Scientific Data as RDF with Arrays: Tight integration of SciSPARQL queries into MATLAB

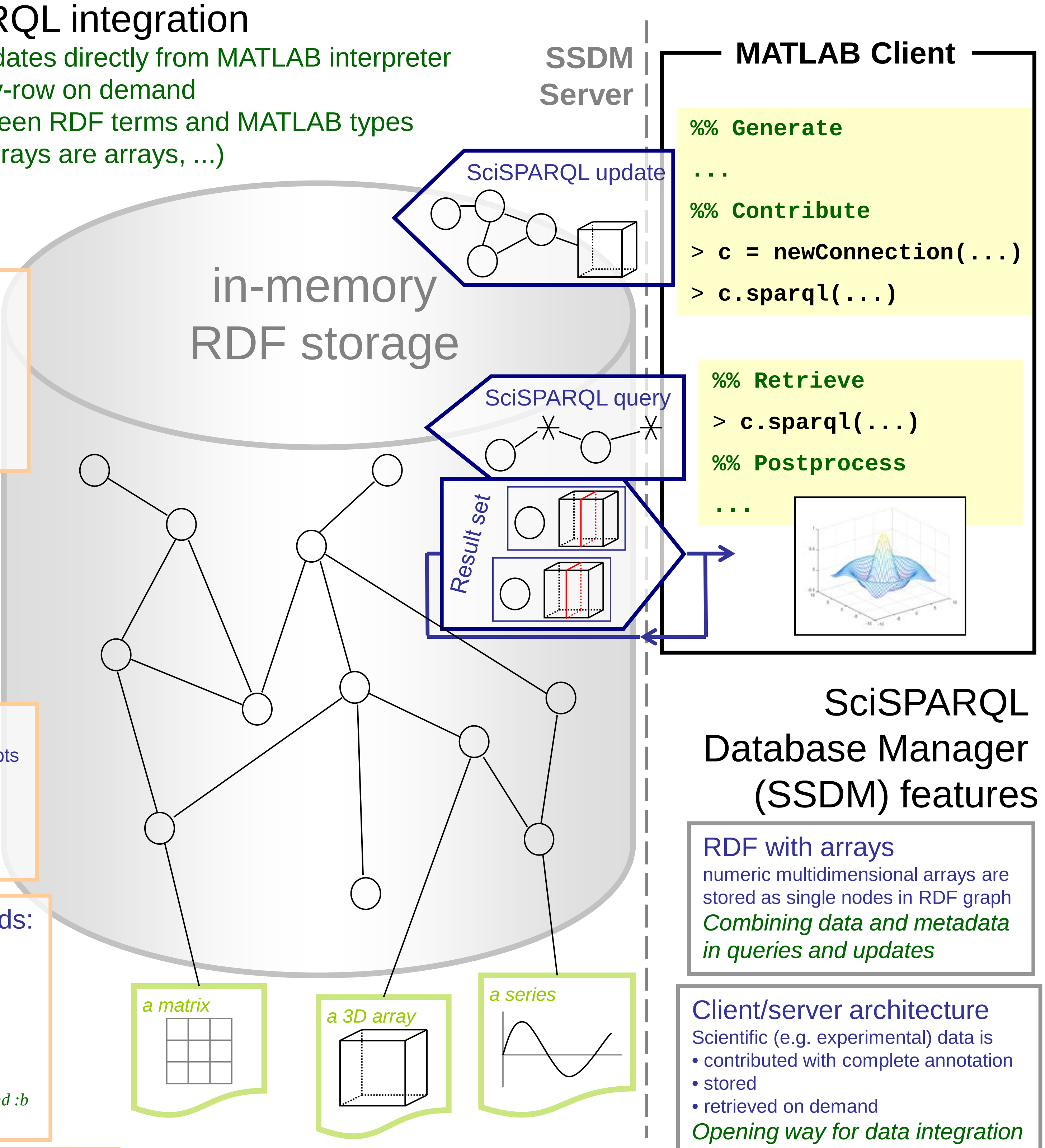
Andrej Andrejev, Xueming He, Tore Risch

Department of Information Technology, Uppsala University

http://it.uu.se/research/group/udbl/SciSPARQL

MATLAB-SciSPARQL integration

- Sending queries and updates directly from MATLAB interpreter
- Retrieving results row-by-row on demand
- Complete mapping between RDF terms and MATLAB types (numbers are numbers, arrays are arrays, ...)



```
%% Generate
...
%% Contribute
> c = newConnection(...)
> c.sparql(...)
```

```
%% Retrieve
> c.sparql(...)
%% Postprocess
...
```

.mat binary files on the server

SciSPARQL features

Array operations:

slicing, projection, transposition

(performed on the server)

```
SELECT (?A[?start:?step:, ?i])
        AS ?result)
 WHERE ...
# slice and project array ?A
```

Intra-array aggregations:

sum(), min(), etc. of array elements

(performed on the server)

```
SELECT (array_sum(?A[:,?i])
        AS ?result)
 WHERE ...
# sum up the column ?i in array ?A
```

Automatic subscript ranges:

variables are bound to available subscripts

```
SELECT ?i, (?A[?i] AS ?result)
 WHERE { ...
        FILTER (mod(?i, 2) = 1)
# return every odd row in ?A
```

Arrays as arithmetic operands:

extended arithmetic operators
and SPARQL 1.1 aggregate functions

(performed on the server)

```
SELECT (AVG(abs(?A-?B))
        AS ?result)
 WHERE { [] :a ?A ; :b ?B }
# get average absolute difference between :a and :b
  properties (element-wise if arrays)
```

Functional RDF views:

parameterized SPARQL queries (and updates)

```
DEFINE FUNCTION sse(?x)
 AS SELECT (array_sum(sqr(?A-?B)) AS ?result)
 WHERE { ?x :a ?A ; :b ?B }
# get sum-of-squared error  between :a and :b properties of ?x
```

Second-order functions:

operate on functions or closures

(performed on the server)

```
SELECT (ARGMIN(sse) AS ?x)
# get the node ?x with minimal SSE

SELECT (ARGMIN(param_sse(*, 1.75)) AS ?result)
# get the node ?x with minimal parameterized SSE
```

Scalable data management

in-memory RDF storage for metadata,
native binary file formats for
massive multidimensional numeric arrays

No performance overhead

Wrapper interface

A mechanism to define RDF views
over storages with different data models
(relational, spreadsheet, etc.)

Making use of existing data

SciSPARQL Database Manager (SSDM) features

RDF with arrays

numeric multidimensional arrays are
stored as single nodes in RDF graph

Combining data and metadata in queries and updates

Client/server architecture

Scientific (e.g. experimental) data is

- contributed with complete annotation
- stored
- retrieved on demand

Opening way for data integration

No data overhead

Massive numeric arrays are projected,
aggregated and filtered as part of
the query answering on the server.

Only the query results are sent over

Extensible server

Foreign functions can be implemented in
Python, Java, C, and used in queries

```
DEFINE FUNCTION
  pyplus(?a ?b)
 AS PYTHON 'foreign.plus';
```

```
def plus(a, b):
  return a+b;
```

Making use of existing libraries

Andrej Andrejev, e-Science / ISWC - October 2014