~~Accelerate Monetization with Zuora Devops~~

Monetization and Devops

Monetization strategies are complex but the customer and user experiences require it to be simple and easy. Your monetization platform must provide a consistent customer experience throughout the journey of the customer starting from subscribing to reading invoices to creating and modifying payment methods. The user experience in the subscription platform is more on the business side where the platform should be easy to use for the accounting, finance and IT admins who are managing subscriptions and revenue.Customers expect services to be available whenever they need them. Any disruption or issues in customer experience impacts customer satisfaction and trust.There is a constant need for the businesses to innovate, maintain and release faster. Implementing a Devops strategy enables the businesses to build a reliable, scalable solution which is resilient to issues, ultimately safeguarding the bottom line and ensuring the success of customer facing initiatives.

The need to “go live faster” and constantly ” iterate” for improving the current application is driven by competition and constant innovation required meeting customer expectations, generating revenue and maintaining agility. It is even all the more compelling for Subscription businesses to consider faster software release cycles for managing customer engagement and key success metrics like MRR, ARR etc.

Businesses need to prioritize product readiness and improve the production environment for reducing time to market. The business value of Devops lies in faster releases, high quality and optimisation of resourcing and tooling costs.

What is Devops?

Devops enables coordination and collaboration between formerly siloed roles through tools, processes and people. Devops is a culture or an organization approach where everyone has access to Devops Tools and is responsible for all aspects of the product from development, IT operations, quality engineering, security and vulnerability compliance to release managers.

Why Devops Strategy is Required for Zuora?

The estimated growth for the Devops market is expected to achieve a compound annual growth rate (CAGR) of 19.7% within a span of 5 years from 2023 till 2028. They key drivers for growth in this market are:

- Increasing demand for continuous and faster application delivery
- Increasing growth of microservices architectures over monolith applications. Microservices are created, tested and deployed independently thus it is easier and faster to support releases and bug fixes whereas a monolith application requires a complete version release for new releases and makes it difficult to introduce changes.
- Increasing demand for streamlining collaboration between IT and operations teams for breaking the silos and involving all the teams and SME engaged in software development and its lifecycle.
  {Source: MarketsandMarkets}

Financial processes implemented on Zuora are varied, ranging from straight forward subscriptions scenarios to highly complex ones. The solutions are tailored to the unique needs of each organization. Zuora provides a range of customizable features from pricing models, payment methods, notifications to customers for managing subscriptions to dunnings, advanced custom fields to extension studios for managing configurations that enable our customers to craft business operation specific solutions. For coping with increasing complexity from both internal and external factors, it is inevitable for businesses to adopt a Devops strategy for achieving :

- Reduce time to market by shorter release cycles for bug fixes and new feature releases through automation
- High quality and governance as a predictable and repeatable process is utilized for development, testing and deployment

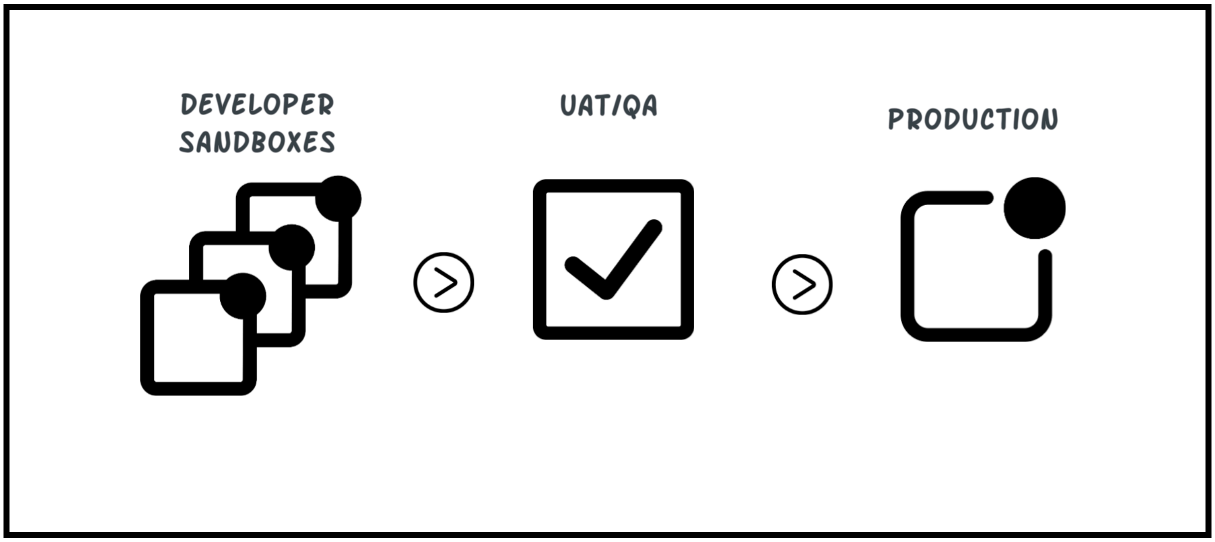
Best Devops Practices for managing Zuora landscape

Managing Zuora Environments

Develop a testing approach and determine which environments will be required to execute the strategy. A good practice is to have at least three or more stages in the deployment process for releasing to production.

1. Development sandbox(es)- The environment where the developers will pursue development and testing.
2. Merging environment-Integrating environment where the code from all the dev environments are synced and UAT is done. Zuora Central Sandbox mirrors production data, this makes it an ideal environment for testing integrations and syncing all the isolated developments into one and testing how they will work in Production.
3. Production environment- It is operationally live environment where subscribers access the catalog and services. It has to be highly stable and reliable for handling large volumes of transactions and it impacts the customer experience directly.

The Developer environments should be used for development, testing new and existing customizations, ensuring that Production data, integrations and applications are not compromised.



There may be a merging environment, often named as SIT environment where all the changes from various developers are merged and tested. The UAT environment is a production- like environment for understanding how the changes will work in the Production environment.

It is recommended to keep refreshing the lower environments with Production data. Baseline development and testing environments with production data like data ensuring that lower environments accurately mirror the production environment. This consistency helps in identifying and resolving issues early in the development lifecycle.

Define a Testing Strategy

Automate tests to cover the large majority of your use cases. Automated Test Scripts programmatically perform validations, ensuring efficiency and accuracy.

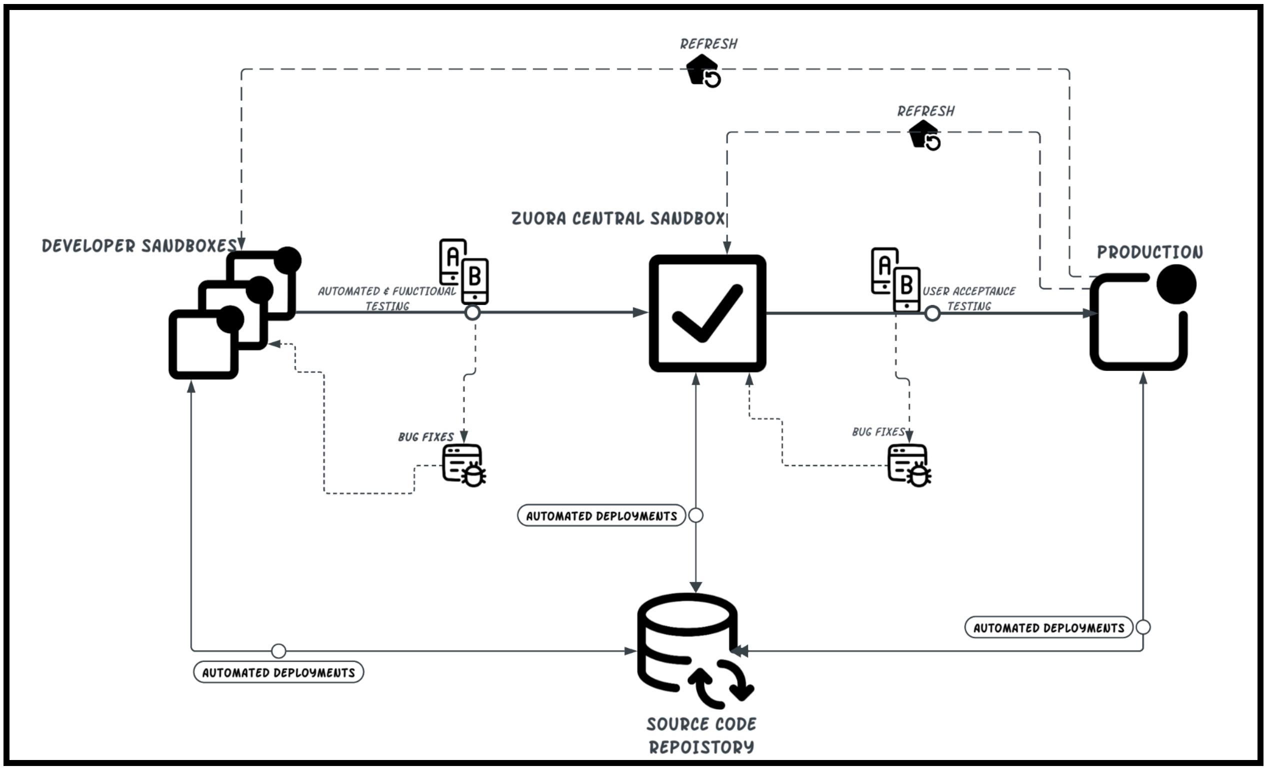
In the case of major releases and changes directly impacting the business and subscriptions, another good practice is to run the UAT in a production-like environment. This will help in understanding the change and certify that it works as expected.The bad practice would be to modify your changes in production, which unfortunately happens more often. As a result,

there is no control over errors and failures, which will directly impact the front end.
Test using relevant production data and users.Usually the trend is to deploy the last successful set of configurations tested in the UAT environment. Regardless of the approach, don't skip testing.

Automated Collaboration

In low code platforms and also working in a shared environment, it becomes difficult to track changes and issues. Implementing a version control is essential for tracking changes to metadata.It facilitates improved collaboration among teams for tasks such as peer reviews and implementing merging and branching strategies for deploying programmatically to various tenants.These strategies are essential for managing releases, bug fixes, and maintaining an audit trail of changes, including details on who initiated the changes and when they occurred.

When the Version Control System that acts as a source of truth for a given application, the continuous merging of the changes triggers a series of automated builds and tests. By merging changes continuously, the bugs identified are earlier, fixed and validated faster and released faster. After the success criteria post testing and integration have been met, the changes can be deployed to production, thus reducing time to market.



Monitoring and Observability

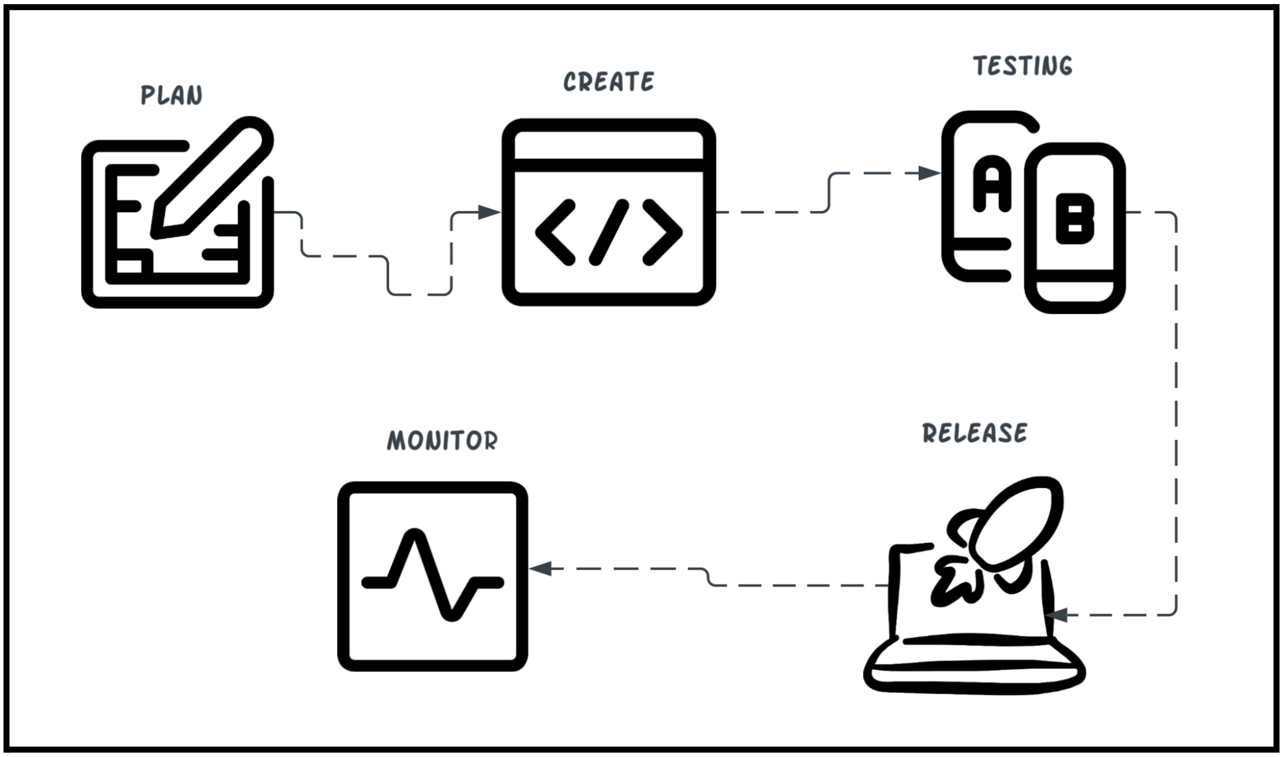
The automated processes integrated with DevOps tools necessitate continuous monitoring and observability across the entire development and release lifecycle. This entails seamless integration between tools and collaboration among IT and operations teams. DevOps monitoring tools play a crucial role in providing visibility throughout these processes, aiding in achieving efficient monitoring and observability across the entire development and release pipeline.

DevOps and Application Lifecycle Management with Zuora

ALM encompasses the entire process of software development, while DevOps focuses on the methodology used within that process. For example, testing is a component of ALM. If an organization decides to incorporate automated testing through source code-driven development, the organization is leveraging DevOps practices. When ALM is synchronized with DevOps, the objective is to attain quicker and higher quality releases.

What is Application Lifecycle Management?

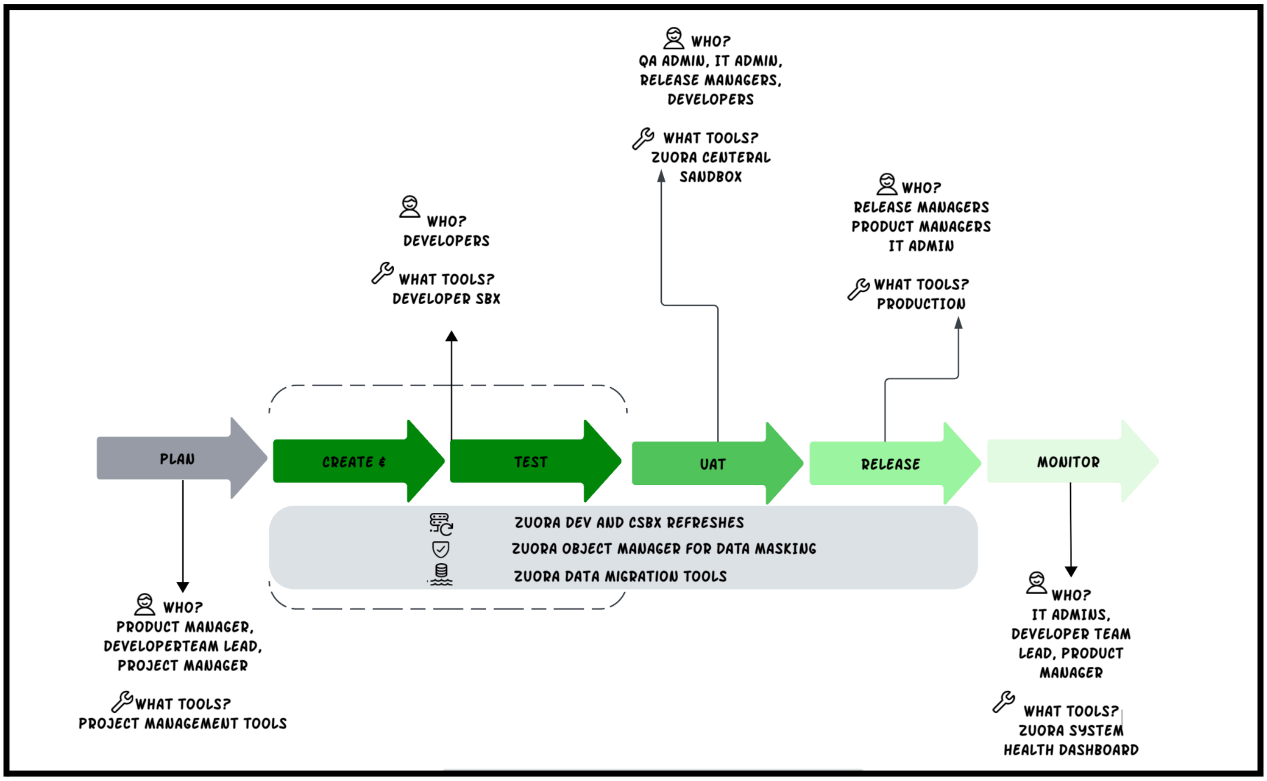
Application Lifecycle Management is a set of processes and tools that an organization can define and implement for maintaining the application, development, testing and deployments. There are five typical phases for ALM.



Zuora Application Lifecycle Management Tools

Zuora provides an extensive framework of tools and isolated sandboxes designed to support its customers at various stages of development, testing and monitoring system health. Both Zuora Central Sandbox and Zuora Developer Sandbox are interconnected with production for unification of data and the developers have access to production data in an isolated environment where they can develop and test. Zuora Developer and Central Sandboxes offer the inherent capability to refresh data from the production environment, enhancing the development and testing processes.

Data security and protecting sensitive information is a pivot for any organization for meeting compliance and regulatory requirements, Zuora Object Manager may be leveraged for downstream masking, encryption, and scrubbing, organizations can enhance data security, protect sensitive information, and ensure compliance with regulatory requirements. Zuora Data Migration tool kit provides more than one way for ingesting data from various sources into Zuora tenant whether the developer requires for a subset of data for testing or migration of data from external sources.


WHO?
QA ADMIN, IT ADMIN, RELEASE MANAGERS, DEVELOPERS
WHAT TOOLS?
ZUORA CENTERAL SANDBOX
WHO?
DEVELOPERS
WHAT TOOLS?
DEVELOPER SBX
WHO?
RELEASE MANAGERS PRODUCT MANAGERS IT ADMIN
WHAT TOOLS?
PRODUCTION
PLAN
CREATE &
TEST
UAT
RELEASE
MONITOR
ZUORA DEV AND CSBX REFRESHES
ZUORA OBJECT MANAGER FOR DATA MASKING
ZUORA DATA MIGRATION TOOLS
WHO?
PRODUCT MANAGER, DEVELOPERTEAM LEAD, PROJECT MANAGER
WHAT TOOLS?
PROJECT MANAGEMENT TOOLS
WHO?
IT ADMINS, DEVELOPER TEAM LEAD, PRODUCT MANAGER
WHAT TOOLS?
ZUORA SYSTEM HEALTH DASHBOARD

1. **Plan**: This stage is about gathering requirements, creating user stories and breaking them down into smaller tasks for the developers to initiate the development.

2. **Create and Test**: The build on the requirements, user stories is initiated. The developers verify the changes they have made before merging work with others.
   As the development expands and the operational business complexity increases, it becomes essential to optimize Application Lifecycle Management. The key area for optimizing the process end to end is strategic use of multiple developer sandboxes of different types.
   Zuora Developer Sandboxes which is a dedicated, isolated environment provided to create, modify and test changes by using a snapshot of production configuration data. It provides frequent refreshes and is an ideal set up for testing their changes and integrations.
3. **Testing**: All the created,modified, customized artifacts are consolidated, bundled together and moved to a higher environment for user acceptance testing. Zuora Central Sandbox is designed to provide an identical copy of the production environment for various development, testing and training purposes. Central Sandboxes are valuable and highly recommended for UAT, performance and load testing.
4. **Release**: Post successful testing and meeting quality benchmarks, deploy the updates to production. Releases in Zuora Tenants can typically fall into one of the two categories:
   1. Minor changes such as Bug fixes or modifying any of the existing billing documents, adding a new batch or communication profile etc.
   2. Major Changes with significant impact that may impact the user experience and data accuracy. For e.g, adding a new workflow for performing mass update action on price changes or data integration from external sources such as Netsuite, Salesforce, SFTP integrations etc.
5. **Monitor**: Zuora Platform enables monitoring your deployments and optimizing your UAT with Zuora System Health Dashboards. These dashboards provide visualizations to usage and performance of Zuora

products, which enable the users in understanding workload, performance and make data driven decisions. System Health dashboards can be configured with threshold based alerts (callouts/emails) to help you continuously monitor through your continuous devops processes.

Zuora continuously improves its software application for providing a better user experience and tools and applications for simplifying the complexities of business unit expansions. You can gain visibility into latest Zuora releases by visiting the provided link and access the impact.

The ALM framework within the Zuora Billing Platform offers guidance on best practices for our customers who utilize Zuora CPQ and Zuora Connect for Salesforce. These best practices will ensure effective utilization of these tools.