

R-1

Deployment Platform

File List

Software Version 5.0

For Windows/UNIX operating systems

May 16, 2011

Copyright © 2011 RepliWeb® Inc., All Rights Reserved

The information in this manual has been compiled with care, but RepliWeb, Inc. makes no warranties as to its accuracy or completeness. The software described herein may be changed or enhanced from time to time. This information does not constitute a commitment or representation by RepliWeb and is subject to change without notice. The software described in this document is furnished under license and may be used and/or copied only in accordance with the terms of this license and the End User License Agreement.

No part of this manual may be reproduced or transmitted, in any form, by any means (electronic, photocopying, recording or otherwise) without the express written consent of RepliWeb, Inc.

Windows, Windows XP and Windows Vista are trademarks of Microsoft Corporation in the US and/or other countries. UNIX is a registered trademark of Bell Laboratories licensed to X/OPEN.

Any other product or company names referred to in this document may be the trademarks of their respective owners.

Please direct correspondence or inquiries to:

RepliWeb, Inc.
6441 Lyons Road
Coconut Creek, Florida 33073
USA

Telephone: (954) 946-2274
Fax: (954) 337-6424

Sales & General Information: info@repliweb.com
Documentation: docs@repliweb.com
Technical Support: <http://support.repliweb.com>
Website: <http://www.repliweb.com>

Table of Contents

1.	Overview	1
	File List Concept - A Multi-Step Process	1
	Example – Customized Report	1
2.	Job Configuration.....	3
3.	Template File.....	6
	File List Concept - A Multi-Step Process	6
	Creating a Template File.....	6
	Execution Variables.....	7
	Examples of Template Files	8
	Customized Reports.....	8
	UNIX File Lists with Parameters	8
	Deleting Source Files after Transfer.....	8
	Command	9
	Formatted HTML Lists	9
	Executing Scripts and File Lists	9
	Inserting File Transfer Statistics into SQL Database	10

1. Overview

A critical aspect of enterprise level data deployment is the ability to easily determine exactly which files were transferred to which systems, and the details associated with the transfer itself. For example, this capability is important for data integrity validation and to governance issues such as Sarbanes-Oxley. The RepliWeb R-1 **File List** feature enables R-1 deployment or replication to automatically create a file containing the above-mentioned information.

File List Concept - A Multi-Step Process

During the transfer stage, a detailed **Transfer Report** is created, containing information about the files that were transferred in the replication process.

Using a text **Template** file, a **File List** file is created on the Center, combining the information from the **Transfer Report**, with the format defined in a **Template** file. The Template file can be easily created in accordance with a user-defined format.

In the **File List** file, each line represents information pertaining to a single file transferred. Once the File List file has been created, it can then be stored locally on the Center, e-mailed to multiple recipients, or even executed as a Windows batch file or UNIX script on the R-1 Center.

From an R-1 perspective, the order of events is as follows:

1. The user creates a Template file, using R-1 environment variables.
2. The job is run with the **Procedures – File List** option set. The job creates the log of transferred files (Detailed or Summary Transfer Report).
3. Upon job completion the job uses the Template file to create the File List file where each line represents a file that was transferred.
4. The File List file is emailed or executed on the Center.

NOTE: The whole process, starting at creating the text Template file, until the File List file is stored or executed, is performed exclusively on the Center.

Example – Customized Report

In this example, a customized report is created, listing the attributes of each of the transferred files. The customized report is saved to a file called `c:\attributes.txt`:

1. The user creates a **Template** file, using the environment variable `%s`, which represents the name of the file that was transferred.

The Template file contains one line. For example:

```
attrib "%s" >> C:\attributes.txt
```

NOTE: The Template File is NOT a batch file. It is a text file being used to create a batch file that will then be executed.

2. The job is run with a **Detailed** or **Summary** transfer report and **Procedures – File List Execute** option set.

The **Detailed Transfer Report** looks like:

```
16:35:07 LAN transfer to node <production_srv>
16:35:07 Sending data started at - Mon Feb 22 16:35:07 2010

Source File name : C:\source\expenses.xls
Target File name : D:\target\expenses.xls
File size : 11330 bytes
Transfer ended at : Mon Feb 22 16:35:07 2010
Transfer succeeded

Source File name : C:\source\contract.doc
Target File name : D:\target\C:\source\
File size : 15297 bytes
Transfer ended at : Mon Feb 22 16:35:07 2010
Transfer succeeded

16:35:08 Successfully Copied <10> files
16:35:08 Transfer completed at - Mon Feb 22 16:35:07 2010
```

3. Upon job completion, the job uses the **Template** file to create the **File List** file, where each line contains information regarding a file that was transferred.

The **File List** looks like:

```
attrib "C:\source\expenses.xls" >> C:\attributes.txt
attrib "C:\source\contract.doc" >> C:\attributes.txt
```

4. The **File List** file is then executed by R-1 on the Center; each line is executed in sequence. Resulting in the following C:\attributes.txt file:

```
A_ C:\source\expenses.xls
A_ C:\source\contract.doc
```

2. Job Configuration

The job configuration consists of the following steps:

1. Click the appropriate General tab - **General** for a Replication job / **Edge General** for a Distribution job.
2. In this tab, set the **Transfer Report Style** properties to either **Summary** or **Detailed**. Any other report style will result in an **empty** file list being created.

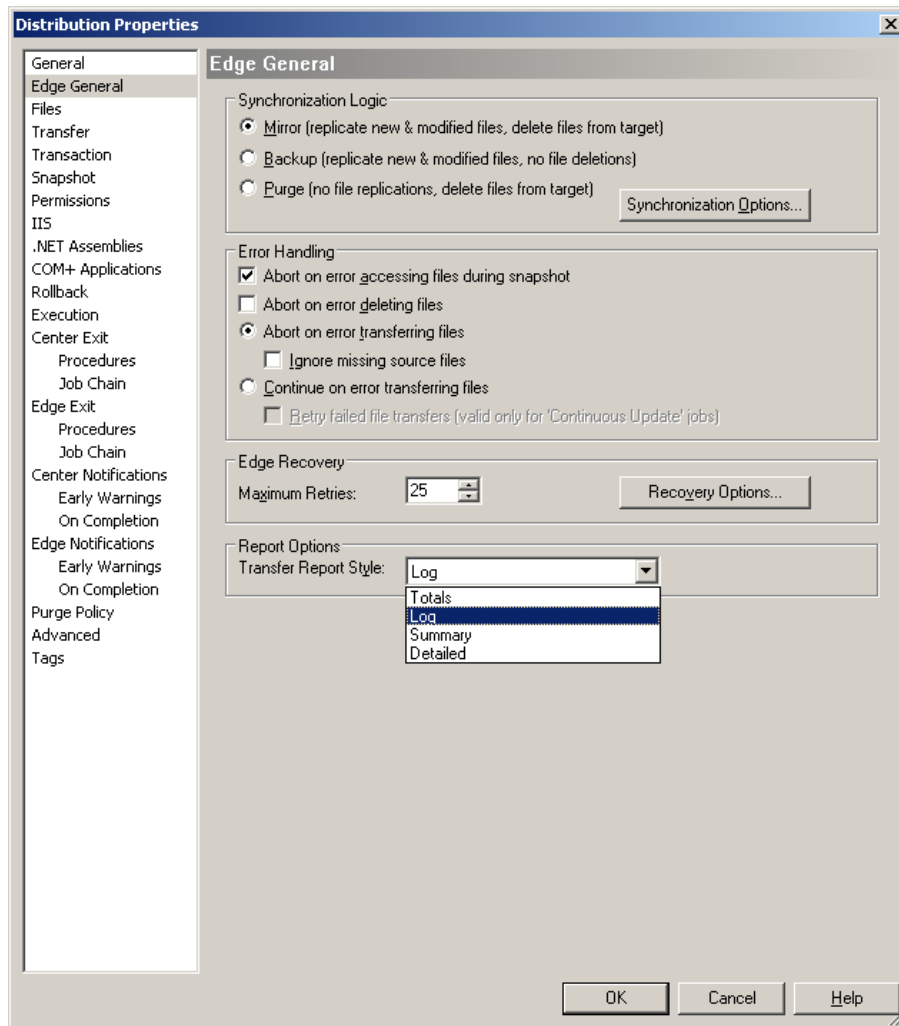


Figure 1 – Edge General Tab – Transfer Report Format

3. Click the **Procedures** tab and specify the **File List** exit procedure – this varies depending on the type of job scheduled and should be enabled in all tabs that apply:
 - In an **Upload / Download** job, the file list exit procedure should be enabled in the **Procedures** tab

- In a **Distribution** job, the file list is created for each edge containing the files unicast. The File List exit procedure should be enabled in the **Edge Exit – Procedures** tab.

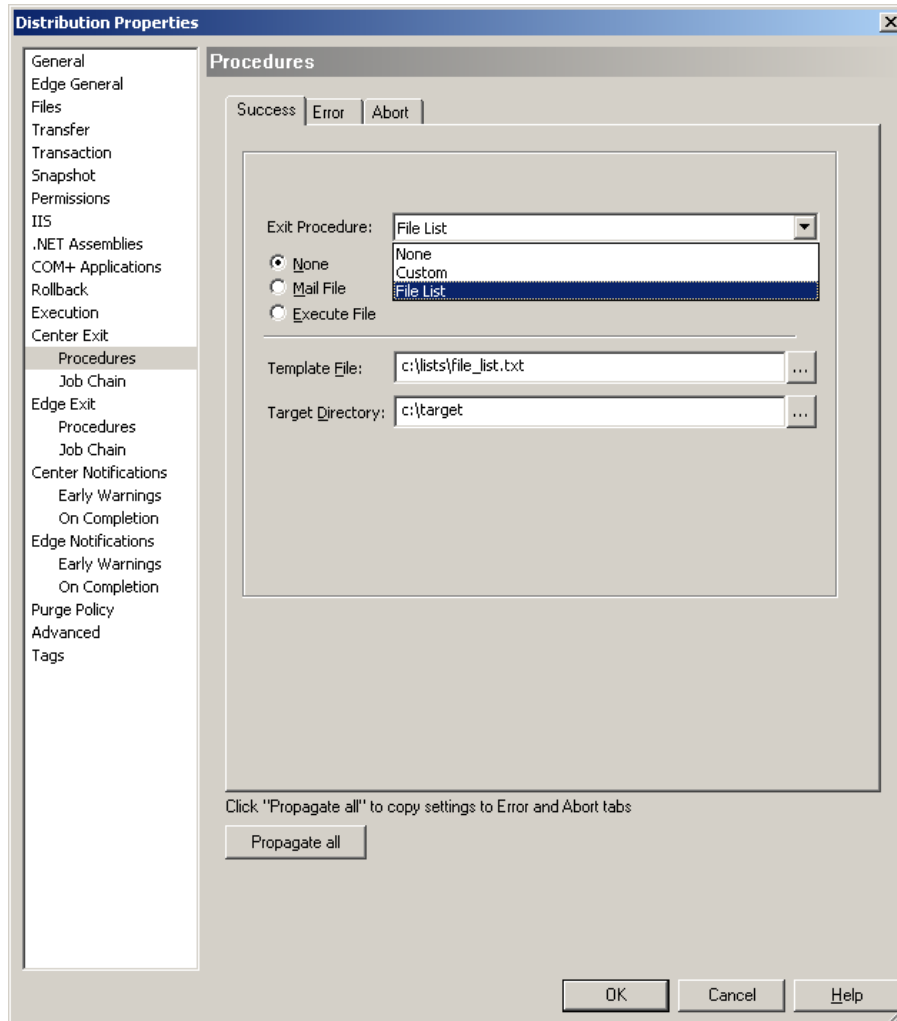


Figure 2 – Procedures – File List

4. Specify the **File List** properties:
 - **None** – Creates the file list on the Center.
 - **Template File** – Mandatory.
The full path to a custom template must be defined. Templates are explained in detail in the following section.
 - **Target Directory** – Mandatory.
Specifies the directory on the Center where the file list will be created. Do not specify a file name, as the file list will be created as **transfer_JOBID.log**, where **JOBID** is the unique id of the distribution or replication job.

- **Mail File**– Mails the file list to the specified recipients.
 - **Template File** – Mandatory.
The full path to a custom template must be defined. Templates are explained in detail in the following section.
 - **Target Directory** – Optional.
Use this feature to keep a copy of the File List on the Center. Specifies the directory on the Center where the file list will be created. Do not specify a file name, as the file list will be created as **transfer_JOBID.log**. *JOBID* is the unique id of the distribution or replication job.
 - **Recipients** – Mandatory.
Specify the e-mail addresses of those receiving the file list. Multiple email addresses should be separated by a comma.
e.g. user1@company.com,user2@company.com
- **Execute File**– Creates a Windows batch file or UNIX script based on the specified template file. Since each line in the file list represents a unique file that was transferred, when the batch file / script is executed on the center, each line is executed in sequence. Some examples of using the **Execute File List** option are provided below.
 - **Template File** – Mandatory.
The full path to a custom template must be defined. Templates are explained in detail in the following section.
 - **Target Directory** – Optional.
Use this feature to keep a copy of the File List batch file or script on the Center. Specifies the directory on the Center where the file list will be created. Do not specify a file name, as the file list will be created as **transfer_JOBID.bat** on windows, and **transfer_JOBID.sh** on UNIX. *JOBID* is the unique id of the distribution or replication job.

For the Execute option, it is ideal to leave this field empty and use the default Job directory. That way the intermediate file will be purged along with the job itself.

3. Template File

File List Concept - A Multi-Step Process

During the transfer stage, a detailed **Transfer** report is created, containing information about the files that were transferred in the replication process.

Using a text **Template** file, a **File List** file is created on the Center, combining the information from the **Transfer Report**, with the format defined in a **Template** file. The Template file can be easily created in accordance with a user-defined format.

In the **File List** file, each line represents information about a single transferred file. Once the File List file is created, it can then be stored locally on the Center, e-mailed to multiple recipients, or executed as a Windows batch file or UNIX script on the R-1 Center.

From an R-1 perspective, the order of events is as follows:

1. The user creates a Template file, using environment variables provided by R-1.
2. The job is run with **Procedures – File List** option set. The job creates the log of transferred files (Detailed or Summary Transfer Report).
3. Upon job completion the job uses the Template file to create the File List file where each line represents a file that was transferred.
4. The File List file is emailed or executed on the Center.

NOTE: The whole process, starting at creating the Template file, until the File List file is stored or executed, is performed exclusively on the Center.

NOTE: Set the properties **Transfer Report Style** under the **Edge General** Tab to either **Summary** or **Detailed**. Any other report style will result in an **empty** file list being created.

The template file dictates the format of the File List that will be created or executed (as explained above).

Creating a Template File

The template file format is as follows:

HEADER
multiple lines written to the file once
BODY
one line written to the file for each file transferred
FOOTER
multiple lines written to the file once

- Section names are case sensitive

- HEADER section may contain multiple lines.
- HEADER section is optional. If omitted, there's no need to write the BODY header. The first line in the file will be regarded as the template line.
- If BODY is specified, FOOTER must exist (but can be empty).
- BODY can contain only one (1) line. Since each line in the file list represents a unique file transferred, the line in the template file is a representation of what each line in the file list will look like.
- The template file line can contain any text, plus the variables listed below. When the file list is created after a deployment / replication, the variables are replaced on a line-by-line basis with the information about the given file.
- A template line can use pre-defined variables for information about the transferred file:

```
RW_JOB_UNIQUE_ID          Job ID
RW_JOB_NAME               Job Name
RW_JOB_COMPLETION_STATUS The job's completion status: SUCCESS|ABORT|ERROR
```

```
%j - job name
%c - source hostname
%n - destination node/hostname
%z - file size (bytes)
%s - source file specification
%d - target file specification
%b - file transfer start time
%e - file transfer start time (SQL time format)
%t - file transfer completion time
%q - file transfer completion time (SQL time format)
```

NOTE: In UNIX, use `$x` while in Windows, use `%x`

Execution Variables

The following variables may be used ONLY in the File List Execute option:

```
RW_JOB_UNIQUE_ID          Job ID

RW_JOB_NAME               Job Name
RW_JOB_COMPLETION_STATUS The job's completion status:
                          SUCCESS|ABORT|ERROR

RW_JOB_DEMANDING_USER    The job's demanding user name
RW_JOB_DEMANDING_DOMAIN The job's demanding domain
```

NOTE: These variables can be used only in Execution command and will not be replaced while using the Email List option.

Examples of Template Files

The following chapter gives multiple examples of Template files that can be used in R-1.

Customized Reports

The following template file can be used to create a report different from the Transfer Report.

Windows:

```
Source:<[%c]%s> Target:<[%n]%d> Size(bytes):<%z> Transfer Ended at:<%t>
```

UNIX:

```
Source:<[$c]$s> Target:<[$n]$d> Size(bytes):<$z> Transfer Ended at:<$t>
```

A file list created with this template would then take on the following format:

```
Source:<[PAUL]C:\source\expenses.xls>  
Target:<[RINGO]C:\target\expenses.xls> Size(bytes):<147456> Transfer  
Ended at:< Mon Feb 22 17:35:07 2010>  
Source:<[PAUL]C:\source\contract.doc>  
Target:<[RINGO]C:\target\contract.doc> Size(bytes):<1247458> Transfer  
Ended at:< Mon Feb 22 17:35:07 2010>
```

UNIX File Lists with Parameters

The following template file can be used on UNIX using R-1 environment variables.

```
HEADER  
BODY  
file %s  
FOOTER  
comp_status $RW_JOB_COMPLETION_STATUS  
job_name $RW_JOB_NAME
```

Examples of Execute Template Files

An execute template file takes on the same form, and uses the same variables as a normal template file, but must contain commands that can be executed from a command-prompt / shell.

The following are examples of execute template files that perform a given task:

Deleting Source Files after Transfer

```
del "%s"
```

Would create the following batch file on the Center, which would then be executed when the job is complete:

```
del "C:\source\expenses.xls"
```

```
del "C:\source\contract.doc"
```

Command

In this example the attributes of each of the transferred files and append them to a file:

```
attrib "%s" >> C:\attributes.txt
```

Would create the following batch file on the Center, which would then be executed when the job is complete:

```
attrib "C:\source\expenses.xls" >> C:\attributes.txt
attrib "C:\source\contract.doc" >> C:\attributes.txt
```

Formatted HTML Lists

In this example, the file list takes on the following form:

```
<li><a href=%s>%s</a> Transferred at %t
```

Resulting in the following file that could be viewed in a web browser:

```
<li><a href="C:\source\expenses.xls">"C:\source\expenses.xls"</a>
Transferred at Mon Feb 22 16:45:07 2010
<li><a href="C:\source\expenses.xls">"C:\source\expenses.xls"</a>
Transferred at Mon Feb 22 16:45:07 2010
```

Executing Scripts and File Lists

To even enhance the user's power at the exit stage, this file can also include header and footer that are in free format, allowing calling other batch file or contain executable commands. This allows both File List and Post Commands as the exit procedure.

The File List Template can then look like this:

```
HEADER
echo Start of Job %RW_JOB_UNIQUE_ID% %RW_JOB_NAME%>> c:\lists\file_list.txt
echo Completion Status: %RW_JOB_COMPLETION_STATUS% >> c:\lists\file_list.txt
d:\repliwebbatchfiles\checkstatus.bat
BODY
echo %c,%n,%z,%t,%q >> c:\lists\file_list.txt
FOOTER
echo End of Job %RW_JOB_UNIQUE_ID%>> c:\lists\file_list.txt
```

At the exit stage, a batch file that is based on the template file is created in the job directory. The batch file is then executed, writing the Job ID, Name and Completion Status using Environment variables that handle the job's completion status (d:\repliwebbatchfiles\checkstatus.bat) and write the information about the transferred files to the log file (d:\lists\file_list.txt). At the end of the list, a line is added to mark the end of the list using the Environment Variable holding the Job ID.

The output file (c:\lists\file_list.txt) will then look like this:

```
Start of Job 277 Weekly Update
Completion Status: SUCCESS
srv,target_srv,1263,Mon Feb 22 10 51 35 2010,2010-02-22 10:51:35
```

```
srv,target_srv,4320,Mon Feb 22 10 51 35 2010,2010-02-22 10:51:35
srv,target_srv,126344,Mon Feb 22 10 51 35 2010,2004-02-22 10:51:35
srv,target_srv,4320,Mon Feb 22 10 51 35 2010,2004-02-22 10:53:40
End of Job 277
```

Inserting File Transfer Statistics into SQL Database

By using the following template file, in addition to the Mail File List feature, file transfer statistics can be inserted into an SQL database.

The e-mail account is POP'ed by the SQL server. The lines in the file list represent 'insert' commands into the SQL database.

```
INSERT INTO File_list
(Source_Hostname, Destination_Hostname, Filesize, Source_Filespec,
Target_Filespec, Transfer_Completion_Time, SQL_Time_Format_Transfer_Comple
tion_Time)
VALUES ('%c', '%n', '%z', '%s', '%d', '%t', '%q')
```

For any additional information, contact us at support.repliweb.com