FP7-212214

Database Model – A List of Tables

2009

TABLE OF TABLES

STRINGS		
NM_TOKENS		
NM_TOKENSS		
NM_TOKEN_REFS		
URNS		*DDI3: r:URNType*
VERSIONS		*DDI3: r:VersionType*
SEARCH (NA)		
SEARCHES	*--*	
SEARCH_RESULTS	*--*	
PERSISTENCE (NA)		
PERSISTENT_IDS	*NA*	
EMBARGO (0)		
EMBARGOS	*0*	*CLASS ID needed for multilingual service*
EMBARGO_REFERENCES (new)	*1000*	
STUDY (0)		
STUDIES	*10*	
STUDY_REFERENCES	*1010*	
STUDY_CATEGORIES	*2010*	*SUPPORT*
DATASETS	*40*	
DATAFILES	*20*	
DATAFILETYPES	*2020*	*SUPPORT*
ANALYSIS_UNITS	*3020*	*SUPPORT*
KIND_OF_DATAS	*25*	*SUPPORT*
KOD_REFERENCES	*1025*	
SAMPLES	*30*	
SAMPLE_DESIGNS	*3030*	*SUPPORT*
COMPARISON (0): *comparative.xsd (c)*		
COMPARISONS	*50*	*DDI3: c:ComparisonType*
COMPARISON_REFS	*1050*	
GENERIC_MAPS	*60*	*DDI3: c:GenericMapType*
GENERIC_MAP_REFS	*1060*	
CORRESPONDENCES	*70*	*DDI3: c:CorrespondenceType*
USER_DEFINED_CORR_PROPERTYS		*DDI3: c:UserDefinedCorrespondencePropertyType*
USER_DEFINED_CORR_PROPERTY_REFS		
ITEM_MAPS		*DDI3: c:ItemMapType*
ITEM_MAP_REFS		
CODE_MAPS		*DDI3: c:CodeMapType*
CODE_MAP_REFS		
VOCABULARY (0)		
VOCABULARIES	*90*	
PROJECT ORGANIZING (100)		
PROJECTS	*100*	
PROJECT_TYPES	*2100*	*SUPPORT*
PRO_RELATIONS	*105*	
PRO_REL_TYPES	*2105*	*SUPPORT*
PROJECT_LAYERS	*110*	
PRO_LAYER_REFERENCES	*1110*	
PRO_LAYER_TYPES	*2110*	*SUPPORT*
PRO_LAYER_RELATIONS	*115*	
PRO_LAYER_REL_TYPES	*2115*	*SUPPORT*
LAYER_NODES	*120*	
LAYER_NODE_TYPES	*2120*	
LAYER_NODE_REFERENCES	*1120*	
LAYER_EDGES	*125*	
LAYER_EDGE_TYPES	*2125*	
LAYER_EDGE_REFERENCES	*1125*	
USER MANAGING (200)		
PARTICIPANTS	*200*	
PARTICIPANT_ROLES	*2200*	*SUPPORT*
USERS	*210*	
USER_ROLES	*2210*	*SUPPORT*
PERSONS	*220*	
AUTHORSHIPS	*230*	
INDIVIDUALS	*240*	
INSTITUTIONS	*250*	
AFFILIATIONS	*260*	
ADDRESS	*270*	
COUNTRIES	*280*	*SUPPORT*
UNIVERSE (300): conceptualcomponent.xsd (c)		
UNIVERSES	*300*	*DDI3: c:UniverseType*
UNIVERSE_REFS	*1300*	
UNI_CATEGORIES	*2300*	*SUPPORT*
UNI_SCHEMES	*301*	*DDI3: c:UniverseSchemeType*
UNI_SCH_REFERENCES	*1301*	

CONCEPT (300) : conceptualcomponent.xsd (c)		
CONCEPTS	*350*	*DDI3: c:ConceptType*
CONCEPT_REFS	*1350*	
CONCEPT_CATEGORIES	*2350*	*SUPPORT*
CONCEPT_SCHEMES	*351*	*DDI3: c:ConceptSchemeType*
CONCEPT_SCH_REFERENCES	*1351*	
CONCEPT_GROUPS	*355*	*DDI3: c:ConceptGroupType*
CONCEPT_GR_REFERENCES	*1355*	
TEXT STORING (400)		
KEYWORDS	*400*	
KEYWORDS_REFERENCES	*1400*	
COMMENTS	*410*	
COMMENT_TYPES	*2410*	*SUPPORT*
PROPERTIES	*420*	
PROPERTY_REFERENCES	*1420*	
TEXTUAL_OBJECTS	*430*	
TEXT_TYPES	*2430*	
LONG_TEXTS	*431*	
TEXTS	*432*	
SHORT_TEXTS	*433*	
REFERENCE (400) : reusable.xsd (r)		
CITATIONS	*440*	*DDI3: r:CitationType*
CITATION_REFS	*1440*	
DC_ELEMENTS		
DC_ELEMENT_TYPES		
DC_ELEMENT_TYPE_REFS		
URIS	*450*	
URI_REFERENCES	*1450*	
MEASUREMENT (500)		
MEASUREMENTS	*500*	
MEA_REFERENCES	*1500*	
MEA_TYPES	*2500*	*SUPPORT*
MEA_CSI_VALUES	*510*	
MEA_CSI_VAL_REFERENCES	*1510*	
MEA_OBS_STRUCTURES	*520*	
MEA_OBS_STRUCTURES_REFERENCES	*1520*	
MEA_VALIDITIES	*530*	
MEA_RELIABILITIES	*540*	
CONCEPTUAL LAYER (600)		
CONCEPTUALS	*600*	
CON_REFERENCES	*1600*	
CON_DIMENSIONS	*610*	
CON_DIM_REFERENCES	*1610*	
CON_DIM_TYPES	*2610*	*SUPPORT*
CON_SPECIFICATIONS	*620*	
CON_SPE_REFERENCES	*1620*	
ATTRIBUTES	*630*	
ATTRIBUTE_REFERENCES	*1630*	
OPERATIONAL LAYER (700)		
OPERATIONALIZATION	*700*	
OPERA_REFERENCES	*1700*	
OPERA_CONCEPTUAL	*710*	
OPERA_INDICATORS	*720*	
OPERA_IND_REFERENCES	*1720*	
OPERA_IND_TYPES	*2720*	*SUPPORT*
OPERA_MEA_VALUES	*730*	
OPERA_MEA_VAL_REFERENCES	*1730*	
DATA CODING LAYER (70)		
DATA_CODINGS	*740*	
DATA_CODING_REFERENCES	*1740*	
DC_VARIABLES	*750*	
DC_VAR_REFERENCES	*1750*	
DC_VAR_VALUES	*751*	
DC_VAR_VAL_REFERENCES	*1751*	
CONVERSIONS	*760*	
CONV_REFERENCES	*1760*	
CONV_VARIABLES	*761*	
CONV_VAR_TYPES	*2761*	*SUPPORT*
ROUTINES	*770*	
VARIABLE (800) : logicalproduct.xsd (r)		
VARIABLES	*800*	*DDI 3: l:VariableType*
VARIABLE_REFS	*1800*	
VARIABLE_SCHEMES	*801*	*DDI 3: l:VariableSchemeType*
VARIABLE_SCHEME_REFS	*1801*	
VARIABLE_GROUPS	*810*	*DDI3: l:VariableGroupType*
VARIABLE_GROUP_REFS	*1810*	
VALUE_RANGES	*820*	
VALUES	*821*	
VALUES_ REFERENCES	*1821*	

QUESTION (800) : datacollection.xsd (d)

QUESTIONNAIRES	*840*	
QUEST_N_REFERENCES	*1840*	
ADMINISTRATIONTYPES	*3840*	*SUPPORT*
DATA_COLLECTIONTYPES	*4840*	*SUPPORT*
QUESTION_ITEMS	*850*	*DDI3: d:QuestionItemType*
QUESTION_ITEM_REFS	*1850*	
MULTIPLE_QUESTION_ITEMS	*855*	*DDI3: d:MultipleQuestionType*
MULTIPLE_QUESTION_ITEM_REFS	*1855*	
QUESTION_SCHEMES	*851*	*DDI 3: d:QuestionSchemeType*
QUESTION_SCHEME_REFS	*1851*	
QUESTION_GROUPS	*860*	*DDI3: d:QuestionGroupType*
QUEST_GROUP_REFERENCES	*1860*	
SPECIFIC_SEQUENCES		*DDI3: d:SpecificSequenceType*
QUESTION_SEQUENCE_TYPES		
ITEMS	*870*	
ITEM_REFERENCES	*1870*	
ITEM_FLOWCONTROL	*875*	
SURVEY_SUPPORTING_ITEMS	*830*	
SUR_SUP_I_TYPES	*2830*	*SUPPORT*
SUR_SUP_I_REFERENCES	*1830*	
ANSWER_CATEGORIE_SETS	*880*	
ANSWER_CATEGORIES	*881*	
ANSWER_CATEGORIE_REFERENCES	*1881*	

DATA DEFINING (900)

R_REPRESENTATIONS		
REPRESENTATIONS		
CCATS_REPRESENTATIONS	*900*	
TEXT_REPRESENTATIONS	*910*	
DATETIME_REPRESENTATIONS	*920*	
NUMERIC_REPRESENTATIONS	*930*	
BOOLEAN_REPRESENTATIONS	*940*	
INCREMENTALS	*933*	
NUMBER_RANGES	*936*	
NUMERIC_TYPE_CODES	*3930*	*SUPPORT*
CATEGORY_RELATION_CODES	*3900*	*SUPPORT*
DATE_TYPE_CODES	*3920*	*SUPPORT*

NOTE:

In the rest of this document a certain construction will show up repeatedly: TABLE_ID, TABLE_ENTRY_ID.

TABLE_ID	The value stored in TABLE_ID identifies a table. The valid values and their associated tables - are stored in the table *CLASSES*.
TABLE_ENTRY_ID	The value stored in TABLE_ENTRY_ID equates the value of the PK of a table, which were referenced by the value of TABLE_ID. "0" is reserved for addressing a "global object" of the table.

The combination ob TABLE_ID and TABLE_ENTRY_ID creates a unique identifier in the database.

Variants are:

TABLE_ID_CONTAINER	q.v. TABLE_ID.
TABLE_ID_REF	q.v. TABLE_ID.
TABLE_ENTRY_ID_REF	q.v. TABLE_ENTRY_ID.
TABLE_ENTRY_ID	q.v. TABLE_ENTRY_ID.

The following combinations are allowed:

	"REFERENCES"	"STANDARD"	"RELATIONS"		
TABLE_ID_CONTAINER	FILLED	FILLED	FILLED		
TABLE_ID_REF	FILLED	EMPTY	EMPTY		
TABLE_ENTRY_ID_REF	FILLED	EMPTY	FILLED		
TABLE_ENTRY_ID	FILLED	FILLED	FILLED		

Tables marked as "REFERENCES" need all four attributes for unique access:
- The TABLE_ID of the containing REFERENCES table
- The TABLE_ID of a referenced table
- The TABLE_ENTRY_ID of said referenced table
- The TABLE_ENTRY_ID of an table, accessed by the REFERENCES table

Tables marked as "RELATIONS" need only three attributes for unique access. A special case of "REFERENCES".
- The TABLE_ID of the containing RELATIONS table
- The TABLE_ENTRY_ID of an table, accessed by the RELATIONS table
- The TABLE_ENTRY_ID of an table, accessed by the RELATIONS table

All other tables, here called "STANDARD", need two attributes for unique access.
- The TABLE_ID of the containing table
- The TABLE_ENTRY_ID of an object in said table

TABLES

TABLE *TABLES (TABLE_ID: NA), (SUPPORT)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
DEFAULT_LABEL	VARCHAR (256)

TABLE_ID: PK
DEFAULT_LABEL: Denomination of a table name

PK: TABLE_ID

TABLE *REF_RELATIONTYPES (TABLE_ID: NA), (SUPPORT)*

REF_RELATIONTYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

REF_RELATIONTYPE_ID: PK
DEFAULT_LABEL: Denomination of the relation type

PK: REF_RELATIONTYPE_ID

NOTE:

REF_RELATIONTYPE_ID	DEFAULT_LABEL
0	“ASSIGNED”
1	“CONTAINED”
2	“REFERENCED”

In DDI3 you can include an object or you can reference it. In case, the object is actual a part of a DDI3-Document, the access is stored with the type “CONTAINED”, otherwise the type is “REFERENCED”. With TABLE_ID, TABLE_ENTRY_ID the result is a link to an object with a fixed type. With “XXX_ID” the result is an object with possible different types.

“ASSIGNED” is used for assigning an object as part of the project context.

TABLE *ACTIVITIES_LOG (TABLE_ID: NA)*
This table documents the activities of the user on the objects in the conversion project

ACTIVITY_LOG_ID	BIGINT NOT NULL AUTO_INCREMENT
TABLE_ID_CONTAINER	SMALLINT UNSIGNED
TABLE_ID_REF	SMALLINT UNSIGNED
TABLE_ENTRY_ID_REF	INT
TABLE_ENTRY_ID	INT
TIMESTAMP	DATETIME
USER_ID	INT
ACTIVITY_TYPE_ID	INT
RATIONALE	VARCHAR (256)

ACTIVITY_LOG_ID: PK

TABLE_ID_CONTAINER:
TABLE_ID_REF,
TABLE_ENTRY_ID_REF:
TABLE_ENTRY_ID:

TIMESTAMP: Date and time of creation or modification
USER_ID: User who created or modified the object

ACTIVITY_TYPE_ID: The nature of the performed action; q.v. *ACTIVITY_TYPES*
RATIONALE: The intent of the performed action (default: NULL)

PK: ACTIVITY_LOG_ID

NOTE:
This Table has to be filled by a TRIGGER.

TABLE *ACTIVITY_TYPES (TABLE_ID: NA), (SUPPORT)*
This table stores the different types of activities

ACTIVITY_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

ACTIVITY_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the activity type

PK: ACTIVITY_TYPE_ID

TABLE *REFERENCES (TABLE_ID:-), (DDI3: reusable.xsd/ReferenceType)*
This table stores the relation between two objects

REFERENCE_ID	INT NOT NULL
TABLE_ID	SMALLINT UNSIGNED
TABLE_ENTRY_ID	INT
MODULE	INT
SCHEME	INT
URN	INT
ID	INT
IDENTIFYING_AGENCY	VARCHAR
VERSION	INT
IS_EXTERNAL	BOOLEAN DEFAULT (‘0’)
URI	VARCHAR
IS_REFERENCE	BOOLEAN DEFAULT (‘1’)
LATE_BOUND	BOOLEAN DEFAULT (‘0’)

REFERENCE_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
MODULE: SCHEME:	FK (***REFERENCES***) FK (***REFERENCES***)
URN:	FK (***URNS***)
ID: IDENTIFYING_AGENCY: VERSION:	FK (***IDS***) FK (***VERSIONS***)
IS_EXTERNAL: URI: IS_REFERENCE: LATE_BOUND:	
PK:	REFERENCE_ID

NOTE:
The fields TABLE_ID, TABLE_ENTRY_ID link to the referenced object as it is stored in the relational database.

TABLE *REFERENCE_REFS (TABLE_ID: -)*

REFERENCE_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
REF_RELATIONTYPE_ID	INT

LABEL_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE:	FK (***REFERENCES***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONTYPES***)
PK:	LABEL_REF_ID

TABLE *SCHEME_REFERENCES (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the

SCHEME_REFERENCE_ID	INT NOT NULL
REFERENCE_TYPE	INT
EXCLUDE	BOOLEAN DEFAULT (‘0’)

SCHEME_REFERENCE_ID: PK

REFERENCE_TYPE: FK (***REFERENCES***)
EXCLUDE: “FLAG” (***EXCLUDES** via **$_REFS***)

PK: SCHEME_REFERENCE_ID

NOTE:
This table implements the element *SchemeReference (DDI 3)*

MAPPING DDI3 IDS - CHARMCATS IDS

TABLE *ABSTRACT_IDENTIFIABLES (TABLE_ID: NA), (DDI 3 Compliance, reusable.xsd:AbstractIdentifiableType)*

ABSTRACT_IDENTIFIABLE_ID	INT NOT NULL AUTO_INCREMENT
NAME	BOOLEAN DEFAULT ('0')
ID	VARCHAR NOT NULL
URN	VARCHAR
ACTION	INT

ABSTRACT_IDENTIFIABLE_ID: PK

NAME: "FLAG" (***INTERNATIONAL_STRINGS** via **$_REFS***)
ID:
URN:
ACTION: FK (***ACTION_CODES***)

PK: ABSTRACT_IDENTIFIABLE_ID

TABLE *ABSTRACT_VERSIONABLES (TABLE_ID: NA), (DDI 3 Compliance, reusable.xsd:AbstractVersionableType)*

ABSTRACT_VERSIONABLE_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_IDENTIFIABLE_TYPE	INT
VERSION_RESPONSIBILITY	VARCHAR
VERSION_RATIONALE	BOOLEAN DEFAULT ('0')
VERSION	VARCHAR
VERSION_DATE	INT

ABSTRACT_VERSIONABLE_ID: PK

ABSTRACT_IDENTIFIABLE_TYPE: FK (***ABSTRACT_IDENTIFIABLES***)

VERSION_RESPONSIBILITY:
VERSION_RATIONALE: "FLAG" (***INTERNATIONAL_STRINGS** via **$_REFS***)
VERSION:
VERSION_DATE: FK (***BASE_DATES***)

PK: ABSTRACT_VERSIONABLE_ID

TABLE *ABSTRACT_MAINTAINABLES (TABLE_ID: NA), (DDI 3 Compliance, reusable.xsd:AbstractMaintainableType)*

ABSTRACT_MAINTAINABLE_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
AGENCY	VARCHAR

ABSTRACT_MAINTAINABLE_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VERSIONABLES***)

AGENCY:

PK: ABSTRACT_MAINTAINABLE_ID

TABLE *INTERNATIONAL_STRINGS (TABLE_ID: -)*
This table stores the different InternationalStringType-Objects

INTERNATIONAL_STRING_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR
XML_LANG	VARCHAR (64)
TRANSLATED	BOOLEAN DEFAULT (‘0’)
TRANSLATABLE	BOOLEAN DEFAULT (‘1’)

INTERNATIONAL_STRING_ID: PK

STRING_VALUE:

XML_LANG
TRANSLATED: Translated? (default: FALSE)
TRANSLATABLE: Translatable? (default: TRUE)

PK: INTERNATIONAL_STRING_ID

TABLE *INTERNATIONAL_STRING_REFS (TABLE_ID: -)*

INTERNATIONAL_STRING_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
INTERNATIONAL_STRING	INT
REF_RELATIONTYPE_ID	INT

INTERNATIONAL_STRING_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
INTERNATIONAL_STRING: FK (***INTERNATIONAL_STRINGS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: INTERNATIONAL_STRING_REF_ID

TABLE *LABELS (TABLE_ID: -)*
This table stores the different LabelType-Objects

LABEL_ID	INT NOT NULL AUTO_INCREMENT
INTERNATIONAL_STRING_TYPE	INT
LOCATION_VARIANT	VARCHAR
VALID_FOR_DATE	INT
TYPE	VARCHAR
MAX_LENGTH	INT

LABEL_ID: PK

INTERNATIONAL_STRING_TYPE: FK (***INTERNATIONAL_STRINGS***)

LOCATION_VARIANT:
VALID_FOR_DATE: FK (***BASE_DATES***)
TYPE:
MAX_LENGTH:

PK: LABEL_ID

TABLE *LABEL_REFS (TABLE_ID: -)*

LABEL_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
LABEL	INT
REF_RELATIONTYPE_ID	INT

LABEL_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
LABEL: FK (***LABELS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: LABEL_REF_ID

TABLE *TYPED_STRINGS (TABLE_ID: -)*
This table stores the different TypedStringType-Objects

TYPED_STRING_ID	INT NOT NULL AUTO_INCREMENT
INTERNATIONAL_STRING_TYPE	INT
TYPE	VARCHAR NOT NULL

TYPED_STRING_ID: PK

INTERNATIONAL_STRING_TYPE: FK (***INTERNATIONAL_STRINGS***)

TYPE:

PK: TYPED_STRING_ID

TABLE *TYPED_STRING_REFS (TABLE_ID: -)*
This table stores the relation between an object and a typed string

TYPED_STRING_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
TYPED_STRING	INT
REF_RELATIONTYPE_ID	INT

TYPED_STRING_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
TYPED_STRING: FK (***TYPED_STRINGS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: TYPED_STRING_REF_ID

TABLE *STRUCTURED_STRINGS (TABLE_ID: -)*
This table stores the different StructuredStringType-Objects

STRUCTURED_STRING_ID	INT NOT NULL AUTO_INCREMENT
BLK_NO_FORM	VARCHAR
XML_LANG	VARCHAR (64)
TRANSLATED	BOOLEAN DEFAULT (‘0’)
TRANSLATABLE	BOOLEAN DEFAULT (‘1’)

STRUCTURED_STRING _ID: PK

BLK_NO_FORM:

XML_LANG
TRANSLATED: Translated? (default: FALSE)
TRANSLATABLE: Translatable? (default: TRUE)

PK: STRUCTURED_STRING_ID

TABLE *STRUCTURED_STRING_REFS (TABLE_ID: -)*
This table stores the relation between an object and a structured string

STRUCTURED_STRING_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
STRUCTURED_STRING	INT
REF_RELATIONTYPE_ID	INT

STRUCTURED_STRING_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
STRUCTURED_STRING: FK (***STRURCTURED_STRINGS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: STRUCTURED_STRING_REF_ID

TABLE *IDENTIFIED_STRUCTURED_STRINGS (TABLE_ID: -)*
This table stores the different IdentifiedStructuredStringType-Objects

IDENTIFIED_STRUCTURED_STRING_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_IDENTIFIABLE_TYPE	INT
IS_IDENTIFIABLE	BOOLEAN DEFAULT (‘1’)
CONTENT	INT

IDENTIFIED_STRUCTURED_STRING _ID: PK

ABSTRACT_IDENTIFIABLE_TYPE: FK (***ABSTRACT_IDENTIFIABLES***)
IS_IDENTIFIABLE:

CONTENT: FK (***STRUCTURED_STRINGS***)

PK: IDENTIFIED_STRUCTURED_STRING_ID

TABLE *IDENTIFIED_STRUCTURED_STRING_REFS (TABLE_ID: -)*
This table stores the relation between an object and an identified structured string

IDENTIFIED_STRUCTURED_STRING_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
IDENTIFIED_STRUCTURED_STRING	INT
REF_RELATIONTYPE_ID	INT

IDENTIFIED_STRUCTURED_STRING_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
STRUCTURED_STRING: FK (***IDENTIFIED_STRURCTURED_STRINGS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: IDENTIFIED_STRUCTURED_STRING_REF_ID

TABLE *DYNAMIC_TEXTS (TABLE_ID: -)*
This table stores the different DynamicTextType-Objects

DYNAMIC_TEXT_ID	INT NOT NULL AUTO_INCREMENT
TEXT	BOOLEAN DEFAULT (‘0’)
XML_LANG	VARCHAR (64)
TRANSLATED	BOOLEAN DEFAULT (‘0’)
TRANSLATABLE	BOOLEAN DEFAULT (‘1’)

DYNAMIC_TEXT_ID: PK

TEXT: “FLAG” (***TEXTS** via **$_REFS***)

XML_LANG
TRANSLATED: Translated? (default: FALSE)
TRANSLATABLE: Translatable? (default: TRUE)

PK: DYNAMIC_TEXT_ID

TABLE *DYNAMIC_TEXT_REFS (TABLE_ID: -)*
This table stores the relation between an object and a dynamic text

DYNAMIC_TEXT_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
DYNAMIC_TEXT	INT
REF_RELATIONTYPE_ID	INT

DYNAMIC_TEXT_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
STRUCTURED_STRING: FK (***DYNAMIC_TEXTS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: DYNAMIC_TEXT_REF_ID

TABLE *TEXTS (TABLE_ID: -)*
This table stores the different TextType-Objects

TEXT_ID	INT NOT NULL AUTO_INCREMENT
LITERAL_TEXT	INT
CONDITIONAL_TEXT	INT
DESCRIPTION	BOOLEAN DEFAULT('0')

TEXT_ID: PK

LITERAL_TEXT: FK (***LITERAL_TEXTS***)
CONDITIONAL_TEXT: FK (***CONDITIONAL_TEXTS***)

DESCRIPTION: "FLAG" (***STRUCTURED_STRINGS*** *via* ***$_REFS***)

PK: TEXT_ID

TABLE *TEXT_REFS (TABLE_ID: -)*
This table stores the relation between an object and a text

TEXT_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
TEXT	INT
REF_RELATIONTYPE_ID	INT

TEXT_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
STRUCTURED_STRING: FK (***TEXTS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: TEXT_REF_ID

TABLE *LITERAL_TEXTS (TABLE_ID: -)*
This table stores the different LiteralTextType-Objects

LITERAL_TEXT_ID	INT NOT NULL AUTO_INCREMENT
TEXT	VARCHAR

LITERAL_TEXT_ID: PK

TEXT:

PK: LITERAL_TEXT_ID

TABLE *CONDITIONAL_TEXTS (TABLE_ID: -)*
This table stores the different ConditionalTextType-Objects

CONDITIONAL_TEXT_ID	INT NOT NULL AUTO_INCREMENT
EXPRESSION	INT

CONDITIONAL_TEXT_ID: PK

EXPRESSION: FK (***CODES***)

PK: CONDITIONAL_TEXT_ID

TABLE *CODES (TABLE_ID: -)*
This table stores the different CodeType-Objects

CODE_ID	INT NOT NULL AUTO_INCREMENT
CODE	VARCHAR
SOURCE_QUESTION_REFERENCE	BOOLEAN DEFAULT (‘1’)
DESCRIPTION	BOOLEAN DEFAULT (‘0’)
PROGRAMMING_LANGUAGE	VARCHAR

CODE_ID: PK

CODE:
SOURCE_QUESTION_REFERENCE: “FLAG” (***REFERENCES** via **$_REFS***)
DESCRIPTION: “FLAG” (***STRUCTURED_STRINGS** via **$_REFS***)

PROGRAMMING_LANGUAGE:

PK: CODE_ID

NOTE:
SOURCE_QUESTION_REFERENCE: Reference to a Question!

TABLE *CODE_REFS (TABLE_ID: -)*
This table stores the relation between an object and a code

CODE_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CODE	INT
REF_RELATIONTYPE_ID	INT

CODE_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
CODE: FK (***CODES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: CODE_REF_ID

TABLE *STRUCTURED_MIXED_RESPONSE_DOMAINS (TABLE_ID: -)*
This table stores the different StructuredMixedResponseDomainType-Objects

STRUCTURED_MIXED_RESPONSE_DOMAIN_ID	INT NOT NULL AUTO_INCREMENT
RESPONSE_TEXT	BOOLEAN DEFAULT ('0')
RESPONSE_DOMAIN	BOOLEAN DEFAULT ('0')

STRUCTURED_MIXED_RESPONSE_DOMAIN_ID: PK

RESPONSE_TEXT: "FLAG" (***DYNAMIC_TEXTS** via **$_REF***)
RESPONSE_DOMAIN: "FLAG" (***R_REPRESENTATIONS** via **$_REF***)

PK: STRUCTURED_MIXED_RESPONSE_DOMAIN_ID

TABLE *STRUCTURED_MIXED_RESPONSE_DOMAIN_REFS (TABLE_ID: -)*
This table stores the relation between an object and a structured mixed response domain

STRUCTURED_MIXED_RESPONSE_DOMAIN_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
STRUCTURED_MIXED_RESPONSE_DOMAIN	INT
REF_RELATIONTYPE_ID	INT

STRUCTURED_MIXED_RESPONSE_DOMAIN_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
STRUCTURED_MIXED_RESPONSE_DOMAINS: FK (***STRUCTURED_STRING***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK:
STRUCTURED_MIXED_RESPONSE_DOMAIN_REF_ID

TABLE *OTHER_MATERIALS (TABLE_ID: -)*
This table stores the different OtherMaterialType-Objects

OTHER_MATERIAL_ID	INT NOT NULL AUTO_INCREMENT
CITATION	INT
EXTERNAL_URL_REFERENCE	VARCHAR
EXTRENAL_URN_REFERENCE	VARCHAR
RELATIONSHIP	BOOLEAN DEFAULT (‘0’)
MIME_TYPE	VARCHAR
TYPE	VARCHAR NOT NULL

OTHER_MATERIAL_ID: PK

CITATION: FK (***CITATIONS***)
EXTERNAL_URL_REFERENCE:
EXTRENAL_URN_REFERENCE:
RELATIONSHIP: “FLAG” (***RELATIONSHIPS*** *via* ***$_REFS***)
MIME_TYPE:

TYPE:

PK: OTHER_MATERIAL_ID

TABLE *OTHER_MATERIAL_REFS (TABLE_ID: -)*
This table stores the relation between an object and an other material

OTHER_MATERIAL_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
OTHER_MATERIAL	INT
REF_RELATIONTYPE_ID	INT

OTHER_MATERIAL_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
OTHER_MATERIAL: FK (***OTHER_MATERIALS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: OTHER_MATERIAL_REF_ID

TABLE *RELATIONSHIPS (TABLE_ID: -)*
This table stores the different RelationshipType-Objects

RELATIONSHIP_ID	INT NOT NULL AUTO_INCREMENT
RELATED_TO_REFERENCE	INT
RELATIONSHIP_DESCRIPTION	BOOLEAN DEFAULT (‘0’)

RELATIONSHIP_ID: PK

RELATED_TO_REFERENCE: FK (***REFERENCES***)
RELATIONSHIP_DESCRIPTION: “FLAG” (***INTERNATIONAL_STRINGS*** *via* ***$_REFS***)

PK: RELATIONSHIP_ID

TABLE *RELATIONSHIP_REFS (TABLE_ID: -)*
This table stores the relation between an object and a relationship

RELATIONSHIP_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
RELATIONSHIP	INT
REF_RELATIONTYPE_ID	INT

RELATIONSHIP_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
RELATIONSHIP: FK (***RELATIONSHIPS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: RELATIONSHIP_REF_ID

TABLE *DATES (TABLE_ID: NA) , (DDI 3 Compliance, reusable.xsd:DateType)*
This table stores the different DateType-Objects; also the date and/or time of any other non-DDI3 object in CHARMCATS (text, comment etc)

DATE _ID	INT NOT NULL AUTO_INCREMENT
SIMPLE_DATE	INT
START_DATE	INT
HIST_START_DATE	INT
END_DATE	INT
HIST_END_DATE	INT
CYCLE	INT
HIST_DATE	INT
CALENDAR	VARCHAR

DATE _ID:

SIMPLE_DATE: FK (***BASE_DATES***)
START_DATE: FK (***BASE_DATES***)
HIST_START_DATE: FK (***HISTORICAL_DATES***)
END_DATE: FK (***BASE_DATES***)
HIST_END_DATE: FK (***HISTORICAL_DATES***)
CYCLE:
HIST_DATE: FK (***HISTORICAL_DATES***)
CALENDAR:

PK: DATE _ID

TABLE *BASE_DATES (TABLE_ID: -)*
This table stores the different BaseDateType-Objects

BASE_DATE_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR

BASE_DATE_ID: PK

STRING_VALUE:

PK: BASE_DATE_ID

TABLE *HISTORICAL_DATES (TABLE_ID: -)*
This table stores the different HistoricalDateType-Objects

HISTORICAL_DATE_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR
HISTORICAL_DATE_FORMAT	VARCHAR

HISTORICAL_DATE_ID: PK

STRING_VALUE:

HISTORICAL_DATE_FORMAT:

PK: HISTORICAL_DATE_ID

TABLE *INTERVIEWER_INSTRUCTION_SCHEMES (TABLE_ID: -)*
This table stores the different InterviewerInstructionSchemeType-Objects

INTERVIEWER_INSTRUCTION_SCHEME_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_MAINTAINABLE_TYPE	INT
IS_MAINTAINABLE	BOOLEAN DEFAULT('1')
LABEL	BOOLEAN DEFAULT ('0')
DESCRIPTION	BOOLEAN DEFAULT ('0')
INTERVIEWER_INSTRUCTION_SCHEME_REF	BOOLEAN DEFAULT ('1')
INSTRUCTION	BOOLEAN DEFAULT ('0')

INTERVIEWER_INSTRUCTION_SCHEME_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_MAINTAINABLE_TYPES***)
IS_VERSIONABLE: Is it maintainable? (default: TRUE)

LABEL: "FLAG" (***LABELS via $_REFS***)
DESCRIPTION: "FLAG" (***DESCRIPTIONS via $_REFS***)
INTERVIEWER_INSTRUCTION_SCHEME_REF: "FLAG" (***SCHEME_REFERENCES via $ _REFS***)
INSTRUCTION: "FLAG" (***INSTRUCTIONS via $_REFS***)

PK: INTERVIEWER_INSTRUCTION_SCHEME_ID

TABLE *INTERVIEWER_INSTRUCTION_SCHEME_REFS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the relation between an object and an interviewer instruction scheme

INTERVIEWER_INSTRUCTION_SCHEME_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
INTERVIEWER_INSTRUCTION_SCHEME	INT
REF_RELATIONTYPE_ID	INT

INTERVIEWER_INSTRUCTION_SCHEME_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
INTERVIEWER_INSTRUCTION_SCHEME: FK (***INTERVIEWER_INSTRUCTION_SCHEMES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: INTERVIEWER_INSTRUCTION_SCHEME_REF_ID

TABLE *INSTRUCTIONS (TABLE_ID: -)*
This table stores the different InstructionType-Objects

INSTRUCTION_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT('1')
LABEL	BOOLEAN DEFAULT ('0')
ASSOCIATED_IMAGE	BOOLEAN DEFAULT ('0')
INSTRUCTION_TEXT	BOOLEAN DEFAULT ('0')

INSTRUCTION_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VERSIONABLES***)
IS_VERSIONABLE: Is it versionable? (default: TRUE)

LABEL: "FLAG" (***LABELS via $ _REFS***)
ASSOCIATED_IMAGE: "FLAG" (***STRINGS via $_REFS***)
INSTRUCTION_TEXT: "FLAG" (***STRUCTURED_STRINGS via $_REFS***)

PK: INSTRUCTION_ID

TABLE *CONTROL_CONSTRUCT_SCHEMES (TABLE_ID: -)*
This table stores the different ControlConstructSchemeType-Objects

CONTROL_CONSTRUCT_SCHEME_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_MAINTAINABLE_TYPE	INT
IS_MAINTAINABLE	BOOLEAN DEFAULT('1')
LABEL	BOOLEAN DEFAULT ('0')
DESCRIPTION	BOOLEAN DEFAULT ('0')
CONTROL_CONSTRUCT_SCHEME_REF	BOOLEAN DEFAULT ('1')
CONTROL_CONSTRUCT	BOOLEAN DEFAULT ('0')

CONTROL_CONSTRUCT_SCHEME_ID: PK

ABSTRACT_MAINTAINABLE_TYPE:
IS_MAINTAINABLE:

LABEL: "FLAG" (***LABELS via $_REFS***)
DESCRIPTION: "FLAG" (***DESCRIPTIONS via $_REFS***)
CONTROL_CONSTRUCT_SCHEME_REF: "FLAG" (***SCHEME_REFERENCES via $_REFS***)
CONTROL_CONSTRUCT: "FLAG" (***CONTROL_CONSTRUCTS via $_REFS***)

PK: CONTROL_CONSTRUCT_SCHEME_ID

TABLE *CONTROL_CONSTRUCT_SCHEME_REFS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the relation between an object and an control-construct scheme

CONTROL_CONSTRUCT_SCHEME_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CONCTROL_CONSTRUCT_SCHEME	INT
REF_RELATIONTYPE_ID	INT

CONTROL_CONSTRUCT_SCHEME_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
CONTROL_CONSTRUCT_SCHEME: FK (***CONTROL_CONSTRUCT_SCHEMES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: CONTROL_CONSTRUCT_SCHEME_REF_ID

TABLE *CONTROL_CONSTRUCTS (TABLE_ID: -)*
This table stores the different ControlConstructType-Objects

CONTROL_CONSTRUCT_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT('1')
EXTERNAL_AID	BOOLEAN DEFAULT ('0')
EXT_INTERVIEWER_INSTRUCT_REF	BOOLEAN DEFAULT ('0')
INTERVIEWER_INSTRUCT_REF	BOOLEAN DEFAULT ('0')

CONTROL_CONSTRUCT_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VARSIONABLES***)
IS_VERSIONABLE: Is is versionable? (default: TRUE)

EXTERNAL_AID: "FLAG" (***OTHER_MATERIALS via $_REF***)
EXT_INTERVIEWER_INSTRUCT_REF: "FLAG" (***EXT_INTERVIEWER_INSTRUCTS via $_REFS***)
INTERVIEWER_INSTRUCT_REF: "FLAG" (***INTERVIEWER_INSTRUCTS via $_REFS***)

PK: CONTROL_CONSTRUCT_ID

TABLE *CONTROL_CONSTRUCT_REFS (TABLE_ID: 1350), (DDI 3 Compliance)*
This table stores the relation between an object and an control-construct

CONTROL_CONSTRUCT_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CONCTROL_CONSTRUCT	INT
REF_RELATIONTYPE_ID	INT

CONTROL_CONSTRUCT_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: CONTROL_CONSTRUCT_REF_ID

TABLE *EXT_INTERVIEWER_INSTRUCTS (TABLE_ID: -)*
This table stores the different ExternalInterviewerInstrutionType-Objects

EXT_INTERVIEWER_INSTRUCT_ID	INT NOT NULL AUTO_INCREMENT
OTHER_MATERIAL	INT
DISPLAY_TEXT	BOOLEAN DEFAULT('1')

EXT_INTERVIEWER_INSTRUCT_ID: PK

OTHER_MATERIAL: FK (***OTHER_MATERIALS***)
DISPLAY_TEXT: Is is a Display Text? (default: TRUE)

PK: EXT_INTERVIEWER_INSTRUCT_ID

TABLE *EXT_INTERVIEWER_INSTRUCT_REFS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the relation between an object and an external interviewer instruction

EXT_INTERVIEWER_INSTRUCT_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
EXT_INTERVIEWER_INSTRUCT	INT
REF_RELATIONTYPE_ID	INT

EXT_INTERVIEWER_INSTRUCT_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
EXT_INTERVIEWER_INSTRUCT_ID: FK (*EXT****_INTERVIEWER_INSTRUCTS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: EXT_INTERVIEWER_INSTRUCT_REF_ID

TABLE *INTERVIEWER_INSTRUCTS (TABLE_ID: -)*
This table stores the different InterviewerInstrutionType-Objects

INTERVIEWER_INSTRUCT_ID	INT NOT NULL AUTO_INCREMENT
REFERENCE	INT
DISPLAY_TEXT	BOOLEAN DEFAULT('1')

INTERVIEWER_INSTRUCT_ID: PK

REFERENCE: FK (***REFERENCES***)
DISPLAY_TEXT: Is is a text to be displyed? (default: TRUE)

PK: INTERVIEWER_INSTRUCT_ID

TABLE *INTERVIEWER_INSTRUCT_REFS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the relation between an object and an interviewer instruction

INTERVIEWER_INSTRUCT_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
INTERVIEWER_INSTRUCT	INT
REF_RELATIONTYPE_ID	INT

INTERVIEWER_INSTRUCT_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
INTERVIEWER_INSTRUCT_ID: FK (***INTERVIEWER_INSTRUCTS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: INTERVIEWER_INSTRUCT_REF_ID

TABLE *LOOPS (TABLE_ID: -)*
This table stores the different LoopType-Objects

LOOP_ID	INT NOT NULL AUTO_INCREMENT
CONTROL_CONSTRUCTS	INT
LOOP_VAR_REF	INT
INITIAL_VALUE	INT
LOOP_WHILE	INT
STEP_VALUE	INT

LOOP_ID: PK

CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS)***

LOOP_VAR_REF: FK (***REFERENCES***)
INITIAL_VALUE: FK (***CODES***)
LOOP_WHILE: FK (***CODES***)
STEP_VALUE: FK (***CODES***)

PK: LOOP_ID

TABLE *IF_THEN_ELSES (TABLE_ID: -)*
This table stores the different IfThenElseType-Objects

IF_THEN_ELSE_ID	INT NOT NULL AUTO_INCREMENT
CONTROL_CONSTRUCTS	INT
IF_CONDITION	INT
THEN_CONSTRUCT_REF	INT
ELSE_CONSTRUCT_REF	INT

IF_THEN_ELSE_ID: PK

CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS)***

IF_CONDITION: FK (***CODES***)
THEN_CONSTRUCT_REF: FK (***REFERENCES***)
ELSE_CONSTRUCT_REF: FK (***REFERENCES***)

PK: IF_THEN_ELSE_ID

TABLE *REPEAT_UNTILS (TABLE_ID: -)*
This table stores the different RepeatUntilType-Objects

REPEAT_UNTIL_ID	INT NOT NULL AUTO_INCREMENT
CONTROL_CONSTRUCTS	INT
UNTIL_CONDITION	INT
UNTIL_CONSTRUCT_REF	INT

REPEAT_UNITL_ID: PK

CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS)***

UNTIL_CONDITION: FK (***CODES***)
UNTIL_CONSTRUCT_REF: FK (***REFERENCES***)

PK: REPEAT_UNTIL_ID

TABLE *REPEAT_WHILES (TABLE_ID: -)*
This table stores the different RepeatWhileType-Objects

REPEAT_WHILE_ID	INT NOT NULL AUTO_INCREMENT
CONTROL_CONSTRUCTS	INT
WHILE_CONDITION	INT
WHILE_CONSTRUCT_REF	INT

REPEAT_WHILE_ID: PK

CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS)***

WHILE_CONDITION: FK (***CODES***)
WHILE_CONSTRUCT_REF: FK (***REFERENCES***)

PK: REPEAT_WHILE_ID

TABLE *SEQUENCES (TABLE_ID: -)*
This table stores the different SequenceType-Objects

SEQUENCE_ID	INT NOT NULL AUTO_INCREMENT
CONTROL_CONSTRUCTS	INT
TYPE	INT
CONTROL_CONSTRUCT_REF	BOOLEAN DEFAULT (‘1’)
CONSTRUCT_SEQUENCE	INT

SEQUENCE_ID: PK

CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS)***

TYPE: FK (***CODE_VALUES***)
CONTROL_CONSTRUCT_REF: “FLAG” (***REFERENCES via $_REFS***)
CONSTRUCT_SEQUENCE: FK (***SPECIFIC_SEQUENCES***)

PK: SEQUENCE_ID

TABLE *COMPUTATION_ITEMS (TABLE_ID: -)*
This table stores the different ComputationItemType-Objects

COMPUTATION_ITEM_ID	INT NOT NULL AUTO_INCREMENT
CONTROL_CONSTRUCTS	INT
CODE	INT
ASSIGNED_VAR_REF	INT

COMPUTATION_ITEM_ID: PK

CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS)***

CODE: FK (***CODES***)
ASSIGNED_VAR_REF: FK (***REFERENCES***)

PK: COMPUTATION_ITEM_ID

TABLE *STATEMENT_ITEMS (TABLE_ID: -)*
This table stores the different StatementItemType-Objects

STATEMENT_ITEM_ID	INT NOT NULL AUTO_INCREMENT
CONTROL_CONSTRUCTS	INT
DISPLAY_TEXT	INT
DESCRIPTION	BOOLEAN DEFAULT (‘0’)

STATEMENT_ITEM_ID: PK

CONTROL_CONSTRUCT: FK (***CONTROL_CONSTRUCTS)***

DISPLAY_TEXT: FK (***DYNAMIC_TEXTS***)
DESCRIPTION: “FLAG” (***DESCRIPTIONS via $_REFS***)

PK: STATEMENT_ITEM_ID

TABLE *CREATORS (TABLE_ID: -)*
This table stores the different CreatorType-Objects

CREATOR_ID	INT NOT NULL AUTO_INCREMENT
INTERNATIONAL_STRING_TYPE	INT
AFFILIATION	VARCHAR

CREATOR_ID: PK

INTERNATIONAL_STRING_TYPE: FK (***INTERNATIONAL_STRINGS***)

AFFILIATION:

PK: CREATOR_ID

TABLE *CREATOR_REFS (TABLE_ID: -)*
This table stores the relation between an object and a creator

CREATOR_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CREATOR	INT
REF_RELATIONTYPE_ID	INT

CREATOR_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
CONCEPT: FK (***CREATORS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: CREATOR_REF_ID

TABLE *CONTRIBUTORS (TABLE_ID: -)*
This table stores the different ContributorType-Objects

CONTRIBUTOR_ID	INT NOT NULL AUTO_INCREMENT
INTERNATIONAL_STRING_TYPE	INT
ROLE	VARCHAR
AFFILIATION	VARCHAR

CONTRIBUTOR_ID: PK

INTERNATIONAL_STRING_TYPE: FK (***INTERNATIONAL_STRINGS***)

ROLE:
AFFILIATION:

PK: CONTRIBUTOR_ID

TABLE *CONTRIBUTOR_REFS (TABLE_ID: -)*
This table stores the relation between an object and a contributor

CONTRIBUTOR_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CONTRIBUTOR	INT
REF_RELATIONTYPE_ID	INT

CONTRIBUTOR_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
CONCEPT: FK (***CONTRIBUTORS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: CONTRIBUTOR_REF_ID

TABLE *GENERAL_INSTRUCTION_REFS (TABLE_ID: -)*
This table stores the relation between an object and a general instruction

GENERAL_INSTRUCTION_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
GENERAL_INSTRUCTION	INT
REF_RELATIONTYPE_ID	INT

GENERAL_INSTRUCTION_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE: GENERAL_INSTRUCTION:	FK (***REFERENCES***) FK (***GENERAL_INSTRUCTIONS***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONTYPES***)
PK:	GENERAL_INSTRUCTION_REF_ID

TABLE *IMPUTATION_REFS (TABLE_ID: -)*
This table stores the relation between an object and a general instruction

IMPUTATION_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
IMPUTATION	INT
REF_RELATIONTYPE_ID	INT

IMPUTATION_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE: IMPUTATION:	FK (***REFERENCES***) FK (***GENERAL_INSTRUCTIONS***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONTYPES***)
PK:	IMPUTATION_REF_ID

TABLE *STANDARD_WEIGHT_REFS (TABLE_ID: -) (new)*
This table stores the relation between an object and a standard weight

STANDARD_WEIGHT_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
STANDARD_WEIGHT	INT
REF_RELATIONTYPE_ID	INT

STANDARD_WEIGHT_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE: STANDARD_WEIGHT:	FK (***REFERENCES***) FK (***IDENTIFIED_STRUCTURED_STRING***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONTYPES***)
PK:	STANDARD_WEIGHT_REF_ID

TABLE *WEIGHT_VARIABLE_REFS (TABLE_ID: -)*
This table stores the relation between an object and a weight variable

WEIGHT_VARIABLE_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
WEIGHT_VARIABLE	INT
REF_RELATIONTYPE_ID	INT

WEIGHT_VARIABLE_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE: WEIGHT_VARIABLE:	FK (***REFERENCES***) FK (***VARIABLES***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONTYPES***)
PK:	WEIGHT_VARIABLE_REF_ID

TABLE *SOURCE_REFS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the Reference to a question or a variable

SOURCE_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
MNENOMIC	VARCHAR

SOURCE_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE: MNEMONIC:	FK (***REFERENCES***)
PK:	SOURCE_REF_ID

TABLE *SOURCE_QUESTION_REFS (TABLE_ID: -)*
This table stores the relation between an object and a source question

SOURCE_QUESTION_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
SOURCE_REFERENCE_TYPE	INT
QUESTION	INT
REF_RELATIONTYPE_ID	INT

SOURCE_QUESTION_ID: PK

TABLE_ID, TABLE_ENTRY_ID: FK (***diff. tables***):

ELEMENT_NAME: <TAG>-Denomination in DDI3

SOURCE_REFERENCE_TYPE: FK (***SOURCE_REFS***)
QUESTION: FK (***QUESTION_ITEMS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: SOURCE_QUESTION_ID

TABLE *SOURCE_VARIABLE_REFS (TABLE_ID: -)*
This table stores the relation between an object and a source variable

SOURCE_VARIABLE_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
SOURCE_REFERENCE_TYPE	INT
VARIABLE	INT
REF_RELATIONTYPE_ID	INT

SOURCE_VARIABLE_ID: PK

TABLE_ID, TABLE_ENTRY_ID: FK (***diff. tables***):

ELEMENT_NAME: <TAG>-Denomination in DDI3

SOURCE_REFERENCE_TYPE: FK (***SOURCE_REFS***)
VARIABLE: FK (***VARIABLES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: SOURCE_VARIABLE_ID

TABLE *CODINGS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different CodingType-Objects

CODINGS_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_IDENTIFIABLE_TYPE	INT
IS_IDENTIFIABLE	BOOLEAN DEFAULT (‘1’)
GENERAL_INSTRUCTION	INT
GENERATION_INSTRUCTION	INT

CODINGS_ID: PK

ABSTRACT_IDENTIFIABLE_TYPE: FK (***ABSTRACT_IDENTIFIABLES***)
IS_IDENTIFIABLE: Is is identifiable? (default: TRUE)

GENERAL_INSTRUCTION: FK (***GENERAL_INSTRUCTIONS***)
GENERATION_INSTRUCTION: FK (***GENERATION_INSTRUCTIONS***)

PK: CODINGS_ID

TABLE *CODING_INSTRUCTION_REFS (TABLE_ID: -)*
This table stores the reference to a coding

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CODING_INSTRUCTION	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***):

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
CODING_INSTRUCTION: FK (***CODINGS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: TABLE_ID, TABLE_ENTRY_ID, CODING_INSTRUCTION

NOTE:
This table implements the element *CodingInstructionReference* from *l:RepresentationType*

TABLE *GENERAL_INSTRUCTIONS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different GeneralInstructionType-Objects

GENERAL_INSTRUCTION_ID	INT NOT NULL AUTO_INCREMENT
DESCRIPTION	BOOLEAN DEFAULT (‘0’)
COMMAND	BOOLEAN DEFAULT (‘0’)
OVERRIDE	INT
IS_OVERRIDE	BOOLEAN

GENERAL_INSTRUCTION_ID: PK

DESCRIPTION: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
COMMAND: “FLAG” (***COMMANDS*** *via* ***$_REFS***)
OVERRIDE: FK (***GENERAL_INSTRUCTION_REFS***)

IS_OVERRIDE: Is is an override?

PK: GENERAL_INSTRUCTION_ID

TABLE *GENERATION_INSTRUCTIONS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different GenerationInstructionType-Objects

GENERATION_INSTRUCTION_ID	INT NOT NULL AUTO_INCREMENT
SOURCE_QUESTION	BOOLEAN DEFAULT (‘0’)
SOURCE_VARIABLE	BOOLEAN DEFAULT (‘0’)
EXTERNAL_INFORMATION	BOOLEAN DEFAULT (‘0’)
DESCRIPTION	BOOLEAN DEFAULT (‘0’)
COMMAND	BOOLEAN DEFAULT (‘0’)
CONTROL_CONSTRUCT_REFERENCE	BOOLEAN DEFAULT (‘1’)
AGGREGATION	INT
IS_DERIVED	BOOLEAN

GENERATION_INSTRUCTION_ID: PK

SOURCE_QUESTION: “FLAG” (***QUESTION_ITEMS*** *via* ***$_REFS***)
SOURCE_VARIABLE: “FLAG” (***SOURCE_VARIABLES*** *via* ***$_REFS***)
EXTERNAL_INFORMATION: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
DESCRIPTION: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
COMMAND: “FLAG” (***COMMANDS*** *via* ***$_REFS***)
CONTROL_CONSTRUCT_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
AGGREGATION: FK (***AGGREGATIONS***)

IS_DERIVED:

PK: GENERATION_INSTRUCTION_ID

TABLE *COMMANDS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different CommandType-Objects

COMMAND_ID	INT NOT NULL AUTO_INCREMENT
COMMAND_TEXT	VARCHAR
COMMAND_FILE	BOOLEAN DEFAULT (‘0’)
STRUCTURED_COMMAND	INT
FORMAL_LANGUAGE	VARCHAR

COMMAND_ID: PK

COMMAND_TEXT:
COMMAND_FILE: “FLAG” (***COMMAND_FILES*** *via* ***$_REFS***)
STRUCTURED_COMMAND: FK (***STRUCTURED_COMMANDS***)

FORMAL_LANGUAGE:

PK: COMMAND_ID

TABLE *STRUCTURED_COMMANDS (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different StructuredCommandType-Objects

STRUCTURED_COMMAND_ID	INT NOT NULL AUTO_INCREMENT
CONTENT	VARCHAR

STRUCTURED_COMMAND_ID: PK

CONTENT:

PK: STRUCTURED_COMMAND_ID

TABLE *COMMAND_FILES (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different CommandFileType-Objects

COMMAND_FILE_ID	INT NOT NULL AUTO_INCREMENT
LOCATION	BOOLEAN DEFAULT (‘0’)
PATH	BOOLEAN DEFAULT (‘0’)
URI	BOOLEAN DEFAULT (‘0’)
FORMAL_LANGUAGE	VARCHAR

COMMAND_FILE_ID: PK

LOCATION: “FLAG” (***STRINGS*** *via* ***$_REFS***)
PATH: “FLAG” (***STRINGS*** *via* ***$_REFS***)
URI: “FLAG” (***URNS*** *via* ***$_REFS***)

PK: COMMAND_FILE_ID

TABLE *CODE_VALUES (TABLE_ID: -)*
This table stores the different CodeValueType-Objects

CODE_VALUE_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR
CODE_LIST_ID	VARCHAR
CODE_LIST_AGENCY	VARCHAR
CODE_LIST_VERSION	VARCHAR DEFAULT (‘1.0’)
OTHER_VALUE	VARCHAR

CODE_VALUE_ID: PK

STRING_VALUE:

CODE_LIST_ID:
CODE_LIST_AGENCY:
CODE_LIST_VERSION:
OTHER_VALUE:

PK: CODE_VALUE_ID

TABLE *ACTION_CODES (TABLE_ID: NA), (DDI 3 Compliance, reusable.xsd:ActionCodeType) (SUPPORT)*
This table stores the different ActionCodeType-Values

ACTION_CODE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR

ACTION_CODE_ID: PK
DEFAULT_LABEL:

PK: ACTION_CODE_ID

NOTE:

ACTION_CODE_ID	DEFAULT_LABEL
1	"ADD"
2	"UPDATE"
3	"DELETE"

TABLE *ADDITIVITY_CODES (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different AdditivityCodeType-Values

ADDITIVITY_CODE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR

ADDITIVITY_CODE_ID: PK
DEFAULT_LABEL:

PK: ADDITIVITY_CODE_ID

NOTE:

ADDITIVITY_CODE_ID	DEFAULT_LABEL
1	"STOCK"
2	"FLOW"
3	"NON_ADDITIVE"

TABLE *AGGREGATION_METHOD_CODES (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different AggregationMethodCodeType-Values

AGGREGATION_METHOD_CODE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR

AGGREGATION_METHOD_CODE_ID: PK
DEFAULT_LABEL:

PK: AGGREGATION_METHOD_CODE_ID

NOTE:

AGGREGATION_METHOD_CODE_ID	DEFAULT_LABEL
1	"SUM"
2	"AVERAGE"
3	"COUNT"
4	"MODE"
5	"MEDIAN"
6	"MAXIMUM"
7	"MINIMUM"
8	"PERCENT"
9	"CUMULATIVE_PERCENT"
10	"PERCENTILE_RANK"

TABLE *CATEGORY_RELATION_CODES (TABLE_ID: -), (DDI 3 Compliance)*
This table stores the different CategoryRelationCodeType-Values

CATEGORY_RELATION_CODE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR

CATEGORY_RELATION_CODE_ID: PK
DEFAULT_LABEL:

PK: CATEGORY_RELATION_CODE_ID

NOTE:

CATEGORY_RELATION_CODE_ID	DEFAULT_LABEL
1	"NOMINAL"
2	"ORDINAL"
3	"INTERVAL"
4	"RATIO"
5	"CONTINUOUS"

TABLE *CONCATENATED_VALUES (TABLE_ID: -)*
This table stores

CONCATENATED_VALUE_ID	INT NOT NULL
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR
REF_RELATIONTYPE_ID	INT

CONCATENATED_VALUE_ID: PK

TABLE_ID, TABLE_ENTRY_ID: FK (***diff. tables***):

ELEMENT_NAME: <TAG>-Denomination in DDI3

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: CONCATENATED_VALUE_ID

USE:
To access the linked VariableReferences (at least two of them) do this:

1. select the concatenated_value_id where table_id = orig. table_id and table_entry_id = pk of the orig. table and where element_name denominates the column name of the orig. table
2. access the table variable_REFS with table_id = table_id concatenated_values and table_entry_id = concatenated_value_id

NOTE:
This table implements the element *ConcatenatedValue* from *l:RepresentationType*

TABLE *EXCLUDES (TABLE_ID: -), (DDI3: reusable.xsd/ExcludeType)*
This table stores the different ExcludeType-Objects

EXCLUDE_ID	INT NOT NULL AUTO_INCREMENT
ID	VARCHAR
VERSION	INT

EXCLUDE_ID: PK

ID:
VERSION: FK (***VERSIONS***)

PK: EXCLUDE_ID

TABLE *IDS (TABLE_ID:-), (DDI3: reusable.xsd/IDType)*
This table stores the different IDType-Objects

ID_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR
TYPE	VARCHAR DEFAULT('ID')

ID_ID: PK

STRING_VALUE:
TYPE:

PK: ID_ID

TABLE *STRINGS (TABLE_ID: -)*
This table stores the different StringType-Objects

STRING_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR

STRING_ID: PK

STRING_VALUE:

PK: STRING_ID

TABLE *NM_TOKENS (TABLE_ID: -)*
This table stores

NM_TOKEN_ID	INT NOT NULL
STRING_VALUE	VARCHAR

NM_TOKEN_ID: PK

STRING_VALUE:

PK: NM_TOKEN_ID

TABLE *NM_TOKENSS (TABLE_ID: -)*
This table stores

NM_TOKENS_ID	INT NOT NULL
LIST	BOOLEAN DEFAULT ('0')

NM_TOKENS_ID: PK

LIST: "FLAG" (***NM_TOKEN via $_REFS***)

PK: NM_TOKENS_ID

TABLE *NM_TOKEN_REFS (TABLE_ID: -) (new)*
This table stores

NM_TOKEN_REF_ID	INT NOT NULL
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
NM_TOKEN	INT
REF_RELATIONTYPE_ID	INT

NM_TOKEN_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

NM_TOKEN: FK (***NM_TOKENS***)

REF_RELATIONTYPE_ID: FK (*REF_RELATIONTYPES*)

PK: NM_TOKEN_REF_ID

TABLE *URNS (TABLE_ID:-) , (DDI3: reusable.xsd/URNType)*
This table stores the different URNType-Objects

URN_ID	INT NOT NULL AUTO_INCREMENT
URI	VARCHAR
TYPE	VARCHAR DEFAULT('URN')

URN_ID: PK

URI:
TYPE:

PK: URN_ID

TABLE *VERSIONS (TABLE_ID: NA), (DDI3: reusable.xsd/VersionType)*
This table stores the different VersionType-Objects

VERSION_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR

VERSION_ID: PK

STRING_VALUE:

PK: VERSION_ID

SEARCH

TABLE *SEARCHES (TABLE_ID: --)*
This table stores the searches

SEARCH_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
PROJECT_ID	INT
TABLE_ID	INT
TABLE_ENTRY_ID	INT

SEARCH_ID: PK
DEFAULT_LABEL:

PROJECT_ID: FK (***PROJECTS***)

TABLE_ID,
TABLE_ENTRY_ID: Search object
(text: q.v. textual object; sample: q.v. table of sample object)

PK: SEARCH_ID

TABLE *SEARCH_RESULTS (TABLE_ID: --)*
This table stores the (marked) search result objects

SEARCH_RESULT_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
SEARCH_ID	INT
TABLE_ID	INT
TABLE_ENTRY_ID	INT
EXPANDED_NODE	BOOLEAN DEFAULT ('0')
MARKED	BOOLEAN DEFAULT ('0')

SEARCH_RESULT_ID: PK
DEFAULT_LABEL:

SEARCH_ID: FK (***SEARCHES***)

TABLE_ID,
TABLE_ENTRY_ID:

EXPANDED_NODE: (default: FALSE)
MARKED: (default: FALSE)

PK: SEARCH_RESULT_ID

NOTE:
Search results from QDB are referenced through the use of *PERSISTENT_IDS*. In this case TABLE_ID and TABLE_ENTRY_ID stay EMPTY. With the SEARCH_RESULT_ID the table *PERSISTENT_IDS* is referenced,

	MARKED: TRUE	MARKED: FALSE
EXPANDED_NODE: TRUE	Exp. Search Result, marked	NA
EXPANDED_NODE: FALSE	Orig. Search Result, marked	Orig. Search Result

PERSISTENCE

TABLE *PERSISTENT_IDS (TABLE_ID: --)*

TABLE_ID_CONTAINER	SMALLINT UNSIGNED
TABLE_ID_REF	SMALLINT UNSIGNED
TABLE_ENTRY_ID_REF	INT
TABLE_ENTRY_ID	INT
PERSISTENCE_ID	VARCHAR

TABLE_ID_CONTAINER:
TABLE_ID_REF,
TABLE_ENTRY_ID_REF:
TABLE_ENTRY_ID:

PERSISTENCE_ID:

PK: TABLE_ID_CONTAINER, TABLE_ID_REF, TABLE_ENTRY_ID_REF, TABLE_ENTRY_ID

EMBARGO

TABLE *EMBARGOS (TABLE_ID: 0) , (DDI 3 Compliance, reusable.xsd:EmbargoType)*
This table stores the embargo state of any object (text, comment etc)

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
DATE	INT
RATIONALE	BOOLEAN DEFAULT (‘0’)
AGENCY_ORG_REF	INT
ENFORCE_AGENCY_ORG_REF	BOOLEAN DEFAULT (‘0’)

TABLE_ID,
TABLE_ENTRY_ID:

DATE: FK (***DATE_TYPES***)
RATIONALE: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
AGENCY_ORG_REF: FK (***PERSONS***)
ENFORCE_AGENCY_ORG_REF: “FLAG” (***REFERENCES*** *via* ***$_REFS***)

PK: TABLE_ID, TABLE_ENTRY_ID

TABLE *EMBARGO_REFS (TABLE_ID: 1000)*
This table stores the relation between an object and an embargo

EMBARGO_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
EMBARGO	INT
REF_RELATIONTYPE_ID	INT

EMBARGO_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
EMBARGO: FK (***EMBARGOS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: EMBARGO_REF_ID

STUDY

TABLE *STUDIES (TABLE_ID: 10)*
This table stores studies

STUDY_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
STUDY_CATEGORY_ID	INT
INSTITUTION_ID	INT
COPYRIGHT	VARCHAR (256)
PRIMARY_INVESTIGATOR_ID	INT
DATE_OF_COLLECTION	DATE

STUDY_ID: PK
DEFAULT_LABEL: Denomination of the study

STUDY_CATEGORY_ID: FK (***STUDY_CATEGORIES***)

INSTITUTION_ID: FK (***INSTITUTIONS***)
COPYRIGHT:
PRIMARY_INVESTIGATOR_ID: FK (***PERSONS***)
DATE_OF_COLLECTION:

PK: STUDY_ID

TABLE *STUDY_REFERENCES (TABLE_ID: 1010)*
This table stores the relation between an object and a study

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
PROJECT_ID	INT
STUDY_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the STUDY

PROJECT_ID: Flag. If set, this indicates that the assigned study is only valid in the context of the project. (default: NULL)

STUDY_ID: The reference to a STUDY; a foreign key (FK), q.v. *STUDIES*

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: TABLE_ID, TABLE_ENTRY_ID, STUDY_ID

TABLE *STUDY_CATEGORIES (TABLE_ID: 2010), (SUPPORT)*
This table stores the different categories or types a study may belong to

STUDY_CATEGORY_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

STUDY_CATEGORY_ID: PK
DEFAULT_LABEL: Denomination of the universe category

PK: STUDY_CATEGORY_ID

NOTE:
“Longitudinal/Cross-cultural”, “Repeated Cross-sectional/Cross-cultural”, ”Cross-sectional/Mono-cultural”

TABLE *DATASETS (TABLE_ID: 40)*
This table stores the different data sets a data file may belong to

DATASET_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
STUDY_ID	INT

DATASET_ID: PK
DEFAULT_LABEL: Denomination of the data set

STUDY_ID: FK (***STUDIES***) - Reference to the study the sample belongs to

PK: DATASET_ID

TABLE *DATAFILES (TABLE_ID: 20)*
This table stores the available data files per study

DATAFILE_ID	INT NOT NULL AUTO_INCREMENT
DATASET_ID	INT
DATAFILETYPE_ID	INT
FILE	VARCHAR (256)
FILE_VERSION	VARCHAR (64)
FILE_URI	VARCHAR (256)
IS_EXTERNAL	BOOLEAN
ANALYSIS_UNIT_ID	INT
DISTRIBUTOR	INT

DATAFILE_ID: PK

DATASET_ID: FK (***DATASETS***) - Reference to the dataset the data file belongs to

DATAFILETYPE_ID: FK (***DATAFILETYPES***)
FILE:
FILE_VERSION:
FILE_URI:
FILE_EXTERN:
ANALYSIS_UNIT_ID: FK (***ANALYSIS_UNITS***)
DISTRIBUTOR: FK (***PERSONS***)

PK: DATAFILE_ID

TABLE *DATAFILETYPES (TABLE_ID: 2020), (SUPPORT)*
This table stores the different types a data file may belong to

DATAFILETYPE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

DATAFILETYPE_ID: PK
DEFAULT_LABEL: Denomination of the data file type

PK: DATAFILETYPE_ID

NOTE:
“Singular data file”, ”Cumulative/integrated dataset”

TABLE *ANALYSIS_UNITS (TABLE_ID: 3020), (SUPPORT)*
This table stores the different types of analysis unit a data file may belong to

ANALYSIS_UNIT_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

ANALYSIS_UNIT_ID: PK
DEFAULT_LABEL: Denomination of the analysis unit

PK: ANALYSIS_UNIT_ID

TABLE *KIND_OF_DATAS (TABLE_ID: 25), (SUPPORT)*
This table stores the different kind of data a data file may store

KIND_OF_DATA_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

KIND_OFR_DATA_ID: PK
DEFAULT_LABEL: Denomination of the kind of data

PK: KIND_OF_DATA_ID

TABLE *KOD_REFERENCES (TABLE_ID: 1025)*
This table stores the relation between a data file and a stored kind of data

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
KIND_OF_DATA_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to a DATAFILE, recipient of the KOD

KIND_OF_DATA_ID: FK (***KIND_OF_DATAS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: TABLE_ID, TABLE_ENTRY_ID, KIND_OF_DATA_ID

TABLE *SAMPLES (TABLE_ID: 30)*
This table stores the sample information from the studies

SAMPLE_ID	INT NOT NULL AUTO_INCREMENT
STUDY_ID	INT
SAMPLING_DESIGN_ID	INT
SAMPLE_SIZE	INT
RESPONSE_RATE	FLOAT
SAMPLING_ERROR	FLOAT
SAMPLING_WEIGHT	FLOAT

SAMPLE_ID: PK

STUDY_ID: FK (***STUDIES***)

SAMPLING_DESIGN_ID: FK (***SAMPLING_DESIGNS***)
SAMPLE_SIZE:
RESPONSE_RATE:
SAMPLING_ERROR:
SAMPLING_WEIGHT:

PK: SAMPLE_ID

TABLE *SAMPLING_DESIGNS (TABLE_ID: 3030), (SUPPORT)*
This table stores the different sampling designs a sample may belong to

SAMPLING_DESIGN_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

SAMPLING_DESIGN_ID: PK
DEFAULT_LABEL: Denomination of the sampling design

PK: SAMPLING_DESIGN_ID

COMPARISON (DDI3 COMPLIANCE)

TABLE *COMPARISONS (TABLE_ID: 50)*
This table stores every ComparisonType-Object

COMPARISON_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_MAINTAINABLE_TYPE	INT
COMPARISON_DESCRIPTION	BOOLEAN DEFAULT (‘0’)
CONCEPT_MAP	BOOLEAN DEFAULT (‘0’)
VARIABLE_MAP	BOOLEAN DEFAULT (‘0’)
QUESTION_MAP	BOOLEAN DEFAULT (‘0’)
CATEGORY_MAP	BOOLEAN DEFAULT (‘0’)
CODE_MAP	BOOLEAN DEFAULT (‘0’)
UNIVERSE_MAP	BOOLEAN DEFAULT (‘0’)
R_NOTE	BOOLEAN DEFAULT (‘0’)

COMPARISON_ID:	PK
ABSTRACT_MAINTAINABLE_TYPE:	
COMPARISON_DESCRIPTION:	“FLAG” (***STRUCTURED_STRING_TYPES*** via ***$_REF***)
CONCEPT_MAP:	“FLAG” (***GENERIC_MAPS*** via ***$_REF***)
VARIABLE_MAP:	“FLAG” (***GENERIC_MAPS*** via ***$_REF***)
QUESTION_MAP:	“FLAG” (***GENERIC_MAPS*** via ***$_REF***)
CATEGORY_MAP:	“FLAG” (***GENERIC_MAPS*** via ***$_REF***)
CODE_MAP:	“FLAG” (***CODE_MAPS*** via ***$_REF***)
UNIVERSE_MAP:	“FLAG” (***GENERIC_MAPS*** via ***$_REF***)
R_NOTE:	“FLAG” (***NOTES*** via ***$_REF***)
PK:	COMPARISON_ID

TABLE *COMPARISON_REFS (TABLE_ID: 1050)*
This table stores the relation between an object and a comparison, or …

COMPARISON_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
COMPARISON	INT
REF_RELATIONTYPE_ID	INT

COMPARISON_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE:	FK (***REFERENCES***), ***COMPARISONS via REFERENCES***
COMPARISON:	FK (***COMPARISONS***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONTYPES***)
PK:	COMPARISON_REF_ID

TABLE *GENERIC_MAPS (TABLE_ID: 60)*
This table stores the different GenericMapType-Objects

GENERIC_MAP_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT(‘1’)
SOURCE_SCHEME_REFERENCE	INT
TARGET_SCHEME_REFERENCE	INT
CORRESPONDENCE	INT
ITEM_MAP	BOOLEAN DEFAULT (‘0’)

GENERIC_MAP_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VERSIONABLES***)
IS_VERSIONABLE: Is it versionable? (default: TRUE)

SOURCE_SCHEME_REFERENCE: FK (***REFERENCES***)
TARGET_SCHEME_REFERENCE: FK (***REFERENCES***)
CORRESPONDENCE: FK (***CORRESPONDENCES***)
ITEM_MAP: “FLAG” (***ITEM_MAPS*** via ***$_REF***)

PK: GENERIC_MAP_ID

TABLE *GENERIC_MAP_REFS (TABLE_ID: 1060)*
This table stores the relation between an object and a generic map

GENERIC_MAP_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
GENERIC_MAP	INT
REF_RELATIONTYPE_ID	INT

GENERIC_MAP_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***), ***GENERIC_MAPS via REFERENCES***
GENERIC_MAP: FK (***GENERIC_MAPS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: GENERIC_MAP_REF_ID

TABLE *CORRESPONDENCES (TABLE_ID: 70)*
This table stores the different CorrespondenceType-Objects

CORRESPONDENCE_ID	INT NOT NULL AUTO_INCREMENT
COMMONALITY	BOOLEAN DEFAULT (‘0’)
DIFFERENCE	BOOLEAN DEFAULT (‘0’)
COMMONALITY_TYPE_CODED	INT
COMMONALITY_WEIGHT	FLOAT
USER_DEFINED_CORR_PROPERTY	BOOLEAN DEFAULT (‘0’)

CORRESPONDENCE_ID: PK

COMMONALITY: “FLAG” (***STRUCTURED_STRING_TYPES*** via ***$_REF***)
DIFFERENCE: “FLAG” (***STRUCTURED_STRING_TYPES*** via ***$_REF***)
COMMONALITY_TYPE_CODED: FK (***CODE_VALUES***)
COMMONALITY_WEIGHT:
USER_DEFINED_CORR_PROPERTY: “FLAG” (***USER_DEFINED_CORR_PROPERTYS*** via ***$_REF***)

PK: CORRESPONDENCE_ID

TABLE *USER_DEFINED_CORR_PROPERTYS (TABLE_ID: -)*
This table stores the different UserDefinedCorrespondencePropertyType-Objects

USER_DEFINED_CORR_PROPERTY_ID	INT NOT NULL AUTO_INCREMENT
NAME	VARCHAR
VALUE	VARCHAR

USER_DEFINED_CORR_PROPERTY_ID: PK

NAME:
VALUE:

PK: USER_DEFINED_CORR_PROPERTY_ID

TABLE *USER_DEFINED_CORR_PROPERTY_REFS (TABLE_ID: -)*
This table stores the relation between an object and an user defined correspondence property

USER_DEFINED_CORR_PROPERTY_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
USER_DEFINED_CORR_PROPERTY	INT
REF_RELATIONTYPE_ID	INT

USER_DEFINED_CORR_PROPERTY_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***), ***USER… via REFERENCES***
USER_DEFINED_CORR_PROPERTY: FK (***USER_DEFINED_CORR_PROPERTY***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: USER_DEFINED_CORR_PROPERTY_REF_ID

TABLE *ITEM_MAPS (TABLE_ID: -)*
This table stores the different ItemMapType-Objects

ITEM_MAP_ID	INT NOT NULL AUTO_INCREMENT
SOURCE_ITEM	VARCHAR
TARGET_ITEM	VARCHAR
CORRESPONDENCE	INT
ALIAS	VARCHAR

ITEM_MAP_ID: PK

SOURCE_ITEM:
TARGET_ITEM:
CORRESPONDENCE: FK (***CORRESPONDENCES***)
ALIAS:

PK: ITEM_MAP_ID

TABLE *ITEM_MAP_REFS (TABLE_ID: -)*
This table stores the relation between an object and an item map

ITEM_MAP_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
ITEM_MAP	INT
REF_RELATIONTYPE_ID	INT

ITEM_MAP_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***), ***ITEM_MAPS via REFERENCES***
ITEM_MAP: FK (***ITEM_MAPS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: ITEM_MAP_REF_ID

TABLE *CODE_MAPS (TABLE_ID: -)*
This table stores the different CodeMapType-Objects

CODE_MAP_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT (‘1’)
SOURCE_SCHEME_REFERENCE	INT
TARGET_SCHEME_REFERENCE	INT
CORRESPONDENCE	INT
GENERATION_INSTRUCTION	BOOLEAN DEFAULT (‘0’)

CODE_MAP_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (*ABSTRACT_VERSIONABLES*)
IS_VERSIONABLE: Is it versionable? (default: TRUE)

SOURCE_SCHEME_REFERENCE: FK (***REFERENCES***)
TARGET_SCHEME_REFERENCE: FK (***REFERENCES***)
CORRESPONDENCE: FK (***CORRESPONDENCES***)
GENERATION_INSTRUCTION: “FLAG” (***GENERATION_INSTRUCTIONS*** via ***$_REF***)

PK: CODE_MAP_ID

TABLE *CODE_MAP_REFS (TABLE_ID: -)*
This table stores the relation between an object and a code map

CODE_MAP_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CODE_MAP	INT
REF_RELATIONTYPE_ID	INT

CODE_MAP_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***), ***CODE_MAPS via REFERENCES***
CODE_MAP: FK (***CODE_MAPS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: CODE_MAP_REF_ID

VOCABULARY

TABLE *VOCABULARIES (TABLE_ID: 90)*
This table stores every vocabulary

VOCABULARY_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

COMPARISON_ID: PK
DEFAULT_LABEL:

PK: VOCABULARY_ID

NOTE:
This table stores references to actual vocabularies using the table *URIS*.

PROJECT ORGANIZING

TABLE *PROJECTS (TABLE_ID: 100)*
The main table for every conversion project

PROJECT_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
PROJECT_TYPE_ID	INT
PUBLISHED_SINCE	DATETIME
OBSOLETE_SINCE	DATETIME

PROJECT_ID: PK
DEFAULT_LABEL: Title of project

PROJECT_TYPE_ID: Type of project, a foreign key (FK), q.v. *PROJECT_TYPES*

PUBLISHED_SINCE: Date of publishing (default: NULL)
OBSOLETE_SINCE: Date of withdrawing (default: NULL)

PK: PROJECT_ID

TABLE *PROJECT_TYPES (TABLE_ID: 2100), (SUPPORT)*
This table stores the available project types

PROJECT_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

PROJECT_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the project layer type

PK: PROJECT_TYPE_ID

NOTE:

PROJECT_TYPE_ID	DEFAULT_LABEL
0	“Harmonization Project”
1	“Documentation of CSI”
2	“Conceptual Basis of CSI”

TABLE *PRO_RELATIONS (TABLE_ID: 105)*
This table stores the existence and type of relation between projects

PROJECT_ID	INT
RELATED_ID	INT
PRO_REL_TYPE_ID	INT

PROJECT_ID: The reference to a PROJECT
RELATED_ID: Stores a reference to a project the PROJECT substitutes, bases on or modifies (default: 0)

PRO_REL_TYPE_ID: A reference to the type of relation. A foreign key (FK), q.v. *PRO_REL_TYPES* (default: NULL)

PK: PROJECT_ID, RELATED_ID

TABLE *PRO_REL_TYPES (TABLE_ID: 2105), (SUPPORT)*
This table stores the different relations of between projects

PRO_REL_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

PRO_REL_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the relation type

PK: PRO_REL_TYPE_ID

NOTE:

PRO_REL_TYPE_ID	DEFAULT_LABEL
0	"Substitutes"
1	"Based on"
2	"Modifies" BESSER "Varied"

"Substitutes": PROJECT substitutes the RELATED project
"Based on": PROJECT bases on the RELATED project
"Modifies": PROJECT modifies or varied the RELATED project

TABLE *PROJECT_LAYERS (TABLE_ID: 110)*
This table stores the project layers

PROJECT_LAYER_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
PRO_LAYER_TYPE_ID	INT
PUBLISHED_SINCE	DATETIME
OBSOLETE_SINCE	DATETIME

PROJECT_LAYER_ID: PK
DEFAULT_LABEL: Denomination of specific project layer

PRO_LAYER_TYPE_ID: Type of project layer, a foreign key (FK), q.v. *PRO_LAYER_TYPES*

PUBLISHED_SINCE: Date of publishing (default: NULL)
OBSOLETE_SINCE: Date of withdrawing (default: NULL)

PK: PROJECT_LAYER_ID

TABLE *PRO_LAYER_REFERENCES (TABLE_ID: 1110)*
This table stores the relation between a project and a project layer

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
PROJECT_LAYER_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to a PROJECT, recipient of the PROJECT LAYER

PROJECT_LAYER_ID: The reference to a project LAYER; a foreign key (FK), q.v. *PROJECT_LAYERS*

REF_RELATIONTYPE_ID: FK (*REF_RELATIONTYPES*)

PK: TABLE_ID, TABLE_ENTRY_ID, PROJECT_LAYER_ID

TABLE *PRO_LAYER_TYPES (TABLE_ID: 2110), (SUPPORT)*
This table stores the available project layer types

PRO_LAYER_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

PRO_LAYER_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the project layer type

PK: PRO_LAYER_TYPE_ID

NOTE:

PRO_LAYER_TYPE_ID	DEFAULT_LABEL
0	“Conceptual”
1	“Operational”
2	“Data coding”

TABLE *PRO_LAYER_RELATIONS (TABLE_ID: 115)*
This table stores the existence and type of relation between project layers

PROJECT_LAYER_ID	INT
RELATED_LAYER_ID	INT
PRO_LAYER_REL_TYPE_ID	INT

PROJECT_LAYER_ID: ID of a PROJECT LAYER
RELATED_LAYER_ID: Stores a reference to a layer the PROJECT LAYER substitutes, bases on or modifies (default: 0)

PRO_LAYER_REL_TYPE_ID: A reference to the type of relation. A foreign key (FK), q.v. *PRO_LAYER_REL_TYPES*

PK: PROJECT_LAYER_ID, RELATED_LAYER_ID

TABLE *PRO_LAYER_REL_TYPES (TABLE_ID: 2115), (SUPPORT)*
This table stores the different relations of between project layers

PRO_LAYER_REL_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

PRO_LAYER_REL_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the relation type

PK: PRO_LAYER_REL_ID

NOTE:

PRO_LAYER_REL_TYPE_ID	DEFAULT_LABEL
0	“Substitutes”
1	“Based on”
2	“Modifies”

“Substitutes”: PROJECT layer substitutes the RELATED layer
“Based on”: PROJECT layer bases on the RELATED layer
“Modifies”: PROJECT layer modifies the RELATED layer

TABLE *LAYER_NODES (TABLE_ID: 120)*
This table stores the nodes belonging to a layer

LAYER_NODE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
TABLE_ID	SMALLINT UNSIGNED
TABLE_ENTRY_ID	INT
X_POS	FLOAT
Y_POS	FLOAT
LAYER_NODE_TYPE_ID	INT

LAYER_NODE_ID: PK
DEFAULT_LABEL: Denomination of a node

TABLE_ID,
TABLE_ENTRY_ID: A reference to the containing layer, e. q. *CONCEPTUALS, OPERATIONALIZATIONS, DATA_CODINGS*

X_POS: Horizontal Position
Y_POS: Vertical Position

LAYER_NODE_TYPE_ID: FK (***LAYER_NODE_TYPES***)

PK: LAYER_NODE_ID

TABLE *LAYER_NODE_TYPES (TABLE_ID: 2120), (SUPPORT)*
This table stores the available layer node types

LAYER_NODE_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

LAYER_NODE_TYPE_ID: PK

DEFAULT_LABEL: Denomination of the node type

PK: LAYER_NODE_TYPE_ID

NOTE:
Types includes Dimension, Indicator, Category, logical operators, arithmetical operators and a free-text-field

TABLE *LAYER_NODE_REFERENCES (TABLE_ID: 1120)*
This table stores the relation between a layer node and another one from a different layer

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
LAYER_NODE_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to a LAYER NODE

LAYER_NODE_ID: The reference to a LAYER NODE; a foreign key (FK), q.v. *LAYER_NODES*

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: TABLE_ID, TABLE_ENTRY_ID, LAYER_NODE_ID

TABLE *LAYER_EDGES (TABLE_ID: 125) (changed)*
This table stores the edges between nodes

LAYER_EDGE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
TABLE_ID	SMALLINT UNSIGNED
TABLE_ENTRY_ID	INT
NODE_SOURCE	INT NOT NULL
NODE_TARGET	INT
LAYER_EDGE_TYPE_ID	INT

LAYER__EDGE_ID: PK
DEFAULT_LABEL: Denomination of a node

TABLE_ID,
TABLE_ENTRY_ID: A reference to the containing layer, e. q. *CONCEPTUALS, OPERATIONALIZATIONS, DATA_CODINGS*
NODE_SOURCE: FK (***LAYER_NODES***)
NODE_TARGET: FK (***LAYER_NODES***)

LAYER_EDGE_TYPE_ID: FK (***LAYER_EDGE_TYPES***)

PK: LAYER_EDGE_ID

TABLE *LAYER_EDGE_TYPES (TABLE_ID: 2125), (SUPPORT)*
This table stores the available layer node types

LAYER_EDGE_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

LAYER_EDGE_TYPE_ID: PK

DEFAULT_LABEL: Denomination of the node type

PK: LAYER_EDGE_TYPE_ID

NOTE:
Types includes Specification, Value

TABLE *LAYER_EDGE_REFERENCES (TABLE_ID: 1125)*
This table stores the relation between an edge and another one from a different layer

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
LAYER_EDGE_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to a LAYER EDGE

LAYER_EDGE_ID: The reference to a LAYER EDGE; a foreign key (FK), q.v. *LAYER_EDGES*

REF_RELATIONTYPE_ID: FK (***REF_RELATIONTYPES***)

PK: TABLE_ID, TABLE_ENTRY_ID, LAYER_EDGE_ID

USER MANAGING

TABLE *PARTICIPANTS (TABLE_ID: 200)*

PARTICIPANT_ID	INT NOT NULL AUTO_INCREMENT
PROJECT_ID	INT NOT NULL
PERSON_ID	INT NOT NULL
PARTICIPANT_ROLE_ID	INT

PARTICIPANT_ID: PK

PROJECT_ID: FK (***PROJECTS***)
PERSON_ID: FK (***PERSONS***)

PARTICIPANT_ROLE_ID: FK (***PARTICIPANT_ROLES***)

PK: PARTICIPANT_ID

TABLE *PARTICIPANT_ROLES (TABLE_ID: 2200), (SUPPORT)*

PARTICIPANT_ROLE_ID	INT NOT NULL
PARTICIPANT_ROLE	VARCHAR (64)

PARTICIPANT_ROLE_ID: PK
PARTICIPANT_ROLE:

PK: PARTICIPANT_ROLE_ID

NOTE:
Typical roles are: 'CREATOR', 'CONTRIBUTOR', 'PUBLISHER'.

TABLE *USERS (TABLE_ID: 210)*

USER_ID	INT NOT NULL AUTO_INCREMENT
PERSON_ID	INT
USER_NAME	VARCHAR (64)
USER_PASSWORD	VARCHAR (256)
USER_ROLE_ID	INT

USER_ID: PK
PERSON_ID: FK (***PERSONS***)
USER:
PASSWORD:
USER_ROLE_ID: FK (***USER_ROLES***)

PK: USER_ID

TABLE *USER_ROLES (TABLE_ID: 2210), (SUPPORT)*

USER_ROLE_ID	INT NOT NULL
USER_ROLE	VARCHAR (64)

USER_ROLE_ID: PK
USER_ROLE:

PK: USER_ROLE_ID

TABLE *PERSONS (TABLE_ID: 220)*

PERSON_ID	INT NOT NULL AUTO_INCREMENT
INDIVIDUAL_ID	INT
INSTITUTION_ID	INT
PHONE	VARCHAR (16)
MOBILE_PHONE	VARCHAR (16)
EMAIL	VARCHAR (320)

PERSON_ID: PK

INDIVIDUAL_ID: FK (***INDIVIDUALS***)
INSTITUTION_ID: FK (***INSTITUTIONS***)

PHONE: q.v. E.164
MOBILE_PHONE: q.v. E.164
EMAIL:

PK: PERSON_ID

TABLE *AUTHORSHIPS (TABLE_ID: 230)*
This table stores the authorship of any object (text, comment etc)

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
PERSON_ID	INT

TABLE_ID,
TABLE_ENTRY_ID:

PERSON_ID: FK (***PERSONS***)

PK: TABLE_ID, TABLE_ENTRY_ID, PERSON_ID

NOTE:
This Table has to be filled by a TRIGGER, though a PROVIDER should be able to overwrite the entry.

TABLE *INDIVIDUALS (TABLE_ID: 240)*

INDIVIDUAL_ID	INT NOT NULL AUTO_INCREMENT
FORM_OF_ADDRESS	VARCHAR (64)
TITLE	VARCHAR (64)
FIRST_NAME	VARCHAR (64)
MIDDLE_NAME	VARCHAR (64)
LAST_NAME	VARCHAR (64)
DATE_OF_BIRTH	DATE
PLACE_OF_BIRTH	VARCHAR (64)
ADDRESS_ID	INT

INDIVIDUAL_ID:	PK
FORM_OF_ADDRESS:	
TITLE:	
FIRST_NAME:	
MIDDLE_NAME:	
LAST_NAME:	
DATE_OF_BIRTH:	
PLACE_OF_BIRTH:	
ADDRESS_ID:	
PK:	INDIVIDUAL_ID

TABLE *INSTITUTIONS (TABLE_ID: 250)*

INSTITUTION_ID	INT NOT NULL AUTO_INCREMENT
INSTITUTION	VARCHAR (64)
DEPARTMENT	VARCHAR (64)
ADDRESS_ID	INT
LOGO_URI	VARCHAR (256)

INSTITUTION_ID:	PK
INSTITUTION:	
DEPARTMENT:	
ADDRESS_ID:	FK (***ADDRESS***)
LOGO_URI:	
PK:	INSTITUTION_ID

TABLE *AFFILIATIONS (TABLE_ID: 260)*
Which individual belongs to which institution?

INDIVIDUAL_ID	INT NOT NULL
INSTITUTION_ID	INT NOT NULL
AFFILIATION	VARCHAR (256)

INDIVIDUAL_ID:	FK (***INDIVIDUALS***)
INSTITUTION_ID:	FK (***INSTITUTIONS***)
AFFILIATION:	
PK:	INDIVIDUAL_ID, INSTITUTION_ID

TABLE *ADDRESS (TABLE_ID: 270)*

ADDRESS_ID	INT NOT NULL AUTO_INCREMENT
COUNTRY_ID	INT
CITY	VARCHAR (64)
ZIP_CODE	VARCHAR (64)
STREET	VARCHAR (64)
STREET_NUMBER	VARCHAR (11)

ADDRESS_ID:	PK
COUNTRY_ID:	FK (***COUNTRIES***)
CITY:	
ZIP_CODE:	
STREET:	
STREET_NUMBER:	
PK:	ADDRESS_ID

TABLE *COUNTRIES (TABLE_ID: 275), (SUPPORT)*

COUNTRY_ID	INT NOT NULL
COUNTRY_CODE	VARCHAR (64)
COUNTRY_CODE_NORM	VARCHAR (64)

COUNTRY_ID:	PK
COUNTRY_CODE:	e. g. „DEU“
COUNTRY_CODE_NORM:	e. g. „ISO 3166-1 Alpha-3“
PK:	COUNTRY_ID

UNIVERSE

TABLE *UNIVERSES (TABLE_ID: 300)*
This table stores universes

UNIVERSE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
UNI_CATEGORY_ID	INT

UNIVERSE_ID: PK
DEFAULT_LABEL:

UNI_CATEGORY_ID: FK (***UNI_CATEGORIES***)

PK: UNIVERSE_ID

TABLE *UNIVERSE_REFS (TABLE_ID: 1300), (DDI 3 Compliance)*
This table stores the relation between an object and a universe

UNIVERSE_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
UNIVERSE	INT
REF_RELATIONTYPE_ID	INT

UNIVERSE_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
UNIVERSE: FK (***UNIVERSES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: UNIVERSE_REF_ID

NOTE:
This table implements the element UniverseReference from l:VariableType, l:VariableGroupType (DDI 3)
The following relation types are additional to be supported:
'subtype' – the referenced universe is a subuniverse to another one.

TABLE *UNI_CATEGORIES (TABLE_ID: 2300), (SUPPORT)*
This table stores the different categories a universe may belong to

UNI_CATEGORY_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

UNI_CATEGORY_ID: PK
DEFAULT_LABEL: Denomination of the universe category

PK: UNI_CATEGORY_ID

TABLE *UNI_SCHEMES (TABLE_ID: 301)*
This table stores universe schemes

UNI_SCHEME_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

UNI_SCHEME_ID: PK
DEFAULT_LABEL: Denomination of the universe scheme

PK: UNI_SCHEME_ID

TABLE *UNI_SCH_REFERENCES (TABLE_ID: 1301)*
This table stores the relation between an object and a universe scheme

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
UNI_SCHEME_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the UNIVERSE SCHEME

UNI_SCHEME_ID: FK (***UNI_SCHEMES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, UNIVERSE_SCHEME_ID

CONCEPT

TABLE *CONCEPTS (TABLE_ID: 350)*
This table stores the different concepts

CONCEPT_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
CONCEPT_CATEGORY_ID	INT

CONCEPT_ID: PK
DEFAULT_LABEL: Denomination of the concept

CONCEPT_CATEGORY_ID: FK (***CONCEPT_CATEGORIES***)

PK: CONCEPT_ID

TABLE *CONCEPT_REFS (TABLE_ID: 1350), (DDI 3 Compliance)*
This table stores the relation between an object and a concept

CONCEPT_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CONCEPT	INT
REF_RELATIONTYPE_ID	INT

CONCEPT_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
CONCEPT: FK (***CONCEPTS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: CONCEPT_REF_ID

TABLE *CONCEPT_CATEGORIES (TABLE_ID: 2350), (SUPPORT)*
This table stores the different categories a concept may belong to

CONCEPT_CATEGORY_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

CONCEPT_CATEGORY_ID: PK
DEFAULT_LABEL: Denomination of the universe category

PK: CONCEPT_CATEGORY_ID

TABLE *CONCEPT_SCHEMES (TABLE_ID: 351)*
This table stores concept schemes

CONCEPT_SCHEME_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

CONCEPT_SCHEME_ID: PK
DEFAULT_LABEL: Denomination of the concept scheme

PK: CONCEPT_SCHEME_ID

TABLE *CONCEPT_SCH_REFERENCES (TABLE_ID: 1351)*
This table stores the relation between an object and a concept scheme

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
CONCEPT_SCHEME_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the CONCEPT SCHEME

CONCEPT_SCHEME_ID: FK (***CONCEPT_SCHEMES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, CONCEPT_SCHEME_ID

TABLE *CONCEPT_GROUPS (TABLE_ID: 355)*
This table stores concept groups

CONCEPT_GROUP_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

CONCEPT_GROUP_ID: PK
DEFAULT_LABEL: Denomination of the concept group

PK: CONCEPT_GROUP_ID

TABLE *CONCEPT_GR_REFERENCES (TABLE_ID: 1355)*
This table stores the relation between an object and a concept group

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
CONCEPT_GROUP_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the CONCEPT GROUP

CONCEPT_GROUP_ID: FK (***CONCEPT_GROUPS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, CONCEPT_GROUP_ID

TEXT STORING

TABLE *KEYWORDS (TABLE_ID: 400)*
This table stores keywords

KEYWORD_ID	INT NOT NULL AUTO_INCREMENT
KEYWORD	VARCHAR (256)
LANG_CODE	VARCHAR (64)
LANG_CODE_NORM	VARCHAR (64)
TRANSLATABLE	BOOLEAN DEFAULT (‘1’)
TRANSLATION	BOOLEAN DEFAULT (‘0’)
TRANSLATION_ORIGIN	INT

KEYWORD_ID: PK

KEYWORD: The stored keyword

LANG_CODE: e. g. „deu“
LANG_CODE_NORM: e. g. „ISO 639-3“
TRANSLATABLE: TRUE, if translatable (default: TRUE)
TRANSLATION: TRUE, if translated (default: FALSE)
TRANSLATION_ORIGIN: FK (***KEYWORDS***) – Ref. to the original language keyword (default: NULL)

PK: KEYWORD_ID

TABLE *KEYWORD_REFERENCES* (***TABLE_ID: 1400***)
This table stores the relation between class objects and keywords

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
PROJECT_ID	INT
KEYWORD_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***)

KEYWORD_ID: FK (***KEYWORDS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, KEYWORD_ID

TABLE *COMMENTS (TABLE_ID: 410)*
This table stores the comments

COMMENT_ID	INT NOT NULL AUTO_INCREMENT
OBJECT_TABLE_ID	SMALLINT UNSIGNED
OBJECT_TABLE_ENTRY_ID	INT
PROJECT_ID	INT
COMMENT_TYPE_ID	INT
SUBJECT	VARCHAR (256)
COMMENT_TEXT	VARCHAR (2000)
LANG_CODE	VARCHAR (64)
LANG_CODE_NORM	VARCHAR (64)
TRANSLATABLE	BOOLEAN DEFAULT (‘1’)
TRANSLATION	BOOLEAN DEAFULT (‘0’)
TRANSLATION_ORIGIN	INT

COMMENT_ID:	PK
OBJECT_TABLE_ID, OBJECT_TABLE_ENTRY_ID:	Reference to the object the commentary aims at
PROJECT_ID:	Flag. If set, this indicates that the assigned comment is only valid in the context of the project. (default: NULL)
COMMENT_TYPE_ID:	A foreign key (FK), q.v. *COMMENT_TYPES*
SUBJECT: COMMENT:	The subject of the stored comment The stored comment text
LANG_CODE: LANG_CODE_NORM: TRANSLATABLE: TRANSLATION: TRANSLATION_ORIGIN:	e. g. „deu“ e. g. „ISO 639-3“ TRUE, if translatable (default: TRUE) TRUE, if translated (default: FALSE) FK (*COMMENTS*) - Ref. to the original language comment (default: NULL)
PK:	COMMENT_ID

TABLE *COMMENT_TYPES (TABLE_ID: 2410), (SUPPORT)*
This table stores the different types of comment available for use.

COMMENT_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

COMMENT_TYPE_ID: DEFAULT_LABEL:	PK Denomination of the type of comment
PK:	COMMENT_TYPE_ID

TABLE *NOTES (TABLE_ID: -)*
This table stores

NOTE_ID	INT NOT NULL AUTO_INCREMENT
SUBJECT	VARCHAR
RELATIONSHIP	BOOLEAN DEFAULT (‘0’)
RESPONSIBILITY	VARCHAR
HEADER	VARCHAR
CONTENT	INT
TYPE	INT

NOTE_ID: PK

SUBJECT:
RELATIONSHIP: “FLAG” (***RELATIONSHIPS*** via ***$_REF***)
RESPONSIBILITY:
HEADER:
CONTENT: FK (***STRUCTURED_STRINGS***)
TYPE: FK (***NOTE_TYPE_CODES***)

PK: CODE_ID

TABLE *NOTE_TYPE_CODES (TABLE_ID: -)*

NOTE_TYPE_CODE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (20)

NOTE_TYPE_CODE_ID: PK
DEFAULT_LABEL:

PK: NOTE_TYPE_CODE_ID

NOTE:

NOTE_TYPE_CODE_ID	DEFAULT_LABEL
1	“PROCESSING”
2	“FOOTNOTE”
3	“ADDENDUM”
4	“SYSTEM”
5	“PROBLEM”
6	“COMMENT”
7	“OTHER”

TABLE *NOTE_REFS (TABLE_ID: -)*
This table stores

NOTE_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
NOTE	INT
REF_RELATIONTYPE_ID	INT

NOTE_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***), ***NOTES via REFERENCES***
NOTE: FK (***NOTES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: NOTE_REF_ID

TABLE *PROPERTIES (TABLE_ID: 420), (CONTEXTUAL INFORMATION)*
This table stores the available contextual information

PROPERTY_ID	INT NOT NULL AUTO_INCREMENT
PROPERTY *(NAME)*	VARCHAR (256)
CONTENT *(VALUE)*	VARCHAR (2000)
TIMESTAMP	DATETIME
LANG_CODE	VARCHAR (64)
LANG_CODE_NORM	VARCHAR (64)
TRANSLATABLE	BOOLEAN DEFAULT (‘1’)
TRANSLATION	BOOLEAN DEFAULT (‘0’)
TRANSLATION_ORIGIN	INT

PROPERTY_ID: PK

PROPERTY (NAME):
CONTENT (VALUE):
TIMESTAMP: Time constraint of the name-value pair

LANG_CODE: e. g. „deu“
LANG_CODE_NORM: e. g. „ISO 639-3“
TRANSLATABLE: TRUE, if translatable (default: TRUE)
TRANSLATION: TRUE, if translated (default: FALSE)
TRANSLATION_ORIGIN: FK (***PROPERTIES***) - Ref. to the original language property (default: NULL)

PK: PROPERTY_ID

TABLE *PROPERTY_REFERENCES (TABLE_ID: 1420)*
This table stores the relation between an object and a property

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
PROJECT_ID	INT
PROPERTY_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the answer value, e. g. an answer category.

PROJECT_ID: Flag. If set, this indicates that the assigned property is only valid in the context of the project. (default: NULL)

PROPERTY_ID: FK (***PROPERTIES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, PROPERTY_ID

TABLE *TEXTUAL_OBJECTS (TABLE_ID: 430)*
This table stores the reference of text for different objects in the conversion project

TEXTUAL_OBJECT_ID	INT NOT NULL AUTO_INCREMENT
OBJECT_TABLE_ID	SMALLINT UNSIGNED
OBJECT_TABLE_ENTRY_ID	INT
PROJECT_ID	INT
TEXT_TYPE_ID	INT
DEFAULT *(new)*	BOOLEAN
TEXT_TABLE_ID	SMALLINT UNSIGNED
TEXT_TABLE_ENTRY_ID	INT
LANG_CODE	VARCHAR (64)
LANG_CODE_NORM	VARCHAR (64)
TRANSLATABLE	BOOLEAN DEFAULT ('1')
TRANSLATION	BOOLEAN DEFAULT ('0')
TRANSLATION_ORIGIN	INT

TEXTUAL_OBJECT_ID: PK

OBJECT_TABLE_ID,
OBJECT_TABLE_ENTRY_ID: Reference to the object whose text is stored

PROJECT_ID: Flag. If set, this indicates that the assigned study is only valid in the context of the project. (default: NULL)

TEXT_TYPE_ID: A reference to the type of text. A foreign key (FK), q.v. *TEXT_TYPES*

DEFAULT *(new)*:

TEXT_TABLE_ID,
TEXT_TABLE_ENTRY_ID: Reference to the text which is stored

LANG_CODE: e. g. „deu"
LANG_CODE_NORM: e. g. „ISO 639-3"
TRANSLATABLE: TRUE, if translatable (default: TRUE)
TRANSLATION: TRUE, if translated (default: FALSE)
TRANSLATION_ORIGIN: Reference to the original language text (default: NULL)

PK: TEXTUAL_OBJECT_ID

TABLE *TEXT_TYPES (TABLE_ID: 2430), (SUPPORT)*
This table stores the different types of text

TEXT_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

TEXT_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the text type

PK: TEXT_TYPE_ID

NOTE:

TEXT_TYPE_ID	DEFAULT_LABEL
0	"ABBR"
1	"ABSTRACT"
2	"DEFINITION"
3	"DESCRIPTION"
4	"INSTRUCTION"
5	"LABEL"
6	"COMMONALITY"
7	"DIFFERENCE"

TABLE *LONG_TEXTS (TABLE_ID: 431)*
This table stores long text for different objects in the conversion project

LONG_TEXT_ID	INT NOT NULL AUTO_INCREMENT
LONG_TEXT	VARCHAR (2000)

LONG_TEXT_ID: PK

LONG_TEXT: The stored long text

PK: LONG_TEXT_ID

NOTE:
LONG_TEXT_ID “0”: “no text available”

TABLE *TEXTS (TABLE_ID: 432)*
This table stores text for different objects in the conversion project

TEXT_ID	INT NOT NULL AUTO_INCREMENT
TEXT	VARCHAR (256)

TEXT_ID: PK

TEXT: The stored text

PK: TEXT_ID

NOTE:
TEXT_ID “0”: “no text available”

TABLE *SHORT_TEXTS (TABLE_ID: 433)*
This table stores short text for different objects in the conversion project

SHORT_TEXT_ID	INT NOT NULL AUTO_INCREMENT
SHORT_TEXT	VARCHAR (64)

SHORT_TEXT_ID: PK

SHORT_TEXT: The stored short text

PK: SHORT_TEXT_ID

NOTE:
SHORT_TEXT_ID “0”: “no text available”

REFERENCE

TABLE *CITATIONS (TABLE_ID: 440)*
This table stores citations

CITATION_ID	INT NOT NULL AUTO_INCREMENT
TITLE	BOOLEAN DEFAULT ('0')
SUBTITLE	BOOLEAN DEFAULT ('0')
ALTERNATE_TITLE	BOOLEAN DEFAULT ('0')
CREATOR	BOOLEAN DEFAULT ('0')
PUBLISHER	BOOLEAN DEFAULT ('0')
CONTRIBUTOR	BOOLEAN DEFAULT ('0')
PUBLICATION_DATE	
LANGUAGE	VARCHAR
INTERNATIONAL_IDENTIFIER	BOOLEAN DEFAULT ('0')
COPYRIGHT	INT
DC_ELEMENTS	INT

CITATION_ID:	PK
TITLE:	"FLAG" (***INTERNATIONAL_STRINGS*** *via* ***$_REFS***)
SUBTITLE:	"FLAG" (***INTERNATIONAL_STRINGS*** *via* ***$_REFS***)
ALTERNATE_TITLE:	"FLAG" (***INTERNATIONAL_STRINGS*** *via* ***$_REFS***)
CREATOR:	"FLAG" (***CREATORS*** *via* ***$_REFS***)
PUBLISHER:	"FLAG" (***INTERNATIONAL_STRINGS*** *via* ***$_REFS***)
CONTRIBUTOR:	"FLAG" (***CONTRIBUTORS*** *via* ***$_REFS***)
PUBLICATION_DATE:	FK (***DATES***)
LANGUAGE:	
INTERNATIONAL_IDENTIFIER:	"FLAG" (***TYPED_STRINGS*** *via* ***$_REFS***)
COPYRIGHT:	FK (***INTERNATIONAL_STRINGS)***
DC_ELEMENTS:	FK (***DC_ELEMENTS***)
PK:	CITATION_ID

TABLE *CITATION_REFS (TABLE_ID: 1440)*
This table stores

CITATION_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
CITATION	INT
REF_RELATIONTYPE_ID	INT

CITATION_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE:	FK (***REFERENCES***)
CITATION:	FK (***CITATIONS***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONSHIP***)
PK:	CITATION_REF_ID

NOTE:
RELATION stores values like: 'ORIGINAL', 'MODIFIED' or 'ADDITIONAL'

TABLE *DC_ELEMENTS (TABLE_ID: NA) , (DDI 3 Compliance)*
This table stores

DC_ELEMENTS_ID	INT NOT NULL AUTO_INCREMENT
TITLE	BOOLEAN DEFAULT (‘0’)
CREATOR	BOOLEAN DEFAULT (‘0’)
SUBJECT	BOOLEAN DEFAULT (‘0’)
DESCRIPTION	BOOLEAN DEFAULT (‘0’)
PUBLISHER	BOOLEAN DEFAULT (‘0’)
CONTRIBUTOR	BOOLEAN DEFAULT (‘0’)
DATE	BOOLEAN DEFAULT (‘0’)
TYPE	BOOLEAN DEFAULT (‘0’)
FORMAT	BOOLEAN DEFAULT (‘0’)
IDENTIFIER	BOOLEAN DEFAULT (‘0’)
SOURCE	BOOLEAN DEFAULT (‘0’)
LANGUAGE	BOOLEAN DEFAULT (‘0’)
RELATION	BOOLEAN DEFAULT (‘0’)
COVERAGE	BOOLEAN DEFAULT (‘0’)
RIGHTS	BOOLEAN DEFAULT (‘0’)

DC_ELEMENTS_ID: PK

TITLE: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
CREATOR: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
SUBJECT: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
DESCRIPTION: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
PUBLISHER: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
CONTRIBUTOR: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
DATE: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
TYPE: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
FORMAT: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
IDENTIFIER: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
SOURCE: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
LANGUAGE: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
RELATION: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
COVERAGE: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)
RIGHTS: “FLAG” (***DC_ELEMENT_TYPES*** *via* ***$_REFS***)

PK: DC_ELEMENTS_ID

TABLE *DC_ELEMENT_TYPES (TABLE_ID: NA) , (DDI 3 Compliance)*
This table stores

DC_ELEMENT_TYPE_ID	INT NOT NULL AUTO_INCREMENT
STRING_VALUE	VARCHAR
XML_LANG	VARCHAR (64)

DC_ELEMENT_TYPE_ID: PK

STRING_VALUE:

XML_LANG:

PK: DC_ELEMENT_TYPE_ID

TABLE *DC_ELEMENT_TYPE_REFS (TABLE_ID: -)*
This table stores

DC_ELEMENT_TYPE_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
DC_ELEMENT_TYPE	INT
REF_RELATIONTYPE_ID	INT

DC_ELEMENT_TYPE_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***)
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE:	FK (***REFERENCES***)
DC_ELEMENT_TYPE:	FK (***DC_ELEMENT_TYPES***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONSHIP***)
PK:	DC_ELEMENT_TYPE_ID

TABLE *URIS (TABLE_ID: 450)*
This table stores URIs

URI_ID	INT NOT NULL AUTO_INCREMENT
URI	VARCHAR (256)
LANG_CODE	VARCHAR (64)
LANG_CODE_NORM	VARCHAR (64)

URI_ID: PK

URI:

LANG_CODE: e. g. „deu“
LANG_CODE_NORM: e. g. „ISO 639-3“

PK: URI_ID

TABLE *URI_REFERENCES (TABLE_ID: 1450)*
This table stores the relation between an object and an URI

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
PROJECT_ID	INT
URI_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the URI

PROJECT_ID: Flag. If set, this indicates that the assigned URI is only valid in the context of the project. (default: NULL)

URI_ID: FK (***URIS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, PROPERTY_ID

MEASUREMENT

TABLE *MEASUREMENTS (TABLE_ID: 500)*
This table stores the different classifications, scales or indexes

MEASUREMENT_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
MEA_TYPE_ID	INT

MEASUREMENT_ID: PK
DEFAULT_LABEL: e.g. "ESeC"

MEA_TYPE_ID: FK (***MEA_TYPES***)

PK: MEASUREMENT_ID

TABLE *MEA_REFERENCES (TABLE_ID: 1500)*
This table stores the relation between an object and a classification, scale or index

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
MEASUREMENT_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the classification, scale or index

MEASUREMENT_ID: FK (***MEASUREMENTS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, MEASUREMENT_ID

TABLE *MEA_TYPES (TABLE_ID: 2500), (SUPPORT)*
This table stores the different types of measurement

MEA_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

MEA_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the measurement type

PK: MEA_TYPE_ID

NOTE:

MEA_TYPE_ID	DEFAULT_LABEL
0	"Classification"
1	"Scale"
2	"Index"

TABLE *MEA_CSI_VALUES (TABLE_ID: 510)*
This table stores the different categories / values

MEA_CSI_VALUE_ID	INT NOT NULL AUTO_INCREMENT
MEASUREMENT_ID	INT
CODE	VARCHAR (256)
DEFAULT_LABEL	VARCHAR (256)
INTERPRETATION	VARCHAR (256)
PARENT_CSI_VALUE_ID	INT

MEA_CSI_VALUE_ID: PK

MEASUREMENT_ID: FK (***MEASUREMENTS***)

CODE: e.g. "3"
DEFAULT_LABEL: e.g. "Intermediate occupations"

INTERPRETATION:

PARENT_CSI_VALUE_ID: References to a parent element in a hierarchical structure – FK (*MEA_CSI_VALUES*)

PK: MEA_CSI_VALUE_ID

TABLE *MEA_CSI_VAL_REFERENCES (TABLE_ID: 1510)*
This table stores the relation between an object and a category or value

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
MEA_CSI_VALUE_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the category or value, q.v. *PRO_LAYER_NODES*.

MEA_CSI_VALUE_ID: FK (***MEA_CSI_VALUES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, MEASUREMENT_ID

TABLE *MEA_OBS_STRUCTURES (TABLE_ID: 520)*
This table stores the observed structure measured through the use of a classification, scale or index.
For example: “Occupation” in case of ESeC.

MEA_OBS_STRUCTURE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

MEA_OBS_STRUCTURE_ID: PK
DEFAULT_LABEL: Denomination of the observed structure

PK: MEA_OBS_STRUCTURE_ID

TABLE *MEA_OBS_STRUCTURES_REFERENCES (TABLE_ID: 1520)*
This table stores the relation between an object and an observed structure

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
MEA_OBS_STRUCTURE_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, such as *CON_DIMENSIONS*, recipient of the observed structure

MEA_OBS_STRUCTURE_ID: FK (***MEA_OBS_STRUCTURES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, MEA_OBS_STRUCTURE_ID

TABLE *MEA_VALIDITIES (TABLE_ID: 530)*
This table stores the validity information for a classification or scale

MEA_VALIDITY_ID	INT NOT NULL AUTO_INCREMENT
MEASUREMENT_ID	INT
TEXT_ID	INT

MEA_VALIDITY_ID: PK

MEASUREMENT_ID: FK (***MEASUREMENTS***)

TEXT_ID: FK (***TEXTUAL_OBJECTS***)

PK: MEA_VALIDITY_ID

TABLE *MEA_RELIABILITIES (TABLE_ID: 540)*
This table stores the reliability information for a classification or scale

MEA_RELIABILITY_ID	INT NOT NULL AUTO_INCREMENT
MEASUREMENT_ID	INT
TEXT_ID	INT

MEA_RELIABILITY_ID: PK

MEASUREMENT_ID: FK (***MEASUREMENTS***)

TEXT_ID: FK (***TEXTUAL_OBJECTS***)

PK: MEA_RELIABILITY_ID

CONCEPTUAL (DERIVATION) LAYER

TABLE *CONCEPTUALS (TABLE_ID: 600)*
This table stores conceptual levels

CONCEPTUAL_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

CONCEPTUAL_ID: PK
DEFAULT_LABEL:

PK: CONCEPTUAL_ID

TABLE *CON_REFERENCES (TABLE_ID: 1600)*

TABLE_ID	SMALLINT UNSIGNED NOT NULl
TABLE_ENTRY_ID	INT NOT NULL
CONCEPTUAL_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the conceptual layer, e.g. *PROJECT_LAYERS*

CONCEPTUAL_ID: FK (***CONCEPTUALS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, CONCEPTUAL_ID

TABLE *CON_DIMENSIONS (TABLE_ID: 610*)
This table stores conceptual dimensions

CON_DIMENSION_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

CON_DIMENSION_ID: PK
DEFAULT_LABEL:

PK: CON_DIMENSION_ID

TABLE *CON_DIM_REFERENCES (TABLE_ID: 1610)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
CON_DIMENSION_ID	INT NOT NULL
CON_DIM_TYPE_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the dimension, e.g. *LAYER_NODES*, *CONCEPTUALS*

CON_DIMENSION_ID: FK (***CON_DIMENSIONS***)

DIMENSION_TYPE_ID: FK (***CON_DIM_TYPES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, CON_DIMENSION_ID

TABLE *CON_DIM_TYPES (TABLE_ID: 2610), (SUPPORT)*
This table stores the different types of dimensions

CON_DIM_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

CON_DIM_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the dimension type

PK: CON_DIM_TYPE_ID

NOTE:

CON_DIM_TYPE_ID	DEFAULT_LABEL
0	"Source"
1	"Auxiliary"
2	"Target"

"Source"; Variable is used as SOURCE dimension
"Auxiliary" Variable is an AUXILIARY dimension
"Target": Variable is the new TARGET dimension

TABLE *CON_SPECIFICATIONS (TABLE_ID: 620)*
This table stores conceptual specifications

CON_SPECIFICATION_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	

CON_SPECIFICATION_ID: PK
DEFAULT_LABEL:

PK: CON_SPECIFICATION_ID

TABLE *CON_SPE_REFERENCES (TABLE_ID: 1620)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
CON_SPECIFICATION_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the specification, e.g. *LAYER_EDGES*, *CONCEPTUALS*

CON_DIMENSION_ID: FK (***CON_SPECIFICATIONS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, CON_SPECIFICATION_ID

TABLE *ATTRIBUTES (TABLE_ID: 630)*
This table stores the different attributes

ATTRIBUTE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

ATTRIBUTE_ID: PK
DEFAULT_LABEL: Denomination of the concept

PK: ATTRIBUTE_ID

TABLE *ATTRIBUTE_REFERENCES (TABLE_ID: 1630)*
This table stores the relation between an object and an attribute

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ATTRIBUTE_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the ATTRIBUTE

ATTRIBUTE_ID: FK (***ATTRIBUTES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, ATTRIBUTE_ID

OPERATIONAL (DEFINITION) LAYER

TABLE *OPERATIONALIZATIONS (TABLE_ID: 700)*
This table stores the operationalization of a conceptual object for harmonization

OPERATIONALIZATION_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

OPERATIONALIZATION_ID: PK

DEFAULT_LABEL:

PK: OPERATIONALIZATION_ID

NOTE:
Reference to the conceptual object for harmonization; q.v. *CON_REFERENCES.*

TABLE *OPERA_REFERENCES (TABLE_ID: 1700)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
OPERATIONALIZATION_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the opera layer, e.g. *PROJECT_LAYERS*

OPERATIONALIZATION_ID: FK (***OPERATIONALIZATIONS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, OPERATIONALIZATION_ID

TABLE *OPERA_CONCEPTUAL (TABLE_ID: 710)*
This table stores the conceptual foundation of a conceptual object for harmonization

OPERA_CONCEPTUAL_ID	INT NOT NULL AUTO_INCREMENT
OPERATIONALIZATION_ID	INT
BASIS	INT
LIMITATION	INT

OPERA_CONCEPTUAL_ID: PK

OPERATIONALIZATION_ID: FK (***OPERATIONALIZATIONS***)

BASIS: FK (***TEXTUAL_OBJECTS***)
LIMITATION: FK (***TEXTUAL_OBJECTS***)

PK: OPERA_CONCEPTUAL _ID

NOTE:
Reference to the concept; q.v. *CONCEPT_REFERENCES.*

TABLE *OPERA_INDICATORS (TABLE_ID: 720)*
This table stores the required information of a classification or scale

OPERA_ INDICATOR_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	

OPERA_INDICATOR_ID: PK

DEFAULT_LABEL:

PK: OPERA_INDICATOR_ID

NOTE:
Reference to a recommended name-value pair; q.v. *PROPERTY_REFERENCES.*
Reference to a recommended variable; q.v. *VAR_REFERENCES.*
Reference to a recommended question; q.v. *QUEST_REFERENCES.*

TABLE *OPERA_IND_REFERENCES (TABLE_ID: 1720)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
OPERA_ INDICATOR_ID	INT NOT NULL
OPERA_IND_TYPE_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the opera indicator, e.g. *OPERATIONALIZATIONS, LAYER_NODES, CON_DIMENSIONS*

OPERA_ INDICATOR_ID: FK (***OPERA_INDICATORS***)

OPERA_IND_TYPE_ID: FK (***OPERA_IND_TYPES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, OPERA_INDICATOR_ID

TABLE *OPERA_IND_TYPES (TABLE_ID: 2720), (SUPPORT)*
This table stores the different types of indicators

OPERA_IND_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

OPERA_IND_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the indicator type

PK: OPERA_IND_TYPE_ID

NOTE:

OPERA_IND_TYPE_ID	DEFAULT_LABEL
0	“Source”
1	“Auxiliary”
2	“Target”

“Source”; Variable is used as SOURCE indicator
“Auxiliary” Variable is an AUXILIARY indicator
“Target”: Variable is the new TARGET indicator

TABLE *OPERA_MEA_VALUES (TABLE_ID: 730)*
This table stores abstract construct: measurement prescriptions / values

OPERA_MEA_VALUE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	

OPERA_ MEA_VALUE_ID: PK

DEFAULT_LABEL:

PK: OPERA_ MEA_VALUE_ID

TABLE *OPERA_MEA_VAL_REFERENCES (TABLE_ID: 1730)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
OPERA_ MEA_VALUE_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the measurement prescription / value, e.g. *OPERATIONALIZATIONS, LAYER_EDGES, CON_SPECIFICATIONS*

OPERA_ MEA_VALUE_ID: The reference to a measurement prescription / value – FK (***OPERA_MEA_VALUES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, OPERA_MEA_VALUE_ID

DATA CODING LAYER

TABLE *DATA_CODINGS (TABLE_ID: 740)*
This table stores the data coding

DATA_CODING_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

DATA_CODING_ID: PK

DEFAULT_LABEL:

PK: DATA_CODING_ID

TABLE *DATA_CODING_REFERENCES (TABLE_ID: 1740)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
DATA_CODING_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the opera layer, e.g. *PROJECT_LAYERS*

DATA_CODING_ID: FK (***DATA_CODINGS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, DATA_CODING_ID

TABLE *DC_VARIABLES (TABLE_ID: 750)*
This table stores the required information of an assigned variable

DC_ VARIABLE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	

DC_VARIABLE_ID: PK

DEFAULT_LABEL:

PK: DC_VARIABLE_ID

NOTE:
Reference to an assigned variable; q.v. *VAR_REFERENCES*.
Reference to an assigned question; q.v. *QUEST_REFERENCES*.

TABLE *DC_VAR_REFERENCES (TABLE_ID: 1750)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
DC_VARIABLE_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the dc variables, e.g. *DATA_CODINGS, LAYER_NODES, OPERA_INDICATORS*

DC_VARIABLE_ID: FK (***DC_VARIABLES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, DC_VARIABLE_ID

TABLE *DC_VAR_VALUES (TABLE_ID: 751)*
This table stores abstract construct: values

DC_VAR_VALUE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	

DC_VAR_VALUE_ID: PK

DEFAULT_LABEL:

PK: DC_VAR_VALUE_ID

NOTE:
DC_VARIABLES and DC_VAR_VALUES are constructed with OPERA_INDICATORS and OPERA_MEA_VALUES as a blue print.

TABLE *DC_VAR_VAL_REFERENCES (TABLE_ID: 1751)*

TABLE_ID	SMALLINT UNSIGNED NOT NULl
TABLE_ENTRY_ID	INT NOT NULL
DC_VAR_VALUE_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the dimension, e.g. *DATA_CODINGS*, *LAYER_NODES*, *OPERA_MEA_VALUES*

DC_VAR_VALUE_ID: The reference to a measurement prescription / Value – FK (***DC_VAR_VALUES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, DC_VAR_VALUE_ID

TABLE *CONVERSIONS (TABLE_ID: 760)*
This table stores the conversions of a project

CONVERSION_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

CONVERSION_ID: PK

PK: CONVERSION_ID

TABLE *CONV_REFERENCES (TABLE_ID: 1760)*

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
CONVERSION_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the conversion, e.g. *DATA_CODINGS*

CONVERSION_ID: FK (***CONVERSIONS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, CONVERSION_ID

TABLE *CONV_VARIABLES (TABLE_ID: 761)*
This table stores the different variables used in a conversion

CONVERSION_ID	INT NOT NULL
VARIABLE_ID	INT
CONV_VAR_TYPE_ID	INT

CONVERSION_ID: FK (***CONVERSIONS***)
VARIABLE_ID: FK (***VARIABLES***)

CONV_VAR_TYPE_ID: FK (***CONV_VAR_TYPES***) (default: NULL)

PK: CONVERSION_ID, CONV_VAR_TYPE_ID

TABLE *CONV_VAR_TYPES (TABLE_ID: 2761), (SUPPORT)*
This table stores the different types of variables

CONV_VAR_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

CONV_VAR_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the variable type

PK: CONV_VAR_TYPE_ID

NOTE:

CONV_VAR_TYPE_ID	DEFAULT_LABEL
0	"Source"
1	"Auxiliary"
2	"Target"

"Source"; Variable is used as SOURCE variable
"Auxiliary" Variable is an AUXILIARY variable
"Target": Variable is the new TARGET variable

TABLE *ROUTINES (TABLE_ID: 770)*
This table stores the different routines used to implement the conversion

ROUTINE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
CONVERSION_ID	INT
STAT_SYSTEM	VARCHAR (256)
STAT_SYSTEM_VERSION	VARCHAR (64)
INSTRUCTION_CODE	VARCHAR (2000)

ROUTINE_ID:	PK
DEFAULT_LABEL:	
CONVERSION_ID:	FK (***CONVERSIONS***)
STAT_SYSTEM:	Name of a statistically program
STAT_SYSTEM_VERSION:	Version of said program
INSTRUCTION_CODE:	The stored instruction text
PK:	ROUTINE_ID

VARIABLE

TABLE *VARIABLES (TABLE_ID: 800)*
This table stores the variables

VARIABLE_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT (‘1’)
LABEL	BOOLEAN DEFAULT (‘0’)
VARIABLE_DEFINITION	BOOLEAN DEFAULT (‘0’)
UNIVERSE_REFERENCE	BOOLEAN DEFAULT (‘1’)
CONCEPT_REFERENCE	INT
QUESTION_REFERENCE	BOOLEAN DEFAULT (‘1’)
EMBARGO_REFERENCE	INT
RESPONSE_UNIT	VARCHAR
ANALYSIS_UNIT	INT
REPRESENTATION	INT
IS_TEMPORAL	BOOLEAN DEFAULT(‘0’)
IS_GEOGRAPHIC	BOOLEAN DEFAULT(‘0’)
IS_WEIGHT	BOOLEAN DEFAULT(‘0’)

VARIABLE_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VERSIONABLES***)
IS_VERSIONABLE:

VERSION_RATIONALE: “FLAG” (***LABELS*** *via* ***$_REFS***)
VARIABLE_DEFINITION: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
UNIVERSE_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
CONCEPT_REFERENCE: FK (***CONCEPT_REFS***)
QUESTION_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
EMBARGO_REFERENCE: FK (***EMBARGO_REFS***)
RESPONSE_UNIT:
ANALYSIS_UNIT: FK (***CODE_VALUES***)
REPRESENTATION: FK (***REPRESENTATIONS***)

IS_TEMPORAL:
IS_GEOGRAPHIC:
IS_WEIGHT:

PK: VARIABLE_ID

NOTE:
This table implements the type *l:VariableType (DDI 3)*

TABLE *VARIABLE_REFS (TABLE_ID: 1800)*
This table stores

VARIABLE_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
VARIABLE	INT
REF_RELATIONTYPE_ID	INT

VARIABLE_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
VARIABLE: FK (***VARIABLES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: VARIABLE_REF_ID

TABLE *VARIABLE_SCHEMES (TABLE_ID: 801)*
This table stores the variable schemes

VARIABLE_SCHEME_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_MAINTAINABLE_TYPE	INT
IS_MAINTAINABLE	BOOLEAN DEFAULT (‘1’)
LABEL	BOOLEAN DEFAULT (‘0’)
DESCRIPTION	BOOLEAN DEFAULT (‘0’)
VARIABLE_SCHEME_REFERENCE	BOOLEAN DEFAULT (‘1’)
VARIABLE	BOOLEAN DEFAULT (‘0’)
VARIABLE_REFERENCE	BOOLEAN DEFAULT (‘1’)
VARIABLE_GROUP	BOOLEAN DEFAULT (‘0’)
VARIABLE_GROUP_REFERENCE	BOOLEAN DEFAULT (‘1’)

VARIABLE_SCHEME_ID: PK

ABSTRACT_MAINTAINABLE_TYPE: FK (***ABSTRACT_MAINTAINABLES***)
IS_MAINTAINABLE:

LABEL: “FLAG” (***LABELS*** *via* ***$_REFS***)
DESCRIPTION: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
VARIABLE_SCHEME_REFERENCE: “FLAG” (***SCHEME_REFERENCES*** *via* ***$_REFS***)
VARIABLE: “FLAG” (***VARIABLES*** *via* ***$_REFS***)
VARIABLE_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
VARIABLE_GROUP: “FLAG” (***VARIABLE_GROUPS*** *via* ***$_REFS***)
VARIABLE_GROUP_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)

PK: VARIABLE_SCHEME_ID

NOTE:
This table implements the type *l:VariableSchemeType (DDI 3)*

TABLE *VARIABLE_SCHEME_REFS (TABLE_ID: 1801)*
This table stores

VARIABLE_SCHEME_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
SCHEME_REFERENCE_TYPE	INT
VARIABLE_SCHEME_ID	INT
REF_RELATIONTYPE_ID	INT

VARIABLE_SCHEME_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

SCHEME_REFERENCE_TYPE: FK (***SCHEME_REFERENCES***)
VARIABLE_SCHEME_ID: FK (***VARIABLE_SCHEMES)***

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: VARIABLE_SCHEME_REF_ID

TABLE *VARIABLE_GROUPS (TABLE_ID: 810)*
This table stores the variable groups

VARIABLE_GROUP_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT (‘1’)
GROUP_TYPE	INT
LABEL	BOOLEAN DEFAULT (‘0’)
DEFINITION	BOOLEAN DEFAULT (‘0’)
UNIVERSE_REFERENCE	BOOLEAN DEFAULT (‘1’)
CONCEPT_REFERENCE	INT
VARIABLE_REFERENCE	BOOLEAN DEFAULT (‘1’)
VARIABLE_GROUP_REFERENCE	BOOLEAN DEFAULT (‘1’)

VARIABLE_GROUP_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VERSIONABLES***)
IS_VERSIONABLE:

GROUP_TYPE: FK (***CODE_VALUES***)
LABEL: “FLAG” (***LABELS*** *via* ***$_REFS***)
DEFINITION: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
UNIVERSE_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
CONCEPT_REFERENCE: FK (***$_REFS***)
VARIABLE_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
VARIABLE_GROUP_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)

PK: VARIABLE_GROUP_ID

NOTE:
This table implements the type *l:VariableGroupType (DDI 3)*

TABLE *VARIABLE_GROUP_REFS (TABLE_ID: 1810)*
This table stores

VARIABLE_GROUP_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
VARIABLE_GROUP	INT
REF_RELATIONTYPE_ID	INT

VARIABLE_GROUP_REF_ID: PK

TABLE_ID,
TABLE_ENTRY_ID: FK (***diff. tables***):

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
VARIABLE_GROUP: FK (***VARIABLE_GROUPS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: VARIABLE_GROUP_REF_ID

TABLE *VALUE_RANGES (TABLE_ID: 820)*
This table groups the values together

VALUE_RANGE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

VALUE_RANGE_ID: PK
DEFAULT_LABEL:

PK: VALUE_RANGE_ID

TABLE *VALUES (TABLE_ID: 821)*
This table stores the different values

VALUE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
DEFAULT_VALUE	VARCHAR (256)
VALID	BOOLEAN
FREQUENCY	INT
FREQ_WEIGHTED	INT
RELATIVE_FREQUENCY	FLOAT

VALUE_ID: PK

DEFAULT_LABEL:
DEFAULT_VALUE:
VALID:

FREQUENCY:
FREQ_WEIGHTED:
RELATIVE_FREQUENCY:

PK: VALUE_ID

TABLE *VALUE_REFERENCES (TABLE_ID: 1821)*
This table stores the relation between an object and a value

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
VALUE_ID	INT
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the value, e. g. a value range.

VALUE_ID: FK (***VALUES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, VALUE_ID

QUESTION

TABLE *QUESTIONNARIES (TABLE_ID: 840)*
This table stores the questionnaires

QUESTIONNAIRE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
ADMINISTRATIONTYPE_ID	INT
DATA_COLLECTIONTYPE_ID	INT

QUESTIONNAIRE_ID: PK
DEFAULT_LABEL: Denomination of the questionnaire, if available

ADMINISTRATIONTYPE_ID: FK (***ADMINISTRATIONTYPES***)
DATA_COLLECTIONTYPE_ID: FK (***DATA_COLLECTIONTYPES***)

PK: QUESTIONNAIRE_ID

TABLE *QUEST_N_REFERENCES (TABLE_ID: 1840)*
This table stores the relation between an object and a questionnaire

TABLE_ID	SMALLINT UNSIGNED NOT NULl
TABLE_ENTRY_ID	INT NOT NULL
QUESTIONNAIRE_ID	INT NOT NULL
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the QUESTIONNAIRE

QUESTIONNAIRE_ID: FK (***QUESTIONNAIRES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID; QUESTIONNAIRE_ID

TABLE *ADMINISTRATIONTYPES (TABLE_ID: 3840), (SUPPORT)*

ADMINISTRATIONTYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

ADMINISTRATIONTYPE_ID: PK
DEFAULT_LABEL: Denomination of the administration type

PK: ADMINISTRATIONTYPE_ID

NOTE:

ADMINISTRATIONTYPE_ID	DEFAULT_LABEL
0	"Interview"
1	"Self-administrated"

TABLE *DATA_COLLECTIONTYPES (TABLE_ID: 4840), (SUPPORT) - (Instrument.type DDI3)*

DATA_COLLECTIONTYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

DATA_COLLECTIONTYPE_ID: PK
DEFAULT_LABEL: Denomination of the data collection type

PK: DATA_COLLECTIONTYPE_ID

NOTE: "PAPI: paper and pencil", "CAPI: computer assisted", "CAPI: computer assisted (self) interview", "CATI: audio assisted / telephone", "ACASI: audio assisted computer self interview"

TABLE *QUESTION_ITEMS (TABLE_ID: 850)*
This table stores the question items

QUESTION_ITEM_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT (‘1’)
QUESTION_TEXT	BOOLEAN DEFAULT (‘0’)
QUESTION_INTENT	BOOLEAN DEFAULT (‘0’)
RESPONSE_DOMAIN	INT
STRUCTURED_MIXED_RESPONSE_DOMAIN	INT
CONCEPT_REFERENCE	BOOLEAN DEFAULT (‘1’)
EXTERNAL_AID	BOOLEAN DEFAULT (‘0’)

QUESTION_ITEM_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VERSIONABLES***)
IS_MAINTAINABLE:

QUESTION_TEXT: “FLAG” (***DYNAMIC_TEXTS*** *via* ***$_REFS***)
QUESTION_INTENT: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
RESPONSE_DOMAIN: FK (***R_REPRESENTATIONS***)
STRUCTURED_MIXED_RESPONSE_DOMAIN: FK (***STRUCTURED_MIXED_RESPONSE_DOMAINS***)
CONCEPT_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
EXTERNAL_AID: “FLAG” (***OTHER_MATERIALS*** *via* ***$_REFS***)

PK: QUESTION_ITEM_ID

NOTE:
This table implements the type *dc:QuestionItemType (DDI 3)*

TABLE *QUESTION_ITEM_REFS (TABLE_ID: 1850)*
This table stores

QUESTION_ITEM_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
QUESTION_ITEM	INT
REF_RELATIONTYPE_ID	INT

QUESTION_ITEM_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
VARIABLE: FK (***QUESTION_ITEMS***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: QUESTION_ITEM_REF_ID

TABLE *QUESTION_REFS (TABLE_ID: -)*
This table stores the relation between an object and a question (Q.Item , Multipe Q. Item)

QUESTION_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
QUESTION	INT
REF_RELATIONTYPE_ID	INT

QUESTION_REF_ID:	PK
TABLE_ID, TABLE_ENTRY_ID:	FK (***diff. tables***): Reference to an OBJECT, recipient of the linked QUESTION, e. g. a question group or a questionnaire.
ELEMENT_NAME:	<TAG>-Denomination in DDI3
REFERENCE_TYPE: QUESTION:	FK (***REFERENCES***) FK (***QUESTION_ITEMS***)
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONSHIP***)
PK:	QUESTION_REF_ID

NOTE:
~~This table implements the element *QuestionReference* from *l:VariableType*~~

TABLE *MULTIPLE_QUESTION_ITEMS (TABLE_ID: 855)*
This table stores the multiple question items

MULTIPLE_QUESTION_ITEM_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_VERSIONABLE_TYPE	INT
IS_VERSIONABLE	BOOLEAN DEFAULT (‘1’)
QUESTION_TEXT	BOOLEAN DEFAULT (‘0’)
QUESTION_INTENT	BOOLEAN DEFAULT (‘0’)
CONCEPT_REFERENCE	BOOLEAN DEFAULT (‘1’)
EXTERNAL_AID	BOOLEAN DEFAULT (‘0’)
SUB_QUESTION_SEQUENCE	INT
SUB_QUESTIONS	INT

MULTIPLE_QUESTION_ITEM_ID: PK

ABSTRACT_VERSIONABLE_TYPE: FK (***ABSTRACT_VERSIONABLES***)
IS_MAINTAINABLE:

QUESTION_TEXT: “FLAG” (***DYNAMIC_TEXTS*** *via* ***$_REFS***)
QUESTION_INTENT: “FLAG” (***STRUCTURED_STRINGS*** *via* ***$_REFS***)
CONCEPT_REFERENCE: “FLAG” (***REFERENCES*** *via* ***$_REFS***)
EXTERNAL_AID: “FLAG” (***OTHER_MATERIALS*** *via* ***$_REFS***)
SUB_QUESTION_SEQUENCE: FK (***SPECIFIC_SEQUENCES***)
SUB_QUESTIONS: FK (***QUESTION_GROUPS***)

PK: MULTIPLE_QUESTION_ITEM_ID

NOTE:
This table implements the type *dc:MultipleQuestionItemType (DDI 3)*

TABLE *MULTIPLE_QUESTION_ITEM_REFS (TABLE_ID: 1855)*
This table stores

MULTIPLE_QUESTION_ITEM_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
REFERENCE_TYPE	INT
MULTIPLE_QUESTION_ITEM	INT
REF_RELATIONTYPE_ID	INT

MULTIPLE_QUESTION_ITEM_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

REFERENCE_TYPE: FK (***REFERENCES***)
MULTIPLE_QUESTION_ITEM: FK (***VARIABLES***)

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: MULTIPLE_QUESTION_ITEM_REF_ID

TABLE *QUESTION_SCHEMES (TABLE_ID: 851)*
This table stores the question schemes

QUESTION_SCHEME_ID	INT NOT NULL AUTO_INCREMENT
ABSTRACT_MAINTAINABLE_TYPE	INT
IS_MAINTAINABLE	BOOLEAN DEFAULT ('1')
LABEL	BOOLEAN DEFAULT ('0')
DESCRIPTION	BOOLEAN DEFAULT ('0')
QUESTION_SCHEME_REFERENCE	BOOLEAN DEFAULT ('1')
QUESTION_ITEM	BOOLEAN DEFAULT ('0')
MULTIPLE_QUESTION_ITEM	BOOLEAN DEFAULT ('0')

QUESTION_SCHEME_ID: PK

ABSTRACT_MAINTAINABLE_TYPE: FK (***ABSTRACT_MAINTAINABLES***)
IS_MAINTAINABLE:

LABEL: "FLAG" (***LABELS** via **$_REFS***)
DESCRIPTION: "FLAG" (***STRUCTURED_STRINGS** via **$_REFS***)
QUESTION_SCHEME_REFERENCE: "FLAG" (***SCHEME_REFERENCES** via **$_REFS***)
QUESTION_ITEM: "FLAG" (***QUESTION_ITEMS** via **$_REFS***)
MULTIPLE_QUESTION_ITEM: "FLAG" (***MULTIPLE_QUESTION_ITEMS** via **$_REFS***)

PK: QUESTION_SCHEME_ID

NOTE:
This table implements the type *dc:QuestionSchemeType (DDI 3)*

TABLE *QUESTION_SCHEME_REFS (TABLE_ID: 1851)*
This table stores

QUESTION_SCHEME_REF_ID	INT NOT NULL AUTO_INCREMENT
TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ELEMENT_NAME	VARCHAR NOT NULL
SCHEME_REFERENCE_TYPE	INT
QUESTION_SCHEME_ID	INT
REF_RELATIONTYPE_ID	INT

QUESTION_SCHEME_REF_ID: PK

TABLE_ID, FK (***diff. tables***)
TABLE_ENTRY_ID:

ELEMENT_NAME: <TAG>-Denomination in DDI3

SCHEME_REFERENCE_TYPE: FK (***SCHEME_REFERENCES***)
QUESTION_SCHEME_ID: FK (***QUESTION_SCHEMES)***

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: QUESTION_SCHEME_REF_ID

TABLE *QUESTION_GROUPS (TABLE_ID: 860), (DDI3: dc:QuestionGroupType)*
This table stores

QUESTION_GROUP_ID	INT NOT NULL AUTO_INCREMENT
QUESTION_ITEM	BOOLEAN DEFAULT ('0')
MULTIPLE_QUESTION_ITEM	BOOLEAN DEFAULT ('0')

QUESTION_GROUP_ID: PK

QUESTION_ITEM: "FLAG" (***QUESTION_ITEMS*** via ***$_REF***)
MULTIPLE_QUESTION_ITEM: "FLAG" (***MULTIPLE_QUESTION_ITEMS*** via ***$_REF***)

PK: QUESTION_GROUP_ID

TABLE *QUEST_GROUP_REFERENCES (TABLE_ID: 1860)*
This table stores the relation between an object and a question group

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
QUEST_GROUP_ID	INT
ORDER_NUM	INT
ORDER_TXT	VARCHAR (64)
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the QUESTION GROUP, e. g. a question group or a questionnaire.

QUESTION_GROUP_ID: FK (***QUESTION_GROUPS***)

ORDER_NUM: The order of this question group compared to other.
ORDER_TXT:

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, QUESTION_GROUP_ID

TABLE *SPECIFIC_SEQUENCES (TABLE_ID: NA), (DDI3: dc:SpecificSequenceType)*
This table stores

SPECIFIC_SEQUENCE_ID	INT NOT NULL
QUESTION_SEQUENCE	INT
ALTERNATE_SEQUENCE	INT

SPECIFIC_SEQUENCE_ID: PK

QUESTION_SEQUENCE: FK (***QUESTION_SEQUENCE_TYPES***)
ALTERNATE_SEQUENCE: FK (***COMMANDS***)

PK: SPECIFIC_SEQUENCE_ID

TABLE *QUESTION_SEQUENCE_TYPES (TABLE_ID: -), (DDI3: dc:QuestionSequenceTypeType)*

QUESTION_SEQUENCE_TYPE_ID	INT NOT NULL
ENUMERATION_VALUE	VARCHAR (20)

QUESTION_SEQUENCE_TYPE_ID: PK
ENUMERATION_VALUE:

PK: QUESTION_SEQUENCE_TYPE_ID

NOTE:

QUESTION_SEQUENCE_TYPE_ID	ENUMERATION_VALUE
1	"IN_ORDER_OF_APPEARANCE"
2	"RANDOM"
3	"ROTATE"
4	"OTHER"

TABLE *ITEMS (TABLE_ID: 870)*
This table stores question items

ITEM_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_TEXT	VARCHAR (256)
DEFAULT_INSTRUCTION	VARCHAR (256)
ANSWER_CATEGORY_SET_ID	INT
TABLE_ID_REPRESENTATION	SMALLINT UNSIGNED
TABLE_ENTRY_ID_REPRESENTATION	INT

ITEM_ID:	PK
DEFAULT_TEXT:	Item text
DEFAULT_INSTRUCTION:	Instruction for the interviewer
ANSWER_CATEGORY_SET_ID:	FK (***ANSWER_CATEGORY_SETS***)
TABLE_ID_REPRESENTATION, TABLE_ENTRY_ID_REPRESENTATION:	Reference to the type representation
PK:	ITEM_ID

TABLE *ITEM_REFERENCES (TABLE_ID: 1870)*
This table stores the relation between an object and an item

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ITEM_ID	INT
ORDER_NUM	INT
ORDER_TXT	VARCHAR (64)
REF_RELATIONTYPE_ID	INT

TABLE_ID, TABLE_ENTRY_ID:	The reference to an OBJECT, recipient of the ITEM, e. g. a question.
ITEM_ID:	FK (***QUESTION_GROUPS***)
ORDER_NUM: ORDER_TXT:	The order of this item compared to other.
REF_RELATIONTYPE_ID:	FK (***REF_RELATIONSHIP***)
PK:	TABLE_ID, TABLE_ENTRY_ID, ITEM_ID

TABLE *ITEM_FLOWCONTROL (TABLE_ID: 875)*
This table stores control flow redirections caused by the chosen answer category

ITEM_FLOWCONTROL_ID	INT NOT NULL AUTO_INCREMENT
ITEM_ID	INT
ANSWER_CATEGORIE_ID	INT
TABLE_ID	SMALLINT UNSIGNED
TABLE_ENTRY_ID	INT

ITEM_FLOWCONTROL_ID:	PK
ITEM_ID:	FK (***ITEMS***)
ANSWER_CATEGORY_ID:	FK (***ANSWER_CATEGORIES***)
TABLE_ID, TABLE_ENTRY_ID:	Reference to the next ITEM or QUESTION
PK:	ITEM_FLOWCONTROL_ID

TABLE *SURVEY_SUPPORTING_ITEMS (TABLE_ID: 830)*
This table stores the survey item components

SURVEY_ITEM_ID	INT NOT NULL AUTO_INCREMENT
SUR_SUP_I_TYPE_ID	INT

SURVEY_ITEM_ID: PK

SUR_SUP_I_TYPE_ID: FK (***SUR_SUP_I_TYPES***)

PK: SURVEY_ITEM_ID

NOTE:
The textual part is stored in *TEXTUAL_OBJECTS* and accessed with the SURVEY_ITEM_ID.

TABLE *SUR_SUP_I_TYPES (TABLE_ID: 2830), (SUPPORT)*

SUR_SUP_I_TYPE_ID	INT NOT NULL
DEFAULT_LABEL	VARCHAR (256)

SUR_SUP_I_TYPE_ID: PK
DEFAULT_LABEL: Denomination of the class, e.g. question type

PK: SUR_SUP_I_TYPE_ID

NOTE:
'Introduction','Motivation','Info_Regard_Cont','Info_Regard_Def','Instruct_Respondent','Instruct_Interviewer','Stimulus' etc

TABLE *SUR_SUP_I_REFERENCES (TABLE_ID: 1830)*
This table stores the relation between an object and a survey item

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
SURVEY_ITEM_ID	INT NOT NULL
ORDER_NUM	INT
ORDER_TXT	VARCHAR (64)
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the SURVEY ITEM, e. g. a question, question group, item etc

SURVEY_ITEM_ID: FK (***SURVEY_ITEMS***)

ORDER_NUM: The order of this survey supporting item compared to other.
ORDER_TXT:

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID; SURVEY_ITEM_ID

TABLE *ANSWER_CATEGORIE_SETS (TABLE_ID: 880)*
This table stores answer categories

ANSWER_CATEGORY_SET_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

ANSWER_CATEGORY_SET_ID: PK
DEFAULT_LABEL:

PK: ANSWER_CATEGORY_SET_ID

TABLE *ANSWER_CATEGORIES (TABLE_ID: 881)*
This table stores answer vales, groups them with answer category sets

ANSWER_ CATEGORIE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)
DEFAULT_VALUE	VARCHAR (256)

ANSWER_CATEGORIE_ID: PK

DEFAULT_LABEL:
DEFAULT_VALUE:

PK: ANSWER_CATEGORIE_ID

TABLE *ANSWER_CATEGORIE_REFERENCES (TABLE_ID: 1881)*
This table stores the relation between an object and an answer value

TABLE_ID	SMALLINT UNSIGNED NOT NULL
TABLE_ENTRY_ID	INT NOT NULL
ANSWER_CATEGORIE_ID	INT
ORDER_NUM	INT
ORDER_TXT	VARCHAR (64)
REF_RELATIONTYPE_ID	INT

TABLE_ID,
TABLE_ENTRY_ID: The reference to an OBJECT, recipient of the answer category, e. g. an answer c. set.

ANSWER_CATEGORIE_ID: FK (***ANSWER_CATEGORIES***)

ORDER_NUM: The order of this answer category compared to other.
ORDER_TXT:

REF_RELATIONTYPE_ID: FK (***REF_RELATIONSHIP***)

PK: TABLE_ID, TABLE_ENTRY_ID, ANSWER_CATEGORIE_ID

DATA DEFINING

TABLE *R_REPRESENTATIONS (TABLE_ID: -)*
This table stores

R_REPRESENTATION_ID	INT NOT NULL AUTO_INCREMENT
RECOMMENDED_DATA_TYPE	INT
GENERIC_OUTPUT_FORMAT	INT
MISSING_VALUE	
BLANK_IS_MISSING_VALUE	BOOLEAN
CLASSIFICATION_LEVEL	

R_REPRESENTATION_ID: PK

RECOMMENDED_DATA_TYPE: FK (***CODE_VALUES***)
GENERIC_OUTPUT_FORMAT: FK (***CODE_VALUES***)

MISSING_VALUE: FK (***NM_TOKENSS***)
BLANK_IS_MISSING_VALUE:
CLASSIFICATION_LEVEL: FK (***CATEGORY_RELATION_CODES***)

PK: R_REPRESENTATION_ID

TABLE *REPRESENTATIONS (TABLE_ID: -)*
This table stores

REPRESENTATION_ID	INT NOT NULL AUTO_INCREMENT
ROLE	VARCHAR
WEIGHT_VARIABLE_REFERENCE	BOOLEAN DEFAULT ('1')
STANDARD_WEIGHT_REFERENCE	INT
IMPUTATION_REFERENCE	INT
CONCATENATED_VALUE	INT
CODING_INSTRUCTION_REFERENCE	INT
VALUE_REPRESENTATION	INT NOT NULL
MEASUREMENT_UNIT	VARCHAR
AGGREGATION_METHOD	INT
ADDITIVITY	INT

REPRESENTATION_ID: PK

ROLE:
WEIGHT_VARIABLE_REFERENCE: "FLAG" (***REFERENCES** via **WEIGHT_VARIABLE_REFS***)
STANDARD_WEIGHT_REFERENCE: FK (***STANDARD_WEIGHT_REFS***)
IMPUTATION_REFERENCE: FK (***IMPUTATION_REFS***)
CONCATENATED_VALUE: FK (***CONCATENATED_VALUES***)
CODING_INSTRUCTION_REFERENCE: FK (***CODING_INSTRUCTION_REFS***)
VALUE_REPRESENTATION: FK (***R_REPRESENTATIONS***)

MEASUREMENT_UNIT:
AGGREGATION_METHOD: FK (***AGGREGATION_METHOD_CODES***)
ADDITIVITY: FK (***ADDITIVITY_CODES***)

PK: REPRESENTATION_ID

TABLE *CCATS_REPRESENTATIONS (TABLE_ID: 900)*
This table stores the data defining representations

REPRESENTATION_ID	INT NOT NULL AUTO_INCREMENT
CATEGORY_RELATION_CODE_ID	INT
REPR_TABLE_ID	SMALLINT UNSIGNED
REPR_TABLE_ENTRY_ID	INT
OBJECT_TABLE_ID	SMALLINT UNSIGNED
OBJECT_TABLE_ENTRY_ID	INT

REPRESENTATION_ID: PK

CATEGORY_RELATION_CODE_ID: FK (***CATEGORY_RELATION_CODES***)

REPR_TABLE_ID,
REPR_TABLE_ENTRY_ID: Reference to the representation type which is stored

OBJECT_TABLE_ID,
OBJECT_TABLE_ENTRY_ID: Reference to the object whose representation is stored

PK: REPRESENTATION_ID

TABLE *TEXT_REPRESENTATIONS (TABLE_ID: 910), (DDI 3 compatibility exists)*
This table stores text representations

TEXT_REPRESENTATION_ID	INT NOT NULL AUTO_INCREMENT
MAX_LENGTH	INT
MIN_LENGTH	INT
REG_EXPR	VARCHAR (256)

TEXT_REPRESENTATION_ID:

MAX_LENGTH:
MIN_LENGTH:
REG_EXP: e. g. “ab+c”

PK: TEXT_REPRESENTATION_ID

NOTE:
q.v. *r:TextRepresentationType (DDI 3)*

TABLE *DATETIME_REPRESENTATIONS (TABLE_ID: 920), (DDI 3 compatibility possible)*
This table stores date time representations

DATE_TIME_REPRESENTATION_ID	INT NOT NULL AUTO_INCREMENT
DATE_TYPE_CODE_ID	INT
FORMAT	VARCHAR (256)

DATE_TIME_REPRESENTATION_ID:

DATE_TYPE_CODE_ID:
FORMAT: e. g. “YYYY-MM-DD HH:MM:SS”

PK: DATE_TIME_REPRESENTATION_ID

NOTE:
q.v. *r:DateTimeRepresentationType (DDI 3)*

TABLE *NUMERIC_REPRESENTATIONS (TABLE_ID: 930), (DDI 3 compatibility possible)*
This table stores numeric representations

NUMERIC_REPRESENTATION_ID	INT NOT NULL AUTO_INCREMENT
NUMBER_FORMAT	VARCHAR (256)
NUMERIC_TYPE_CODE_ID	INT
SCALE	INT
DECIMAL_POSITIONS	INT

NUMERIC_REPRESENTATION_ID:

NUMBER_FORMAT: e. g. "1.11.53.1023.2"

NUMERIC_TYPE_CODE_ID:
SCALE:
DECIMAL_POSTIONS:

PK: NUMERIC_REPRESENTATION_ID

NOTE:
q.v. *r:NumericRepresentationType (DDI 3)*

TABLE *BOOLEAN_REPRESENTATIONS (TABLE_ID: 940)*
This table stores boolean representations

BOOLEAN_REPRESENTATION_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

BOOLEAN_REPRESENTATION_ID:
DEFAULT_LABEL:

PK: BOOLEAN_REPRESENTATION_ID

TABLE *INCREMENTALS (TABLE_ID: 933), (DDI 3 compatibility possible)*
This table stores numeric representations

INCREMENTAL_ID	INT NOT NULL AUTO_INCREMENT
NUMERIC_REPRESENTATION_ID	INT
START_VALUE	FLOAT
END_VALUE	FLOAT
INTERVAL	FLOAT

INCREMENTAL_ID:

NUMERIC_REPRESENTATION_ID: FK (*NUMERIC_REPRESENTATIONS*)

START_VALUE:
END_VALUE:
INTERVAL:

PK: INCREMENTAL_ID

NOTE:
q.v. *r:NumericRepresentationType (DDI 3)*

TABLE *NUMBER_RANGES (TABLE_ID: 936), (DDI 3 compatibility possible)*
This table stores number ranges

NUMBER_RANGE_ID	INT NOT NULL AUTO_INCREMENT
NUMERIC_REPRESENTATION_ID	INT
BOUND_FORMAT	VARCHAR (256)
LOW_VALUE	DECIMAL
LOW_INCLUDED	BOOLEAN
HIGH_VALUE	DECIMAL
HIGH_INCLUDED	BOOLEAN
TOP_CODE	INT
BOTTOM_CODE	INT
REG_EXPR	VARCHAR (256)

NUMBER_RANGE_ID:

NUMERIC_REPRESENTATION_ID: FK (***NUMERIC_REPRESENTATIONS***)

BOUND_FORMAT: e.g. “[8.5,99.0[“

LOW_VALUE: e.g. “8.5”
LOW_BOOLEAN: e.g. “TRUE”
HIGH_VALUE: e.g. “99.0”
LOW_BOOLEAN: e.g. “FALSE”
TOP_CODE:
BOTTOM_CODE:
REG_EXPR:

PK: NUMBER_RANGE_ID

NOTE:
q.v. *r:NumberRangeType, r:NumberRangeValueType (DDI 3)*

TABLE *NUMERIC_TYPE_CODES (TABLE_ID: 3930), (SUPPORT)*
This table stores the different numeric types a numeric representation may belong to

NUMERIC_TYPE_CODE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

NUMERIC_TYPE_CODE_ID: PK
DEFAULT_LABEL: Denomination of the numeric type code

PK: NUMERIC_TYPE_CODE_ID

NOTE:
"BigInteger", "Integer", "Long", "Short", "Decimal", "Float", "Double", "Count", "Incremental"

q.v. *r:NumericTypeCodeType (DDI 3)*

TABLE *CATEGORY_RELATION_CODES (TABLE_ID: 3900), (SUPPORT)*
This table stores the different category relation codes a representation may belong to

CATEGORY_RELATION_CODE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

CATEGORY_RELATION_CODE_ID: PK
DEFAULT_LABEL: Denomination of the category relation code

PK: CATEGORY_RELATION_CODE_ID

NOTE:
"Nominal", "Ordinal", "Interval", "Ratio", "Continuous"

q.v. *r:CategoryRelationCodeType (DDI 3)*

TABLE *DATE_TYPE_CODES (TABLE_ID: 3920), (SUPPORT)*
This table stores the different date type codes a date type representation may belong to

DATE_TYPE_CODE_ID	INT NOT NULL AUTO_INCREMENT
DEFAULT_LABEL	VARCHAR (256)

DATE_TYPE_CODE_ID: PK
DEFAULT_LABEL: Denomination of the category relation code

PK: DATE_TYPE_CODE_ID

NOTE:
"DateTime", "Date", "Time", "Year", "Month", “Day”, “MonthDay”, “YearMonth”, “Duration”, “Timespan”

q.v. *r:DateTypeCodeType (DDI 3)*