

FOCUS ON FUNCTION

> **not coding**

{ YOUR GUIDE TO BUILDING & AUTOMATING
WORKFLOWS FASTER AND MORE RELIABLY }

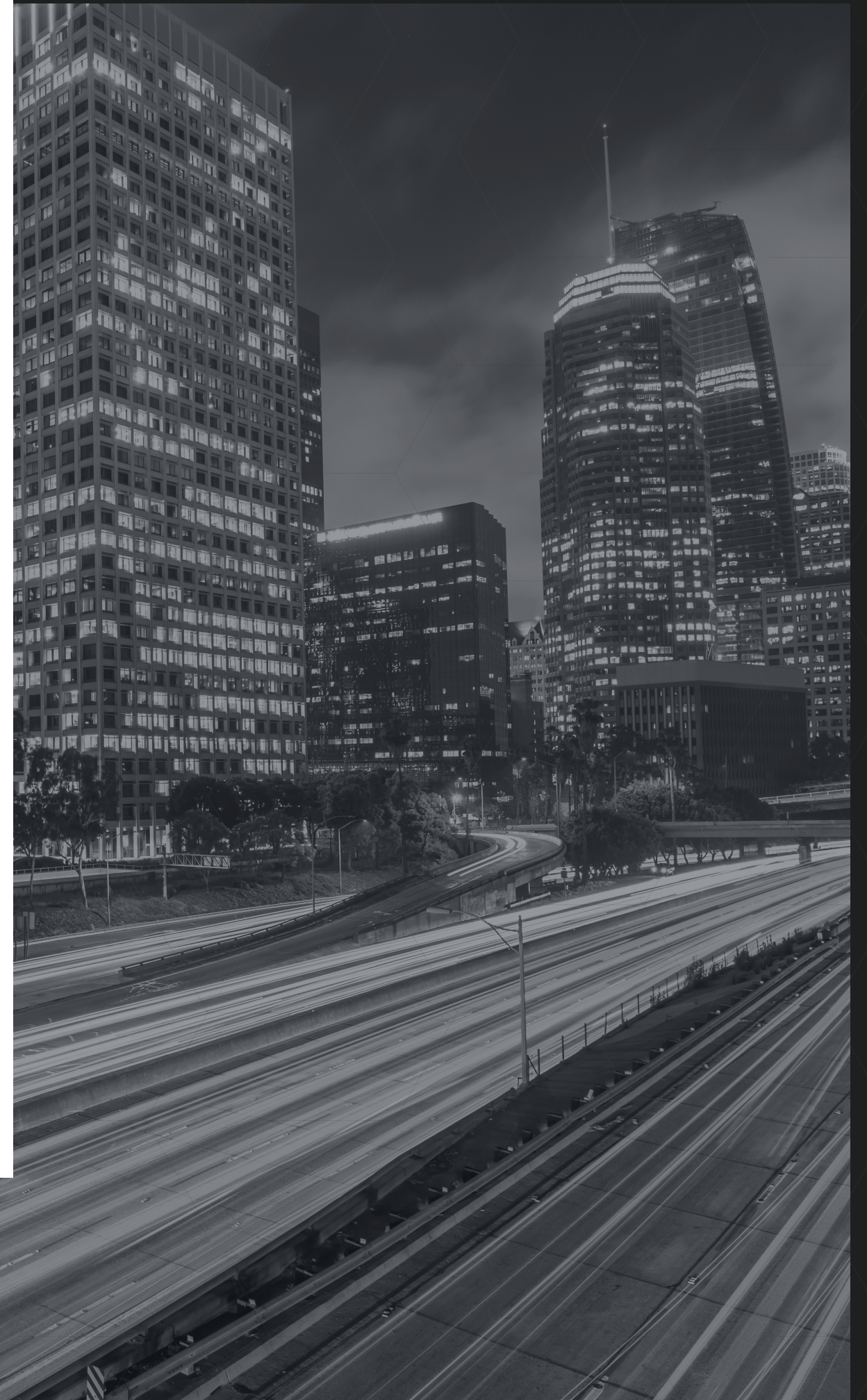


In the world of business and IT, we are surrounded by change.

Continuous advances in technology are ensuring that the volume, variety, and velocity of data that IT needs to manage is increasing daily, and constant new developments in the industry almost guarantee that this influx of data will not be decelerating any time soon. Managing the growth in data and velocity while simultaneously fulfilling a multitude of other duties is resulting in the need for IT Teams to do more, faster than ever before.

To keep pace with the industry and with competitors, it is imperative that organizations become more agile and responsive to change. This shift requires IT organizations to be able to build, automate, test, change, audit, revise, optimize, stop, and start their workflows faster and more reliably.

This guide will introduce ways that IT can increase agility by using solutions that allow IT to focus on the function of their workflows, not the coding behind them. It will also highlight key examples and use cases from ActiveBatch IT Automation that demonstrate how shifting your focus to objectives, rather than processes, saves both time and resources.



The Problem:

For IT professionals, there is never enough time in the day. There are too many business processes, external applications, shifting dependencies, data sources, and more that IT must manage daily while simultaneously facing the realities of rather stagnant IT budgets.

To address this problem, many IT organizations have turned to writing custom scripts and code to automate these processes and integrate them with other applications, systems, and technologies. But coding, writing, testing, rewriting, retesting, introducing new code, etc. can take hours for developers. Additionally, scripts are hardly ever centrally located; they're often stored in unsystematic, scattered locations across the organization. Only adding to these issues is the fact that scripts are simply not scalable, nor are they commonly well documented.

The focus of IT needs to shift, and to do this IT teams need a solution that enables them to think differently.

According to the latest research from Gartner, IT spending is "projected to total \$3.5 trillion in 2017, a 2.4 percent increase from 2016" and this growth is "up from the previous quarter's forecast of 1.4%." While these numbers do show projected growth in IT budgets, the rate of increase is hardly quick enough to match the rate of increase in available computing resources.

The Solution:

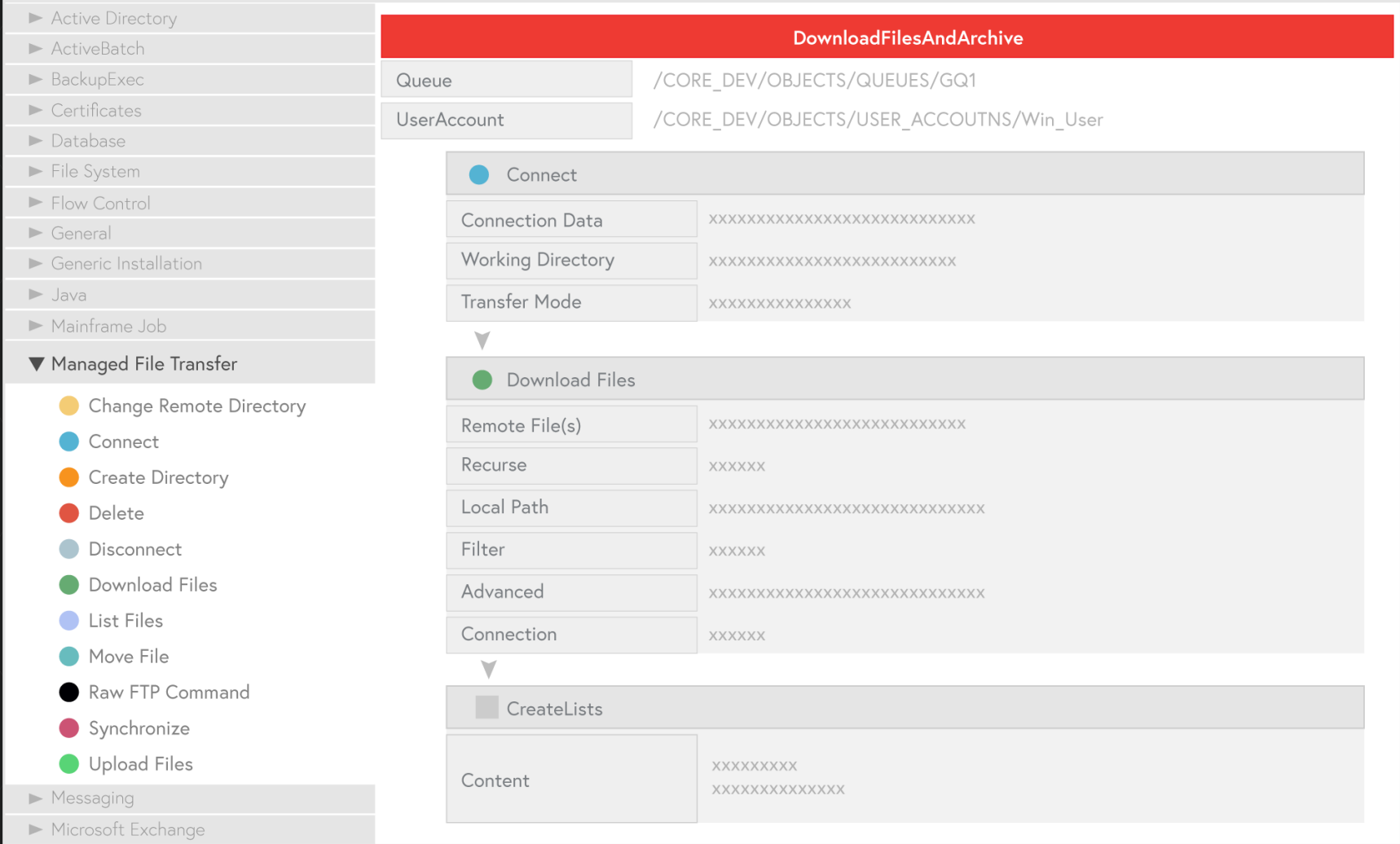
What organizations should be looking for is a comprehensive, **end-to-end IT Automation solution** that allows developers to *focus on the function* of their workflows and processes, *rather than the coding* behind them.

Where to Begin:

A modern, intelligent enterprise IT Automation solution enables developers to reduce their focus on coding by providing prebuilt, pretested drag-and-drop Job Steps for commonly scripted actions and functions. The right solution enables developers to build and automate complex workflows in less than half the time, without the need for custom scripting.

Here's an example:

ActiveBatch IT Automation's Integrated Jobs Library allows developers to focus on function, not coding. The Integrated Jobs Library is the largest library of automated integrations on the market. A rich content library filled with hundreds of prebuilt, pretested, production-ready Job Steps, the Jobs Library allows developers to simply drag-and-drop what are essentially building blocks into end-to-end workflows, minimizing the need for scripting. This results in a few things: it allows developers to concentrate on the purpose of their workflows, reduces the time it takes to build them out, and reduces the amount of testing that needs to be done. ActiveBatch Engineers have done the work so developers don't have to—hundreds of prebuilt jobs have already been tested and are ready to be assembled into workflows. This saves time for developers and allows them to concentrate on the function of their workflows, not the coding behind them.



Jason Van Pee, a Database Administrator for Sub-Zero Group, Inc. explained how his team is able to focus on function through utilizing the Integrated Jobs Library. Van Pee reported that Sub-Zero "uses the Integrated Jobs Library almost exclusively," and added, "Having a nice user interface is great for building jobs. It allows me to simply select the task from the list instead of searching for how to script it." Instead of spending time researching and coding his processes, he is able to save time and resources by simply focusing on the task at hand. -Sub-Zero Group, Inc. Innovates with Self-Service Automation for Business Users

API Jobs:

ActiveBatch's unique Service Library augments the **Integrated Jobs Library** and extends its automation capabilities. The previous example highlighted the prebuilt Job Steps within the Integrated Jobs Library, but what if there is no direct out-of-the-box integration for an application or technology you are using? This is where the Service Library comes into play. Should the Integrated Jobs Library not offer an out-of-the-box integration that you need, the Service Library can create API Jobs.

The Service Library can take the REST APIs or Web Services methods from any application or technology you want to integrate into your workflow and seamlessly turn those methods into reusable drag-and-drop Job Steps within the Jobs Library. API Jobs aren't limited to Web Services. Developers can invoke **Stored Procedures**, **.Net Assemblies**, or a **Command Line Interface** for example, and use the logic as reusable Job Steps as well.

The screenshot displays the configuration interface for a REST Adapter. The 'Properties' window is open, showing the 'REST Adapter' selected. The 'Category Name' is set to 'Taleo REST Adapter'. Under the 'Operations' section, the 'FindInstance' operation is selected, with a 'GET' method and an endpoint URL of 'https://tbe.taleo.net/MANAGER/dispatcher/api/v1/serviceURL/ADVASYST2.xml'. The 'Response' tab is active, showing a 'Response type' of 'XML' and a 'Schema' of 'JSON (complex property)'. The 'Name' is 'root' and the 'Value' is '<ResponsePropertys>'. A table at the bottom lists job steps: FindInstance, Login, Logout, ServiceDiscovery, SearchEmpl, and CreateEmpl.

Fact: Skilled IT resources are becoming harder and harder to find, making the need for prebuilt logic that reduces the need for coding more important than ever. According to Gartner's 2017 CIO Agenda: Global Perspectives on Seizing the Digital Ecosystem Opportunity, **35%** of CIOs in North America across various industries reported that the biggest talent gap in regard to information, technology, or digital business, was in the category of Information/Analytics/Data Science/and Business Intelligence—and other regions of the world fell reported around the same percentage. Additionally, overall, **26%** of CIOs from around the world reported Skills/Resources as their largest barrier to achieving their objectives, with Funding & Budgets coming in second place at **17%**. These two challenges facing CIOs (lack of skilled IT resources and stagnant IT budgets) can be alleviated with an IT Automation solution that provides prebuilt logic and allows IT to focus on function not coding, enabling less experienced developers to complete tasks that once required their senior level counterparts.

It's Your Automation - You Can Script If You Want To

While an IT Automation solution may reduce the need for custom scripting, it doesn't have to eliminate scripts altogether, nor should it do so. Many organizations have a large investment in scripts that they have written and tested for specific products, applications, business processes, or services that they use—and an IT Automation solution should only serve to augment these scripts. For example, think of a car that has adaptive cruise control (ACC). You can certainly use ACC, and it will make controlling the speed of the car easier and more reliable, but if you enjoy and feel more comfortable driving without it, there's no requirement for you to use the ACC. Just as your adaptive cruise control doesn't replace your ability to manually operate the vehicle and use the gas pedal to manage your speed, the right IT Automation solution will not replace your existing script investment or require you to start from scratch.

The right solution will allow you to use your existing scripts to get up and running quickly and allows you to build upon your scripts and offer capabilities to enhance your existing scripts.

Here's How:

ActiveBatch allows you to place your scripts into workflows both upstream and downstream from prebuilt Job Steps and actions, enabling you to build your scripts into **end-to-end processes** and enhance them without additional coding. These demonstrate a few capabilities that an IT Automation solution should provide when it comes to protecting and using your existing scripts.

ActiveBatch IT Automation offers **script vaulting** where scripts can be held and executed as needed and enhanced with lifecycle management capabilities. ActiveBatch is script language independent and supports all scripting languages, like PowerShell, Python, Perl, Ruby, and more. By utilizing your scripts within ActiveBatch, you can leverage ActiveBatch capabilities such as audits, revision rollback, alerting and monitoring, and more.

The screenshot displays the ActiveBatch workflow editor interface. On the left is a 'File System' category with a list of actions: Archive, Calculated File Checksum, Copy File, Copy Folder, Create Folder, Delete File, Delete Folder, For-Each-File, Get File Info, If-Compare Files, If-File, If-Folder, Move File, Read Text From File, Set File Attribute, Set Security, Unarchive, and Write Text to File. Below this are expandable sections for Flow Control, General, Generic Installation, Java, Mainframe Job, Messaging, Microsoft Exchange, and Networking. The 'PowerShell' section is expanded, showing 'Execute Powershell Script' and 'Format PowerShell Object'.

The main workspace shows three actions in a sequence:

- Copy File**: Source is 'C:\Testing_Suite\Job Steps\PowerShell\OriginalData.txt', Destination is 'C:\Testing_Suite\Job Steps\PowerShell\ArchivedData\ParsedData.txt', DateType is 'CreationDate', and Overwrite is 'True'.
- ExecutePowerShellScript**: Machine is '.', Script is 'Get-Content "C:\Testing_Suite\Job Steps\PowerShell\ArchivedData\ParsedData.txt" | select-string -pattern "Failed to execute"', InputObject is '<default>', Implicit Remote is 'False', Imported Command Properties is '<default>', PowerShell Console File is empty, Format Output Objects is 'True', No profile is 'False', and Display Objects Immediately is 'False'.
- ForEachItem**: ContextName is 'Value', Collection is '=ExecutePowerShellScript.ReturnValue', and an 'Expression' field is highlighted in red.

In this next section we'll look at 3 more ActiveBatch features that allow organizations to focus on function rather than coding:

1. **Variables**—pull variables into your object properties to eliminate the need for repetitive hardcoding of certain values
2. **Reference Functionality**—easily create jobs that mimic the logic of another without writing code
3. **Revision History**—if something goes wrong, get back up and run faster, without having to revise code

Variables

ActiveBatch also embraces the use of **variables**, which result in increased object reusability by reducing the need for hardcoding values. Instead of hardcoding a value within an object, developers can specify a variable to represent a result or data source, that can be used to pass information between plans and jobs. The use of variables guarantees that all new jobs that run will take into account the latest values set for them. This saves developers the time it would take to re-code the value in every workflow object that it is used in. Variables allow developers to focus on the function of their objects and workflows, rather than the coding of each value within them. Variable use can further be leveraged in ActiveBatch by dynamically passing and updating variable values within an executing workflow.

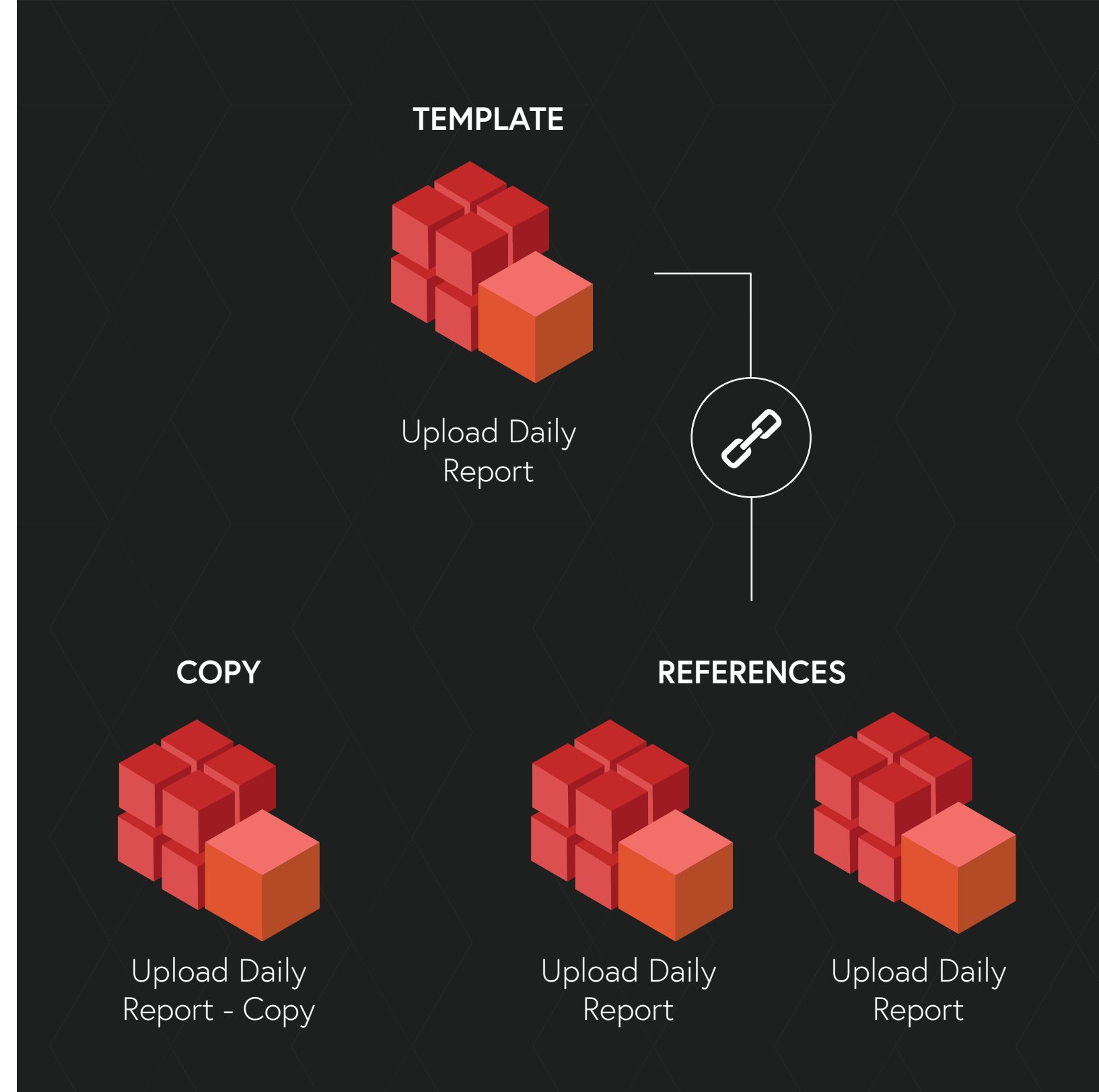
The screenshot shows the configuration for an 'UploadFiles' object. The 'Local File(s)' field contains the path `\\server_repo1\encrypted_reports\daily_${today_mmddyy}`. A dropdown menu is open, showing a list of variables including `today_mmddyy`, which is highlighted by a mouse cursor. A tooltip for the selected variable is visible in the bottom right corner.

▼ UploadFiles	
Local File(s)	\\server_repo1\encrypted_reports\daily_\${today_mmddyy}
Recurse	False
Remote Path	/daily_reports/\${today_mmddyy}
▼ Filter	Filter
▼ Time Filter	Absolute Time
Newer Than	<code>\${today_mm}</code>
Older Than	
File Size Filter	
▼ Advanced	
Force Overwrite	
Error Action	
Log Successful Transfers	
Connection	

Active Variable: Abat.General.DateAritmethics
DateTagExpression: <TODAY/FORMAT=mmddyy>
Defined at '/EXAMPLES'

Reference Functionality

ActiveBatch provides **Reference Functionality** that allows developers to reuse, rather than rewrite to simplify workflow maintenance. As many developers know, it is not uncommon to execute the same job or plan within different workflows or for different customers. Instead of having to recode new logic to mirror another job's logic or use copy-and-paste functionality (which is helpful when needing to quickly duplicate an exact job, but as a one-off, will only increase the number of jobs to maintain, adding to the complexity of the automation environment) to mirror another job's logic, developers can benefit from ActiveBatch References. ActiveBatch's Reference Functionality allows users to designate a job or plan as a template job and then create Reference Jobs that mimic the same logic as the Template Job to be used in various workflows across your IT environment. With this functionality, a single change to the Template Job or Plan will be passed down to all of the Reference Jobs or Plans throughout your different workflows. This not only saves time when creating the workflows, but simplifies workflow maintenance in the future.



Ciana Barrueco, a Database Administrator at Subway, spoke about the time they saved through utilizing ActiveBatch's Reference Jobs. Barrueco said leveraging references was "great, because it meant [they] only had to make the changes once at a template level and those changes are passed down automatically to the referenced jobs. It's a big time saver." Barrueco's team was able to save time in thinking about the function of the changes, rather than the time it would take to go in and make edits to the code of all of the various jobs and workflows.

Revision History & Changesets

Revision History enables ActiveBatch users to focus on the function of their workflows. Within ActiveBatch, every change made is audited to help developers comply with corporate governance requirements. Revision History provides developers with a comparison of any changes made between versions of any ActiveBatch Object. After comparing changes made, if the need arises to revert back to a previous version, the changes can be simply rolled back and reverted to any earlier version of that object.

With ActiveBatch **Changesets**, developers can associate and identify a series of changes as a part of a single check-in. If necessary, this allows developers to rollback sets of multiple changes at one time.

Mistakes happen, requirements change, and sometimes workflow edits need to be rolled back. A coding-based approach would require a developer to locate the change made that needs to be undone and spend time revising code, just to undo that change. Developers using ActiveBatch, however, don't need to worry about these time-consuming processes and can instead check Revision History to see a comparison of the changes made to the ActiveBatch Object in question and can roll back the unwanted change efficiently.

History of ChangeSetsExample				
Changeset ID	User	Date/Time	Objects	Comment
47	COMPANY\johnsmith	2/21/20XX 3:19PM	7	Checked in by COMPANY\johnsmith
48	COMPANY\johnsmith	2/21/20XX 3:18PM	8	Checked in by COMPANY\johnsmith
46	COMPANY\johnsmith	2/13/20XX 1:10PM	11	'Job5' checked in implicitly by COMPANY\johnsmith
45	COMPANY\johnsmith	2/13/20XX 1:10PM	5	'Job4' checked in implicitly by COMPANY\johnsmith
44	COMPANY\johnsmith	2/13/20XX 1:10PM	4	'Job3' checked in implicitly by COMPANY\johnsmith
43	COMPANY\johnsmith	2/13/20XX 1:10PM	12	'Job2' checked in implicitly by COMPANY\johnsmith
42	COMPANY\johnsmith	2/13/20XX 1:10PM	2	'Job1_d' checked in implicitly by COMPANY\johnsmith
41	COMPANY\johnsmith	2/13/20XX 1:10PM	4	'Job1_c' checked in implicitly by COMPANY\johnsmith
40	COMPANY\johnsmith	2/13/20XX 1:10PM	1	'Job1_b' checked in implicitly by COMPANY\johnsmith
39	COMPANY\johnsmith	2/13/20XX 1:10PM	21	'Job1' checked in implicitly by COMPANY\johnsmith

Included in Objects in Changeset 47		
Name	Revision	Operation
[-] /Sandbox/RA		
[-] ChangeSetsExample	2	Modify
[-] PlanA	3	Modify
● Job1	6	Modify
● Job2	3	Modify
[-] PlanB	3	Modify
● Job1	2	Modify
● Job2	3	Modify

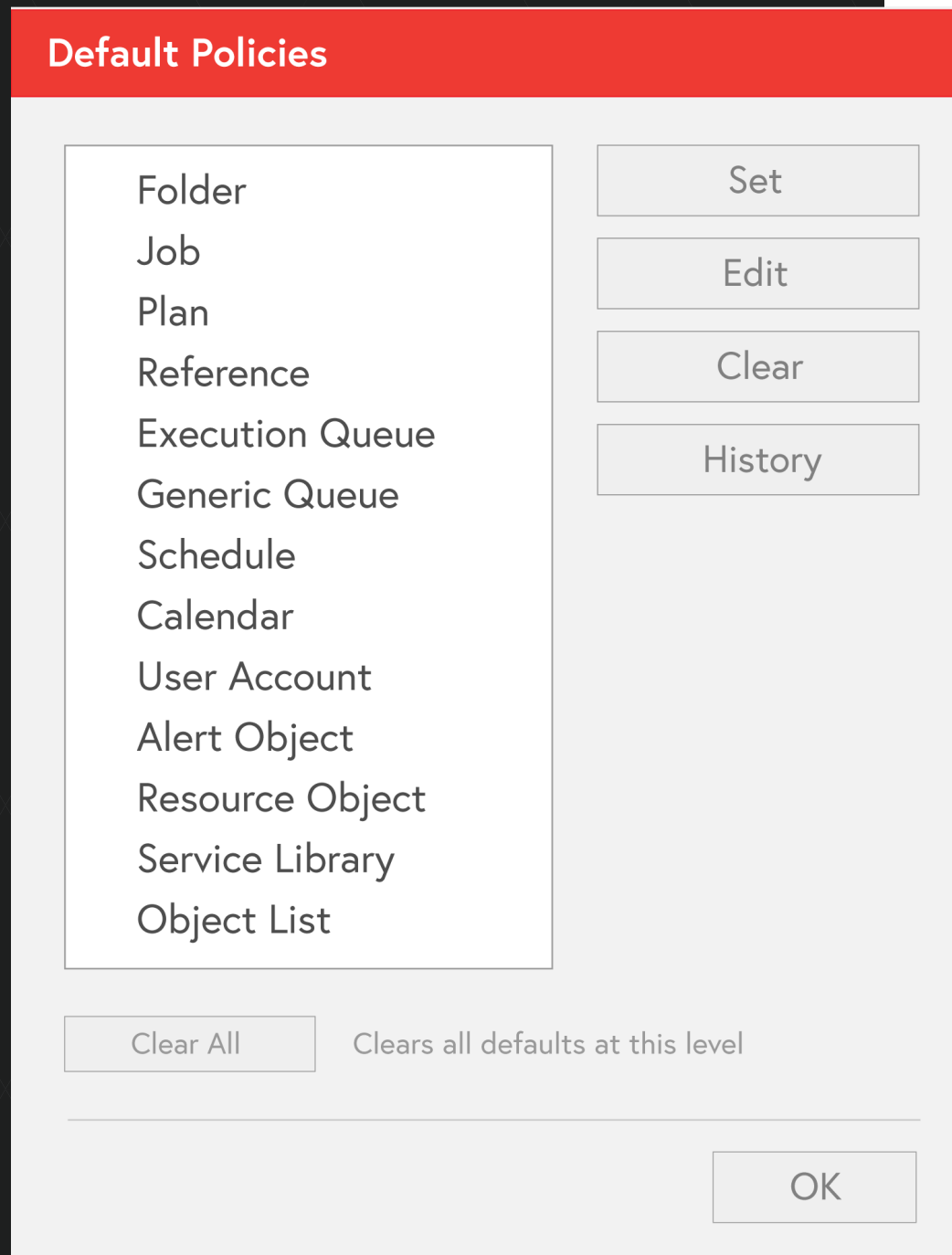
Did you know...

ActiveBatch supports the use of policies to ensure a more secure automation environment. With policies, developers have the ability to

designate customizable audit fields to be mandatory on specific ActiveBatch objects. By doing so, ActiveBatch would require users to obtain approval or authorization before they are able to modify, trigger, or update any ActiveBatch objects. Policies can be set to prevent specific users from seeing, modifying, or running designated objects, jobs, or plans within ActiveBatch.

Policies play a large role for ActiveBatch users who handle secure information—think about banks who manage financial records and funds, or healthcare organizations who must ensure they comply with HIPAA legislation. RelayHealth for example is an ActiveBatch customer who manages medical, pharmaceutical, and dental claims with sensitive, private patient information, 24 hours a day. Before ActiveBatch, they were using scripts to automate the management and processing of these claims, but this method came with complexity. ActiveBatch policies enable users to limit the risk of unauthorized changes to any ActiveBatch object by requiring managerial approval for any changes made to Jobs, Plans, or Variables. Tracy Thompson, Senior Director of Technical Services for RelayHealth, reported, "if any user attempts to change a variable, modify the frequency of a job executing, a plan variable, etc., we are notified of the change via ActiveBatch alert mechanisms and have to approve it."

Many organizations are automating critical processes to reduce manual intervention and reduce the risk of human error. This reduction can be magnified using policies and ActiveBatch developers can have increased control over the entire automation environment.





Lamar Advertising is one of the largest outdoor advertising companies in North America, producing and operating over 315,000 displays across the continent. The organization was facing challenges meeting their real-time business demands with their homegrown job scheduler. Specifically, the advent of digital billboards required real-time data transfers. Between keeping up with requirements like constant image refreshes on billboards and meeting the needs of Lamar's accounting and operations functions, the organization needed efficient IT operations and computing processes. Lamar discovered ActiveBatch in their search for a solution that was flexible, feature-rich, and affordable.

ActiveBatch fulfilled the requirements that Lamar proposed and more—noting Advanced Systems Concepts' highly rated Technical Support Team as a major selling point for them. Lamar reported that the Integrated Jobs Library allowed them to reduce the amount of time and resources that they previously spend on scripting and that ActiveBatch's advanced automation capabilities, such as its event-driven automation (which they utilize heavily for file triggers) have improved their overall IT agility.

Using ActiveBatch, Lamar can use prebuilt Job Steps for virtually every scheduled job they have. Edgar Kirchem, MIS Operations Manager at Lamar said that they've "made heavy use of the Integrated Jobs Library, which has given [them] a lot more time back that [they] used to spend creating and maintaining scripts."

Kirchem also reported that today with ActiveBatch they run "approximately 10 to 12 thousand jobs a day, with a 99% success rate." By leveraging the prebuilt, out-of-the-box functionality provided by ActiveBatch, Lamar Advertising can focus on the function of their jobs and workflows, rather than the custom scripts needed to build them. Prebuilt logic has given Lamar the ability to run more Jobs per day, with a higher success rate. The organization has increased their overall IT efficiency and agility by implementing an IT Automation solution.

Conclusion

It is becoming increasingly important for IT to build and automate their workflows faster and more reliably—and it is critical that if a change needs to be made, which in today's world is far from uncommon, it is easy to access and edit the workflows, jobs, or plans that need to be modified. A solution needs to be designed for change. To stay agile and efficient in today's constantly changing industry, IT organizations need to look to a solution that allows them to focus on the big picture.

An Enterprise IT Automation solution that provides prebuilt logic, a drag-and-drop Workflow Designer, Script Investment Protection, Revision History, as well as Reference Functionality and support for Variables, allows developers to focus on the function of their workflows and IT processes, rather than the coding and custom scripts behind them.

Name	ID	State	Execution Time	Duration	Exit Code	Exit Code Description
Sales_Reporting_Workflow	11401	Succeeded	5/16/2017 2:34:21 PM	67.85 mi...	0	
Sales_Reporting_Workflow	11289	Succeeded	5/15/2017 2:49:01 PM	68.05 mi...	0	
Sales_Reporting_Workflow	11279	Succeeded	5/15/2017 2:47:12 PM	44.12 mi...	0	
Sales_Reporting_Workflow	11276	Succeeded	5/15/2017 2:47:07 PM	44.47 mi...	0	
Sales_Reporting_Workflow	11273	Succeeded	5/15/2017 2:47:03 PM	44.00 mi...	0	
Sales_Reporting_Workflow	11191	Succeeded	5/12/2017 10:51:33 AM	43.95 mi...	0	
Sales_Reporting_Workflow	11079	Succeeded	5/9/2017 12:35:55 PM	45.41 mi...	0	



 ActiveBatch®

Start Redefining Your IT Automation. Learn More
or Request a Demo www.ActiveBatch.com