(NOTE Feb 9, 2016 - the following is a sample of my writing from 2012 when I was a Product Manager for a Fortune 1000 Company. It contains no sensitive or proprietary information. In this copy of the document I have replaced the name of the Company with "Company" and changed a few other terms to generic equivalents. I prepared the original document at the request of my Company to fulfil the requirements for a Statement on Standards for Attestation Engagements (SSAE) audit. In it I make a statement as to the purpose, design, development and testing of the software component for which I was responsible.)

Common Banking Services (CBS)

SSAE16 Audit Narrative
June 1, 2012
Craig Heath – Product Manager

Overview

Common Banking Services (CBS) provides re-usable banking services that can be accessed by the various clients within and outside Company in a standard, safe and secure manner. While CBS has been accessed directly by client applications for some time, all access to CBS services will soon be accomplished by an application first calling the Financial Services Gateway (FSG). See below for more details of this.

As a Domain Service, CBS is responsible for connecting end user-facing applications (such as SDP) to data stores maintained by Company (Informix DB) and especially to DPV host systems maintained within each of our client Financial Institutions. The latter is accomplished through interfaces maintained by Company that CBS uses to connect to host systems and return user and account data for display and use in the end user product.

CBS also provides business logic in which configuration settings are used to manipulate and enhance the data retrieved in ways that serve product functionality goals. While Company maintained data retrieval is a key part of the service, the prime function CBS serves is access to host data, with the value-add of business logic application.

CBS is part of the Services Oriented Architecture (SOA) infrastructure of the Company product line, and provides its services in a standard manner that allows any and all customer-facing products to create a consistent experience across the entire suite of offerings.

The business services currently provided by CBS are:

- Get Financial Institution
- Authenticate User
- Create/Update Challenge Questions
- Validate Challenge Questions
- Get Principal End User Details
- Update IB Startup Page

- Update User Last Mobile Login
- Update User Last Activity
- Preferences Service
- Invalidate User Cache
- Get Account List
- Get Transactions List
- Retrieve Check Images
- Deposit Slip Image Functionality
- Funds Transfer - Implementation Overview
- Scheduled Transfer add
- Edit Scheduled Transfer
- Scheduled Transfer Delete
- Get AchIds
- FinancialInfo Service
- Reset User
- getFiProduct
- Validate Host Pin
- get AccountXRef

These services are exposed by CBS using Durable API's that provide a consistent and robust method of accessing and utilizing the services using RESTful web services. Client applications can access CBS directly in this manner, but the Durable API's are also used as the access points for calls coming to the Financial Services Gatweway.

NOTE: At present, most client applications consuming CBS services do so through the Financial Services Gateway (FSG), a customized Enterprise Service Bus (ESB) implementation which exposes standardized, Durable API's to client applications which are used to route calls to the underlying domain service(s) required for the functional purpose. In the coming months those services and products who now contact CBS directly will do so through the gateway, in fulfillment of the overall architectural plan for the Company Data Center to which all Company infrastructure is now moving.

The FSG provides access to other underlying domain services in addition to CBS and provides value-add capabilities such as security, logging, transformation and orchestration capabilities that enhance the capabilities of the domain services themselves. While a critical part of the overall infrastructure, the FSG is not the subject of this overview.

Requirements Management

The requirements for new services or enhancements to existing services largely come from the various client development teams whose products consume and use the services to develop feature/functionality in their offerings. While the CBS team maintains a backlog of technical work that relates to the long-term improvement, maintenance and enhancement of the services overall, the needs of product

offerings are communicated to the CBS team through a system designed for this purpose.

The system is Quckbase-managed and is referred to as the Shared Services Request Quickbase, with each request document referred to as an "SSR". Team members from the client teams enter requests in which they outline their requirements, attach or link to relevant documentation, and identify their planned release dates and other information relevant to the work being asked of the CBS Team. CBS team members then review the request, add notes and documentation that outline the proposed solution, and provide high-level estimates for the work.

As SSR's move on to become backlog stories in the CBS Rally system, the SSR's are updated to reflect the status of the work, the CBS release in which they will be included, and becomes the document-of-record for the lifespan of the work, until the request is fulfilled and the SSR is closed.

The Rally story that reflects the work in an SSR also provides additional documentation of the solution, the purpose, and the task work involved in realizing the goal of the requesting team. While Rally is used to communicate and manage work in hand for the CBS team at any given time, the SSR system is used to track and communicate the status of work between the CBS Product Manager and the client teams, as well as serve as a queue system for work that is not yet part of a release.

Agile Development Processes

SCRUM, an agile software development framework, is the core model for CBS development. Efforts are structured in cycles called sprints, iterations of work that are typically three weeks in duration. During each sprint, a sprint backlog is built from a prioritized list of business requirements in the SSR system so that the features that are developed first are of the highest value to the client teams. At the end of each sprint, a potentially shippable product is delivered. Rally is the agile SCRUM process automation tool used to manage the story backlog and progress over the course of each sprint.

All releases are accompanied with a release note, which is a terse summary of recent changes, enhancements and bug fixes in a particular software release. Release notes are documents which are distributed along with software products. The Release notes for CBS are delivered Company-wide via email.

Deployment

CBS is currently being deployed in three Company data centers: Data Center 1 (DC1), Data Center 2 (DC2) and Data Center 3 (DC3). Within the next fiscal year, the DC1 data center will be shut down and all future CBS deployments will be to DC2 and DC3 only. The current arrangement is to manage a phased migration of all products and services into DC2/DC3 before the DC1 center can be closed.

Environments into which CBS is deployed follow a similar pattern with some differences between the DC's:

- **DC1**:
  - Development
  - Development QA
  - Formal QA
  - Performance
  - Beta
  - Pre-Production
  - Production
- **DC2/DC3**
  - Development
  - Formal QA
  - Performance
  - Beta
  - Staging
  - Production

CBS is currently deployed in the DC1 data center using legacy methods that have been in place in that data center for many years, but CBS is currently being deployed to DC2/DC3 using the Dev-to-Deploy (D2D) methodology that is required for any deployments to those centers. Beginning with the August release, CBS will also be implementing D2D deployments in DC1.

Quality Assurance

Software development and control processes include the Quality Assurance process, which is a planned and systematic approach to the evaluation of the quality of and adherence to software product standards, processes, and procedures. Software testing verifies that the software meets its requirements.

The quality of testing is assured by verifying that project and business requirements are satisfied and that the testing process is in accordance with the test plans and established procedures. A test plan is a document that describes the objectives, scope, approach, and focus of a software testing effort. A test case is a document that describes an input, action, or event, and an expected and actual response, to determine if a feature of an application is working correctly.

QA is involved in the project from the beginning. This helps the teams communicate and understand the problems and concerns, and also provides time to set up the testing environment and configuration. Actual testing starts as features become available in the case of Agile projects. Test plans are written, reviewed and approved based on the design documentation and test results are documented and retained. Quality Assurance must approve test results prior to implementing the change into production.

CBS benefits from having a large set of test cases maintained in a centralized library that can be used by any member of the QA team. These tests can be run manually, but CBS also maintains automation scripts that allow automated regression testing to be run at any time. This allows hundreds of scenarios to be run without manual intervention, both speeding the process and allowing for more frequent test runs to ensure quality at various points in the life cycle of a release.

Defect Management and Early Defect Removal

Once a defect has been discovered via inspection, testing or operations, the defect is recorded in the defect tracking tool. When a defect is validated, it enters the change management process and is subject to the controls previously described.

CBS uses a standard defect tracking tool (Mercury Quality Center) for the tracking and management of defects discovered during more formal operations, such as QA, Beta testing, etc. But the CBS QA engineers also maintain a separate, Quickbase-driven tool for managing issues and defects that arise in the Development and Dev/QA environments, which are not usually tracked in a formal way. This makes it possible for defects to be discovered at earlier phases in the release cycle and mitigated as part of the development work itself, instead of being handled at later stages in the work. This helps keep the actual "bug count" low as they are handled before they become actual defects.