* HTML5 Pilot Project
  + Microservices:
    - Asset Summary REST Service
      * Retrieves assets/datachannels from Mongo
    - DOL REST Service
      * Public interface(API?) to access protected REST services
    - User Auth Service
      * Custom token-based authentication using Mongo
    - User Data Service
      * Retrieves data pertaining to user
  + Libraries:
    - Authentication Attribute
      * C# Attribute for requiring authentication on MVC class
    - Basic Authentication
      * Authenticates user by comparing password in header
    - REST Client Library
      * Interface for making synchronous REST calls
  + Other:
    - Unit/Integration tests for every service and library
    - Continuous integration environment
    - Consistent logging at multiple levels
    - Custom exceptions and unified handling
* Packet Counter Service
  + Service to reduce load on database by offloading packet count operation from database
  + Reads message queue into hash table in memory
  + Periodically counts packets and sends count to database
  + About 30x speedup on packet count UI page
  + Removes need to perform over 2+ million operations on DB per minute
  + Moves business logic out of database
  + SOLID design principles - interface, factory, templating, DI
  + Unit testing
  + Logging
* Google Maps
  + Cleanup of core maps code
    - Reduced pollution of global namespace
    - Moved business logic into duck-typed interface
    - Organized code through prototypes and local data structures
  + Addition of Baidu Maps
    - Implemented interface with Baidu Maps API
    - Allow Chinese customers unable to use Google access to our mapping system
  + Maps Overhaul
    - Cleaned up Maps
      * Reduced number of user states
      * Simplified presentation of data
      * Simplified navigation
      * Enforced consistency