

# XML Schema Recommendations

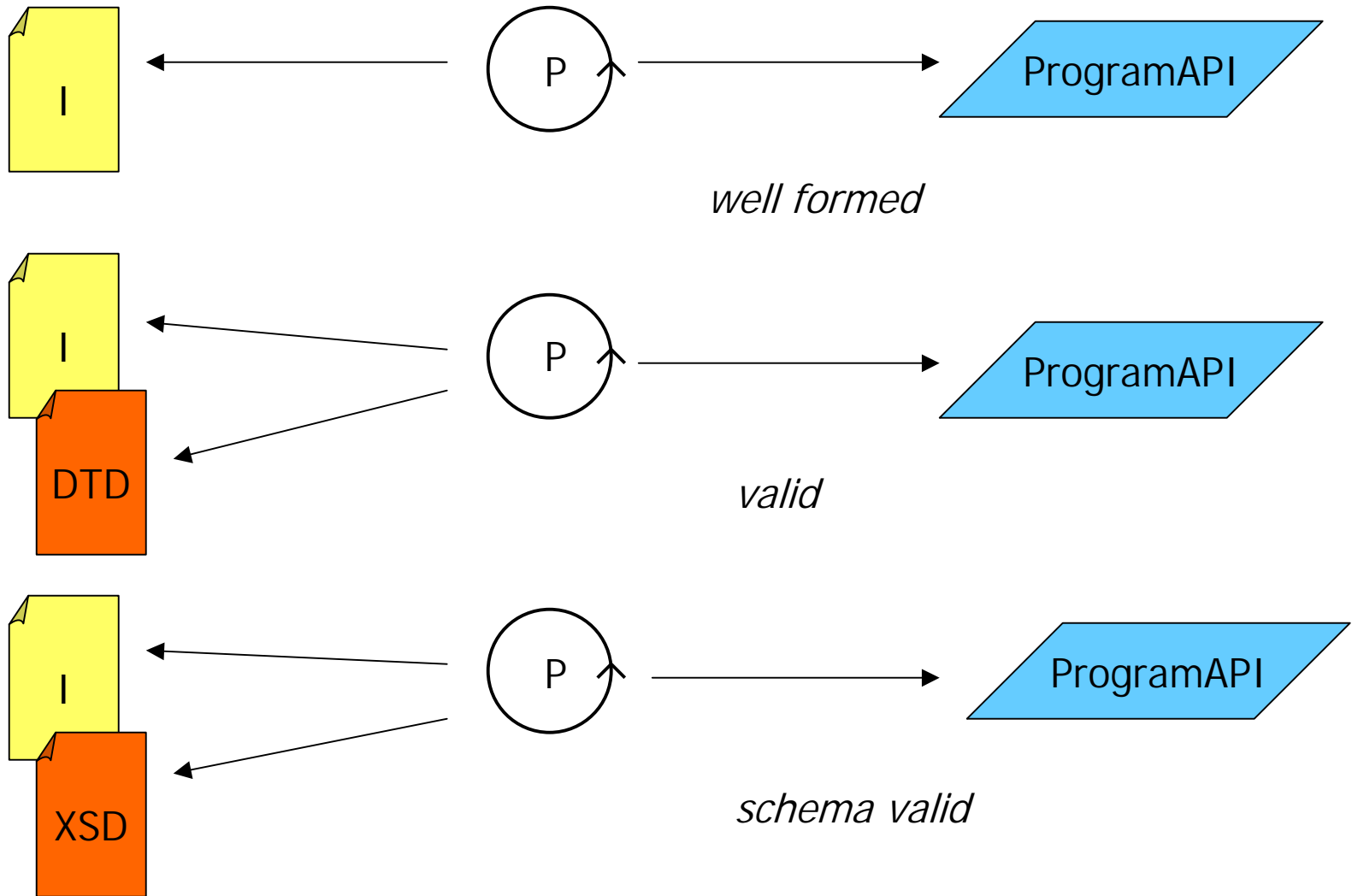
***NACS POS/Backoffice Committee  
contributed by David Ezell  
VeriFone, Inc.***



# Using XML Schema

- **Always think in terms of the XML infoset**
- **Follow consistent rules in designing schemas**
  - Use composition to modularize the schema
  - Be consistent in use of attributes vs. elements
  - Be aware of the difference between “types” and “names”
  - Plan for extensions
- **Use XPath to test effectiveness of schema design**

# XML Processing Terminology



# XML Infoset: sample document

```
<?xml version="1.0" ?>
<mv:person xmlns:mv="http://www.exmpl3.com"
  mv:ssn="555-55-5555" mv:status="US-Citizen" >
  <mv:name xml:lang="US-en" mv:order="normal" >David Ezell</mv:name>
  <mv:address>
    <mv:street>444 Homestead Cir.</mv:street>
    <mv:city>Tampa</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:address>
  <mv:workAddress>
    <mv:street>300 S. Park Place Blvd.</mv:street>
    <mv:subStreet>Suite 100</mv:subStreet>
    <mv:city>Clearwater</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:workAddress>
</mv:person>
```

# XML Infoset: definitions

- **Information Items**

- Document Information item

- Child elements (ordered), exactly one
    - Processing instruction information item
    - Notation information items (unordered) (designation of processing models)
    - Entity information items (unordered) (may contain notations)

- Element Information Items

- URI:LocalName
    - Ordered list of element, PI, reference to Unexpanded Entities, and character information items
    - Unordered list of attribute information items (Unicode)

- Attribute Information Items

- URI:LocalName
    - Ordered list of character information items (Unicode)

- **Namespaces**

# XML Infoset

```
<?xml version="1.0" ?>
<mv:person xmlns:mv="http://www.exmpl3.com"
           mv:ssn="555-55-5555" mv:status="US-Citizen" >
  <mv:name xml:lang="US-en" mv:order="normal">David Ezell</mv:name>
  <mv:address>
    <mv:street>444 Homestead Cir.</mv:street>
    <mv:city>Tampa</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:address>
  <mv:workAddress>
    <mv:street>300 S. Park Place Blvd.</mv:street>
    <mv:subStreet>Suite 100</mv:subStreet>
    <mv:city>Clearwater</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:workAddress>
</mv:person>
```

## – Document Information item

- Child elements (ordered), exactly one
- Processing instruction information item
- Notation information items (unordered) (designation of processing models)
- Entity information items (unordered) (may contain notations)

# XML Infoset

```
<?xml version="1.0" ?>
<mv:person xmlns:mv="http://www.exmpl3.com"
            mv:ssn="555-55-5555" mv:status="US-Citizen" >
  <mv:name xml:lang="US-en" mv:order="normal" >David Ezell</mv:name>
  <mv:address>
    <mv:street>444 Homestead Cir.</mv:street>
    <mv:city>Tampa</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:address>
  <mv:workAddress>
    <mv:street>300 S. Park Place Blvd.</mv:street>
    <mv:subStreet>Suite 100</mv:subStreet>
    <mv:city>Clearwater</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:workAddress>
</mv:person>
```

## – Element Information Items

- URI:LocalName
- Ordered list of element, PI, reference to Unexpanded Entities, and character information items
- Unordered list of attribute information items (Unicode)

# XML Infoset

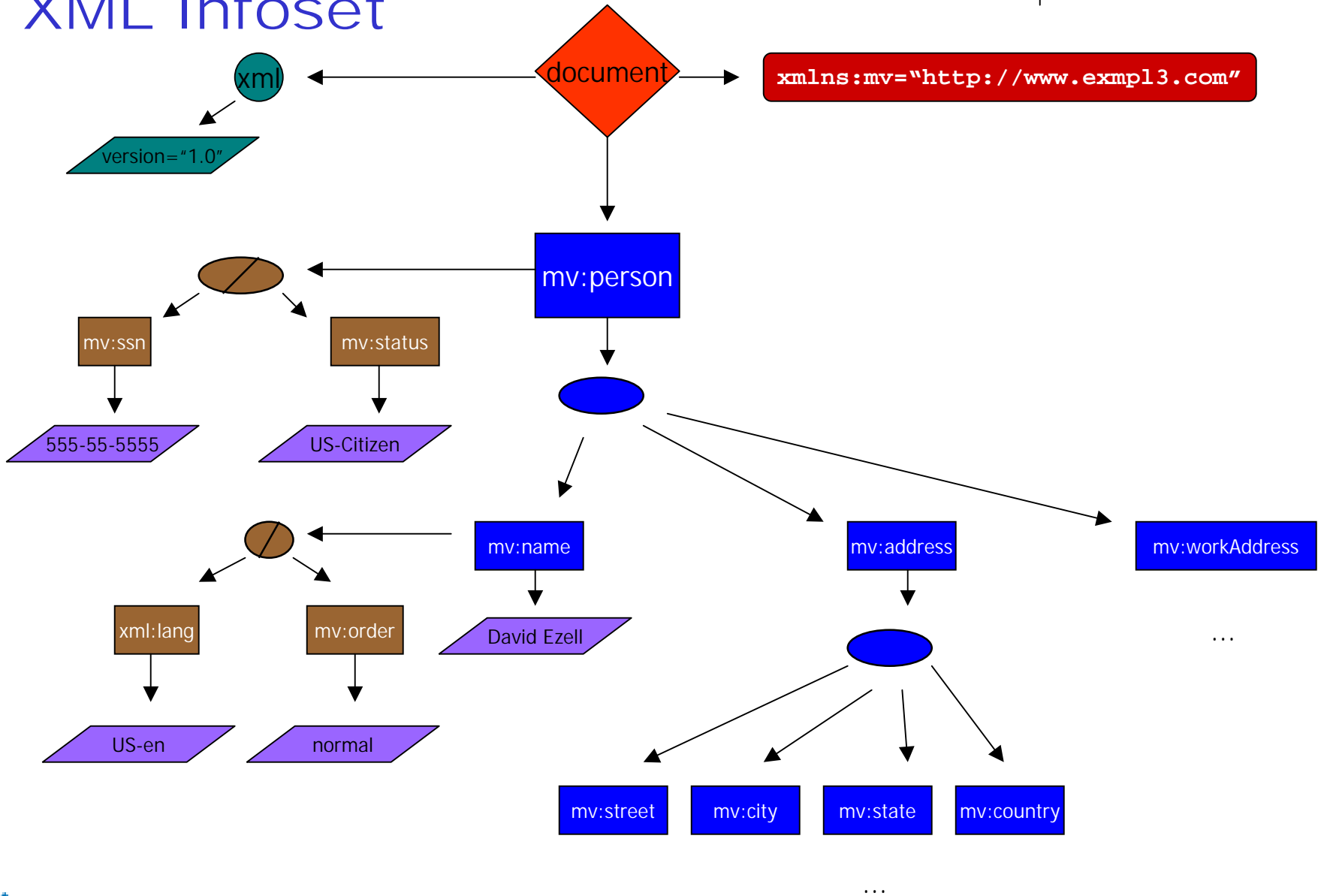
```
<?xml version="1.0" ?>
<mv:person xmlns:mv="http://www.exmpl3.com"
            mv:ssn="555-55-5555" mv:status="US-Citizen" >
  <mv:name xml:lang="US-en" mv:order="normal" >David Ezell</mv:name>
  <mv:address>
    <mv:street>444 Homestead Cir.</mv:street>
    <mv:city>Tampa</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:address>
  <mv:workAddress>
    <mv:street>300 S. Park Place Blvd.</mv:street>
    <mv:subStreet>Suite 100</mv:subStreet>
    <mv:city>Clearwater</mv:city>
    <mv:state>FL</mv:state>
    <mv:country>USA</mv:country>
  </mv:workAddress>
</mv:person>
```

## – Attribute Information Items

- URI:LocalName
- Ordered list of character information items (Unicode)



# XML Infoset



# Using XML Schema: Composition

- use “include” to expand a namespace (probably committee only)
- use “import” to use items from another namespace (for vendors)
- use “redefine” only with great care (if ever)

# Using XML Schema: Attributes vs. Elements

- **use attributes for “key” values**
  - helps speed server-side (SAX) apps
- **limit attributes to information “about” an element**
  - avoid putting essential state information in attributes

## Using XML Schema: Types vs. Names

type name

```

  ↓   ↓
int a;
long b;

```

```

struct foo {
  int a;
  long b;
};

```

```

struct bar {
  struct foo foo_tag;
  int a;
  long b;
};

```

**struct bar baz;****baz.foo\_tag.a = 15;**

```

  ↙   ↘   ↗
baz.foo_tag.a = 15;

```

name name name

no types

type name

# Using XML Schema: Planning Extensions

- Use “substitutionGroup” for planned “unions”
- Use “xsi:type” to allow vendor extensibility
- Use “any” with namespace constraint “##other” as a last resort

# XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>  
<xsd:schema targetNamespace="http://www.exmpl3.com"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
xmlns:mv="http://www.exmpl3.com"  
elementFormDefault="unqualified">
```

• • •

```
<xsd:schema>
```

# XML Schema: defining types

```
<xsd:attribute name="order" type="xsd:string"/>

<xsd:simpleType name="stateType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="GA"/>
    <xsd:enumeration value="FL"/>
    <xsd:enumeration value="SC"/>
    <xsd:enumeration value="NY"/>
    <xsd:enumeration value="OH"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="nameType">
  <xsd:simpleContent>
    <xsd:restriction base="xsd:string">
      <xsd:attribute ref="mv:order" use="required"
        form="qualified"/>
    </xsd:restriction>
  </xsd:simpleContent>
</xsd:complexType>
```

# XML Schema: defining types

```
<xsd:complexType name="addressType">
  <xsd:sequence>
    <xsd:element name="street" type="xsd:string"/>
    <xsd:element name="subStreet" type="xsd:string" minOccurs="0"/>
    <xsd:element name="city" type="xsd:string"/>
    <xsd:element name="state" type="mv:stateType"/>
    <xsd:element name="country" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="address" type="mv:addressType"/>
```



# XML Schema: defining vocabulary

```
<xsd:element name="person">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="name" type="mv:nameType"/>
      <xsd:element ref="mv:address"/>
      <xsd:element name="workAddress"
        type="mv:addressType"/>
      <xsd:element ref="mv:address" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="ssn" type="xsd:string" use="required"/>
    <xsd:attribute name="status" type="xsd:string"
      use="required"/>
  </xsd:complexType>
</xsd:element>
```

# XML Schema: extending types

```
<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema targetNamespace="http://www.exmpl4.com"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:mv0="http://www.exmpl3.com"
  xmlns:mv1="http://www.exmpl4.com"
  elementFormDefault="unqualified">

  <xsd:import namespace="http://www.exmpl3.com"
    schemaLocation="/transactions/samp1.xsd"/>

  <xsd:complexType name="milAddressType">
    <xsd:complexContent>
      <xsd:extension base="mv0:addressType">
        <xsd:choice>
          <xsd:element name="addrExtension"
            type="xsd:string"/>
          <xsd:element name="baseQual"
            type="xsd:string"/>
        </xsd:choice>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:element name="militaryAddress" type="mv1:milAddressType"
    substitutionGroup="mv0:address"/>

</xsd:schema>
```

# XML Schema: extension using substitution

```

<?xml version="1.0" encoding="UTF-8"?>
<mv0:person xmlns:mv0="http://www.exmpl3.com"
  xmlns:mv1="http://www.exmpl4.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xml="http://www.w3.org/TR/REC-xml"
  xsi:schemaLocation="http://www.exmpl4.com /transactions/samp2.xsd"
  ssn="555-55-5555" status="US-Citizen">
  <name mv0:order="normal">David Ezell</name>
  <mv0:address>
    <street>444 Homestead Cir.</street>
    <city>Tampa</city>
    <state>FL</state>
    <country>USA</country>
  </mv0:address>
  <workAddress>
    <street>300 S. Park Place Blvd.</street>
    <subStreet>Suite 100</subStreet>
    <city>Clearwater</city>
    <state>FL</state>
    <country>USA</country>
  </workAddress>
  <mv1:militaryAddress>
    <street>231 Hardship Street</street>
    <city>Columbus</city>
    <state>GA</state>
    <country>USA</country>
    <baseQual>Fort Benning</baseQual>
  </mv1:militaryAddress>
</mv0:person>

```

# XML Schema: extension using xsi:type

```
<?xml version="1.0" encoding="UTF-8"?>
<mv0:person xmlns:mv0="http://www.exmpl3.com"
  xmlns:mv1="http://www.exmpl4.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xml="http://www.w3.org/TR/REC-xml"
  xsi:schemaLocation="http://www.exmpl4.com /transactions/samp2.xsd"
  ssn="555-55-5555" status="US-Citizen">
  <name mv0:order="normal">David Ezell</name>
  <mv0:address>
    <street>444 Homestead Cir.</street>
    <city>Tampa</city>
    <state>FL</state>
    <country>USA</country>
  </mv0:address>
  <workAddress>
    <street>300 S. Park Place Blvd.</street>
    <subStreet>Suite 100</subStreet>
    <city>Clearwater</city>
    <state>FL</state>
    <country>USA</country>
  </workAddress>
  <mv0:address xsi:type="mv1:milAddressType">
    <street>231 Hardship Street</street>
    <city>Columbus</city>
    <state>GA</state>
    <country>USA</country>
    <baseQual>Fort Benning</baseQual>
  </mv0:address>
</mv0:person>
```