Choose a tool for replication

You can use seven different tools for replicating and synchronizing databases:

- Briefcase replication is a convenient tool for keeping a database file up to date. For example, you can take a database file that resides on your office computer and drag it onto the **My Briefcase** icon on your laptop, which creates a replica. For more information, click ».

- Replication commands on the Microsoft Access **Tools** menu in a database and in a project allow you to create replicas and synchronize them on demand as you work in Microsoft Access. For more information about replication in a database, click ». For more information about replication in a project, click ».

- Data Access Objects (DAO) and Jet and Replication Objects (JRO) include methods and properties developers can use to programmatically replicate and synchronize databases and projects. For more information about DAO, click ». For more information about JRO, click ».

- Microsoft Replication Manager is a full-featured tool for managing replicas, setting synchronization schedules, and viewing members of the replica set. It is available to owners of the Microsoft Office 2000 Developer. For more information, click ».

- Web server replication is a tool you can use to work on files even when you are no longer connected to a Web server. For more information, click ».

Microsoft Replication Manager

You can use Microsoft Replication Manager to:

- Manage a large number of replicas.
- Support laptop users who aren't always connected to a network.
- Easily create replicas of more than one database.
- Set schedules for synchronizing members of the replica set.
- Troubleshoot errors.

Replication Manager provides a visual interface for designating Design Masters, creating replicas, viewing the connections between members of a replica set, and setting the properties for replicas.

Note If you move or copy a global replica that is a hub replica for local or anonymous replicas by using any tool other than the **Move Replica** command in Replication Manager, you will not be able to synchronize the global replica to the local or anonymous replicas in the replica set. And the local or anonymous replicas will not be able to synchronize with the global replica that created them.

What's new in replication?

Replication commands on the Microsoft Access **Tools** menu in a Microsoft Access database (.mdb) and in a Microsoft Access project (.adp) allow you to create replicas and synchronize them on demand as you work in Microsoft Access.

- **Replication in an Access project** Replicate data in an Access project by creating publications and subscriptions.

- **Jet and Replication Objects (JRO)** JRO includes methods and properties developers can use to programmatically replicate and synchronize databases and Access projects.

- **Web server replication** Work on files even when you are no longer connected to a Web server.

- **Replica priority** A replica in a replica set is assigned a priority when it is created. The highest priority replica wins in the case of a synchronization conflict.

- **Prevent deletes option** Prevent users from deleting records in a replica.

- **Local and anonymous replicas** Create local replicas and anonymous replicas. Local and anonymous replicas can synchronize only with their parent, global replica. Anonymous replicas are recommended for Internet applications if you need a large number of replicas.

- **Conflict Viewer** The **Conflict Viewer** is the default tool in Access to reconcile and resolve synchronization conflicts.

- **Row Level Tracking property** When this table property is set to **True**, it indicates that conflicts are tracked based at the row-level of a table. Conflicts are tracked at the column-level by default.

- **ConflictFunction property** This property is used to replace the Microsoft

Access Conflict Viewer with a customized procedure that assists users in resolving synchronization conflicts.

- **Retention Period** Controls the number of days nonsynchronized records are retained in the system tables.

- **Synchronization conflicts** A single mechanism is now used to record and resolve conflicts and errors, making it easier to resolve conflicts. Whenever a conflict occurs, a winning change is selected and applied in all replicas and the losing change is recorded as a conflict at all replicas.

- **Column-level conflict resolution** Conflicts are determined at the column level. Changes to the same record in two different replicas cause a synchronization conflict only if the same column or field is changed.

Troubleshoot replication problems in a database

Why has my table or query been named something else?

When I change data in one member of the replica set, it doesn't change in another member.

When I change the design of an object in the Design Master, it doesn't change in any of the replicas.

I can't find the Design Master.

I can't open the Design Master.

I can't make my replica the Design Master.

There are two Design Masters in the replica set.

I want to use a different program for resolving synchronization conflicts.

I don't want to use replication when I drag a database file into My Briefcase. How do I stop Briefcase replication?

Make a replica the new Design Master

1. Make sure that neither the replica nor the current Design Master is in use elsewhere.
2. Open the replica that you want to make the new Design Master.
3. On the **Tools** menu, point to **Replication**, and then click **Synchronize Now**.
4. Select the current Design Master as the member to synchronize with.
5. Check the **Make *'Filename'* The Design Master** check box.
6. Click **OK**.

Recover the Design Master

You can recover a Design Master if the file has been lost, damaged, or moved. However, this option is unavailable if the currently opened replica is a partial replica or if the Design Master is moved with a tool other than Replication Manager. For information on moving a Design Master, click ».

1. Make sure that there is no Design Master in the replica set.

2. Open the replica that you want to make the new Design Master.

3. On the **Tools** menu, point to **Replication**, and then click **Synchronize Now**.

4. Click a replica in the list, and then click **OK**.

5. Repeat steps 2 through 4 as many times as necessary to synchronize with each remaining replica in the replica set.

   Important Make sure that you synchronize with all other replicas in the replica set.

6. On the **Tools** menu, point to **Replication**, and then click **Recover Design Master**.

7. Click **Yes**.

Create a replica of your database

1. Open the database you want to replicate. If you are working in a multiuser environment, make sure that all other users have closed the database.

   Important   Before replicating your database, it is recommended that you read about the changes that are made to a database when it's replicated. To read about these changes, click ».

2. If your database is protected by a database password, remove the password.

   How?

3. On the **Tools** menu, point to **Replication**, and then click **Create Replica**.

4. Click **Yes** when you are asked whether you want to close the database.

5. Click either **Yes** or **No** (**Yes** is recommended) when you are asked whether you want to make a backup of the database (this dialog box is displayed only if the database must first be converted to a Design Master before the first replica is created).

6. In the **Location of New Replica** dialog box, select the location where you want to place the new replica.

7. In the **Location of New Replica** dialog box, click the **Priority** button, type the priority for the replica, and click **OK**. For more information about setting replica priority, click ».

8. In the **Location of New Replica** dialog box, check the **Prevent deletes**

check box if you do not want users to delete records in the replica.

9. In the **Location of New Replica** dialog box, select the replica visibility from the **Save as type** drop-down list box. For more information about replica visibility, click ».

10. Click **OK**.

Notes

- When you replicate a database, all the tables, queries, forms, reports, macros, modules are replicated by default, as well as the data. For more information on how to keep your forms reports, macros, and modules local, click ».

- For other considerations when replicating a Microsoft Access project, click ».

Create a replica with Briefcase

1. Make sure that Briefcase is installed on your computer. If Briefcase is installed, you will see a **My Briefcase** icon on the Windows desktop.

   If Briefcase isn't installed, you must do both of the following two steps:

   - Click the Windows **Start** button, point to **Settings**, click **Control Panel**, click **Add/Remove Programs**, click the **Windows Setup** tab, and then double-click **Accessories**. Then follow the instructions on the screen.

   - To install Briefcase replication, run the Microsoft Access Setup program. For more information, click ».

2. If your database is protected by a database password, remove the password.

   How?

3. From Windows Explorer, select the folder where the Microsoft Access database is located.

4. Drag the database file to the **My Briefcase** icon on the Windows desktop.

   Important Before replicating your database, you should read about the changes that are made to a database when it's replicated. To read about these changes, click ».

Remove a replica from the replica set

1. From Windows Explorer, find the replica you want to delete.

2. Click the replica that you want to delete, and press the DEL key.

3. From another replica set member, attempt to synchronize with the deleted member. (This procedure broadcasts the deletion to the other members.)

Caution Do not delete the Design Master for the replica set unless the Design Master is no longer functioning. Deleting the Design Master prevents you from changing the design of the database. Moreover, it's a good idea to check a replica for existing conflicts. Otherwise, the record of any existing conflicts will be lost when you delete (or move) the replica.

A comparison of global, local, and anonymous replica visibility

There are three types of replica visibility that are available in Microsoft Access: global, local, and anonymous.

This topic provides reference information about:

Replica use

Disconnected environment on the Internet

Custom application on the Internet

Replica creation

Use as a Design Master

Database replica visibility in Access 95 and 97

Conflict resolution

Synchronization

Scheduling

Briefcase support

Replica use

• Global	Serves as a hub in a replica set
• Local	Control topology
• Anonymous	Control topology/Internet

Return to top

Disconnected environment on the Internet

• Global	Yes
• Local	Yes
• Anonymous	Yes. Anonymous replicas are recommended for Internet applications if you need a large number of replicas.

Return to top

Custom applications on the Internet

• Global	Yes
• Local	Yes
• Anonymous	Yes. Use the Internet as a distribution mechanism for a large number of replicas.

Return to top

Replica creation

• Global	Yes. Create a global, local, or anonymous replica.
• Local	Yes. Create new local replicas.
• Anonymous	Yes. Create new anonymous replicas.

Return to top

Use as a Design Master

• Global	Supported. When a database is replicated, the first replica that is created (the Design Master) is a global replica.
• Local	Not supported
• Anonymous	Not supported

Return to top

Database replica visibility in Access 95 and 97

Global	Default
Local	Not applicable
Anonymous	Not applicable

Return to top

Conflict resolution

• Global	Conflict resolution can be managed at any global replica.
• Local	All conflicts are recorded and resolved at the hub (global) replica.
• Anonymous	All conflicts are recorded and resolved at the hub (global) replica.

Return to top

Synchronization

• Global	Synchronize with any global replica in the replica set, or any local or anonymous replica that the set created.
• Local	Synchronizes only with its hub (global) replica.
• Anonymous	Synchronizes only with its hub (global) replica.

Return to top

Scheduling

- Global

  Replication Manager can schedule synchronization with any global or local (child) replica. Synchronization cannot be scheduled to anonymous replicas.

- Local

  Hub replicas can schedule an exchange to a local replica. And local replicas can schedule synchronization with their hub (global) replica.

- Anonymous

  Anonymous replicas can schedule synchronization with their hub (global) replica.

Return to top

Briefcase support

Global	Yes
Local	No
Anonymous	No

Return to top

About setting replica priority

Replica priority affects how changes to records are resolved when there is a synchronization conflict.

This topic provides reference information about:

Priority differences in Access

Determining the replica priority

Simple case

Complex case

More information

Priority differences in Access

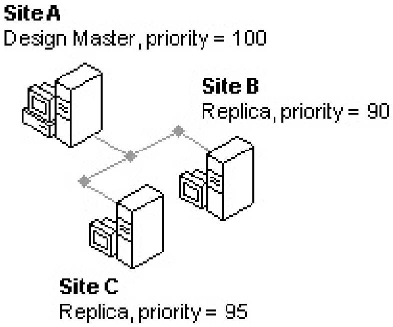
In Microsoft Access 95 and 97, synchronization conflicts were resolved by using the rule in which the winning record is the one changed most often. In Access 2000, a replica in a replica set is assigned a priority when it is created, and the highest priority replica wins in the case of a synchronization conflict. Where priorities are equal, the replica with the lowest **ReplicaID** property wins. Learn about properties that are added to your database when you replicate it.

Return to top

Determining the replica priority

Every replica is assigned a priority number between 0 and 100, with 100 as the highest priority. When a database is made replicable, the replica’s default priority is set to 90. Subsequent replicas have a default priority equal to 90 percent of the hub's priority. Priorities for local and anonymous replicas are always 0. Local and anonymous replicas automatically lose if their changes conflict with their global hub replica. If a local or anonymous replica sends a non-conflicting change to the hub, the hub assumes ownership of the change.

Access carries the historical priority for each record in a replica set. During synchronization, the historical priority setting is evaluated and the record with the highest priority wins in any conflicts. To understand the priority scheme, consider the example of one Design Master, Replica A, with a priority of 100. Replica A replicates to two replicas, Replica B, which has a priority of 90, and Replica C, which has a priority of 95.



Return to top

Simple case

The highest priority change wins. Suppose that all three sites agree initially that Replica A created version one of the record, and no subsequent updates have occurred. If Replicas A and B update the record simultaneously, then Replica A’s update is the conflict winner because it has the highest priority.

Return to top

Complex case

If multiple changes occurred to the same record after the last synchronization, then the replica that has changes with the highest historical priority is used to determine the conflict winner. For example, suppose that Replica A makes version two, and then sends it on to Replica B. Replica B makes version three, and then sends it back to Replica A, where it is merged. Then suppose that Replica C has also made a version two, and reconciles with Replica A. Choosing the highest historical priority of changes that occurred after the changes to the original record (Replica A with a replica priority 100 and 95 for the version of the row that Replica C has), Replica A and B’s joint changes are the priority winner when compared to the change made by Replica C. However, because Replica A was created before Replica B, Replica A is the conflict winner. One benefit of this decision rule is that no changes made at the highest priority replica ever get reversed by the conflict resolution mechanism.

Return to top

More information

About synchronizing data in your replica set

Create a replica of your database

Return to top

About the replica set retention period setting

The replica set retention period setting controls the number of days non-synchronized records are retained in system tables. The retention period is established when the database is initially made replicable. If you replicate the database by using Replication Manager, Data Access Objects (DAO), or Jet and Replication Objects (JRO), the default retention period is 60 days. If you replicate the database by using Microsoft Access or Briefcase, the default retention period is 1000 days.

The retention period can be changed in a Design Master by using Replication Manager or through the JRO **RetentionPeriod** property. The retention period must be between 5 and 32,000 days. The replica set should have a large retention period if the replicas do not synchronize frequently. However, if the replicas synchronize frequently and you want to keep the replica size small, specify a shorter retention period.

A replica or Design Master must receive, as well as send, data and schema within the retention period. When you open a replica in Microsoft Access or Replication Manager that expires within 5 days, a message reminds you that the replica expires soon. If the retention period expires for a replica, Replication Manager and Access will not synchronize changes between the expired replica and the other replicas in the replica set. If a replica does not synchronize with another replica in the set within the retention period, the next time you attempt to synchronize the replica, it gets removed from the replica set. If the Design Master expires, you must recover the design master to change an existing or new replica into the design master.

More information

Create a replica of your database with Access

Create a replica with Briefcase

Microsoft Replication Manager

Replication with Data Access Objects (DAO)

Replication with Jet and Replication Objects (JRO)

About synchronizing data in your replica set

A comparison of row and column level conflict resolution

In Microsoft Access 95 and 97, conflicts are determined at the row-level. Conflicts will occur whenever the same record is changed in two separate replicas, even if different fields are changed at each replica. For example, changing a customer's phone number in one replica and that same customer's zip code in another replica will cause a conflict.

In Access 2000, conflicts are determined at the column-level. Changes to the same record in two different replicas cause a synchronization conflict only if the same column, or field, is changed. So in the previous example, there would no longer be a synchronization conflict since the two users changed different fields.

Column-level conflict resolution is the default when a database is made replicable. To specify row-level conflict resolution for a table, you must set it prior to making the table replicable.

You can set the conflict tracking on a database or an individual table prior to making it replicable. Once you create a replica, you can’t change the setting. However, you can change the value of individual tables to row-level tracking by setting the table **RowLevelTracking** property.

Note By using Jet and Replication Objects (JRO) you can only specify the default tracking for the database, not for individual tables. When making a database replicable, you can set **ColumnLevelTracking** to either **True** or **False** by using the JRO **MakeReplicable** method.

More information

Create a replica of your database

Replication with Jet and Replication Objects (JRO)

About replication in a Microsoft Access project

About synchronization conflicts

Conflicts can occur when data changes are made at two or more replicas.

This topic provides reference information about:

Conflict behavior differences in Access

Simultaneous update conflict

Unique key conflict

Table level validation conflict

Update referential integrity constraint conflict

Delete referential integrity constraint conflict

Locking conflict

Foreign key violation conflict

Case sensitivity in replica sets

Conflicts in partial replicas

Resolving conflicts

Conflict behavior differences in Access

In Microsoft Access 95 and 97, there were synchronization conflicts and synchronization errors. Synchronization conflicts occurred when two users updated the same record in two different databases in a replica set. Synchronizing the two databases would succeed, but only one of the two sets of changes would be applied to both databases. Synchronization errors occurred when a change to data in one database in a replica set could not be applied to another database in the replica set because it would violate referential integrity or some other constraint.

In Access 2000, the events that cause synchronization conflicts and errors are both viewed simply as synchronization conflicts. A single mechanism is used to record and resolve the conflicts, making the process much easier. Whenever a conflict occurs, a winning change is selected and applied in all replicas, and the losing change is recorded as a conflict at all replicas. The **Conflict Viewer** ,the default tool in Access, is used to reconcile and resolve synchronization conflicts.

Return to top

Simultaneous update conflict

A simultaneous update conflict occurs when two replicas both update the same record. The losing record is logged in the conflict table. Learn about tracking changes to your data.

Return to top

Unique key conflict

A unique key conflict can occur in one of the following ways:

- Two replicas both enter a new record with the same value in a field that is a primary key or has an unique index.

- The Design Master creates a unique index, and a replica simultaneously adds two or more records with the same key value. When the schema change reaches the first replica with duplicate key records, then the first duplicate key record remains in the base table. Subsequent records are marked for deletion and are written to the conflict table.

Return to top

Table-level validation conflict

A table-level validation conflict occurs when data that breaks a table-level validation rule that restricts the values or types of data that can be entered into a table is entered. If you add a table-level validation rule to the Design Master without determining if any existing data violates the rule, you may encounter a conflict when you synchronize the design changes to the rest of the replica set. In this case, the records are deleted and logged in the conflict table.

Return to top

Update referential integrity conflict

An update referential integrity conflict occurs when the primary key is updated at one replica and new child records that reference the original primary key value are added at a different replica.

Return to top

Delete referential integrity conflict

A delete referential integrity conflict occurs when a primary key record is deleted in one replica while new child records that reference the deleted primary key are added at a second replica. When the two replicas are synchronized, the new child records are marked for deletion and added to the conflict table.

Return to top

Locking conflict

A locking conflict occurs when the record cannot be applied during synchronization because another user locked the table. Microsoft Access will try to update the record several times, but if unsuccessful, the synchronization will stop, and the entire transaction will return to its original status. An error is returned but no conflict is logged.

Return to top

Foreign key violation conflict

A foreign key violation conflict occurs when there is an invalid primary key record. This could be caused by any of the other conflict types.

Return to top

Case-sensitivity in replica sets

The number of conflicts can increase in replica sets that reside across multiple database types — Access and Microsoft SQL Server for instance — if language sort orders or the case-sensitivity of the sorting is different. This is because unique key values in one database may not be unique in the database with different sorting. The problem can affect both indexed text data and metadata, but it is not an issue for non-indexed text data. Creating replicas with different sorting will require special care to ensure that metadata and indexed text will always be unique for both methods of sorting.

Return to top

Conflicts in partial replicas

A partial replica receives conflicts associated with all of the rows in the partial replica, even for those rows that are added to the partial replica during synchronization.

Return to top

Resolving conflicts

If the same record in a replicated database is changed in one or more replicas, conflicts result when you synchronize a replica with the replica set. When you choose to resolve these conflicts, Access calls the **Conflict Viewer**. The wizard presents each conflict, and you must manually determine which changed record contains the correct information. How to resolve synchronization conflicts between replica set members.

Return to top

About synchronizing data in your replica set

In Microsoft Access, Replication Manager, and Jet and Replication Objects (JRO), you can choose between three different methods to synchronize your data: direct, indirect and Internet.

This topic provides reference information about:

Direct synchronization

Indirect synchronization

Internet synchronization

Moving and copying global replicas

More information

Direct synchronization

Direct synchronization is an excellent solution when replicas are connected directly to the local area network and are available through shared network folders. Direct synchronization is not the best solution when remote synchronizing over a Remote Access Service (RAS) or dial-up connection is required. In this case you should use indirect or Internet synchronization.

If you attempt to do a direct synchronization to a replica that’s in the list of known replicas, and that replica isn’t found, that replica will then be removed from the replica set.

Return to top

Indirect synchronization

Indirect synchronization is useful when you work in a disconnected environment, such as when you are travelling with a laptop.

Indirect synchronization can only be configured by using Replication Manager, which is available through the Microsoft Office 2000 Developer (MOD). Once indirect synchronization is configured with Replication Manager, you can execute an indirect synchronization by using Microsoft Access, Replication Manager, or JRO.

Return to top

Internet Synchronization

Internet synchronization is a convenient way to synchronize replicas in a disconnected environment in which an Internet Server is configured.

You must use Replication Manager, which is available through the Microsoft Office 2000 Developer (MOD) to configure Internet synchronization.

Return to top

Moving and copying global replicas

If you move or copy a global replica that is a hub replica for local or anonymous replicas using any tool other than the **Move Replica** command in Replication Manager, you will not be able to synchronize the global replica to the local or anonymous replicas in the replica set. And the local or anonymous replicas will not be able to synchronize with their parent global replica.

Return to top

More information

Microsoft Replication Manager

A comparison of replica visibility

About synchronization conflicts

Synchronize with another replica set member

About replication in a Microsoft Access project

Return to top

Resolve synchronization conflicts between replica set members

Conflicts can occur if data has changed in the same record of two replica set members that have been synchronized.

1. Start the Microsoft Replication **Conflict Viewer** by using one of the following steps:
   - Open the database that you want to check for conflicts. If there are conflicts, you are asked whether you want to resolve them. Click **Yes** to proceed.
   - On the **Tools** menu, point to **Replication**, and then click **Resolve Conflicts**.
2. In the **Conflict Viewer**, select a table that has conflicts from the list, and click **View**.
3. For each conflict shown in the **Conflict Viewer**, choose one of the following options.
   - To keep the data in Replica 1, click **Keep existing data**.
   - To modify the data in Replica 1 and copy it to Replica 2 on the next synchronization, click **Keep revised data**.
   - To keep the data in Replica 2 and overwrite the data in Replica 1 on the next synchronization, click **Overwrite with conflicting data**.
   - To modify the data in Replica 2 and overwrite the data in Replica 1 on the next synchronization, click **Overwrite with revised data**.
4. Click **Resolve**.
5. Repeat steps 3 and 4 as many times as necessary to resolve each conflict in the table.

Notes

- The record of conflicts can be lost if you delete or move a replica. Before deleting or moving a replica, check it for conflicts.

- For more information about synchronization conflicts, click ».

Synchronize a replica by using Briefcase

Synchronization is the process of updating two members of a replica set by exchanging all updated records and objects in each member. Two members are synchronized when the changes in each member have been applied to the other member.

1. Click the **My Briefcase** icon, and then click the database file.
2. On the **Briefcase** menu, click **Update Selection**.
3. Click **Update** to have Briefcase exchange the updates.

Note Briefcase does not support local or anonymous replicas.

Synchronize with another replica set member in a database

Synchronization is the process of updating two members of a replica set by exchanging all updated records and objects in each member. Two replica set members are synchronized when the changes in each have been applied to the other.

1. Open the replica set member you want to synchronize.

2. On the **Tools** menu, point to **Replication**, and then click **Synchronize Now**. (If a Synchronizer on the Internet manages any replica set member, you will see that the managed replica appears in the **Directly with Replica** drop-down list box as http://*ServerName*, where *ServerName* is the name of the Internet Server.

3. Enter the path and file name of the replica set member you want to synchronize with the current database.

4. If your database is the Design Master and you want to make the other replica the Design Master, select the **Make *'Filename'* The Design Master** check box. (This option is unavailable if either of the replicas to be synchronized is a partial replica.)

5. Click **OK**.

6. When asked whether you want to close and reopen the database, click **Yes**. (If you click **No**, you might not see all of the changes.)

Notes

- For other considerations when replicating a Microsoft Access project, click ».
- For more information about synchronizing data in your replica set, click ».

Convert a replica set to Microsoft Access 2000

Caution It's important that you do not synchronize the test Design Master you create by following this procedure and replicas with members of your working replica set. Otherwise, unintended changes to your data or structure of your working replica set might occur.

1. Make a copy of the current Design Master (created in Microsoft Access 97), and put the copy on a different computer or isolate it completely from any other members of the replica set.

2. In Microsoft Access 97, make the isolated copy the new Design Master by pointing to **Replication** on the **Tools** menu and then clicking **Recover Design Master**.

3. Create some replicas based on the new Design Master.

4. On the **Tools** menu, point to **Database Utilities**, and then click **Convert Database** to convert the new Design Master to a Microsoft Access 2000 database.

5. Repeat step 4 for all replicas in your replica set.

6. Synchronize the new converted Design Master with the newly converted replicas.

7. Run any tests you'd like to try.

If you want to change some objects, you can either make the changes again in the original Design Master after it is converted, or keep the copy around long enough to import the changed objects from it.

8. After you are satisfied that the copy works, delete the new Design Master and all of its replicas. Make sure that all users who will access the Design Master and replica databases have installed Microsoft Access 2000.

9. Convert the original Design Master to Microsoft Access 2000, and convert all the original replicas.

   You can't open the converted replicas again in Microsoft Access 97.

Note For other considerations when replicating a Microsoft Access project, click ».

Make a replicated database a regular database

After you've converted a database into a Design Master, you can't convert it back to its former status as a regular database. However, if you no longer want to use replication and want to decrease the size of a replicated database, you can create a new, regular database that contains all of the objects and data in your replicated database without the additional system fields, tables, and properties associated with replication.

Note For each table in the new database, create the same index used in the replica table. If the s_GUID is a primary key (and foreign keys in other tables refer to it), you'll want the s_GUID field in the new table. If the s_GUID field is not used as a primary key, there is no reason to preserve it in the new table. The s_Lineage and s_Generation fields must not be copied over to the new table.

1. Create a new database, and open it.

2. On the **File** menu, point to **Get External Data**, and then click **Import**.

3. Click the replica that contains the database objects that you want in the regular database, and then click **Import**.

4. Click the database objects — except tables — that you want to import into the new database.

5. Click **OK** to import these objects.

6. In the new database, create a query for each table in the replica that takes all the data in the old table and puts it into a new table.

   How?

7. For each table in the new database, create the same index used in the replica's table.

   How?

8. For each table in the new database, create the relationships that existed for the replica's table.

   How?

9. Save your new database.

Replication with earlier versions of Microsoft Access

Versions of Microsoft Access before Microsoft Access 7.0 can't replicate databases and likewise cannot use replicated databases.

You can convert a replica set that you created in Microsoft Access 97 by synchronizing your replica set and then converting every replica individually in Access 2000. For information about converting a replica set for use in Microsoft Access 2000, click ».

AutoNumber fields and replication in a database

If you plan to replicate your database, you need to consider what field size is appropriate for the AutoNumber field used as a table's primary key. If you use an AutoNumber field as the primary key for a table in the replicated database, set its **FieldSize** property to either **Long Integer** or **Replication ID**.

- If fewer than 100 records are routinely added between synchronizing replicas, use a **Long Integer** setting for the **FieldSize** property to take up less disk space.

- If more than 100 records are routinely added between synchronizing replicas, you should use **Replication ID** for the **FieldSize** property setting to prevent records from being assigned the same primary key value in each replica. Note, however, that an AutoNumber field with a Replication ID field size produces a 128-bit value that will require more disk space.

Changes made to your database when you use replication

When you replicate a Microsoft Access database, a number of changes are automatically made to your database.

What changes do you want to read about?

Fields that are added to each existing table in the database

New tables that are added to the database

New properties that are added to the database

Changes to the behavior of AutoNumber fields

Changes to the size of the database file

Suggestions when changing the design of a replicated database

- Synchronize all replicas in the replica set before making design changes.

- Check all replicas to make sure that the name you are selecting for a new field or object isn't already being used as the name of a local object. (If you do choose the same name, the local object will be renamed.)

- Don't create relationships between tables when you intend to replicate some tables but keep others local.

- Make sure that the number of bytes you add to a field or the number of fields you add to a record doesn't exceed the maximum size for a replicated database. For more information on the size limitations of a replicated database, click ».

- If you use linked tables, keep in mind that linked tables aren't automatically replicated. If you create a linked table in the Design Master, it's created as a local object and remains local until you make it replicable. When you make the table replicable, the link and the table are sent to all the replicas in the set. After the table appears in each replica, you can change the link in an individual replica or in the Design Master without affecting the other members of the replica set. To refresh the link in all members of the replica set, change the replicable linked table to a local table (in the Design Master), refresh the link, and change the table back to a replicable table.

Secure a replicated database

To secure a replicated database, you can set user permissions on replicated database objects. These permissions will not interfere with synchronization of the database. However, you can't use a password to protect your replicated database. A password-protected database can't be made replicable, nor can a password be assigned to a replicated database.

You can also keep a replica in an unshared directory and use indirect synchronization (through Microsoft Replication Manager). If direct synchronization is required, set the permissions on the share where the replicas reside so that only specific users can access them. Do not put replicas in read-only shares, because synchronizations with other replica set members will not occur.

For other considerations when replicating a Microsoft Access project, click ».

Replication with Jet and Replication Objects (JRO)

You can create and maintain a replicated database by using Jet and Replication Objects (JRO) methods and properties.

This topic provides reference information about:

Programming replicated databases

Methods

Properties

Methods and properties available in a partial replica

More information

Programming replicated databases

Use JRO if you require programmatic control over the exchange of data and design information among members of the replica set. For example, you can use JRO to write a procedure that automatically synchronizes a user's replica with the rest of the set when the user opens the database. The database must be closed to programmatically make the database replicable.

Return to top

Methods

CreateReplica method

GetObjectReplicability method

MakeReplicable method

SetObjectReplicability method

Synchronize method

Return to top

Properties

ActiveConnection property

ConflictFunction property

ConflictTables property

DesignMasterId property

ReplicaId property

ReplicaType property

RetentionPeriod property

TableName property

Visibility property

Return to top

Methods and properties available in a partial replica

Append method

Delete method

PopulatePartial method

FilterCriteria property

FilterType property

Priority property

Return to top

More information

Create partial replicas

Return to top

Replication with Data Access Objects (DAO)

You can create and maintain a replicated database by using DAO methods and properties. Use DAO if you require programmatic control over the exchange of data and design information among members of the replica set. For example, you can use DAO to write a procedure that automatically synchronizes a user's replica with the rest of the set when the user opens the database.

You can use the following methods and properties to create and maintain a replicated database:

MakeReplica method

Synchronize method

ConflictTable property

DesignMasterID property

KeepLocal property

Replicable property

ReplicaID property

ReplicationConflictFunction property

Microsoft Jet provides these additional methods and properties for creating and maintaining partial replicas (replicas that contain a subset of the records in a full replica):

ReplicaFilter property

PartialReplica property

PopulatePartial method

Learn about creating partial replicas.

Note To learn about these properties and methods, open the Visual Basic Editor and access Visual Basic Editor Help.

When I change data in one member of the replica set, it doesn't change in another member.

1. Open the member of the replica set that you want to synchronize.

2. On the **Tools** menu, point to **Replication**, and then click **Synchronize Now**.

3. Enter the path and file name of the other replicated database with which you want the open database to synchronize.

4. Click **OK**.

5. Close and reopen the database to view the updated records.

6. If your data still doesn't appear, you may have changed data in local objects, which are not synchronized with the members of the replica set, or there may have been a conflict with another member. For information on resolving conflicts, click ».

When I change the design of an object in the Design Master, it doesn't change in any of the replicas.

1. Open the Design Master.

2. On the **Tools** menu, point to **Replication**, and then click **Synchronize Now**.

I can't find the Design Master.

The Design Master might have been moved from its original location or might be damaged. To find the new location:

1. Open a replica in the replica set.
2. On the **Tools** menu, point to **Replication**, and then click **Synchronize Now**.
3. Click **Browse** to locate other members of the replica set on your computer or network.
4. Select a member of the replica set. If the **Make** '***Filename***' **The Design Master** check box appears, the member you are synchronizing with is the Design Master.

I can't open the Design Master.

The Design Master might be damaged. Make a replica of the new Design Master. For information about making a replica of the new Design Master, click ».

I can't make my replica the Design Master.

Your replica might be in use by another operation or application and can't be opened exclusively. Wait a few minutes, and try again.

If you still can't make your replica the Design Master and haven't received another error message, your replica may be damaged. Delete the replica and create a new one. For information about deleting a replica, click ».

There are two Design Masters in the replica set.

Having more than one Design Master in a replica set prevents the members of the set from synchronizing correctly. If this occurs, do the following:

1. Determine which Design Master is the original and whether it should continue to be the Design Master.

2. Delete the Design Master that should not exist.

3. Delete all replicas that had their most recent synchronization with the deleted Design Master or with other replicas that have had their most recent synchronization with the deleted Design Master.

   How?

4. Create new replicas from the remaining Design Master to replace the replica or replicas you deleted.

I want to use a different program for resolving synchronization conflicts.

1. Open the Design Master for the replica set.
2. On the **File** menu, click **Database Properties**.
3. Click the **Custom** tab.
4. In the **Name** box, type **ReplicationConflictFunction**.
5. In the **Type** box, click **Text**.
6. In the **Value** box, type the name of the custom function followed by **( )**. For example, if the name of the custom function is MyResolver, type **MyResolver( )**.

For more information about the **ReplicationConflictFunction** property, click ».

I don't want to use replication when I drag a database file into My Briefcase. How do I stop Briefcase replication?

Briefcase replicates Microsoft Access database files that you drag into the Briefcase. If you don't want to replicate the database file, change the extension from .mdb (or .mde) to another extension.

Prevent forms, reports, macros, and modules from being replicated

Note This process must be performed before a database is replicated.

1. On the **File** menu, click **Database Properties**.
2. Click the **Custom** tab.
3. Select the **ReplicateProject** property in the **Properties** box.
4. Click **No** next to **Value**.
5. Click **Modify**.

Properties that are added to your database when you replicate it

When you replicate a database or specify a database object as local, the following new properties may be added to the **Properties** collection of objects:

Property	Description
Replicable or ReplicableBool	A database or object property. When the property is set to T (or True for **ReplicableBool**), it indicates that the database or the table or query is now replicable. The **Replicable** or **ReplicableBool** properties may be used interchangeably.
KeepLocal	A property appended to a table or query. When the property is set to T, it indicates that the object should not be replicated when the database is replicated. An object that is already replicated can't have its

	KeepLocal property set.
ReplicaID	A property that provides each member of the replica set with unique identification. This property is read-only and is stored in the MSysReplicas system table.
DesignMasterID	The **ReplicaID** of the Design Master. It is stored in the MSysRepInfo system table as SchemaMaster.
ColumnLevelTracking	A database or table property. When the property is set to True (default), it indicates that conflicts are tracked based at the column level of a table. For more information about tracking changes to your data, click ».
Replication ConflictFunction	The property is used to replace the Microsoft Access **Conflict Viewer** with a customized procedure that assists users in resolving synchronization conflicts.

Notes

- Microsoft Access objects (forms, reports, macros, and modules) are either all replicated or are all local in Access 2000. Therefore, the **Replicable** and **KeepLocal** properties apply only to tables and queries. For more information on how to prevent forms, reports, macros, and modules from being replicated, click ».

- To learn about the Replicable, KeepLocal, ReplicaId, and DesignMasterID properties, open the Visual Basic Editor and access Visual Basic Editor Help.

About converting a replica set to Microsoft Access 2000

To convert a replica created in Microsoft Access 97 so that it can be used in Microsoft Access 2000, you must convert each replica individually. It is recommended that you first synchronize your replica set prior to conversion. If your replica set is secured, it is also recommended that you convert the system.mdw file, which contains permissions for the database, but it is not necessary. For the conversion to work, all members of the replica set must be using computers with Microsoft Access 2000 installed.

Take a conservative approach when converting a replica set. This procedure allows you to test a temporary second replica set before committing your original replica set to the conversion.

More information

Convert a replica set to Microsoft Access 2000

Fields that are added to your tables when you replicate a database

When you replicate a database, Microsoft Access adds the following system fields to each replicated table in the replicated database:

Field	Description
s_GUID	A globally unique identifier for each record.
s_Lineage	A binary field that contains information about the history of changes to each record.
s_Generation	A field that stores information regarding groups of changes.

Notes

- Additional fields (named Gen_*Fieldname*) are added for every Memo data type or OLE Object data type field in a table.

- The s_GUID, s_Lineage, and s_Generation system fields are system objects that may or may not be visible in your tables. To make these fields visible in replicated tables, click **Options** on the **Tools** menu, click the **View** tab, and then select the **System Objects** check box. To see the fields, open a replicated table .

- You can't change the information stored in the system fields in a replicated table.

- For other considerations when replicating a Microsoft Access project, click ».

Tables that are added to your database when you replicate it

When a Microsoft Access database is replicated and certain conditions occur, new system tables may be added to the database. Of these, the following new system tables may be of interest:

Table	Description
MSysSidetables	This table exists only if a conflict has occurred between the user's replica and another replica in the replica set. This table isn't replicated. It's supplied for information only, and its contents can be modified or deleted by custom conflict-resolution routines or by the user. All side tables are named *table*_conflict, where *table* is the original name of the table.
MSysSchemaProb	This table is present only when an error has occurred while updating the design of a replica. It provides additional details about the cause of the error. This is a local table and isn't replicated.
MSysReplicas	This table stores details, such as Replica ID and path, of all known replicas in the replica set. This table appears in all members of the replica set, but it is not replicated.
MSysTransAddress	This table stores addressing information for the Synchronizer and defines the set of synchronizers known to this replica set. This replicated table appears in all members of the replica set.
MSysTombstone	This table stores information about deleted records, and allows deletions to be dispersed to other replicas during the synchronization process. This table appears in all members of the replica set, but it is not a replicated table.
	This table stores information about the entire replica set, including the identity (GUID) of the Design

MSysRepInfo	Master. The table contains a single record. This replicated table appears in all members of the replica set.
MSysExchangeLog	This table stores information about replica synchronizations that have taken place. This is a local table and isn't replicated.

Notes

- These tables are visible when system objects are visible. To make system objects visible, click **Options** on the **Tools** menu, click the **View** tab, and then select the **System Objects** check box.

- You can't change the information stored in the MSysSchemaProb, MSysExchangeLog, MSysReplicas, MSysTransAddress, MSysTombstone, or MSysRepInfo system tables in a replicated database.

Changes to the behavior of AutoNumber fields when you replicate a database

When you replicate a database, any incremental AutoNumber fields in your tables change to random numbering. All AutoNumber fields in existing records retain their values, but AutoNumber values for inserted records are random. In other words, the record numbers don't reflect the order in which the records were inserted and, consequently, the record inserted last doesn't necessarily have the highest value.

Before you replicate a database, determine whether any of your applications or users rely on AutoNumber field's incremental numbering. If so, you can use an additional Date/Time field to provide sequential ordering information.

Note Replicas in an Access database that were created from a Microsoft SQL Server publication retain the AutoNumber functionality to support SQL Server's Named Range feature. You must manage these ranges in each replica. For more information about replication in a Microsoft Access project, click ».

Additional size limitations on a replicated database

The addition of three fields (s_GUID, s_Lineage, and s_Generation) imposes two limitations on your replicated tables:

- The Microsoft Jet database engine allows a maximum of 2,048 bytes (not counting Memo or OLE Object fields) in a record. Replication uses a minimum of 54 bytes to store unique identifiers, indexes, and information about changes to the record. If the record contains either Memo or OLE Object fields, replication uses an additional 4 bytes for each of those fields. The total number of bytes available in a record in a replicated table can be calculated as follows:

  2,048 bytes
  - 54 bytes for replication overhead
  - (4 bytes * the number of Memo fields)
  - (4 bytes * the number of OLE Object fields)
  = the maximum number of bytes available

- The Microsoft Jet database engine allows a maximum of 255 fields in a table, of which at least three fields are used by replication. The total number of fields available in a replicated table can be calculated as follows:

  255 fields
  - 3 system fields
  - the number of Memo and OLE Object fields
  = the number of fields available

Few well-designed applications use all the available fields in a table or all the bytes in a record. However, if you have a large number of Memo or OLE Object fields in your table, you might run into constraints.

Note In addition to setting limits on the available number of characters and fields, the Microsoft Jet database engine also limits the number of nested transactions allowed. You can have a maximum of seven nested transactions in a nonreplicated database, but a replicated database can have a maximum of six nested transactions.

The addition of several new system tables also adds to the size of your database. Most of these new tables contain only a few records, but some of the new tables can grow significantly, depending upon the frequency of synchronization between replicas.

The size of your database file is significant for two reasons:

- The Microsoft Jet database engine supports files up to a maximum of 1 gigabyte in size, but many users have limited space available on their hard disks. A replicated database that is larger than the original database file might use up a significant portion of the available disk space.

- If you choose to make a backup copy of the nonreplicable form of your database, you must have sufficient space on your hard disk for both the replicated database and the backup copy.

- **Note** For other considerations associated with a Microsoft Access project, click ».