

Semi-structured Data

1 - Introduction

Outline

- Structured Data
- Semi-structured Data
- Why Semi-structured Data?
- The Data Model
- Store Semi-structured Data

Structured Data

- Data is structured in semantic chunks - **entities**

VIE, Vienna International, Vienna

LHR, London Heathrow, London

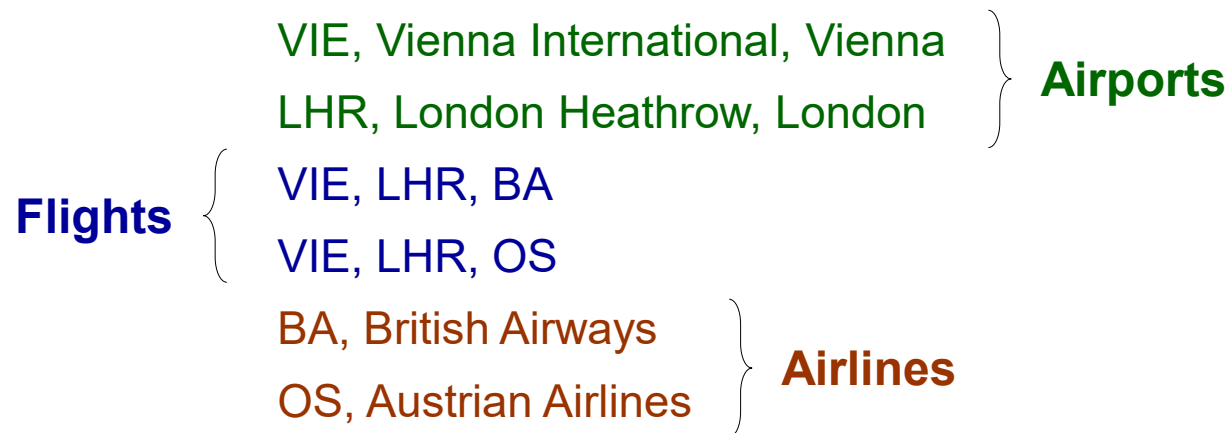
VIE, LHR, BA

VIE, LHR, OS

BA, British Airways

OS, Austrian Airlines

- Similar entities are grouped together - **classes**



Structured Data

- Entities in the same class have the same descriptions - **attributes**

Airports

(VIE, Vienna International, Vienna)

(LHR, London Heathrow, London)

(Airport_Code, Name, City)

Airlines

(BA, British Airways)

(OS, Austrian Airlines)

(Airline_Code, Name)

Flights

(VIE, LHR, BA)

(VIE, LHR, OS)

(Origin, Destination, Airline)

Structured Data

- Entities in the same class have the same descriptions - **attributes**

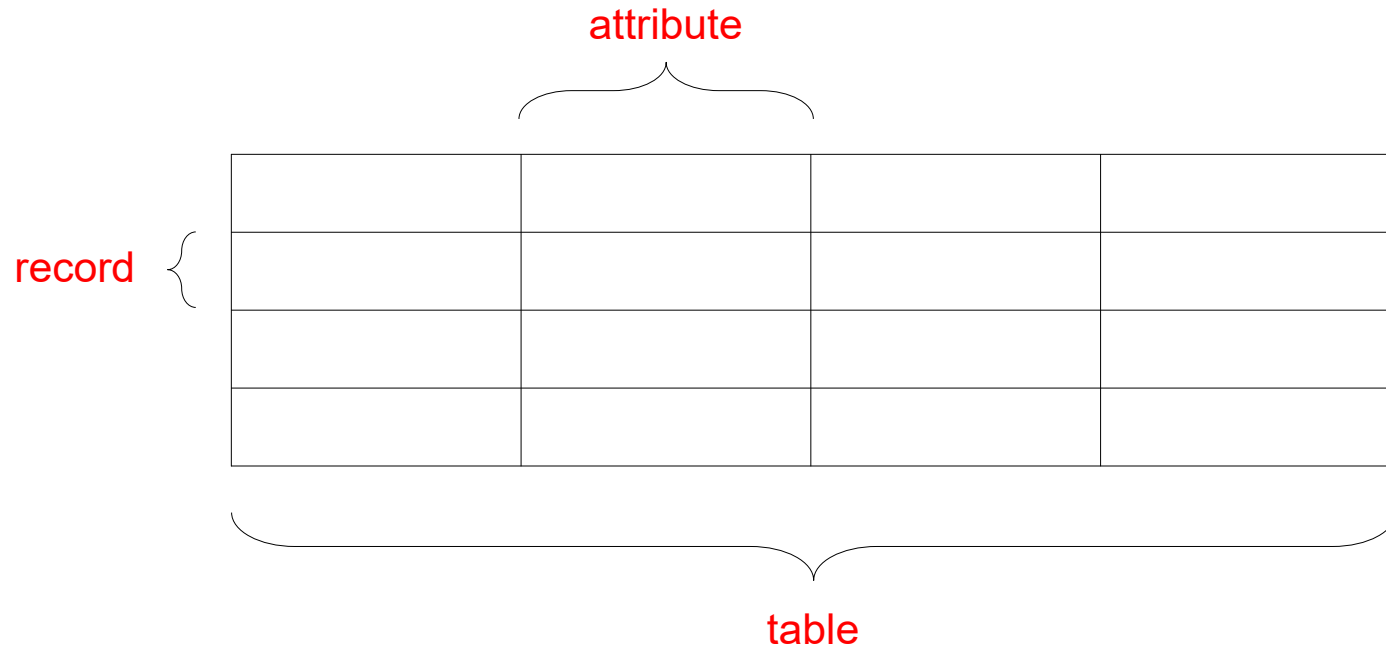
Airports	Flights	Airlines
(VIE, Vienna International, Vienna)	(VIE, LHR, BA)	(BA, British Airways)
(LHR, London Heathrow, London)	(VIE, LHR, OS)	(OS, Austrian Airlines)
(Airport_Code, Name, City)	(Origin, Destination, Airline)	(Airline_Code, Name)

- Attributes in similar entities {
 - same format (string, integer, date, etc.)
 - predefined length
 - all present
 - same order

... strict structure forced by a schema!!!

Structured Data - Relational Model

- Database model for structured data: entities → **records** (or tuples)
classes → **tables** (or relations)
- Records grouped in tables**



Structured Data: “On the Fly” Example

Airports	Code	Name	City
	VIE	Vienna International	Vienna
	LHR	London Heathrow	London
	LGW	London Gatwick	London
	LCA	Larnaca International	Larnaca
	GLA	Glasgow	Glasgow
	EDI	Edinburgh	Edinburgh

Airlines	Code	Name
	BA	British Airways
	OS	Austrian Airlines
	U2	EasyJet

Flights	Origin	Destination	Airline
	VIE	LHR	BA
	VIE	LHR	OS
	LHR	EDI	BA
	LGW	GLA	U2

“Persons” Example

Gerti Kappel, 18870, 18896, gerti@big.tuwien.ac.at

Wolfgang J., Dvořák, dvorak@dbai.tuwien.ac.at, 18441, 918441

Wolfgang Fischl, wfischl@dbai.tuwien.ac.at, 740050

Bill, Robert, 188316, bill@big.tuwien.ac.at

Semi-structured Data (SSD)

- Data is structured in semantic entities
- Similar entities are grouped in classes
- Entities in the same class may not have the same attributes
- Attributes of similar entities
 - may have different format
 - may have different length
 - not all required
 - may have different order

**there is
structure**

**but not
too much
structure**

Semi-structured Data: “Persons” Example

Gerti Kappel, 18870, 18896, gerti@big.tuwien.ac.at

Wolfgang J., Dvořák, dvorak@dbai.tuwien.ac.at, 18441, 918441

Wolfgang Fischl, wfischl@dbai.tuwien.ac.at, 740050

Bill, Robert, 188316, bill@big.tuwien.ac.at

- **There is structure**
 - Each row is a semantic entity - **person**
 - All entities are grouped in a class - **persons**
- **But not too much structure**
 - Entities have no regular structure
 - Structure of future entities is unpredictable

Why Semi-structured Data?

- There are data sources that we would like to treat as **databases**, but which **cannot be constraint by a schema**
- Flexible format for **exchanging data** between different places

... the WEB

GOAL: Reconcile document view (web) with strict structures (databases)

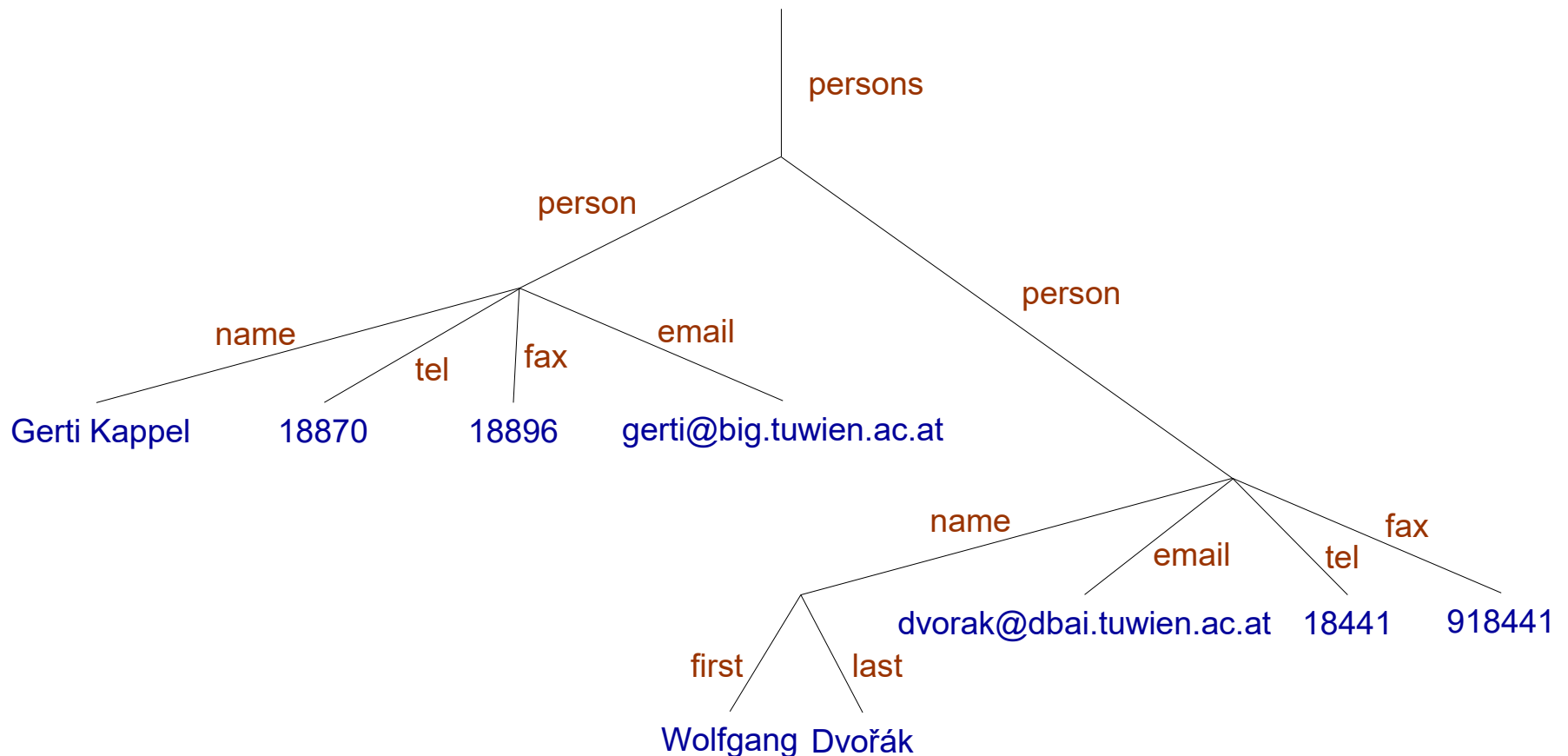
Data Model

- We need an effective way to represent semi-structured data
- Like the relational model for structured data

Trees as Data Model

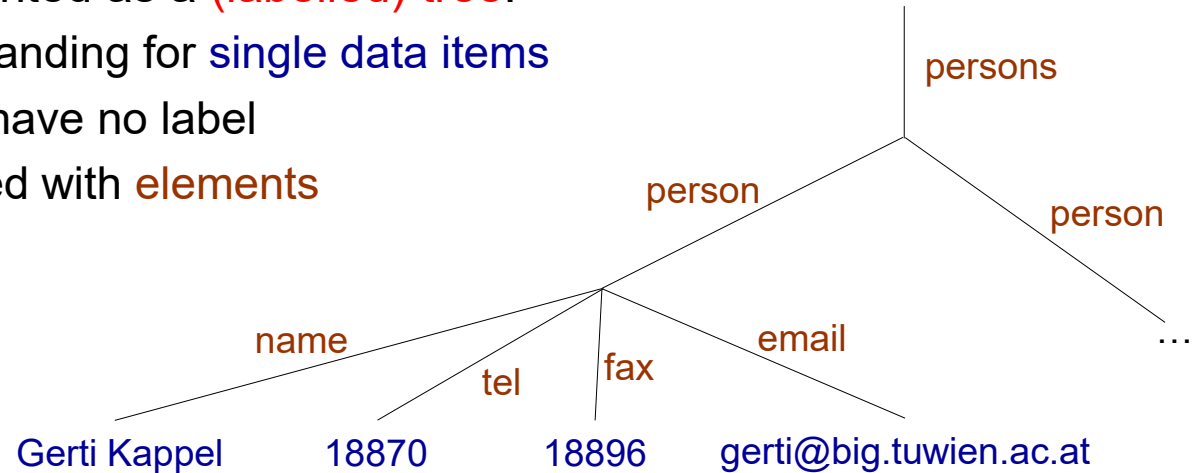
Gerti Kappel, 18870, 18896, gerti@big.tuwien.ac.at

Wolfgang, Dvořák, dvorak@dbai.tuwien.ac.at, 18441, 918441



Trees as Data Model

- SSD can be represented as a (labelled) tree:
 - leaf nodes standing for single data items
 - inner nodes have no label
 - edges labelled with elements

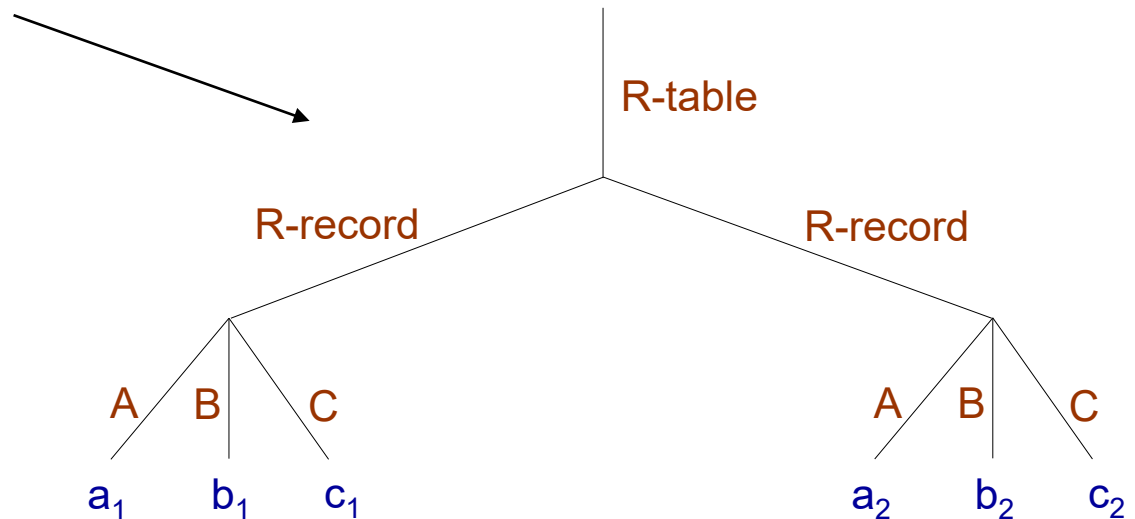


- Such a model is called self-describing - information that is usually associated with a schema is contained within the data
- Data carries its own description

SSD: Representing Relational Data

Structured data is a **special case** of semi-structured data
+
relational data can be represented as a tree (with an overhead)

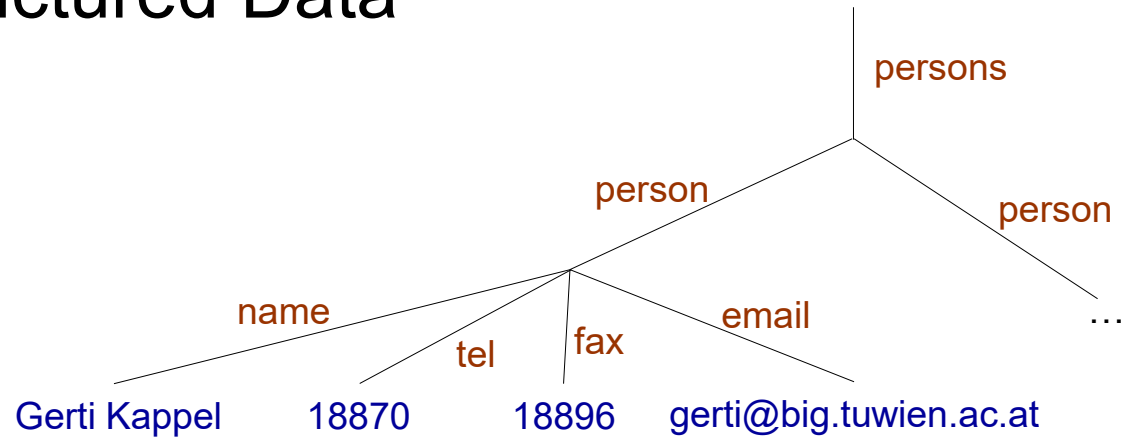
R	A	B	C
	a ₁	b ₁	c ₁
	a ₂	b ₂	c ₂



Store Semi-structured Data

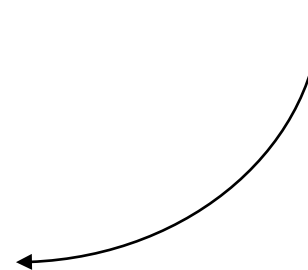
- There are **various formalisms** to store semi-structured data
 - Object Exchange Model (OEM)
 - JavaScript Object Notation (JSON)
 - eXtensible Markup Language (XML)

Store Semi-structured Data

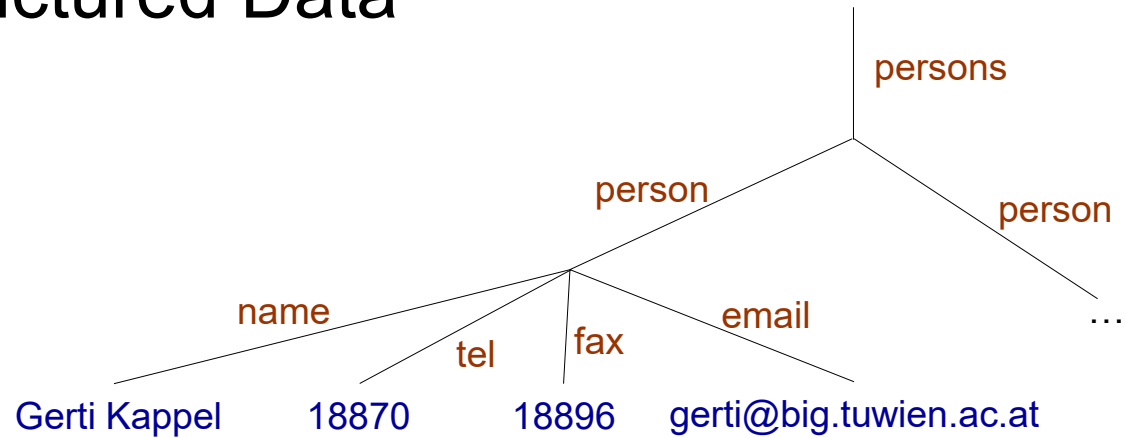


```
{persons:
  {person:
    {name: "Gerti Kappel"
      tel: 18870
      fax: 18896
      email: "gerti@big.tuwien.ac.at"}}
  {person:
    {name:
      {first: "Wolfgang",
       last: "Dvořák"}}
    email: "dvorak@dbai.tuwien.ac.at"
    tel: 18441
    fax: 918441}
}
```

OEM Representation

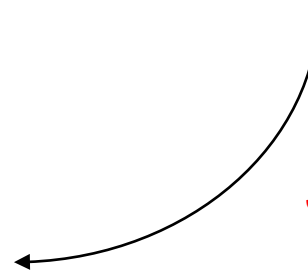


Store Semi-structured Data

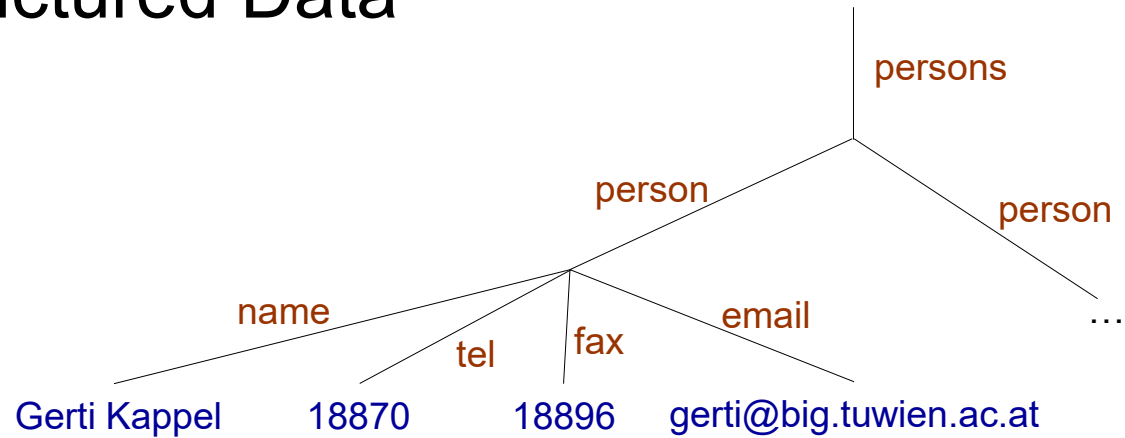


```
{ "persons":
  [ {"person":
    { "name": "Gerti Kappel",
      "tel": 18870,
      "fax": 18896,
      "email": "gerti@big.tuwien.ac.at"}
    },
    { "person":
      { "name":
        { "first": "Wolfgang",
          "last": "Dvořák"},
        "email": "dvorak@dbai.tuwien.ac.at",
        "tel": 18441,
        "fax": 918441}
      }
    ]
}
```

JSON Representation



Store Semi-structured Data



```
<persons>
  <person>
    <name> Gerti Kappel </name>
    <tel> 18870 </tel>
    <fax> 18896 </fax>
    <email> gerti@big.tuwien.ac.at </email>
  </person>
  <person>
    <name>
      <first> Wolfgang </first>
      <last> Dvořák </last>
    </name>
    <email> dvorak@dbai.tuwien.ac.at </email>
    <tel> 18441 </tel>
    <fax> 918441 </fax>
  </person>
</persons>
```

XML Representation

Store Semi-structured Data

- There are **various formalisms** to store semi-structured data
 - Object Exchange Model (OEM)
 - JavaScript Object Notation (JSON)
 - eXtensible Markup Language (XML)
- Different syntax
- Different mechanisms for self-describing
- Different description mechanisms
 - Which attributes are allowed/required
 - Which values are allowed/required
- Different query languages and manipulation mechanisms

but the goal is the same:

store SSD

Sum Up

- Structured Data
 - Similar entities grouped in classes
 - Similar entities have a regular structure
 - Relational Model
- Semi-structured Data
 - Similar entities grouped in classes
 - Similar entities have irregular structure
 - Trees as a Model
- Store Semi-structured Data
 - Various formalisms
 - eXtensible Markup Language (XML)