



New York State Experience with Corporation Tax Modernized e-File

James Lieb, Director – Common Services
Chris Chiesa, Manager – Corp Tax e-File
NYS Department of Taxation and Finance



Business Problem

- Did not have a Corporation Tax e-file program for NY returns
- Did not have a way for software developers to transmit directly to DTF
 - Our direct file program for PIT had come into our processing partner, not DTF. That program was discontinued in 2004 when we converted completely to a Fed/State model for PIT.
- IRS CT e-file program and mandate on certain corporations created a demand for NY to implement an e-file program for those corporations.
- IRS offered a Fed/State Corp Tax program under the MeF platform, but we were not experienced with the technology.

Business and Technical Challenges

- No experience in accepting or parsing complex XML
- Headers, footers and tracking numbers had to be created for the internal processing of the submissions
- We needed to display the data to our business users in a way they could understand
- DTF never stored or accessed XML
- IRS MeF framework new to us
- Although we had participated in TIGERS since the inception of the project, our development work did not begin until September of 2006 for a January 2007 implementation

3

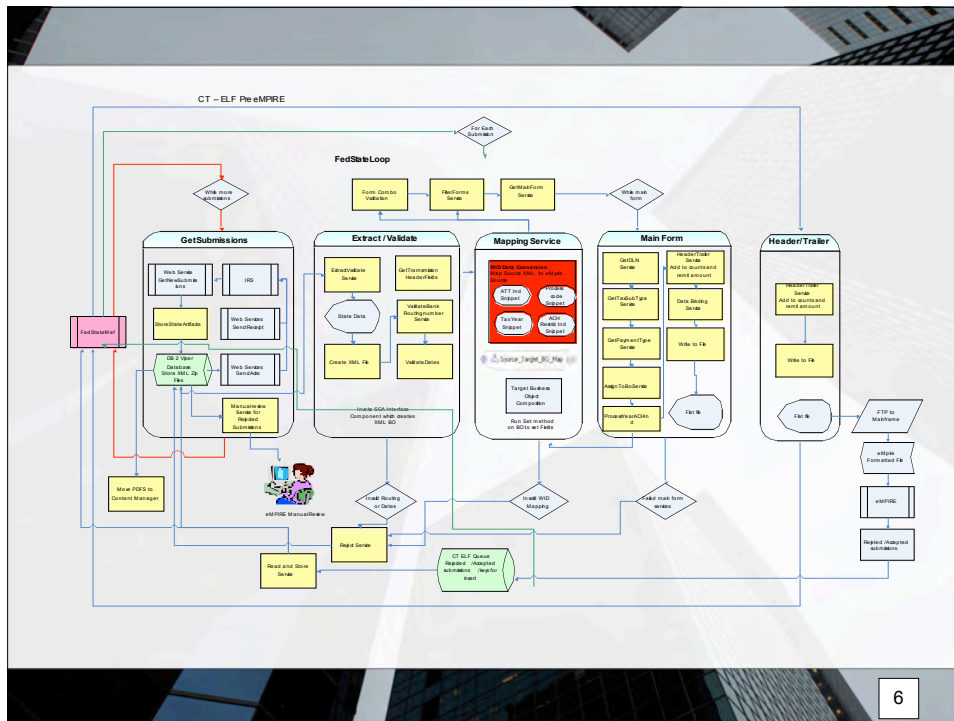
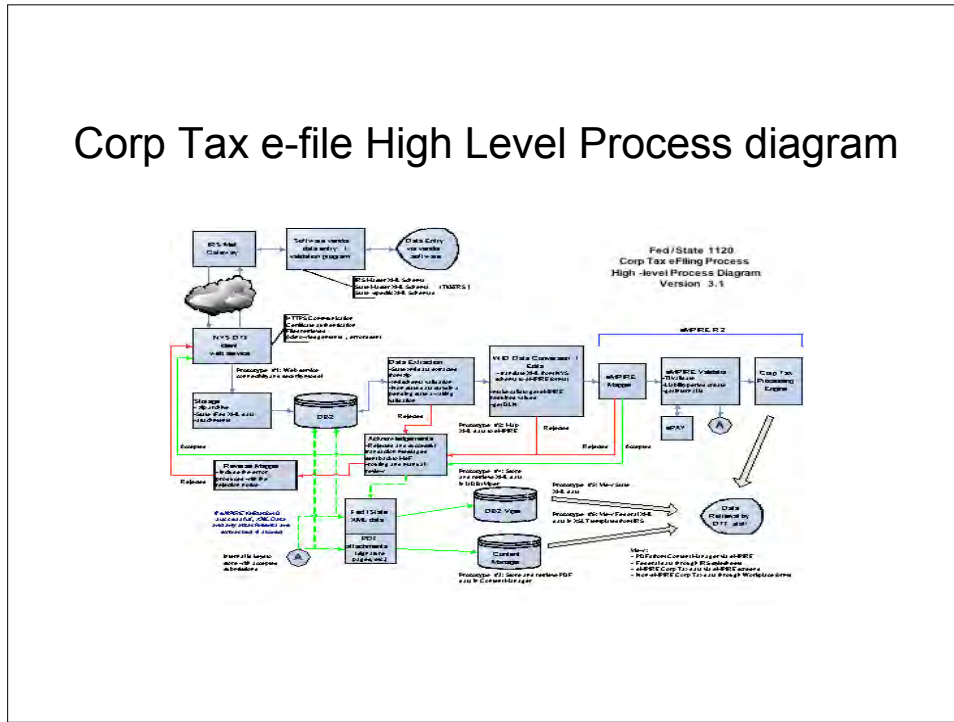
SOA Concepts and Techniques

A Development View

- Top-Down Design
- Focus on process and integration
- Leverage tools and products
- Use industry standards (e.g. XML) to increase capabilities of tools
- Develop standard design and integration patterns to facilitate future projects

4

Corp Tax e-file High Level Process diagram



XML Forms Mapping

```

<USAddress>
- <Address>
  <AddressLine1> PO BOX 228 </AddressLine1>
  <City> SCHENECTADY </City>
  <State> NY </State>
  <ZIPCode> 123060000 </ZIPCode>
</USAddress>
- </Address>
</USAddress>
</StateOfIncorporation>
<HdrCode>
- <FederalReturnFiledOther>String</FederalReturnFiledOther>
  <FilerClassificationCode>AA3</FilerClassificationCode>
  <FormType>CT5</FormType>
  <ReturnTypeCode>CT5</ReturnTypeCode>
  <SoftwareDeveloper>
    <DeveloperName>
<BusinessNameLine1>Sunrise Investments Inc</BusinessNameLine1>
<BusinessNameLine2>A A</BusinessNameLine2>

```

9

Project Completion Information

- In Production Jan. 2007 (on time)
- Over 2,000 data elements mapped. (only 6 data issues found in production)
- Of 34,000 federal returns, NYS processed 13,500
- 14 DTF ELF eforms were in production day one
- Used software (Process Server, DB2 V9, Workplace Forms) all evaluated after June 2006 and used for the first time in production

10

What We Did Right!

- TIGERS participation
- Involve business users in the development process
 - Not just technology driven
- Prototyping
 - Theories tested with simple working prototypes prior to development
- Communication with Software Developers
- Minimize impact on end users

11

Questions?

Kansas Modernized e-File Development

Planning and Implementation
Statistics
Future Development
.Net Technical Walk-through



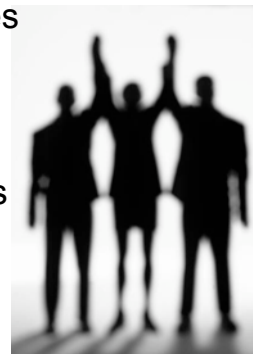
Terry Hunt, e-File Coordinator

Andrew Sheley, Application Developer

Planning & Implementation



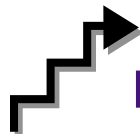
- Agency-wide support of MeF initiatives
 - IT commitment of time and resources
 - Developed project plans
 - Business and Technical staff teamwork
- Published state schema and business rules
- Partnered with software developer



Planning & Implementation



- 14 Forms and schedules included in launch of Kansas Corporate e-File
- Total development time, including backend system changes – approx. 1 year
- Lessons Learned:
 - Need additional software developers to support program
 - XML parse errors are difficult for practitioners to read and understand
 - Practitioners struggle with how to scan documents, create and attach pdf documents

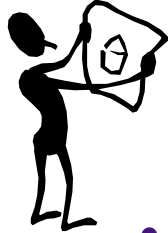


Kansas MeF Stats



- 3,993 Submissions received through MeF
 - 3,170 Corporate, 494 Partnership, 329 failed schema validation
- 3 Software developers: CCH, Creative Solutions, Drake
- Kansas and Wisconsin first states to “go-live” with Corporate MeF in June 2006
- Kansas and Maryland first states to “go-live” with Partnership MeF in January 2007

Future Development



- Moving away from “category-based” data structure to a “forms-based” approach
 - Will be easier for software developers to code
 - Will improve parse error readability
 - Reduce development time for year-end updates
- MeF 1040 conversion
 - Starting development work now
 - Opportunity to capture more data, more forms

Technical Walk-through

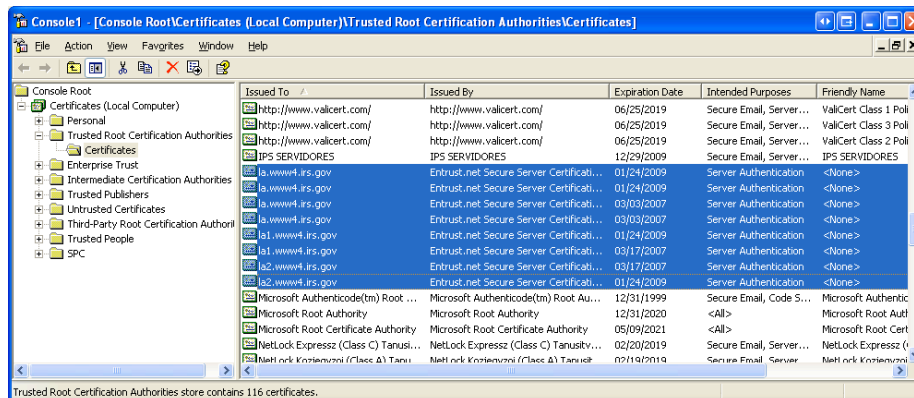


- Web Services Development
- Schema Development
- Parsing program
- Lessons Learned

Web Services Development



- Load trusted root certificates from IRS onto Server through MMC (Microsoft Management Console). Import certificates into the Trusted Root Certification Authorities folder.



Web Services Development



- Create Client Proxy classes from IRS supplied WSDL files.
 - Use .Net utility to create proxy classes. These tell the App how to communicate with the MeF web service.
 - A few modifications need to be made. The class to inherit, and URL of the MeF web service.
 - WSDL and Proxy Code example next slides.

Web Services Development



```

C:\Temp\wsdl\MeFMSIServices_SeparateServices.wsdl
File Edit View Favorites Tools Help
Back Forward Stop Search Favorites
Address C:\Temp\wsdl\MeFMSIServices_SeparateServices.wsdl
To help protect your security, Internet Explorer has restricted this file from showing active content that could access your computer. Click here for options...
</wsdl:input>
- <wsdl:output>
  <soap:body parts="LogoutResponse" use="literal" />
  <soap:header message="tns:LogoutResponseMessage" part="MeFHeader" use="literal" />
</wsdl:output>
- <wsdl:fault name="fault">
  <soap:fault name="fault" use="literal" />
</wsdl:fault>
</wsdl:operation>
</wsdl:binding>
- <wsdl:binding name="Login" type="tns:Login">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
- <wsdl:operation name="Login">
  <soap:operation soapAction="http://www.irs.gov/a2a/mef/Login" />
- <wsdl:input>
  <soap:body parts="Login" use="literal" />
  <soap:header message="tns:LoginRequestMessage" part="MeFHeader" use="literal" />
</wsdl:input>
- <wsdl:output>
  <soap:body parts="LoginResponse" use="literal" />
  <soap:header message="tns:LoginResponseMessage" part="MeFHeader" use="literal" />
</wsdl:output>
- <wsdl:fault name="fault">
  <soap:fault name="fault" use="literal" />
</wsdl:fault>
</wsdl:operation>
</wsdl:binding>
- <wsdl:service name="ChangePassword">
  <wsdl:port name="ChangePassword" binding="tns:ChangePassword">
    <soap:address location="http://www.irs.gov/a2a/mef/ChangePassword" />
  </wsdl:port>
</wsdl:service>
- <wsdl:service name="EtinRetrieval">
  <wsdl:port name="EtinRetrieval" binding="tns:EtinRetrieval">

```

Web Services Development



```

End Class
'<remarks/>
<System.Diagnostics.DebuggerStepThroughAttribute(),
System.ComponentModel.DesignerCategoryAttribute("code"),
System.Web.Services.WebServiceBindingAttribute(Name="Login", [Namespace]="http://www.irs.gov/a2a/mef/MeFMSIServices")>
Public Class Login
    Inherits WebServicesClientProtocol
    'Replace this ***** Inherits System.Web.Services.Protocols.SoapHttpClientProtocol

    Public MeF As MeFHeaderType

    'remarks/>
    Public Sub New()
        MyBase.New
        'Replace this ***** Me.Url = "http://www.irs.gov/a2a/mef/Login"
        Me.Url = "https://la.vvv4.irs.gov/a2a/mef/Login/"
    End Sub

    'remarks/>
    <System.Web.Services.Protocols.SoapHeaderAttribute("MeF", Direction:=System.Web.Services.Protocols.SoapHeaderDirection.InOut),
    System.Web.Services.Protocols.SoapDocumentMethodAttribute("http://www.irs.gov/a2a/mef/Login", Use:=System.Web.Services.Descript
    Public Function Login(<System.Xml.Serialization.XmlElementAttribute([Namespace]="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd"
    Dim results() As Object = Me.Invoke("Login", New Object() {LoginRequest})
    Return CType(results(0), LoginResponseType)
End Function

'<remarks/>
Public Function BeginLogin(ByVal LoginRequest As LoginRequestType, ByVal callback As System.AsyncCallback, ByVal asyncState As O
Return Me.BeginInvoke("Login", New Object() {LoginRequest}, callback, asyncState)
End Function

'<remarks/>
Public Function EndLogin(ByVal asyncResult As System.IAsyncResult) As LoginResponseType
Dim results() As Object = Me.EndInvoke(asyncResult)
Return CType(results(0), LoginResponseType)
End Function
End Class

'<remarks/>
<System.Xml.Serialization.XmlTypeAttribute([Namespace]="http://www.irs.gov/a2a/mef/MeFHeader.xsd"),
System.Xml.Serialization.XmlRootAttribute("MeF", [Namespace]="http://www.irs.gov/a2a/mef/MeFHeader.xsd", IsNullable:=false)>
Public Class MeFHeaderType
    Inherits System.Web.Services.Protocols.SoapHeader
    'remarks/>
    Public MessageID As String

```

Web Services Development



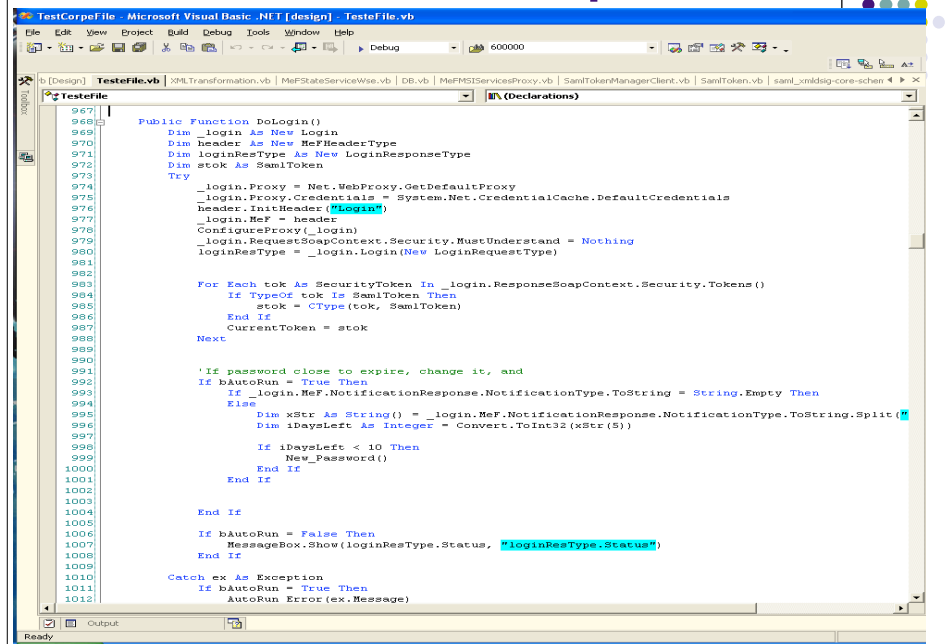
- Create .Net Application to connect to the MeF web service.
 - Include WSE (Web Service Enhancements) in the project. This allows for extended methods and properties for working with web services.
 - Set up Security in the project to use SAML security tokens. Create SAML security token class.
 - Add proxy classes to the application to make the function calls to the MeF web service (i.e. Login, GetNewSubmissions, etc.)
- See code example next slide.

Web Services Development



```
MeFMSIServicesClient - Microsoft Visual Basic .NET [design] - SamToken.vb
File Edit View Project Build Debug Tools Window Help
Debug 600000
Start Page TestFile.vb [Design] TestFile.vb XMLTransformation.vb MeFStateServiceWeb.vb DB.vb MeFMSIServicesProxy.vb SamTokenManagerClient.vb SamToken.vb
[General] [Declarations]
43 MyAssertion = New AssertionType
44 Dim utcTime As String = Format(currentTime, "yyyy-MM-dd") & "T" & Format(currentTime, "HH:mm:ss") &
45
46 MyAssertion.AssertionID = "d-8d9020be2c090a0ef1481e6d2cbddf56a66f9ad"
47 MyAssertion.MajorVersion = "1.0"
48 MyAssertion.MinorVersion = "1.0"
49 MyAssertion.Issuer = "http://www.forkumvz.com/sentry"
50 MyAssertion.IssueInstant = utcTime
51
52 Dim cond As New ConditionsType
53 cond.NotBeforeSpecified = True
54 cond.NotBefore = utcTime
55 cond.NotOnOrAfterSpecified = True
56 cond.NotOnOrAfter = Format(currentTime.AddYears(1), "yyyy-MM-dd") & "T" & Format(currentTime.AddYear
57 MyAssertion.Conditions = cond
58
59
60 Dim authst As New AuthenticationStatementType
61
62 MyAssertion.Items = Array.CreateInstance(GetType(StatementAbstractType), 2)
63 MyAssertion.Items(0) = authst
64
65 authst.AuthenticationInstant = utcTime
66 authst.AuthenticationMethod = "urn:oasis:names:tc:SAML:1.0:am:password"
67
68 Dim subject As New SubjectType
69 authst.Subject = subject
70
71
72 Dim nameID As New NameIdentifierType
73
74 subject.NameIdentifier = nameID
75 nameID.Format = "urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName"
76 nameID.Value = "UID=" & userID & ";OID=" & _OID
77
78 Dim subjconf As New SubjectConfirmationType
79 subject.SubjectConfirmation = subjconf
80 subjconf.ConfirmationMethod = Array.CreateInstance(GetType(String), 1)
81 subjconf.ConfirmationMethod(0) = "urn:oasis:names:tc:SAML:1.0:cm:sender-vouches"
82
83 Dim attst As New AttributeStatementType
84 MyAssertion.Items(1) = attst
85 attst.Subject = subject
86
87 Dim att As New AttributeType
88 attst.Attribute = Array.CreateInstance(GetType(AttributeType), 1)
89 attst.Attribute(0) = att
```

Web Services Development



```
967 Public Function Dologin()  
968 Dim _login As New Login  
969 Dim header As New MeFHeaderType  
970 Dim loginResType As New LoginResponseType  
971 Dim stok As SAMLToken  
972 Try  
973     _login.Proxy = Net.WebProxy.GetDefaultProxy  
974     _login.Proxy.Credentials = System.Net.CredentialCache.DefaultCredentials  
975     header.InitHeader("MeFLogin")  
976     _login.MeF = header  
977     ConfigureProxy(_login)  
978     _login.RequestSoapContext.Security.MustUnderstand = Nothing  
979     loginResType = _login.Login(New LoginRequestType)  
980  
981     For Each tok As SecurityToken In _login.ResponseSoapContext.Security.Tokens()  
982         If TypeOf tok Is SAMLToken Then  
983             stok = CType(tok, SAMLToken)  
984         End If  
985     Next  
986     CurrentToken = stok  
987  
988     'If password close to expire, change it, and  
989     If bAutoRun = True Then  
990         If _login.MeF.NotificationResponse.NotificationType.ToString = String.Empty Then  
991             Dim xStr As String() = _login.MeF.NotificationResponse.NotificationType.ToString.Split(" ")  
992             Dim iDaysLeft As Integer = Convert.ToInt32(xStr(5))  
993             If iDaysLeft < 10 Then  
994                 New_Password()  
995             End If  
996         End If  
997     End If  
998  
999     If bAutoRun = False Then  
1000         MessageBox.Show(loginResType.Status, "LoginResType.Status")  
1001     End If  
1002  
1003     Catch ex As Exception  
1004         If bAutoRun = True Then  
1005             AutoRun_Error(ex.Message)  
1006         End If  
1007     End Try  
1008 End Function
```

Web Services Development

- In your application create calls to the proxy functions to perform the necessary actions to retrieve, and acknowledge returns.
- Typical sequence of events:
 - ***Initial Login – 1 time only to establish account and new password.
 - Login – Login to MeF site using user id and password. A session is established and SAML token is returned. This is used for subsequent transmissions to the MeF site.
 - GetNewSubmissions – Get Submissions from MeF for your state. Set maximum results. Returns are received as DIME attachment - *.ZIP file.
 - SendSubmissionReceipts – Receipt XML is created for each return received – just says we received the return. Stored in ZIP file and added as DIME attachment and sent back to MeF web service.
 - Process the XML return data. Covered in XML parsing slides.
 - SendAcks – After Returns are processed, generate XML acknowledgements. This says if the state accepted or rejected the Return. If its an error, error reasons are returned in the acknowledgment. Also stored in ZIP file and added as DIME attachment and sent to MeF web service.
 - Logout – End session with MeF web service.

Schema Development



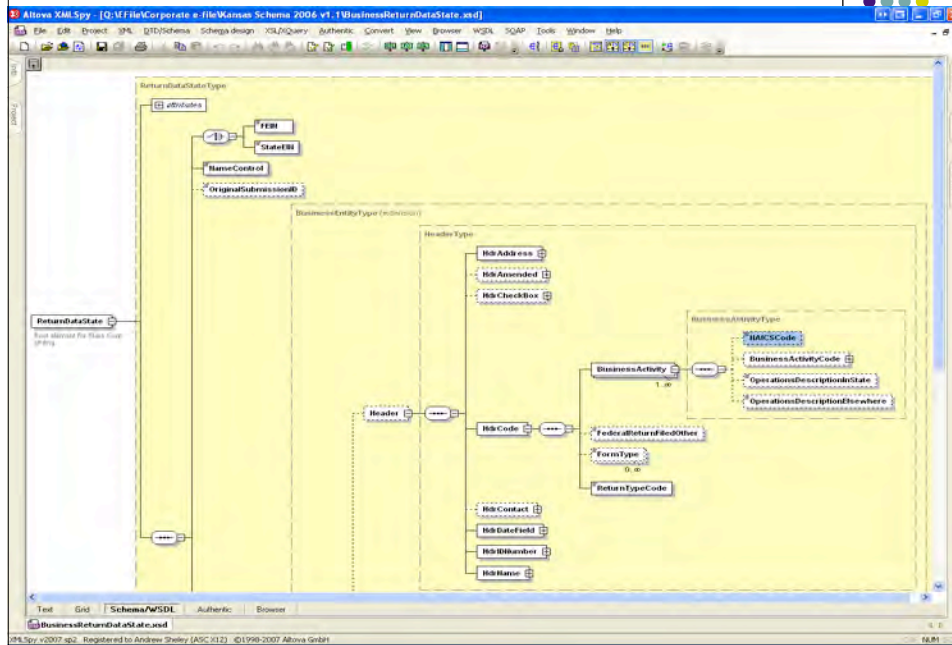
- Category based. Will move to Forms based eventually if passed by TIGERS.
- Derive State schema from Master schema based on the Data Dictionary – Spreadsheet that assigns Tax Form line items to XML elements.
- We use a program to pull only state elements from Master schema.
- States can restrict State schema from Master, but not expand.
- Build sample XML file from State schema using XML spy and validate all elements of Data Dictionary exist in the XML file created from the State schema. Some may not validate – choice elements.
- Make sure sample XML file validates back to Master schema.

Schema Development



Form #	Line #	Field Name	Schema Category	Xpath
K-120	Header	Method used to determine Income of Corporation in Kansas	General Information	/Return/State/BusinessEntity/GeneralInformation/BusinessInfo/MethodToDetermineIncome/Year
K-120	Header	Business Activity Code (NAICS)	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Code/BusinessActivity/NAICS/Code
K-120	Header	Date Business Began in Kansas	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Date/Field/Date/Began/In/State
K-120	Header	Date Business Discontinued in Kansas	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Date/Field/Date/Ended/In/State
K-120	Header	State of Incorporation	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Address/State/Of/Incorporation/US/Address/State
K-120	Header	Month and Year of Incorporation	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Date/Field/Date/Of/Incorporation
K-120	Header	State of Commercial Domicile	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Address/State/Of/Domicile/US/Address/State
K-120	Header	Type of Federal Return Filed	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Code/Federal/Return/Filed/Other
K-120	Header	Kansas Form K-120EL Indicator	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Check/Box/Schedule/Attached
K-120	Header	EIN of this Entity	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/ID/Number/FEIN
K-120	Header	EIN Federal Consolidated Parent	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/ID/Number/Consolidated/Parent/FEIN
K-120	Header	Original Federal due date	General Information	/Return/State/BusinessEntity/GeneralInformation/Fed/Return/Info/Other/Due/Date/Federal
K-120	Header	Taxpayer information has changed since last return was filed	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Check/Box/Name/Or/Address/Change
K-120	Header	Amended Return Indicator	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Check/Box/Amended/Return/Indicator
K-120	Header	Amended Return Reason	Header .xsd	/Return/State/BusinessEntity/Header/Hdr/Amended/Amended/Reason/Code
K-120	1	Federal taxable income	Income .xsd	/Return/State/BusinessEntity/Income/Inc/Fed/Tax/Inc/Federal/Taxable/Income
K-120	2	Total state and municipal interest	Additions .xsd	/Return/State/BusinessEntity/Additions/Add/Adjustments/Interest/State/And/Municipal
K-120	3	Taxes on or measured by income or fees or payments in lieu of in	Additions .xsd	/Return/State/BusinessEntity/Additions/Add/Taxes/Taxes/Based/On/Income
K-120	4	Federal net operating loss deduction	Additions .xsd	/Return/State/BusinessEntity/Additions/Add/Adjustments/Federal/Net/Operating/Loss
K-120	5	Other additions to federal taxable income	Additions .xsd	/Return/State/BusinessEntity/Additions/Add/Miscellaneous/Other/Additions/Amount
K-120	6	Total additions to federal taxable income	Additions .xsd	/Return/State/BusinessEntity/Additions/Add/Totals/Total/Additions
K-120	7	Interest on U.S. government obligations	Subtractions .xsd	/Return/State/BusinessEntity/Subtractions/Sub/Deductions/Interest/US/Obligations
K-120	8	IRC Section 78 and 80% of foreign dividends	Subtractions .xsd	/Return/State/BusinessEntity/Subtractions/Sub/Deductions/IRC/Section/78/and/80/Percent
K-120	9	Other subtractions from federal taxable income	Subtractions .xsd	/Return/State/BusinessEntity/Subtractions/Sub/Other/Other/Subtractions
K-120	10	Total subtractions from federal taxable income	Subtractions .xsd	/Return/State/BusinessEntity/Subtractions/Sub/Total/Total/Subtractions
K-120	11	Net income before apportionment	Tax Due Overpayment	/Return/State/BusinessEntity/Tax/Due/Overpayment/Tax/Income/Inc/After/Reconciliation/Adjust
K-120	12	Nonbusiness income - Total Company	Income .xsd	/Return/State/BusinessEntity/Income/Inc/Non/Bus/Inc/Nonbusiness/Income/Total
K-120	13	Apportionable business income	Tax Due Overpayment	/Return/State/BusinessEntity/Tax/Due/Overpayment/Tax/Income/Inc/Apportionable/Income
K-120	14a	Average percent to Kansas	Allocation And App	/Return/State/BusinessEntity/Allocation/And/Apportionment/State/All/All/Property/Total/Property/Ratio
K-120	14b	Average percent to Kansas	Allocation And App	/Return/State/BusinessEntity/Allocation/And/Apportionment/State/All/All/Payroll/Pay/Payroll/Ratio
K-120	14c	Average percent to Kansas	Allocation And App	/Return/State/BusinessEntity/Allocation/And/Apportionment/State/All/All/Sales/Sales/Fraction/Ratio
K-120	14	Average percent to Kansas	Tax Due Overpayment	/Return/State/BusinessEntity/Tax/Due/Overpayment/Tax/Apportionment/State/Apportionment/Factor
K-120	15	Amount to Kansas	Tax Due Overpayment	/Return/State/BusinessEntity/Tax/Due/Overpayment/Tax/Income/Inc/Amount/State
K-120	16	Nonbusiness income - Kansas	Income .xsd	/Return/State/BusinessEntity/Income/Inc/Non/Bus/Inc/Nonbusiness/Income/State
K-120	17	Kansas net income before NOL deduction	Tax Due Overpayment	/Return/State/BusinessEntity/Tax/Due/Overpayment/Tax/Income/Inc/Net/Income/Before/NOL
K-120	18	Kansas net operating loss deduction	Deductions .xsd	/Return/State/BusinessEntity/Deductions/Ded/NOL/Net/Loss/Preceding/Taxable/NOL/Loss

Schema Development



Parsing/XML Processing

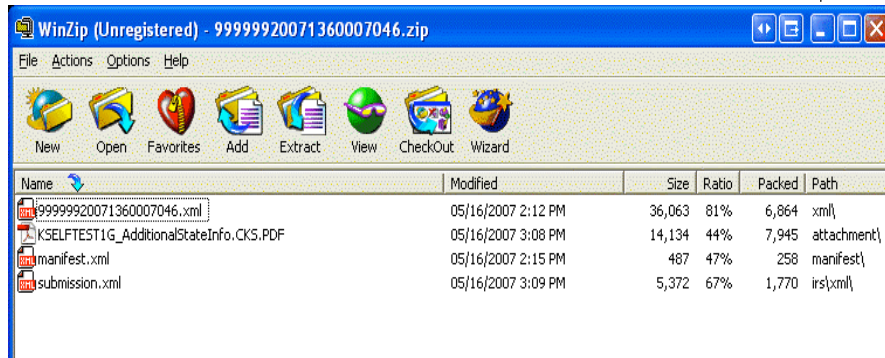


- Basic steps for processing XML return.
 - Validate XML return against state schema.
 - Parse and retrieve data elements from XML document using .Net framework classes.
 - Generate text/flat file for passing to our back end tax system.
 - Create Acknowledgement data.

Parsing/XML Processing



- Zip file structure.



Parsing/XML Processing



- Validate XML return against state schema.

```
Public Class XML_Validation
    Public bErr As Boolean
    Public errarr As New Collections.ArrayList

    Public Function Validate_XML(ByVal strFile As String, ByVal sTaxYear As String) As Collections.Array

        Dim tr As New XmlTextReader(strFile)
        Dim vr As New XmlValidatingReader(tr)

        bErr = True

        AddHandler vr.ValidationEventHandler, AddressOf ValidationErrorHandler

        Try
            Select Case sTaxYear
                Case "2005"
                    vr.Schemas.Add("http://www.irs.gov/efile", Configuration.ConfigurationSettings.AppSe)
                Case "2006"
                    vr.Schemas.Add("http://www.irs.gov/efile", Configuration.ConfigurationSettings.AppSe)
            End Select

            vr.ValidationType = ValidationType.Schema

            While vr.Read()
                'Parse Through Elements
            End While

            Catch e As Exception
                'Error Encountered, set message, close, and return false
                errarr.Add(String.Format("Error: (0)", e.Message.Trim))
                bErr = False
                Return errarr
            Finally
                vr.Close()
            End Try
    End Function
End Class
```

Parsing/XML Processing



- Parse and retrieve data elements from XML document using .Net framework classes.
 - Retrieve data from the XML document using its Xpath via .Net framework XML classes.
 - Build text/flat file with data retrieved for passing on to back end tax system.
 - See next slides that show typical XML state return, and programming to retrieve data from it via Xpaths.

Parsing/XML Processing



```
Altova XMLSpy - [C:\apps\CorpEfile\PDFZips\SubmissionZip070702112241\58
File Edit Project XML DTD/Schema Schema design XSL/XQuery Authentic Conv
1 <?xml version="1.0"?>
2 <ReturnDataState StateBusinessVersionNumber="2006v1.1" xmlns="http://www.irs.g
3 <FEIN>110000002</FEIN>
4 <NameControl>HIDE</NameControl>
5 <BusinessEntity>
6 <Header>
7 <HdrAddress>
35 <HdrCheckBox>
38 <HdrCode>
39 <BusinessActivity>
40 <NAICSCode>111900</NAICSCode>
41 </BusinessActivity>
42 <FederalReturnFiledOther>1</FederalReturnFiledOther>
43 <ReturnActionCode>K120</ReturnActionCode>
44 </HdrCode>
45 <HdrContact>
46 <ContactPerson>
59 <CorporateOfficer>
60 <Title>Chief Executive Of</Title>
61 <FirstName>DOUG</FirstName>
62 <LastName>DOE</LastName>
63 </CorporateOfficer>
64 <Preparer>
65 <PreparerName>Johnny Appleseed</PreparerName>
66 <Address>
67 <USAddress>
68 <AddressLine1>100 Efile Drive</AddressLine1>
69 <City>Anytown</City>
70 <State>LA</State>
```

Parsing/XML Processing



```
MeFStateServiceWise.vb | DB.vb | MeFMSIServicesProxy.vb | SamlTokenManagerClient.vb | SamlToken.vb | saml_xmldsig-core-schema.vb | ILogin.vb | XSLTransformation.vb
Project Build Debug Tools Window Help
Debug 600000
ParseFormK120_2006

strBldrFormK120.AppendFormat("{0,-6}", "K120") 'Form Type
strBldrFormK120.AppendFormat("{0,-12}", "") 'Document locator number
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrIDNumber/ConsolidatedParentFEIN}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrAddress/Address/USAddress/AddressLine1}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrAddress/Address/USAddress/City}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrAddress/Address/USAddress/State}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrAddress/Address/USAddress/ZIPCode}", strBldrFormK120)
ParseFilingMethod_2006(strBldrFormK120) 'Filing method
ParseElement("/{ReturnDataState/BusinessEntity/GeneralInformation/BusinessInfo/MethodToDetermineIncome}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrCode/BusinessActivity/NAICSCode}", strBldrFormK120)
ParseDateElement("/{ReturnDataState/BusinessEntity/Header/HdrDateField/DateBeganInState}", strBldrFormK120)
ParseDateElement("/{ReturnDataState/BusinessEntity/Header/HdrDateField/DateEndedInState}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrAddress/StateOfIncorporation/USAddress/State}", strBldrFormK120)
ParseDateElement("/{ReturnDataState/BusinessEntity/Header/HdrDateField/DateOfIncorporation}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrAddress/StateOfDomicile/USAddress/State}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/Header/HdrCode/FederalReturnFiledOther}", strBldrFormK120)
ParseCheckBox("/{ReturnDataState/BusinessEntity/Header/HdrCheckBox/ScheduleAttached}", strBldrFormK120)
ParseElement("/{ReturnDataState/BusinessEntity/GeneralInformation/FedReturnInfo/OtherDueDateFederal}", strBldrFormK120)
ParseCheckBox("/{ReturnDataState/BusinessEntity/Header/HdrName/NameOrAddressChange}", strBldrFormK120)
ParseCheckBox("/{ReturnDataState/BusinessEntity/Header/HdrCheckBox/NameOrAddressChange}", strBldrFormK120)
```

Parsing/XML Processing



```
MeFStateServiceWise.vb | DB.vb | MeFMSIServicesProxy.vb | SamlTokenManagerClient.vb | SamlToken.vb | saml_xmldsig-core-schema.vb | ILogin.vb | XSLTransformation.vb
Project Build Debug Tools Window Help
Debug namespace
XMLTransformation.vb | MeFStateServiceWise.vb |
Start Page | TestFile.vb [Design] | TestFile.vb | XMLTransformation.vb | MeFStateServiceWise.vb |
ParseElement

2674 End Sub
2675
2676
2677
2678 Private Function ParseElement(ByVal XPath As String, ByRef strBldr As StringBuilder, ByVal strFormat As
2679 'Procedure to get the XmlNode based on the XPath, and get the text from that node.
2680 Dim xNode As XmlNode
2681 Dim strNodeText As String
2682
2683 If XPath.Trim().Equals("/") Then
2684 strBldr.AppendFormat(strFormat, "")
2685 Return Nothing
2686 End If
2687
2688
2689 'Added by Andy Shely - Needed to add Namespace manager to get Xpaths from XML document
2690 Dim ns As New XmlNamespaceManager(xDoc.NameTable)
2691 ns.AddNamespace("ns1", "http://www.irs.gov/xsl")
2692
2693 XPath = XPath.Replace("/", "ns1:")
2694 XPath = XPath.Replace("/ns1:ReturnDataState/", "//")
2695
2696
2697 If xReferenceNode Is Nothing Then xReferenceNode = xDoc
2698 Try
2699 xNode = xReferenceNode.SelectSingleNode(xpath, ns)
2700
2701 If xNode Is Nothing Then
2702 If strBlank = Nothing Then
2703 strNodeText = ""
2704 Else
2705 Dim sb As New StringBuilder
2706 If strBlank = "" Then
2707 sb.Append(" ", sLength)
2708 Else
2709 sb.Append(strNodeText, sLength)
2710 End If
2711 strNodeText = sb.ToString
2712 End If
2713 Else
2714 strNodeText = xNode.InnerText
2715 End If
2716
2717 If strFormat = Nothing Then
2718 If IsNumeric(strNodeText) And strNodeText.IndexOf(".") > -1 Then
2719
```


Parsing/XML Processing



```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<AcknowledgementList xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
<Count>2</Count>
<Acknowledgement>
  <SubmissionId>90591320071020488596</SubmissionId>
  <EFIN>905913</EFIN>
  <GovernmentCode>KSST</GovernmentCode>
  <SubmissionType>K120</SubmissionType>
  <TaxYear>2006</TaxYear>
  <SubmissionCategory>CORP</SubmissionCategory>
  <FilingStatus>Accepted</FilingStatus>
  <StatusDate>2007-04-12</StatusDate>
  <EIN>11000004</EIN>
</Acknowledgement>
<Acknowledgement>
  <SubmissionId>12345620071060001286</SubmissionId>
  <EFIN>123456</EFIN>
  <GovernmentCode>KSST</GovernmentCode>
  <SubmissionType>K120</SubmissionType>
  <TaxYear>2006</TaxYear>
  <SubmissionCategory>CORP</SubmissionCategory>
  <FilingStatus>Rejected</FilingStatus>
  <StatusDate>2007-04-16</StatusDate>
  <EIN>110000001</EIN>
<ErrorList errorCount="12">
  <Error errorId="1">
    <ErrorCategory>XML Validation</ErrorCategory>
    <ErrorMessage>Error: The 'http://www.irs.gov/efile:Month' element has an invalid value according to its data type. An err
    <RuleNumber>XML Validation</RuleNumber>
    <Severity>99</Severity>
  </Error>
  <Error errorId="2">
    <ErrorCategory>XML Validation</ErrorCategory>
    <ErrorMessage>Error: The 'http://www.irs.gov/efile:Month' element has an invalid value according to its data type. An err
    <RuleNumber>XML Validation</RuleNumber>
    <Severity>99</Severity>
  </Error>
  <Error errorId="3">
    <ErrorCategory>XML Validation</ErrorCategory>
```

Lessons Learned



- Need for automation.
 - Added programming to run the steps in the proper sequence and added error trapping. This was done by adding a .Net Timer to the program to facilitate automation. It runs until the MeF response of more submissions is False.
- Time Outs.
 - Added timeout of 600,000 milliseconds (approx. 10 minutes) to web service client programs to handle delays in the web service and network traffic. Also set maximum results to 10 on GetNewSubmissions.
- Transmission Failures
 - Retrieve submissions by message id to make sure no submissions were lost.
 - IRS also supplies spreadsheet of un-acknowledged returns by submission id.
- On going process. MeF is a new system and we are still learning as we go.
 - Joan Barr, Xan Ostro, Tom Erickson, and the rest of the IRS has been extremely helpful when we have had problems.

Questions



- Terry Hunt
(785) 296-4066
Terry_Hunt@kdor.state.ks.us
- Andrew Sheley
(785) 296-0545
Andrew_Sheley@kdor.state.ks.us