eMerge Development Summary

August 16

2012

Documenting the current .net developments for eMerge-IT, including recommendations with regards the deployment of infrastructure for supporting a shared development process.

By Simon Barnett

Shared Development Platform

With a view to:

- the development of the projects listed below by a larger, geographically disparate team
- maintaining the latest versions of source code in a centralised location

It is suggested that all developers use a single terminal server such that all code resides in a single location.

This configuration will also allow developers sessions to be monitored for support and management

For speed / ease of access this terminal service should be hosted in the cloud.

Online, remote online, local physical media and remote physical backups should all be implemented to ensure security of data

Additionally this platform should host a development wiki, containing a knowledge base of documentation / installation instructions available from anywhere via http

It may also prove useful to install team-speak to provide a multi user open voice channel for developer support.

Hardware

- Cloud Based Terminal server @ dev.emerge-it.co.uk
- Windows 2008 server
- UNC access
- Physical Media Backup

Software

- Terminal services
- Team Foundation server (to prevent issues with multiple users of the same codebase)
- Visual Studio 2008
- Visual Studio 2010
- Mono (for Android)
- IIS
- MSSQL
- Priority (for testing / creating Priority Batch files)

Projects

Mobile

General Mobile Design Concepts

Polymorphism

Polymorphism is the ability of a piece of code to reconfigure itself based on a dynamic data set. This is important because it means that each framework can be re-used between multiple customers, while still being configurable to the individual needs of a specific customer.

Polymorphic developments create a reusable framework layer that is shared between all customers and isolates customer specific code in such a way that additional customer modules can easily be created and distributed.

This separation of code into re-useable framework and isolated customer specific development means that 90-95% of any given project is re-usable and that updates / fixes made to the framework are automatically included in each customer rebuild.

The ability to make customer specific interfaces is key to delivery of the type of solutions described by the “target audience” section.

Target Audience

Mobile developments currently focus on industrial applications. These can be defined as “Turn key” applications where only a subset of Priority data is used and the emphasis is on data capture, often including the use of barcodes and/or rfid.

As such our mobile applications do not seek to compete with other superficially similar Medatech developments including web based access to Priority or similar remote access to management data.

Industrial applications are necessarily process driven, and developments focus heavily on streamlining that process for the end user. It is therefore vital to be able to heavily customise the user interface as described in the section on polymorphism, as opposed to the “one-size-fits-all” approach favoured by Medatech.

Development and Deployment Tools

Delivery time is a function of profitability and so we must seek to minimise this time as much as possible.

The aim here is to produce such tools – installers, configuration tools etc – that the development and deployment time is minimised and can be undertaken largely by project managers, and without significant input from the development team.

GUI design Interface

The development time of customer specific modules should be minimised, and the development of interface by project managers should be supported by the creation of

tool sets that provide graphical interfaces for editing the customer specific xml data used by the projects (e.g. forms, columns, data definitions etc).

The aim is to have as much of the development as possible undertaken by the project manager working directly with their customer as possible.

Auto Configuration

Auto configuration is basically how slick the install is. This can be measured directly as the amount of time elapsed between receipt of purchase order and the delivery of the project.

Configuration development should be viewed as a sliding scale - ranging from "nightmare install" through to "unattended install" - with diminishing returns. I.e. there is no point spending a week fine tuning the install for a time saving of only a few seconds each time.

Within reason however we should be aiming for as close to the "unattended install" as possible (with the gap filled by proper installation documentation).

Such "Auto configuration" developments have already reduced the installation time of the SOAP service from 3 days to 1.

Documentation

The introduction of more developers makes it increasingly important that proper documentation of the projects exist, especially in the area of deploying projects.

Such resources will also reduce the required know-how during the deployment process and so assist project managers to install these projects for themselves without the assistance of the development team.

As mentioned under the shared development platform topic, a wiki is likely to be the best tool for centralised sharing of such resources between the development team and project managers.

Priority Mobile Data Capture

Design Concepts

Offline mobility

Central to the concept of eMerge's Mobile Data Capture platform is the use of on-board data and burst data synchronisation. This allows data collection to proceed in fully disconnected scenarios.

HTML (of any number) is always downloaded from a web server, and will therefore always require some level of connectivity that could limit the fully disconnected functionality required by the contemporary field based personnel (i.e. the reality is that they don't always have connection).

Portability between devices

With the recent proliferation of mobile device types we are back circa 1985, choosing between Commodore, Dragon, Acorn, Sinclair and 1000 other device types. Eventually a winner will emerge, cornering the market like PCs did in the 90s.

Till that point we need to be able to support multiple platforms, although to minimise development costs we should do so using .net as the common development platform.

The release of a MONO (open source net) dot net CF Framework for Android should enable us to reuse code between Android and Windows Mobile. It should also be possible to port the project to Windows Phone.

Given the current market share (in terms of deployed hardware and projected new device sales) the order of precedence we should be following with regards porting the Windows Mobile project to other platforms should be as follows:

- Android
- Windows Phone
- IOS

Project Status

Current State of Development

Version 3, the polymorphic version of the original service engineer application is currently under development for Strauss.

Scope for Future Development

The nature of the development is such that it will support a wide range of customer modules.

Initially a service engineer application similar in scope to v2 is under development, but this platform can be the basis for a wide range of customer led future developments.

Priority Shop Floor Data Capture

Design Concepts

Embedded Windows

Fortunately a clear market leader has already been established in the embedded market. Windows CE is almost ubiquitous for a vast range of industrial applications, being used in everything from set-top boxes and white-goods through to process control in large scale manufacturing equipment control units.

A wide variety of handheld barcode scanners support this operating system, and tend to come in ruggedized form factors ideal for industrial environments.

Project Status

Current State of Development

Already enjoying a wide customer base the framework enables customer specified modules to be written based on a range of pre-existing warehouse transactions.

Scope for Future Development

A version 3 SFDC would move a number of coded properties in the customer modules into xml files, enabling a GUI development tool to be implemented to allow project managers to tailor standard process to their customers' specification.

Priority Web Integration

General Web Design Concepts

Offsite hosting

Most business web hosting is offsite and not local to the installation of ERP software. To support this topology we must provide data from Priority to a remote web server. This is done using xml feeds to get data from Priority and by using our customers SOAP service to load captured data back to Priority.

3rd Party implementations

All web development data consumed by Priority web interfaces should also be made available in raw form (i.e. without an interface) for use by third party developers to build web applications from.

XML portable data sources

To enable off-site hosting and third party implementation all web data sources should be readily portable. XML is a text based data hierarchical format that can easily be deployed between multiple platforms.

Separation of functionality from design

Separation of function and design is a key concept in web development. Priority CMS does this by using the “master page” functionality of .NET.

The master pages contain the look-and-feel of the rendered pages but it is noted that there is currently a deficit in our ability to deliver high quality page design

Content Manager

Description of Development

An XML based web content WYSIWYG management system

Current State of Development

Production

Scope for Future Development

This module can be expanded to incorporate customer specific module, including those already pre-developed modules listed below.

e-Commerce Module

Description of Development

Using the Realex integration tools and a Priority XML part feed to provide inline card validation from an eMerge CMS website

Current State of Development

Production

Scope for Future Development

Customer driven implementation of alternative authentication providers

Service Desk Module

Description of Development

Enables customer self care for logging support calls with the service desk

Current State of Development

Production

Scope for Future Development

Customer led customisation of service

Customer Site Survey Module

Description of Development

Allows customers to complete Priority site surveys on-line from their Priority CMS website

Current State of Development

Production (internal)

Scope for Future Development

Customer led

Billboard Module

Description of Development

Displays rotating web pages based on a Priority XML feed on a wall display

Current State of Development

Beta

Scope for Future Development

Customer led

Windows Developments

Integrations

Royal Mail Postcode Integration (rmxcli)

Description of Development

Imports postcode data to Priority from a centralised Royal Mail XML service

Current State of Development

Production (internal)

Scope for Future Development

XML server database requires import routines creating to maintain updates

Full update of the current database

World First Integration

Description of Development

To integrate automatic foreign currency transactions into Priority

Current State of Development

Production

Scope for Future Development

Customer Led

GPS Tracking Integration

Description of Development

To integrate GPS tracking data into Priority

Current State of Development

Prospect

Realex Merchant Services Integration

Description of Development

Priority Merchant services integration to enable credit card authentication and transaction, direct from Priority or via the Priority CMS e-commerce module

Current State of Development

Production (both web and from Priority)

Scope for Future Development

Customer led

Telephony Tools

Telephony Screen Pop

Description of Development

Displays the Priority screen correlating to the telephone number of an inbound phone call

Current State of Development

Release Candidate

Scope for Future Development

Tidy TAPI interaction

Test on multiple platforms

Customer led

Priority Fax Client

Description of Development

Send faxes from a network FAX service

Current State of Development

Release Candidate

Scope for Future Development

Auto configuration

Customer Led

WPF Tools

wpfDesktop HTTP

Description of Development

To enable activation of local executable files from a cloud based Priority installation

Current State of Development

Release Candidate

Scope for Future Development

none

PDF Printer

Description of Development

To automatically print PDF documents from a cloud based Priority installation

Current State of Development

Release Candidate

Scope for Future Development

None, although similar customer led developments using the wpfDesktop are possible

Code Generation Tools

Precursor

Description of Development

Automatically generates Priority SQL for writing cursor functions

Current State of Development

Production

Scope for Future Development

Internally led

PDAGen

Description of Development

GUI tool for the generation of xml configuration data for Mobile Data Capture framework

Current State of Development

Alpha

Scope for Future Development

Rewrite for v3

Logging Tools

Mail Proxy (SMTProxy)

Description of Development

Captures inbound/outbound STMP email and recording it in Priority

Current State of Development

Beta

Scope for Future Development

Fix the bug in handling of binary data transmission

Windows Event Logger for Priority

Description of Development

Enables writing to the Windows Event Log from within Priority SQL

Current State of Development

Production

Scope for Future Development

None

HookPriority

Description of Development

Hooks the winform.exe of Priority such that logged on/off times can be recorded for desktop uses of Priority

Current State of Development

Beta

Scope for Future Development

Auto configuration

Graphical Tools

lSpace

Description of Development

Graphically display template and cut-sheet stock dimensions to enable users to visually inspect relative size

Current State of Development

Alpha

Scope for Future Development

Connect to Priority data source