Deloitte.

DevOps in the US Federal Sector

Deloitte's US Federal practice has been helping its clients realize value sooner and achieve higher levels of efficiency by using DevOps practices. The approach is to lead with DevOps capability and performance assessments leading to small pilots, quick wins and culture change that scales up for transformation at an enterprise level.

Below is an example of a DevOps implementation by the Deloitte US Federal practice.

Case study: DevOps implementation at a large federal client

Business issue
The client IT department develops and maintains the IT systems that support 240+ official locations all over the world. Due to the criticality of changes being deployed, IT needed to reduce cycle time to deliver technology solutions and enhanced applications.

Root cause
The client turned to Deloitte for analysis and leading practices. The Deloitte team identified time constraints related to manual software builds with inconsistent/ad-hoc processes. Manual processes created re-work, wait-states, and longer deployment times. To address these issues, Deloitte saw an opportunity to introduce software build automation using Continuous Integration (CI) to reduce the overall lead time.
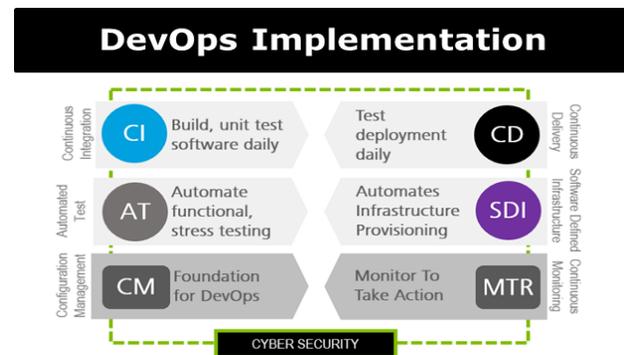
The approach
Continuous Integration drives process consistency, reduction of repetitive tasks, and an ability to catch issues and defects faster to drive rapid deployment of new functionality to the customer.
* **Migrated source code management** from Subversion into Team Foundation Server (TFS) for better organization and control of source code assets.
* **Execute CI Process upon code check in with a Controlled Code Promotion:** TFS was configured to incorporate the changes from a specific branch and to kick off unit test execution during the build. At build completion, the tool provided the pass/fail results and warnings/error log enabled fast debugging.

Results Achieved
Significant benefits were realized early in the project due to the approach.
* **Early defect detection**: Enabled faster defect resolution gaining higher code quality sooner
* **Predictable software builds**: Configuration management was automated and streamlined; software builds became predictable and auditable using the automation tools
* **Quicker response to implement critical changes**: Multiple release branches could be managed effectively with automation to reduce churn and errors to implement critical legislative mandates
* **Reduced human error**: TFS Build definitions and automation reduced human errors
* **Increased developer productivity**: Faster feedback-loops helped developers address changes and defects with greater efficiency compared to waiting for downstream test and evaluation cycles

Deloitte.