# eSaude – Data Migration Tool

## Matching File Description

### Change Log

| # | Author | Date | Description |
|---|---|---|---|
| 1 | Valério João | 13-10-2014 | Basic document |

## Tuple Sheet

Answers the question *"what to migrate?"* Each entry represent records (tuples) of a specific OpenMRS table. Tuples are based on global concept and can be related to other concepts.

Tuple contains the columns listed and described in Table 1Tuple Sheet table columns.

| COLUMN | DESCRIPTION |
|---|---|
| **ID** | Is the unique identifier of the tuple, it's specific of DMT internal processes and given by the user. |
| **TERMINOLOGY** | Is the concept name that the tuple is based on, it is formulated in a way that the user can best understand. It has no use for the DMT. |
| **TABLE** | Indicate the OpenMRS table where the tuple goes into |
| **PREDECESSOR** | Indicate the sequence of tuple execution based on OpenMRS database relationships. It points to the ID of the tuple that must be processed before the current one, or N/A if the tuple is the first in the execution sequence (root). |
| **DESC** | Anything else that can be used to provide more understanding about the tuple. It has no use for the DMT |

*Table 1Tuple Sheet table columns*

### Insertion rules

While inserting data into tuple sheet, the following rules must be respected:

1. ID must not be empty
2. ID must be a number and unique.
3. ID must not be 0 (zero)
4. TABLE must not be empty
5. TABLE must be a valid OpenMRS database table
6. PREDECESSOR must not be empty
7. PREDECESSOR must point to an existing tuple ID or N/A if *root* tuple
8. PREDECESSOR, there is only one *root* tuple (the first tuple in the execution sequence), it must have N/A value.

### Insertion guidelines

In order to provide easy comprehension by human readers, consider the following guidelines:

1. For ID column use positive numbers in ascending order
2. Use a TERMINOLOGY name in the paper form or more approximated to it.
3. Use upper case letters for TABLE names
4. Provide a comprehensive description about the tuple.

## Match-L-to-R Sheet

Answers the question *"how to migrate from left to right?"* It contains the information of the two databases involved in the migration, where the left side is always the target database (OpenMRS) and right side is the source database. This sheet contains the matching/mapping between target and source. Matches put together the concepts of target and source databases and provide technical information that the DMT engine requires to process the matches. Each match is a column of a TABLE in **Tuple Sheet**, excluding **foreign key (FK)** columns.

Match-L-to-R contains the columns listed and described in Table 2 Match-L-to-R table columns.

| COLUMN | DESCRIPTION |
|---|---|
| **TUPLE** | Is the ID of the Tuple whose the match belongs to |
| **TERMINOLOGY** | Is the concept name that the match represents, it is formulated in a way that the user best comprehends. It has no use for the DMT. |
| **LEFT:COLUMN** | Is a valid OpenMRS column that belongs to the TABLE indicated by the tuple whose the match belongs to |
| **LEFT:DATATYPE** | Indicate the datatype of the column in target database |
| **LEFT:SIZE** | Indicate the size of the column in target database |
| **LEFT:REQUIRED?** | Indicate whether or not the column in target database can take a NULL value |
| **LEFT:PK** | Indicate whether or not the column is a PK in target database |
| **RIGHT:TABLE** | Indicate the table in source database that contains the column whose concept matches the target one. |
| **RIGHT:COLUMN** | Indicate the column in the source database whose concept matches the target one. |
| **RIGHT:DATATYPE** | Indicate the datatype of column in source database |
| **RIGHT:SIZE** | Indicate the size of column in source database |
| **RIGHT:REQUIRED?** | Indicate whether or not the column in source database can take a NULL value |
| **MATCH ID** | Is a unique identifier of the match, it is specific of DMT internal processes and given by the user. |
| **OBSERVATION** | Is any technical description that provides more understanding about the match to the user. It has no use for the DMT. |

| | |
|---|---|
| **DEFAULT VALUE** | Is the value that must be used by the DMT in case the match doesn't have a source side or the source side is not required |
| **VALUE MATCH** | Indicates the ID of the Value Match in which the match must comply with |

*Table 2 Match-L-to-R table columns*

## Insertion Rules

While inserting data into match-l-to-r sheet, the following rules must be respected:

1. TUPLE must not be empty
2. TUPLE must be an existing ID value of **Tuple sheet**
3. Matches of the same tuple must be grouped together
4. LEFT:COLUMN must not be empty
5. LEFT:COLUMN is mandatory, it cannot be N/A
6. LEFT:DATATYPE must not be empty
7. LEFT:DATATYPE is mandatory, it cannot be N/A
8. LEFT:SIZE must not be empty
9. LEFT:SIZE is mandatory, it cannot be N/A
10. LEFT:REQUIRED is mandatory, it must only take the value YES or NO
11. LEFT:PK is mandatory, it must only take the value YES or NO
12. RIGHT:TABLE must not be empty
13. RIGHT:TABLE is not mandatory, it can be N/A
14. RIGHT:COLUMN must not be empty
15. RIGHT:COLUMN must not be N/A, unless RIGHT:TABLE is N/A
16. RIGHT:DATATYPE must not be empty
17. RIGHT:DATATYPE must not be N/A, unless RIGHT:TABLE is N/A
18. RIGHT:SIZE must not be empty
19. RIGHT:SIZE must not be N/A, unless RIGHT:TABLE is N/A
20. RIGHT:REQUIRED must not be empty
21. RIGHT:REQUIRED must take the values YES/NO, or N/A if RIGHT:TABLE is N/A
22. MATCH ID must not be empty
23. MATCH ID must be a number and unique.
24. MATCH ID must not be 0 (zero)
25. DEFAULT VALUE must not be empty
26. DEFAULT VALUE can be any value including pre-defined constants (AI, NULL, N/A, TOP, SKIP, AI/SKIP/TRUE and AI/SKIP/FALSE)
27. VALUE MATCH must not be empty
28. VALUE MATCH must be the ID of Value Match sheet, or N/A.

## Insertion guidelines

In order to provide easy comprehension by human readers, consider the following guidelines:

1. For Match ID column use positive numbers in ascending order
2. Use a TERMINOLOGY name in the paper form or more approximated to it.

3. Use upper case letters for Table, Datatype, Required, PK and Default Value constant names
4. Use lower case letters for Column names
5. Provide a comprehensive technical observation of the matches.

## L-References Sheet

This sheet is named L-References (Left References) because it contains relationship references of left side database, indicates the foreign key columns of tuples, the referenced side and the value that the FK column must take. A L-Reference can be of two types, namely:

**Direct Reference –** directly references a tuple TABLE, in other words it contains a column (REFERENCE:COLUMN) that is a FK in the tuple that the reference belongs to. Direct reference points directly to the FK value or indirectly through indirect references.

**Indirect Reference –** indirectly references a tuple TABLE, it's only related to the tuple TABLE through a direct reference that it follows. A good example for an indirect reference is the existence of a link table between the tuple TABLE and the table that provides the FK value.

A matching query is built from the RIGHT:TABLE and RIGHT:COLUMN of the match, unless these values are not available (N/A) then the query is completely built from REFERENCED TABLE and REFERENCED COLUMN of l-references of the match.

L-References contain the columns listed and described in Table 3 L-References table columns.

| COLUMN | DESCRIPTION |
|---|---|
| **ID** | Is an unique identifier of the reference, it is specific of DMT internal processes and given by the user. |
| **TUPLE ID** | Is the ID of the Tuple whose the reference belongs to |
| **REFERENCE TABLE** | Is the owner table of the relationship, it contains the FK. This table indicate whether the reference is direct or indirect. |
| **REFERENCE COLUMN** | Is the FK column of the relationship |
| **REFERENCED TABLE** | Is the contained side of the relationship, it contains the PK. |
| **REFERENCED COLUMN** | Is the PK column of the relationship |
| **DATATYPE** | Indicate the datatype of the relationship |
| **SIZE** | Indicate the size of the relationship. In case the PK and FK don't share the same size, the greater must be used. |
| **REFERENCED VALUE** | Is the value that the REFERENCE COLUMN must take |
| **SEQUENCE** | Indicate the sequence of references composition for a tuple. If the sequence is N/A, then the reference is direct. |

| | |
|---|---|
| | For indirect references, it is equal to the ID of the previous reference in the sequence. |
| **NAME/DESC** | Anything else that can be used to provide more understanding of the reference. It has no use for the DMT |

*Table 3 L-References table columns*

### Insertion Rules

While inserting data into l-references, the following rules must be respected:

1. ID must not be empty
2. ID must be a number and unique
3. TUPLE ID must not be empty
4. TUPLE ID must be an existing ID value of Tuple sheet
5. REFERENCE TABLE must not be empty
6. REFERENCE TABLE must be a valid OpenMRS table
7. REFERENCE COLUMN must not be empty
8. REFERENCE COLUMN must be a valid OpenMRS database FK column
9. REFERENCED TABLE must not be empty
10. REFERENCED TABLE must be a valid OpenMRS table
11. REFERENCED COLUMN must not be empty
12. REFERENCED COLUMN must be a valid OpenMRS database PK column
13. DATATYPE must not be empty
14. DATATYPE must be valid for OpenMRS database
15. SIZE must not be empty
16. SIZE must be the greater of REFERENCE COLUMN and REFERENCED COLUMN
17. REFERENCED VALUE cannot be empty
18. REFERENCED VALUE can be any value including pre-defined constants (TOP, EQUALS and NULL).

### Insertion guidelines

In order to provide easy comprehension by human readers, consider the following guidelines:

1. For ID column use positive numbers in ascending order
2. Use upper case letters for table, datatype and constant names
3. Use lower case letters for column names
4. Provide a comprehensive name/description about the references

## R-References Sheet

This sheet is named R-References (Right References) because it contains information of *how to retrieve data from left side database*, it indicate how to get the CURRS of a tuple and the conditions to retrieve data to be used in a particular matching. A R-Reference can be of two types, namely:

**Direct Reference** – directly references a match TABLE, the REFERENCE TABLE is N/A or equal to the TABLE of the Match-L-to-R that owns the reference. Direct reference points directly to the match value or indirectly through indirect references.

**Indirect Reference** – indirectly references a match TABLE, it's only related to the match TABLE through a direct reference that it follows. A good example of an indirect reference is the existence of a link table between the match TABLE and the table that provide the match value.

R-References contain the columns listed and described in Table 4 R-References table columns.

| COLUMN | DESCRIPTION |
|---|---|
| **ID** | Is a unique identifier of the reference, it is specific of DMT internal processes and given by the user. |
| **MATCH ID** | Is the ID of the Match-L-to-R whose the reference belongs to |
| **REFERENCE TABLE** | Is the owner table of the relationship, it contains the FK. This table indicate whether the reference is direct or indirect. |
| **REFERENCE COLUMN** | Is the FK column of the relationship |
| **REFERENCED TABLE** | Is the contained side of the relationship, it contains the PK. |
| **REFERENCED COLUMN** | Is the PK column of the relationship |
| **DATATYPE** | Indicate the datatype of the relationship |
| **SIZE** | Indicate the size of the relationship. In case the PK and FK don't share the same size, the greater must be used. |
| **REFERENCED VALUE** | Is the value that the REFERENCED COLUMN must take |
| **SEQUENCE** | Indicate the sequence of references composition for a match. If the sequence is N/A, then the reference is direct. For indirect references, it's equal to the ID of the previous reference in the sequence. |
| **NAME/DESC** | Anything else that can be used to provide more understanding about the reference. It has no use for the DMT |

*Table 4 R-References table columns*

## Insertion Rules

While inserting data into r-references, the following rules must be respected:

1. ID must not be empty
2. ID must be a number and unique
3. MATCH ID must not be empty
4. MATCH ID must be an existing ID value of Match-L-to-R sheet
5. REFERENCE TABLE must not be empty
6. REFERENCE TABLE must be equal to the Match-L-to-R TABLE that owns the reference or equal to N/A
7. REFERENCE COLUMN must not be empty

8. REFERENCE COLUMN must be equal to the Match-L-to-R RIGHT:COLUMN that owns the reference or equal to N/A only if REFERENCE TABLE is N/A.
9. REFERENCED TABLE must not be empty
10. REFERENCED TABLE must be a valid source database table
11. REFERENCED COLUMN must not be empty
12. REFERENCED COLUMN must be a valid source database PK column
13. DATATYPE must not be empty
14. DATATYPE must be a valid for source database
15. SIZE must not be empty
16. SIZE must be the greater of REFERENCE COLUMN and REFERENCED COLUMN
17. REFERENCED VALUE cannot be empty
18. REFERENCED VALUE can be any value including pre-defined constants (ALL, CURR, EQUALS and NULL)

## Insertion guidelines

In order to provide easy comprehension by human readers, consider the following guidelines:

1. For ID column use positive numbers in ascending order
2. Use upper case letters for table, datatype and constant names
3. Use lower case letters for column names
4. Provide a comprehensive name/description about references

## R-References to retrieve CURRS

A CURR is the value(s) of the parameter(s) used to select the parent tuple and/or itself. CURRS is a list of CURR of a particular tuple, it indicates the number of records that should be processed and the CURR that should be used as the parameter to retrieve data from source database in all the non PK matches of the tuple. For e.g.: The tuple with table PERSON needs to know all the persons in the source that must be processed/migrated (CURRS) and which parameter must be used to find and retrieve each person from source database (CURR). The DMT uses the r-reference of PK match of a tuple to query CURRS.

As an example, the tuple with ID equal to 1 has a PK match with Match ID equal to 1, the r-references of this very match is used to retrieve the CURRS of tuple 1.

| | Manual Data | | OpenMRS | | | | | SESP | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | Tuple | Terminology | Column | Datatype | Size | Required? | PK | Table | Column | Datatype | Size | Required? | Match ID |
| 3 | 1 | N/A | person_id | INT | 11 | YES | YES | N/A | N/A | N/A | N/A | N/A | 1 |
| 4 | 1 | Sexo | gender | VARCHAR | 50 | NO | NO | T_PACIENTE | sexo | TEXT | 50 | YES | 2 |
| 5 | 1 | Data de Nascimento | birthdate | DATE | 3 | NO | NO | T_PACIENTE | datanasc | DATE/TIME | 8 | NO | 3 |
| 6 | 1 | N/A | birthdate_estimated | TINYINT | 1 | YES | NO | N/A | N/A | N/A | N/A | N/A | 4 |
| 7 | 1 | N/A | dead | SMALLINT | 6 | YES | NO | N/A | N/A | N/A | N/A | N/A | 5 |
| 8 | 1 | N/A | death_date | DATETIME | 8 | NO | NO | N/A | N/A | N/A | N/A | N/A | 6 |

*Figure 1 Match-L-to-R sheet*

The r-reference of match with Match ID equal to 1 (r-reference with ID equal to 1), retrieves ALL the **nid** column values from table T_PATIENT, these **nid** values are the CURRS and each **nid** value in CURRS is the CURR used to query a non PK match of the same tuple with ID equal to 1.

| | ID | Match ID | Reference Table | Reference Column | Referenced Table | Referenced Column | Datatype | Size | Referenced Value | Sequence | Name/Desc |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | N/A | N/A | T_PACIENTE | nid | TEXT | 50 | ALL | N/A | |
| 2 | 2 | 2 | N/A | N/A | T_PACIENTE | nid | TEXT | 50 | CURR | N/A | |
| 3 | 3 | 3 | N/A | N/A | T_PACIENTE | nid | TEXT | 50 | CURR | N/A | |
| 4 | 4 | 8 | N/A | N/A | T_PACIENTE | nid | TEXT | 50 | CURR | N/A | |
| 5 | 5 | 11 | N/A | N/A | T_PACIENTE | nid | TEXT | 50 | CURR | N/A | |

*Figure 2 R-References sheet*

# Value Match Sheet

Value match represents the correspondences between values of equivalent groups (lists) in both sides of the migration. The group/list of values must be of single selection and the match must point to the group/list of value matches. If a match points to a value match, it says to the DMT: *"take the value that you found in the right side query of the match and find its equivalent, then set it into the left side of the match"*.

Value Match contain the columns listed and described in Table 5 Value Match table columns.

| COLUMN | DESCRIPTION |
|---|---|
| **ID** | Is the unique identifier of the value match group, it is specific of DMT internal processes and given by the user. |
| **GROUP NAME** | Is the concept name that the group represents, it's formulated in a way that the user best comprehends. It has no use for the DMT. |
| **LEFT:TABLE** | Indicate the target database table that keeps the group/list |
| **LEFT:COLUMN** | Indicate the target database column that keeps the name of the group/list element |
| **LEFT:VALUE** | Is the value of a group/list element in the target database |
| **LEFT:PK** | Is the PK of the group/list element in the target database |
| **RIGHT:TABLE** | Indicate the source database table that keeps the group/list |
| **RIGHT:COLUMN** | Indicate the source database column that keeps the name of the group/list element |
| **RIGHT:VALUE** | Is the name of the group/list element in the source database |
| **RIGHT:PK** | Is the PK of the group/list element in the source database |

*Table 5 Value Match table columns*

## Insertion Rules

While inserting data into Value-Match, the following rules must be respected:

1. ID must not be empty
2. ID must be a number and unique
3. GROUP NAME must not be empty
4. LEFT:TABLE must not be empty
5. LEFT:TABLE must be a valid OpenMRS table
6. LEFT:TABLE must be the same for all the value matches of the group
7. LEFT:COLUMN must not be empty

8. LEFT:COLUMN must be a valid OpenMRS column
9. LEFT:COLUMN must be the same for all the value matches of the group
10. LEFT:VALUE must not be empty
11. LEFT:VALUE must be a valid element of the group
12. LEFT:PK must not be empty
13. LEFT:PK must be a valid PK of LEFT:VALUE
14. RIGHT:TABLE must not be empty
15. RIGHT:TABLE must be a valid source database table
16. RIGHT:TABLE must be the same for all the value matches of the group
17. RIGHT:COLUMN must not be empty
18. RIGHT:COLUMN must be a valid source database column
19. RIGHT:COLUMN must be the same for all the value matches of the group
20. RIGHT:VALUE must not be empty
21. RIGHT:VALUE must be a valid element of the group or N/A if there is no match for the RIGHT:VALUE
22. LEFT:PK must not be empty
23. LEFT:PK must be a valid PK of LEFT:VALUE or N/A if RIGHT:VALUE is N/A

## Insertion guidelines

In order to provide easy comprehension by human readers, consider the following guidelines:

1. For ID column use positive numbers in ascending order
2. Use upper case letters for table names