SciQL

Bridging the Gap between Science and Relational DBMS

Ying Zhang, Martin Kersten, Niels Nes
CWI Amsterdam

Dec. 03, 2012
Who needs arrays anyway?

<table>
<thead>
<tr>
<th>Field</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seismology</td>
<td>1-D waveforms, 3-D spatial data</td>
</tr>
<tr>
<td>Astronomy</td>
<td>temporal ordered rasters</td>
</tr>
<tr>
<td>Climate simulation</td>
<td>temporal ordered grid</td>
</tr>
<tr>
<td>Remote sensing</td>
<td>images of 2-D or higher</td>
</tr>
<tr>
<td>Genomics</td>
<td>ordered DNA strings</td>
</tr>
<tr>
<td>Forensics</td>
<td>images, strings, graphs</td>
</tr>
</tbody>
</table>

*Scientists love arrays: FITS, MSEED, NETCDF, HDF5... but also use: lists, tables, XML, ...*
Arrays In DBMS

- Research issues already in the 80’s
- Algebraic frameworks
  - (S)RAM, AQL, AML, ...
- SQL language extension
  - RasQL, AQuery, SRQL, ...
  - a notion of order
  - collection type, C-style arrays
  - aggregation functions over arrays
- OODB, multi-dimensional DBMS, Sequence DBMS, ...
- The Longhorn Array Database
- RasDaMan
  - Store large arrays in chunks as BLOBs
  - Array query (RasQL) optimisation on top of DBMS
  - Known to work up to 12 TBs!
- PostgreSQL 8.1
- SciDB
  - Array DBMS from scratch
  - Overlapping chunks for parallel execution
What Not RDBMS?

- SQL is difficult
  - Appropriate array denotations?
  - Functional complete operation set?

- DBMSs are slow
  - Too much overhead
  - Size limitations (due to BLOB representations)?
  - Existing foreign files?
  - Scale
  - ...

An array query language based on SQL:2003

- Lower the entrance fee to RDBMSs
- Reveal optimisation opportunities

Distinguish features (Kersten et al., AD’11; Zhang et al. IDEAS2011):

- Arrays and tables as first class citizens of DBMSs
- Seamless integration of relational and array paradigms
- Named dimensions with constraints
- Flexible structure-based grouping
CREATE ARRAY A1 ( 
  x INT DIMENSION [0:1:4], y INT DIMENSION [0:1:4], 
  v FLOAT DEFAULT 0.0);
CREATE ARRAY A1 (x INT DIMENSION [0:1:4], y INT DIMENSION [0:1:4], v FLOAT DEFAULT 0.0);

dimensions, any scalar data type
CREATE ARRAY A1 (
  x INT DIMENSION [0:1:4], y INT DIMENSION [0:1:4],
  v FLOAT DEFAULT 0.0;
)

dimensional range: 
[(start|*) : (step|*) : (stop|*)]
Array Definitions

CREATE ARRAY A1 (
    x INT DIMENSION [0:1:4], y INT DIMENSION [0:1:4],
    v FLOAT DEFAULT 0.0);

Cell values, any column data type
CREATE ARRAY A1 (  
  x INT DIMENSION [0:1:4], y INT DIMENSION [0:1:4],  
  v FLOAT DEFAULT 0.0);
CREATE ARRAY A2 ( 
  x INT DIMENSION[1:*],
  y INT DIMENSION,
  v FLOAT DEFAULT 0.0);
CREATE ARRAY A2 (  
x INT DIMENSION[1:*],
y INT DIMENSION,
v FLOAT DEFAULT 0.0);

INSERT INTO A2 VALUES  
(1,0,5.5), (1,1,0.4), (2,2,4.5);

Unbounded array
Array Definitions

Unbounded array

CREATE ARRAY A2 (  
x INT DIMENSION[1:*],  
y INT DIMENSION,  
v FLOAT DEFAULT 0.0);  

INSERT INTO A2 VALUES  
(1,0,5.5), (1,1,0.4), (2,2,4.5);  

```
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>4.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.4</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.5</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Array Definitions

Unbounded array

CREATE ARRAY A2 (x INT DIMENSION[1:*], y INT DIMENSION, v FLOAT DEFAULT 0.0);

INSERT INTO A2 VALUES (1,0,5.5), (1,1,0.4), (2,2,4.5);

```sql
CREATE ARRAY A2 (x INT DIMENSION[1:*], y INT DIMENSION, v FLOAT DEFAULT 0.0);
```

![Unbounded array diagram](image)
CREATE ARRAY A1 (  
x INT DIMENSION[0:1:4],  
y INT DIMENSION[0:1:4],  
v FLOAT DEFAULT 0.0);

SELECT x, y, v FROM A1;

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>v</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>0</td>
<td>3</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0.0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0.0</td>
</tr>
</tbody>
</table>

full materialisation!
CREATE TABLE T2 (x INT, y INT, v FLOAT DEFAULT 0.0);

INSERT INTO T2 VALUES (1,0,5.5), (1,1,0.4), (2,2,4.5), (1,1,1.3);

An unbounded array

- dimension ranges derived from the minimal bounding box
- cells values from the table or the column default
- duplicates are overwritten arbitrarily

SELECT [x], [y], v FROM T2;
DELETE FROM A1 WHERE x = 1;

creates holes in the array
UPDATE A1 SET v = 0.5 WHERE y = 1;

INSERT INTO A1 VALUES (0,1,0.5), (1,1,0.5), (2,1,0.5), (3,1,0.5);

set/change cell values
overwrite existing values
CREATE VIEW A2 AS
SELECT [x], [y], v FROM A1[1:3][1:3];

Array view A2
CREATE ARRAY VIEW A3 (  
x INT DIMENSION [-1:1:5],
y INT DIMENSION [-1:1:5],
w FLOAT DEFAULT 0.0) AS  
SELECT [x], [y], v FROM A1;
CREATE ARRAY VIEW A3 (  
x INT DIMENSION [-1:1:5],  
y INT DIMENSION [-1:1:5],  
w FLOAT DEFAULT 0.0) AS  
SELECT [x], [y], v FROM A1;
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2];
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2];
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2];

Anchor point: A1[x][y]
```
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2];
```
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2];
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2];
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2];
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x:x+2][y:y+2]
HAVING x mod 2 = 0 AND y mod 2 = 0;

Anchor point: A1[x][y]
SELECT [x], [y], AVG(v) FROM A1
GROUP BY A1[x-1][y], A1[x][y-1],
        A1[x][y], A1[x+1][y], A1[x][y+1];
the LOw Frequency ARray radio telescope
zone (Gray et al. 2006)

ra DOUBLE, decl DOUBLE, 
ra_err DOUBLE, decl_err DOUBLE, 
flux DOUBLE, ...

2012-12-03

Ying Zhang
CREATE ARRAY LOFARsrc (  
  zone INT DIMENSION[-90:1:91], mrdn INT DIMENSION[0:1:360],  
  ts TIMESTAMP DIMENSION, freq INT DIMENSION[30:10:241],  
  id INT DIMENSION[0:1:*], stks CHAR(1) DIMENSION  
    CHECK(stks=`I' OR stks=`Q' OR stks=`U' OR stks=`V'),  
  ra DOUBLE, decl DOUBLE, ra_err DOUBLE, decl_err DOUBLE,  
  flux DOUBLE, ...);
Similarity of the flux of a LOFAR source at frequencies 30 MHz and 200 MHz

cross-correlation of two time series
Cross-Correlation

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>1</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>idx</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Cross-Correlation

**Cr.idx = -3**

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

**F [3 : 4]**

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

**G [0 : 1]**

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>1</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

**Cr**

<table>
<thead>
<tr>
<th>idx</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Cross-Correlation

#### Cr.idx = -2

<table>
<thead>
<tr>
<th>idx</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>2</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### F [2 : 4]

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

#### G [0 : 2]

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>1</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>
## Cross-Correlation

The image illustrates the concept of cross-correlation with three tables:

### Table F

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

### Table G

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>1</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

### Table Cr

<table>
<thead>
<tr>
<th>idx</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>2</td>
<td>16</td>
<td>47</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The notation $\text{Cr}$.idx = -1 indicates the cross-correlation index.

The vectors $\text{F}[1:4]$ and $\text{G}[0:3]$ are also shown in the image.
Cross-Correlation

\[
\text{Cr.idx} = 0
\]

\[
\begin{align*}
\text{F} & : \begin{array}{c|cccc} 
\text{idx} & 0 & 1 & 2 & 3 \\
\text{val} & 4 & 3 & 6 & 2 \\
\end{array} \\
\text{G} & : \begin{array}{c|ccc} 
\text{idx} & 0 & 1 & 2 \\
\text{val} & 1 & 5 & 7 \\
\end{array} \\
\text{Cr} & : \begin{array}{c|cccc} 
\text{idx} & -3 & -2 & -1 & 0 & 1 & 2 \\
\text{val} & 2 & 16 & 47 & 61 \\
\end{array}
\end{align*}
\]
### Cross-Correlation

#### Cr.idx = 1

<table>
<thead>
<tr>
<th>idx</th>
<th>val</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3</td>
<td>2</td>
</tr>
<tr>
<td>-2</td>
<td>16</td>
</tr>
<tr>
<td>-1</td>
<td>47</td>
</tr>
<tr>
<td>0</td>
<td>61</td>
</tr>
<tr>
<td>1</td>
<td>41</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

#### F [0 : 2]

<table>
<thead>
<tr>
<th>idx</th>
<th>val</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

#### G [1 : 3]

<table>
<thead>
<tr>
<th>idx</th>
<th>val</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
</tr>
</tbody>
</table>

#### F

<table>
<thead>
<tr>
<th>idx</th>
<th>val</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

#### G

<table>
<thead>
<tr>
<th>idx</th>
<th>val</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
</tr>
</tbody>
</table>

#### Cr

<table>
<thead>
<tr>
<th>idx</th>
<th>val</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3</td>
<td>2</td>
</tr>
<tr>
<td>-2</td>
<td>16</td>
</tr>
<tr>
<td>-1</td>
<td>47</td>
</tr>
<tr>
<td>0</td>
<td>61</td>
</tr>
<tr>
<td>1</td>
<td>41</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
Cross-Correlation

Cr.idx = 2

F [0 : 1]

G [2 : 3]
Cross-Correlation

<table>
<thead>
<tr>
<th>Cr.idx</th>
<th>2</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Cr.idx</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>2</td>
<td>16</td>
<td>47</td>
<td>61</td>
<td>41</td>
<td>28</td>
</tr>
</tbody>
</table>

| Cr.idx = 2 |
|---|---|

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

| G [2 : 3] |
|---|---|

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>1</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

| F [0 : 1] |
|---|---|

<table>
<thead>
<tr>
<th>idx</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>val</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>
DECLARE fcnt INT, gcnt INT;
SET fcnt = SELECT COUNT(*) FROM LOFARsrc[*][*][*][30][11]['I'];
SET gcnt = SELECT COUNT(*) FROM LOFARsrc[*][*][*][200][11]['I'];

CREATE ARRAY VIEW F (idx INT DIMENSION[0:1:fcnt], flux DOUBLE DEFAULT 0.0) AS SELECT flux FROM LOFARsrc[*][*][*][30][11]['I'];
CREATE ARRAY VIEW G (idx INT DIMENSION[0:1:gcnt], val DOUBLE DEFAULT 0.0) AS SELECT flux FROM LOFARsrc[*][*][*][200][11]['I'];

CREATE ARRAY CrCorr30_200 (idx INT DIMENSION[-fcnt+1:1:gcnt], val DOUBLE DEFAULT 0.0);
INSERT INTO CrCorr SELECT SUM(F.flux * G.flux) FROM F, G, CrCorr30_200 AS C
GROUP BY F[MAX(0, -C.idx) : MIN(fcnt, gcnt-C.idx)], G[MAX(0, C.idx) : MIN(gcnt, fcnt+C.idx)];
DECLARE fcnt INT, gcnt INT;
SET fcnt = SELECT COUNT(*) FROM LOFARsrc[*][*][30][11]['I'];
SET gcnt = SELECT COUNT(*) FROM LOFARsrc[*][*][200][11]['I'];

CREATE ARRAY VIEW F (idx INT DIMENSION[0:1:fcnt], flux DOUBLE DEFAULT 0.0) AS SELECT flux FROM LOFARsrc[*][*][30][11]['I'];
CREATE ARRAY VIEW G (idx INT DIMENSION[0:1:gcnt], val DOUBLE DEFAULT 0.0) AS SELECT flux FROM LOFARsrc[*][*][200][11]['I'];

CREATE ARRAY CrCorr30_200 (idx INT DIMENSION[-fcnt+1:1:gcnt], val DOUBLE DEFAULT 0.0);
INSERT INTO CrCorr SELECT SUM(F.flux * G.flux) FROM F, G, CrCorr30_200 AS C
GROUP BY F[MAX(0, -C.idx) : MIN(fcnt, gcnt-C.idx)], G[MAX(0, C.idx) : MIN(gcnt, fcnt+C.idx)];

ra DOUBLE,
decl DOUBLE,
ra_err DOUBLE,
decl_err DOUBLE,
flux DOUBLE,
...
LOFAR Use Case

 DECLARE fcnt INT, gcnt INT;
 SET fcnt = SELECT COUNT(*) FROM LOFARsrc[*][*][30][11]['I'];
 SET gcnt = SELECT COUNT(*) FROM LOFARsrc[*][*][200][11]['I'];

 CREATE ARRAY VIEW F (idx INT DIMENSION[0:1:fcnt], flux DOUBLE DEFAULT 0.0) AS SELECT flux FROM LOFARsrc[*][*][30][11]['I'];
 CREATE ARRAY VIEW G (idx INT DIMENSION[0:1:gcnt], val DOUBLE DEFAULT 0.0) AS SELECT flux FROM LOFARsrc[*][*][200][11]['I'];

 CREATE ARRAY CrCorr30_200 (idx INT DIMENSION[-fcnt+1:1:gcnt], val DOUBLE DEFAULT 0.0);
 INSERT INTO CrCorr SELECT SUM(F.flux * G.flux) FROM F, G, CrCorr30_200 AS C
 GROUP BY F[MAX(0, -C.idx) : MIN(fcnt, gcnt-C.idx)], G[MAX(0, C.idx) : MIN(gcnt, fcnt+C.idx)];
Your use cases?
Your problems?
What do you need?
How can SciQL help?