Ontology-Based Data Access to Slegge

Roman Kontchakov

Department of Computer Science & Inf. Systems, Birkbeck, University of London

http://www.dcs.bbk.ac.uk/~roman

Optique

joint work with **Dag Hovland, Martin G. Skjæveland, Arild Waaler and M. Zakharyaschev**

Data Gathering at Statoil (1)

Slegge 'sledgehammer' database contains tables

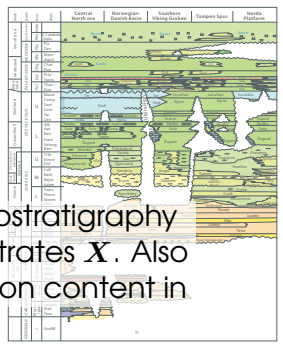
```
WELLBORE, STRATIGRAPHIC_ZONE, ROCK_FEATURE, COMPONENT_MATERIAL,
DATA_COLLECTION, DATA_COLLECTION_CONTENT, MATERIAL_CLASS,
CLASSIFICATION_SYSTEM, MATERIAL_CLASSIFICATION, ...
```

Data Gathering at Statoil (1)

Slegge 'sledgehammer' database contains tables

```
WELLBORE, STRATIGRAPHIC_ZONE, ROCK_FEATURE, COMPONENT_MATERIAL,
DATA_COLLECTION, DATA_COLLECTION_CONTENT, MATERIAL_CLASS,
CLASSIFICATION_SYSTEM, MATERIAL_CLASSIFICATION, ...
```

LITHOSTRATIGRAPHIC CHART
NORWEGIAN NORTH SEA



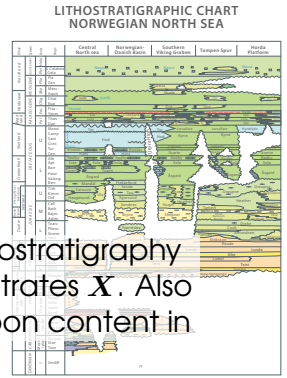
information need of a geoscientist:

In my area of interest, return the wellbores penetrating a given chronostratigraphic unit $\boldsymbol{X}$ and return information about the lithostratigraphy and the hydrocarbon content in the wellbore interval that penetrates $\boldsymbol{X}$. Also return information about other wellbore intervals with hydrocarbon content in the wellbores with hydrocarbon in $\boldsymbol{X}$.

Data Gathering at Statoil (1)

Slegge 'sledgehammer' database contains tables

```
WELLBORE, STRATIGRAPHIC_ZONE, ROCK_FEATURE, COMPONENT_MATERIAL,
DATA_COLLECTION, DATA_COLLECTION_CONTENT, MATERIAL_CLASS,
CLASSIFICATION_SYSTEM, MATERIAL_CLASSIFICATION, ...
```



information need of a geoscientist:

In my area of interest, return the wellbores penetrating a given chronostratigraphic unit X and return information about the lithostratigraphy and the hydrocarbon content in the wellbore interval that penetrates X. Also return information about other wellbore intervals with hydrocarbon content in the wellbores with hydrocarbon in X.

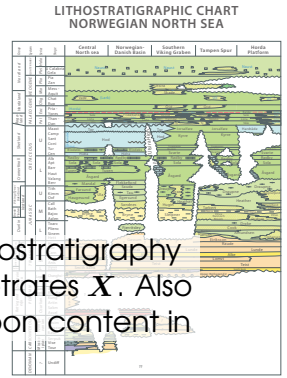
solution 1 use **predefined SQL queries** of the in-house system to retrieve information about (a) wellbores penetrating chronostratigraphic unit X, (b) lithostratigraphy of wellbores, (c) hydrocarbon content in wellbore intervals, etc.

and **integrate** the results using a spreadsheet

Data Gathering at Statoil (1)

Slegge 'sledgehammer' database contains tables

```
WELLBORE, STRATIGRAPHIC_ZONE, ROCK_FEATURE, COMPONENT_MATERIAL,
DATA_COLLECTION, DATA_COLLECTION_CONTENT, MATERIAL_CLASS,
CLASSIFICATION_SYSTEM, MATERIAL_CLASSIFICATION, ...
```



information need of a geoscientist:

In my area of interest, return the wellbores penetrating a given chronostratigraphic unit $\boldsymbol{X}$ and return information about the lithostratigraphy and the hydrocarbon content in the wellbore interval that penetrates $\boldsymbol{X}$. Also return information about other wellbore intervals with hydrocarbon content in the wellbores with hydrocarbon in $\boldsymbol{X}$.

solution 1 use **predefined SQL queries** of the in-house system to retrieve information about (a) wellbores penetrating chronostratigraphic unit $\boldsymbol{X}$, (b) lithostratigraphy of wellbores, (c) hydrocarbon content in wellbore intervals, etc. and **integrate** the results using a spreadsheet

time-consuming, error-prone (e.g., different units of measurement), **difficult to repeat**

Data Gathering at Statoil (2)

solution 2 ask an **IT expert** to translate the information need into SQL

```
SELECT
    WB.IDENTIFIER AS wellbore,
    SZ1.STRAT_UNIT_IDENTIFIER AS chronostrat_unit,
    SZ2.STRAT_UNIT_IDENTIFIER AS lithostrat_unit,
    SZ1.STRAT_ZONE_ENTRY_MD AS top_md_m
FROM
    WELLBORE WB,
    STRATIGRAPHIC_ZONE SZ1,
    ROCK_FEATURE RF1,
    COMPONENT_MATERIAL CM1,
    DATA_COLLECTION COL1,
        ... (24 tables)
WHERE
    WB.REF_EXISTENCE_KIND = 'actual' AND
    WB.IDENTIFIER = SZ1.WELLBORE AND
        ...
    CM1.ENTITY_TYPE_NAME = 'COMPONENT_MATERIAL' AND
        ...
    COL1.REF_DATA_COLLECTION_TYPE = 'stratigraphic hierarchy' AND
        ...
    (lots of other conditions)
```

Data Gathering at Statoil (2)

solution 2 ask an **IT expert** to translate the information need into SQL

```
SELECT
    WB.IDENTIFIER AS wellbore,
    SZ1.STRAT_UNIT_IDENTIFIER AS chronostrat_unit,
    SZ2.STRAT_UNIT_IDENTIFIER AS lithostrat_unit,
    SZ1.STRAT_ZONE_ENTRY_MD AS top_md_m
FROM
    WELLBORE WB,
    STRATIGRAPHIC_ZONE SZ1,
    ROCK_FEATURE RF1,
    COMPONENT_MATERIAL CM1,
    DATA_COLLECTION COL1,
        ... (24 tables)
WHERE
    WB.REF_EXISTENCE_KIND = 'actual' AND
    WB.IDENTIFIER = SZ1.WELLBORE AND
        ...
    CM1.ENTITY_TYPE_NAME = 'COMPONENT_MATERIAL' AND
        ...
    COL1.REF_DATA_COLLECTION_TYPE = 'stratigraphic hierarchy' AND
        ...
    (lots of other conditions)
```

time-consuming (a few days to a couple of weeks)

Data Gathering at Statoil (2)

solution 2 ask an **IT expert** to translate the information need into SQL

```
SELECT
    WB.IDENTIFIER AS wellbore,
    SZ1.STRAT_UNIT_IDENTIFIER AS chronostrat_unit,
    SZ2.STRAT_UNIT_IDENTIFIER AS lithostrat_unit,
    SZ1.STRAT_ZONE_ENTRY_MD AS top_md_m
FROM
    WELLBORE WB,
    STRATIGRAPHIC_ZONE SZ1,
    ROCK_FEATURE RF1,
    COMPONENT_MATERIAL CM1,
    DATA_COLLECTION COL1,
        ... (24 tables)
WHERE
    WB.REF_EXISTENCE_KIND = 'actual' AND
    WB.IDENTIFIER = SZ1.WELLBORE AND
        ...
    CM1.ENTITY_TYPE_NAME = 'COMPONENT_MATERIAL' AND
        ...
    COL1.REF_DATA_COLLECTION_TYPE = 'stratigraphic hierarchy' AND
        ...
    (lots of other conditions)
```

time-consuming (a few days to a couple of weeks)

knowledge of the **geological domain** and **database structure**

(1545 tables and 1727 views)

Data Gathering at Statoil (2)

solution 2 ask an **IT expert** to translate the information need into SQL

```
SELECT
    WB.IDENTIFIER AS wellbore,
    SZ1.STRAT_UNIT_IDENTIFIER AS chronostrat_unit,
    SZ2.STRAT_UNIT_IDENTIFIER AS lithostrat_unit,
    SZ1.STRAT_ZONE_ENTRY_MD AS top_md_m
FROM
    WELLBORE WB,
    STRATIGRAPHIC_ZONE SZ1,
    ROCK_FEATURE RF1,
    COMPONENT_MATERIAL CM1,
    DATA_COLLECTION COL1,
        ... (24 tables)
WHERE
    WB.REF_EXISTENCE_KIND = 'actual' AND
    WB.IDENTIFIER = SZ1.WELLBORE AND
        ...
    CM1.ENTITY_TYPE_NAME = 'COMPONENT_MATERIAL' AND
        ...
    COL1.REF_DATA_COLLECTION_TYPE = 'stratigraphic hierarchy' AND
        ...
    (lots of other conditions)
```

time-consuming (a few days to a couple of weeks)

knowledge of the **geological domain** and **database structure**
(1545 tables and 1727 views)

encode the domain knowledge in an ontology

Data Gathering at Statoil (2)

solution 2 ask an **IT expert** to translate the information need into SQL

```
SELECT
    WB.IDENTIFIER AS wellbore,
    SZ1.STRAT_UNIT_IDENTIFIER AS chronostrat_unit,
    SZ2.STRAT_UNIT_IDENTIFIER AS lithostrat_unit,
    SZ1.STRAT_ZONE_ENTRY_MD AS top_md_m
FROM
    WELLBORE WB,
    STRATIGRAPHIC_ZONE SZ1,
    ROCK_FEATURE RF1,
    COMPONENT_MATERIAL CM1,
    DATA_COLLECTION COL1,
        ... (24 tables)
WHERE
    WB.REF_EXISTENCE_KIND = 'actual' AND
    WB.IDENTIFIER = SZ1.WELLBORE AND
        ...
    CM1.ENTITY_TYPE_NAME = 'COMPONENT_MATERIAL' AND
        ...
    COL1.REF_DATA_COLLECTION_TYPE = 'stratigraphic hierarchy' AND
        ...
    (lots of other conditions)
```

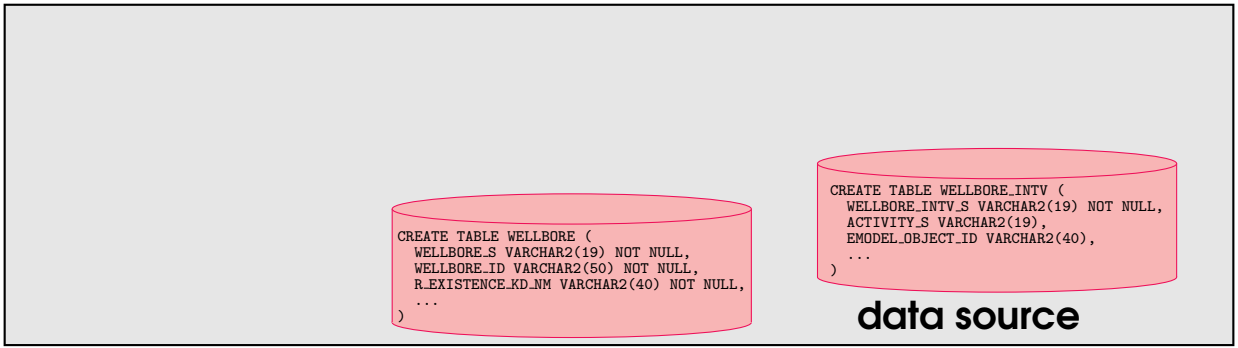
time-consuming (a few days to a couple of weeks)

knowledge of the **geological domain** and **database structure**

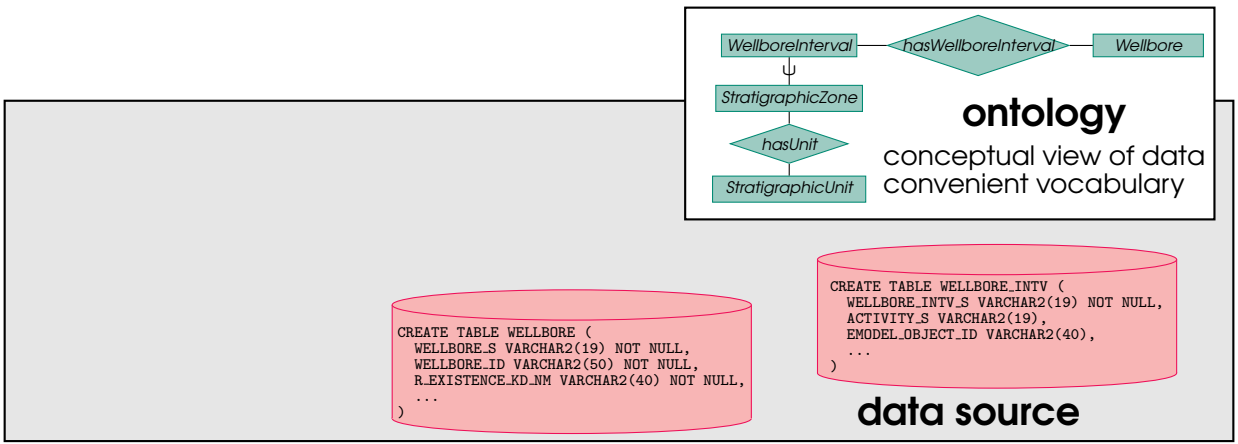
(1545 tables and 1727 views)

encode the domain knowledge in an ontology and database structure in mappings

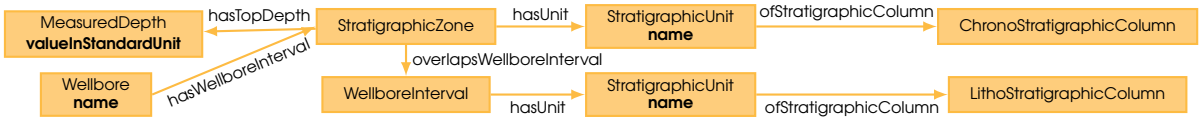
Ontology-Based Data Access



Ontology-Based Data Access



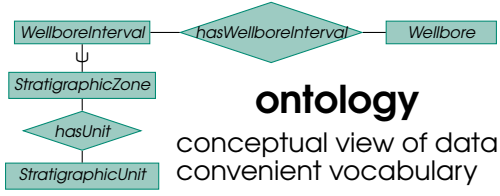
Ontology-Based Data Access



Visual Query System

```
SELECT ?wellbore ?chronostrat_unit ?top_md_m ?lithostrat_unit WHERE {
    ?w a :Wellbore; :name ?wellbore; :hasWellboreInterval ?intv.
    ?intv a :StratigraphicZone; :hasUnit ?cu; :hasTopDepth ?top.
    ?cu :name ?chronostrat_unit;
        :ofStratigraphicColumn [a :ChronoStratigraphicColumn ].
    ?top a :MeasuredDepth; :valueInStandardUnit ?top_md_m.
    ?intv :overlapsWellboreInterval ?litho_intv.
    ?litho_intv :hasUnit ?lu.
    ?lu :name ?lithostrat_unit;
        :ofStratigraphicColumn [a :LithoStratigraphicColumn ].
}
```

SPARQL query



ontology

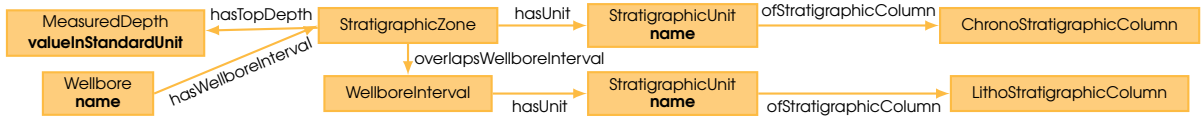
conceptual view of data
convenient vocabulary

```
CREATE TABLE WELLBORE (
  WELLBORE_S VARCHAR2(19) NOT NULL,
  WELLBORE_ID VARCHAR2(50) NOT NULL,
  R_EXISTENCE_KD_NM VARCHAR2(40) NOT NULL,
  ...
)
```

```
CREATE TABLE WELLBORE_INTV (
  WELLBORE_INTV_S VARCHAR2(19) NOT NULL,
  ACTIVITY_S VARCHAR2(19),
  EMODEL_OBJECT_ID VARCHAR2(40),
  ...
)
```

data source

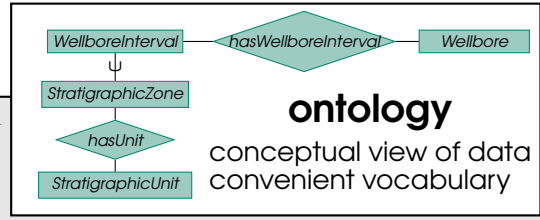
Ontology-Based Data Access




```
SELECT ?wellbore ?chronostrat_unit ?top_md_m ?lithostrat_unit WHERE {
    ?w a :Wellbore; :name ?wellbore; :hasWellboreInterval ?intv.
    ?intv a :StratigraphicZone; :hasUnit ?cu; :hasTopDepth ?top.
    ?cu :name ?chronostrat_unit;
        :ofStratigraphicColumn [a :ChronoStratigraphicColumn ].
    ?top a :MeasuredDepth; :valueInStandardUnit ?top_md_m.
    ?intv :overlapsWellboreInterval ?litho_intv.
    ?litho_intv :hasUnit ?lu.
    ?lu :name ?lithostrat_unit;
        :ofStratigraphicColumn [a :LithoStratigraphicColumn ].
}
```

SPARQL query

Visual Query System



ontology

conceptual view of data
convenient vocabulary

```
map:m-00008 a rr:TriplesMap ;
  rr:logicalTable [ rr:tableName "STRATIGRAPHIC_ZONE" ] ;
  rr:predicateObjectMap [
    rr:predicate expl:hasUnit ;
    rr:objectMap [ rr:termType rr:IRI ;
      rr:template "StratigraphicUnit-{STRAT_COLUMN_ID}-{...}" ]
  ...
```

mappings

```
CREATE TABLE WELLBORE_INTV (
  WELLBORE_INTV_S VARCHAR2(19) NOT NULL,
  ACTIVITY_S VARCHAR2(19),
  EMODEL_OBJECT_ID VARCHAR2(40),
  ...
)
```

```
CREATE TABLE WELLBORE (
  WELLBORE_S VARCHAR2(19) NOT NULL,
  WELLBORE_ID VARCHAR2(50) NOT NULL,
  R_EXISTENCE_KD_NM VARCHAR2(40) NOT NULL,
  ...
)
```

data source

reduced time for translating information needs into (SPARQL) queries

days ⟶ minutes

From Information Needs to Queries and Ontology

73 information needs collected from end-users at Statoil over a period of 4 years

From Information Needs to Queries and Ontology

73 information needs collected from end-users at Statoil over a period of 4 years

39 are **beyond the scope** of the Slegge database

user interface configuration / data entry processes / data unavailable in Slegge

34 are basic **competency questions** for the Subsurface Exploration Ontology

From Information Needs to Queries and Ontology

73 information needs collected from end-users at Statoil over a period of 4 years

39 are **beyond the scope** of the Slegge database

user interface configuration / data entry processes / data unavailable in Slegge

34 are basic **competency questions** for the Subsurface Exploration Ontology

96 SPARQL queries (vagueness / possible interpretation in SPARQL in different ways)

`my area of interest' = `a particular field'

or `a geographical area in the North Sea identified by coordinates'

From Information Needs to Queries and Ontology

73 information needs collected from end-users at Statoil over a period of 4 years

39 are **beyond the scope** of the Slegge database

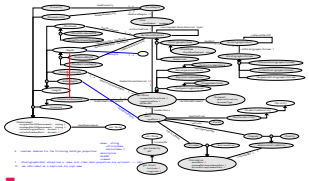
user interface configuration / data entry processes / data unavailable in Slegge

34 are basic **competency questions** for the Subsurface Exploration Ontology

96 SPARQL queries (vagueness / possible interpretation in SPARQL in different ways)

'my area of interest' = 'a particular field'

or 'a geographical area in the North Sea identified by coordinates'



Subsurface Exploration Ontology

captures terms from the information needs

71 classes (`Wellbore`, `Core`, `WellboreInterval` with subclasses `Reservoir` and `StratigraphicZone`, etc.)

NPD glossary, interviews at Statoil

From Information Needs to Queries and Ontology

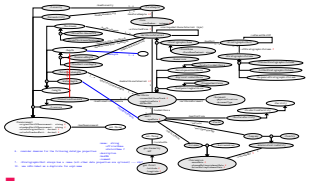
73 information needs collected from end-users at Statoil over a period of 4 years

39 are **beyond the scope** of the Slegge database
user interface configuration / data entry processes / data unavailable in Slegge

34 are basic **competency questions** for the Subsurface Exploration Ontology

96 SPARQL queries (vagueness / possible interpretation in SPARQL in different ways)

'my area of interest' = 'a particular field'
or 'a geographical area in the North Sea identified by coordinates'



Subsurface Exploration Ontology
captures terms from the information needs

71 classes (`Wellbore`, `Core`, `WellboreInterval` with subclasses `Reservoir` and `StratigraphicZone`, etc.) NPD glossary, interviews at Statoil

`Measurement` **taxonomy** (porosity, permeability, measured depth, true vertical depth, etc.)
measurements can be specified in the **standard units** / **original units**
(magic values, e.g., 9999 ft)

From Information Needs to Queries and Ontology

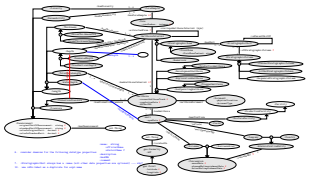
73 information needs collected from end-users at Statoil over a period of 4 years

39 are **beyond the scope** of the Slegge database
user interface configuration / data entry processes / data unavailable in Slegge

34 are basic **competency questions** for the Subsurface Exploration Ontology

96 SPARQL queries (vagueness / possible interpretation in SPARQL in different ways)

'my area of interest' = 'a particular field'
or 'a geographical area in the North Sea identified by coordinates'



Subsurface Exploration Ontology
captures terms from the information needs

71 classes (`Wellbore`, `Core`, `WellboreInterval` with subclasses `Reservoir` and `StratigraphicZone`, etc.) NPD glossary, interviews at Statoil

`Measurement` **taxonomy** (porosity, permeability, measured depth, true vertical depth, etc.)
measurements can be specified in the **standard units** / **original units**
(magic values, e.g., 9999 ft)

46 object properties & **34** data properties

Slegge Database

Slegge is an Oracle database (700 GB, since late 1990s)

based on **Epicentre v2.2** (Petrotechnical Open Standards Consortium, POSC)

object-oriented logical model and its projection to relational databases

Slegge Database

Slegge is an Oracle database (700 GB, since late 1990s)

based on **Epicentre v2.2** (Petrotechnical Open Standards Consortium, POSC)

object-oriented logical model and its projection to relational databases

- extensive (multiple) **inheritance** hierarchies
  - a table per subtype
  - a single table for all subtypes with a **discriminating column** (magic values)

1545 tables with 19 719 columns (1141 tables are **empty**)

1722 views (1632 contain no joins and no WHERE clauses)

Slegge Database

Slegge is an Oracle database (700 GB, since late 1990s)

based on **Epicentre v2.2** (Petrotechnical Open Standards Consortium, POSC)

object-oriented logical model and its projection to relational databases

- extensive (multiple) **inheritance** hierarchies
  - a table per subtype
  - a single table for all subtypes with a **discriminating column** (magic values)

  1545 tables with 19 719 columns (1141 tables are **empty**)
  1722 views (1632 contain no joins and no WHERE clauses)

- **denormalisation**: many columns are duplicated to avoid joins in queries

Slegge Database

Slegge is an Oracle database (700 GB, since late 1990s)

based on **Epicentre v2.2** (Petrotechnical Open Standards Consortium, POSC)

object-oriented logical model and its projection to relational databases

- extensive (multiple) **inheritance** hierarchies
  - a table per subtype
  - a single table for all subtypes with a **discriminating column** (magic values)

  1545 tables with 19 719 columns (1141 tables are **empty**)
  1722 views (1632 contain no joins and no WHERE clauses)

- **denormalisation**: many columns are duplicated to avoid joins in queries

- **lack of foreign keys**: many relationships involve multiple tables for subtypes
  → **conditional** foreign keys (not supported by the DBMS)

  3112 foreign keys but 2727 refer to just 18 reference tables such as `R_UOM`
  most of the remaining 385 foreign keys refer to 'single-purpose' reference tables

Slegge Database

Slegge is an Oracle database (700 GB, since late 1990s)

based on **Epicentre v2.2** (Petrotechnical Open Standards Consortium, POSC)

object-oriented logical model and its projection to relational databases

- extensive (multiple) **inheritance** hierarchies
  - a table per subtype
  - a single table for all subtypes with a **discriminating column** (magic values)

  1545 tables with 19 719 columns (1141 tables are **empty**)
  1722 views (1632 contain no joins and no WHERE clauses)

- **denormalisation**: many columns are duplicated to avoid joins in queries

- **lack of foreign keys**: many relationships involve multiple tables for subtypes
  → **conditional** foreign keys (not supported by the DBMS)

  3112 foreign keys but 2727 refer to just 18 reference tables such as `R_UOM`
  most of the remaining 385 foreign keys refer to 'single-purpose' reference tables

semantics of data is unclear / documentation is outdated & missing

From Database to Ontology: Mappings

2996 ProSource queries (1801 single-table, but 9 queries are joins of 51–92 tables)
predefined queries for generating reports, filling in forms, etc.

From Database to Ontology: Mappings

2996 ProSource queries (1801 single-table, but 9 queries are joins of 51–92 tables)
predefined queries for generating reports, filling in forms, etc.

mapping design method: **decompose** complex queries into simpler fragments

logical tables for ontology classes & properties

join of the fragments = original query

62 logical tables
180 mapping assertions

# SQL tables/views	1	2	3	4	5	6
# R2RML logical tables	32	8	13	2	2	5

most have filters, 2 have GROUP BY and 8 stored procedure calls

From Database to Ontology: Mappings

2996 ProSource queries (1801 single-table, but 9 queries are joins of 51–92 tables)
predefined queries for generating reports, filling in forms, etc.

mapping design method: **decompose** complex queries into simpler fragments

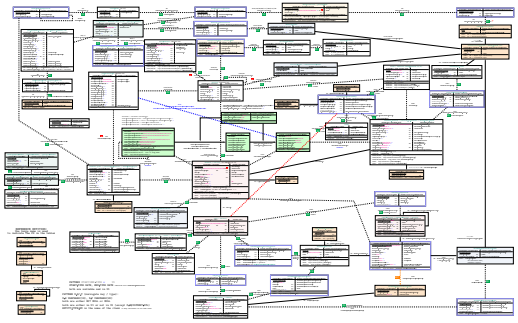
logical tables for ontology classes & properties

join of the fragments = original query

62 logical tables
180 mapping assertions

# SQL tables/views	1	2	3	4	5	6
# R2RML logical tables	32	8	13	2	2	5

most have filters, 2 have GROUP BY and 8 stored procedure calls



small database schema

66 tables with 379 columns, 55 views,
5 materialised views, 4 stored procedures

only 47 foreign keys, 11 of which refer to R_UOM
only 3 foreign keys refer to entity tables

would be required for semantic query optimisation

Summary

we publish a **complete OBDA specification** of the Statoil use case *first!*

purl.org/slegge

- information needs and SPARQL queries
- Subsurface Exploration Ontology
- R2RML mappings to Slegge database
- Slegge database schema SQL script with integrity constraints

Summary

we publish a **complete OBDA specification** of the Statoil use case *first!*

purl.org/slegge

- information needs and SPARQL queries
- Subsurface Exploration Ontology
- R2RML mappings to Slegge database
- Slegge database schema SQL script with integrity constraints
- synthetic data generator is our current work in progress (statistics, implicit constraints) for benchmarking OBDA systems

Summary

we publish a **complete OBDA specification** of the Statoil use case *first!*

purl.org/slegge

- information needs and SPARQL queries
- Subsurface Exploration Ontology
- R2RML mappings to Slegge database
- Slegge database schema SQL script with integrity constraints
- synthetic data generator is our current work in progress (statistics, implicit constraints) for benchmarking OBDA systems

better tools for ontology and *mapping* design / validation are needed