Previews of course books are provided as an opportunity to see the quality of material used at TDWI educational events. The selected pages are representative of the entire course book. These pages cannot be printed.

The previews are intended to help you select the courses that best fit your needs.

TDWI Education

In-Depth Business Intelligence and Data Warehousing Education

Data Conversion, Consolidation, and Cleansing—Practical Skills

THE DATA WAREHOUSING INSTITUTE

TABLE OF CONTENT

Data Conversion Overview

Typical Project Steps

Step 1. Define Data Mapping

Step 2. Extract, Transform, Load

Step 3. Drown in Data Problems

Step 4. Find Scapegoat

Data Conversion Overview

Typical Project Steps

OVERVIEW

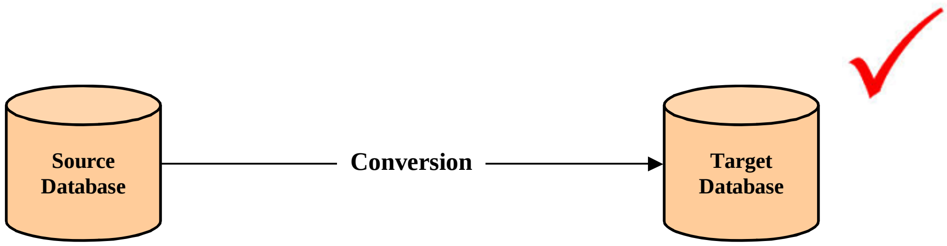
A typical project starts with the definition of data mapping between source and target databases. The data mapping is usually defined based on existing data models and data dictionaries.

The next step is so extract the data from the source databases, perform necessary transformations, and load the resultant dataset into the target database.

Usually some data is lost in the process, cannot be loaded, or is converted incorrectly because the mapping is totally ignorant of the actual data content and quality. Some problems are discovered during loading and testing and the project team scrambles to patch the process on the fly. This typically leads to more problems. The majority of the problems ate gradually uncovered in the month and years after conversion.

Data Conversion Overview

Keys to Success – Defining Objective



Target Data Quality Levels	≥	Pre-Defined Benchmarks
Target Data Quality Levels	≥	Source Data Quality Levels

Data Conversion Overview

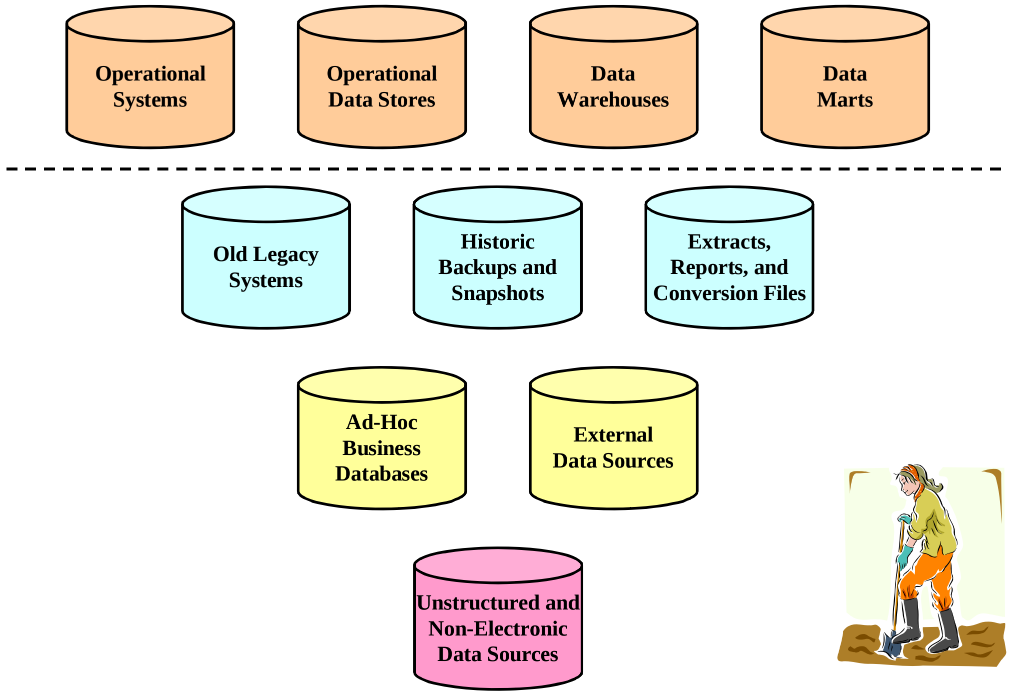
Keys to Success – Defining Objective

OVERVIEW

It is common to think about the objective of data conversion as "to move the data from table of system A to appropriate tables of system B". This is a very dangerous view. The true objective of the data conversion is to convert all relevant data from source to target database, while at least retaining the data quality level. Data quality of course, is defined as the fitness to the purpose of use. If data quality levels are not retained the conversion cannot be viewed as successful. In fact, data conversion is a good time to improve data quality. So even better objective is to achieve a certain predefined level of data quality in the target database.

Data Gathering

Identifying Data Sources



Data Gathering

Identifying Data Sources

OVERVIEW

Data sources vary considerably in origin, function, ownership, and use type. The most obvious data sources are operational systems, operational data stores, data warehouses, and data marts. However numerous other data sources that can be extremely useful for data conversion and cleansing exist but are typically overlooked.

Legacy systems are old information systems inherited from past generations of software design. They are often barely kept alive or even shutdown and the data is relocated to the tape library. Legacy systems ironically have one big advantage over modern applications – they are not well suited for data purging and thus typically contain wealth of historic data. In fact you can often expect (and be on the lookout) to find data that neither users nor documentation anticipated in a legacy system. Whether you like it or not, legacy systems are often the best source of historical data.

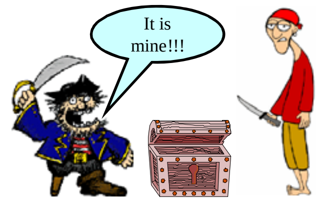
Historic backups and snapshots are typically created as a part of disaster recovery programs or due to the fiduciary policies, such as an often-encountered corporate policy requiring seven-year retention of year-end payroll backup files. They are created for most databases and stored much longer than one would anticipate, often hidden in the far corners of corporate tape libraries. Backups are often a great source of historic information.

"Ad-hoc" databases are systems and documents created by business users for their internal purposes. "Ad-hoc" databases are rarely well documented and are typically not developed on solid data modeling and database design foundation. However, they almost always contain highly reliable data simply because people who create them are at the same time their users and so are likely very diligent, motivated, and detail-oriented.

Data Gathering

Subject De-Duplication

Subject matching during data loading to the staging area is a good time to perform subject de-duplication.



These two records clearly belong to different subjects but have identical SSN. One of the SSNs is erroneous.

These two records are exact duplicates.

E_EMPLOYEE_PROFILE

EmpID	SSN	LastName	FirstName	BirthDate	Gender	CompanyCode
141857	834-51-2788	ACCARDI	LAURENE	9/30/1954	F	B8
235572	770-27-1938	CADY	CONRAD	11/19/1950	M	B3
275356	770-27-1938	BURR	LINCOLN	1/17/1958	M	N4
186860	850-67-8703	BUBB	CHARLES E	8/4/1942	M	B6
346981	988-65-1411	ABRAHAM	MILLARD	7/19/1943	M	N2
155107	988-65-1411	ABRAHAM	MILLARD	7/19/1943	M	N2
247826	989-85-2689	CHIU	HARLAN	1/1/1922	M	N2
39159	991-96-8738	ABRUZZO	DIEGO	1/1/1922	M	N2
39160	996-96-8738	ABRUZZO	DIEGO	1/1/1922	M	N2
145944	997-77-5120	BART	MANFRED	1/11/1931	M	NC

10 of 10 Figure 1 FLTR SRT ALL

DIEGO ABRUZZO has 2 records with different SSNs, but identical BirthDate (1/1/1922). Further analysis shows that the SSNs have only 1 different digit. This is a case of complex duplicate subjects.

Data Gathering

Subject De-Duplication

OVERVIEW

It is desirable that that every real world subject represented in the data is uniquely identifiable and can be distinguished from all other subjects of the same type. In relational databases primary key is used as a proxy for the real identity key. While primary keys are usually shown in data models and enforced in databases, this does not guarantee proper subject identity.

While we are matching subjects across data sources, it is a good time to ensure unique identification of all subjects in each source. This is done by subject de-duplication. Comprehensive de-duplication requires complex techniques. Fortunately, various tools are available on the market for de-duplication of records for persons or businesses.

EXAMPLE

`E_EMPLOYEE_PROFILE` table lists all employees along with their basic indicative data. The table has surrogate key attribute `EMP_ID` declared as a primary key and enforced by the database. Of course uniqueness of `EMP_ID` is guaranteed by design yet does not mean that each employee is truly uniquely identified in the data (see Abraham Millard).

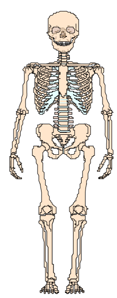
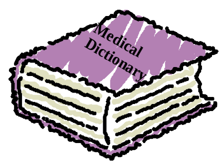
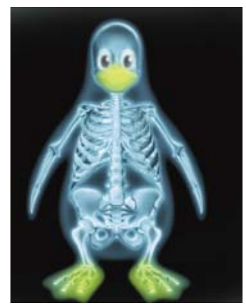
Attribute `SSN` is nominated for the primary key in the relational data model, though its uniqueness is not enforced in the database. Obviously all employees must have unique `SSN`. The records for Conrad Cady and Lincoln Burr violate this constraint. One of the records has incorrect `SSN`.

Yet even uniqueness of `SSN` does not guarantee that all employees are truly uniquely identified in the data. For instance, same employee may be listed twice – once with correct and once with erroneous `SSN` (see Diego Abruzzo). A combination of full name and date of birth can be used for fuzzy matching and as a best proxy for true identity key.

Data Profiling

Overview

Data catalogue is a collection of basic metadata about data attributes. Relational data models and state-transition models describe the structure of the data. Data models and catalogues are the source of initial knowledge about data. In real life data models and catalogues are usually incomplete, incorrect, and obsolete.

Data Models	Reference Data	Data Profiles
		
Describe Structure of the Data	Describe Meaning of the Data	Describe Actual Content of the Data

- Data profiling is aimed at examining the data and understanding hidden structure and dependencies
- Data profiling is key to building correct data mapping and data quality rules
- Data profiling is in itself a valuable exercise and produces metadata useful for many purposes

Data Profiling

Overview

DATA CATALOGUE

Data catalogue is a collection of basic metadata about data attributes. It includes basic attribute listings, detailed descriptions and usage patterns, as well as reference information, including valid values and their meanings, default values, etc.

DATA MODELS

Subject area models define main data subjects – categories of high-level business objects whose data is stored in the database.

Relational data models depict logical relationships between various entities and attributes.

State-transition models describe the life cycle of complex state-dependent objects.

Temporal models describe chronological structure of time-dependent data and event histories.

DATA PROFILING

Data models and catalogues are the source of initial knowledge about data. Data profiling is a group of experimental techniques aimed at examining the data and understanding its actual structure and dependencies.

The reason data profiling is so important is that actual data is often very different from what is theoretically expected. Over time data models and dictionaries become inaccurate. Data profiling is like an X-ray showing the hidden truth. It is key to building correct data mappings and quality rules. As a rule of thumb, the more in-depth analysis and profiling we conduct the easier it is to design a comprehensive set of data mappings and quality rules and achieve greater success in data conversion and consolidations.

Data Profiling

Profiling Techniques

Data profiling is often mistakenly equated to attribute profiling. The cause of that mistake is the proliferation of efficient attribute profiling tools. However, comprehensive data profiling is a far broader exercise.

Subject Profiling

Relationship Profiling

Attribute Profiling

Timeline Profiling

State-Transition Model Profiling

Dependency Profiling

In this class we do not address all data profiling techniques, but only those that are of significant value in data conversion and consolidation projects. The main objective here is to understand where the data can be found and what is their meaning. We will skip advanced state-transition model profiling and dependency profiling techniques aimed at understanding hidden data relationship. These techniques are however an important part of data quality assessment.

Data Profiling

Profiling Techniques

OVERVIEW

Data profiling is often mistakenly equated to attribute profiling. The cause of that mistake is the proliferation of efficient attribute profiling tools. However, comprehensive data profiling is a far broader exercise.

Subject profiling examines subjects in different tables or on different systems and helps to find where the information about each subject is stored.

Relationship profiling is an exercise in identifying entity keys and relationships as well as counting occurrences for each relationship in the data model. It is necessary to validate existing relational data models or build them when none are available.

Attribute profiling examines values of individual data attributes and provides information about frequencies and distributions of their values. It helps to identify meaning and allowed values for an attribute.

Timeline profiling looks for patterns in historical data, such as temporal distribution of the data, patterns of values for different time periods, etc.

State-transition model profiling examines lifecycle of state-dependent objects and provides actual information about the order and characteristics of states and actions. It helps build or validate state-transition models.

Dependency profiling uses various pattern recognition techniques to find hidden relationships between attribute values.

Data Quality Assessment

Approaches

Complete Data Validation

- Manual validation of each data element against a "trusted" source
- Often impossible as "trusted" source may not be easily available
- Impractical for all, but the smallest databases

Sample Data Validation

- Manual validation of a data sample with further extrapolation of the results to the untested data
- Impractical for large databases
- Does not provide detailed information about erroneous data elements

Using Data Quality Rules

- Use constraints that validate data accuracy and consistency and can be implemented in computer programs
- The same setup can be reused reassess data quality periodically

Data Quality Assessment

Approaches

COMPLETE DATA VALIDATION

The only way to be sure that a piece of data is correct is to compare it with some "trusted" source, that is a source which is correct 100% of the time. Such source may not always exist or at least may not be readily available.

Another problem is the time constraint on data quality assessment. Outside of the very small databases, total manual data validation is impractical.

SAMPLE DATA VALIDATION

Sampling approaches were suggested, mostly drawing on the experience of quality management in other industries. This scales the problem down a bit, but the solution remains impractical for larger databases or on an enterprise-wide scale. Also, sampling does not provide detailed information about erroneous data elements and thus is of limited use for data quality improvement.

USING DATA QUALITY RULES

Modern databases have two important characteristics that distinguish data from all other products. First, they allow the data to be accessed and processed with dramatic speeds. Secondly, myriads of data elements stored in them are tied by equally huge numbers of data relationships. The combination of these two factors allows validating the data in mass by computer.

The main tool of a data quality assessment professional is a data quality rule – a constraint that validates a data element or a relationship between several data elements and can be implemented in a computer program. The solution relies on the design and implementation of hundreds and thousands of such data quality rules and using them to identify all data inconsistencies. The same setup can then be reused over and over again to reassess data quality periodically with minimal effort.

Data Quality Assessment

Challenges

Mt Completeness

Designing all the rules and making sure that they indeed identify all data errors is the first challenge.

Mt Imperfection

Minimizing imperfection in the error reports and accounting for it is the second challenge.

Mt Organization

Organizing data quality metadata into a comprehensive metadata warehouse is the third challenge.

Data Quality Assessment

Challenges

COMPLETENESS

The objective of data quality assessment is to identify all data errors. Considering the volume and structural complexity of a typical database this is a daunting task. Data quality rules are a perfect tool as they can test large quantities of data pieces in seconds. Yet it will require hundreds or thousands of them to do the job.

<u>Designing all the rules and making sure that they indeed identify all data errors is the first challenge.</u>

IMPERFECTION

Data quality rules are inexact by their nature. They miss some errors and falsely identify others; they may not tell you which data element is erroneous even when the error is identified; they may identify the same error in many different ways. In other words, data quality metadata may suffer from the same malady as the data itself – poor quality. This imperfection, if not understood and controlled, will overrun and doom any data quality assessment effort.

<u>Minimizing the imperfection in the data quality metadata and accounting for it in the results is the second challenge.</u>

ORGANIZATION

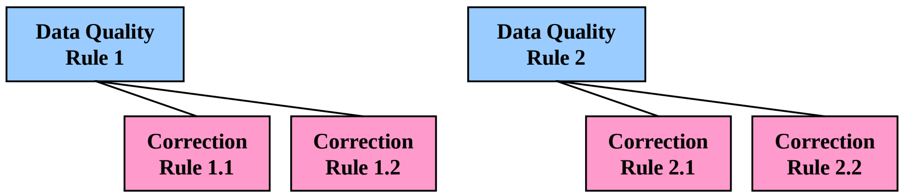
Data quality rules produce endless reports of data errors. Each error applies to one or several data elements from one or several tables for one or several subjects. And the error reports are just the tip of the iceberg. Other categories of metadata produced in the process of data quality assessment include data profiles, aggregated data quality scores, etc.

<u>Organizing data quality metadata into a comprehensive Data Quality Metadata Warehouse is the third challenge.</u>

Data Cleansing

Introduction to Correction Rules

Rule-driven approach to data cleansing relies on design of correction rules to fix errors found by various data quality rules. In this way, data cleansing is decomposed into many small steps.



It is important to understand that objective of each correction rule is to correct certain type of errors, rather than to make specific records accurate. It takes many rules working together to achieve ultimate data accuracy.

Example

Data quality rules may identify duplicate events in the employment status history. Correction rule will remove the duplicate. This does not guarantee that the remaining event information is accurate.

Data Cleansing

Introduction to Correction Rules

OVERVIEW

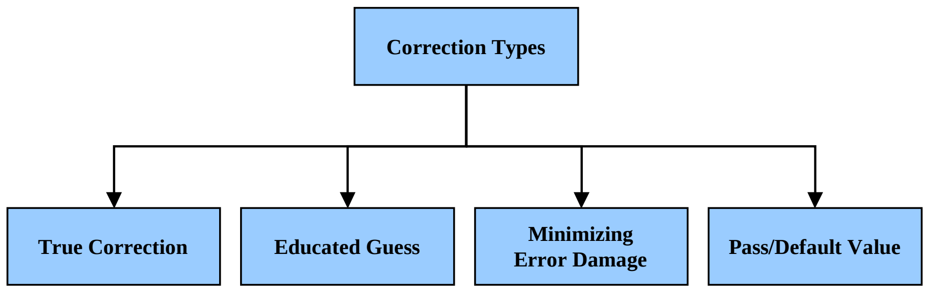
Rule-driven approach to data cleansing relies on design of correction rules to fix errors found by various data quality rules. Each correction rule is a subroutine called inside or immediately after execution of the data quality rule, in case an error is identified. This approach allows to decompose data cleansing into many small steps, each dealing with a specific solution to a unique subset of data problems.

It is important to understand that objective of each correction rule is to correct certain type of errors, rather than to make specific records accurate. It takes many rules working together to achieve ultimate data accuracy. For instance, data quality rule may identify duplicate events in the employment status history. Correction rule will remove the duplicate. This does not guarantee that the remaining event information is accurate.

Data Cleansing

Correction Types

Not all corrections must achieve 100% accuracy. Sometimes a perfect correction rule may not be designed, but an improvement can be made to the data quality at a lower cost than exhaustive manual research.



Data Cleansing

Correction Types

OVERVIEW

A common misconception is that correction must make the data accurate. In reality the objective of data cleansing is not to achieve 100% accuracy level (which is totally impractical and usually impossible) but rather to increase data quality up to a certain acceptable level. This allows to use some creative approaches to corrections when it is impossible to make 100% accurate correction with the data available.

One technique is to use “educated guess”, i.e. some heuristic algorithm that makes accurate correction more often than not. Say, a group of 100 employees is known to work full-time, but their weekly scheduled hours are missing. However, it is known that for their positions 90% of all employees are scheduled to work 35 hours per week. Then it is a good correction to set missing schedule to 35. While we are likely to make 10 mistakes we will correct 90 and data quality will certainly improve. It is still possible to later identify which corrections were made using such inexact techniques using corrections catalogue.

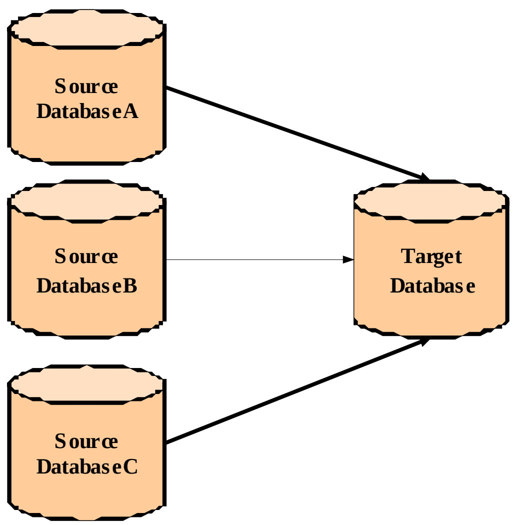
Alternative technique is to “minimize error damage”. For instance, if the data is used to produce employee benefit statements and the same weekly schedule is not available, yet it might influence benefit eligibility. Using value 40 would be in favor of employee, while 35 could potentially make them ineligible. Implications of the first approach may be far less expensive than the second, and so it may be financially better to choose this incorrect correction over a more accurate one.

Finally, in some cases a correction rule cannot be identified, yet something must be done, as the data in its existing form may be unacceptable. For example, during conversion some values may not fit into the target system and must be replaced even if correct values are not available. The solution is to use some default/pass values.

Data Consolidation

Winners-Losers Matrix

The common approach is to build a winners-losers matrix, which shows order of mapping from various data sources. The fallacy of the approach is that it assumes precedence of one source over another. Since in reality no data is perfect (especially legacy data) such assumption inevitably fails.



Target Data Element	1st Choice	2nd Choice	3rd Choice
Date of Birth	A	B	C
…	C	A	B

Data Consolidation

Winners-Losers Matrix

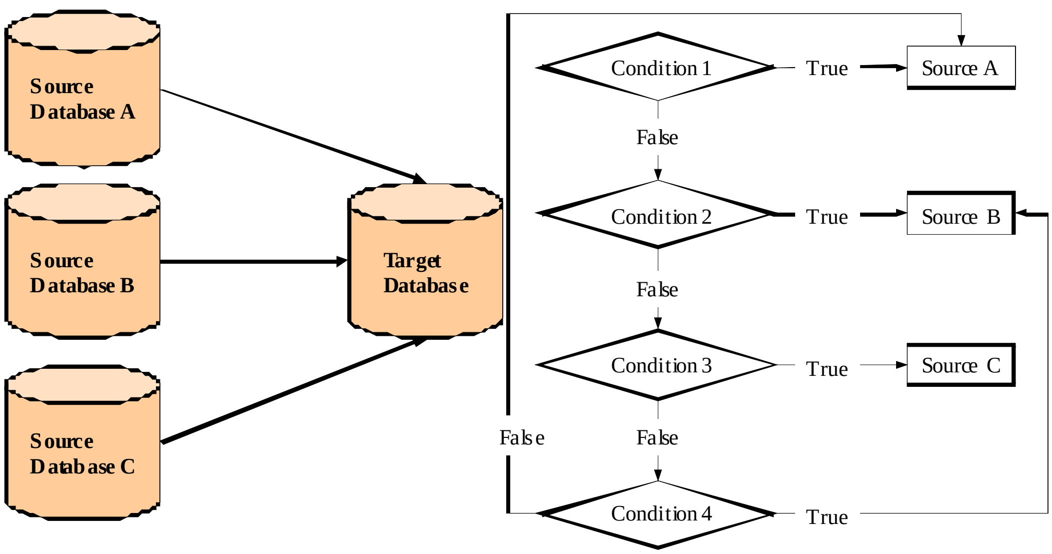
OVERVIEW

The traditional approach is to setup a winner-loser matrix indicating which source data element is picked up in case of a conflict. For instance, date of birth will be taken from System A if present, from System B otherwise, and from System C if it is missing in both A and B. This rarely works because it assumes that data on System A is always correct – a laughable assumption.

Data Consolidation

Winners-Losers Hierarchy

To mitigate the problem, the winner-loser matrix is usually transformed into a complex conditional hierarchy. However the decision tree soon becomes very complex, hard to understand and manage, and still rarely yields good results for complex historical data.



Data Consolidation Overview

Winners-Losers Hierarchy

OVERVIEW

To mitigate the problem, the winner-loser matrix is usually transformed into a complex conditional hierarchy. Now we take the date of birth from System A for all males born after 1956 in California, except if that date of birth is January 1, 1970, in which case we take it from System B, unless of course the record on System B is marked as edited by John Doe who was fired for playing games on the computer while doing data entry, in which case we pull it from Spreadsheet C…

At some point the decision tree becomes so complex, that nobody really understands what is going on. It becomes impossible to manage and rarely yields good results for all, but the simple indicative data elements. The approach inevitably fails for complex historical data, such as event histories and state-transition histories.

Even more serious issue is that in this model we absolutely cannot cleanse the data before conversion, because it is impossible to determine which data elements will really be used.

