Content QA

The Problem: Production Content Quality Assurance

The Solution:
Agile Zone "Content QA"!!
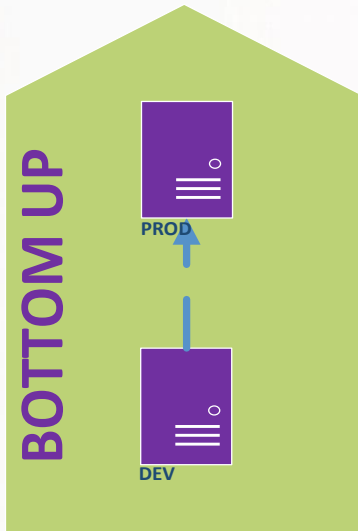

Figure 1: The "Bottom-Up Model

Not All Environments Are Created Equal

Since the beginning of time, the SDLC has followed a "bottom up" model. Deployment begins in the development environment, followed by quality assurance, performance, and eventually the production LIVE environment. While the bottom up model works well with conventional Application Server managed systems, it is not the right solution for Content Managed systems.

The basic RULE of CMS is *content development*, which is often misunderstood as *an application development activity*. When working with a CMS, our goal should be to move past development models to a system that caters to content creation in a function-oriented environment. The Agile Zone "Content QA" Adobe Experience Manager Add-On is just such a system.

A Functional Approach to CMS

In the CMS environment, we can identify three distinct – though sometimes overlapping – functions for content development. The first function is the **content creator**, who we refer to as the Author. Once content has been created, it must be **approved** by the Content QA function. From there, the content passes to a Publisher, whose function is to **publish** the content after the approval process is complete. These functions require proper interfaces and well-defined processes to assure that the CMS system works as intended.

In an organization with a large content footprint or sensitive content sites, correctly implementing these CMS functions is critical. Proper implementation supports the organization's security and business requirements, and eliminates the inadvertent content release to LIVE servers.
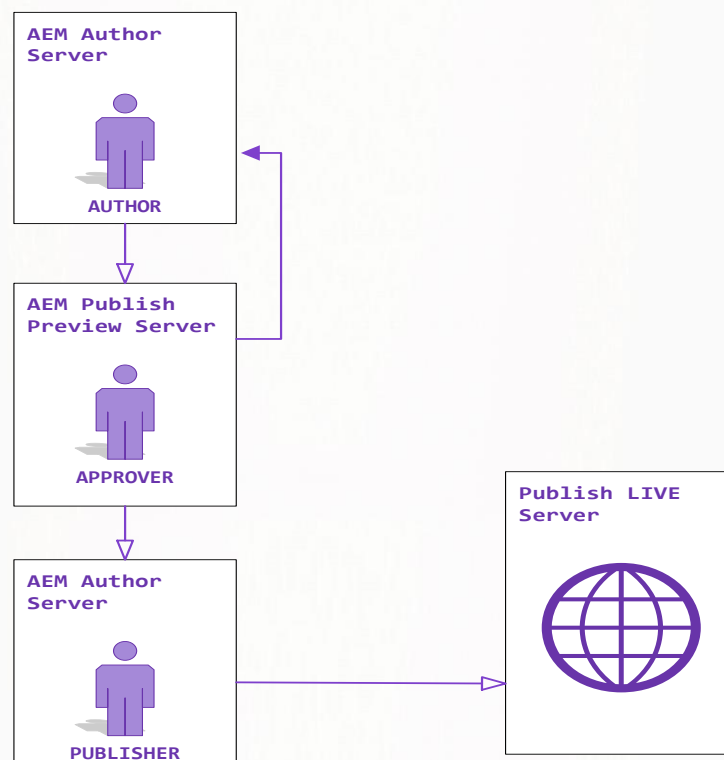

<div align="center">Figure 2: The CMS Function Environment</div>

The Adobe Experience Manager

The Agile Zone "Content QA" Adobe Experience Manager (AEM) Add-On is specifically designed for use with AEM implementations. Why does the typical AEM implementation need Agile Zone's "Content QA"? In the simplest terms, it comes down to providing effective means to achieve **separation of duties**, assuring that the three CMS functions work as intended. Let's start with a look at the AEM in its "out-of-the-box" (OOTB) state.

Anyone who has worked with AEM knows that it is a remarkably flexible and versatile CMS, even straight OOTB. But like any complex system, it has gaps — functional shortcomings that require modifications or workarounds. In my experience, these solutions range from the "hope-for-the-best" quick fix to more complex approaches designed to mimic the bottom-up model.

For example, we've had clients that rely on the Author to perform the Content QA function, simply hoping for the best. We've seen clients create workflows, adding an approval step before content is published. A few have even added a separate AEM stack dedicated solely to the Content QA function, creating a separate space where content approval happens. Then the content is migrated to LIVE production servers. We've seen all of these solutions, and nearly everything in between, too. It's what led us to create the Agile Zone "Content QA."

Each of these approaches create problems of their own, but absent any better options, organizations have had no choice but to improvise and work within the AEM boundaries. If yours is an organization that has implemented AEM OOTB, with or without workarounds, here are some of the issues you may have encountered:

- No good separation of functional duties, so no proper vetting.
- Approvers must work with a clunky and technical Author preview interface.
- Using WCM mode to disable the Author preview interface for a given URL means losing navigation aspects, since as soon as the approver navigates to a different URL, she ends up with the clunky Author preview interface again.
- The Author interface is not true site code and content, meaning that the content seen by an end user on the LIVE site may look and behave differently.
- Personalization is not available in Author server preview.
- Third-party integration is not available in Author server preview.
- Workarounds like a bottom-up approach or introducing a dedicated production preview stack can entail huge hardware and support costs.
- A separate preview hardware stack complicates the process because of the reference changes between the preview Author server and the LIVE author server.
- Moving tens or hundreds of gigabytes of content between servers can be a time-consuming and risky proposition.
- Security may be an issue if approvers are situated outside your intranet (for example, marketing people), leaving the most important and vulnerable layer of your Author server susceptible to intrusion.

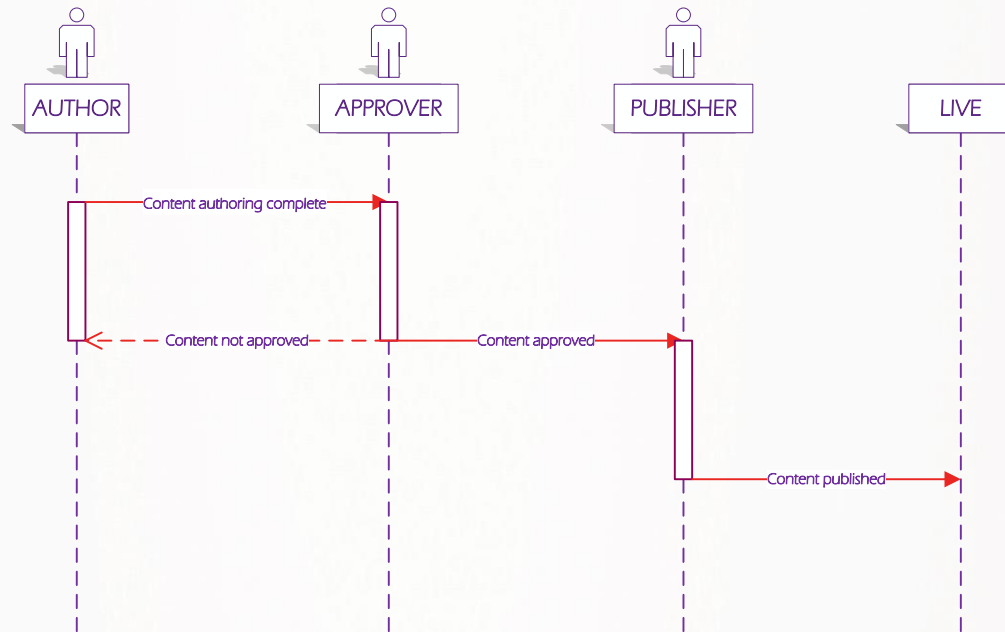**The Solution: Agile Zone's "Content QA" AEM Add-On**

Figure 3: "Content QA" Workflow Process

Agile Zone's "Content QA" is designed to solve all of these problems, and more. Created with all three of the functions integral and essential to production content creation in mind, "Content QA" provides secure and simple interfaces for authors, approvers, and publishers. By establishing a reliable system to separate production content roles, "Content QA" offers a robust and comprehensive solution for production content quality assurance.

"Content QA" converts one of the existing (or new) production AEM publish servers into a **preview server**. Once an author has changed content, "Content QA" provides a lightweight, easy-to-use approval interface for quick and easy approval or disapproval of those changes. If proposed changes are rejected, Content QA notifies the author via email that a proposed change is disapproved and must be re-done. If changes are approved, those changes can go live once the approval process is complete.

At the Author level, "Content QA" augments AEM's OOTB Author server with an easy-to-use Author interface that allows new content to be sent for approval to the Publish server.
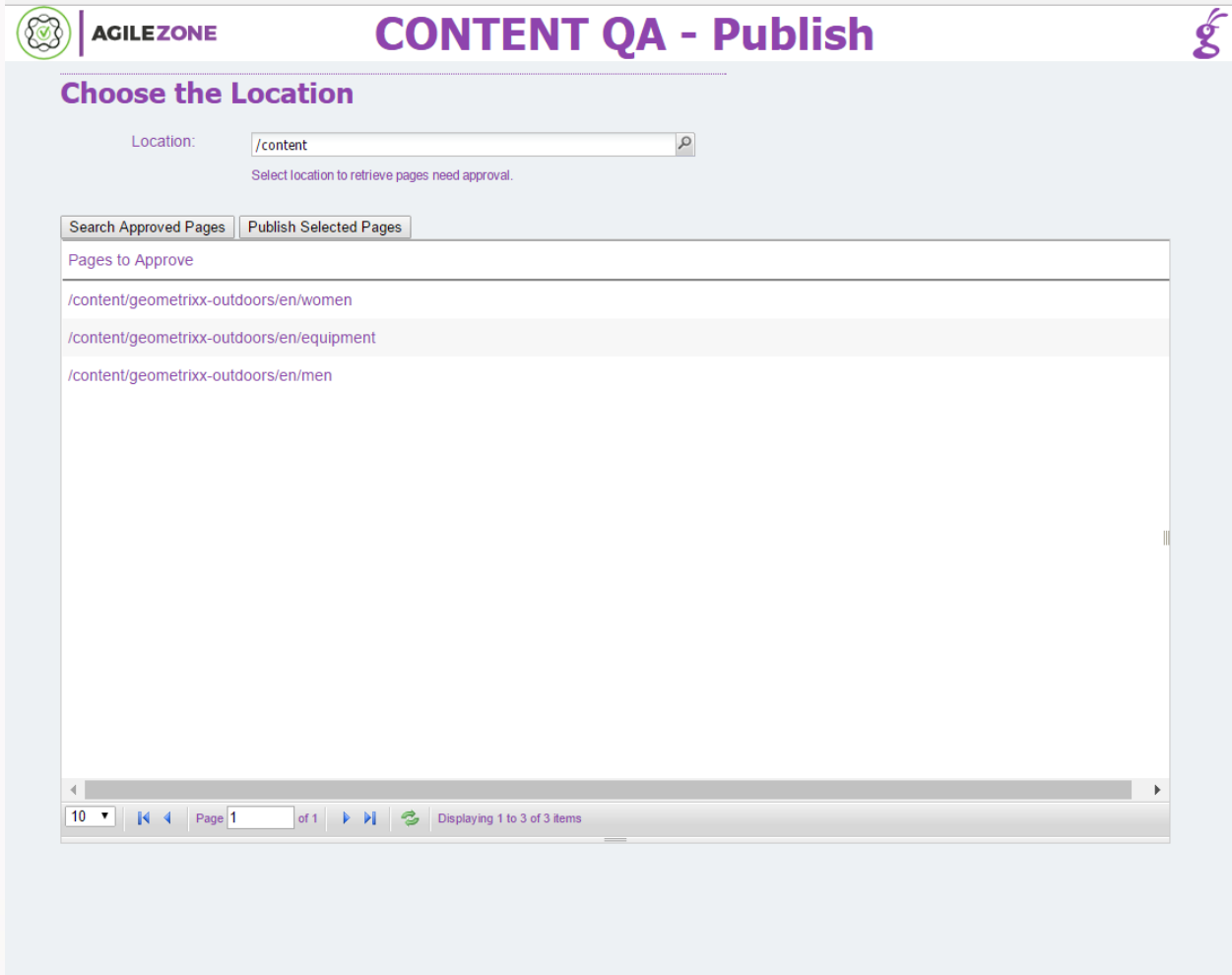
Figure 4: An Agile Zone "Content QA" Publish User Interface

"Content QA" also adds an approver interface on the AEM Publish server to streamline the approval of new production content once the quality assurance process is complete. This interface is the same as the end-user or consumer interface, so the approver knows exactly what she is approving and how it will look and behave when it goes LIVE and is available to the end user.

Finally, "Content QA" addresses the third group of users who work within the CMS environment – the Publisher. By adding a new and user-friendly **publish interface** on the Author server, "Content QA" allows newly-approved content to be published in bulk or on a page-by-page basis.

"Content QA" Advantages

"Content QA" offers you important and valuable functionality that isn't available in AEM OOTB, or in any other tool on the market. And it does all this without impacting AEM stack performance or otherwise interfering with AEM's code or contents. Here are just some of the advantages you can expect from implementing Agile Zone's "Content QA" solution:

- "Content QA" is a lightweight, non-intrusive AEM add-on with a small footprint, but it offers powerful functionality. It is written in Java, JSP and JQuery.
- "Content QA" uses OOTB AEM libraries and does not require the installation of third-party libraries.
- "Content QA" does not modify the AEM core code-base, meaning that it will not interfere with future AEM upgrades.
- "Content QA" does not require new hardware for its production preview environment. It works with the existing production publish server without requiring significant server changes.
- "Content QA" does not add additional processing load on the AEM author server.
- "Content QA" provides a true end-user LIVE environment for content QA testing: the same codebase, third-party integration, vanity URLs and personalization.
- "Content QA" provides the solution without a need to expose the Author Server to public internet.
- "Content QA" delivers the CMS separation of duties you expect, distinguishing between Author, Approver, and Publisher functions.
- "Content QA" is also flexible enough to accommodate the organization that combines the approver and publisher roles, or that sometimes combines all three roles into one.
- "Content QA" is simple to configure and can be deployed in a few easy steps. It can be installed, configured, and made operational in less than one day.

Reference URLs:
Agile Zone – http://www.agilezone.com
Content QA URL – http://ww.agilezone.com/contentqa
Adobe Marketplace -
https://marketing.adobe.com/resources/content/resources/en/exchange/marketplace/apps/content-qa.html