2019: A Data Odyssey × Real World W01-25

Data Modeling: Real World Entity
Conceptual Level |
| Logical Level |
| Physical Level
Computable Entity

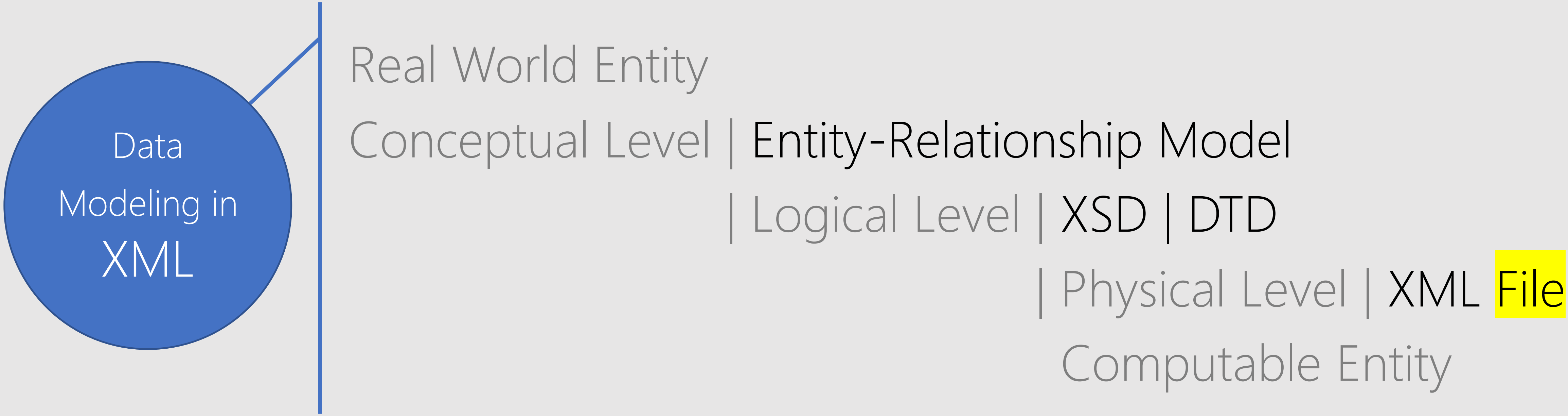
Data Modeling × Logical Level

There are other representations as well.
There are other logical models as well.
There are other data models at logical level as well.

~1960: Object Oriented
~1969: Relational: Mathematical Relation~~ships~~
~1996: XML

2019: A Data Odyssey × XML

1



Data Modeling in XML

Real World Entity

Conceptual Level | Entity-Relationship Model

| Logical Level | XSD | DTD

| Physical Level | XML File

Computable Entity

Data Modeling × XML 2

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director>
                <Id>1</Id>
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

Data Modeling × Relational DBMS

Movie			
Id	Title	RunningTime	ReleaseDate
1	*Inception*	*148*	*2010*

Director				
Id	FirstName	LastName	DateOfBirth	PlaceOfBirth
1	*Christopher*	*Nolan*	*1970-06-30*	*England*

MovieGenre	
MovieId	GenreId
1	*6*
1	*1*

MovieDirector	
MovieId	DirectorId
1	*1*

Genre	
Id	Title
1	*Sci-fi*
2	*Drama*
3	*Crime*
4	*Mystery*
5	*Thriller*
6	*Adventure*
7	*Horror*
8	*Action*

XML × Markup Language

Inception is a 2010 science fiction action and adventure film directed by Christopher Nolan. The film is 148 minutes long. Christopher Nolan (born 30 July 1970) is an English filmmaker

XML × Markup Language

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director>
                <Id>1</Id>
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

XML × Markup Language

A markup language is a system for *annotating* (tagging) a document in a way that is syntactically distinguishable from the text.

Machine-readable

Markup languages include HTML, XML, XHTML

HTML × HyperText

In HTML (HyperText Markup Language), tags are standardised by W3C so that all web browsers have same presentation. E.g.,

```
<a href="http://ls3.rnet.ryerson.ca">Visit our lab!</a>
<img src="smiley.gif" height="42" width="42">
<p>this a paragraph</p>
```

The World Wide Web Consortium (W3C) is an international community that develops open standards to ensure the long-term growth of the Web

XML × eXtensible

Unlike HTML, XML is able to be extended to new user-defined tags! E.g.,

```
<Movie></Movie>
<Director></Director>
<Id></Id>
<Title></Title>
<Genre></Genre>
```

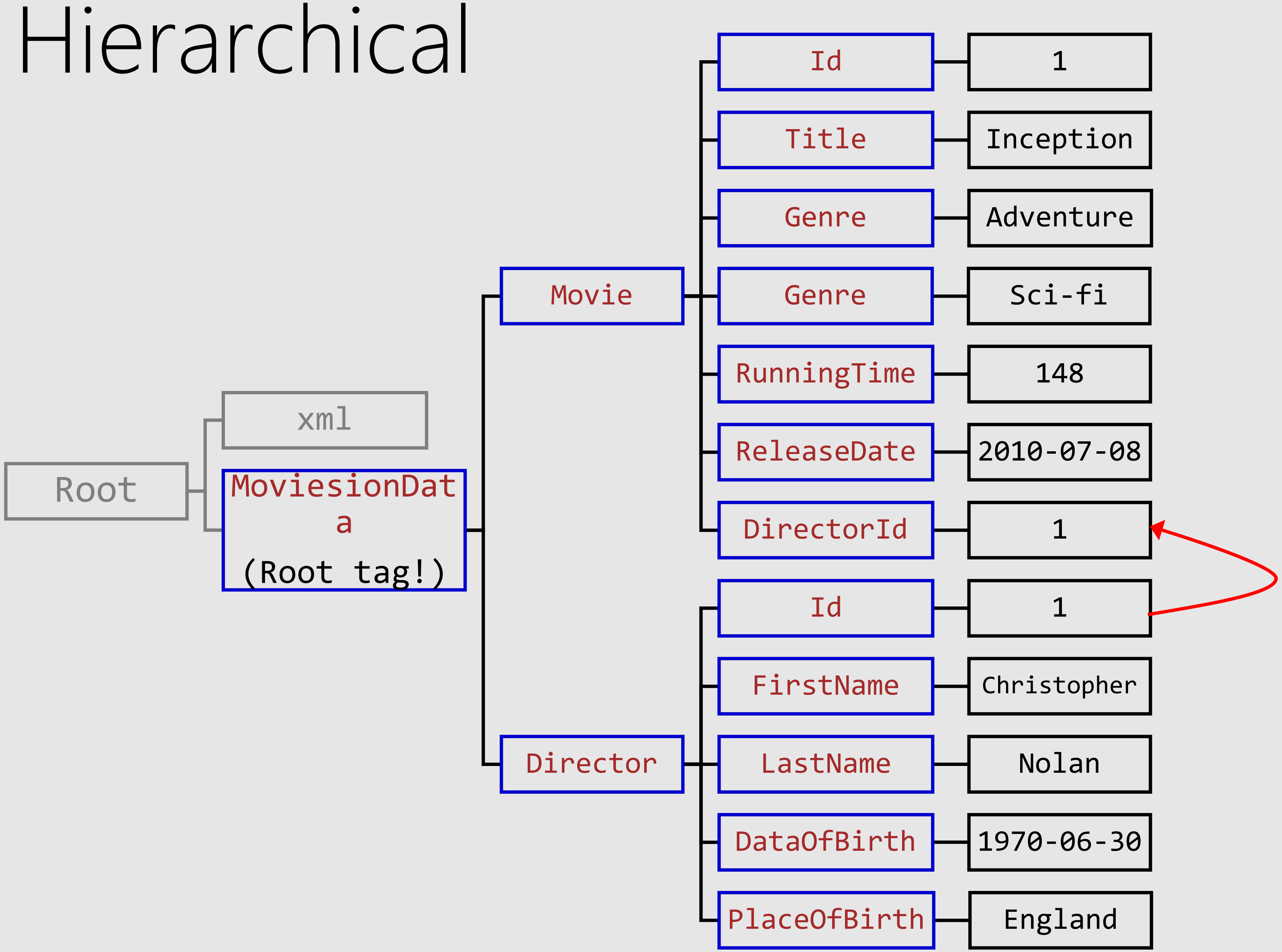
XML × eXtensible

Unlike HTML, XML's tag can be human-readable! E.g.,

```
<Movie></Movie>
<Director></Director>
<Id></Id>
<Title></Title>
<Genre></Genre>
```

As a result, XML is self-explanatory.

XML × Hierarchical



XML × Anatomy

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director>
                <Id>1</Id>
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

XML × Element

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director>
                <Id>1</Id>
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

Element 01

XML × Element

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director>
                <Id>1</Id>
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

Element 02

Element 03

XML × Element

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>                                                  Element 04
                <Title>Inception</Title>                                    Element 05
                <Genre>Adventure</Genre>                                    Element 06
                <Genre>Sci-fi</Genre>                                       Element 07
                <RunningTime>148</RunningTime>                              Element 08
                <ReleaseDate>2010-07-08</ReleaseDate>                       Element 09
                <DirectorId>1</DirectorId>                                  Element 10
        </Movie>
        <Director>
                <Id>1</Id>                                                  Element 11
                <FirstName>Christopher</FirstName>                          Element 12
                <LastName>Nolan</LastName>                                  Element 13
                <DateOfBirth>1970-06-30</DateOfBirth>                       Element 14
                <PlaceOfBirth>England</PlaceOfBirth>                        Element 15
        </Director>
</MoviesionData>
```

XML × Element

- Made up of a opening tag and an end tag with same name

```
<Title>Inception</Title>
```

- May include properly nested elements

```
<Movie>
      <Id>1</Id>
      <Title>Inception</Title>
</Movie>
```

```
<Movie>
      <Id>1</Id>
      <Title>Inception
</Movie></Title>
```

Improperly Nested! NOT well-formed!

XML × Element

- Tags are case-sensitive

```
<Title></Title> ≠ <title></title> ≠ <TiTle></TiTle>
```

- Empty elements can be abbreviated

```
<Title></Title> = <Title/>
```

- Multiple elements may have same name

```
<Movie>
      <Genre>Adventure</Genre>
      <Genre>Sci-fi</Genre>
</Movie>
```

- Order matters

XML × Attribute

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director>
                <Id>1</Id>
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

XML × Attribute

- An element may have multiple {key}="{value}" in opening tag, called attributes.

- Per element attribute's key must be unique.

- Attributes cannot be nested.

XML × Attribute vs. Element

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie>
                <Id>1</Id>
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director>
                <Id>1</Id>
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

XML × Attribute vs. Element

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie Id="1" Title="Inception">
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Director Id="1">
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

XML × Attribute vs. Element

Attributes are part of markup (tag), however, elements are part of the contents (data instances).

Representing elements as attributes are NOT encouraged.

Rules always have exceptions:
It is encouraged to put PK elements like Id as attributes.
This way the XML element can be uniquely accessible by Id attribute.

XML × Semi-structured

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie Id="1">
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
                <DirectorId>1</DirectorId>
        </Movie>
        <Movie Id="2">
                <Title>Inception</Title>
                <ReleaseDate>2010-07-08</ReleaseDate>
        </Movie>
</MoviesionData>
```

✓

Well-formed

XML × Schema

To do DML on XML document, we need to parse it.
To do DML on XML document, we need to know minimum structure.
To do DML on XML document, we have to preserve data integrity.
To do DML on XML document, what are legal elements & attributes.
E.g.,

- MoviesionData element has Movie and Director elements
- Movie must have Id element, but may have Title, . . .
- Should Movie element has DirectorId element, the values must be from Id element of Director elements, . . .

To enforce a structure to an XML document, we need to define schema.

XML × Schema

DTD: Document Type Definition

Old, but widely used

XSD: XML Schema Definition

Recommended by World Wide Web Consortium (W3C)

An XML document might be well-formed, but not valid against a defined schema.

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE {root element}[
    <!ELEMENT {root element} ({nested elements})>
    <!ELEMENT {subelement} (...)>
    <!ELEMENT ...>
]>

<{root element}>
    <{subelement}>
        ...
    </{subelement}>
</{root element}>
```

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
]>
```

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
]>
```

Regular Expression for element's content:

| Either **`Movie`** or **`Director`**

+ Minimum One Occurrence of a **`Movie`** or a **`Director`**

* Zero or More Occurrences

? Zero or None

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
]>
```

XML document is valid if it has:

A) No movie but one director
B) One movie but no director
C) Two movies and then three directors
D) Two movies, three directors, and one last movie
E) 100 movies only
F) 100 directors only
G) Empty, i.e., no movie and no director

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
]>
```

XML document is valid if it has:

A) No movie but one director
B) One movie but no director
C) Two movies and then three directors
D) Two movies, three directors, and one last movie
E) 100 movies only
F) 100 directors only
G) Empty, i.e., no movie and no director

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
        <!ELEMENT MoviesionData (Movie+ | Director*)>
]>
```

XML document is valid if it has:

A) No movie but one director
B) One movie but no director
C) Two movies and then three directors
D) Two movies, three directors, and one last movie
E) 100 movies only
F) 100 directors only
G) Empty, i.e., no movie and no director

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
        <!ELEMENT MoviesionData (Movie+ | Director*)>
]>
```

XML document is valid if it has:

A) No movie but one director

B) One movie but no director

C) Two movies and then three directors

D) Two movies, three directors, and one last movie

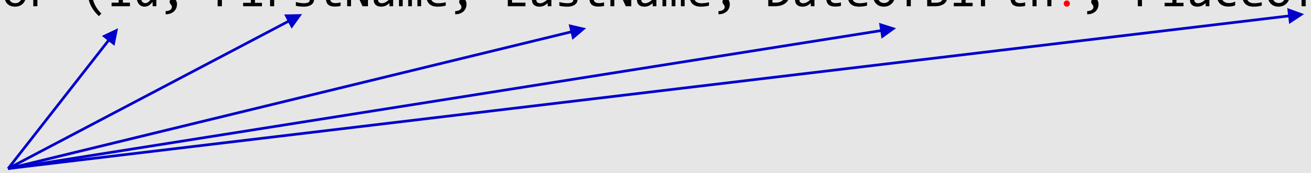
E) 100 movies only

F) 100 directors only

G) Empty, i.e., no movie and no director

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
        <!ELEMENT Movie (Id, Title, Genre*, RunningTime?, ReleaseDate?, DirectorId+)>
        <!ELEMENT Director (Id, FirstName, LastName, DateOfBirth?, PlaceOfBirth?)>
]>
```



Order matters!

XML × DTD × !ELEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
        <!ELEMENT Movie (Id, Title, Genre*, RunningTime?, ReleaseDate?, DirectorId+)>
        <!ELEMENT Director (Id, FirstName, LastName, DateOfBirth?, PlaceOfBirth?)>
        <!ELEMENT Id (#PCDATA)>
        <!ELEMENT Title (#PCDATA)>
        <!ELEMENT Genre (#PCDATA)>
        <!ELEMENT RunningTime (#PCDATA)>
        <!ELEMENT ReleaseDate (#PCDATA)>
        <!ELEMENT DirectorId (#PCDATA)>
        <!ELEMENT FirstName (#PCDATA)>
        <!ELEMENT LastName (#PCDATA)>
        <!ELEMENT DateOfBirth (#PCDATA)>
        <!ELEMENT PlaceOfBirth (#PCDATA)>
]>
```

#PCDATA Parsed Character DATA is the only data type in DTD!

XML × DTD × !ATTLIST

Declare the attribute list of an element

```
<!ATTLIST  {element} {key} {type} #REQUIRED|#IMPLIED|#FIXED "{default value}">
```

{type}:	CDATA	The value is character data
	(enum1|enum2|...)	The value must be one from an enumerated list
	ID	The value is a unique id
	IDREF	The value is the id of another element
	IDREFS	The value is a list of other ids
	…	

XML × DTD × !ATTLIST

```
<!ELEMENT Movie (Id, Title, Genre*, RunningTime?, ReleaseDate?, DirectorId+)>
<!ATTLIST Movie Company CDATA #REQUIRED>

✓ <Movie Company="Warner Bros">
       <Id>1</Id>
       <Title>Inception</Title>
       <Genre>Adventure</Genre>
       <DirectorId>1</DirectorId>
  </Movie>
✗ <Movie>
       <Id>2</Id>
       <Title>Memento</Title>
       <Genre>Mystery</Genre>
       <DirectorId>1</DirectorId>
  </Movie>
```

XML × DTD × !ATTLIST

```
<!ELEMENT Movie (Id, Title, Genre*, RunningTime?, ReleaseDate?, DirectorId+)>
<!ATTLIST Movie Company CDATA #IMPLIED>

✓ <Movie Company="Warner Bros">
       <Id>1</Id>
       <Title>Inception</Title>
       <Genre>Adventure</Genre>
       <DirectorId>1</DirectorId>
  </Movie>
✓ <Movie>
       <Id>2</Id>
       <Title>Memento</Title>
       <Genre>Mystery</Genre>
       <DirectorId>1</DirectorId>
  </Movie>
```

XML × DTD × !ATTLIST

```
<!ELEMENT Movie (Id, Title, Genre*, RunningTime?, ReleaseDate?, DirectorId+)>
<!ATTLIST Movie Company CDATA #FIXED "Warner Bros">

✓<Movie Company="Warner Bros">
       <Id>1</Id>
       <Title>Inception</Title>
       <Genre>Adventure</Genre>
       <DirectorId>1</DirectorId>
  </Movie>
✗ <Movie Company="Newmarket">
       <Id>2</Id>
       <Title>Memento</Title>
       <Genre>Mystery</Genre>
       <DirectorId>1</DirectorId>
  </Movie>
```

XML × DTD × !ATTLIST

```
<!ELEMENT Movie (Id, Title, Genre*, RunningTime?, ReleaseDate?, DirectorId+)>
<!ATTLIST Movie Company (Warner Bros|Newmarket) "Warner Bros">
```

```
✓<Movie Company="Warner Bros">
       <Id>1</Id>
       <Title>Inception</Title>
       <Genre>Adventure</Genre>
       <DirectorId>1</DirectorId>
 </Movie>
✓ <Movie Company="Newmarket">
       <Id>2</Id>
       <Title>Memento</Title>
       <Genre>Mystery</Genre>
       <DirectorId>1</DirectorId>
 </Movie>
✓ <Movie>
       <Id>3</Id>
       <Title>2001: A Space Odyssey</Title>
       <Genre>Sci-fi</Genre>
       <DirectorId>10</DirectorId>
 </Movie>
```

XML × DTD × !ATTLIST × ID

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData[
        <!ELEMENT MoviesionData (Movie | Director)+>
        <!ELEMENT Movie (~~Id~~, Title, Genre*, RunningTime?, ReleaseDate?~~, DirectorId+~~)>
        <!ATTLIST Movie Id ID #REQUIRED>
        <!ATTLIST Movie DirectorId IDREFS #IMPLIED>
        <!ELEMENT Director (~~Id~~, FirstName, LastName, DateOfBirth?, PlaceOfBirth?)>
        <!ATTLIST Director Id ID #REQUIRED>
        <!ELEMENT Title (#PCDATA)>
        <!ELEMENT Genre (#PCDATA)>
        <!ELEMENT RunningTime (#PCDATA)>
        <!ELEMENT ReleaseDate (#PCDATA)>
        <!ELEMENT FirstName (#PCDATA)>
        <!ELEMENT LastName (#PCDATA)>
        <!ELEMENT DateOfBirth (#PCDATA)>
        <!ELEMENT PlaceOfBirth (#PCDATA)>
]>
```

XML × DTD × !ATTLIST × ID

```
...DTD...
<MoviesionData>
        <Movie Id="1" DirectorId="1">
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
        </Movie>
        <Director Id="1">
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

✓

Well-formed

✗

No valid!

XML × DTD × !ATTLIST × ID

- Must be unique across all the ID values in the XML document, ignoring the elements name!

- Has to start with a letter or ':' or '_'. It cannot be numbers only!

- An attribute declared of type ID cannot be defined as either #FIXED.

XML × DTD × !ATTLIST × ID

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData>
        <Movie Id="M1" DirectorId="D1">
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
        </Movie>
        <Director Id="D1">
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

✓

Well-formed

✓

Valid

XML × DTD × External

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MoviesionData SYSTEM "MoviesionData.dtd">
<MoviesionData>
        <Movie Id="M1" DirectorId="D1">
                <Title>Inception</Title>
                <Genre>Adventure</Genre>
                <Genre>Sci-fi</Genre>
                <RunningTime>148</RunningTime>
                <ReleaseDate>2010-07-08</ReleaseDate>
        </Movie>
        <Director Id="D1">
                <FirstName>Christopher</FirstName>
                <LastName>Nolan</LastName>
                <DateOfBirth>1970-06-30</DateOfBirth>
                <PlaceOfBirth>England</PlaceOfBirth>
        </Director>
</MoviesionData>
```

XML × DTD × Limitations

- Doesn't have datatypes. Only #PCDATA

XML × DTD × Limitations

- Doesn't have datatypes. Only #PCDATA
- IDs aren't a good implementation of keys (Why?)

XML × DTD × Limitations

- Doesn't have datatypes. Only #PCDATA
- IDs aren't a good implementation of keys (Why?)
- IDs, IDREF, and IDREFS are untyped

  E.g., `<Movie Id="M1" DirectorId="M2">`

XML × DTD × Limitations

- Doesn't have datatypes. Only #PCDATA
- IDs aren't a good implementation of keys (Why?)
- IDs, IDREF, and IDREFS are untyped

  E.g., `<Movie Id="M1" DirectorId="M2">`
- Not exactly following XML syntax

XML × XSD

XML Schema (Definition) is a more sophisticated schema language which addresses the drawbacks of DTDs.

+ Follows XML syntax, i.e., itself is an XML document
+ Built-in datatypes: integer, string, date, …
+ User-defined datatypes: complex types
+ Constraints on min/max values
+ Constraints on uniqueness
+ Constraints on foreign key constraints
+ Inheritance
- Significantly more complicated than DTDs.

XML × XSD

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
</xs:schema>
```

`xmlns:xs="{ns}"` the element names (tags) are from `{ns}` namespace, e.g., World Wide Web Consortium (W3C)

`xs` is a name, could be any name, to refer to `{ns}`,
e.g., `xs:schema` means that the element schema is from `xs` namespace
which is actually refer to `{ns}` namespace.

XML × XSD

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="MoviesionData">
    <xs:complexType>
     <xs:sequence>
       <xs:element minOccurs="0" maxOccurs="unbounded" ref="Movie"/>
       <xs:element minOccurs="0" maxOccurs="unbounded" ref="Director"/>
     </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Like * in DTD

Defining the root element.

e.g., **`MoviesionData`** is a complex type and has sequence of **`Movie`** and **`Director`**

Order matters. Movies come first.

XML × XSD

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
...
  <xs:element name="Movie">
   <xs:complexType>
     <xs:sequence>
       <xs:element name="Title" type="xs:string"/>
       <xs:element name="Genre" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
       <xs:element name="RunningTime" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
       <xs:element name="ReleaseDate" type="xs:date" minOccurs="0" maxOccurs="1"/>
     </xs:sequence>
     <xs:attribute name="Id" type="xs:integer" use="required"/>
     <xs:attribute name="DirectorId" type="xs:integer" use="optional"/>
   </xs:complexType>
  </xs:element>
</xs:schema>
```

Like ? in DTD

Defining the **`Movie`** element.

XML × XSD

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
...
  <xs:element name="Director">
   <xs:complexType>
     <xs:sequence>
       <xs:element name="FirstName" type="xs:string"/>
       <xs:element name="LastName" type="xs:string"/>
       <xs:element name="DateOfBirth" type="xs:date" minOccurs="0" maxOccurs="1"/>
       <xs:element name="PlaceOfBirth" type="xs:string" minOccurs="0" maxOccurs="1"/>
     </xs:sequence>
     <xs:attribute name="Id" type="xs:integer" use="required"/>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Defining the **`Director`** element.

XML × XSD

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="MoviesionData">
    <xs:complexType>
     <xs:sequence>
       <xs:element maxOccurs="unbounded" minOccurs="0" ref="Movie"/>
       <xs:element maxOccurs="unbounded" minOccurs="0" ref="Director"/>
     </xs:sequence>
   </xs:complexType>
  </xs:element>
<xs:element name="Movie">
   <xs:complexType>
     <xs:sequence>
       <xs:element name="Title" type="xs:string"/>
       <xs:element name="Genre" type="xs:string" maxOccurs="unbounded" minOccurs="0"/>
       <xs:element name="RunningTime" type="xs:decimal" minOccurs="0"/>
       <xs:element name="ReleaseDate" type="xs:date" minOccurs="0"/>
     </xs:sequence>
     <xs:attribute name="Id" type="xs:integer" use="required"/>
     <xs:attribute name="DirectorId" type="xs:integer" use="required"/>
   </xs:complexType>
  </xs:element>
<xs:element name="Director">
   <xs:complexType>
     <xs:sequence>
       <xs:element name="FirstName" type="xs:string"/>
       <xs:element name="LastName" type="xs:string"/>
       <xs:element name="DateOfBirth" type="xs:date" minOccurs="0"/>
       <xs:element name="PlaceOfBirth" type="xs:string" minOccurs="0"/>
     </xs:sequence>
     <xs:attribute name="Id" type="xs:integer" use="required"/>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

XML × XSD × External

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
               xs:noNamespaceSchemaLocation="MoviesionData.xsd">
       <Movie Id="1" DirectorId="1">
               <Title>Inception</Title>
               <Genre>Adventure</Genre>
               <Genre>Sci-fi</Genre>
               <RunningTime>148</RunningTime>
               <ReleaseDate>2010-07-08</ReleaseDate>
       </Movie>
       <Director Id="1">
               <FirstName>Christopher</FirstName>
               <LastName>Nolan</LastName>
               <DateOfBirth>1970-06-30</DateOfBirth>
               <PlaceOfBirth>England</PlaceOfBirth>
       </Director>
</MoviesionData>
```

XML × XSD × key

```
<xs:element name="MoviesionData">
...
  <xs:key name="PK_Movie_Id">
    <xs:selector xpath="Movie"/>
    <xs:field xpath="@Id"/>
  </xs:key>
  <xs:key name="PK_Director_Id">
    <xs:selector xpath="Director"/>
    <xs:field xpath="@Id"/>
  </xs:key>
</xs:element>
```

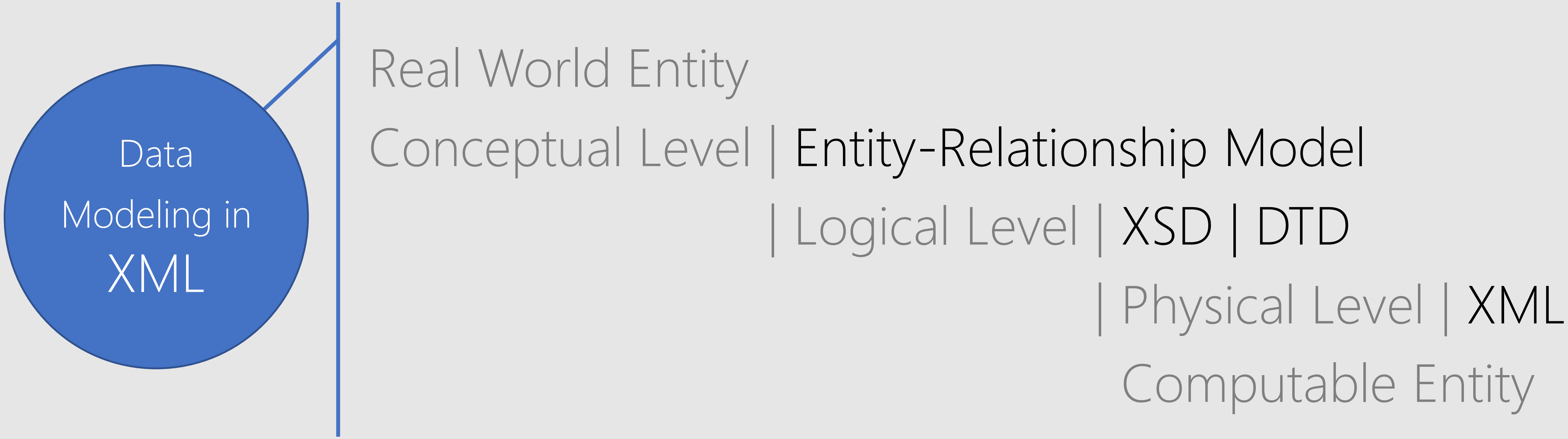
Only check uniqueness for **`Id`** attribute in **`Movie`** elements

Only check uniqueness for **`Id`** attribute in **`Director`** elements

XML × XSD × keyref

```
<xs:element name="MoviesionData">
 ...
 <xs:key name="PK_Movie_Id">
    <xs:selector xpath="Movie"/>
    <xs:field xpath="@Id"/>
  </xs:key>
  <xs:key name="PK_Director_Id">
    <xs:selector xpath="Director"/>
    <xs:field xpath="@Id"/>
  </xs:key>
  <xs:keyref name="FK_Movie_DirectorId_2_Director_Id" refer="PK_Director_Id">
    <xs:selector xpath="Movie"/>
    <xs:field xpath="@DirectorId"/>
  </xs:keyref>
</xs:element>
```

2019: A Data Odyssey × XML



Data Modeling in XML

Real World Entity

Conceptual Level | Entity-Relationship Model

| Logical Level | XSD | DTD

| Physical Level | XML

Computable Entity

XML × DML

Different query languages have been proposed for XML:

- XPath
- XQuery
- XQL
- XML-QL

XML × XPath

How do you organize your files in a PC?
How do you find your files in a PC?

XML × XPath

XPath looks like a path in a file system:

```
/
/hossein
./lab1
../../coe848
```

Instead of files, it returns set of elements and subelements.

→ https://www.w3schools.com/xml/xpath_syntax.asp

XML × XPath × SELECT

"/"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
          <Movie Id="1" DirectorId="1">
                    <Title>Inception</Title>
                    <Genre>Adventure</Genre>
                    <Genre>Sci-fi</Genre>
                    <RunningTime>148</RunningTime>
                    <ReleaseDate>2010-07-08</ReleaseDate>
          </Movie>
          <Movie Id="2" DirectorId="10">
                    <Title>Avatar</Title>
                    <Genre>Mystery</Genre>
                    <Genre>Sci-fi</Genre>
                    <RunningTime>161</RunningTime>
                    <ReleaseDate>2009-12-10</ReleaseDate>
          </Movie>
          <Director Id="1">
                    <FirstName>Christopher</FirstName>
                    <LastName>Nolan</LastName>
                    <DateOfBirth>1970-06-30</DateOfBirth>
                    <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × SELECT

"/MoviesionData"

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                  <Title>Inception</Title>
                  <Genre>Adventure</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>148</RunningTime>
                  <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                  <Title>Avatar</Title>
                  <Genre>Mystery</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>161</RunningTime>
                  <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                  <FirstName>Christopher</FirstName>
                  <LastName>Nolan</LastName>
                  <DateOfBirth>1970-06-30</DateOfBirth>
                  <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × SELECT

"/MoviesionData/Movie"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                   <Title>Inception</Title>
                   <Genre>Adventure</Genre>
                   <Genre>Sci-fi</Genre>
                   <RunningTime>148</RunningTime>
                   <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                   <Title>Avatar</Title>
                   <Genre>Mystery</Genre>
                   <Genre>Sci-fi</Genre>
                   <RunningTime>161</RunningTime>
                   <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                   <FirstName>Christopher</FirstName>
                   <LastName>Nolan</LastName>
                   <DateOfBirth>1970-06-30</DateOfBirth>
                   <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × SELECT

"/MoviesionData/Movie/Title"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                  <Title>Inception</Title>
                  <Genre>Adventure</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>148</RunningTime>
                  <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                  <Title>Avatar</Title>
                  <Genre>Mystery</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>161</RunningTime>
                  <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                  <FirstName>Christopher</FirstName>
                  <LastName>Nolan</LastName>
                  <DateOfBirth>1970-06-30</DateOfBirth>
                  <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × SELECT

"/MoviesionData/Movie/Genre"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                  <Title>Inception</Title>
                  <Genre>Adventure</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>148</RunningTime>
                  <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                  <Title>Avatar</Title>
                  <Genre>Mystery</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>161</RunningTime>
                  <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                  <FirstName>Christopher</FirstName>
                  <LastName>Nolan</LastName>
                  <DateOfBirth>1970-06-30</DateOfBirth>
                  <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × SELECT

"/MoviesionData/Movie/@Id"

For Attribute use @

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                  <Title>Inception</Title>
                  <Genre>Adventure</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>148</RunningTime>
                  <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                  <Title>Avatar</Title>
                  <Genre>Mystery</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>161</RunningTime>
                  <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                  <FirstName>Christopher</FirstName>
                  <LastName>Nolan</LastName>
                  <DateOfBirth>1970-06-30</DateOfBirth>
                  <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × SELECT

"//@Id"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                  <Title>Inception</Title>
                  <Genre>Adventure</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>148</RunningTime>
                  <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                  <Title>Avatar</Title>
                  <Genre>Mystery</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>161</RunningTime>
                  <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                  <FirstName>Christopher</FirstName>
                  <LastName>Nolan</LastName>
                  <DateOfBirth>1970-06-30</DateOfBirth>
                  <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × WHERE

"/MoviesionData/Movie[@Id=2]"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                   <Title>Inception</Title>
                   <Genre>Adventure</Genre>
                   <Genre>Sci-fi</Genre>
                   <RunningTime>148</RunningTime>
                   <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                   <Title>Avatar</Title>
                   <Genre>Mystery</Genre>
                   <Genre>Sci-fi</Genre>
                   <RunningTime>161</RunningTime>
                   <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                   <FirstName>Christopher</FirstName>
                   <LastName>Nolan</LastName>
                   <DateOfBirth>1970-06-30</DateOfBirth>
                   <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × WHERE

```
"/MoviesionData/Movie[Title='Inception']"
```

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                  <Title>Inception</Title>
                  <Genre>Adventure</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>148</RunningTime>
                  <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                  <Title>Avatar</Title>
                  <Genre>Mystery</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>161</RunningTime>
                  <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                  <FirstName>Christopher</FirstName>
                  <LastName>Nolan</LastName>
                  <DateOfBirth>1970-06-30</DateOfBirth>
                  <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × WHERE

"/MoviesionData/Movie[Genre='Sci-fi']"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
         <Movie Id="1" DirectorId="1">
                  <Title>Inception</Title>
                  <Genre>Adventure</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>148</RunningTime>
                  <ReleaseDate>2010-07-08</ReleaseDate>
         </Movie>
         <Movie Id="2" DirectorId="10">
                  <Title>Avatar</Title>
                  <Genre>Mystery</Genre>
                  <Genre>Sci-fi</Genre>
                  <RunningTime>161</RunningTime>
                  <ReleaseDate>2009-12-10</ReleaseDate>
         </Movie>
         <Director Id="1">
                  <FirstName>Christopher</FirstName>
                  <LastName>Nolan</LastName>
                  <DateOfBirth>1970-06-30</DateOfBirth>
                  <PlaceOfBirth>England</PlaceOfBirth>
```

XML × XPath × WHERE

"/MoviesionData/*[@Id=1]"

```
<?xml version="1.0" encoding="UTF-8"?>
<MoviesionData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:noNamespaceSchemaLocation="MoviesionData.xsd">
           <Movie Id="1" DirectorId="1">
                      <Title>Inception</Title>
                      <Genre>Adventure</Genre>
                      <Genre>Sci-fi</Genre>
                      <RunningTime>148</RunningTime>
                      <ReleaseDate>2010-07-08</ReleaseDate>
           </Movie>
           <Movie Id="2" DirectorId="10">
                      <Title>Avatar</Title>
                      <Genre>Mystery</Genre>
                      <Genre>Sci-fi</Genre>
                      <RunningTime>161</RunningTime>
                      <ReleaseDate>2009-12-10</ReleaseDate>
           </Movie>
           <Director Id="1">
                      <FirstName>Christopher</FirstName>
                      <LastName>Nolan</LastName>
                      <DateOfBirth>1970-06-30</DateOfBirth>
                      <PlaceOfBirth>England</PlaceOfBirth>
           </Director>
```

XML × XPath × Syntax

```
"/MoviesionData/Movie/*"
"/MoviesionData/Movie/@*"
"/MoviesionData/Movie/* | /MoviesionData/Movie/@*"
"/MoviesionData/Movie[1]/* | /MoviesionData/Movie[last()]/*"
"/MoviesionData/Movie[position()>1 and position()<3]"

"/MoviesionData/Movie[@DirectorId]"
"/MoviesionData/Movie/@DirectorId"

"/MoviesionData/Movie[@DirectorId]/Title"
"/MoviesionData/Movie[@DirectorId=1]/Title[text()]“
```

Back to <key> and <keyref> and see the XML power!

perfected 1st
romains 43.3
diametre
140
BRUNELLESCHI
model built with
DONATELLO & BANCO
brick construction
no concrete
octagonal
403
Romains
Granite,
chapiteaux marbre
16 corinthian
columns
Egypte

XML × Why?

XML × Why? Data Exchange

XML has become the *de facto* basis for all data exchange.
XML document is self-describing due to presence of tags.

Person 2 Person
Software 2 Person (Export)
Person 2 Software (Import)
Software 2 Software (Browser, Web Service, API)

Choice of Data Model

I) When you need maximum flexibility in design

Relational tables follow a fairly rigid model. Normalizing can be very difficult if the schema changes often.

As such, representing it as XML document is a better choice. XML schemas can be evolved over time.

Choice of Data Model

II) When you need maximum performance for data retrieval

If performance is more of an issue than flexibility, relational data might be the better choice.

Some expense is associated with serializing and interpreting XML document.

Choice of Data Model

III) When the ratio of data complexity to volume is high

Many situations involve highly structured information in very small quantities. Representation of that data with a relational model can involve complex tables where most of the tables have only a few rows.

A better way to represent this data is to use an XML document.

Choice of Data Model

IV) ACID properties

Choice of Data Model

IV) ACID properties

Relational

XML

Why?

Relational Data 2 XML Document?

Relational Data 2 XML Document?

Data in almost all data models can be serialized to XML. So what?

Relational Data 2 XML Document?

Data in almost all data models can be serialized to XML.

So, data in any data model can be exchanged.

XML Document 2 Relational Data?

XML Document 2 Relational Data?

Somehow! Still, we loose

- Order!
- Heterogeneous data!
- …

JOAQUIN PHOENIX
JOKER
PUT ON A HAPPY FACE
OCTOBER 4
DC