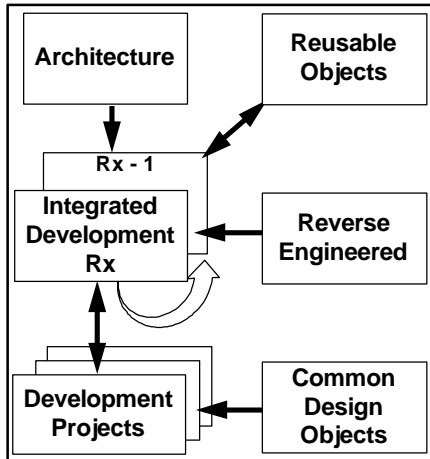


Model Types (cont.)

- Project Models
 - Represent the scope of the project within a specific life-cycle phase
 - Support work-in-progress for development and maintenance teams
 - Responsibility of project team (e.g., Team Model Manager)



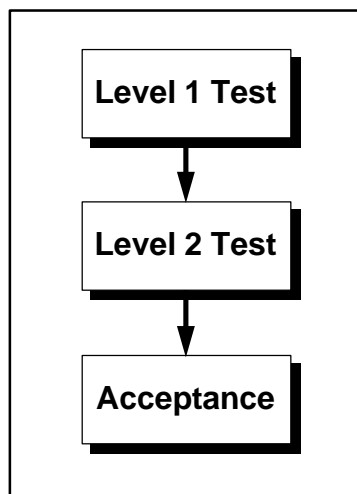
Development Models



- Potential Centralized Models
 - Architecture Model
 - Reusable Objects Model
 - Integrated Development Model
 - Reverse Engineered or Current System Model
 - Common Design Objects Model



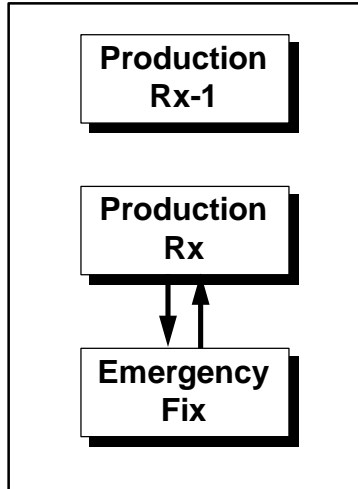
Testing Models



- Potential Centralized Models
 - Level 1 Test Model
 - Level 2 Test Model
 - Level n Test Model
 - Acceptance Model



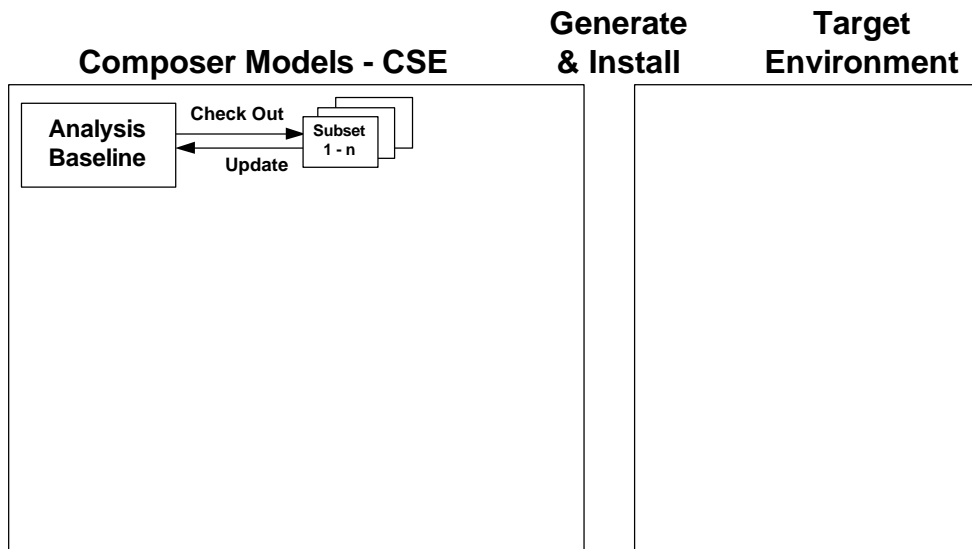
Production Support Models



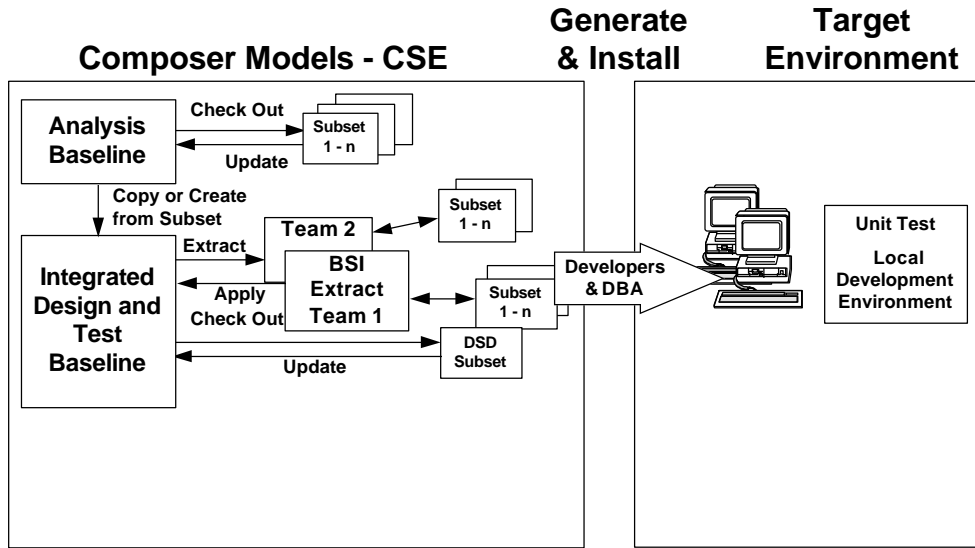
- Potential Centralized Models
 - Production
 - Previous Production Release
 - Emergency Fix Model



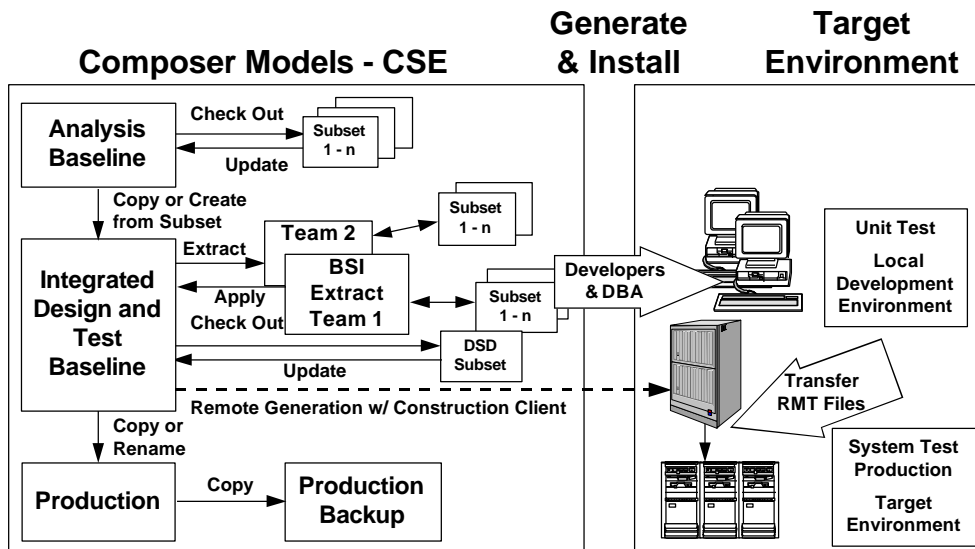
Simple Example – Analysis



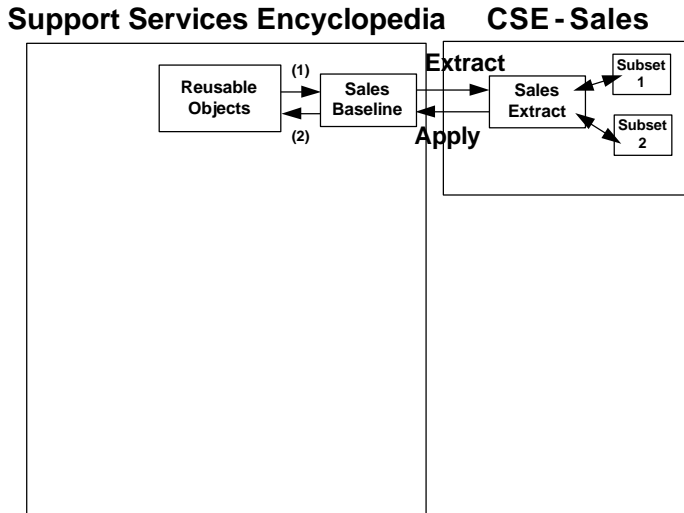
Simple Example – Design



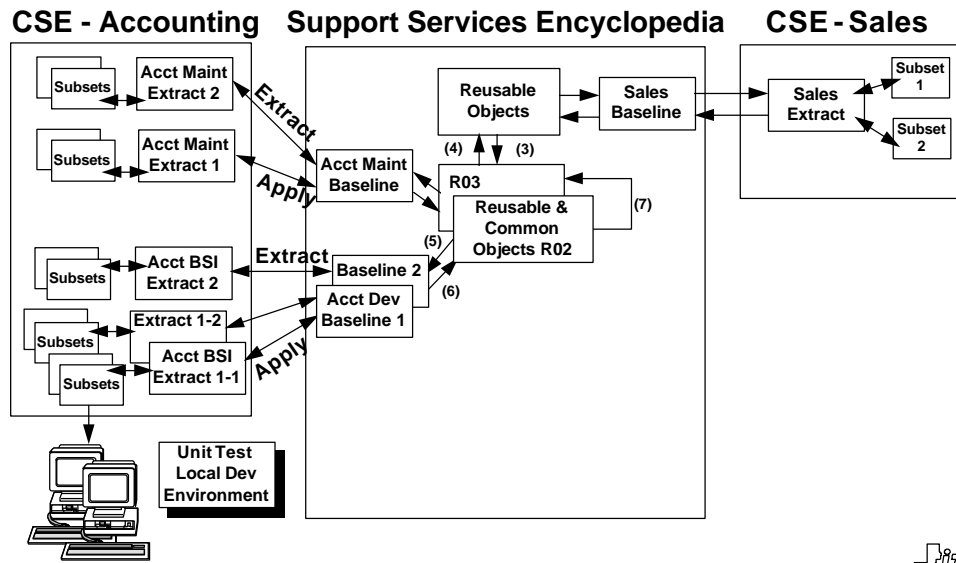
Simple Example – Test and Production



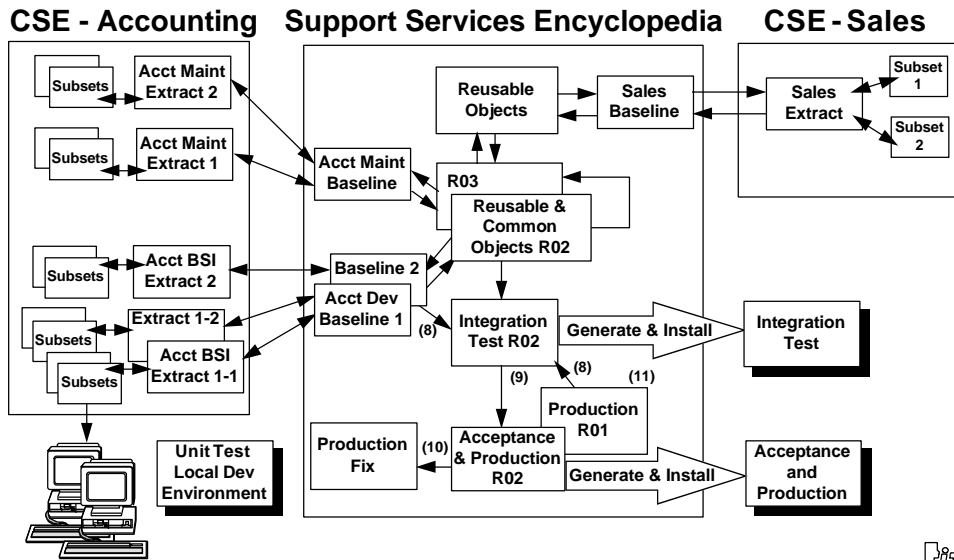
Complex Example – Analysis



Complex Example – Design



Complex Example – Test and Production



Completing the Strategy Definition

- Based on model architecture diagram
 - Model definitions
 - Flow (interrelationships) definitions
 - Detailed procedures and associated roles to manage models and flows



Model Definition Example

Model	Integrated Analysis Model
Custodian	Information Architect
Purpose	To provide an integrated, stable, and consistent data and activity model to facilitate sharing of objects across project models.
Contents	The object types include: subject areas, entity types, attributes, relationships, functions, and processes.
Creation Method	Create a new model on the Composer workstation, add the preliminary analysis information, and update the encyclopedia with or without checkin.



Model Definition Example (cont.)

Model	Integrated Analysis Model
Update Method	The majority of the change will be migrated from the BAA project models and Integrated Design Model upon approval of the Information Architect.
Duration	Indefinite. It is an active model and will be updated with new analysis objects and modifications to existing analysis objects.
Versioning	This model will always contain the versions of objects being used by active BAA projects. In addition, this model may be copied at significant milestones.
Encyclopedia Name	Integrated Analysis Model Vxx Rxx A O



Flow Definition Example

Flow	From a project analysis model to Integrated Analysis Model
Object Types	Subject areas, entity types, attributes, relationships, functions, and processes.
Method	Migration of new and modified objects; and Delete/Rename Objects within a Model for deleted objects.
Condition	Part of Change Management Procedures, change request has been approved and migration prescheduled with custodian project team.
Performed by	Enterprise Model Manager
Procedure	Refer to Implementation of Approved Change Request procedures



Procedure Definition Example

Create Project Analysis Model

Task	Responsibility
List entity types and activities from Planning Model and Integrated Analysis Model which team will be the custodians and will reference/use.	Project Manager
List project team encyclopedia users and required authorizations.	Project Manager
Verify new project scope and custodian assignment. Consult with project manager to develop approved list.	Information Architect
Create subset definition against the Integrated Analysis Model; contains the most current and detailed versions of objects.	Model Manager



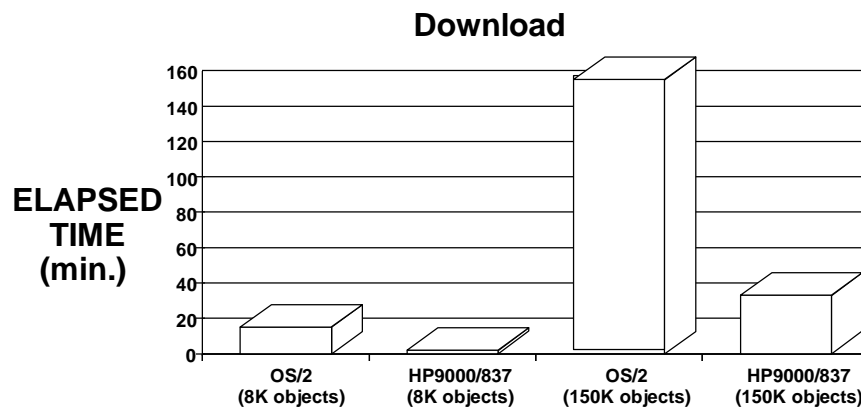
Procedure Definition Example (cont.)

Create Project Analysis Model

Task	Responsibility
Create project model from existing subset definition.	Model Manager
Migrate remaining listed objects from the Planning Model.	Model Manager
Notify Network and Encyclopedia Administrators of project model creation.	Model Manager
Establish project team network users ids and authorizations.	Network Administrator
Establish encyclopedia(s) users ids and model authorizations.	Encyclopedia Administrator



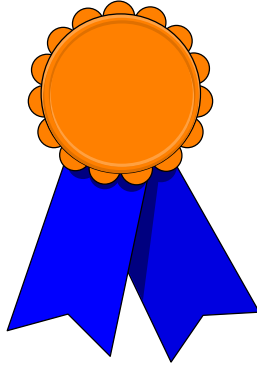
Factors for Success



Right-Sizing Encyclopedia Resources



Practices for Success



- Organizational
 - All users must support and follow encyclopedia usage guidelines
 - Infrastructure procedures (security, maintenance)
 - Training is critical
- Minimizing contention
 - Effective subsetting
 - Offload large, contentious activities
 - Additional extract models and encyclopedias
 - Encyclopedia Concurrence Matrix



Summary



- Scalable encyclopedia architecture with functionality at each level
- A strategy is driven by business and technical requirements. Factors influence:
 - Number and type of models
 - Interrelationships or flows between models
- A strategy must support organizational objectives while minimizing development cost and complexities
- Right-sizing architecture and development practices are essential to development success



Model Management for the Composer Encyclopedias

Session 130

Adel Harris
Texas Instruments

