



# Instructions for Setup and Use of RDA's AutoReplicator 1.1

by

**Resource Development Associates'**

© Copyright 1999-2001 by Dennis Santoro and Resource Development Associates. All rights reserved.

## Installation and Use

RDA's AutoReplicator is a tool for Paradox developers who want to manage remotely running copies of their applications by updating them from a master code source. By using this tool you can build in the functionality of having your applications check for updates at the master copy every time they start. If there are no updates, the code execution is completely invisible. If there is an update, RDA's AutoReplicator manages the copying of components from the master to the local copy. As a developer you only need to add 3 lines of code to your application! The time saved on application management and updates should pay for the product on the first update. It is designed to be easy to use and requires very little processing overhead. It also comes with a bonus custom toolbar you can add to your applications. It contains the following components:

**The AutoReplicator Library**

**The AutoReplicator Interface Form**

**Tables for Managing Update Synchronization**

## Installation

To install RDA's Autoreplicator your application must have 2 aliases and 2 separate directories. One must be the alias and directory where the master copy of your program to be replicated resides. This is the alias from which all updates will be copied and will be referred to as the Update alias. This alias must point to a directory which all users have read access to. The other is the alias from which users run the program. This will be referred to as the Main alias. It is not required that you change existing aliases from which your application runs, however. For example, if your application is a client tracking database you might have the aliases ClientTrackMain and ClientTrackUpdate.

For users who run the application direct from the Update alias the path to the Main and Update alias will be the same. For users who run the application local to themselves, or from a site other than the primary program site on the server, the Update Alias is the server or master location of the program and the Main Alias is the location from which the application actually runs for the user. Data, of course will always be on the server for multiuser applications. For non multiuser applications the data can be local but the AutoReplicator should not be used to modify data tables since, unless you restructure the table and ensure that all components of the tables meet the date requirements of the most recent change, the AutoReplicator will not copy such infrequently

changed files such as .Val files. Failure to copy a .Val file can cause data corruption. Data and Programs do not have to be in separate directories, however but that is always good practice and we highly recommend it. Putting data in the Main or Update directories risks damaging the data and is discouraged since the AutoReplicator is designed to only exclude .lck and hidden files and all directory folders within the Update Alias.

## **Adding AutoReplicator to Your Applications**

In the Main Alias directory from which the application runs you need to place the AutoRep.lcl library and the lastupdt.db. In the Update Alias directory you need to place the AutoRep.lcl library, the lastchng.db, lastupdt.db, updtlist.db and updtlist.px files.

In your opening form or script you need to place in the Uses method (note the Objectpal keyword on the Uses line):

### **Uses Objectpal**

```
Autorep(updatedir string, maindir string)
```

```
;//if you want to use the toolbar  
RDAToolbar  
endUses
```

and in Var you need to place:

### **Var**

```
;//You can use any variable you wish for the library variable here  
;//but this helps keep library calls clear.  
lbAutoRep library
```

```
;//this line is only needed if you are using the toolbar  
rdatb toolbar  
endVar
```

and in open, arrive or run you need to place:

```
if lbAutoRep.open(":MainAlias:AutoRep.lcl") then  
lbAutoRep.Autorep("UpdateAlias", "MainAlias")  
else  
errorshow("Library not opened")  
endif
```

substituting the actual application Main Alias for MainAlias in the lbAutoRep.open() statement and the actual Update Alias and Main Alias for the string variables in the lbAutoRep.Autorep() line (exclude the colons from the alias strings in this line). Error trapping code is recommended.

You should also place the form AutoRep.fdl in the Update directory only. This form is used to set the change and update dates.

## Using RDA's AutoReplicator

To use the AutoReplicator function the Last Change Date from Lastchnng.db needs to have the date of the earliest file in the latest update. The Last Update Date in the lastupdt.db file in the Update Alias directory should be earlier than the date in lastchnng.db. It should not be blank. This should also be the date in the copy of lastupdt.db that is placed in the Main Alias prior to the first time the AutoReplicator is run for any user. Afterward, the date in the lastupdt.db in the Main Alias directory will indicate the last time the running copy was updated from the master copy. Any time this date is older than the date in the lastchnng.db in the Update Alias the AutoReplicator will update the running copy from the update copy when it is invoked on starting the application. If the user is running the application from the update copy then both the Update Alias and the Main Alias will be the same and the AutoReplicator will not attempt to perform any updates.

Note: if you set up your aliases so that various components are stored in separate directories (e.g. queries in one directory, reports in another and forms in a third) you can set the AutoReplicator to check and update from as many directories as you wish to as many as you wish. Each set of directories must be paired, however, and must be set up as described above. You can then simply run the AutoReplicator on startup as many times as needed to check all your directories and update from them. You can not, however, use the AutoReplicator to update from several directories into only one local directory unless all updates use progressively later last change dates and you run the AutoReplicator sequentially through the dates.

## RDA's Autoreplicator System Interface

To use this interface open the AutoRep.fdl form and type in the Update and Main Aliases for the

**Auto Replication System Date Modifier**

*To use this function the Last Change Date from Lastchnng.db needs to have the date of the earliest file in the latest update. Lastchnng.db needs to be located in the program's Update Alias directory. The file Lastupdt.db should also be located in the program's Update Alias directory. The Last Update Date in this lastupdt.db file should be earlier than the date in lastchnng.db It should not be blank.*

*There must be an Update Alias which points to the master program directory from which the updates are made. The Update alias below needs to be set to the correct system. Then, if the user runs this application from the server the updates will not run. If they run the application local to themselves the update will occur whenever they log in and there is an update waiting.*

*To use this interface type in the Update and the Main Aliases for the application you want to update (without colons). Then press the Go Button to collect the actual Paths for the Aliases. **If Main and Update are the same then you are in the correct directory and can update the system. Otherwise do NOT continue!** If you are in the correct directory then you can update the Last Change Date. This controls the update process for remote machines. Only change the Last update date if it is blank.*

Application Update Alias:

Application Main Alias:  Update Date:

Last Change Date (located in the same directory as this form):

Application Main Alias:

Application Update Alias:

Current Work Alias:

*Do not update this date unless it is Blank. If it is blank set it to a date earlier than the the Last Change Date.*

selected Application. You should be opening the copy of the interface located in that application's Update directory. Then press the Go Button to collect the actual paths for Main, Update and Work Aliases. If Main and Update are the same then you are in the correct directory and can update the system. Otherwise do NOT continue! The system will warn you if that is the case. If you are in the correct directory then you can update the Last Change Date.

Once you change the last change date in the update directory to the date of the earliest file in the latest update, users who run RDA's AutoReplicator when their local copy of the application starts will have an update run if their local Last Update date is earlier than the Last Change date in the update directory.

AutoReplicator will correctly manage updates even if a user does not log in to the system during a period in which you add multiple updates. This is possible because the local last update is managed at the users' Main directory while the Update dates are handled in the Update directory.

### **RDAtoolbar()**

This method is included as a bonus and allows you to deploy the same toolbar that shows up with the AutoReplicator System Interface form. It includes cut, copy, paste, undo, locate, locate next, vcr controls, edit toggle, post record, filter field, multi-field filter, help and close form buttons. You should keep the undo.bmp with the library or in the current working directory to enable the undo button.

### **Licensing**

RDA believes in clear and simple licensing. RDA's AutoReplicator requires the components including the autorep.fdl, autorep.ldl, lastchng.db, lastupdt.db, updtlist.db and updtlist.px files.

Each license for RDA's AutoReplicator allows the developer to use the AutoReplicator at their development site and one production site if that is different from the development site.

A site is defined as all users having access to a server where the AutoReplicator is installed and the master program being replicated is housed (LAN, WAN or stand alone PC).

There are no user limits at a site and RDA's AutoReplicator may be used for all applications housed on that server.

Additional licenses must be purchased for additional sites. For example, if you build an application which includes RDA's AutoReplicator and install it at 2 different clients, 2 licenses are needed regardless of the user count. On the other hand, if you build 2 applications which include RDA's AutoReplicator and install them both at the same site, only 1 license is required. Systems requiring different versions of AutoReplicator (for different versions of Paradox running at the same) must purchase one license per site for each version of the AutoReplicator.

Contact us to discuss volume terms or other licensing issues.