DDI4 and tall data layouts

## Prologue

DDI4 has the potential to model data arranged in a tall format. To do so it will need a couple of additions. This document gives some examples to explain what is meant by a “Tall” format and then includes a class diagram showing four proposed classes that would enable DDI4 to model Tall layouts.

An appendix shows what DDI4 XML for one of the examples would look like with the new classes.

## Examples

First though - examples of the difference between the traditional “wide” data format which is modeled in DDI4 with a UnitDataRecord and the “tall” layout which is sometimes used for certain analytical procedures and for data where the set of measurements needs to be flexible (like event data). Both examples use the same underlying set of measurements.

## Wide

The wide example table shows a set of three blood pressure measurements.

Table 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| patient | measurementDatetime | systolic | diastolic | position |
| 34BG21 | 26Dec2000 6:00:00 | 110 | 70 | 1 |
| 4yAD93 | 25Dec2000 0:00:00 | 185 | 100 | 1 |
| 4yAD93 | 25Dec2000 0:15:00 | 120 | 80 | 2 |

The variables at collection performed certain roles:

* patient – identifier
* measurementDatetime – identifier
* systolic – measure of interest
* diastolic – measure of interest
* position – measurement attribute (1=sitting, 2=standing)

## Tall

The exact same set of measurements could be transformed into a “tall” representation as in the table below.

Table 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| patient | measurementDatetime | Source | Label | Value1 |
| 34BG21 | 26Dec2000 6:00:00 | systolic | systolic pressure in mm Hg | 110 |
| 34BG21 | 26Dec2000 6:00:00 | diastolic | diastolic pressure in mm Hg | 70 |
| 34BG21 | 26Dec2000 6:00:00 | position | Position during measurement | 1 |
| 4yAD93 | 25Dec2000 0:00:00 | systolic | systolic pressure in mm Hg | 185 |
| 4yAD93 | 25Dec2000 0:00:00 | diastolic | diastolic pressure in mm Hg | 100 |
| 4yAD93 | 25Dec2000 0:00:00 | position | Position during measurement | 1 |
| 4yAD93 | 25Dec2000 0:15:00 | systolic | systolic pressure in mm Hg | 120 |
| 4yAD93 | 25Dec2000 0:15:00 | diastolic | diastolic pressure in mm Hg | 80 |
| 4yAD93 | 25Dec2000 0:15:00 | position | Position during measurement | 2 |

Variables in this case also serve certain roles.

* patient – identifier
* measurementDatetime – identifier
* Label – measurement attribute
* Source – both an identifier and a special measurement attribute, a **code** which indicates a specific logical and physical representation for an InstanceVariable
* Value1 – a special kind of variable, a **generic** representation with the value domain dependent on the value in Source.

The special nature of Value1 above can be seen more clearly below if the original position variable had had a text representation.

Many software platforms would not allow the direct transformation from wide to tall with mixed numeric and character value domains. We can work around this by representing all values in “Value2” below as generic text, Note that this is commonly done when data are written to something like a csv file.

Table 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| patient | measurementDatetime | Source | Label | Value2 |
| 34BG21 | 26dec2000:06:00:00 | systolic | systolic pressure in mm Hg | 110 |
| 34BG21 | 26dec2000:06:00:00 | diastolic | diastolic pressure in mm Hg | 70 |
| 34BG21 | 26dec2000:06:00:00 | position | Position during measurement | sitting |
| 4yAD93 | 25dec2000:00:00:00 | systolic | systolic pressure in mm Hg | 185 |
| 4yAD93 | 25dec2000:00:00:00 | diastolic | diastolic pressure in mm Hg | 100 |
| 4yAD93 | 25dec2000:00:00:00 | position | Position during measurement | sitting |
| 4yAD93 | 25dec2000:00:15:00 | systolic | systolic pressure in mm Hg | 120 |
| 4yAD93 | 25dec2000:00:15:00 | diastolic | diastolic pressure in mm Hg | 80 |
| 4yAD93 | 25dec2000:00:15:00 | position | Position during measurement | supine |

As of the prototype for DDI4, we do not have the ability to specify that a code for variable “Source” above represents the description of the associated “Value2” value.

We could do this by allowing a code in “Source” above to represent a **DDI4 ValueMapping**. A ValueMapping formats a DataPoint which, in turn, is described by an InstanceVarlable. We currently use a ValueMapping to describe the physical layout of a column in a physical unit record (like a csv). This would allow us to specify the mapping from text to the true value domain: that the 110 above is numeric, has a certain scale, etc., as well as the properties defined by the associated InstanceVariable (the units of measurement, the associated Concept, etc. ).

We would also need to be able to describe in the table above that “Value2” was described be “Source”. We could do this with a VariableRelation with a semantic describing that “Source” contains the ValueMapping for “Value2”

This example points out why we needed the ValueMapping for physical representations of unit record data. All of the “110” representations of a designation above represent the same “signified” value - the systolic pressure measured on patient 34BG21 at 26dec2000:06:00:00. A common InstanceVariable can describe the Act and the Capture, the unit of measurement, the intended datatype.

This might show a problem, though, in including “physicalDataType” at the InstanceVariable level.

What we don’t have explicitly in the model is an instantiable class for the “signified”. It is not a DataPoint since we can see that it shows up in different DataPoints in different layouts. It is not a Datum as we’ve defined it, since that is defined as a designation. Each of the tables above has a different “110” Datum (as we’ve defined it) for the same signified associated with it.

# Possible Additions to the DDI4 Prototype to Handle Tall Skinny

The model below describes potential new elements to achieve description of Tall skinny layouts like in the table 3. A UML model diagram is followed by possible XML to describe the table.

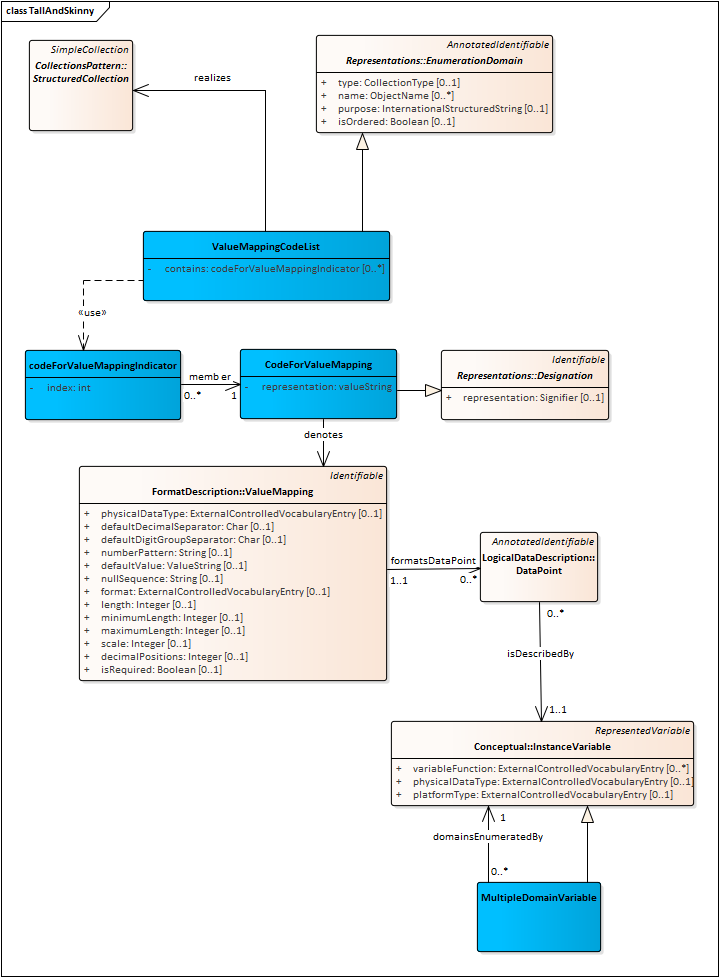
The classes in blue in the figure below do not currently exist in the DDI4 prototype.

**ValueMappingCodeList** – This class is a Collection to contain CodeForValueMapping objects. It defines the value domain of columns like the Source column in the examples above. This clas extends an EnumerationDomain.

**CodeForValueMapping -** This class allows the definition of a code that points to a ValueMapping. The ValueMapping, in turn, references a DataPoint which references an InstanceVariable. The ValueMapping describes the physical representation of a value (e.g. decimal separator, number pattern) . The InstanceVariable contains information about the meaning of the value ( e.g. Concept, units of measurement).

**CodeForValueMappingIndicator –** This class is part of the plumbing of the collections pattern. It currently indicates membership and order of objects in a Collection. It may disappear if the Collections Pattern is updated.

**MultipleDomainVariable** – This class allows for a column like the Value1 or Value2 columns in the examples above. It allows for the association with an InstanceVariable with a SubstantativeValueDomain of a ValueMappingCodeList that defines the meaning and value domain of each of its generic values.



### XML for table 3

In the XML below the classes not currently in the DDI4 Prototype are shown in green. UnitDataViewPoint may have the wrong semantic for this but the content works.

<?xml version="1.0" encoding="UTF-8"?>  
<DDI xmlns="urn:ddi.org:4"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="urn:ddi.org:4 file:/C:/schemas/DDI4/ddi4\_2019\_04\_17/ddi-alliance-ddi-views-61893f02fc7e/bindings/xsd/DDI\_4-DR0.2.xsd">  
 <DocumentInformation>  
 <Agency>example.org</Agency>  
 <Id>IDtallExample</Id>  
 <Version>1</Version>  
 <OfType>DR0.2</OfType>  
 </DocumentInformation>  
   
   
  
   
 <IdentifierRole>  
 <Agency>example.org</Agency>  
 <Id>IdIdentifiers</Id>  
 <Version>1</Version>  
 <Contains>urn:ddi:example.org:IDpatient:1</Contains>  
 <Contains>urn:ddi:example.org:IdMeasurementDatetime:1</Contains>  
 </IdentifierRole>  
 <MeasureRole>  
 <Agency>example.org</Agency>  
 <Id>IdMeasures</Id>  
 <Version>1</Version>  
 <Contains>urn:ddi:example.org:IdValue2:1</Contains>  
 </MeasureRole>  
 <AttributeRole>  
 <Agency>example.org</Agency>  
 <Id>IdMeasures</Id>  
 <Version>1</Version>  
 <Contains>urn:ddi:example.org:IdSource:1</Contains>  
 </AttributeRole>  
   
 <UnitDataViewpoint>  
 <Agency></Agency>  
 <Id>IdViewPoint</Id>  
 <Version></Version>  
 <HasIdentifierRole typeOfClass="IdentifierRole">urn:ddi:example.org:IdIdentifiers:1</HasIdentifierRole>  
 <HasMeasureRole typeOfClass="MeasureRole">urn:ddi:example.org:IdMeasures</HasMeasureRole>  
 <HasAttributeRole typeOfClass="AttributeRole">urn:ddi:example.org:IdAttributes</HasAttributeRole>  
 </UnitDataViewpoint>  
   
 <!-- Traditional variable type. systolic -->  
 <InstanceVariable>  
 <Agency>example.org</Agency>  
 <Id>IdSystolic</Id>  
 <Version>1</Version>  
 <Name><Content>systolic</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdSystolicDomain:1  
 </TakesSubstantiveValuesFrom>  
 </InstanceVariable>  
 <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdSystolicDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe a systolic pressure</LanguageSpecificStructuredString></DisplayLabel>  
 </SubstantiveValueDomain>   
   
 <!-- Traditional variable type. diastolic -->  
 <InstanceVariable>  
 <Agency>example.org</Agency>  
 <Id>IdDiastolic</Id>  
 <Version>1</Version>  
 <Name><Content>Diastolic</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdDiastolicDomain:1  
 </TakesSubstantiveValuesFrom>  
 </InstanceVariable>  
 <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdDiastolicDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe a Diastolic pressure</LanguageSpecificStructuredString></DisplayLabel>  
 </SubstantiveValueDomain>   
  
  
 <!-- Traditional variable type. position -->  
 <InstanceVariable>  
 <Agency>example.org</Agency>  
 <Id>IdPosition</Id>  
 <Version>1</Version>  
 <Name><Content>Position</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdPositionDomain:1  
 </TakesSubstantiveValuesFrom>  
 </InstanceVariable>  
 <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdPositionDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe a patient position during measurement</LanguageSpecificStructuredString></DisplayLabel>  
 </SubstantiveValueDomain>   
   
  
  
 <!-- Traditional variable type. patient -->  
 <InstanceVariable>  
 <Agency>example.org</Agency>  
 <Id>IDpatient</Id>  
 <Version>1</Version>  
 <Name><Content>patient</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdPatientDomain:1  
 </TakesSubstantiveValuesFrom>  
 </InstanceVariable>  
 <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdPatientDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe a patient id</LanguageSpecificStructuredString></DisplayLabel>  
 </SubstantiveValueDomain>  
   
 <!-- Traditional variable type. measurementDatetime -->  
 <InstanceVariable>  
 <Agency>example.org</Agency>  
 <Id>IdMeasurementDatetime</Id>  
 <Version>1</Version>  
 <Name><Content>measurementDatetime</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdMeasurementDatetimeDomain:1  
 </TakesSubstantiveValuesFrom>  
 </InstanceVariable>  
 <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdMeasurementDatetimeDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe a measurementDateTime id</LanguageSpecificStructuredString></DisplayLabel>  
 </SubstantiveValueDomain>  
   
 <!-- Traditional variable type Label -->  
 <InstanceVariable>  
 <Agency>example.org</Agency>  
 <Id>IdLabel</Id>  
 <Version>1</Version>  
 <Name><Content>Source</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdTextDomain:1  
 </TakesSubstantiveValuesFrom>  
 </InstanceVariable>  
 <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdTextDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe text </LanguageSpecificStructuredString></DisplayLabel>  
 </SubstantiveValueDomain>  
   
  
 <!-- New variable type Value2 -->  
 **<MultipleDomainVariable>  
 <Agency>example.org</Agency>  
 <Id>IdLabel</Id>  
 <Version>1</Version>  
 <Name><Content>Source</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdTextDomain:1  
 </TakesSubstantiveValuesFrom>  
 <DomainsEnumeratedBy type="InstanceVariable">urn:ddi:example.org:IdSource:1</DomainsEnumeratedBy>  
 </MultipleDomainVariable>** <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdTextDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe text </LanguageSpecificStructuredString></DisplayLabel>  
 <DescribedBy>urn:ddi:example.org:IdSource:1</DescribedBy>  
 </SubstantiveValueDomain>

<!-- New variable type. Source -->  
 <InstanceVariable>  
 <Agency>example.org</Agency>  
 <Id>IdSource</Id>  
 <Version>1</Version>  
 <Name><Content>Source</Content></Name>  
 <TakesSubstantiveValuesFrom typeOfClass="SubstantiveValueDomain">  
 urn:ddi:example.org:IdSourceDomain:1  
 </TakesSubstantiveValuesFrom>  
 </InstanceVariable>  
 <SubstantiveValueDomain>  
 <Agency>example.org</Agency>  
 <Id>IdSourceDomain</Id>  
 <Version>1</Version>  
 <DisplayLabel><LanguageSpecificStructuredString>This has elements to describe a variable that points to ValueMappings </LanguageSpecificStructuredString></DisplayLabel>  
 <EnumeratedValueDomain typeOfClass="ValueMappingCodeList">urn:ddi:example.org:IdValueMappingCodeList:1</EnumeratedValueDomain>  
 </SubstantiveValueDomain>  
  
 <ValueMappingCodeList>  
 <Agency></Agency>  
 <Id>IdValueMappingCodeList</Id>  
 <Version>1</Version>   
 <Contains><Index>1</Index><Member typeOfClass="CodeForValueMapping">urn:ddi:example.org:IdSystolicCode:1</Member></Contains>  
 <Contains><Index>2</Index><Member typeOfClass="CodeForValueMapping">urn:ddi:example.org:IdDiastolicCode:1</Member></Contains>  
 <Contains><Index>3</Index><Member typeOfClass="CodeForValueMapping">urn:ddi:example.org:IdPositionCode:1</Member></Contains>   
 </ValueMappingCodeList>  
   
 <CodeForValueMapping>  
 <representation>systolic</representation>  
 <denotes type="ValueMapping">urn:ddi:example.org:IdValuemappingSystolic</denotes>  
 </CodeForValueMapping>  
 <CodeForValueMapping>  
 <representation>diastolic</representation>  
 <denotes type="ValueMapping">urn:ddi:example.org:IdValuemappingDiastolic</denotes>  
 </CodeForValueMapping>  
 <CodeForValueMapping>  
 <representation>position</representation>  
 <denotes type="ValueMapping">urn:ddi:example.org:IdValuemappingPosition</denotes>  
 </CodeForValueMapping>  
   
 <ValueMapping>  
 <Agency>example.org</Agency>  
 <Id>IdValuemappingSystolic</Id>  
 <Version>1</Version>  
 <FormatsDataPoint typeOfClass="DataPoint">urn:ddi:example.org:IdSystolicDataPoint:1</FormatsDataPoint>  
 </ValueMapping>  
 <DataPoint>  
 <Agency>example.org</Agency>  
 <Id>IdSystolicDataPoint</Id>  
 <Version>1</Version>  
 <IsDescribedBy typeOfClass="InstanceVariable">urn:ddi:example.org:IdSystolic:1</IsDescribedBy>  
 </DataPoint>   
   
 <ValueMapping>  
 <Agency>example.org</Agency>  
 <Id>IdValuemappingdiastolic</Id>  
 <Version>1</Version>  
 <FormatsDataPoint typeOfClass="DataPoint">urn:ddi:example.org:IdDiastolicDataPoint:1</FormatsDataPoint>  
 </ValueMapping>  
 <DataPoint>  
 <Agency>example.org</Agency>  
 <Id>IdDiastolicDataPoint</Id>  
 <Version>1</Version>  
 <IsDescribedBy typeOfClass="InstanceVariable">urn:ddi:example.org:IdDiastolic:1</IsDescribedBy>  
 </DataPoint>   
   
 <ValueMapping>  
 <Agency>example.org</Agency>  
 <Id>IdValuemappingPosition</Id>  
 <Version>1</Version>  
 <FormatsDataPoint typeOfClass="DataPoint">urn:ddi:example.org:IdPositionDataPoint:1</FormatsDataPoint>  
 </ValueMapping>  
 <DataPoint>  
 <Agency>example.org</Agency>  
 <Id>IdPositionDataPoint</Id>  
 <Version>1</Version>  
 <IsDescribedBy typeOfClass="InstanceVariable">urn:ddi:example.org:IdPosition:1</IsDescribedBy>  
 </DataPoint>   
   
   
</DDI>

Brief notes from 2019-03-24 meeting

How does source relate to a dimension? Or is it like the measure?

Is Value like the Value in DDI3 for inline data? How is that described?

Analogy to Data cube

* Role, (dimension, measure, attribute)
* Gender (m, f) age( y,o)

Is there an analogy to the OSI stack in tying together the above with data cubes.

# Appendix

Many software packages have the capability to convert between wide and tall layouts. In SAS below that employs the generalized transpose facility.

## SAS code to generate tables 1 an 2

**Proc** **format**;

value pos

**1**="sitting"

**2**="supine"

;

/\* example rectangular file \*/

**data** bpWideNumeric;

input patient $ measurementDatetime :datetime18. systolic diastolic position;

label patient="ID of patient"

MeasurementDatetime="DateTime of measurement"

systolic="systolic pressure in mm Hg"

diastolic="diastolic pressure in mm Hg"

position="Position during measurement";

format position pos.;

format measurementDatetime datetime18.;

datalines;

4yAD93 25dec2000:00:00:00 185 100 1

4yAD93 25dec2000:00:15:00 120 80 2

34BG21 26dec2000:06:00:00 110 70 1

;

**run**;

**proc** **sort** data=work.bpWideNumeric;

by patient measurementDatetime;

**PROC** **TRANSPOSE** DATA=bpWideNumeric

OUT=WORK.bpTallNumeric (LABEL="Transposed WORK.bpWideNumeric")

PREFIX=Value

NAME=Source

LABEL=Label

;

BY patient MeasurementDatetime;

VAR systolic diastolic position;

;

## SAS code to generate table 3

**Proc** **format**;

value pos

**1**="sitting"

**2**="supine"

;

/\* example rectangular file as text \*/

**data** bpWideText;

input patient $ measurementDatetime :$18. systolic $ diastolic $ position $;

label patient="ID of patient"

MeasurementDatetime="DateTime of measurement"

systolic="systolic pressure in mm Hg"

diastolic="diastolic pressure in mm Hg"

position="Position during measurement";

datalines;

4yAD93 25dec2000:00:00:00 185 100 sitting

4yAD93 25dec2000:00:15:00 120 80 supine

34BG21 26dec2000:06:00:00 110 70 sitting

;

**run**;

**proc** **sort** data=work.bpWideText;

by patient measurementDatetime;

**PROC** **TRANSPOSE** DATA=bpWideText

OUT=WORK.bpTallText(LABEL="Transposed WORK.bpWideText")

PREFIX=Value

NAME=Source

LABEL=Label

;

BY patient MeasurementDatetime;

VAR systolic diastolic position;

;