Thought Leadership White Paper

Workload Automation: Accelerate Digital Services Delivery

Table of Contents

Executive Summary

In most industries today, delivering digital services to customers and employees is essential to business success. Consequently, there is more pressure on the enterprise IT organization to provide the applications that deliver these services faster. This increased speed of application delivery combined with frequent application changes is forcing IT to look for ways to become more agile and scalable, and to accelerate processes and work.

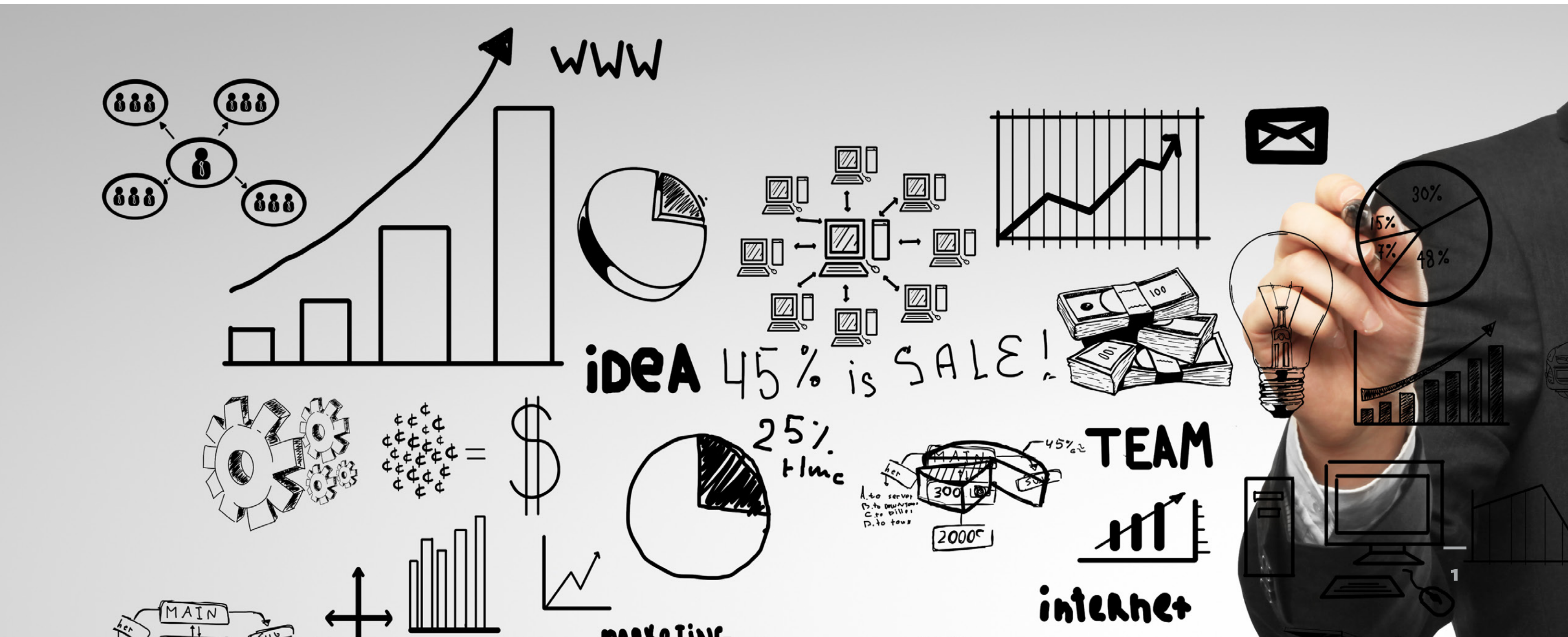
According to industry analysts, 60-70 percent of the application processing within organizations is scheduled and repeatable batch processes. In most organizations, IT has automated the scheduling, monitoring, and management of these processes. But the downstream effort of creating the batch jobs and workflows largely remains a manual process. If IT organizations automate this downstream process, they can substantially accelerate the time to implement an application.

Instead of scripting homegrown wrappers that will later be automated by generic scheduling tools, you can employ a workload automation solution and automate the entire batch process—from design to delivery.

A well-architected workload automation solution will:

- Provide native integration with enterprise applications
- Monitor application performance against service level agreements (SLAs)
- Analyze workload completion status
- Automatically take corrective action in the event of failures
- Proactively alert IT staff of problems that require attention

This paper discusses the obstacles IT organizations face in developing and managing application batch work and how a workload automation solution can remove these barriers while decreasing the time and costs of delivering digital services. It also presents the requirements that such a solution should meet to be effective.



A BUMPY PATH FROM DEVELOPMENT TO PRODUCTION

Whether developing a new application or changing an existing one, IT faces numerous challenges when moving from development to production. This includes designing and managing the batch processes associated with these applications.

Lack of Comprehensive Scheduling Tools

When scheduling batch processes for new applications, application developers commonly use one of the following tools:

- Platform-specific schedulers such as CRON (for UNIX® environments) or Microsoft® Windows® Task Scheduler
- The native scheduling tools available with commercial business applications such as SAP Computing Center Management System (CCMS) and Oozie for Apache™ Hadoop®
- Homegrown schedulers developed in-house by the application developers

Both platform-specific and homegrown schedulers require some form of scripting—a time-consuming and error-prone effort that the application developer usually undertakes. The problem is that application developers are highly skilled in a particular application area and scripting does not leverage this expertise. Using highly specialized and expensive developers for the more mundane task of scripting distracts them from their primary strategic tasks and drives up project costs.

Application-specific schedulers permit scheduling only in their native environments. They are also limited in functionality, sophistication, and ability to manage dependencies, including the workload dependencies within the application.

Difficulty in Meeting Service Delivery Standards

When moving a new application to production, application developers often find a gap between their expectations and those of the production control team with regard to how the application's batch activity needs to be managed. For example, the production control team may expect the application to notify the operations staff when service level agreements (SLAs) are missed or about to be missed. Or they may want it to automatically recover from common job failures, such as when a job ends with a certain exit code or with an error message in its output. Some stakeholders may expect the application to automatically generate incident tickets for the service desk, so that the organization can track to problem resolution when jobs fail.

In many cases, the application designers, architects, and developers are not fully aware of all the additional functionality that the operations staff requires. As a result, the developer may be asked to retrofit these for the application. Because these additions are typically afterthoughts, rather than built in from the beginning, they often create application inefficiencies and fragility. They also add more time and cost to development efforts.

Islands of Automation

Batch services typically span multiple disparate operating systems (e.g., z/OS®, Windows, UNIX, Linux®, and iSeries®) and different enterprise business applications (e.g., SAP, Hadoop, data integration and business intelligence, and file transfers). Some of these platforms and applications provide native scheduling tools, but they only support scheduling within the native application environment. This creates "islands of automation."

To manage batch activity in these heterogeneous environments, the production control team members must navigate multiple tools and acquire a wide range of expertise. That's just the beginning. The team members then resort to scripting to glue together the dependencies between the different environments. This fragmented approach lacks standardization, making it difficult to monitor, maintain, support, and update batch services.

Communication Barrier

A lack of a common language between developers, schedulers, and production control personnel hampers collaboration and creates additional inefficiencies. In many cases, application developers are not aware that there is a standard workload automation solution in place until they hand off the application to production control. At this point, they have already spent precious time and resources.

Even if developers are aware of the workload automation solution in place, their requests for new workflows or changes to existing workflows have to be communicated to the scheduling team. However, because of the communication barrier, the developers' requests may not be in a format that the schedulers can easily understand. That introduces confusion that creates more delay.

FROM PATH TO SUPERHIGHWAY

Just smoothing the bumps in the development-to-production path is not the most effective approach. To maximize efficiency and cost reduction, organizations should look into an approach that upgrades processes and removes the bumps to create a high-speed thoroughfare. This approach mirrors the DevOps discipline guidelines, which are intended to speed the applications release lifecycle by creating a more collaborative and productive relationship between development and operations teams.

You can implement a high-speed thoroughfare using a well-architected workload automation solution. To be effective, the solution must meet a number of stringent requirements.

Eliminating Scripting

Using scripting to schedule batch work is one of the primary reasons it takes so long to move applications from development to production. A workload automation solution that offers native integration with enterprise applications can eliminate the effort and costs resulting from scripting by interfacing directly with the application APIs. This permits highly skilled developers to focus their expertise and time on strategic activities. In addition, such a solution provides a standardized scheduling method across all workloads.

In addition to eliminating scripting, the workload automation solution enables you to:

- Analyze workload output
- Determine whether workloads complete successfully or fail
- Configure automatic recovery from errors
- Generate alerts when needed
- Restart workloads from point of failure

A well-architected solution also tracks and correlates changes with workflows. This way, you can quickly ascribe failures to recent changes, contact the change owner, compare workload versions, and if needed, roll back to previous workload versions to restore operation. You can do all this without requiring the production control team to login to the problematic application server and manually edit a script, possibly resulting in lengthy downtime. As a result, the production control team spends less time on the phone with the on-duty application developer or database administrator (DBA) when failures occur outside business hours.

Sophisticated Scheduling and Forecasting

Most popular scheduling tools, such as the native schedulers in enterprise business applications, are limited in the scheduling granularity they offer. For example, they can schedule a workflow to run every Monday between 2:00 p.m. and 3:00 p.m.

An effective workload automation solution allows more sophisticated scheduling. For example, it can schedule a workflow to run on the last day of each month, unless it is a holiday or the schedule conflicts with another type of calendar. The solution also supports intricate dependencies, such as “run job C only if job A completed with a particular exit code and job B didn’t complete successfully.”

An effective solution also simplifies scheduling workloads that run across multiple geographical locations and time zones. This capability eliminates the need for application developers or production control personnel to account for differences when they schedule workloads that run in remote locations.

Predictive Analytics

Unanticipated scheduling problems can have serious business consequences. Failure to update point-of-sale terminals can result in wrong prices being charged to customers. Failure to update a just-in-time inventory system can halt a manufacturing line. It is essential to ensure batch work can be executed successfully and within their specified time windows.

Consequently, it’s important to be able to forecast how batch work is expected to execute at future times. For example, if the solution knows when a batch process is scheduled to run, it can provide a predictive analytics capability that indicates, based on analysis of historical data, whether that process will complete within its specified SLA at the future date and time.

Another use of a predictive capability is to enable “what-if” analysis. For example, the solution can predict what will happen if a batch workflow runs 20 percent longer than normal due to a seasonal peak in transaction volume. Or it can predict what will happen if a particular server is taken down for maintenance from 2:00 p.m.to 4:00 p.m. next Thursday.

Supporting Production Control Requirements

In an enterprise environment, applications must comply with IT standards and integrate with existing monitoring, management, and security systems. That's why the workload automation solution must enable the developer to define and schedule production control functions before moving the application to production. With this capability, developers can build required production control capabilities into the application during the development phase—rather than tacking them on later. This eliminates the need to retrofit applications and shuffle them between development and production. This results in shorter development cycles and more efficient, durable applications.

To include production control functions, the workload automation solution must integrate with other IT systems. Such integration is not practical in a script-based scheduling environment.

Integration with an IT service management (ITSM) system enables the workload automation solution to generate an incident ticket when a workload fails or when a batch service is delayed. In addition, the corrective action taken to handle these failures or delays should be associated with those tickets.

Integration with an authentication repository, such as LDAP or Active Directory®, eliminates the need to manage redundant user credentials and enables higher levels of security. It also relieves the application developer and production control team from duties that are usually within the purview of the IT security team.

Finally, integration with an IT change management system ensures that the batch aspect is taken into consideration when planning a change such as maintaining a server that participates in a batch activity. A comprehensive workload automation solution can recommend a time window to perform the server maintenance that will not disrupt batch operation.

Associating Workloads with Business Services

IT organizations continually strive to align more closely with the business. This requires associating batch workflows with business services and associated SLAs. This association gives the production control team visibility into the business relevance of workloads. With this visibility, the team can make informed decisions and prioritize how they will respond to failures or delays.

Allowing application developers to associate workloads to business services and SLAs at the early stages of a project eliminates risk, ensures that production control will have the needed visibility into the business context of the services, and shortens the project schedule. Waiting to do this until the later stages of the project may require application developers to retroactively map workloads to business services weeks or months after they have been created.

The workload automation solution should issue an alert whenever a workflow or associated batch service is running slower than allowed by the relevant SLA. This alert warns production control and the business service owner that an SLA is in danger of being violated. Advance notification enables production control to move proactively to handle issues before they disrupt the business.

Batch services may run slower than expected when the host servers are running beyond capacity. Attempting to balance these workloads in a script-based scheduling environment is extremely difficult. It requires production control to understand each of the individual workloads and their associated SLAs, so they can assign the appropriate priorities. It may also require production control to manually modify scripts to defer non-critical workloads to off-peak hours or to reassign critical workloads to alternate servers than are running under capacity.

A workload automation solution that performs automatic, policy-based workload balancing eliminates the effort and risk involved with manually modifying scripts. The solution also shortens the development time when the application developer opts to implement load balancing within the new application.

Self Service

The workload automation solution should provide an easy-to-use, self-service portal that permits application developers to submit requests for new workloads or changes to existing ones, without requiring scripting or workload automation expertise. A graphical interface allows developers to assemble workloads into an automated process using simple drag-and-drop operations to establish workflow and dependencies.

The solution should provide pre-defined templates to ensure proper naming conventions and site-standard polices for workloads. Developers should be able to assign pertinent attributes to the workloads such as time constraints, target destination, and relationships to the business services they support.

The graphical interface should also allow application developers to monitor workloads in pre-production environments such as during application testing and quality control phases.

When an alert is issued for a failed or delayed workload in the production environment, the graphical interface should permit the application developer to examine the workload log and output. This information enables the developer to decide how to correct the problem. The interface shortens the problem resolution cycle by allowing access to the log and output from a remote location without requiring login to the application.

What's more, a self-service portal that provides information that is meaningful to both the application developers and the production control team eliminates the communication barrier between the two groups. As a result, they can collaborate more effectively to reduce problem analysis and resolution times.

ELIMINATING ISLANDS OF AUTOMATION

Managing dependencies among workflows that span multiple systems and platforms is extremely difficult in a script-based environment. It typically involves both scripting and using file transfers to enable communication among the different systems.

A workload automation solution that supports heterogeneous environments eliminates islands of automation and related inefficiencies. It replaces multiple, disparate scheduling tools with a single device that requires only a single skill set. It also standardizes applications, making them easier to monitor, maintain, support, and update. Finally, it enables developers to specify, in seconds, dependencies among workloads that span multiple systems, regardless of the platforms, applications, geographical locations, or time zones in which the workloads run.

Here are two examples of how a workload automation solution can help eliminate islands of automation.

Introducing Hadoop to the Enterprise

Hadoop is a software framework for storing and processing large volumes of data. It helps IT drive down costs by using a cluster of low-cost commodity hardware rather than an expensive high-end, high-capacity machine.

A common Hadoop implementation involves integrating Hadoop processes with other enterprise applications such as data integration (ETL) and business intelligence tools, ERP applications, database queries, and file transfers.

Here's a typical example:

> Workflow initiates when the system receives data from an external vendor. The data is loaded into the Hadoop Distributed File System (HDFS™) and processed by MapReduce or PIG™ programs. The processed data is then transferred into an SAP data warehouse, which generates a business report.

This workflow cannot be automated with a Hadoop-specific scheduler such as Oozie, because it is restricted to the Hadoop environment and cannot be used to automate non-Hadoop processes or integrate Hadoop with non-Hadoop applications.

A workload automation solution that natively integrates with all of the systems involved in such a workflow enables the developer to schedule the workflow across all systems—without requiring any scripting.

Automating SAP Processes

Some ERP applications offer native scheduling capabilities. However, they support scheduling only within the native ERP environment. For example, SAP CCMS operates solely within the SAP environment and does not allow SAP applications to integrate with systems outside the SAP environment. Here again, a workload automation solution that offers native integration with SAP and other systems enables scheduling within the SAP environment and the integration of SAP processes with other environments.

CONCLUSION

The faster delivery of applications is essential for success in most organizations today. Consequently, it's imperative that the IT organization has the agility to implement the applications the business needs in the time frames required. That means compressing the application development cycle as much as possible.

Managing application batch processing may consume a significant portion of the application developers' time and effort, which drives up application development times. Simplifying batch management presents an attractive opportunity for time and cost savings.

A well-architected workload automation solution eliminates the batch-scheduling obstacles that delay development cycles. Such a solution enables IT to accelerate the delivery of applications and their digital services, resulting in a faster time-to-application value.

FOR MORE INFORMATION

To learn more about Workload Automation, please visit **bmc.com/it-solutions/workload-automation.html**

BMC delivers software solutions that help IT transform digital enterprises for the ultimate competitive business advantage. From mainframe to cloud to mobile, we pair high-speed digital innovation with robust IT industrialization—allowing our customers to provide amazing user experiences with optimized IT performance, cost, compliance, and productivity. We believe :

- **Technology is the heart of every business**
- **IT drives business to the digital age**

BMC – Bring IT to Life