Advantage Gen Encyclopedia

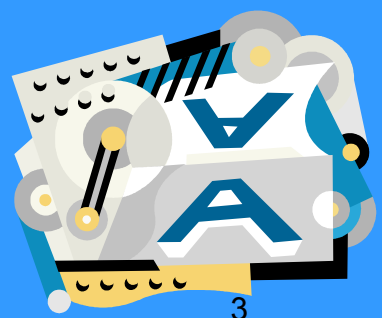
Tony R. Pierce
ActiveDevelopment Group, Inc.
NU301
May 20, 2002

What This is About…

- Encyclopedia functions overview
- Details of the encyclopedia functions
- Administration guide
- Technical design of the encyclopedias
- Common functions tips
- Encyclopedia queries

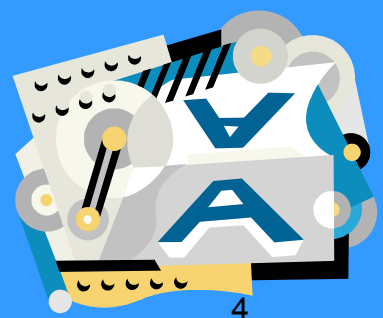
Encyclopedia Functions

- Provides a storage mechanism for models
  - A repository for model objects and associations
- Is implemented as a database and logic
- Provides access and management functions
  - Provides secured shared access to objects
  - Provides versioning and migration functions
  - Provides reports on models and encyclopedia objects



Encyclopedia Functions

- Model Checkin and Checkout (Subsetting)
- Adoption
- Migration
- Model Management (copy, delete, …)
- Reporting (comparison, conflicts, validation…)
- User Administration and Access



First, What is a Model?

- The basis of Advantage Gen development
- A set of specifications that describe a part of the “real world”
- A set of interrelated diagrams that support all phases of development
- A representation of business requirements
  - Data
  - Activities
  - User interface
  - Application navigation

Model Evolution

Planning and Analysis

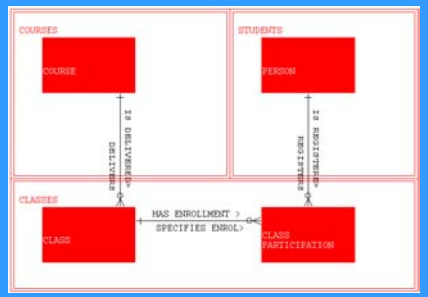
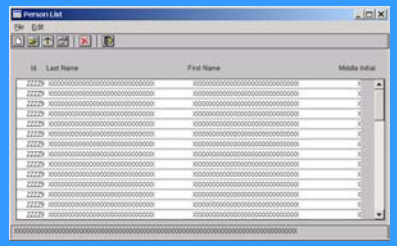
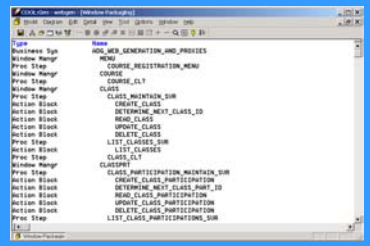
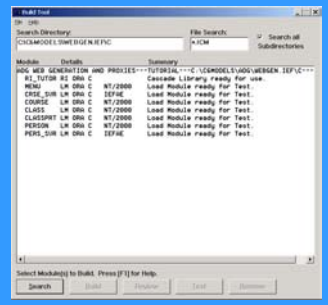
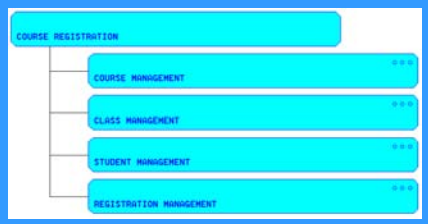
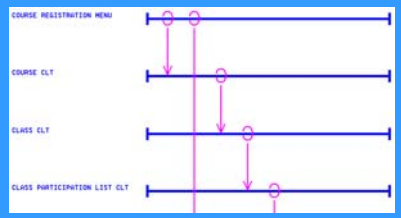
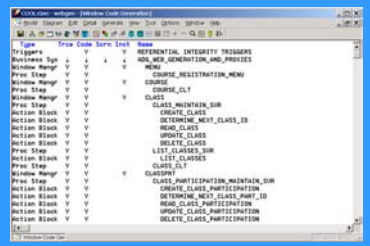
Design and Construction

Implementation

Advantage Gen provides:

- A set of layered diagrams used to specify business requirements
- Techniques for populating the diagrams
- An integrated development environment

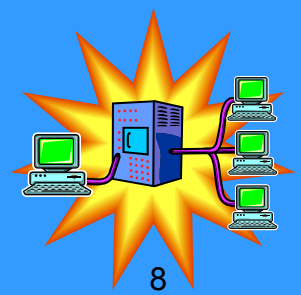
Layered Diagrams

Planning and Analysis	Design	Construction	Implementation
			
Data Model	User Interface	Load Module Definition	Implementation Toolkit
			
Activity Hierarchy	Application Navigation	Source Generation	Executables
	Logic Diagrams	Target Environment	

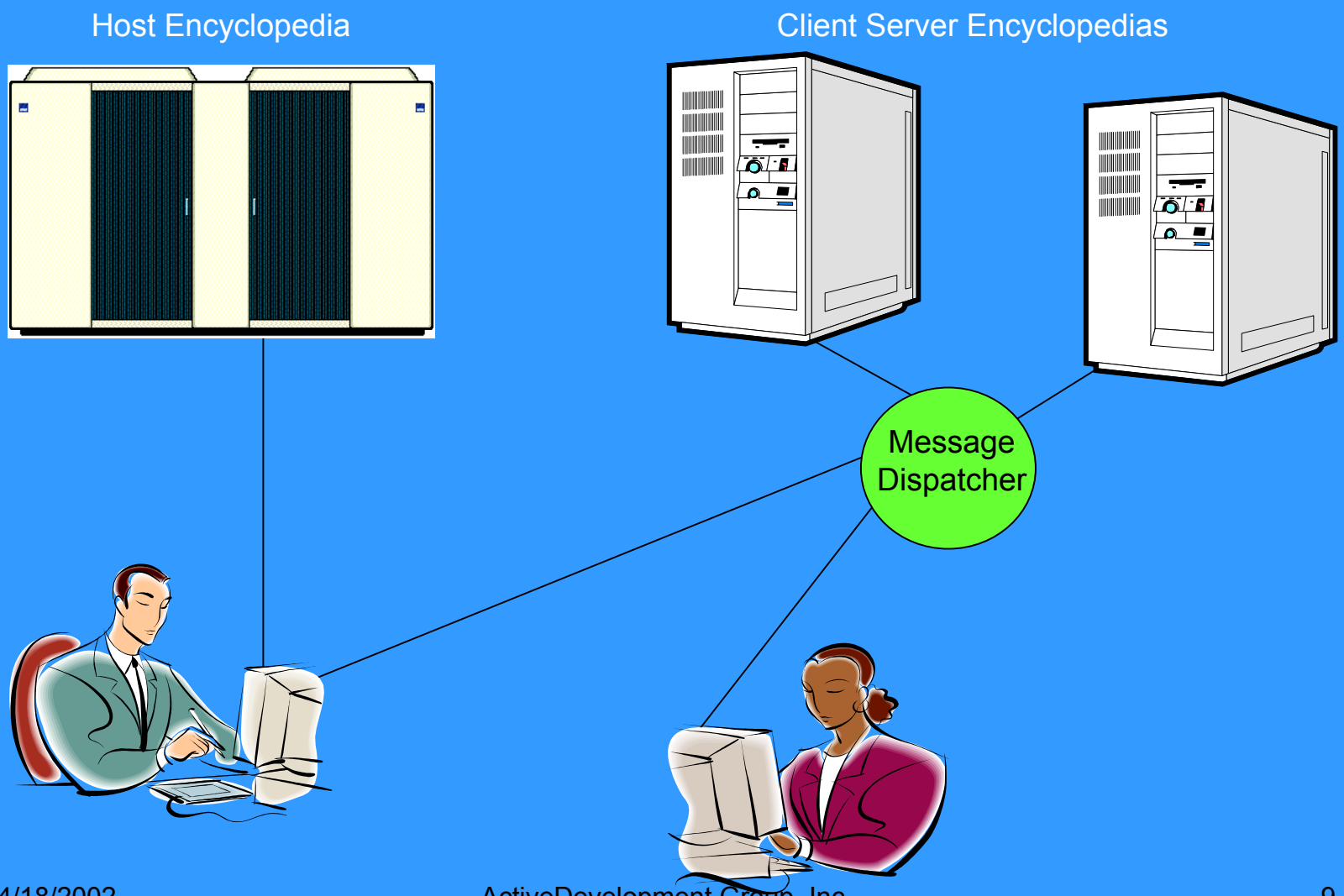
Application Development Lifecycle

What is an Encyclopedia?

- Advantage Gen provides two types of encyclopedias
  - Host Encyclopedia for the IBM MVS mainframe
    - Access through TSO panels
    - Uses DB2 for repository database
  - Client/Server Encyclopedia for open systems
    - Access through GUI clients in a distributed processing environment
    - Uses Oracle or SQL Server for repository database

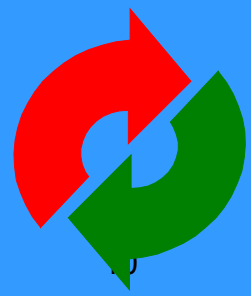


Encyclopedias in Development

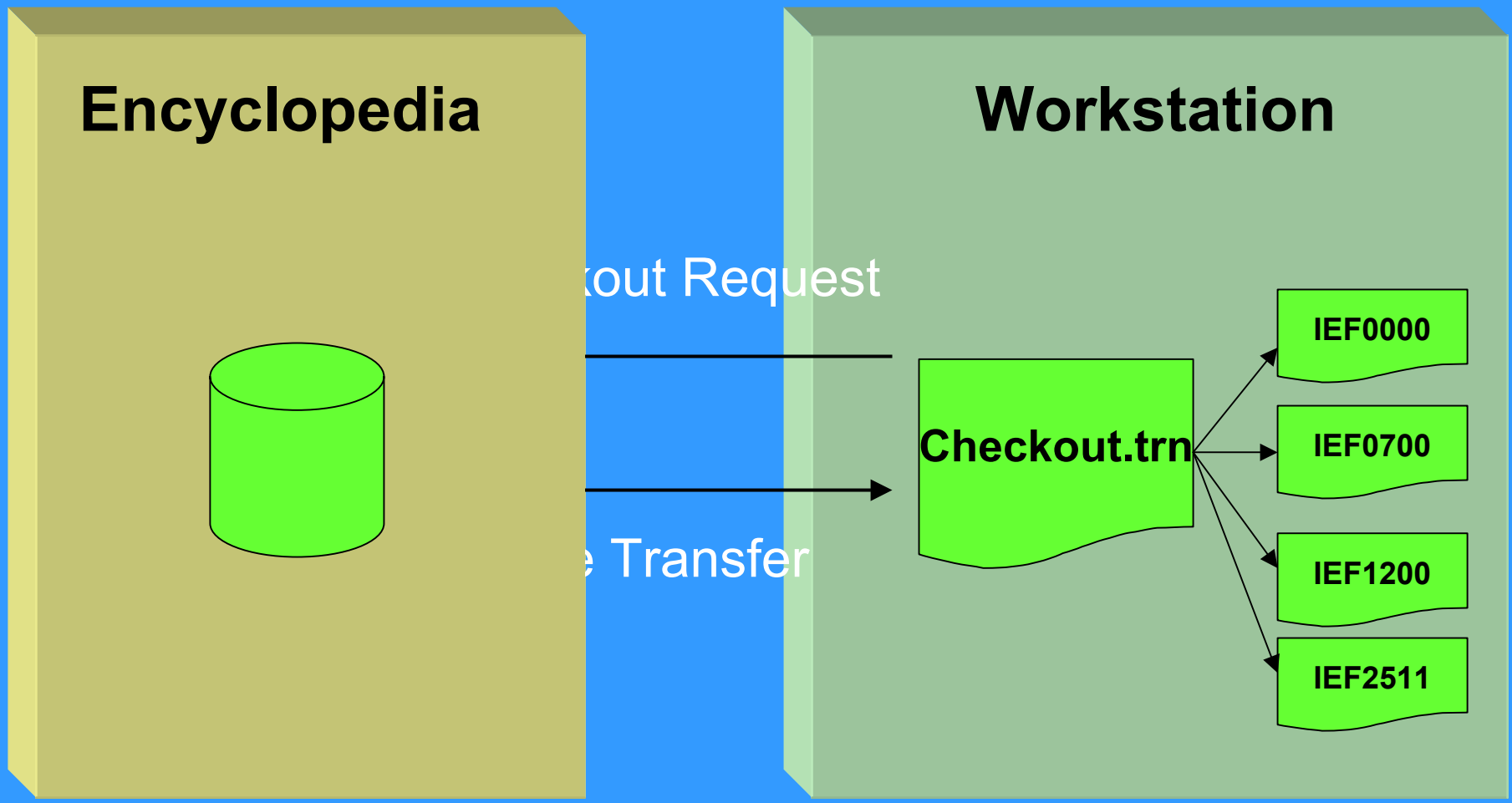


Model Checkin and Checkout

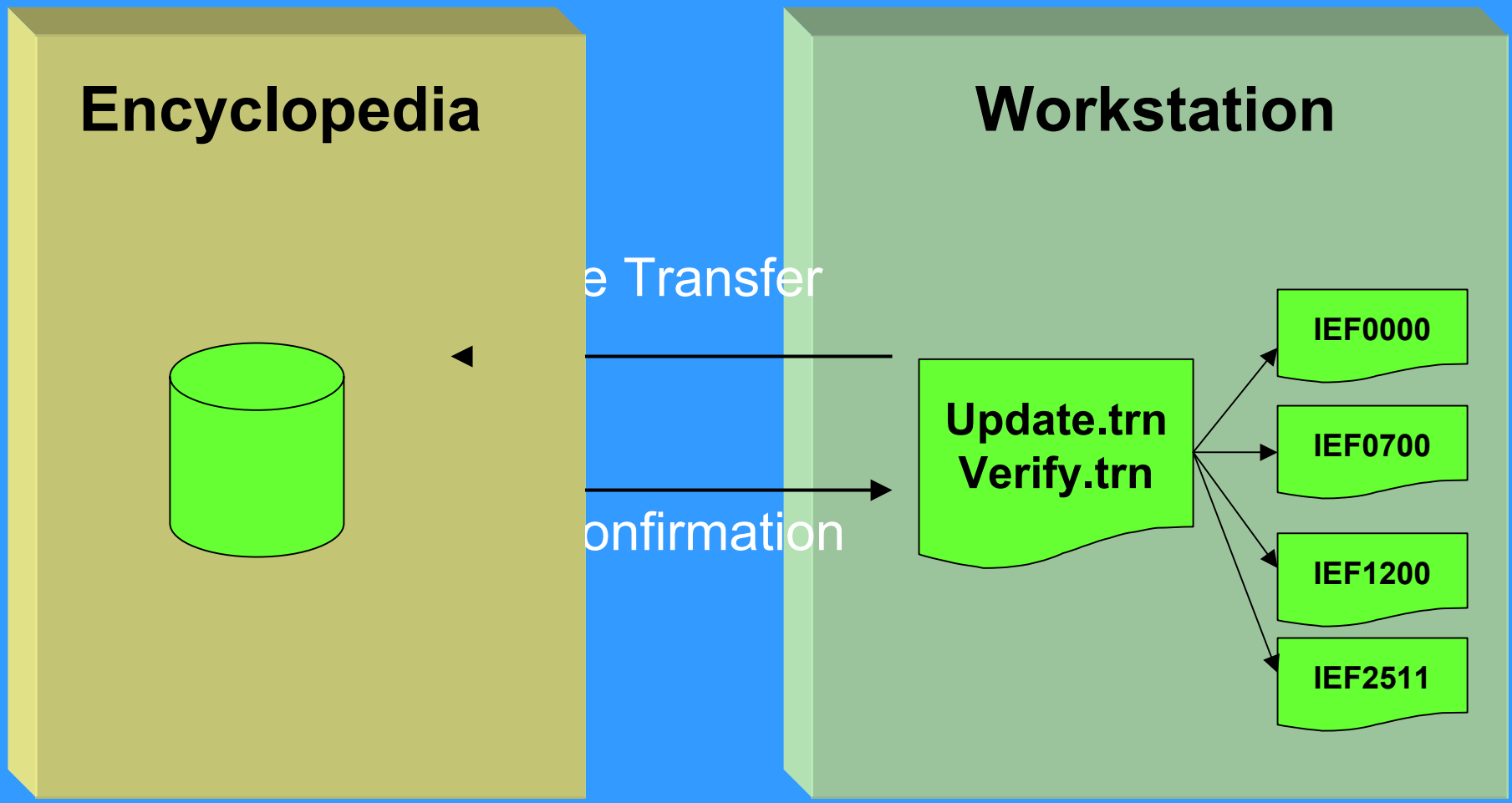
- Checking Out
  - Transfer of model (or subset) from an encyclopedia to a development workstation
  - Security and current checked out status verified
- Checking In
  - Transfer of model (or subset) from a development workstation to an encyclopedia
  - Security and current checked out status verified



Checkout Process

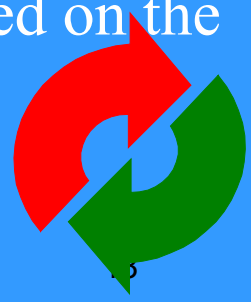


Checkin Process



Checkin Details

- With Checkin
  - The encyclopedia is updated with any changes made to the model on the development workstation
  - The active copy of the model (or subset) is returned to the encyclopedia
- Without Checkin
  - The encyclopedia is updated with any changes made to the model on the development workstation
  - The active copy of the model (or subset) is maintained on the workstation

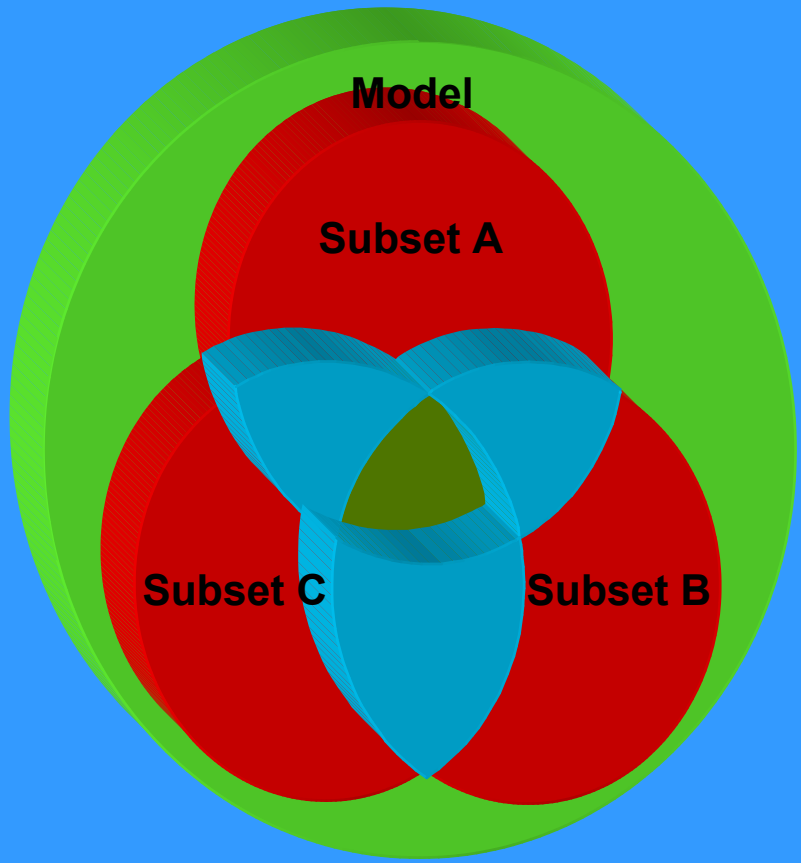


Subsetting

- Subsetting allows multiple developers to work on the same model simultaneously
- Subsetting allows objects to be shared among members of the development team
- Security is applied to a subset in the same manner as to models
- Subsets are checked in and out in the same manner as models

Subsetting

- A subset is a named set of model objects



Version Control

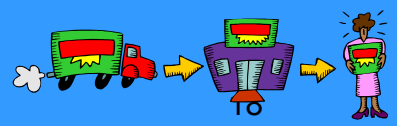
- A set of tools and techniques that allow the maintenance of multiple versions of a model or object and promote sharing of objects between models
  - Move changes from a development model to a testing model
  - Move common objects from a shared model to a specific project model
  - Typically a shared responsibility of the model creator and the encyclopedia administrator

Adoption

- Adoption is the process of establishing common ancestry between objects in different models
- Ancestry is the basis on which objects are shared across models

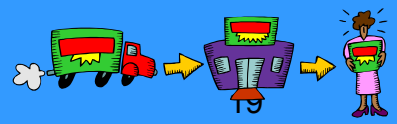
Adoption

- Select the Adopter model as the source
  - The model with the correct original object IDs
  - Assure that you have at least read only rights to the model
- Select the Adoptee model as the destination
  - The model with the incorrect original object IDs
  - Assure that the adoptee model is checked in and you have migrate to rights on the model
- Select the objects to adopt
  - Choose either all objects or select them from the list
- Proceed with the adoption
- Examine the adoption results



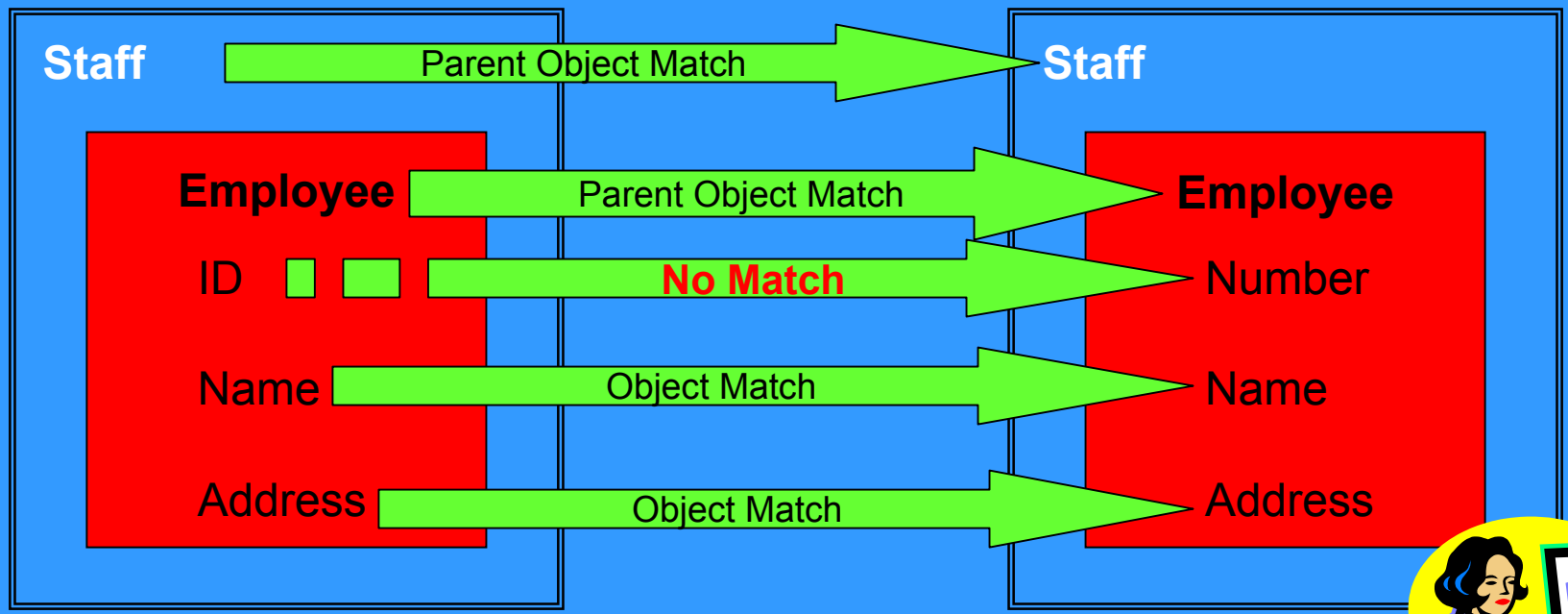
Trial Adoption

- The trial adoption proceeds exactly like the adoption and generates an identical adoption report
  - This is a good technique to verify that the adoption will be successful
- At the end of the process the trial adoption rolls back any changes to the encyclopedia database



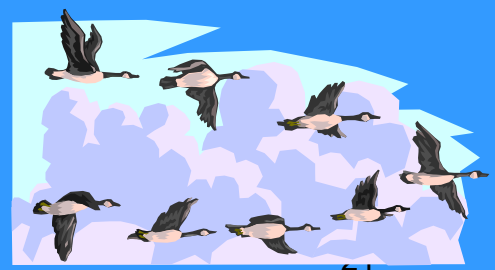
Adoption Rules

- Objects are adopted based on object name and type, as well as parent original object id



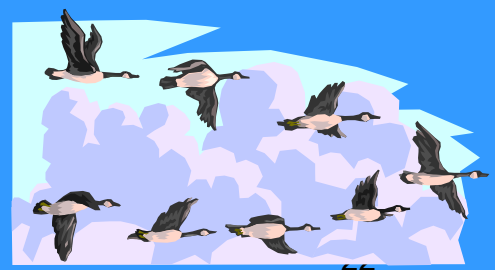
Migration

- The process of copying new or changed objects from one model to another
  - It is critical that common ancestry be established to prevent new copies of the changed objects from being created



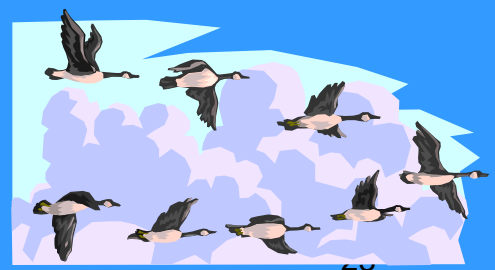
Migration Process

- Select the source model
  - The model with the current objects
  - Assure that you have at least read only rights to the model and that the desired objects are current in the encyclopedia
- Select the destination model
  - Assure that you have at migrate to rights to the model and that it is checked in to the encyclopedia
- Select the objects to migrate
- Proceed with the migration
- Examine the migration results



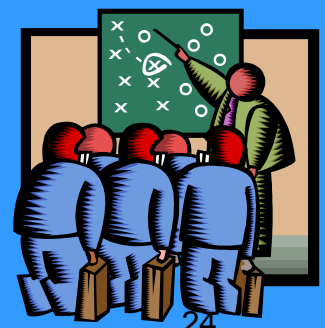
Trial Migration

- The trial migration proceeds exactly like the migration and generates an identical migration report
  - This is a good technique to verify that the migration will be successful
- At the end of the process the trial migration rolls back any changes to the encyclopedia database



Model Management

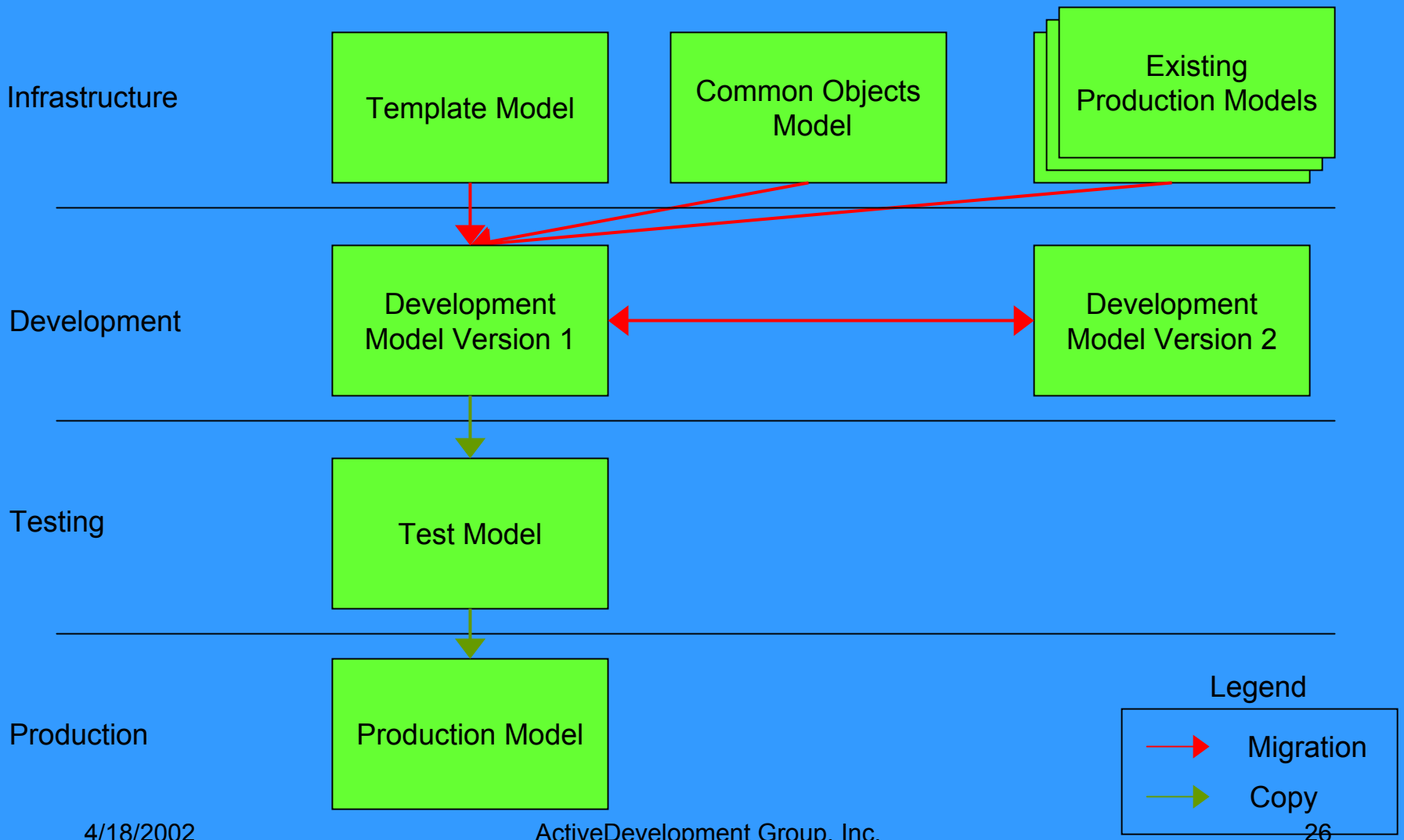
- Model Management has two primary purposes
  - To leverage corporate Advantage Gen knowledge by starting new development with infrastructure and pre-existing elements
  - To maintain version control of developed applications



Model Management

- It is unusual to start a new project with an empty model
  - Development templates speed the development process by providing a starting point for application development
  - Re-used components prevent the re-development of pre-existing logic

Model Management Strategy



Reporting

- The Encyclopedia provides a variety of reports to assist in development and development coordination
- The Host Encyclopedia includes many different reports
- The Public Interface provides a mechanism by which users can query the Encyclopedia

User Administration

- Developers
  - Primary interactions with the encyclopedia include checkin, checkout, subset creation
- Model Owners
  - Primary interactions are similar to developers with the added responsibility of granting model level access to other users
- Encyclopedia Administrators
  - Grants access to the encyclopedias, authorizes model owners, removes inactive models, etc.

User Access & Privileges

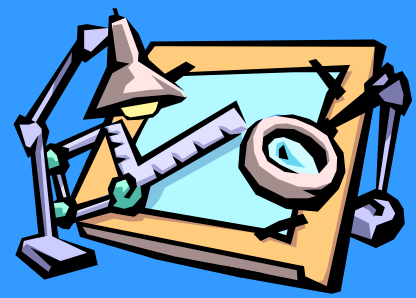
- Access privileges are assigned either to an individual or group of users
  - Users are granted access to the encyclopedia
  - Users are granted access to models
- Access is either explicitly granted or inherited
  - i.e. The encyclopedia administrator inherits all of the rights of the model and subset owners

Encyclopedia Groups

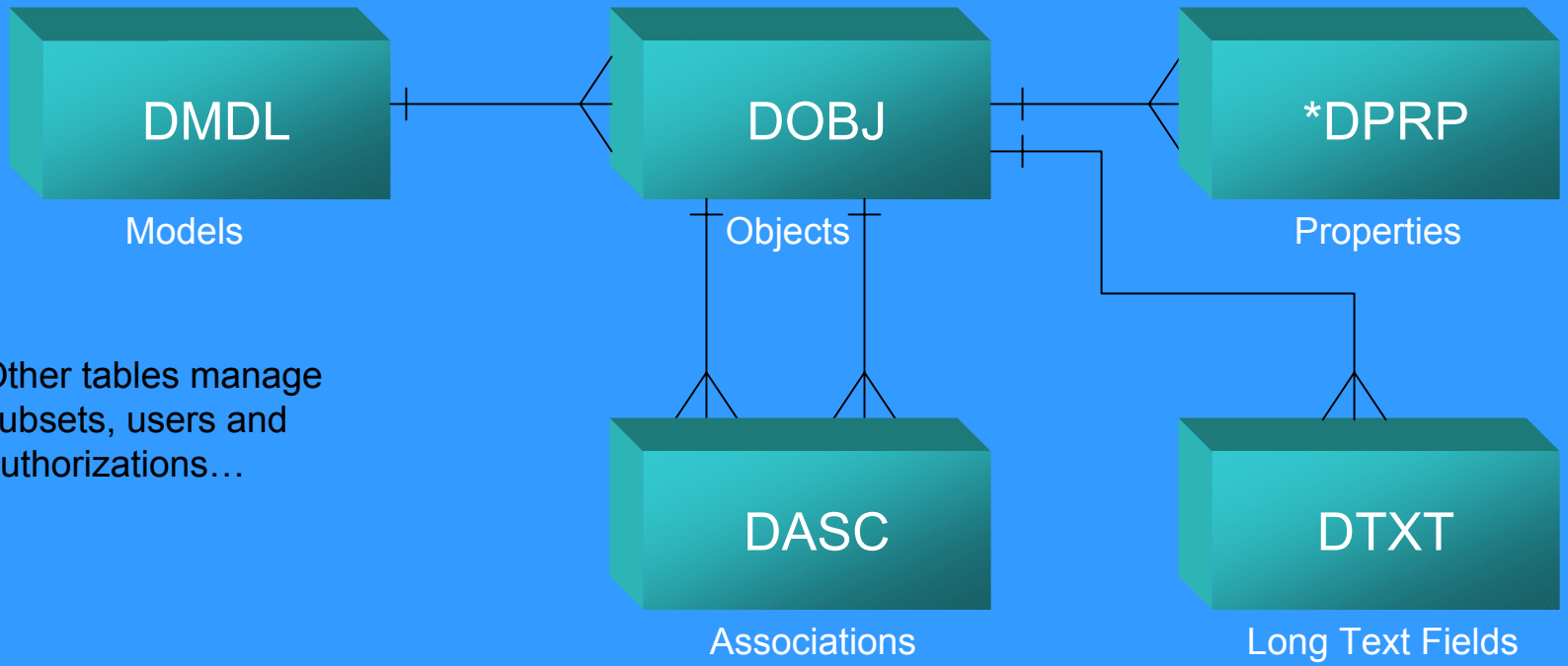
- A group contains encyclopedia users
  - A user can belong to more than one group
  - Any encyclopedia user can create a group
  - An encyclopedia administrator or model owner can assign access privileges to a group

Technical Design

- A set of processes that interact with a relational database
- The database is a meta-model composed of three parts
  - Data tables (store objects, object properties, subset definitions, checkout status and associations)
  - Schema tables (store constants and types)
  - Public interface



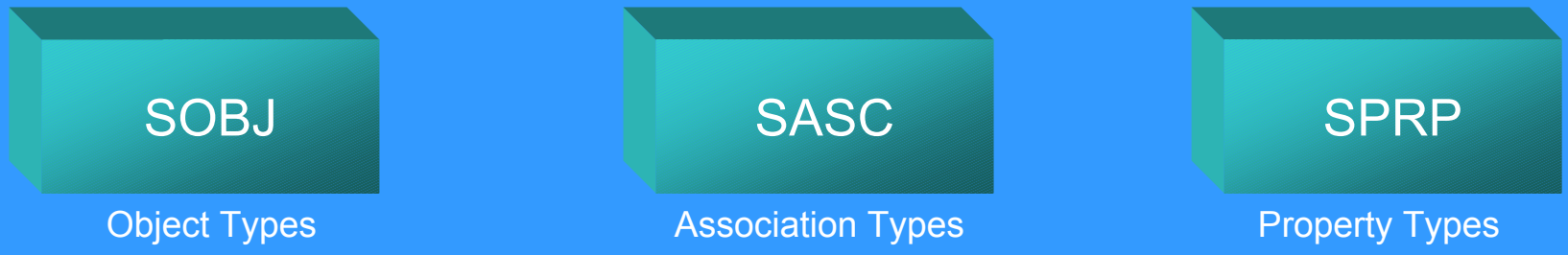
Primary Data Tables



Other tables manage subsets, users and authorizations…

* Host

Schema Tables



The schema manages what is valid for objects, their associations and properties, for an Advantage Gen release.

Public Interface & APIs

- Consists of “views” defined at the DBMS level
- Provides a friendlier interface to a model’s contents
- Can be accessed by user-created queries and application interfaces
- API consists of read-only and update application interfaces

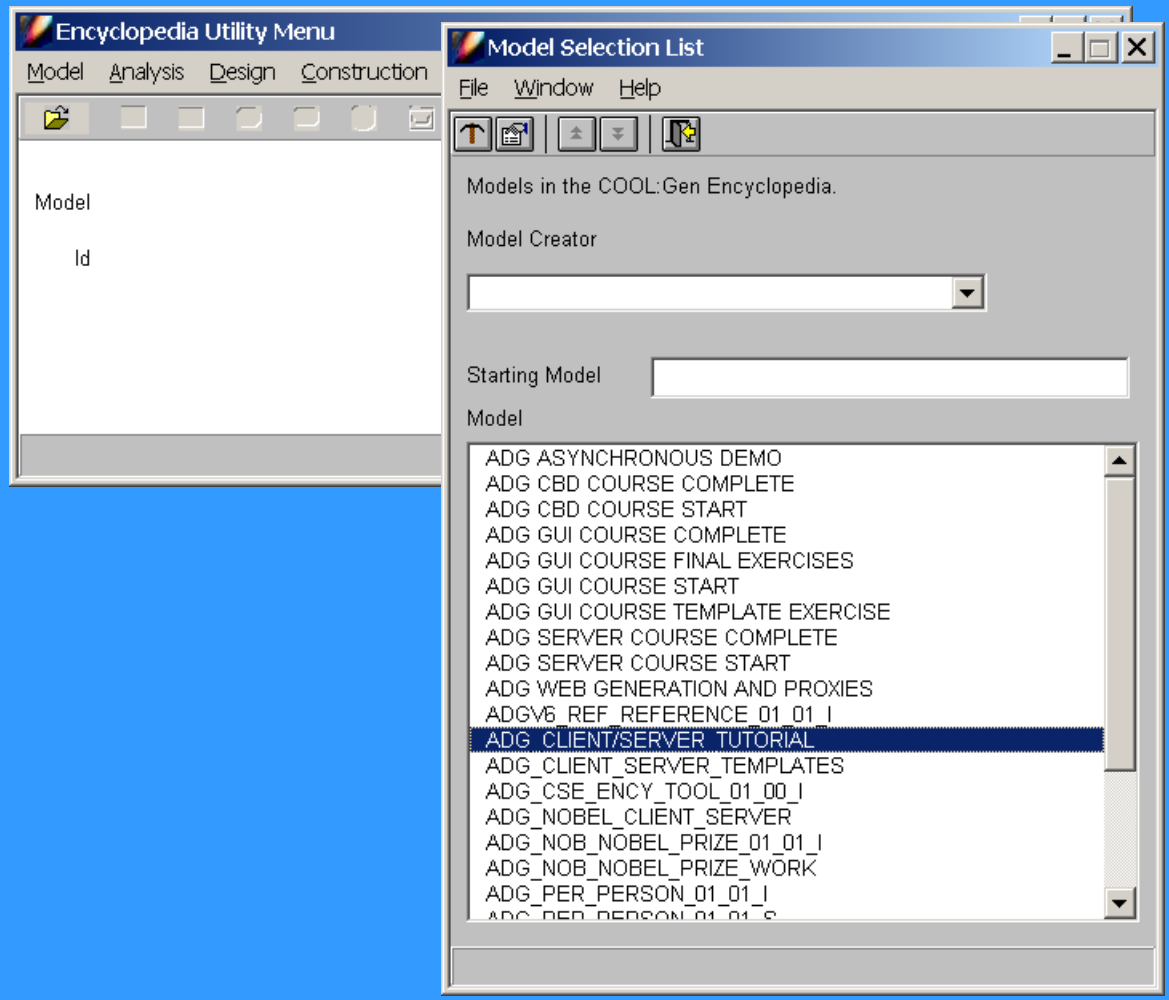
Common Functions Tips

- Coordinate version control activities across teams
- Limit migrations to small sets of objects
- Perform migrations at a time when conflicts with development activities will be minimized
- Break up the development work so developers do not encounter conflicts with team members
- Use small, task-oriented subsets in concurrent development

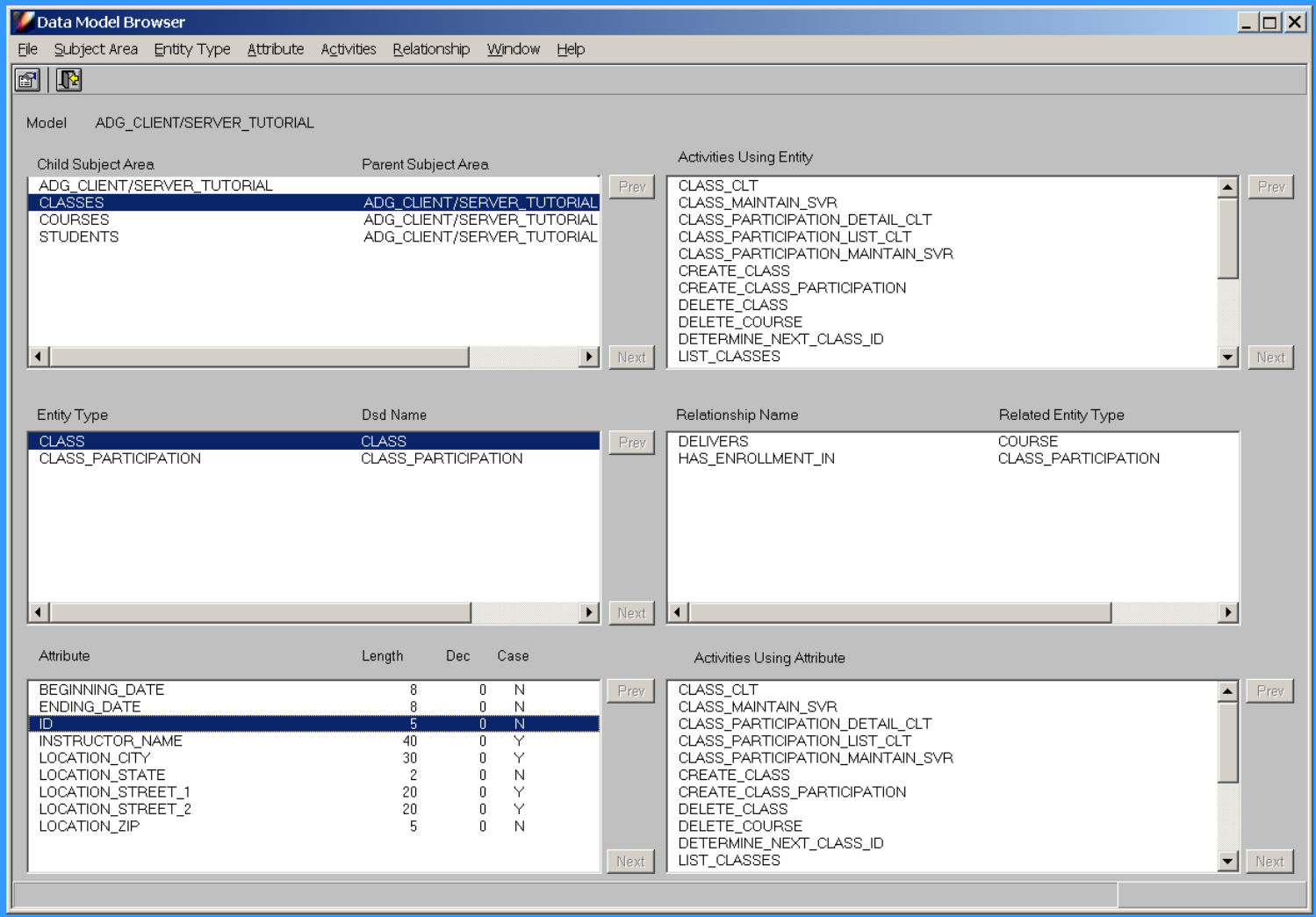
Common Functions Tips

- Schedule regular encyclopedia database reorganization
- Monitor encyclopedia for accumulation of unneeded models
- Develop a software reuse strategy

Queries Demonstration



Queries Demonstration



Questions and Comments