

An Electronic Forms Solution with DataPower

WebSphere User Group
IBM Bedfont, 3rd March 2009

S C Smith
Technical Architect
Smart421Ltd

Presentation is also available from
<http://www.smart421.com/aboutus/news/030309.asp>

- Quick DataPower and XI50 Overview
- Presentation Positioning
- Problem Statement
- Requirements breakdown
- High Level Design
- Detailed Design
- Testing Approach
- Questions?

- Part of the KCOM Group (£500M, 2500 people)
- IBM Premier Business Partner
- IBM Tier 1 WebSphere Integration Services Partner

- Smart421 SOA Consultancy
- Largest DataPower practice in Europe
- Experience from numerous WebSphere projects

XML Accelerator XA35

Feature Overview

A Subset...



- XML Parsing Acceleration
- XML Transformation Acceleration
- SSL Acceleration
- Schema Validation

XML Security Gateway XS40

Feature Overview

Everything from the slide before plus...



- XML Firewall
 - Message Routing
- Threat Protection
 - Denial Of Service, SQL Injection, Malicious XML etc.
- Message and Field Level Security
 - Encryption, Decryption and Message Signing
- Authentication, Authorisation and Auditing
 - LDAP, WS-Security, SAML, SPNEGO, LTPA etc...

Integration Appliance XI50 Feature Overview



- Everything from the slide before plus...
- Protocol Bridging
 - HTTP, HTTPS, MQ, JMS etc.
- Multi-Protocol Gateway
- Transformation between XML & Non-XML Formats

- A national revenue service has a regulatory requirement to support electronic submission for diamond mine company quarterly tax return levies
- There is also the opportunity to bring many other tax collection forms online (e.g. Other utilities, other mining, etc)

- Many different forms exist (e.g. Electricity, gas, water, mining specific to the industry, etc)
- All are currently manually completed, posted to revenue, sent back to company for correction, etc.
- Some require regular monthly or quarterly submission
- Company submissions, so low volume (1000s not millions)
- High value

- Regulatory – aggressive timescales, electronic requirements
- Efficiency (back office)
- Streamlined/Quicker revenue collection
- Company preference for electronic submission

- Quick development, test and deployment of diamond levy form
- Ease of use by mining companies
- Must support complex business rules validation to reduce errors in completed form
- Non-Repudiation (reference numbers, guaranteed delivery, etc)
- Integration with existing back office revenue system (existing data format)

- Validation should be automated and support a high degree of complexity

E.g. Not just:

`'field4 is a decimal between 0.00 and 100.00'`

But:

`'field4 must be completed if field104 is completed, unless field6 is completed'`

Or:

`'field7 should total fields 5,6,10,12 or 11,13,4 if foreign worker
allowing for $20 rounding error'`

Or:

`'fieldX should pass luhn/modulus10 validation'`

Or:

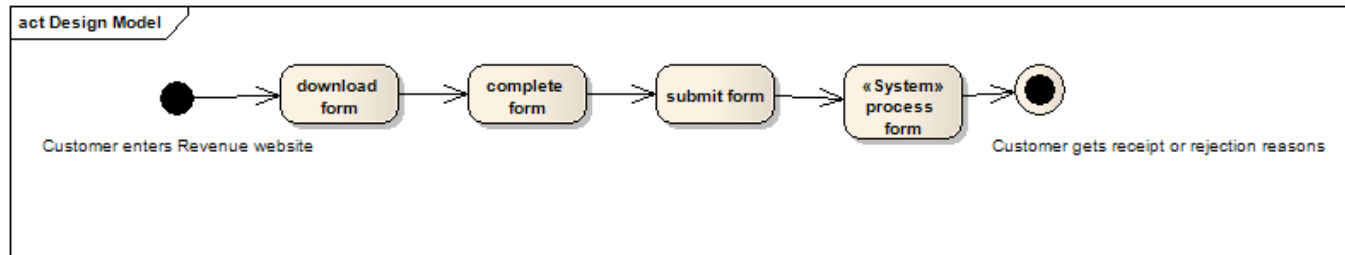
`'fields10-40 should only be completed if date of birth results in them being over 60 if male,
65 if female'`

- Validation business rules should be written and maintained once (no duplication, no validation in multiple systems leading to out of sync rules, etc)

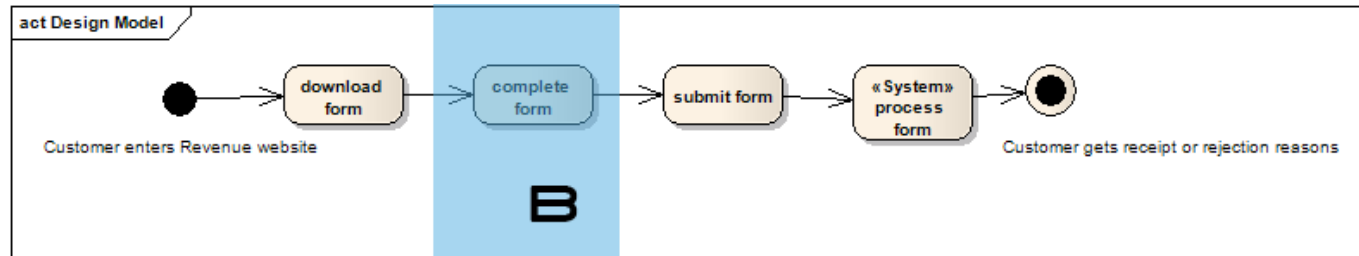
- The resultant design framework should support the addition of more forms with their own validation
- Support future B2B aspirations
- Robust
- Secure
- Must be delivered in about a week !

You might have noticed.....

- No mention of web services
- No mention of SOAP or XML
- No mention of transformation
- No mention of 'XML acceleration'
- So, why DataPower ?



- A. Download form : customer should be able to download the electronic form
- B. Complete form: customer able to complete the form offline
- C. Submit form: customer submits tax return
- D. Process form: system validates the form. If valid, returns receipt to customer. Otherwise rejects form with rejection reasons.



- MS Excel chosen
- All customers have it and will be reasonably familiar with it
- Very quick to develop form and protect it from change
- Can be printed out by customer and manually completed as last resort
- Other possibilities (Adobe Forms, etc) may be looked at in the future.
- Online Forms (application server) would require extra hardware, integration, development effort and would offer no offline completion facility.

- XML Backing/Mapping of spreadsheets supported since Office XP
- Mapping of spreadsheet fields to XML can be established quickly
- Offers limited flexibility in XML and spreadsheet layout to be successful
- Good match if you have control of both the XML (XSD) and the spreadsheet design
- Add a bit of VB behind some spreadsheet buttons and you have a simple application
- Customer presses button to export tax return (in XML format)
- NOTE: Weak typing in Excel means output XML is not guaranteed to be valid to schema.

Excel XML Mapping

Microsoft Excel - DEL162 EDIT.xls

File Edit View Insert Format Tools Data Window Help

Type a question for help

D14 fishes

DIAMOND EXPORT LEVY DEL 162

Diamond export levy (DEL) return
Section 2 of the Diamond Levy Act, 2007

1. REGISTRANT PARTICULARS

a. Client code 00281124

b. Assessment period from (YYYY-MM-DD) 10/12/2007 To 10/12/2008

c. Client types Producer

d. Particulars of Registrant

Registered name Fish

Trade name fishes

Physical address 12 fasf ssis

ksd adsdas

klklkls Postal code 1234

2. LEVIABLE EXPORT TRANSACTIONS

a. Total export value for this period (Ref: DEL 162A 1b) R 200

b. Less: Exemption producers (Ref: DEL162B 1A 2b) R 468

c. Less: Exemption Beneficiary (Ref: DEL162B 1Bb) R 78

d. Less: Election (Producer / Beneficiary) (Ref: DEL162C 1b) R 53

e. Less: Temporary Exports (Ref: DEL162D 1b) R 611

f. Less: Temporary Imports (Ref: DEL162E 1b) R 134

g. Imports credits (Ref: DEL162F 1b) R 877

h. Total: Nett value R 344

i. Nett: Levy payable (Ref: DEL162 2h x 5%) R 17.20

Page 1

DEL 162

XML Source

XML maps in this workbook:

SARS_DEL162_REQUEST_Map

ns1:SARS_DEL162_REQUEST

- ns1:Registrant
 - ns1:ClientCode
 - ns1:ClientType
 - ns1:AssessmentPeriodFrom
 - ns1:AssessmentPeriodTo
 - ns1:RegisteredName
 - ns1:TradingName
- ns1:Address
 - ns1:AddressLine1
 - ns1:AddressLine2
 - ns1:AddressLine3
 - ns1:Postcode
- ns1:LeviableExportTransactions
 - ns1:TotalExportValueForPeriod
 - ns1:LessExceptionProducers
 - ns1:LessExceptionBeneficiary
 - ns1:LessElection
 - ns1:LessTemporaryExports
 - ns1:LessTemporaryImports
- ns1:ImportCredits
 - ns1:PDBCredits
 - ns1:TotalNetValue
 - ns1:NetLevyPayable

To map non-repeating elements, drag the elements from the tree onto the worksheet where you want the data to appear.

To import data, use the Import XML Data button on the List toolbar.

Options XML Maps...

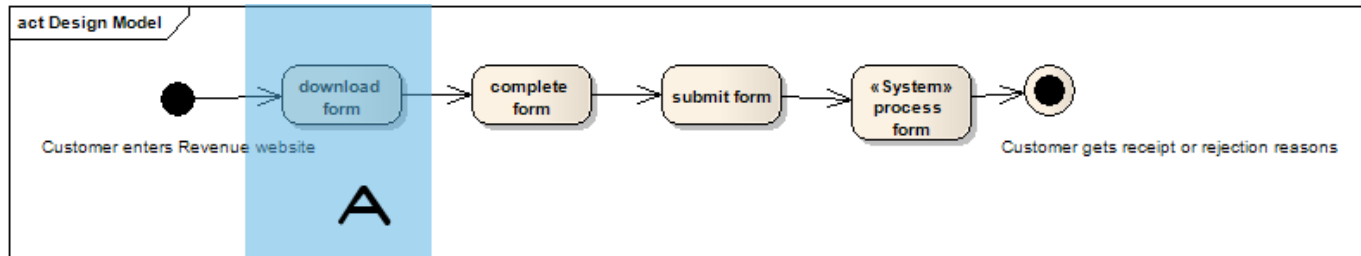
Verify Map for Export...

Tips for mapping XML

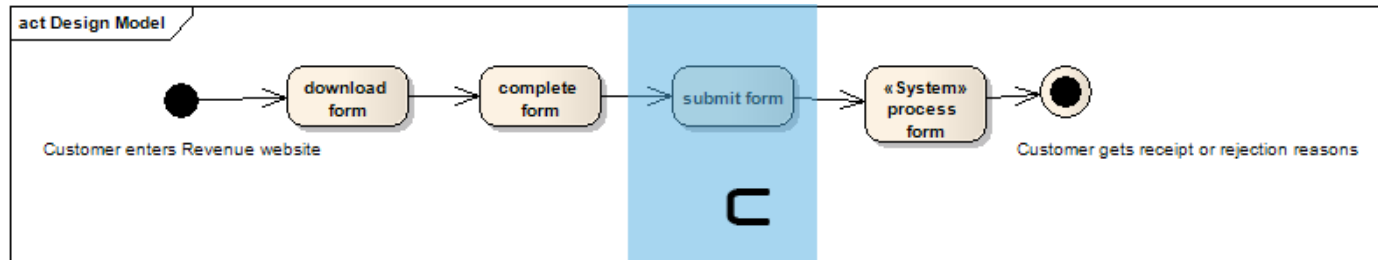
Ready

Con... Inbo... C:\U... Edit... Alto... IBM ... Ima... H\... 01... dp p... sma... Mic...

19:20

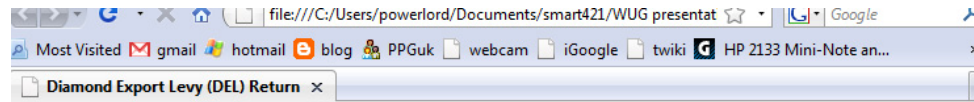


- DataPower has a built-in HTTP server
- A static webpage can be served to user allowing them to download MS Excel spreadsheet directly from XI50



Submit can be broken into 2 parts:

- Static webpage allowing the user to enter their username and password and upload tax return (XML file generated from spreadsheet)
- The actual form submission function



DIAMOND EXPORT LEVY

DEL 162

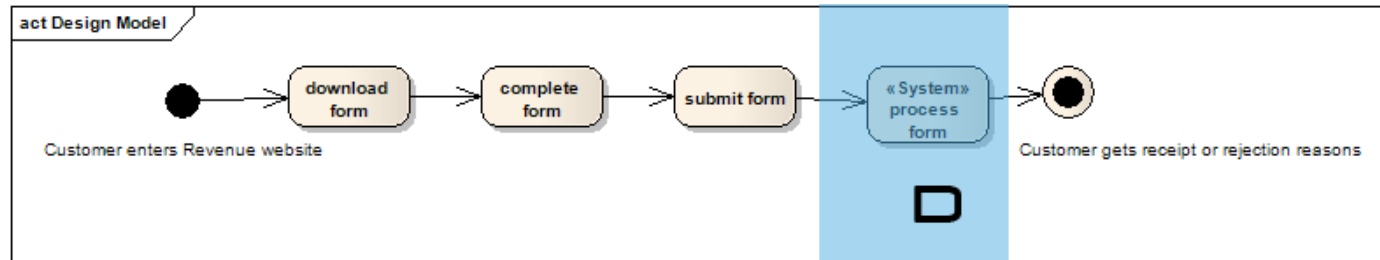
Diamond export levy (DEL) return

Stage 1

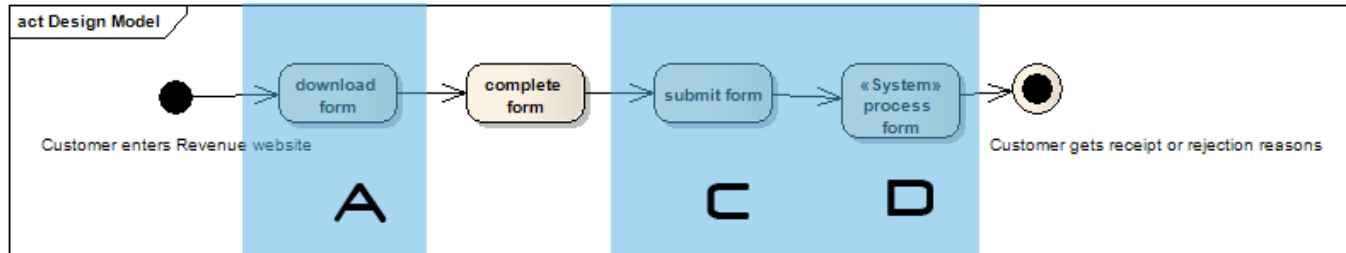
In order to file your DEL162 return, please [click here to download this Microsoft EXCEL Spreadsheet](#) to your local disk.

[I have completed Stage 1. Take me to Stage 2](#)

Done



- Regular web-service. Accepts XML request, validates and then transforms and forwards to back-end system (in correct format)
- Should be extendible to cover all other forms in future
- Supports B2B aspirations
- Can use standard WS-SECURITY functionality
- Can use DataPower XSLT extensions to generate reference number
- MQ to backend for reliable transmission
- XSLT to transform to back-end format

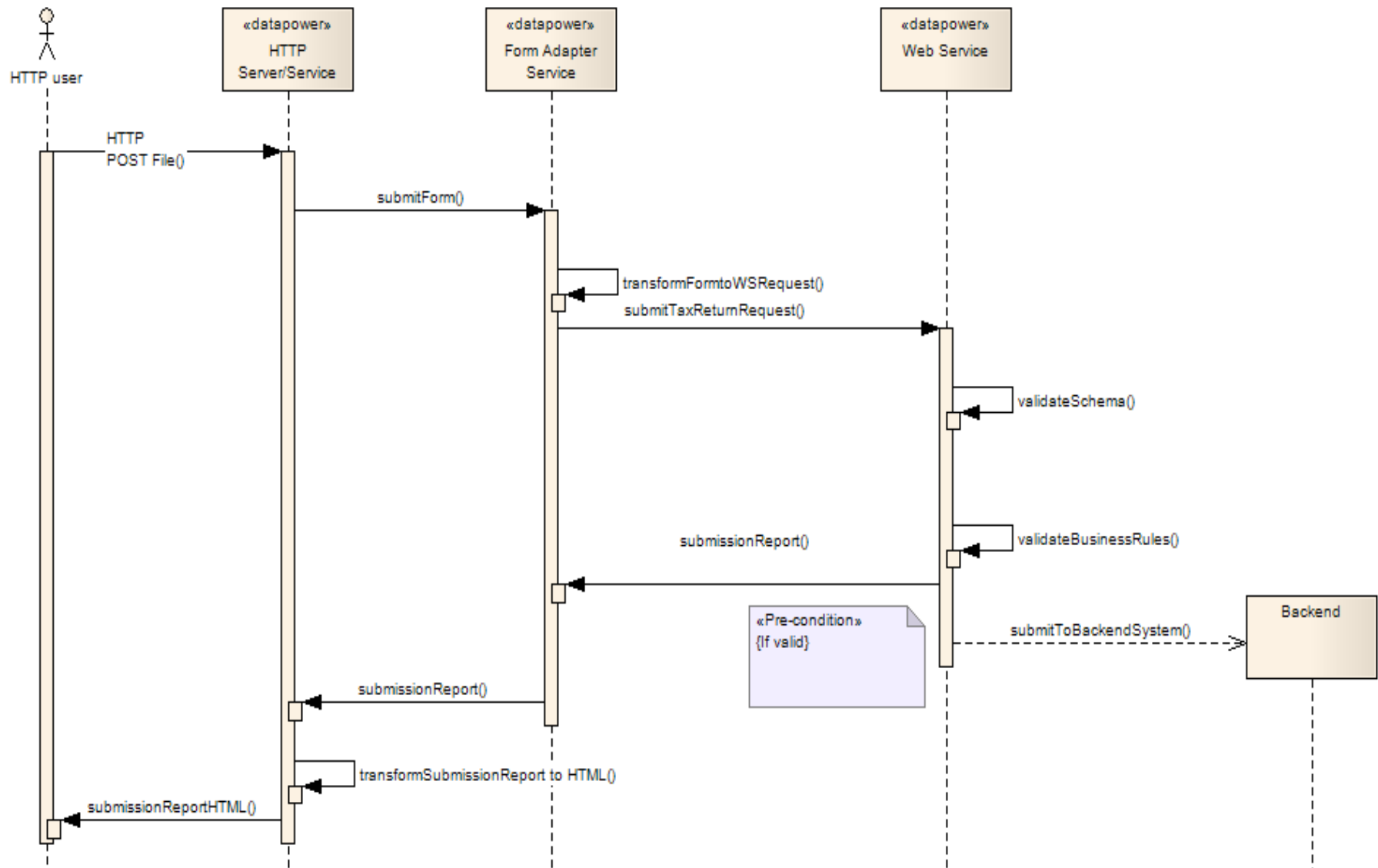


If DataPower can accept a form submission (file upload), we can do stages A, C and D in the XI50 !

- This can be seen as an adapter pattern – sitting in front of the web-service implementation.
- Its job is to accept a form submission and process this into a web-services request
- Send this request to the web-service (stage D)
- Accept the web-service response and process it into a human friendly webpage to present back to the user.

Sequence Diagram

sd SARS - HTTP File submission



Multi-step Walkthrough

DataPower XI50 - Processing Policy: - Mozilla Firefox

https://172.29.129.45:9090/?skipNav=true&action=show&screen=StylePolicyEditor&policyNameSelect=processForm2&service=XMLFirewalService

The running configuration of the domain contains unsaved changes. [Review changes.](#)
Troubleshooting Enabled (The performance of the device may be impacted!)

Policy:

Policy Name: *

[Export](#) | [View Log](#) | [View Status](#) | [Close Window](#)

Rule:

Rule Name: Rule Direction:

Create rule: Click New, drag action icons onto line. Edit rule: Click on rule, double-click on action

Filter Sign Verify Validate Encrypt Decrypt Transform Route AAA Results Advanced

CLIENT → [Action Icons] → ORIGIN SERVER

Configured Rules

Order	Rule Name	Direction	Actions	
↑ ↓	processForm2_rule_0	Client to Server	[Action Icons]	delete rule
↑ ↓	processForm2_rule_1	Error	[Action Icons]	delete rule

[Scroll to top](#)

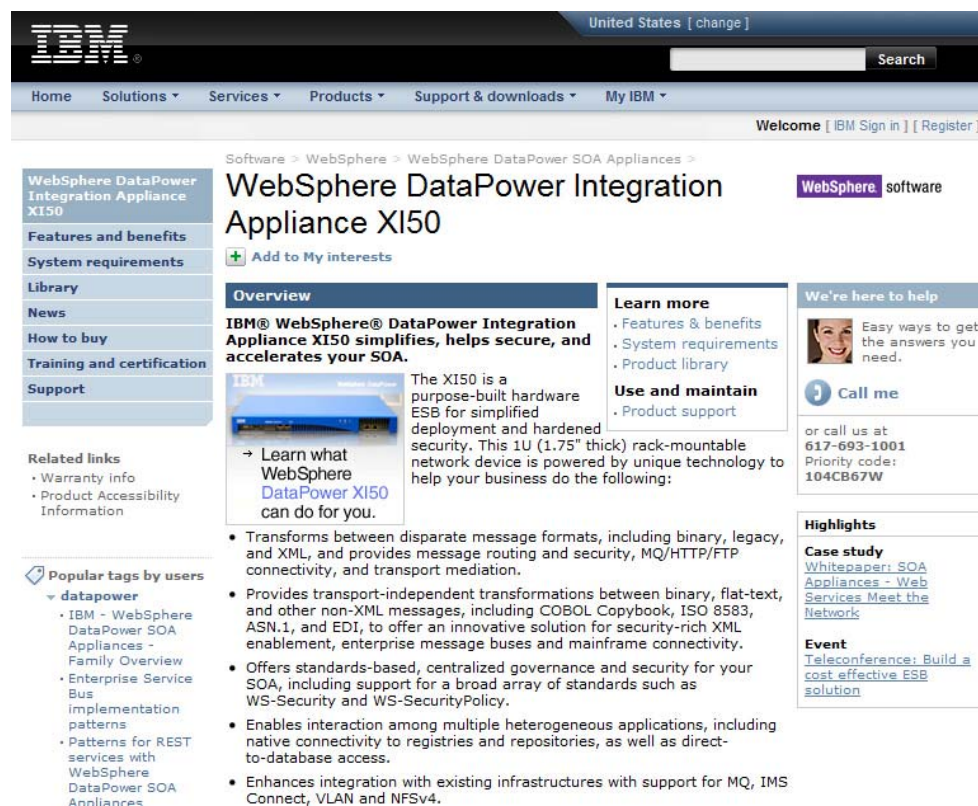
Done 172.29.129.45:9090

- Error Handling via DataPower variables
- Parsing into user friendly strings
- Limitation of XSD validation: only get first error

- Complex validation possible
- Document parsed for all errors
- DataPower variables used to store errors for later retrieval
- Error handler pattern (with errors defined in XML backing file)

- Most of functionality is concentrated in the actual web-service
- Regular web-service tools can be used for testing
- Can be as simple as a series of batch files calling curl commands with input messages
- CI can be built by building unit test manager as an XSLT, and running this in its own service. This can then be called by DataPower timer and alter if/when build breaks.

<http://www-01.ibm.com/software/integration/datapower/xi50/>



The screenshot shows the IBM website for the WebSphere DataPower Integration Appliance XI50. The page layout includes a top navigation bar with the IBM logo, a search bar, and links for Home, Solutions, Services, Products, Support & downloads, and My IBM. A secondary navigation bar shows the user is in the United States and provides links for Welcome, IBM Sign in, and Register.

The main content area features a breadcrumb trail: Software > WebSphere > WebSphere DataPower SOA Appliances > WebSphere DataPower Integration Appliance XI50. The product title is prominently displayed, followed by a 'WebSphere software' badge. A sidebar on the left lists various sections: Features and benefits, System requirements, Library, News, How to buy, Training and certification, and Support. Below this, 'Related links' include Warranty info and Product Accessibility Information. A 'Popular tags by users' section lists tags like 'datapower', 'IBM - WebSphere DataPower SOA Appliances - Family Overview', 'Enterprise Service Bus implementation patterns', and 'Patterns for REST services with WebSphere DataPower SOA Appliances'.

The main content area includes an 'Overview' section with a sub-header: 'IBM® WebSphere® DataPower Integration Appliance XI50 simplifies, helps secure, and accelerates your SOA.' This is followed by a description of the XI50 as a purpose-built hardware ESB for simplified deployment and hardened security. A small image of the appliance is shown. To the right, a 'Learn more' section lists links for Features & benefits, System requirements, and Product library. Below this, a 'Use and maintain' section lists a link for Product support.

On the right side of the page, there is a 'We're here to help' section with a 'Call me' button and contact information: 'or call us at 617-693-1001', 'Priority code: 104CB67W'. Below this, a 'Highlights' section includes a 'Case study' link for 'Whitepaper: SOA Appliances - Web Services Meet the Network' and an 'Event' link for 'Teleconference: Build a cost effective ESB solution'.

At the bottom of the page, the email address 'ssmith@smart421.com' is displayed.

ssmith@smart421.com