

# Mapping DDI2 to DDI4, Larry Hoyle and Joachim Wackerow

- This poster describes a mapping of leaf DDI2 elements into corresponding DDI4 properties. The mapping can be described as a multi-column, machine actionable, table.
- DDI2 and the later versions differ in the underlying reliance on reusable objects. This makes mapping from DDI2 to DDI4 complicated. Software using this table can also collapse content that is repeated in DDI2 (like CodeLists) into single reused DDI4 objects.
- The table is derived from a spreadsheet listing DDI2 elements important to CESSDA. That table was further refined at the DDI4 Norway Sprint. The table below right explains the role of each column in the mapping table. Those columns contain the information needed to move information from one model to the other. Examples in the table below are color coded to corresponding cells in the table above.

The example below is taken from the European Social Survey file ESS1e06.6.xml

Sample rows from the mapping table. The full table currently lists 181 DDI25 XPath. This table can grow as new element types are found in DDI2 instances.

DDI25	DDI4PropertyName	IdentifiablesMapping	AbstractSubstitution	ParameterValues	IsIdRef	DDI4IdRefPath	DDI4IdRefAppend	Notes
/codeBook/dataDscr/var/anlysUnit	Study/hasInstanceVariable/usesUnitType/definition	/codeBook:Study, /codeBook/dataDscr/var:InstanceVariable, /codeBook/dataDscr/var/anlysUnit:UnitType						
/codeBook/dataDscr/var/qstn/preQtxt	Study/hasInstanceVariable/sourceCapture/hasInstruction/instructionText/textContent/text/content	/codeBook:Study, /codeBook/dataDscr/var:InstanceVariable, /codeBook/dataDscr/var/qstn:RepresentedQuestion, /codeBook/dataDscr/var/qstn:Instruction	Capture: RepresentedQuestion, DynamicTextContent: LiteralText	RepresentedQuestion/hasInstruction/instructionText/textContent/textContent/purpose/languageSpecificStructuredString/content='PreQuestionText'				
/codeBook/dataDscr/var/@ID	Study/hasInstanceVariable/localId/localIdValue	/codeBook:Study, /codeBook/dataDscr/var:InstanceVariable			FALSE			Program will need to match DDI2 variable ID to DDI4 DdiUrn for references to the variable like varGrp
/codeBook/dataDscr/varGrp/@var	Study/hasVariableCollection/hasAnnotation/summary/languageSpecificString/content	/codeBook:Study, /codeBook/dataDscr/varGrp:VariableCollection		Study/hasVariableCollection/hasAnnotation/summary/languageSpecificString/scope='ddi2 variable ids'	TRUE	contains/member	TRUE, FALSE	requires matching DDI2 ID with DDI4 DdiUrn and inserting DDI4URN references into the VariableCollection
/codeBook/dataDscr/var/Lab1	Study/hasInstanceVariable/displayLabel/languageSpecificStructuredString/content	/codeBook:Study, /codeBook/dataDscr/var:InstanceVariable						

## DDI2 XML

### DDI2 varGrp "Weights"

```
<varGrp ID="WEIGHTS" type="other" var="V564 V565 V566">
  <lab1> Weights </lab1>
</varGrp>
```

### DDI2 var "V564"

```
<var ID="V564" name="dweight"
wgt="wgt"
files="F1" dcml="2" intrvl="contin">
  <location width="4"/>
  <lab1> Design weight </lab1>
  <qstn>
    <qstnLit> R34 Design weight </qstnLit>
  </qstn>
  <valrng>
    <range UNITS="REAL"
      min="0.0044"max="4.34"/>
  </valrng>
  <sumStat type="vald"> 42359
  </sumStat>
  <sumStat type="invd"> 0 </sumStat>
  <varFormat type="numeric"
    schema="other"/>
</var>
```

### DDI2 var "V565"

```
<var ID="V565" name="pspwght"
wgt="wgt" files="F1"
dcml="2" intrvl="contin">
  <location width="4"/>
  <lab1> Post-stratification weight including design weight </lab1>
  <qstn>
    <qstnLit> R35 Post-stratification weight including design weight </qstnLit>
  </qstn>
  <valrng>
    <range UNITS="REAL"
      min="0.000750077458907839"
      max="6.85496650535486"/>
  </valrng>
  <sumStat type="vald"> 42359 </sumStat>
  <sumStat type="invd"> 0 </sumStat>
  <varFormat type="numeric" schema="other"/>
</var>
```

## DDI4 Objects

VariableCollection object  
Id = 84f810fb-7555-4bb8-af88-2148baca7d85

InstanceVariable object  
Id = b17688c1-97a9-41f0-8ff1-4b7fe367c3f3  
Name = "dweight"  
displayLabel =

LabelForDisplay StructuredDatatype  
languageSpecificStructuredString =  
locationVariant =  
validDates =  
maxLength =

LanguageSpecificStructuredStringType  
StructuredDatatype  
Content = "Design weight"  
language =  
scope =  
...  
isPlainText =  
structureUsed =  
otherDefined =

InstanceVariable object  
Id = b10f7a22-97f6-4971-99cf-481e2cc12367

VariableStatistics object  
Id = 00091102-8197-4387-b738-a1bd8eb9c736  
hasSummaryStatistic =

SummaryStatistic StructuredDatatype  
Content = "Design weight"  
typeOfStatistic =  
hasStatistic =

Statistic StructuredDatatype  
Content = "Design weight"  
content = "42359"  
isWeighted =  
computationBase =  
typeOfNumericValue =

## Column Descriptions for the Table Above

Column	Function	Example	Details
DDI25	The unpredicated XPath of a DDI2 text or attribute node	/codeBook/dataDscr/var/anlysUnit	Each piece of information to be imported from DDI2 should have a corresponding XPath listed.
DDI4PropertyName	The corresponding path to a leaf in DDI4	Study/hasInstanceVariable/usesUnitType/definition	The first node in this path is a DDI4 class. The remaining nodes are properties in a chain down to a leaf value. The value of some properties are references to other objects, that object may need to be created. Other values are "structured datatypes".
IdentifiablesMapping	This maps a DDI2 sub-path to a DDI4 Identifiable class. An object of that class will need to be created for each unique instance of that DDI2 sub-path.	/codeBook:Study, /codeBook/dataDscr/var:InstanceVariable, /codeBook/dataDscr/var/anlysUnit:UnitType	In the example to the left, for each unique var element in a DDI2 instance, the same DDI4 InstanceVariable needs to be used. A predicated XPath identifies a specific DDI4 object, e.g. /codeBook[1]/dataDscr[2]/var[7] indicates a specific InstanceVariable. In this example the 7th variable in the 2nd dataDscr element of the codebook always maps to the same reusable InstanceVariable.
AbstractSubstitution	Some references in the DDI4 model are to abstract classes. In these cases it is necessary to specify which extension of the abstract class should be used in the mapping.	Capture:RepresentedQuestion, DynamicTextContent:LiteralText	In this example a sourceCapture associates with the abstract class Capture. The mapping will use the RepresentedQuestion extension of the Capture.
ParameterValues	Some values in DDI4 can use additional explanatory metadata. This column lists the path and the value for that information.	RepresentedQuestion/hasInstruction/instructionText/textContent/purpose/languageSpecificStructuredString/content='PreQuestionText'	The LiteralText above is further described as "PreQuestionText"
IsIdRef	Is this value a reference to an ID in the DDI2 XML (an xs:IDREF)?, if so this will ultimately need to be transformed into a proper reference in DDI4 through a DDI URN.	TRUE	An example is the @var attribute of the DDI2 varGrp. This will need to be implemented as a reference to an InstanceVariable in a VariableCollection in DDI4.
DDI4IdRefPath	This is the sub-path within the last DDI4 identifiable object for the DDI URN of the referenced object.	contains/member	In the case of a varGrp, the VariableCollection has a contains/member value that is the URN of the DDI4 InstanceVariable created to match the DDI2 var referenced by the @var IDREF.
DDI4IdRefAppend	This describes whether to append or replace values in the DDI4 path.	TRUE, FALSE	In the case above, there may be more than one member property under contains, but there can only be one value for member.
Notes	Used to describe any notes for the mapping	requires matching DDI2 ID with DDI4 DdiUrn and inserting DDI4URN references into the VariableCollection	In this example, it describes what is needed in the matching process of DDI2 IDs and DDI4 URNs.

- DDI25, DDI4PropertyName, and IdentifiablesMapping columns from DDI2 anlysUnit
- AbstractSubstitution and ParameterValues from DDI2 preQtxt
- IsIdRef, DDI4IdRefPath, DDI4IdRefAppend, and Notes from DDI2 varGrp/@var

## Post Processing

- As DDI2 instances are processed, XPaths of leaf elements with content that do not have an entry in the mapping table can be recorded for possible inclusion in future versions of the mapping table.
- Some elements in DDI2, like varGrp, will make reference to elements that appear later in the document. This means that assigning InstanceVariables to a VariableCollection or InstanceVariables to VariableStatistics should be done after a first pass through the DDI2 document.
- DDI4 allows reuse. Another type of post-processing can collapse duplicated content like repeated codelists into references to just one instance. This process might require parameters to dictate how aggressive the collapsing is. For example should all identical categories be collapsed into a single one, or only those together in duplicate CategorySets?
- We're working on adding some of these features to the DDI4 R package.