

# Package: rddi (via r-universe)

November 1, 2024

**Title** R Interface to DDI Codebook 2.5

**Version** 0.1.1

**Date** 2022-10-14

**Description** A direct interface to the underlying XML representation of DDI Codebook 2.5 with flexible API creation.

**Depends** R (>= 3.4.0)

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.1

**Roxygen** list(markdown = TRUE)

**Suggests** testthat, covr, knitr, rmarkdown, magrittr

**Imports** rlang, glue, xml2

**VignetteBuilder** knitr

**Config/Needs/readme** rmarkdown

**Repository** <https://nyuglobalties.r-universe.dev>

**RemoteUrl** <https://github.com/nyuglobalties/rddi>

**RemoteRef** HEAD

**RemoteSha** 0947df15a000fb8e162274f0db50a0d2342843d1

## Contents

as_ddi . . . . .	3
as_xml . . . . .	4
as_xml_string . . . . .	5
ddi_anlyInfo . . . . .	5
ddi_anlysUnit . . . . .	7
ddi_boundPoly . . . . .	7
ddi_catgry . . . . .	9
ddi_citation . . . . .	10

ddi_codeBook . . . . .	12
ddi_codingInstructions . . . . .	13
ddi_cohort . . . . .	14
ddi_concept . . . . .	15
ddi_contact . . . . .	16
ddi_controlledVocabUsed . . . . .	17
ddi_dataAccs . . . . .	18
ddi_dataColl . . . . .	19
ddi_dataDscr . . . . .	23
ddi_dataFingerprint . . . . .	24
ddi_dataItem . . . . .	25
ddi_dataSrc . . . . .	26
ddi_derivation . . . . .	27
ddi_developmentActivity . . . . .	28
ddi_dimensns . . . . .	30
ddi_distStmt . . . . .	31
ddi_dmns . . . . .	33
ddi_docDscr . . . . .	34
ddi_docSrc . . . . .	36
ddi_embargo . . . . .	37
ddi_exPostEvaluation . . . . .	38
ddi_fileDscr . . . . .	39
ddi_fileStrc . . . . .	40
ddi_fileTxt . . . . .	41
ddi_frameUnit . . . . .	43
ddi_geoBndBox . . . . .	44
ddi_imputation . . . . .	46
ddi_labl . . . . .	47
ddi_location . . . . .	48
ddi_locMap . . . . .	49
ddi_method . . . . .	50
ddi_mrow . . . . .	51
ddi_nCube . . . . .	52
ddi_nCubeGrp . . . . .	54
ddi_notes . . . . .	55
ddi_otherMat . . . . .	56
ddi_othrStdyMat . . . . .	57
ddi_point . . . . .	59
ddi_polygon . . . . .	60
ddi_prodStmt . . . . .	61
ddi_producer . . . . .	63
ddi_qstn . . . . .	64
ddi_qualityStatement . . . . .	66
ddi_range . . . . .	67
ddi_recGrp . . . . .	68
ddi_resource . . . . .	69
ddi_respUnit . . . . .	70
ddi_row . . . . .	71

ddi_rspStmt . . . . .	72
ddi_sampleFrame . . . . .	73
ddi_security . . . . .	75
ddi_serStmt . . . . .	76
ddi_setAvail . . . . .	77
ddi_sources . . . . .	79
ddi_srcChar . . . . .	80
ddi_srcDocu . . . . .	80
ddi_srcOrig . . . . .	81
ddi_standard . . . . .	82
ddi_standardsCompliance . . . . .	83
ddi_stdyDscr . . . . .	84
ddi_stdyInfo . . . . .	85
ddi_studyAuthorization . . . . .	87
ddi_studyDevelopment . . . . .	88
ddi_subject . . . . .	89
ddi_sumDscr . . . . .	90
ddi_table . . . . .	92
ddi_targetSampleSize . . . . .	93
ddi_tbody . . . . .	94
ddi_tgroup . . . . .	95
ddi_thead . . . . .	96
ddi_titl . . . . .	97
ddi_titlStmt . . . . .	98
ddi_txt . . . . .	99
ddi_universe . . . . .	100
ddi_usage . . . . .	101
ddi_useStmt . . . . .	103
ddi_valrng . . . . .	105
ddi_var . . . . .	107
ddi_varGrp . . . . .	110
ddi_verStmt . . . . .	111
validate_codebook . . . . .	112

**Index****114**

as\_ddi

*Convert XML trees to DDI objects***Description**

Convert XML trees to DDI objects

**Usage**

as\_ddi(x, ...)

**Arguments**

x                    An xml\_node object.  
...                   Arguments to pass to methods.

**Value**

The DDI equivalent of the XML tree.

---

as\_xml                    *Get XML representation of ddi\_node objects*

---

**Description**

Get XML representation of ddi\_node objects

**Usage**

```
as_xml(x, ...)
```

**Arguments**

x                    A ddi\_node object (or subclass).  
...                   Arguments to pass to methods.

**Value**

An xml\_document or xml\_node object whether the object is a root node or not, respectively.

**Examples**

```
cb <- ddi_codeBook(ddi_stdyDscr(ddi_citation(ddi_titlStmt(ddi_titl("Sample")))))  
as_xml(cb)
```

---

as_xml_string	<i>Shortcut to text representation of DDI XML</i>
---------------	---

---

**Description**

Functionally equivalent to `as.character(as_xml(ddi_node_obj))`.

**Usage**

```
as_xml_string(x, ...)
```

**Arguments**

x	A ddi_node object.
...	Arguments forwarded to <code>as_xml()</code> .

**Value**

A string containing the text representation of XML.

**Examples**

```
cb <- ddi_codeBook(ddi_stdyDscr(ddi_citation(ddi_titlStmt(ddi_titl("Sample")))))
as_xml_string(cb)
```

---

ddi_anlyInfo	<i>anlyInfo and its child nodes</i>
--------------	-------------------------------------

---

**Description**

Information on data appraisal.

**Usage**

```
ddi_anlyInfo(...)
```

```
ddi_dataAppr(...)
```

```
ddi_EstSmpErr(...)
```

```
ddi_respRate(...)
```

**Arguments**

...	Child nodes or attributes.
-----	----------------------------

## Details

### *Parent nodes*

anlyInfo is contained in method.

### *anlyInfo specific child nodes*

- `ddi_dataAppr()` are other issues pertaining to data appraisal. Describe here issues such as response variance, nonresponse rate and testing for bias, interviewer and response bias, confidence levels, question bias, etc. Attribute type allows for optional typing of data appraisal processes and option for controlled vocabulary.
- `ddi_EstSmpErr()` are estimates of sampling error. This element is a measure of how precisely one can estimate a population value from a given sample.
- `ddi_respRate()` is the response rate. The percentage of sample members who provided information. This may include a broader description of stratified response rates, information affecting response rates etc.

## Value

A `ddi_node` object.

## References

[anlyInfo documentation](#)

[dataAppr documentation](#)

[respRate documentation](#)

[EstSmpErr documentation](#)

## Examples

```
ddi_anlyInfo()

# Functions that need to be wrapped in ddi_anlyInfo()

ddi_dataAppr("These data files were obtained from the United States House of
  Representatives, who received them from the Census Bureau
  accompanied by the following caveats...")

ddi_EstSmpErr("To assist NES analysts, the PC SUDAAN program was used to
  compute sampling errors for a wide-ranging example set of
  proportions estimated from the 1996 NES Pre-election Survey
  dataset...")

ddi_respRate("For 1993, the estimated inclusion rate for TEDS-eligible
  providers was 91 percent, with the inclusion rate for all
  treatment providers estimated at 76 percent (including privately
  and publicly funded providers).")
```

---

ddi_anlysUnit	<i>anlysUnit node</i>
---------------	-----------------------

---

### Description

Provides information regarding whom or what the variable/nCube describes. The element may be repeated only to support multiple language expressions of the content. More information on this element, especially its allowed attributes, can be found in the references.

### Usage

```
ddi_anlysUnit(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

anlysUnit is contained in nCube and var.

### Value

A ddi\_node object.

### References

[anlysUnit documentation](#)

### Examples

```
ddi_anlysUnit("This variable reports election returns at the constituency level.")
```

---

ddi_boundPoly	<i>boundPoly and its child nodes</i>
---------------	--------------------------------------

---

## Description

The geographic bounding polygon field allows the creation of multiple polygons to describe in a more detailed manner the geographic area covered by the dataset. It should only be used to define the outer boundaries of a covered area. For example, in the United States, such polygons can be created to define boundaries for Hawaii, Alaska, and the continental United States, but not interior boundaries for the contiguous states. This field is used to refine a coordinate-based search, not to actually map an area. If the boundPoly element is used, then geoBndBox MUST be present, and all points enclosed by the boundPoly MUST be contained within the geoBndBox. Elements westBL, eastBL, southBL, and northBL of the geoBndBox should each be represented in at least one point of the boundPoly description. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_boundPoly(...)
```

## Arguments

... Child nodes or attributes.

## Details

*Parent nodes*

boundPoly is contained in sumDscr.

## Value

A ddi\_node object.

## Shared and complex child nodes

- [ddi\\_polygon\(\)](#)

## References

[boundPoly documentation](#)

## Examples

```
# ddi_boundPoly requires ddi_polygon(). ddi_polygon then requires ddi_point()
# which requires ddi_gringLat() and ddi_gringLon()
```

```
ddi_boundPoly(ddi_polygon(
  ddi_point(
    ddi_gringLat("42.002207"),
    ddi_gringLon("-120.005729004")
  )
)
```



---

ddi_catgry	<i>catgry, catgryGrp and their child nodes</i>
------------	--

---

### Description

catgry is a description of a particular categorical response. ddi\_catgryGrp() groups the responses together. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

ddi\_catgry(...)

ddi\_catgryGrp(...)

ddi\_catStat(...)

ddi\_catValu(...)

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

catgry and catgryGrp is contained in var.

#### *catgry and catgryGrp specific child nodes*

- ddi\_catStat() is a category level statistic. May include frequencies, percentages, or crosstabulation results. The attribute "type" indicates the type of statistics presented - frequency, percent, or crosstabulation. If a value of "other" is used for this attribute, the "otherType" attribute should take a value from a controlled vocabulary. This option should only be used when applying a controlled vocabulary to this attribute. Use the complex element controlledVocabUsed to identify the controlled vocabulary to which the selected term belongs.

#### *catgry specific child nodes*

- ddi\_catValu() is the category value.

### Value

A ddi\_node object.

### Shared and complex child nodes

- [ddi\\_labl\(\)](#)
- [ddi\\_mrow\(\)](#)
- [ddi\\_txt\(\)](#)

## References

[catgry documentation](#)  
[catgryGrp documentation](#)  
[catStat documentation](#)  
[catValu documentation](#)

## Examples

```
ddi_catgry(missing = "Y", missType = "inap")
ddi_catgryGrp(missing = "N")

# Functions that need to be wrapped in ddi_catgry() or ddi_catgryGrp()

ddi_catStat(type = "freq", "256")

# Functions that need to be wrapped in ddi_catgry()

ddi_catValu("9")
```

---

ddi\_citation

*citation, sourceCitation, fileCitation and their child nodes*

---

## Description

Citation entities for the study including general citations and source citations. Citation is a required element in the DDI-Codebook. `fileCitation` provides a full bibliographic citation option for each data file described in `fileDscr`. The minimum element set includes: `titl`, `IDNo`, `authEnty`, `producer`, and `prodDate`. If a DOI is available for the data enter this in the `IDNo`. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_citation(...)
ddi_sourceCitation(...)
ddi_fileCitation(...)
ddi_biblCit(...)
ddi_holdings(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

citation is contained in the following elements: docDscr; othRefs; otherMat; relMat; relPubl; relStdy; and stdyDscr. sourceCitation is contained in the sources element. fileCitation is included in the fileTxt element.

### *citation, sourceCitation, and fileCitation specific child nodes*

ddi\_biblCit() is the complete bibliographic reference containing all of the standard elements of a citation that can be used to cite the work. The "format" attribute is provided to enable specification of the particular citation style used, e.g., APA, MLA, Chicago, etc.

ddi\_holdings() is information concerning either the physical or electronic holdings of the cited work. Attributes include: location—The physical location where a copy is held; callno—The call number for a work at the location specified; and URI—A URN or URL for accessing the electronic copy of the cited work.

## Value

A ddi\_node object..

## Shared and complex child nodes

- [ddi\\_distStmt\(\)](#)
- [ddi\\_notes\(\)](#)
- [ddi\\_prodStmt\(\)](#)
- [ddi\\_rspStmt\(\)](#)
- [ddi\\_serStmt\(\)](#)
- [ddi\\_titlStmt\(\)](#)
- [ddi\\_verStmt\(\)](#)

## References

[citation documentation](#)

[fileCitation documentation](#)

[sourceCitation documentation](#)

[biblCit documentation](#)

[holdings documentation](#)

## Examples

```
ddi_citation()
```

```
ddi_sourceCitation()
```

```
ddi_fileCitation()
```

```
# An example using the ddi_biblCit() child:
```

```
ddi_citation(  
  ddi_biblCit(format = "APA", "Full citation text")  
)  
  
# An example using the ddi_holdings() child:  
  
ddi_citation(  
  ddi_holdings(location = "ICPSR DDI Repository",  
    callno = "inap.",  
    URI = "http://www.icpsr.umich.edu/DDIrepository/",  
    "Marked-up Codebook for Current Population Survey, 1999: Annual Demographic File")  
)
```

---

ddi\_codeBook

*Codebook*

---

## Description

The root node of a DDI 2.5 Codebook file. This file must contain `stdyDscr`. More information on this element, especially the allowed attributes, can be found in the references.

## Usage

```
ddi_codeBook(...)
```

## Arguments

... Child nodes or attributes.

## Value

A `ddi_node` object

## Shared and complex child nodes

- [ddi\\_dataDscr\(\)](#)
- [ddi\\_docDscr\(\)](#)
- [ddi\\_fileDscr\(\)](#)
- [ddi\\_otherMat\(\)](#)
- [ddi\\_stdYDscr\(\)](#)

## References

[codeBook documentation](#)

## Examples

```
# All ddi_codeBook() functions must contain ddi_stdyDscr(),  
# which also has ddi_citation() as a required child element.  
  
ddi_codeBook(ddi_stdyDscr(ddi_citation()))
```

---

ddi\_codingInstructions

*codingInstructions and its child nodes*

---

## Description

Describe specific coding instructions used in data processing, cleaning, assessment, or tabulation. Element `relatedProcesses` allows linking a coding instruction to one or more processes such as `dataProcessing`, `dataAppr`, `cleanOps`, etc. Use the `txt` element to describe instructions in a human readable form. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_codingInstructions(...)
```

```
ddi_command(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

`codingInstructions` is contained in method.

### *codingInstructions specific child nodes*

- `ddi_command()` provides command code for the coding instruction. The `formalLanguage` attribute identifies the language of the command code.

## Value

A `ddi_node` object.

## Shared and complex child nodes

- [ddi\\_txt\(\)](#)

**References**

[codingInstructions documentation](#)  
[command documentation](#)

**Examples**

```
ddi_codingInstructions()

# Functions that need to be wrapped in ddi_codingInstructions()

ddi_command(formalLanguage = "SPSS",
             "RECODE V1 TO V100 (10 THROUGH HIGH = 0)")
```

---

ddi\_cohort

*cohort and its child nodes*


---

**Description**

The element cohort is used when the nCube contains a limited number of categories from a particular variable, as opposed to the full range of categories. The attribute "catRef" is an IDREF to the actual category being used. The attribute "value" indicates the actual value attached to the category that is being used. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_cohort(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent nodes*

cohort is contained in dmns.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_range\(\)](#)

## References

[cohort documentation](#)

## Examples

```
ddi_cohort(catRef = "CV24_1", value = "1")
```

---

ddi_concept	<i>concept node</i>
-------------	---------------------

---

## Description

The general subject to which the parent element may be seen as pertaining. This element serves the same purpose as the keywords and topic classification elements, but at the data description level. The "vocab" attribute is provided to indicate the controlled vocabulary, if any, used in the element, e.g., LCSH (Library of Congress Subject Headings), MeSH (Medical Subject Headings), etc. The "vocabURI" attribute specifies the location for the full controlled vocabulary. More information on this element, especially its allowed attributes, can be found in the references.

## Usage

```
ddi_concept(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

concept is contained in the following elements: anlyUnit; anlysUnit; collMode; dataKind; geogCover; geogUnit; nCubeGrp; nation; resInstru; sampProc; srcOrig; timeMeth; universe; var; and varGrp.

## Value

A ddi\_node object.

## References

[concept documentation](#)

**Examples**

```
ddi_concept(vocab = "LCSH",
            vocabURI = "http://lcweb.loc.gov/catdir/cpsol/lcco/lcco.html",
            source = "archive",
            "more experience")
```

---

 ddi\_contact

*contact node*


---

**Description**

Names and addresses of individuals responsible for the work. Individuals listed as contact persons will be used as resource persons regarding problems or questions raised by the user community. The URI attribute should be used to indicate a URN or URL for the homepage of the contact individual. The email attribute is used to indicate an email address for the contact individual. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_contact(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent nodes*

contact is contained in the following elements: distStmt and useStmt.

**Value**

A ddi\_node object.

**References**

[contact documentation](#)

**Examples**

```
ddi_contact(affiliation = "University of Wisconsin",
            email = "jsmith@...",
            "Jane Smith")
```



---

`ddi_controlledVocabUsed`*controlledVocabUsed and its child nodes*

---

**Description**

Provides a code value, as well as a reference to the code list from which the value is taken. Note that the CodeValue can be restricted to reference an enumeration. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**`ddi_controlledVocabUsed(...)``ddi_codeListAgencyName(...)``ddi_codeListID(...)``ddi_codeListName(...)``ddi_codeListSchemeURN(...)``ddi_codeListURN(...)``ddi_codeListVersionID(...)`**Arguments**

... Child nodes or attributes.

**Details***Parent node*

`controlledVocabUsed` is contained in `docDscr`.

*controlledVocabUsed specific child nodes*

- `ddi_codeListAgencyName()` is the agency maintaining the code list.
- `ddi_codeListID()` identifies the code list that the value is taken from.
- `ddi_codeListName()` identifies the code list that the value is taken from with a human-readable name.
- `ddi_codeListSchemeURN()` identifies the code list scheme using a URN.
- `ddi_codeListURN()` identifies the code list that the value is taken from with a URN.
- `ddi_codeListVersionID()` is the version of the code list. (Default value is 1.0).

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_usage\(\)](#)

**References**

[controlledVocabUsed documentation](#)

[codeListAgencyName documentation](#)

[codeListID documentation](#)

[codeListName documentation](#)

[codeListSchemeURN documentation](#)

[codeListURN documentation](#)

[codeListVersionID documentation](#)

**Examples**

```
ddi_controlledVocabUsed(ddi_codeListID("TimeMethod"),
                        ddi_codeListName("Time Method"),
                        ddi_codeListAgencyName("DDI Alliance"),
                        ddi_codeListVersionID("1.2"),
                        ddi_codeListURN("urn:ddi-cv:TimeMethod:1.2"),
                        ddi_codeListSchemeURN("
                            http://www.ddialliance.org/Specification/
                            DDI-CV/TimeMethod_1.2_Genericcode1.0_DDI-CVProfile1.0.xml"),
                        ddi_usage())
```

---

ddi\_dataAccs

*dataAccs and its children*

---

**Description**

This section describes data access conditions and terms of use for the data collection. In cases where access conditions differ across individual files or variables, multiple access conditions can be specified. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

ddi\_dataAccs(...)

**Arguments**

... Child nodes or attributes.

**Details**

*Parent node*

dataAccs is contained in stdyDscr.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_notes\(\)](#)
- [ddi\\_setAvail\(\)](#)
- [ddi\\_useStmt\(\)](#)

**References**

[dataAccs documentation](#)

**Examples**

```
ddi_dataAccs()
```

---

ddi\_dataColl

*dataColl and its children*

---

**Description**

Information about the data collection methodology employed in the codebook. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_dataColl(...)
```

```
ddi_actMin(...)
```

```
ddi_cleanOps(...)
```

```
ddi_collectorTraining(...)
```

```
ddi_collMode(...)
```

```
ddi_collSitu(...)
```

```
ddi_ConOps(...)
```

```

ddi_dataCollector(...)
ddi_deviat(...)
ddi_frequenc(...)
ddi_instrumentDevelopment(...)
ddi_resInstru(...)
ddi_sampProc(...)
ddi_timeMeth(...)
ddi_weight(...)

```

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

dataColl is contained in method.

#### *dataColl specific child nodes*

- ddi\_actMin() is the summary of actions taken to minimize data loss. Includes information on actions such as follow-up visits, supervisory checks, historical matching, estimation, etc.
- ddi\_cleanOps() are the methods used to "clean" the data collection, e.g., consistency checking, wild code checking, etc. The "agency" attribute permits specification of the agency doing the data cleaning.
- ddi\_collectorTraining() describes the training provided to data collectors including interviewer training, process testing, compliance with standards etc. This is repeatable for language and to capture different aspects of the training process. The type attribute allows specification of the type of training being described.
- ddi\_collMode() is the method used to collect the data; instrumentation characteristics.
- ddi\_collSitu() is the description of noteworthy aspects of the data collection situation. Includes information on factors such as cooperativeness of respondents, duration of interviews, number of call-backs, etc.
- ddi\_ConOps() are control operations. These are methods to facilitate data control performed by the primary investigator or by the data archive. Specify any special programs used for such operations. The "agency" attribute maybe used to refer to the agency that performed the control operation.
- ddi\_dataCollector() is the entity (individual, agency, or institution) responsible for administering the questionnaire or interview or compiling the data. This refers to the entity collecting the data, not to the entity producing the documentation.

- `ddi_deviat()` are major deviations from the sample design. This is information indicating correspondence as well as discrepancies between the sampled units (obtained) and available statistics for the population (age, sex-ratio, marital status, etc.) as a whole.
- `ddi_frequenc()` is the frequency of data collection. It's for data collected at more than one point in time.
- `ddi_instrumentDevelopment()` describes any development work on the data collection instrument.
- `ddi_resInstru()` is the type of data collection instrument used.
- `ddi_sampProc()` is the type of sample and sample design used to select the survey respondents to represent the population. May include reference to the target sample size and the sampling fraction.
- `ddi_weight()` defines the weights used to produce accurate statistical results within the sampling procedures. Describe here the criteria for using weights in analysis of a collection. If a weighting formula or coefficient was developed, provide this formula, define its elements, and indicate how the formula is applied to data.

### Value

A `ddi_node` object.

### Shared and complex child nodes

- [ddi\\_sampleFrame\(\)](#)
- [ddi\\_sources\(\)](#)
- [ddi\\_targetSampleSize\(\)](#)

### References

[dataColl documentation](#)  
[actMin documentation](#)  
[cleanOps documentation](#)  
[collectorTraining documentation](#)  
[collMode documentation](#)  
[collSitu documentation](#)  
[ConOps documentation](#)  
[dataCollector documentation](#)  
[deviat documentation](#)  
[frequenc documentation](#)  
[instrumentDevelopment documentation](#)  
[resInstru documentation](#)  
[sampProc documentation](#)  
[timeMeth documentation](#)  
[weight documentation](#)

**Examples**

```
ddi_dataColl()

# Functions that need to be wrapped in ddi_dataColl()

ddi_actMin("To minimize the number of unresolved cases and reduce the
          potential nonresponse bias, four follow-up contacts were made with
          agencies that had not responded by various stages of the data
          collection process.")

ddi_cleanOps("Checks for undocumented codes were performed, and data were
            subsequently revised in consultation with the principal investigator.")

ddi_collectorTraining(type = "interviewer training",
                    "Describe research project, describe population and
                    sample, suggest methods and language for approaching
                    subjects, explain questions and key terms of survey instrument.")

ddi_collMode("telephone interviews")

ddi_collSitu("There were 1,194 respondents who answered questions in face-to-face
            interviews lasting approximately 75 minutes each.")

ddi_ConOps(agency = "ICPSR",
           "Ten percent of data entry forms were reentered to check for accuracy.")

ddi_dataCollector(abbr = "SRC",
                 affiliation = "University of Michigan",
                 role = "questionnaire administration",
                 "Survey Research Center")

ddi_deviat("The suitability of Ohio as a research site reflected its similarity
          to the United States as a whole. The evidence extended by Tuchfarber
          (1988) shows that Ohio is representative of the United States in
          several ways: percent urban and rural, percent of the population
          that is African American, median age, per capita income, percent
          living below the poverty level, and unemployment rate. Although
          results generated from an Ohio sample are not empirically
          generalizable to the United States, they may be suggestive of what
          might be expected nationally.")

ddi_frequenc("monthly")

ddi_instrumentDevelopment(type = "pretesting",
                        "The questionnaire was pre-tested with split-panel
                        tests, as well as an analysis of non-response rates
                        for individual items, and response distributions.")

ddi_resInstru("structured")

ddi_sampProc("National multistage area probability sample")
```

```
ddi_weight("The 1996 NES dataset includes two final person-level analysis weights
which incorporate sampling, nonresponse, and post-stratification
factors. One weight (variable #4) is for longitudinal micro-level
analysis using the 1996 NES Panel. The other weight (variable #3)
is for analysis of the 1996 NES combined sample (Panel component
cases plus Cross-section supplement cases). In addition, a Time
Series Weight (variable #5) which corrects for Panel attrition was
constructed. This weight should be used in analyses which compare
the 1996 NES to earlier unweighted National Election Study data
collections.")
```

---

ddi\_dataDscr

*dataDscr and its children*

---

## Description

Description of variables within the Codebook. More information on this element, especially the allowed attributes, can be found in the references.

## Usage

```
ddi_dataDscr(...)
```

## Arguments

... Child nodes or attributes.

## Details

*Parent node*

dataDscr is contained in codeBook.

## Value

A ddi\_node object

## Shared and complex child nodes

- [ddi\\_nCube\(\)](#)
- [ddi\\_nCubeGrp\(\)](#)
- [ddi\\_notes\(\)](#)
- [ddi\\_var\(\)](#)
- [ddi\\_varGrp\(\)](#)

## References

[dataDscr documentation](#)

**Examples**

```
ddi_dataDscr()
```

---

ddi\_dataFingerprint    *dataFingerprint and its child nodes*

---

**Description**

Allows for assigning a hash value (digital fingerprint) to the data or data file. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_dataFingerprint(...)  
ddi_algorithmSpecification(...)  
ddi_algorithmVersion(...)  
ddi_digitalFingerprintValue(...)
```

**Arguments**

...                    Child nodes or attributes.

**Details***Parent nodes*

dataFingerprint is contained in fileDscr.

*dataFingerprint specific child nodes*

- ddi\_algorithmSpecification()
- ddi\_algorithmVersion()
- ddi\_digitalFingerprintValue()

**Value**

A ddi\_node object.

**References**

[dataFingerprint documentation](#)  
[algorithmSpecification documentation](#)  
[algorithmVersion documentation](#)  
[digitalFingerprintValue documentation](#)



**Examples**

```

ddi_dataFingerprint()

# Functions that need to be wrapped in ddi_Fingerprint()

ddi_algorithmSpecification()

ddi_algorithmVersion()

ddi_digitalFingerprintValue()

```

---

ddi_dataItem	<i>dataItem and its child nodes</i>
--------------	-------------------------------------

---

**Description**

Identifies a physical storage location for an individual data entry, serving as a link between the physical location and the logical content description of each data item. . It is used to describe the physical location of aggregate/tabular data in cases where the nCube model is employed. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```

ddi_dataItem(...)

ddi_CubeCoord(...)

ddi_physLoc(...)

```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

dataItem is contained in locMap.

*dataItem specific child nodes*

- ddi\_CubeCoord() is an empty element containing only the attributes listed below. It is used to identify the coordinates of the data item within a logical nCube describing aggregate data. CubeCoord is repeated for each dimension of the nCube giving the coordinate number ("coordNo") and coordinate value ("coordVal"). Coordinate value reference ("cordValRef") is an ID reference to the variable that carries the coordinate value. The attributes provide a complete coordinate location of a cell within the nCube.

- `ddi_physLoc()` is an empty element containing only the attributes listed below. Attributes include "type" (type of file structure: rectangular, hierarchical, two-dimensional, relational), "recRef" (IDREF link to the appropriate file or `recGrp` element within a file), "startPos" (starting position of variable or data item), "endPos" (ending position of variable or data item), "width" (number of columns the variable/data item occupies), "RecSegNo" (the record segment number, deck or card number the variable or data item is located on), and "fileid" (an IDREF link to the `fileDscr` element for the file that includes this physical location).

### Value

A `ddi_node` object.

### References

[dataItem documentation](#)

[CubeCoord documentation](#)

[physLoc documentation](#)

### Examples

```
ddi_dataItem()

# Functions that need to be wrapped in ddi_dataItem()

ddi_CubeCoord(coordNo = "1", coordVal = "3")

ddi_physLoc(type = "rectangular",
            recRef = "R1",
            startPos = "55",
            endPos = "57",
            width = "3")
```

---

`ddi_dataSrc`

*dataSrc node*

---

### Description

Used to list the book(s), article(s), serial(s), and/or machine-readable data file(s)—if any—that served as the source(s) of the data collection. More information on this element, especially its allowed attributes, can be found in the references.

### Usage

```
ddi_dataSrc(...)
```

### Arguments

... Child nodes or attributes.

**Details***Parent nodes*

dataSrc is contained in the following elements: sources and resource.

**Value**

A ddi\_node object.

**References**

[dataSrc documentation](#)

**Examples**

```
ddi_dataSrc('"Voting Scores." CONGRESSIONAL QUARTERLY ALMANAC 33 (1977), 487-498.')
```

---

ddi_derivation	<i>derivation and its child nodes</i>
----------------	---------------------------------------

---

**Description**

Used only in the case of a derived variable, this element provides both a description of how the derivation was performed and the command used to generate the derived variable, as well as a specification of the other variables in the study used to generate the derivation. The "var" attribute provides the ID values of the other variables in the study used to generate this derived variable. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_derivation(...)
```

```
ddi_drvdesc(...)
```

```
ddi_drvcmd(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

derivation is included in var.

*derivation specific child nodes*

- `ddi_drvcmd()` is the actual command used to generate the derived variable. The "syntax" attribute is used to indicate the command language employed (e.g., SPSS, SAS, Fortran, etc.). The element may be repeated to support multiple language expressions of the content.
- `ddi_drvdesc()` is a textual description of the way in which this variable was derived. The element may be repeated to support multiple language expressions of the content.

### Value

A `ddi_node` object.

### References

[derivation documentation](#)

[drvcmd documentation](#)

[drvdesc documentation](#)

### Examples

```
ddi_derivation()

# Functions that need to be wrapped in ddi_derivation()

ddi_drvcmd(syntax = "SPSS",
           "RECODE V1 TO V3 (0=1) (1=0) (2=-1) INTO DEFENSE WELFARE HEALTH.")

ddi_drvdesc("VAR215.01 'Outcome of first pregnancy' (1988 NSFG=VAR611 PREGOUT1)
           If R has never been pregnant (VAR203 PREGNUM EQ 0) then OUTCOM01 is
           blank/inapplicable. Else, OUTCOM01 is transferred from VAR225
           OUTCOME for R's 1st pregnancy.")
```

---

`ddi_developmentActivity`

*developmentActivity and its child nodes*

---

### Description

Describe the activity, listing participants with their role and affiliation, resources used (sources of information), and the outcome of the development activity.

### Usage

`ddi_developmentActivity(...)`

`ddi_description(...)`

`ddi_outcome(...)`

`ddi_participant(...)`

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

developmentActivity is contained in studyDevelopment.

*developmentActivity specific child nodes*

- ddi\_description() describes the development activity.
- ddi\_outcome() describes the outcome of the development activity.
- ddi\_participant() lists the participants conducting or designing the development activity.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_resource\(\)](#)

**References**

[developmentActivity documentation](#)

[description documentation](#)

[outcome documentation](#)

[participant documentation](#)

**Examples**

```
ddi_developmentActivity(type = "checkDataAvailability")

# Functions that need to be wrapped in ddi_developmentActivity()

ddi_description("A number of potential sources were evaluated for content,
                consistency and quality")

ddi_outcome("Due to quality issues this was determined not to be a viable
            source of data for the study")

ddi_participant(affiliation = "NSO",
                role = "statistician",
                "John Doe")
```

---

ddi_dimensns	<i>dimensns, recDimnsn and their child nodes</i>
--------------	--

---

### Description

Dimensions of the overall digital or physical file. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

ddi\_dimensns(...)

ddi\_caseQty(...)

ddi\_logRecl(...)

ddi\_recDimnsn(...)

ddi\_recNumTot(...)

ddi\_recPrCas(...)

ddi\_varQty(...)

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

dimensns is contained in fileTxt. recDimensn is contained in recGrp.

#### *dimensns and recDimnsn shared nodes*

- ddi\_caseQty() is the number of cases, observations, or records.
- ddi\_logRecl() is the logical record length, i.e., number of characters of data in the record.
- ddi\_varQty() is the overall variable count.

#### *dimensns specific nodes*

- ddi\_recNumTot() is the overall record count in file. Particularly helpful in instances such as files with multiple cards/decks or records per case.
- ddi\_recPrCas() is the number of records per case in the file. This element should be used for card-image data or other files in which there are multiple records per case.

### Value

A ddi\_node object.

**References**

[dimensns documentation](#)  
[caseQty documentation](#)  
[logRecL documentation](#)  
[recDimnsn documentation](#)  
[recNumTot documentation](#)  
[recPrCas documentation](#)  
[varQty documentation](#)

**Examples**

```
ddi_dimensns()  
  
ddi_recDimnsn()  
  
# Functions that need to be wrapped in ddi_dimensns() or ddi_recDimnsn()  
ddi_caseQty("1011")  
  
ddi_logRecL("27")  
  
ddi_varQty("27")  
  
# Functions that need to be wrapped in ddi_dimensns  
ddi_recNumTot("2400")  
  
ddi_recPrCas("5")
```

---

ddi\_distStmt

*ddi\_distStmt and its children*

---

**Description**

Distribution statement for the work at the appropriate level: marked-up document; marked-up document source; study; study description, other material; other material for study. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_distStmt(...)  
  
ddi_depDate(...)  
  
ddi_depositr(...)
```

ddi\_distDate(...)

ddi\_distribtr(...)

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

distStmt is contained in the following elements: citation; docSrc; fileCitation; and sourceCitation.

#### *distStmt specific child nodes*

ddi\_depDate() is the date that the work was deposited with the archive that originally received it. The ISO standard for dates (YYYY-MM-DD) is recommended for use with the "date" attribute.

ddi\_depositr() is the name of the person (or institution) who provided this work to the archive storing it.

ddi\_distDate() is the date that the work was made available for distribution/presentation. The ISO standard for dates (YYYY-MM-DD) is recommended for use with the "date" attribute. If using a text entry in the element content, the element may be repeated to support multiple language expressions.

ddi\_distribtr() is the organization designated by the author or producer to generate copies of the particular work including any necessary editions or revisions. Names and addresses may be specified and other archives may be co-distributors. A URI attribute is included to provide an URN or URL to the ordering service or download facility on a Web site.

### Value

A ddi\_node object.

### Shared and complex child nodes

- [ddi\\_contact\(\)](#)

### References

[distStmt documentation](#)

[depDate documentation](#)

[depositr documentation](#)

[distDate documentation](#)

[distribtr documentation](#)



**Examples**

```

ddi_distStmt()

# Functions that need to be wrapped in ddi_distStmt()

ddi_depDate(date = "2022-01-01", "January 1, 2022")

ddi_depositr(abbr = "BJS",
             affiliation = "U.S. Department of Justice",
             "Bureau of Justice Statistics")

ddi_distDate(date = "2022-01-01", "January 1, 2022")

ddi_distrbtr(abbr = "ICPSR",
             affiliation = "Institute for Social Research",
             URI = "http://www.icpsr.umich.edu",
             "Ann Arbor, MI: Inter-university Consortium for Political and Social Research")

```

---

ddi\_dmns

*dmns and its child nodes*


---

**Description**

This element defines a variable as a dimension of the nCube, and should be repeated to describe each of the cube's dimensions. The attribute "rank" is used to define the coordinate order (rank="1", rank="2", etc.). The attribute "varRef" is an IDREF that points to the variable that makes up this dimension of the nCube. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_dmns(...)
```

**Arguments**

...                   Child nodes or attributes.

**Details***Parent nodes*

dmns is contained in nCube.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_cohort\(\)](#)

**References**

[dmns documentation](#)

**Examples**

```
ddi_dmns(rank = "1", varRef = "var01")
```

---

ddi\_docDscr

*docDscr and its children*

---

**Description**

The Document Description consists of bibliographic information describing the DDI-compliant document itself as a whole. This Document Description can be considered the wrapper or header whose elements uniquely describe the full contents of the compliant DDI file. Since the Document Description section is used to identify the DDI-compliant file within an electronic resource discovery environment, this section should be as complete as possible. The author in the Document Description should be the individual(s) or organization(s) directly responsible for the intellectual content of the DDI version, as distinct from the person(s) or organization(s) responsible for the intellectual content of the earlier paper or electronic edition from which the DDI edition may have been derived. The producer in the Document Description should be the agency or person that prepared the marked-up document. Note that the Document Description section contains a Documentation Source subsection consisting of information about the source of the DDI-compliant file—that is, the hardcopy or electronic codebook that served as the source for the marked-up codebook. These sections allow the creator of the DDI file to produce version, responsibility, and other descriptions relating to both the creation of that DDI file as a separate and reformatted version of source materials (either print or electronic) and the original source materials themselves. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_docDscr(...)
```

```
ddi_docStatus(...)
```

```
ddi_guide(...)
```

**Arguments**

... Child nodes or attributes.

## Details

### *Parent node*

docDscr is contained in codeBook.

### *docDscr specific child nodes*

- `ddi_docStatus()` indicates if the documentation is being presented/distributed before it has been finalized. Some data producers and social science data archives employ data processing strategies that provide for release of data and documentation at various stages of processing. The element may be repeated to support multiple language expressions of the content.
- `ddi_guide()` is the list of terms and definitions used in the documentation. Provided to assist users in using the document correctly.

## Value

A `ddi_node` object.

## Shared and complex child nodes

- [ddi\\_citation\(\)](#)
- [ddi\\_controlledVocabUsed\(\)](#)
- [ddi\\_docSrc\(\)](#)
- [ddi\\_notes\(\)](#)

## References

[docDscr documentation](#)

[docStatus documentation](#)

[guide documentation](#)

## Examples

```
ddi_docDscr()  
  
# Functions that need to be wrapped in ddi_docDscr()  
  
ddi_docStatus("This marked-up document includes a provisional data dictionary...")  
  
ddi_guide("List of terms and definitions")
```

---

`ddi_docSrc`*docSrc and its child nodes*

---

**Description**

Citation for the source document. This element encodes the bibliographic information describing the source codebook, including title information, statement of responsibility, production and distribution information, series and version information, text of a preferred bibliographic citation, and notes (if any). Information for this section should be taken directly from the source document whenever possible. If additional information is obtained and entered in the elements within this section, the source of this information should be noted in the source attribute of the particular element tag. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**`ddi_docSrc(...)`**Arguments**

... Child nodes or attributes.

**Details**

*Parent node*

`docSrc` is contained in `docDscr`.

**Value**

A `ddi_node` object.

**Shared and complex child nodes**

- `ddi_biblCit()`
- `ddi_distStmt()`
- `ddi_holdings()`
- `ddi_notes()`
- `ddi_prodStmt()`
- `ddi_rspStmt()`
- `ddi_serStmt()`
- `ddi_titlStmt()`
- `ddi_verStmt()`

**References**

[docSrc documentation](#)

**Examples**

```
ddi_docSrc()
```

---

ddi\_embargo

*embargo node*

---

**Description**

Provides information on variables/nCubes which are not currently available because of policies established by the principal investigators and/or data producers. This element may be repeated to support multiple language expressions of the content.

**Usage**

```
ddi_embargo(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent nodes*

embargo is contained in nCube and var.

**Value**

A ddi\_node object.

**References**

[embargo documentation](#)

**Examples**

```
ddi_embargo(event = "notBefore",
             date = "2001-09-30",
             "The data associated with this variable/nCube will not become
             available until September 30, 2001, because of embargo provisions
             established by the data producers.")
```

---

ddi\_exPostEvaluation *exPostEvaluation and its child nodes*

---

## Description

Post Evaluation Procedures describes evaluation procedures not addressed in data evaluation processes. These may include issues such as timing of the study, sequencing issues, cost/budget issues, relevance, institutional or legal arrangements etc. of the study. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_exPostEvaluation(...)
```

```
ddi_evaluationProcess(...)
```

```
ddi_evaluator(...)
```

```
ddi_outcomes(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

exPostEvaluation is contained in stdyInfo.

### *exPostEvaluation specific child nodes*

- ddi\_evaluationProcess() describes the evaluation process followed.
- ddi\_evaluator() identifies persons or organizations involved in the evaluation.
- ddi\_outcomes() describes the outcomes of the evaluation.

## Value

A ddi\_node object.

## References

[exPostEvaluation documentation](#)

[evaluationProcess documentation](#)

[evaluator documentation](#)

[outcomes documentation](#)

**Examples**

```
ddi_exPostEvaluation()

# Functions that need to be wrapped in ddi_exPostEvaluation()

ddi_evaluationProcess("This dataset was evaluated using the following methods...")

ddi_evaluator(affiliation = "United Nations",
              abbr = "UNSD",
              role = "consultant",
              "United Nations Statistical Division")

ddi_outcomes("The following steps were highly effective in increasing response
             rates, and should be repeated in the next collection cycle...")
```

---

ddi\_fileDscr                      *fileDscr and its children*

---

**Description**

Information about the data file(s) that comprises a collection. This section can be repeated for collections with multiple files. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_fileDscr(...)
```

**Arguments**

...                      Child nodes or attributes.

**Details**

*Parent node*

fileDscr is contained in codeBook.

**Value**

A ddi\_node object

**Shared and complex child nodes**

- [ddi\\_fileTxt\(\)](#)
- [ddi\\_locMap\(\)](#)
- [ddi\\_notes\(\)](#)

**References**

[fileDscr documentation](#)

**Examples**

```
ddi_fileDscr()
```

---

ddi_fileStrc	<i>fileStrc and its child nodes</i>
--------------	-------------------------------------

---

**Description**

Type of file structure. The file structure is fully described in the first fileTxt within the fileDscr and then the fileStrc in subsequent fileTxt descriptions would reference the first fileStrcRef attribute rather than repeat the details. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_fileStrc(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent node*

fileStrc is contained in fileTxt.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_notes\(\)](#)
- [ddi\\_recGrp\(\)](#)

**References**

[fileStrc documentation](#)

**Examples**

```
ddi_fileStrc()
```



---

ddi_fileTxt	<i>fileTxt and its children</i>
-------------	---------------------------------

---

### Description

Provides descriptive information about the data file. More information on these elements, especially their allowed attributes, can be found in the references.

*Parent nodes*

### Usage

ddi\_fileTxt(...)

ddi\_dataChck(...)

ddi\_dataMsng(...)

ddi\_fileCont(...)

ddi\_fileName(...)

ddi\_filePlac(...)

ddi\_fileType(...)

ddi\_format(...)

ddi\_ProcStat(...)

### Arguments

... Child nodes or attributes.

### Details

fileTxt is contained in fileDscr.

*fileTxt specific child nodes*

- [ddi\\_dataChck\(\)](#) are the types of checks and operations performed on the data file at the file level.
- [ddi\\_dataMsng\(\)](#) can be used to give general information about missing data, e.g., that missing data have been standardized across the collection, missing data are present because of merging, etc.
- [ddi\\_fileCont\(\)](#) are the file contents. It is the abstract or description of the file. A summary describing the purpose, nature, and scope of the data file, special characteristics of its contents, major subject areas covered, and what questions the PIs attempted to answer when they created

the file. A listing of major variables in the file is important here. In the case of multi-file collections, this uniquely describes the contents of each file.

- `ddi_fileName()` contains a short title that will be used to distinguish a particular file/part from other files/parts in the data collection. The element may be repeated to support multiple language expressions of the content.
- `ddi_filePlac()` indicates where the file was produced, whether at an archive or elsewhere.
- `ddi_fileType()` are the types of data files. These include raw data (ASCII, EBCDIC, etc.) and software-dependent files such as SAS datasets, SPSS export files, etc. If the data are of mixed types (e.g., ASCII and packed decimal), state that here.
- `ddi_format()` is the physical format of the data file: Logical record length format, card-image format (i.e., data with multiple records per case), delimited format, free format, etc. The element may be repeated to support multiple language expressions of the content.
- `ddi_ProcStat()` is the processing status of the file. Some data producers and social science data archives employ data processing strategies that provide for release of data and documentation at various stages of processing.

### Value

A `ddi_node` object.

### Shared and complex child nodes

- `ddi_dataFingerprint()`
- `ddi_dimensns()`
- `ddi_fileCitation()`
- `ddi_fileStrc()`
- `ddi_software()`
- `ddi_verStmt()`

### References

[fileTxt documentation](#)  
[dataChck documentation](#)  
[dataMsng documentation](#)  
[fileCont documentation](#)  
[fileName documentation](#)  
[filePlac documentation](#)  
[fileType documentation](#)  
[format documentation](#)  
[ProcStat documentation](#)

**Examples**

```

ddi_fileTxt()

# Functions that need to be wrapped in ddi_fileTxt()

ddi_dataChck("Consistency checks were performed by Data Producer/ Principal Investigator.")

ddi_dataMsg('The codes "-1" and "-2" are used to represent missing data.')
```

```

ddi_fileCont("Part 1 contains both edited and constructed variables describing demographic...")

ddi_fileName(ID = "File1", "Second-Generation Children Data")

ddi_filePlac("Washington, DC: United States Department of Commerce, Bureau of the Census")

ddi_fileType(charset = "US-ASCII", "ASCII data file")

ddi_format("comma-delimited")

ddi_ProcStat("Available from the DDA. Being processed.")
```

---

ddi_frameUnit	<i>frameUnit and its children</i>
---------------	-----------------------------------

---

**Description**

Provides information about the sampling frame unit. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```

ddi_frameUnit(...)

ddi_unitType(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

frameUnit is contained in sampleFrame.

*frameUnit specific child nodes*

- ddi\_unitType() describes the type of sampling frame unit. The attribute "numberOfUnits" provides the number of units in the sampling frame.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_txt\(\)](#)

**References**

[frameUnit documentation](#)

[unitType documentation](#)

**Examples**

```
ddi_frameUnit()

# Functions that need to be wrapped in ddi_frameUnit()

ddi_unitType(numberOfUnits = 150000,
             "Primary listed owners of published phone numbers in the City of St. Paul")
```

---

ddi\_geoBndBox

*geoBndBox and its child nodes*

---

**Description**

The fundamental geometric description for any dataset that models geography. `geoBndBox` is the minimum box, defined by west and east longitudes and north and south latitudes, that includes the largest geographic extent of the dataset's geographic coverage. This element is used in the first pass of a coordinate-based search. If the `boundPoly` element is included, then the `geoBndBox` element **MUST** be included. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_geoBndBox(...)

ddi_eastBL(...)

ddi_northBL(...)

ddi_southBL(...)

ddi_westBL(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

geoBndBox is contained in sumDscr.

### *geoBndBox specific child nodes*

- `ddi_eastBL()` is the easternmost coordinate delimiting the geographic extent of the dataset. A valid range of values, expressed in decimal degrees (positive east and positive north), is:  $-180,0 \leq \text{East Bounding Longitude Value} \leq 180,0$ .
- `ddi_northBL()` is the northernmost coordinate delimiting the geographic extent of the dataset. A valid range of values, expressed in decimal degrees (positive east and positive north), is:  $-90,0 \leq \text{North Bounding Latitude Value} \leq 90,0$  ;  $\text{North Bounding Latitude Value} \geq \text{South Bounding Latitude Value}$ .
- `ddi_southBL()` is the southernmost coordinate delimiting the geographic extent of the dataset. A valid range of values, expressed in decimal degrees (positive east and positive north), is:  $-90,0 \leq \text{South Bounding Latitude Value} \leq 90,0$  ;  $\text{South Bounding Latitude Value} \leq \text{North Bounding Latitude Value}$ .
- `ddi_westBL()` is the westernmost coordinate delimiting the geographic extent of the dataset. A valid range of values, expressed in decimal degrees (positive east and positive north), is:  $-180,0 \leq \text{West Bounding Longitude Value} \leq 180,0$ .

## Value

A `ddi_node` object.

## References

[geoBndBox documentation](#)

[eastBL documentation](#)

[northBL documentation](#)

[southBL documentation](#)

[westBL documentation](#)

## Examples

```
ddi_geoBndBox()  
  
# Functions that need to be wrapped in ddi_geoBndBox()  
  
ddi_eastBL("90")  
  
ddi_northBL("17")  
  
ddi_southBL("45")
```

```
ddi_westBL("-10")
```

---

ddi_imputation	<i>imputation node</i>
----------------	------------------------

---

### Description

According to the Statistical Terminology glossary maintained by the National Science Foundation, this is "the process by which one estimates missing values for items that a survey respondent failed to provide," and if applicable in this context, it refers to the type of procedure used. When applied to an nCube, imputation takes into consideration all of the dimensions that are part of that nCube. This element may be repeated to support multiple language expressions of the content. More information on this element, especially its allowed attributes, can be found in the references.

### Usage

```
ddi_imputation(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

imputation is contained in nCube and var.

### Value

A ddi\_node object.

### References

[imputation documentation](#)

### Examples

```
ddi_imputation("This variable contains values that were derived by substitution.")
```

---

ddi_labl	<i>labl node</i>
----------	------------------

---

### Description

A short description of the parent element. In the variable label, the length of this phrase may depend on the statistical analysis system used (e.g., some versions of SAS permit 40-character labels, while some versions of SPSS permit 120 characters), although the DDI itself imposes no restrictions on the number of characters allowed. More information on this element, especially its allowed attributes, can be found in the references.

### Usage

```
ddi_labl(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

labl is contained in the following elements: catgry; catgryGrp; nCube; nCubeGrp; otherMat; recGrp; sampleFrame; var; and varGrp.

### Value

A ddi\_node object.

### References

[labl documentation](#)

### Examples

```
ddi_labl(level = "variable", lang = "en", "short variable description")
```

---

ddi_location	<i>location node</i>
--------------	----------------------

---

### Description

The physical or digital location of the variable. It is an empty element. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

```
ddi_location(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

location is contained in nCube and var.

### Value

A ddi\_node object.

### References

[location documentation](#)

### Examples

```
ddi_location(StartPos = "55",  
             EndPos = "57",  
             RecSegNo = "2",  
             fileid = "CARD-IMAGE")
```



---

`ddi_locMap`*locMap and its child nodes*

---

## Description

This element maps individual data entries to one or more physical storage locations. It is used to describe the physical location of aggregate/tabular data in cases where the nCube model is employed. More information on these elements, especially their allowed attributes, can be found in the references.

*Parent nodes*

## Usage

```
ddi_locMap(...)
```

## Arguments

... Child nodes or attributes.

## Details

locMap is contained in fileDscr.

## Value

A ddi\_node object.

## Shared and complex child nodes

- [ddi\\_dataItem\(\)](#)

## References

[locMap documentation](#)

## Examples

```
ddi_locMap()
```

---

ddi_method	<i>method and its child nodes</i>
------------	-----------------------------------

---

### Description

This section describes the methodology and processing involved in a data collection. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

```
ddi_method(...)
```

```
ddi_dataProcessing(...)
```

```
ddi_stdyclas(...)
```

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

method is contained in stdyDscr.

#### *method specific child nodes*

- `ddi_dataProcessing()` describes various data processing procedures not captured elsewhere in the documentation, such as topcoding, recoding, suppression, tabulation, etc. The "type" attribute supports better classification of this activity, including the optional use of a controlled vocabulary.
- `ddi_stdyclas()` is generally used to give the data archive's class or study status number, which indicates the processing status of the study. May also be used as a text field to describe processing status. This element may be repeated to support multiple language expressions of the content.

### Value

A `ddi_node` object.

### Shared and complex child nodes

- [ddi\\_anlyInfo\(\)](#)
- [ddi\\_codingInstructions\(\)](#)
- [ddi\\_dataColl\(\)](#)
- [ddi\\_notes\(\)](#)

**References**

[method documentation](#)  
[dataProcessing documentation](#)  
[stdyClas documentation](#)

**Examples**

```
ddi_method()

# Functions that need to be wrapped in ddi_method()

ddi_dataProcessing(type = "topcoding",
                  "The income variables in this study (RESP_INC, HHD_INC, and
                  SS_INC) were topcoded to protect confidentiality.")

ddi_stdyclas("ICPSR Class II")
```

---

ddi_mrow	<i>mrow and its child nodes</i>
----------	---------------------------------

---

**Description**

mrow or Mathematical Row is a wrapper containing the presentation expression *mi*. It creates a single string without spaces consisting of the individual elements described within it. It can be used to create a single variable by concatenating other variables into a single string. It is used to create linking variables composed of multiple non-contiguous parts, or to define unique strings for various category values of a single variable. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_mrow(...)

ddi_mi(...)
```

**Arguments**

...                    Child nodes or attributes.

**Details***Parent nodes*

mrow is contained in *catgry*.

*mrow specific child nodes*

- `ddi_mi()` is the mathematical identifier. This is the token element containing the smallest unit in the mrow that carries meaning.

**Value**

A ddi\_node object.

**References**

[mrow documentation](#)

[mi documentation](#)

**Examples**

```
ddi_mrow()

# Functions that need to be wrapped in ddi_mrow()

ddi_mi("1")
```

---

ddi\_nCube

*nCube and its child nodes*

---

**Description**

Describes the logical structure of an n-dimensional array, in which each coordinate intersects with every other dimension at a single point. The nCube has been designed for use in the markup of aggregate data. Repetition of the following elements is provided to support multi-language content: anlysUnit, embargo, imputation, purpose, respUnit, and security. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_nCube(...)

ddi_measure(...)

ddi_purpose(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent nodes*

nCube is contained in dataDscr.

*nCube specific child nodes*

- `ddi_measure()` indicates the measurement features of the cell content: type of aggregation used, measurement unit, and measurement scale. An origin point is recorded for anchored scales, to be used in determining relative movement along the scale. Additivity indicates whether an aggregate is a stock (like the population at a given point in time) or a flow (like the number of births or deaths over a certain period of time). The non-additive flag is to be used for measures that for logical reasons cannot be aggregated to a higher level - for instance, data that only make sense at a certain level of aggregation, like a classification. Two nCubes may be identical except for their measure - for example, a count of persons by age and percent of persons by age. Measure is an empty element.
- `ddi_purpose()` explains the purpose for which a particular nCube was created.

### Value

A `ddi_node` object.

### Shared and complex child nodes

- `ddi_anlysUnit()`
- `ddi_embargo()`
- `ddi_imputation()`
- `ddi_labl()`
- `ddi_location()`
- `ddi_notes()`
- `ddi_respUnit()`
- `ddi_security()`
- `ddi_txt()`
- `ddi_universe()`
- `ddi_verStmt()`

### References

[nCube documentation](#)  
[measure documentation](#)  
[purpose documentation](#)

### Examples

```
ddi_nCube()  
  
# Functions that need to be wrapped in ddi_nCube()  
  
ddi_measure(aggrMeth = "sum", additivity = "stock")  
  
ddi_purpose("Meets reporting requirements for the Federal Reserve Board")
```

---

ddi_nCubeGrp	<i>nCubeGrp and its child nodes</i>
--------------	-------------------------------------

---

### Description

A group of nCubes that may share a common subject, arise from the interpretation of a single question, or are linked by some other factor. This element makes it possible to identify all nCubes derived from a simple presentation table, and to provide the original table title and universe, as well as reference the source. Specific nesting patterns can be described using the attribute nCubeGrp. nCube groups are also created this way in order to permit nCubes to belong to multiple groups, including multiple subject groups, without causing overlapping groups. nCubes that are linked by the same use of the same variable need not be identified by an nCubeGrp element because they are already linked by a common variable element. Note that as a result of the strict sequencing required by XML, all nCube Groups must be marked up before the Variable element is opened. That is, the mark-up author cannot mark up a nCube Group, then mark up its constituent nCubes, then mark up another nCube Group. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

```
ddi_nCubeGrp(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

nCubeGrp is contained in dataDscr.

### Value

A ddi\_node object.

### Shared and complex child nodes

- [ddi\\_concept\(\)](#)
- [ddi\\_defntn\(\)](#)
- [ddi\\_labl\(\)](#)
- [ddi\\_notes\(\)](#)
- [ddi\\_txt\(\)](#)
- [ddi\\_universe\(\)](#)

### References

[nCubeGrp documentation](#)

**Examples**

```
ddi_nCubeGrp(name = "Group 1")
```

---

ddi\_notes

*notes node*

---

**Description**

For clarifying information/annotation regarding the parent element. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_notes(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

notes is contained in the following elements: citation; dataAccs; dataDscr; docDscr; docSrc; fileCitation; fileDscr; fileStrc; invalrng; method; nCube; nCubeGrp; otherMat; setAvail; sourceCitation; stdyDscr; stdyInfo; valrng; var; varGrp; and verStmt.

**Value**

A ddi\_node object.

**References**

[notes documentation](#)

**Examples**

```
ddi_notes(resp = "Jane Smith", "The source codebook was produced from original  
hardcopy materials using Optical Character Recognition (OCR).")
```

---

`ddi_otherMat`*otherMat and its children*

---

### Description

This section allows for the inclusion of other materials that are related to the study as identified and labeled by the DTD/Schema users (encoders). The materials may be entered as PCDATA (ASCII text) directly into the document (through use of the "txt" element). This section may also serve as a "container" for other electronic materials such as setup files by providing a brief description of the study-related materials accompanied by the attributes "type" and "level" defining the material further. Other Study-Related Materials may include: questionnaires, coding notes, SPSS/SAS/Stata setup files (and others), user manuals, continuity guides, sample computer software programs, glossaries of terms, interviewer/project instructions, maps, database schema, data dictionaries, show cards, coding information, interview schedules, missing values information, frequency files, variable maps, etc. More information on this element, especially the allowed attributes, can be found in the references.

### Usage

```
ddi_otherMat(...)
```

### Arguments

```
...          Child nodes or attributes.
```

### Details

*Parent nodes*

otherMat is contained in the following elements: codeBook and otherMat.

### Value

A ddi\_node object

### Shared and complex child nodes

- `ddi_citation()`
- `ddi_labl()`
- `ddi_notes()`
- `ddi_otherMat()`
- `ddi_table()`
- `ddi_txt()`

### References

[otherMat documentation](#)



**Examples**

```
ddi_otherMat()
```

---

ddi_othrStdyMat	<i>othrStdyMat and its child nodes</i>
-----------------	--

---

**Description**

Other study description materials relating to the study description. This section describes other materials that are related to the study description that are primarily descriptions of the content and use of the study, such as appendices, sampling information, weighting details, methodological and technical details, publications based upon the study content, related studies or collections of studies, etc. This section may point to other materials related to the description of the study through use of the generic citation element, which is available for each element in this section. This maps to Dublin Core Relation element. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_othrStdyMat(...)
```

```
ddi_othRefs(...)
```

```
ddi_relMat(...)
```

```
ddi_relPubl(...)
```

```
ddi_relStdy(...)
```

**Arguments**

```
... Child nodes or attributes.
```

**Details***Parent nodes*

othrStdyMat is contained in stdyDscr.

*othrStdyMat specific child nodes*

- ddi\_othRefs() indicates other pertinent references. can take the form of natural language text and/or bibliographic citations using ddi\_citation().
- ddi\_relMat() describes materials related to the study description, such as appendices, additional information on sampling found in other documents, etc. Can take the form of natural language text and/or bibliographic citations using ddi\_citation(). This element can contain either PCDATA or a citation or both, and there can be multiple occurrences of both the citation

and PCDATA within a single element. May consist of a single URI or a series of URIs comprising a series of citations/references to external materials which can be objects as a whole (journal articles) or parts of objects (chapters or appendices in articles or documents).

- `ddi_relPubl()` are bibliographic and access information about articles and reports based on the data in this collection. Can take the form of natural language text and/or bibliographic citations using `ddi_citation()`.
- `ddi_relStdy()` is information on the relationship of the current data collection to others (e.g., predecessors, successors, other waves or rounds) or to other editions of the same file. This would include the names of additional data collections generated from the same data collection vehicle plus other collections directed at the same general topic. Can take the form of natural language text and/or bibliographic citations using `ddi_citation()`.

### Value

A `ddi_node` object.

### References

[othrStdyMat documentation](#)

[othRefs documentation](#)

[relMat documentation](#)

[relPubl documentation](#)

[relStdy documentation](#)

### Examples

```
ddi_othrStdyMat()

# Functions that need to be wrapped in ddi_othrStdyMat()

ddi_othRefs("Part II of the documentation, the Field Representative's Manual,
            is provided in hardcopy form only.")

ddi_relMat("Full details on the research design and procedures, sampling
            methodology, content areas, and questionnaire design, as well as
            percentage distributions by respondent's sex, race, region, college
            plans, and drug use, appear in the annual ISR volumes MONITORING
            THE FUTURE: QUESTIONNAIRE RESPONSES FROM THE NATION'S HIGH SCHOOL
            SENIORS.")

ddi_relPubl("Economic Behavior Program Staff. SURVEYS OF CONSUMER FINANCES.
            Annual volumes 1960 through 1970. Ann Arbor, MI: Institute for
            Social Research.")

ddi_relStdy("ICPSR distributes a companion study to this collection titled
            FEMALE LABOR FORCE PARTICIPATION AND MARITAL INSTABILITY, 1980:
            [UNITED STATES] (ICPSR 9199).")
```

---

ddi_point	<i>point and its child nodes</i>
-----------	----------------------------------

---

### Description

0-dimensional geometric primitive, representing a position, but not having extent. In this declaration, point is limited to a longitude/latitude coordinate system.

### Usage

```
ddi_point(...)
```

```
ddi_gringLat(...)
```

```
ddi_gringLon(...)
```

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

point is contained in polygon.

#### *point specific child nodes*

- `ddi_gringLat()` is the latitude (y coordinate) of a point. Valid range expressed in decimal degrees is as follows: -90,0 to 90,0 degrees (latitude).
- `ddi_gringLon()` is the longitude (x coordinate) of a point. Valid range expressed in decimal degrees is as follows: -180,0 to 180,0 degrees (longitude).

### Value

A `ddi_node` object.

### References

[point documentation](#)

[gringLat documentation](#)

[gringLon documentation](#)

### Examples

```
# ddi_point() which requires ddi_gringLat() and ddi_gringLon()
```

```
ddi_point(ddi_gringLat("42.002207"), ddi_gringLon("-120.005729004"))
```

---

`ddi_polygon`*polygon and its child nodes*

---

### Description

The minimum polygon that covers a geographical area, and is delimited by at least 4 points (3 sides), in which the last point coincides with the first point. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

```
ddi_polygon(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

polygon is contained in boundPoly.

### Value

A ddi\_node object.

### Shared and complex child nodes

- [ddi\\_point\(\)](#)

### References

[polygon documentation](#)

### Examples

```
# ddi_polygon requires ddi_point() which requires ddi_gringLat() and ddi_gringLon()
ddi_polygon(ddi_point(
  ddi_gringLat("42.002207"),
  ddi_gringLon("-120.005729004")
))
```

---

ddi_prodStmt	<i>prodStmt and its child nodes</i>
--------------	-------------------------------------

---

### Description

Production statement for the work at the appropriate level: marked-up document; marked-up document source; study; study description, other material; other material for study. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

ddi\_prodStmt(...)

ddi\_copyright(...)

ddi\_fundAg(...)

ddi\_grantNo(...)

ddi\_prodDate(...)

ddi\_prodPlac(...)

ddi\_software(...)

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

prodStmt is contained in the following elements: citation; docSrc; fileCitation; and sourceCitation.

#### *prdStmt specific child nodes*

ddi\_copyright() is the copyright statement for the work at the appropriate level. Copyright for data collection (codeBook/stdyDscr/citation/prodStmt/copyright) maps to Dublin Core Rights. Inclusion of this element is recommended.

ddi\_fundAg() is the source(s) of funds for production of the work. If different funding agencies sponsored different stages of the production process, use the "role" attribute to distinguish them.

ddi\_grantNo() is the grant/contract number of the project that sponsored the effort. If more than one, indicate the appropriate agency using the "agency" attribute. If different funding agencies sponsored different stages of the production process, use the "role" attribute to distinguish the grant numbers.

ddi\_prodDate() is the date when the marked-up document/marked-up document source/data collection/other material(s) were produced (not distributed or archived). The ISO standard for dates

(YYYY-MM-DD) is recommended for use with the date attribute. Production date for data collection (codeBook/stdyDscr/citation/prodStmt/prodDate) maps to Dublin Core Date element.

ddi\_prodPlac() is the address of the archive or organization that produced the work.

ddi\_software() is the software used to produce the work. A "version" attribute permits specification of the software version number. The "date" attribute is provided to enable specification of the date (if any) for the software release. The ISO standard for dates (YYYY-MM-DD) is recommended for use with the date attribute.

## Value

A ddi\_node object.

## References

[prodStmt documentation](#)

[copyright documentation](#)

[fundAg documentation](#)

[grantNo documentation](#)

[prodDate documentation](#)

[prodPlac documentation](#)

[software documentation](#)

## Examples

```
ddi_prodStmt()

# Functions that need to be wrapped in ddi_prodStmt()

ddi_copyright("Copyright(c) ICPSR, 2000")

ddi_fundAg(abbr = "NSF", role = "infrastructure", "National Science Foundation")

ddi_grantNo(agency = "Bureau of Justice Statistics", "J-LEAA-018-77")

ddi_prodDate(date = "2022-01-01", "January 1, 2022")

ddi_prodPlac("Place of production")

ddi_software(version = "6.12", "SAS")
```

---

ddi_producer	<i>producer node</i>
--------------	----------------------

---

### Description

The producer is the person or organization with the financial or administrative responsibility for the physical processes whereby the document was brought into existence. Use the "role" attribute to distinguish different stages of involvement in the production process, such as original producer. Producer of data collection (codeBook/stdyDscr/citation/prodStmt/producer) maps to Dublin Core Publisher element. The "producer" in the Document Description should be the agency or person that prepared the marked-up document. More information on this element, especially its allowed attributes, can be found in the references.

### Usage

```
ddi_producer(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

producer is contained in the following elements: prodStmt and standard.

### Value

A ddi\_node object.

### References

[producer documentation](#)

### Examples

```
ddi_producer(abbr = "MNPoll",  
             affiliation = "Minneapolis Star Tribune Newspaper",  
             role = "origianl producer",  
             "Star Tribune Minnesota Poll")
```

---

 ddi\_qstn

*qstn and its child nodes*


---

### Description

- `ddi_qstn()` is the question asked. The element may have mixed content. The element itself may contain text for the question, with the subelements being used to provide further information about the question. Alternatively, the question element may be empty and only the subelements used. The element has a unique question ID attribute which can be used to link a variable with other variables where the same question has been asked. This would allow searching for all variables that share the same question ID, perhaps because the questions was asked several times in a panel design.

### Usage

```
ddi_qstn(...)
```

```
ddi_backward(...)
```

```
ddi_forward(...)
```

```
ddi_ivuInstr(...)
```

```
ddi_postQTxt(...)
```

```
ddi_preQTxt(...)
```

```
ddi_qstnLit(...)
```

### Arguments

```
...          Child nodes or attributes.
```

### Details

# *Parent nodes*

`qstn` is contained in `var`.

*qstn specific child nodes*

- `ddi_backward()` contains a reference to IDs of possible preceding questions. The "qstn" IDREFS may be used to specify the question IDs.
- `ddi_forward()` contains a reference to IDs of possible following questions. The "qstn" IDREFS may be used to specify the question IDs.
- `ddi_ivuInstr()` are specific instructions to the individual conducting an interview.
- `ddi_postQTxt()` is the text describing what occurs after the literal question has been asked.



- `ddi_preQTxt()` is the pre-question text. This is the text describing a set of conditions under which a question might be asked.
- `ddi_qstnLit()` is the text of the actual, literal question asked.

### Value

A `ddi_node` object.

### References

[qstn documentation](#)

[backward documentation](#)

[forward documentation](#)

[ivuInstr documentation](#)

[postQTxt documentation](#)

[preQTxt documentation](#)

[qstnLit documentation](#)

### Examples

```
ddi_qstn("When you get together with your friends, would you say you discuss
political matters frequently, occasionally, or never", ID = "Q125")

# Functions that need to be wrapped in ddi_qstn()

# Including ddi_preQTxt within a ddi_qstn with content
ddi_qstn("When you get together with your friends, would you say you discuss
political matters frequently, occasionally, or never", ID = "Q125",
        ddi_preQTxt("For those who did not go away on a holiday of four days or more in 1985..."))

ddi_qstn(ddi_postQTxt("The next set of questions will ask about your financial situation"))

# Using IDREFS in ddi_backward() and ddi_forward()

ddi_backward(qstn = "Q143")

ddi_forward("If yes, please ask questions 120-124", qstn = "Q120 Q121 Q122 Q123 Q124")

# Other child elements

ddi_ivuInstr("Please prompt the respondent if they are reticent to answer this question.",
            lang = "en")

ddi_qstnLit("Why didn't you go away in 1985?")
```

---

ddi\_qualityStatement *qualityStatement and its child nodes*

---

### Description

The Quality Statement consists of two parts, standardsCompliance and otherQualityStatement. In standardsCompliance list all specific standards complied with during the execution of this study. Note the standard name and producer and how the study complied with the standard. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

```
ddi_qualityStatement(...)
```

```
ddi_otherQualityStatement(...)
```

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

qualityStatement is contained in stdyInfo.

#### *qualityStatement specific child nodes*

- `ddi_otherQualityStatement()` holds additional quality statements.

### Value

A ddi\_node object.

### Shared and complex child nodes

- [ddi\\_standardsCompliance\(\)](#)

### References

[qualityStatement documentation](#)

[otherQualityStatement documentation](#)

### Examples

```
ddi_qualityStatement()
```

```
# Functions that need to be wrapped in ddi_qualityStatement()
```

```
ddi_otherQualityStatement("Additional quality statements not addressed in standardsCompliance.")
```

---

ddi_range	<i>range node</i>
-----------	-------------------

---

## Description

This is the actual range of values. The "UNITS" attribute permits the specification of integer/real numbers. The "min" and "max" attributes specify the lowest and highest values that are part of the range. The "minExclusive" and "maxExclusive" attributes specify values that are immediately outside the range. This is an empty element consisting only of its attributes. More information on this element, especially its allowed attributes, can be found in the references.

## Usage

```
ddi_range(...)
```

## Arguments

... Child nodes or attributes.

## Details

*Parent nodes*

range is contained in the following elements: valrng; invalrng; and cohort.

## Value

A ddi\_node object.

## References

[range documentation](#)

## Examples

```
ddi_range(min = "1", maxExclusive = "20")
```

---

ddi\_recGrp                      *recGrp and its child nodes*

---

## Description

Used to describe record groupings if the file is hierarchical or relational. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_recGrp(...)
```

## Arguments

...                      Child nodes or attributes.

## Details

*Parent nodes*

recGrp is contained in fileStrc.

## Value

A ddi\_node object.

## Shared and complex child nodes

- [ddi\\_labl\(\)](#)
- [ddi\\_recDimnsn\(\)](#)

## References

[recGrp documentation](#)

## Examples

```
ddi_recGrp()
```

---

ddi\_resource                      *resource and its child nodes*

---

## Description

Resources used in the development of the activity. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_resource(...)
```

## Arguments

...                      Child nodes or attributes.

## Details

*Parent nodes*

resource is contained in developmentActivity.

## Value

A ddi\_node object.

## Shared and complex child nodes

- [ddi\\_dataSrc\(\)](#)
- [ddi\\_srcChar\(\)](#)
- [ddi\\_srcDocu\(\)](#)
- [ddi\\_srcOrig\(\)](#)

## References

[resource documentation](#)

## Examples

```
ddi_resource()
```

---

ddi_respUnit	<i>respUnit node</i>
--------------	----------------------

---

### Description

Provides information regarding who provided the information contained within the variable/nCube, e.g., respondent, proxy, interviewer. This element may be repeated only to support multiple language expressions of the content. More information on this element, especially its allowed attributes, can be found in the references.

### Usage

```
ddi_respUnit(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

respUnit is contained in nCube and var.

### Value

A ddi\_node object.

### References

[respUnit documentation](#)

### Examples

```
ddi_respUnit("Head of household")
```

---

ddi_row	<i>row and its child nodes</i>
---------	--------------------------------

---

### Description

Each row represents a table row. More information on this element, especially the allowed attributes, can be found in the references.

### Usage

```
ddi_row(...)
```

```
ddi_entry(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

row can be found in tbody and thead.

*Child node*

entry is each table entry in the row.

### Value

A ddi\_node object.

### References

[row documentation](#)

[entry documentation](#)

### Examples

```
ddi_row()
```

```
# Functions that need to be wrapped in ddi_row()
```

```
ddi_entry("row contents")
```

---

 ddi\_rspStmt

*rspStmt and its child nodes*


---

### Description

Responsibility statement for the creation of the work at the appropriate level: marked-up document; marked-up document source; study; study description, other material; other material for study. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

ddi\_rspStmt(...)

ddi\_AuthEnty(...)

ddi\_othId(...)

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

rspStmt is contained in the following elements: citation; docSrc; fileCitation; and sourceCitation.

#### *rspStmt specific child nodes*

ddi\_AuthEnty() is the person, corporate body, or agency responsible for the work's substantive and intellectual content. Repeat the element for each author, and use "affiliation" attribute if available. Invert first and last name and use commas. Author of data collection (codeBook/stdyDscr/citation/rspStmt/AuthEnty) maps to Dublin Core Creator element. Inclusion of this element in codebook is recommended.

The "author" in the Document Description should be the individual(s) or organization(s) directly responsible for the intellectual content of the DDI version, as distinct from the person(s) or organization(s) responsible for the intellectual content of the earlier paper or electronic edition from which the DDI edition may have been derived.

ddi\_othId() are the statements of responsibility not recorded in the title and statement of responsibility areas. Indicate here the persons or bodies connected with the work, or significant persons or bodies connected with previous editions and not already named in the description. For example, the name of the person who edited the marked-up documentation might be cited in codeBook/docDscr/rspStmt/othId, using the "role" and "affiliation" attributes. Other identifications/acknowledgments for data collection (codeBook/stdyDscr/citation/rspStmt/othId) maps to Dublin Core Contributor element.

### Value

A ddi\_node object.



## References

[rspStmt documentation](#)

[AuthEnty documentation](#)

[othId documentation](#)

## Examples

```
ddi_rspStmt()

# Functions that need to be wrapped in ddi_rspStmt()

ddi_AuthEnty(affiliation = "Organization name",
             "LastName, FirstName")

ddi_othId(role = "Data Manager",
          affiliation = "Organization name",
          "LastName, FirstName")
```

---

ddi_sampleFrame	<i>sampleFrame and its children</i>
-----------------	-------------------------------------

---

## Description

Sample frame describes the sampling frame used for identifying the population from which the sample was taken. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_sampleFrame(...)

ddi_custodian(...)

ddi_referencePeriod(...)

ddi_sampleFrameName(...)

ddi_updateProcedure(...)

ddi_validPeriod(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

sampleFrame is contained in dataColl.

### *sampleFrame specific child nodes*

- ddi\_custodian() identifies the agency or individual who is responsible for creating or maintaining the sample frame.
- ddi\_referencePeriod() indicates the period of time in which the sampling frame was actually used for the study in question. Use ISO 8601 date/time formats to enter the relevant date(s).
- ddi\_sampleFrameName() is the name of the sample frame.
- ddi\_updateProcedure() is the description of how and with what frequency the sample frame is updated.
- ddi\_validPeriod() defines a time period for the validity of the sampling frame. Enter dates in YYYY-MM-DD format.

## Value

A ddi\_node object.

## Shared and complex child nodes

- [ddi\\_frameUnit\(\)](#)
- [ddi\\_labl\(\)](#)
- [ddi\\_txt\(\)](#)
- [ddi\\_universe\(\)](#)
- [ddi\\_useStmt\(\)](#)

## References

[sampleFrame documentation](#)

[custodian documentation](#)

[referencePeriod documentation](#)

[sampleFrameName documentation](#)

[updateProcedure documentation](#)

[validPeriod documentation](#)

## Examples

```
ddi_sampleFrame()
```

```
# Functions that need to be wrapped in ddi_sampleFrame()
```

```
ddi_custodian("DEX Publications")
```

```

ddi_referencePeriod(event = "single",
                    "2009-06-01")

ddi_sampleFrameName("City of St. Paul Directory")

ddi_updateProcedure("Changes are collected as they occur through registration
                    and loss of phone number from the specified geographic
                    area. Data are compiled for the date June 1st of odd
                    numbered years, and published on July 1st for the following
                    two-year period.")

ddi_validPeriod(event = "start", "2009-07-01")

```

---

ddi\_security

*security node*


---

### Description

Provides information regarding levels of access, e.g., public, subscriber, need to know. The ISO standard for dates (YYYY-MM-DD) is recommended for use with the date attribute. More information on this element, especially its allowed attributes, can be found in the references.

### Usage

```
ddi_security(...)
```

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

security is contained in nCube and var.

### Value

A ddi\_node object.

### References

[security documentation](#)

### Examples

```

ddi_security(date = "1998-05-10",
            "This variable has been recoded for reasons of confidentiality.
            Users should contact the archive for information on obtaining access.")

```

---

ddi_serStmt	<i>serStmt and its child nodes</i>
-------------	------------------------------------

---

### Description

Series statement for the work at the appropriate level: marked-up document; marked-up document source; study; study description, other material; other material for study. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

```
ddi_serStmt(...)
```

```
ddi_serInfo(...)
```

```
ddi_serName(...)
```

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

serStmt is contained in the following elements: citation; docSrc; fileCitation; and sourceCitation.

#### *serStmt specific child nodes*

ddi\_serInfo() is the series information. This element contains a history of the series and a summary of those features that apply to the series as a whole.

ddi\_serName() is the name of the series to which the work belongs.

### Value

A ddi\_node object.

### References

[serStmt documentation](#)

[serInfo documentation](#)

[serName documentation](#)

**Examples**

```
ddi_serStmt()  
  
# Functions that need to be wrapped in ddi_serStmt()  
  
ddi_serInfo("Series abstract...")  
  
ddi_serName(abbr="SN", "Series Name")
```

---

ddi_setAvail	<i>setAvail and its children</i>
--------------	----------------------------------

---

**Description**

Information on availability and storage of the data set collection. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_setAvail(...)  
ddi_accsPlac(...)  
ddi_avlStatus(...)  
ddi_collSize(...)  
ddi_complete(...)  
ddi_fileQnty(...)  
ddi_origArch(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent node*

setAvail is contained in dataAccs.

*setAvail specific child nodes*

ddi\_accsPlac() is the location where the data collection is currently stored. Use the URI attribute to provide a URN or URL for the storage site or the actual address from which the data may be downloaded.

ddi\_avlStatus() is the statement of collection availability. An archive may need to indicate that a collection is unavailable because it is embargoed for a period of time, because it has been superseded, because a new edition is imminent, etc.

ddi\_collSize() summarizes the number of physical files that exist in a collection, recording the number of files that contain data and noting whether the collection contains machine-readable documentation and/or other supplementary files and information such as data dictionaries, data definition statements, or data collection instruments.

ddi\_complete() is the completeness of study stored. This item indicates the relationship of the data collected to the amount of data coded and stored in the data collection. Information as to why certain items of collected information were not included in the data file stored by the archive should be provided.

ddi\_fileQty() is the total number of physical files associated with a collection.

ddi\_origArch() is the archive from which the data collection was obtained; the originating archive.

### Value

A ddi\_node object.

### Shared and complex child nodes

- [ddi\\_notes\(\)](#)

### References

[setAvail documentation](#)  
[accsPlac documentation](#)  
[avlStatus documentation](#)  
[collSize documentation](#)  
[complete documentation](#)  
[fileQty documentation](#)  
[origArch documentation](#)

### Examples

```
ddi_setAvail()

# Functions that need to be wrapped in ddi_setAvail()

ddi_accsPlac(URI = "https://dataverse.harvard.edu/",
            "Harvard Dataverse")

ddi_avlStatus("This collection is superseded by CENSUS OF POPULATION, 1880...")

ddi_collSize("1 data file + machine-readable documentation (PDF) + SAS data definition statements.")

ddi_complete("Because of embargo provisions, data values for some variables have been masked...")
```

```
ddi_fileQty("5 files")  
ddi_origArch("Zentralarchiv fuer empirische Sozialforschung")
```

---

ddi\_sources                      *sources and its child nodes*

---

## Description

Description of sources used for the data collection. The element is nestable so that the sources statement might encompass a series of discrete source statements, each of which could contain the facts about an individual source. This element maps to Dublin Core Source element. More information on this element, especially its allowed attributes, can be found in the references.

## Usage

```
ddi_sources(...)
```

## Arguments

...                      Child nodes or attributes.

## Details

*Parent nodes*

sources is contained in the following elements: dataColl and sources.

## Value

A ddi\_node object.

## Shared and complex child nodes

- [ddi\\_dataSrc\(\)](#)
- [ddi\\_sourceCitation\(\)](#)
- [ddi\\_sources\(\)](#)
- [ddi\\_srcChar\(\)](#)
- [ddi\\_srcDocu\(\)](#)
- [ddi\\_srcOrig\(\)](#)

## References

[sources documentation](#)

## Examples

```
ddi_sources()
```

---

ddi_srcChar	<i>srcChar node</i>
-------------	---------------------

---

**Description**

Assessment of characteristics and quality of source material. May not be relevant to survey data. This element may be repeated to support multiple language expressions of the content. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_srcChar(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

srcChar is contained in the following elements: sources and resource.

**Value**

A ddi\_node object.

**References**

[srcChar documentation](#)

**Examples**

```
ddi_srcChar("Assessment of source material(s).")
```

---

ddi_srcDocu	<i>srcDocu node</i>
-------------	---------------------

---

**Description**

Level of documentation of the original sources. May not be relevant to survey data. This element may be repeated to support multiple language expressions of the content. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_srcDocu(...)
```



**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

srcDocu is contained in the following elements: sources and resource.

**Value**

A ddi\_node object.

**References**

[srcDocu documentation](#)

**Examples**

```
ddi_srcDocu("Description of documentation of source material(s).")
```

---

ddi\_srcOrig

*srcOrig documentation*

---

**Description**

For historical materials, information about the origin(s) of the sources and the rules followed in establishing the sources should be specified. May not be relevant to survey data. This element may be repeated to support multiple language expressions of the content. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_srcOrig(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

srcOrig is contained in the following elements: sources and resource.

**Value**

A ddi\_node object.

## References

[srcOrig documentation](#)

## Examples

```
ddi_srcOrig("Origin of source material(s).")
```

---

ddi_standard	<i>standard and its child nodes</i>
--------------	-------------------------------------

---

## Description

Standard describes a standard with which the study complies. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_standard(...)
```

```
ddi_standardName(...)
```

## Arguments

... Child nodes or attributes.

## Details

*Parent nodes*

standard is contained in standardsCompliance.

*standard specific child nodes*

- `ddi_standardName()` contains the name of the standard with which the study complies.

## Value

A `ddi_node` object.

## Shared and complex child nodes

- [ddi\\_producer\(\)](#)

## References

[standard documentation](#)

[standardName documentation](#)

## Examples

```
ddi_standard()

# Functions that need to be wrapped in ddi_standard()

ddi_standardName(date = "2009-10-18",
                 version = "3.1",
                 URI = "http://www.ddialliance.org/Specification/DDI-Lifecycle/3.1/",
                 "Data Documentation Initiative")
```

---

ddi\_standardsCompliance

*standardsCompliance and its child nodes*

---

## Description

The standards compliance section lists all specific standards complied with during the execution of this study. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_standardsCompliance(...)

ddi_complianceDescription(...)
```

## Arguments

... Child nodes or attributes.

## Details

*Parent nodes*

standardsCompliance is contained in qualityStatement.

*standardsCompliance specific child nodes*

- ddi\_complianceDescription describes how the study complied with each standard.

## Value

A ddi\_node object.

## Shared and complex child nodes

- [ddi\\_standard\(\)](#)

**References**

[standardsCompliance documentation](#)  
[complianceDescription documentation](#)

**Examples**

```
# Note: ddi_standard() is a required child for ddi_standardsCompliance()
ddi_standardsCompliance(ddi_standard())

# Functions that need to be wrapped in ddi_standardsCompliance()

ddi_complianceDescription("This study complied to X standard by...")
```

---

ddi_stdyDscr	<i>stdyDscr and its children</i>
--------------	----------------------------------

---

**Description**

All DDI codebooks must have a study description which contains information about the study overall. The Study Description consists of information about the data collection, study, or compilation that the DDI-compliant documentation file describes. This section includes information about how the study should be cited, who collected or compiled the data, who distributes the data, keywords about the content of the data, summary (abstract) of the content of the data, data collection methods and processing, etc. At least one citation must be present, capturing the whole study. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_stdyDscr(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent node*

stdyDscr is contained in codeBook.

**Value**

A ddi\_node object

**Shared and complex child nodes**

- [ddi\\_citation\(\)](#)
- [ddi\\_dataAccs\(\)](#)
- [ddi\\_method\(\)](#)
- [ddi\\_notes\(\)](#)
- [ddi\\_othrStdyMat\(\)](#)
- [ddi\\_stdyInfo\(\)](#)
- [ddi\\_studyAuthorization\(\)](#)
- [ddi\\_studyDevelopment\(\)](#)

**References**

[stdyDscr documentation](#)

**Examples**

```
# ddi_citation() is required in ddi_stdyDscr()
ddi_stdyDscr(ddi_citation())
```

---

ddi\_stdyInfo

*stdyInfo and its child nodes*

---

**Description**

stdyInfo is the study scope. It contains information about the data collection's scope across several dimensions, including substantive content, geography, and time. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_stdyInfo(...)  
  
ddi_abstract(...)  
  
ddi_studyBudget(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

stdyInfo is contained in stdyDscr.

*stdyInfo specific child nodes*

- `ddi_abstract()` is an unformatted summary describing the purpose, nature, and scope of the data collection, special characteristics of its contents, major subject areas covered, and what questions the PIs attempted to answer when they conducted the study. A listing of major variables in the study is important here. In cases where a codebook contains more than one abstract (for example, one might be supplied by the data producer and another prepared by the data archive where the data are deposited), the "source" and "date" attributes may be used to distinguish the abstract versions. Maps to Dublin Core Description element. Inclusion of this element in the codebook is recommended. The "date" attribute should follow ISO convention of YYYY-MM-DD.
- `ddi_studyBudget()` is used to describe the budget of the project in as much detail as needed.

**Value**

A `ddi_node` object.

**Shared and complex child nodes**

- `ddi_exPostEvaluation()`
- `ddi_notes()`
- `ddi_qualityStatement()`
- `ddi_subject()`
- `ddi_sumDscr()`

**References**

[stdyInfo documentation](#)

[abstract documentation](#)

[studyBudget documentation](#)

**Examples**

```
ddi_stdyInfo()

# Functions that need to be wrapped in ddi_stdyInfo()

ddi_abstract(date = "1999-01-28",
             contentType = "abstract",
             "Data on labor force activity for the week prior to the survey
             are supplied in this collection. Information is available on the
             employment status, occupation, and industry of persons 15 years
             old and over. Demographic variables such as age, sex, race, marital
             status, veteran status, household relationship, educational
```

background, and Hispanic origin are included. In addition to providing these core data, the May survey also contains a supplement on work schedules for all applicable persons aged 15 years and older who were employed at the time of the survey. This supplement focuses on shift work, flexible hours, and work at home for both main and second jobs.”)

```
ddi_studyBudget("The budget for the study covers a 5 year award period
distributed between direct and indirect costs including:
Staff, ...")
```

---

```
ddi_studyAuthorization
```

*studyAuthorization and its child nodes*

---

## Description

Study Authorization provides structured information on the agency that authorized the study, the date of authorization, and an authorization statement. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_studyAuthorization(...)
ddi_authorizationStatement(...)
ddi_authorizingAgency(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

studyAuthorization is contained in stdyDscr.

### *studyAuthorization specific child nodes*

- ddi\_authorizationStatement() is the text of the authorization.
- ddi\_authorizingAgency() is the name of the agent or agency that authorized the study.

## Value

A ddi\_node object.

**References**

[studyAuthorization documentation](#)  
[authorizationStatement documentation](#)  
[authorizingAgency documentation](#)

**Examples**

```
ddi_studyAuthorization()

# Functions that have to be wrapped in ddi_studyAuthorization()

ddi_authorizationStatement("Required documentation covering the study purpose,
                             disclosure information, questionnaire content, and
                             consent statements was delivered to the OUHS on
                             2010-10-01 and was reviewed by the compliance officer.
                             Statement of authorization for the described study
                             was issued on 2010-11-04.")

ddi_authorizingAgency(affiliation = "Purdue University",
                       abbr = "OUHS",
                       "Office for Use of Human Subjects")
```

---

ddi\_studyDevelopment    *studyDevelopment and its child nodes*

---

**Description**

Describe the process of study development as a series of development activities. These activities can be typed using a controlled vocabulary. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_studyDevelopment(...)
```

**Arguments**

...                    Child nodes or attributes.

**Details**

*Parent nodes*

studyDevelopment is contained in stdyDscr.

**Value**

A ddi\_node object.



**Shared and complex child nodes**

- [ddi\\_developmentActivity\(\)](#)

**References**

[studyDevelopment documentation](#)

**Examples**

```
ddi_studyDevelopment()
```

---

ddi\_subject

*subject and its child nodes*

---

**Description**

Subject describes the data collection's intellectual content. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_subject(...)
```

```
ddi_keyword(...)
```

```
ddi_topcClas(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent nodes*

subject is contained in `stdyInfo`.

*subject specific child nodes*

- `ddi_keyword()` are words or phrases that describe salient aspects of a data collection's content. Can be used for building keyword indexes and for classification and retrieval purposes. A controlled vocabulary can be employed. Maps to Dublin Core Subject element.
- `ddi_topcClas()` indicates the broad substantive topic(s) that the data cover. Library of Congress subject terms may be used here. Maps to Dublin Core Subject element. Inclusion of this element in the codebook is recommended.

**Value**

A `ddi_node` object.

## References

[subject documentation](#)  
[keyword documentation](#)  
[topcClas documentation](#)

## Examples

```
ddi_subject()

# Functions that need to be wrapped in ddi_subject()

ddi_keyword(vocab = "ICPSR Subject Thesaurus",
            vocabURI = "http://www.icpsr.umich.edu/thesaurus/subject.html",
            "quality of life")

ddi_topcClas(vocab = "LOC Subject Headings",
            vocabURI = "http://www.loc.gov/catdir/cpsol/lcco/lcco.html",
            "Public opinion -- California -- Statistics")
```

---

ddi\_sumDscr

*sumDscr and its child nodes*

---

## Description

This is the summary data description and it contains information about the geographic coverage of the study and unit of analysis. More information on these elements, especially their allowed attributes, can be found in the references.

## Usage

```
ddi_sumDscr(...)

ddi_anlyUnit(...)

ddi_collDate(...)

ddi_dataKind(...)

ddi_geogCover(...)

ddi_geogUnit(...)

ddi_nation(...)

ddi_timePrd(...)
```

## Arguments

... Child nodes or attributes.

## Details

### *Parent nodes*

sumDscr is contained in stdyInfo.

### *sumDscr specific child nodes*

- `ddi_anlyUnit()` is the basic unit of analysis or observation that the file describes: individuals, families/households, groups, institutions/organizations, administrative units, etc.
- `ddi_collDate()` contains the date(s) when the data were collected. Maps to Dublin Core Coverage element. Inclusion of this element in the codebook is recommended.
- `ddi_dataKind()` is the type of data included in the file: survey data, census/enumeration data, aggregate data, clinical data, event/transaction data, program source code, machine-readable text, administrative records data, experimental data, psychological test, textual data, coded textual, coded documents, time budget diaries, observation data/ratings, process-produced data, etc. This element maps to Dublin Core Type element.
- `ddi_geogCover()` is information on the geographic coverage of the data. Includes the total geographic scope of the data, and any additional levels of geographic coding provided in the variables. Maps to Dublin Core Coverage element.
- `ddi_geogUnit()` is the lowest level of geographic aggregation covered by the data.
- `ddi_nation()` indicates the country or countries covered in the file. Attribute "abbr" may be used to list common abbreviations; use of ISO country codes is recommended. Maps to Dublin Core Coverage element. Inclusion of this element is recommended.
- `ddi_timePrd()` is the time period to which the data refer. This item reflects the time period covered by the data, not the dates of coding or making documents machine-readable or the dates the data were collected. Also known as span. Maps to Dublin Core Coverage element. Inclusion of this element is recommended.

## Value

A `ddi_node` object.

## Shared and complex child nodes

- [ddi\\_boundPoly\(\)](#)
- [ddi\\_geoBndBox\(\)](#)
- [ddi\\_universe\(\)](#)

## References

[sumDscr documentation](#)

[anlyInfo documentation](#)

[collDate documentation](#)

[dataKind documentation](#)  
[geogCover documentation](#)  
[geogUnit documentation](#)  
[nation documentation](#)  
[timePrd documentation](#)

## Examples

```

ddi_sumDscr()

# Functions that need to be wrapped in ddi_sumDscr()

ddi_anlyUnit("individuals")

ddi_collDate(event = "single",
             date = "1998-11-10",
             "10 November 1998")

ddi_dataKind(type = "numeric",
             "survey data")

ddi_geogCover("State of California")

ddi_geogUnit("state")

ddi_nation(abbr = "GB",
           "United Kingdom")

ddi_timePrd(event = "start",
            date = "1998-05-01",
            "May 1, 1998")

```

---

ddi_table	<i>table and its child nodes</i>
-----------	----------------------------------

---

## Description

Used to create a table in DDI 2.5. More information on this element, especially the allowed attributes, can be found in the references.

## Usage

```
ddi_table(...)
```

## Arguments

... Child nodes or attributes.

**Details***Parent nodes*

table is contained in the following elements: key; notes; otherMat; and txt.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_tgroup\(\)](#)
- [ddi\\_titl\(\)](#)

**References**

[table documentation](#)

**Examples**

```
ddi_table()
```

---

ddi\_targetSampleSize    *targetSampleSize and its children nodes*

---

**Description**

Provides both the target size of the sample (this is the number in the original sample, not the number of respondents) as well as the formula used for determining the sample size. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_targetSampleSize(...)
```

```
ddi_sampleSize(...)
```

```
ddi_sampleSizeFormula(...)
```

**Arguments**

...                    Child nodes or attributes.

**Details***Parent nodes*

targetSampleSize is contained in dataColl.

*targetSampleSize specific child nodes*

- ddi\_sampleSize() provides the targeted sample size in integer format.
- ddi\_sampleSizeFormula() includes the formula that was used to determine the sample size.

**Value**

A ddi\_node object.

**References**

[targetSampleSize documentation](#)

[sampleSize documentation](#)

[sampleSizeFormula documentation](#)

**Examples**

```
ddi_targetSampleSize()

# Functions that need to be wrapped in ddi_targetSampleSize()

ddi_sampleSize(385)

ddi_sampleSizeFormula("n0=Z2pq/e2=(1.96)2(.5)(.5)/(.05)2=385 individuals")
```

---

ddi\_tbody

*tbody and its child nodes*

---

**Description**

This is the body of the table. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_tbody(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

tbody is contained in tgroup.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_row\(\)](#)

**References**

[tbody documentation](#)

**Examples**

```
ddi_tbody(valign = "middle")
```

---

ddi\_tgroup

*tgroup and its child nodes*


---

**Description**

This is the table group. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_tgroup(...)
```

```
ddi_colspec(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

tgroup is contained in table.

*tgroup specific child node*

- `ddi_colspec()` is the column specification for each column. It is an empty element.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_tbody\(\)](#)
- [ddi\\_thead\(\)](#)

**References**

[tgroup documentation](#)  
[colspec documentation](#)

**Examples**

```
ddi_tgroup()

# Functions that must be wrapped in ddi_tgroup()

ddi_colspec(align = "left")
```

---

ddi_thead	<i>thead and its child nodes</i>
-----------	----------------------------------

---

**Description**

This is the table header. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_thead(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent nodes*  
thead is contained in tgroup.

**Value**

A ddi\_node object.



**Shared and complex child nodes**

- [ddi\\_row\(\)](#)

**References**

[thead documentation](#)

**Examples**

```
ddi_thead(valign = "middle")
```

---

ddi_titl	<i>titl node</i>
----------	------------------

---

**Description**

titl is the full authoritative title for the work at the appropriate level: marked-up document; marked-up document source; study; other material(s) related to study description; other material(s) related to study. The study title will in most cases be identical to the title for the marked-up document. 'A full title should indicate the geographic scope of the data collection as well as the time period covered. Title of data collection '(codeBook/stdyDscr/citation/titlStmt/titl) maps to Dublin Core Title element. This element is required in the Study Description citation. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_titl(...)
```

**Arguments**

... Child nodes or attributes.

**Details**

*Parent nodes*

titl is contained in the following elements: table and titlStmt.

**Value**

A ddi\_node object.

**References**

[titl documentation](#)

**Examples**

```
ddi_titl("Census of Population, 1950 [United States]: Public Use Microdata Sample")
```

---

ddi_titlStmt	<i>titlStmt and its child nodes</i>
--------------	-------------------------------------

---

**Description**

Title statement for the work at the appropriate level: marked-up document; marked-up document source; study; study description, other materials; other materials for the study. Both `titlStmt` and `titl` are required elements in the `citation` branch of a DDI-Codebook. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_titlStmt(...)
```

```
ddi_altTitl(...)
```

```
ddi_IDNo(...)
```

```
ddi_parTitl(...)
```

```
ddi_subTitl(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

`titlStmt` is contained in the following elements: `citation`; `docSrc`; `fileCitation`; and `sourceCitation`.

*titlStmt specific child nodes*

`ddi_altTitl()` is the alternative title. A title by which the work is commonly referred, or an abbreviation of the title.

`ddi_IDNo()` is the identification number. This is a unique string or number (producer's or archive's number). Can be a DOI. An "agency" attribute is supplied. Identification Number of data collection maps to Dublin Core Identifier element.

`ddi_parTitl()` is the parallel title. Title translated into another language.

`ddi_subTitl()` is the subtitle. A secondary title used to amplify or state certain limitations on the main title. It may repeat information already in the main title.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_titl\(\)](#)

**References**

[titlStmt documentation](#)

[altTitl documentation](#)

[IDNo documentation](#)

[parTitl documentation](#)

[subTitl documentation](#)

**Examples**

```
ddi_titlStmt()  
  
# Functions that need to be wrapped in ddi_titlStmt()  
  
ddi_altTitl("Alternative Title of work")  
  
ddi_IDNo(agency = "agency name", "ID number")  
  
ddi_parTitl(lang = "fr", "French translation of the title")  
  
ddi_subTitl("Subtitle of work")
```

---

ddi_txt	<i>txt node</i>
---------	-----------------

---

**Description**

Lengthier description of the parent element. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_txt(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

txt is contained in the following elements: anlyUnit; anlysUnit; catgry; catgryGrp; codingInstructions; collMode; dataKind; frameUnit; geogCover; geogUnit; nCube; nCubeGrp; nation; otherMat; resInstru; sampProc; sampleFrame; srcOrig; timeMeth; universe; var; and varGrp.

**Value**

A ddi\_node object.

**References**

[txt documentation](#)

**Examples**

```
ddi_txt("The following five variables refer to respondent attitudes toward
national environmental policies: air pollution, urban sprawl, noise
abatement, carbon dioxide emissions, and nuclear waste.")
```

---

ddi\_universe

*universe node*

---

**Description**

The group of persons or other elements that are the object of research and to which any analytic results refer. Age, nationality, and residence commonly help to delineate a given universe, but any of a number of factors may be involved, such as sex, race, income, veteran status, criminal convictions, etc. The universe may consist of elements other than persons, such as housing units, court cases, deaths, countries, etc. In general, it should be possible to tell from the description of the universe whether a given individual or element (hypothetical or real) is a member of the population under study. More information on this element, especially its allowed attributes, can be found in the references.

**Usage**

```
ddi_universe(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

universe is contained in the following elements: nCube; nCubeGrp; sampleFrame; sumDscr; var; and varGrp.

**Value**

A ddi\_node object.

**References**

[universe documentation](#)

**Examples**

```
ddi_universe(clusion = "I", "Individuals 15-19 years of age.")
```

---

ddi\_usage

*usage and its child nodes*

---

**Description**

Defines where in the instance the controlled vocabulary which is identified is utilized. A controlled vocabulary may occur either in the content of an element or in an attribute on an element. The usage can either point to a collection of elements using an XPath via the selector element or point to a more specific collection of elements via their identifier using the specificElements element. If the controlled vocabulary occurs in an attribute within the element, the attribute element identifies the specific attribute. When specific elements are specified, an authorized code value may also be provided. If the current value of the element or attribute identified is not in the controlled vocabulary or is not identical to a code value, the authorized code value identifies a valid code value corresponding to the meaning of the content in the element or attribute. More information on this element, especially the allowed attributes, can be found in the references.

**Usage**

```
ddi_usage(...)
```

```
ddi_attribute(...)
```

```
ddi_selector(...)
```

```
ddi_specificElements(...)
```

**Arguments**

... Child nodes or attributes.

## Details

### *Parent nodes*

usage is contained in controlledVocabUsed.

### *usage specific child nodes*

- `ddi_attribute()` identifies an attribute within the element(s) identified by the selector or `specificElements` in which the controlled vocabulary is used. The fully qualified name used here must correspond to that in the instance, which is to say that if the attribute is namespace qualified, the prefix used here must match that which is defined in the instance.
- `ddi_selector()` identifies a collection of elements in which a controlled vocabulary is used. This is a simplified XPath which must correspond to the actual instance in which it occurs, which is to say that the fully qualified element names here must correspond to those in the instance. This XPath can only identify elements and does not allow for any predicates. The XPath must either be rooted or deep.
- `ddi_specificElements()` identifies a collection of specific elements via their identifiers in the `refs` attribute, which allows for a tokenized list of identifier values which must correspond to identifiers which exist in the instance. The `authorizedCodeValue` attribute can be used to provide a valid code value corresponding to the meaning of the content in the element or attribute when the identified element or attribute does not use an actual valid value from the controlled vocabulary.

## Value

A `ddi_node` object.

## References

[usage documentation](#)

[attribute documentation](#)

[selector documentation](#)

[specificElements documentation](#)

## Examples

```
ddi_usage(ddi_selector("/codeBook/stdyDscr/method/dataColl/timeMeth"))
```

```
ddi_usage(ddi_selector("/codeBook/stdyDscr/method/dataProcessing"), ddi_attribute("type"))
```

```
ddi_usage(ddi_specificElements(refs = "ICPSR4328timeMeth", authorizedCodeValue = "CrossSection"))
```

---

ddi_useStmt	<i>useStmt and its children</i>
-------------	---------------------------------

---

### Description

Information on terms of use for the data collection. This element may be repeated only to support multiple language expressions of the content. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

```

ddi_useStmt(...)
ddi_citReq(...)
ddi_conditions(...)
ddi_confDec(...)
ddi_deposReq(...)
ddi_disclaimer(...)
ddi_restrctn(...)
ddi_specPerm(...)

```

### Arguments

... Child nodes or attributes.

### Details

#### *Parent nodes*

useStmt is contained in the following elements: dataAccs and sampleFrame.

#### *useStmt specific child nodes*

- ddi\_citReq() is the citation requirement. This is the text of requirement that a data collection should be cited properly in articles or other publications that are based on analysis of the data.
- ddi\_conditions() indicates any additional information that will assist the user in understanding the access and use conditions of the data collection.
- ddi\_confDec() is the confidentiality declaration. This element is used to determine if signing of a confidentiality declaration is needed to access a resource.
- ddi\_deposReq() is the deposit requirement. This is information regarding user responsibility for informing archives of their use of data through providing citations to the published work or providing copies of the manuscripts.

- `ddi_disclaimer()` is information regarding responsibility for uses of the data collection. This element may be repeated to support multiple language expressions of the content.
- `ddi_restrctn()` are any restrictions on access to or use of the collection such as privacy certification or distribution restrictions should be indicated here. These can be restrictions applied by the author, producer, or disseminator of the data collection. If the data are restricted to only a certain class of user, specify which type.
- `ddi_specPerm()` is used to determine if any special permissions are required to access a resource.

### Value

A `ddi_node` object.

### Shared and complex child nodes

- [ddi\\_contact\(\)](#)

### References

[useStmt documentation](#)  
[citReq documentation](#)  
[conditions documentation](#)  
[confDec documentation](#)  
[deposReq documentation](#)  
[disclaimer documentation](#)  
[restrctn documentation](#)  
[specPerm documentation](#)

### Examples

```
ddi_useStmt()

# Functions that need to be wrapped in ddi_useStmt()

ddi_citReq(lang = "en",
  "Publications based on ICPSR data collections should acknowledge
  those sources by means of bibliographic citations. To ensure that
  such source attributions are captured for social science
  bibliographic utilities, citations must appear in footnotes or in
  the reference section of publications.")

ddi_conditions(lang = "en",
  "The data are available without restriction. Potential users
  of these datasets are advised, however, to contact the original
  principal investigator Dr. J. Smith (Institute for Social Research,
  The University of Michigan, Box 1248, Ann Arbor, MI 48106),
  about their intended uses of the data. Dr. Smith would also
  appreciate receiving copies of reports based on the datasets.")
```



```

ddi_confDec(formNo = "1",
            "To download this dataset, the user must sign a declaration of confidentiality.")

ddi_deposReq("To provide funding agencies with essential information about
            use of archival resources and to facilitate the exchange of
            information about ICPSR participants' research activities, users
            of ICPSR data are requested to send to ICPSR bibliographic
            citations for, or copies of, each completed manuscript or thesis
            abstract. Please indicate in a cover letter which data were used.")

ddi_disclaimer("The original collector of the data, ICPSR, and the relevant
            funding agency bear no responsibility for uses of this collection
            or for interpretations or inferences based upon such uses.")

ddi_restrctn("ICPSR obtained these data from the World Bank under the terms of
            a contract which states that the data are for the sole use of
            ICPSR and may not be sold or provided to third parties outside
            of ICPSR membership. Individuals at institutions that are not
            members of the ICPSR may obtain these data directly from the
            World Bank.")

ddi_specPerm(formNo = "4",
            "The user must apply for special permission to use this dataset
            locally and must complete a confidentiality form.")

```

---

ddi\_valrng

*valrng, invalrng, and their child nodes*


---

### Description

Values for a particular variable that represent legitimate responses (*valrng*) or illegitimate response (*invalrng*). Must include item or range as a child element.

### Usage

```
ddi_valrng(...)
```

```
ddi_invalrng(...)
```

```
ddi_item(...)
```

```
ddi_key(...)
```

### Arguments

... Child nodes or attributes.

**Details***Parent nodes*

valrng and invalrng are contained in var.

*valrng and invalrng specific child nodes*

ddi\_item() is the counterpart to range; used to encode individual values. This is an empty element consisting only of its attributes. The "UNITS" attribute permits the specification of integer/real numbers. The "VALUE" attribute specifies the actual value.

ddi\_key() is the range key. This element permits a listing of the category values and labels. While this information is coded separately in the Category element, there may be some value in having this information in proximity to the range of valid and invalid values. A table is permissible in this element.

**Value**

A ddi\_node object.

**Shared and complex child nodes**

- [ddi\\_notes\(\)](#)
- [ddi\\_range\(\)](#)

**References**

[valrng documentation](#)

[invalrng documentation](#)

[item documentation](#)

[key documentation](#)

**Examples**

# ddi\_valrng() and ddi\_invalrng() requires either the ddi\_item() or ddi\_range() child node.

```
ddi_valrng(ddi_item())
```

```
ddi_invalrng(ddi_item())
```

```
ddi_valrng(ddi_range())
```

```
ddi_invalrng(ddi_range())
```

# Functions that must be wrapped in ddi\_valrng() or ddi\_invalrng()

```
ddi_item(VALUE = "1")
```

```
ddi_key("05 (PSU) Parti Socialiste Unifie et extreme gauche (Lutte Ouvriere)
[United Socialists and extreme left (Workers Struggle)] 50 Les Verts
[Green Party] 80 (FN) Front National et extreme droite [National Front
and extreme right]")
```

---

ddi_var	<i>var and its child nodes</i>
---------	--------------------------------

---

### Description

This element describes all of the features of a single variable in a social science data file. The following elements are repeatable to support multi-language content: `anlysUnit`, `embargo`, `imputation`, `respUnit`, `security`, `TotlResp`. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

`ddi_var(varname, ...)`

`ddi_catLevel(...)`

`ddi_codInstr(...)`

`ddi_geomap(...)`

`ddi_stdCatgry(...)`

`ddi_sumStat(...)`

`ddi_TotlResp(...)`

`ddi_undocCod(...)`

`ddi_varFormat(...)`

### Arguments

<code>varname</code>	The variable name.
<code>...</code>	Child nodes or attributes.

### Details

#### *Parent nodes*

`var` is contained in `dataDscr`.

#### *var specific child nodes*

- `ddi_catLevel()` is used to describe the levels of the category hierarchy.
- `ddi_codInstr()` are coder instructions. These are any special instructions to those who converted information from one form to another for a particular variable. This might include the reordering of numeric information into another form or the conversion of textual information into numeric information.

- `ddi_geomap()` is a geographic map. This element is used to point, using a "URI" attribute, to an external map that displays the geography in question.
- `ddi_stdCatgry()` are standard category codes used in the variable, like industry codes, employment codes, or social class codes.
- `ddi_sumStat()` is one or more statistical measures that describe the responses to a particular variable and may include one or more standard summaries, e.g., minimum and maximum values, median, mode, etc.
- `ddi_TotlResp()` are the number of responses to this variable. This element might be used if the number of responses does not match added case counts. It may also be used to sum the frequencies for variable categories.
- `ddi_undocCod()` is the list of undocumented codes where the meaning of the values are unknown.
- `ddi_varFormat()` is the technical format of the variable in question.

### Value

A `ddi_node` object.

### Shared and complex child nodes

- `ddi_anlysUnit()`
- `ddi_catgry()`
- `ddi_catgryGrp()`
- `ddi_concept()`
- `ddi_derivation()`
- `ddi_embargo()`
- `ddi_imputation()`
- `ddi_invalrng()`
- `ddi_labl()`
- `ddi_location()`
- `ddi_notes()`
- `ddi_qstn()`
- `ddi_respUnit()`
- `ddi_security()`
- `ddi_txt()`
- `ddi_universe()`
- `ddi_valrng()`
- `ddi_verStmt()`

## References

[var documentation](#)

[catLevel documentation](#)

[codInstr documentation](#)

[geomap documentation](#)

[stdCatgry documentation](#)

[sumStat documentation](#)

[TotlResp documentation](#)

[undocCod documentation](#)

[varFormat documentation](#)

## Examples

```
ddi_var(varname = "var01")

# Functions that need to be wrapped in ddi_var()

ddi_catLevel(ID = "Level1", levelnm = "Broader sectors")

ddi_codInstr("Use the standard classification tables to present responses to
             the question: What is your occupation? into numeric codes.")

ddi_geomap(URI = "https://mapURL.com")

ddi_stdCatgry(date = "1981",
              "U. S. Census of Population and Housing, Classified Index of
              Industries and Occupations")

ddi_sumStat(type = "min", "0")

ddi_TotlResp("1,056")

ddi_undocCod("Responses for categories 9 and 10 are unavailable.")

ddi_varFormat(type = "numeric",
              formatname = "date.iso8601",
              schema = "XML-Data",
              category = "date",
              URI = "http://www.w3.org/TR/1998/NOTE-XML-data/",
              "19541022")
```

---

`ddi_varGrp`*varGrp and its child nodes*

---

### Description

A group of variables that may share a common subject, arise from the interpretation of a single question, or are linked by some other factor. Variable groups are created this way in order to permit variables to belong to multiple groups, including multiple subject groups such as a group of variables on sex and income, or to a subject and a multiple response group, without causing overlapping groups. Variables that are linked by use of the same question need not be identified by a Variable Group element because they are linked by a common unique question identifier in the Variable element. Note that as a result of the strict sequencing required by XML, all Variable Groups must be marked up before the Variable element is opened. That is, the mark-up author cannot mark up a Variable Group, then mark up its constituent variables, then mark up another Variable Group. More information on these elements, especially their allowed attributes, can be found in the references.

### Usage

`ddi_varGrp(...)``ddi_defntn(...)`

### Arguments

... Child nodes or attributes.

### Details

*Parent nodes*

`varGrp` is contained in `dataDscr`.

*varGrp specific child nodes*

- `ddi_defntn()` is the rationale for why the variable group was constituted.

### Value

A `ddi_node` object.

### Shared and complex child nodes

- [ddi\\_concept\(\)](#)
- [ddi\\_labl\(\)](#)
- [ddi\\_notes\(\)](#)
- [ddi\\_txt\(\)](#)
- [ddi\\_universe\(\)](#)

**References**

[varGrp documentation](#)

[defnfn documentation](#)

**Examples**

```
ddi_varGrp()

# Functions that need to be wrapped in ddi_varGrp()

ddi_defnfn("The following eight variables were only asked in Ghana.")
```

---

ddi\_verStmt

*verStmt and its child nodes*


---

**Description**

This is the version statement for the work at the appropriate level: marked-up document; marked-up document source; study; study description, other material; other material for study. More information on these elements, especially their allowed attributes, can be found in the references.

**Usage**

```
ddi_verStmt(...)

ddi_verResp(...)

ddi_version(...)
```

**Arguments**

... Child nodes or attributes.

**Details***Parent nodes*

verStmt is contained in the following elements: citation; docSrc; fileCitation; fileTxt; nCube; sourceCitation; and var.

*verStmt specific child nodes*

- ddi\_verResp() is the organization or person responsible for the version of the work.
- ddi\_version() is also known as release or edition. If there have been substantive changes in the data/documentation since their creation, this statement should be used at the appropriate level. The ISO standard for dates (YYYY-MM-DD) is recommended for use with the "date" attribute.

**Value**

A `ddi_node` object.

**Shared and complex child nodes**

- [ddi\\_notes\(\)](#)

**References**

[verStmt documentation](#)

[verResp documentation](#)

[version documentation](#)

**Examples**

```
ddi_verStmt()

# Functions that need to be wrapped in ddi_verStmt()

ddi_verResp("Zentralarchiv fuer Empirische Sozialforschung")

ddi_version(type = "edition",
            date = "1999-01-25",
            "Second ICPSR Edition")
```

---

validate\_codebook

*Validate generated codebook against DDI Codebook 2.5*

---

**Description**

Validates your constructed codebook against the DDI Codebook 2.5 schema. While all built-in `ddi_` functions are written with the schema in mind, this is useful if you create your own DDI nodes (there are many and it will take a while to implement all of them).

**Usage**

```
validate_codebook(codebook)
```

**Arguments**

`codebook` The codebook root node, output of `ddi_codeBook()`.

**Value**

A logical (with attributes containing any errors) that indicates passing or failing.



**Examples**

```
cb <- ddi_codeBook(ddi_stdyDscr(ddi_citation(ddi_titlStmt(ddi_titl("Sample")))))  
validate_codebook(cb)
```

# Index

as\_ddi, 3  
as\_xml, 4  
as\_xml\_string, 5

ddi\_abstract (ddi\_stdyInfo), 85  
ddi\_accsPlac (ddi\_setAvail), 77  
ddi\_actMin (ddi\_dataColl), 19  
ddi\_algorithmSpecification  
    (ddi\_dataFingerprint), 24  
ddi\_algorithmVersion  
    (ddi\_dataFingerprint), 24  
ddi\_altTitl (ddi\_titlStmt), 98  
ddi\_anlyInfo, 5  
ddi\_anlyInfo(), 50  
ddi\_anlysUnit, 7  
ddi\_anlysUnit(), 53, 108  
ddi\_anlyUnit (ddi\_sumDscr), 90  
ddi\_attribute (ddi\_usage), 101  
ddi\_AuthEnty (ddi\_rspStmt), 72  
ddi\_authorizationStatement  
    (ddi\_studyAuthorization), 87  
ddi\_authorizingAgency  
    (ddi\_studyAuthorization), 87  
ddi\_avlStatus (ddi\_setAvail), 77  
ddi\_backward (ddi\_qstn), 64  
ddi\_biblCit (ddi\_citation), 10  
ddi\_biblCit(), 36  
ddi\_boundPoly, 7  
ddi\_boundPoly(), 91  
ddi\_caseQnty (ddi\_dimensns), 30  
ddi\_catgry, 9  
ddi\_catgry(), 108  
ddi\_catgryGrp (ddi\_catgry), 9  
ddi\_catgryGrp(), 108  
ddi\_catLevel (ddi\_var), 107  
ddi\_catStat (ddi\_catgry), 9  
ddi\_catValu (ddi\_catgry), 9  
ddi\_citation, 10  
ddi\_citation(), 35, 56, 85  
ddi\_citReq (ddi\_useStmt), 103

ddi\_cleanOps (ddi\_dataColl), 19  
ddi\_codeBook, 12  
ddi\_codeListAgencyName  
    (ddi\_controlledVocabUsed), 17  
ddi\_codeListID  
    (ddi\_controlledVocabUsed), 17  
ddi\_codeListName  
    (ddi\_controlledVocabUsed), 17  
ddi\_codeListSchemeURN  
    (ddi\_controlledVocabUsed), 17  
ddi\_codeListURN  
    (ddi\_controlledVocabUsed), 17  
ddi\_codeListVersionID  
    (ddi\_controlledVocabUsed), 17  
ddi\_codingInstructions, 13  
ddi\_codingInstructions(), 50  
ddi\_codInstr (ddi\_var), 107  
ddi\_cohort, 14  
ddi\_cohort(), 34  
ddi\_collDate (ddi\_sumDscr), 90  
ddi\_collectorTraining (ddi\_dataColl), 19  
ddi\_collMode (ddi\_dataColl), 19  
ddi\_collSitu (ddi\_dataColl), 19  
ddi\_collSize (ddi\_setAvail), 77  
ddi\_colspec (ddi\_tgroup), 95  
ddi\_command (ddi\_codingInstructions), 13  
ddi\_complete (ddi\_setAvail), 77  
ddi\_complianceDescription  
    (ddi\_standardsCompliance), 83  
ddi\_concept, 15  
ddi\_concept(), 54, 108, 110  
ddi\_conditions (ddi\_useStmt), 103  
ddi\_confDec (ddi\_useStmt), 103  
ddi\_ConOps (ddi\_dataColl), 19  
ddi\_contact, 16  
ddi\_contact(), 32, 104  
ddi\_controlledVocabUsed, 17  
ddi\_controlledVocabUsed(), 35  
ddi\_copyright (ddi\_prodStmt), 61

- ddi\_CubeCoord (ddi\_dataItem), 25
- ddi\_custodian (ddi\_sampleFrame), 73
- ddi\_dataAccs, 18
- ddi\_dataAccs(), 85
- ddi\_dataAppr (ddi\_anlyInfo), 5
- ddi\_dataChck (ddi\_fileTxt), 41
- ddi\_dataChck(), 41
- ddi\_dataColl, 19
- ddi\_dataColl(), 50
- ddi\_dataCollector (ddi\_dataColl), 19
- ddi\_dataDscr, 23
- ddi\_dataDscr(), 12
- ddi\_dataFingerprint, 24
- ddi\_dataFingerprint(), 42
- ddi\_dataItem, 25
- ddi\_dataItem(), 49
- ddi\_dataKind (ddi\_sumDscr), 90
- ddi\_dataMsng (ddi\_fileTxt), 41
- ddi\_dataMsng(), 41
- ddi\_dataProcessing (ddi\_method), 50
- ddi\_dataSrc, 26
- ddi\_dataSrc(), 69, 79
- ddi\_defntn (ddi\_varGrp), 110
- ddi\_defntn(), 54
- ddi\_depDate (ddi\_distStmt), 31
- ddi\_depositr (ddi\_distStmt), 31
- ddi\_deposReq (ddi\_useStmt), 103
- ddi\_derivation, 27
- ddi\_derivation(), 108
- ddi\_description
  - (ddi\_developmentActivity), 28
- ddi\_developmentActivity, 28
- ddi\_developmentActivity(), 89
- ddi\_deviat (ddi\_dataColl), 19
- ddi\_digitalFingerprintValue
  - (ddi\_dataFingerprint), 24
- ddi\_dimensns, 30
- ddi\_dimensns(), 42
- ddi\_disclaimer (ddi\_useStmt), 103
- ddi\_distDate (ddi\_distStmt), 31
- ddi\_distribtr (ddi\_distStmt), 31
- ddi\_distStmt, 31
- ddi\_distStmt(), 11, 36
- ddi\_dmns, 33
- ddi\_docDscr, 34
- ddi\_docDscr(), 12
- ddi\_docSrc, 36
- ddi\_docSrc(), 35
- ddi\_docStatus (ddi\_docDscr), 34
- ddi\_drvcmd (ddi\_derivation), 27
- ddi\_drvdesc (ddi\_derivation), 27
- ddi\_eastBL (ddi\_geoBndBox), 44
- ddi\_embargo, 37
- ddi\_embargo(), 53, 108
- ddi\_entry (ddi\_row), 71
- ddi\_EstSmpErr (ddi\_anlyInfo), 5
- ddi\_evaluationProcess
  - (ddi\_exPostEvaluation), 38
- ddi\_evaluator (ddi\_exPostEvaluation), 38
- ddi\_exPostEvaluation, 38
- ddi\_exPostEvaluation(), 86
- ddi\_fileCitation (ddi\_citation), 10
- ddi\_fileCitation(), 42
- ddi\_fileCont (ddi\_fileTxt), 41
- ddi\_fileCont(), 41
- ddi\_fileDscr, 39
- ddi\_fileDscr(), 12
- ddi\_fileName (ddi\_fileTxt), 41
- ddi\_fileName(), 42
- ddi\_filePlac (ddi\_fileTxt), 41
- ddi\_filePlac(), 42
- ddi\_fileQty (ddi\_setAvail), 77
- ddi\_fileStrc, 40
- ddi\_fileStrc(), 42
- ddi\_fileTxt, 41
- ddi\_fileTxt(), 39
- ddi\_fileType (ddi\_fileTxt), 41
- ddi\_fileType(), 42
- ddi\_format (ddi\_fileTxt), 41
- ddi\_format(), 42
- ddi\_forward (ddi\_qstn), 64
- ddi\_frameUnit, 43
- ddi\_frameUnit(), 74
- ddi\_frequenc (ddi\_dataColl), 19
- ddi\_fundAg (ddi\_prodStmt), 61
- ddi\_geoBndBox, 44
- ddi\_geoBndBox(), 91
- ddi\_geogCover (ddi\_sumDscr), 90
- ddi\_geogUnit (ddi\_sumDscr), 90
- ddi\_geomap (ddi\_var), 107
- ddi\_grantNo (ddi\_prodStmt), 61
- ddi\_gringLat (ddi\_point), 59
- ddi\_gringLon (ddi\_point), 59
- ddi\_guide (ddi\_docDscr), 34
- ddi\_holdings (ddi\_citation), 10
- ddi\_holdings(), 36

- ddi\_IDNo (ddi\_titlStmt), 98
- ddi\_imputation, 46
- ddi\_imputation(), 53, 108
- ddi\_instrumentDevelopment (ddi\_dataColl), 19
- ddi\_invalrng (ddi\_valrng), 105
- ddi\_invalrng(), 108
- ddi\_item (ddi\_valrng), 105
- ddi\_ivuInstr (ddi\_qstn), 64
- ddi\_key (ddi\_valrng), 105
- ddi\_keyword (ddi\_subject), 89
- ddi\_labl, 47
- ddi\_labl(), 9, 53, 54, 56, 68, 74, 108, 110
- ddi\_location, 48
- ddi\_location(), 53, 108
- ddi\_locMap, 49
- ddi\_locMap(), 39
- ddi\_logRecl (ddi\_dimensns), 30
- ddi\_measure (ddi\_nCube), 52
- ddi\_method, 50
- ddi\_method(), 85
- ddi\_mi (ddi\_mrow), 51
- ddi\_mrow, 51
- ddi\_mrow(), 9
- ddi\_nation (ddi\_sumDscr), 90
- ddi\_nCube, 52
- ddi\_nCube(), 23
- ddi\_nCubeGrp, 54
- ddi\_nCubeGrp(), 23
- ddi\_northBL (ddi\_geoBndBox), 44
- ddi\_notes, 55
- ddi\_notes(), 11, 19, 23, 35, 36, 39, 40, 50, 53, 54, 56, 78, 85, 86, 106, 108, 110, 112
- ddi\_origArch (ddi\_setAvail), 77
- ddi\_otherMat, 56
- ddi\_otherMat(), 12, 56
- ddi\_otherQualityStatement (ddi\_qualityStatement), 66
- ddi\_othId (ddi\_rspStmt), 72
- ddi\_othRefs (ddi\_othrStdyMat), 57
- ddi\_othrStdyMat, 57
- ddi\_othrStdyMat(), 85
- ddi\_outcome (ddi\_developmentActivity), 28
- ddi\_outcomes (ddi\_exPostEvaluation), 38
- ddi\_participant (ddi\_developmentActivity), 28
- ddi\_parTitl (ddi\_titlStmt), 98
- ddi\_physLoc (ddi\_dataItem), 25
- ddi\_point, 59
- ddi\_point(), 60
- ddi\_polygon, 60
- ddi\_polygon(), 8
- ddi\_postQTxt (ddi\_qstn), 64
- ddi\_preQTxt (ddi\_qstn), 64
- ddi\_ProcStat (ddi\_fileTxt), 41
- ddi\_ProcStat(), 42
- ddi\_prodDate (ddi\_prodStmt), 61
- ddi\_prodPlac (ddi\_prodStmt), 61
- ddi\_prodStmt, 61
- ddi\_prodStmt(), 11, 36
- ddi\_producer, 63
- ddi\_producer(), 82
- ddi\_purpose (ddi\_nCube), 52
- ddi\_qstn, 64
- ddi\_qstn(), 108
- ddi\_qstnLit (ddi\_qstn), 64
- ddi\_qualityStatement, 66
- ddi\_qualityStatement(), 86
- ddi\_range, 67
- ddi\_range(), 14, 106
- ddi\_recDimnsn (ddi\_dimensns), 30
- ddi\_recDimnsn(), 68
- ddi\_recGrp, 68
- ddi\_recGrp(), 40
- ddi\_recNumTot (ddi\_dimensns), 30
- ddi\_recPrCas (ddi\_dimensns), 30
- ddi\_referencePeriod (ddi\_sampleFrame), 73
- ddi\_relMat (ddi\_othrStdyMat), 57
- ddi\_relPubl (ddi\_othrStdyMat), 57
- ddi\_relStdy (ddi\_othrStdyMat), 57
- ddi\_resInstru (ddi\_dataColl), 19
- ddi\_resource, 69
- ddi\_resource(), 29
- ddi\_respRate (ddi\_anlyInfo), 5
- ddi\_respUnit, 70
- ddi\_respUnit(), 53, 108
- ddi\_restrctn (ddi\_useStmt), 103
- ddi\_row, 71
- ddi\_row(), 95, 97
- ddi\_rspStmt, 72
- ddi\_rspStmt(), 11, 36
- ddi\_sampleFrame, 73
- ddi\_sampleFrame(), 21

- ddi\_sampleFrameName (ddi\_sampleFrame),  
73
- ddi\_sampleSize (ddi\_targetSampleSize),  
93
- ddi\_sampleSizeFormula  
(ddi\_targetSampleSize), 93
- ddi\_sampProc (ddi\_dataColl), 19
- ddi\_security, 75
- ddi\_security(), 53, 108
- ddi\_selector (ddi\_usage), 101
- ddi\_serInfo (ddi\_serStmt), 76
- ddi\_serName (ddi\_serStmt), 76
- ddi\_serStmt, 76
- ddi\_serStmt(), 11, 36
- ddi\_setAvail, 77
- ddi\_setAvail(), 19
- ddi\_software (ddi\_prodStmt), 61
- ddi\_software(), 42
- ddi\_sourceCitation (ddi\_citation), 10
- ddi\_sourceCitation(), 79
- ddi\_sources, 79
- ddi\_sources(), 21, 79
- ddi\_southBL (ddi\_geoBndBox), 44
- ddi\_specificElements (ddi\_usage), 101
- ddi\_specPerm (ddi\_useStmt), 103
- ddi\_srcChar, 80
- ddi\_srcChar(), 69, 79
- ddi\_srcDocu, 80
- ddi\_srcDocu(), 69, 79
- ddi\_srcOrig, 81
- ddi\_srcOrig(), 69, 79
- ddi\_standard, 82
- ddi\_standard(), 83
- ddi\_standardName (ddi\_standard), 82
- ddi\_standardsCompliance, 83
- ddi\_standardsCompliance(), 66
- ddi\_stdCatgry (ddi\_var), 107
- ddi\_stdYClas (ddi\_method), 50
- ddi\_stdYDscr, 84
- ddi\_stdYDscr(), 12
- ddi\_stdYInfo, 85
- ddi\_stdYInfo(), 85
- ddi\_studyAuthorization, 87
- ddi\_studyAuthorization(), 85
- ddi\_studyBudget (ddi\_stdYInfo), 85
- ddi\_studyDevelopment, 88
- ddi\_studyDevelopment(), 85
- ddi\_subject, 89
- ddi\_subject(), 86
- ddi\_subTitl (ddi\_titlStmt), 98
- ddi\_sumDscr, 90
- ddi\_sumDscr(), 86
- ddi\_sumStat (ddi\_var), 107
- ddi\_table, 92
- ddi\_table(), 56
- ddi\_targetSampleSize, 93
- ddi\_targetSampleSize(), 21
- ddi\_tbody, 94
- ddi\_tbody(), 96
- ddi\_tgroup, 95
- ddi\_tgroup(), 93
- ddi\_thead, 96
- ddi\_thead(), 96
- ddi\_timeMeth (ddi\_dataColl), 19
- ddi\_timePrd (ddi\_sumDscr), 90
- ddi\_titl, 97
- ddi\_titl(), 93, 99
- ddi\_titlStmt, 98
- ddi\_titlStmt(), 11, 36
- ddi\_topcClas (ddi\_subject), 89
- ddi\_TotlResp (ddi\_var), 107
- ddi\_txt, 99
- ddi\_txt(), 9, 13, 44, 53, 54, 56, 74, 108, 110
- ddi\_undocCod (ddi\_var), 107
- ddi\_unitType (ddi\_frameUnit), 43
- ddi\_universe, 100
- ddi\_universe(), 53, 54, 74, 91, 108, 110
- ddi\_updateProcedure (ddi\_sampleFrame),  
73
- ddi\_usage, 101
- ddi\_usage(), 18
- ddi\_useStmt, 103
- ddi\_useStmt(), 19, 74
- ddi\_validPeriod (ddi\_sampleFrame), 73
- ddi\_valrng, 105
- ddi\_valrng(), 108
- ddi\_var, 107
- ddi\_var(), 23
- ddi\_varFormat (ddi\_var), 107
- ddi\_varGrp, 110
- ddi\_varGrp(), 23
- ddi\_varQnty (ddi\_dimensns), 30
- ddi\_verResp (ddi\_verStmt), 111
- ddi\_version (ddi\_verStmt), 111
- ddi\_verStmt, 111
- ddi\_verStmt(), 11, 36, 42, 53, 108

`ddi_weight` (`ddi_dataColl`), [19](#)  
`ddi_westBL` (`ddi_geoBndBox`), [44](#)  
`validate_codebook`, [112](#)