Business Intelligence

Lecture 6

STTM: Source To Target Mapping

Source to Target Mapping

When moving data from one system to another, it's almost impossible to have a situation where the source and the target system have the same schema.

- Need for a mechanism that allows users to map their attributes.
- This process becomes more complicated than it already is when there is data that has to be moved to a central data warehouse from various data sources, each having different schemas.

Source to Target Mapping

Source to Target mapping can be defined as a set of instructions that define how the structure and content in a source system would be transferred and stored in the target system.

- Guidelines for the ETL (Extract, Transform, Load) process
- Instructions on dealing with multiple data types, unknown members and default values, foreign key relationships, metadata, etc.

Source to Target Mapping

TARGET						SOURCE					
Mapping Change Date	Target Table	Target Column	Nullable	PK	Data-type, Length	Source Table	Source Column	Data-type, Length	Expression, Transformation	Default Value	Error Types and Handling
Y/M/D	Name of target table	Target table column name	Whether a field can be null	Primary key field for target	Data type & length for target column	Name of source table	Column in source table from which data is extracted	Data type and length for this source column	Decodes, aggregates, conversions, if statements, lookup functions	Value to use in target field when source field is null	Used to document Not null, value if looked up, pk, fk, etc...comments, issues

Why Source-Target Mapping?

Source to Target mapping is an integral part of the data management process.

Source to Target Mapping assists in three processes of the ETL pipeline:

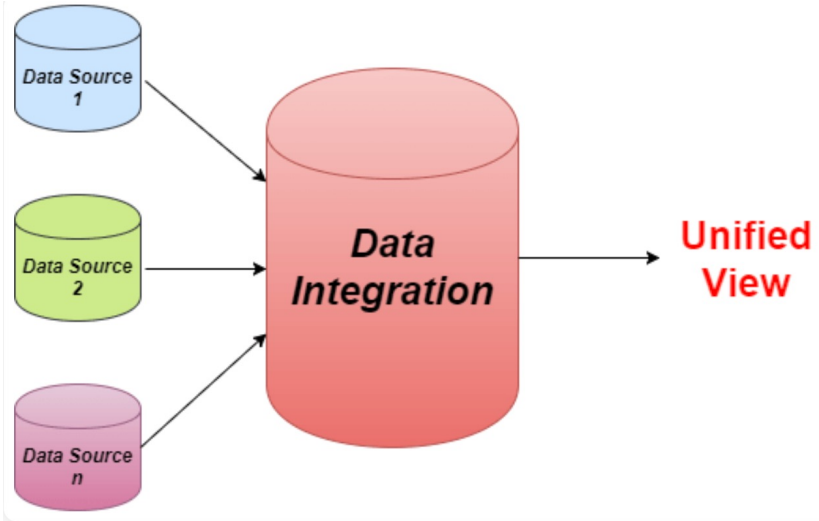
Data Integration

Data Migration

Data Transformation

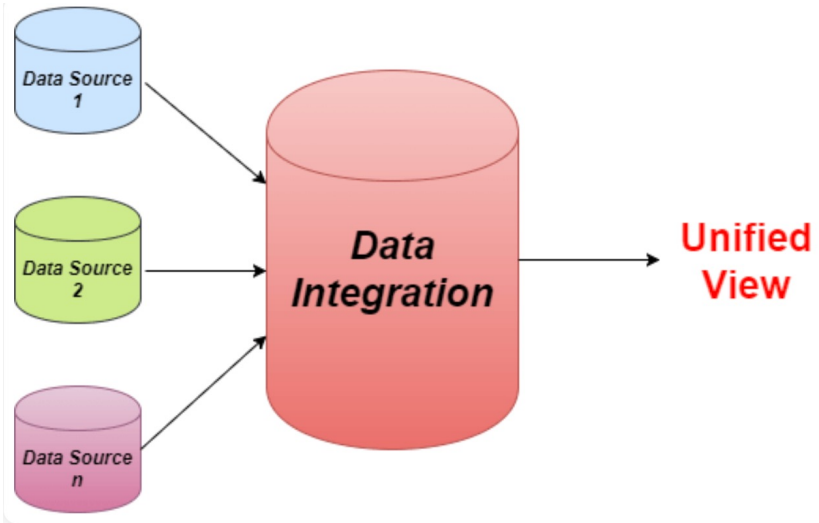
Data Integration

- Moving data from one system to another.
- In most cases, this movement of data is from the **operational database** to the **data warehouse**.



Data Integration

The mapping defines how data sources relate to the data warehouse during data integration. Instructions on how multiple data sources intersect with each, which data record is preferred if duplicate data is found, etc.



Data Migration

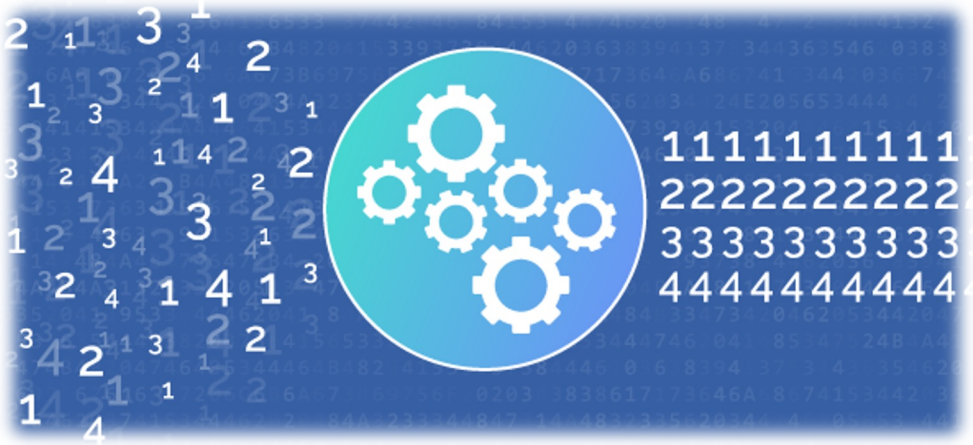
The movement of data from one system to another performed as a one-time process.

In most cases, it is done to ensure that multiple systems have a copy of the same data.

Data Transformation

The conversion of data at the source system to a format required by the destination system.

- Includes operations such as data type transformation, handling missing data, etc.
- Defines how to map, modify, join, filter, or aggregate data as required by the destination system.



Steps in Source to Target Mapping

Step 1: Defining the Attributes

- Defining which tables and which attributes in those tables are to be transferred.
- If data integration is being performed, the frequency of integration is also defined in this step.

Step 2: Mapping the Attributes

Once the data to be transferred has been defined, it has to be mapped according to the destination system's attributes.

Steps in Source to Target Mapping

Step 3: Transforming the Data

Converting the data in a form suitable to be stored in the DW

Step 4: Testing the Mapping Process

Once the first three steps have been completed, it has to be tested on some sample data sources to ensure that the right data attributes in the proper form are mapped correctly with the destination system.

Steps in Source to Target Mapping

Step 5: Deploying the Mapping Process

Upon completion of testing and successful data transfer, migration or integration can be scheduled on the live data as per the user's requirements.

Step 6: Maintaining the Mapping Process

The Source to Target Mapping process must be maintained and updated periodically to handle large datasets and any new data sources if required.

Source to Target Techniques

Manual - manually code the connection between the source and the destination system.

This process can only be used in case the mapping is to be performed for only a few sources that don't have much data.

Advantages

Flexible, Completely customizable to the exact needs of the user

Disadvantages

Manual, Time-consuming, Resource-intensive, Code-dependent, Error-prone.

Source to Target Techniques

Automated – used when the number of sources and the volume of data will increase with each round of data transfer.

Advantages

No technical knowledge required, Fast, Easy to scale, Eliminates human error.

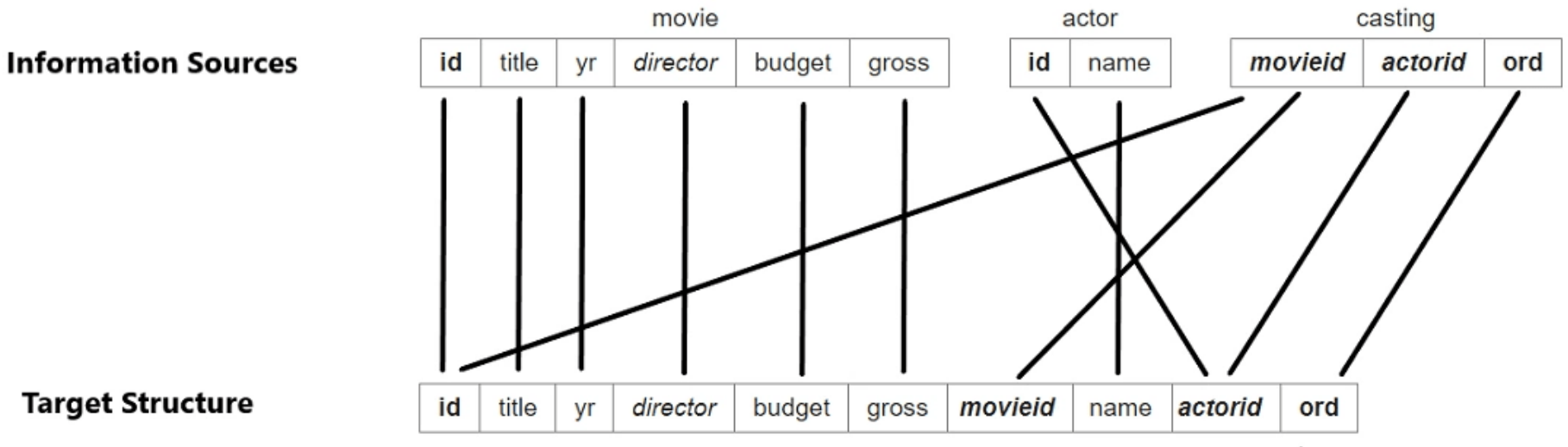
Disadvantages

Training required for use, In-house solutions are expensive to build.

Source to Target Mapping

TARGET						SOURCE					
Mapping Change Date	Target Table	Target Column	Nullable	PK	Data-type, Length	Source Table	Source Column	Data-type, Length	Expression, Transformation	Default Value	Error Types and Handling
Y/M/D	Name of target table	Target table column name	Whether a field can be null	Primary key field for target	Data type & length for target column	Name of source table	Column in source table from which data is extracted	Data type and length for this source column	Decodes, aggregates, conversions, if statements, lookup functions	Value to use in target field when source field is null	Used to document Not null, value if looked up, pk, fk, etc...comments, issues

Source to Target Mapping



Source to Target Mapping

TARGET DIMENSION: T_DIM_PRODUCT					SOURCE DATA					Transformation
Column	datatype	size	PK	SCD	origin		data element	column	datatype	
product_key	NUMBER	12	PK	1	-	-	-	-	-	Surrogate key
product_natural_Key	NUMBER	10	-	1	sales_src	database	view_products	src_id	INTEGER	
product_name	VARCHAR2	30	-	1	sales_src	database	view_products	src_name	VARCHAR2(30)	
product_brand	VARCHAR2	30	-	1	sales_src	database	view_products	src_brand	VARCHAR2(30)	
product_category	VARCHAR2	20	-	1	sales_src	database	view_products	src_category_id	CHAR(5)	SELECT c.src_name FROM view_categories c, view_products p WHERE c.src_id = p.src_category_id
					sales_src	database	view_categories	src_id	CHAR(5)	
					sales_src	database	view_categories	src_name	VARCHAR	
product_size_package	VARCHAR2	20	-	2	sales_src	database	view_products	src_height	INTEGER	concatenate (src_height, 'x', src_width, 'x', src_depth)
					sales_src	database	view_products	src_width	INTEGER	
					sales_src	database	view_products	src_depth	INTEGER	
product_type_package	VARCHAR2	30	-	2	sales_src	database	view_products	src_pack_type	VARCHAR2(30)	

product_diet_type	VARCHAR2	15	-	2	sales_src	database	view_products	src_calories_100g	INTEGER	IF (src_calories_100g) THEN 'EXTRA-LIGHT' ELSE
										IF (src_calories_100g)<30 THEN 'LIGHT' ELSE
										IF (src_calories_100g<70) THEN 'REGULAR' ELSE 'FAT'

product_liquid_weight	NUMBER	8,2	-	2	sales_src	database	view_products	src_liq_weight	NUMBER(8,2)	