An Overview of Distributing Data
with Geodatabases

An Overview of Distributing Data with Geodatabases

An ESRI Technical Paper

Contents   **Page**

An Overview of Distributing Data with Geodatabases

Introduction

Data distribution involves creating copies of data and distributing data between two or more geodatabases. To keep the copies in sync, data distribution also includes a means to apply edits made to each copy across all copies. This paper discusses cases where you will need to distribute your data and introduces the options available. Please refer to the online documentation or software development kit (SDK) help for complete details of each option discussed.

When to Distribute Data

Data distribution can serve as a means to improve data availability and performance. Data availability refers to your ability to access data. Poor data availability can result from slow or unreliable networks. For example, field-workers may have problems accessing data in the office from the field depending on their location. It is also possible to have cases where, due to security requirements or lack of infrastructure, systems are disconnected and have no data availability. Data distribution allows you to make copies of the geodatabases such that availability is optimal.

Performance issues can arise from many users accessing the same geodatabase at once. For example, you may have a system with many editors as well as many readers accessing the data at the same time. By distributing the data, you can spread the load across several geodatabases.

The following describes some use cases where you would want to consider distributing the data in your geodatabase:

- **Mobile users:** If workers in the field can't access or are having problems accessing the geodatabase in the central office, consider using a data distribution technique. Each mobile user works with a local copy of some data from the central office. When the mobile users come into the office, they send their changes to and receive updates from the central office geodatabase.
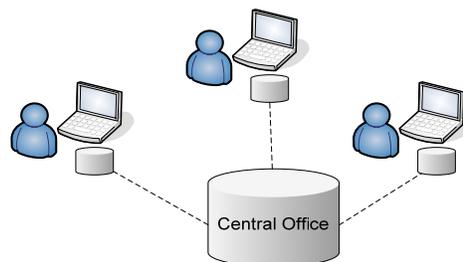
- **Multiple offices per region:** Larger organizations may have several offices located throughout a region, each requiring access to the data for that region. Data distribution can be used such that each office has its own copy for day-to-day use. A

process can also be run periodically to synchronize changes so that each office has the most up-to-date information.
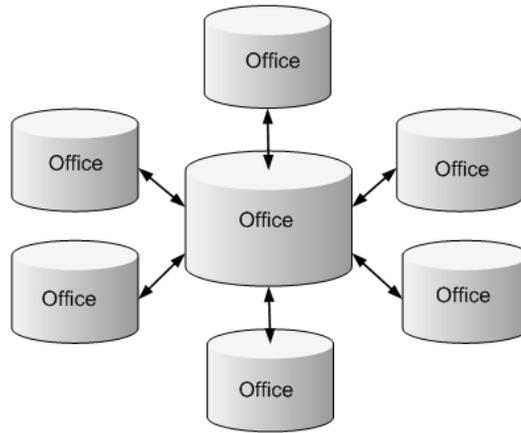
- **Multi-level organizations:** Some organizations have offices at multiple levels. For example, a company can have offices at the national level, state level, and city level. Data distribution allows each office to locally manage the data applicable to its area and also share with the levels above and below.
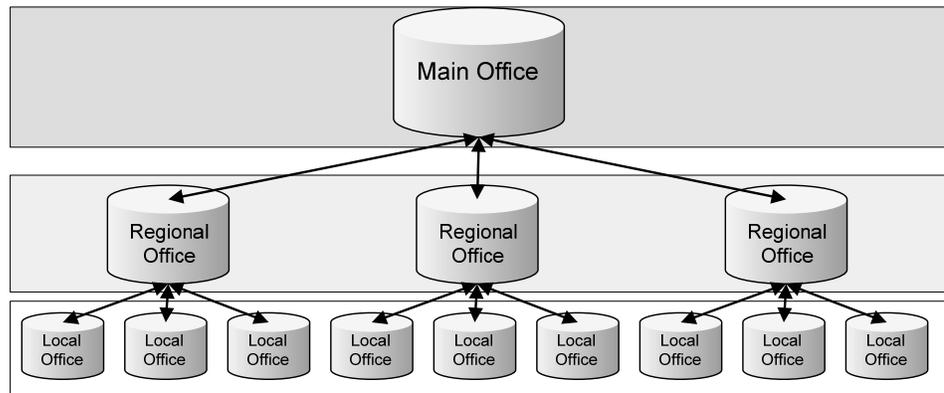
- **Working with contractors:** Some organizations need to contract out work that requires access to their geodatabase. With data distribution, a copy is used to share a portion of the geodatabase with the contractor without interrupting day-to-day activities. The contractor can then provide periodic updates that are incorporated into the company's geodatabase.
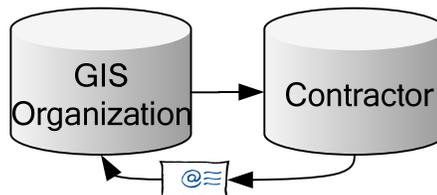
■ **Production and Publication Geodatabases:** Some systems require support for editors and support for users with read only access. Since editors and readers access the database for different reasons, the system needs to cover the requirements of each type of user. One way to build these types of system is to distribute the data across two geodatabases. The load can be divided so that editors work on one geodatabase while readers access a read-only publication copy of this geodatabase. Periodically processes can be run to update the publication geodatabase with changes from the production geodatabase.
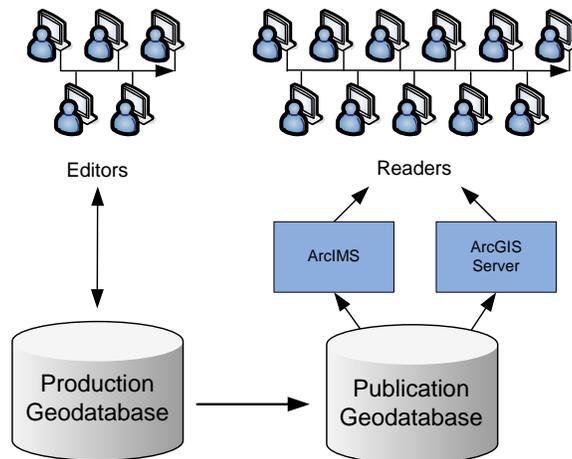
■ **Multigroup geodatabases:** Within an organization, data management may be divided among several different groups. For example, one group may be in charge of managing the utility networks while another is tasked with managing the land base data for the same area. Within these geodatabases, some of the data may also require security clearance where access is restricted. Data distribution can be used to provide each group with a copy of only the data it needs. This method prevents nonauthorized users from accessing security-sensitive data in the central database.
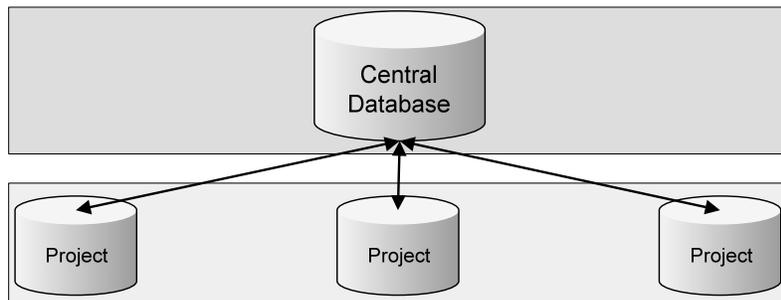
■ **Failover systems:** Some organizations require their data to always be available. Data distribution can be used to achieve this by building failover systems. If the primary geodatabase goes down, a second geodatabase—which is an exact copy of the original—can be placed online. In this way, operations on the system are not interrupted.

Data Distribution Techniques

Several different data distribution techniques are available. Deciding which to use involves considering the requirements of your system as well as the benefits and limitations of each technique. In some cases, more than one technique may be used to meet the system requirements.

The following describes each technique:

- **Geodatabase replication:** Geodatabase replication allows you to distribute data across two or more geodatabases such that edits can be made independently and synchronized periodically. It has built-in safeguards against data loss, data redundancy, and system instability. Geodatabase replication requires at least one versioned ArcSDE® geodatabase.

- **DBMS replication:** ArcSDE geodatabases are built on top of DBMSs that include technology for replicating at the database level. Geodatabases, like other applications built on top of these DBMSs, can be used with this technology. Using DBMS replication with geodatabases requires knowledge of how geodatabase data structures are implemented at the database level. ArcGIS® does not provide out-of-the-box tools for implementing these systems like it does for geodatabase replication.

- **Data copying and loading tools:** Another technique for distributing data involves simply copying data from one geodatabase to another. This technique is useful for systems with simple requirements. For example, a field-worker updates a feature class and needs to copy that feature class to the ArcSDE geodatabase in the office each night. This technique can also be used where the data is nonversioned or where only personal or file geodatabases are involved. However, it has no built-in safeguards against data loss or data redundancy.

Geodatabase Replication

Geodatabase replication is a data distribution method provided by ArcGIS. With geodatabase replication, tools are provided to distribute data across two or more geodatabases and to synchronize changes between the geodatabases.

Replica creation involves defining the data to replicate from a source geodatabase, then running a process to create replicas. The process copies data from the source geodatabase to a target geodatabase, then creates a replica in each geodatabase. The replica describes what data has been replicated and contains the information needed to synchronize changes. The replica in the source geodatabase is called the parent replica, and the replica in the target geodatabase is called the child replica. Each combination of a child and parent replica is called a replica pair. The term *relative replica* is a role-independent term that refers to the other replica in a replica pair. For example, the relative replica for the parent replica is the child, and vice versa.

Geodatabase replication allows you to replicate specific datasets within a geodatabase. It also allows you to apply filters to define which features within these datasets are involved in the replica.

Geodatabase replication is built on top of versioning. The parent replica must be hosted in an ArcSDE geodatabase, and the data involved must be versioned. Depending on the replica type, the child replica may also require an ArcSDE geodatabase and versioned data. When you create a replica, you define the version within the geodatabase to

replicate. During synchronization, edits applied to this version can be sent to the relative replica. Changes received from the relative replica are also imported into this version.

You can choose from any of the following types of replication:

■ **Checkout/Check-in replication:** Checkout/Check-in replication allows you to make edits to the child replica data, then synchronize these changes with the parent. Once the data has been synchronized, you can no longer synchronize additional edits. If additional edits are required, you must create a new checkout replica. For a checkout replica, the parent must be hosted in ArcSDE while the child can be hosted in either an ArcSDE, file, or personal geodatabase. The data in ArcSDE must be versioned for both the parent and the child.

Checkout/Check-in replication was first introduced in ArcGIS 8.3 as disconnected editing, then renamed checkout/check-in replication in ArcGIS 9.2.
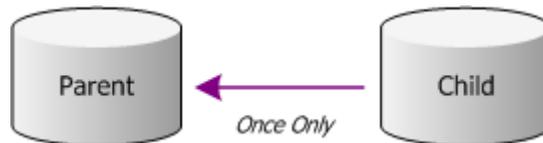
■ **One-way replication:** One-way replication allows data changes to be sent multiple times from the parent replica to the child replica. Changes cannot be sent from the child replica to the parent replica. The parent replica's data is editable, but the data in the child replica is considered read-only. One-way replication is only supported between ArcSDE geodatabases. It requires the data in the parent to be versioned but has an option that allows the data in the child to be unversioned.

■ **Two-way replication:** Two-way replication allows data changes to be sent multiple times from the parent replica to the child replica or from the child replica to the parent replica. Two-way replication is only supported between ArcSDE geodatabases. The data must be versioned in both the parent and the child.

Geodatabase replication is supported on both connected (LAN or WAN) and disconnected systems. A disconnected system is one in which the replica geodatabases are not on the same network. If disconnected, replica creation and synchronization are

accomplished through workflows involving file transport. Connected systems involve local connections on a LAN or using ArcGIS Server for remote access on a WAN.

Geodatabase replication is also fault tolerant. If the system fails during synchronization or if files are lost in transit, the replicas will remain in a consistent state. The system automatically detects when it needs to resend changes, thus safeguarding against data loss. Functionality is also built in to deal with conflicting edits or schema changes made independently to each replica geodatabase.

Geodatabase replication supports cross-DBMS replication. For example, a replica can be created between a geodatabase running on an Oracle DBMS and a geodatabase running on a SQL Server DBMS.

The ArcObjects™ API provides methods for developers to write applications involving geodatabase replication. Additional methods are also provided that allow you to distribute version differences. Using version differences, you can write code to move changes between versioned ArcSDE geodatabases without using geodatabase replication. Note, however, that the version differences approach has no safeguards against data loss or data redundancy.

Implementing Geodatabase Replication: Use Cases

In general, replica creation first involves identifying the data to replicate, then preparing the data for replication. The Preparing data for replication help topic describes this process. It is also recommended that you have a well-defined data model before creating replicas. Schema changes can be applied across replicas after replica creation, but extra work is required and not all types of schema changes are supported. See Working with schema changes for more information.

The next step involves creating replicas, then determining a synchronization strategy. The section below describes techniques for accomplishing this for each workflow.

Note that in addition to the information in this paper, it is also recommended that you review the Working with geodatabase replication help topic before implementing one of these systems.

Geodatabase Replication for Mobile Users

Mobile users running ArcEditor™ for desktop on field laptops or Tablet PCs can take advantage of geodatabase replication. Data from a geodatabase in the central office is replicated to a geodatabase (ArcSDE geodatabase for SQL Server Express, personal or file geodatabase) on the users' field laptops or Tablet PCs.

- **Creating replicas:** When determining the data to replicate for mobile users, be sure to replicate enough data for each mobile user's day-to-day needs. For example, if the mobile user works on jobs all around the city, you'll want to replicate more data than for a mobile user who always works in the same area.

When choosing a replica type, consider two-way replication. With two-way replication, you can synchronize multiple times from the field to the office or from the office to the field. The two-way replication requirement for an ArcSDE geodatabase in the field can be covered by using an ArcSDE geodatabase for SQL Server Express. Checkout replicas can also be used; this requires you to re-create the checkout replica after each synchronization.
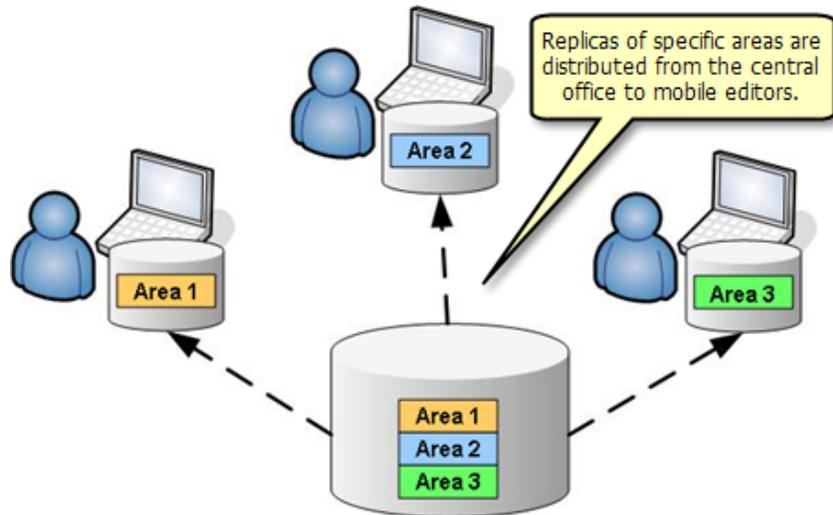
- **Synchronization strategy:** It is important to schedule synchronizations such that the changes from the field can be smoothly integrated into the geodatabase in the central office. One way to do this is by using the Synchronize Changes geoprocessing tool and the Windows scheduler to synchronize the replicas periodically. For example, if field-workers dock their laptops at the end of the day, synchronizations can run overnight. The next morning, the laptops are up-to-date and ready to go with the latest data.

The order in which the field replicas are synchronized may be important. For example, consider the case where each night several field geodatabases synchronize with the geodatabase in the central office. One strategy would be for each field geodatabase to synchronize in both directions with the central office geodatabase. The field geodatabase sends changes to the central office geodatabase, then the central office geodatabase sends changes to the field geodatabase.
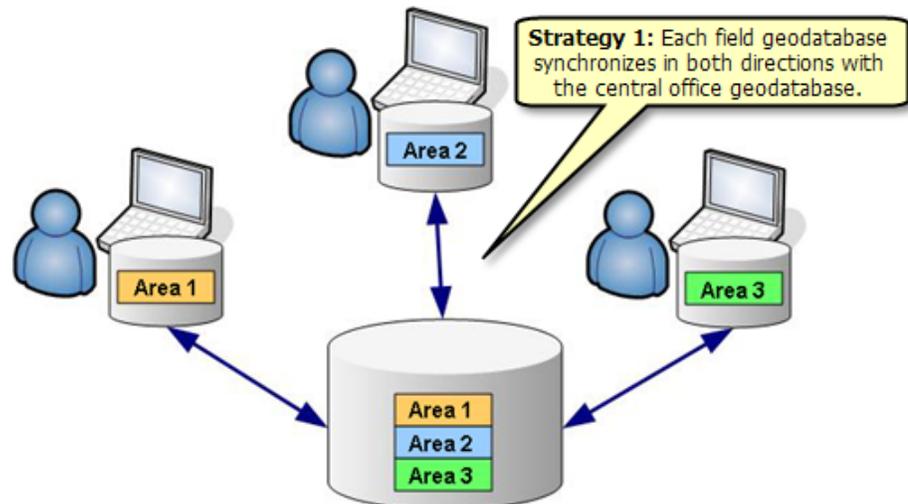
Another strategy is for each field geodatabase to first send its changes to the central office geodatabase. The central office geodatabase incorporates all the changes, then sends changes back to each field geodatabase.
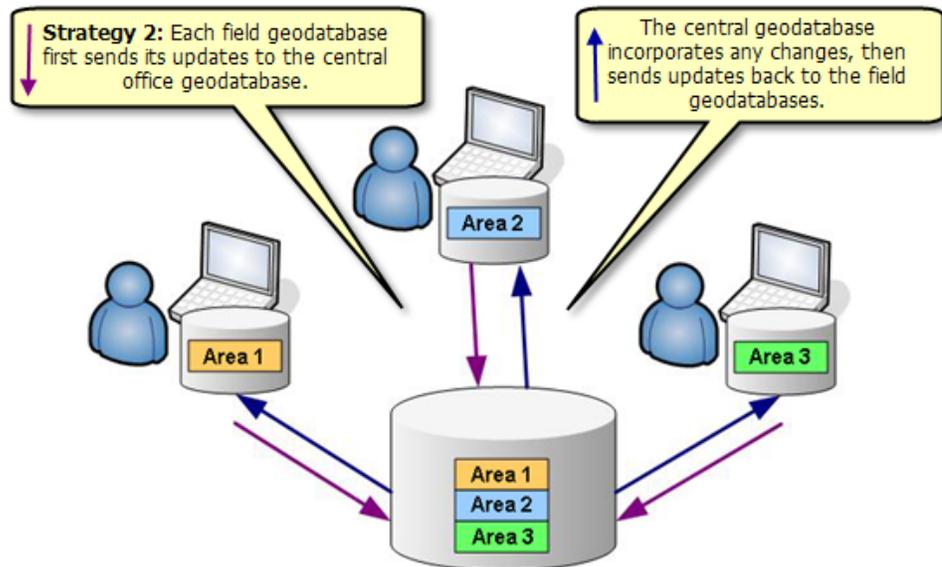
In the first case, the central office geodatabase is sending only its changes, while in the second case it is also sending changes incorporated from other replicas. Depending on the requirements of your system, one strategy may be more appropriate than the other.

Working with Conflicts

Several mobile users can replicate the same data. For example, the area covered by one field crew can overlap with the area covered by other field crews. It is also possible that changes are made in the central office between synchronizations. In these cases, conflicts can happen while synchronizing changes.

If conflicts occur during synchronization, you can choose to automatically resolve the conflicts in favor of either the incoming changes or the changes already in the receiving geodatabase. You can also choose to resolve conflicts at a later time.

If you choose to resolve the conflicts at a later time, the incoming changes are stored in a synchronization version. When you are ready to resolve the conflicts, you must connect to the synchronization version and reconcile and post into the replica version. Conflicts can be resolved manually or as part of another custom process.

Note: Geodatabase replication requires hardware capable of running software for editing geodatabases. Laptops and Tablet PCs are examples of hardware that meets these requirements and can be used in the field. If you require lighter-weight devices, however, consider using ArcGIS Mobile technology.

Multiple Offices per Region

For cases where there are multiple offices in a region, one office can serve as a hub from which replicas are created and distributed to the other offices. Once replicas are created,

each office has its own copy that it can work with directly. Offices can then share changes with one another by synchronizing with the central hub office. Synchronization involves sending updates to the central hub office while, in return, receiving the most up-to-date information.

■ **Creating replicas:** In this scenario, all offices replicate the same data from the central hub office.
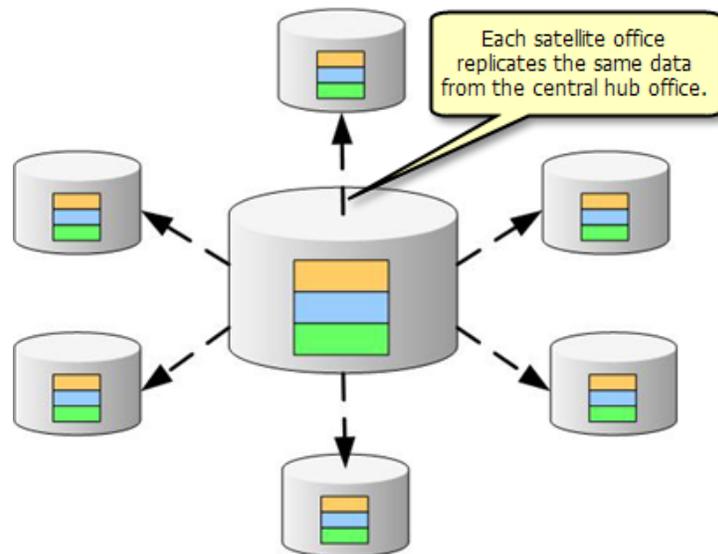

If the volume of data is not too large, create the replicas using the standard replica creation process. However, if the data volumes are large, you may want to consider using another technique for copying the data to each office.

One approach is to use the DBMS administration tools to export from the central hub office and import into each satellite office. For example, with an ArcSDE geodatabase for SQL Server Express, commands are provided to manage the geodatabase with ArcEditor or ArcInfo® for desktop. This includes a command to detach the DBMS database corresponding to a geodatabase. Once detached, the DBMS database can be copied and attached to another SQL Server Express instance. With the data copied, replicas can be created using the option to register with existing data. Using this option, replicas are created based on existing, precopied data. It is important to ensure that the values in the globalid columns match across the geodatabases. To ensure that they match, add the globalid columns to the data in the central hub office before copying to the satellite offices.

In this scenario, two-way replicas are required in order to share changes from one satellite office with the other offices.

■ **Synchronization strategy:** When working with multiple offices where each office performs local edits, it is important to devise a strategy for sharing changes.
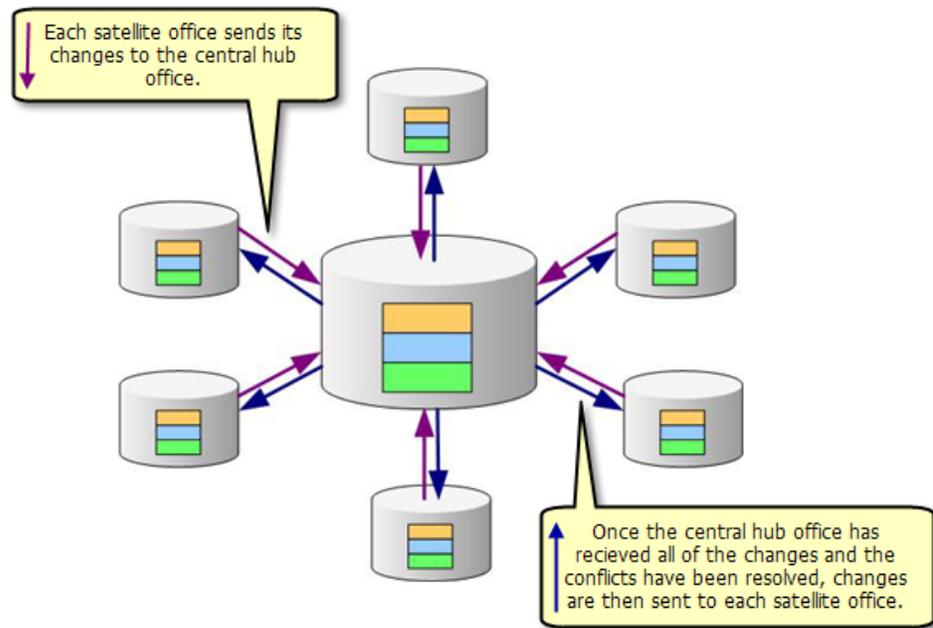
A recommended synchronization strategy is to have all the satellite offices send their changes to the central hub office on a regular basis. This can be done by scheduling the synchronizations from the satellite offices to the central hub office.

Conflicts may occur when synchronizing if each office is editing the same region of data. These conflicts can be resolved automatically during synchronization. They can also be reconciled and posted into the replica version afterward using a manual process or as part of another automated process.

Once the central hub office has received all the changes and the conflicts have been resolved, changes are then sent to each satellite office. This is done by scheduling synchronizations from the central hub office to the satellite offices. These changes are an amalgamation of all the changes from the satellite offices as well as any updates made directly to the central hub office geodatabase. After this synchronization, each office will have all the most recent updates throughout the entire region as of the date of the last synchronization to the central hub office.

Conflicts may also occur when getting changes from the central hub office. This may happen if edits have been made in the satellite office since the last time it sent changes to the central hub office. How you choose to resolve the conflicts depends on the system. For example, you may choose to resolve in favor of the satellite office since it is more up-to-date than the central hub office. You may also choose to resolve in favor of the central hub office if, in your system, the central hub office is more trusted.

Multilevel Organizations

Multilevel organizations have offices at any of several levels throughout the organization. For example, there may be an office at the main level that shares information with offices at the regional level which, in turn, share information with offices at the local level. This creates a hierarchical tree structure where data flows between the levels. The offices at each level are responsible for a subset of data. For example, each local office manages

the data for its area and shares it with the regional office in the level above it. Each regional office, in turn, shares with the main office above it.

The majority of the edits happen at the local offices, but the regional offices and main office may also make regional-level edits to the data. In this case, two-way replicas should be used.

■ **Creating replicas:** Starting with the main office, two-way replicas are created for each of the specific regions. For example, if the main office is at the national level (as seen in the graphic below), it creates multiple replicas, one for each state, and distributes to each of the regional offices at the state level only the data pertaining to that state. Likewise, the regional offices create two-way replicas for each city from its state-level data and distribute to each of the local offices only the data pertaining to that city.
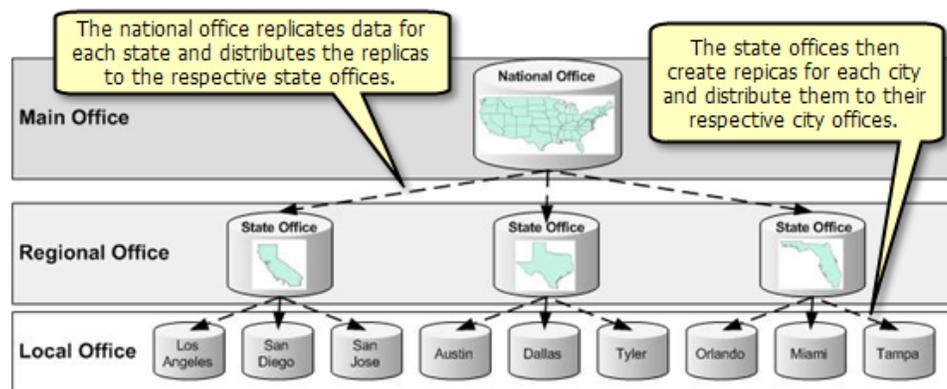
For the most part, the data involved in each replica does not overlap. Each replica contains data specific to the region. However, there may be cases where there is some overlap. For example, adjacent states may both contain features that overlap the state border. One way to approach this issue is to allow each state to replicate the overlapping features. In this case, you may get conflicts when synchronizing changes. See the Multiple Offices per Region section above for a discussion on how to handle the conflicts. Other approaches include splitting the feature at the boundary or assigning responsibility for the overlapping features to one of the replicas.

■ **Synchronization strategy:** The synchronization process for this scenario starts with the local offices sending edits to the regional offices. Since the data in each local office is normally specific to that office, each local office can synchronize independently. During synchronization, changes made in the local office are sent to the regional office, and changes in the regional office are sent to the local offices. The changes in the regional office include edits made locally and edits incorporated from the last time the regional synchronized with the main office. Once a regional office incorporates changes from all the local offices, it then synchronizes with the main office. Edits made locally in the regional office, as well as edits collected from the local offices, are passed up to the main office. Changes made in the main office are also sent down to the regional office. As with the local office level, each regional office can synchronize changes with the main office independently.
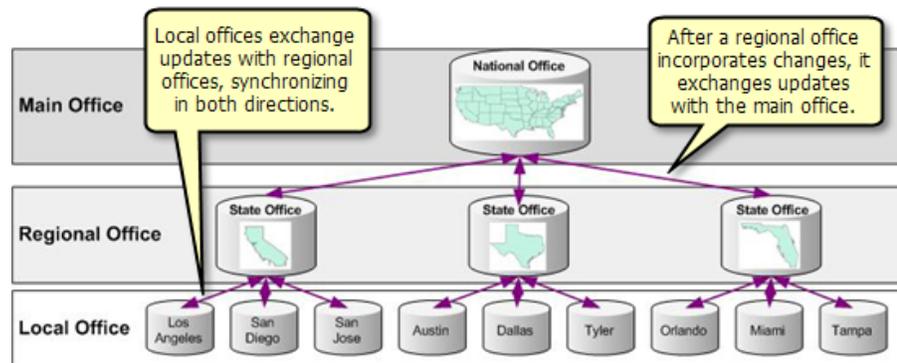
Synchronizations can be scheduled such that iterations through this synchronization pattern can be repeated on a regular basis. In some cases, you may synchronize between the regional and local offices more regularly than between the regional and main offices. The examples described above involve three levels, but this structure can be expanded to as many levels as needed.

Conflicts can occur when edits made at the upper level conflict with edits made at the lower level. In this case, you can choose a policy to favor one over the other depending on which level is more trusted.

Working with Contractors

Geodatabase replication can be used to distribute data to contractors. By replicating a portion of the data from a company's geodatabase, the contractor can work away from the main office on a local copy. As work progresses, updates are sent back to the main office.

- **Creating replicas:** Two-way replication is ideal here since it allows for changes to be sent back to the origin geodatabase multiple times. If the contractor is sending changes back one time only, checkout/check-in replication can also be used.

  Often an organization will work with versions that represent stages in the development of a project. When working with a contractor, a good practice is to replicate a specific version for the contractor's use.

- **Synchronization strategy:** As the contractor completes each stage, changes are synchronized into the replica version. Once the project is complete, changes in the replica version can be appraised, then reconciled and posted into the main working version of the geodatabase. When the project is completed, the replica can be unregistered.

J-9546
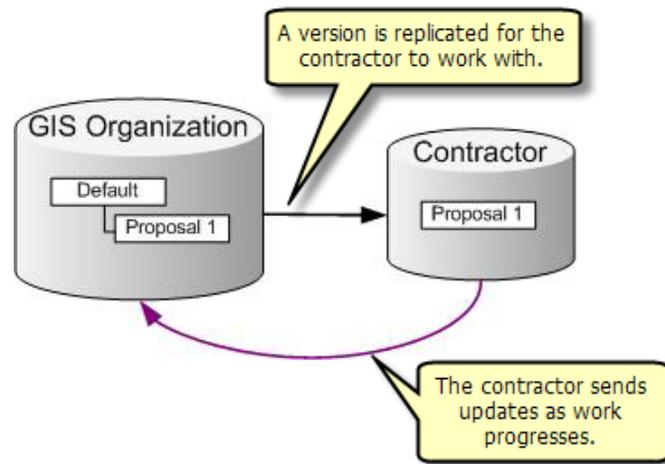
Production and Publication Geodatabases

One way to model a system is to distribute the load by configuring a geodatabase for editors and a replica geodatabase for readers. In this example, the geodatabase is replicated so that editors can work on a production geodatabase while readers view the data from a publication geodatabase. The editors edit the data on the production geodatabase, quality check it, then synchronize with the publication geodatabase, giving readers access to more up-to-date information.

- **Creating replicas:** One-way replication works well in this scenario. Since the readers won't be doing any editing, changes only need to be sent from the production geodatabase to the publication geodatabase. You can also replicate only the data that is needed by the readers. If the publication geodatabase needs only simple features, you can choose to use a simple one-way replica where the data is not versioned on the publication geodatabase.
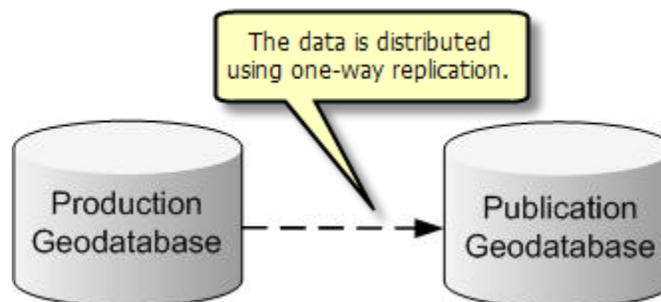
- **Synchronization strategy:** Synchronization should be done at a time when users are least likely to be accessing the data on the publication geodatabase such as overnight. With one-way replication, changes are applied directly so that conflicts do not occur. The decision to synchronize changes should come after the editor updates have gone through a quality assessment process on the production geodatabase and are deemed to be appropriate for public consumption. The parent replica on the production geodatabase can then initiate a synchronization process and send updates to the publication geodatabase.
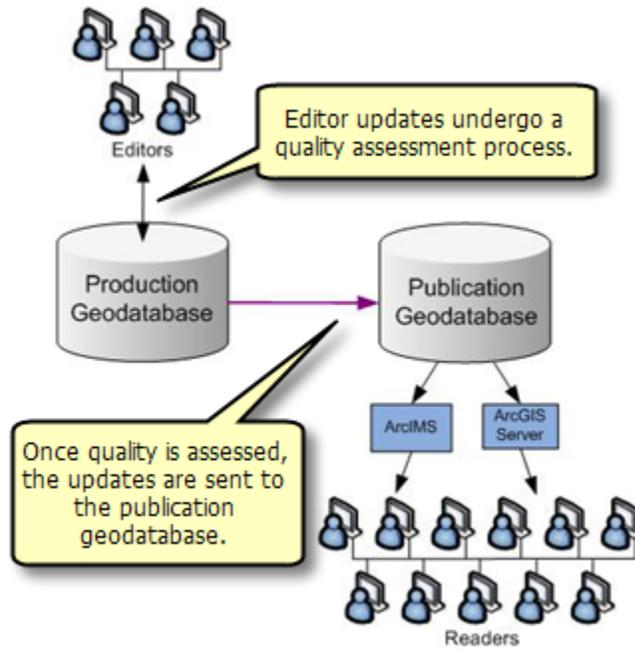
Multigroup Geodatabases

Often an organization will divide its work into multiple projects, distributing data for each project to specific work groups. Geodatabase replication can be used to accommodate this by replicating only the data that is needed for each group. The projects are independent with each group working on different datasets and synchronizing with the central geodatabase on different schedules.

Creating replicas: Since each group works on independent projects, the replicated data is different. For example, the utilities group edits the utilities data and the land base group edits the land base data. A replica is created for each group. Two-way replication is used in this scenario so that updates can be exchanged between each project group and the central database multiple times. Because of this, ArcSDE geodatabases are required at the central database as well as at each project group's geodatabase.
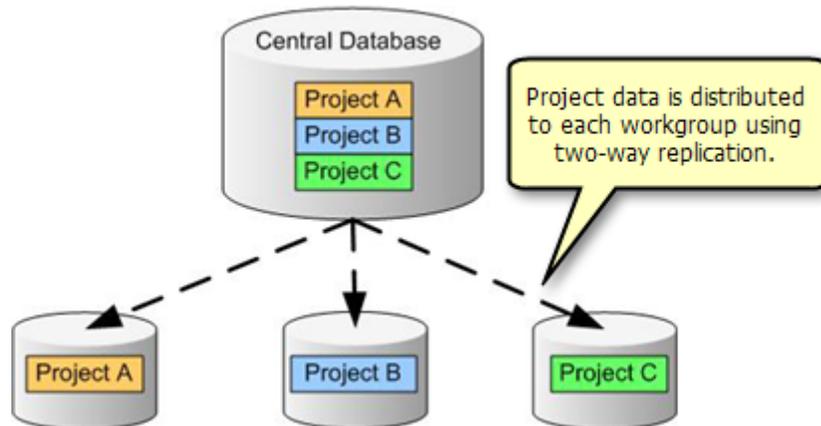
■ **Synchronization strategy:** Scheduling synchronization in this scenario is a little easier since the different groups are basically separate replication systems. Synchronization between a group and the central database can happen at any time that fits with the project's workflow. Because each group is working on separate projects, conflicts may only occur if edits are independently made on the central geodatabase.
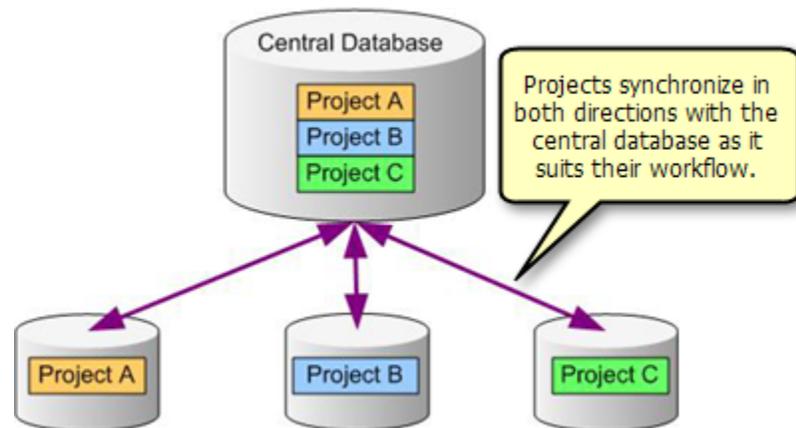
DBMS Replication

DBMS replication refers to the replication functionality provided by the DBMS on which an ArcSDE geodatabase is built. It is possible to take advantage of this technology with geodatabases, but there are limitations.

This type of replication works well for cases where you want to replicate the entire contents of a geodatabase to a read-only copy in a connected system. It can be configured to perform in synchronous mode where updates to a geodatabase are applied to a copy when changes are committed. These features make DBMS replication a good choice for the fail over system and, in some cases, the load balancing system use cases discussed above. With DBMS replication, any change—not just those made to a specific version— can be replicated.

DBMS replication, however, is not geodatabase aware. Therefore, options available through geodatabase replication, such as spatial filters, cannot be used. This also means that special logic needed for complex geodatabase types, such as geometric networks, is not applied when synchronizing changes. This may not cause problems when replicating all the data but can cause inconsistencies in the data structure when replicating a subset. Problems with object identity in the geodatabase may also arise with two-way DBMS replication. Relationships, for example, may be dropped if additional logic to preserve object identity isn't applied during the synchronization process.

Building a system with DBMS replication also requires understanding how geodatabases are implemented on top of the DBMS. There are no out-of-the-box tools included with ArcGIS for supporting DBMS replication. Work is under way to provide documentation describing case studies where DBMS replication is used with the geodatabase.

Data Copying and Loading Tools

This technique involves using copies to distribute data across geodatabases. To accomplish this, you can take advantage of the tools ESRI provides for copying and loading data into a geodatabase. The following describes how these tools can be used.

Initially copies of data are distributed from a source geodatabase to destination geodatabases using an appropriate method. For example, to copy the schema and all rows for the datasets involved, the copy and paste tools in ArcCatalog™ can be used. To schedule this event or to make this a batch process, use the copy geoprocessing tool. To copy the schema and a subset of rows or the schema only, use the Extract Data wizard in ArcMap™. If you are interested in using a published interchange format, you can use the XML workspace export and import tools.

Once created, changes can then be applied to each copy individually. To apply changes from one copy to another, you can choose to replace, append, or import as new datasets. For example, to get changes from a copy into your geodatabase, first identify the datasets with changes that you want to import. Then choose any of the following:

■ **Replace:** Before importing, first delete the corresponding datasets from your geodatabase. Then replace these datasets with those from the copy using the same copy methods that were used to initially create the copies.

■ **Append:** To append, use the Data loader tools in either ArcCatalog or ArcMap to append the rows from the copy geodatabase into geodatabase. Tools are provided to load records directly or using an XML recordset document, which is in a public interchange format.

■ **Import as new datasets:** You can also copy the datasets into your geodatabase as new datasets and leave your original copy unaltered.

A disadvantage to this technique is that it can result in data loss or data redundancy. Replacing the data can result in data loss where local changes are removed before import. Importing as new datasets or appending can result in data redundancy where there are multiple copies of the same data.

Applications can be written to make comparisons before importing to try to prevent data loss and data redundancy. However, if this is a requirement, it is recommended that you look into using geodatabase replication, which has this functionality built in.

Heterogeneous Systems

Heterogeneous systems involve distributing data across several different data storage formats. The data distribution techniques discussed to this point have involved systems that use only geodatabases as the data storage format.

ESRI provides technology for distributing data between geodatabases and nongeodatabase ESRI formats (e.g., shapefiles). Examples include the ArcPad® checkout and check-in solution as well as the ArcGIS Mobile technology for developers. Geoprocessing tools and desktop commands are also available to export and import between geodatabase and nongeodatabase ESRI formats.

In addition, there are techniques that you can use to distribute data between geodatabases and non-ESRI formats. These techniques involve using open XML specifications

published by ESRI and the ArcGIS Data Interoperability extension to transform data between formats. Since the XML format is published, you can also build your own tools to support your specific requirements. The Geodatabase XML specifications include workspace, recordset, and updategram documents.

Another option is to use third-party products, which are also available for working with these heterogeneous systems.

Summary   Data distribution can be used in a number of cases to improve the data availability and performance of your geodatabase. ArcGIS provides a number of tools for supporting data distribution techniques, each having advantages and disadvantages. The techniques discussed include the following:

- **Geodatabase replication:** Geodatabase replication contains out-of-the-box tools to support data distribution with geodatabases. It has built-in safeguards against data redundancy and data loss. Geodatabase replication is designed to work with versioned data in ArcSDE geodatabases.

- **DBMS replication:** It is possible to use the tools provided by the DBMS to replicate geodatabases. However, since the DBMS is not geodatabase aware, problems may arise when replicating geodatabase data structures. In addition, there are no out-of-the-box tools provided in ArcGIS for building these systems. This limits the use cases that can be implemented using DBMS replication.

- **Data copying and loading tools:** Systems built using these tools are simple and flexible. However, they do not have built-in safeguards against data loss or data redundancy.

- **Heterogeneous systems:** Distributing data across a heterogeneous system requires translating between the storage formats. ESRI provides tools and open formats that can be used for these purposes.