Integration Architectural Overview

Subtitle Here

Table of Contents

[1 Introduction 4](#_Toc424652061)

[1.1 Purpose 4](#_Toc424652062)

[1.2 Scope 4](#_Toc424652063)

[1.3 Definitions, Acronyms and Abbreviations 4](#_Toc424652064)

[1.4 Functional Overview 4](#_Toc424652065)

[1.4.1 Solution 5](#_Toc424652066)

[1.4.2 Key Features 5](#_Toc424652067)

[2 Architecture 6](#_Toc424652068)

[2.1 Master Data 6](#_Toc424652069)

[2.2 Scalability 6](#_Toc424652070)

[2.3 Data Retention and Backups 7](#_Toc424652071)

[3 Implementation Considerations 8](#_Toc424652072)

[3.1 Master Data 8](#_Toc424652073)

[3.1.1 File Types 8](#_Toc424652074)

[3.1.2 Document Framework 8](#_Toc424652075)

[3.2 Distributor On-boarding 8](#_Toc424652076)

[3.2.1 File Types 8](#_Toc424652077)

[3.2.2 Document Framework 9](#_Toc424652078)

[3.3 Client Configuration Settings 9](#_Toc424652079)

[3.3.1 Company Resolutions 9](#_Toc424652080)

[3.3.2 Cross References and Fuzzy Matching Resolutions 9](#_Toc424652081)

[3.3.3 Product Resolution 9](#_Toc424652082)

[3.4 Endpoint Files 10](#_Toc424652083)

[4 Web Based Application 11](#_Toc424652084)

[4.1 Underlying Technology 11](#_Toc424652085)

[4.2 Application Security 12](#_Toc424652086)

[4.3 Application User Roles 12](#_Toc424652087)

[4.4 Configuration Settings Management 13](#_Toc424652088)

[4.5 Exception Resolution 13](#_Toc424652089)

# Introduction

This document provides a high level overview architecture of the Data Acquisition Process, which includes a process engine component and a user interface for setting client specific configurations, exception resolution and reporting.

## Purpose

The purpose of this document is to provide an architectural overview of the framework by which appropriate business functions can be implemented, supported, and securely accessed by users of Data Acquisition Application.

## Scope

This document discusses the architecture of the Data Acquisition process engine and user application interface mainly from a data integration implementations stand point, but will also describe the architecture from several different perspectives to allow multiple stakeholders to get a high level overview.

## Definitions, Acronyms and Abbreviations

* DA – Data Acquisition System
* AFS – Advanced Food Systems
* ASP.NET – Microsoft web application platform
* HTTP – Hypertext Transfer Protocol
* MS-SQL – relational database management system (RDMS)
* WWW – World Wide Web
* UML – Unified Modeling Language
* PE – Processing Engine
* Client – Member entity of this multi-tenant system
* MS-SSIS – Microsoft’s Extract, Transform and Load utility
* DF – Document Framework
* ETL – Extract, Transform, Load. Refers to a process to migrate data from one database to another.

## Functional Overview

Distributors and Operators (often referred to as buyers & sellers) record and identify products and customers differently by using their own internal identifications and naming conventions. This data is then communicated to the Manufacturer providing claim and velocity information in an infinite number of layouts across dozens of different formats in both paper and electronic form. The prerequisite to the settlement and reporting of a manufacturer’s trade spend is the standardization and mapping of this data.

Data Acquisition is a robust master data management service that will capture, cleanse, map, and standardize a client’s inbound claim and velocity data for use in trade management, reporting, and analysis solutions. Using matching algorithms developed over 25 years of managing foodservice data, Data Acquisition maps and standardizes the buyer and seller entities contained to your organization’s master data.

### Solution

* Capture and standardize the hundreds of incoming data formats provided in Distributor and Operator Claim and Velocity files.
* Resolve relationships between key business entities such as Distributors, Operators and Products, and the client’s master data.
* Control and Monitoring of the service performance.
* Provide intuitive exceptions workflows and rich reporting capabilities.

### Key Features

* **Awareness**: AFS Data Acquisition knows when to expect your trading partner’s inbound files, understands what they should contain, and which formats are acceptable for consumption by our services. Inbound files are monitored throughout each phase in the mapping process, recording audit data, and producing alerts based on configurable tolerances.
* **Duplication**: AFS Data Acquisition includes routines to identify and mitigate duplicate files and data rows.
* **Flexibility**: Can be leveraged to use the AFS Master Data Warehouse or configured as a standalone service using a client’s master data. A wide array of customizable formats can be registered as consumable file endpoints, and virtually all elements of the matching process can be adjusted.
* **Resolution**: Using rich and independently-customizable algorithms, AFS Data Acquisition can match company and product information, minimizing the need for manual data mapping.
* **Scalability**: Data Acquisition was designed to be easily scaled to meet any customer growth needs.
* **Portability**: Data Acquisition can be leveraged by our TPM solutions or implemented as a stand-alone service to be used by your in-house or third-party systems.
* **Reporting**: Analyse the health of your trade programs and trading partner relationships from key points of view with built in reports.

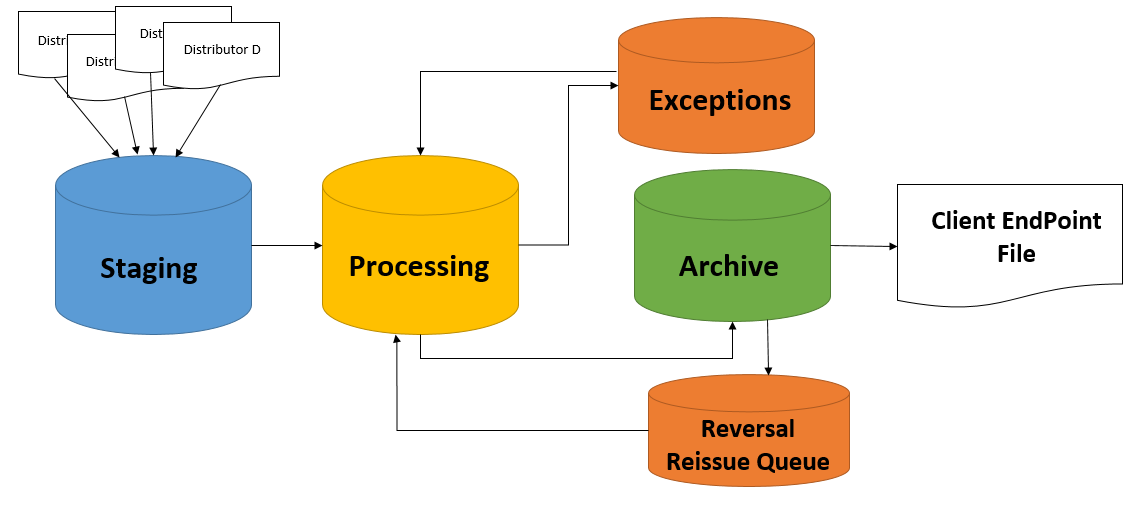


Figure 1

# Architecture

## Master Data

Data Acquisition utilizes a centralized repository for master company and product data. Company data can be time. More on master data will be covered in the implementation section of this document.

## Scalability

The Data Acquisition system utilizes a centralized repository for master data, while the processing systems can be housed on different servers. This allows for client data to be processed quickly without being impacted by the volume being processed by other servers. The User Application is also housed on a single application server. The multi-tenant architecture allows for a single unified codebase, while the multiple processing servers keep the impact of a multi-tenant environment related to performance to a minimum. See Figure 2.

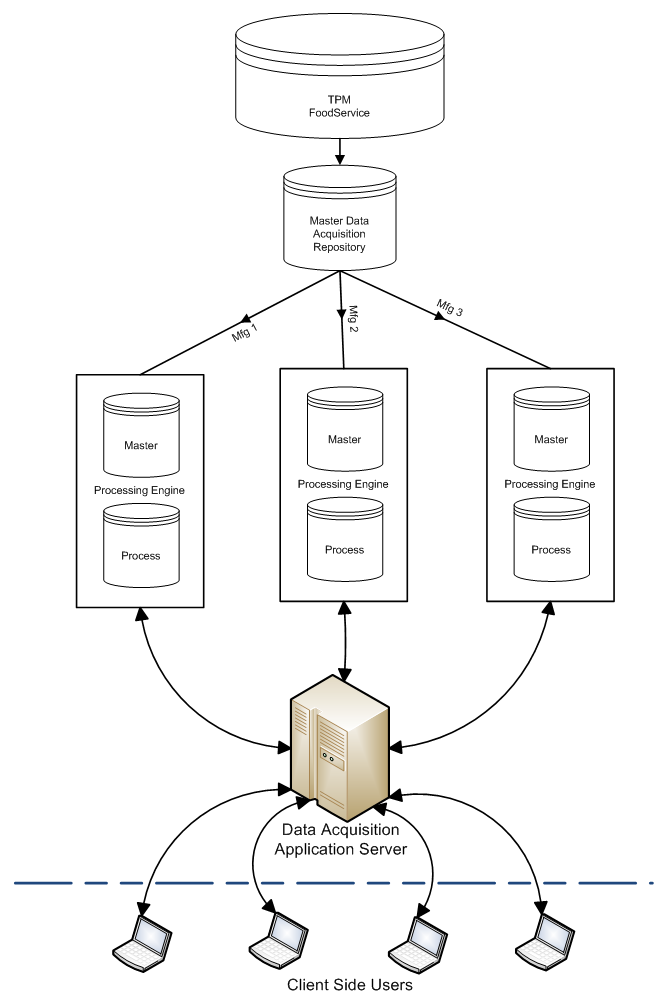


Figure 2

## Data Retention and Backups

Our data retention policy for resolved archived data is three years plus current, within the normalized relational database model. Generated endpoint files are kept on local data stores in a zipped format for the same timeframe.

Backups are performed prior to each processing run and are stored at an AFS hosted server.

# Implementation Considerations

## Master Data

The Data Acquisition processing engine requires Master Product and Company (Customer) data from each client account. Master data can come organically from the Food Service TMP system or can be received into Data Acquisition via a custom import process.

### File Types

AFS has a preferred file layout for both Company and Product master data inbound files. These layouts are considered ITB, but custom solutions can be created to support specific business requirements.

### Document Framework

All inbound files to AFS are registered with the Document Framework system. This system watches for file arrival, logs properties of the file and stages it to be consumed by an ETL. After consumption, the Document Framework directs the ETL as to the location for archival of the original inbound files, and additional properties of the file (e.g. arrival date, process date, row counts) can be stored in the Document Framework for reporting purposes.

## Distributor On-boarding

Data Acquisition supports two types of file processing. Velocity data or distributor sales into the operator (Type = SALES) and distributor/operator bill-back (Type = CLAIM). Sales data is a dump of all transactions between the distributor (Seller) and the distributor’s customers (Buyer) of a given Data Acquisition client’s (Manufacturer) goods (Products). Claim feeds are always a subset of the larger velocity feed, but the purpose is to bill back against contracts between the Manufacturer and Operator for guaranteed pricing. This discounting is factored into the pricing the buyer receives from the seller and therefore the seller is billing back to the manufacturer for this loss (Rebate).

### File Types

Each distributor has their own proprietary layouts for these types of feeds. These can be custom text feeds, Excel, or EDI. AFS has established relationships with most food service distributors and has created custom ETL for receiving data via these feeds.

### Document Framework

Claim and Sales feeds are also registered with the Document Framework (described above). Data Acquisition supports a foreign key from the Document Framework so that gap reporting can be provided to identify missing and untimely files.

## Client Configuration Settings

Client setting are used to configure how the Data Acquisition process engine resolves each client’s Sales and Claim data, and includes which elements of that data are resolved, how to handle unresolved data, etc.

### Company Resolutions

Data Acquisition supports the resolution of 5 different company roles as it pertains to Sales and Claim Data. The five company roles are Claimant, Seller, Buyer, Buyer HQ, and Billed to Buyer. Claimant is a required resolution but all other roles are optional. If the resolution is performed, the client can also configure whether or not a successful resolution to master data is required before sending to the client in the standardized endpoint feed.

### Cross References and Fuzzy Matching Resolutions

Data Acquisition utilizes client-specific cross-references for matching companies to the client’s master data. When no cross-references exist, Data Acquisition also employs a custom proprietary algorithm for auto-matching called Fuzzy Matching. The distributor’s raw data is compared to the client’s master company records and a probable match score is captured for each unique instance of raw company data. The client can customize how tight or loose this auto-matching algorithm is evaluated during the resolution phase of the process by setting minimum thresholds for the overall probable match score, or of the atomic pieces (name, address, city, state, zip), giving the client a wide range of options to fine-tune this process to minimize unnecessary unmatched exceptions. Cross references always take precedence over Fuzzy matching.

### Product Resolution

Products are matched via a direct lookup of either the reported GTIN, Manufacturer SKU, or UPC. If the lookups do not result in a match, cross-references can be created against the reported product number in a precedence order. If GTIN matching is on, this is the gold standard, followed by seller product SKU cross-references, manufacture SKU cross-references, and UPC cross-references.

## Endpoint Files

Once data has been resolved according to the client’s own configurations, any data that is eligible for transmission is exported to a standardized set of text files that will be transmitted to the client via FTP/SFTP for consumption by the client or client’s third-party provider. The endpoint file has all of the client’s business keys for customer and products, making consumptions for claim verification and reporting very easy. AFS can publish endpoint files to an AFS FTP server or push to the client’s own FTP server(s).

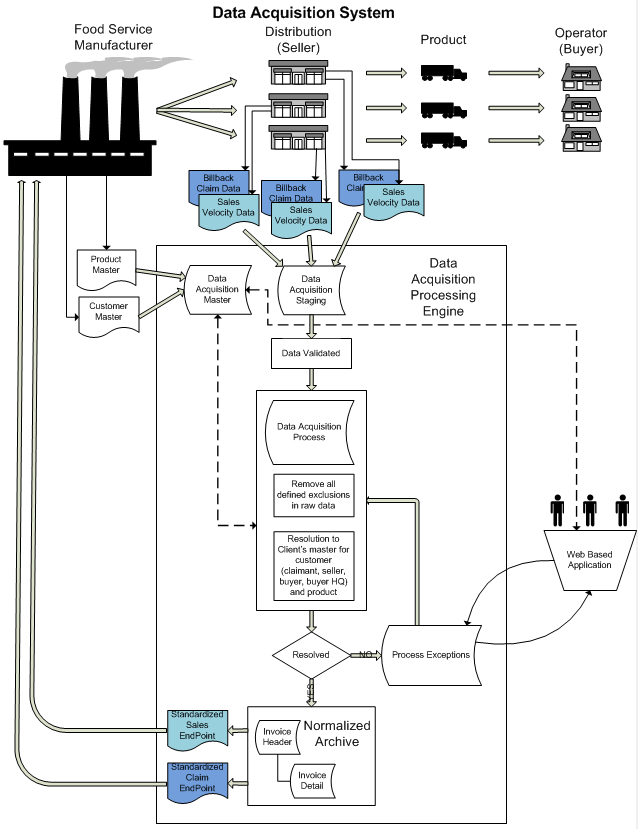


Figure 3

# Web Based Application

## Underlying Technology

The Data Acquisition Application is a Multitenant Software as a Service (SaaS) web application running on the Microsoft Windows technology stack, and was built using the C# MVC 4.0 programming language in a responsive HTML5 framework. The application can be accessed over a secure internet connection (SSL) and supports Microsoft Internet Explorer 9 or later, Mozilla Firefox, Apple Safari, or Google Chrome, on both Windows and Apple devices.

* ASP.NET MVC 4.0, HTML5 & JavaScript
* Lightweight and fluid, empowering users across all platforms.
* Using latest Microsoft technology stack.
* Highly flexible architecture providing deep data mining and management capabilities.
* Highly scalable architecture supporting both scaling-up and out.
* Service Oriented Architecture
* Recommended Browsers: IE 11, Firefox, Safari or Chrome.

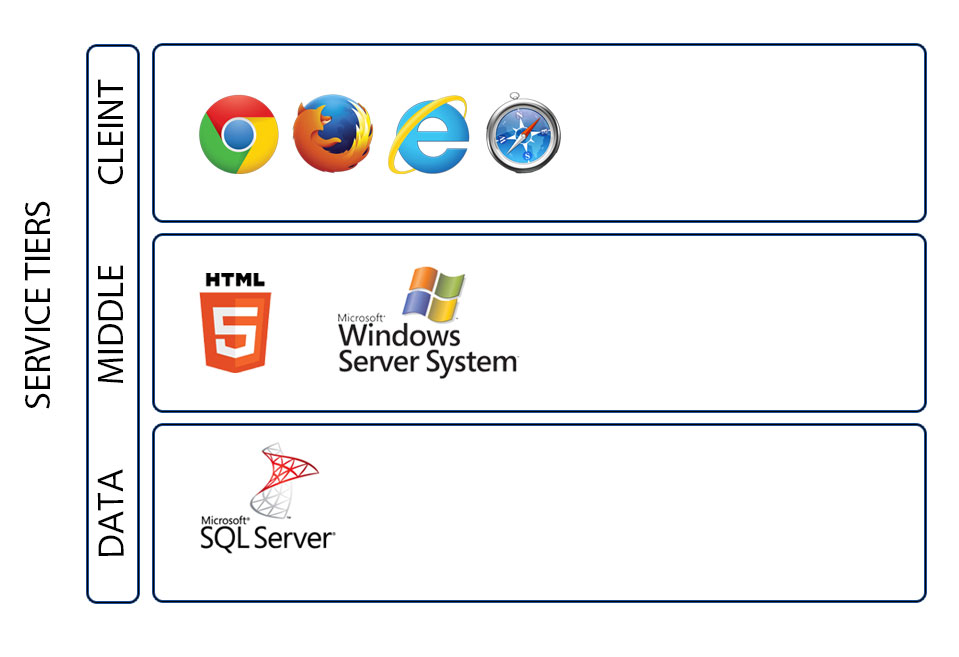
****

Figure 4

## Application Security

Data is validated at the client, server, and module interfaces, then again through certain security controls at the database level. The application makes use of industry standard methods of security, and care is taken to protect the application from common attack types such as SQL Injection and Cross Site Scripting. User input is validated at multiple stages to ensure both application security and data integrity.

## Application User Roles

The application has four main roles: Super Administrator, Super Company Administrator, Company Administrator, and User. The matrix below defines which roles have access to specific features.



Figure 5

## Configuration Settings Management

The Data Acquisition Application provides access to manage the configuration of the system’s processing engine and application to users assigned to certain roles. This extensive set of configuration options is designed to reduce implementation time while giving the user control to fine-tune or update the way the application processes and displays data.

## Exception Resolution

If the client’s configuration settings require that a company role (Claimant/Seller/Buyer/Buyer HQ/Billed-to Buyer) or product be matched to the client’s master data, the processing engine will move these lines to an Exception holding table and make these records available in this section of the application. Users can access the lines in exceptions and manually search for matches in master data. Worked Exceptions data is then reprocessed on a configured schedule and exported to an endpoint file for transmission to the client.