

LESSON 7 : SCALABLE DATA IMPORT/EXPORT

LOGICBLOX TRAINING

SO FAR : DATA LOAD BY DELTA RULES

```
// creating wines
+core:wine:wine(x), +core:wine:wine_id(x:1000),
+core:wine:hasDescription(x,"San Martino Riserva 2007"),
+core:wine:ofYear(x,year),
+core:wine:cost(x,3.6f),
+core:wine:retail(x,6.99f),
+core:wine:hasOrigin(x,italy),
+core:wine:ofType(x,type)
  <- core:attributes:year_id[year] = 2007,
    core:attributes:country_id[italy] = "Italy",
    core:attributes:wineType_id[type] = "red".

+core:wine:wine(x), +core:wine:wine_id(x:1001),
+core:wine:hasDescription(x,"Santa Ana Chardonnay Viognier 2010"),
+core:wine:ofYear(x,year),
+core:wine:cost(x,3.6f),
+core:wine:retail(x,6.99f),
+core:wine:hasOrigin(x,country),
+core:wine:ofType(x,type)
  <- core:attributes:year_id[year] = 2010,
    core:attributes:country_id[country] = "Argentina",
    core:attributes:wineType_id[type] = "white".
```



SO FAR : DATA LOAD BY DELTA RULES

```
// creating wines
+core:wine:wine(x), +core:wine:wine_id(x:1000),
+core:wine:hasDescription(x,"San Martino Riserva 2007"),
+core:wine:ofYear(x,year),
+core:wine:cost(x,3.6f),
+core:wine:retail(x,6.99f),
+core:wine:hasOrigin(x,italy),
+core:wine:ofType(x,type)
  <- core:attributes:year_id[year] = 2007,
    core:attributes:country_id[italy] = "Italy",
    core:attributes:wineType_id[type] = "red".

+core:wine:wine(x), +core:wine:wine_id(x:1001),
+core:wine:hasDescription(x,"Santa Ana Chardonnay Viognier 2010"),
+core:wine:ofYear(x,year),
+core:wine:cost(x,3.6f),
+core:wine:retail(x,6.99f),
+core:wine:hasOrigin(x,country),
+core:wine:ofType(x,type)
  <- core:attributes:year_id[year] = 2010,
    core:attributes:country_id[country] = "Argentina",
    core:attributes:wineType_id[type] = "white".
```

Imagine loading 100,000 wines...



COMMON PRACTICE

- delimited File

```
ID|DESC|YEAR|COST|RETAIL|ORIGIN|TYPE
1000|San Martino Riserva 2007|2007|3.6|6.99|Italy|red
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99||
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99|Chile|
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99||red
...
```

- ought to be simple



IN PRACTICE

- typical decisions an integration consultant needs to make:
 - Where is an {input, output} file located?
 - What if the file is not there?
 - Is the {input, output} file compressed?
 - Do I check that the input file is correct and complete?
 - Does the input file have a header?
 - Do I also need an {import, export} procedure?
 - Will I have to handle a full file and retractions?
 - How do I handle optional columns in logic?
 - Do I need to create entity elements?
 - Do I generate assertions for already existing data?
 - Do I use a vector format for compression?
 - How do I script the execution of the batch procedure?
 - Do I add options for logging database details in case of issues?



DELIMITED FILE SERVICES

- standardization of irrelevant file format variations
 - e.g. delimited files have headers
- high-level specification of file format
 - suitable for communication with customer
- specification for binding file to predicates
 - get import, export, replace all from one specification
 - generator handles details of file definition (optional columns)
- services and HTTP
 - HTTP handles details of compression, local file location irrelevant



SIMPLICITY

- HTTP methods are a good match for integration needs
 - GET- retrieve current data
 - PUT - replace current data with given data
 - POST - modify current data
- demo:
 - <http://localhost:8080/dw/delim-wines>

```
$ curl -X POST
  -H "Content-Type: text/csv"
  --data-binary data/extrawines.dlm
  http://localhost:8080/dw/delim-wines
```

```
$ wget -q -O -
  http://localhost:8080/dw/delim-wines
```



USING DELIMITED FILE SERVICE

- file definition
 - delimiter, column headers, expected format
- file -> predicate binding specification
 - what column goes into what predicate
 - required vs. optional columns
 - import/export functions (e.g. string:trim)
- service configuration
 - URI, file binding



FILE DEFINITION

```
ID|DESC|YEAR|COST|RETAIL
```

```
1000|San Martino Riserva 2007|2007|3.6|6.99
```

```
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
```

```
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
```

```
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
```

```
...
```



EXAMPLE : /DW/DELIM-WINES

```
ID|DESC|YEAR|COST|RETAIL
1000|San Martino Riserva 2007|2007|3.6|6.99
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
...
```

logiql/services/tdx/wines.logic

```
// defining file import of wine information
file_definition_by_name["wines"] = fd,
file_definition(fd) {
  file_delimiter[] = "|",
  column_headers[] = "ID,DESC,YEAR,COST,RETAIL",
  column_formats[] = "int,string,int,float,float"
}.
```



FILE → PREDICATE BINDING

```
ID|DESC|YEAR|COST|RETAIL
1000|San Martino Riserva 2007|2007|3.6|6.99
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
...
```

core:wine:hasDescription[wine] = description

core:wine:ofYear [wine] = year

core:wine:cost [wine] = cost

core:wine:retail [wine] = retail



FILE → PREDICATE BINDING

```
ID|DESC|YEAR|COST|RETAIL
1000|San Martino Riserva 2007|2007|3.6|6.99
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
...
```

ID

DESC

YEAR

COST

RETAIL

core:wine:hasDescription[wine] = description

core:wine:ofYear [wine] = year

core:wine:cost [wine] = cost

core:wine:retail [wine] = retail



FILE → PREDICATE BINDING

```
ID|DESC|YEAR|COST|RETAIL
1000|San Martino Riserva 2007|2007|3.6|6.99
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
...
```

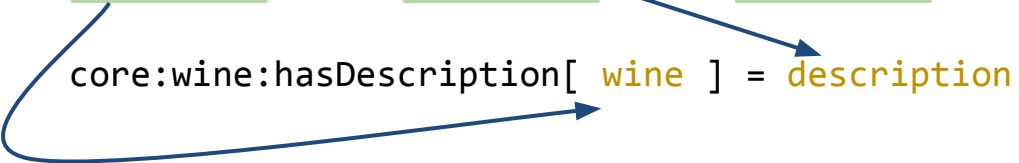
ID

DESC

YEAR

COST

RETAIL



core:wine:ofYear [wine] = year

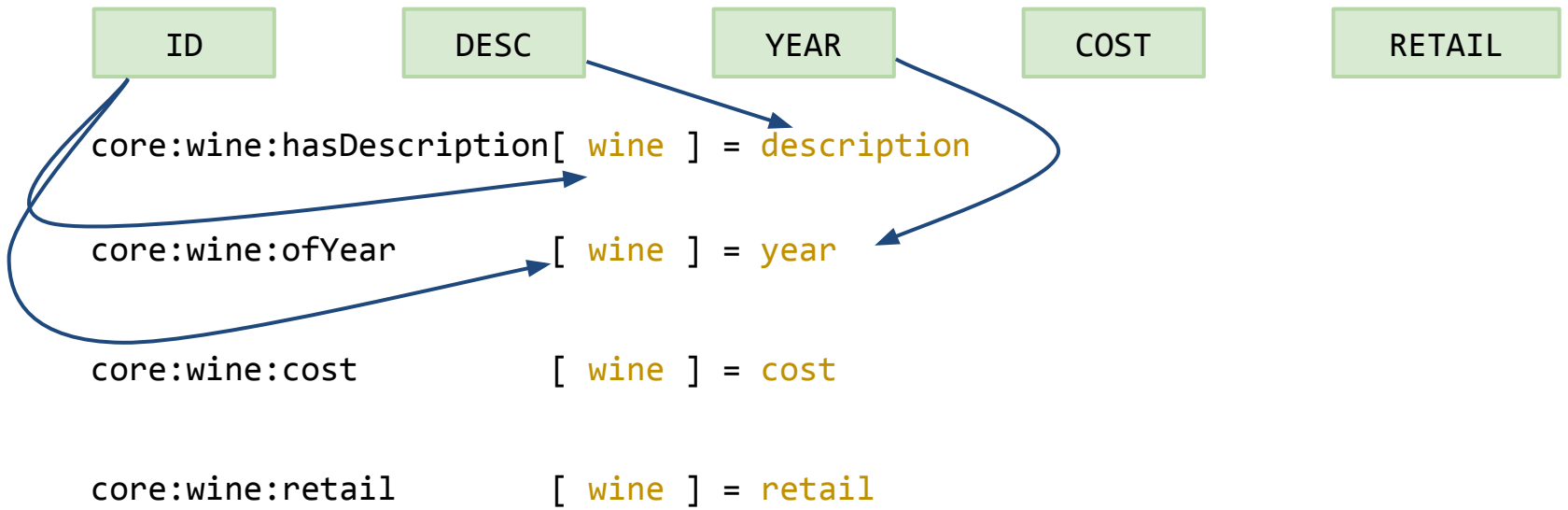
core:wine:cost [wine] = cost

core:wine:retail [wine] = retail



FILE → PREDICATE BINDING

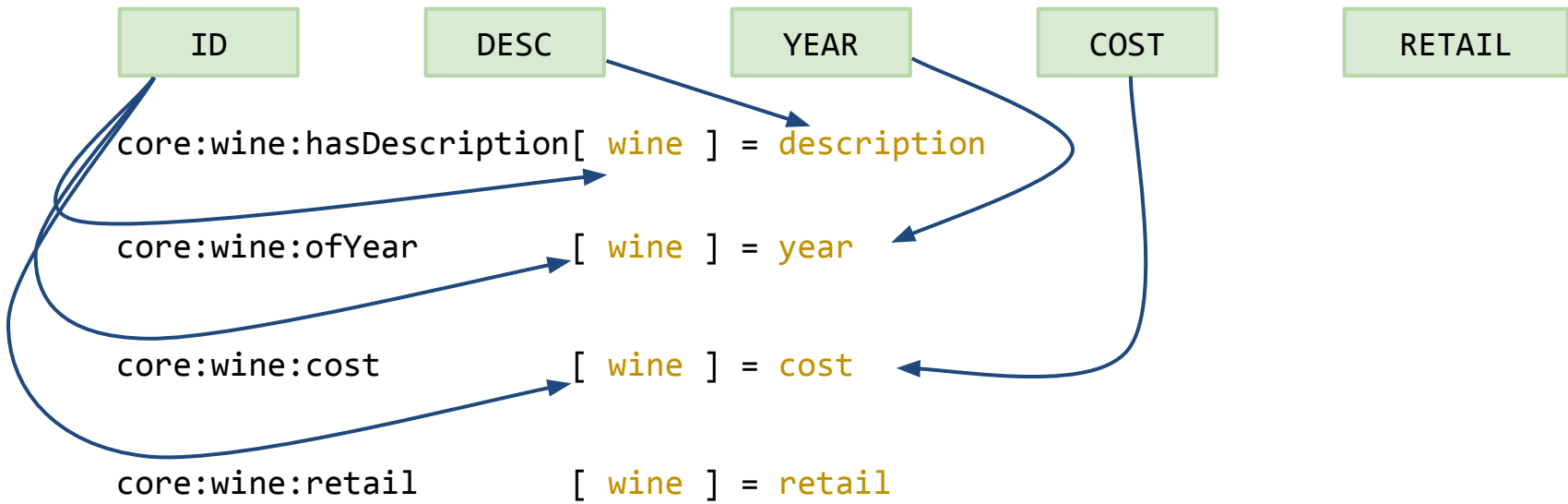
```
ID|DESC|YEAR|COST|RETAIL
1000|San Martino Riserva 2007|2007|3.6|6.99
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
...
```





FILE → PREDICATE BINDING

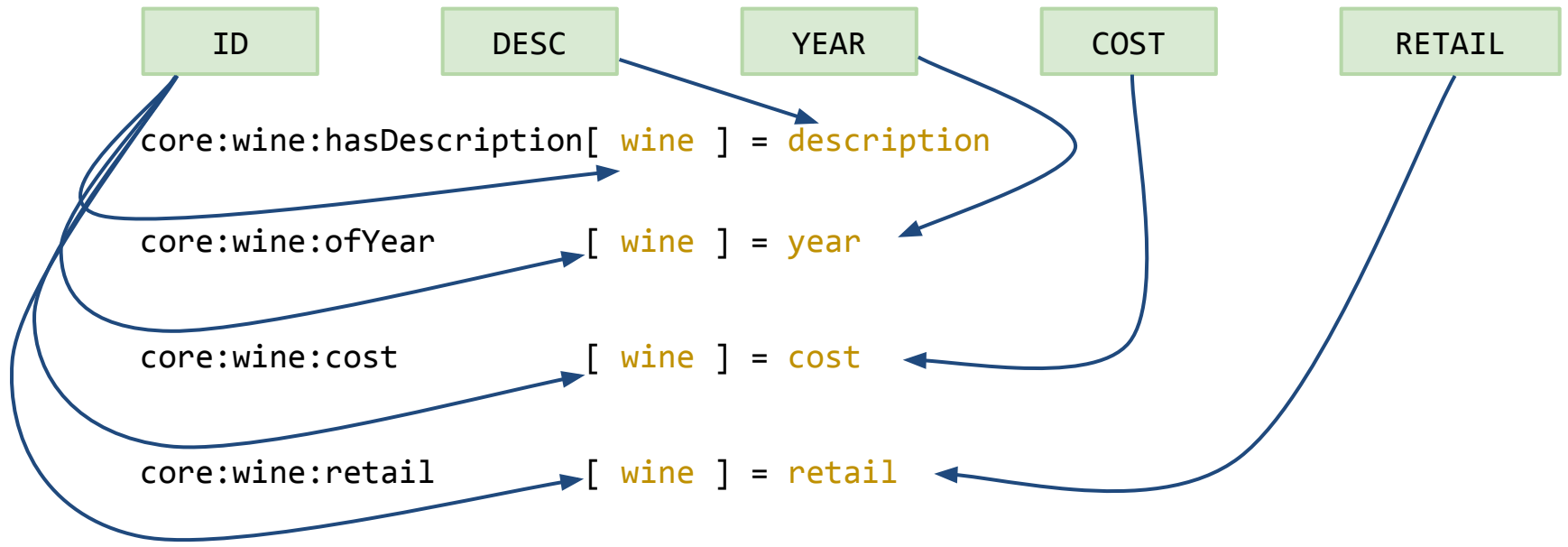
```
ID|DESC|YEAR|COST|RETAIL
1000|San Martino Riserva 2007|2007|3.6|6.99
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
...
```





FILE → PREDICATE BINDING

```
ID|DESC|YEAR|COST|RETAIL
1000|San Martino Riserva 2007|2007|3.6|6.99
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99
1003|Santa Rita Medalla Real Leyda Valley Pinot Noir 2008|2008|4.5|9.99
...
```




```
// defining file import of wine information
file_definition_by_name["wines"] = fd,
file_definition(fd) {
  file_delimiter[] = "|",
  column_headers[] = "ID,DESC,YEAR,COST,RETAIL",
  column_formats[] = "integer,string,integer,float,float"
}.

file_binding_by_name["wine_bindings"] = fb,
file_binding(fb) {
  file_binding_definition_name[] = "wines",
  file_binding_entity_creation[] = "accumulate",

  predicate_binding_by_name["core:wine:hasDescription"] =
  predicate_binding(_) {
    predicate_binding_columns[] = "ID,DESC"
  },
  predicate_binding_by_name["core:wine:ofYear"] =
  predicate_binding(_) {
    predicate_binding_columns[] = "ID,YEAR"
  },
  predicate_binding_by_name["core:wine:cost"] =
  predicate_binding(_) {
    predicate_binding_columns[] = "ID,COST"
  },
  predicate_binding_by_name["core:wine:retail"] =
  predicate_binding(_) {
    predicate_binding_columns[] = "ID,RETAIL"
  }
}.
```



SERVICE CONFIGURATION


logiql/services/tdx/wines.logic

```
delim_service(x) {  
  service_by_prefix("/dw/delim-wines"),  
  delim_file_binding("wine_bindings")  
}
```



OPTIONAL COLUMNS

```
ID|DESC|YEAR|COST|RETAIL|ORIGIN|TYPE
1000|San Martino Riserva 2007|2007|3.6|6.99|Italy|red
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99||
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99|Chile|
...
```

- File definition 
- File -> predicate binding specification
- Service configuration
 - URI, file binding



OPTIONAL COLUMNS

```
ID|DESC|YEAR|COST|RETAIL|ORIGIN|TYPE
1000|San Martino Riserva 2007|2007|3.6|6.99|Italy|red
1001|Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99||
1002|Santa Ines Limari Chardonnay 2010|2010|4.2|7.99|Chile|
...
```

logiq/services/tdx/wines_opt.logic

```
file_definition_by_name["wines_opt"] = fd,
file_definition(fd) {
  file_delimiter[] = "|",
  column_headers[] = "ID,DESC,YEAR,COST,RETAIL,ORIGIN,TYPE",
  column_formats[] =
    "int,string,int,float,float,string,string",

  file_columns_optional[] = "ORIGIN,TYPE"
}
```



TESTING DELIMITED FILE SERVICES

....

```
class TestDelimService(lb.web.testcase.PrototypeWorkspaceTestCase):
```

```
    prototype = "discountwines_test"
```

```
    def setUp(self):
```

```
        super(TestDelimService, self).setUp()
```

```
        self.maxDiff = None
```

```
        self.client = get_client("wines")
```

```
    def post_wines(self,client):
```

```
        client.post("""
```

```
            ID|DESC|YEAR|COST|RETAIL
```

```
            1000|Posted San Martino Riserva 2007|2007|3.6|6.99
```

```
            1001|Posted Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
```

```
        """)
```

```
        self.assertDelimEqual(client.get(), """
```

```
            ID|DESC|YEAR|COST|RETAIL
```

```
            1000|Posted San Martino Riserva 2007|2007|3.6|6.99
```

```
            1001|Posted Santa Ana Chardonnay Viognier 2010|2010|3.6|6.99
```

```
        """)
```

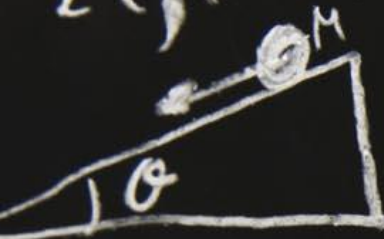
```
    def test_post_wines(self):
```

```
        client = get_client("wines")
```

```
        self.post_wines(client)
```

....

$$= \frac{1}{2} \rho (\omega r)^2 dV = \frac{1}{2} \omega^2 \rho (\alpha^2 + y^2) dV \quad r(\varphi)$$



$$U = Mgh = Mg \rho \sin \alpha \frac{d\omega}{d\varphi} = -\frac{1}{r^2} \frac{dr}{d\varphi}$$

$$K = \frac{1}{2} M v^2 + \frac{1}{2} I \omega^2$$

$$= \frac{dr^2}{d\varphi^2} \left(\frac{J}{\mu r^2} \right) - \frac{2}{r^3} \frac{J}{\mu} \left(\frac{dr}{d\varphi} \right) \frac{J}{\mu r^2} \frac{d\omega}{d\varphi} = \dots$$

$$\frac{d^2 \omega}{d\varphi^2} = \frac{1}{r^2} \frac{d^2 r}{d\varphi^2} + \frac{2}{r^3} \left(\frac{dr}{d\varphi} \right)^2$$



LESSON 7 LAB EXERCISES

$$\frac{d^2 v}{dt^2} = \frac{1}{r^2} \left(\frac{J}{\mu} \right) \frac{d^2 \omega}{d\varphi^2} \Rightarrow \frac{d^2 \omega}{d\varphi^2} + \omega = \frac{\mu G M \mu_1}{J^2}$$

$$\frac{dJ}{dt} + \omega \mu \vec{S} = \vec{N} \quad \alpha = \frac{I_2 - I_1}{I_1} \omega_1$$

$$\vec{F} = \frac{c}{r^2} \vec{r} \quad F = -\frac{\partial U}{\partial r} = \frac{c}{r^2}$$

$$\langle k \rangle = \frac{1}{t_0} \int_0^{t_0} k dt = \frac{1}{t_0} M \omega_0^2 \int_0^{t_0} \cos^2(\omega_0 t) dt$$

$$\int_0^{2\pi} \cos^2 \omega_0 t dt = \frac{1}{2\pi} \int_0^{2\pi} \cos^2 y dy = \frac{1}{2} \int_0^{2\pi} (1 + \cos 2y) dy = \frac{1}{2} (2\pi + 0) = \pi$$

$$U = \frac{1}{2} c x^2 = \frac{1}{2} c A^2 \sin^2 \omega_0 t$$



THANK YOU.

