PLANNING FOR DEVOPS IN MAINFRAME ENVIRONMENTS

– KEY CONSIDERATIONS –

A mountain of information is available around the implementation and practice of DevOps, collectively encompassing people, processes and technology. This ranges significantly from the implementation of tool chains to discussions around DevOps principles, devoid of technology. While technology itself does not guarantee a successful DevOps implementation, the right tools and solutions can help companies better enable a DevOps culture.

Unfortunately, the majority of conversations regarding DevOps center around distributed environments, neglecting the platform that still runs more than 70% of the business and transaction systems in the world; the mainframe. It is estimated that 5 billion lines of new COBOL code are added to live systems every year. Furthermore, 55% of enterprise apps and an estimated 80% of mobile transactions touch the mainframe at some point.

Like their distributed brethren, mainframe applications benefit from DevOps practices, boosting speed and lowering risk. Instead of continuing to silo mainframe and distributed development, it behooves practitioners to increase communication and coordination between these environments to help address the pressure of delivering release cycles at an accelerated pace, while improving product and service quality. And some vendors are doing just that.

Today, vendors are increasing their focus as it relates to the modernization of the mainframe, to not only better navigate the unavoidable generation shift in mainframe staffing, but to also better bridge mainframe and distributed DevOps cultures.

DevOps is a significant topic incorporating community and culture, application and infrastructure development, operations and orchestration, and many more supporting concepts. In short, DevOps incorporates multiple tools and approaches that will vary depending on the unique needs of the customer. With this, the following offers some key thoughts and considerations to help in the evaluation and adoption of a mainframe inclusive DevOps culture.

1 How does the proposed technology facilitate effective communication and collaboration, bringing together processes, tools and staff across mainframe and distributed development, operations and management teams?

Will it facilitate communication across environments and IT disciplines?

Today's "mobile-to-mainframe" business applications are rarely isolated to distributed or mainframe environments. Coordination between front-end developers and system of record API creators, for example, is pertinent to minimize miscommunication and subsequent delays.

Will it connect silos without dictating conformance to one another's workflow / development methodologies?

A key issue DevOps helps address is one of silos. They negatively isolate work into separate units within an organization, typically resulting in poor communication with predictable complications. Silos increase difficulty around collisions of opposing processes, cultures, and technologies. The first step is removing the segregation between organizational silos and moving to a collaborative model where teams and individuals can work together. A highly configurable solution however, will allow existing processes and technologies to be used while coordinating them as part of a larger whole.

Are all stakeholders included?	More than collaboration between development and operations, a DevOps solution needs to extend to everyone with a stake (e.g., development, QA, operations, product management, and business executives). Incorporating business analysts, initial product planning, and ultimately customer feedback can speed delivery while improving quality, security, and business outcomes.

2 Will the solutions integrate well into the existing environment, providing open, extensive and flexible tooling?

Does the solution integrate with existing systems, processes and agile methodologies?	Every organization incorporates different best practices around software development. Processes are often unique to how different teams work together, possibly even within the same organization. Similarly, there are a plethora of tools available for different elements and stages of DevOps. More than simply integrating with various tools, a solution needs to be able to link them together across the lifecycle. Synchronizing information across these tools facilitates a smooth transition as code and metadata (e.g., requirement and configuration) flow from one state to the next without manual transcription between tools. A highly configurable solution helps meet the needs of these individualized workflows and methodologies, adapting to processes as opposed to forcing customers to adapt to an implemented solution, which would otherwise increase friction and decrease user adoption.
How is user acceptance and adoption addressed?	Adoption of modern tools and techniques throughout the organization can sometimes be patchy and inconsistent. For example, multiple teams doing agile development may already have varying degrees of discipline and automation, while using traditional techniques in other areas. Forcing users to migrate to new tools introduces learning curves and inevitably will be met with resistance. A solution that can integrate with existing tools allows developers and operations to pick and choose, helping increase user acceptance and rapid adoption.
Can developers continue using their preferred IDE, or are they required to switch?	While a solution should integrate with existing tools, it should also leave room for the introduction of new tools that can be used in parallel. For example, well-seasoned mainframe developers can continue to use their preferred methods and interfaces. As they retire, new incoming millennial developers can simultaneously use visual drag and drop tools that better mimic their distributed counterparts.
How is integration achieved?	Some of the best individual tools fall short on integration. Must a customer compromise in some areas to achieve a comprehensive system that works well together in order to automate the full DevOps process? Or is a vendor solution able to overcome these shortcomings? Partnering with third party vendors to provide out of the box integration, or the use of plug-ins (e.g., Eclipse), can significantly accelerate integration efforts. But when pre-existing integration is not available, is a framework to ease the integration with even the most obscure COTS or home grown solutions available? How well existing solutions are integrated (i.e., quality over quantity) can be a significant decision factor.
How many third party solutions are required?	Although capitalizing on a customer's existing investments is important, they may still elect to replace any number of tools, or simply require new tools to address current shortcomings. Can a prospective vendor address all these areas or will additional third party solutions be required? Even when third party tools are integrated, organically grown solutions from a single vendor will typically provide better integration for a more seamless experience.

3 How does the solution enable an organization to deliver DevOps practices across the Software Development Lifecycle, mobile to mainframe—including agile development, continuous testing, continuous deployment, and agile operations?

Will it facilitate adoption of modern agile development methodologies?	Does the solution offer intuitive, enterprise grade development tools that support both millennial and experienced staff with modern methodologies and coding languages, helping them adapt as agile processes evolve? Effective solutions reduce time to market by lowering barriers associated with skills and code base age, and help make sure that the time used and the effort expended are appropriate and valuable.
Will it reduce long lead times in enabling Continuous Testing?	An encompassing solution provides facilities for automated testing and debugging, but further helps streamline the test environment setup, manage complicated test scripts and inter-relationships, and fulfills 100% of testing data needs. Ideally, a solution provides facilities for complete defect/issue management, test case and test plan management, and test automation without a requirement for third party solutions; all while ensuring comprehensive security and audit integrity.
Will it reduce time to market through reliable Continuous Deployment?	A reliable continuous delivery process automates and orchestrates key application testing measures to ensure that multi-team, cross-application composite releases can rapidly, but reliably, be deployed to production with high quality. Support for continuous deployment is the next step, meaning that every change that passes automated testing is deployable either automatically or through an approval process where constrained by regulatory or other requirements.
Can it use data and predictive intelligence to deliver a superior customer experience?	A superior customer experience derived from efficient IT operations requires an agile operations process that can unify monitoring and performance management across multi-platform applications for better mean-time-to-resolution and lower resource costs. Integrated performance monitoring for applications not only provides feedback for development teams but can quickly isolate code responsible for excessive CPU consumption to reduce hardware costs and more importantly customer turnover.

4 Is the solution enterprise-class, providing the scalability, auditability, and data security required to meet the needs of multiple departments that must coordinate on software delivery?

Will it provide end-to-end managerial visibility?	While most DevOps discussions center around the development and operations teams, executive leadership and other stakeholders need to be able to measure and monitor performance. While individual tools may provide their own form of reporting or dashboards, these offer only a snapshot of their own individually pertinent information. A complete DevOps solution that facilitates different teams and individuals to collaborate in one console should also provide universal access to relevant application and development intelligence in order to optimize software strategies. This ideally would include the necessary tools to create custom reports that best suit the needs of the user.
Does it include complete traceability?	Similarly, individual tools often (but not always) provide some form of relationship management of program components, versioning, and/or auditing to provide tracking of changes and facilitate recovery to older versions. From an end-to-end perspective, an implemented solution should make these details available through

a single source, meaning that customers can use any combination of third party products but have access to complete audit trail information without a requirement to manually switch between multiple tools and interfaces.

Are there facilities for intelligent alert management?

Rare is the case today where management tools don't provide some form of notification. What varies however is the type, channel and/or extent of notification. Well articulated monitoring and alert management solutions support more complex notification generation with highly dynamic content that can include reports or other attachment data. In combination with workflow engines, notifications can be finely controlled making them far more useful in and of themselves (i.e., self contained) without a requirement to force users to revert to the respective solution console.

Can it help facilitate future application planning?

While Project Planning and Project Management are not to be confused with governance itself, a well-designed solution can bring together processes, tools, and staff across development, operations, and management teams, helping bring all stakeholders into consideration. They facilitate sharing ideas with other stakeholders, the capture and management of changes to business needs and expectations, and support the collection of feedback on ideas and business needs. Similar to the way that visualizations help when looking for the route cause of failures, they can provide valuable insight during the planning process.

5 What is the potential for improvement in Time to Market (TTM) and Mean Time to Resolution (MTTR) by leveraging this toolset? Is there an overall positive ROI?

Will it enable Bimodal IT, improving the value returned by existing investments?

Whether or not customers are planning to implement aspects of agile into their mainframe development, the implementation of a DevOps culture is not about throwing out existing, open source, and/or best of breed solutions and replacing them with large integrated suites of commercial software. A solution needs to be able to integrate and capitalize on existing investments while simultaneously helping renovate these legacy assets to play an essential role in the digital economy. A solution that supports Bimodal implementations recognizes the need to adopt differentiated but simultaneous styles and approaches, while providing a predictable evolution of products and technologies.

More than just a collection of tools, does a vendor solution help coordinate these various tools?

DevOps relies on tool chains to automate all parts of the end-to-end software development and deployment process. This can include tools that were built internally, tools that are purchased, open source tools, and proprietary tools. Many organizations may also have a semblance of automation already in place. Will a solution allow established incumbents to remain in place and be integrated as part of larger automated workflow processes?

Will staff need to be retrained?

A requirement to abruptly switch tools versus simply integrating existing known tools adds migration and training costs. It can also decrease user acceptance and subsequently increase the probability of failure. The ability to coordinate and include existing tools virtually eliminates retraining requirements. More than simply maintaining the status quo however, a solution should enable the introduction of new tools and processes to help customers achieve any desired advancement up the agile and DevOps spectrums. This Bimodal approach can help build momentum and buy-in of transformations.

What are the hidden costs?

While the feature functionality between two vendors may be comparable for a fully implemented solution, how many different product licenses are required? Can a

vendor even provide all the necessary tools or do additional third party products (and licensing) need to be introduced; even if they are integrated? An extensive (and well integrated) vendor portfolio can help address the many varied needs of a customer's unique situation.

Are professional services available? Finally, the availability of Professional Services for implementation and transition can help customers to quickly implement a DevOps methodology. Can a selected vendor provide its own long-standing experts or are implementation partners required to supplement?

Adopting tools doesn't magically create a DevOps culture, but designing and sharing the right tool chain for the organization can bring about important benefits. A clearly laid out tool chain illustrates the flow of work from inception to production, which improves visibility into the work to be done and promotes continuous feedback. Typically such a tool chain requires infrastructure, provisioning, and configuration management tools for testing and deployment, but also build/delivery pipelines to move code from source control all the way to running in production.

Just like its distributed counterpart, DevOps tool chains inclusive of the mainframe deliver major competitive advantages by enabling much more frequent delivery of mainframe code, while making the platform much more accessible to DevOps teams.

While the focus here has been on implementing a DevOps culture for mainframe application development, it is still important to consider interaction and collaboration outside of the mainframe. Today's business applications are rarely isolated to mainframe or distributed platforms. As discussed, an implemented DevOps culture should allow staff to continue using their existing toolsets and processes, distributed or mainframe, but integrate them together to provide a better overall view of processes, and help staff better communicate and coordinate efforts for increased productivity between teams so mainframe and distributed teams can work together as one.

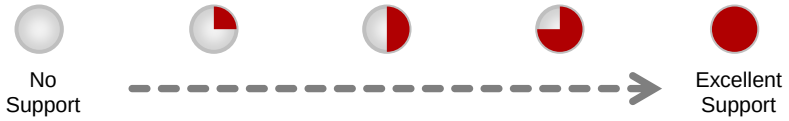
DEVOPS FOR MAINFRAME

– A COMPARATIVE ASSESSMENT –

The following valuation provides a brief look at two vendor solutions that offer DevOps capabilities for the mainframe. The vendors we reviewed approach DevOps from different perspectives: CA Technologies, anchored by CA Application Lifecycle Conductor leverages the embrace and transition approach, and Compuware, anchored by Compuware Topaz Workbench leverages the new tooling for the enterprise approach.

Functional Area	Compuware	CA
Project Planning and Management	Partnered with Atlassian® for JIRA® integration	CA Application Lifecycle Conductor, CA Agile Central
Analysis	Compuware Topaz™ for Program Analysis	N/A
Integrated Development Environment	Compuware Topaz Workbench	CA Development Environment for z Systems or Eclipse plugins for IBM Developer for z Systems (formerly IBM Rational Developer for z Systems)
Source Code Management	ISPW™	CA Endevor® SCM / CA Harvest SCM
Data Visualization, Editing and Management	Compuware File-AID, Compuware Topaz for Enterprise Data	CA File Master™ Plus, CA RC/Extract™ for DB2 for z/OS, CA Fast Load® for DB2 for z/OS, CA Fast Unload® for DB2 for z/OS, CA Quick Copy for DB2 for z/OS
Test Data Generation & Masking	Compuware File-AID for Data Privacy	CA Test Data Manager (including automation and data masking), CA Test Case Optimizer
Testing / Debugging	Compuware Xpediter, Compuware Hiperstation	CA InterTest™ Solutions, CA Verify, CA Application Lifecycle Conductor Mainframe DevOps Catalog
Release Automation	ISPW Deploy	CA Release Automation
Diagnostics	Compuware Abend-AID	CA SymDump® Solutions
Application Tuning / Monitoring	Compuware Strobe™	CA Mainframe Application Tuner

While prospective customers may not be left wanting where these vendors compare directly, there may be significant personal preferences that influence their decision with regard to implementation strategies. For example, while CA provides an application lifecycle management solution that has significant value with regard to allowing staff to continue to use their existing products, Compuware is focusing on its IDE to help bridge the gap between distributed and mainframe development, hopefully helping address the impending mainframe skills shortage.



Core Evaluation Factors		
Feature	**CA DevOps for Mainframe**	**Compuware DevOps for Mainframe**
Platform Support		
Installation Process		
Server OS Support (installation)		
Client OS Support		
Database Support (Installation)		
User Interface Browser Support		
Software Development Lifecycle		
Overall Interface(s)		
Problem Ticket / Issue Management		
Program Analysis		
Agile / Project Planning and Management		
Workflow Engine		
Collaboration		
Source Code Management		
Artifact / Package Management		
Relationship Management		
Versioning		
IDE / Workbench		
Compile / Build		

Core Evaluation Factors		
Feature	**CA DevOps for Mainframe**	**Compuware DevOps for Mainframe**
Continuous Integration		
Debugging		
Unit and Integration Testing / QA		
Approval / Change Management		
Release Automation		
Deployment		
Data Management		
Test Data Generation and Automation		
Data Security and Compliance		
Roll Back Facilities		
Monitoring (Application Performance Management)		
Integration and Extension of Enterprise Assets		
Extension of Solution with Vendor Portfolio		
Additional Third Party Integration / Support		
Configuration and Administrative Management		
Auditing		
Security (Authentication and Authorization)		
Monitoring, Notifications and Alerts		
Maintenance (Deployment, Rollout, and Upgrading)		
Reporting and Dashboards		

While this assessment was commissioned by CA Inc., Zibis Group does not endorse either vendors' solution, rather profiling them in this instance to illustrate several areas of consideration, specific to the use of modern technology and techniques, and towards the adoption of a mainframe inclusive DevOps strategy.

ABOUT ZIBIS GROUP

Zibis Group Inc. ("Zibis Group") is a leading source for market research and advisory services on the Information and Communications Technology (ICT) industry and related sectors. We assist vendors, government institutions, academia, non-profits and other key industry stakeholders by providing them with the tools they need to make informed decisions.

Zibis Group is a niche play in the marketplace for customized ICT sector research. We understand that all companies have distinct interests and unique business challenges. We thrive on rising to the occasion and in seeing our clients succeed. Technology is changing and we're here to help.

For more information regarding Zibis Group, please visit **www.zibisgroup.com**.

ZibisGroup
555 Legget Drive, Suite 304 Tower A
Ottawa, Ontario
Canada K2K 2X3
Tel: +1 613-518-8006
www.zibisgroup.com