BUILDING YOUR XML FOR MEF

Terry Garber.
FTA

August 2016

THIS HOUR

- NOT a comprehensive technical class in XML
- Lightning overview of XML and how to approach building your state's MeF Individual schema set
- If you remember nothing else: Building your XML should be a joint effort between the business folks and the technical folks
  - Every XML schema is a little computer program written in XML code – there is no way to make XML non-technical.
  - MeF Efile is a business function - the XML must be built to the business rules for your state program.

GETTING STARTED

- Altova XML Spy is needed for everyone actively working on the schemas
  - De facto standard for communicating XML
- Example: E-Standards StateIdividualPackage
  - You are not starting from a blank page!
  - Even if you have last year's schema set, start from the standard and apply your customization and insert your own ReturnDataState (your own state's forms)
  - Do not crib from another state
    - They may or may not be following best practices
    - Their business rules may not match yours

WHERE THE E-STANDARDS COME IN

- All this is in fact documented
  - "TIGERS MEF Standards"
    - A bit out of date, but usable
    - E-Standards site www.statemef.com – bottom link on the home page
- E-Standards Review Process
  - Combined state/industry team
  - Review all new programs or major changes
  - Or, whenever you want another set of eyes

BASIC XML CONCEPTS

- XML documents are made up of elements
- Elements can be simple or complex
- Simple elements are single pieces of data
- Complex elements are made up of sub-elements, such as an address made up of street, city, state, zip
- Elements are all described by “types”
  - SimpleTypes specify if a simple element is a number, a monetary amount, or text and give length and other restrictions such as eligible values.
  - ComplexTypes show the simple elements that make up the complex element, each with its SimpleType

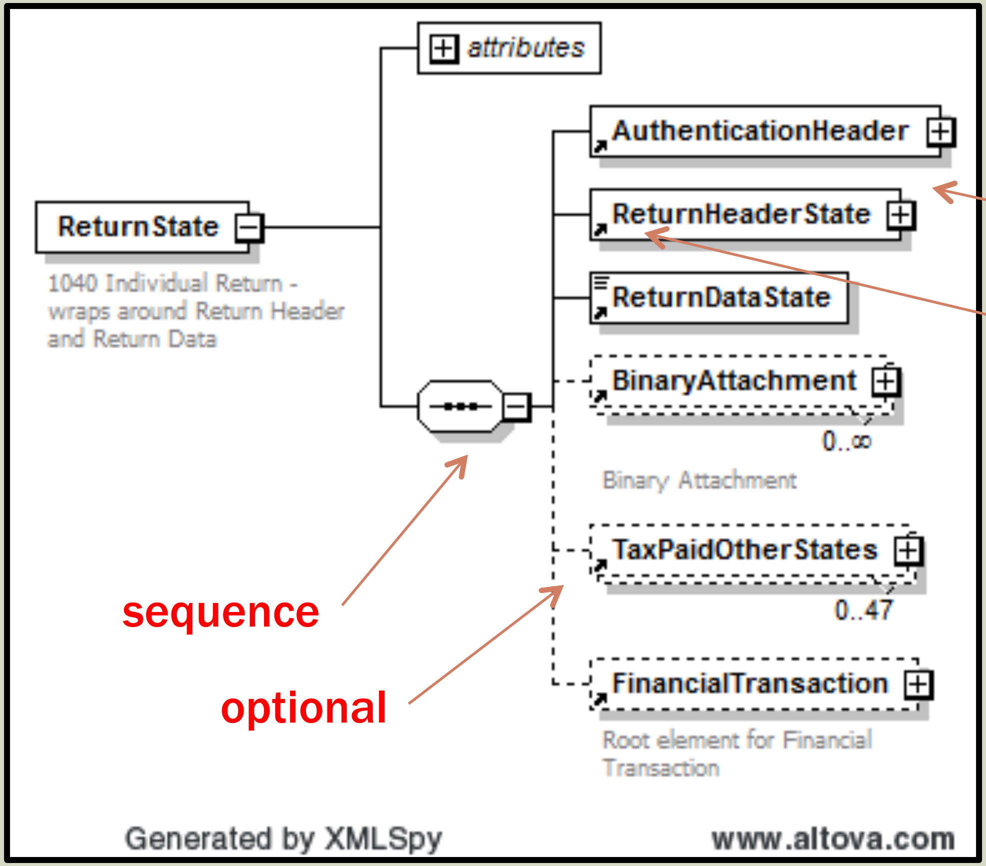
PACKAGED “BUILDING BLOCKS”

- We call them “efileTypes” – term coined by IRS
- EfileTypes are copied directly from IRS
- StateeFileTypes are state building blocks
- Both simple and complex types
- Define standard formats for elements, such as names, addresses, and amounts
- You can (and probably will) set up your own XXeFileTypes, but you are encouraged to use the standard efileTypes wherever possible

TRICKIER CONCEPTS

- xsd:sequence – so all the returns come in with fields in the same order
- xsd:choice – the ability to say you must give us one of these, but not more than one
- xsd:include – the ability to include one schema as part of a larger schema, so a large schema can be broken into subject areas
- xsd:enumeration – the ability to list acceptable values for an element
- xsd:pattern – the ability to specify patterns of characters making up an element

E-STANDARDS FRAMEWORK



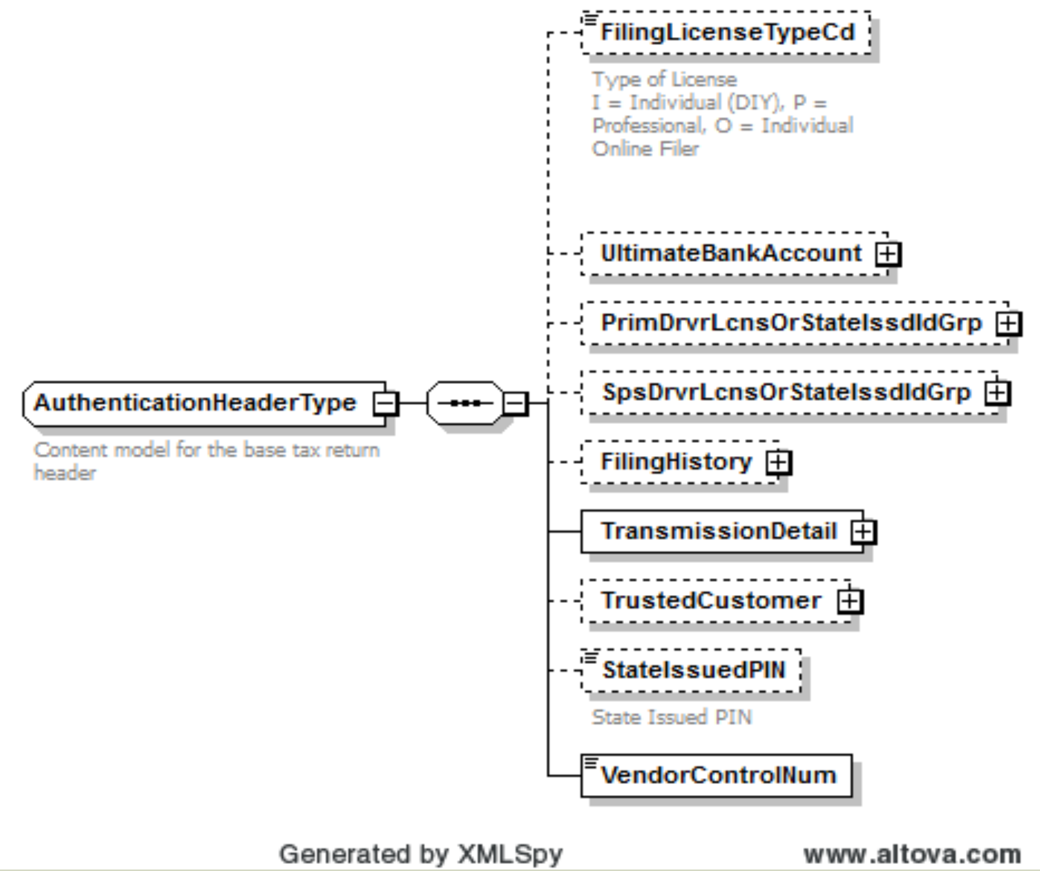
mandatory

include

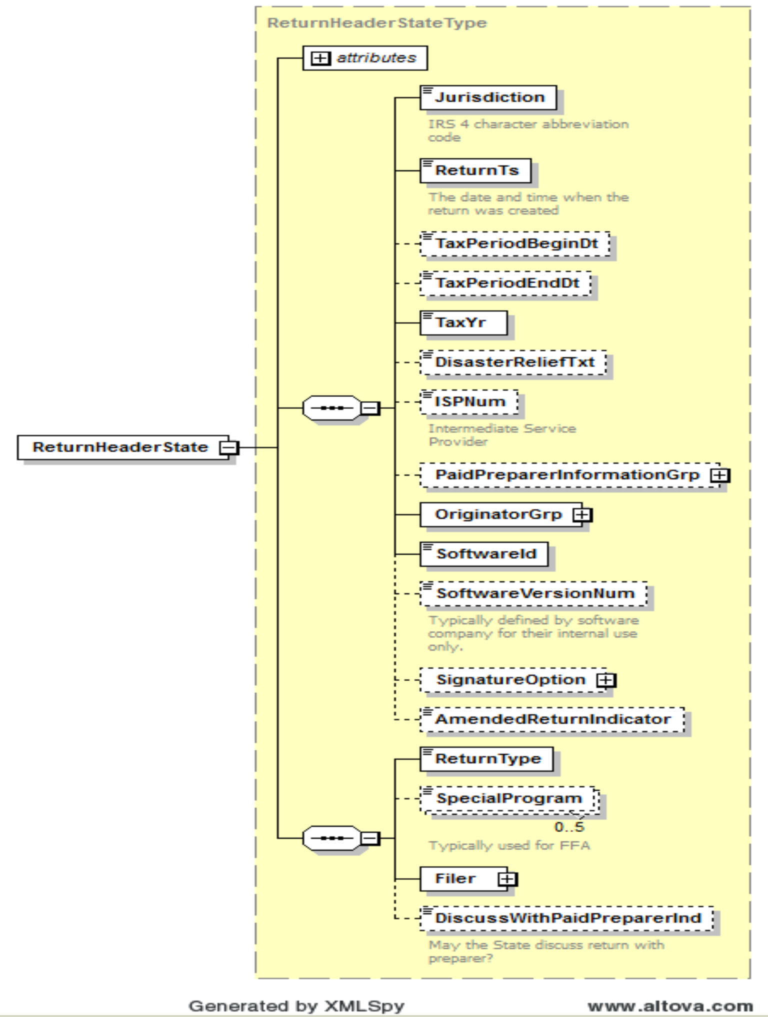
sequence

optional

AUTHENTICATION HEADER



STATE RETURN HEADER



STATE ENUMERATIONS IN THE HEADER

```
<xsd:simpleType name="ReturnTypeType">
        <xsd:annotation>
            <xsd:documentation>Enumerate the specific Return Types that your state accepts</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="String20Type">
            <!--for example:
            <xsd:enumeration value="Form1"/>
            <xsd:enumeration value="Form1A"/>
            -->
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="SpecialProgramType">
        <xsd:annotation>
            <xsd:documentation>Enumerate the specific special programs that your state supports</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="String20Type">
            <!--for example:
            <xsd:enumeration value="Freefile"/>
            <xsd:enumeration value="Spanish"/>
            -->
        </xsd:restriction>
    </xsd:simpleType>
```

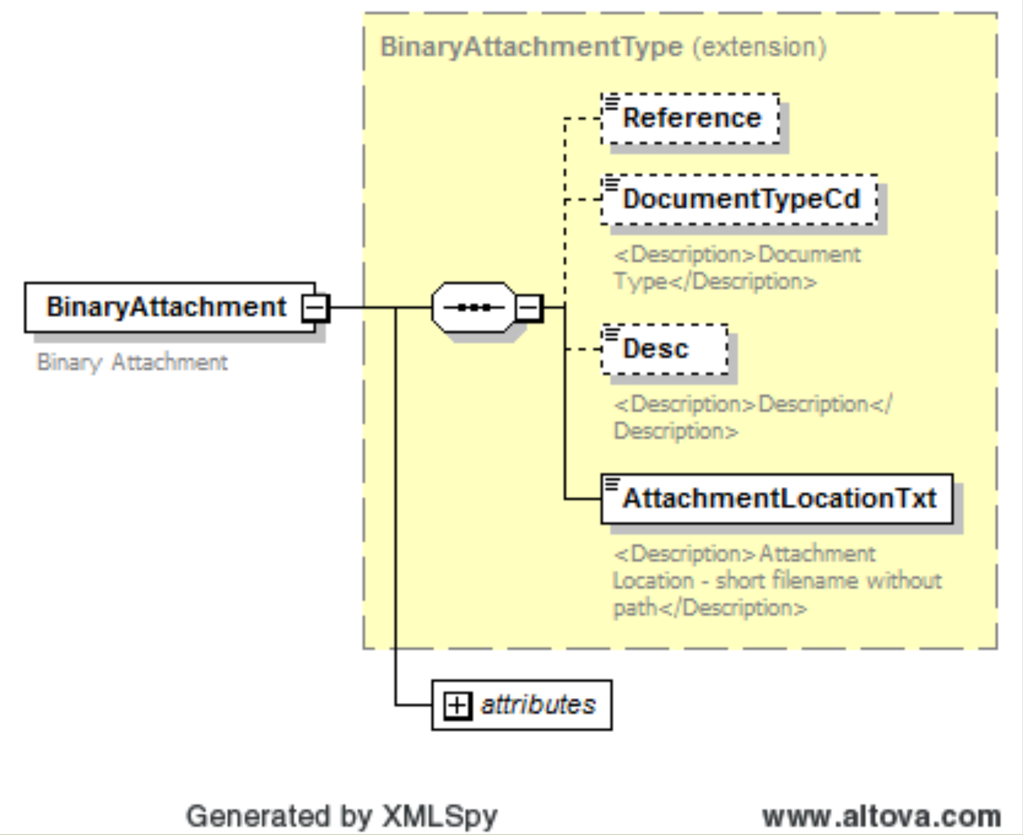
RETURN DATA STATE – YOUR FORMS

- The meat of the efile – the tax data
  - Form by form, all the elements not in the E-Standards schemas (do not duplicate!)
    - Structure? Length? Repeats? Allowed values?
    - Checkboxes, multiple choice check-offs
    - Critical for all elements: mandatory or optional?
    - Critical for amounts: Positive? Non-negative?
  - Go through all your forms and list the business rules first, then code the schema
  - Two-edged sword: The XML return will be rejected if XML schema coded business rules fail

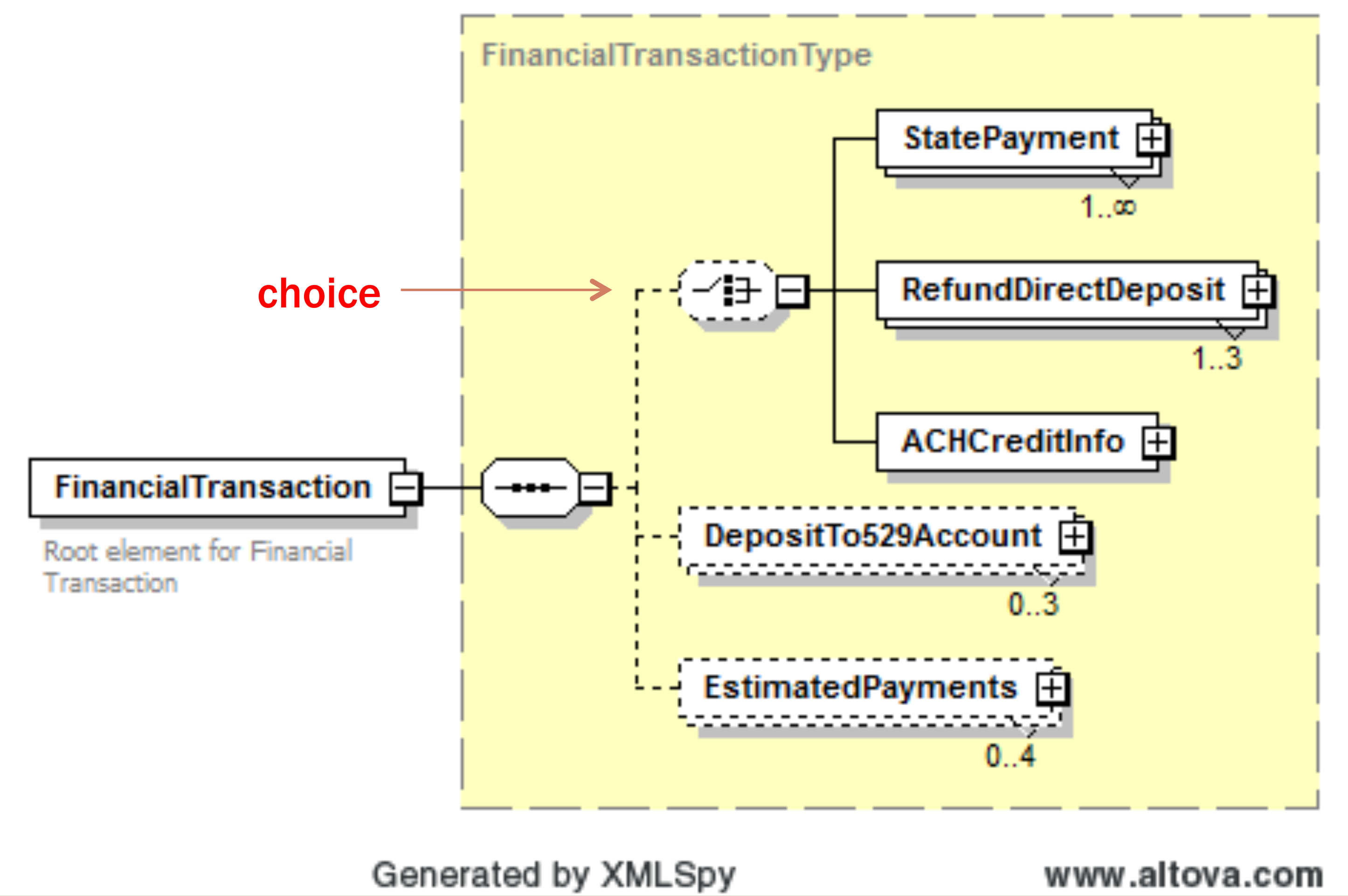
E-STANDARDS AND BEST PRACTICES

- Standards – E-Standards Review Team requires
  - Element names less than 30 characters
  - Element names “upper camel case”
- Best Practices – Review Team points out
  - Easier to code or maintain
  - Do NOT use line numbers in element names!!
  - Do not have a mandatory complex element with all optional children – your could get an empty element
  - Use elements rather than attributes

BINARY ATTACHMENTS


BinaryAttachmentType (extension)
Reference
DocumentTypeCd
<Description>Document Type</Description>
BinaryAttachment
Binary Attachment
Desc
<Description>Description</Description>
AttachmentLocationTxt
<Description>Attachment Location - short filename without path</Description>
attributes
Generated by XMLSpy
www.altova.com

FINANCIAL TRANSACTION



- You don’t have to do this alone!
- Send email to statemef@rsimail.com and cc Terry.Garber@taxadmin.org
- We really are ready to help you!!