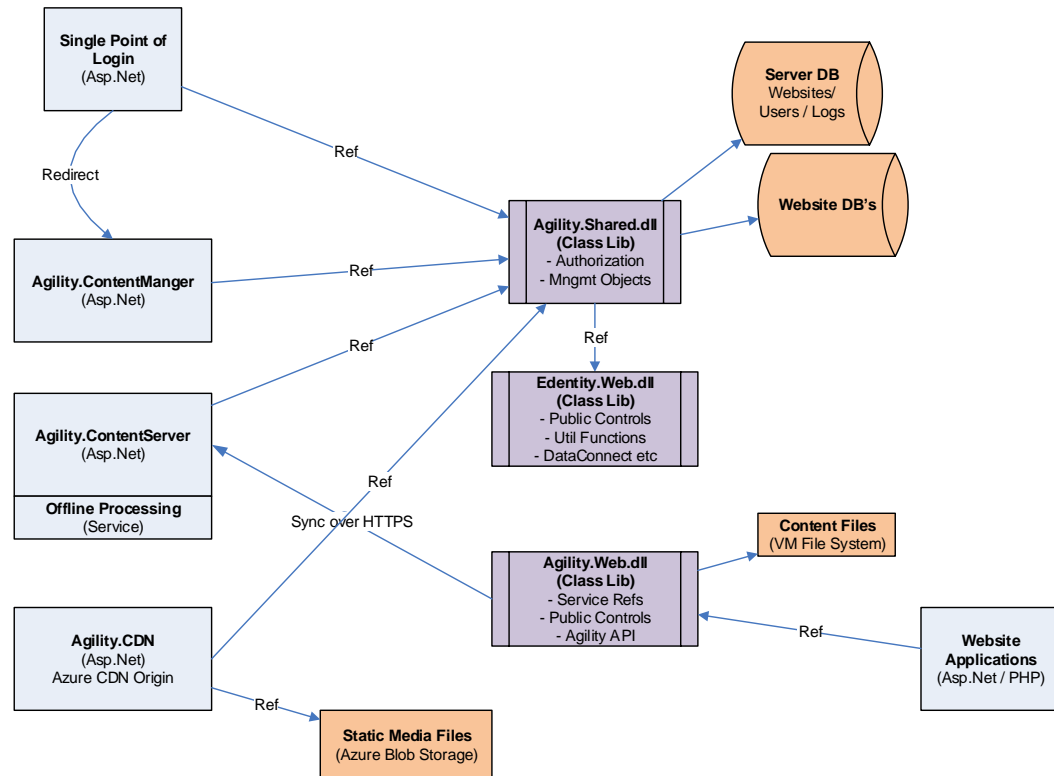# Agility CMS- Systems Overview

Agility Application Architecture - Overview
Azure Hosted Systems

Architecture Notes

The Agility CMS system consists of several multi-tenanted components that deliver website data to Web Servers via HTTP or HTTPS. These components are hosted on Microsoft Azure on Web Roles (PaaS). See the above diagram for a simple overview of the applications and critical reference points for data access.

These components are:

- Content Manager (Asp.net, web application used for customer access)
- Content Server (Asp.net, web application used as an API endpoint)
- Offline Processing (windows service used to perform offline processing)
- CDN –ASP.Net Website used as the origin endpoint for Azure CDN

Website Data is housed in Azure SQL Databases, 1 db per customer. All connection info and user login information is stored in a central Server DB in Azure SQL Server.   All user passwords are encrypted.

All database access is performed via the Agility.Shared.dll Assembly, using utility classes contained in Edentity.Web.dll, both reviewed by Joel Varty, Director of R&D.

All login and password access in the application is done via SSL, however the system itself is primarily accessed via HTTP, so that non-SSL resources referenced by the website can be used with the content manager easily. An encrypted Session Cookie Hash is used to store the authorized user id.

All systems are load balanced for redundancy and performance, with automatic scaling enabled with the Azure Cloud Service deployment.

Testing

Agility performs the following tests on our systems before releasing code:

Development Environment

- **Developer Peer testing**
- These tests are done within 24-48 hours of code being completed and allow a developer to self - test or peer-test their code.
- **"Bug Bash" events**
- All hands on deck – Agility employees are invited to spend dedicated time testing Agility from a customer's perspective.

Staging Environment

- **Feature Testing**
- Each feature is tested by an assigned QA staff.
- **User Acceptance Testing**
- Internal and/or partner and/or customer stakeholders can test individual features or platform components.
- **Load Testing**
- When applicable the system is load tested. We typicall use a cloud service called "LoadImpact.com" for this.
- **Regression Testing**
- A set of regression test cases is followed to all components impacted by a particular feature set that is to be released.
- **Live Deployment Testing**
- On the day of a deployment, all configuration or platform variable settings and files are reviewed by Joel Varty and QA or designated developer(s).

Live Deployment Testing

- **Critical Path**
- After deployment on a Live environment, a predefined set of "critical paths" are tested to determine with a roll-back may need to happen.
- **Smoke Testing**
- The system is tested with a common set of actions to determine if the deployment was successful or needs to be rolled back or fixed.
- **Feature Testing**
- Each feature that was newly deployed is tested by QA staff.

Release Management

All code deployed in Agility's system is tested in-house. We have a team of 3 individuals who are tasked with deploying code to our production environment.

All code must be tested in a mirror environment in a separate Azure subscription before being deployed to the live environment. Compiled code is stored in a versioned folder in Azure Blob storage and deployed by an automated system to each Instance in the deployment's Web Role. This system has been in place since 2011 in Azure, and since 2006 before that when the system was hosted in Q9 networks.

Director of Research & Development Joel Varty has been the individual in tasked with each code deployment, and with review of all code in the system.

Since our new version has been in development, code to the current live system has been frozen outside of critical fixes. Since the system is load balanced, critical issues can be fixed in real time in an emergency, however normal code releases are scheduled for off-hours, and customers are notified in advance.

Code Management

All code is housed currently in TFS in-house at Agility, or in VisialStudio Online, hosted in Azure.

Change and conflict management is handled by Joel Varty, Director of R&D, as is branch management. Tasks, Bugs and Features are also housed in TFS and have change-sets directly associated to them to provide context for why a certain change was made.

Future Plans

As the Agility system grows and evolves, we plan to outsource at least some of our Quality Assurance workload to a 3rd party company, however that has not been finalized yet, as we are still in negotiations.