Data Models and Database Management Systems (DBMSs)
Data Models in the 1960s, 1970s, and 1980s

Hierarchical

1956 Vern Watts was IMS's chief architect for many years. Watts joined IBM in 1956 and worked at IBM's Silicon Valley development labs until his death on April 4, 2009.

1966 IBM began the design of IMS with Rockwell and Caterpillar for the Apollo program, where it was used to inventory the very large bill of materials (BOM) for the Saturn V moon rocket and Apollo space vehicle.

1968 The first "IMS READY" message appeared on an IBM 2740 terminal in Downey, California, on 14 August 1968.

Network (Graph)

1965 CODASYL (Conference / Committee on Data Systems Languages) formed a List Processing Task Force.

1969 the network CODASYL database model was published.

1980s interest in CODASYL gradually faded due to growing interest in relational databases.

Relational – Schema (Model) - first

1970 Dr. E. F. Codd develops the relational database model.
1978 IBM develops the predecessor to SQL.
1979 Relational Software, Inc. (later renamed Oracle) releases the first relational DBMS, Oracle.
1982 IBM releases their first RDBMS, SQL/DS (SQL/Data System).
1985 IBM released DB2 (Database 2).
1987 Microsoft releases SQL Server.
1989 ANSI publishes first SQL standards (ANSI/ISO SQL-89, or SQL1).

See humorous video here
Dr. E. F. Codd

1. Humans only know how to store data on Disk or in Memory in the form of Mathematical Relations (Tables).
2. So, the Data Model should be Relational (i.e., Tables).
3. Also, DBMSs should have an ad hoc query language.

A mathematical Relation is a subset of the cross-product of a set of Domains. e.g.,

Example 1

<table>
<thead>
<tr>
<th>&lt;</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Example 2

<table>
<thead>
<tr>
<th>D1: People</th>
<th>D2: Ages</th>
<th>D3: Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phil</td>
<td>60</td>
<td>M</td>
</tr>
<tr>
<td>Phil</td>
<td>60</td>
<td>F</td>
</tr>
<tr>
<td>Phil</td>
<td>65</td>
<td>M</td>
</tr>
<tr>
<td>Phil</td>
<td>65</td>
<td>F</td>
</tr>
<tr>
<td>Rita</td>
<td>60</td>
<td>M</td>
</tr>
<tr>
<td>Rita</td>
<td>60</td>
<td>F</td>
</tr>
<tr>
<td>Rita</td>
<td>65</td>
<td>M</td>
</tr>
<tr>
<td>Rita</td>
<td>65</td>
<td>F</td>
</tr>
</tbody>
</table>
Traditional Data Management in the 1980s, 1990s, and 2000s

1956 Vern Watts was IMS's chief architect for many years. He joined IBM in 1956 and worked at IBM's silicon Valley development labs until his death on April 4, 2009.

1965 CODASYL (Conference on Data System Languages) formed the Object Design Task Force.

1969 The network CODASYL database model was published.

1980s interest in CODASYL gradually faded due to growing interest in relational databases.

1970 Dr. E. F. Codd develops the relational database model.

1978 IBM develops the predecessor to SQL.

1979 Relational Software, Inc. (later renamed Oracle) releases the first relational DBMS, Oracle.

1982 IBM releases their first RDBMS, SQL/DS (SQL/Data System).

1985 IBM released DB2 (Database 2).

1987 Microsoft releases SQL Server.

1989 ANSI publishes first SQL standards (ANSI/ISO SQL-89, or SQL1).


cs347 Data Management

NoSQL – Schema(Model)-later

Big Data – Schema(Model)-never

Relational (YeSQL) – Schema-first

The Complexity goes Somewhere!
Relational, Graph, and Property Graph Data Models

**Relational Model**

<table>
<thead>
<tr>
<th>EMPNO</th>
<th>ENAME</th>
<th>JOB</th>
<th>MGR</th>
<th>HIREDATE</th>
<th>SAL</th>
<th>COMM</th>
<th>DEPTNO</th>
</tr>
</thead>
<tbody>
<tr>
<td>7369</td>
<td>SMITH</td>
<td>CLERK</td>
<td>7902</td>
<td>17-DEC-80</td>
<td>800</td>
<td>0</td>
<td>20</td>
</tr>
</tbody>
</table>

**Graph Model**

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>emp class</td>
<td></td>
<td>RowGUID1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7369</td>
</tr>
<tr>
<td></td>
<td></td>
<td>17-DEC-80</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SMITH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CLERK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7920</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
</tr>
</tbody>
</table>

**Property Graph Model**

- **Node**
  - emp node 1
    - empno = 7369
    - ename = SMITH
    - job = CLERK
    - mgr = 7902
    - hiredate = 17-DEC-80
    - sal = 800
    - deptno = 20

- **Relationship**
  - department
    - employee

- **Node**
  - dept node 1
    - deptno = 7369
    - dname = SMITH
    - loc = CLERK

Dr. Philip Cannata
Relational Model

CREATE TABLE EMP
(EMPNO NUMBER(7) NOT NULL,
CONSTRAINT emp_pkey PRIMARY KEY (empno),
ENAME VARCHAR2(10),
JOB VARCHAR2(9),
MGR NUMBER(4),
HIREDATE DATE,
SAL NUMBER(7, 2),
COMM NUMBER(7, 2),
DEPTNO NUMBER(2));

CREATE TABLE DEPT
(DEPTNO NUMBER(2),
DNAME VARCHAR2(14),
LOC VARCHAR2(13));

Graph Model

Not needed but allowed and standardized by W3C. We’ll see this later with rdf, rdfs, and owl.

Property Graph Model

Not allowed.
Phils-MacBook-Pro:~$ jython
Jython 2.7.1b2 (default:8fb35eb165e+5194b3d04dca+, Jan 20 2016, 15:32:05)
[Java HotSpot(TM) 64-Bit Server VM (Oracle Corporation)] on java1.8.0_51
Type "help", "copyright", "credits" or "license" for more information.

>>> conn = connectTo 'OracleNoSQL' 'C##cs329e_U TEId' 'orcl_U TEId' 'rdf_mode' 'A0' nodebug
**Relational, Graph, and Property Graph Data Models**

**Relational Model**

SQL:
- SQL on conn "INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7369, 'SMITH', 'CLERK', 7902, TO_DATE('17-DEC-1980', 'DD-MON-YYYY'), 800, NULL, 20) ;"
- SQL on conn "INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (20, 'RESEARCH', 'DALLAS') ;"

**Graph Model**

RDF Java API:
- In addQuad, stmt is: :dept, :7580ebed-e755-4de1-a490-f37c32542f0b, :deptno, :20
- In addQuad, stmt is: :dept, :7580ebed-e755-4de1-a490-f37c32542f0b, :dname, :RESEARCH
- In addQuad, stmt is: :dept, :7580ebed-e755-4de1-a490-f37c32542f0b, :loc, :DALLAS
- In addQuad, stmt is: :dept, :7580ebed-e755-4de1-a490-f37c32542f0b, :employees, :dfb8b0a9-4dca-4ecc-922e-d4b515043010

**Property Graph Model**

Neo4j OpenCypher:
- Neo4j on conn "CREATE (:dept { DEPTNO : 20, DNAME : 'RESEARCH', LOC : 'DALLAS' }) ;"
- Neo4j on conn "MATCH (a:emp),(b:dept) WHERE a.deptno = 20 AND b.deptno = 20 CREATE (a)<-[ :employees ]->(b) ;"
- Neo4j on conn "MATCH (a:emp),(b:dept) WHERE a.deptno = 20 AND b.deptno = 20 CREATE (a)-[ :dept ]->(b) ;"
Relational, Graph, and Property Graph Data Models

### Relational Model

**SQL:**
- SQL on conn "select * from emp";
- SQL on conn "select * from dept"
- SQL on conn "select e.ename, d.dname from emp e join dept d on(e.deptno = d.deptno)"

---

### Graph Model

**SPARQL:**
See the next page.

---

### Property Graph Model

*Neo4j OpenCypher:*
- Neo4j on conn "MATCH(a:emp) RETURN a.ename; "
- Neo4j on conn "MATCH(a:dept) RETURN a.dname; "
- Neo4j on conn "MATCH(a:emp)<-[employees]-(b:dept) RETURN a.ename, b.dname; "

Dr. Philip Cannata
Relational, Graph, and Property Graph Data Models

Relational Model

<table>
<thead>
<tr>
<th>EMPNO</th>
<th>ENAME</th>
<th>JOB</th>
<th>MGR</th>
<th>HIREDATE</th>
<th>SAL</th>
<th>COMM</th>
<th>DEPTNO</th>
</tr>
</thead>
<tbody>
<tr>
<td>7369</td>
<td>SMITH</td>
<td>CLERK</td>
<td>7902</td>
<td>17-DEC-80</td>
<td>800</td>
<td>(null)</td>
<td>20</td>
</tr>
</tbody>
</table>

CLERK

Graph Model

Property
Graph Model

<table>
<thead>
<tr>
<th>Node</th>
<th>Relationship</th>
<th>Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>emp node 1</td>
<td>department</td>
<td>dept node 1</td>
</tr>
<tr>
<td>empno = 7369</td>
<td></td>
<td>deptno = 7369</td>
</tr>
<tr>
<td>ename = SMITH</td>
<td></td>
<td>dname = SMITH</td>
</tr>
<tr>
<td>job = CLERK</td>
<td></td>
<td>loc = CLERK</td>
</tr>
<tr>
<td>mgr = 7902</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hidedate = 17-DEC-80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sal = 800</td>
<td></td>
<td></td>
</tr>
<tr>
<td>deptno = 20</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Relational, Graph, and Property Graph Data Models

SCHEMA

('SCHEMA', 'dept', 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'http://www.w3.org/2000/01/rdf-schema#Class')
('SCHEMA', 'emp', 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'http://www.w3.org/2000/01/rdf-schema#Class')

('dept', '94f693ee-c71d-4caa-8f76-0d51152635af', 'deptno', 20)
('dept', '94f693ee-c71d-4caa-8f76-0d51152635af', 'dname', 'RESEARCH')
('dept', '94f693ee-c71d-4caa-8f76-0d51152635af', 'employees', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89')
('dept', '94f693ee-c71d-4caa-8f76-0d51152635af', 'loc', 'DALLAS')

('dept_SCHEMA', '94f693ee-c71d-4caa-8f76-0d51152635af', 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'dept')
('dept_SCHEMA', 'deptno', 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'http://www.w3.org/2002/07/owl#DatatypeProperty')
('dept_SCHEMA', 'deptno', 'http://www.w3.org/2000/01/rdf-schema#domain', 'dept')
('dept_SCHEMA', 'deptno', 'http://www.w3.org/2000/01/rdf-schema#range', 'http://www.w3.org/2001/XMLSchema#string')
('dept_SCHEMA', 'dname', 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'http://www.w3.org/2002/07/owl#DatatypeProperty')
('dept_SCHEMA', 'dname', 'http://www.w3.org/2000/01/rdf-schema#domain', 'dept')
('dept_SCHEMA', 'dname', 'http://www.w3.org/2000/01/rdf-schema#range', 'http://www.w3.org/2001/XMLSchema#string')
('dept_SCHEMA', 'loc', 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'http://www.w3.org/2002/07/owl#DatatypeProperty')
('dept_SCHEMA', 'loc', 'http://www.w3.org/2000/01/rdf-schema#domain', 'dept')
('dept_SCHEMA', 'loc', 'http://www.w3.org/2000/01/rdf-schema#range', 'http://www.w3.org/2001/XMLSchema#string')
Relational, Graph, and Property Graph Data Models

**Relational Model**

<table>
<thead>
<tr>
<th>EMPNO</th>
<th>ENAME</th>
<th>JOB</th>
<th>MGR</th>
<th>HIREDATE</th>
<th>SAL</th>
<th>COMM</th>
<th>DEPTNO</th>
</tr>
</thead>
<tbody>
<tr>
<td>7369</td>
<td>SMITH</td>
<td>CLERK</td>
<td>7902</td>
<td>17-DEC-80</td>
<td>800</td>
<td>(null)</td>
<td>20</td>
</tr>
</tbody>
</table>

('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'comm', 0)
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'dept', '94f693ee-c71d-4caa-8f76-0d51152635af')
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'deptno', 20)
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'empno', 7369)
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'ename', 'SMITH')
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'hiredate', '17-DEC-80')
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'job', 'CLERK')
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'mgr', 7902)
('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'sal', 800).org/2001/XMLSchema#string'

**Graph Model**

- **department**
- **employee**

- employee:
edname = SMITH
  
  job = CLERK
  
  mgr = 7902
  
  hiredate = 17-DEC-80
  
  sal = 800
  
  deptno = 20

- department:
dname= SMITH
  
  loc= CLERK

- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'comm', 0)
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'dept', '94f693ee-c71d-4caa-8f76-0d51152635af')
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'deptno', 20)
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'empno', 7369)
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'ename', 'SMITH')
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'hiredate', '17-DEC-80')
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'job', 'CLERK')
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'mgr', 7902)
- ('emp', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', 'sal', 800).org/2001/XMLSchema#string')
Relational, Graph, and Property Graph Data Models

### Relational Model

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>RowGUID1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RowGUID2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Graph Model

- **emp** class
- **dept** class

<table>
<thead>
<tr>
<th>Node</th>
<th>Relationship</th>
<th>Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>('emp_SCHEMA', 'a9e1615f-c8df-478d-8c48-ea4b2ccd1a89', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'comm', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'comm', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'comm', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'deptno', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'deptno', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'deptno', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'empno', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'empno', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'empno', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'ename', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'ename', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'ename', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'hiredate', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'hiredate', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'hiredate', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'job', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'job', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'job', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'mgr', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'mgr', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'mgr', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'sal', '<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>', '<a href="http://www.w3.org/2002/07/owl#DatatypeProperty">http://www.w3.org/2002/07/owl#DatatypeProperty</a>')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'sal', '<a href="http://www.w3.org/2000/01/rdf-schema#domain">http://www.w3.org/2000/01/rdf-schema#domain</a>', 'emp')</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('emp_SCHEMA', 'sal', '<a href="http://www.w3.org/2000/01/rdf-schema#range">http://www.w3.org/2000/01/rdf-schema#range</a>', '<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>')</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
class/subclass Inferencing is just the beginning

For more information on Inferencing, see Chapter 9 of the Learning SPARQL book (see the Books and Papers Tab on the class website).

Dr. Philip Cannata
All Data Models are Ignoring the First Data Model

2 Visualizing Aristotle’s 10-fold Division & His Varieties of Attribution

Figure 1 allows us to get a visual feel for Aristotle’s more fine-grained, 10-fold division, and the ways in which he thinks attributions work (and don’t work). Here, we are drawing on both the Categories readings and the Topics readings.

Table 2 describes four different varieties of Aristotelian attribution which we can use to classify the attributions in Figure 1.

Table 2: Four varieties of Aristotelian attribution.

---

1 In Figure 1, the arrows represent attributions. For instance, “Human —> Socrates” (i.e., attribution 2 in Figure 1) is to be understood as the attribution “Socrates is [a] Human.” Not all of the arrows drawn in Figure 1 represent genuine attributions within Aristotle’s theory. For instance, arrows 6 and 7 are not genuine attributions, as far as Aristotle is concerned. Why not? Can you think of other non-genuine attributions?

2 Can you find relevant passages in the Categories and/or the Topics which justify (or refute!) each of the claims made in this handout?
Except for SIM, which was implemented in the 1980s and still exists at UT with Me

Based on the 1981 paper “Database Description with SDM: A Semantic Database Model” by Michael Hammer and Dennis McLeod, see class calendar for a pdf version of the paper.
Except for SIM, which was implemented in the 1980s and still exists at UT with Me