



The Metanautix Quest Data Compute Engine and Media Applications

Larry Cutler, VP Solutions, Metanautix

Nikos Michalakakis, Lead Engineer, Metanautix

Metanautix Quest Background

Metanautix Quest Background



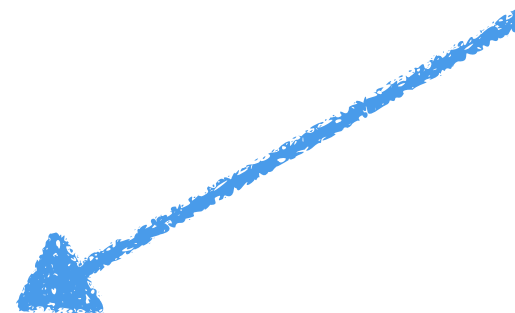
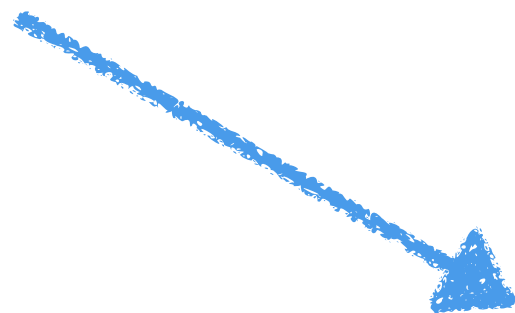
Google
Dremel



Metanautix Quest Background



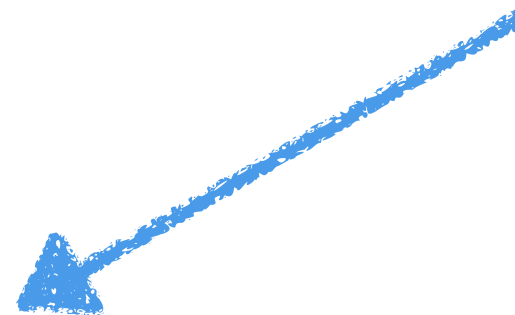
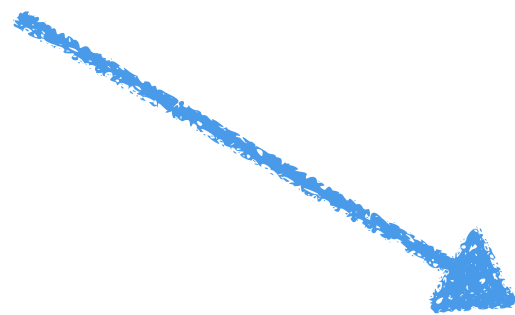
Google
Dremel



Metanautix Quest Background



Google
Dremel

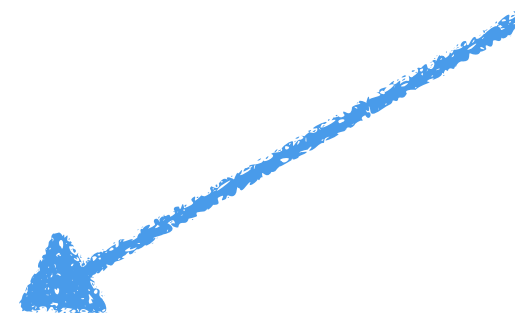
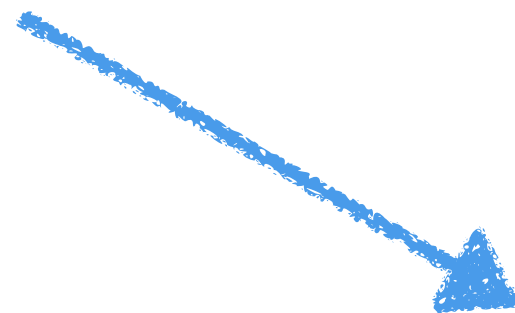


 **Metanautix**

Metanautix Quest Background



Google
Dremel



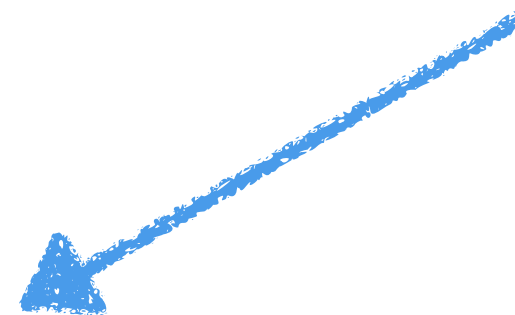
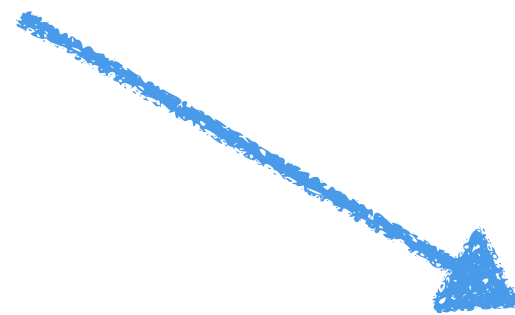
 **Metanautix**



Metanautix Quest Background



Google
Dremel

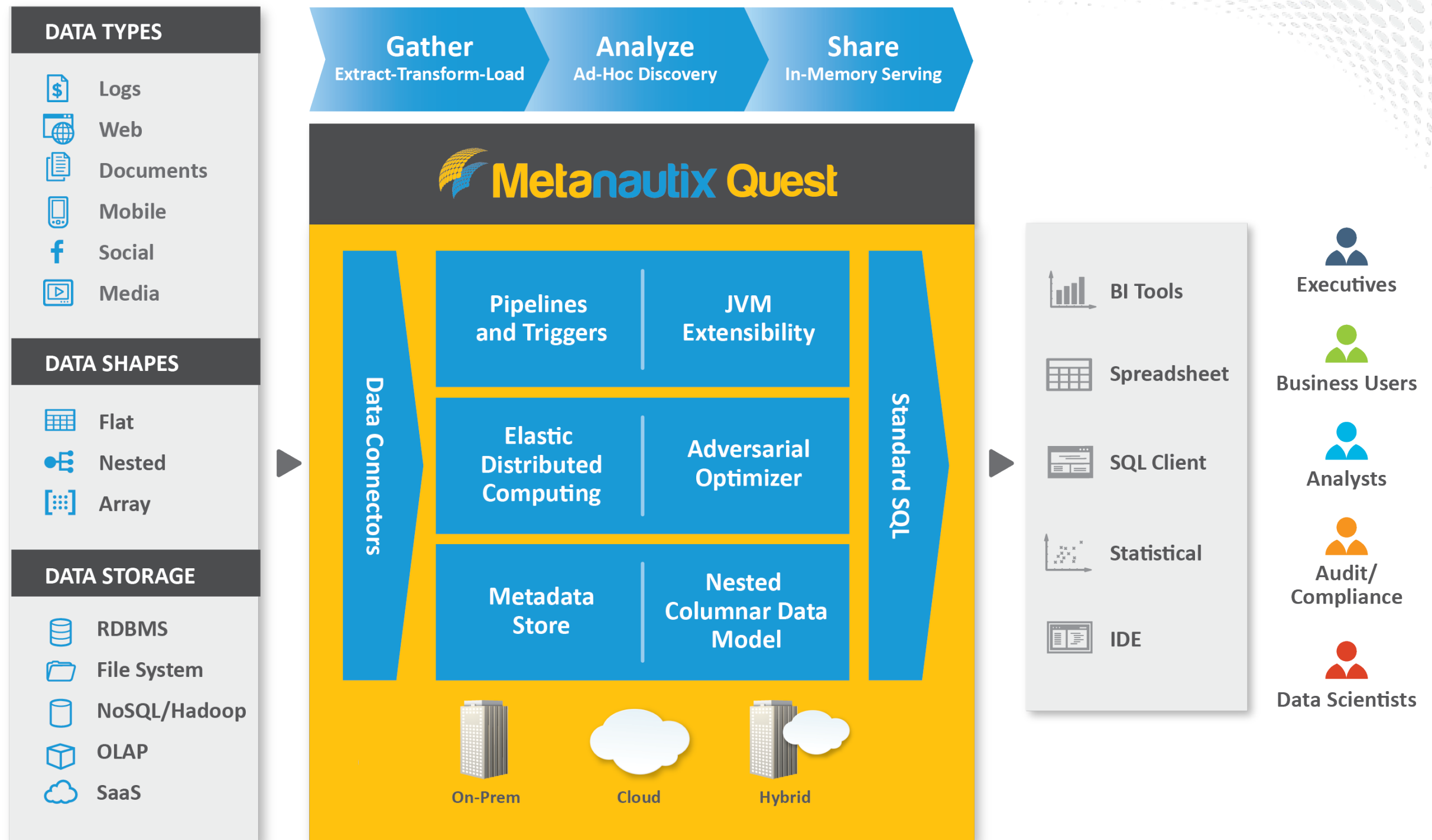


 **Metanautix**

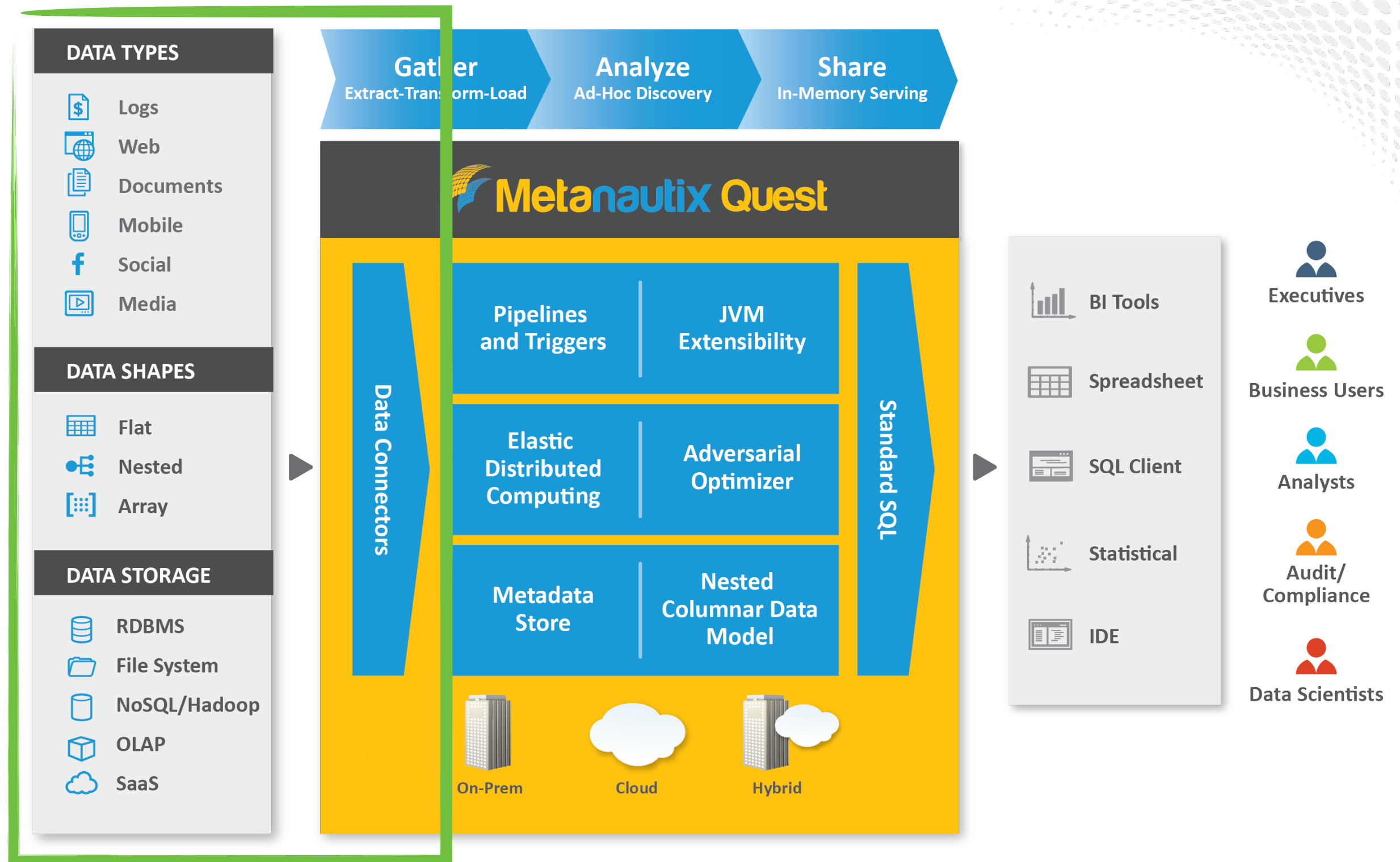


QuestTM

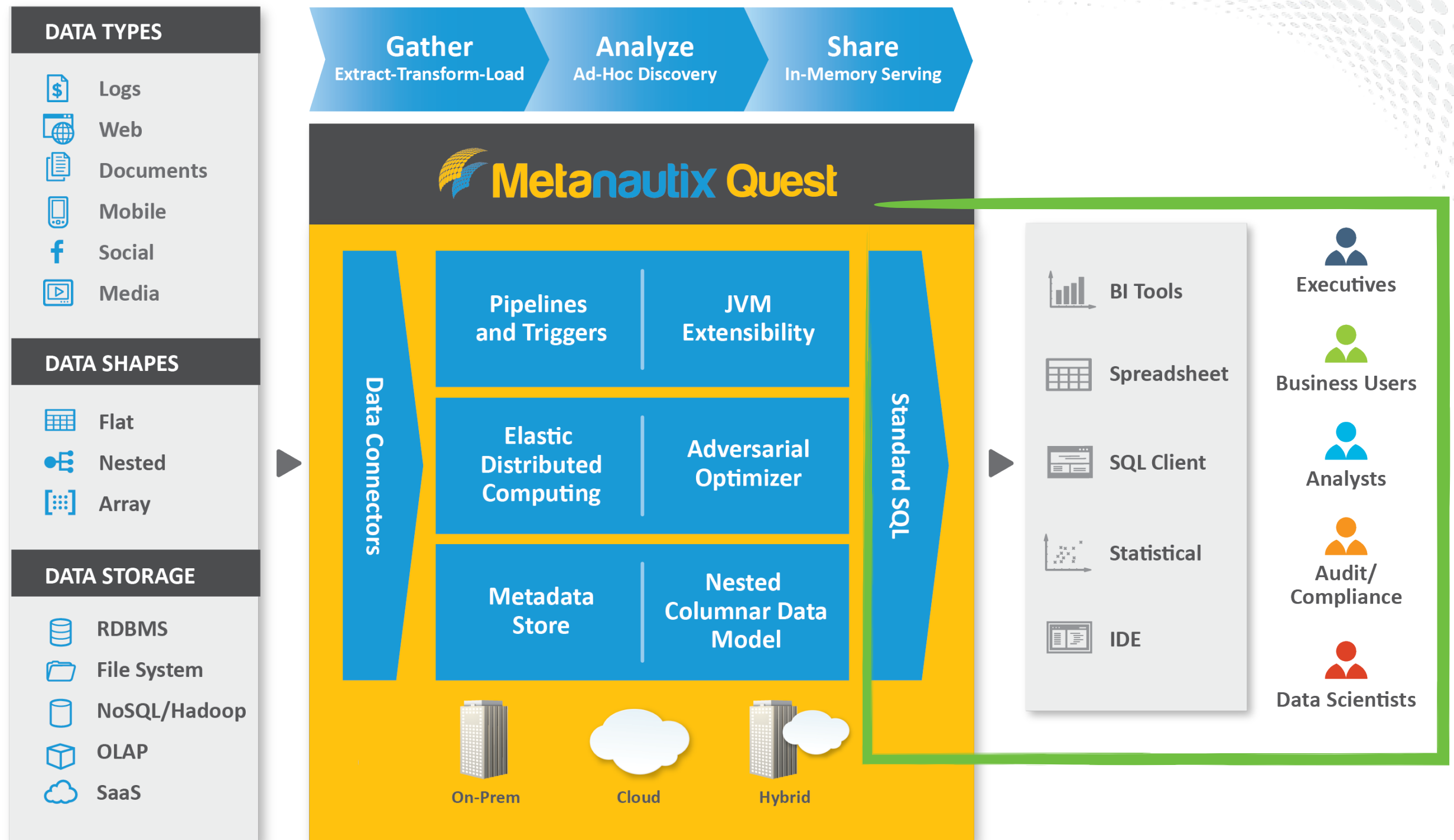
Quest: Converged ETL and Ad-Hoc Analysis



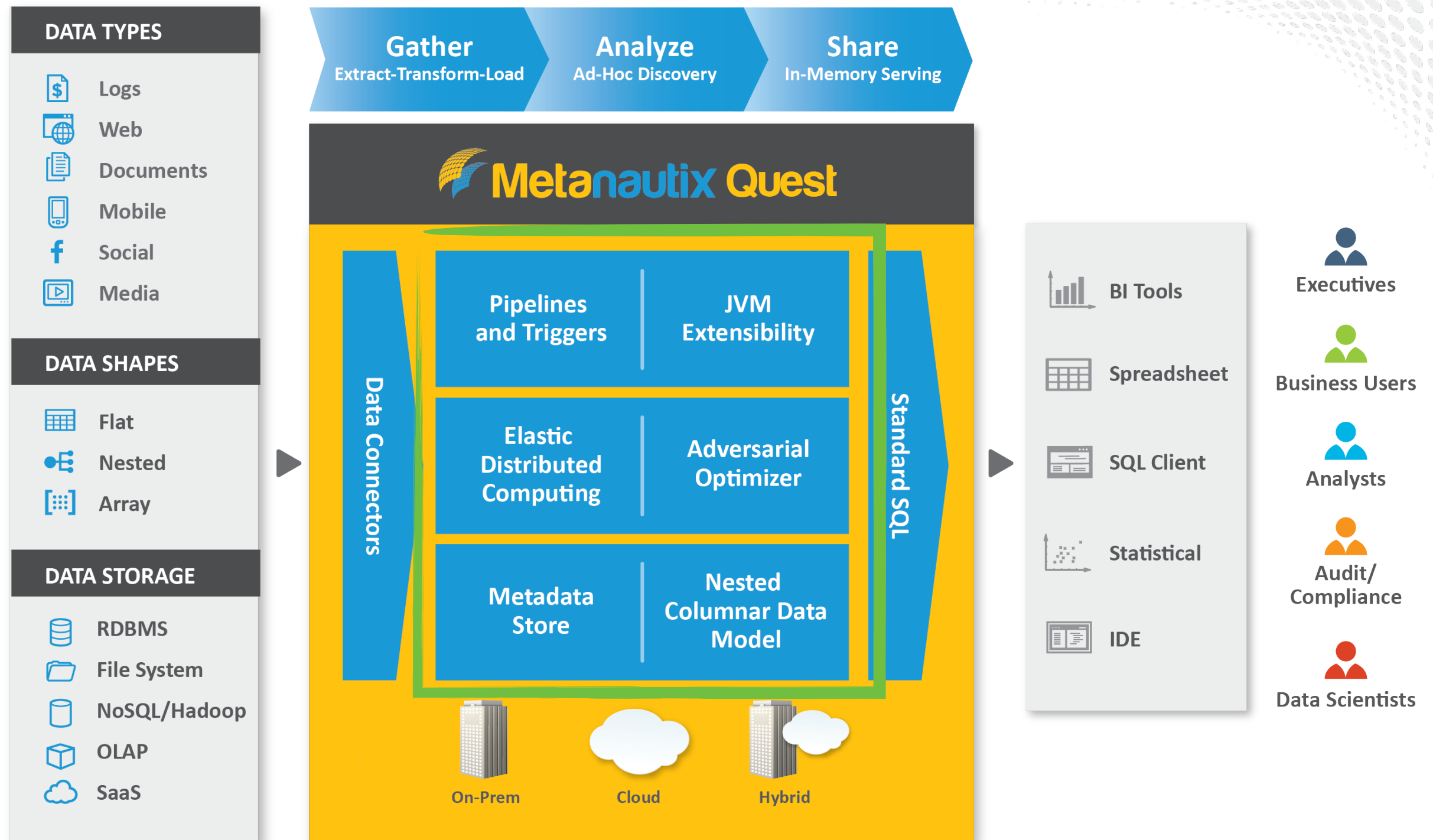
Quest: Converged ETL and Ad-Hoc Analysis



Quest: Converged ETL and Ad-Hoc Analysis

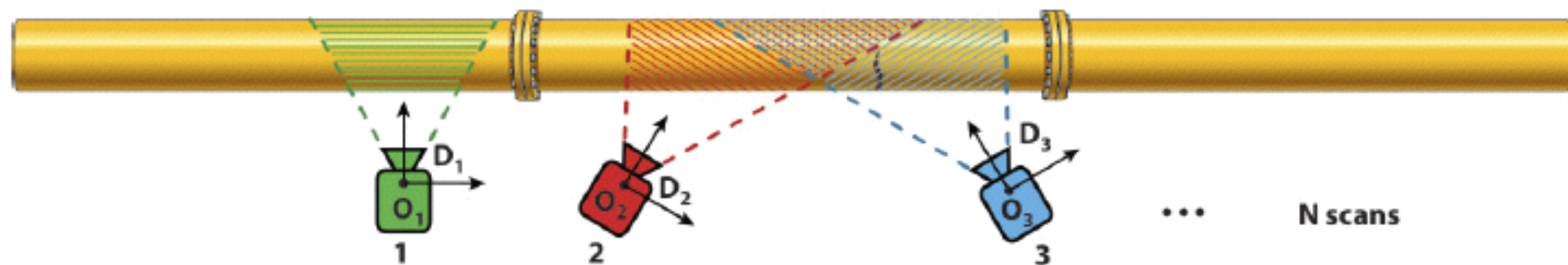


Quest: Converged ETL and Ad-Hoc Analysis



Detecting Failures in Utility Gas Pipes

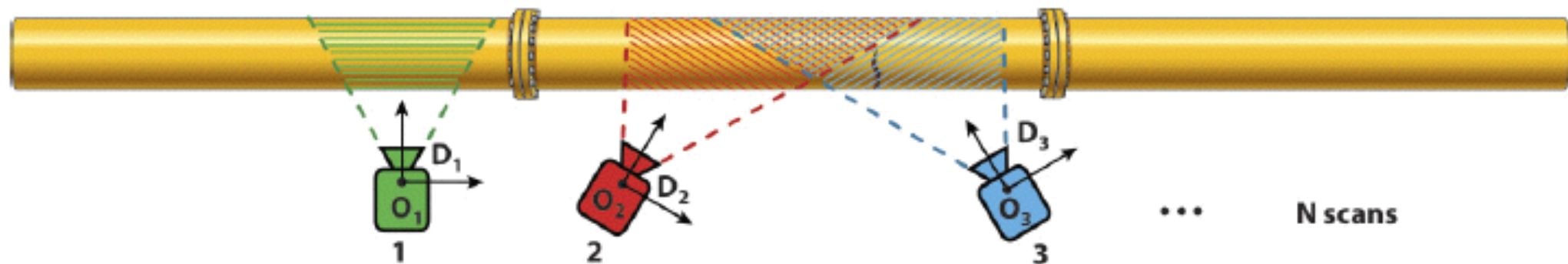
Detecting Failures in Utility Gas Pipes



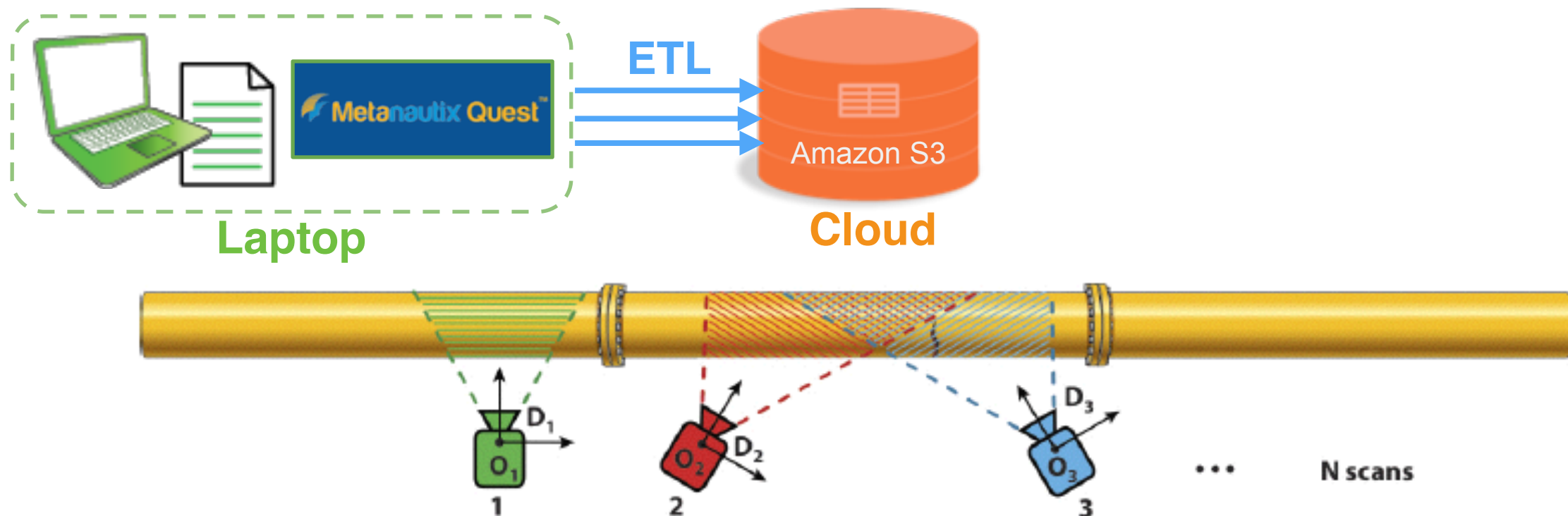
Detecting Failures in Utility Gas Pipes



Laptop



Detecting Failures in Utility Gas Pipes



Detecting Failures in Utility Gas Pipes

```
CREATE VIEW latest
AS SELECT
FROM ply_files
WHERE
w > max_watermark
```



Detecting Failures in Utility Gas Pipes

```
CREATE VIEW latest  
AS SELECT  
FROM ply_files  
WHERE  
w > max_watermark
```

```
CREATE TABLE  
LOCATION 's3'  
AS SELECT  
FROM latest
```



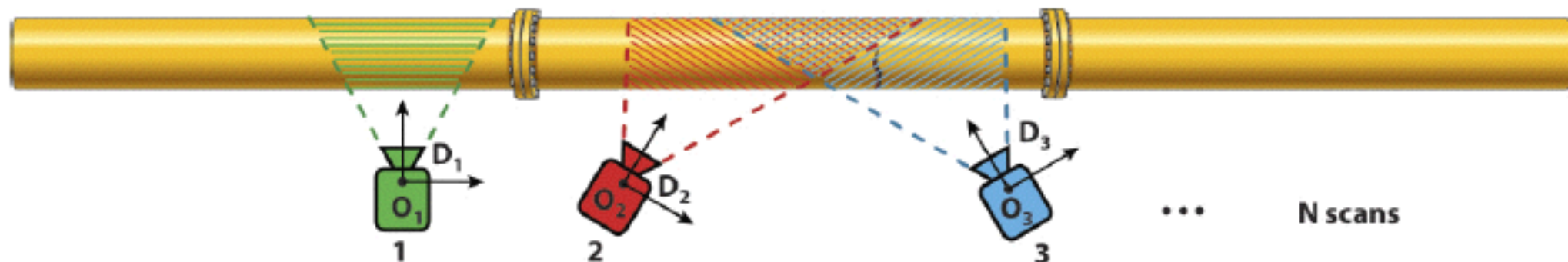
Laptop

ETL

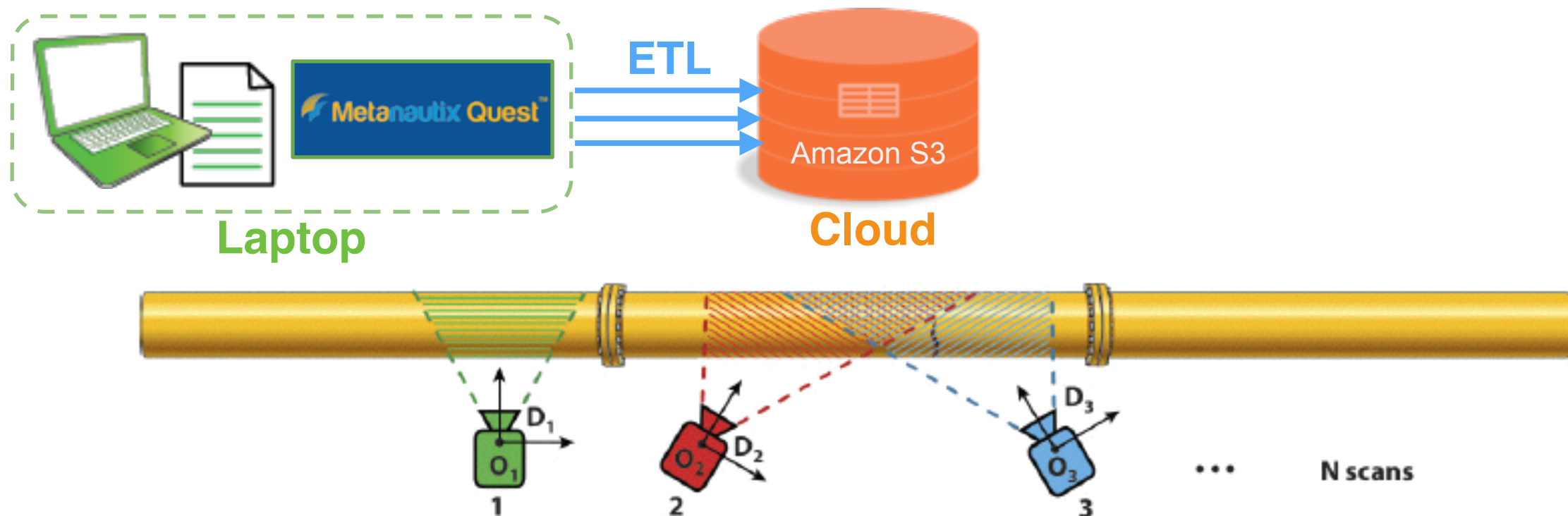
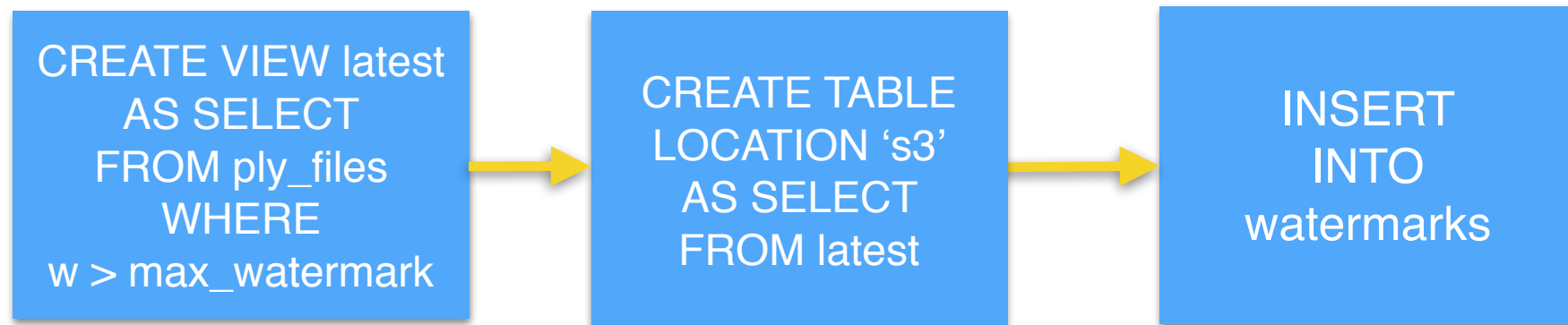


Amazon S3

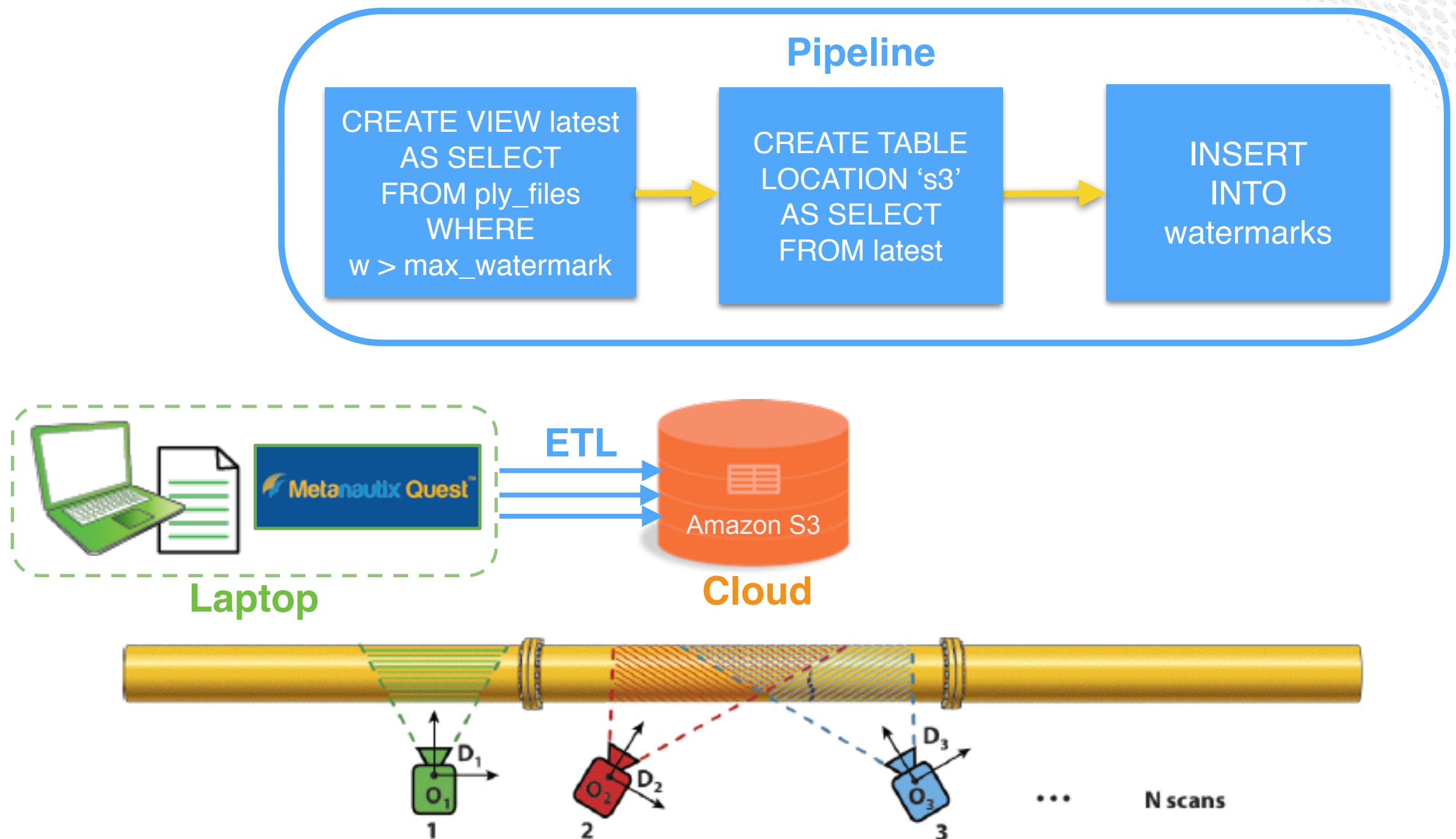
Cloud



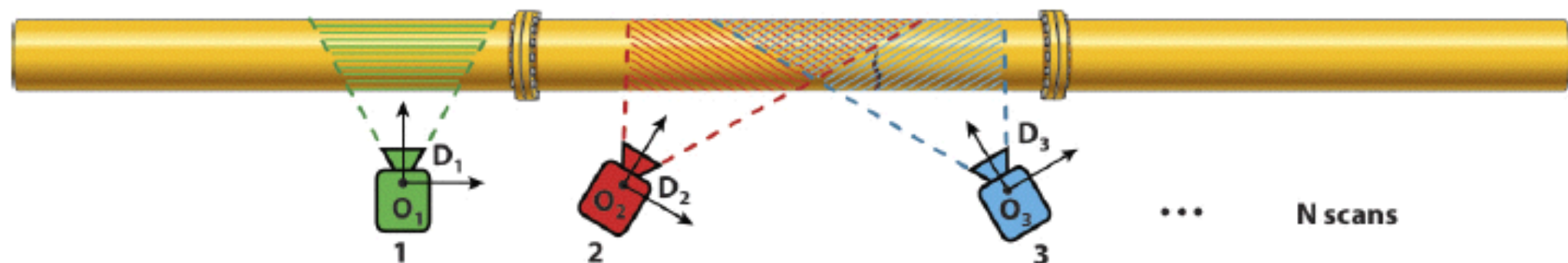
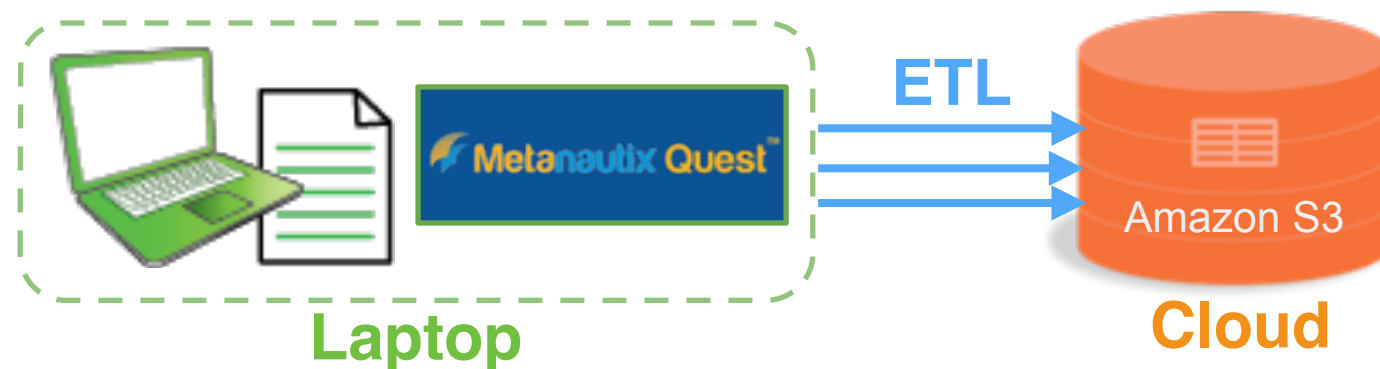
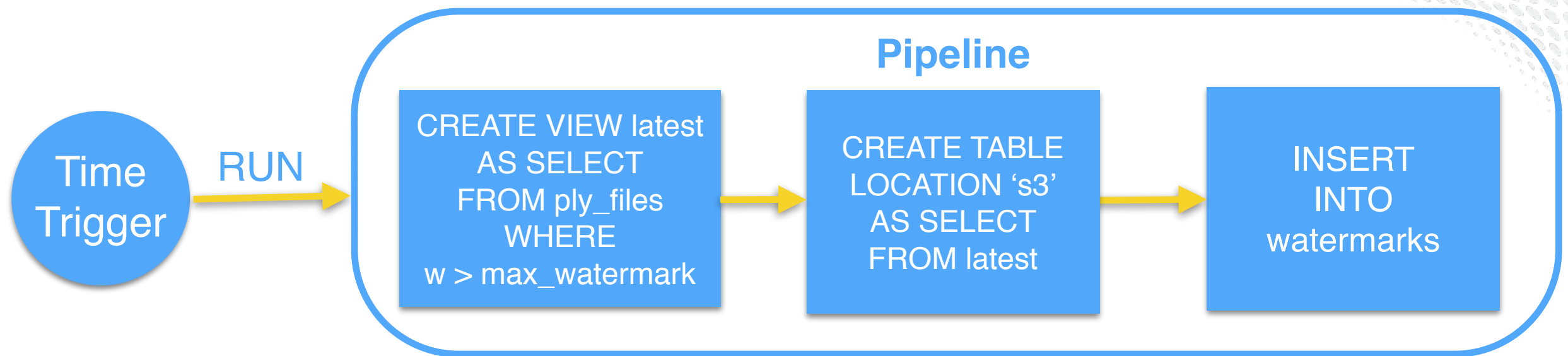
Detecting Failures in Utility Gas Pipes



Detecting Failures in Utility Gas Pipes



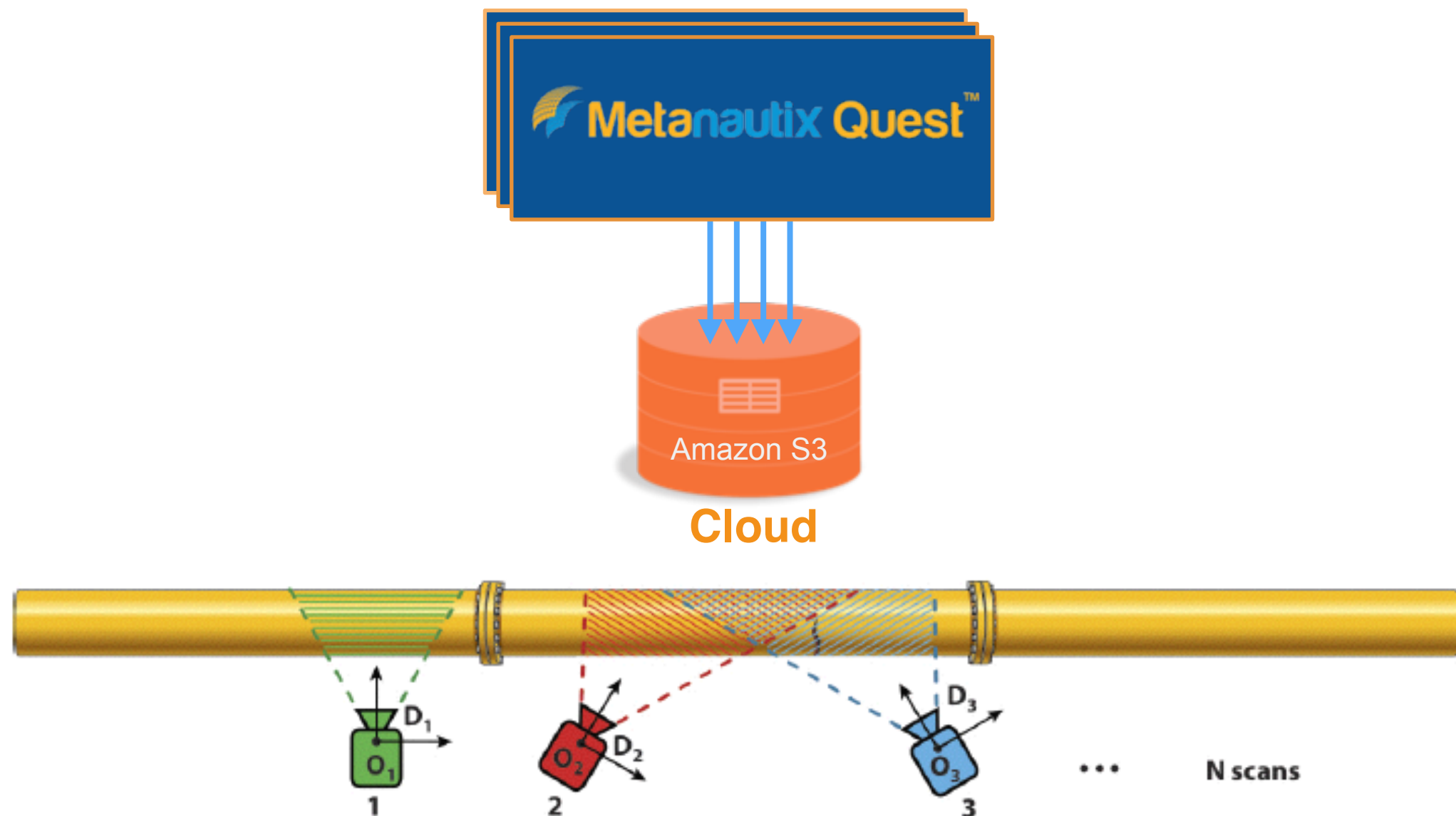
Detecting Failures in Utility Gas Pipes



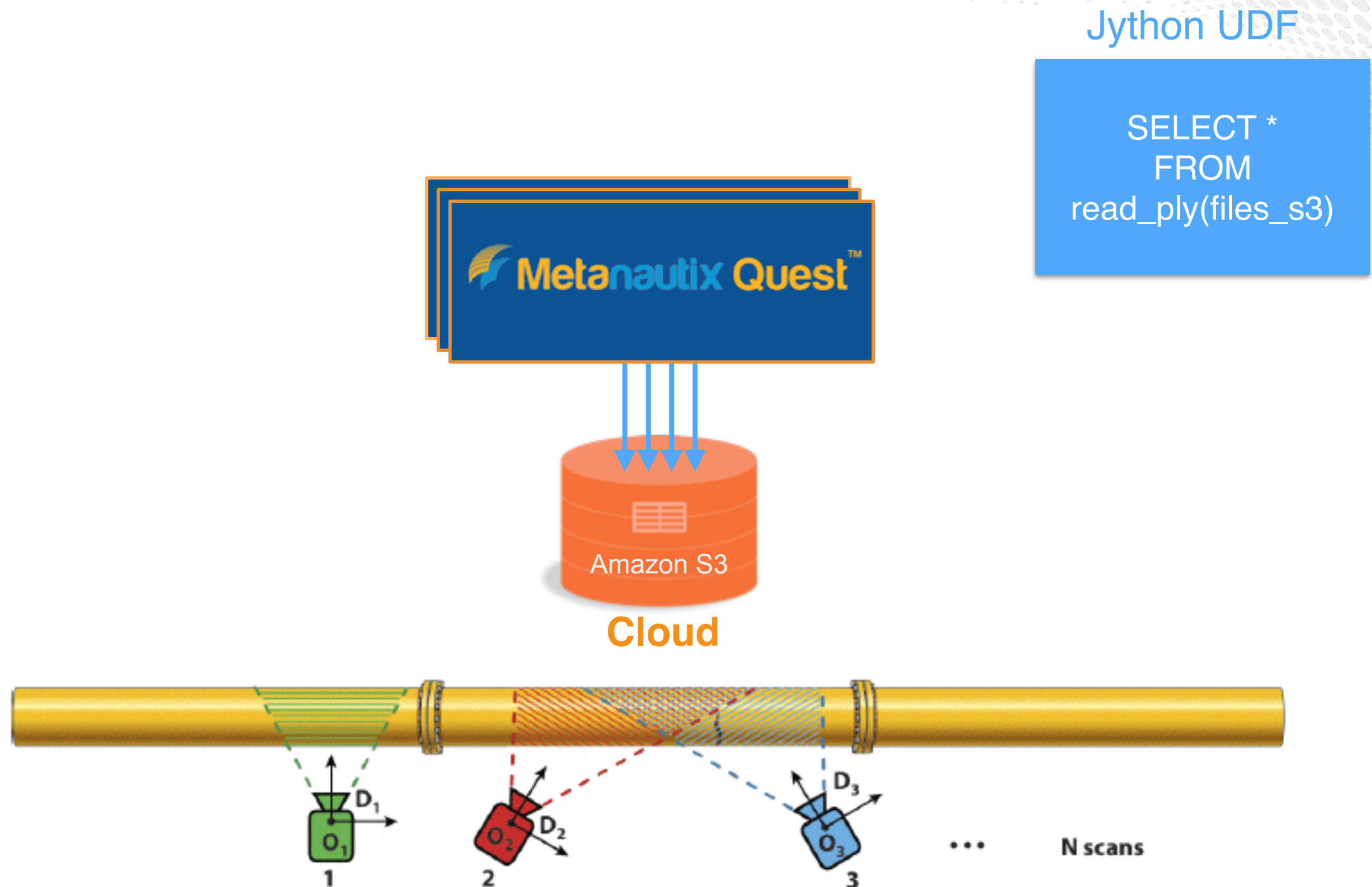
Detecting Failures in Utility Gas Pipes



Detecting Failures in Utility Gas Pipes



Detecting Failures in Utility Gas Pipes



Detecting Failures in Utility Gas Pipes

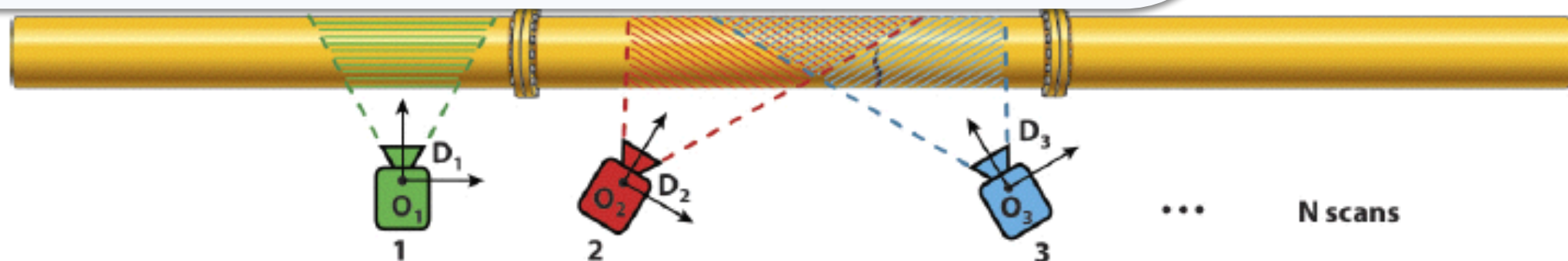
```
CREATE TYPE ModelData AS (vidx INTEGER, fidx INTEGER, x  
REAL, y REAL, z REAL, r REAL, g REAL, b REAL, v0 INTEGER,  
v1 INTEGER, v2 INTEGER);
```

```
CREATE FUNCTION read_ply(f VARCHAR) RETURNS ModelData  
LANGUAGE JYTHON AS $$
```

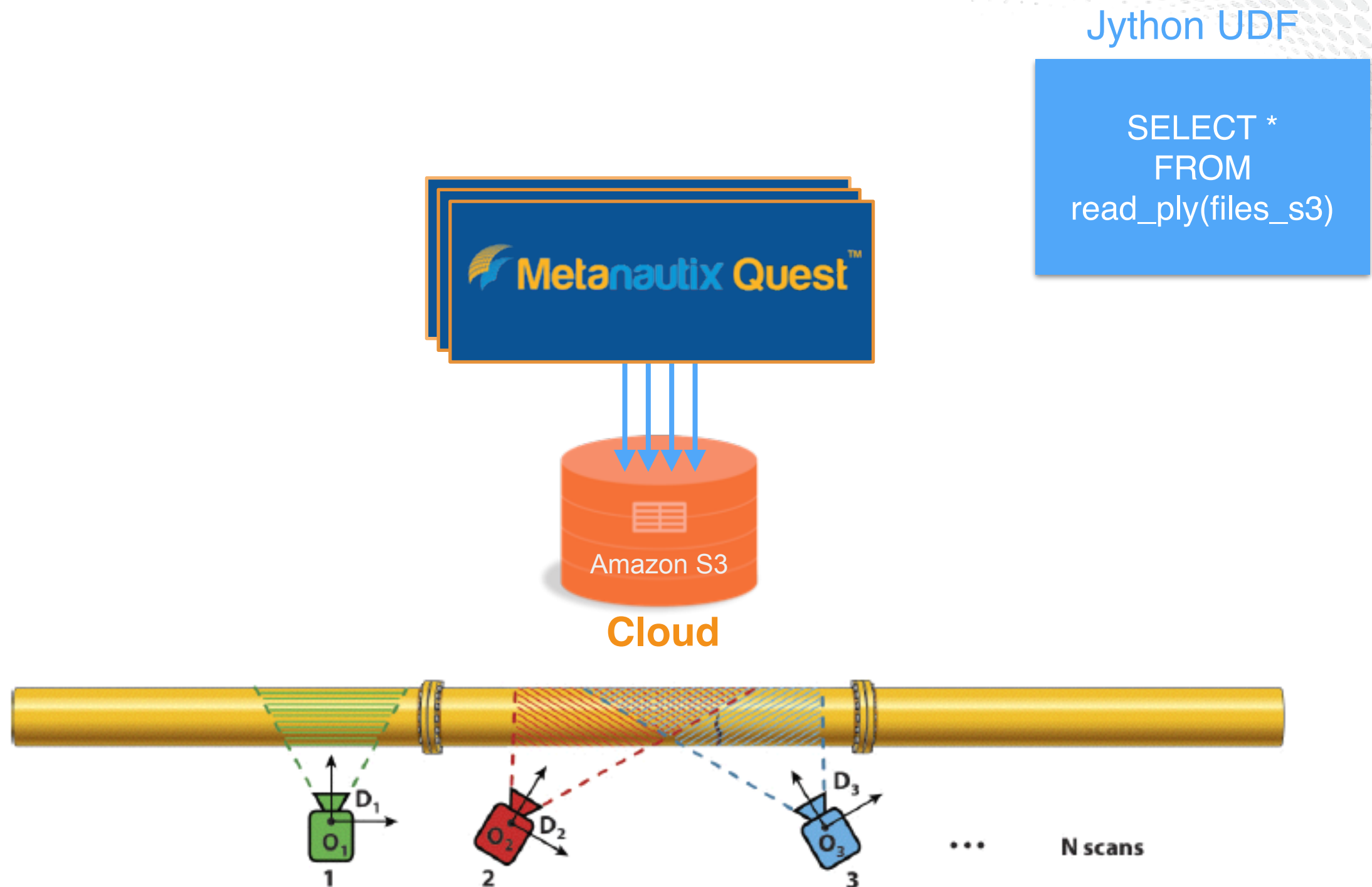
```
def skw_read_3d_data(f):  
    from metanautix.skwreader import SkwReader  
    from metanautix.scangrid import ScanGrid  
    m=SkwReader(f).get_scan_grid().get_downsampled().  
        get_model()  
    vs=m.get_vertices()  
    fs=m.get_faces()  
    result=[]  
    for i in xrange(0,len(vs)):  
        . . .
```

Jython UDF

```
SELECT *  
FROM  
read_ply(files_s3)
```



Detecting Failures in Utility Gas Pipes



Detecting Failures in Utility Gas Pipes

Metadata

Key	Value
Origin_GPS	O_1^A
Direction	D_1^R
Operator	Bud

Points

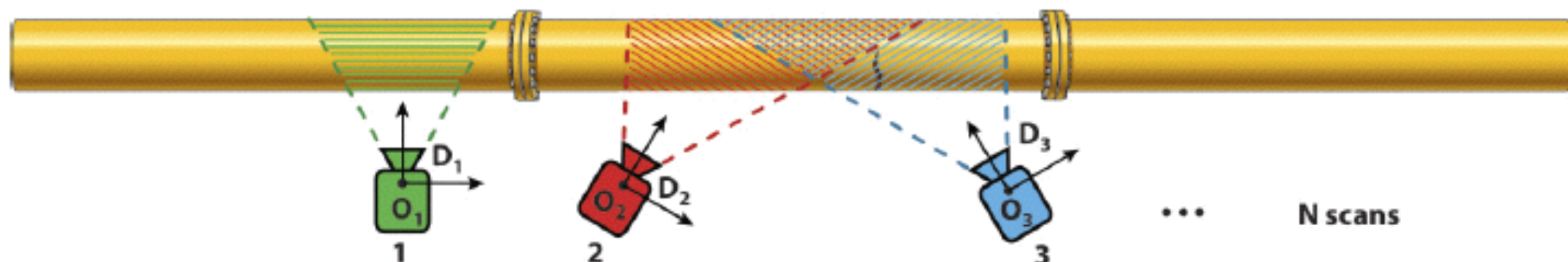
Position	Color
$P_{1,1}^R$	
$P_{1,2}^R$	
$\vdots$	
$P_{1,M}^R$	

Jython UDF

```
SELECT *  
FROM  
read_ply(files_s3)
```



Cloud



Detecting Failures in Utility Gas Pipes

Metadata

Key	Value
Origin_GPS	O_1^A
Direction	D_1^R
Operator	Bud

Points

Position	Color
$P_{1,1}^R$	
$P_{1,2}^R$	
$\vdots$	
$P_{1,M}^R$	



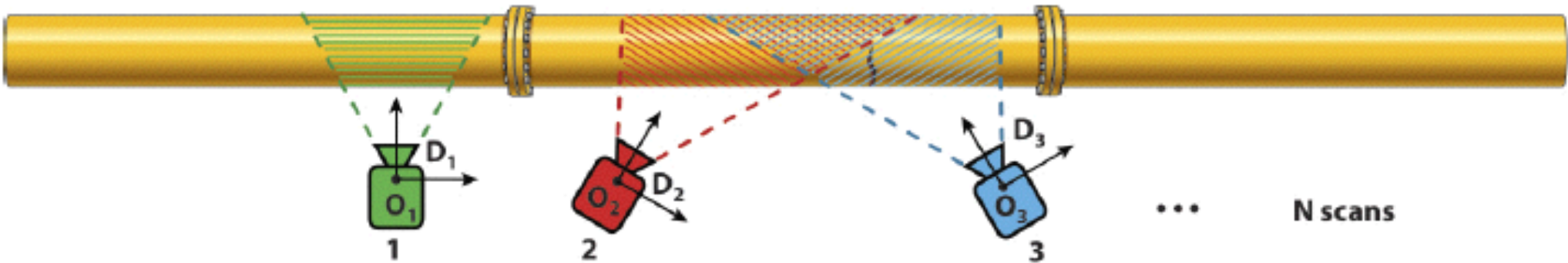
Cloud

Jython UDF

```
SELECT *  
FROM  
read_ply(files_s3)
```

Pipeline

```
RUN  
calc_rstreng(points)
```



Detecting Failures in Utility Gas Pipes

CREATE PIPELINE

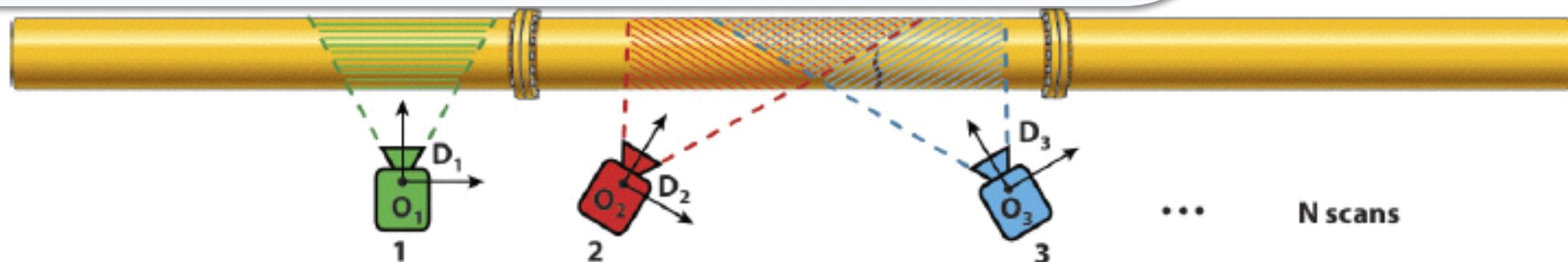
```
calc_rstreng("river_bottom_filepath", "filename",  
"rstreng_table", "root_location")  
BEGIN  
  
-- step 1: connect to the river bottom profile csv file  
DROP FOREIGN TABLE IF EXISTS "$rstreng_table_profile";  
CREATE FOREIGN TABLE "$rstreng_table_profile"  
LOCATION 'file://$river_bottom_filepath'  
OPTIONS(  
  'format' IS 'csv',  
  'has_header_row' IS 'true',  
  'delimiter' IS ',',  
  'append_line_num_column' IS 'row_num',  
  'refresh' IS 'never'  
);  
. . . .
```

Jython UDF

```
SELECT *  
FROM  
read_ply(files_s3)
```

Pipeline

```
RUN  
calc_rstreng(points)
```

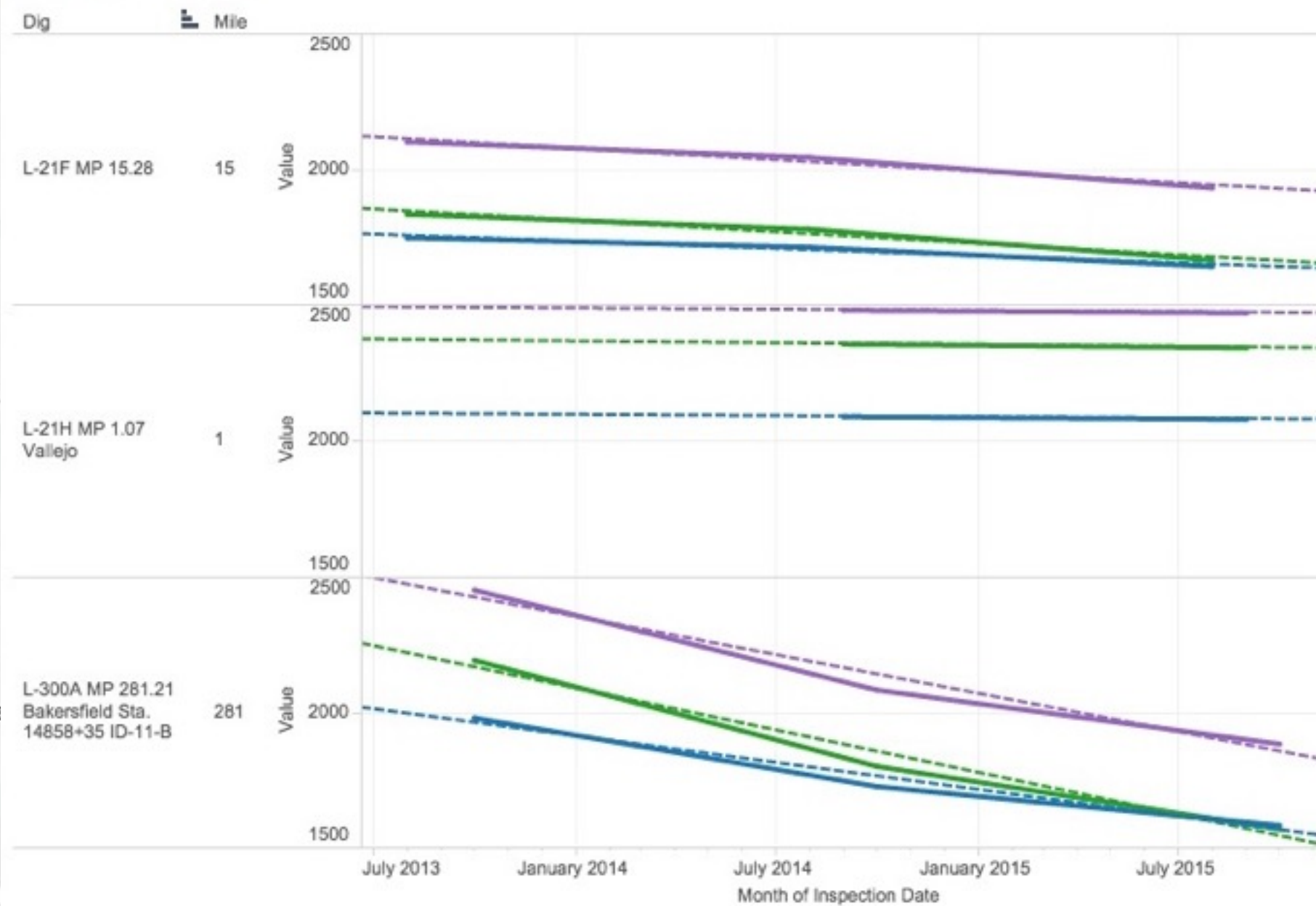


Detecting Failures in Utility Gas Pipes

Remember my changes ▾

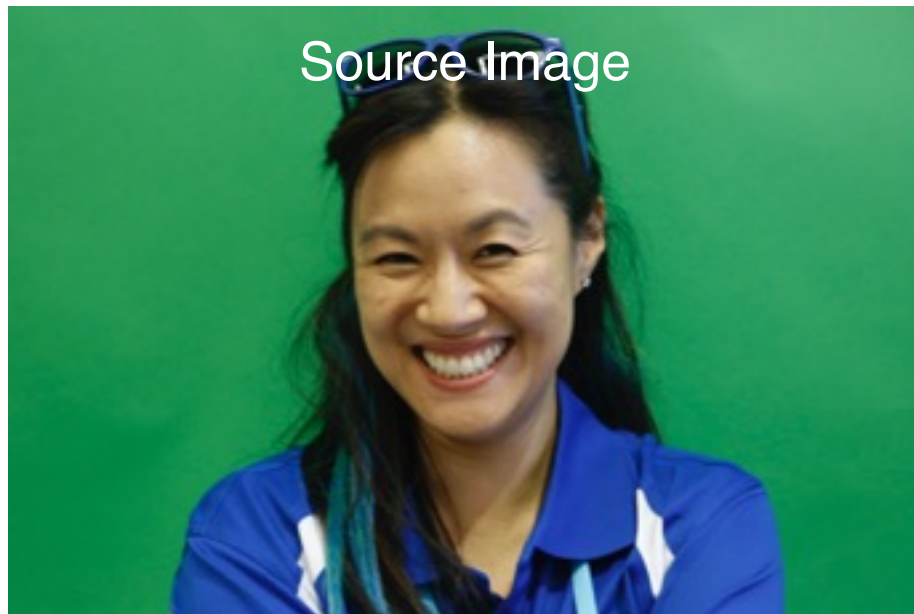


RStreng Time Trend

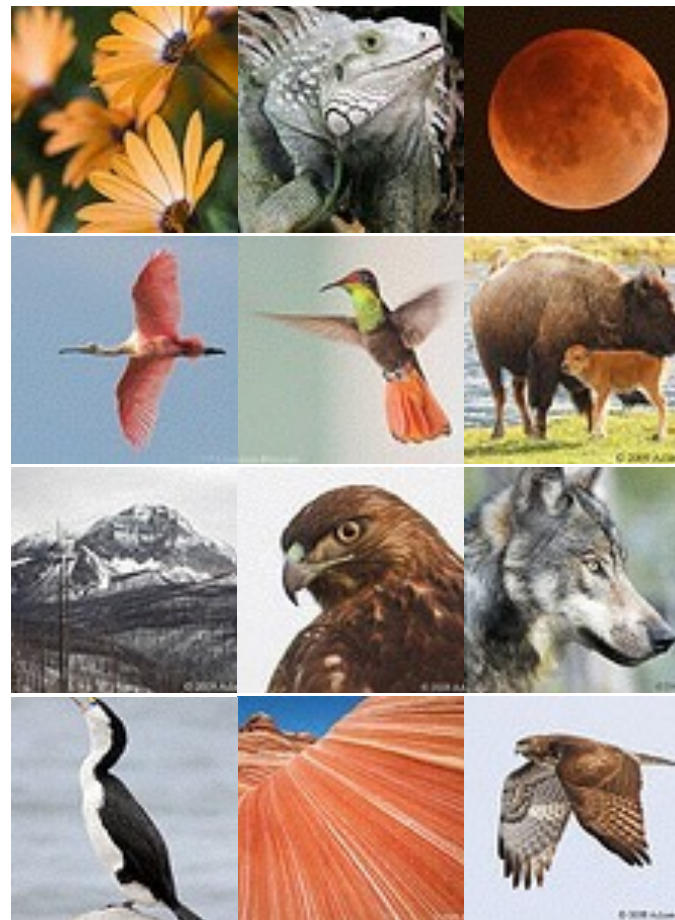
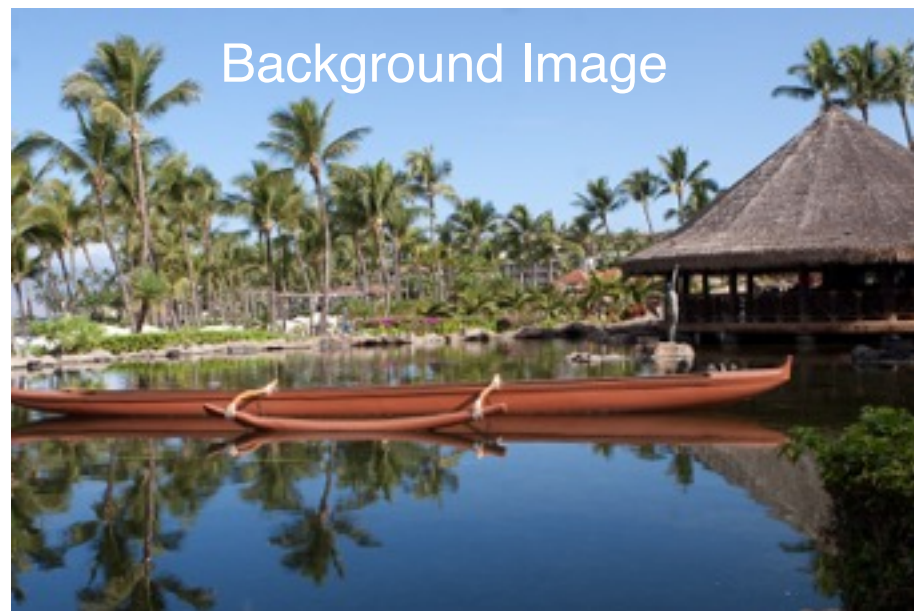


Declarative Media Processing

Source Image



Background Image



Gallery Images

SQL Queries

```
SELECT
  m.x*g.size + g.x AS x,
  m.y*g.size + g.y AS y,
  g.r, g.g, g.b,
  m.scale_r,
  m.scale_g,
  m.scale_b
FROM SourceMatchedPixels m
JOIN GalleryPixels g
  ON m.gallery = g.image
```

Declarative Media Processing



Declarative Media Processing



Declarative Media Processing



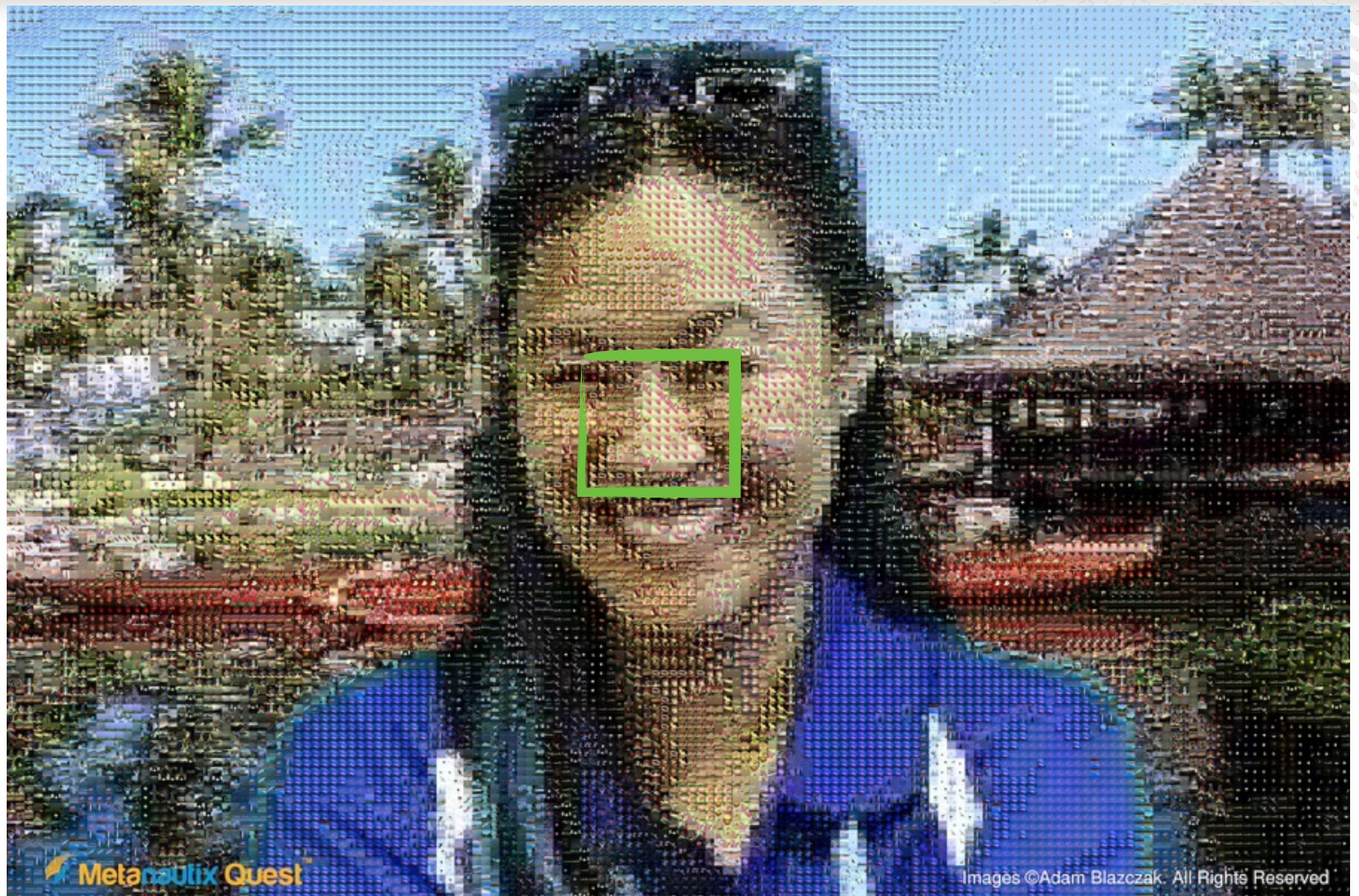
Declarative Media Processing



Metanautix Quest™

Images ©Adam Blazczak. All Rights Reserved

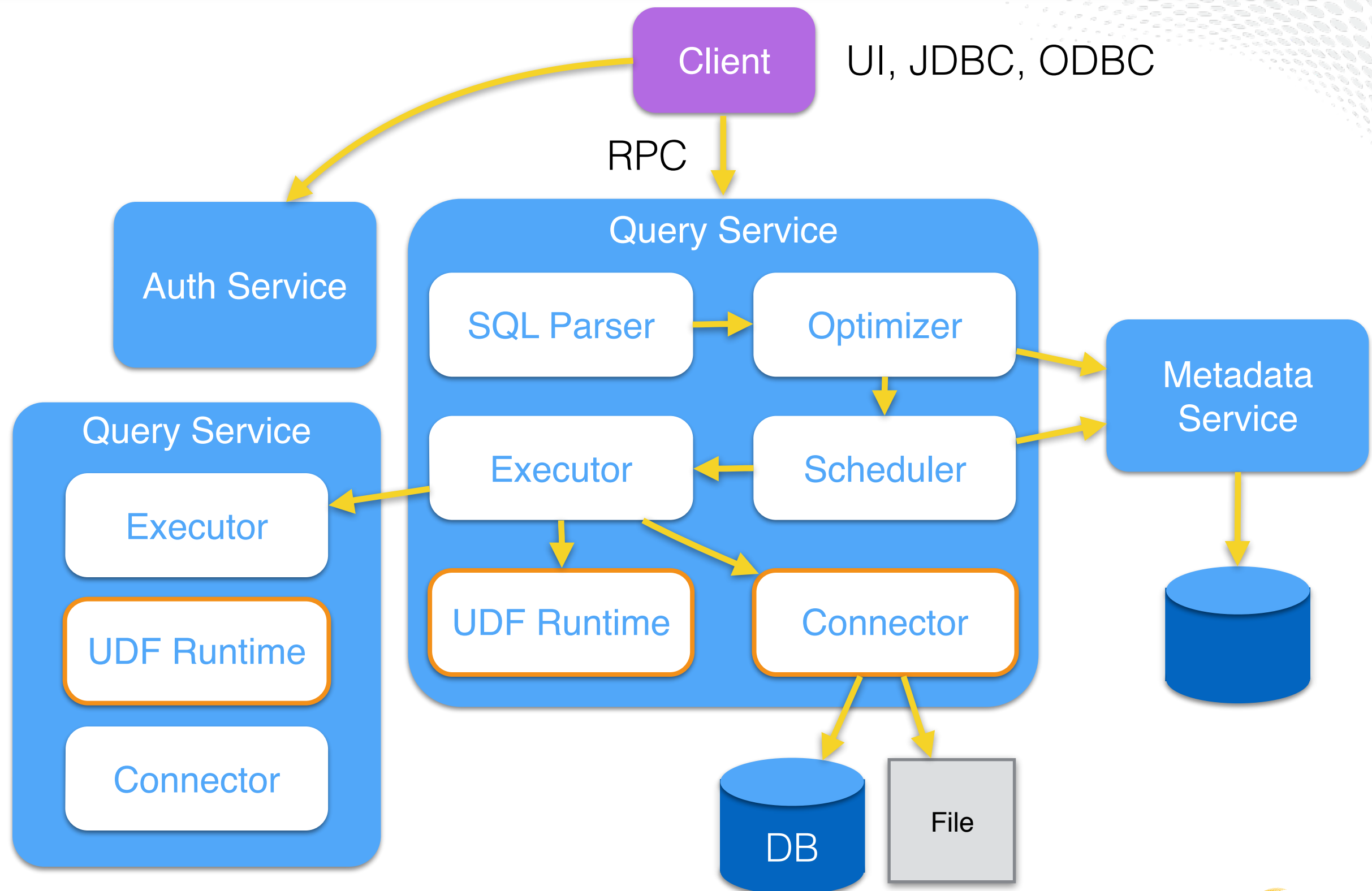
Declarative Media Processing



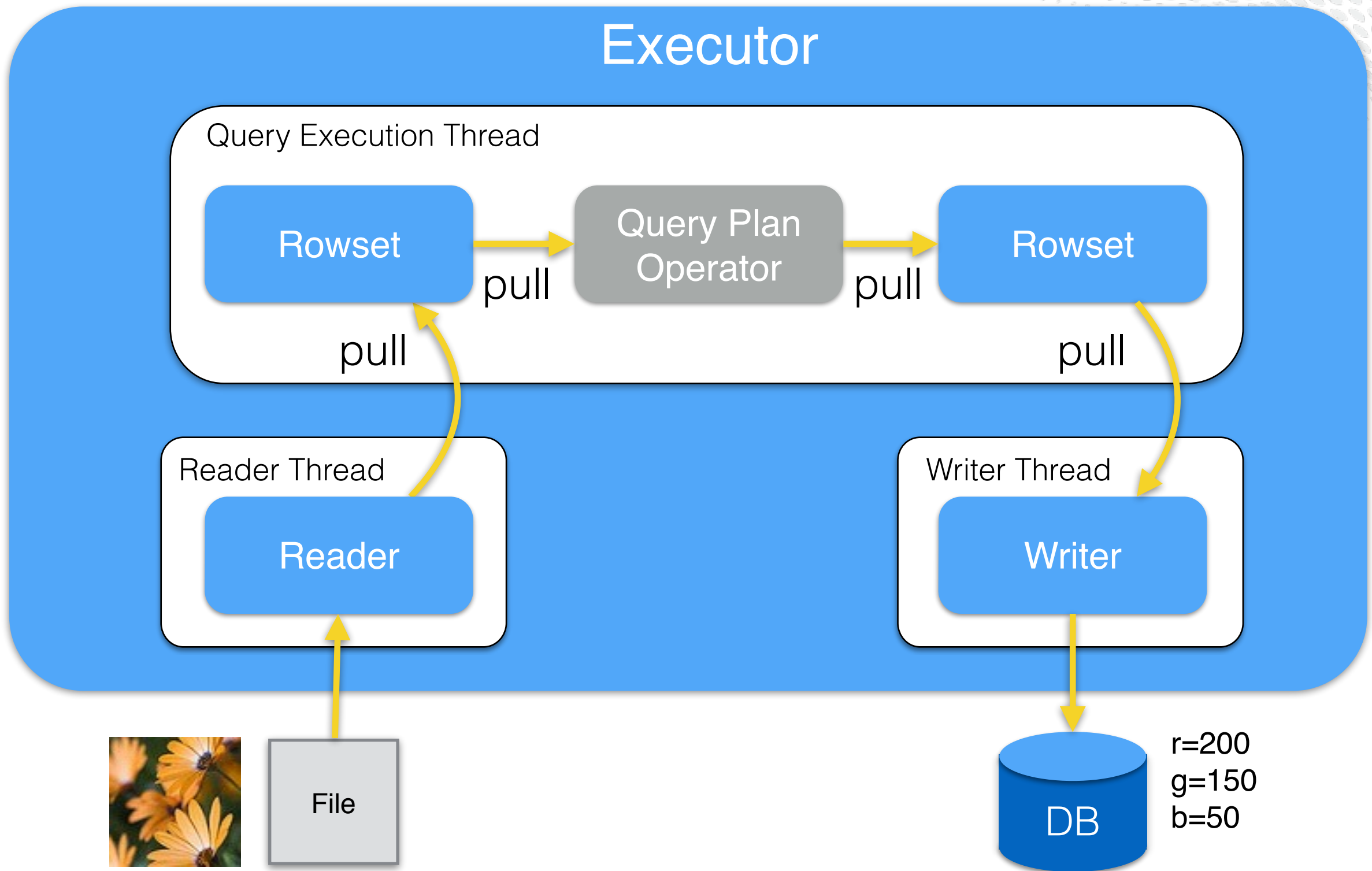
Declarative Media Processing



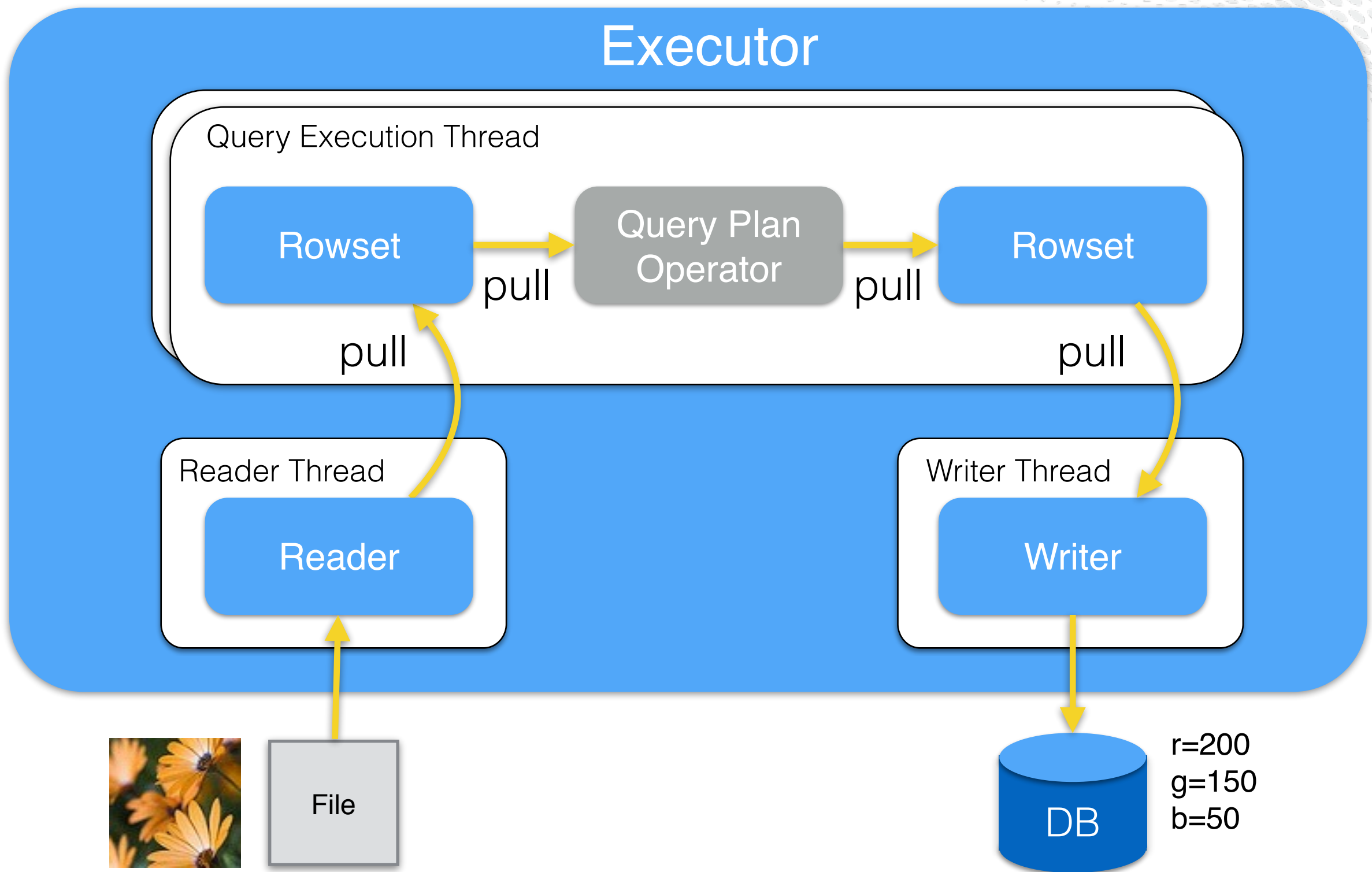
Metanautix Quest Architecture



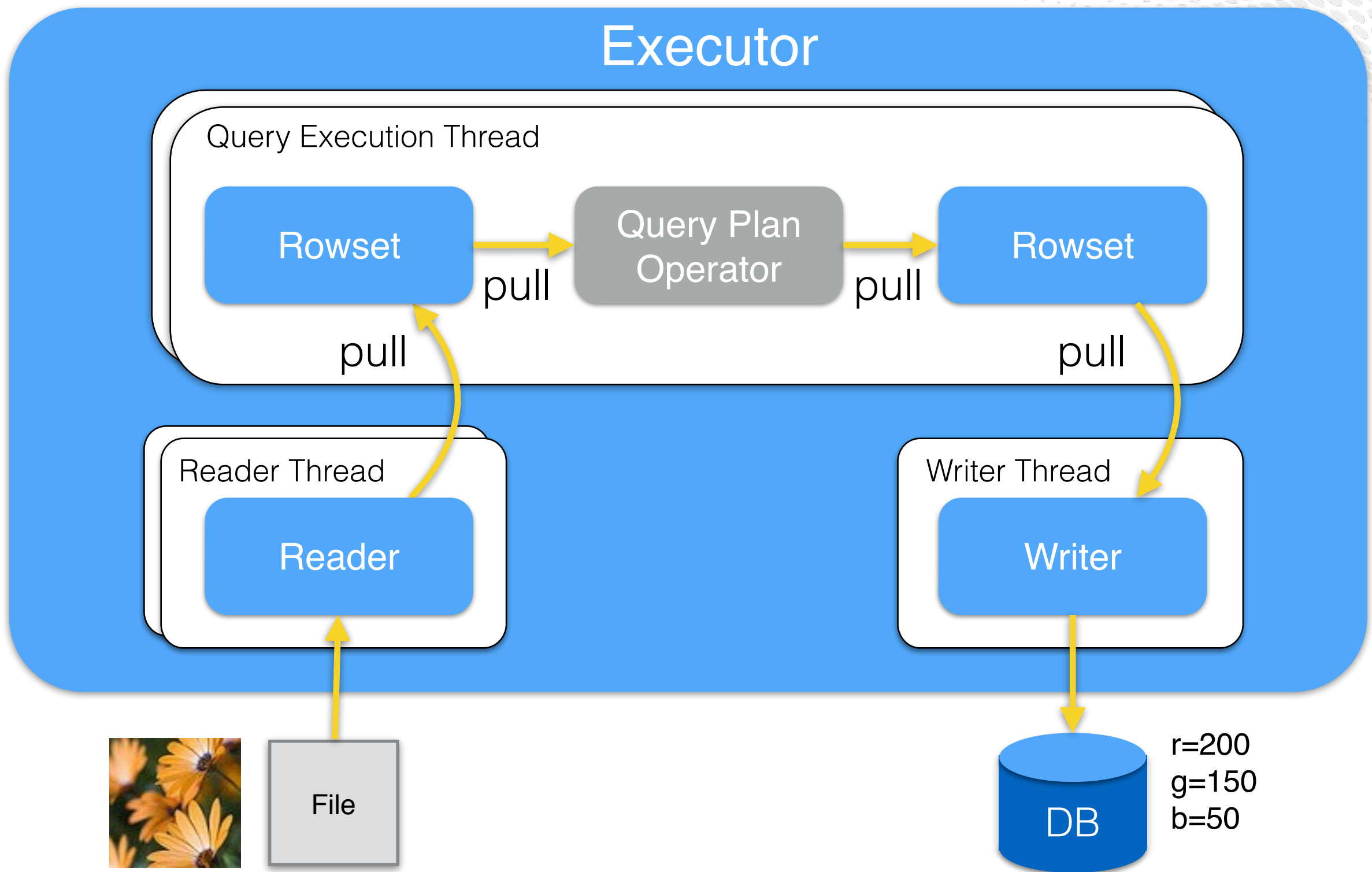
Connecting To External Stores



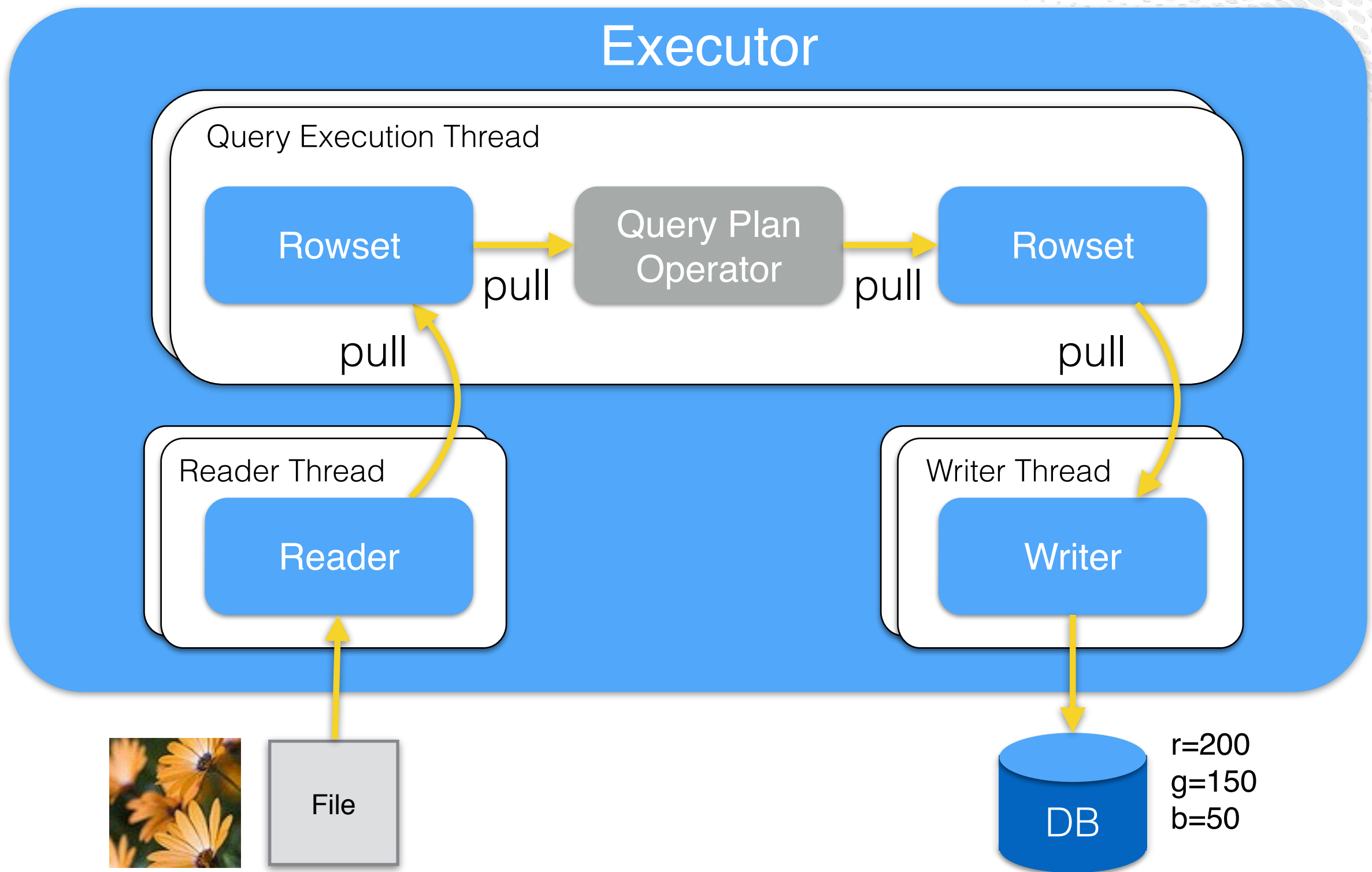
Connecting To External Stores



Connecting To External Stores



Connecting To External Stores

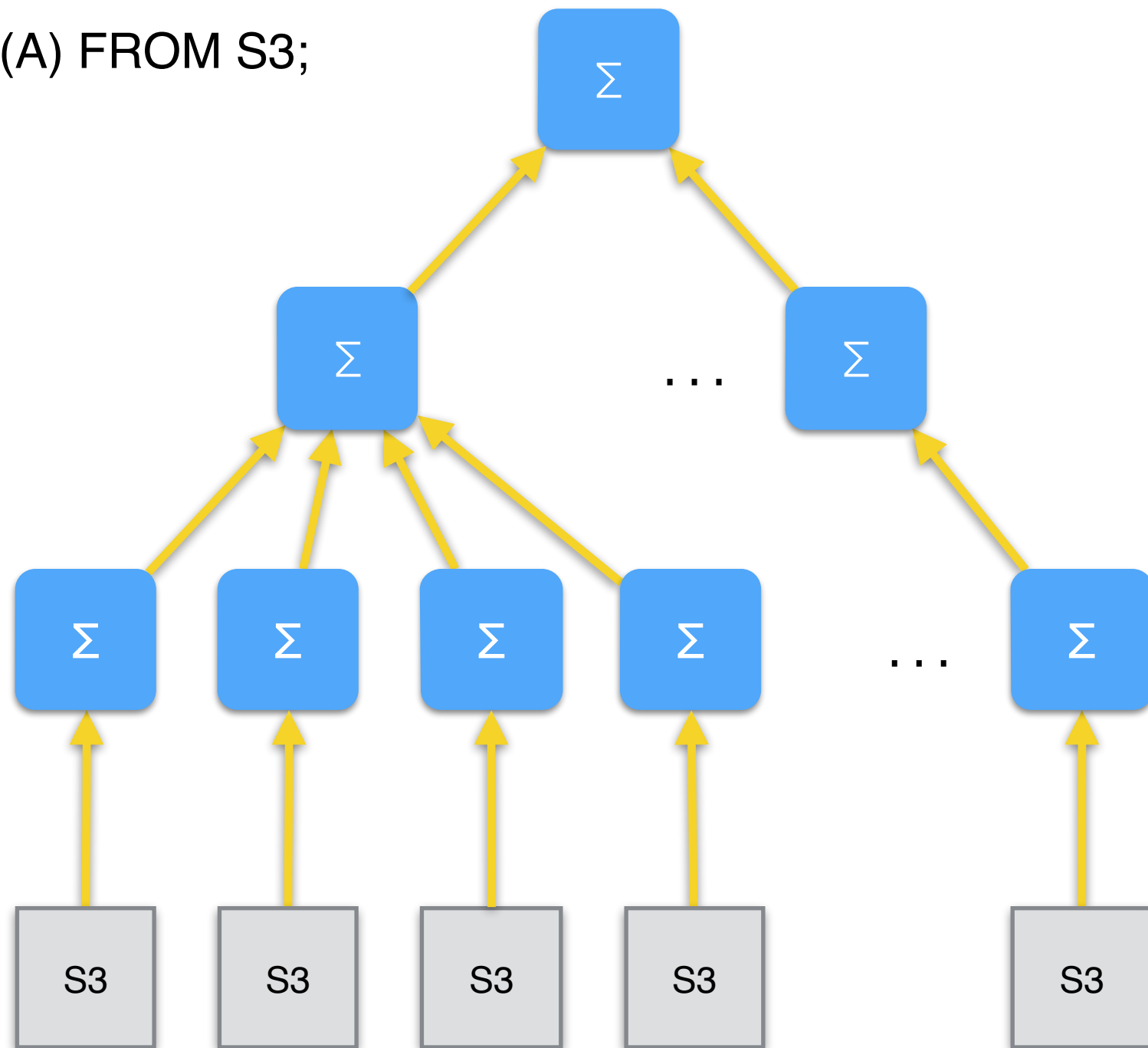


Caching

	resides in	used if	invalidated when
Results	LevelDB	same query plan	externally modified or table option
Query Plans	RAM	same query plan minus constants, pushdowns, limits, field names	evicted only
Contents	LevelDB	same table	externally modified or via user option
Shard Metadata	LevelDB	same table	externally modified or user option

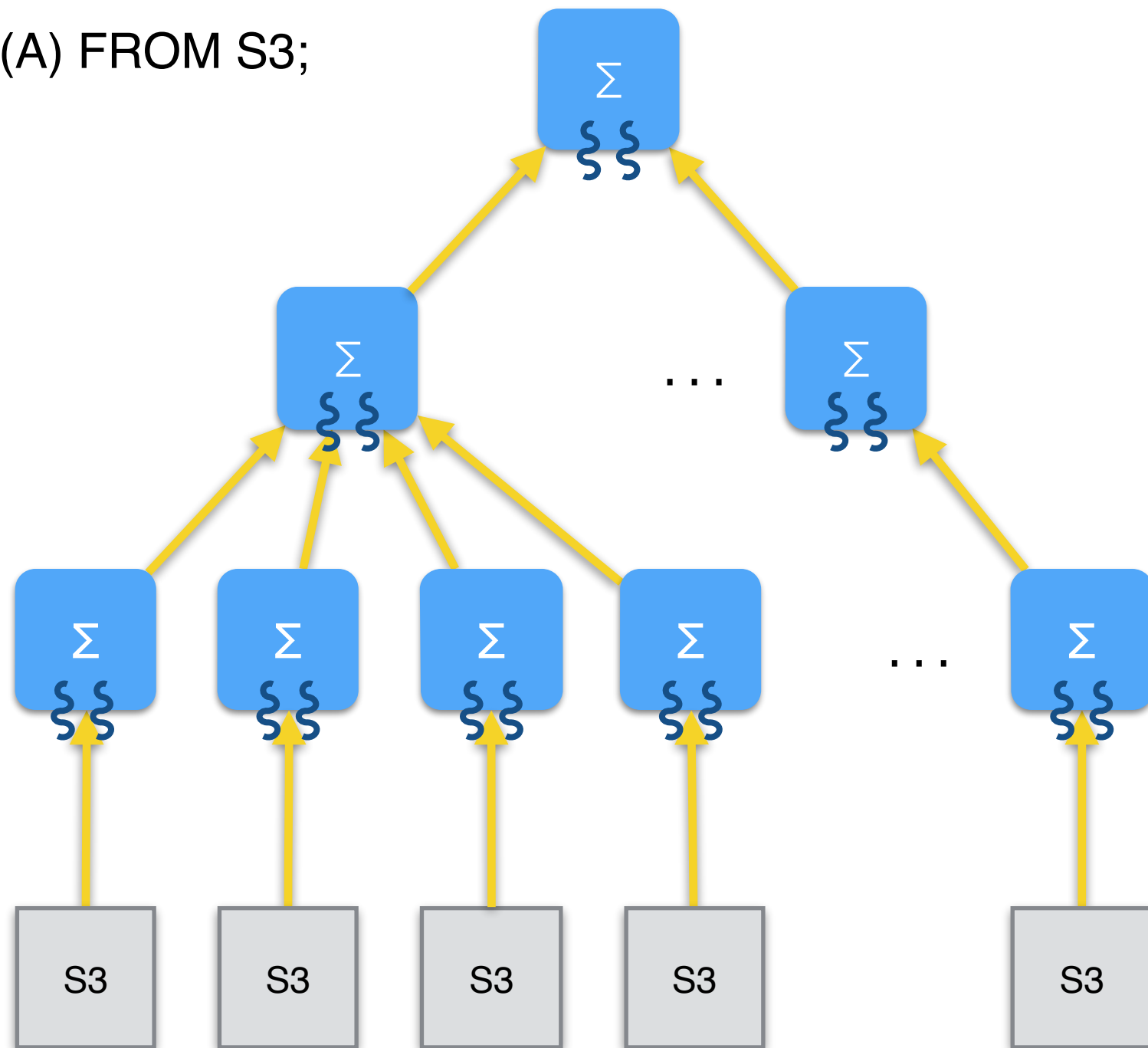
Distributed Execution

SELECT sum(A) FROM S3;



Distributed Execution

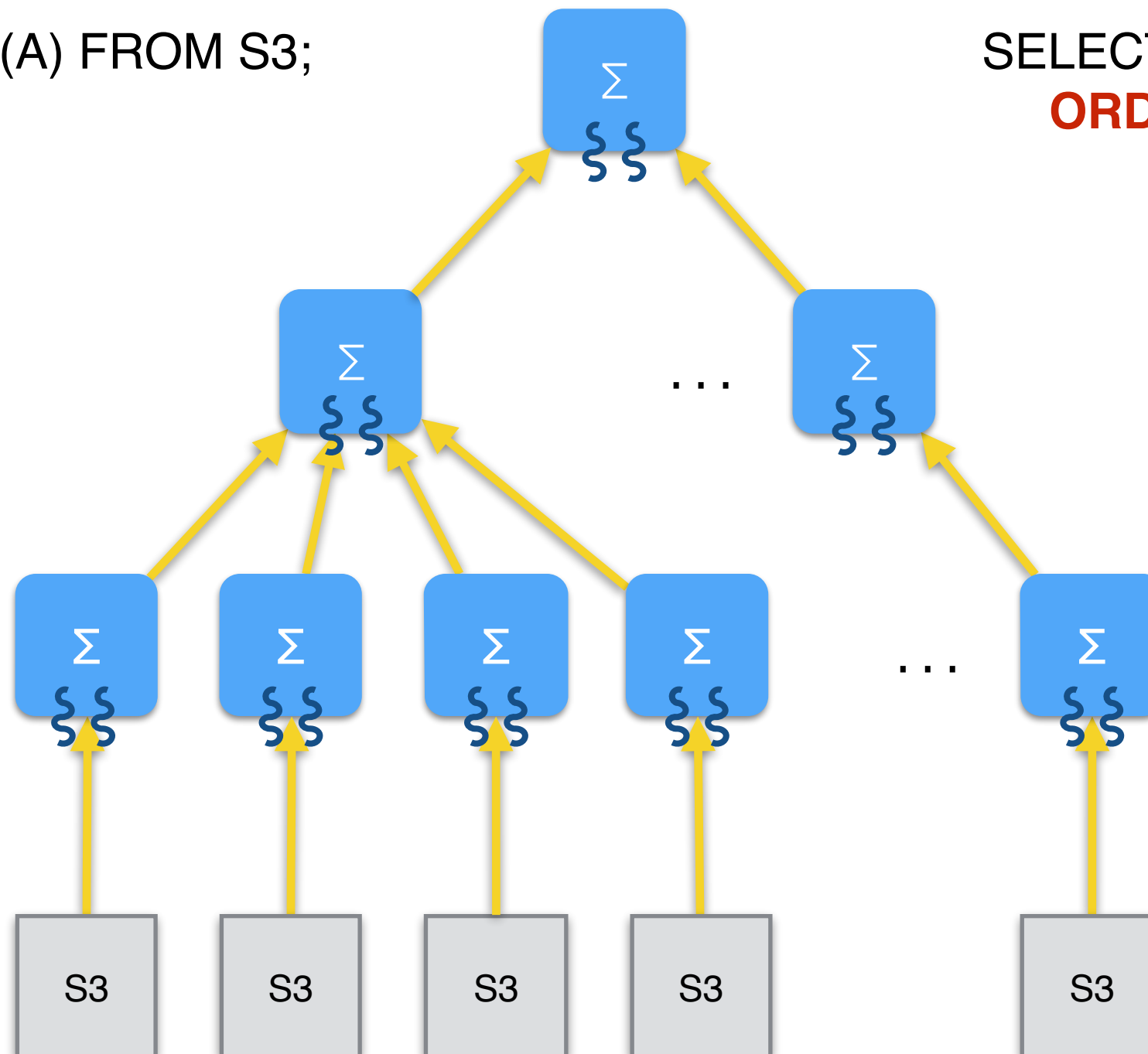
SELECT sum(A) FROM S3;



Distributed Execution

SELECT sum(A) FROM S3;

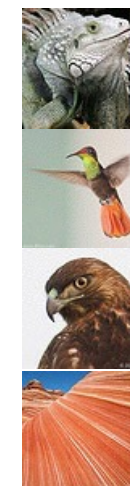
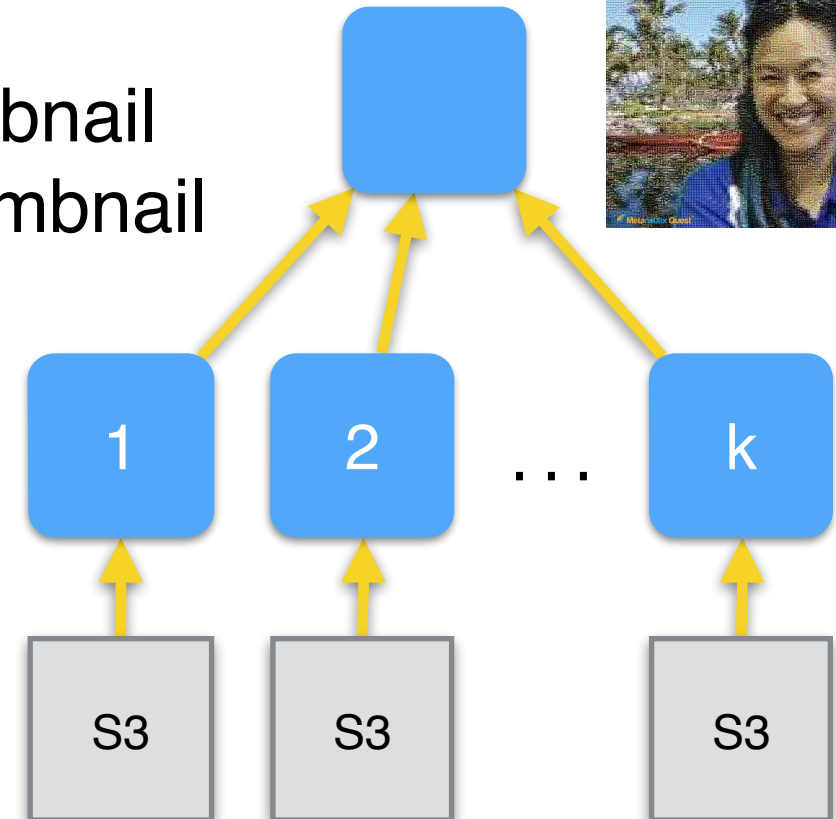
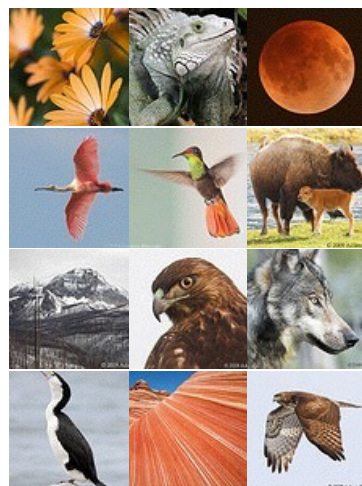
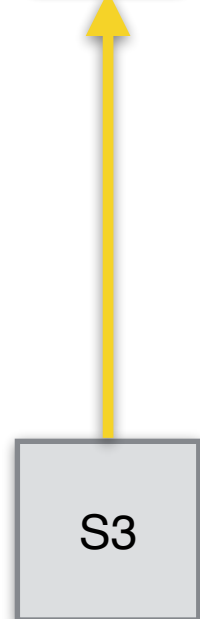
SELECT * FROM S3
ORDER BY A;



Optimizer And Machine Learning

Model:

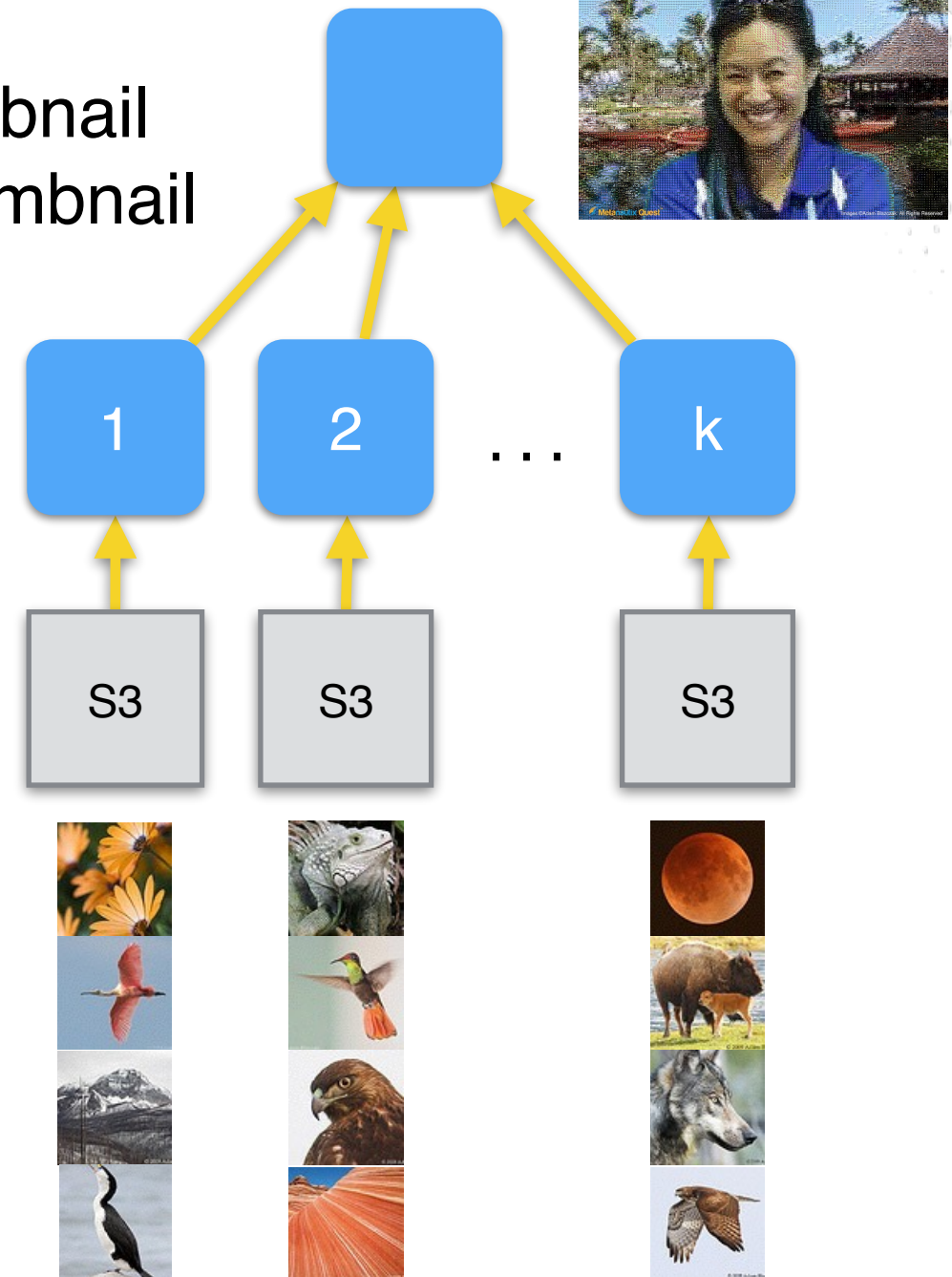
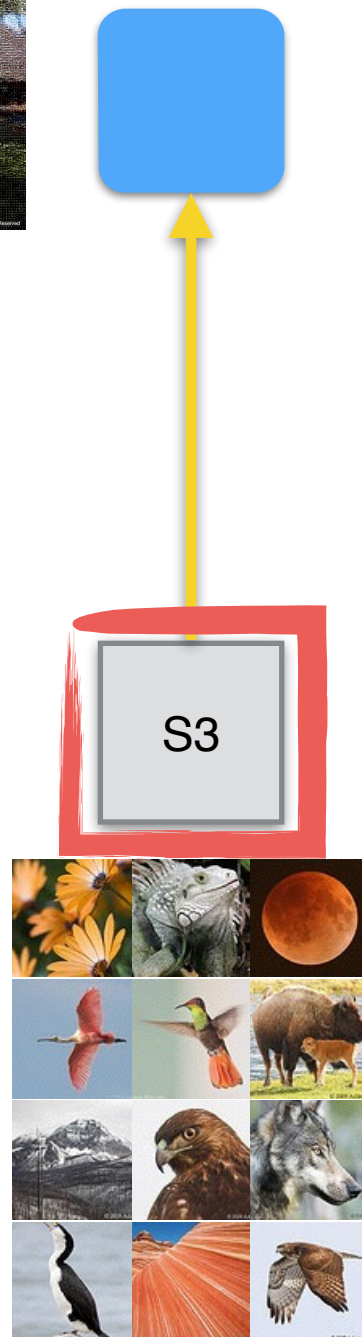
- w_1 : read picture files
- w_2 : compute RGB+thumbnail
- w_3 : aggregate RGB+thumbnail
- w_4 : stitch mosaic



Optimizer And Machine Learning

Model:

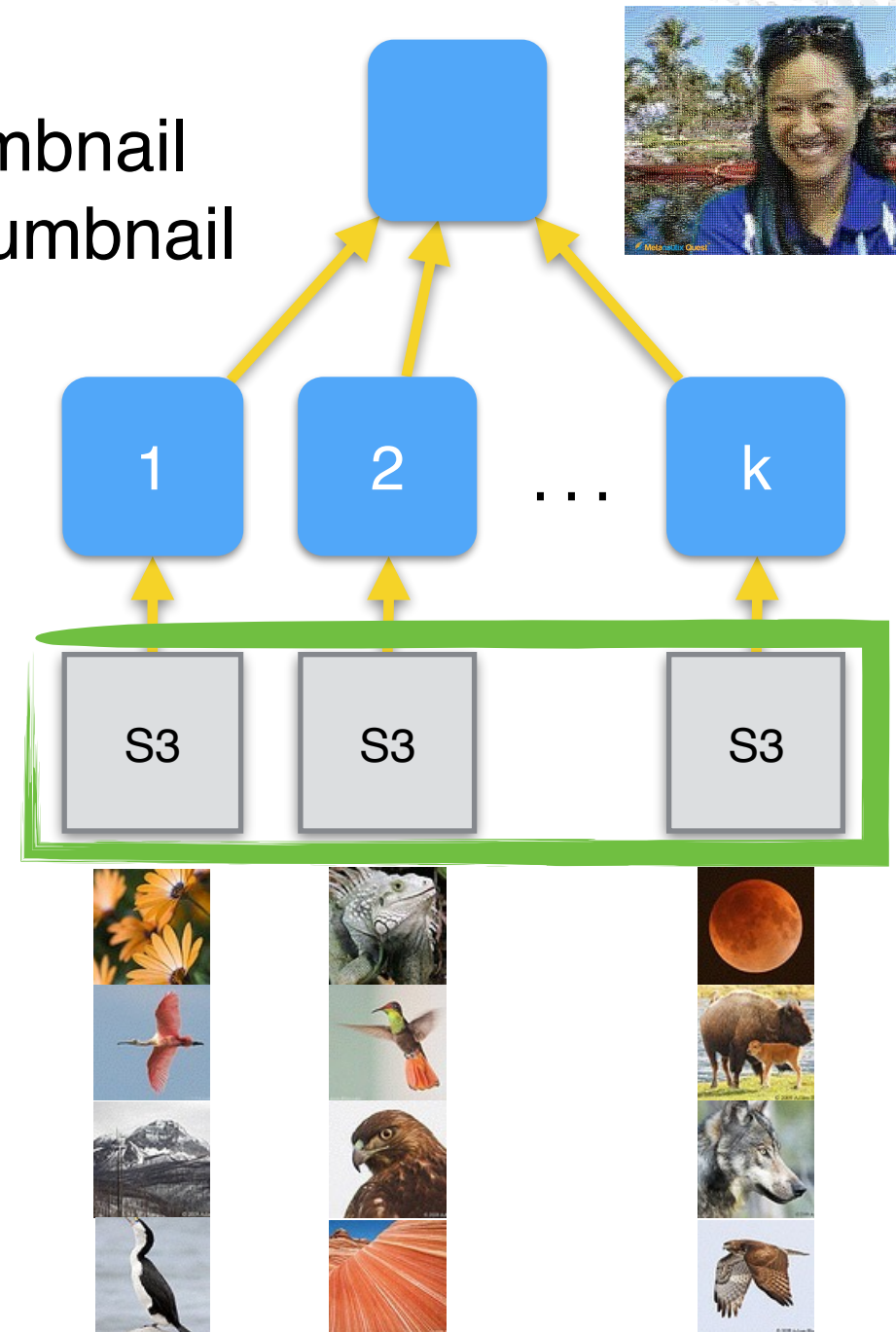
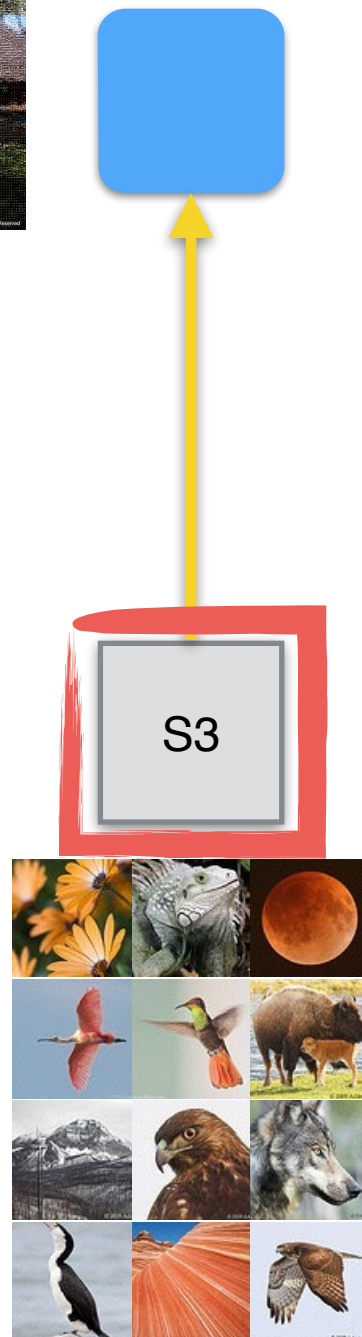
- w_1 : read picture files
- w_2 : compute RGB+thumbnail
- w_3 : aggregate RGB+thumbnail
- w_4 : stitch mosaic



Optimizer And Machine Learning

Model:

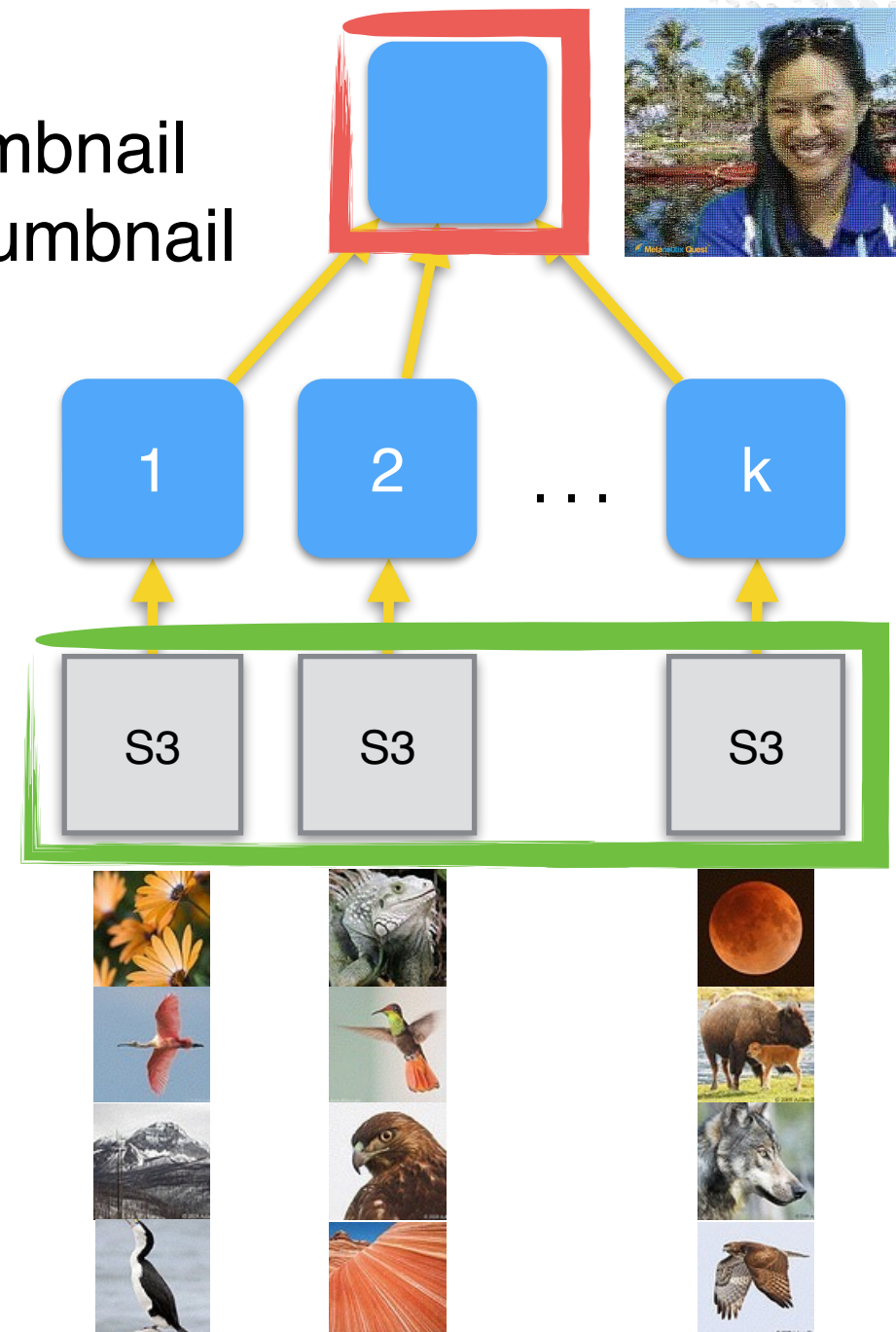
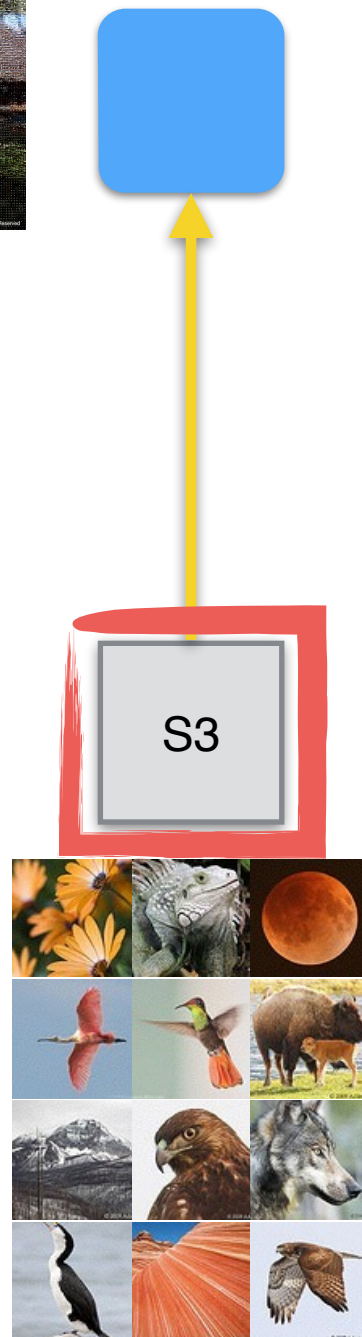
- w_1 : read picture files
- w_2 : compute RGB+thumbnail
- w_3 : aggregate RGB+thumbnail
- w_4 : stitch mosaic



Optimizer And Machine Learning

Model:

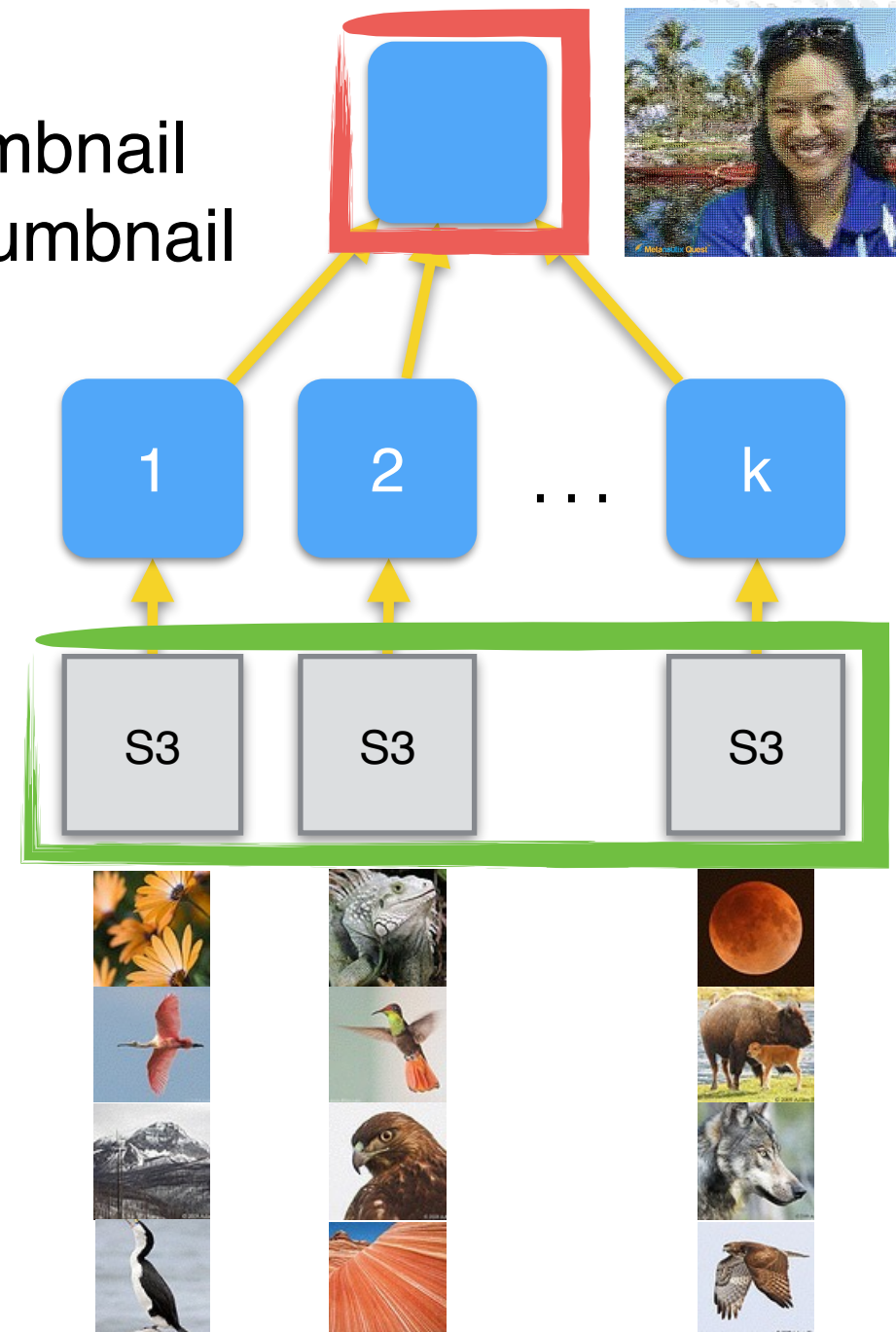
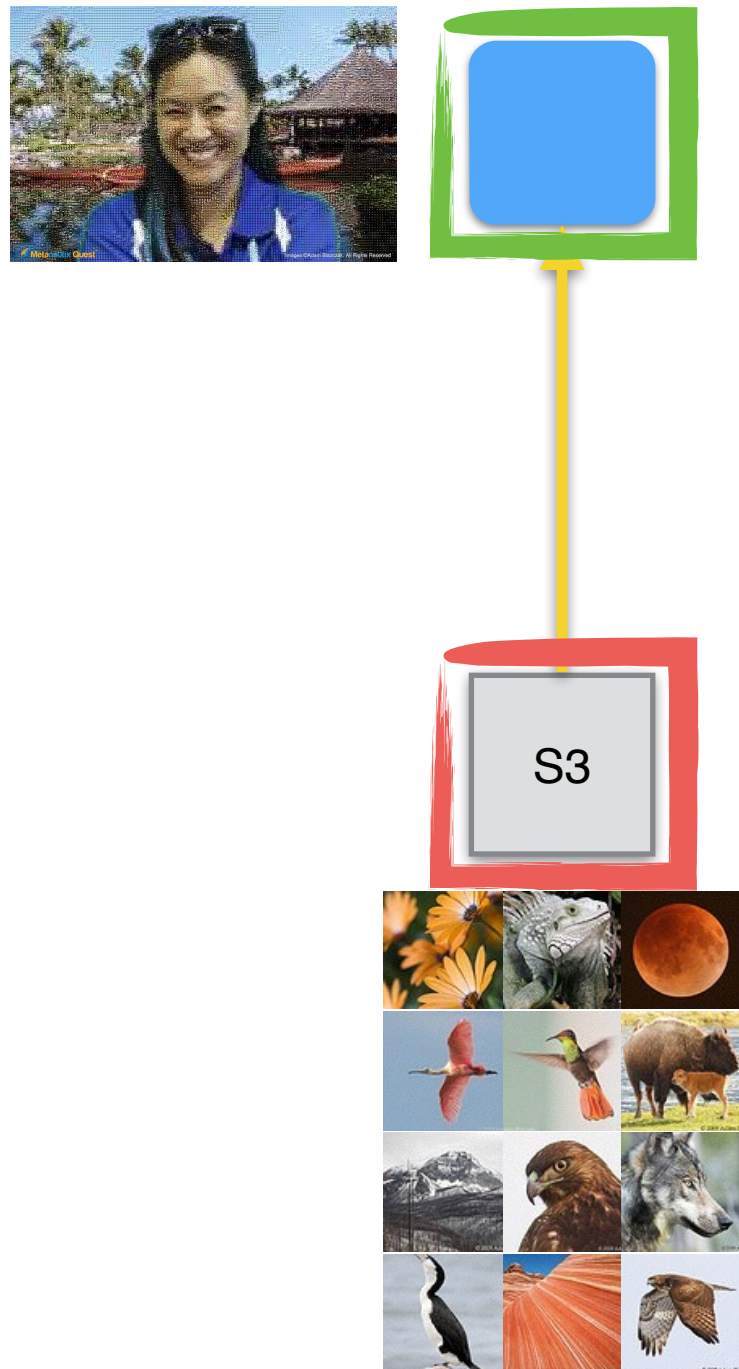
- w_1 : read picture files
- w_2 : compute RGB+thumbnail
- w_3 : aggregate RGB+thumbnail
- w_4 : stitch mosaic



Optimizer And Machine Learning

Model:

- w_1 : read picture files
- w_2 : compute RGB+thumbnail
- w_3 : aggregate RGB+thumbnail
- w_4 : stitch mosaic



Performance

- Performance of single node
 - We run *all* queries of popular analytical benchmark
 - *Without* touching the SQL queries
 - Comparable performance with IBM Big SQL report [11/2014]
 - 1/10th of load at 1/10th of CPU/MEM resources (single node)
- Performance of multi-node (WIP)
 - We run ~90% of the queries of the same benchmark
 - We see linear scale with some fractional overhead.

Future Work

- More work on benchmarks
 - Distributed computation
 - Validate by third party
- Make Connector SDK available to 3rd parties
- Continue machine learning work
 - Fully integrate Adaptive Scheduler
 - Also provide ML primitives as part of the product

Questions?

More on Metanautix Quest at:
<https://metanautix.com/product/>

Special Thanks To Our Engineers

*Patrick Nguyen, David Kryze, Sreeni Viswanadha, Eric Karlson,
George Baxter, Laurent Nguyen, Shizhen Wang, Adam Blazczak,
Carly Williams, Thanh Doan, Sunny Sun, Max Lerch*