Attunity RepliWeb PowerShell API Developer's Guide

Software Version 6.0

November 27, 2012

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. Introduction

This document describes the Attunity RepliWeb PowerShell API. It provides detailed explanations of the Attunity RepliWeb objects, methods, and properties required to operate and control Attunity RepliWeb functionality and obtain information regarding Attunity RepliWeb jobs using PowerShell.

PowerShell is an easy-to-use object-oriented command-line shell and scripting language designed to provide IT professionals with greater system control, operation capabilities, and automation.

Requirements

You must have the following resources to use PowerShell on Attunity RepliWeb:

- R-1 v4.1 (or later) Deployment Console

- Deployment computer with Microsoft .NET Framework (version 2.0 or later) installed

- PowerShell (version 1.0 or later)

2. Installation Guidelines

Please refer to the relevant scenario below for installation guidelines.

Installing Attunity RepliWeb on Environment with PowerShell

When installing Attunity RepliWeb on PowerShell-installed environments, the following automatic PowerShell-related processes take place:

- Attunity RepliWeb installation copies relevant PowerShell DLLs to `~\RepliWeb\R1\API\PowerShell`

- Attunity RepliWeb installation performs PowerShell DLL registration

Installing Attunity RepliWeb on Environment without PowerShell

When installing Attunity RepliWeb on environments that do not include PowerShell, Attunity RepliWeb will copy relevant PowerShell DLLs to `~\RepliWeb\R1\API\PowerShell`. However, since PowerShell is not installed, Attunity RepliWeb will not perform PowerShell DLL registration.

Installing PowerShell on Environment with Attunity RepliWeb

After installing PowerShell on Attunity RepliWeb-installed environments, you must manually register the PowerShell DLLs.

This can be done using cmd or PowerShell. You can redirect the output to a file by using `" > filename.log"` at the end of the command.

In order to use these DLLs, register the Attunity RepliWeb snap-in by running the following .NET installation utility:

- %windir%\Microsoft.NET\Framework\v2.0.50727\installutil.exe ~\Repliweb\R1\API\PowerShell\ps4r1.dll

3. Attunity RepliWeb Snap-in

When opening a new PowerShell instance, use the `Add-PSSnapin` cmdlet to add the required Attunity RepliWeb snap-in. Attunity RepliWeb snap-in adds the `R1Session` cmdlet into your PowerShell session.

On each new PowerShell session, manually run the `add-pssnapin r1_ps_api` command.

Manually Adding Attunity RepliWeb Snap-in

When opening a new PowerShell instance, you must use the `Add-PSSnapin r1_ps_api` cmdlet to add the required Attunity RepliWeb snap-in. Attunity RepliWeb snap-in adds the `Get-R1Session` cmdlet into your PowerShell session.

For help on the usage of `Get-R1Session`, type the command: `Get-Help GetR1Session`.

Automating the Attunity RepliWeb Snap-in Process

You may customize your profile to run the "`add-pssnapin r1_ps_api`" command automatically at the start of each new PowerShell session, as follows:

Note: Prior to performing this procedure, make sure the **My Documents\WindowsPowerShell** folder exists, otherwise create it.

To automate the Attunity RepliWeb Snap-in process:

1. Access the **My Documents\WindowsPowerShell** folder.

2. Locate the **profile.ps1** file, if it does not exist, create it.

3. In the profile.ps1 file, type "**add-pssnapin r1_ps_api**" and save the file.

   `Add-pssnapin r1_ps_api` will run automatically whenever you open a new PowerShell session.

4. Handling Escape Characters

When creating a command string using Attunity RepliWeb PowerShell, please take the following into consideration:

- Precede any special character (`$()*+.[]?\/^{}|"` etc.) with a backquote (`` ` ``).

  For example, `-edge_password=`"`@aB2q`$WiM`"`

- Enclose in quotation marks (`""`) any qualifier value that contains spaces.

  For example, `-name=`"My First Job`"`

- Use double backslashes (`\\`) when specifying paths which are enclosed in quotation marks (`""`). For example, `-target_dir=`"C:\\my directory`"`

5. Objects

This chapter provides low-level details of the Attunity RepliWeb objects used on the PowerShell platform, including methods, properties, syntax definitions and code samples for each method.

Note: When running the code directly on PowerShell, each command should be ended in the same line. From script files, it is possible to run commands in blocks.

The following Attunity RepliWeb objects are available:

- R1Session
- R1Job
    - ○ Replication Job
    - ○ Distribution Job
    - ○ Replication Rollback Job
    - ○ Distribution Rollback Job
- QueryFilter
- R1JobInfo

R1Session

This is the first object presented to you whenever you start a RepliWeb PowerShell session. Use R1Session to connect and submit requests to Attunity RepliWeb.

This object is loaded by running the `Get-R1Session` cmdlet, and includes the following properties and methods:

R1Session Properties

Note: These properties may be set by the user, or retrieved at a later time.

Type	Name	Description	Valid Values
String	`Center`	The name or IP of the Center being accessed.	IP address / computer name
String	`Domain`	The valid domain name for the user-specified account.	Domain name
String	`Password`	The password for the user-specified account.	
Integer	`Port`	The port for Console-Center communication.	
QueryFilter	[QueryFilter](#)	The query parameter settings before retrieving the job list.	
String	`ScrambledPassword`	The valid scrambled password for the user-specified account.	Use the [ScramblePassword](#) method
Boolean	`SSL`	Indicates if the connection is through SSL.	
String	`UserName`	Username that connects to the Center.	
Enum Views	`View`	Display view for the jobs list.	Flat Hierarchy

R1Session Methods

Return Values	Name	Description
Boolean	Connect	Starts a session on the Center. You can connect in three ways: `Connect()` `Connect(ConnectInfo)` `cmdlet qualifiers`
Void	Disconnect	Terminates a PowerShell session by logging it off from the Attunity RepliWeb server.
R1Job[]	GetJobList	Retrieves all of the jobs, according to `QueryFilter`, from the R1Session database.
String	ScramblePassword	Returns encrypted passwords for greater security.
Void	SetDeploymentMode	Allows granting specific users with access and permissions to jobs. If the user is in the job's authorized users/groups list, The user can view, monitor and control jobs if the user is included in the jobs' authorized users/groups list.
Void	SetView	Displays the required jobs according to the `QueryFilter` settings in one of two modes.
R1Job	Submit	Submits a new job.
R1Job	DeploymentSubmit	Submits a new MSI Deployment job.
Void	Purge	Purges jobs in the specified states.

Connect

You may connect to the Attunity RepliWeb server in three ways, as follows:

Using Connect()

Remarks

Before invoking a `Connect()` call, set: `Center, UserName`, and either `Password` or `ScrambledPassword`.

Syntax

```
R1Session.Connect()
```

Example

Starts an Attunity RepliWeb session after setting the session's properties:

```
$R1Session = Get-R1Session
$pass = $R1Session.ScramblePassword("plain text password")
$R1Session.Center = "localhost"
$R1Session.User = "Administrator1"
$R1Session.ScrambledPassword = $pass
$R1Session.Domain = "Domain1"
$R1Session.Connect()
```

Using Connect(ConnectInfo)

Syntax

```
R1Session.Connect( String center,
                   String user,
                   String password,
                   String domain,
                   String ScrambledPassword,
                   Boolean useSSl,
                   Int32 port ) }
```

Parameters

- **String Center** – The name or IP of the Center being accessed.
- **String User** - The username that connects to the Center.
- **String Password** - The password for the user-specified account.
- **String Domain** - The valid domain name for the user-specified account.
- **String ScrambledPassword** - The valid scrambled password for the user-specified account.
- **Boolean SSL** - Indicates if the connection is through SSL.
- **Int32 Port** - The port for Console-Center communication.

Example

Starts an Attunity RepliWeb session by including the relevant parameters in the connect call:

```
$R1session = Get-R1Session
$R1Session.Connect( "localhost", "Administrator1", "Password1",
"", "", $false , "2837" )
```

Using cmdlet qualifiers

Syntax

```
Get-R1Session
-Center [Center IP]
-User [User]
-Password [Password]
-Domain [Center domain]
-ScrambledPassword [Scrambled Password]
-UseSSL [$true/$false]
-port [Center port]
```

Example

Starts an Attunity RepliWeb session by requesting the user's credentials during runtime:

```
$cred = Get-Credential
$R1Session = Get-R1Session -Center [localhost] -credential $cred
```

Disconnect

Syntax

```
R1Session.Disconnect()
```

GetJobList

Remarks

The History and ChildJobs properties become available when
QueryFilter.IncludeHistoryJobs is set to "true".

Syntax

```
R1Session.GetJobList()
```

Example

Resets the QueryFilter settings and returns scheduled Distribution jobs:

```
$R1Session.QueryFilter.Reset()
$jobs = $R1Session.GetJobList()
$dist = $jobs | where{ $_.Type -match "DIST" -and $_.state -match
"sched" }
```

ScramblePassword

Syntax

```
R1Session.ScramblePassword( string plain_text_password )
```

Example

Set a scrambled password in the session:

```
$scramble = $session.ScramblePassword( "my password" )
$R1Session.ScrambledPassword = $scramble
```

SetDeploymentMode

After setting `DeploymentMode`, use the [GetJobList](#) method to retrieve the jobs that can be accessed by the user who was set through the `SetDeploymentMode` method.

Remarks

`SetDeploymnetMode` may be called for only one user at a time.

Syntax

```
R1Session.SetDeploymentMode( Boolean UseDeploymentMode,
                             string UserName,
                             string Domain )
```

Parameters

- **UseDeploymentMode** - Indicates to use deployment mode. This parameter must be used in conjunction with `UserName` and `domain`.
- **UserName** - Indicates the name of the Deployment user.
- **Domain** - When `UserName` is part of a Windows domain, this qualifier must be added. If `UserName` is not part of a Windows domain (i.e. a UNIX user, or a local user on a Windows machine), this qualifier can be omitted.

Example

Sets Deployment Mode to User1 and Domain1:

```
$R1Session.SetDeploymentMode( $true, "User1", "Domain1" )
```

Disables Deployment Mode:

```
$R1Session.SetDeploymentMode( $false, $null, $null )
```

SetView

Remarks

The parameters are case-insensitive.

In order to view spawned jobs as well, in the QueryFilter object, set the `IncludeHistoryJobs` and `IncludeChildJobs` fields to "**True**".

Syntax

```
R1Session.SetView(string view)
```

Parameters

- **Flat** - Retrieves all existing jobs on the server
- **Hierarchy** - Retrieves only scheduled jobs and individual (jobs with no `ParentID` value) jobs

Example

Sets view to Flat mode:

```
$R1Session.Setview( "flat" )
```

Sets view to Hierarchy mode and retrieves scheduled job execution history:

```
$R1Session.Setview( "hierarchy" )
$R1Session.QueryFilter.IncludeHistoryJobs = $true
$R1Session.QueryFilter.JobState = "R1_API_SCHEDULER_JOB"
$sched = $session.GetJobList()
```

Submit

Remarks

For more information, refer to the Submit Qualifiers section in the *Attunity RepliWeb User Guide*.

Syntax

```
R1Session.Submit(String Submit_CLI_qualifiers)
```

Example

Submits a Replication job:

```
$qual = " -name=replication -edge=edge_addr -user=username
-edge_password=secret -type=replication -source_dir=c:\source
-target_directory=c:\target"
$R1Session.Submit($qual)
```

DeploymentSubmit

Remarks

For more information, refer to the Deployment_Submit Command section in the *Attunity RepliWeb User Guide*.

Syntax

```
R1Session.DeploymentSubmit(String
DeploymentSubmit_CLI_qualifiers)
```

Example

Installs MSI file `ABC.msi` with installation-related files and parameters:

```
$qual = "-name=MSI_job -edge=edge_name -user=username
-edge_password=password -type=replication -msi_action=install
-msi_list=`"`[c:\\ABC.msi,,true,c:\\file1.dll;c:\\file2.txt,true,
param1=value param2=value,true`]`""
$R1Session.DeploymentSubmit($qual)
```

Purge

Syntax

```
R1Session.Purge (String Purge_CLI_Qualifiers)
```

Example

Purges completed jobs without confirmation:

```
R1Session.Purge("-state=completed -noconfirm")
```

R1Job

Replication through Attunity RepliWeb is handled by jobs. Each job type is responsible for a different deployment process.

R1Job is an abstract object that comprises the attributes and functions (members) shared by the different Attunity RepliWeb job types. Each job type includes its own set of properties and methods, and inherits a set of generic ones from the R1Job object.
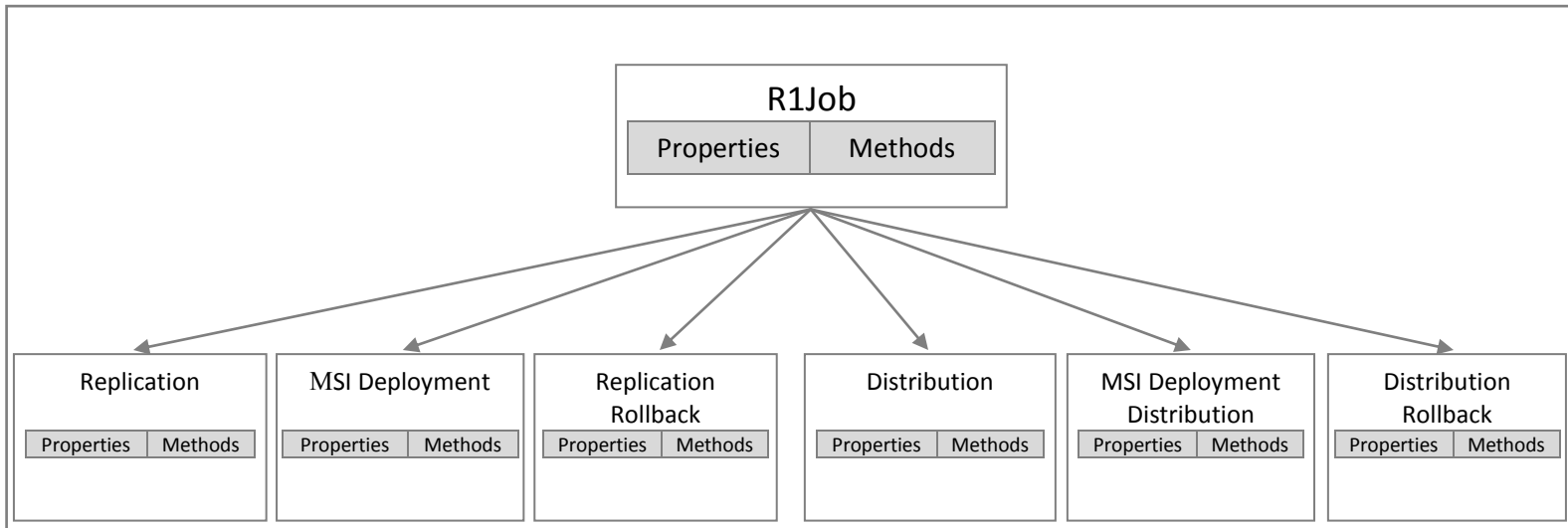
Each job type includes its own specific set of methods and properties and inherits a shared set of general properties and methods from the R1Job object.

For the job-specific properties and methods, refer to the relevant job:

- Replication Job
- Distribution Job
- MSI Deployment Job
- MSI Deployment Distribution Job
- Replication Rollback Job
- Distribution Rollback Job

R1Job Properties

Type	Name	Description	Valid Values
String	Center	Center IP or name.	
String	Description	Job description, as entered by the user.	
DateTime	EndTime	Job's end time.	DateTime object containing the job-submitted timestamp.
String	ExistingReports	Job status reports.	general transfer pre_center pre_edge post_center post_edge deleted_files excluded_files rollback_recorder rollback_reloader mb_transfer edge_general sharepoint_transfer solution_deployment_transfer
Integer	Id	Unique system-generated job ID.	
String	Name	Job name, as entered by the user or inherited from the parent job.	
Scheduling	Scheduling	Properties related to the scheduling of Replication and Distribution jobs	
Enum R1_JOB_STATES	State	Current job state. For example, Aborted or Completed.	R1_API_SUBMITTED R1_API_INITIALIZING R1_API_RUNNING R1_API_COMPLETED_SUCCESS R1_API_COMPLETED_ERROR R1_API_INTERUPTTED R1_API_ABORTED R1_API_SCHEDULER_JOB R1_API_HOLD_RECOVERING R1_API_WAIT_FOR_CONFIRM R1_API_HOLD

Type	Name	Description	Valid Values
			Note: For R1_API_HOLD, API_SCHED_HOLD_MODE indicates the reason for holding the job, as follows: **1** – Execution **2** - Priority **3** – Administrative
String	SystemMessage	Job-specific messages that are generated by Attunity RepliWeb while the job is running.	
String	TagsList	List of user-created job tags. **Note:** Tags are separated by a ";".	
Enum R1_OPERATION_TYPES	Type	Job's type.	R1_DISTRIBUTION, R1_EDGE, R1_REPLICATION, R1_DISTRIBUTION_ROLLBACK, R1_EDGE_ROLLBACK, R1_REPLICATION_ROLLBACK

R1Job Methods

Return Values	Name	Description
R1Job	Abort	Aborts the job.
R1Job	Confirm	Confirms the job.
void	Delete	Deletes the job
String array	GetExistingReports	Gets the list of the job's available reports.
String array or void	GetReport	Generates the job's report.
R1Job	Hold	You may use Hold in two ways: Hold - Puts a job on hold indefinitely. Hold(DateTime time) - Puts a job on hold until the specified date and time.

Return Values	Name	Description
String(XML)	**Preview**	Returns the job's comparative snapshot for the user to review and edit before confirming the job using the `Confirm` method.
R1Job	**Refresh**	Gets the job's updated state and info.
R1Job	**Resume**	Reactivates a job that is on hold.

Abort

Syntax

```
R1Job.Abort()
```

Example

Aborts all scheduled jobs:

```
$R1Session.QueryFilter.JobState = "R1_API_SCHEDULER_JOB"
$jobs = $session.GetJobList()
foreach( $R1Job in $jobs ){ $R1Job.Abort() }
```

Confirm

You may confirm and activate a new job in two ways, as follows:

Using Connect Syntax

Syntax

```
R1Job.Confirm()
```

Example

Confirms all jobs that are pending confirmation in Attunity RepliWeb:

```
$R1Job = $session.GetJobList()
$waitList = $R1Job | where{ $_.State -match "CONFIRM" }
foreach ( $Job in $waitList ) { $Job.Confirm() }
```

Using Confirm(String ComparativeSnapshotXML)

Syntax

```
R1Job.Confirm(String ComparativeSnapshotXML)
```

Example

Confirms all jobs pending Edge:

```
$jobs = $session.GetJobList()
$waitList = $jobs | where{ $_.State -match "CONFIRM" }
foreach ( $job in $waitList )
{ [xml]$Preview = $job.Preview()
$edge = $Preview.RootConfirmation.EdgeName
If ( $edge.Equals("1.1.1.1") )
{ echo "Confirming job: " $job.Id
$job.Confirm( $Preview.Get_outerXML() ) } }
```

Delete

Syntax

```
R1Job.Delete()
```

Example

Removes all jobs that completed successfully:

```
$R1Session.QueryFilter.JobState = "R1_API_COMPLETED_SUCCESS"
$jobCompletedSuccessfully = $R1session.GetJobList()
foreach( $index in $jobCompletedSuccessfully){$index.Delete()}
```

GetExistingReports

Syntax

```
R1Job.GetExistingReports()
```

Example

Retrieves a list of report types available for a job:

```
$ExistingReport = $job.GetExistingReports()
```

GetReport

Remarks

Use the GetExistingReports return values as parameters in GetReport.

Only reports that are returned by [GetExistingReports](#) method can be viewed.

Syntax

```
R1Job.GetReport(string ReportType,
                string FileNamePath,
                Boolean OpenNotepad)
```

Parameters

`ReportType` – Taken from `GetExistingReports`.
`FileNamePath` – Relative or absolute path, file name must be included.
`OpenNotepad` – Indicates if to open Windows Notepad with the returned report.

Example

Retrieves the job's "General" report, and opens a notepad file:

```
$R1Job.GetReport( "general", "d:\reports\report.txt", $true )
```

Retrieves the job's "General" report and stores it in a variable:

```
$jobReport = $R1Job.GetReport( "general", $false )
```

Hold

You may put a job on hold in two ways, as follows:

Using Hold()

Syntax

```
R1Job.Hold()
```

Example

Holds all scheduled jobs:

```
$R1Session.QueryFilter.JobState = "R1_API_SCHEDULER_JOB"
$jobs = $R1session.GetJobList()
foreach( $R1Job in $jobs ){ $R1Job.Hold() }
```

Using Hold(DateTime time)

Syntax

```
R1Job.Hold(DateTime time)
```

Parameters

DateTime `Time`

Example

Holds a job for 10 days:

```
$scheduledJobs = $R1session.GetJobList() | where{ $_.state -match
'schedule' }
$time = Get-Date
$time = $time.AddDays( 10 )
foreach($R1Job in $scheduledJobs ){ $R1Job.Hold($time ) }
```

Preview

Syntax

```
R1Job.Preview()
```

Example

Displays all Edges that are pending confirmation:

```
[xml]$Preview = $job.Preview()
echo $Preview.RootConfirmation.EdgeName
```

Refresh

Syntax

```
R1Job.Refresh()
```

Example

Queries the current job's state, aborts, refreshes, and then queries the job's new state:

```
if( $R1Job.state –match 'schedule' )
 {
   $job.Abort()
  $job = $job.Refresh()
write-output $R1Job.State
}
```

Resume

Syntax

```
R1Job.Resume()
```

Example

Resumes held jobs:

```
$R1Session.QueryFilter.JobState = "R1_API_HOLD"
$jobs = $session.GetJobList()
foreach($R1Job in $jobs ){ $R1Job.Resume() }
```

ReplicationJob

Replication includes the following properties and methods.

ReplicationJob Properties

Type	Name	Description	Valid Values
Enum RDS_JOB_DIRECTION	Direction	Job's replication direction, "Upload" or "Download".	RDS_API_DOWNLOAD, RDS_API_UPLOAD
String	Edge	Edge IP or name.	
String	ParentID	Unique ID of the job that spawned this job.	
String	GroupID	Unique ID of the group this job belongs to.	
Enum HOLD_MODES	HoldMode	Indicates if a job is on hold.	NONE EXECUTION_TIMEFRAME PRIORITY ADMINISTRATIVE WAIT_FOR_COMMIT
DateTime	HoldUntil	Indicates when a held job will be resumed.	
ReplicationInfo	Info	Information about the job's settings.	
ReplicationJob[]	History	Stores children jobs that were spawned by this scheduled Replication job.	
Boolean	RollbackEnabled	Indicates if rollback is enabled for a job.	
Scheduling	Scheduling	Job's running schedule.	For example, Daily, Weekly.
String	SourceDirectory	Source directory.	
String	TargetDirectory	Target directory.	

Type	Name	Description	Valid Values
Enum R1_REPLICATION_ JOB_STAGES	`Stage`	Current stage in a job's lifecycle. R1_API_CHECK_LICEN SE.	R1_API_INIT = 0, R1_API_START, R1_API_GET_REMOTE_INFO, R1_API_CHECK_LICENSE, R1_API_MAKE_REMOTE_WORKI NG_DIR , R1_API_PRE_COMMANDS, R1_API_MAKE_LISTS, R1_API_COMPARE_LISTS, R1_API_TRANSFORM_LISTSR1_ API_PRE_TRANSFER_DELETION, R1_API_SPLIT_REGULAR_TRANS FER_LIST =11, R1_API_TRANSFER_2_TARGET, R1_API_POST_TRANSFER_DELET ION, R1_API_POST_COMMANDS, R1_API_CLEAN_REMOTE_END, R1_API_CLEAN_LOCAL_END , R1_API_END
DateTime	`SubmitTime`	Job's "submit" time.	DateTime object containing the job-submitted timestamp.

Replication Methods

Return Values	Name	Description
R1Job	Commit	Commits jobs that are set to "Commit on User Request".
R1Job	Resubmit	Resubmits an existing job by modifying the job's settings using the `Submit()` method's qualifiers. See the Submit Command section in the *Attunity RepliWeb User Guide*.
R1Job	RollbackAll	Rolls back the target directory to its state before replication.
R1Job	RollbackToInstance	Rolls back to a specific backup instance.
R1Job	RollbackToTime	Rolls back to a specific backup time.
R1Job	SubmitNow	Submits a scheduled job immediately.
R1Job	Modify	Allows modifying a job's properties.

Commit

Syntax

```
R1Job.Commit()
```

Example

Commits a Replication job pending for user approval where the Edge is '1.1.1.1:

```
If ( $R1Job.HoldMode –match 'WAIT_FOR_COMMIT' –and ( $R1Job.edge –
eq '1.1.1.1' ) ) { $R1Job.Commit() }
```

Resubmit

Syntax

```
R1Job.Resubmit
```

Parameters

string CLI Qualifiers

Example

Resubmits a Replication job without changing any properties

```
$R1Job.Resubmit("")
```

Resubmits a Replication job by modifying the Edge

```
$qualifiers = "-edge=other_address –user=other_user
–password=other_password"
$R1Job.Resubmit($qualifiers)
```

RollbackAll

Syntax

```
R1Job.RollbackAll()
```

Example

Rolls back all instances of a job with an Edge whose IP address is 1.1.1.1:

```
If ( $R1Job.type –match 'replication –and
$R1Job.Info.RollbackEnabled –and $R1Job.edge –eq '1.1.1.1') {
$job.RollbackAll() }
```

RollbackToInstance

Syntax

```
R1Job.RollbackToInstance(int instance)
```

Example

Performs rollback to instance 5:

```
If ( $R1Job.Info.RollbackEnabled ) { $R1Job.RollbackToInstance(5) }
```

RollbackToTime

Syntax

```
R1Job.RollbackToTime(DateTime time)
```

Example

Rolls the job back 10 days:

```
$time = Get-Date
$time = $time.AddDays( -10 )
If ( $R1Job.Info.RollbackEnabled ) { $R1Job.RollbackToTime($time)
```

SubmitNow

Syntax

```
R1Job.SubmitNow()
```

Example

Checks if the job is scheduled and then submits:

```
If ( $R1Job.State -match 'SCHEDULER' ) { $R1Job.SubmitNow() }
```

Modify

Syntax

```
R1Job.Modify()
```

Example

```
If( $R1Job.type -match 'replication' -and $job.state -match
'scheduler')
{
$R1Job.Modify("-password=new_password")
```

```
    }
```

DistributionJob

Distribution includes the following properties and methods.

DistributionJob Properties

Type	Name	Description	Valid Values
DistributionJob[]	History	Stores children jobs that were spawned by this **scheduled** Replication job. **Note:** History is available when `QueryFilter.IncludeHistoryJobs` is set to "true".	
String	ParentID	Unique ID of this job's parent job.	
Enum HOLD_MODE	HoldMode	Indicates if a job is on hold.	NONE EXECUTION_TIMEFRAME PRIORITY ADMINISTRATIVE WAIT_FOR_COMMIT
DateTime	HoldUntil	Indicates when a held job will be resumed.	
DistributionInfo	Info	Information about the job's settings.	
ReplicationJob[]	ChildJobs	Stores all Center-Edge Replication jobs spawned by this Distribution job. **Note:** `ChildJobs` is available when `QueryFilter.IncludeChildJobs` is set to "true".	
Boolean	RollbackEnabled	Indicates if rollback is enabled for this job.	

Type	Name	Description	Valid Values
Scheduling	Scheduling	Job's running schedule.	
String	SourceDirectory	Source directory on the Center.	
String	TargetDirectory	Target directory on the Edge.	
Enum R1_DISTRIBUTION_JOB_STAGES	Stage	Current stage in a job's lifecycle.	R1D_API_INIT, R1D_API_VALIDITY, R1D_API_LICENSE_CHECK, R1D_API_PRE_DISTRIBUTION_DOWNLOAD, R1D_API_PRE_COMMANDS, R1D_API_SUBMIT_CORE_JOBS, R1D_API_GATHER_REMOTE_INFO, R1D_API_ENABLE_CORE_PRE_CMD, R1D_API_WAIT_TO_CORE_PRE_CMD, R1D_API_SUBMIT_LOCAL_ML, R1D_API_TRANSFER, R1D_API_CHECK_UNICAST_TRANSFER, R1D_API_ENABLE_CORE_POST_CMD, R1D_API_CORES_COMPLETION, R1D_API_POST_COMMANDS, R1D_API_CLEAN_LOCAL_END, R1D_API_END
DateTime	SubmitTime	Job's "submit" time.	DateTime object containing the job-submitted timestamp.

Distribution Methods

Return Values	Name	Description
R1Job	Commit	Commits the job in case it was set to "Commit on User Request".
R1Job	Resubmit	Resubmits an existing job by changing the required settings through the included qualifiers.
R1Job	RollbackAll	Rolls back all backed up instances.

Return Values	Name	Description
R1Job	RollbackToInstance	Rolls back to a specific backup instance.
R1Job	RollbackToTime	Rolls back to a specific backup time.
R1Job	SubmitNow	Submits a scheduled job immediately.
R1Job	Modify	Allows modifying a job's properties.

Commit

Syntax

```
R1Job.Commit()
```

Example

Commits a distribution job whose source is "C:\source", and is pending user approval:

```
If ($R1Job.HoldMode –match 'WAIT_FOR_COMMIT' –and
$R1Job.SourceDirectory -eq 'C:\source' )
{
 $R1Job.Commit()
}
```

Resubmit

Syntax

```
R1Job.Resubmit
```

Parameter

(String Qualifiers)

RollbackAll

Syntax

```
R1Job.RollbackAll()
```

Example

```
If ($R1Job.Info.RollbackEnabled ) { $R1Job.RollbackAll() }
```

RollbackToInstance

Syntax

```
R1Job.RollbackToInstance(int instance)
```

Example

Performs rollback to instance 5:

```
If ($R1Job.Info.RollbackEnabled ) {
$R1Job.RollbackToInstance(5) }
```

RollbackToTime

Syntax

```
R1Job.RollbackToTime(DateTime time)
```

Example

```
$time = Get-Date
$time = $time.AddDays( -10 )
If ($R1Job.Info.RollbackEnabled ) {
$R1Job.RollbackToTime($time) }
```

SubmitNow

Syntax

```
R1Job.SubmitNow()
```

Example

Checks if the job is scheduled and then submits:

```
If ( $R1Job.state –match 'SCHEDULER' ) { $R1Job.SubmitNow() }
```

Modify

Syntax

```
R1Job.Modify
```

Example

```
If($R1Job.type -match 'replication' -and $R1Job.State -match
'scheduler') {
$R1Job.Modify("-password=new_password")
}
```

MsiJob

MsiJob includes the following properties and methods.

Note: The Rollback option is not available for MSI Deployment jobs.

MsiJob Properties

Type	Name	Description	Valid Values
Enum RDS_JOB_DIRECTION	Direction	Job's replication direction, "Upload" or "Download".	RDS_API_DOWNLOAD, RDS_API_UPLOAD
String	Edge	Edge IP or name.	
String	ParentID	Unique ID of the job that spawned this job.	
String	GroupID	Unique ID of the group this job belongs to.	
Enum HOLD_MODES	HoldMode	Indicates if a job is on hold.	NONE EXECUTION_TIMEFRAME PRIORITY ADMINISTRATIVE WAIT_FOR_COMMIT
DateTime	HoldUntil	Indicates when a held job will be resumed.	
ReplicationInfo	Info	Information about the job's settings.	
ReplicationJob[]	History	Stores children jobs that were spawned by this scheduled Replication job.	
Boolean	RollbackEnabled	Indicates if rollback is enabled for a job.	
Scheduling	Scheduling	Job's running schedule.	For example, Daily, Weekly.

Type	Name	Description	Valid Values
R1_MSI_ACTION_TYPES	MsiAction	MSI deployment operation to apply to the target.	R1_MSI_ACTION_INSTALL R1_MSI_ACTION_UPGRADE_REPAIR R1_MSI_ACTION_REMOVE
String	MsiConfiguration	Specify each MSI file, including deployment settings, you wish to deploy on the target.	XML formatted string For the XML file's format, refer to Appendix C: MSI Configuration XML.
Enum R1_REPLICATION_JOB_STAGES	Stage	Current stage in a job's lifecycle. R1_API_CHECK_LICENSE.	R1_API_INIT = 0, R1_API_START, R1_API_GET_REMOTE_INFO, R1_API_CHECK_LICENSE, R1_API_MAKE_REMOTE_WORKING_DIR , R1_API_PRE_COMMANDS, R1_API_MAKE_LISTS, R1_API_COMPARE_LISTS, R1_API_TRANSFORM_LISTSR1_API_PRE_TRANSFER_DELETION, R1_API_SPLIT_REGULAR_TRANSFER_LIST =11, R1_API_TRANSFER_2_TARGET, R1_API_POST_TRANSFER_DELETION, R1_API_POST_COMMANDS, R1_API_CLEAN_REMOTE_END, R1_API_CLEAN_LOCAL_END , R1_API_END
DateTime	SubmitTime	Job's "submit" time.	DateTime object containing the job-submitted timestamp.

MSI Deployment Methods

Return Values	Name	Description
R1Job	Resubmit	Resubmits an existing job by changing the required settings through the included qualifiers.
R1Job	SubmitNow	Submits a scheduled job immediately.

Return Values	Name	Description
R1Job	Modify	Allows modifying a job's properties.

Resubmit

Syntax

```
R1Job.Resubmit
```

Parameter

(String Qualifiers)

SubmitNow

Syntax

```
R1Job.SubmitNow()
```

Example

Submits Scheduled job 15:

```
$session.QueryFilter.JobId=15
 $job=$session.GetJobList()[0]
 $job.SubmitNow()
```

Modify

Syntax

```
R1Job.Modify
```

Example

```
If($R1Job.type -match 'replication' -and $R1Job.State -match
'scheduler') {
$R1Job.Modify("-msi_action=remove   -run_option=on_demand")
}
```

MsiDistributionJob

MsiDistributionJob includes the following properties and methods.

Note: The Rollback option is not available for MSI Deployment jobs.

MsiDistributionJob Properties

Type	Name	Description	Valid Values
DistributionJob[]	`History`	Stores children jobs that were spawned by this **scheduled** Replication job. **Note:** `History` is available when `QueryFilter.IncludeHistoryJobs` is set to "true".	
String	`ParentID`	Unique ID of this job's parent job.	
Enum HOLD_MODE	`HoldMode`	Indicates if a job is on hold.	NONE EXECUTION_TIMEFRAME PRIORITY ADMINISTRATIVE WAIT_FOR_COMMIT
DateTime	`HoldUntil`	Indicates when a held job will be resumed.	
DistributionInfo	`Info`	Information about the job's settings.	
ReplicationJob[]	`ChildJobs`	Stores all Center-Edge Replication jobs spawned by this Distribution job. **Note:** `ChildJobs` is available when `QueryFilter.IncludeChildJobs` is set to "true".	
Scheduling	`Scheduling`	Job's running schedule.	

Type	Name	Description	Valid Values
R1_MSI_ACTION_TYPES	`MsiAction`	MSI deployment operation to apply to the target(s).	R1_MSI_ACTION_INSTALL R1_MSI_ACTION_UPGRADE_REPAIR R1_MSI_ACTION_REMOVE
String	`MsiConfiguration`	Specify each MSI file, including deployment settings, you wish to deploy on the target.	XML formatted string For the XML file's format, refer to Appendix C: MSI Configuration XML.
Enum R1_DISTRIBUTION_JOB_STAGES	`Stage`	Current stage in a job's lifecycle.	R1D_API_INIT, R1D_API_VALIDITY, R1D_API_LICENSE_CHECK, R1D_API_PRE_DISTRIBUTION_DOWNLOAD, R1D_API_PRE_COMMANDS, R1D_API_SUBMIT_CORE_JOBS, R1D_API_GATHER_REMOTE_INFO, R1D_API_ENABLE_CORE_PRE_CMD, R1D_API_WAIT_TO_CORE_PRE_CMD, R1D_API_SUBMIT_LOCAL_ML, R1D_API_TRANSFER, R1D_API_CHECK_UNICAST_TRANSFER, R1D_API_ENABLE_CORE_POST_CMD, R1D_API_CORES_COMPLETION, R1D_API_POST_COMMANDS, R1D_API_CLEAN_LOCAL_END, R1D_API_END
DateTime	`SubmitTime`	Job's "submit" time.	DateTime object containing the job-submitted timestamp.

MSI Deployment Distribution Methods

Return Values	Name	Description
R1Job	Resubmit	Resubmits an existing job by changing the required settings through the included qualifiers.
R1Job	SubmitNow	Submits a scheduled job immediately.

Return Values	Name	Description
R1Job	[Modify](#)	Allows modifying a job's properties.

Resubmit

Syntax

```
R1Job.Resubmit
```

Parameter

(String Qualifiers)

SubmitNow

Syntax

```
R1Job.SubmitNow()
```

Example

Checks if the job is scheduled and then submits:

```
If ( $R1Job.state –match 'SCHEDULER' ) { $R1Job.SubmitNow() }
```

Modify

Syntax

```
R1Job.Modify
```

Example

```
If($R1Job.type -match 'replication' -and $R1Job.State -match
'scheduler') {
$R1Job.Modify("-password=new_password")
}
```

ReplicationRollbackJob

Replication Rollback includes the following properties.

ReplicationRollbackJob Properties

Type	Name	Description	Valid Values
String	ParentID	Unique ID of the job that spawned this job.	
String	GroupID	Unique ID of the group a job belongs to.	
ReplicationRollbackInfo	Info	Information about the job's settings.	
String	OriginalJobID	Original ID of a rolled back job.	
Enum ROLLBACK_METHOD	RollbackMethod	Rollback method used to create this job.	ALL, TO_ABSOLUTE_TIME, TO_RELATIVE_TIME, TO_NUM_OF_INSTANCE
Object	RollbackValue	Relevant RollbackMethod value.	**NULL** if RollbackMethod= ALL, **DateTime object** if RollbackMethod= TO_ABSOLUTE_TIME/TO_ RELATIVE_TIME **Integer** if RollbackMethod= TO_NUM_OF_INSTANCE

DistributionRollbackJob

Distribution Rollback includes the following properties.

DistributionRollbackJob Properties

Type	Name	Description	Valid Values
ReplicationJob[]	ChildJobs	Stores the Center-Edge Replication Rollback jobs spawned by this Distribution rollback job.	
String	ParentID	Unique ID of the job that spawned this job.	
DistributionRollbackInfo	Info	Information about the job's settings.	
String	OriginalJobId	Original ID of the job that has been rolled back.	
Enum ROLLBACK_METHOD	RollbackMethod	Rollback method used to create this job.	ALL, TO_ABSOLUTE_TIME, TO_RELATIVE_TIME, TO_NUM_OF_INSTANCE
Object	RollbackValue	Relevant RollbackMethod value.	**NULL** if RollbackMethod= ALL, **DateTime object** if RollbackMethod= TO_ABSOLUTE_TIME/TO_RELATIVE_TIME **Integer** if RollbackMethod= TO_NUM_OF_INSTANCE

QueryFilter

QueryFilter provides advanced filtering options for retrieving jobs. Use this object to fine tune the filtering settings for retrieving jobs using the `R1Session.GetJobList` method.

Note: Each R1Session object includes one QueryFilter object. To retrieve different job lists, use different R1Session objects.

QueryFilter Properties

Set these properties to define the query for retrieving jobs.

Type	Name	Description	Valid Values
String	EdgeId	Unique Edge ID.	
String	EdgeName	IP address or name of the Edge.	
String	ParentId	Unique parent job ID code.	
Boolean	IncludeChildJobs	Indicates if the Center or Edge Replication job will be returned with the Distribution jobs' members. **Note:** Using this feature may affect performance.	
Boolean	IncludeHistoryJobs	Indicates if to include children jobs that were spawned by **scheduled** jobs. **Note:** Using this feature may affect performance.	
String	JobId	Job ID.	
String	JobName	Job name, as entered by the user.	

Type	Name	Description	Valid Values
String[]	JobState	Job state.	R1_API_SUBMITTED R1_API_INITIALIZING R1_API_RUNNING R1_API_COMPLETED_SUCCESS R1_API_COMPLETED_ERROR R1_API_INTERUPTTED R1_API_ABORTED R1_API_SCHEDULER_JOB R1_API_HOLD_RECOVERING R1_API_WAIT_FOR_CONFIRM R1_API_HOLD **Note:** For R1_API_HOLD, API_SCHED_HOLD_MODE indicates the reason for holding the job, as follows: **1** – Execution **2** - Priority **3** – Administrative
String[]	JobType	Job type.	R1_DISTRIBUTION R1_DISTRIBUTION_ROLLBAC C R1_EDGE R1_EDGE_ROLLBACK R1_REPLICATION R1_REPLICATION_ROLLBAC K

QueryFilter Methods

Use these methods to retrieve jobs.

Return Values	Name	Description
String[]	GetJobTypes	Retrieves all job types.
String[]	GetJobStates	Retrieves all job states.
Void	Reset	Resets all QueryFilter properties to default.

GetJobTypes

Syntax

```
QueryFilter.GetJobTypes()
```

Example

Gets all rollback jobs:

```
$operations = $session.QueryFilter.GetJobTypes() | where { $_ -
match 'rollback' }
```

GetJobStates

Syntax

```
QueryFilter.GetJobStates()
```

Example

Gets job states that are on hold:

```
$states = $session.QueryFilter.GetJobStates() | where{ $_ -match
'hold' }
```

Reset

Remarks

QueryFilter default settings are:

```
JobId              : NULL
ParentId           : NULL
JobName            : NULL
EdgeName           : NULL
EdgeId             : NULL
JobState           : {All}
JobType            : {All}
IncludeHistoryJobs : False
IncludeChildJobs   : False
```

Syntax

```
QueryFilter.Reset()
```

Parameters

```
JobId
```

```
ParentId
JobName
EdgeName
EdgeId
JobState
JobType
IncludeHistoryJobs
IncludeChildJobs
```

Example

Resets `QueryFilter` to the default settings:

```
$R1Session.QueryFilter.Reset()
```

R1JobInfo

This section displays the JobInfo object properties required for setting up, configuring and executing Attunity RepliWeb jobs.

- Replication JobInfo
- Distribution JobInfo
- MSI Deployment JobInfo
- MSI Deployment Distribution JobInfo
- Replication Rollback JobInfo
- Distribution Rollback JobInfo

ReplicationInfo Properties

Type	Property	Description	Valid Values
Object	`IisMetabaseInfo`	Job's IIS 6 Metabase Replication-related information.	
Object	`ScheduleInfo`	Job's scheduling information.	
Enum R1_ASSEMBLY_REPLACE_OPTION	`AssemblyReplaceOption`	(Assembly Replication mode) Indicates the rule for replacing replicated .NET assemblies in GAC.	R1_ASSEMBLY_REPLACE_ON_NEWER_SOURCE , R1_ASSEMBLY_REPLACE_ON_DIFFERENT_SOURCE
Enum RDS_JOB_DIRECTION	`Direction`	Job's replication direction.	RDS_API_DOWNLOAD, RDS_API_UPLOAD
Enum RDS_JOB_LOGIC	`SyncLogic`	Job's replication logic.	RDS_API_MIRROR, RDS_API_BACKUP, RDS_API_PURGE
Enum R1_REPLICATION _JOB_STAGES	`Stage`	Job's current low-level phase.	R1_API_INIT = 0, R1_API_START, R1_API_GET_REMOTE_INFO, R1_API_CHECK_LICENSE, R1_API_MAKE_REMOTE_WORKING_DIR , R1_API_PRE_COMMANDS, R1_API_MAKE_LISTS, R1_API_COMPARE_LISTS, R1_API_TRANSFORM_LISTS, R1_API_PRE_TRANSFER_DELETION, R1_API_SPLIT_REGULAR_TRANSFER_LIST =11, R1_API_TRANSFER_2_TARGET, R1_API_POST_TRANSFER_DELETION, R1_API_POST_COMMANDS, R1_API_CLEAN_REMOTE_END, R1_API_CLEAN_LOCAL_END , R1_API_END

Type	Property	Description	Valid Values
Enum R1_JOB_STATES	`State`	Job's high level state.	R1_API_SUBMITTED R1_API_INITIALIZING R1_API_RUNNING R1_API_COMPLETED_SUCCESS R1_API_COMPLETED_ERROR R1_API_INTERUPTTED R1_API_ABORTED R1_API_SCHEDULER_JOB R1_API_HOLD_RECOVERING R1_API_WAIT_FOR_CONFIRM R1_API_HOLD **Note:** For R1_API_HOLD, API_SCHED_HOLD_MODE indicates the reason for holding the job, as follows: **1** – Execution **2** - Priority **3** – Administrative
Enum R1_RUN_TYPE	`RunType`	Job's scheduling type.	R1_RUN_ONCE, R1_SCHEDULED, R1_CONTINUOUS_UPDATE
Boolean	`AbortOnTransferMissingFiles`	Indicates if to abort the job in case a file to be transferred is missing from the source directory. For example, if a file has been deleted or renamed between the time the snapshot was generated and the actual file transfer.	
Boolean	`AclDetect`	Specifies if the Access Control List is detected.	
Int32	`ActualMultiStreamsNum`	Number of concurrent transfers to be performed.	
Int32	`ActualStreamsNumber`	Number of streams into which each file is divided and sent concurrently.	
Boolean	`AdsDetectType`	Alternate data streams represented in NTFS as Data Streams. When transferring a file, this option enables replicating, for example, Macintosh Resource Fork files.	
Boolean	`AllFiles`	Replicates all files.	
Int32	`Attempts`	Number of recovery attempts.	

Type	Property	Description	Valid Values
Boolean	Confirm	Indicates if the job requires confirmation.	
Boolean	CopyDirPermissionOnCreation	Indicates if to copy a newly created directory's permissions.	
Boolean	CopyRootSecurity	(For Unix systems) Indicates If to replicate root directory permissions. This option is available both for Windows <-> Windows and UNIX <-> UNIX replication.	
Boolean	CopySharesMask	Indicates if share definitions will be copied from source to target.	
Int32	CuRetryFailedFiles	(For Continuous Update jobs) Indicates the number of transfer retries, in case files failed transfer.	
Boolean	CuStatisticsDeleted	(For Continuous Update jobs) Total number of files deleted by the CU job.	
Boolean	DeleteBeforeTransfer	Indicates if to prevent replication from failing due to a lack of disk space on the target machine. The default is to delete files from the target directory AFTER transferring changed files.	
Boolean	ExistingReports	Number of existing reports.	
Boolean	AbortTransferOnError	Indicates if to abort job in case of error in transferring the files.	
Int32	FcBandwidthStyle	(For WAN jobs) Required bandwidth option.	**None** – No bandwidth limit is set. **Absolute** – Set absolute bandwidth limit in MB|Mb|KB|Kb. **Relative** – Set relative bandwidth limit in % **Expression** – Set bandwidth using an expression.
Int32	FcDifferentialMinSize	(For WAN jobs) minimum bytes differential between the source and target which triggers differential transfer.	
Boolean	FcDifferentialTransfer	(For WAN jobs) Indicates if only changes between a given file will be transferred by the job (down to the block level).	

Type	Property	Description	Valid Values
Int32	`InternalLanPort`	(For LAN jobs) Internal LAN port number.	
Boolean	`OverrideAfterComplet ion`	Indicates if transactional update will transfers file/s to a temp directory, and then to the target.	
Int32	`FcPort`	(For WAN jobs) WAN Transfer Engine port. Default is 5745.	
Int32	`TimeTolerance`	Indicates if the job's pre-replication command will be executed with each new attempt to run the job.	
Int32	`FcUseAlternatePort`	(For WAN jobs) Indicates if and which alternative port to 5745 is being used.	
Int32	`FilesCopied`	Number of files copied in the job.	
Int32	`FilesDeleted`	Number of files deleted in the job.	
Int32	`FilesFailedToCopy`	Number of files that failed to be copied in the job.	
Int32	`FilesToTransfer`	Number of files to be transferred in the job.	
Boolean	`FollowUnixLinksToDir ectories`	Indicates if Attunity RepliWeb will follow the source directory link and include the files from that linked directory in its snapshot as files that reside in a directory with the link name	
Int32	`HandlingUnixLinksToD irectories`	Required action for cases where a link to a directory is detected on the source.	
Boolean	`HandlingUnixLinksToF iles`	Required action for cases where a link to a file is detected in the source.	
Boolean	`IgnoreUnixHiddenFile s`	Indicates if hidden UNIX files will be replicated in the job. For example, files beginning with a period.	
Boolean	`IgnoreWinInvalidFile names`	If the source is UNIX and the target is Windows, UNIX files whose names include invalid characters in Windows will not be transferred. If this option is not selected, and files of that kind exist, the job will fail.	

Type	Property	Description	Valid Values
Boolean	`LfaCompareAfterTrans fer`	(For Large File Accelerator jobs) Attunity RepliWeb will analyze each distributed target file to ensure that it matches the source. Enabling this option can significantly decrease performance.	
Int32	`LfaCompressionLevel`	(For Large File Accelerator jobs) Compression level to use according to the compressibility of the files and the network conditions.	
Int32	`LfaCompressionMinSiz e`	(For Large File Accelerator jobs) Smallest possible file size, in bytes, that may be transferred. Applying differential transfer to small files can actually increase the total replication time.	File size is in Bytes.
Int32	`LfaConnectTimeout`	(For Large File Accelerator jobs) TCP connect session timeout, in seconds, to be used during transfer using the LFA engine. For OS, default is 0.	
Boolean	`LfaDifferentialTrans fer`	(For Large File Accelerator jobs) Attunity RepliWeb will only transfer the changes in a given file during transfer, down to the block level.	
Int32	`LfaFileTransferSessi ons`	(For Large File Accelerator jobs) Defined number of concurrent sessions to be performed. In order to accelerate transfer of large files, files can be broken into smaller streams, or sessions, and transferred in parallel.	Files smaller than 128KB will not be divided.
Boolean	`LfaMaintainPriorSecu rityOnTarget`	(For Large File Accelerator jobs when the target system is UNIX) Attunity RepliWeb will maintain the target file's UNIX security attributes (ownership, group, and protection mask. This is applicable only if a file of the same name exists on the target, prior to file transfer.	
Int32	`LfaReceiveTimeout`	(For Large File Accelerator jobs) TCP receive operation timeout, in seconds, to be used during transfer using the LFA engine. For OS system, default is 0.	

Type	Property	Description	Valid Values
Int32	`LfaSendTimeout`	(For Large File Accelerator jobs) TCP send operation timeout, in seconds, to be used during transfer using the LFA engine. For OS system, default is 0.	
Boolean	`LfaSsl`	(For Large File Accelerator jobs) Use SSL protocol in LFA engine for data encryption and authentication between the Center and Edge. For a detailed explanation about Attunity RepliWeb security, see the <u>Attunity RepliWeb SSL Guide</u>.	
Boolean	`LfaUseCompressionSizeLimit`	(For Large File Accelerator jobs)	**True** - "lfa_compression_min_size" is set
Boolean	`LocalPostCommandExecuted`	Performs the command script, on Center, after transfer is completed.	
Boolean	`LocalPreCommandExecuted`	Perform the command script, on Center, before transfer begins.	
Int32	`MaintainPriorSecurityOnTarget`	(For LAN or Large File Accelerator jobs) Attunity RepliWeb will maintain the target file's UNIX security attributes (ownership, group, and protection mask). This is applicable only if a file of the same name exists on the target, prior to file transfer.	
Int32	`MinimalPurgeAge`	Attunity RepliWeb will not purge files that were created or modified within the past n seconds.	
Int32	`MultiStreamsNumber`	Number of multi streams.	
Int32	`PostCommandTimeout`	Time to wait (in seconds), until post-command script on the Center and/or Edge will execute. If the timeout has passed, the execution of the pre-command fails.	
Boolean	`PreserveUnixFileOwnership`	(Unix operating system) Preserves file ownership.	
Boolean	`PreserveUnixFileOwnershipToTemp`	(Unix operating system) Preserves file ownership in the temp directory.	
Boolean	`PreserveUnixFilePermissions`	(Unix operating system) Preserves file permissions.	

Type	Property	Description	Valid Values
Boolean	`PreserveUnixFilePermissionsToTemp`	(Unix operating system) Preserves file permissions in the temp directory.	
Int32	`PreCommandsTolerance`	Determines if to execute a job's pre-Replication command with each new attempt to run the job. For example, to execute a job's pre-Replication command whenever the job enters recovery, set this period as follows: If this period ends before recovery begins, the Pre-Replication command will be executed.	
Int32	`PreCommandTimeout`	Time to wait (in seconds), until pre-command script on the Center and/or Edge will execute. If the timeout has passed, the execution of the pre-command fails.	
Boolean	`PurgeFolders`	Purges folders on the target machine that do not exist on the source machine.	
Boolean	`RemotePostCommandExecuted`	Executes post command on remote computer.	
Boolean	`RemotePreCommandExecuted`	Execute pre command on remote computer.	
Boolean	`ReplicateAssemblies`	Replicates .NET Global assemblies residing in the GAC, and not just their definition in the IIS Metabase. The actual DLL's will be replicated as well. Set these replication properties in the **.NET Assemblies tab**.	
Boolean	`ReplicateComplusApplications`	Replicates COM+ components. Set these replication properties in the COM+ **Applications tab**.	
Boolean	`ReplicateContent`	Enables replicating files.	
Boolean	`ReplicateIisMetabase`	Replicates ONLY IIS 6 Metabase and not the Web Site files.	
Boolean	`ReplicateOnEmptySource`	Continues transfer in case of empty source.	
Boolean	`RollbackAbortOnError`	Aborts rollback job in case of error.	

Type	Property	Description	Valid Values
Boolean	RollbackCompress	Stores all recorded data in a compressed form. While preserving disk-space, this feature will increase recording and rollback times.	
Boolean	RollbackEnabled	Enables rollback recording on the target.	
Boolean	RollbackDiskSpaceEnabled	Enforce disk space limits used for the job's recording during the Rollback process.	
Int32	RollbackMaxDiskSpace	Set the rollback directory's maximum size, in MB.	
Boolean	RollbackEncrypt	Encrypt rollback information.	
Int32	RollbackPurgeAfter	Purges data that exists for periods longer than this one.	
Int32	ScheduledTime	(For Scheduled jobs) The start of the next scheduled child job. The time is displayed as a time value, in msec, counted since 1970.	
Boolean	SetDirPermission	Enables setting directory permissions.	
Int32	SetDirPermissionOnCreation	Sets the value of the directory permissions to the directories and sub-directories created during the replication process.	**0-777** Octal
Int32	SetFilePermission	Sets the value of the directory permissions, to all the files transferred during the replication process.	**0-777** Octal
Int32	SetRootPermission	Sets the permissions of the distribution target's root directory.	**0-777** Octal
Boolean	AbortOnDeleteError	Aborts the job on error deleting files during the synchronization process.	
Boolean	SnapshotAbortOnInaccesibleFile	During the Snapshot phase, job will fail on error accessing files (mainly due to permission issues).	
Boolean	AbortOnTransferError	Aborts the job if failure occurs while transferring files.	
Int32	SnapshotReduceCpuLevelCenter	Reduces CPU usage on the Center during Snapshot by increasing the value of this qualifier.	**0-10**
Int32	SnapshotReduceCpuLevelEdge	Reduces CPU usage on the Edge during Snapshot by increasing the value of this qualifier.	**0-10**

49

Type	Property	Description	Valid Values
Boolean	SnapshotSkipIndication	(For Continuous Update jobs) Performs a Continuous Update process without Snapshot	
Int32	SnapshotTolerance	Validity period, in seconds, for Snapshot generation.	
Int32	SnapshotWanCompression	Compresses the Snapshot file prior to its transfer to the Center.	
Int32	StatisticsSourceDirsAclModify	Number of directories whose ACL will be modified.	
Int32	StatisticsSourceDirsAdsModify	Number of directories whose ADL will be modified.	
Int32	StatisticsSourceDirsUnixSecurityModify	Number of UNIX directories whose security permissions will be modified.	
Int32	StatisticsSourceFilesAclModify	Number of files whose ACL is to be modified.	
Int32	StatisticsSourceFilesAdsModify	Number of files whose ADL is to be modified.	
Int32	StatisticsSourceFilesFailed	Number of failed files.	
Int32	StatisticsSourceFilesReplicate	Number of replicated files.	
Int32	StatisticsSourceFilesUnixSecurityModify	Number of UNIX files whose security permissions will be modified.	
Int32	StatisticsTargetDirsPostDelete	Number of target directories deleted after replication.	
Int32	StatisticsTargetDirsPreDelete	Number of target directories deleted before replication.	
Int32	StatisticsTargetFilesPostDelete	Number of target files deleted after replication.	
Int32	StatisticsTargetFilesPreDelete	Number of target files deleted before replication.	
Boolean	SubmitTime	Job "submit" time.	
Int32	TimeoutConnect	TCP "connect" session timeout, in seconds, for general communication within Attunity RepliWeb components. Default is 0 seconds.	
Int32	TimeoutLanConnect	TCP "connect" session timeout, in seconds, to be used during transfer using the LAN engine. For OS system, default is 0.	

Type	Property	Description	Valid Values
Int32	TimeoutLanRecv	TCP "receive" operation timeout, in seconds to be used during transfer using the LAN engine. For OS system, default is 0.	
Int32	TimeoutLanSend	TCP "send" operation timeout, in seconds, to be used during transfer using the LAN engine. For OS system, default is 0.	
Int32	TimeoutRecv	TCP "receive" operation timeout, in seconds, for general communication within Attunity RepliWeb components. For OS system, default, enter is 0.	
Int32	TimeoutSend	TCP "send" operation timeout, in seconds, for general communication within Attunity RepliWeb components. For OS system, default is 0.	
Int32	TimeoutWanConnect	TCP "connect" session timeout, in seconds, to be used during transfer using the WAN engine. Default is 60 seconds.	
Int32	TimeoutWanRecv	TCP "receive" operation timeout, in seconds, to be used during transfer using the WAN engine. Default is 60.	
Int32	TimeoutWanSend	TCP "send" operation timeout, in seconds, to be used during transfer using the WAN engine. Default is 60.	
Int32	TotalFilesFailed	Total files that failed transfer.	
Int32	TotalFilesTransfered	Total files that transferred to the target.	
Int32	TransferEngine	Transfer engine used.	WAN LAN LFA
Int32	TransferGid	Replicates group ownership of the files selected for replication.	
Int32	TransferReportsNum	Number of transfer reports generated for the job.	
Int32	TransferReportStyle	(For WAN jobs) Report type to display.	Totals Log Summary Detailed
Int32	TransferSgid	Replicates SGID of the files selected for replication.	

Type	Property	Description	Valid Values
Int32	`TransferSuid`	Replicates SUID of the files selected for replication.	
Int32	`TransferUid`	Replicates User ownership of the files selected for replication.	
Int32	`UdlSourceType`	Specifies how the snapshot will be created.	User Defined List Pre-sorted file list
Boolean	`UseSsl`	Use SSL.	
Object	`Assemblies`	Transfers the following .NET assemblies in the job.	
Object	`CompletionMsg`	Job's completion message.	
Object	`ComplusApplications`	Transfers the following COM+ applications in the job.	
Object	`EmailAttachReportsAbort`	Reports attached to the notification email sent in case of "abort".	
Object	`EmailAttachReportsError`	Reports attached to the notification email sent in case of "error".	
Object	`EmailAttachReportsSuccess`	Reports attached to the notification email sent in case of "success".	
Object	`EmailAttachUserFilesAbort`	User files attached to the notification email sent on "abort".	
Object	`EmailAttachUserFilesError`	User files attached to the notification email sent in case of "error".	
Object	`EmailAttachUserFilesSuccess`	User files attached to the notification email sent in case of "success".	
Object	`EmailRecipientsAbort`	Recipients of the notification email sent in case of "abort".	
Object	`EmailRecipientsError`	Recipients of the notification email sent in case of "error".	
Object	`EmailRecipientsSuccess`	Recipients of the notification email sent in case of "success".	
Object	`ExcludeNonRecursiveSpecs`	Excludes files on the level specified from replication.	
Object	`ExcludeRecursiveSpecs`	Excludes all files, including sub-directories, in the tree matching the specification.	
Object	`FileListEmailRecipientsAbort`	In case of "abort", emails file list to these recipients.	
Object	`FileListEmailRecipientsError`	In case of "error", emails file list to these recipients.	

Type	Property	Description	Valid Values
Object	`FileListEmailRecipie ntsSuccess`	In case of "success", emails list to these recipients.	
Object	`IncludeNonRecursiveS pecs`	Includes files on the level specified from replication.	
Object	`IncludeRecursiveSpec s`	Includes all files and sub-directories in the tree that match the specification.	
Object	`MacSpecs`	Replicates alternate data streams, such as Macintosh Resource Fork files.	
Object	`MultiStreamsInfo`	Information about each stream used.	**0** - Idle **1** - Working **2** - Finished **3** - Error
Object	`NumOfFilesCopiedPerS tream`	Number of files transferred to target per stream.	
Object	`NumOfFilesFailedPerS tream`	Number of files failed transfer to target per stream.	
Object	`NumOfFilesPerStream`	Number of files transferred per stream.	
Object	`ReportsExisting`	Job's existing reports.	general transfer pre_center pre_edge post_center post_edge deleted_files excluded_files rollback_recorder rollback_reloader mb_transfer edge_general sharepoint_transfer solution_deployment_transfer
String	`CenterDomain`	Center domain.	
String	`CenterPlatform`	Center's operating system.	UNIX Windows
String	`CenterUser`	Center user.	
String	`CuLastTotalBytesTran sfered`	Number of bytes transferred in the last continues update.	
String	`CuTotalBytesToTransf er`	Number of bytes transferred in the continuous update job.	
String	`Description`	Job description.	

Type	Property	Description	Valid Values
String	DirPermission	Value of the directory permissions.	**0-777** Octal
String	FcBandwidth	Set the desired bandwidth option.	**None** – No bandwidth limit is set. **Absolute** – Set absolute bandwidth limit in MB|Mb|KB|Kb. **Relative** – Set relative bandwidth limit in % **Expression** – Set bandwidth using an expression.
String	FcBandwidthExt		
String	FcCompression	Select whether Attunity RepliWeb will compress files before transfer. Compression is performed transparently, without performing any intermediate spooling of compressed data.	
String	FcCenterTempDir		
String	FcDomain		
String	FcDoNotCompress	Files of the specified types will not be compressed.	
String	FcEdgeTempDir		
String	FcVerification		
String	FileListActionTypeAbort	File List exit procedure in case of "abort".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.
String	FileListActionTypeError	File List exit procedure in case of "error".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.

Type	Property	Description	Valid Values
String	FileListActionTypeSu ccess	File List exit procedure in case of "success".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.
String	FileListTemplateAbor t	File List template to use in case of "abort".	
String	FileListTemplateErro r	File List template to use in case of "error".	
String	FileListTemplateSucc ess	File List template to use in case of "success".	
String	FileListTempDirAbort	File List temp directory to use in case of "abort".	
String	FileListTempDirError	File List temp directory to use in case of "error".	
String	FileListTempDirSucce ss	File List temp directory to use in case of "success".	
String	FilePermission	Value of the directory permissions.	0 - 777 Octal
String	HandlingDuplicateFil eNames	Action performed when encountering duplicate file names on Target and only one of either the Source or Target systems is case sensitive (i.e. UNIX).	**Error Exit** - exit the job with an error. **Select Newest** - transfer the file with the newest modification date. **Select Oldest** - transfer the file with the oldest modification date. **Select Lexicographic first** - transfer the file that is first alphanumerically. **Select Lexicographic last** - transfer the file that is last alphanumerically.
String	JobId	Job ID.	
String	LfaCompression	Select whether Attunity RepliWeb will compress files before transfer. Compression is performed transparently, without performing any intermediate spooling of compressed data.	

Type	Property	Description	Valid Values
String	`LfaDifferentialMinSize`	Applying differential transfer to small files can actually increase the total replication time. File size is in Bytes.	
String	`LfaDontCompress`	Specified files will not be compressed.	
String	`LocalPostCommand`	Center's post-Replication command.	
String	`LocalPostCommandParam`	Center's post-replication command parameters to pass.	
String	`LocalPreCommand`	Center's pre-Replication command.	
String	`LocalPreCommandParam`	Center's pre-replication command parameters to pass.	
String	`Name`	Job name.	
String	`RemoteOs`	Operating system used on remote computer.	**For Unix:** Solaris 2.6 and later (Sun Sparc) HP-UX 11 and later (Itanium) Red Hat Linux 7.0 and later (Kernel 2.6.x) / Linux (SUSE) Red Hat Enterprise Linux 3.0 and later / Fedora Red Hat Enterprise Linux 4.4 and later (64-bit) AIX 4.3 and later (IBM RS/600) **For Windows:** Windows Server 2000 Windows Server 2003 (32/64-bit Editions) Windows XP Windows Server 2008 (32/64-bit Editions)
String	`RemotePostCommand`	Edge's post-Replication command.	
String	`RemotePostCommandParam`	Edge's post-replication command parameters to pass.	
String	`RemotePreCommand`	Edge's pre-Replication command.	
String	`RemotePreCommandParam`	Edge's pre-replication command parameters to pass.	
Boolean	`UseCommitCommands`	Enable the use of Pre/Post Commit commands.	

Type	Property	Description	Valid Values
Boolean	`UseCommitCommandsTimeout`	Enable setting commit command timeouts.	
int	`PreCommitCommandTimeout`	Specify the time to wait, in seconds, until the Pre-Commit command on the target is executed.	
string	`PreCommitCommandParam`	Specify the parameters to be used by `PreCommitCommand`.	
string	`PreCommitCommand`	Enter the path to the Pre-Commit command script. This script will be executed on the target before the Commit Stage.	
int	`PostCommitCommandTimeout`	Specify the time to wait, in seconds, until the Post-Commit command on the target is executed.	
String	`PostCommitCommandParam`	Specify the parameters to be used by `PostCommitCommand`.	
String	`PostCommitCommand`	Enter the path to the Post-Commit command script. This script will be executed on the target after the Commit Stage.	
String	`RenameHook`	Specifies the name of the rename hook script.	For RenameHook to work, you must enable overwrite_after_completion.
String	`RenameHookParam`	Renames Hook command parameters to pass.	
String	`RollbackRecordingDir`	Directory that stores rollback recording results.	
String	`RootPermission`	Replicates root directory permissions.	
String	`Edge`	Edge machine.	
String	`SnapshotWanBandwidth`	Bandwidth limit for fetching remote list.	
String	`SourceDir`	Source directory.	
String	`StateMsg`	Job state message.	See StateMessages.
String	`TargetDir`	Target directory.	
String	`TotalBytesTransfered`	Total bytes transferred.	
String	`TotalTransferedFileSize`	Sum total, in bytes, of data transferred in the job.	
String	`UdlName`	UDL file name.	

Type	Property	Description	Valid Values
String	`Iis7Content`	Specifies the IIS 7 items whose configurations are to be replicated.	XML-represented string
Boolean	`Iis7ReplicationDoBackup`	Indicates that the target IIS 7 configuration will be backed up for rollback.	
Boolean	`Iis7ReplicationEnabled`	Indicates that IIS 7 configuration replication is enabled.	
RDS_JOB_LOGIC	`Iis7SyncLogicType`	Determines the synchronization logic for replication IIS 7 configuration.	RDS_API_MIRROR,

DistributionInfo Properties

Type	Property	Description	Valid Values
Int32	AbortedEdges	Number of Edges that aborted.	
Object	AbortedList	List of aborted Edges.	
String	AbortExitCommand	On Exit command to be executed in case of "abort".	
String	AbortExitCommandParam	On Exit command parameters to pass in case of "abort".	
Int32	AllEdgesRequired	Distribution must be successful on all Edges defined.	
Int32	CompletedEdges	Number of Edges that succeeded Distribution.	
Object	CompletedList	List of Edges that succeeded Distribution.	
Object	CompletionMsg	Job's completion message.	
Int32	Confirm	Job requires confirmation before execution.	
Int32	ContinueEmptySource	Does not abort if the source directory is empty.	
String	Description	Job description.	
String	DistributionExternalSubset	Replicates to these Edges during Distribution.	
Object	DistributionList	List of job's Edges.	
Object	EdgesIds	IDs of Edges in this job.	
Int32	EdgesNum	Number of Edges in this job.	
Object	EdgeDirList	List of target directories.	
Object	EmailAttachReportsAbort	Reports attached to the notification email sent in case of "abort".	
Object	EmailAttachReportsError	Reports attached to the notification email sent in case of "error".	
Object	EmailAttachReportsSuccess	Reports attached to the notification email sent in case of "success".	
Object	EmailAttachUserFilesAbort	User files attached to the notification email sent in case of "abort".	

Type	Property	Description	Valid Values
Object	EmailAttachUserFilesError	User files attached to the notification email sent in case of "error".	
Object	EmailAttachUserFilesSuccess	User files attached to the notification email sent in case of "success".	
Object	EmailRecipientsAbort	Recipients of the notification email sent in case of "abort".	
Object	EmailRecipientsError	Recipients of the notification email sent in case of "error".	
Object	EmailRecipientsSuccess	Recipients of the notification email sent in case of "success".	
Int32	EndTime	Job's end time.	
String	ErrorExitCommand	On Exit command to be executed in case of "error".	
String	ErrorExitCommandParam	On Exit command parameters to pass in case of "error".	
Int32	FailedEdges	Number of Edges that failed Distribution.	
Object	FailedList	List of Edges that failed Distribution.	
String	FileListActionTypeAbort	File List exit procedure in case of "abort".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.
String	FileListActionTypeError	File List exit procedure in case of "error".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.

Type	Property	Description	Valid Values
String	`FileListActionTypeSuccess`	File List exit procedure in case of "success".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.
Object	`FileListEmailRecipientsAbort`	In case of "abort", emails file list to these recipients.	
Object	`FileListEmailRecipientsError`	In case of "error", emails file list to these recipients.	
Object	`FileListEmailRecipientsSuccess`	In case of "success", emails list to these recipients.	
String	`FileListTemplateAbort`	File List template to use in case of "abort".	
String	`FileListTemplateError`	File List template to use in case of "error".	
String	`FileListTemplateSuccess`	File List template to use in case of "success".	
String	`FileListTempDirAbort`	File List temp directory to use in case of "abort".	
String	`FileListTempDirError`	File List temp directory to use in case of "error".	
String	`FileListTempDirSuccess`	File List temp directory to use in case of "success".	
Boolean	`InPreview`	Edges waiting for confirmation.	
String	`JobId`	Job ID.	
String	`LocalPostCommand`	Center's Post-Replication command.	
String	`LocalPostCommandParam`	Center's Post-Replication command parameters to pass.	
String	`LocalPreCommand`	Center's Pre-Replication command.	
String	`LocalPreCommandParam`	Center's Pre-Replication command parameters to pass.	
String	`Name`	Distribution job's name.	

Type	Property	Description	Valid Values
Int32	PreviewType	Preview mode used for reviewing the files that are about to be transferred or deleted during the Job process.	Preview Distribution Window Replication Window
Int32	PreDistributionDownloads	Download job to run before Distribution begins.	
Object	PreDistributionDownloadsList	List of the job's Pre-Distribution Download Edges.	
Object[]	PreDistributionDownloadsEdgesIds	IDs of the job's Pre-Distribution Download jobs.	
Object	ReportsExisting	Job's existing reports.	general transfer pre_center pre_edge post_center post_edge deleted_files excluded_files rollback_recorder rollback_reloader mb_transfer edge_general sharepoint_transfer solution_deployment_transfer
Int32	RollbackEnabled	Rollback is enabled.	
Object	RollbackHostsList	List of Edges that store rollback information.	
Object	RollbackRecordingDirs	Directories for recording rollback information.	
Int32	RunningEdges	Number of Edges currently running.	
Object	RunningList	List of Edges currently running.	
Enum R1_RUN_TYPE	RunType	Job's Schedule Mode.	R1_RUN_ONCE, R1_SCHEDULED, R1_CONTINUOUS_UPDATE
Int32	ScheduledTime	The start time of the next scheduled child job. The time is displayed in msec, counted since 1970.	
Object	ScheduleInfo	Job's scheduling information.	

Type	Property	Description	Valid Values
Enum R1_RUN_OPTION	ScheduleOption	Schedule option selected.	R1_IMMEDIATE, R1_HOLD_UNTIL, R1_EVERY_MINUTES, R1_DAILY, R1_WEEKLY, R1_FILE_EXISTENCE, R1_FILE_SPEC, R1_SAT_FILE_EXISTENCE, R1_CONT_OR_SAT_FILE_EXISTENCE, R1_CONT_AND_SAT_FILE_EXISTENCE, R1_ON_DEMAND
String	SourceDirectory	Source directory on the Center.	
Boolean	SourceIsEmpty	Indicates if to abort on empty source.	
Enum R1_DISTRIBUTION_JOB_STAGES	Stage	Current low-level phase of the job.	R1D_API_INIT, R1D_API_VALIDITY, R1D_API_LICENSE_CHECK, R1D_API_PRE_DISTRIBUTION_DOWNLOAD, R1D_API_PRE_COMMANDS, R1D_API_SUBMIT_CORE_JOBS, R1D_API_GATHER_REMOTE_INFO, R1D_API_ENABLE_CORE_PRE_CMD, R1D_API_WAIT_TO_CORE_PRE_CMD, R1D_API_SUBMIT_LOCAL_ML, R1D_API_TRANSFER, R1D_API_CHECK_UNICAST_TRANSFER, R1D_API_ENABLE_CORE_POST_CMD, R1D_API_CORES_COMPLETION, R1D_API_POST_COMMANDS, R1D_API_CLEAN_LOCAL_END, R1D_API_END
String	StageMsg	Current stage description.	

Type	Property	Description	Valid Values
Enum R1_JOB_STATES	State	Job's high-level state.	R1_API_SUBMITTED R1_API_INITIALIZING R1_API_RUNNING R1_API_COMPLETED_SUCCESS S R1_API_COMPLETED_ERROR R1_API_INTERUPTTED R1_API_ABORTED R1_API_SCHEDULER_JOB R1_API_HOLD_RECOVERING R1_API_WAIT_FOR_CONFIRM R1_API_HOLD **Note:** For R1_API_HOLD, API_SCHED_HOLD_MODE indicates the reason for holding the job, as follows: **1** – Execution **2** - Priority **3** – Administrative
String	StateMsg	Job state message.	See StateMessages.
Int32	SubmitTime	Job's submit time.	
String	SuccessExitCommand	On Exit command to execute in case of "success".	
String	SuccessExitCommandParam	On Exit command parameters to pass in case of "success".	
Int32	SyncAfterPreCmd	Edge Jobs will synchronize once they have all executed pre-command.	
Int32	SyncBeforePostCmd	Edge Jobs will synchronize once they have all reached post-command.	
Int32	SyncBeforePreCmd	Edge Jobs will synchronize once they have all reached pre-command.	
Object	TargetDirList	List of target directories.	
Enum R1_TRANSPORT_ ENGINE_TYPES	TransportEngine	Transport engine used.	R1_API_REPLIWEB

Type	Property	Description	Valid Values
Enum R1_OPERATION_ TYPES	Type	Job type.	R1_DISTRIBUTION, R1_EDGE, R1_REPLICATION, R1_DISTRIBUTION_ROLLBACK, R1_EDGE_ROLLBACK, R1_REPLICATION_ROLLBACK
Int32	UnicastTransferProgress		
Int32	UseSsl	Indicates that SSL is used. For a detailed explanation about Attunity RepliWeb security, see Attunity RepliWeb SSL Guide.	

MsiInfo Properties

Type	Property	Description	Valid Values
Int32	AbortOnFailure	Abort the job if an error deploying an MSI file is encountered.	**0** – continue **1** - abort
Int32	Attempts	Number of recovery attempts.	
String	CenterDomain	Center domain.	
String	CenterPlatform	Center's operating system.	UNIX Windows
String	CenterUser	Center user.	
Object	CompletionMsg	Job's completion message.	
Enum R1_DEPLOYMENT_TYPES	DeploymentType	Job's deployment type.	Solution MSI
String	Description	Job description.	
Enum RDS_JOB_DIRECTION	Direction	Job's replication direction.	RDS_API_DOWNLOAD, RDS_API_UPLOAD
String	Edge	Edge machine.	
Object	EmailAttachReportsAbort	Reports attached to the notification email sent in case of "abort".	
Object	EmailAttachReportsError	Reports attached to the notification email sent in case of "error".	
Object	EmailAttachReportsSuccess	Reports attached to the notification email sent in case of "success".	
Object	EmailAttachUserFilesAbort	User files attached to the notification email sent on "abort".	
Object	EmailAttachUserFilesError	User files attached to the notification email sent in case of "error".	
Object	EmailAttachUserFilesSuccess	User files attached to the notification email sent in case of "success".	
Object	EmailRecipientsAbort	Recipients of the notification email sent in case of "abort".	
Object	EmailRecipientsError	Recipients of the notification email sent in case of "error".	

Type	Property	Description	Valid Values
Object	`EmailRecipientsSuccess`	Recipients of the notification email sent in case of "success".	
Int32	`ExistingReports`	Number of existing reports.	
String	`JobId`	Job ID.	
String	`LocalPostCommand`	Center's post-Replication command.	
Int32	`LocalPostCommandExecuted`	Performs the command script, on Center, after transfer is completed.	**0** – not executed **1** – executed
String	`LocalPostCommandParam`	Center's post-replication command parameters to pass.	
String	`LocalPreCommand`	Center's pre-Replication command.	
Int32	`LocalPreCommandExecuted`	Perform the command script, on Center, before transfer begins.	**0** – not executed **1** – executed
String	`LocalPreCommandParam`	Center's pre-replication command parameters to pass.	
Int32	`MinimalPurgeAge`	Attunity RepliWeb will not purge files that were created or modified within the past n seconds.	
Enum R1_MSI_ACTION _TYPES	`MsiAction`	Job's MSI Deployment action.	Install Upgrade Remove
String	`MsiConfiguration`	MSI deployment configuration in XML format.	XML formatted string For the XML file's format, refer to Appendix C: MSI Configuration XML.
String	`Name`	Job name.	
Int32	`PostCommandTimeout`	Time to wait (in seconds), until post-command script on the Center and/or Edge will execute. If the timeout has passed, the execution of the pre-command fails.	

Type	Property	Description	Valid Values
Int32	PreCommandTimeout	Time to wait (in seconds), until pre-command script on the Center and/or Edge will execute. If the timeout has passed, the execution of the pre-command fails.	
Int32	PreCommandsTolerance	Determines if to execute a job's pre-Replication command with each new attempt to run the job. For example, to execute a job's pre-Replication command whenever the job enters recovery, set this period as follows: If this period ends before recovery begins, the Pre-Replication command will be executed.	
String	RemoteOs	Operating system used on remote computer.	**For Unix:** Solaris 2.6 and later (Sun Sparc) HP-UX 11 and later (Itanium) Red Hat Linux 7.0 and later (Kernel 2.6.x) / Linux (SUSE) Red Hat Enterprise Linux 3.0 and later / Fedora Red Hat Enterprise Linux 4.4 and later (64-bit) AIX 4.3 and later (IBM RS/600) **For Windows:** Windows Server 2000 Windows Server 2003 (32/64-bit Editions) Windows XP Windows Server 2008 (32/64-bit Editions)
String	RemotePostCommand	Edge's post-Replication command.	
Int32	RemotePostCommandExecuted	Executes post command on remote computer.	**0** – not executed **1** – executed
String	RemotePostCommandParam	Edge's post-replication	

Type	Property	Description	Valid Values
		command parameters to pass.	
String	RemotePreCommand	Edge's pre-Replication command.	
Int32	RemotePreCommandExecuted	Execute pre command on remote computer.	**0** – not executed **1** – executed
String	RemotePreCommandParam	Edge's pre-replication command parameters to pass.	
Object	ReportsExisting	Job's existing reports.	general transfer pre_center pre_edge post_center post_edge deleted_files excluded_files rollback_recorder rollback_reloader mb_transfer edge_general sharepoint_transfer solution_deployment_transfer
Object	ReportsSuccessExiting	Specifies the reports to be attached when job completes successfully	
Int32	RollbackAbortOnError	Aborts rollback job in case of error.	**0** – continue **1** – abort
Int32	RollbackCompress	Stores all recorded data in a compressed form. While preserving disk-space, this feature will increase recording and rollback times.	
Int32	RollbackDiskSpaceEnabled	Enforce disk space limits used for the job's recording during the Rollback process.	**0** – do not enforce **1** – enforce
Int32	RollbackEnabled	Enables rollback recording on the target.	**0** – disable **1** – enable
Int32	RollbackEncrypt	Encrypt rollback information.	**0** – do not encrypt **1** – encrypt
Int32	RollbackMaxDiskSpace	Set the rollback directory's maximum size, in MB.	
Int32	RollbackPurgeAfter	Purges data that exists for periods longer than this one.	
String	RollbackRecordingDir	Directory that stores rollback recording results.	

Type	Property	Description	Valid Values
String	RollbackRecordingUniqueDir		
Enum R1_RUN_TYPE	RunType	Job's schedule type.	R1_CONTINUOUS_UPDATE R1_RUN_ONCE R1_SCHEDULED
ScheduleInfo	ScheduleInfo	Job's scheduling information.	
Enum R1_RUN_OPTION	ScheduleOption	Job's schedule option.	R1_IMMEDIATE R1_HOLD_UNTIL R1_EVERY_MINUTES R1_DAILY R1_WEEKLY R1_FILE_EXISTENCE R1_FILE_SPEC R1_SAT_FILE_EXISTENCE R1_CONT_OR_SAT_FILE_EXISTENCE R1_CONT_AND_SAT_FILE_EXISTENCE R1_ON_DEMAND
Int32	ScheduledTime	(For Scheduled jobs) The start of the next scheduled child job. The time is displayed as a time value, in msec, counted since 1970.	
String	Stage	Job's current stage.	
Enum R1_JOB_STATES	State	Job's high level state.	R1_API_SUBMITTED R1_API_INITIALIZING R1_API_RUNNING R1_API_COMPLETED_SUCCESS R1_API_COMPLETED_ERROR R1_API_INTERUPTTED R1_API_ABORTED R1_API_SCHEDULER_JOB R1_API_HOLD_RECOVERING R1_API_WAIT_FOR_CONFIRM R1_API_HOLD **Note:** For R1_API_HOLD, API_SCHED_HOLD_MODE indicates the reason for holding the job, as follows: 1 – Execution 2 - Priority 3 – Administrative

Type	Property	Description	Valid Values
String	StateMsg	Job state message.	See StateMessages.
Int32	SubmitTime	Job "submit" time.	
Int32	UseSsl	Use SSL.	
Int32	UseUserTarget	(Used with **UserTarget**) Enables saving a copy of the job's MSI files on a target directory.	
String	UserTarget	(Used with **–UseUserTarget**) Path to the directory where copies of the job's MSI files will be saved.	

MsiDistributionInfo Properties

Type	Property	Description	Valid Values
String	AbortExitCommand	On Exit command to be executed in case of "abort".	
String	AbortExitCommandParam	On Exit command parameters to pass in case of "abort".	
Int32	AbortedEdges	Number of Edges that aborted.	
Object	AbortedList	List of aborted Edges.	
Int32	AllEdgesRequired	Distribution must be successful on all Edges defined.	
Int32	CompletedEdges	Number of Edges that succeeded Distribution.	
Object	CompletedList	List of Edges that succeeded Distribution.	
Object	CompletionMsg	Job's completion message.	
Enum R1_DEPLOYMENT_TYPES	DeploymentType	Job's deployment type (MSI / Solution).	R1_DEPLOY_MSI
String	Description	Job description.	
String	DistributionExternalSubset	Replicates to these Edges during Distribution.	
Object	DistributionList	List of job's Edge names.	
Object	EdgeDirList	List of target directories.	
String	Edges		
Object	EdgesIds	IDs of Edges in this job.	
Object	EdgesNames	Job's Edge names.	
Int32	EdgesNum	Number of job's Edges.	
Object	EmailAttachReportsAbort	Reports attached to the notification email sent in case of "abort".	
Object	EmailAttachReportsError	Reports attached to the notification email sent in case of "error".	
Object	EmailAttachReportsSuccess	Reports attached to the notification email sent in case of "success".	

Type	Property	Description	Valid Values
Object	`EmailAttachUserFilesAbort`	User files attached to the notification email sent in case of "abort".	
Object	`EmailAttachUserFilesError`	User files attached to the notification email sent in case of "error".	
Object	`EmailAttachUserFilesSuccess`	User files attached to the notification email sent in case of "success".	
Object	`EmailRecipientsAbort`	Recipients of the notification email sent in case of "abort".	
Object	`EmailRecipientsError`	Recipients of the notification email sent in case of "error".	
Object	`EmailRecipientsSuccess`	Recipients of the notification email sent in case of "success".	
Int32	`EndTime`	Job's end time.	
String	`ErrorExitCommand`	On Exit command to be executed in case of "error".	
String	`ErrorExitCommandParam`	On Exit command parameters to pass in case of "error".	
Int32	`FailedEdges`	Number of Edges that failed Distribution.	
Object	`FailedList`	List of Edges that failed Distribution.	
Int32	`HandledEdges`	Number of Edges that succeeded Distribution.	
String	`JobId`	Distribution job ID.	
Object	`JobIdList`	Distribution's Edge job IDs.	
String	`LocalPostCommand`	Center's Post-Replication command.	
String	`LocalPostCommandParam`	Center's Post-Replication command parameters to pass.	
String	`LocalPreCommand`	Center's Pre-Replication command.	
String	`LocalPreCommandParam`	Center's Pre-Replication command parameters to pass.	
Int32	`MinNodesOpt`		
Int32	`MinimumNodes`		

Type	Property	Description	Valid Values
Enum R1_MSI_ACTION _TYPES	MsiAction	MSI deployment operation to apply to the target.	R1_MSI_ACTION_INSTALL R1_MSI_ACTION_UPGRAD E_REPAIR R1_MSI_ACTION_REMOVE
String	Name	Distribution job's name.	
Int32	PreDistributionDownloads	Download job to run before Distribution begins.	
Object	PreDistributionDownloadsLi st	List of the job's Pre-Distribution Download Edges.	
Object	ReportsExisting	Job's existing reports.	general transfer pre_center pre_edge post_center post_edge deleted_files excluded_files rollback_recorder rollback_reloader mb_transfer edge_general sharepoint_transfer solution_deployment_tran sfer
Int32	RollbackEnabled	Rollback is enabled.	
Object	RollbackHostsList	List of Edges that store rollback information.	
Object	RollbackRecordingDirs	Directories for recording rollback information.	
Enum R1_RUN_TYPE	RunType	Job's Schedule Mode.	R1_RUN_ONCE, R1_SCHEDULED, R1_CONTINUOUS_UPDATE
Int32	RunningEdges	Number of Edges currently running.	
Object	RunningList	List of Edges currently running.	
ScheduleInfo	ScheduleInfo	Job's scheduling information.	

Type	Property	Description	Valid Values
Enum R1_RUN_OPTION	`ScheduleOption`	Schedule option selected.	R1_IMMEDIATE, R1_HOLD_UNTIL, R1_EVERY_MINUTES, R1_DAILY, R1_WEEKLY, R1_FILE_EXISTENCE, R1_FILE_SPEC, R1_SAT_FILE_EXISTENCE, R1_CONT_OR_SAT_FILE_EXISTENCE, R1_CONT_AND_SAT_FILE_EXISTENCE, R1_ON_DEMAND
Int32	`ScheduledTime`	The start time of the next scheduled child job. The time is displayed in msec, counted since 1970.	
Enum R1_DEPLOY_DISTRIBUTION_JOB_STAGES	`Stage`	Current low-level phase of the job.	R1D_API_INIT, R1D_API_VALIDITY, R1D_API_LICENSE_CHECK, R1D_API_PRE_DISTRIBUTION_DOWNLOAD, R1D_API_PRE_COMMANDS, R1D_API_SUBMIT_CORE_JOBS, R1D_API_GATHER_REMOTE_INFO, R1D_API_ENABLE_CORE_PRE_CMD, R1D_API_WAIT_TO_CORE_PRE_CMD, R1D_API_SUBMIT_LOCAL_ML, R1D_API_TRANSFER, R1D_API_CHECK_UNICAST_TRANSFER, R1D_API_ENABLE_CORE_POST_CMD, R1D_API_CORES_COMPLETION, R1D_API_POST_COMMANDS, R1D_API_CLEAN_LOCAL_END, R1D_API_END
String	`StageMsg`	Current stage description.	

Type	Property	Description	Valid Values
Enum R1_JOB_STATES	State	Job's high-level state.	R1_API_SUBMITTED R1_API_INITIALIZING R1_API_RUNNING R1_API_COMPLETED_SUCCESS R1_API_COMPLETED_ERROR R1_API_INTERUPTTED R1_API_ABORTED R1_API_SCHEDULER_JOB R1_API_HOLD_RECOVERING R1_API_WAIT_FOR_CONFIRM R1_API_HOLD **Note:** For R1_API_HOLD, API_SCHED_HOLD_MODE indicates the reason for holding the job, as follows: **1** – Execution **2** - Priority **3** – Administrative
String	StateMsg	Job state message.	See [StateMessages](#).
Int32	SubmitTime	Job's submit time.	
String	SuccessExitCommand	On Exit command to execute in case of "success".	
String	SuccessExitCommandParam	On Exit command parameters to pass in case of "success".	
Int32	SyncAfterPreCmd	Edge Jobs will synchronize once they have all executed pre-command.	
Int32	SyncBeforePostCmd	Edge Jobs will synchronize once they have all reached post-command.	
Int32	SyncBeforePreCmd	Edge Jobs will synchronize once they have all reached pre-command.	
Object	TargetDirList	List of target directories.	
Enum R1_OPERATION_TYPES	Type	Job type.	R1_FLAT_MODE 1R1_DISTRIBUTION R1_EDGE R1_REPLICATION R1_DISTRIBUTION_ROLLBACK

Type	Property	Description	Valid Values
			R1_EDGE_ROLLBACK R1_REPLICATION_ROLLBAC K R1_BACKUP R1_ORG_SYNC R1_DEPLOY_TYPES_MASK R1_DEPLOYMENT_DISTRIB UTION R1_DEPLOYMENT_EDGE R1_DEPLOYMENT R1_DEPLOYMENT_DISTRIB UTION_ROLLBACK R1_DEPLOYMENT_EDGE_R OLLBACK R1_DEPLOYMENT_ROLLBA CK
Int32	`UnicastEdges`		
Object	`UnicastList`		
Int32	`UnicastTransferProgress`		
Int32	`UseSsl`	Indicates that SSL is used. For a detailed explanation about Attunity RepliWeb security, see Attunity RepliWeb SSL Guide.	

ReplicationRollbackInfo Properties

Type	Property	Description	Valid Values
Boolean	AbortExitCommand	On Exit command to execute in case of "abort".	
Boolean	AbortExitCommandParam	On Exit command parameters to pass in case of "abort".	
String	Description	Job description.	
Object	EmailAttachReportsAbort	Reports attached to the notification email sent in case of "abort".	
Object	EmailAttachReportsError	Reports attached to the notification email sent in case of "error".	
Object	EmailAttachReportsSuccess	Reports attached to the notification email sent in case of "success".	
Object	EmailAttachUserFilesAbort	User files attached to the notification email sent in case of "abort".	
Object	EmailAttachUserFilesError	User files attached to the notification email sent in case of "error".	
Object	EmailAttachUserFilesSuccess	User files attached to the notification email sent in case of "success".	
Object	EmailRecipientsAbort	Recipients of the notification email sent in case of "abort".	
Object	EmailRecipientsError	Recipients of the notification email sent in case of "error".	
Object	EmailRecipientsSuccess	Recipients of the notification email sent in case of "success".	
String	ErrorExitCommand	On Exit command to execute in case of "error".	
String	ErrorExitCommandParam	On Exit command parameters to pass in case of "error".	

Type	Property	Description	Valid Values
String	FileListActionTypeAbort	File List exit procedure in case of "abort".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.
String	FileListActionTypeError	File List exit procedure in case of "error". Options are:.	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.
String	FileListActionTypeSuccess	File List exit procedure on "success".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined. **Execute File** - creates and runs the batch file at end of job.
Object	FileListEmailRecipientsAbort	In case of "abort", emails file list to these recipients.	
Object	FileListEmailRecipientsError	In case of "error", emails file list to these recipients.	
Object	FileListEmailRecipientsSuccess	In case of "success", emails list to these recipients.	
String	FileListTemplateAbort	File List template to use in case of "abort".	
String	FileListTemplateError	File List template to use in case of "error".	
String	FileListTemplateSuccess	File List template to use in case of "success".	
String	FileListTempDirAbort	File List temp directory to use in case of "abort".	
String	FileListTempDirError	File List temp directory to use in case of "error".	

Type	Property	Description	Valid Values
String	FileListTempDirSuccess	File List temp directory to use in case of "success".	
Object	IisMetabaseInfo	IIS 6 Metabase content to distribute.	
String	JobId	Job ID.	
String	LocalPostCommand	Center's Post-Replication command.	
String	LocalPostCommandParam	Center's Post-Replication command parameters to pass.	
String	LocalPreCommand	Center's Pre-Replication command.	
String	LocalPreCommandParam	Center's Pre-Replication command parameters to pass.	
String	Name	Job name.	
String	OriginalJobId	ID of rolled back job.	
Int32	Port	Port used for the job. Default is 2837.	
Int32	PostCommandTimeout	Time to wait (in seconds) until post command script on the Center and/or Edge executes. If timeout has passed, the execution of the post-command fails.	
Int32	PreCommandTimeout	Time to wait (in seconds) until pre-command script on the Center and/or Edge executes. If timeout has passed, the execution of the pre-command fails.	
String	RemoteDomain	Edge domain.	
String	RemoteNode	Edge name or IP.	
String	RemotePostCommand	Center's Pre-Replication command.	
String	RemotePostCommandParam	Center's Pre-Replication command parameters to pass.	
String	RemotePreCommand	Center's Pre-Replication command.	
String	RemotePreCommandParam	Center's Pre-Replication command parameters to pass.	

Type	Property	Description	Valid Values
String	RemoteUser	Edge user name.	
Object	ReportsExisting	Job's existing reports.	general transfer pre_center pre_edge post_center post_edge deleted_files excluded_files rollback_recorder rollback_reloader mb_transfer edge_general sharepoint_transfer solution_deployment_transfer
Int32	RollbackCompress	Stores all recorded data in a compressed form. While preserving disk-space, this feature will increase recording and rollback times.	
Int32	RollbackEncrypt	Encrypts rollback information.	
Int32	RollbackMethod	Rollback method to be used in case of rollback.	**0** – To time (All, Absolute, Relative) **1** – To instance
String	RollbackToTime	Enabled if RollbackMethod = 0. Point in time to rollback to.	Time represented by "ddMMyyyyhhmm"
Int32	RollbackNumOfInstances	Enabled if RollbackMethod = 1. Indicates the number of instances the original job is rolled back to.	Number of instances defined
String	RollbackRecordingDir	Directory that stores rollback recordings.	
Object	ScheduleInfo	Job scheduling information.	

Type	Property	Description	Valid Values
Enum R1_REPLICATION_J OB_STAGES	Stage	Current low-level phase of the job..	R1_API_INIT = 0, R1_API_START, R1_API_GET_REMOTE_INFO , R1_API_CHECK_LICENSE, R1_API_MAKE_REMOTE_W ORKING_DIR , R1_API_PRE_COMMANDS, R1_API_MAKE_LISTS, R1_API_COMPARE_LISTS, R1_API_TRANSFORM_LISTS, R1_API_PRE_TRANSFER_DE LETION, R1_API_SPLIT_REGULAR_TR ANSFER_LIST =11, R1_API_TRANSFER_2_TARG ET, R1_API_POST_TRANSFER_D ELETION, R1_API_POST_COMMANDS, R1_API_CLEAN_REMOTE_E ND, R1_API_CLEAN_LOCAL_END , R1_API_END
String	StageMsg	Description of job's "stage".	
String	SuccessExitCommand	On Exit command to execute in case of "success".	
String	SuccessExitCommandParam	On Exit command parameters to pass in case of "success".	
String	TargetDir	Target directory.	
Int32	TimeoutConnect	TCP "connect" session timeout, in seconds, for general communication within Attunity RepliWeb components. Default is 0 seconds.	

Type	Property	Description	Valid Values
Int32	TimeoutRecv	TCP "receive" operation timeout, in seconds, for general communication within Attunity RepliWeb components. Default is 0 seconds.	
Int32	TimeoutSend	TCP "send" operation timeout, in seconds, for general communication within Attunity RepliWeb components. Default is 0 seconds.	

DistributionRollbackInfo Properties

Type	Property	Description	Valid Values
Int32	AbortedEdgesNum	Number of aborted Edges.	
Object	AbortedList	List of aborted Edges.	
String	AbortExitCommand	On Exit command to execute in case of "abort".	
String	AbortExitCommandParam	On Exit command parameters to pass in case of "abort".	
Int32	AllEdgesRequired	Indicates that in order for Distribution to complete successfully, you should roll back Replication on all of the job's Edges that completed successfully.	
Int32	CompletedEdgesNum	Number of Distribution Rollback Edge jobs on which Replication rollback completed successfully.	
Object	CompletedList	List of the Distribution Rollback Edge jobs on which Replication rollback completed successfully.	
String	Description	Parent Distribution job's description.	
Object	DistributionList	List of Edges.	
Object	EmailAttachReportsAbort	Reports attached to the notification email sent in case of "abort".	
Object	EmailAttachReportsError	Reports attached to the notification email sent in case of "error".	
Object	EmailAttachReportsSuccess	Reports attached to the notification email sent in case of "success".	
Object	EmailAttachUserFilesAbort	User files attached to the notification email sent on "abort".	
Object	EmailAttachUserFilesError	User files attached to the notification email sent in case of "error".	
Object	EmailAttachUserFilesSuccess	User files attached to the notification email sent in case of "success".	

Type	Property	Description	Valid Values
Object	EmailRecipientsAbort	Recipients of the notification email sent in case of "abort".	
Object	EmailRecipientsError	Recipients of the notification email sent in case of "error".	
Object	EmailRecipientsSuccess	Recipients of the notification email sent in case of "success".	
String	ErrorExitCommand	On Exit command to execute on "error".	
String	ErrorExitCommandParam	On Exit command parameters to pass on "error".	
Int32	FailedEdgesNum	Number of Edges that failed Distribution.	
Object	FailedList	List of Edges that failed Distribution.	
String	FileListActionTypeAbort	File List exit procedure in case of "abort".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined **Execute File** - creates and runs the batch file at end of job.
String	FileListActionTypeError	File List exit procedure in case of "error".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined **Execute File** - creates and runs the batch file at end of job.
String	FileListActionTypeSuccess	File List exit procedure in case of "success".	**None** - batch file is created but no action performed on it. **Mail File** - sends the created batch via email to recipients defined **Execute File** - creates and runs the batch file at end of job.
Object	FileListEmailRecipientsAbort	In case of "abort", emails file list to these recipients.	

Type	Property	Description	Valid Values
Object	`FileListEmailRecipientsError`	In case of "error", emails file list to these recipients.	
Object	`FileListEmailRecipientsSuccess`	In case of "success", emails list to these recipients.	
String	`FileListTemplateAbort`	File List template to use in case of "abort".	
String	`FileListTemplateError`	File List template to use in case of "error".	
String	`FileListTemplateSuccess`	File List template to use in case of "success".	
String	`FileListTempDirAbort`	File List temp directory to use in case of "abort".	
String	`FileListTempDirError`	File List temp directory to use in case of "error".	
String	`FileListTempDirSuccess`	File List temp directory to use in case of "success".	
String	`JobId`	Distribution rollback job ID.	
Object	`JobIdList`	List of spawned job IDs.	
String	`LocalPostCommand`	Center's Post-Replication command.	
String	`LocalPostCommandParam`	Center's Post-Replication command parameters to pass.	
String	`LocalPreCommand`	Center's Pre-Replication command.	
String	`LocalPreCommandParam`	Center's Pre-Replication command parameters to pass.	
Int32	`MinimumNodes`	Number of required Edges that must succeed.	
Int32	`MinNodesOpt`	Indicates the number of failed Edges allowed by the Distribution job.	
String	`Name`	Job name.	
String	`OriginalJobId`	ID of the Distribution job that was rolled back.	
Int32	`OriginalSubmitTime`	Submit time of rolled back job.	
String	`OriginalSubmitTimeStr`		

Type	Property	Description	Valid Values
Int32	PostCommandTimeout	Time to wait (in seconds) until the post-command script on the Center and/or Edge executes. If timeout has passed, the execution of the post-command fails.	
Int32	PreCommandTimeout	Time to wait (in seconds) until pre-command script on the Center and/or Edge executes. If timeout has passed, the execution of the pre-command fails.	
Object	RecordingDirs	List of directories that store rollback recordings.	
Object	ReportsExisting	Job's existing reports.	general transfer pre_center pre_edge post_center post_edge deleted_files excluded_files rollback_recorder rollback_reloader mb_transfer edge_general sharepoint_transfer solution_deployment_transfer
Int32	RollbackMethod	Rollback method to be used in case of rollback.	**0** – To time (All, Absolute, Relative) **1** – To instance
String	RollbackToTime	Enabled if RollbackMethod = 0. Point in time to rollback to.	Time represented by "ddMMyyyyhhmm"
Int32	RollbackNumOfInstances	Enabled if RollbackMethod = 1. Number of instances the original job is rolled back to.	Number of instances defined
Int32	RunningEdgesNum	Number of Edges currently running in the Distribution job.	
Object	RunningList	List of currently running jobs.	
Object	ScheduleInfo	Job's scheduling information.	

Type	Property	Description	Valid Values
Int32	SchedType	Schedule mode.	Run Once Continuous Update Scheduled
Enum R1_JOB_STATES	State	Job's high-level state.	R1_API_SUBMITTED R1_API_INITIALIZING R1_API_RUNNING R1_API_COMPLETED_SUCCESS R1_API_COMPLETED_ERROR R1_API_INTERUPTTED R1_API_ABORTED R1_API_SCHEDULER_JOB R1_API_HOLD_RECOVERING R1_API_WAIT_FOR_CONFIRM R1_API_HOLD **Note:** For R1_API_HOLD, API_SCHED_HOLD_MODE indicates the reason for holding the job, as follows: **1** – Execution **2** - Priority **3** – Administrative
String	StateMsg	Job state message.	See [StateMessages](#).
String	SuccessExitCommand	On Exit command to execute on success.	
String	SuccessExitCommandParam	On Exit command parameters to pass on success.	
Int32	SyncBeforePostCommand	All Edge job post commands will execute after all Edges have finished their transfer stage.	
Int32	SyncBeforePreCommand	All Edge job pre commands will execute together.	
Int32	SyncBeforeReload	Center and Edge(s) will synchronize before Edges reload.	

Type	Property	Description	Valid Values
Int32	`TotalEdgesNum`	Number of Edge jobs in the Distribution job.	

A - Exit Messages

Attunity RepliWeb can report events to the Microsoft Windows Event Viewer and the UNIX syslog. The current implementation issues are reported upon the completion of an Attunity RepliWeb distribution or replication job. Job completion means the job has completed successfully, has been aborted by the user, or has exhausted all retry attempts and failed.

Replication Exit Messages

The following messages describe the state of the Replication job.

Success

"R1-S-Replication completed successfully"

Information

"R1-I-Waiting for Transaction Commit Condition:"
"R1-I-Waiting for Comparative Snapshot confirmation"
"R1-I-Hold by user instruction"

Abort

"R1-E-Aborted due to a non active Distribution job"

Failure

"R1-F-Unable to set default FASTCopy qualifiers"
"R1-F-Unable to set default FASTCopy transfer qualifiers"
"R1-F-Unable to establish connection to RepliWeb's Scheduler"
"R1-F-Unable to get RepliWeb server name"
"R1-E-The specified recording directory is not valid"

"R1-E-RepliWeb Event Driven Agent exited unexpectedly"
"R1-E-Unknown state of Replication job <>"
"is not supported on UNIX.\ "
"R1-E-Configuration file is not valid"
"R1-E-RDS is not installed"
"R1-F-Failed to get configuration file field"
"R1-F-Failed to set configuration file field"
"R1-F-Failed to get job state from configuration file"
"R1-F-Failed to initialize configuration file\ "
"R1-F-Failed at initialization"
"R1-F-Failed to get transaction type"
"R1-E-Failed to connect using dial-up"
"R1-E-Failed to disconnect dial-up"

"R1-E-Failed to initialize the retrial of failed files\ "
"R1-E-Could not delete failed files list\ "
"R1-E-Failed to prepare failed files list\ "
"R1-E-Failed to handle failed files\ "
"R1-E-Failed to get remote Edge information"
"R1-E-Failed to reconnect to Edge"
"R1-E-Failed in paths check"
"R1-E-Failed in failover validity"
"R1-E-Failed to convert remote Edge information"
"R1-E-Failed to check break point"
"R1-E-Failed to get rollback user's definitions"
"R1-E-Failed at IIS 6 Metabase validation\ "
"R1-E-Failed at IIS 7 Configuration validation\ "
"R1-E-Failed at .NET Assemblies validation\ "
"R1-E-Failed at COM+ Services validation\ "
"R1-E-Failed at license validation"
"R1-E-Failed to execute Pre-Replication commands"
"R1-E-Failed to execute Post-Replication commands"
"R1-E-Failed to start RepliWeb Event Driven Agent"
"R1-E-Failed to communicate with Distribution Event Driven Agent"
"R1-E-Failed to handle Event Driven Agent periodic full synchronization\ "
"R1-E-Failed to generate snapshots\ "
"R1-E-Failed to compare snapshots"
"R1-E-Failed to transform comparative snapshot"
"R1-E-Failed to delete ignored files from Center and Edge"
"R1-E-Failed to check validity of streams in recovery"
"R1-E-Failed to generate IIS 6 Metabase snapshots"
"R1-E-Failed to compare IIS 6 Metabase snapshots"
"R1-E-Failed to create target IIS 6 Metabase"
"R1-E-Failed to generate .NET Shared Assemblies snapshot"
"R1-E-Failed comparing COM+ applications"
"R1-E-Failed to check if < > of type < > exists:\ "
"R1-E-Failed to record rollback data"
"R1-E-Failed at post transform Comparative Snapshot"
"R1-E-Failed to set preview"
"R1-E-Failed to remove confirm preview indication"
"R1-E-Failed at pre transfer deletion"
"R1-E-failed at post transfer deletion"
"R1-E-Failed to transfer files"
"R1-E-Failed to rename files"
"R1-E-Failed to replicate IIS 6 Metabase"
"R1-E-Failed to replicate IIS 7 Configuration\ "
"R1-E-Failed replicating .NET Assemblies"
"R1-E-Failed transferring .NET Assembly < >\n"
"R1-E-Failed to replicate COM+ applications"
"R1-E-Failed to transfer COM+ application < >\n"
"R1-E-Failed to register COM+ application < >\n"
"R1-E-Failed to compare COM+ application < >\n"

"R1-E-Failed to clean up temporary files from the Edge"
"R1-E-Failed to create Edge directories"
"R1-E-Failed due to remote platform conflict"
"R1-E-Failed to create Comparative Snapshot confirmation information"
"R1-E-Failed to consolidate information after snapshot confirmation"
"R1-E-Failed to initialize winsock"
"R1-E-Failed to initialize report"
"R1-E-Failed to allocate configuration file handle"
"R1-E-Failed to set configuration file table"
"R1-E-Failed to read configuration file"
"R1-E-Failed to update configuration file"
"R1-E-Insufficient disk space on <center machine name>. Minimum disk space require at <drive or partition>: <%ld> MB, disk space available: <%ld> MB."
"R1-E-Insufficient disk space on <edge machine name>. Minimum disk space required at <drive or partition> : <%ld> MB, disk space available: <%ld> MB."
"R1-I-Deleting recordings since <date> to clear recording space."

Distribution Exit Messages

The following messages describe the state of the Distribution job.

Success

"R1-S-Distribution completed successfully"
"R1-I-Edge Deployment Groups committed successfully"

Information

"R1-I-Hold by user instruction"
"R1-F-Unknown state in Distribution manager job"
"R1-I-Waiting to gather remote information from Edge jobs"
"R1-I-Waiting for Edge jobs(s) to reach pre command stage"
"R1-I-Waiting for Edge jobs(s) to complete performing the pre command stage"
"R1-I-Waiting to gather preview information from Edge jobs"

"R1-I-Waiting for Edge jobs(s) to reach post command stage"
"R1-I-Waiting for Edge jobs(s) to complete performing the post command stage"
"R1-I-Waiting for Edge Group <> to complete"
"R1-I-Waiting for Commit Condition:"
"R1-I-Holding for Comparative Snapshot confirmation\n"
"R1-I-Commit Condition is Edge Deployment Groups"

Abort

"R1-E-Aborted by Center Distribution Job"
"R1-E-Aborted on empty source snapshot"

Failure

"R1-S- Distribution stopped at debug checkpoint"
"R1-E-Distribution source directory does not exist"
"R1-E-R1 is not installed"
"R1-E-RepliWeb is not installed"
"R1-E-Unable to meet Distribution success condition"
"%s is not supported on UNIX.\n"
"R1_E-Failed at job initialization"
"R1-E-Failed to initialize network module"
"R1-E-Failed to initialize scheduler connection"
"R1-E-Failed to release Edge jobs after hold"
"R1-E-Failed to hold Edge jobs upon user request"
"R1-E-Failed to read hold request information"
"R1-E-Failed to check distribution manager validity"
"R1-E-Failed at Distribution manager license check"
"R1-E-Failed at Pre-Distribution Download(s)"
"R1-E-Failed to run Center pre commands"
"R1-E-Failed at Center post commands"

"R1-E-Failed in Edges pre commands"
"R1-E-Failed in Edges post commands"
"R1-E-Failed to submit Edge jobs"
"R1-E-Failed at gathering remote information from Edges "
"R1-E-Failed to Calculate Filter Parameters"
"R1-E-Failed to start RepliWeb Event Driven Agent"
"R1-E-Failed to generate local snapshot"
"R1-E-Failed to control Edges operation"
"R1-E-Failed to signal Edge jobs to Comparative Snapshot Technology"
"R1-E-Failed to move Edges to unicast transfer"
"R1-E-Failed to check unicast progress"
"R1-E-Failed to wait for Edge jobs completion"
"R1-E-Failed to handle Edges commit completion"
"R1-E-Failed to set last break point"
"R1-E-Failed to set wakeup condition"
"R1-E-Failed to gather preview information from Edge jobs"
"R1-E-Failed to distribute preview information to Edge jobs"
"R1-E-FAILED Distribution did not reach the minimum Edges required for success\n"
"R1-E-Failed during RepliWeb Event Driven Agent(s) continuous update monitoring"
"\tEdge <> Unique ID < > failed with error < >\n"
"\tPre-Distribution Download <> (job) failed\"
"R1-E-Failed at Center license validation\n"
"R1-E-Failed to load rw_glob dll function\n Please contact support for new version\n"
"R1-E-Failed to load rw_misc dll function\n Please contact support for new version\n"
"R1-E-Failed to load rw_client dll function\n Please contact support for new version\n"
"R1-E-Failed to load sched_api dll function\n Please contact support for new version\n"
"R1-E-Insufficient disk space on <center machine name>. Minimum disk space require at <drive or partition>: <%ld> MB, disk space available: <%ld> MB."
"R1-E-Insufficient disk space on <edge machine name>. Minimum disk space required at <drive or partition> : <%ld> MB, disk space available: <%ld> MB."
"R1-I-Deleting recordings since <date> to clear recording space."

B - Examples

Note: When appending values that are to be used as CLI qualifiers into a text file, you should use the following command: `out-file "your text" -encoding ascii`

Script 1

This script searches for and commits held Distribution jobs only if a specific Edge and target directory exists:

```
#connect to center
$session= Get-R1Session -Center 10.0.61.55 -User administrator -
ScrambledPassword U72VT71GU0QIK1PMU0T2FKKVK1
$session.QueryFilter.JobType= "R1_DISTRIBUTION"
$session.QueryFilter.IncludeChildJobs= $true

#get all relevant jobs
$jobs= $session.GetJobList()
$waitForCommit= $jobs | Where {$_.holdmode -match 'commit'}

#security code+write to log
if( $waitForCommit -eq $null)
{
 $date =get-Date
 $date.toString()  >> "c:\r1_ps.log"
 echo "No jobs found!" >> "c:\r1_ps.log"
 Exit 1
}
#iterate jobs and commits all jobs tha fits condition
foreach($waitJob in $waitForCommit)
{
 foreach($child in $waitJob.ChildJobs)
 {
     if($child.Edge -eq "127.0.0.1" -and ($child.TargetDirectory -eq
"c:\target"))
     {
         $waitJob.Commit()
         $date =get-Date
         $date.toString()  >> "c:\r1_ps.log"
         echo "$waitforcommit : Commited" >> "c:\r1_ps.log"
         break
     }
 }
}
```

Script 2

This script searches for all scheduled Distribution jobs and adds a commit condition to be activated by trigger file for a specific Edge:

```
#security code
if($Args.length -lt 3)
{
 Write-Error "send edge name, trigger file path and center name through
script arguments"
 exit 1
}
$edgename= $Args[0]
$trigger= $Args[1]
$center= $Args[2]
$modFile= "c:\mod.txt"

#create modifier file
echo "edge_override $edgename mod -commit_trigger_file_type=edge -
transaction_commit_condition=trigger_file
-transactional_update_type=job -commit_edge_trigger_file=$trigger" |
Out-File -FilePath $modFile -Encoding ASCII

#connect to center by prompting credentials request
$cred = Get-Credential
$session= Get-R1Session -Center $center -Credential $cred

#set query filter
$session.QueryFilter.JobType= "R1_DISTRIBUTION"
$session.QueryFilter.JobState= "R1_API_SCHEDULER_JOB"

#get all relevant jobs
$jobs= $session.GetJobList()

foreach($job in $jobs)
{
 #check if edge exists
 foreach($edge in $job.info.edgesNames)
 {
     if($edge -eq $edgename)
     {
                 $job.Modify("-mod=$modFile")
         break
     }
 }
}
```

C - MSI Configuration XML

MSI files and settings are specified in an XML file. For example:

```
<?xml version="1.0" encoding="UTF-8"?>
<MSIList>
    <default>
        <UseDefaults>false</UseDefaults>
        <UseParameters>false</UseParameters>
        <Parameters></Parameters>
        <UseDependencies>false</UseDependencies>
        <Dependencies/>
        <UninstallBefore>false</UninstallBefore>
    </default>
    <MSIEntry>
        <File>C:\MyInstaller.msi</File>
        <UseMSI>true</UseMSI>
        <UseDefaults>true</UseDefaults>
        <UseParameters>false</UseParameters>
        <Parameters></Parameters>
        <UseDependencies>false</UseDependencies>
        <Dependencies/>
        <UninstallBefore>false</UninstallBefore>
    </MSIEntry>
</MSIList>
```

For more information about each setting, refer to the **–msi_list** qualifier in the _Attunity RepliWeb User Guide_.