DATA MANAGEMENT FOR BUSINESS INTELLIGENCE

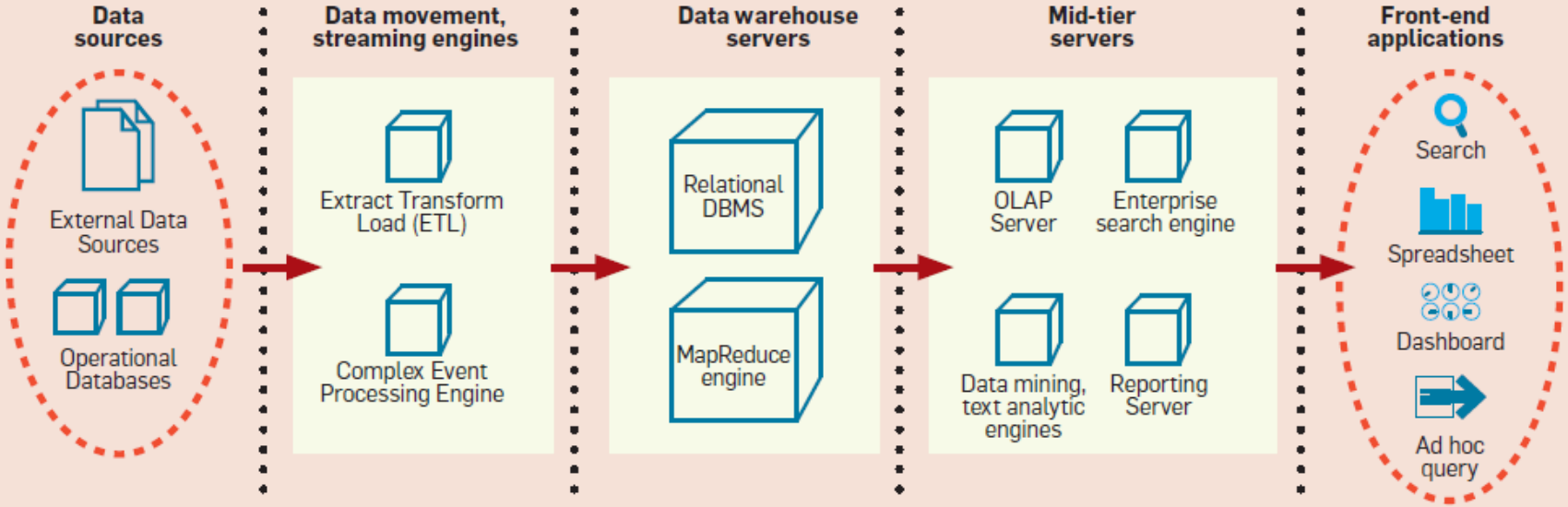
Data Access: Files

Salvatore Ruggieri

Computer Science Department, University of Pisa

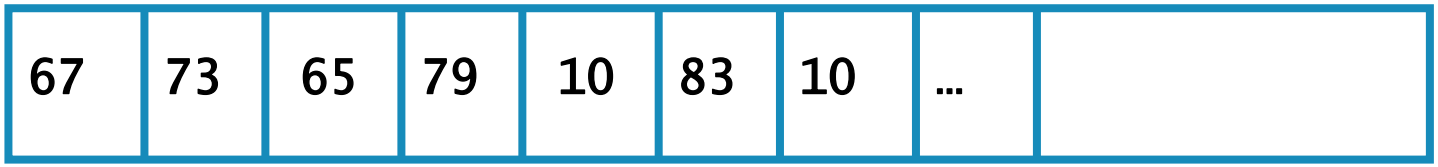
BI Architecture

Figure 1. Typical business intelligence architecture.



What is a file?

- File = sequence of bytes



How bytes are mapped to chars?

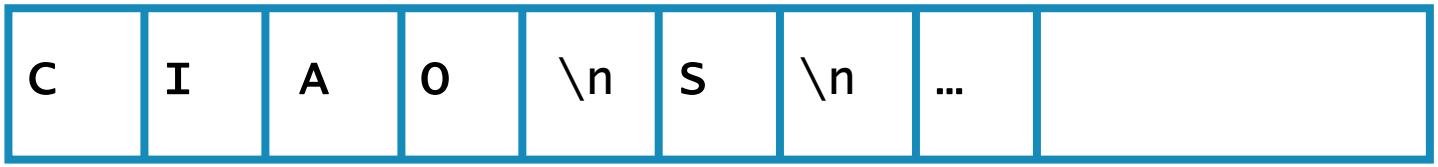
- Character set = alphabet of characters
- Coding bytes by means of a character set
  - ASCII, EBCDIC (1 byte per char)
  - UNICODE (1/2/4 bytes per char)

American Standard Code for Information Interchange

CODE	CHAR	CODE	CHAR	CODE	CHAR	CODE	CHAR	CODE	CHAR
0	NUL	26	SUB	52	4	78	N	104	h
1	SOH	27	ESC	53	5	79	O	105	i
2	STX	28	FS	54	6	80	P	106	j
3	ETX	29	GS	55	7	81	Q	107	k
4	EOT	30	RS	56	8	82	R	108	l
5	ENQ	31	US	57	9	83	S	109	m
6	ACK	32	SP	58	:	84	T	110	n
7	BEL	33	!	59	;	85	U	111	o
8	BS	34	"	60	<	86	V	112	p
9	HT	35	#	61	=	87	W	113	q
10	LF	36	$	62	>	88	X	114	r
11	VT	37	%	63	?	89	Y	115	s
12	FF	38	&	64	@	90	Z	116	t
13	CR	39	'	65	A	91	[	117	u
14	SO	40	(	66	B	92	\	118	v
15	SI	41	)	67	C	93	]	119	w
16	DLE	42	*	68	D	94	^	120	x
17	DC1	43	+	69	E	95	_	121	y
18	DC2	44	,	70	F	96	`	122	z
19	DC3	45	-	71	G	97	a	123	{
20	DC4	46	.	72	H	98	b	124	|
21	NAK	47	/	73	I	99	c	125	}
22	SYN	48	0	74	J	100	d	126	~
23	ETB	49	1	75	K	101	e	127	DEL
24	CAN	50	2	76	L	102	f		
25	EM	51	3	77	M	103	g		

Text file = file+character set

- Text file = sequence di characters

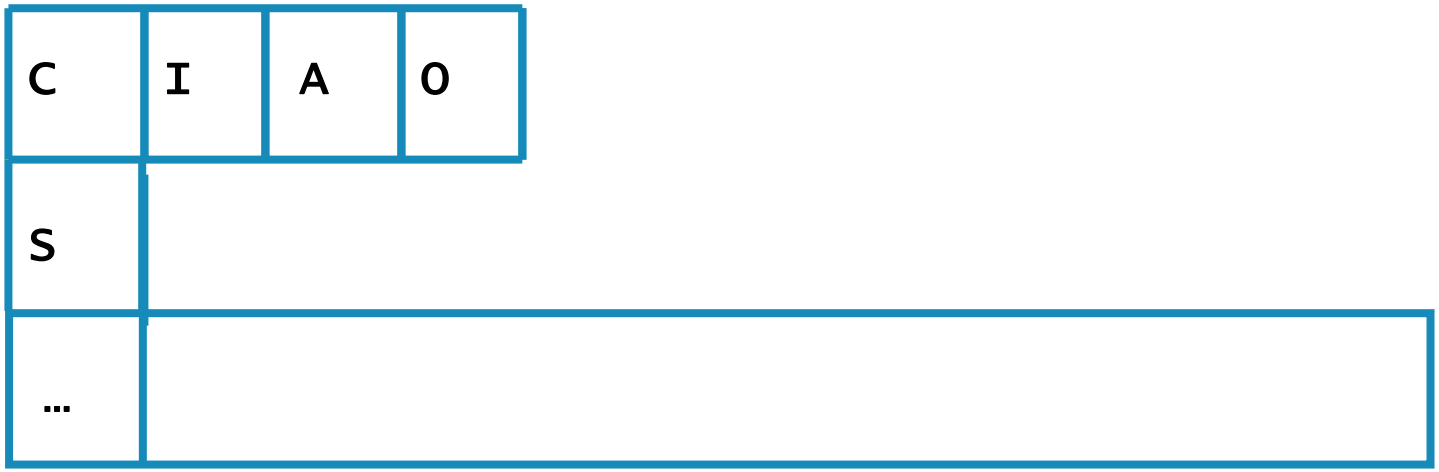


Viewing text files

- By a text editor
  - Emacs, Nodepad++,TextPad, GEdit, Vi, etc.
- "Carriage return" character
  - Start a new line
  - Coding
    - Unix: 1 char ASCII(0A) ('\n' in Java)
    - Windows: 2 chars ASCII(0D 0A) ("\r\n" in Java)
    - Mac: 1 char ASCII(0D) ('\r' in Java)
  - Conversions
    - > **dos2unix**
    - > **unix2dos**

Text file = file+character set

- Text file = sequence di **lines**



Tabular data format

Column

Row

Mario	Bianchi	23	Student
Luigi	Rossi	30	Workman
Anna	Verdi	50	Teacher
Rosa	Neri	20	Student

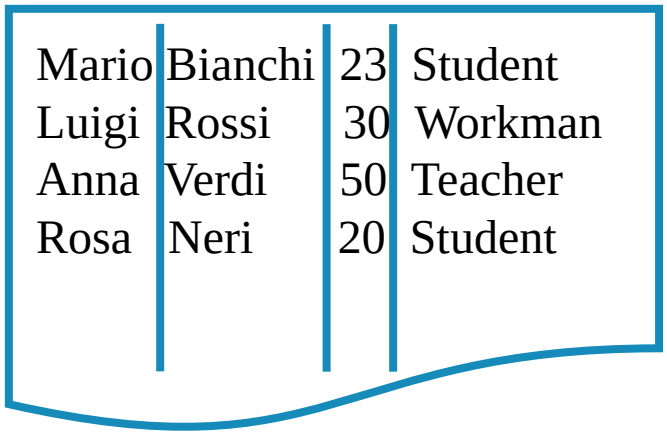
Representing tabular data in text files

- **Comma Separated Values (CSV)**
  - A row per line
  - Column values in a line separated by a special character
  - Delimiters: comma, tab, space

```
Mario,Bianchi,23,Student
Luigi,Rossi,30,Workman
Anna,Verdi,50,Teacher
Rosa,Neri,20,Student
```

Representing tabular data in text files

- **Fixed Length Values (FLV)**
  - A row per line
  - Column values occupy a fixed number of chars
    - Allow for random access to elements
    - Higher disk space requirements



Quoting

- What happens in CSV if a delimiter is part of a value?
  - Format error
- Solution: **quoting**
  - Special delimiters for start and end of a value (ex. " … ")

Mario Bianchi 23 Student
Luigi Rossi 30 Workman
Anna Verdi 50 Teacher
Rosa Neri 20 Student

"Mario Bianchi" 23 Student
"Luigi Rossi" 30 Workman
"Anna Verdi" 50 Teacher
"Rosa Neri" 20 Student

Missing values

- How to represent missing values in CSV or FLV?
  - A reserved string: "?", "null", ""

"Mario Bianchi" 23 Student
"Luigi Rossi" 30 **?**
"Anna Verdi" 50 Teacher
"Rosa Neri" **?** Student

Meta-data

- Describe properties of data
  - Table name, column name, column type, …

name	surname	age	occupation
string	string	int	string
Mario	Bianchi	23	Student
Luigi	Rossi	30	Workman
Anna	Verdi	50	Teacher
Rosa	Neri	20	Student

How to represent meta-data in text files?

- One or two rows: names and types

name	surname	age	occupation
string	string	int	string


```
name,surname,age,occupation
string,string,int,string
```

Meta-data and data in text files

- In the same file
  - Meta-data first (header), then data

name	surname	age	occupation
string	string	int	string
Mario	Bianchi	23	Student
Luigi	Rossi	30	Workman
Anna	Verdi	50	Insegnante
Rosa	Neri	20	Studente

name,surname,age,occupation
string,string,int,string
Mario,Bianchi,23,Studente
Luigi,Rossi,30,Operaio
Anna,Verdi,50,Insegnante
Rosa,Neri,20,Studente

Two issues

- **Where** are my files?
  - Local file systems
  - Distributed file systems
  - Network protocols
- Which **format** is file data in?
  - Text
    - CSV, JSON

Data interchange issue

- Problem: **data interchange** between applications
  - Proprietary data format do not allow for easy interchange
    - CSV with different delimiters, or column orders
    - Similar limitations of FLV, ARFF, binary data, etc.

- Solution:
  - definition of an interchange format…
  - … marking data elements with their meaning …
  - … so that any other party can easily interpret them.

JSON http://www.json.org/

- Objects:
  - comma-separated list of pairs in the form
    - name : value

```
{
    "name":"John",
    "surname":"Doe",
    "age":25
}
```

- Name is a string
- Value data types:
  - strings ("John")
  - integer, real (25)

JSON

- Value data types:
  - Arrays: comma-separated list of values

```
{
    "name":"John",
    "surname":"Doe",
    "age":25,
    "courses": ["BD", "DM", "AI"]
}
```

JSON

- Value data types:
  - Objects

```
{
    "name":"John",
    "surname":"Doe",
    "age":25,
    "courses": ["BD", "DM", "AI"],
    "address": {"street":"5th Av.", "city":"NY"},
    "friends": [ {"name":"Ed", "surname":"May"},
                 {"name":"Al", "surname":"Black"} ]
}
```

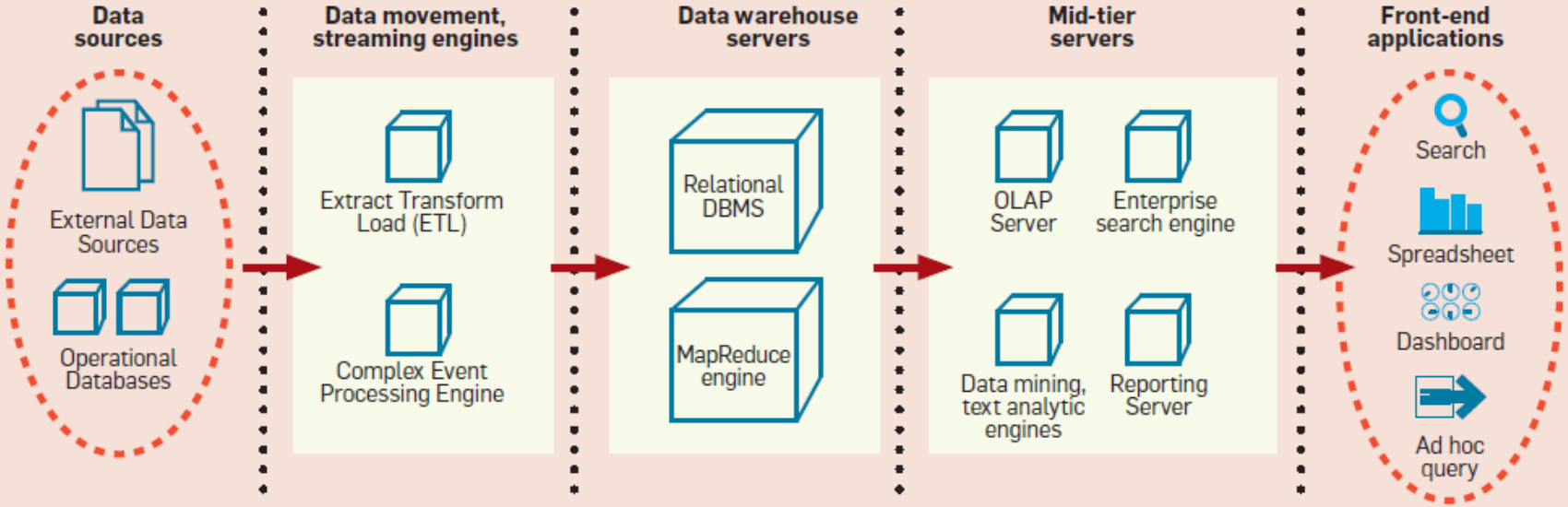
DATA MANAGEMENT FOR BUSINESS INTELLIGENCE

Data Access: Relational Data Bases

Master in Big Data Analytics and Social Mining

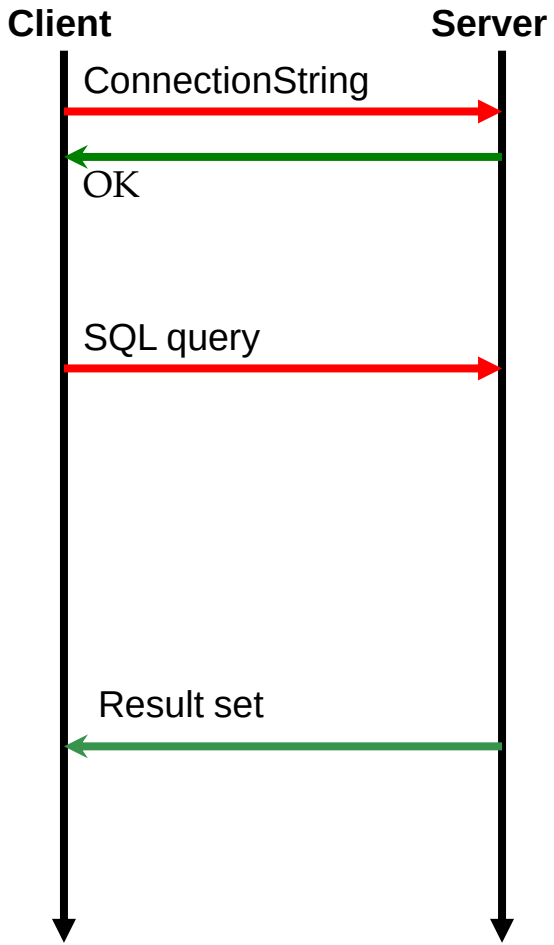
BI Architecture

Figure 1. Typical business intelligence architecture.



Connecting to a RDBMS

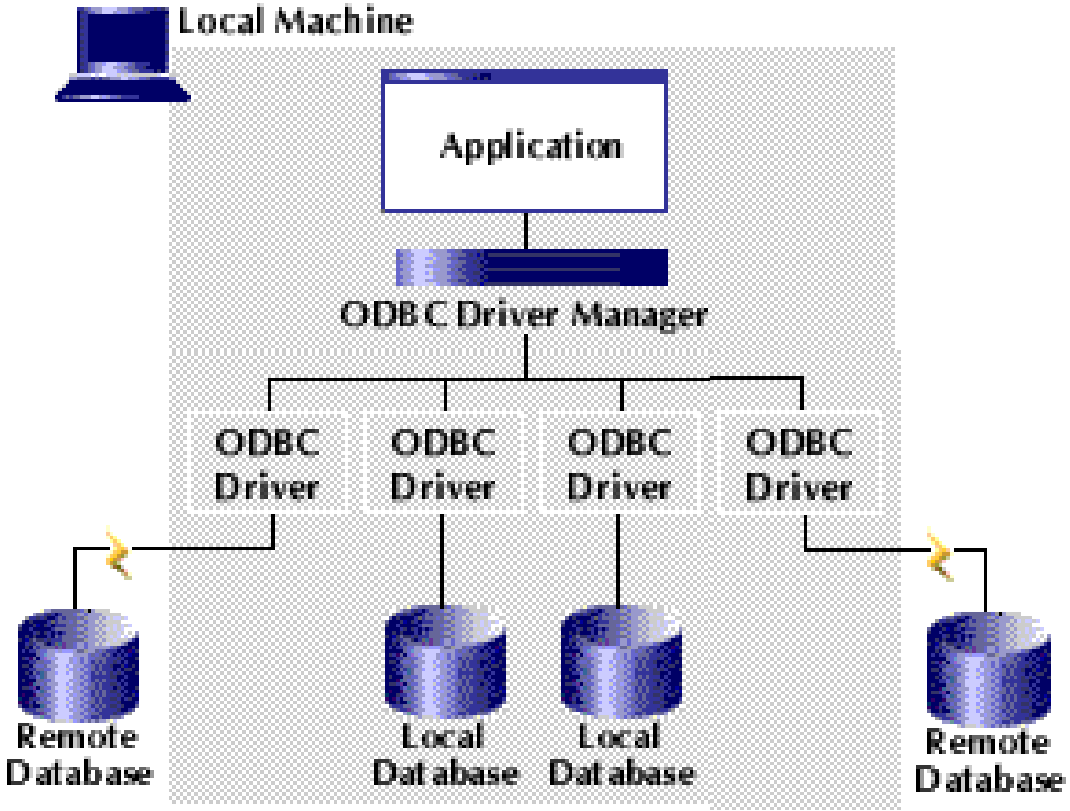
- **Connection protocol**
  - locate the RDBMS server
  - open a connection
  - user autentication
- **Querying**
  - query SQL
    - SELECT
    - UPDATE/INSERT/CREATE
  - stored procedures
  - prepared query SQL
- **Scan Result set**
  - scan row by row
  - access result meta-data



Connection Standards

- ODBC - Open DataBase Connectivity
  - Windows: odbc Linux: unixodbc, iodbc
  - Tabular Data
- JDBC
  - Java APIs for tabular data
- OLE DB (Microsoft)
  - Tabular data, XML, multi-dimensional data
- ADO (Microsoft)
  - Object-oriented API on top of OLE DB
- ADO.NET
  - Evolution of ADO in the .NET framework

ODBC Open DataBase Connectivity



DATA MANAGEMENT FOR BUSINESS INTELLIGENCE

ETL – Extract, Transform and Load

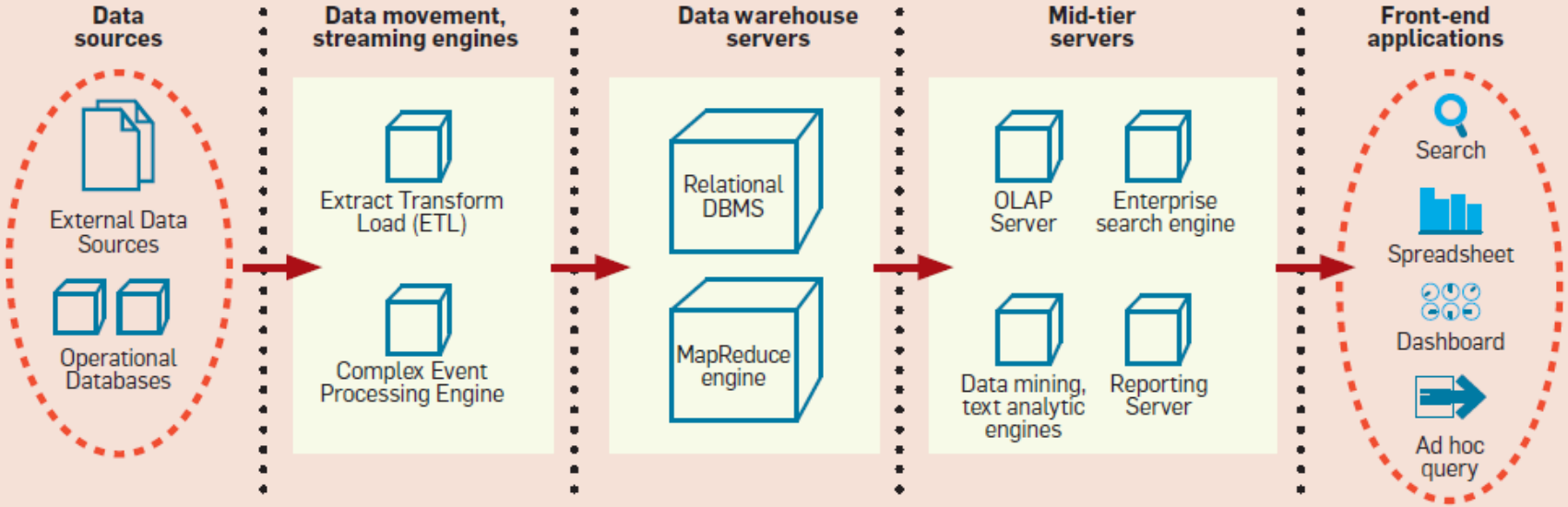
Salvatore Ruggieri

Computer Science Department, University of Pisa

Master in Big Data Analytics and Social Mining

BI Architecture

Figure 1. Typical business intelligence architecture.



Extract, Transform and Load

ETL (extract transform and load) is the process of extracting, transforming and loading data from heterogeneous sources in a data base/warehouse.

- Typically supported by (**visual**) tools.

No.	List of ETL Tools	Version	ETL Vendors
1.	Oracle Warehouse Builder (OWB)	11gR1	Oracle
2.	Data Services	XI 3.2	SAP Business Objects new!
3.	IBM Information Server (Datastage)	9.1	IBM
4.	SAS Data Integration Studio	4.21	SAS Institute new!
5.	PowerCenter	9.0	Informatica
6.	Elixir Repertoire	7.2.2	Elixir
7.	Data Migrator	7.7	Information Builders new!
8.	SQL Server Integration Services	10	Microsoft
9.	Talend Open Studio & Integration Suite	4.0	Talend
10.	DataFlow Manager	6.5	Pitney Bowes Business Insight
11.	Data Integrator	9.2	Pervasive
12.	Open Text Integration Center	7.1	Open Text
13.	Transformation Manager	4.1.4	ETL Solutions Ltd.
14.	Data Manager/Decision Stream	8.2	IBM (Cognos)
15.	Clover ETL	2.9.2	Javlin
16.	Centerprise	5.0	Astera new!
17.	DB2 Warehouse Edition	9.1	IBM
18.	Pentaho Data Integration	4.1	Pentaho
19	Adeptia Integration Suite	5.1	Adeptia

ETL tasks

- **Extract**: access data sources
  - Local, distributed, file format, connectivity standards
- **Transform**: data manipulation for quality improvm
  - Selecting data
    - remove unnecessary, duplicated, corrupted, out of limits (ex., age=999) rows and columns, sampling, dimensionality reduction
  - Missing data
    - fill with default, average, filter out
  - Coding and normalizing
    - to resolve format (ex., CSV, ARFF), measurement units (ex., meters vs inches), codes (ex., person id), times and dates, min-max norm, …
  - Attribute Splitting/merging
    - of attributes (ex., address vs street+city+country)

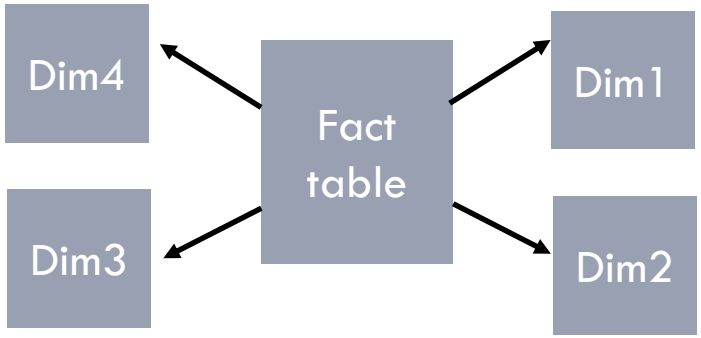
ETL tasks

- Managing surrogate key & Slowly changing dimensions
  - generation and lookup
- Aggregating data
  - At a different granularity. Ex., grain “orders” (id, qty, price) vs grain“customer” (id, no. orders, amount), discretization into bins, …
- Deriving calculated attributes
  - Ex., margin = sales – costs
- Resolving inconsistencies – record linkage
  - Ex., Dip. Informatica Via Buonarroti 2 is (?) Dip. Informatica Largo B. Pontecorvo 3
- Data merging-purging
  - from two or more sources (ex., sales database, stock database)

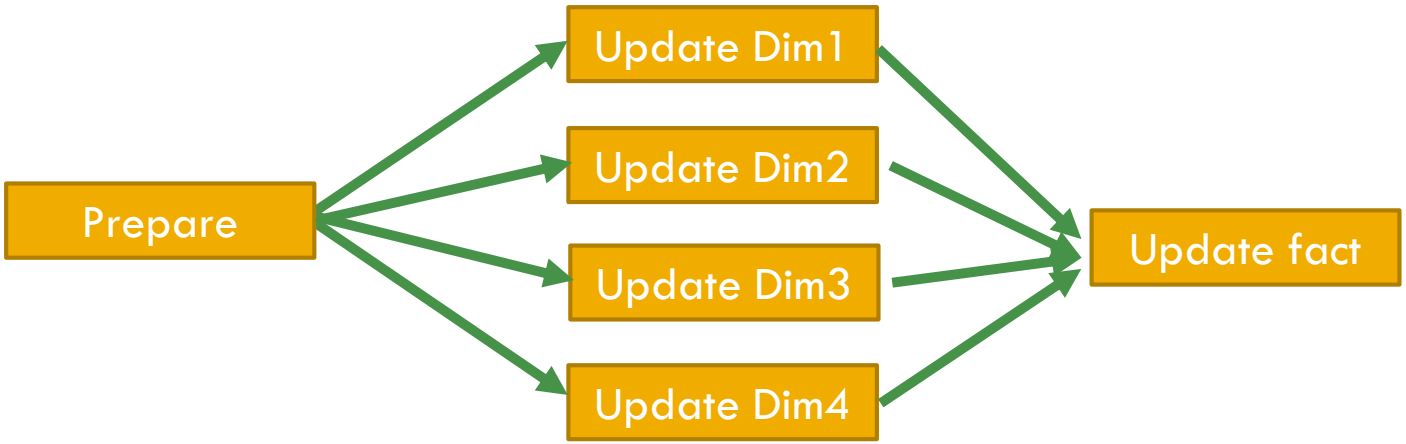
ETL tasks

- **Load**
  - Data staging area
    - Area containing intermediate, temporary, partially processed data
  - Types of loading:
    - Initial load (of the datawarehouse)
    - Incremental load
      - Types of updates: append, destructive merge, constructive merge
    - Full refresh

ETL process for DW

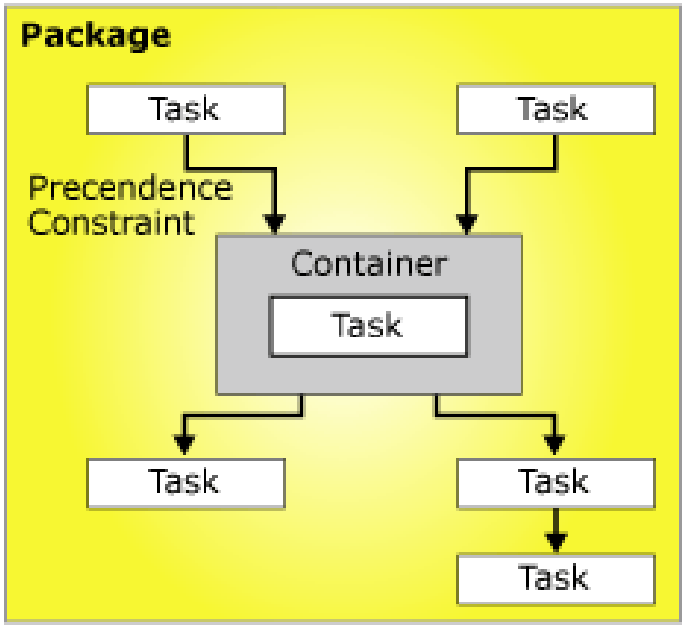


Control Flow



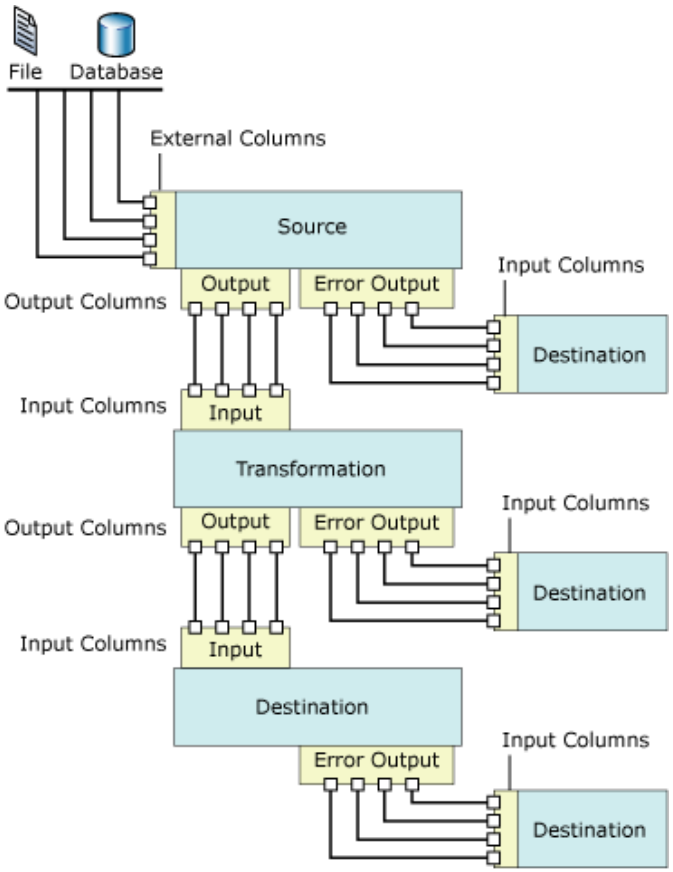
Control flow / Jobs

- Tasks & Precedence
  - Tasks
    - E.g. data flows / transformations
  - Container
    - For grouping and iteration
  - Precedence
    - Arrows connecting tasks specify precedence type

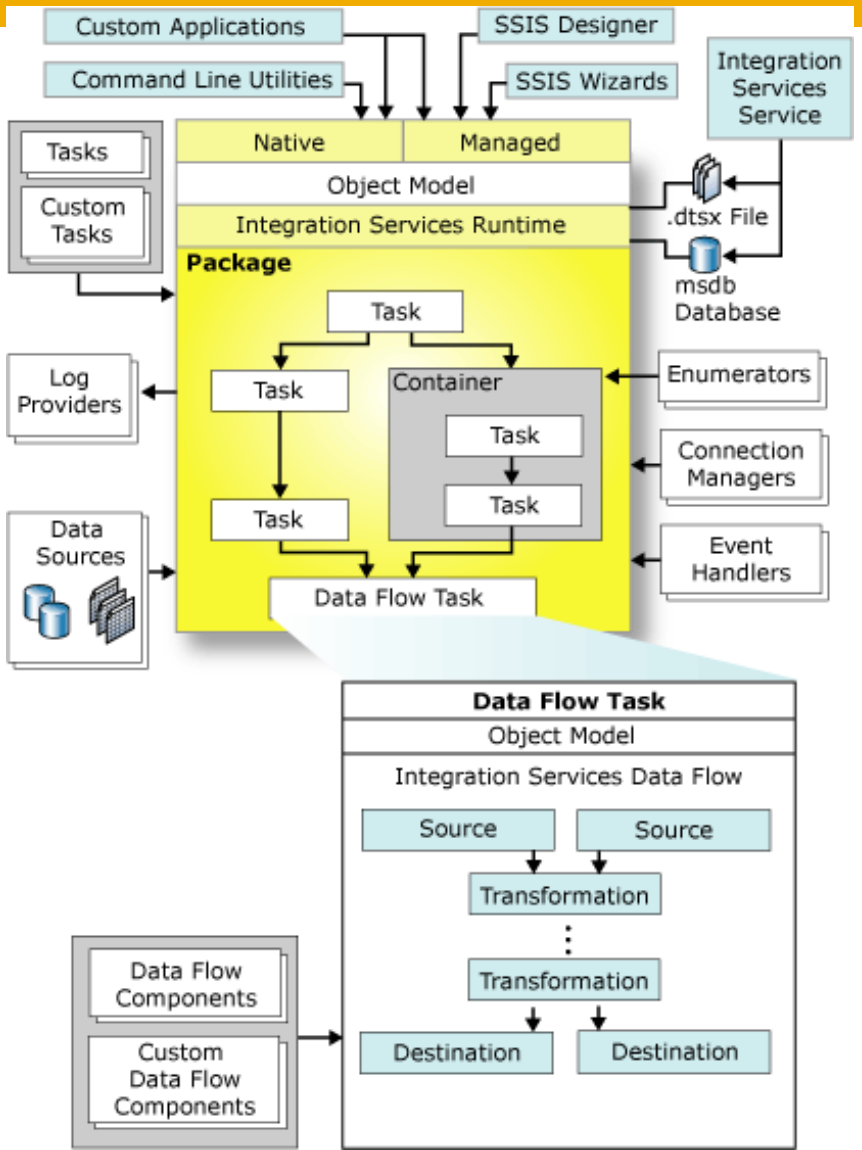


Special tasks: data flow / transformations

- Define pipelines of data flows from sources to destination
  - Data flow sources
  - Data flow transformation
  - Data destination
  - Toolbox panel for list



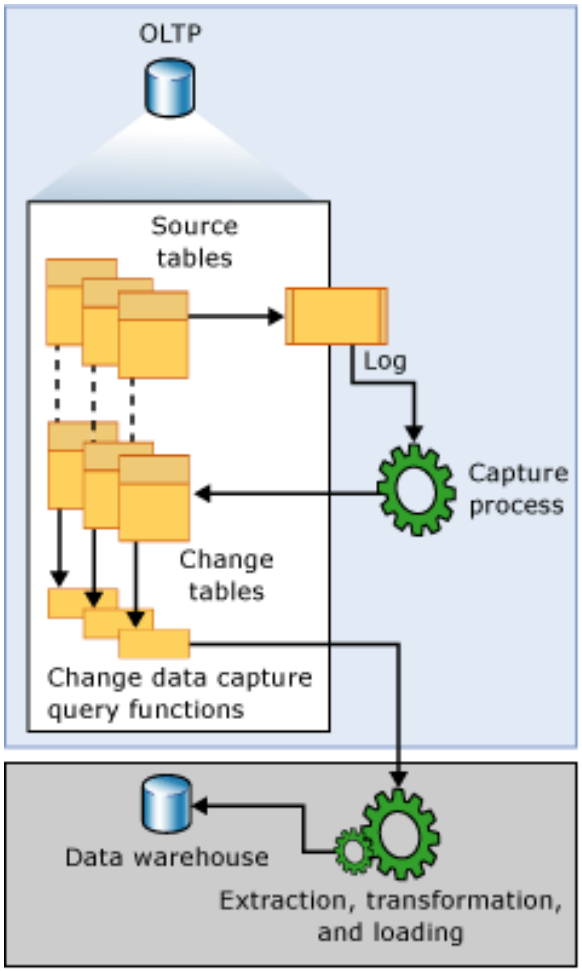
ETL projects structure



Data types

- ETL tools define a set of reference data types
- Data type from sources are mapped into ETL types
- ETL transformations work on ETL types
- ETL types are mapped to destination data types

Change data capture



BUSINESS INTELLIGENCE LABORATORY

ETL Demo: Pipeline, Sampling and Surrogate Keys

Pipeline

- Consider the Foodmart sales database
- Design an ETL project for writing to a CSV file the list of products ordered descending by gain
  - Gain of a single sale is defined as (store_sales – store_cost)*unit_sales
  - Gain of a product is the sum of gains for all product sales
- Do not use views or queries! Do all work in ETL.

Stratified subsampling

- Consider the census table in the MasterBigData db
- Design an ETL project for writing to a CSV a random sampling of 30% stratified by sex
  - 30% of males plus 30% of females
- Do not use views or queries! Do all work in ETL.

BUSINESS INTELLIGENCE LABORATORY

Lab exercise on ETL: SCD

SCD: background

- **Slowly Changing Dimensions**
  - Datawarehouse dimensions members updates
  - Three types:
    - Type 1: overwrite previous value
    - Type 2: keep all previous values
    - Type 3: keep last N previous values (N ~ 1, 2, 3)
  - Each attribute of the dimension can have its own type
    - Type 1: name, surname, …
    - Type 2: address, …

SCD: input and output tables

- Database FoodMart in MySQL
- Input
  - table **customer**
- Output in the MasterBigData database
  - create a table **<surname>_customer_dim**
    - columns
      - surrogate_key (PK), customer_id, customer_name, address, date_start, date_end
    - with
      - surrogate_key being a surrogate key, customer_name including name and surname, address made of address1-city-zip-province-country, date_start and date_end are dates

SCD: type 1 updates

- Overwrite previous value
- Changes on the input table **customer**
  - On 10/3/2007
    - 231, Mario Rosi, Via XXV Aprile Pisa
  - On 12/3/2007
    - 231, Mario Rossi, Via XXV Aprile Pisa
  - Surname has been corrected

SCD: type 1 updates

- The DW **<surname>_customer_dim** table looks as:
  - On 10/3/2007, and up to 12/3/2007

surrogate_key, customer_id, name, address, date_start, date_end

874, 231, Mario Rosi, Via XXV Aprile Pisa, 10/3/2007, NULL

  - On 12/3/2007

surrogate_key, customer_id, name, address, date_start, date_end

874, 231, Mario Rossi, Via XXV Aprile Pisa, 10/3/2007, NULL

SCD: type 2 updates

- Keep all previous values
- Changes on the input table **customer**
  - On 12/3/2007
    - 231, Mario Rossi, Via XXV Aprile Pisa
  - On 25/9/2008
    - 231, Mario Rossi, Via Risorgimento Pisa
  - Customer has changed his address

SCD: type 2 updates

- The DW **<surname>_customer_dim** table looks as:
  - On 12/3/2007, and up to 25/9/2008

surrogate_key, customer_id, name, address, date_start, date_end

874, 231, Mario Rossi, Via XXV Aprile Pisa, 10/3/2007, NULL

  - On 25/9/2008

surrogate_key, customer_id, name, address, date_start, date_end

870, 231, Mario Rossi, Via Roma Pisa, 1/1/2006, 10/3/2007

874, 231, Mario Rossi, Via XXV Aprile Pisa, 10/3/2007, 25/9/2008

987, 231, Mario Rossi, Via Risorgimento Pisa, 25/9/2008, NULL

Today exercise

- Design an ETL project to update **<surname>_customer_dim** starting from **customer** as follows:
  - Customers in **customer** that are not in **<surname>_customer_dim** are added to it
  - Updates of **customer_name** are of Type 1
  - Updates of **address** are of Type 2

Open lab

- A sale was done during a travel if the store of the sale was not in the city of residence of the customer. It is required to produce a CSV with a row for every customer and her/his percentage of sales done during travel over the total sales of the customer.

- The frequency of purchases in weekends (FPW) is the number of distinct weekend days (Saturdays or Sundays) in which the customer made a purchase. It is required to produce a CSV with a row for every customer his/her FPW.

Open lab

- The quarter value QV of a customer id C at the date id T is the sum of revenues minus costs of the products that the customer C buys in the quarter of T. It is required to produce a CSV le with three columns: customer id, time it, QV.

- A product id is female-specific if the number of distinct women customers who bought the product is at least 1.5 times the number of distinct male customers who bought it. It is required to produce a CSV with all female-specific product id's.