The E-Marketing Corporation: *Dev Ops Recommendation*

Hasan Zaidi, Brandon Yu, Troy Yoakum

Existing Environment - (TROY)

Development tasks in backlog kept on sticky notes, no capacity/sprint planning in place

Current development environment runs ad-hoc local versioning from individual workstations on-site – No source control in place

Package building done ad-hoc, by an individual – Requires replacement

Existing Production Environment Resources configured for BLUE/GREEN Deployment (explain verbally what that is ~ 30 sec), hosted on AWS IaaS – Requires Restructuring

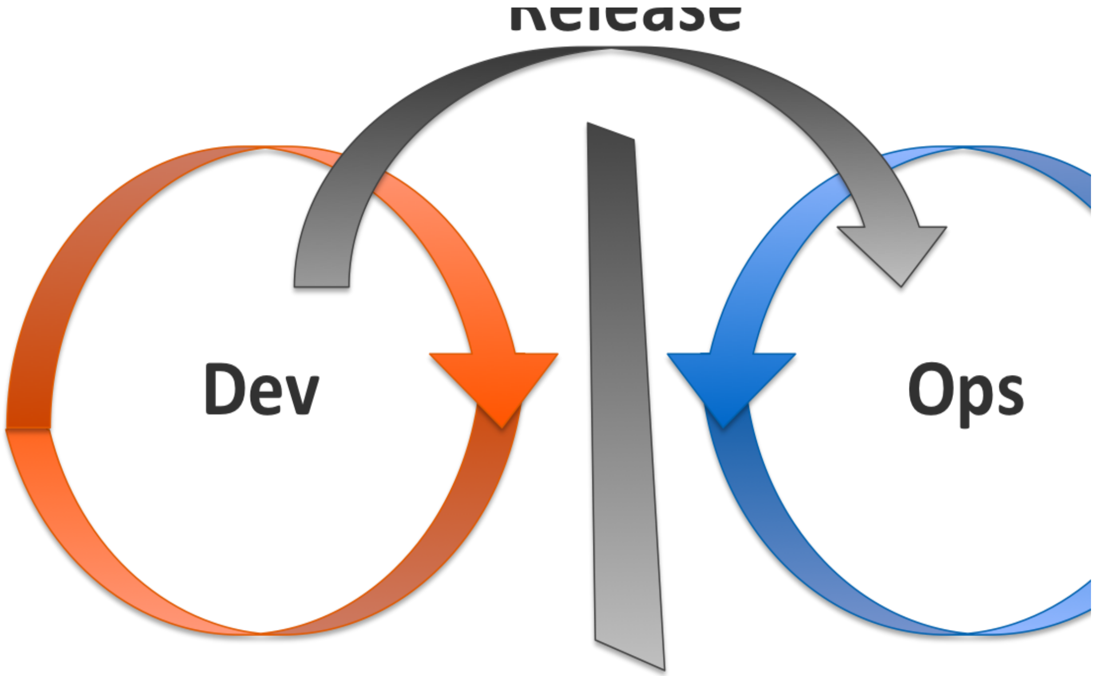
Proposed Solution - Plan/Code (BRANDON)

- Software developer backlog should be using Azure Boards for 'big picture' planning.
- Use Eclipse IDE to develop our coded features in the green environment before it's deployed and receives online traffic, giving us the opportunity to utilize iterative improvements in our programming.
- Git Branches are used to cement finalized code into our project and create avenues for new improvements to be made at any given time.
- Eclipse is connected to Git Repository for safe, timely storing of newly-pushed programs.

Proposed Solution - Build/Test (BRANDON)

- Eclipse IDE compiler is used to automate and test each iteration of our program in our unit testing phase. This uncovers possible issues which may be detrimental to our workflow process in the long run.
- Jenkins is used to build and 'push' our built application packages into the green testing environment for more realistic testing parameters.
- Thoroughly test the new version using Regression or Integrated Testing in the Green environment to ensure it is working as expected.

Proposed Solution – Release / Deploy (HASAN)



Route

Once validated, seamlessly route production traffic from the Blue environment to the Green environment, making the new version live.

Roll back

If issues are encountered, quickly roll back by routing traffic back to the Blue environment.

Requirements

This method will allow them to roll back quickly, as outlined in their requirements.

Proposed Solution – Operate / Monitor (TROY)

- Business objective is for agile, responsive control of production and indev software.
- Ansible fulfills these requirements while maintaining excellent support for existing AWS resources, preferable to other CD & Orchestration solutions.
- Ansible offers Operate/Monitor Capability of deployed software instances.

Solution Implementation (HASAN)

Environment Before

- Sticky Notes in Product Owner's office
- No version control
- After Build a team member manually deploys

Proposed Solution

- Azure Boards
- Introducing Version control for centralized repository
- For CI/CD implementing Jenkins to automatically build, test, and package the application.
- Leverage the Blue/Green deployment strategy to enable quick rollbacks if needed