

Rich Client Platforms

1 Introduction

There has been a lot of interest recently in 'rich clients', clients that enable a user to have 'rich' interactions with software applications. The richness is in contrast to 'poor' interactions in existing applications with a HTML-only web browser client.

Much of the literature today is focused on improved web browser based interactions, and not on what really the user needs. What users need is a way to interact with applications that automate their current business processes. Current business processes deal with information manifested as data, documents, images, video, etc. For instance, contracts, purchase orders, invoices, etc. are all documents. All the rich client platforms today address specific types of information, and are therefore not suitable for all types of applications.

The purpose of this paper is to first define a canonical model for 'rich client', define attributes that can be used to characterize various rich client platforms, and then define many of the existing rich client platforms using these characteristics. The hope is that it will be a useful guide for designing application interfaces and for deciding which rich client platform(s) would be most suitable for their needs.

1.1 Scope

- Traditional
- General Interface
- Laszlo
- JackBe
- Asperon
- Nexaweb
- Isomorphic
- Java Desktop
- SAP Web DynPro
- IBM Workplace
- Microsoft Avalon
- Microsoft Office
- Adobe IDP

1.2 Revisions

Revision	Date	Comments
0.9	02-17-2005	Initial draft

0.91	04-14-2005	Updated JackBe information and References
------	------------	---

2 Canonical Model

To analyze the various rich client platforms and compare them using common characteristics, a canonical model for a rich client platform is presented below.

A rich client platform typically consists of 3 relevant (to this analysis) tiers:

1. User Interface Tier

This tier presents information to the end user (in a suitable format), reacts to events that are generated because of interactions with the user and/or changes to the information, and interacts with the presentation server tier for sending/receiving information or performing operations from/to the EIS.

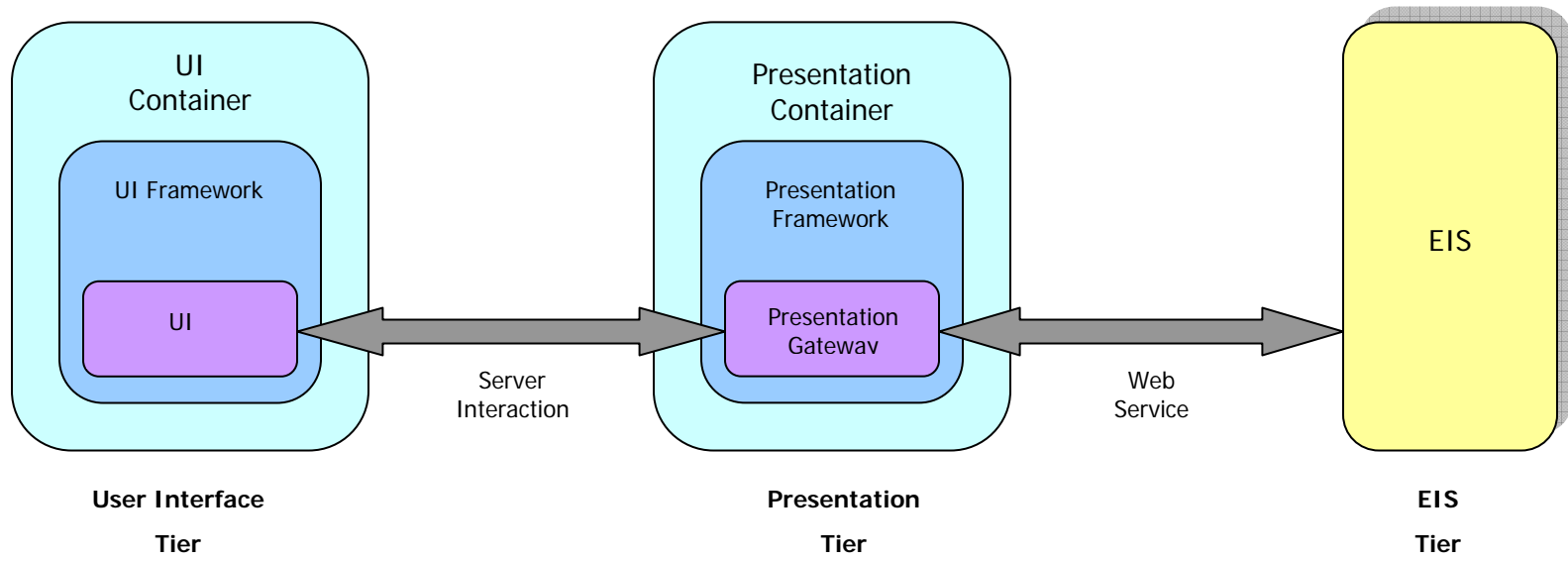
2. Presentation Tier

This tier acts as a gateway between the user interface and enterprise information system(s). It manages the lifecycle of the rich client UI framework and application, the UI description, conversion of information between the EIS and UI tier formats, and forwards application events to the EIS.

3. Enterprise Information System (EIS) Tier

This tier hosts the business logic of a single or multiple applications and exposes a web service interface to the presentation server.

Rich Client Platforms



3 Characteristics

UI Container

The container for the rich client user interface framework. For instance, a web browser, but can be other applications like Eclipse.

UI Framework

Framework that provides a container for the rich client user interface defined for an application.

UI

The rich client user interface developed for a specific application.

Presentation Container

The container for the presentation framework. Usually, a Servlet container but can be anything else.

Presentation Framework

Framework that is the server counterpart of the UI framework layer. It provides a container for the presentation logic – screen/page description and flow – developed for an application.

Presentation Gateway

Presentation Logic for a specific application. This also translates user requests and operations captured by the UI to invocations on EIS(s).

Server Interaction

Mechanism used by the UI to interact with the presentation gateway.

Information Type

The type of information that a user can consume or interact with, via the UI. Information types include data, text, image, graphics, audio, and video.

Operations

What operations can be performed on the information. This is identified as a subset of CRUD.

Openness

Whether the platform is open and can be used with multiple EIS, or is restricted to specific EIS or technologies.

Size

The size of the UI Framework, essentially the cost of initial download.

Link-ability

Ability to refer to a page/window/screen of the rich client UI by a URL.

Information quality

Mechanisms to ensure that the information entered by the user is accurate. These can range from simple data type checking to enforcing business rules in the user

	interface.
IDE	Environment for developing the user interface.
Offline	Whether the rich client can be operated when the user is not connected to the network.
Provisioning	Mechanism to version the user interface – automatic or manual.
Desktop Integration	Other applications on the user's desktop that the user interface can integrate with.

4 Rich Client Technologies

<Brief description of the technologies surveyed – TBD>

5 Comparison

Characteristic	Traditional	General Interface	Laszlo	JackBe	Asperon
UI Container	Web Browser – IE 6.0, possibly other browsers	Web Browser – IE 6.0 Other browsers upcoming	Flash plug-in	Web Browser – IE, Netscape	Java Plug-in
UI Framework	HTML/JavaScript interpreter	JavaScript	Flash player	Mini Assembler (DHTML engine)	KWT
UI	HTML, JavaScript	XML, JavaScript	Flash	HTML	Java
Information Type	Data, Images, Audio, Video	Data, Images	Data, Images, Audio, Video	Data, Images	Data, Images
Size	0	280K	Flash plug-in	20K	700K???
Operations	Data (CRUD), Image (R), Video (R)	Data (CRUD), Image (R)	Data (CRUD), Image (R), Video (R)	Data (CRUD), Image(R)	Data (CRUD), Image(R)
Link-ability	All pages can be accessed with a unique URL	Pages cannot be accessed by URL	Pages cannot be accessed via URL	???	Screens cannot be accessed via URL
Information Quality	JavaScript	JavaScript	LZX (Script???)	DHTML, JScript	Screen XML (proprietary language)
IDE	Any IDE that can be used to develop HTML/JS/JSP	Browser-based IDE	LZX, JavaScript IDE4Laszlo Eclipse plug-in	DHTML, JScript Proprietary Browser-based IDE (JackBuilder) developed using itself	Java, XML IDE
Development	Java, JSP	None	LZX libraries, Java	JackBe enabled forms	XML, Java (data buffers, commands)
Server Interaction	Single Synchronous HTTP	Multiple, Concurrent Synchronous, Asynchronous, Real-time HTTP, SOAP, extendible	Single Synchronous, Asynchronous, Real-time XML/HTTP	Single, Synchronous, HTTP	Single Synchronous Binary encrypted form XML/HTTP

Rich Client Platforms

	Servlet Engine	Web Service	Servlet Engine		
Presentation Container	Servlet Engine	Not required	Servlet Engine	Servlet Engine	Servlet Engine
Presentation Framework	Struts, etc.	Not required	LZX libraries, Data binding framework	JackBe framework	AppProjector Controller
Presentation Gateway	Application specific	Not required	Application specific	Application specific	Application specific commands
Offline	No	Yes (as long as browser is open?)	No	No	No
Provisioning	No provisioning needed	Yes	Yes (assuming flash player exists)	Yes	Yes
Desktop integration	No	No	No	No	No
Openness	Open	No	No	No	No

Rich Client Platforms

Characteristic	Nexaweb	Isomorphic
UI Container	Java plug-in	Web Browser – IE, N, M
UI Framework	Applet	JavaScript, DHTML (ISC Client)
UI	Java	DHTML, JavaScript
Information Type	Data, Image, Vector Graphics	Data, Image
Size	150K	???
Operations	Data(CRUD), Image(R)	Data(CRUD), Image(R)
Link-ability	Pages cannot be accessed via URL	Pages cannot be accessed via URL
Information Quality	Java code	JavaScript
IDE	XUL, SVG Eclipse based IDE, IBM WSAD	XML, JavaScript IDE
Development	XUL, SVG Java	XML tag language Java (to connect to data sources), JavaScript (to extend client libraries)
Server Interaction	Single Synchronous, Real-time XML/HTTP Servlet Engine	Single Synchronous, Real-time ??? Servlet Engine
Presentation Container	Servlet Engine	Servlet Engine
Presentation Framework	Nexaweb Server	ISC Server
Presentation Gateway	Application specific Java code	Application specific Java/C# code

Rich Client Platforms

Offline	No	No
Provisioning	Yes	Yes
Desktop integration	No	No
Openness	Yes (relatively)	No

Rich Client Platforms

Characteristic	SAP Web Dynpro	IBM Workplace ¹	MS Office	MS Avalon	Adobe IDP
UI Container	Web Browser	Eclipse	Office application		Acrobat Reader
UI Framework	JavaScript	Rich Client Technology Platform	.NET		Acrobat Reader
UI	HTML, JavaScript	Markup language (that defines components, views, layout and update information)	C#		PDF
Information Type	Data, Images	Data, Images, Text Others depending on editors supported	Data, Images, Text, Image, Graphics		Data, Images
Size	Size of CSF	Size of the application module	Size of XEP		Initial install of acrobat reader and subsequent updates
Operations	Data (CRUD), Image (R)	Data (CRUD), Image (R), Text (CRUD) Others depending on editors supported by the platform	Data (CRUD), Images (CRUD), Text (CRUD), Image (CRUD), Graphics (CRUD)		Data (CRUD), Image (R)
Link-ability	Application level	Not supported	Not supported		Not supported
Information Quality	JavaScript	Java code (in application specific parts)	XML Schema		ECMA Script embedded in PDF
IDE	IDE – plug-in to Eclipse	Eclipse	Visual Studio		Adobe Designer LiveCycle
Development	UI declarative language Java code for event handlers	Java, J2EE, XML	XEP development using C# Smart tags and		Includes complete platform. Adapters to exchange

¹ Does not include browser interface for the rich client edition, or the micro edition

Rich Client Platforms

			documents IBF		data with enterprise systems
Server Interaction	XML/HTTP???	<anything> that supports invocation of EJB services from the client	Web Services		XML/HTTP
Presentation Container	Web Dynpro Runtime	Eclipse	Not required		Adobe LiveCycle
Presentation Framework	Web Dynpro Framework	Rich Client Technology Platform	Not required		Adobe LiveCycle
Presentation Gateway	Java code to map UI events to EIS operations	EJB access proxy	Not required		Java code for adapters
Offline	No	Yes	Yes (very limited)		Yes (very limited)
Provisioning	Yes	Yes	Yes		Yes
Desktop integration	No	Yes	Yes		No
Openness	Proprietary	Open	Proprietary		Proprietary

6 References

- General Interface.
http://www.tibco.com/software/business_optimization/generalinterface.jsp.
- Laszlo
<http://www.openlaszlo.org/download/eclipse/>
<http://www.laszlosystems.com>
- JackBe
<http://www.jackbe.com>
- Asperon
<http://www.asperon.com>
- Nexaweb
<http://www.nexaweb.com>
- Isomorphic
<http://www.isomorphic.com>
- SAP Web Dynpro
<https://www.sdn.sap.com/sdn/developerareas/webdynpro.sdn?node=linkDnode6-2>
- IBM Workplace
<http://www-306.ibm.com/software/info/workplace/index.jsp>
- MS Office
<http://www.informationbridgeframework.com/>
- MS Avalon
<http://msdn.microsoft.com/Longhorn/understanding/pillars/avalon/default.aspx>
- Adobe IDP
<http://www.adobe.com/enterprise/idp.html>