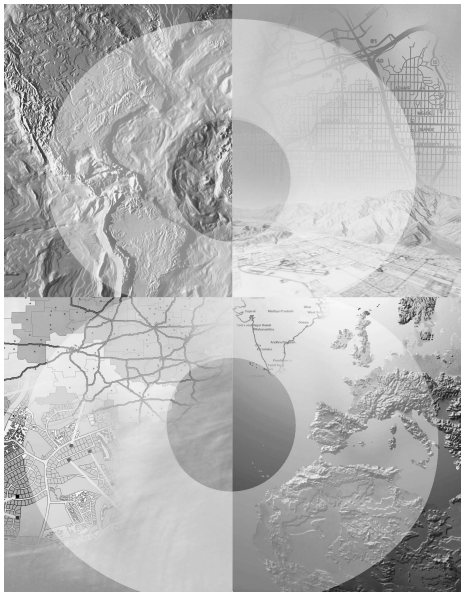
An ESRI White Paper



Managing Workflow with Versions

October 2000

ESRI 380 New York St., Redlands, CA 92373-8100, USA • TEL. 909-793-2853 • FAX 909-793-5953 • E-MAIL info@esri.com • WEB www.esri.com

Copyright © 2000 Environmental Systems Research Institute, Inc.
All rights reserved.
Printed in the United States of America.

The information contained in this document is the exclusive property of Environmental Systems Research Institute, Inc. This work is protected under United States copyright law and other international copyright treaties and conventions. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage or retrieval system, except as expressly permitted in writing by Environmental Systems Research Institute, Inc. All requests should be sent to Attention: Contracts Manager, Environmental Systems Research Institute, Inc., 380 New York Street, Redlands, CA 92373-8100, USA.

The information contained in this document is subject to change without notice.

U.S. GOVERNMENT RESTRICTED/LIMITED RIGHTS

Any software, documentation, and/or data delivered hereunder is subject to the terms of the License Agreement. In no event shall the U.S. Government acquire greater than RESTRICTED/LIMITED RIGHTS. At a minimum, use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in FAR §52.227-14 Alternates I, II, and III (JUN 1987); FAR §52.227-19 (JUN 1987) and/or FAR §12.211/12.212 (Commercial Technical Data/Computer Software); and DFARS §252.227-7015 (NOV 1995) (Technical Data) and/or DFARS §227.7202 (Computer Software), as applicable. Contractor/Manufacturer is Environmental Systems Research Institute, Inc., 380 New York Street, Redlands, CA 92373-8100, USA.

ESRI, ARC/INFO, ArcCAD, ArcView, *Business*MAP, MapObjects, PC ARC/INFO, SDE, and the ESRI globe logo are trademarks of Environmental Systems Research Institute, Inc., registered in the United States and certain other countries; registration is pending in the European Community. 3D Analyst, ADF, ARC COGO, the ARC COGO logo, ARC GRID, the ARC GRID logo, the ARC/INFO logo, AML, ARC NETWORK, the ARC NETWORK logo, *ArcNews*, ARC TIN, the ARC TIN logo, ArcInfo, the ArcInfo logo, ArcInfo LIBRARIAN, ArcInfo—Professional GIS, ArcInfo—The World's GIS, ArcAtlas, the ArcAtlas logo, the ArcCAD logo, the ArcCAD WorkBench logo, ArcCatalog, the ArcData logo, the ArcData Online logo, ArcDoc, ARCEDIT, the ARCEDIT logo, ArcEurope, the ArcEurope logo, ArcEditor, ArcExplorer, the ArcExplorer logo, ArcExpress, the ArcExpress logo, ArcFM, the ArcFM logo, the ArcFM Viewer logo, ArcGIS, ArcIMS, the ArcIMS logo, ArcLogistics, the ArcLogistics Route logo, ArcMap, ArcObjects, ArcPad, the ArcPad logo, ARCPLOT, the ARCPLOT logo, ArcPress, the ArcPress logo, the ArcPress for ArcView logo, ArcScan, the ArcScan logo, ArcScene, the ArcScene logo, ArcSchool, ArcSDE, the ArcSDE logo, the ArcSDE CAD Client logo, ArcSdl, ArcStorm, the ArcStorm logo, ArcSurvey, ArcToolbox, ArcTools, the ArcTools logo, ArcUSA, the ArcUSA logo, *ArcUser*, the ArcView GIS logo, the ArcView 3D Analyst logo, the ArcView Business Analyst logo, the ArcView Data Publisher logo, the ArcView Image Analysis logo, the ArcView Internet Map Server logo, the ArcView Network Analyst logo, the ArcView Spatial Analyst logo, the ArcView StreetMap logo, the ArcView StreetMap 2000 logo, the ArcView Tracking Analyst logo, ArcVoyager, ArcWorld, the ArcWorld logo, Atlas GIS, the Atlas GIS logo, AtlasWare, Avenue, the Avenue logo, the *Business*MAP logo, DAK, the DAK logo, DATABASE INTEGRATOR, DBI Kit, the Digital Chart of the World logo, the ESRI Data logo, the ESRI Press logo, ESRI—Team GIS, ESRI—The GIS People, FormEdit, Geographic Design System, Geography Matters, GIS by ESRI, GIS Day, the GIS Day logo, GIS for Everyone, GISData Server, *Insite*MAP, MapBeans, MapCafé, the MapCafé logo, the MapObjects logo, the MapObjects Internet Map Server logo, ModelBuilder, MOLE, the MOLE logo, NetEngine, the NetEngine logo, the PC ARC/INFO logo, PC ARCEDIT, PC ARCPLOT, PC ARCSHELL, PC DATA CONVERSION, PC NETWORK, PC OVERLAY, PC STARTER KIT, PC TABLES, the Production Line Tool Set logo, *Route*MAP, the *Route*MAP logo, the *Route*MAP IMS logo, Spatial Database Engine, the SDE logo, SML, StreetEditor, StreetMap, TABLES, The World's Leading Desktop GIS, *Water Writes*, and Your Personal Geographic Information System are trademarks; and ArcData, ArcOpen, ArcQuest, *ArcWatch*, ArcWeb, Rent-a-Tech, Geography Network, the Geography Network logo, www.geographynetwork.com, @esri.com, and www.esri.com are service marks of Environmental Systems Research Institute, Inc.

The names of other companies and products herein are trademarks or registered trademarks of their respective trademark owners.

Managing Workflow with Versions

An ESRI White Paper

Contents **Page**

J-8560

Contents **Page**

Managing Workflow with Versions

Note: This white paper is an abridged version of Chapter 7 from Modeling our World—An ESRI Guide to Geodatabase Design. *The book is included as a manual with ArcGIS™ software. Additional copies of* Modeling our World—An ESRI Guide to Geodatabase Design *can be ordered from ESRI or commercial booksellers such as Amazon or Barnes and Noble.*

Many applications of geographic information system (GIS) involve long-term design efforts that require the cooperation of a number of persons and departments. These design activities take place at the organizations that build things—utilities, municipal and regional governments, and departments of transportation. This white paper documents the fundamentals of versioned geodatabases and shows how they can be employed with some workflow scenarios.

Using Versions

These organizations have established workflow processes for design, construction, and maintenance. The general steps include the initial engineering design, exploration of design alternatives, selection and approval of a design, the construction of the design, and updating maps with the construction features as they have been built in the field.

When you use GIS in these workflow scenarios, it is necessary that multiple persons be able to simultaneously edit a geodatabase. They also need to have a transacted view of the geodatabase so that only the changes they or their coworkers make are visible to them. Further, the workflow structure needs to emulate the business practices of various departments in an organization.

The geodatabase data model serves these needs through a data management framework called *versioning.* This framework lets you create versions of a geodatabase for the states of a project, reconcile differences between versions, and update the master version of a geodatabase with the design as built.

Design Scenario

To illustrate how versioned geodatabases are used in a multiuser environment, follow the scenario from a water utility.

A municipal water utility keeps a comprehensive geodatabase with the current state of its field assets. All the water pipes, valves, pumps, and other components of the water system are recorded as features in a geodatabase that is updated daily.

This water utility has a number of departments that are responsible for different phases of constructing and maintaining the water system. Because of this organizational structure, this utility uses a versioned geodatabase served through ArcSDE™ software.

A versioned geodatabase has a top-level version that is always called "default." The default version of the geodatabase represents the water system in its best known as-built state. It is the starting point for creating new designs and construction activities.

Continuous Editing of the Geodatabase

The mapping department is responsible for the daily maintenance of the geodatabase. To support the new line extension, an engineer reviews field notes from that area and updates the water meter features. Another engineer adds new survey data points that were collected in a field survey in advance of the line extension. These edits are made directly to the default version of the geodatabase because they represent improved knowledge of the water system and are not part of a design cycle.

Creating Versions by Department

The information systems department is responsible for the corporate database that supports customer billing and asset management. To support the line extension, Fritz creates a new version, uses locators to match billing records to network features, and extracts and summarizes water usage data. This summarized data is intended only for this line extension project, so this version is temporary and discarded at the end of the design process.

The engineering department takes the data collected by the other departments and creates two versions for two engineering designs. Petra and Taylor create an engineering design based on using 16-inch pipe for the new main line. They simultaneously work on this version and create a proposed design. Filly creates another engineering design based on a 24-inch pipe to examine whether the increased pipe cost is offset by greater efficiency in handling present and future water usage. She discovers that the 24-inch pipe will serve projected water demand for 12 more years and that the greater initial construction cost is justified. Her design gets posted to the line extension version. When construction is complete, the line extension project version is posted into the default version.

This scenario is a simple example of how versions can be used to support a rich modeling environment for organizations that build complex systems.

Long Transactions and the Geodatabase

ArcInfo 8 is a milestone in the integration of GIS and relational database technology. GIS has now joined the mainstream of information technologies. Following are some of the benefits of storing all your geographic data in commercial relational databases.

- You can integrate geographic data with corporate or agency databases.
- You can use standard database administration tools for managing your geographic data.
- You can create very large geographic databases that can be displayed and edited quickly.
- You can deploy geodatabases on the commercial relational database of your choice.
- You can serve geographic data to a wide variety of clients, such as view-only applications, CAD applications, or Internet applications.

The geodatabase extends standard relational (and object-relational) databases to support the special requirements of representing geographic data. Following are some of the capabilities that a geodatabase adds to a relational database.

- You can represent and store geographic data in the form of raster datasets, feature datasets, TIN datasets, and location data.
- You can execute spatial and topological analysis on geographic data.
- You can perform rich cartographic display and produce high-quality maps.
- You can add intelligence to features by defining attributes, topological associations, relationships, and validation rules.
- You can enable many users to simultaneously display and edit geographic data.

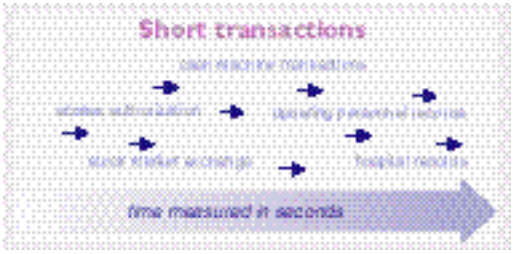
This last capability, providing multiple users with read-and-write access to a geodatabase, is called versioning and is a critical requirement for many organizations. Versioning is a key function of geodatabases served through ArcSDE.

Transactions in a Database

A central idea of relational databases is a *transaction.* Simply put, a transaction is a group of atomic data operations that comprise a complete operational task. Transactions preserve the consistency and integrity of the database by ensuring that either all or none of the atomic operations are executed for a task.

Short Transactions

When you access data in a database, you have two basic goals: that the data is accurate and that it is timely. Relational databases satisfy that requirement with *short transactions,* which represent operational tasks that can be completed in fractions of a second, or a minute or two at most. During the very brief time that a short transaction is being committed, no other updates to the affected rows are possible.



Short transactions represent most of the information tasks that people engage in, such as drawing money from an account at an automated teller machine, updating hours worked in a payroll application, or entering medical records.

Once a short transaction is committed in a relational database, it is not easy to undo that transaction or to reconstruct the state of the database at a historic point in time. There is only one state with a relational database: its status as of the most recently completed short transaction.

The short transaction model works very well for many critical applications that require instant access to a uniform view of data, but geographic data requires a longer view of updating data.

Editing Geographic Data

While a short transaction is in progress, the relational database applies locks on affected rows in database tables so that data being updated is protected from changes until the transaction is complete. When the short transaction is completed, the locks are released.

When multiple people are simultaneously editing geographic data, this type of row locking is impractical because even for short edit tasks, the locks must be held for several minutes.

Another reason that row locking is deficient for GIS is that features in a geodatabase coexist in a rich context of network connections, topological associations, and relationships.

To understand why, consider this scenario: You are editing an electric utility's database and adding lines, poles, transformers, and other devices. If another person were to edit a nearby feature while you were editing this transformer, your network could quickly fall into an inconsistent state.

Short transactions fail for this scenario because when you are adding this transformer, you are not really editing a single feature, but you are editing a larger, more complex object—the network as a whole.

Another reason that short transactions are deficient in the multiuser geographic editing environment is that you must always be able to see the current state of the database as displayed on a map. Every time someone else made a change, your system would have to redraw the map, which might take a number of seconds for a complex map. This is unacceptable.

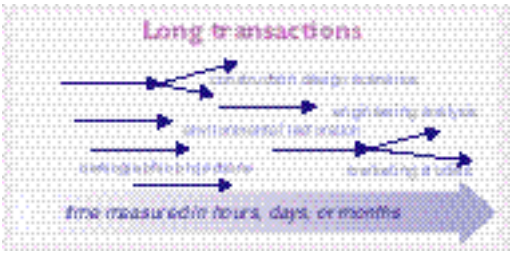
Long Transactions

To enable multiple users to edit geographic data, what you need is a transaction type that can do the following:

- Allow multiple persons to simultaneously edit the same complex system such as a network.
- Span all the edits that you need to perform on a work unit, whether it takes an hour or a month.
- Let you have a private view of your data so that no one else sees incomplete work.
- Permit you to define the scope of work to match your business's work order system.

This type of transaction is a *long transaction.* Long transactions have other uses besides representing construction work units. You can use long transactions to model any type of "what if" scenario.

J-8560



During the scope of the long transaction, you can freely add proposed features, perform geographic analysis, and produce maps—all without affecting your nominal database. When the scenario is done, you can post the changes to the database if it is built or discard it if it is not.

Concurrency Model

Long transactions implement a data management approach called *optimistic concurrency*. This means that when you start a long transaction, no locks are applied to features. The absence of locks permits the introduction of editing conflicts, but this is mitigated by an environment that makes it easy to detect, reconcile, and post these conflicts.

Optimistic concurrency is suitable for GIS applications because the volume of edits is small compared to the size of the geographic database. In real workflow practices, editing conflicts are not frequent, and the cost of reconciling conflicts is minor when compared to the savings from not having to lock or check out features for the duration of a long transaction.

Versioning is ArcInfo™ software's implementation of long transactions against central multiuser relational databases served by ArcSDE. It is an advanced data management system that lets you adopt any of a variety of workflow practices when editing geodatabases in a multiuser environment.

Versioning can be implemented on multiuser geodatabases served by ArcSDE. You cannot implement versioning on personal geodatabases.

Basic Concepts

Versioning lets multiple users directly edit a geodatabase without explicitly applying feature locks or duplicating data. The following are the essential facts about versions.

Version—A Named State of a Geodatabase

You can use versions to represent engineering designs, construction jobs, snapshots in time of geodatabases, and any type of scenario that involves the posing of "what if" questions in studying a result.

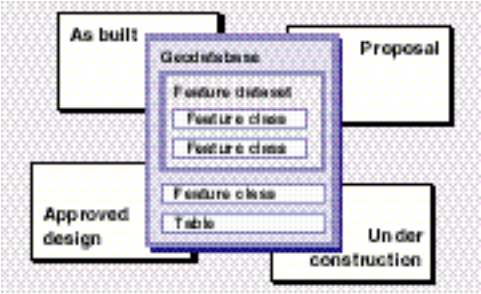
How a Version Spans a Geodatabase and Has Properties

You can define which objects in a geodatabase are versioned and can selectively specify which feature datasets, feature classes, and tables are versioned. When you specify that a feature dataset is versioned, all its tables and feature classes are automatically versioned. You can control the visibility of a version to other users by setting its permission.

A Geodatabase Having Multiple Coexisting Versions

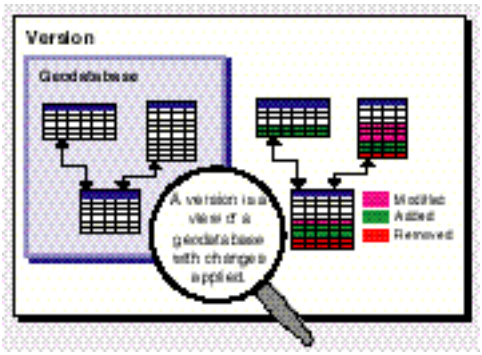
Each version lets you perform all the same display and analytic functions as a nonversioned geodatabase.

J-8560



Differences Only in Row State, Not in Schema

A version presents you with a seamless view of all the edits applied since the version was created. The row state reflects all added, removed, and modified objects. The row state information about each version is stored (or persisted) in the geodatabase. The schema, the definition of tables and their fields, can be modified on a geodatabase; schema changes are applied to all versions of the geodatabase.



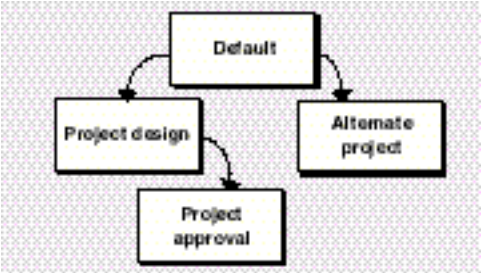
Internally, the geodatabase has tables to keep track of modified, added, and removed features for each version, but that is not apparent to you when you use versions. It appears to be an integral copy of a geodatabase.

A Default Version for Every Versioned Geodatabase

The default version can be thought of as the "as-built" version. It usually represents the nominal state of the geodatabase. The default version is the geodatabase. Most users will edit the default version.

A Version Created from Another Version

Starting with the default version, you can create any number of versions. Every version, except for the default version, has exactly one parent version. You can create a complex hierarchical version tree as appropriate for your organization's workflow requirements.



J-8560

The Capability to Connect to Any Version

A user will start editing a version based on the project or project stage they are working on. A user can work on any version for which they have been granted permission.

Why Versions Perform Well

When you start using versioning, you will notice considerable performance improvement and greater ease of use over previous data management systems, such as checked out datasets, tiled libraries of datasets, or copied datasets.

The reason that versioning works quickly and well is that versions do not require any duplication or replication of data. Internally, a versioned geodatabase uses internal identifiers and manages additional tables that record which features and objects are added, removed, or modified.

Reconciling Versions

Reconciliation is the process of merging features and objects from a target version into the current edit session. A target version can be any version in the direct ancestry of the version being edited. Reconciliation must be done before posting changes to another version.

Because a version spans all the versioned feature datasets, feature classes, and tables in a geodatabase, all objects and features in these classes will be merged into the edit session. The great majority of features and objects will pass straight through reconciliation from the target version to your edit session.

Handling Conflicts During Reconciliation

A small percentage of features and objects will have conflicts when compared between the target version and the edit session.

There are two types of conflicts.

1. When the same feature is updated in both the target version and the edit session
2. When the same feature is updated in one version and deleted in the other

For most reconciliation operations, no conflicts will be encountered. That is because at most organizations, projects and versions represent distinct geographic areas. If you and your coworkers are editing different parts of the map, it is generally not possible to introduce conflicts. Conflicts usually arise when people are editing features that are in close proximity.

When conflicts do arise, you will see an interactive conflict resolution dialog. This dialog lets you examine and zoom to any conflicts between the two versions. The conflicts are grouped by *conflict classes*, which are feature classes and tables for which conflicts are detected.

For each conflict, you can choose whether to replace the feature in your edit session with the conflict feature from the target version, keep it as it is in your edit session, or revert it to its state at the beginning of your edit session.

Posting Versions

You can post a version to the target version after you have successfully completed the reconciliation. The posting operation synchronizes the row state of your edit session with the target version. They are identical at this point.

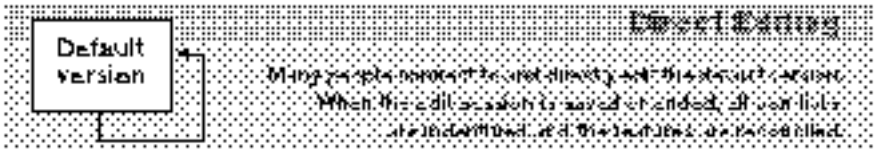
At this point, you may continue to make more edits in your edit session, but you will need to undergo the reconciliation, conflict resolution, and posting process again if you want to apply these changes to the target version. If a posting marks the end of your project, you can terminate that part of your workflow by removing the version you have been editing.

Types of Workflows

When you institute versioned geodatabases in your organization, you can select from one or several types of workflows that match your business practices. The following is a summary of the basic workflows supported by versioning. Your implementation can be one or a combination of these workflows.

Direct Editing

The simplest workflow for multiuser access on a geodatabase is for many users to directly edit the default version. As each person opens the default version for editing, a temporary version is created. The editor is not explicitly aware that a version is created and does not give it a name. Whenever the editor saves the work or ends the edit session, then that temporary version is automatically reconciled with and posted to the default version.

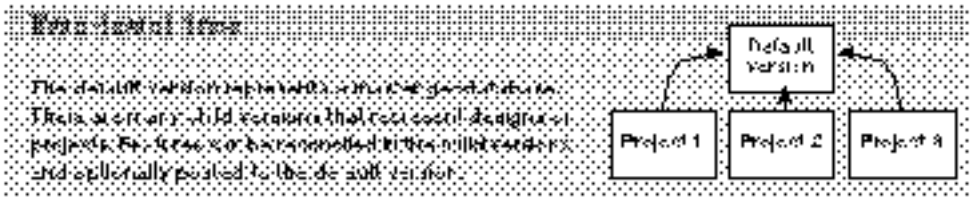


If there are conflicts, you must resolve them with the conflict resolution dialog before you can successfully save your edits. If no conflicts are detected, the edits are directly posted to the default version.

This workflow has the virtue of simplicity. It is most appropriate for situations in which the units of work are fairly modest in scale and no design alternatives have been explored or historical snapshots made.

Two-Level Tree

Many organizations employ a more structured process that tracks discrete work units of construction or maintenance. These work units typically span a time interval of days, weeks, or months and represent tasks such as adding new phone service, adding a new line extension with pipes or poles and wires, and building a new pump station or electric substation.

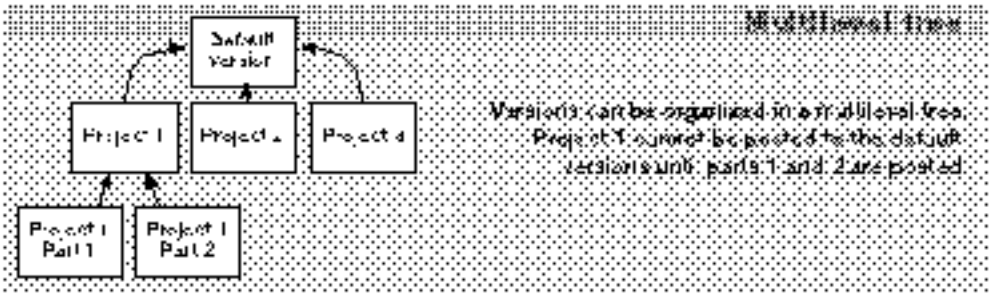


When a work order or project is initiated, a version is created. One person or several people work on this version until the design or construction is complete. At that point,

reconciliation and posting are done to merge the work order features into the default versions, and then the work order version can be removed.

Multilevel Tree

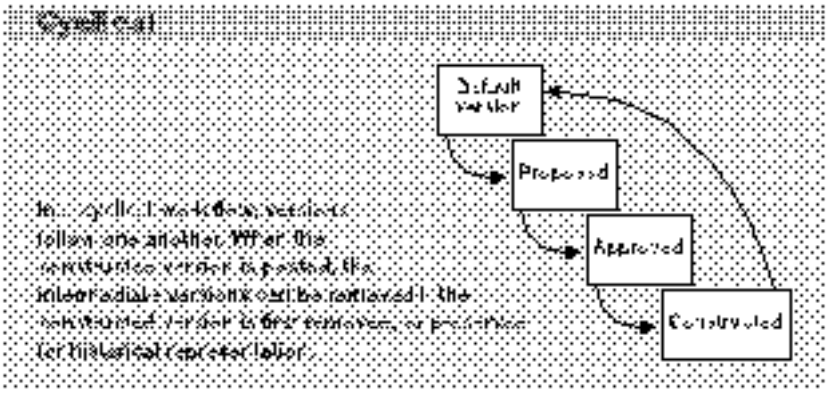
Some organizations' projects have a higher level of structure and can be subdivided into functional or geographic parts. For example, a project to design and construct a new shopping mall might have phases of construction, be subdivided into eastern and western parts, or be subdivided by construction activities such as structure, gas and water, and electric.



For larger projects with departments and teams, a multilevel version tree is an effective way to organize workflow. The teams that are working on each aspect of the project have their own version with which they can maintain a private view of their designs and then post the designs when constructed.

Cyclical

Many projects go through a prescribed or regulated set of stages that require engineering, administrative, or legal approval before proceeding to the next stage. A version represents each stage of this process. A cyclical workflow can capture the design at each stage, and when the last stage is reached and finished, the design can be posted directly to the default version, which represents the nominal state of the database.

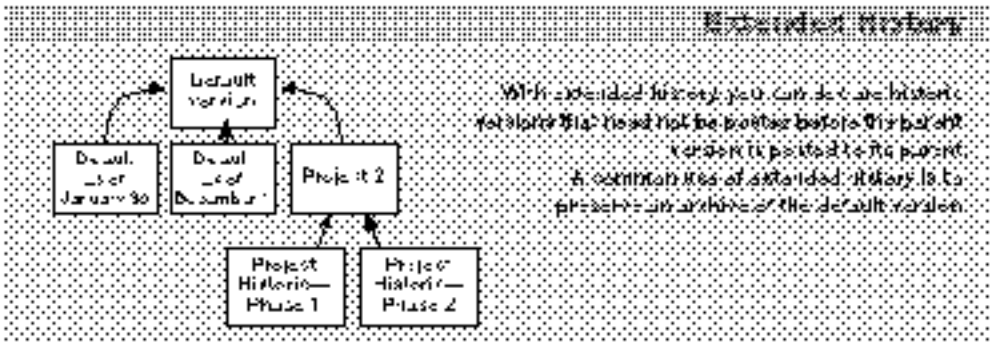


This workflow saves the effort of progressively posting changes up the version tree; you can bypass the immediate parent versions and post directly to the default or another version.

J-8560

Extended History

For some projects, it is desirable to preserve a version that reflects a historic state of a project. You can define a historic version on a project version, and when the project version is posted to its parent version, the historic version remains as a snapshot in time.



Summary

In practice, you will probably either apply the direct editing workflow or some combination of the others. An understanding of the elements of workflow management will improve the effectiveness of your geodatabase design.

For a comprehensive guide to GIS data modeling in general, and to the geodatabase model in particular, obtain the book *Modeling Our World*. It shows how to make the right decisions about modeling data—decisions that will impact every aspect of a GIS project, from database design and data capture to spatial analysis and visual presentation.