Metadata XML for Medication Modifiers

Metadata XML for Medication Modifiers

XML Specification for Descriptive Files of Values

Version 3.02

In the Observation_fact table of the Clinical Research Chart of i2b2, the following columns are used to describe values associated with concepts:

ValType_CD (Varchar(50))

NVal_num (number)

TVal_char (Varchar(255) – standard, sized per implementation)

ValueFlag_CD (Varchar(50))

Quantity_CD (number)

Units_CD (Varchar(50))

Observation_blob (blob)

Values that are expected to occur in these columns are specified in a column of the Ontology Cell named "Metadata_XML". Modifiers may often have values associated with them, and the above columns are filled and the Metadata_XML is specified in the same way as the concept_cd, however the value applies to the modifier rather than the concept_cd. When there is a value in these columns and a modifier is present in the modifier_cd column, the value ALWAYS applies to the modifier_cd coded element. Note that if the value is desired to apply to the base concept_cd, the modifier_cd column is specified to be "@".

Values are defined according to the ValType_CD column, which can be construed as an identifier for the type of value object that fills out the rest of the above columns. There are two identifiers that are used for medication modifiers:

N = numeric value

T = enumerated or string value

Numeric values are normalized in one of two ways. In versions through 1.5, the values are assumed to be equivalent to a "Normal" set of units. In the Metadata_XML file this is specified as <NormalUnits>. This implies that all values are normalized to those units during loading. Any value in the Units_CD column is ignored. In versions 1.6 RC4 and above, the Units_CD column is used to specify several possible values for the units. Conversions will occur between units as long as they are specified in the Metadata_XML file. It should be recognized however that queries may run considerable slower when specifying values in this mode.

Quantity is part of the value specification, and has units that are in the denominator of the unit's string. If the value of quantity is null, then the value is assumed to be 1.

The required tags are Version, CreationDateTime, TestID, DataType, and Flagstouse. All tags except data in the Analysis tag is expected to persist from one build to the next. Note that the Metadata_XML file can be used during data loading in order to provide data type and normalization information for the values that are being loaded (especially important when this information is not specified by the source)

Version will contain a real number with the XML specification version. This is version "3.02"

CreationDateTime is the date and time the file was created, in the format MM/DD/YY HH:MM:SS using the 24 hour time.

TestID is the Code (concept_cd or modifer_cd) associated with the file.

CodeType is equivalent to the "symantic type" and will be fully specified at a later date.

TestName is the literal name of the test. It is used for the convenience of the Analyst.

DataType will contain the code for what kind of data to expect for this test. Possible values are:

PosInteger – domain of all positive integers

Integer – domain of all integers

PosFloat – domain of all positive real numbers

Float – domain of all real numbers

Enum – domain of enumerated values

String – domain of free text, NOT enumerated text values, which would be the Enum data type.

Flagstouse is a string of concatinated flags that are valid for this TestID. For example, for most PosIntegers it would be "LNH" Acceptable values are L (low), N (normal), H (high), A (abnormal), T (toxic)

Oktousevalues will contain a "Y" or a message, which indicates why it is not OK to use values. Nothing indicates that values cannot be used and the user may only specify values using flags.

UnitValues is the parent tag of a set of possibly repeating tags. It contains data when the datatype is PosInteger, Integer, Float, and PosFloat. All units are always LOWER CASE.

NormalUnits can exist only once, it contains a string that a user would recognize which represents the units of the value as we have it in the data warehouse.

EqualUnits can repeat, it contains other strings that are numerically equal to the NormalUnits string.

ConvertingUnits can repeat, it contains other strings, Units, and the factor, MultiplyingFactor, such that values in theses units need to be multiplied by the multiplyingfactor to convert them into values of NormalUnits. For example, if NormalUnits was in feet, and ConvertingUnits was in yards, the MultiplyingFactor would be 0.333.

ExcludingUnits can repeat, it contains units that will cause the test with these units to be excluded from the query (and in versions 1.5 and prior, should not be included in the data load). These are units that can not be converted with a simple multiplier to NormalUnits. These concepts will need a new code, or if grouped will need to go into their own group, in order to be queried by value.

EnumValues is used to specify the list of acceptable enumerated values, each enclosed in the Val tag. Enumerated values that indicate an invalid test result (for the enum datatype) can be enclosed in the ExcludingVal tag. The "descriptions" parameter exists to allow a humanly readable value to be presented for the enumerated value in choice boxes of user intergaces. ExcludingVal is directed to loading processes, and specifies values not to lead into the database or display in a user interface (for example, "pending").

MaxStringLength will contain a postitive integer or 0, representing the longest acceptable string length, if the datatype is "string".

LowofLowValue specifies the lowest of the low range values for PosInteger, Integer, Float, and PosFloat datatypes.

HighofLowValue specifies the highest of the low range values for PosInteger, Integer, Float, and PosFloat datatypes.

LowofHighValue specifies the lowest of the high range values for PosInteger, Integer, Float, and PosFloat datatypes.

HighofHighValue specifies the highest of the high range values for PosInteger, Integer, Float, and PosFloat datatypes.

LowofToxicValue specifies the lowest of the toxic range values for PosInteger, Integer, Float, and PosFloat datatypes.

HighofToxicValue specifies the highest of the toxic range values for PosInteger, Integer, Float, and PosFloat datatypes (rarely used).

Analysis contains tags that are used to retain data for the analysis of a valuemetadata element. However, it is transient, and will not be retained from one build to the next. It generally only reflects the current state of the metadata database. The New tag is intended to contain new unitvalues, enumvalues, and such.

```
<ValueMetadata>
    <Version>3.02</Version>
    <CreationDateTime>
    <TestID/>
    <TestName/>
    <DataType>PosInteger, Integer, Float, PosFloat, Enum, String</DataType>
    <Flagstouse/>
    <Oktousevalues/>
    <UnitValues>
        <NormalUnits/>
        <EqualUnits/>
        <ConvertingUnits/>
        <Units/>
        <MultiplyingFactor/>
        <ConvertingUnits/>
        <ExcludingUnits/>
    </UnitValues>
    <EnumValues>
```

```
        <Val description=""/>
        <ExcludingVal description=""/>
    </EnumValues>
    <MaxStringLength/>
    <CommentsDeterminingExclusion>
        <Com></Com>
    </CommentsDeterminingExclusion>
    <LowofLowReference/>
    <HighofLowReference/>
    <LowofHighReference/>
    <HighofHighReference/>
    <LowofToxicReference/>
    <HighofToxicReference/>
    <Analysis>
        <Enums/>
        <Counts/>
        <New/>
    </Analysis>
</ValueMetadata>
```

Rolling up group codes

Groups are one hierarchical level above the leaf codes, and the XML file that represents this group code will be computed from all the XML files that represent the children in the group. The following fields are computed form the children's Metadata_XML files in the following manner:

Flagstouse is a string of concatinated flags that are valid for any Children in this group.

Oktousevalues will contain a Y or nothing. If the Children in the group are not all Numbers (Float, PosFloat, Integer, PosInteger) or all Enums, or all Strings, then Oktousevalues is usually a short explanatory string.

UnitValues is the parent tag of a set of possibly repeating tags. It contains data when the datatype is PosInteger, Integer, Float, and PosFloat. Each unique XML element and value in the Children of this group is simply repeated as necessary under this tag.

EnumValues is used to specify the list of acceptable enumerated values for all children in the group, each enclosed in the Val tag. Enumerated values that indicate an invalid result (for the enum datatype) can be enclosed in the ExcludingVal tag. These (like UnitValues) are simply repeated for each unique value found in the Children.

MaxStringLength will contain a postitive integer or 0, representing the longest acceptable string length of all the Children in the Group.

LowofLowReference specifies the lowest of the low range values for PosInteger, Integer, Float, and PosFloat datatypes of all the Children in the Group.

HighofLowReference specifies the highest of the low range values for PosInteger, Integer, Float, and PosFloat datatypes of all the Children in the Group..

LowofHighReference specifies the lowest of the high range values for PosInteger, Integer, Float, and PosFloat datatypes of all the Children in the Group..

HighofHighReference specifies the highest of the high range values for PosInteger, Integer, Float, and PosFloat datatypes of all the Children in the Group..

LowofToxicReference specifies the lowest of the toxic range values for PosInteger, Integer, Float, and PosFloat datatypes of all the Children in the Group.

HighofToxicReference specifies the highest of the toxic range values for PosInteger, Integer, Float, and PosFloat datatypes of all the Children in the Group.

Analysis is not used in the group, and therefore is always nothing.

Examples of medication modifiers XML_Metadata codes

DOSE:

<?xml version="1.0"?>

<ValueMetadata>

<Version>3.02</Version>

<CreationDateTime>01/26/2011 00:00:00</CreationDateTime>

<TestID>MED:DOSE</TestID>

<TestName>Medication Dose</TestName>

<DataType>PosFloat</DataType>

<Flagstouse></Flagstouse>

<Oktousevalues>Y</Oktousevalues>

<EnumValues></EnumValues>

<UnitValues>

<NormalUnits >mg/dose</NormalUnits>

<EqualUnits >mg/tablet</ EqualUnits>

<EqualUnits>gm/liter</EqualUnits>

<EqualUnits>mg/ml</EqualUnits>

<ExcludingUnits>%</ExcludingUnits>

<ExcludingUnits>iu</ExcludingUnits>

<ExcludingUnits>iu/ml</ExcludingUnits>

<ExcludingUnits>mcg/inh</ExcludingUnits>

<ExcludingUnits>mEq</ExcludingUnits>

<ExcludingUnits>mg/inh</ExcludingUnits>

<ExcludingUnits>u/gm</ExcludingUnits>

<ExcludingUnits>u/ml</ExcludingUnits>

<ConvertingUnits>

<Units>gm/tablet</Units>

<MultiplyingFactor>1000</MultiplyingFactor>

</ConvertingUnits>

<ConvertingUnits>

<Units>gm/15ml</Units>

<MultiplyingFactor>66.66</MultiplyingFactor>

</ConvertingUnits>

<ConvertingUnits>

<Units>gm/50ml</Units>

<MultiplyingFactor>20</MultiplyingFactor>

</ConvertingUnits>

<ConvertingUnits>

<Units>mcg/tablet</Units>

<MultiplyingFactor>0.001</MultiplyingFactor>

</ConvertingUnits>

<ConvertingUnits>

```
            <Units>mcg/ml</Units>
            <MultiplyingFactor>0.001</MultiplyingFactor>
        </ConvertingUnits>
        <ConvertingUnits>
            <Units>mg/0.5ml</Units>
            <MultiplyingFactor>2</MultiplyingFactor>
        </ConvertingUnits>
        <ConvertingUnits>
            <Units>mg/15ml</Units>
            <MultiplyingFactor>0.066</MultiplyingFactor>
        </ConvertingUnits>
        <ConvertingUnits>
            <Units>mg/5ml</Units>
            <MultiplyingFactor>0.2</MultiplyingFactor>
        </ConvertingUnits>
    </UnitValues>
</ValueMetadata>
```

FREQUENCY

```
<?xml version="1.0"?>
<ValueMetadata>
    <Version>3.02</Version>
    <CreationDateTime>01/26/2011 00:00:00</CreationDateTime>
    <TestID>MED:FREQ</TestID>
    <TestName>Medication Frequency</TestName>
    <DataType>Enum</DataType>
    <Flagstouse></Flagstouse>
    <Oktousevalues>N</Oktousevalues>
    <EnumValues>
        <Val description="Before meals">AC</Val>
        <Val description="Twice per day">BID</Val>
        <Val description="Once per day">QD</Val>
        <Val description="Once at night">QHS</Val>
        <Val description="Three times per day">TID</Val>
    </EnumValues>
    <UnitValues></UnitValues>
</ValueMetadata>
```

ROUTE

```
<?xml version="1.0"?>
<ValueMetadata>
```

```
    <Version>3.02</Version>
    <CreationDateTime>01/26/2011 00:00:00</CreationDateTime>
    <TestID>MED:ROUTE</TestID>
    <TestName>Medication Route</TestName>
    <DataType>Enum</DataType>
    <Flagstouse></Flagstouse>
    <Oktousevalues></Oktousevalues>
    <EnumValues>
        <Val description="">inhalation</Val>
        <Val description="">injection</Val>
        <Val description="Intravenous">IV</Val>
        <Val description="By Mouth">PO</Val>
        <Val description="By Rectum">PR</Val>
        <Val description="">topical</Val>
        <Val description="">transdermal</Val>
    </EnumValues>
    <UnitValues></UnitValues>
</ValueMetadata>
```

PHARMACY ID

```
<?xml version="1.0"?>
    <ValueMetadata>
    <Version>3.02</Version>
    <CreationDateTime>01/26/2011 00:00:00</CreationDateTime>
    <TestID>MED:NCPDPID</TestID>
    <TestName>NCPDP Provider ID</TestName>
    <DataType>String</DataType>
    <MaxStringLength>255</MaxStringLength>
    <Flagstouse></Flagstouse>
    <Oktousevalues></Oktousevalues>
    <EnumValues></EnumValues>
    <UnitValues></UnitValues>
</ValueMetadata>
```

PHARMACY BENEFITS MANAGER (PBM) NUMBER

```
<?xml version="1.0"?>
<ValueMetadata>
    <Version>3.02</Version>
    <CreationDateTime>01/26/2011 00:00:00</CreationDateTime>
    <TestID>MED:PBM</TestID>
    <TestName>Pharmacy Benefits Manager Number</TestName>
```

<DataType>String</DataType>

<MaxStringLength>255</MaxStringLength>

<Flagstouse></Flagstouse>

<Oktousevalues></Oktousevalues>

<EnumValues></EnumValues>

<UnitValues></UnitValues>

</ValueMetadata>

QUANTITY OF PILLS DISPENSED:

<?xml version="1.0"?>

<ValueMetadata>

<Version>3.02</Version>

<CreationDateTime>01/26/2011 00:00:00</CreationDateTime>

<TestID>MED:DSPQ</TestID>

<TestName>Quantity Dispensed</TestName>

<DataType>PosFloat</DataType>

<Flagstouse></Flagstouse>

<Oktousevalues>Y</Oktousevalues>

<EnumValues></EnumValues>

<UnitValues>

<NormalUnits>tablets</NormalUnits>

</UnitValues>

</ValueMetadata>

NUMBER OF DAYS SUPPLY GIVEN:

<?xml version="1.0"?>

<ValueMetadata>

<Version>3.02</Version>

<CreationDateTime>01/26/2011 00:00:00</CreationDateTime>

<TestID>MED:DDS</TestID>

<TestName>Number of Days Supply Given</TestName>

<DataType>PosFloat</DataType>

<Flagstouse></Flagstouse>

<Oktousevalues>Y</Oktousevalues>

<EnumValues></EnumValues>

<UnitValues>

<NormalUnits>days</NormalUnits>

</UnitValues>

</ValueMetadata>