Planning: Enterprise Geodatabase Solutions

John Alsup

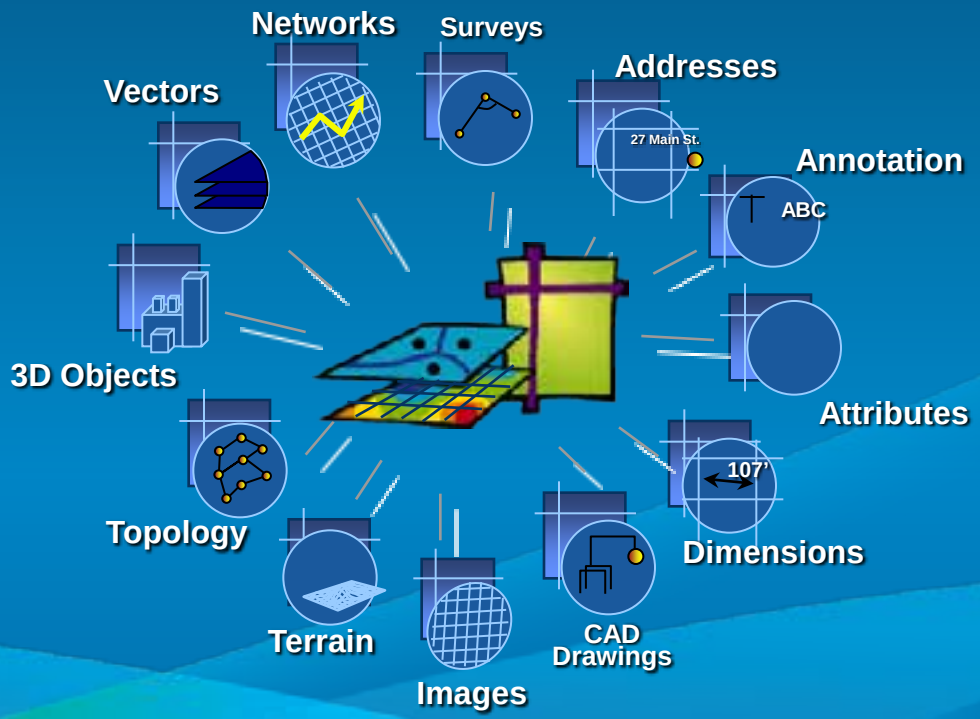
Matt Bottenberg

Agenda

- Overview
- Database Design
- Data Maintenance
- Infrastructure Design and Data Distribution
- Security
- Database Maintenance
- Performance

What is a Geodatabase?

- A database or file structure used to store, query and manipulate spatial data.
- Data and functionality
- Three types:
  - File geodatabase
  - Personal geodatabase
  - ArcSDE geodatabase
    - DB2
    - Informix
    - Oracle
    - PostgreSQL
    - SQL Server



Enterprise GIS

- GIS technology regarded by users and IT as key to business operations
  - May be considered mission critical
- Mainstream IT – deployed and managed like any other IT system
  - Architecture, Interfaces, Development tools, Deployment strategies, Standards
- Integrated with other enterprise systems
- Requires a higher level of planning, integration, testing and support

What is an Enterprise Geodatabase?

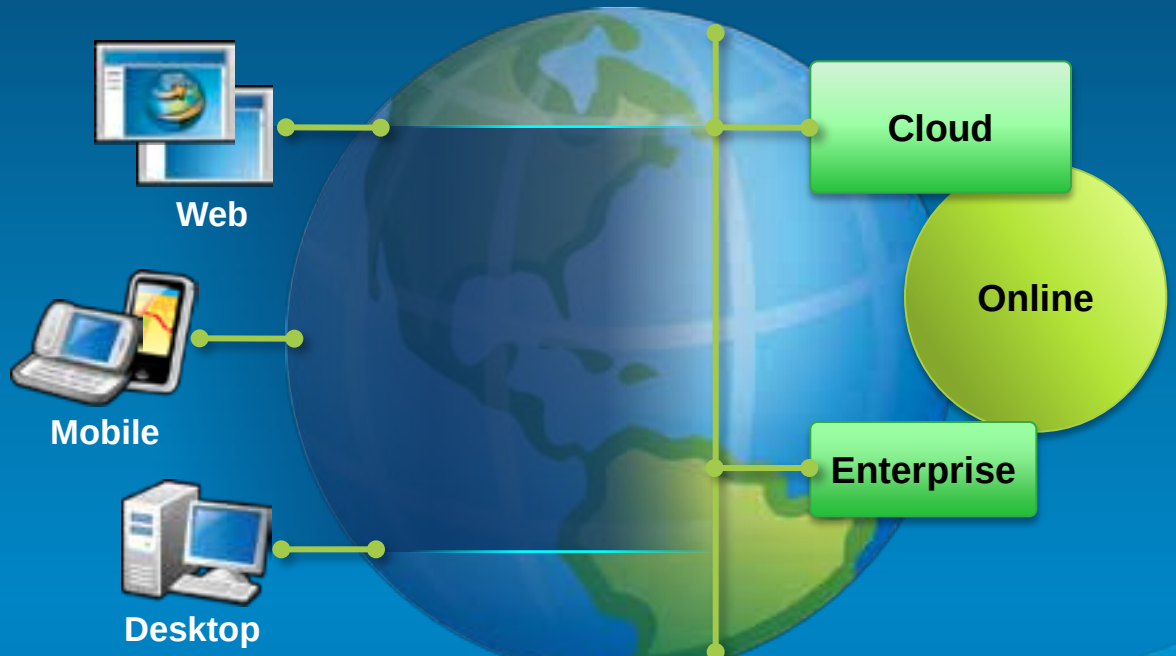
- **Data**
  - Serves data promptly and efficiently
  - Supports multiple users and departments concurrently
  - Provides seamless data
  - Centralized data management
  - Data integrity

- **Functionality**
  - SQL support
  - Collaborative editing, and long transactions
  - Quality control and quality assurance
  - Infrastructure for distributing and replicating data
  - Integrates spatial and business data with other systems
  - Leverages existing GIS and IT skills and resources

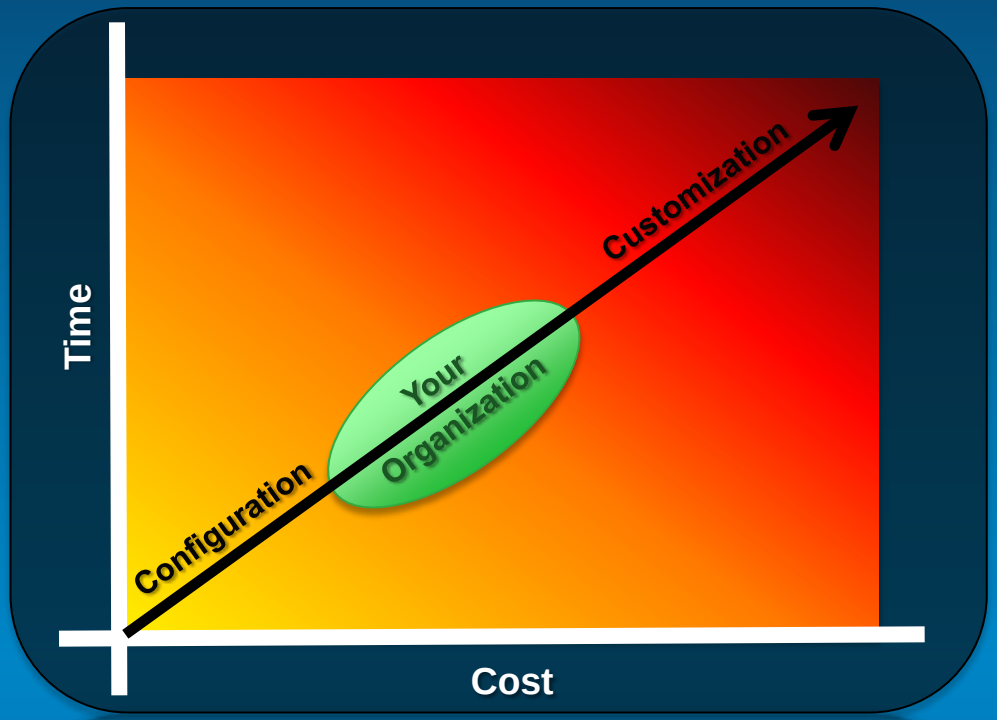
Enterprise GIS Use Patterns

ArcGIS

Is Esri's Complete System for Enterprise Geographic Information



Configuration is the Key to ArcGIS



Why Plan an Enterprise Geodatabase?

- **Some key reasons:**
  - **Foundation for enterprise-wide use of GIS.**
  - **Geodatabase projects are complex**
  - **Enterprise geodatabases and GIS application design requires diligent alignment**
  - **Large geodatabase projects span organizational groups and disciplines**
  - **Impacts almost every part of an enterprise GIS solution**

Spatial data is a key component of an enterprise GIS architecture . . .

. . . delivery of spatial data must be fast, and this requires planning.

Geodatabase Project Scales

- **Larger Multi-phased Approach**
  - Elaborate, large databases
  - Custom applications
  - Large user base
  - Potentially outsourced, dedicated project management

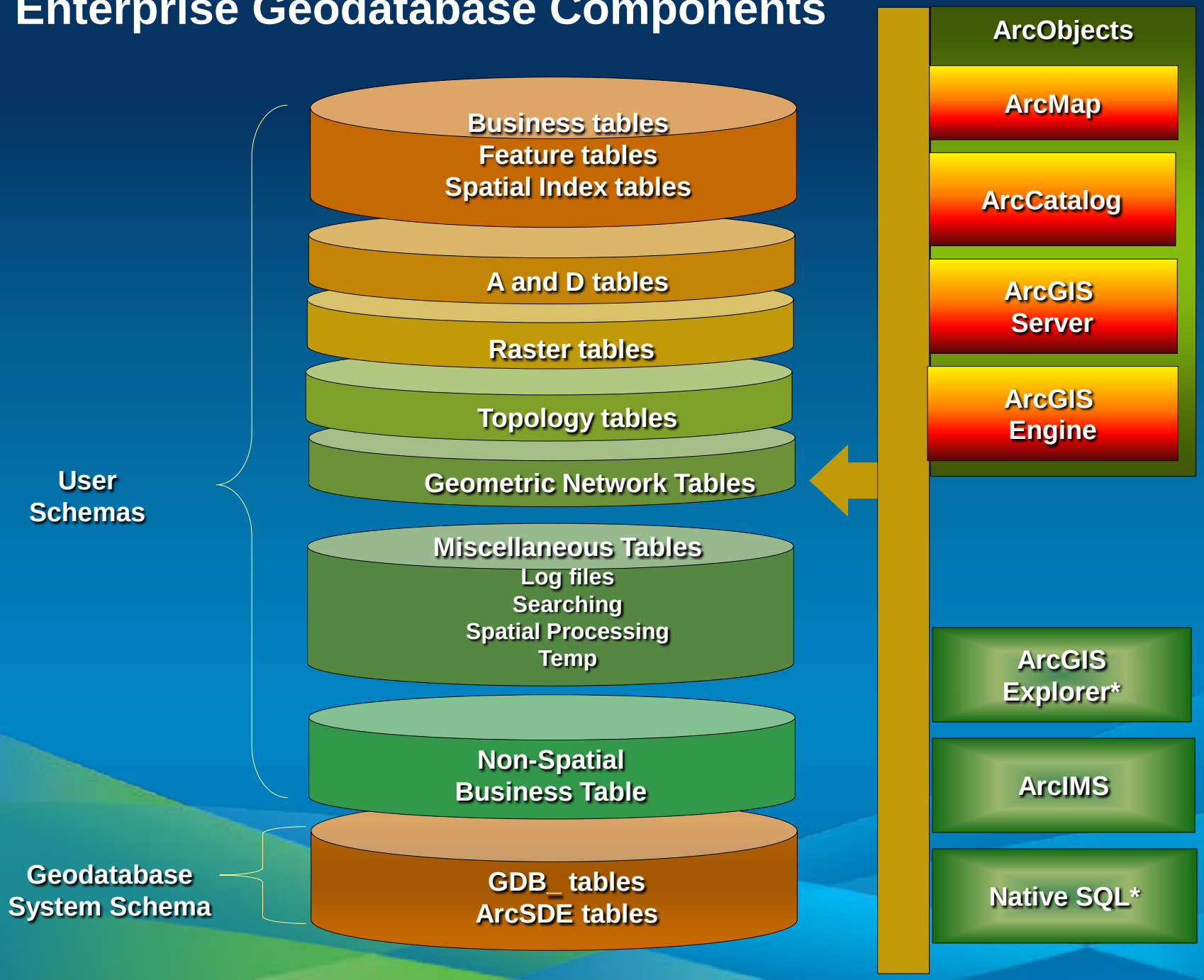
- **Lighter Workgroup Approach**
  - Evolve the geodatabase, gradually upgrade data and applications
  - COTS application functionality where possible
  - Built in-house, part-time project management

All enterprise geodatabase projects require planning …

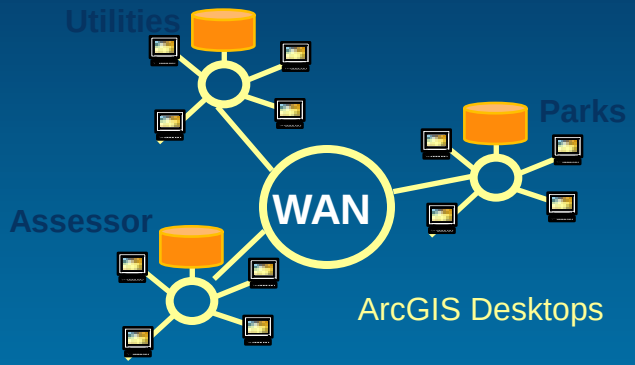
Agenda

- Overview
- Database Design
- Infrastructure Design and Data Distribution
- Data Maintenance
- Security
- Database Maintenance
- Performance

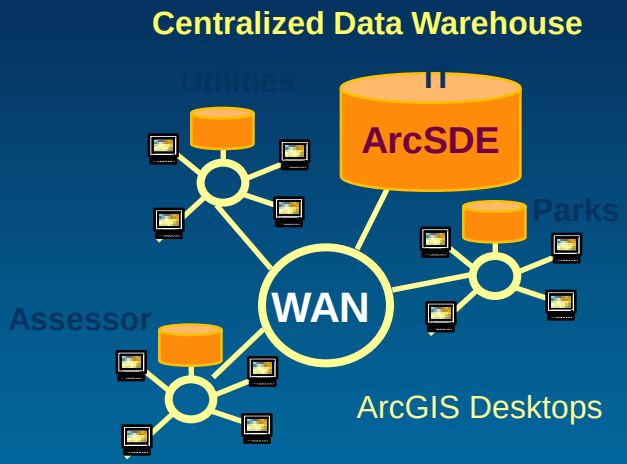
Enterprise Geodatabase Components



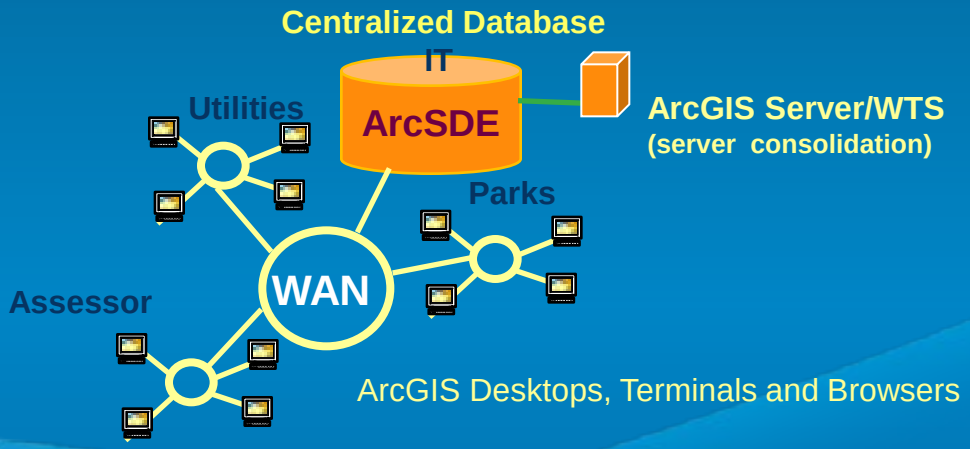
Organizational GIS Configurations



Distributed Client/Server
Departmental GIS



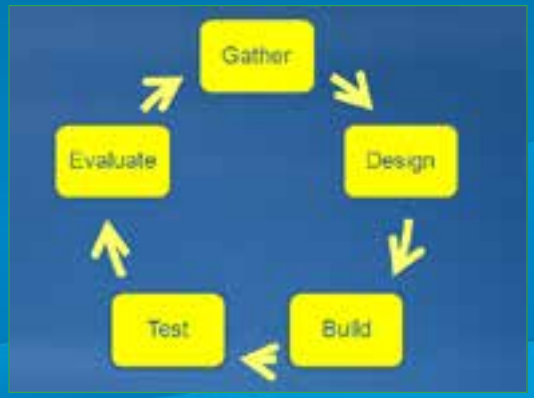
Data Warehouse
Departmental GIS Operations
Centralized Data Sharing



Centralized Database
Enterprise GIS Operations
Centralized Data Administration

Some Considerations on Design

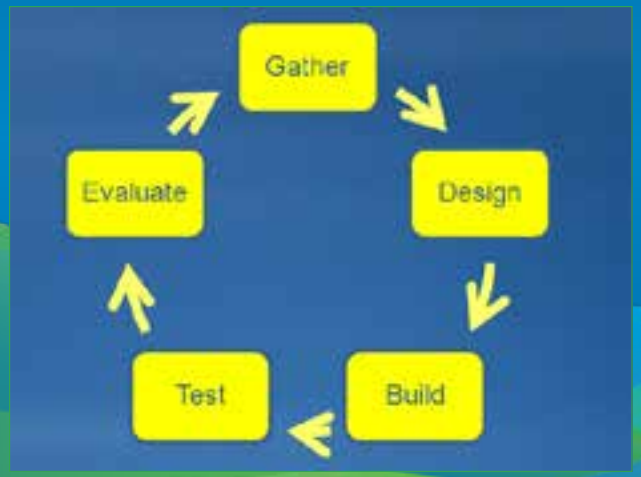
- Core enterprise GIS design task
- Foundation and blueprint for the capabilities of the GIS
- Development of the “data model”
- Data models sets the limits for application functionality
- Data maintenance is expensive
- Performance



Geodatabase design impacts almost every area of the enterprise GIS...

Challenges and Risks

- Application development has a critical dependency
- Normalization in the data model
- Updating the model “downstream” is expensive
- Thorough review of model among teams
- Optimizing for publication and maintenance

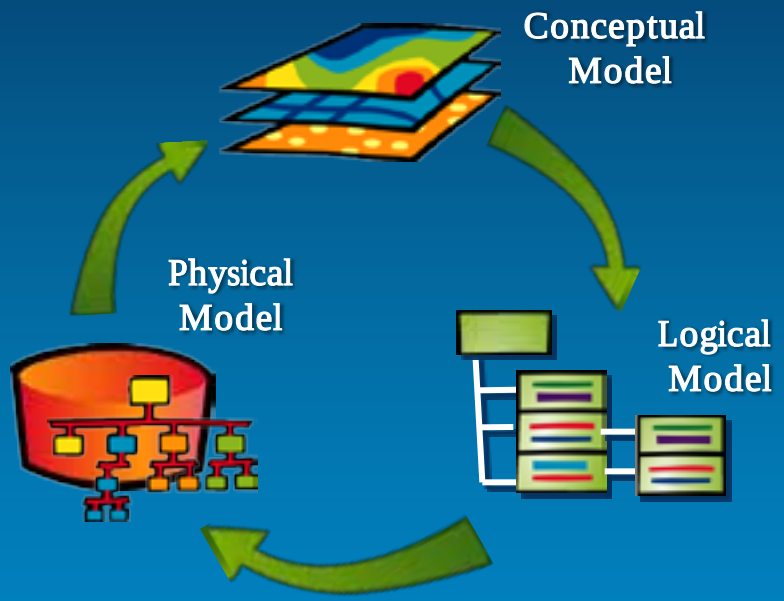


Geodatabase Design

- **Elements of good geodatabase design**
  - **Data model reflects requirements**
  - **Scalable**
  - **Avoids redundant storage of data items**
  - **Efficient access to data**
  - **Maintains data integrity over time**
  - **Clearly documented**
  - **Provides for analysis and behavior**

Data Modeling Methodology

Three Stages



Conceptual Design Tasks:

- Identify business needs
- Identify thematic layers
- Identify required applications
- Leverage data model template
- Document

Logical Design Tasks:

- Define tabular database structure
- Define relationships
- Determine spatial properties
- Document

Physical Design Tasks:

- Create and implement model design
- Generate physical schema in the DBMS
- Testing and validation
- Document

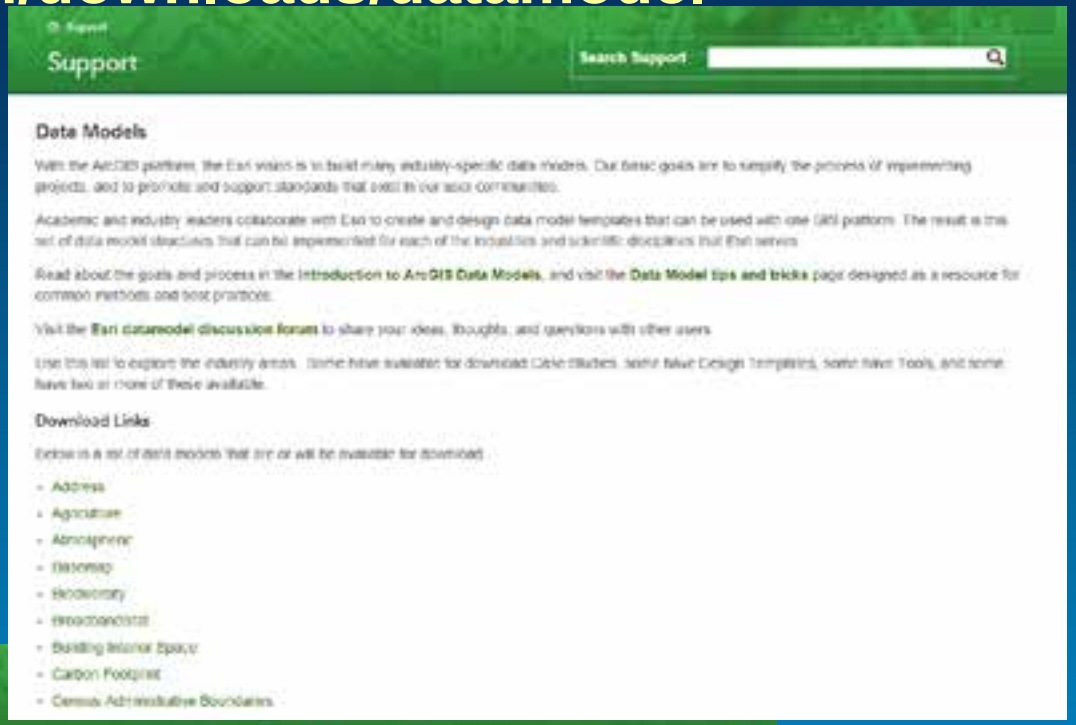
Conceptual Model

- **Identify and Document:**
  - Business needs - requirements
  - Thematic layers
  - Required applications and system interfaces

- **Leverage existing model templates**
  - Esri Resource Center pre-designed schema of data objects
  - Best practices

ArcGIS Data Models Web site:
http://support.esri.com/downloads/datamodel

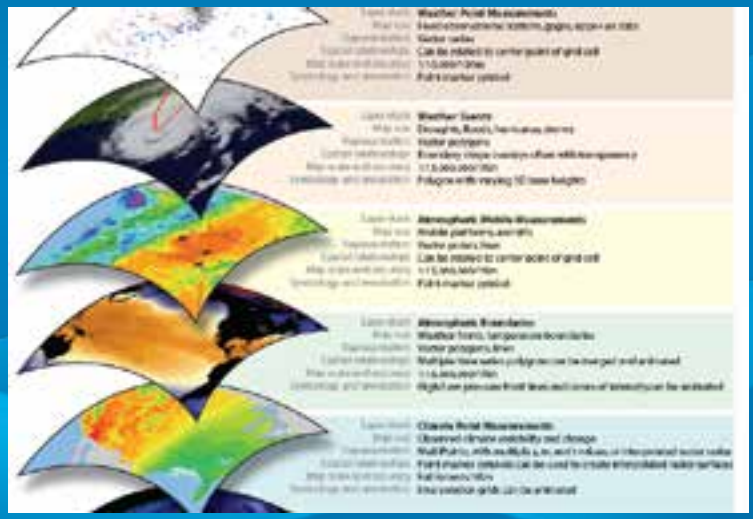
- Over 25 industry-specific data models
- Conceptual and logical diagrams, sample geodatabase schemas
- Case studies
- Tips and Tricks documents
- Developed and maintained by user and industry communities



ArcGIS Resources

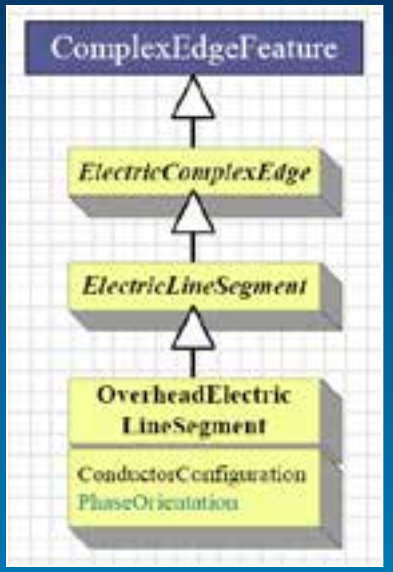
Logical Model Design

- Refine conceptual model based on documented requirements
- Define and clarify all feature classes, tables, attributes and relationship classes
- Use subtypes to control object behavior
- Attribute domains and complex coding
- Define network and topological properties and rules
- Define spatial reference properties
- Map placement considerations



Logical Model Design

- Identification of database rules, categories and data integrity
- Complex data types, network connectivity and topology
- Documentation
  - Diagrams
  - Data dictionary
  - Source data mapping
  - Naming conventions



Overhead Electric Line Segment

Overhead electric line segments carry electric current and are supported by structures including power poles, H-frames, and towers. The abstract class *OverheadElectricLineSegment* inherits the attributes of *ElectricComplexEdge* and *ElectricLineSegment*.

OverheadElectricLineSegment defines the following attributes:

ConductorConfiguration: text — Code indicating configuration of conductors; domain values include Compact, Horizontal, Spacer, etc.

PhaseOrientation: text — Utility-specific order or configuration of phases present on conductor

Subtype: Single Phase Primary Overhead, Two Phase Primary Overhead, Three Phase Primary Overhead, Single Phase Secondary Overhead, Three Phase Secondary Overhead, and Streetlight Conductor

Important Considerations

- **RDBMS Geometry Storage Format**

RDBMS	Geometry Storage
DB2	ST_Geometry, SDEBinary
Informix	ST_Geometry, SDEBinary
SQL Server	Geometry, Geography, , SDEBinary
Oracle	ST_Geometry, SDO, SDEBinary
PostgreSQL	ST_Geometry or Geometry
Netezza	VarChar (Shape)

Important Considerations

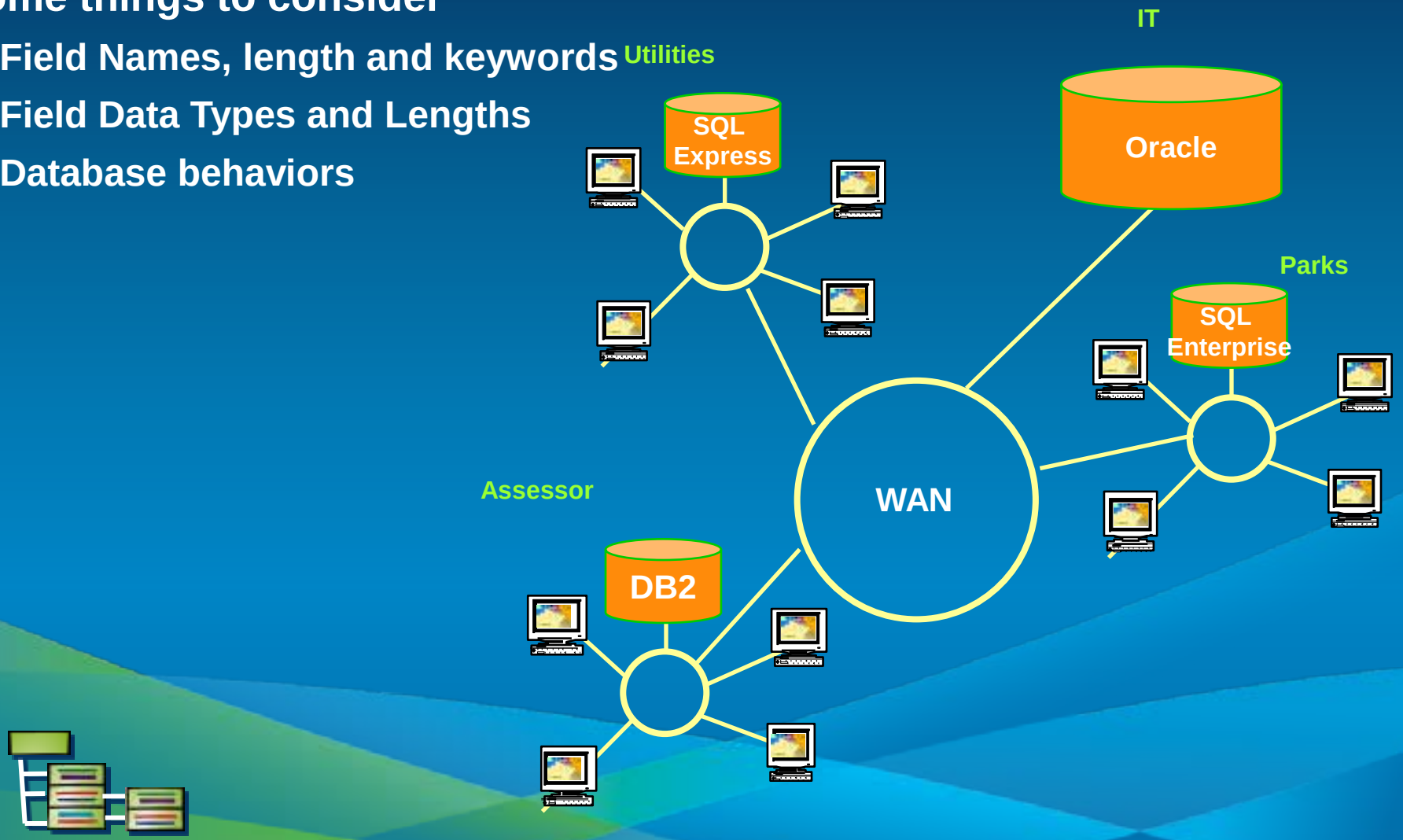
- **External systems and interfaces – key for enterprise GIS**
  - **CRM, WMS, Financials, Reporting**
  - **Number of interfaces depends upon the organization**
  - **Consider data sharing - field data types, naming and length**

External System Interface

- ETL
- Database Level, duplicating data
  - Triggers
  - Update tables
- Database Views
  - Joins data from same or different databases

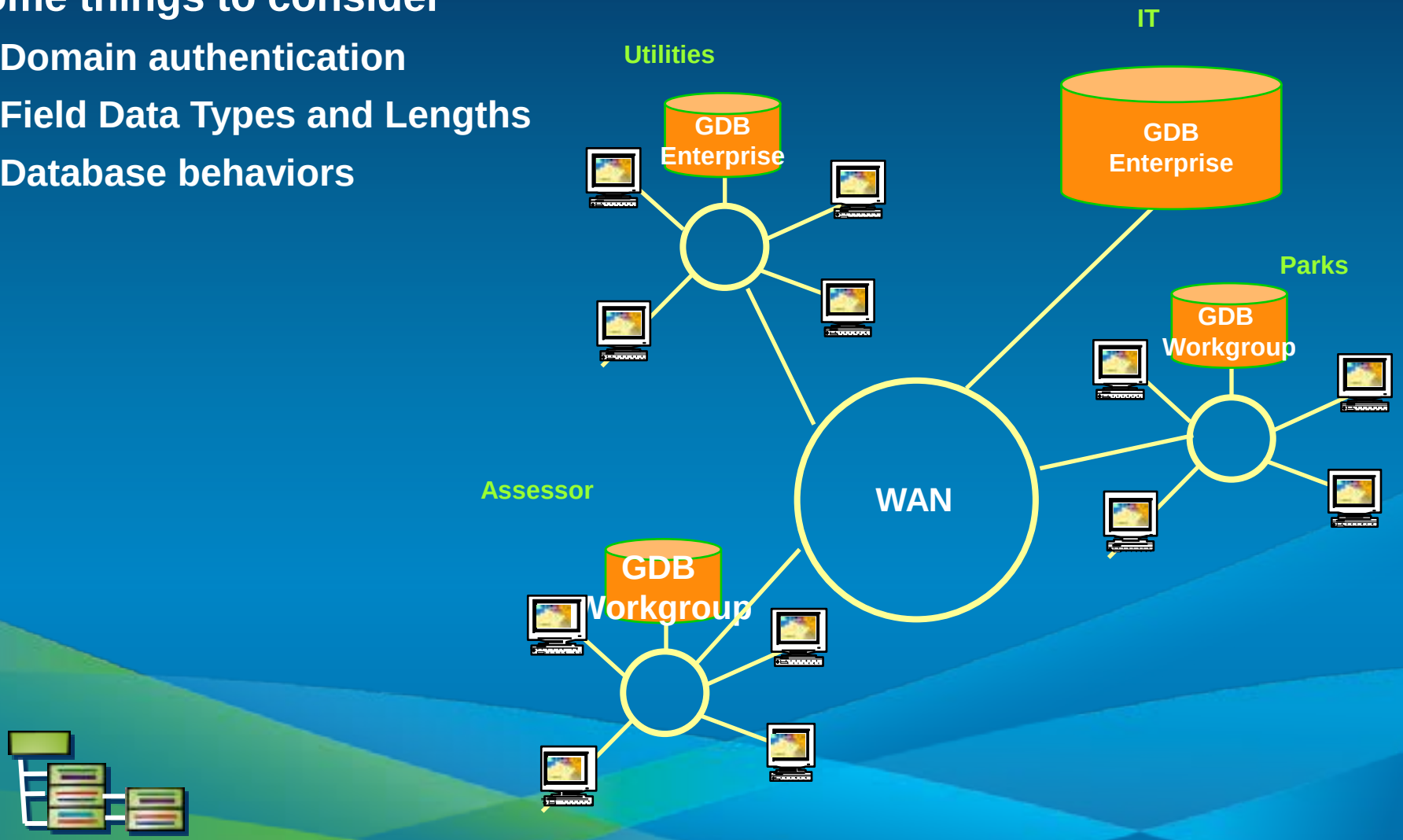
Mixed RDBMS Environments

- Some things to consider
  - Field Names, length and keywords
  - Field Data Types and Lengths
  - Database behaviors



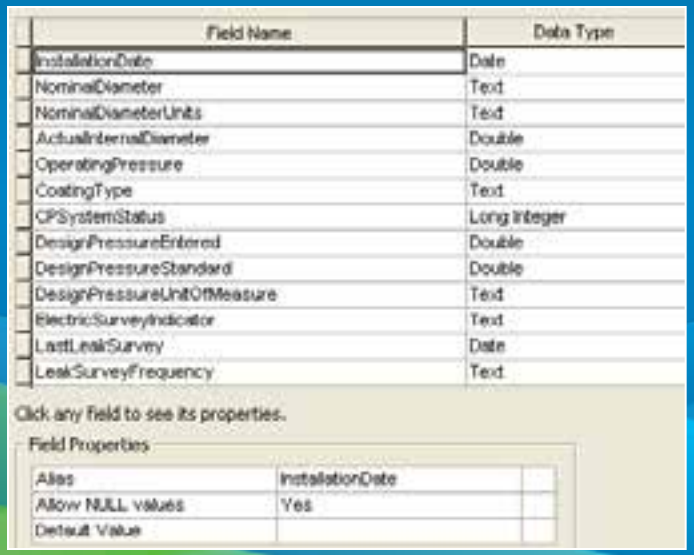
Mixed RDBMS GDB License Levels

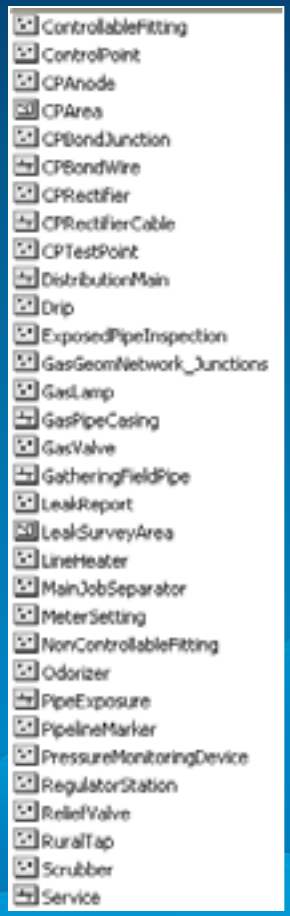
- Some things to consider
  - Domain authentication
  - Field Data Types and Lengths
  - Database behaviors



Physical Model Design

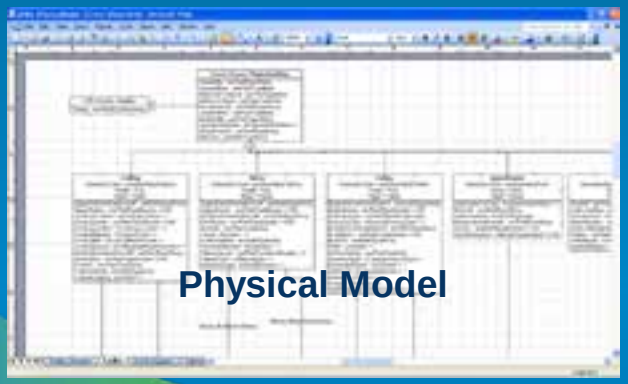
- Implementing the physical geodatabase - prototype, test, review, and refine
- Documenting the design for distribution and efficient updating
- Test, refine and tune data model design for deployment



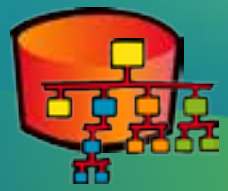


Creating Structure

- Look to existing tools
  - CASE and UML tools – Visio, Rational Rose, etc.
  - Other tools (some free) and samples may work depending on approach
- Inheritance, re-use of objects through abstract and concrete classes



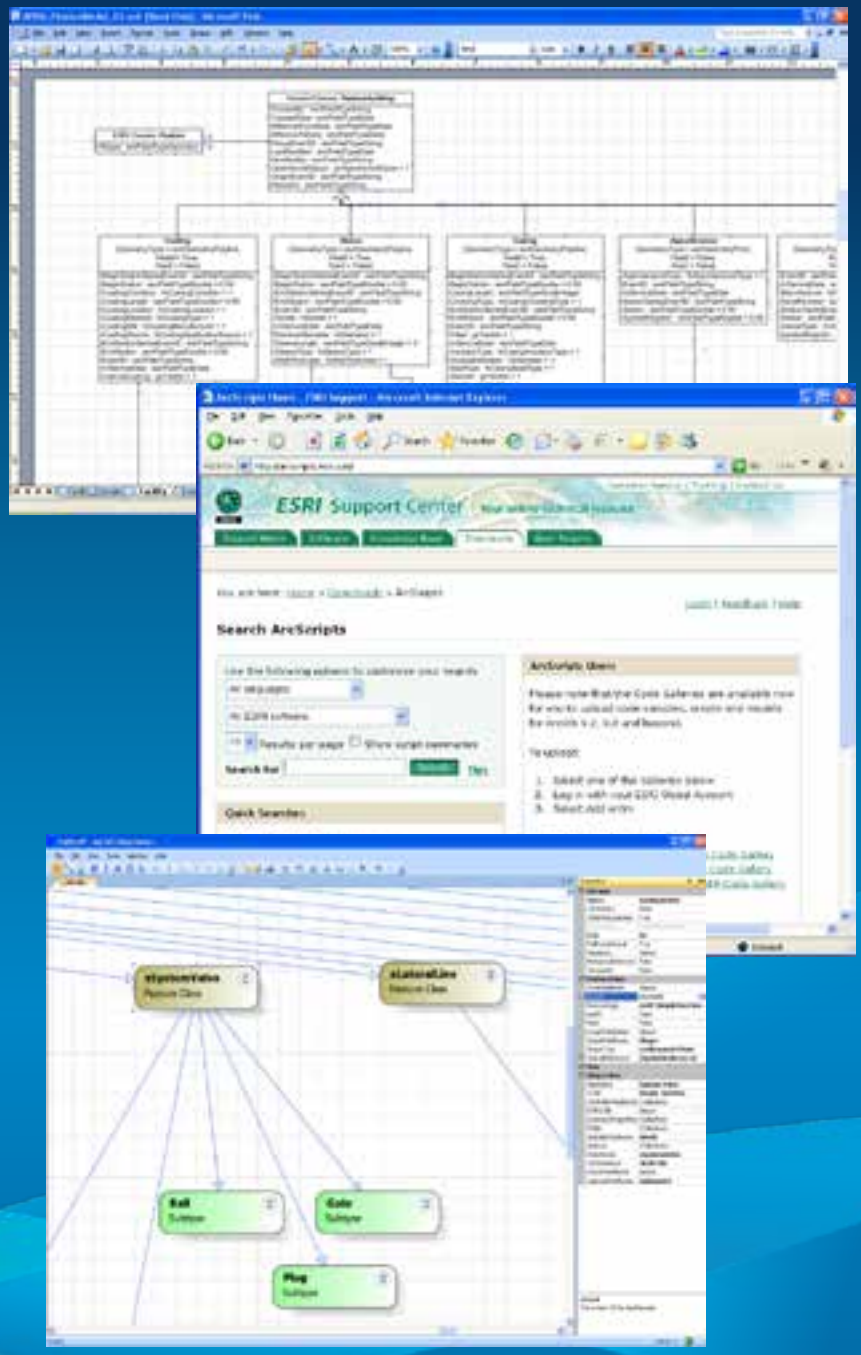
XMI (XML Design)



Data Modeling Tools

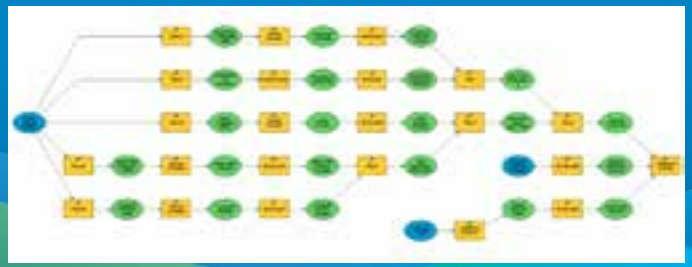
- Visio
- Rational Rose
- Free ESRI Tools on ArcScripts:
  - ArcGIS Diagrammer
  - GDB Xray
  - Geodatabase Diagrammer
  - Geodatabase Designer

Free Tools are not supported



Testing and Refining

- Small pilot data migration with sample data
- Configuration/Application testing – Test workflows
  - Functionality
  - Performance
  - Flexibility and consistency
- Team review and demonstration
  - Show how tasks are performed using GIS
  - Show maps, reports, online demos



Data Planning

- Migration and Conversion
  - Migration deals with moving existing geospatial data between different GIS environments or platforms
  - Conversion refers to development of new data by creating new digital geospatial data
    - Conversion is typically more significant and costly than migration
- Data procurement
  - Landbase
  - Imagery
- Data loading
  - Tools – In-house or outsourced
  - Procedures
- Online Data mashup
  - Landbase
  - Imagery

Agenda

- Overview
- Database Design
- Data Maintenance
- Infrastructure Design and Data Distribution
- Security
- Database Maintenance
- Performance

Overview of Data Maintenance

- Plan and manage the maintenance workflow in the geodatabase

- Key Tasks
  - Analyze and build on business process requirements
  - QA/QC
  - Design your maintenance strategy
  - Plan for versioning
  - Define maintenance workflows

Consider QA / QC

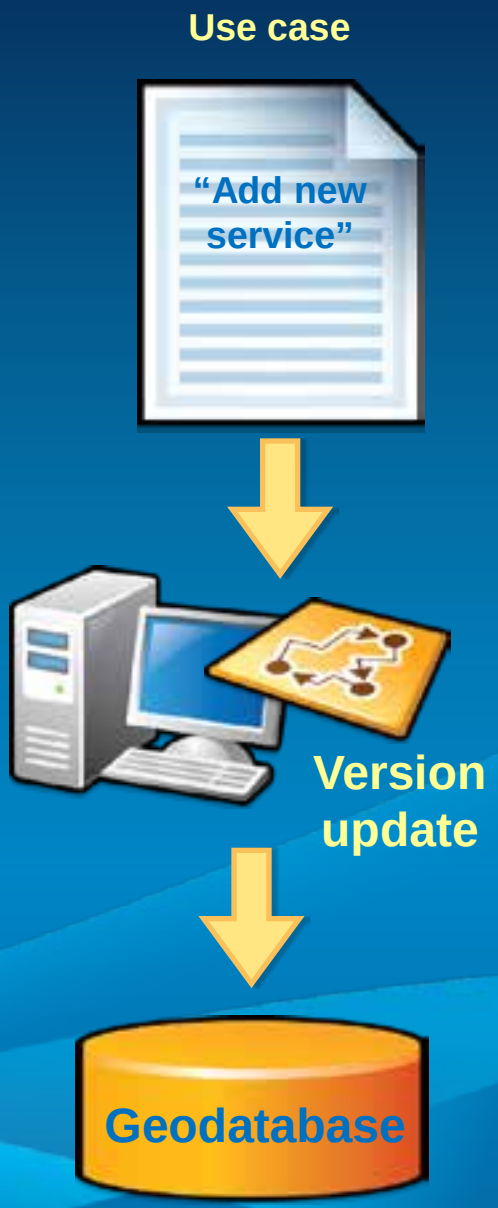
- Ensure data is captured, loaded and maintained accurately
- Quality Assurance
  - Review data to discover errors and perform data cleaning activities to improve quality.
- Quality Control
  - Ensure data products are designed to meet or exceed data requirements.
- QA/QC Plan
  - Versioning
  - Manual and automated procedures
  - Validations

Data Maintenance and Editing Workflows

- **A data maintenance strategy is essential for consistent data quality**
  - QA/QC
  - Versioning strategy
  - Editing workflows
- **Editing Workflows are part of the business model**
  - Business needs
  - Data and schema changes
  - ESRI and non-ESRI client access

User Workflows

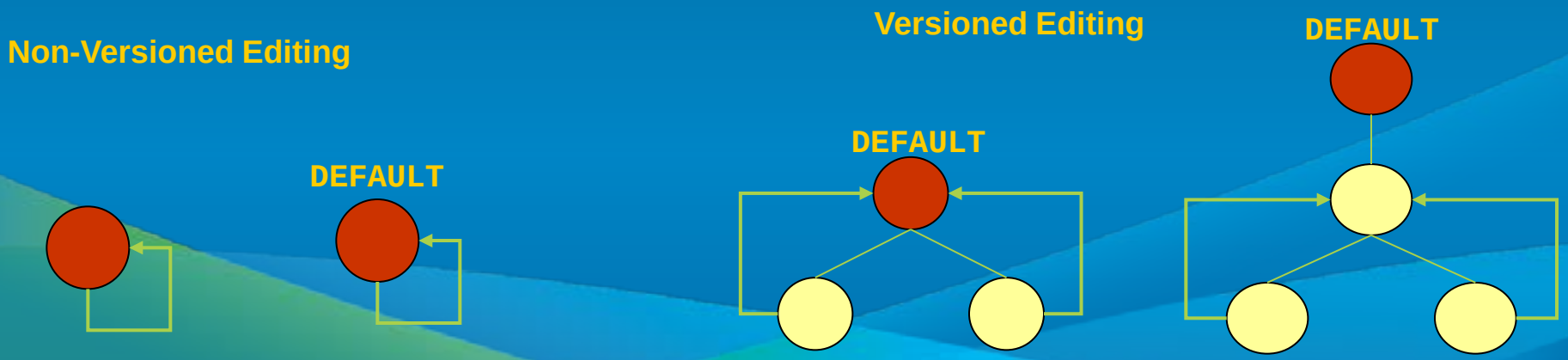
- Document with Use Cases
  - A description of the task you need to perform:
    - “Add new parcel”, “Update new asset”
- Evaluate business needs:
  - What data needs to be edited and in what order
  - Tracking of data changes
  - Conflict detection and resolution
- Security – user roles, etc.
- QA/QC steps – enforced through application or database



Versioning and Multiuser Geodatabase

- **Defining versioning specifications and workflows:**
  - Versioning structure
  - Reconcile, post, compress regimes
  - Edit volumes, version durations

All impact performance…



Considerations for Versions

- Decide how versions will be handled:
  - Lifespan
  - Reconciling
  - Conflict management
  - Naming conventions
  - Structure
    - Staging or QC version between user versions and DEFAULT
    - Security
    - Versions for groups or departments

- Workflow Management Systems for Handling Versions
  - Can provide workflows and efficiencies , some examples:
    - Job Tracking for ArcGIS (JTX)
    - ArcFM and Network Engineer – In the Utility Area

Agenda

- Overview
- Database Design
- Data Maintenance
- Infrastructure Design and Data Distribution
- Security
- Database Maintenance
- Performance

Key Decisions

- System Availability
- Connectivity and Access
- Database Architecture
- Replication and Clustering
- Storage
- Virtualization

Why does System Availability Matter?

- **Down Time**
- **Hardware and Software cost**
  - **More servers or more complex servers**
  - **More servers means more software**
  - **More administration**
- **Maintenance windows**
  - **Compress**
  - **Reconcile services**
  - **Posting services**
  - **Database schema changes**
  - **Database statistics**
  - **Software patching**

System Availability

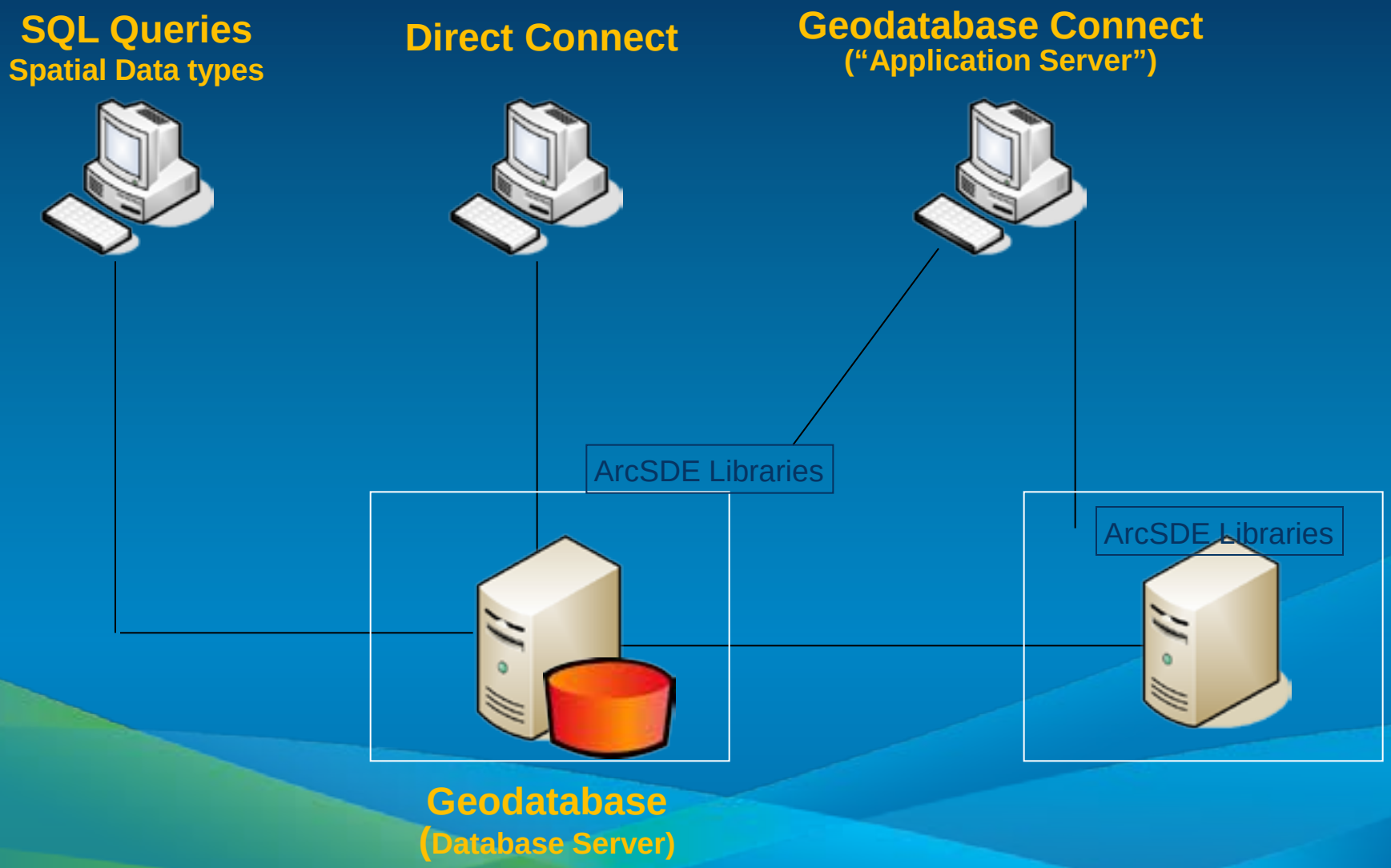
- Primary availability hours
  - 24x7/365
  - Epic “Five 9’s”?

Number of 9s	Percentage Availability	Downtime
1	98.9%	4 Days, 35 minutes
2	99.0%	3 Days, 15 hours
3	99.9%	8 hours, 35 Minutes
4	99.99%	33 minutes
5	99.999%	15 minutes

Availability Options

- Fail-over options
  - Manual –vs.– automated
    - Session Failover not supported
- Clusters
  - Oracle RAC
    - Shared Disk
  - MS Cluster Server
    - Separate Disk
- Replication
  - Database
  - Geodatabase
- Cloud Services
  - 24/7 availability
  - Cost

Geodatabase Connection Architectures



Why connection architecture is important

- Affects system resources on server side
  - Direct Connect uses less on the database server, but more on the client side
- SQL Access
  - May help you decide on storage formats
  - Use database views when using SQL Access
    - Gives the administrators more control of what is accessed
    - Removes versioning complexity from end user
    - More control over how updates are performed
    - Pre-defined queries

Data Access

- **Essential Tasks**
  - **Identify non-GIS application needs**
    - **GIS attribute data**
    - **Business reports based on GIS data or processing**
    - **Reading GIS Geometry data**
    - **Will updates of attribute data occur**
    - **Will updates of geometry occur**
  - **Define and configure the application interfaces based on application needs**
    - **Network configuration (host and ports)**
    - **Client libraries (e.g. SQLNet, Java libs, ArcSDE client libs, etc.)**

Database Architecture

- Multiple instances on the same physical hardware?
  - They are competing for all system resources
  - All background process duplicated (wasteful)
  - One bad apple can spool the bushel
    - One runaway process can affect all databases
- Volume of data
  - If indexes are used properly, this should not be an issue
- Schemas and data ownership

Infrastructure

- Building the hardware and software infrastructure for the Geodatabase instance, and all the related data services
- Essential Tasks
  - Hardware Sizing
    - Identify hardware and software requirements based functional and system needs
      - Development and test
      - Production
      - Licensing
      - System capacity and growth
      - Storage needs
      - Host CPU, RAM
      - Network throughput

Clustering

- **Why use a Cluster**
  - **Fault Tolerance**
  - **Load balancing**
  - **Scalability**

Data Replication

- Essential Tasks
  - Requirements
    - Identify replication uses and benefits
    - Identify data to be replicated
    - Identify QoS requirements
      - how fast should changes replicate?
  - Analysis and Design
    - Define replication architecture
  - Implementation
    - Prototype and test architecture (crucial)
      - Key data modifications
      - Typical and peak loads
    - Procure, install, and configure replication architecture

Data Replication

- Data replication
  - Why replicate
    - Recovery
    - Mobility
    - Accessibility
    - Performance/load balancing
    - Scalability
  - Issues to Consider
    - 2 Way complexity
    - Data model

Data Replication

- **Data replication**
  - **Review replication options**
    - **Device level, OS level, DBMS level, Geodatabase**
    - **RDBMS Types**
      - **Snapshot**
      - **Multi-master/merge**
      - **Transactional**
      - **Hybrid**
      - **Cannot edit using RDMBS replicas, only parent can be edited**

Geodatabase Replication

- Decide what is going to be replicated
  - Specific feature classes and feature datasets
- Decide on data to be replicated
  - Complete
  - By area
  - By attribute
  - Non-spatial tables
- Decide on type of replication
  - Checkout/checkin
  - One way
    - Versioned
    - Non-versioned
  - Two way
- How to perform synchronization
  - On line or off line

Data Replication cont'd

- Deliverables
  - Document requirements and design
  - Full cycle of prototyping
    - Procure and configure replication software/hardware
    - Build master database
    - Modify data, and measure success and performance of replica
  - Configured and tested replication system

Agenda

- Overview
- Database Design
- Data Maintenance
- Infrastructure Design and Data Distribution
- Security
- Database Maintenance
- Performance

Security

- **Preventing Unauthorized access or editing of the Geodatabase**
- **Essential Tasks**
  - **Understand Geodatabase model and security effects**
  - **Review DBMS authentication schemes**
  - **Identify anticipated users (GIS and business applications), and accessible objects**
  - **Database Service Accounts**

Security

- DBMS authentication schemes
  - Integrated with OS and network domain security
  - Standard DBMS security
  - Mixed mode
  - Users and roles

Security

- Geodatabase
  - Feature Classes
  - Relationship Classes
    - Simple (1-1, 1-N)
    - Complex (M-N)
      - Creates underlying join table
  - Feature Datasets
    - Feature Classes
    - Complex objects
      - Networks, Topologies, etc

Security

- **Feature Datasets**
  - **Designed to house objects that work together in some way**
    - **Geometric Network**
  - **Feature Datasets**
    - **Common Spatial Reference**
    - **Common Permissions**
    - **All locked at same time**
    - **Non-Visible elements**

Security

- **Relationship Classes**
  - **Related objects can have different permissions**
  - **Could affect workflow and/or editor permissions**

Security

- **Object Level Security**
  - **ROW Level Security or Fine Grained Access**
  - **Very complex to implement**
  - **Sometimes, better implemented at application level**

Security

- Challenges and Risks
  - Sharing a DBMS login
    - SDE_logfile contention point
    - Difficult to identify which process belongs to which user
    - Security
  - Access to too many objects can impact performance
- Note
  - It’s easier to grant access to users later, than it is to revoke later.

Agenda

- Overview
- Logical Design
- Infrastructure Design and Data Distribution
- Data Maintenance
- Security
- Database Maintenance
- Performance

Database Maintenance

- **Common Tasks**
  - **Backups**
  - **Statistics**
  - **Fragmentation**
  - **Compress**
  - **Batch Reconcile**

Data Backup & Recovery Considerations

- **Key considerations**
  - System availability
  - Backup sizes
  - Speed of recovery
  - Transportability
  - Acceptable loss of edits
  - Consistency
  - Affects on performance

Database Statistics

- All RDBMS optimizers use statistics (metadata) to develop execution plan
- Many DBA's want to estimate as opposed to compute
  - Quicker
  - Estimating only works well if data is uniform
- Better statistics, better execution plan
- Key questions
  - How long will it take
  - When can it be performed
  - Can it be down while users are connected

Fragmentation

- Index Fragmentation
  - When to rebuild?
    - One of the great mysteries of life
  - How to rebuild (Oracle bug in rebuild command)
    - On line
    - Off line (Saves redo log)
    - Drop and recreate
- Table Fragmentation
  - Rarely causes problems
  - Only a concern when reading a large number of blocks

Database Monitoring

- Monitoring geodatabase components
- Version/Stat info
  - Replication versions
  - State info
  - State lineage info
  - Number of features
- Data access time
  - Monitor performance of queries, especially spatial
    - Spa-stats or spatial stats

Agenda

- Overview
- Logical Design
- Infrastructure Design and Data Distribution
- Data Maintenance
- Security
- Database Maintenance
- Performance

Performance

- Deliverables
  - Document requirements
  - Execute performance, analyze, optimize iterations
  - Tuning DBMS, tuning application
  - Scaling strategy
    - Scale out vs. up
- Challenges and Risks
  - Data too granular
    - Group features
  - Overloading your application
    - Overloading application table of contents
    - Building batch-like operations into application

Performance Objectives

- Define performance metrics
  - Identify key tasks
  - Establish initial goal
- Proto-type database
  - Reasonable sample database
  - Spatial density
  - Model behavior
  - Spatial reference and bounds

Data Performance and Scalability

- Measure, assess, and optimize the performance of key functionality using the geodatabase instance.
- Essential Tasks
  - Review anticipated data loads
    - Volume (data file growth management)
    - Volatility (storage partitioning)
  - Identify key business transactions
    - Maintenance operations
    - Publication operations
  - Identify performance requirements for key business transactions
    - Response time
    - Initial and scheduled user loads
    - Throughput
    - Testing

Performance

- Geodatabase designs
  - Potential performance issues related to database design
    - Relationships
      - Both # and Type
      - Schema Cache can help reduce performance cost
    - Size of data stored in records
    - Projection on the fly
    - Number of records returned in a query
    - Density of data, both number of features and number of vertices
- Application design
  - Can have a significant affect on performance; e.g.,
    - Frequently opening a table
    - Retrieving features one at a time vs bulk

Question & Answer

Contact info:

John Alsup
jalsup@esri.com

Matt Bottenberg
mbottenberg@esri.com