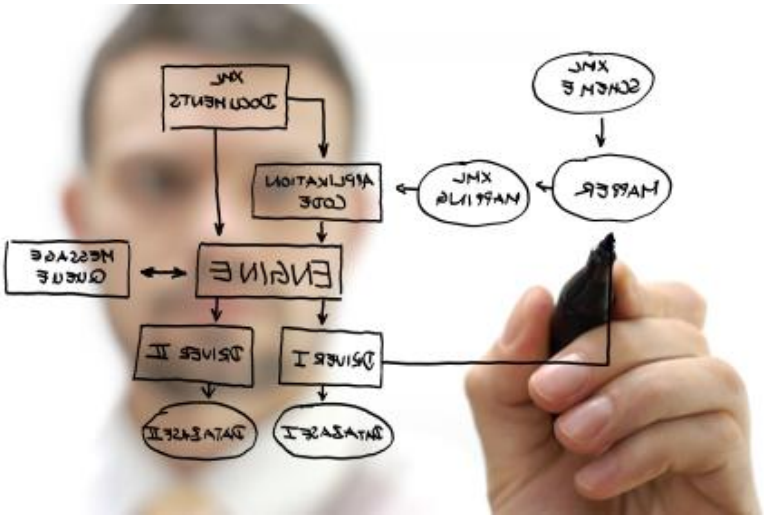
2010

Axcess Consulting &
Axcess Canada Inc.

Axcess Development
Team



# [ARMNET ARCHITECTURE OVERVIEW]

Logical and Physical Layout of the ARMnet Financial Product Management Solution

## CONTENTS

SEPT 23, 2010

PREPARED BY

AXCESS CONSULTING PTY LTD & AXCESS CANADA INC.

VERSION 1.0

This document contains confidential and privileged information.

Any enquiries should be made to Axcess Canada,

922 Cedarpointe Court, Collingwood, ON L9Y5C7

## COPYRIGHT WARNING



## CONFIDENTIALITY WARNING

## REVISION HISTORY

| VERSION | DATE | AMENDED BY | DESCRIPTION OF CHANGE |
|---|---|---|---|
| V0.9 | May 20, 2010 | Geoff Bennett | Draft |
| V1.0 | September 23, 2010 | Tim Shkolnik | Final |

# OVERVIEW

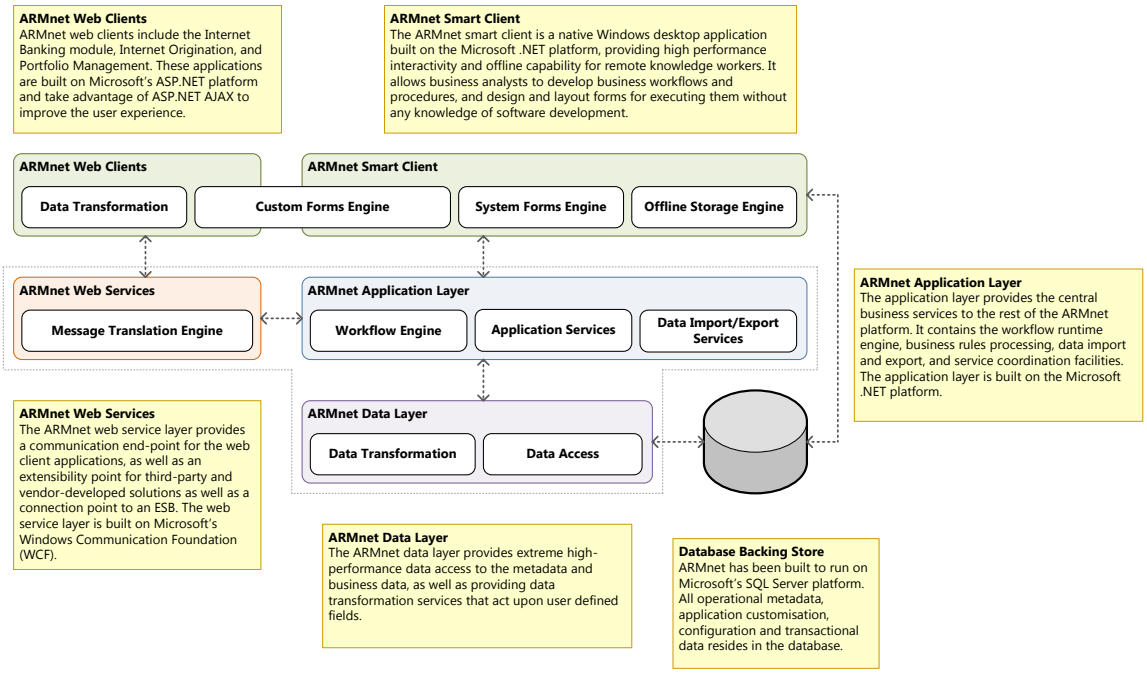
This document will provide an introduction to the architecture and layout of the ARMnet Financial Product Management Solution origination and servicing platform. It is designed to help the reader gain an understanding of the impact of an ARMnet Financial Product Management Solution deployment on their infrastructure, and to help evaluate the implementation strategy and requirements that will affect the business.

While the document is basically technical in nature, it has been written with consideration of non-technical executive staff and other executive management people involved in the sales and implementation process, both within and external to the Axcess Consulting Group Pty Ltd., and Axcess Canada Inc.

# 1.0 - LOGICAL ARCHITECTURE

ARMnet Financial Product Management Software uses a hybrid smart-client/n-tier approach, providing an extremely flexible deployment model, and simple configuration and maintenance. This allows for high-performance data transfer, and flexible coupling to external services and third-party platforms.

**ARMnet Logical Architecture**




## 1.1 - SMART CLIENT APPLICATION

The smart client itself is capable of stand-alone, offline operation utilizing a local data store based on Microsoft's SQL Server. This can be a local server instance, or the SQL Server Compact embedded database. Synchronization happens automatically when the smart client finds the network server is present.

A single workstation environment is capable of executing multiple instances of the smart client, configured for different server and local store endpoints, maintaining separate configuration and execution caches.

## 1.2 - WEB CLIENTS

The web clients install into a standard Microsoft Internet Information Server environment, and utilize the standard .NET mechanism to configure them. The configuration can be altered simply and quickly using Microsoft's own configuration tools, or a simple text editor.

The websites are capable of – and have been designed to – run within a certificated environment, and have been defensively developed to avoid SQL injection and cross-site scripting vulnerabilities. The Internet Banking application has been extensively testing using Fortify SCA (https://www.fortify.com/ssa-elements/testing.html).

The user authentication mechanism is configurable between standard text-box style entry of credentials, or an on-screen keyboard for entering of the password. Credentials are encrypted in the browser, using the industry standard RSA algorithm, before being transmitted to the server as an added precaution.

It is possible to run multiple web clients as individual web applications, each with their own configuration, and each communicating with a separate backend. This allows for extensive brand customisation and integration with existing internet presences. It also allows ARMnet customers to segregate portals to different marketing verticals (i.e. Customer/Client, Broker and/or Dealer) and still utilize one database instance.

## 1.3 - WEB SERVICES

The web service layer is written using Microsoft's Windows Communication Foundation, which provides standards compliant web service endpoints, along with a clear configuration environment and native tools.

The endpoint configurations are available to administrative staff, and can be modified to suit the make-up of your data centre and to follow any operating environment guidelines your organisation must follow.

The web service API surface area covers a large portion of the actual operation of the ARMnet platform, and as such provides an excellent connection point to third-party applications and corporate ESBs.

The web services are hosted inside Microsoft Internet Information Server, and can be secured using Secured Socket Layer (SSL) certificates. It's also possible to completely change the communication mechanism used by WCF to suit your operating environment.

## 1.4 - APPLICATION LAYER

The application layer is the heart of the ARMnet Financial Product Management Software platform. It is a suite of libraries and service executables that perform the operational aspect of the system.

Each component of the ARMnet platform maintains a set of these binary libraries to ensure that a single set of metadata is usable across the breadth of the deployment. This also permits the entire application to be developed from a single source-code base, making maintenance and product enhancement significantly easier for us to manage.

The application layer contains the workflow engine, which is the heart of the end-user configurability of ARMnet, allowing business analysts to modify and extend the platform with no knowledge of software development.

The application services in the application layer provide the runtime interpretation environments for different system services exposed through the workflow engine, such as transaction processing, status changes, and documentation generation, to name a few.

The final major component of the application layer is the data transfer services. This extensive library of transfer functions allows you to import and export data just about any file-based format you require, and coupled with the file system monitoring functionality, provides another extensibility point through file operation triggered automated import and export.

## 1.5 - DATA LAYER

ARMnet Financial Product Management Software is an inherently data driven application. Not just from the transactional data point-of-view, but the entire system is driven off metadata, allowing you unprecedented configuration of the final product.

This configuration comes from two major components; the data transformation module, and the data access module.

The data access module is chiefly concerned with reading and writing data against the SQL server. It has been designed to communicate as quickly as possible with the database, providing minimal processing overhead.

The data transformation module provides services unique to ARMnet Financial Product Management Software, in that it can process the resultant datasets for use in the UI layer by transforming embedded ARMnet field codes into workable data atoms. This occurs without the higher layers being aware.

## 1.6 - DATABASE BACKING STORE

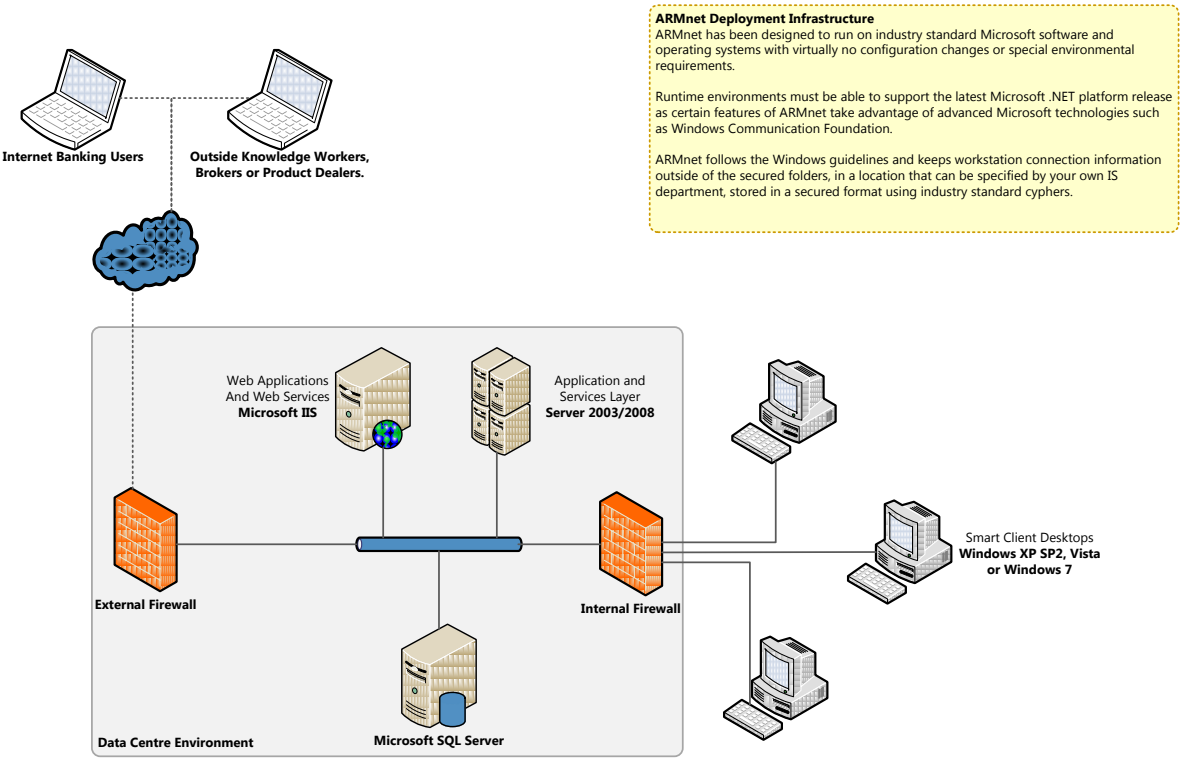
ARMnet was designed to operate on the Microsoft SQL Server platform using base-level T-SQL statements. This allows the system to be deployed into any SQL Server environment from 2005 onward.

This level of adaptability is what also allows the embedded SQL Server software to be used as a local data cache.

# 2.0 - PHYSICAL ARCHITECTURE

ARMnet does not impose any constraints on the deployment environment other than meeting minimum operating system and server requirements. This allows for quite a flexible deployment allowing you to fit ARMnet Financial Product Management into your existing infrastructure with a minimum of configuration effort.

**ARMnet Physical Architecture**




## 2.1 - EXTERNAL CLIENTS

External clients will connect using one or more of the web client applications. There are no environmental requirements on these machines, other than a relatively recent web browser. The existing web applications have been tested on the most recent versions of all the major browsers (i.e. Internet Explorer 8.0+, Mozilla Firefox, Opera and Google Chrome) with only minimal aesthetic differences and no functionality issues.

## 2.2 - DATA CENTRE

The environment required in the data centre again is minimal for a basic deployment. The application layer should be deployed onto a Windows Server 2003 or 2008 environment, with sufficient resources as to support your target user load. As this is impacted as much with user numbers as your individual ARMnet metadata configuration, performance will have to be gauged on a site-by-site basis.

The default database is a SQL Server 2005 compatibility level file, ready to drop onto an existing database server, but due to the heavily data-driven nature of ARMnet, preference for a stand-alone server is strong. Again, the performance of this aspect of the architecture is heavily dependent on the individual customisations made to your application and can only be gauged on a site-by-site basis.

The web services utilise a standard Microsoft Internet Information Services installation to host the endpoints. The configuration of the endpoints defaults to the WS-Basic profile, but using the configuration tools supplied by Microsoft, or a text editor, it is possible to change the profile and communication protocol to whatever your operational requirements are.

## 2.3 - WORKSTATIONS

The workstation environment is supported by any Microsoft operating system capable of sustaining the most recent versions of the .NET platform. At the time of this revision of the document, this was Microsoft .NET Platform Version 3.5 Service Pack 1.

As some workflow processing is offloaded on to the workstation to enhance scalability of the back-end services, a performance increase can be achieved by installing sufficient amounts of RAM and a reasonably recent processor.

The .NET Windows Forms architecture is reasonably graphics intensive, but most recent release on-board graphics cards should be capable of providing more than sufficient performance.

Microsoft Terminal Services also provides an extremely credible way to manage desktop environments and scale that properly. This eliminates the need to manage BIN's locally on each desktop and typically provides enhanced performance over trying to connect the Hosted SQL Server database to desktop ARMnet BIN's via Virtual Private Network connections. On Terminal Services you are essentially passing keystrokes back and forth so traffic is minimize.

In providing this Terminal Services will allow you to simply host the application remotely OR an entire desktop environment. If the later is chosen it is important to recognize optimal performance benchmarks to insure the user experience is sound and minimize bandwidth. This is reviewed during the Joint Scoping Analysis.

## 3.0 - SUMMARY

This document should help you to understand the impact of an ARMnet Financial Product Management Software deployment in your operating environment. From the topics covered, you should now have an understanding of the way the individual elements of the ARMnet platform relate to each other, and the technologies utilised to provide them.

As the ARMnet platform is heavily data driven, and highly configurable, it is not possible to provide concrete performance benchmarks due to the highly variable nature of each installation. However, we have endeavoured to provide as much leverage as possible into the environment to allow you to configure the system to extract the maximum performance from your infrastructure.

There is no other financial product management software package available today, short of an actual software development environment, that provides the level of customisability and flexibility that ARMnet Financial Product Management Software does. You are free to define your business processes, data entry, and data interrogation as you see fit, not as defined by the architecture of the application, and using business analysts instead of software developers. This allows you to leverage one of your biggest business assets, your people, and have a direct and positive influence over your ARMnet Financial Product Management Software implementation.