TELVENT
Miner & Miner

IT for a Sustainable and Secure World

Successful Large Utility GIS Implementations

John Dirkman, P.E.
Sr. Project Manager
Telvent Miner & Miner

Presentation Focus

This presentation will focus on *large* utility companies, but is applicable to *any size* utility

Also, the challenges inherent in *implementing* are similar to those inherent in *upgrading*

Success Requirements

What is a successful implementation project?

- On schedule
- On budget
- Meets business requirements

If you want a successful project:

- Set a realistic schedule and meet each milestone
- Set an adequate scope and contain it
- Get resource commitments from Business and IT
- Establish a collaborative team environment

Top Project Killers

1. Changing requirements
2. Overly aggressive schedule
3. Inadequate testing
4. Inadequate user training
5. Lack of corporate support
6. Core team changes

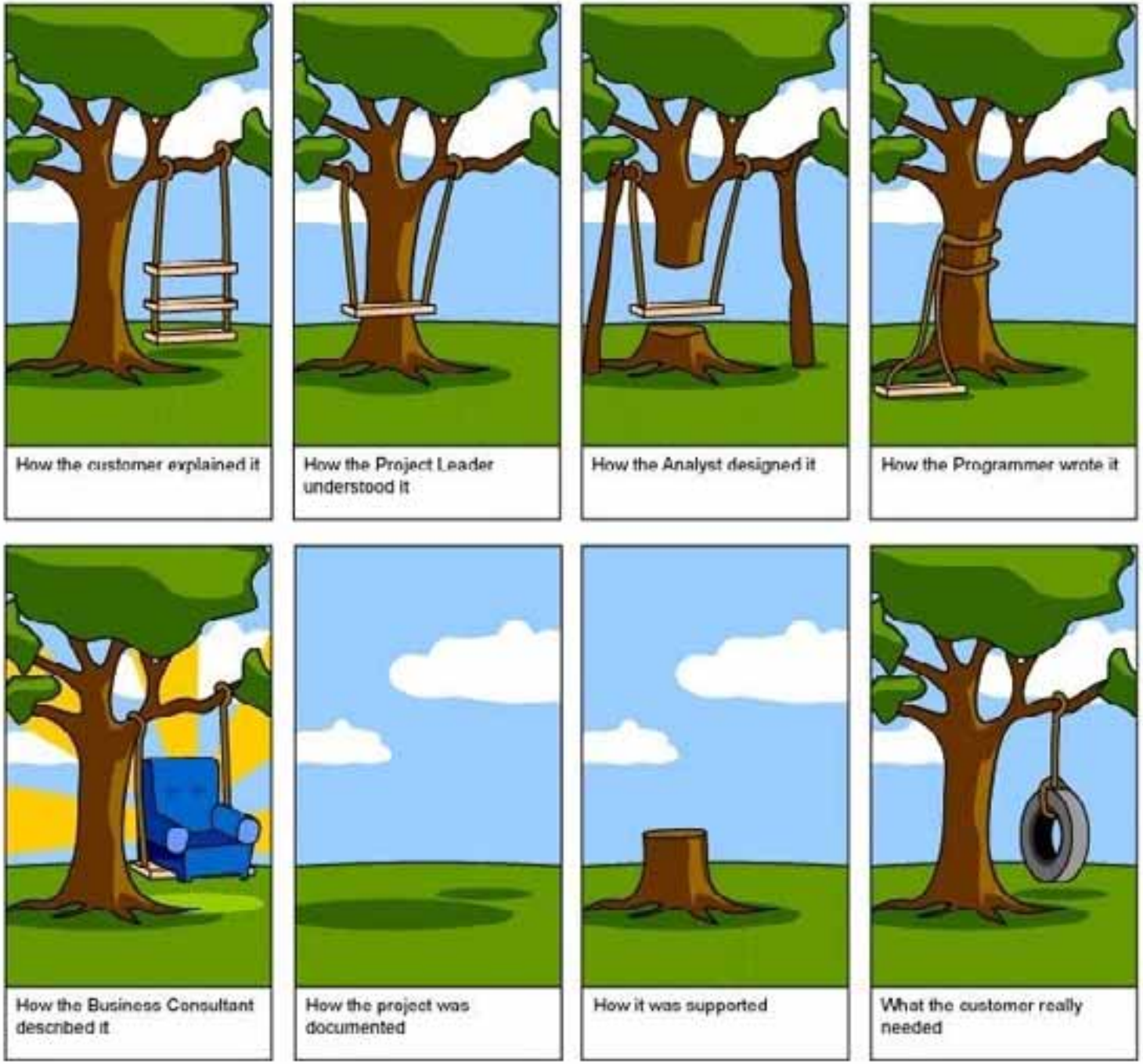
It's about managing expectations and risk.

Top Schedule Drivers

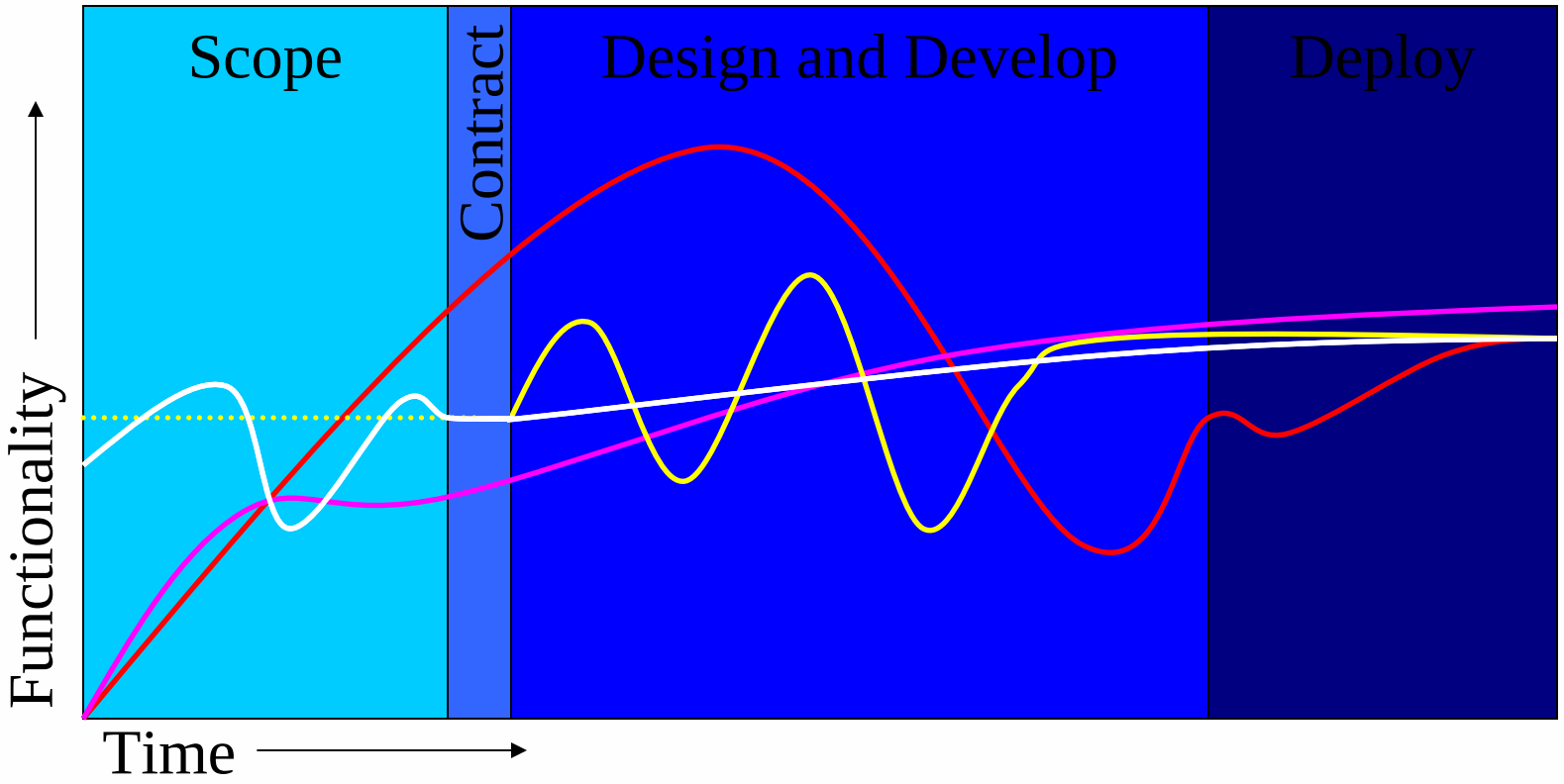
- Customization
  - Complexity
  - Number of interfaces
- Conversion of data
  - Quantity, Complexity, Quality
  - Phased Implementation
- Configuration of ArcGIS and ArcFM
  - Business requirement understanding
- Across all 3 activities: corporate experience
  - Implementation experience: IT, GIS, ArcGIS, ArcFM
  - Experience with corporate structure, legacy systems, and business processes

Successful Large Utility Upgrades

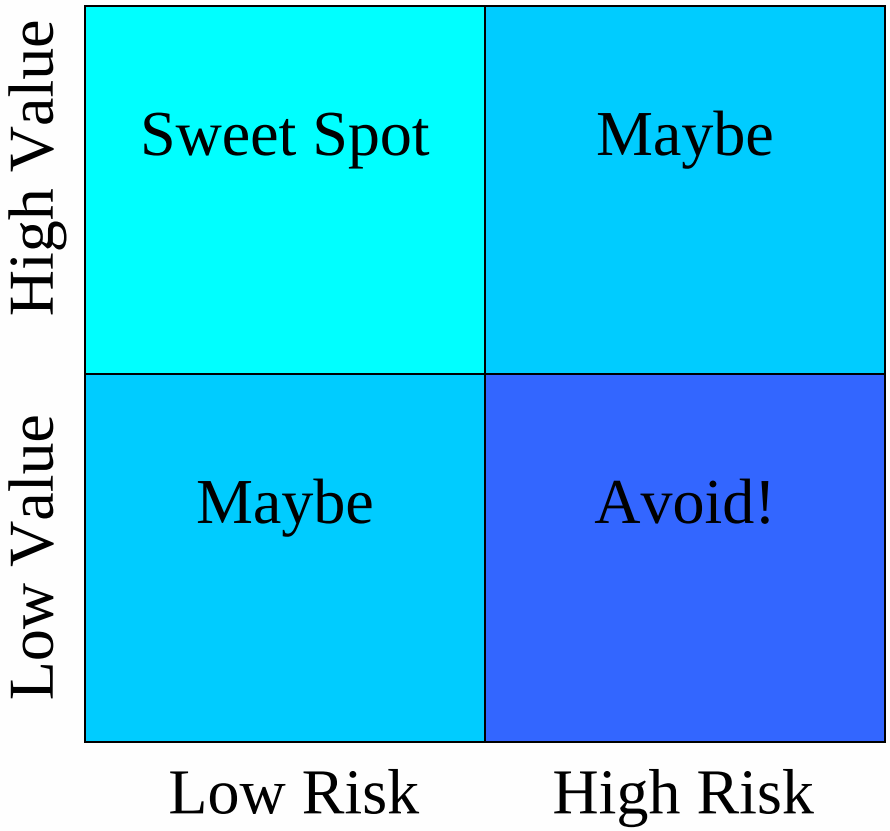
1. Expectation Management
2. Methodology & Risk Mitigation


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
How it was supported
What the customer really needed

User Expectations



Risk and Value



Risk Evaluator

From Agile Estimating and Planning, by Mike Cohn

Expectation Management

- User expectations
  - Faster
  - More stable
  - More functionality
  - Intuitive and easy to use
  - Not too different
- Business expectations
  - Cheaper to operate
  - Cheaper to maintain
  - On schedule and budget
- It’s the game of expectation management
- Make sure you understand the project drivers

Controlling Expectations

Per Building Your ArcFM:

1. Expose the technology early and often to both users and management
2. Involve key users in the Core Team to build internal advocates
3. Don't give the technology to users until they are ready for it, and vice versa

Also establish a Steering Committee – your information conduit upwards

Methodology & Risk Mitigation

1. Gap Analysis
2. Data Migration
3. Collaborative Design, Programming, Configuration, and Testing
4. Version Schemes and Version Maintenance
5. Strategies for Successful Training, Deployment, and Post-Deployment Support
6. Performance

Gap Analysis

- Answers the question: What do you need the product to do that it can't do out of the box
- You may need to think creatively; sometimes a small change in the business requirement or workflow can reduce customization
- Bring in users who are familiar with the current system as SME's
- Look at the application's What's New (upgrades)
- Develop a list of likes and dislikes of the current system and prioritize this list
- This will help define expectations that lead to requirements

Data Conversion/Migration

- Focus on careful data modeling and build in multiple migration iterations and QC checks
- Avoid changing the data model late in the game
- Use SQL queries for feature counts and attribute checks
- Track and manage data model changes closely, changes will impact migration and configuration

Comments on Feature and Object Classes:

#	Feature/Object Class	Comment	Reviewer	Action	Verified
1.	Anomaly	Set the default values for SymbolRotation and BeginStationValue to null.	Tom N.	Done → KL 5/1/06	
2.	ControlPoint	Set the default value for SymbolRotation to null.	Tom N.	Done → KL 5/1/06	
3.	LeakRepairReport	Add LeakReportedBy (text5) field. Assign Source Reported By domain (new domain, see below).	JD	Done → KL 5/1/06	
4.	LeakRepairReport	Delete the Address field	Tom N.	Done → KL 5/1/06	
5.	LSO	Add NAD27X (double), NAD27Y (double) and CollectSoilIndicator (text5) fields. Assign the Yes/No Indicator domain to the CollectSoilIndicator field.	Tom N.	Done → KL 5/1/06	
6.	RemovedGasPipe	Delete this object class.	Tom N.	Done → KL 5/1/06	

Environment Management

- Careful environment management is essential

Environment	App Server	Installer	Integration Server	Installer	Database	Queue
Electric/Land Production	ArcFM1	A35	ILS1	I23	P5151	INBOUND.DEFAULT
Electric/Land Testing	ArcFM3	A36	ILSDR2	I24	T5151	INBOUND.TEST2
Electric/Land Training	ArcFM1	A35	ILS2	I23	T5252	INBOUND.TEST4
Gas Testing	ArcFM2	A52	ILSDR1	I28	T5858	INBOUND.TEST1
Designer Testing	ArcFM4	D10	ILS3	I28	T5656	INBOUND.TEST3
Gas Cutover	ArcFM2	A52	ILSDR3	I28	T5252 (temp)	

Collaboration

- All design and testing must be collaborative
- Programming and configuration may be collaborative as well
- Challenges of collaborative programming
  - Error handling
  - Base class utilization
  - Technical support
  - It isn’t easy, it adds risk, and it may end up being more expensive
- The benefit is that the utility builds experience in customization, troubleshooting, and maintenance

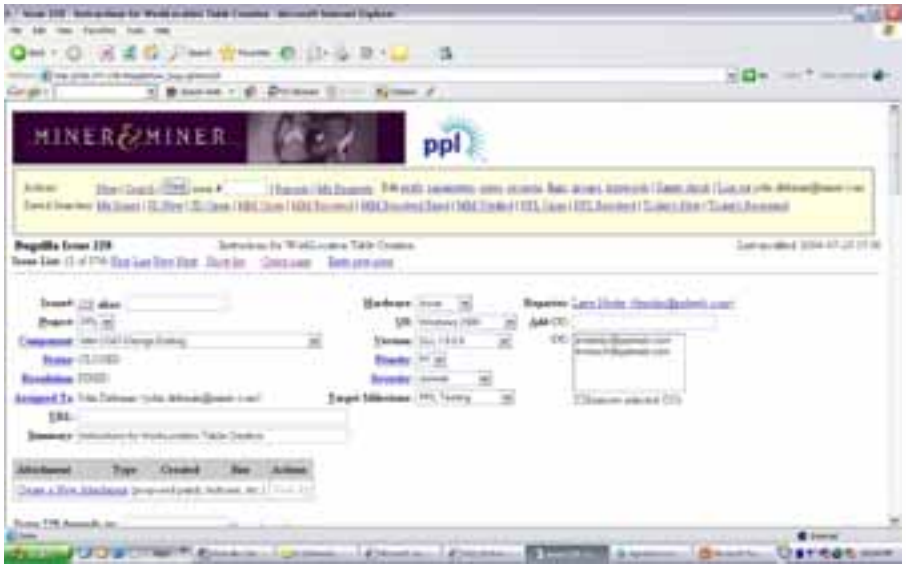
Collaboration

For collaborative programming and configuration:

- Make sure your programmers and configurers are properly trained
- Establish standard error handling and class utilization techniques
- Establish a vehicle for technical support
- Try to be more than one deep
- The more collaboration and customization, the more testing required
- For maintenance, plan for a knowledge transfer and a support period
- Provide access to Development environment via VPN or Webex Access Anywhere

Testing

- Conduct a daily wrap-up
- Have clearly defined acceptance criteria
- Define your P1, P2, and P3 categories
- Use an issue tracking system – Bugzilla
- Bugzilla becomes a knowledge base
- For PPL, we tracked 1514 issues; issues included bugs, enhancements, tasks, training issues, etc.
- Stress-test your system via formal stress tests and during training



Testing

Where do you test – Factory or Site?

- All tests at client site – Pros
  - Essential with interfaces for end-to-end testing
  - Knowledge transfer starts earlier
  - Errors in Test Procedures caught earlier
  - More time for Change Requests
- All tests at client site – Cons
  - First round of testing can be ugly
- The terms FAT and SAT don't fit – try "Integrated Testing"

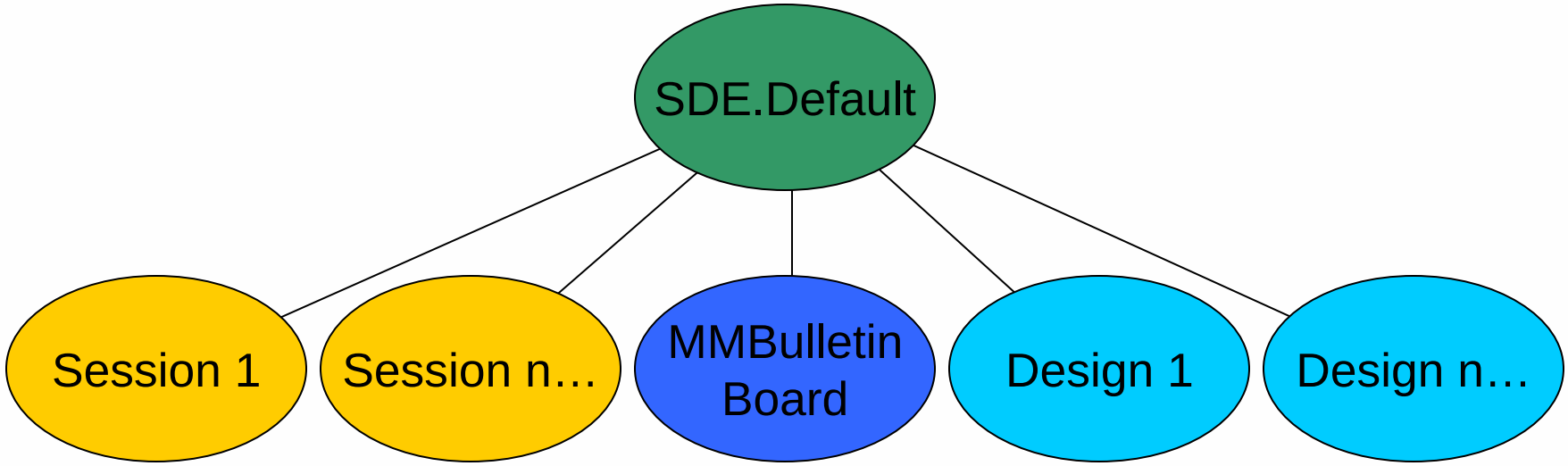
Testing

- Traditional
  - Install
  - Run All Tests
  - Resolve High Priority Issues – Code, Configuration, Data
  - Repeat
- Accelerated
  - Same as above for first round of testing
  - Subsequent tests as follows:
  - Install
  - Repeat Failed Tests
  - Resolve High Priority Issues – Code, Configuration, Data
  - Repeat – New Installer every evening

FAT 1	Rework/ Retest	FAT 2	Rework/ Retest	SAT 1	Rework/ Retest	SAT 2	14 WEEKS
2 WEEKS	2 WEEKS	2 WEEKS	2 WEEKS	2 WEEKS	2 WEEKS	2 WEEKS	

Version Schemes

- For long-term transactions, use one off Default
- MMBulletinBoard as sibling with Designs and Sessions



Reconcile and Posting Services

- Good opportunity for labor and cost savings
- Batch Post
  - No need to tie up ArcFM while the user waits for their version to reconcile/post
  - Versions in conflict are bypassed
- Batch Reconcile
  - Run at night and on weekends (multiple services)
  - Compress afterwards
  - Keeps the state tree shorter and improves performance
- Conflict Filters
  - Automatically accept the parent, but log the results (via XML) for next-day resolution
  - Look at the benefits of more advanced conflict filtering

Training

- Design: Train the Core Team
- Deploy: Train-the-Trainer
- Get your trainers involved in the Design and Testing process
- Testing is a form of training, but it can be a little rough; manage expectations
- Provide Just-in-Time training
- Provide a “sandbox” environment for users to play in

Deployment

- Have a Cutover Plan and practice it
- Host an open house and/or demos to the field
- Do a pilot, then adjust training
- Have work ready for users – backlog
- Carefully control scope
- Set up your help desk
- Establish on-site front-line support (former testers) to fortify your help desk
- Throw a great party

Post-Deployment Support

- Users are going to try things that you never suspected – set a standard normal.mxt and lock it
- Have internal resources available for troubleshooting
- Have a contract vehicle in place for post-deployment support
- Provide follow-on training opportunities
- Establish a user committee and look for more quick wins

Performance

- Benchmark your system
- Be consistent
- Sample tasks may include:
  - Opening Stored Display
  - Opening Session/Design
  - Placing various features: placement 1 and N
  - Trace a Conductor
  - Save Session/Design
  - Run QA/QC
  - Plot
  - Reconcile
  - Post
- Solicit users for additional frequently used tasks
- Plan for required performance; user counts and roles, hardware, and network
- Nightly Reconcile/Compress/Analyze
- Tune your system periodically

Performance

Happy User(s) or Viewer(s)

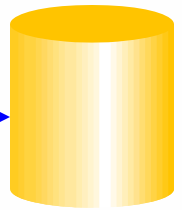
- Training
- Quantity

•Network
•Message Queues

Legacy System(s)

Network

Network



User's PC's

- Operating System
- PC Specs

Application & Integration Server(s)

- Citrix Load Balancing
- Operating System
- ArcGIS
- ArcFM
- Stored Display
- Customizations
- Server Specs

Database Server

- Operating System
- DMBS (Oracle, SQLServer, etc.)
- Server Specs

Database

- Maintenance (R/P/D/C/A)
- Tuning

Optimizing Stored Displays for Performance

1. Set reasonable scale suppression for all feature classes within the stored display.
2. Include all related features and objects. If the related item is in the stored display ArcMap has the information local and does not need to continually go back to the database to retrieve it.
3. Avoid joins and relates.
4. Avoid halos and masks.
5. Avoid definition queries (if possible).
6. Include network junctions and all features participating in the geometric network if the stored display will be used for editing features that participate in the geometric network.
7. Remove any unnecessary feature classes from the stored display. Organize stored displays based on the type of work being accomplished.
8. Avoid intermediate tables in 1:1 and 1:M relationships; instead, store the foreign key in the child/destination object class.

Optimizing Stored Displays for Usability

1. Organize the stored display in the following order (from top to bottom): annotation, point features, lines, then polygons
2. Use Group Layers to logically group layers to allow users to easily turn on and off groups of data.
3. Alphabetize layers within group layers and after organizing by anno, point, line, and polygon features. This will make it much easier and faster for users to find layers.
4. Reorder attributes (via ArcCatalog): move required attributes up to the top, and group attributes logically.
5. Rename the data frame to match the name of the stored display. This makes it easy to see which stored display you are in.
6. Make landbase layers unselectable in the Selection Tab of the Table of Contents for stored displays that will not be used for landbase edits. This helps prevent inadvertent moving and deleting of landbase layers.

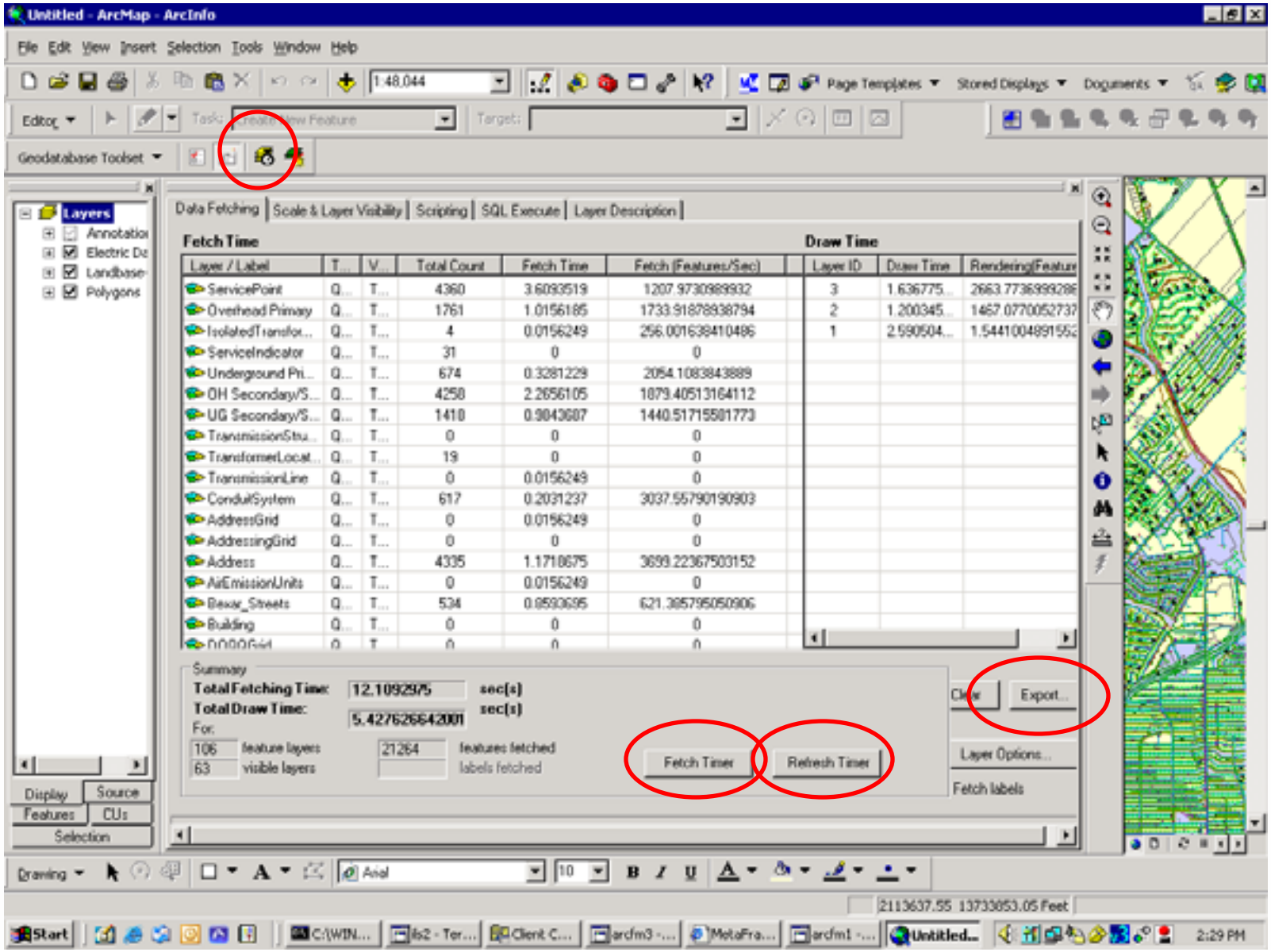
Geodatabase Toolset (GDBT)

- Geodatabase Toolset:
  - Contains tools used to investigate and diagnose geodatabase performance issues
  - Provides different toolsets for ArcCatalog and ArcMap
  - ArcMap: used to review characteristics of Stored Displays, refresh performance, scripting, and DBMS tracing
  - ArcCatalog: used to review enterprise geodatabase information
  - Not a substitute for tuning

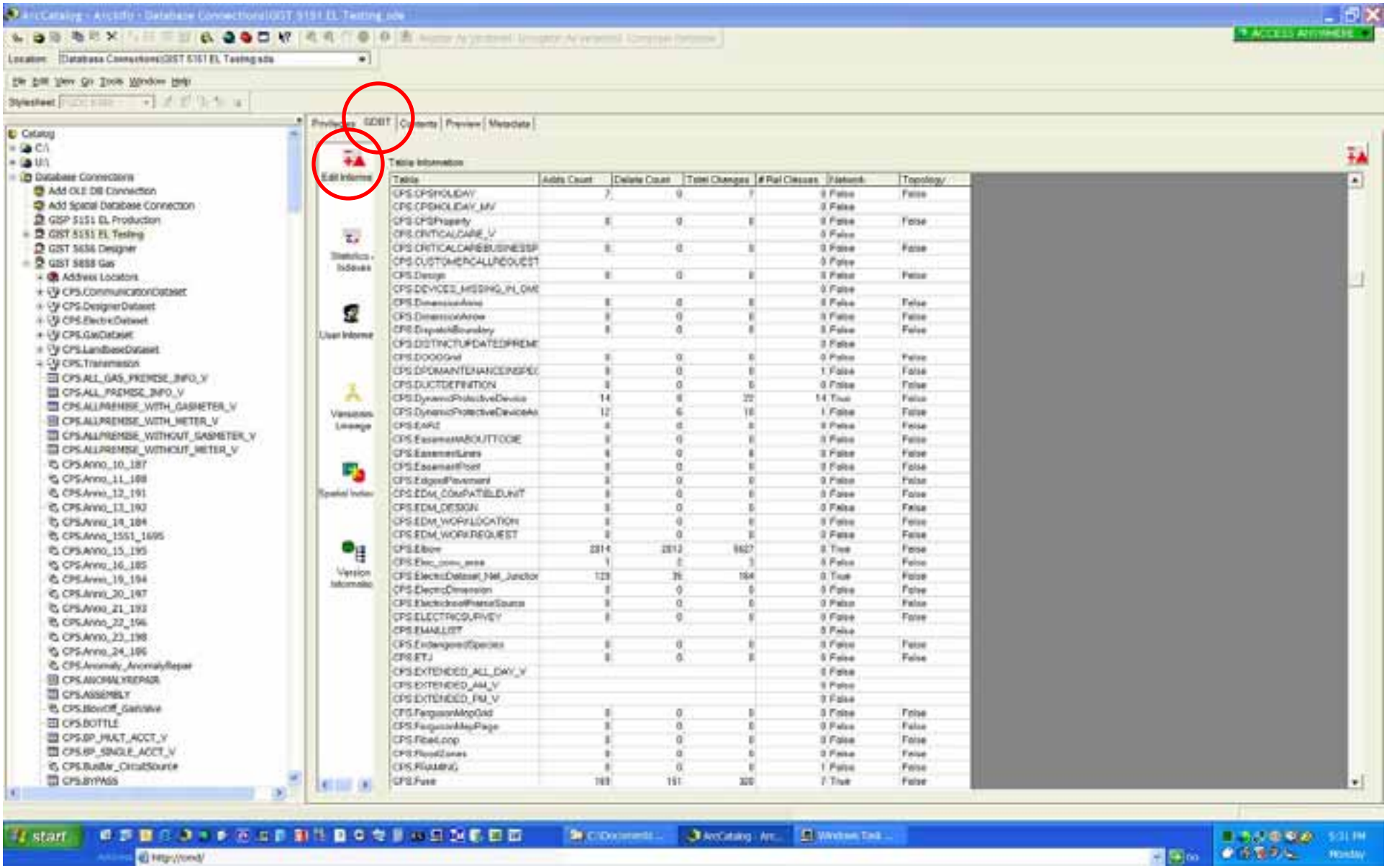
http://www.esri.com/software/arcgis/extensions/gdbt/download.html or http://www.miner.com/freetools/

Alternative: MXDPERFSTAT from arcscripts.esri.com

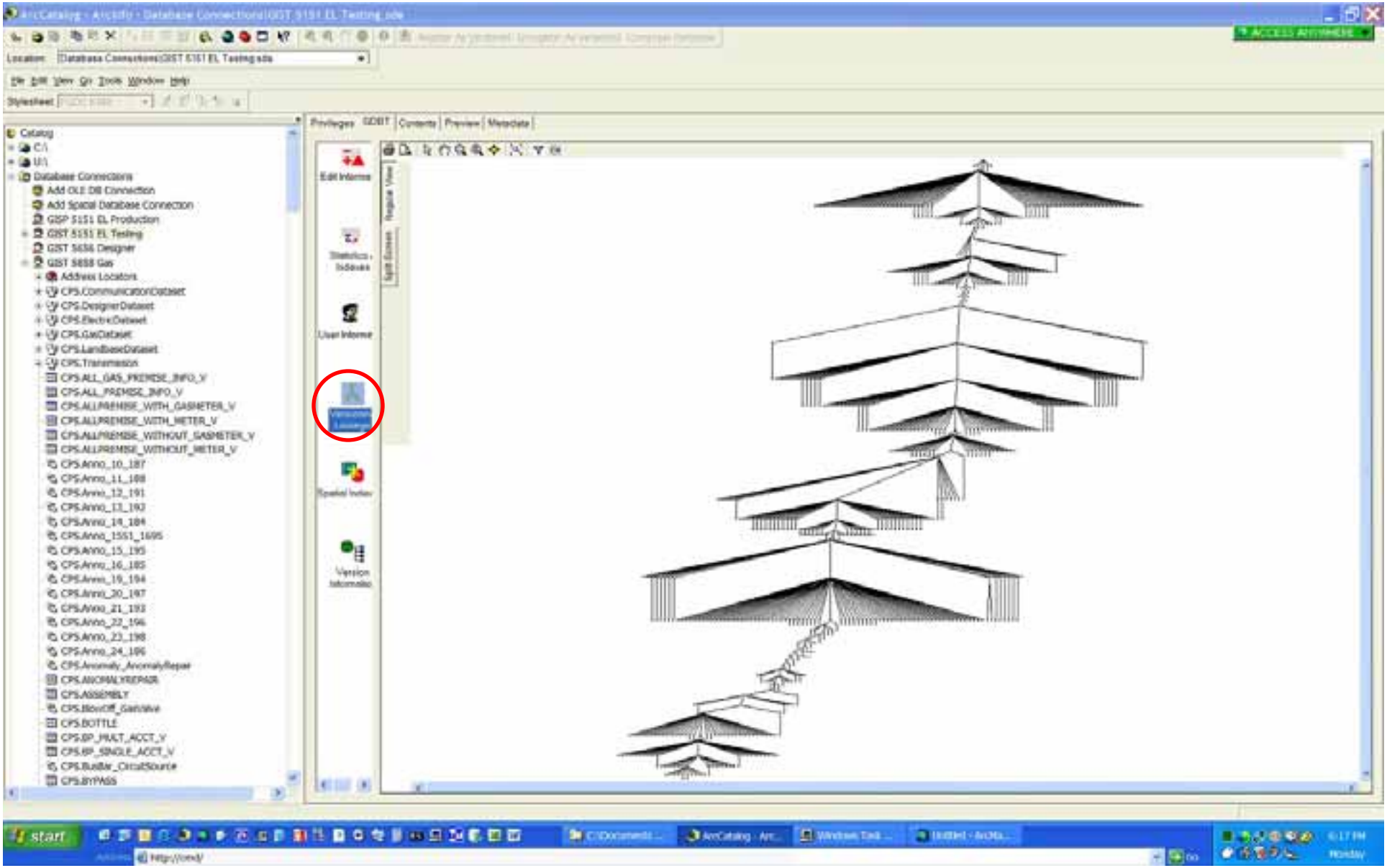
ArcMap - Data Fetching



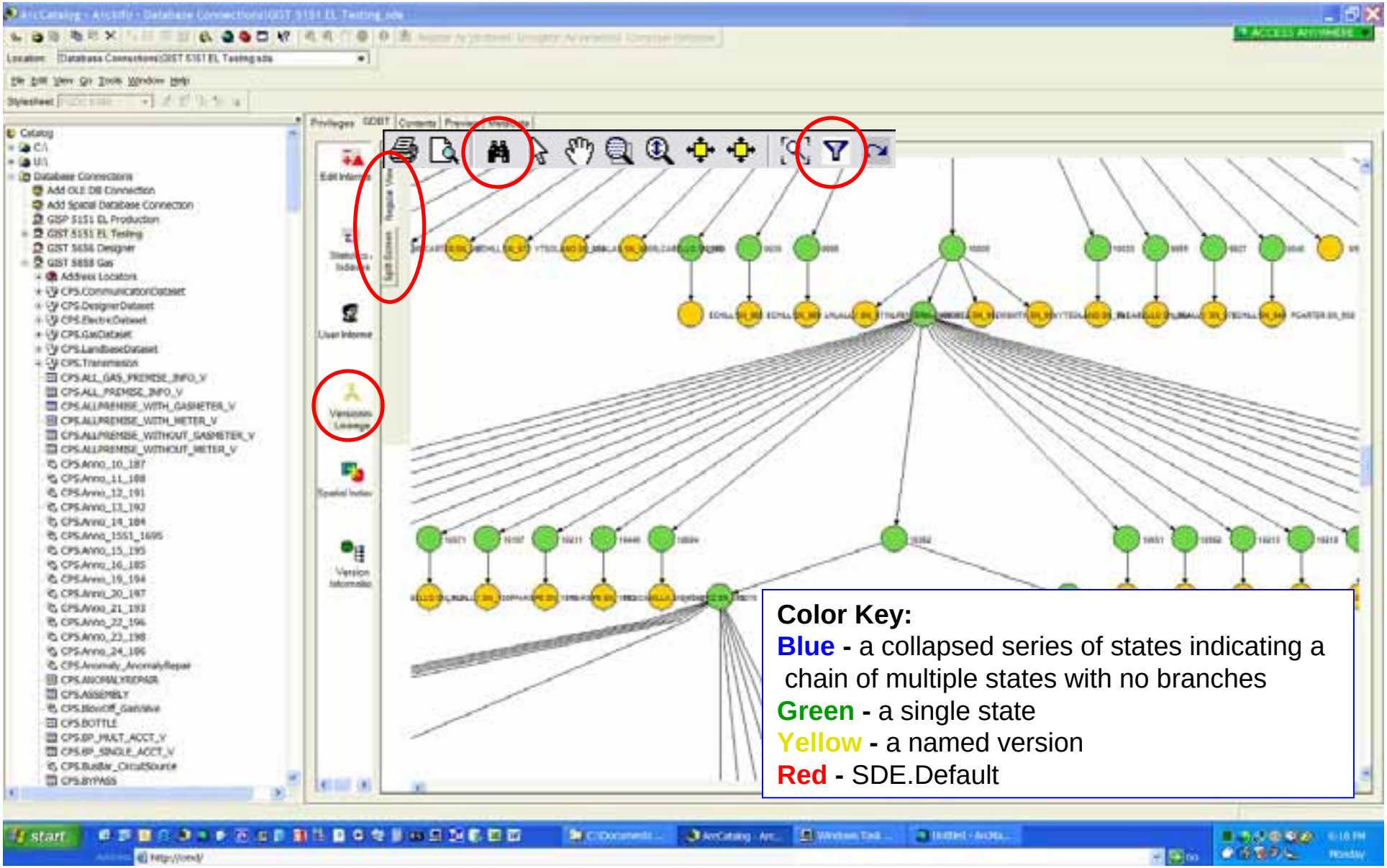
ArcCatalog - Edit Information



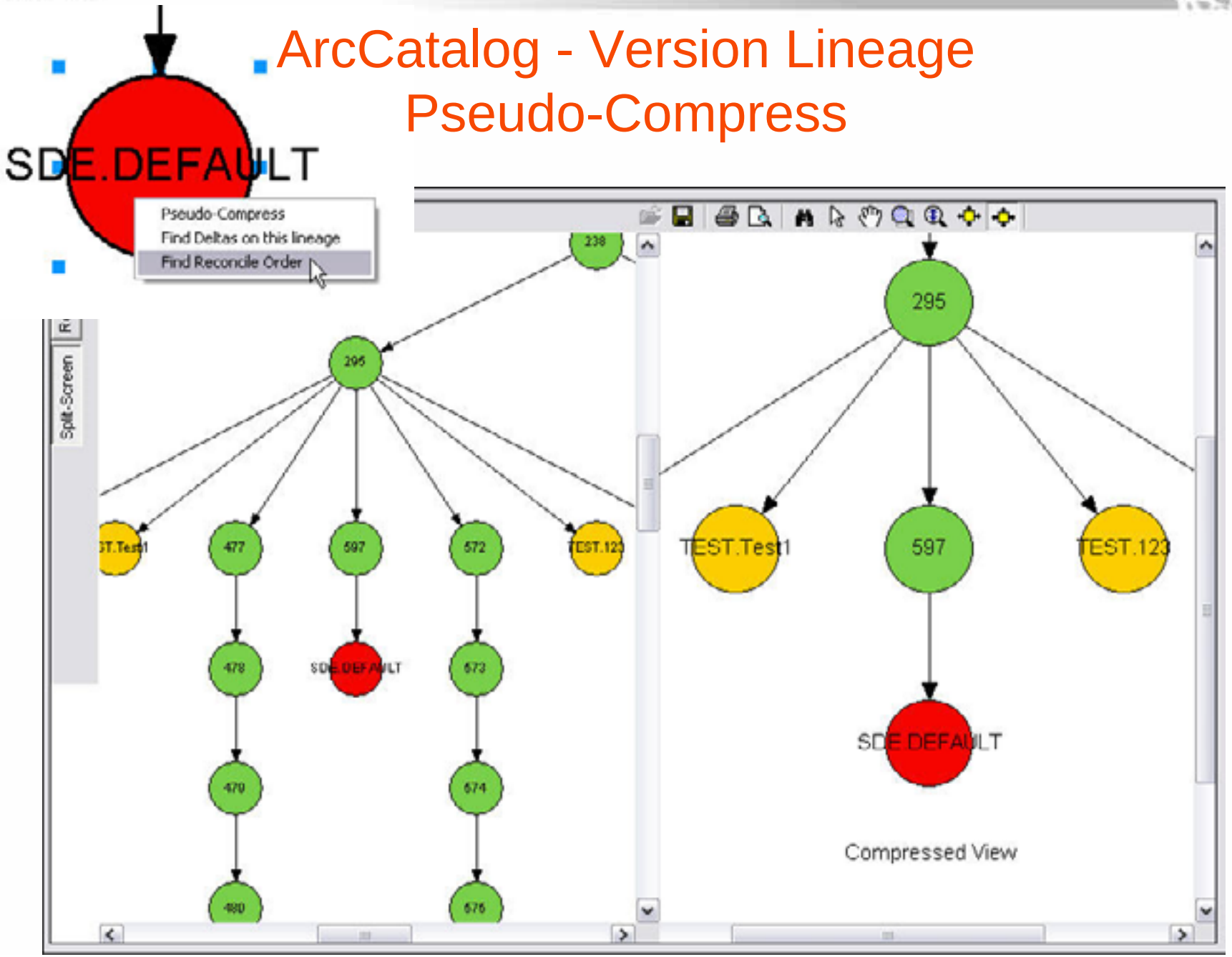
ArcCatalog - Version Lineage

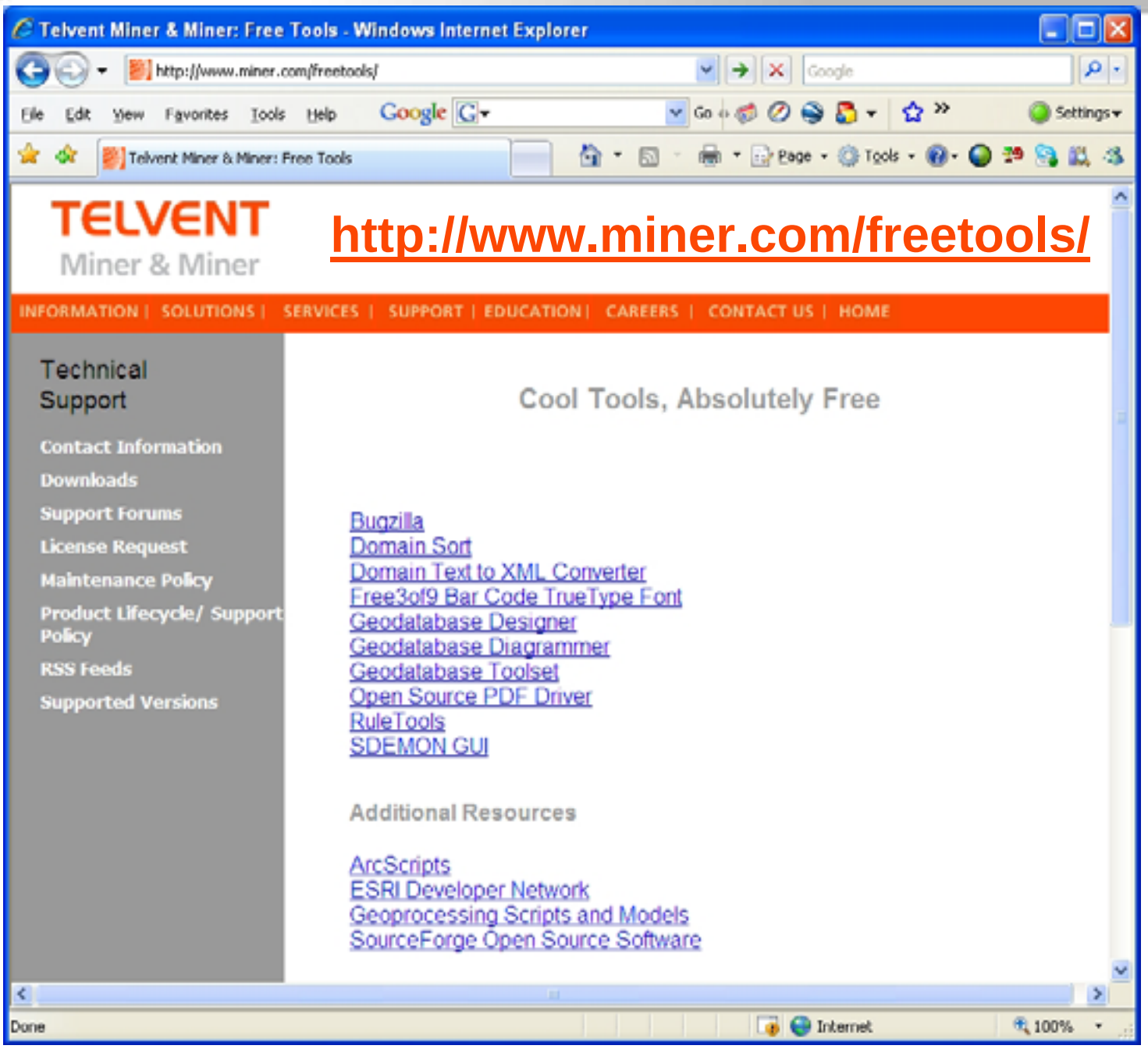


ArcCatalog - Version Lineage



ArcCatalog - Version Lineage Pseudo-Compress




Telvent Miner & Miner: Free Tools - Windows Internet Explorer
http://www.miner.com/freetools/
Telvent Miner & Miner: Free Tools
TELVENT
Miner & Miner
http://www.miner.com/freetools/
INFORMATION | SOLUTIONS | SERVICES | SUPPORT | EDUCATION | CAREERS | CONTACT US | HOME
Technical Support
Contact Information
Downloads
Support Forums
License Request
Maintenance Policy
Product Lifecycle/ Support Policy
RSS Feeds
Supported Versions
Cool Tools, Absolutely Free
Bugzilla
Domain Sort
Domain Text to XML Converter
Free3of9 Bar Code TrueType Font
Geodatabase Designer
Geodatabase Diagrammer
Geodatabase Toolset
Open Source PDF Driver
RuleTools
SDEMON GUI
Additional Resources
ArcScripts
ESRI Developer Network
Geoprocessing Scripts and Models
SourceForge Open Source Software
Done
Internet
100%

Incentives

- Take the old tools away
- Police the tool shed
- Tech/Tech Clerk Case Study

Conclusion

- Work to understand and manage expectations
- Use the 3D methodology and don't cut corners.
  Every $1 spent in design = $100 in development.
- Monitor top management support for the project, and keep management appraised frequently
- Know the strengths & weaknesses of your team; leverage the strengths, mitigate the weaknesses
- Establish a collaborative team environment
- Set a realistic schedule and keep it
- Set an adequate scope and contain it
- Get resource commitments from Business and IT

Conclusion

May every project be successful!

Is there really any other option?

Questions?