

Frequently Asked Questions About Microsoft® Access Replication

By Mary Chipman, Michael Kaplan, Paul Litwin and Steve Thompson

1. What is a GUID?	2
2. Can I use a GUID as a Primary Key field?	2
3. Why did replication change all my AutoNumber fields to Random Increment?	2
4. Can I use a field with a GUID in a criteria for a DLookup? In a parameter query? In a search?	3
5. Why do strange numbers sometimes appear in my list boxes and combo boxes?	3
6. What are some design issues I should consider with replication?	3
7. Why do some of my objects get renamed?	3
8. How can I synchronize in a single direction?	3
9. Can I use database level passwords with replication?	4
10. How do I implement security with replication?	5
11. How do I refresh linked tables in a replicated database?	5
12. How do I replicate an application that has implemented split data and code MDBs?	5
13. How do I compact a replicated database?	5
14. How do I repair a corrupted Design Master?	6
15. What is Replication Manager?	6
16. My RAS connection for synchronization is really slow, how can I speed up the synchronization process?	6
17. What is indirect synchronization and how do I make it work?	6
18. Can I use DAO to perform indirect synchronization?	7
19. How do I create a partial replica?	7
20. The Partial Replica wizard only lets me define a partial replica filter to one table. How do I create additional replica table filters?	7
21. Can I make replication work from Microsoft SQL Server and Microsoft Access?	8
22. Why does my replica report that it has expired? How can I fix it?	8
23. Can I create MDEs for use with replication?	8
24. Can I Import/Export data from a replicated database?	8
25. Why do I get a conflict when I synchronize my replicas?	8
26. How can I avoid (or at least minimize) data conflicts?	8
27. How does Microsoft Access resolve synchronization conflicts?	9
28. How do I prevent replication data errors?	9
29. Once a data error does exist, how can I get rid of it?	9
30. Is there any way to create my own replication conflict manager?	9
31. Will replication work with Banyan® Vines and LANtastic®?	9
32. Troubleshooting tips for Novell networks	10
33. How do I get rid of replication errors for a replica that no longer exists?	10
34. Why don't some of my design changes propagate to the replicas?	10
35. I am having frequent database corruption problems. What to do?	10
36. How do I remove a "dead" replica?	10
37. I get the error message, "'MyData.mdb' isn't an index in this table. Look in the indexes collection of the TableDef object to determine the valid index names."	10
38. Should I use Briefcase manager to handle replication?	10

39. Can I use replication with Visual Source Safe?	11
40. How do I remove replication?	11
41. How do I implement a progress meter to display synchronization status?	11
42. Should I use backup utilities with my replicas?	11
43. Can I replicate through e-mail instead of over a network or dialup connection?	12
44. How do I recover a lost or corrupted Design Master?	12
45. How can I transfer Design Master status to another replica member?	12
46. Can I have more than one Design Master in a replica set?	12
47. How can I detect that the current replica member is the Design Master?	12
48. Why shouldn't I use replication for transactional processing applications?	13
49. How do I set up Internet replication?	13
50. Can I use Replication Manager to schedule synchronizations over the Internet?	13
51. How can I synchronize over the Internet using Visual Basic® for Applications code?	13
52. I installed Internet Explorer 4.0 and can no longer replicate my Microsoft Access 95 applications.	14
53. My RWOP queries in my secured application won't replicate over the Internet.	14
54. How do I convert a Microsoft Access 95 replica set to Microsoft Access 97?	14
55. If I must use Replication Manager and Synchronizer for Internet synchronization, must I purchase a copy of the Microsoft Office 97 Developer Edition (ODE) for each user?	14
Additional References and Resources	14

1. What is a GUID?

Replication needs a methodology to uniquely identify various items such as each record, table, and replica in a replicated database. The identifier is a GUID (Globally Unique Identifier) created by the Microsoft Jet database engine in such a manner as to guarantee uniqueness across space and time. GUIDs are not unique to Jet or replication; many programs create and use GUIDs. For example, the Windows® operating system assigns a GUID to every ActiveX control installed on your system.

When a database is converted to a replica set, several fields are added to each table, including a field named `s_GUID`. GUIDs are created by using a combination of the network node ID, a time value, a clock sequence value, and a version value. For more detailed information on how GUID's are created, please see the *Database Replication white paper*.

Here is an example of a GUID:

```
36B295B1-D128-11D0-81F9-0000F649884F
```

2. Can I use a GUID as a Primary Key field?

Absolutely. However, it's not required that you do this. If you decide to use a GUID as the primary key for a table, you must do so prior to converting the database to a replica set: for the Primary Key field, choose the AutoNumber data type; select ReplicationID as the FieldSize property setting. When the database is converted to a replica, Jet will use that AutoNumber field as the GUID and not add the new `s_GUID` field, as described in item 1, "What is a GUID?"

3. Why did replication change all my AutoNumber fields to Random Increment?

The default behavior of AutoNumber fields, which is to increment by 1 each time a record is added, would not work in a replicated application. If it did, each member of the replica set would be producing new records independently of each other with identical primary key values, causing duplicate primary key errors on every synchronization. So when you make a replica out of a regular database, this behavior is automatically changed to a randomly incrementing value, which greatly reduces the likelihood of two replicas assigning the same primary key value. If you are using AutoNumber both as a primary key value and to number records sequentially, where

the numbers are important in a business context (such as an invoice number), then you need to consider other alternatives before converting your database into a replica.

One possible alternative scheme is to create a custom AutoNumber routine that assigns numbers using code executed from your forms. This code would have to either incorporate a site or replica ID as part of the primary key or perhaps use different sets of values for each replica.

4. Can I use a field with a GUID in a criteria for a DLookup? In a parameter query? In a search?

You can't use a GUID as part of the criteria for a DLookup (or any of the other domain functions). You can use it in a parameter query by setting the parameter data type as GUID. You can also use it in a search.

The Microsoft Jet database engine stores GUIDs as arrays of type Byte. However, Microsoft Access can't return Byte data from a control on a form or report. In order to return the value of a GUID from a control, you must either:

- a) Convert it to a string. To convert a GUID to a string, use the StringFromGUID function. To convert a string to a GUID, use the GUIDFromString function.
- b) Use the Text property of the control rather than the Value property. When a GUID is bound to a control, the canonical (string) form of the GUID is displayed. The Value property is the binary data, and the Text property is the canonical form.

5. Why do strange numbers sometimes appear in my list boxes and combo boxes?

In Microsoft Access 95, Forms and Form controls were not "replication aware;" therefore, if list box or combo box controls were bound directly to a table record source, there were situations that would cause the (normally) hidden table replication fields to display. Using a record source based on queries returning specified fields is a much safer technique. This was fixed in Microsoft Access 97.

6. What are some design issues I should consider with replication?

A book could probably be written on this subject alone!

Consider the physical changes that occur to your database. Each replicated table will have 3 new fields added (more if a table contains Long Binary data types) that will add 24 bytes to the length of each record. Each replicated database will get approximately 14 new tables that are replication system tables. The Microsoft Jet database engine still enforces the 255 fields per table, 2048 byte record length per record. When planning for replication, be sure that the additional fields added to each table will not exceed the maximum number of fields per table or the maximum number of bytes per record.

When a database is converted to a replicated set, placement of the Design Master (DM) deserves special consideration. The DM is the only place you can make revisions to the design of database objects, so you want to make sure that you don't place it in a situation where you inadvertently synchronize partial changes.

7. Why do some of my objects get renamed?

In Microsoft Access 97, if a table is created at the Design Master that happens to have the same name as a local object in any of the replica members, rather than trigger a design error, the local object in the replica gets renamed to '<local object>_Local'.

8. How can I synchronize in a single direction?

Use VBA/DAO and the Synchronize method. The default is bi-directional, but you can choose to synchronize in a "one way" direction, either Export or Import. Keep in mind that any design changes will be bi-directional.

The syntax is:

`database.Synchronize pathname [,exchange]`

The Synchronize method syntax has the following parts.

Part	Description	
Database	An object variable pointing to a replicable Database object that you want synchronized.	
Pathname	Path to the target database with which database will be synchronized. The trailing .mdb can be optionally omitted.	
Exchange	A constant indicating which direction to synchronize changes between the two databases. This can be one of the following Integer constants:	
	dbRepExportChanges	Send changes from database to pathname.
	dbRepImportChanges	Receive changes from pathname.
	dbRepImpExpChanges	(Default) Bidirectional exchange.
	dbRepSyncInternet	Microsoft Access 97. Exchanges data between files connected by an Internet/Intranet pathway. The pathname to synchronize would contain a Uniform Resource Location (URL) as an identifier. In addition, you can 'add' either the Export or Import constant to the Internet exchange option to achieve a send or receive action via the Internet.

Here's a sample procedure to tailor for your own use. The procedure expects the database/path name to synchronize from, the path/database target name, and a synchronization action. You need to specify the intSync argument as 1 for bi-directional, 2 for export changes, 3 for import changes, and 4 for Internet syncs. Here's an example of calling the function to export changes only:

```

Call SynchronizeDBs("C:\MyRepl.mdb", "L:\OtherRepl.mdb", 2)

Sub SynchronizeDBs(strDBName As String, strSyncTargetDB As String, _
    intSync As Integer)

    Dim dbs As DATABASE

    Set dbs = DBEngine(0).OpenDatabase(strDBName)

    Select Case intSync
    Case 1 'Synchronize replicas (bidirectional exchange).
        dbs.Synchronize strSyncTargetDB, dbRepImpExpChanges
    Case 2 'Synchronize replicas (Export changes).
        dbs.Synchronize strSyncTargetDB, dbRepExportChanges
    Case 3 'Synchronize replicas (Import changes).
        dbs.Synchronize strSyncTargetDB, dbRepImportChanges
    Case 4 'Synchronize replicas (Internet).
        dbs.Synchronize strSyncTargetDB, dbRepSyncInternet
    End Select

    dbs.Close
End Sub

```

9. Can I use database level passwords with replication?

Microsoft Access 95 and Microsoft Access 97 introduced the database password feature. The database password allows you to set a single password on a database instead of having multiple logins with regular Microsoft

Access security. This feature is incompatible with replication. You cannot replicate a database with a password set, nor can you set a database password on a replica. You can always implement user-level security in Microsoft Access; merely have all of your users log on using the same ID and password. This achieves the same effect and is a more robust solution since the database password is notoriously unreliable. However, it does require that you distribute your secured workgroup file (system.mdw) with your replicas.

10. How do I implement security with replication?

Both replication and security are advanced features, which require a lot of advance planning in order to implement successfully. You can combine the two, but you need to consider how your secured, replicated application is going to work. You can secure your application in the normal way, and any changes you make to the permissions of your database objects will be distributed during synchronization. However, you can't synchronize workgroup files, so you need to distribute your workgroup file (or system.mdw) separately to each site participating in replication. This can make administration a headache if you need to add and remove users all the time. Another option is to create common groups in different workgroup files that have identical names and PIDs. This would allow each site to maintain its own workgroup file while still allowing access to replicated objects. For more information on how to set up Microsoft Access security, see the *Security FAQ*, which is available on Microsoft's Web site.

11. How do I refresh linked tables in a replicated database?

If you have code that deletes the linked tables and re-creates them, this will fail in all replicas except for the Design Master, because deleting linked tables and adding new ones are both considered design changes. In order to refresh the links, you'll need to write code for your front-end replicated application that updates the Connect property (with the path/mdb name of the back-end database) of each attached TableDef object. Follow the update to the Connect property with the RefreshLink Method for each TableDef object to let the Jet database engine know where the new data lives. Fortunately, this is not considered a design change for the purposes of replication, so you don't have to worry about the path propagating inappropriately to other replicas during synchronization.

12. How do I replicate an application that has implemented split data and code MDBs?

You can replicate the application database, the table database, or both. However, you may find it more efficient in practice to only replicate the table database. Although design changes are replicable, synchronizing them can cause problems. There have been reported instances of design changes to forms and modules not propagating successfully to all replicas in a replica set. If possible, develop your application fully before replicating and distributing it. If you need to make design changes later, then distribute the front-end database separately from the tables.

13. How do I compact a replicated database?

When compacting a replica in Microsoft Jet Database Engine 3.5x, there are four issues to keep in mind:

- Make sure the replica is copied back to the original location/name after the compact and before synchronizing with any other replica. If you don't, then you'll be adding another replica to the replica set and if the replica is the Design Master, it'll no longer be the Design Master (because it moved). The best approach is to make a backup of the replica before compacting, and then compact it to the same location/name.
- If you compact a replica that is corrupted, it will lose its replicable status (and Design Master status if it's the Design Master). Compacting a corrupted replica causes the replica to return to a normal, non-replicated database, but with all of the hidden system tables and fields still present.
- For best results, compact the replica twice. The first compact performs the normal consolidation process and marks replicated objects that it needs to delete but doesn't actually delete them. The second compact deletes these objects and recovers space associated with the deleted replicated objects.

- Although the above advice on compaction is important all the time, it is *vital* in the Design Master, especially when making design changes to Microsoft Access-specific objects. This is because of how Microsoft Access “versions” its objects. If you make 80 changes to a form and save it 80 times, there will be 80 copies of that form stored in your database so that the Jet database engine can apply those changes somewhere else. If you compact the database before synchronizing, the Jet database engine will notice that 79 of those changes are irrelevant and do not need to be kept. If you synchronize first, however, you will be unable to reclaim the space and all 80 changes will have to be applied to every replica. So again, *always compact the Design Master before synchronizing*.

14. How do I repair a corrupted Design Master?

Don't even try – throw it away. Compacting or repairing **any** replica, whether it's a Design Master or not, doesn't work. To replace a corrupted Design Master, synchronize the surviving replicas and designate one to be the new Design Master. Also see item 44 "How do I recover a lost or corrupted Design Master?".

15. What is Replication Manager?

Replication Manager is an optional tool that ships with the Office 97 Developer Edition (ODE) (or the Microsoft Access 95 Developer's Toolkit) that assists you in some aspects of replication. Replication Manager includes features that allow you to schedule and execute unattended synchronizations without programming. In addition, it lets you use indirect synchronization (see item 17 "What is indirect synchronization and how do I make it work?") and Internet synchronization, and provides tools for examining the synchronization history for a replica. One of the components of Replication Manager is the Synchronizer (called the Transporter in Microsoft Access 95). This component performs the unattended synchronizations that you set up with Replication Manager.

16. My RAS connection for synchronization is really slow, how can I speed up the synchronization process?

Consider using indirect synchronization described in item 17, “What is indirect synchronization and how do I make it work?”

17. What is indirect synchronization and how do I make it work?

Indirect synchronization requires use of Replication Manager. When you synchronize two replicas using indirect synchronization, a synchronizer on one computer exchanges changes with a synchronizer on another computer, thus avoiding the direct opening of a database over a wide area network (WAN).

There are several benefits of using this process:

- It greatly reduces the possibility of database corruption that could be caused by a line drop during a database update.
- Should an indirect synchronization fail, Replication Manager will simply re-send any changes on the next pass.

Replication Manager is included with the Office 97 Developers Edition. For Microsoft Access 95, it is packaged with the Microsoft Access Developers Toolkit (ADT). You will need to configure the Replication Manager at both the local and remote sites and create separate dropbox locations for each Replication Manager to use indirect synchronization.

This involves setting up the dropboxes (or shared folders), which will be used to store the changes. You don't want to actually locate the replicas themselves in the shared folders or else indirect synchronization will be bypassed in favor of a direct synchronization.

Here are the steps to configure Replication Manager on the **local** computer:

1. Install ODE with Replication Manager (or use Office Developers Edition Setup wizard).

2. Create a shared folder, which will be the Synchronizer dropbox. (**Do not** put the replica in ANY shared folder, or else indirect syncs will be bypassed).
3. Configure Replication Manager for indirect synchronization by checking the option to support indirect synchronization. Follow the rest of the steps in the wizard by specifying the shared folder you created as the dropbox, by naming the Synchronizer, and by selecting a synchronizer log file.
4. Open Replication Manager and select the replica to be managed.
5. Optionally set up a synchronization schedule.

On the **remote** computer:

6. Install Replication Manager and create a share for the Synchronizer drop box, as in steps 1 and 2 above.
7. Copy the replica from the local to the remote computer (using a regular file copy), and place it in an **unshared** folder. If the replica is in a shared folder, or the subdirectory of a shared folder (this includes the root), indirect syncs will be bypassed and a direct exchange will occur.
8. Open the newly copied replica in the unshared folder on the client computer in Replication Manager, and elect to manage the replica.
9. Connect back to the local computer and synchronize, in order to let the local computer know about the location of the copy on the remote computer.

Whenever the Replication Manager manages a replica, it will write the location of its dropbox into the replica. With remote connections, you must now notify other replicas of the dropbox location, which is why you need to do step 4 above. You notify other replicas of the new address, or dropbox, in one of two ways: either you create a new replica from this managed replica, or you synchronize with an existing replica.

18. Can I use DAO to perform indirect synchronization?

No. Replication Manager must manage indirect synchronization. Look for this enhancement in a future version of Microsoft Access (Jet). You can, however, handle Internet synchronization via DAO (once the Internet server is properly configured), which is a special form of indirect synchronization (for more information, see item 49, "How do I set up Internet replication?").

19. How do I create a partial replica?

The easiest way to get started is to download the Partial Replica Wizard from the Microsoft web and experiment! DAO can also be used to create Partial Replicas, and is required to create Partial Replicas with more than one Partial Replica Filter. Once you have created a new partial replica, do not delete the full replica on which it was based, because it is acting as a sort of "backup" from before you started creating partial replicas.

20. The Partial Replica wizard only lets me define a partial replica filter to one table. How do I create additional replica table filters?

You need to use DAO to add a filter to more than one table in a partial replica table. Follow the online help links for "Create Partial Replicas" -- there are steps and code samples. The three primary database methods you will use are ReplicaFilter, PartialReplica, and PopulatePartial. You can only populate a partial replica from the Design Master when it is open exclusively, as shown in the following code:

```
Sub PopulatePartialDB
    Dim db As Database
    Dim StrFullDB As String

    Set db = OpenDatabase("C:\MyApp.mdb", True)
    StrFull = "k:\direct.mdb"
```

```

    db.PopulatePartial StrFullDB
    db.Close
End Sub

```

For more information, see some of the references at the end of this document in “**Additional References and Resources.**”

21. Can I make replication work from Microsoft SQL Server and Microsoft Access?

You can set up one-way replication from SQL Server to an ODBC DSN, which can be a Microsoft Access database. SQL Server uses "publish and subscribe" replication where one database publishes, or replicates, information to subscriber databases. There's no bi-directional heterogeneous replication available at this time – look for it in the next versions of Microsoft Access and SQL Server.

22. Why does my replica report that it has expired? How can I fix it?

If your replica was created using DAO or the Replication Manager, the default expiration period is 60 days. If the replica set was created through the UI, the default expiration period is 1000 days. Unfortunately, the expiration date is not exposed to DAO. You need to use the Replication Manager to change the expiration date. If you don't have the Replication Manager installed, you can always just open your Design Master and synchronize with the replica member, even though it reports it has expired.

Of course, this will not work if you have made design changes that the “expired” replica never received. If this is the case, then the Jet database engine will not be able to get the two replicas to have the same schema, and the expired replica will never be able to synchronize with the other replicas again.

23. Can I create MDEs for use with Replication?

Yes. However, a replicated database cannot be converted to an MDE. A database in MDE format can be converted to a replicated database. To convert a replicated database to MDE format would require converting the database to a non-replicated database, as described in item 39, "Can I use replication with Visual Source Safe?"

24. Can I Import/Export data from a replicated database?

Yes. Just use the standard import/export process. If you import into a non-Design Master replica, however, and create a new table, you won't be able to replicate it. Any new tables must be created in the Design Master in order for them to be replicable.

25. Why do I get a conflict when I synchronize my replicas?

Because more than one replica has updated data in the same record since the last synchronization.

26. How can I avoid (or at least minimize) data conflicts?

The best way to minimize conflicts is to keep them from happening in the first place. You could write an entire book on this topic, but here are a few examples of ways to avoid or minimize conflicts:

1. ***Adding new records instead of editing existing ones.*** For example, when tracking inventory, rather than keeping a single field that has the quantity, create a log table that lists the change. This way, when two different people remove some of the same item, you will not get a conflict for the two of them editing the same row.
2. ***Splitting up records when the different users edit different fields in the same record.*** By splitting up such fields into two tables with a 1-1 relationship between them, you can avoid getting conflicts in cases where different fields in the same table are being modified.
3. ***Creating smarter synchronization schedules.*** The general rule is to set up your synchronization schedule

based the way your application works. If one replica contains data that another one will need right away, your application should make the synchronization right away. This avoids potential conflicts with someone else changing the same record before the first set of changes has propagated. It will also help you avoid angry customers who wonder why they just gave someone an address change and why they have to do it all over again for someone else!

27. How does Microsoft Access resolve synchronization conflicts?

The most often updated replica member wins. In other words, if you update a record 5 times in one replica, and 2 times in another replica, the replica where the record was updated 5 times wins. Any change to a record counts as a single update, regardless of how many fields are involved. Only the losing replica will receive the conflict table.

In case of a tie, the replica with the lowest ReplicaID wins. Replica IDs are handled internally -- you cannot change the ReplicaID for a replica. The only way to guarantee that one replica will have a lower ReplicaID at creation is to create both replicas in the same session of Jet or Microsoft Access. When you do this, the one created first will have the lower ReplicaID.

28. How do I prevent replication data errors?

Deleting, re-adding or changing the Replicable property of the table will not fix the underlying problem. Consider these factors to minimize design errors:

- When you initiate design changes to the replica, there cannot be any open tables on that replica.
- Be sure you're not using bound forms, which may hold a table open, blocking successful replication
- Synchronize all replicas in the replica set before making any more restrictive changes to table level validation (TLV) rules. After you have made the change, synchronize to all replicas again as soon as possible to minimize the chance that someone has entered data in the interim that breaks the new rule.
- Use the above rules for other schema changes that are more restrictive, such as lessening the maximum number of characters of a field, changing a relationship from unenforced to "enforce referential integrity", adding a unique index, or any other change where someone may be violating the new rule in a replica at the same time you are adding the rule to the Design Master.
- Compact the DM twice before initiating the synchronization
- Use primary keys that are either difficult or impossible for users of two different replicas to duplicate, to avoid the "Duplicate primary key" data error.

29. Once a data error does exist, how can I get rid of it?

You will want to look at MSysSchemaProb, a local table in the replica by un hiding the system tables (Tools/Options/View...). It usually provides details on the nature of the problem. You'll need to remove the problem element from the Replica, and then re-synchronize with the DM.

For example, if a record in this table mentions a duplicate primary key error, you will have to change one of the primary key values and then re-synchronize the replicas.

30. Is there any way to create my own replication conflict manager?

This is fairly involved. See *Access 97 Developer's Handbook* by Litwin, Getz, and Gilbert for more details on several custom conflict manager routines.

31. Will replication work with Banyan® Vines and LANtastic®?

Unfortunately, these networks are not supported for Microsoft Access Replication at this time.

32. Troubleshooting tips for Novell networks

One problem could be that the client computer is running Windows or Windows NT® and using the Novell Client. If so, then load the Microsoft Client. If the Microsoft Client is installed, then make sure to get the SR1 patch for Windows 95, which includes an updated version of the Microsoft Client for Novell Networks.

On the network side, be sure the number of record locks per connection is set to 10,000 and the Max record locks the server can handle is set to 200,000.

33. How do I get rid of replication errors for a replica that no longer exists?

Attempt to synchronize to the path where the replica used to be. When the sync fails because it cannot find the database, it will automatically remove the replica errors for that non-existent replica. If you have deleted the folder, you may need to recreate it for this to succeed.

34. Why don't some of my design changes propagate to the replicas?

Make sure that the replicas are closed before you attempt to synchronize design changes. As with any synchronization, compact all the replicas twice before synchronizing as well. If the problem is with code not running, you can open the database with the /decompile command-line switch and then recompile all the code.

35. I am having frequent database corruption problems. What to do?

Microsoft Access does not handle data collisions well. Make sure your network is not faulty because this will cause your database to crash and to become corrupted. Once a replica is corrupted, it cannot participate in synchronization.

Before attempting to replicate design changes, make sure to synchronize all of your other replicas first until the data is consistent across all members of your replica set. Compact all members before synchronizing.

One other suggestion is not to create relationships between your tables and to rely on code and queries to maintain referential integrity. This can make resolving conflicts on unique indexes easier to deal with if you don't also have to do battle with resolving relationships.

If you are synchronizing over a slow WAN line, consider using indirect synchronization (see item 17, "What is indirect synchronization and how do I make it work?").

36. How do I remove a "dead" replica?

If you try to synchronize with a deleted replica from within Microsoft Access, when Microsoft Access cannot find the replica, it will ask if you want to remove the database from the replica set. If you answer yes, the replica should no longer appear on the Replication Manager map. Replication Manager looks at the MSysReplicas table to create the map. You cannot open the MSysReplicas table yourself and delete records from it because it has read-only permissions set by Microsoft Access.

37. I get the error message, "'MyData.mdb' isn't an index in this table. Look in the indexes collection of the TableDef object to determine the valid index names."

This is a known corruption issue with Microsoft Access 97. You can obtain a new repair/compact utility for Microsoft Access 97 on the Microsoft Web site (www.microsoft.com).

38. Should I use Briefcase manager to handle replication?

We recommend using either the Microsoft Access UI, Replication Manager or VBA/DAO. Briefcase is fine for simple replication, but it has some limitations:

- It will only work on computers which have a copy of Microsoft Access installed where the Briefcase Replication has been selected during installation. Otherwise, it simply replaces the MDB as a file copy based on the date/time stamp of the file.

- Briefcase will only synchronize on demand in a bi-directional manner. You can't set up a schedule or use DAO to synchronize one-way only.

39. Can I use replication with Visual Source Safe?

You need to remove the database from source code control before you can replicate it. You can set properties such as KeepLocal on objects that you do not want to be replicated, so that it will be easy to replicate the database once it is removed from source code control.

40. How do I remove Replication?

If you are just starting out and haven't made many changes to your data and objects, chances are that you created a backup copy of your original database. Search for it in the folder where the replicated database is – it will have the same base file name as the replicated database with a "BAK" extension. If you've gone beyond that point, you can download the Replica To Regular wizard from the Microsoft Web site, which will take care of the drudgery for you. If you do it manually, you need to follow these steps:

1. Create a new database and import all the objects from the replicated database, except tables.
2. Close the new database and open the replicated database. Create a new query and select the first table in the Show Table dialog box. Add all the fields except for the replication fields (s_Generation, s_Guid, s_Lineage), unless they are used in your application. If so, then add them.
3. Select the Make Table query option and use the current database name and the database name of the database you just created in step 1. Run the query. Repeat this process for every table in the database.
4. You will then need to recreate all indexes, properties, and relationships that existed in the original replicated database.
5. When you're done, make sure to compact and repair the new database.

If you would like to avoid the overhead of having to re-create all of the properties and indexes on your tables, and you don't mind having the s_Guid field but would like the other fields to be removed, you can do the following:

1. Either delete all of the relationships in your database or make them unenforced. You must do this step because the Jet database engine will not allow you to have a relationship that enforces referential integrity between a replicated table and a local one.
2. Right-click each replicated table, choose "Properties" and uncheck the "replicated" check box.
3. Import the tables from the replicated database as you did the other objects.
4. Recreate the relationships in your database (or change their type back to enforce referential integrity). All of the properties (such as InputMask, Format, Caption, and so on.) and all of the indexes will still be there.

41. How do I implement a progress meter to display synchronization status?

Unfortunately this property is not exposed to developers, so there is no way to do this. Maybe this useful feature will be available in a future version of Microsoft Access.

42. Should I use backup utilities with my replicas?

You don't really need to – replication itself is a good mechanism for creating backups. Use another replica member on another physical drive or computer to back up your replicas. Synchronize on a regular basis to ensure a minimum amount of downtime should you ever need to restore from the backup replica. The general rule of thumb is, how much time can you (or your customer) afford to take to re-create the data lost? That answer would help determine the minimum synchronization interval.

Should a replica member become corrupted or lost due to media theft or failure, simply create a new replica from another replica in the replica set. Should you lose the Design Master, see item 44, "How do I recover a lost or corrupted Design Master?"

43. Can I replicate through e-mail instead of over a network or dialup connection?

No. This feature is not supported at this time.

44. How do I recover a lost or corrupted Design Master?

Synchronize all of the replicas in your replica set to make sure that all existing members are up to date. Then choose one of them to be the new Design Master. You can make it the Design Master either from the Replication menus, the Replication Manager, or from code by setting the DesignMasterID property.

45. How can I transfer Design Master status to another replica member?

Microsoft Access 97 allows transfer of the DM status by using the UI (Tools/Replication/Synchronize Now). In the 'Synchronize Database <dbname>' dialog box, enable the check box to make the target replica the new Design Master. Behind the scenes, the Jet database engine is transferring the DesignMasterID property to the ReplicaID property of the target replica. The following DAO code example will work in Microsoft Access 97 and Microsoft Access 95. The source for this code is *Microsoft Jet Database Engine Programmers Guide*.

```
Sub SetNewDesignMaster(stroldDM, strNewDM)
    Dim dbs As Database
    Dim newdmdb As Database

    ' Open current Design Master in exclusive mode.
    Set dbs = OpenDatabase(stroldDM, True)
    ' Open database that will become the new Design Master.
    Set newdmdb = OpenDatabase(strNewDM)

    dbs.DesignMasterID = newdmdb.ReplicaID
    dbs.Synchronize strNewDM, dbRepImpExpChanges
    dbs.Close
    newdmdb.Close
End Sub
```

46. Can I have more than one Design Master in a replica set?

This is seldom a good idea, except perhaps to transfer the DM status to another replica member for a specific developer. Design changes should only be made in one location. Should the need arise to transfer the Design Master status to another replica member, see item 45, "How can I transfer Design Master status to another replica member?"

Some people have tried to use replication as a limited form of version control. Although it does not keep historical information, it does allow you to limit changes to one developer at a time. The act of "checking out" the database is simply transferring Design Master status to that replica. This technique, however, is not a good idea unless you synchronize *very* frequently, because you can get into a state where some replicas become confused as to who the Design Master is and how to apply schema changes that the different Design Masters have made.

47. How can I detect that the current replica member is the Design Master?

If the ReplicaID is equal to the DesignMasterID, then the database name passed as an argument to this function is the Design Master, as shown in the following code:

```
Public Function IsDesignMaster(strDBName) As Boolean
    Dim dbs As Database
    ' Initialize return code to False
    IsDesignMaster = False
    ' Open Database passed as argument
    Set dbs = OpenDatabase(strDBName)
```

```

If Len(dbs.ReplicaID & "") > 0 Then
  If dbs.DesignMasterID = dbs.ReplicaID Then
    IsDesignMaster = True
  End If
End If
End Function

```

48. Why shouldn't I use replication for transactional processing applications?

Transactions are generally defined in database parlance as a 'unit of work.' In the classic example of a banking application – an individual transfers \$100.00 from a savings account to a checking account. The bank would only want the transaction to succeed if the amount was withdrawn from savings and deposited in the checking account. From a programmatic point of view, the customer's savings table would need an SQL Update statement and the checking table would require a separate SQL Update statement. Imagine the customer's dismay if \$100.00 were debited from the savings account and not credited to the checking account! This transaction should succeed on an all-or-nothing basis. In other words the updates to two separate tables should either complete in their entirety or completely fail.

In a banking environment, there is generally one central database against which all transactions (such as updates/deletes) would be processed, insuring data consistency and integrity.

Microsoft Access Replication was not designed to be used with transactions, because replication is not a transaction-based system. Individual transactions run on any one replica will work as a transaction for *that* replica. Only the *results* of the transaction will get replicated to other replica members in the replica set. The heart of the problem is maintaining transactional consistency between replica members.

As an example to illustrate the replication consistency problem, consider these two transactions performed on the same customer from two different replicas.

	Replica A	Replica B
Starting combined balance	\$100	\$100
Savings	(\$20)	-
Checking	-	(\$20)
Ending combined balance	\$80	\$80

We would have a conflict record of the ending combined balance based on the results of the transaction. Each replica reports the correct balance, generated by a different transaction. Neither the Savings debit in Replica A nor the Checking debit in Replica B would generate a conflict record because these updates were performed on different tables.

If you're looking for absolute transaction consistency, consider using a product like SQL Server, which offers true transaction journaling, better recovery options in the event of a crash, rollback capability, better handling of potential deadlock situations and superior multi-user capabilities.

49. How do I set up Internet replication?

This is a complex process, which has many steps. There is a white paper on the Microsoft Web site that details the process. Another source of information is the *Access 97 Developer's Handbook* by Litwin, Getz, and Gilbert.

50. Can I use Replication manager to schedule synchronizations over the Internet?

No, you can't. This feature is not supported for Internet synchronizations. However, you can perform Internet synchronizations using VBA/DAO (see the next question).

51. How can I synchronize over the Internet using Visual Basic for Applications code?

The Synchronize method uses the following syntax:

```
<database object>.Synchronize pathname, [exchange]
```

When synchronizing replicas over a local area network, you must specify the local area network path of the replica you want to synchronize with for the pathname argument. When synchronizing replicas over the Internet, you must specify the Uniform Resource Locator (URL) address of the Internet server for the pathname argument, instead of specifying a local area network path. In addition, you must specify the `dbRepSyncInternet` constant for the exchange argument.

When supplying the URL address of the Internet server, your code does not need to supply the full path to the replica on the server. For instance, if your Internet server name is "MyServer" and contains a replica named "Northwind.mdb" in a shared "Scripts" folder, you would use the following syntax:

```
Sub SyncReplicas()
    Dim db As Database
    Set db = CurrentDb()
    db.Synchronize "http://MyServer", dbRepSyncInternet + dbRepImpExpChanges
End Sub
```

52. I installed Internet Explorer 4.0 and can no longer replicate my Microsoft Access 95 applications.

This is a known bug with Microsoft Access 95. If you try to replicate a Microsoft Access database, you will get the following error message: "Microsoft Access cannot complete this operation because it can't find or initialize the dynamic-link library Msjtrclr." Apparently, Internet Explorer 4.0 exposed an OLE problem that shipped with Microsoft Jet Database Engine 3.0 and Briefcase Manager. There is no fix at this writing. The current workarounds are to upgrade the application to Microsoft Access 97, to use DAO to create the replica set and synchronize, or to use Briefcase replication.

53. My Run With Owners Permissions (RWOP) queries in my secured application won't replicate over the Internet.

This is a known bug – there is no fix. RWOP queries can be successfully replicated by doing a direct exchange; only indirect or Internet exchanges fail synchronizing RWOP queries on secured databases. The only workaround is not to attempt to synchronize design changes to queries with the Run permissions set to Owner's. Instead, import them into each replica.

54. How do I convert a Microsoft Access 95 replica set to Microsoft Access 97?

1. Synchronize all members in the replica set and compact all of the databases.
2. Convert the replica set Design Master to Microsoft Access 97.
3. Re-synchronize with all other members in the replica set. All replicas will be automatically upgraded to Microsoft Access 97.

55. If I must use Replication Manager and Synchronizer for indirect synchronization, must I purchase a copy of the ODE for each user?

No, the license for the ODE allows you to distribute freely all needed replication components.

Additional References and Resources

- Chapter 13, "Mastering Replication," *Access 97 Developers Handbook*, Sybex, (make sure you get the Sybex one, not that other one) by Litwin, Getz and Gilbert. 1997.
- Chapter 7, "Database Replication," *Microsoft Jet Database Engine Programmer's Guide*, Microsoft Press, by Haught and Ferguson. 1995.
- *Access/VB/Office Advisor*, published monthly by Advisor Publications. Call 800-336-6060 for subscription information or visit their Web site at <http://www.advisor.com>.

- *SmartAccess* newsletter, published monthly by Pinnacle Publishing. Call 800-788-1900 for subscription information or visit their Web site at <http://www.pinpub.com>. A couple of articles in particular deal with some advanced topics:
 - “Dealing with Replication Conflicts”, May 1996 issue (by Kaplan and Shanklin).
 - “Partial Replication with Jet 3.5”, December 1996 issue (by Kaplan and Shanklin).
- <http://www.microsoft.com> has a complete listing of Knowledge Base articles and the Replication White Paper. In addition, there is also a new Replication Web site, which is located at:

<http://support.microsoft.com/support/access/content/repl/replication.asp>

The Replication White Paper can be found at:

<http://support.microsoft.com/download/support/mslfiles/REPLIC35.EXE>

or

<http://www.microsoft.com/AccessDev/Articles/repjet2.htm>