

MAPR Academy

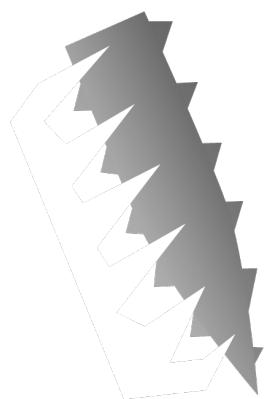
Data Analysis with Drill

Lesson 1: SQL Queries with Drill

© 2015 MapR Technologies **MAPR** 1



What is Drill?



APACHE
DRILL

© 2015 MapR Technologies **MAPR** 2



PROPRIETARY AND CONFIDENTIAL INFORMATION
©2015 MapR Technologies, Inc. All Rights Reserved



What is Drill?

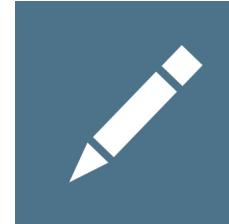
© 2015 MapR Technologies  3

Learning Goals

- ▶ Perform Familiar SQL Queries: Structured Data
- ▶ Perform Familiar SQL Queries: Range of Data Types
- ▶ Demonstrate Drill & Semi-structured Data

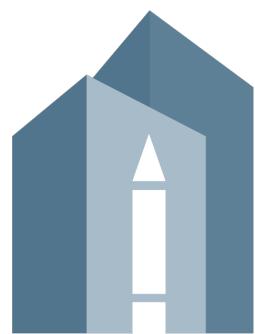
© 2015 MapR Technologies  4

 Use Case



© 2015 MapR Technologies  5

 Use Case



**BIG OFFICE
SUPPLY CO.**



© 2015 MapR Technologies  6



PROPRIETARY AND CONFIDENTIAL INFORMATION
©2015 MapR Technologies, Inc. All Rights Reserved



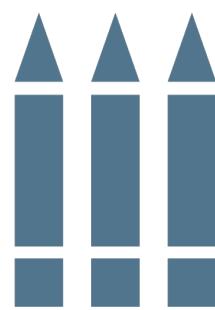
Familiar SQL Queries: Structured Data



© 2015 MapR Technologies **MAPR** 7



Familiar SQL Queries: Structured Data



© 2015 MapR Technologies **MAPR** 8





Familiar SQL Queries: Structured Data

```
orders(order_id BIGINT, month STRING, cust_id BIGINT,
      state STRING, prod_id BIGINT, order_total INT)
```

order_id	month	cust_id	state	prod_id	order_total
67212	June	10001	ca	909	13
70302	May	10004	ga	420	11



Familiar SQL Queries: Structured Data

Query 1

```
SELECT `month`, SUM(order_total) as sales
FROM hive.orders
GROUP BY `month`
ORDER BY sales desc;
```

Query 2

```
SELECT `month`, `state`, SUM(order_total) as sales
FROM hive.orders
WHERE `month`='June'
GROUP BY `month`, `state`
ORDER BY sales desc;
```

Query 3

```
SELECT `prod_id`, SUM(order_total) as sales
FROM hive.orders
GROUP BY `prod_id`
ORDER BY 2 desc limit 10;
```





Refer to Lab Guide Exercise 1.1



Knowledge Check

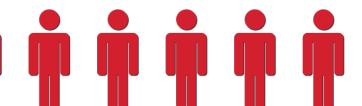
Please indicate if the following statement is true or false:

I can use Drill to easily perform standard ANSI SQL queries on structured data





Familiar SQL Queries: Range of Data Types



© 2015 MapR Technologies 13



Familiar SQL Queries: Range of Data Types

```
[@jallen - root@maprdemo:~/mapr/demo.mapr.com/data/nested] jallen - root@maprdemo:~/mapr/demo.mapr.com/data/nested@clicks.sh - 225x56
["trans_id":31928,"date":"2014-04-26","time":"12:17:12","user_info":{"cust_id":22526,"device":"IOS5","state":"11","trans_info":{"prod_id":1374,21,"purch_flag":"false"}}
("trans_id":31026,"date":"2014-04-28","time":"13:58:29","user_info":{"cust_id":16368,"device":"AOS4.2","state":"n/a","trans_info":{"prod_id":[],"purch_flag":"false"}}
("trans_id":33848,"date":"2014-04-18","time":"04:44:42","user_info":{"cust_id":21449,"device":"IOS5","state":"oh","trans_info":{"prod_id":[],"purch_flag":"false"}}
("trans_id":32383,"date":"2014-04-18","time":"06:27:47","user_info":{"cust_id":26323,"device":"IOS5","state":"oh","trans_info":{"prod_id":178,47,"purch_flag":"false"}}
("trans_id":32384,"date":"2014-04-18","time":"06:27:47","user_info":{"cust_id":26324,"device":"IOS5","state":"oh","trans_info":{"prod_id":178,47,"purch_flag":"false"}}
("trans_id":34022,"date":"2014-04-23","time":"01:46:05","user_info":{"cust_id":15957,"device":"IOS7","state":"sc","trans_info":{"prod_id":[],"purch_flag":"false"}}
("trans_id":35898,"date":"2014-04-18","time":"13:28:56","user_info":{"cust_id":28677,"device":"IOS7","state":"ny","trans_info":{"prod_id":[],"purch_flag":"false"}}
("trans_id":32421,"date":"2014-04-15","time":"11:00:53","user_info":{"cust_id":23599,"device":"IOS5","state":"i1","trans_info":{"prod_id":1,0,87,170,445,6,0,2,7,7,118,"purch_flag":true}}
("trans_id":32422,"date":"2014-04-15","time":"11:00:53","user_info":{"cust_id":16000,"device":"IOS5","state":"i1","trans_info":{"prod_id":1,0,87,170,445,6,0,2,7,7,118,"purch_flag":true}}
("trans_id":35373,"date":"2014-04-22","time":"06:53:28","user_info":{"cust_id":15342,"device":"IOS5","state":"m1","trans_info":{"prod_id":1,0,87,170,445,6,0,2,7,7,118,"purch_flag":true}}
("trans_id":30778,"date":"2014-04-13","time":"01:20:05","user_info":{"cust_id":16996,"device":"AOS4.2","state":"oh","trans_info":{"prod_id":149,499,151,"purch_flag":true}}
("trans_id":32120,"date":"2014-04-28","time":"06:58:19","user_info":{"cust_id":21402,"device":"IOS5","state":"m1","trans_info":{"prod_id":[],"purch_flag":false}}
("trans_id":32120,"date":"2014-04-28","time":"06:58:19","user_info":{"cust_id":21402,"device":"IOS5","state":"m1","trans_info":{"prod_id":[],"purch_flag":false}}
("trans_id":32086,"date":"2014-04-15","time":"21:08:15","user_info":{"cust_id":15344,"device":"IOS5","state":"ny","trans_info":{"prod_id":1,0,87,170,445,6,0,2,7,7,118,"purch_flag":false}}
("trans_id":32086,"date":"2014-04-15","time":"21:08:15","user_info":{"cust_id":15344,"device":"IOS5","state":"ny","trans_info":{"prod_id":1,0,87,170,445,6,0,2,7,7,118,"purch_flag":false}}
("trans_id":32965,"date":"2014-04-15","time":"20:41:58","user_info":{"cust_id":15493,"device":"AOS4.4","state":"u1","trans_info":{"prod_id":242,0,2,3,165,26,8,0,"purch_flag":false}}
("trans_id":32643,"date":"2014-04-26","time":"15:07:00","user_info":{"cust_id":15137,"device":"AOS4.2","state":"tx","trans_info":{"prod_id":1452,19,3,15,954,41,"purch_flag":false}}
```

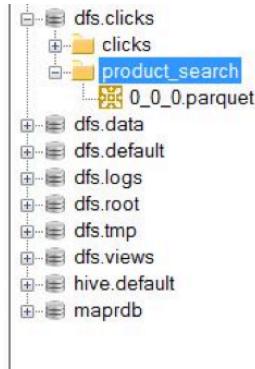
```
[@jallen - root@maprdemo:~/mapr/demo.mapr.com/data/flat] jallen - root@maprdemo:~/mapr/demo.mapr.com/data/flat/logs/2014/6 - ssh - 181x57
["trans_id":24102,"date":"06/29/2014","time":"05:16:08","cust_id":4,"device":"IOS5","state":"hi","camp_id":0,"keywords":"on","prod_id":133,"purch_flag":false}
("trans_id":24105,"date":"06/25/2014","time":"01:23:44","cust_id":3414,"device":"IOS5","state":ut,"camp_id":5,"keywords":go,"prod_id":95,"purch_flag":false}
("trans_id":24109,"date":"06/17/2014","time":19:32:51,"cust_id":43,"device":IOS7,"state":f1,"camp_id":9,"keywords":boys,"prod_id":247,"purch_flag":false}
("trans_id":24124,"date":"06/22/2014","time":05:19:40,"cust_id":4,"device":IOS5,"state":oh,"camp_id":0,"keywords":work,"prod_id":14,"purch_flag":false}
("trans_id":24124,"date":"06/22/2014","time":05:19:40,"cust_id":4,"device":IOS5,"state":oh,"camp_id":0,"keywords":work,"prod_id":14,"purch_flag":false}
("trans_id":24125,"date":"06/11/2014","time":06:45:09,"cust_id":814,"device":IOS5,"state":wa,"camp_id":3,"keywords":ready,"prod_id":287,"purch_flag":false}
("trans_id":24125,"date":"06/11/2014","time":06:45:09,"cust_id":814,"device":IOS5,"state":wa,"camp_id":3,"keywords":ready,"prod_id":287,"purch_flag":false}
("trans_id":24128,"date":"06/18/2014","time":16:44:37,"cust_id":1918,"device":IOS5,"state":tn,"camp_id":0,"keywords":believe,"prod_id":91,"purch_flag":false}
("trans_id":24129,"date":"06/30/2014","time":03:39:47,"cust_id":66537,"device":IOS5,"state":ca,"camp_id":3,"keywords":believe,"prod_id":91,"purch_flag":false}
("trans_id":24152,"date":"06/21/2014","time":13:18:18,"cust_id":0,"device":AOS4.3,"state":oh,"camp_id":12,"keywords":be,"prod_id":388,"purch_flag":false}
("trans_id":24164,"date":"06/12/2014","time":06:41:21,"cust_id":8,"device":IOS5,"state":tx,"camp_id":10,"keywords":my,"prod_id":33,"purch_flag":false}
("trans_id":24166,"date":"06/15/2014","time":22:30:54,"cust_id":0,"device":IOS5,"state":tomorrow,"prod_id":21,"purch_flag":false}
("trans_id":24277,"date":"06/27/2014","time":16:47:21,"cust_id":278,"device":AOS4.4,"state":oh,"camp_id":9,"keywords":sometime,"prod_id":430,"purch_flag":false}
("trans_id":24178,"date":"06/14/2014","time":22:10:50,"cust_id":12,"device":IOS5,"state":az,"camp_id":14,"keywords":want,"prod_id":227,"purch_flag":false}
("trans_id":24187,"date":"06/27/2014","time":13:37:15,"cust_id":2508,"device":md,"camp_id":1,"keywords":help,"prod_id":0,"purch_flag":false}
```

© 2015 MapR Technologies 14





Familiar SQL Queries: Range of Data Types



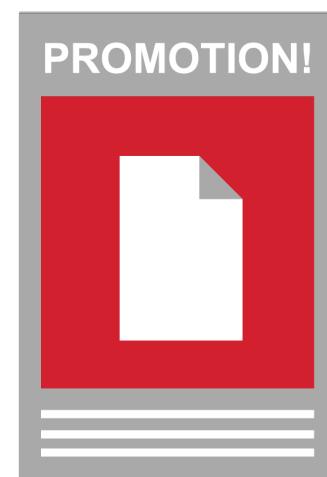
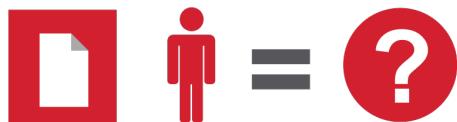
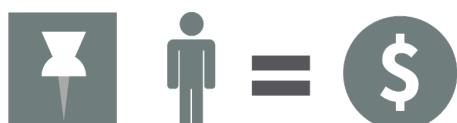
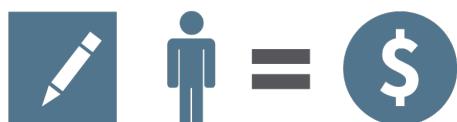
1 Metadata does not exist

Data Preview:

	cust_id	order_total	state	prod_id
▶ 1	10001	13	ca	2
2	10004	11	ga	31
3	10011	76	fl	752

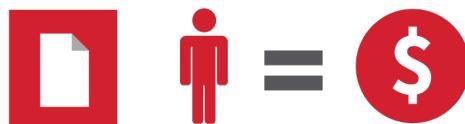
© 2015 MapR Technologies  15

Familiar SQL Queries: Range of Data Types

© 2015 MapR Technologies  16



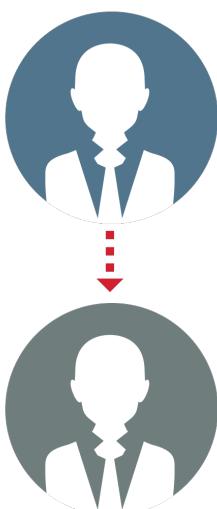
Familiar SQL Queries: Range of Data Types



PROMOTION!

© 2015 MapR Technologies  17

Familiar SQL Queries: Range of Data Types

**Without Drill:**

- Enlist help from Engineer
- Lengthy ETL process
- Increased effort, cost & time

© 2015 MapR Technologies  18



Familiar SQL Queries: Range of Data Types



With Drill:

- Easy query
- Discover data on the fly
- No upfront schema definitions
- Manipulate nested data without flattening at design or run time
- Standard ANSI SQL
- Better expected results

© 2015 MapR Technologies  19



Familiar SQL Queries: Range of Data Types

```
products(HBASE_ROW_KEY,details:name,  
details:category,pricing:price)
```

HBASE_ROW_KEY	name	category	price
5	12-1/2 Diameter Round Wall Clock	Office Furnishings	20
6	12 Colored Short Pencils	Pens & Art Supplies	3

© 2015 MapR Technologies  20





Familiar SQL Queries: Range of Data Types

Query 1

```
SELECT *
FROM (SELECT name, count(*)
      FROM (SELECT
              FLATTEN(clicks.trans_info.prod_id) name
             FROM dfs.`data/nested/clicks/clicks.parquet` clicks) a
        GROUP BY name
       ORDER BY 2 desc) b limit 4;
```

Query 2

```
SELECT clicks.user_info.cust_id
      FROM dfs.`data/nested/clicks/clicks.json` clicks
     WHERE clicks.trans_info.purch_flag = 'false';
```

© 2015 MapR Technologies  21

Familiar SQL Queries: Range of Data Types

Query 3

```
SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date`
      FROM hive.products as prod,
           (SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id, clk.`date`
              FROM maprdb.customers as cust,
                   (SELECT cast(clicks.user_info.cust_id as bigint) as cust_id, cast
                      (FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id, clicks.trans_info.purch_flag as
                       purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd') as `date`
                     FROM dfs.clicks.`clicks.json` as clicks) as clk
                 WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false') as cust_clk
           WHERE cust_clk.prod_id = prod.prod_id
           and cust_clk.`date` between '2014-01-01' and '2014-03-31'
```

© 2015 MapR Technologies  22



Refer to Lab Guide Exercise 1.2



Knowledge Check

Please select all of the answers that apply

With Drill, we can use ANSI SQL to query:

- Structured data in Hive or HBase
- Unstructured JSON or parquet data
- Flat text files
- Nested data
- Any combination of these data types



 Drill & Semi-structured Data



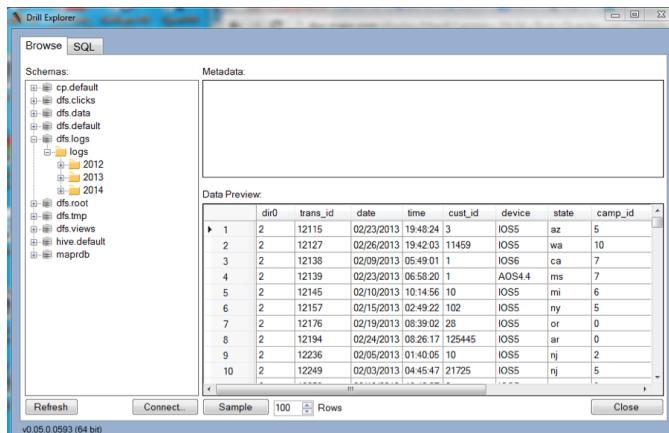
© 2015 MapR Technologies  25

 Drill & Semi-structured Data









© 2015 MapR Technologies  26



Drill & Semi-structured Data

Drill Explorer interface showing semi-structured data.

Schemas:

- cp.default
- dfs.clicks
- dfs.data
- dfs.default
- dfs.logs
- dfs.root
- dfs.tmp
- dfs.views
- hive.default
- maprdb

Metadata: (Empty)

Data Preview:

drn0	trans_id	date	time	cust_id	device	state	camp_id	
1	2	12115	02/23/2013	19:48:24	3	IOS5	az	5
2	2	12127	02/26/2013	19:42:03	11459	IOS5	wa	10
3	2	12138	02/09/2013	05:49:01	1	IOS6	ca	7
4	2	12139	02/23/2013	06:58:20	1	AOS4.4	ms	7
5	2	12145	02/10/2013	10:14:56	10	IOS5	mi	6
6	2	12157	02/15/2013	02:49:22	102	IOS5	ny	5
7	2	12176	02/19/2013	08:39:02	28	IOS5	or	0
8	2	12194	02/24/2013	08:26:17	125445	IOS5	ar	0
9	2	12236	02/05/2013	01:40:05	10	IOS5	nj	2
10	2	12249	02/03/2013	04:45:47	21725	IOS5	nj	5

Buttons: Refresh, Connect..., Sample, 100 Rows, Close

v0.05.0.0593 (64 bit)

© 2015 MapR Technologies 27

Drill & Semi-structured Data

Drill Explorer interface showing semi-structured data.

Schemas:

- cp.default
- dfs.clicks
- dfs.data
- dfs.default
- dfs.logs
- dfs.root
- dfs.tmp
- dfs.views
- hive.default
- maprdb
- orders
- customers
- embeddedclicks

Metadata:

Column_Index	Data_Type	Value_Width
0	Varchar(5)	5
1	Varchar(22)	22
2	Varchar(4)	4
3	Varchar(8)	8
4	Varchar(8)	8
5	Varchar(2)	2

Data Preview:

column_0	column_1	column_2	column_3	column_4	column_5
16841	"Lisa Wells"	"b"	"FEMALE"	"26-35"	40
19905	"Neely Nova"	"va"	"MALE"	"15-20"	45
17718	"William Bush"	"l"	"FEMALE"	"51-100"	29
21975	"Gerald Staples"	"ma"	"MALE"	"26-35"	28
20468	"William Conners"	"de"	"MALE"	"36-50"	32
16670	"John Rubio"	"oh"	"MALE"	"15-20"	42
18600	"Jay Holland"	"l"	"FEMALE"	"21-25"	43
23438	"James Hogsett"	"ne"	"FEMALE"	"15-20"	30
20669	"Florence Black"	"pa"	"FEMALE"	"26-35"	24
22880	"Judith Stuart"	"ga"	"MALE"	"21-25"	25
18467	"Dale Johnson"	"ga"	"FEMALE"	"15-20"	24
19828	"Patrick Cote"	"ms"	"MALE"	"36-50"	45

Buttons: Refresh, Connect..., Sample, 100 Rows, Close

v0.05.0.0593 (64 bit)

© 2015 MapR Technologies 28



Drill & Semi-structured Data

Drill Explorer

Browse SQL

Schemas:

- cp.default
- dfs.clicks
- dfs.data
- dfs.default
- dfs.logs
 - logs
 - 2012
 - 1
 - log.json
 - 10
 - 11
 - 12
 - 2
 - 3
 - 4
 - 5
 - 6
 - 7
 - 8
 - 9
 - 2013
 - 2014
 - dfs.root

Metadata:

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1	Metadata does not exist		

Data Preview:

	trans_id	date	time	cust_id	device	state	camp_id	keywords
1	100	01/30/2012	17:14:14	4	AOS4.2	ms	16	i
2	120	01/29/2012	06:52:05	0	IOS5	va	4	crime
3	138	01/30/2012	10:03:10	0	IOS5	ky	5	laughing
4	154	01/07/2012	20:24:28	2	AOS4.2	ky	10	that
5	160	01/25/2012	11:43:53	9107	AOS4.2	oh	9	me
6	169	01/17/2012	12:14:27	1119	IOS5	ny	17	the
7	180	01/13/2012	17:57:47	5145	IOS5	nc	7	you
8	182	01/09/2012	17:05:30	1	AOS4.4	ny	13	sir
9	184	01/04/2012	14:59:08	7927	AOS4.2	ar	3	one
10	192	01/21/2012	11:14:40	21254	IOS5	ca	10	maka

© 2015 MapR Technologies 29

Drill & Semi-structured Data

Drill Explorer

Browse SQL

Schemas:

- cp.default
- dfs.clicks
- dfs.data
- dfs.default
- dfs.logs
 - logs
 - 2012
 - 2013
 - 2014
 - dfs.root
 - dfs.tmp
 - dfs.views
 - hive.default
 - orders
 - maprdb
 - customers
 - address
 - loyalty
 - personal
 - embeddedclicks
 - products

Metadata: hive.default.orders

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1	order_id	BIGINT	YES
2	month	VARCHAR	YES
3	cust_id	BIGINT	YES
4	state	VARCHAR	YES
5	prod_id	BIGINT	YES

Data Preview:

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	nj	104	134
7	68553	June	10021	ca	117	67
8	68109	June	10022	tx	337	10
9	68526	June	10025	mi	11	63
10	69362	June	10028	tx	430	65
11	68624	June	10030	fl	808	51

Refresh Connect... Sample 100 Rows Close

© 2015 MapR Technologies 30



Drill & Semi-structured Data

Drill Explorer

Browse SQL

View Definition SQL:

```
select * from `clicks/clicks.json` limit 2;
```

Total Number of Records:: 2

	trans_id	date	time	user_info	trans_info
▶	1	2014-04-26	12:17:12	{ "cust_id": 22526, "device": "IOS5", "state": "il"}	{ "prod_id": [174, 2], "purch_flag": "false" }
	2	2014-04-20	13:50:29	{ "cust_id": 16368, "device": "AOS4.2", "state": "nc"}	{ "prod_id": [], "purch_flag": "false" }

© 2015 MapR Technologies  31

Drill & Semi-structured Data

Tableau - Book2

Data

Dimensions

- Age
- CustId
- Gender
- Membership
- Name
- State
- Measure Names

Measures

- Agg Rev
- Latitude (generated)
- Longitude (generated)
- Number of Records
- Measure Values

Pages

Filters

Marks

Rows

Columns

Age / Membership

Number of Records

Gender

TABLEAU CHART DATA (Approximate values)

Age / Membership	Gender	Number of Records
"15-20"	MALE	~100
"15-20"	OTHER	~10
"15-20"	FEMALE	~80
"21-25"	MALE	~100
"21-25"	OTHER	~10
"21-25"	FEMALE	~80
"26-35"	MALE	~100
"26-35"	OTHER	~10
"26-35"	FEMALE	~80
"36-50"	MALE	~80
"36-50"	OTHER	~10
"36-50"	FEMALE	~70
"51-100"	MALE	~80
"51-100"	OTHER	~10
"51-100"	FEMALE	~70

Sheet 1

42 marks 1 row by 15 columns SUM(Number of Records): 1,561

© 2015 MapR Technologies  32





Refer to Lab Guide Exercise 1.3



Knowledge Check

- View all data sources available to Drill
- Explore data saved in selected sources
- See meta data for selected source
- Choose between exploring data sources or SQL use to query this data

4

COLUMN_NAME	DATA_TYPE	IS_NULLABLE
order_id	BIGINT	YES
month	VARCHAR	YES
cust_id	BIGINT	YES
state	VARCHAR	YES
prod_id	BIGINT	YES

1

2

3

Refresh Connect... Sample 100 Rows Close

Schemas:

- if- dfs cp.default
- if- dfs clicks
- if- dfs data
- if- dfs default
- if- dfs logs
 - logs
 - 2012
 - 2013
 - 2014
- if- dfs root
- if- dfs tmp
- if- dfs views
- if- hive default
 - orders
- if- maprdb
 - customers
 - address
 - loyalty
 - personal
 - embeddedclicks
 - products





Next Steps



Lesson 2:
Querying
Self Describing Data

© 2015 MapR Technologies **MAPR** 35



MAPR Academy

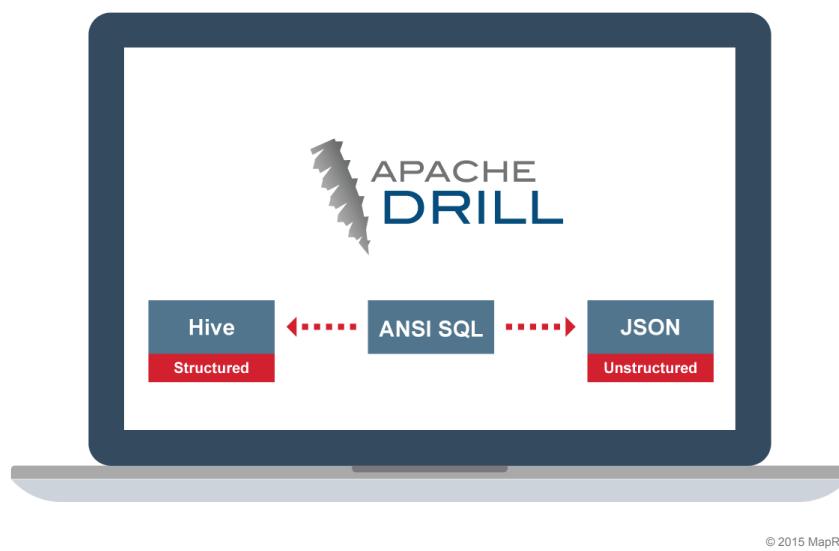
Data Analysis with Drill

Lesson 2: Query Self Describing Data

© 2015 MapR Technologies **MAPR** 1



What is Drill?



© 2015 MapR Technologies **MAPR** 2



PROPRIETARY AND CONFIDENTIAL INFORMATION
©2015 MapR Technologies, Inc. All Rights Reserved



What is Drill?

Drill Explorer

Browse SQL

Schemas:

- cp.default
- dfs.clicks
- dfs.data
- dfs.default
- dfs.logs
 - logs
 - 2012
 - 2013
 - 2014
- dfs.root
- dfs.tmp
- dfs.views
- hive.default
 - orders
- maprdb
 - customers
 - address
 - loyalty
 - personal
 - embeddedclicks
 - products

Metadata: hive.default.orders

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1	order_id	BIGINT	YES
2	month	VARCHAR	YES
3	cust_id	BIGINT	YES
4	state	VARCHAR	YES
5	prod_id	BIGINT	YES

Data Preview:

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	nj	104	134
7	68553	June	10021	ca	117	67
8	68109	June	10022	tx	337	10
9	68526	June	10025	mi	11	63
10	69362	June	10028	tx	430	65
11	68624	June	10030	fl	808	51

Refresh

Connect...

Sample

100 Rows

© 2015 MapR Technologies  3



What is Drill?

Drill Explorer

Browse SQL

Schemas:

- cp.default
- dfs.clicks
- dfs.data
- dfs.default
- dfs.logs
 - logs
 - 2012
 - 2013
 - 2014
- dfs.root
- dfs.tmp
- dfs.views
- hive.default
 - orders
- maprdb
 - customers
 - address
 - loyalty
 - personal
 - embeddedclicks
 - products

Metadata: hive.default.orders

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1	order_id	BIGINT	YES
2	month	VARCHAR	YES
3	cust_id	BIGINT	YES
4	state	VARCHAR	YES
5	prod_id	BIGINT	YES

Data Preview:

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	nj	104	134
7	68553	June	10021	ca	117	67
8	68109	June	10022	tx	337	10
9	68526	June	10025	mi	11	63
10	69362	June	10028	tx	430	65
11	68624	June	10030	fl	808	51

Refresh

Connect...

Sample

100 Rows

© 2015

4

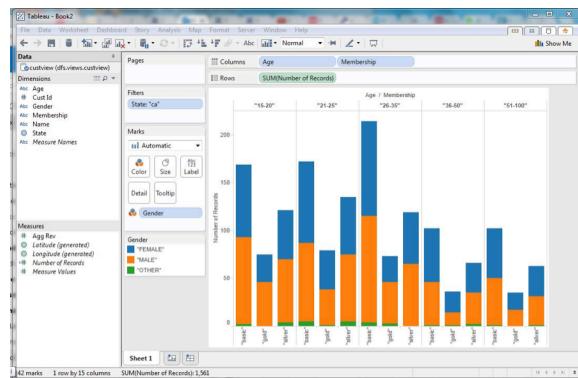




What is Drill?



MicroStrategy



© 2015 MapR Technologies **MAPR** 5



Learning Goals

- ▶ Define Data Types
- ▶ Define Data Interaction & Schema Detection
- ▶ Create Complex Query Examples
- ▶ Create Views for use with BI Tools

© 2015 MapR Technologies **MAPR** 6



Define Data Types


APACHE DRILL

Hive
 Structured




ANSI SQL




JSON
 Unstructured

© 2015 MapR Technologies 7

Define Data Types

```
First Query:  

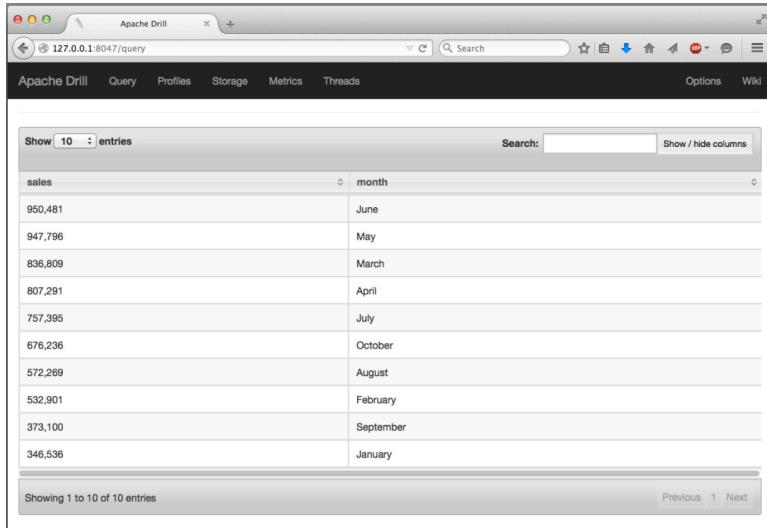
SELECT `month`,  

SUM(order_total) as  

sales FROM hive.orders  

GROUP BY `month` ORDER  

BY sales desc;
```

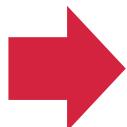


sales	month
950,481	June
947,796	May
836,809	March
807,291	April
757,395	July
676,236	October
572,269	August
532,901	February
373,100	September
346,536	January

© 2015 MapR Technologies 8



Define Data Types



order_id	month	cust_id	state	prod_id	order_total
67212	June	10001	ca	909	13
70302	May	10004	ga	420	11
15451	Feburary	10009	in	107	26
95413	August	10019	oh	594	40

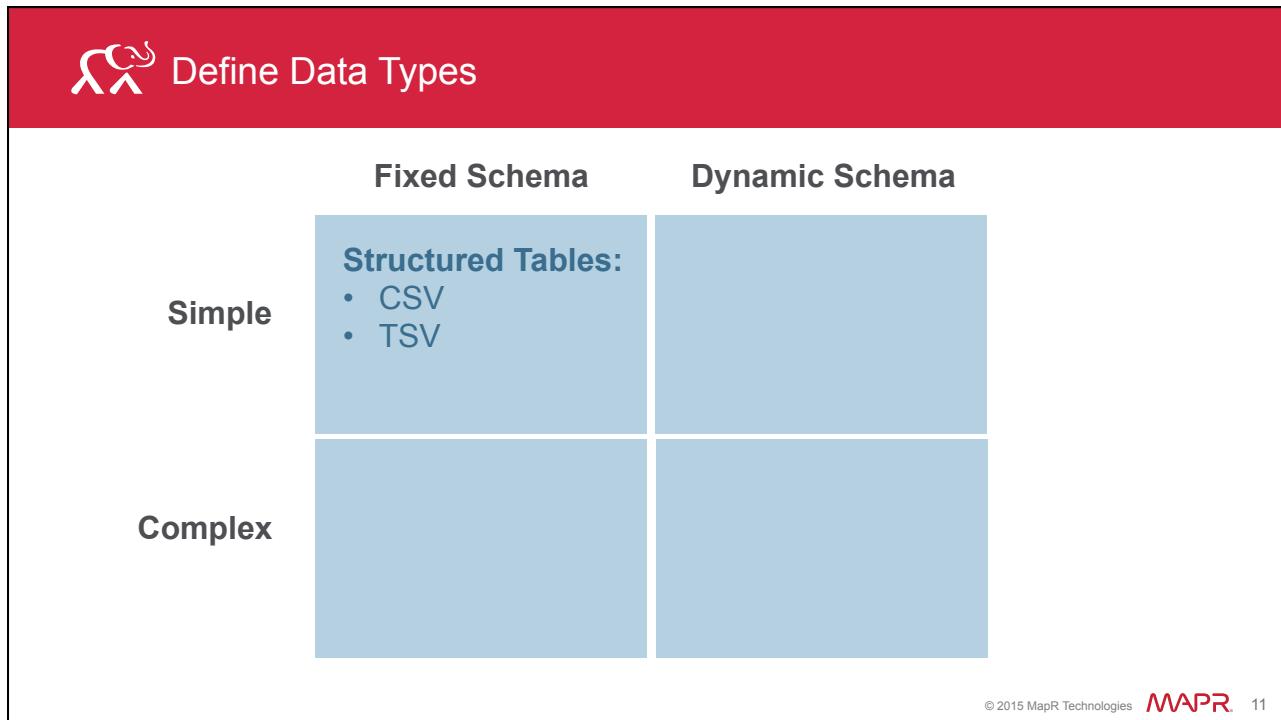
© 2015 MapR Technologies  9

Define Data Types



order_id	month	cust_id	state	prod_id	order_total
67212	June	10001	ca	909	13
70302	May	10004	ga	420	11
15451	Feburary	10009	in	107	26
95413	August	10019	oh	594	40

© 2015 MapR Technologies  10



 Define Data Types

cust_id	name	state	gender	age	agg_rev	membership
16841	Lisa Wells	tx	FEMALE	26-35	40	basic
19905	Neely Nova	va	MALE	15-20	45	basic
17718	William Bush	il	FEMALE	51-100	29	basic
21975	Gerald Staples	ma	MALE	26-35	28	basic

© 2015 MapR Technologies  12





Define Data Types

cust_id	name	state	gender	age	agg_rev	membership	reviews
16841	Lisa Wells	tx	FEMALE	26-35	40	basic	NULL
19905	Neely Nova	va	MALE	15-20	45	basic	NULL
17718	William Bush	il	FEMALE	51-100	29	basic	NULL
21975	Gerald Staples	ma	MALE	26-35	28	basic	TiqA9ybgewk

© 2015 MapR Technologies  13



Define Data Types

Simple
Complex

Fixed Schema

- Structured Tables:**
- CSV
 - TSV

Dynamic Schema

- Semi-Structured Table :**
- Hbase
 - MapR-DB

© 2015 MapR Technologies  14



Define Data Types

cust_id	name	state	gender	age	agg_rev	membership	reviews
16841	Lisa Wells	tx	FEMALE	26-35	40	basic	NULL
19905	Neely Nova	va	MALE	15-20	45	basic	NULL
17718	William Bush	il	FEMALE	51-100	29	basic	NULL
21975	Gerald Staples	ma	MALE	26-35	28	basic	TiqA9ybgewk

© 2015 MapR Technologies 15

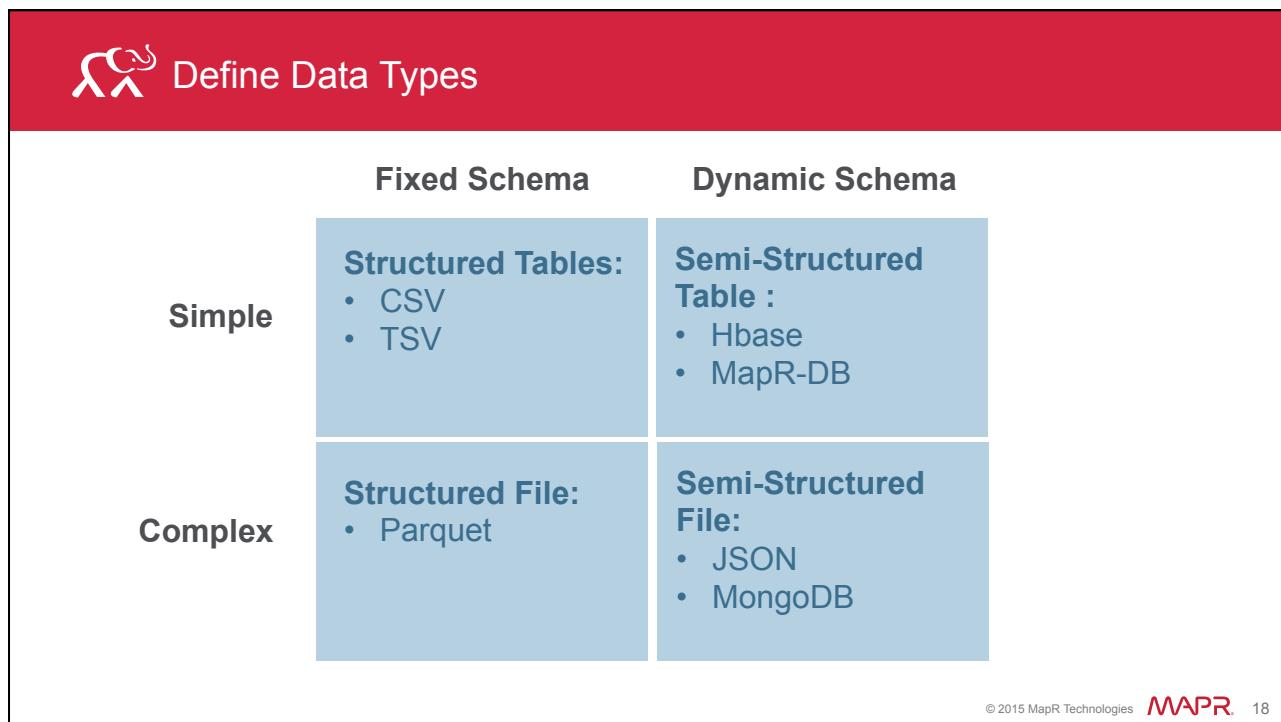
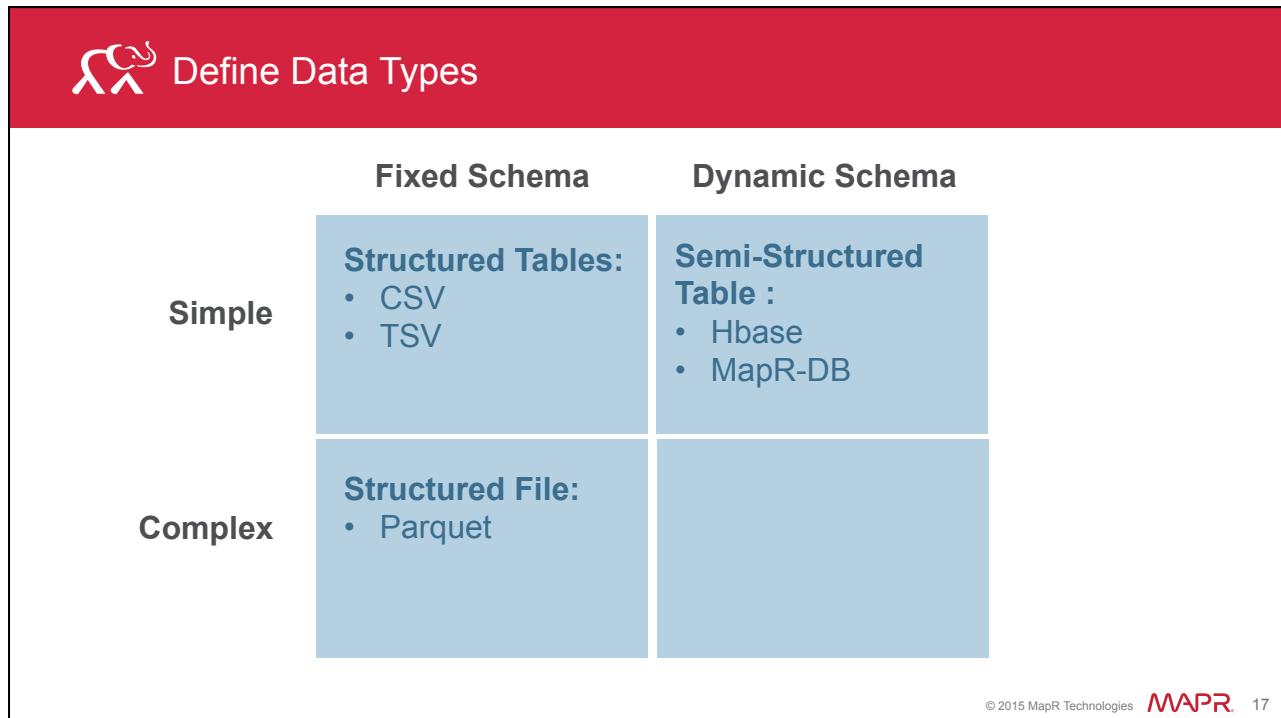
Define Data Types

Data Preview:

	trans_id	date	time	user_info	trans_info
► 1	31920	2014-04-26	12:17:12	{ "cust_id": 22526, "device": "IOS5", "state": "il" }	{ "prod_id": [174, 2], "purch_flag": "false" }
2	31026	2014-04-20	13:50:29	{ "cust_id": 16368, "device": "AOS4.2", "state": "nc" }	{ "prod_id": [], "purch_flag": "false" }
3	33848	2014-04-10	04:44:42	{ "cust_id": 21449, "device": "IOS6", "state": "oh" }	{ "prod_id": [582], "purch_flag": "false" }
4	32383	2014-04-18	06:27:47	{ "cust_id": 20323, "device": "IOS5", "state": "oh" }	{ "prod_id": [710, 47], "purch_flag": "false" }
5	32359	2014-04-19	23:13:25	{ "cust_id": 15360, "device": "IOS5", "state": "ca" }	{ "prod_id": [0, 8, 170, 173, 1, 124, 46, 764, 30, 711, 0, 3, 25], "purch_flag": "false" }
6	30422	2014-04-23	01:46:05	{ "cust_id": 15957, "device": "IOS7", "state": "sc" }	{ "prod_id": [628], "purch_flag": "false" }
7	35898	2014-04-18	13:28:56	{ "cust_id": 20677, "device": "IOS7", "state": "ny" }	{ "prod_id": [], "purch_flag": "false" }
8	32421	2014-04-15	11:00:53	{ "cust_id": 23599, "device": "IOS5", "state": "ri" }	{ "prod_id": [1, 0, 87, 170, 445, 6, 0, 2, 7, 7, 110], "purch_flag": "false" }
9	39447	2014-04-21	01:34:06	{ "cust_id": 16122, "device": "IOS6", "state": "il" }	{ "prod_id": [26, 504, 1, 47, 31, 116, 167, 71, 20, 305, 554, 0, 189, 384], "purch_flag": "false" }
10	35373	2014-04-02	06:53:29	{ "cust_id": 15342, "device": "IOS5", "state": "ms" }	{ "prod_id": [15, 0], "purch_flag": "false" }
11	30778	2014-04-13	01:20:05	{ "cust_id": 16996, "device": "AOS4.2", "state": "oh" }	{ "prod_id": [40, 499, 15], "purch_flag": "true" }
12	32120	2014-04-28	06:58:19	{ "cust_id": 21402, "device": "IOS5", "state": "nm" }	{ "prod_id": [], "purch_flag": "false" }
13	30706	2014-04-15	21:08:15	{ "cust_id": 15344, "device": "IOS7", "state": "ny" }	{ "prod_id": [27], "purch_flag": "false" }
14	38207	2014-04-28	18:57:55	{ "cust_id": 16841, "device": "AOS4.2", "state": "mo" }	{ "prod_id": [181, 0, 33, 708, 92], "purch_flag": "false" }
15	32955	2014-04-15	20:41:58	{ "cust_id": 15492, "device": "AOS4.4", "state": "wi" }	{ "prod_id": [242, 0, 2, 3, 166, 26, 0, 0], "purch_flag": "false" }
...

© 2015 MapR Technologies 16







Knowledge Check

	Fixed Schema	Dynamic Schema
Simple	Structured Tables: <ul style="list-style-type: none">• CSV• TSV	Semi-Structured Table : <ul style="list-style-type: none">• Hbase• MapR-DB
Complex	Structured File: <ul style="list-style-type: none">• Parquet	Semi-Structured File: <ul style="list-style-type: none">• JSON• MongoDB

© 2015 MapR Technologies  19

Define Data Interaction & Schema Detection

```
SELECT `month`, SUM(order_total) as sales
FROM hive.orders GROUP BY `month` ORDER BY sales desc;
```

© 2015 MapR Technologies  20



Define Data Interaction & Schema Detection

```
SELECT `month`, SUM(order_total) as sales
FROM hive.orders GROUP BY `month` ORDER BY sales desc;
```

© 2015 MapR Technologies  21



Define Data Interaction & Schema Detection

```
SELECT `month`, SUM(order_total) as sales
FROM hive.orders GROUP BY `month` ORDER BY sales desc;
```

© 2015 MapR Technologies  22





Define Data Interaction & Schema Detection

The screenshot shows the Drill Explorer interface. On the left, there's a tree view of schemas: cp default, dfs.clicks, dfs.data, dfs.default, dfs.logs (with subfolders 2012, 2013, 2014), dfs.root, dfs.tmp, dfs.views, hive.default, maprdb (with subfolders customers, address, loyalty, personal, embeddedclicks, products). Below the tree are Refresh and Connect... buttons. On the right, a red callout points to the word "hive" in the SQL query with the text "Storage system type".

```
SELECT `month`, SUM(order_total)
as sales FROM hive.orders GROUP
BY `month` ORDER BY sales desc;
```

© 2015 MapR Technologies 23



Define Data Interaction & Schema Detection

The screenshot shows the Drill Explorer interface. On the left, there's a tree view of schemas: cp.default, dfs.clicks, dfs.data, dfs.default, dfs.logs (with subfolders 2012, 2013, 2014), dfs.root, dfs.tmp, dfs.views, hive.default, maprdb (with subfolders customers, address, loyalty, personal, embeddedclicks, products). Below the tree are Refresh and Connect... buttons. On the right, two red callouts point to the word "orders" in the SQL query. The top callout is labeled "Storage system type" and the bottom one is labeled "Table Name".

```
SELECT `month`, SUM(order_total)
as sales FROM hive.orders GROUP
BY `month` ORDER BY sales desc;
```

© 2015 MapR Technologies 24





Define Data Interaction & Schema Detection

```
SELECT clicks.user_info.cust_id  
FROM dfs.`data/nested/clicks/clicks.json` clicks  
WHERE clicks.trans_info.purch_flag = 'false';
```

© 2015 MapR Technologies  25



Define Data Interaction & Schema Detection

Storage system type

```
SELECT clicks.user_info.cust_id  
FROM dfs.`data/nested/clicks/clicks.json` clicks  
WHERE clicks.trans_info.purch_flag = 'false';
```

© 2015 MapR Technologies  26



Define Data Interaction & Schema Detection

Storage system type

```
SELECT clicks.user_info.cust_id
FROM dfs.`data/nested/clicks/clicks.json` clicks
WHERE clicks.trans_info.purch_flag = 'false';
```

Path Name

© 2015 MapR Technologies 27

Define Data Interaction & Schema Detection

The screenshot shows the Drill Explorer interface. On the left, the 'Schemas' tree view is expanded to show various databases and tables, including 'dfs.clicks', 'dfs.logs', 'hive.default', and 'maprdb'. A red box highlights this tree view. On the right, the 'Metadata' tab for the 'hive.default.orders' table is displayed, showing columns: order_id, month, cust_id, state, prod_id, and order_total, along with their data types (BIGINT, VARCHAR) and nullability (YES). Below the metadata is a 'Data Preview' table showing sample data rows.

COLUMN_NAME	DATA_TYPE	IS_NULLABLE
order_id	BIGINT	YES
month	VARCHAR	YES
cust_id	BIGINT	YES
state	VARCHAR	YES
prod_id	BIGINT	YES

order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909
2	70302	June	10004	ga	420
3	69090	June	10011	fl	44
4	68834	June	10012	ar	0
5	71220	June	10018	az	411
6	61287	June	1001	nj	104
7	68553	June	10021	ca	117
8	68109	June	10022	tx	337
9	68526	June	10025	mi	11
10	69362	June	10028	tx	430
11	68624	June	10030	fl	808

© 2015 MapR Technologies 28



Define Data Interaction & Schema Detection

dfs

Hive

HBase, MapR-DB

Refresh Connect...

© 2015 MapR Technologies 29

Define Data Interaction & Schema Detection

Metadata: hive.default.orders

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1	order_id	BIGINT	YES
2	month	VARCHAR	YES
3	cust_id	BIGINT	YES
		VARCHAR	YES
		BIGINT	YES

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	nj	104	134
7	68553	June	10021	ca	117	67
8	68109	June	10022	tx	337	10
9	68526	June	10025	mi	11	63
10	69362	June	10028	tx	430	65
11	68624	June	10030	fl	808	51

Refresh Connect... Sample 100 Rows Close

© 2015 MapR Technologies 30





Define Data Interaction & Schema Detection

```
SELECT `month`, SUM(order_total) as sales
FROM hive.orders GROUP BY `month` ORDER BY sales desc;
```

Table Name

The screenshot shows the Drill Explorer interface. On the left, the 'Schemas' tree view shows various databases and tables, with 'hive.default.orders' selected. On the right, the 'Metadata: hive.default.orders' panel displays the table structure with columns: order_id, month, cust_id, state, prod_id, and order_total. The 'IS_NULLABLE' column indicates that all columns are nullable. Below this, the 'Data Preview' section shows a sample of 11 rows from the table.

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	nj	104	134
7	68553	June	10021	ca	117	67
8	68109	June	10022	bx	337	10
9	68526	June	10025	mi	11	63
10	69362	June	10028	bx	430	65
11	68624	June	10030	fl	808	51

© 2015 MapR Technologies 31



Define Data Interaction & Schema Detection

This screenshot is identical to the one above, showing the Drill Explorer interface with the 'orders' table selected. The 'Metadata' and 'Data Preview' sections are the same, displaying the table structure and a sample of 11 rows.

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	nj	104	134
7	68553	June	10021	ca	117	67
8	68109	June	10022	bx	337	10
9	68526	June	10025	mi	11	63
10	69362	June	10028	bx	430	65
11	68624	June	10030	fl	808	51

© 2015 MapR Technologies 32





Define Data Interaction & Schema Detection

The screenshot shows the Drill Explorer interface. On the left, there's a tree view of schemas: cp.default, dfs.clicks, dfs.data, dfs.default, dfs.logs (with logs for 2012, 2013, 2014), dfs.root, dfs.tmp, dfs.views, hive.default (with orders), maprdb (with customers, address, loyalty, personal, embeddedclicks, products). In the center, a table titled 'Metadata: hive.default.orders' is displayed with columns: COLUMN_NAME, DATA_TYPE, and IS_NULLABLE. The rows show the schema for the 'orders' table: order_id (BIGINT, YES), month (VARCHAR, YES), cust_id (BIGINT, YES), state (VARCHAR, YES), prod_id (BIGINT, YES). Below it is a 'Data Preview' table with 11 rows of sample data. The first row of the preview table is highlighted.

COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1 order_id	BIGINT	YES
2 month	VARCHAR	YES
3 cust_id	BIGINT	YES
4 state	VARCHAR	YES
5 prod_id	BIGINT	YES

order_id	month	cust_id	state	prod_id	order_total
1 67212	June	10001	ca	909	13
2 70302	June	10004	ga	420	11
3 69090	June	10011	fl	44	76
4 68834	June	10012	ar	0	81
5 71220	June	10018	az	411	24
6 61287	June	1001	nj	104	134
7 68553	June	10021	ca	117	67
8 68109	June	10022	bx	337	10
9 68526	June	10025	mi	11	63
10 69362	June	10028	bx	430	65
11 68624	June	10030	fl	808	51

© 2015 MapR Technologies 33



Define Data Interaction & Schema Detection

The screenshot shows the Drill Explorer interface. On the left, there's a tree view of schemas: cp.default, dfs.clicks, dfs.data, dfs.default, dfs.logs (with logs for 2012, 2013, 2014), dfs.root, dfs.tmp, dfs.views, hive.default (with orders), maprdb (with customers, address, loyalty, personal, embeddedclicks, products). In the center, a table titled 'Metadata: hive.default.orders' is displayed with columns: COLUMN_NAME, DATA_TYPE, and IS_NULLABLE. The rows show the schema for the 'orders' table: order_id (BIGINT, YES), month (VARCHAR, YES), cust_id (BIGINT, YES), state (VARCHAR, YES), prod_id (BIGINT, YES). Below it is a 'Data Preview' table with 11 rows of sample data. The first row of the preview table is highlighted.

COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1 order_id	BIGINT	YES
2 month	VARCHAR	YES
3 cust_id	BIGINT	YES
4 state	VARCHAR	YES
5 prod_id	BIGINT	YES

order_id	month	cust_id	state	prod_id	order_total
1 67212	June	10001	ca	909	13
2 70302	June	10004	ga	420	11
3 69090	June	10011	fl	44	76
4 68834	June	10012	ar	0	81
5 71220	June	10018	az	411	24
6 61287	June	1001	nj	104	134
7 68553	June	10021	ca	117	67
8 68109	June	10022	bx	337	10
9 68526	June	10025	mi	11	63
10 69362	June	10028	bx	430	65
11 68624	June	10030	fl	808	51

© 2015 MapR Technologies 34





Define Data Interaction & Schema Detection

cust_id	name	state	gender	age	agg_rev	membership	reviews
16841	Lisa Wells	tx	FEMALE	26-35	40	basic	NULL
19905	Neely Nova	va	MALE	15-20	45	basic	NULL
17718	William Bush	il	FEMALE	51-100	29	basic	NULL
21975	Gerald Staples	ma	MALE	26-35	28	basic	TiqA9ybgewk

```
{
  "trans_id": 31929, "date": "2014-04-26", "time": "12:17:12", "user_info": {"cust_id": 22526, "device": "I056", "state": "il"}, "trans_info": {"prod_id": [174, 2], "purch_flag": "false"}},
  {"trans_id": 31826, "date": "2014-04-26", "time": "13:58:29", "user_info": {"cust_id": 16368, "device": "A054, 2", "state": "nc"}, "trans_info": {"prod_id": [1], "purch_flag": "false"}},
  {"trans_id": 33848, "date": "2014-04-18", "time": "184:44:42", "user_info": {"cust_id": 21449, "device": "I056", "state": "oh"}, "trans_info": {"prod_id": [582], "purch_flag": "false"}},
  {"trans_id": 32383, "date": "2014-04-18", "time": "06:27:47", "user_info": {"cust_id": 20233, "device": "I055", "state": "oh"}, "trans_info": {"prod_id": [710, 47], "purch_flag": "false"}},
  {"trans_id": 32359, "date": "2014-04-19", "time": "23:13:25", "user_info": {"cust_id": 15368, "device": "I055", "state": "ca"}, "trans_info": {"prod_id": [8, 8, 170, 173, 1, 124, 46, 764, 30, 711, 0, 3, 25], "purch_flag": "true"}},
  {"trans_id": 32358, "date": "2014-04-19", "time": "07:01:58", "user_info": {"cust_id": 20233, "device": "I055", "state": "ca"}, "trans_info": {"prod_id": [8, 8, 170, 173, 1, 124, 46, 764, 30, 711, 0, 3, 25], "purch_flag": "true"}},
  {"trans_id": 35899, "date": "2014-04-18", "time": "180:55:42", "user_info": {"cust_id": 20233, "device": "I055", "state": "ca"}, "trans_info": {"prod_id": [8, 8, 170, 173, 1, 124, 46, 764, 30, 711, 0, 3, 25], "purch_flag": "true"}},
  {"trans_id": 32423, "date": "2014-04-15", "time": "11:00:53", "user_info": {"cust_id": 23599, "device": "I055", "state": "il"}, "trans_info": {"prod_id": [1, 0, 87, 178, 445, 6, 8, 2, 7, 7, 110], "purch_flag": "false"}},
  {"trans_id": 39447, "date": "2014-04-21", "time": "01:34:06", "user_info": {"cust_id": 16122, "device": "I056", "state": "il"}, "trans_info": {"prod_id": [26, 504, 1, 47, 31, 116, 167, 71, 20, 305, 554, 0, 189, 384, 1, 71, 4, 934], "purch_flag": "false"}},
  {"trans_id": 35373, "date": "2014-04-02", "time": "06:53:28", "user_info": {"cust_id": 15342, "device": "I055", "state": "es"}, "trans_info": {"prod_id": [15, 0], "purch_flag": "false"}},
  {"trans_id": 38085, "date": "2014-04-13", "time": "08:08:05", "user_info": {"cust_id": 16992, "device": "I055", "state": "oh"}, "trans_info": {"prod_id": [149, 0, 15], "purch_flag": "true"}},
  {"trans_id": 31238, "date": "2014-04-09", "time": "180:49:58", "user_info": {"cust_id": 15344, "device": "I057", "state": "ny"}, "trans_info": {"prod_id": [1, 0, 44, 171, 173, 1, 124, 46, 764, 30, 711, 0, 3, 25], "purch_flag": "true"}},
  {"trans_id": 38706, "date": "2014-04-15", "time": "12:00:15", "user_info": {"cust_id": 15344, "device": "I057", "state": "ny"}, "trans_info": {"prod_id": [1, 0, 44, 171, 173, 1, 124, 46, 764, 30, 711, 0, 3, 25], "purch_flag": "true"}},
  {"trans_id": 38287, "date": "2014-04-28", "time": "18:57:55", "user_info": {"cust_id": 16841, "device": "A054, 2", "state": "mp"}, "trans_info": {"prod_id": [181, 0, 33, 768, 92], "purch_flag": "false"}},
  {"trans_id": 32955, "date": "2014-04-15", "time": "12:41:58", "user_info": {"cust_id": 15492, "device": "A054, 4", "state": "wl"}, "trans_info": {"prod_id": [242, 0, 2, 3, 166, 26, 0, 0], "purch_flag": "false"}}
}
```

© 2015 MapR Technologies 35



Define Data Interaction & Schema Detection

cust_id	name	state	gender	age	agg_rev	membership
16841	Lisa Wells	tx	FEMALE	26-35	40	basic
19905	Neely Nova	va	MALE	15-20	45	basic
17718	William Bush	il	FEMALE	51-100	29	basic
21975	Gerald Staples	ma	MALE	26-35	28	basic

Data Preview:

	column_0	column_1	column_2	column_3	column_4	column_5
1	16841	"Lisa Wells"	"tx"	"FEMALE"	"26-35"	40
2	19905	"Neely Nova"	"va"	"MALE"	"15-20"	45
3	17718	"William Bush"	"il"	"FEMALE"	"51-100"	29
4	21975	"Gerald Staples"	"ma"	"MALE"	"26-35"	28
5	20468	"William Conners"	"de"	"MALE"	"36-50"	32
6	16670	"John Rubio"	"oh"	"MALE"	"15-20"	42

© 2015 MapR Technologies 36



Define Data Interaction & Schema Detection

```
["trans_id":31920,"date":"2014-04-26","time":"12:17:12","user_info":{"cust_id":22526}, {"trans_id":31026,"date":"2014-04-20","time":"13:50:29","user_info":{"cust_id":16368}, {"trans_id":33848,"date":"2014-04-10","time":"04:44:42","user_info":{"cust_id":21449}, {"trans_id":32383,"date":"2014-04-18","time":"06:27:47","user_info":{"cust_id":20323}, {"trans_id":32359,"date":"2014-04-19","time":"23:13:25","user_info":{"cust_id":15360}, {"trans_id":30422,"date":"2014-04-23","time":"01:46:05","user_info":{"cust_id":15957}, {"trans_id":35898,"date":"2014-04-18","time":"13:28:56","user_info":{"cust_id":20677}, {"trans_id":32421,"date":"2014-04-15","time":"11:00:53","user_info":{"cust_id":23599},
```

Data Description:"cust_id"
Data Content: :15360

© 2015 MapR Technologies 37

Define Data Interaction & Schema Detection

Drill Explorer

	Column_Index	Data_Type	Value_Width
1	0	VARCHAR(5)	5
2	1	VARCHAR(22)	22
3	2	VARCHAR(4)	4
4	3	VARCHAR(8)	8
5	4	VARCHAR(8)	8
6	5	VARCHAR(2)	2

Metadata:

Schemas:

- cp default
- dfs clicks
- dfs data
- dfs default
- dfs logs
- dfs root
- apps
- data
- flat
- nested
- orders
- products
- customers all csv
- drill-beta-demo
- hbase
- oozie
- tables
- tmp
- user
- var
- dfs tmp
- dfs views
- hive default
- orders
- maprdb
- customers
- embeddedclicks

Data Preview:

	column_0	column_1	column_2	column_3	column_4	column_5
1	16841	"Lisa Wells"	"tk"	"FEMALE"	"26-35"	40
2	19905	"Neely Nova"	"va"	"MALE"	"15-20"	45
3	17718	"William Bush"	"il"	"FEMALE"	"51-100"	29
4	21975	"Gerald Staples"	"ma"	"MALE"	"26-35"	28
5	20468	"William Conners"	"de"	"MALE"	"36-50"	32
6	16670	"John Rubio"	"oh"	"MALE"	"15-20"	42
7	18600	"Jay Holland"	"f"	"FEMALE"	"21-25"	43
8	23438	"James Hogsett"	"ne"	"FEMALE"	"15-20"	30
9	20669	"Florence Black"	"pa"	"FEMALE"	"26-35"	24
10	22880	"Judith Stuart"	"ga"	"MALE"	"21-25"	25
11	18467	"Dale Johnson"	"ga"	"FEMALE"	"15-20"	24
12	19828	"Patrick Cote"	"ms"	"MALE"	"36-50"	45

Refresh Connect... Sample 100 Rows Close v0.05.0.0593 (64 bit)

© 2015 MapR Technologies 38





Knowledge Check

Match the following definitions

- A. `hive.orders`
- B. `Dfs.'data/nested/clicks/clicks.json'`
- C. `HCATALOG_COLUMNS`
- D. `"user_info":{"cust_id":5324,"device":"AOS4.3","state":"ca"}`

- Tells Drill what table to query
- Metadata used to discover the schema of a Hive table
- Tells Drill what file to query
- Self describing JSON data



Complex Query Examples

Simple
Complex

Fixed Schema

- Structured Tables:**
- Hive
 - CSV
 - TSV

Dynamic Schema

- Semi-Structured Table :**
- Hbase
 - MapR-DB

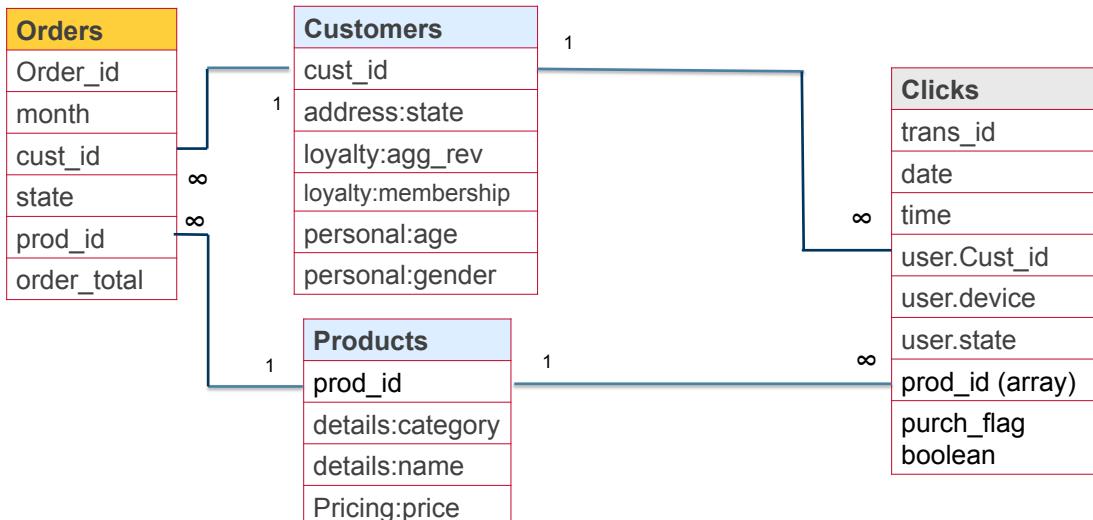
- Structured File:**
- Parquet

- Semi-Structured File:**
- JSON
 - MongoDB





Complex Query Examples



© 2015 MapR Technologies 41



Complex Query Examples

```

SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date` FROM
hive.products as prod,
(SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id,
clk.`date`
FROM maprdb.customers as cust, (SELECT cast(clicks.user_info.cust_id as bigint)
as cust_id, cast(FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id,
clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd')
as `date` FROM dfs.clicks.`/clicks/clicks.json` as clicks) as clk
WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false')
as cust_clk
WHERE cust_clk.prod_id = prod.prod_id
and cust_clk.`date` between '2014-01-01' and '2014-03-31';
  
```

© 2015 MapR Technologies 42



Complex Query Examples

The screenshot displays three separate Drill Explorer windows side-by-side, each showing a complex query interface:

- Hive:** Shows a schema browser with tables like `customers`, `orders`, and `products`. A data preview window shows a table with columns `order_id`, `month`, `cust_id`, `state`, and `prod_id`.
- HBase:** Shows a schema browser with tables like `customers` and `orders`. A data preview window shows a table with columns `column_0` through `column_4`.
- JSON:** Shows a schema browser with tables like `customers` and `orders`. A data preview window shows a table with columns `trans_id`, `date`, `time`, `cust_id`, `device`, `state`, `camp_id`, and `keyword`.

© 2015 MapR Technologies 43

Complex Query Examples

```

SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date` FROM
hive.products as prod,
(SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id,
clk.`date`
FROM maprdb.customers as cust, (SELECT cast(clicks.user_info.cust_id as bigint)
as cust_id, cast(FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id,
clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd')
as `date` FROM dfs.clicks.`/clicks/json` as clicks) as clk
WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false')
as cust_clk
WHERE cust_clk.prod_id = prod.prod_id
and cust_clk.`date` between '2014-01-01' and '2014-03-31';

```

© 2015 MapR Technologies 44



Complex Query Examples

```
SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date`  
FROM hive.products as prod
```

The screenshot shows the MapR UI interface. On the left, there is a 'Schemas' tree view containing various HDFS and MapR databases like cp.default, dfs.clicks, dfs.logs, and maprdb. A red arrow points from the 'products' table in the schema tree to the 'Metadata' and 'Data Preview' sections on the right.

Metadata: hive default.'orders'

COLUMN_NAME	DATA_TYPE	IS_NULLABLE
order_id	BIGINT	YES
month	VARCHAR	YES
cust_id	BIGINT	YES
state	VARCHAR	YES
prod_id	BIGINT	YES

Data Preview:

order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909
2	70302	June	10004	ga	420
3	69090	June	10011	fl	44
4	68834	June	10012	ar	0
5	71220	June	10018	az	411
6	61287	June	1001	nj	104
7	68553	June	10021	ca	117
8	68109	June	10022	tx	337
9	68526	June	10025	mi	11
10	69362	June	10028	tx	430

© 2015 MapR Technologies **MAPR** 45

Complex Query Examples

```
SELECT cast(cust.personal.name as varchar(40)) as cust_name,  
clk.prod_id, clk.`date`  
FROM maprdb.customers as cust
```

The screenshot shows the MapR UI interface. On the left, there is a 'Schemas' tree view containing various HDFS and MapR databases like cp.default, dfs.clicks, and maprdb. A red arrow points from the 'customers' table in the schema tree to the 'Metadata' and 'Data Preview' sections on the right.

Metadata: 'maprdb'.'customers'

COLUMN_NAME	DATA_TYPE
row_key	ANY
address	(VARCHAR(1), ANY) MAP
loyalty	(VARCHAR(1), ANY) MAP
personal	(VARCHAR(1), ANY) MAP

Data Preview:

row_key	address	loyalty	personal
0x3130303031	{ "state": "In2hlg=="} { "agg_rev": "MTk3", "membership": "InNpbHZicl=="} { "age": "i E1LTlwlg==", "gr		
2	{ "state": "Imulg=="} { "agg_rev": "MjMw", "membership": "InNpbHZicl=="} { "age": "i j2LTM1tg==", "gr		
3	{ "state": "ImNhlg=="} { "agg_rev": "MjUw", "membership": "InNpbHZicl=="} { "age": "i j2LTM1tg==", "gr		
4	{ "state": "Im1llg=="} { "agg_rev": "MjYz", "membership": "InNpbHZicl=="} { "age": "i jULTEwMCi=", "gr		
5	{ "state": "Imf1ulg=="} { "agg_rev": "MjA", "membership": "InNpbHZicl=="} { "age": "i jULTEwMCi=", "gr		
6	{ "state": "Imhplg=="} { "agg_rev": "MTV5", "membership": "InNpbHZicl=="} { "age": "i jULTEwMCi=", "gr		
7	{ "state": "Im80lg=="} { "agg_rev": "MTU1", "membership": "InNpbHZicl=="} { "age": "i jLT1Ilg==", "gr		
8	{ "state": "Im5qlg=="} { "agg_rev": "MTk5", "membership": "InNpbHZicl=="} { "age": "i jELTlwlg==", "gr		
9	{ "state": "Im55lg=="} { "agg_rev": "MjQ4", "membership": "InNpbHZicl=="} { "age": "i jULTEwMCi=", "gr		
10	{ "state": "ImN0lg=="} { "agg_rev": "MTA5MQ==", "membership": "ImdvbGQi"} { "age": "i jLT1Ilg==", "gr		

© 2015 MapR Technologies **MAPR** 46



Complex Query Examples

```
SELECT cast(clicks.user_info.cust_id as bigint) as cust_id, cast(FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id, clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd') as `date` FROM dfs.clicks.`/clicks/clicks.json` as clicks
```

The screenshot shows the MapR SQL interface. On the left, the 'Schemas' tree view shows the 'clicks' schema containing 'clicks.campaign.json' and 'clicks.json'. On the right, the 'Metadata' panel displays the 'hive.default.orders' table with one column: 'COLUMN_NAME' (with value 'Metadata does not exist'). Below it is a 'Data Preview' section.

© 2015 MapR Technologies 47

Complex Query Examples

```
cast(cust.personal.name as varchar(40))
```

The screenshot shows the MapR SQL interface. On the left, the 'Schemas' tree view shows the 'maprdb' schema containing 'customers'. On the right, the 'Metadata' panel displays the 'maprdb.customers' table with four columns: 'COLUMN_NAME' (row_key, address, loyalty, personal), 'DATA_TYPE' (ANY, (VARCHAR(1), ANY) MAP, (VARCHAR(1), ANY) MAP, (VARCHAR(1), ANY) MAP), and 'IS_NULLABLE' (NO, NO, NO, NO). Red arrows point from the 'personal' column in the code snippet to the 'personal' column in the table metadata.

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
▶ 1	row_key	ANY	NO
2	address	(VARCHAR(1), ANY) MAP	NO
3	loyalty	(VARCHAR(1), ANY) MAP	NO
4	personal	(VARCHAR(1), ANY) MAP	NO

© 2015 MapR Technologies 48





Complex Query Examples

`cast(clicks.user_info.cust_id as bigint)`

Metadata: hive.default.orders

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1	order_id	BIGINT	YES
2	month	VARCHAR	YES
3	cust_id	BIGINT	YES
4	state	VARCHAR	YES
5	prod_id	BIGINT	YES

Data Preview:

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	ni	104	124

© 2015 MapR Technologies  49



Complex Query Examples

`cast(FLATTEN(clicks.trans_info.prod_id) as bigint)`

Metadata: hive.default.orders

	COLUMN_NAME	DATA_TYPE	IS_NULLABLE
1	order_id	BIGINT	YES
2	month	VARCHAR	YES
3	cust_id	BIGINT	YES
4	state	VARCHAR	YES
5	prod_id	BIGINT	YES

Data Preview:

	order_id	month	cust_id	state	prod_id	order_total
1	67212	June	10001	ca	909	13
2	70302	June	10004	ga	420	11
3	69090	June	10011	fl	44	76
4	68834	June	10012	ar	0	81
5	71220	June	10018	az	411	24
6	61287	June	1001	ni	104	124

© 2015 MapR Technologies  50





Complex Query Examples

```
(FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id,
clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd') as `date` FROM dfs.clicks.`/clicks@clicks.json` as clicks) as clk
```

	trans_info
:22526	"device": "IOS5", "state": "il"} { "prod_id": [174, 2], "purch_flag": "false"}
:16368	"device": "AOS42", "state": "nc"} { "prod_id": [], "purch_flag": "false"}
:21449	"device": "IOS6", "state": "oh"} { "prod_id": [582], "purch_flag": "false"}
:20323	"device": "IOS5", "state": "oh"} { "prod_id": [710, 47], "purch_flag": "false"}
:15360	"device": "IOS5", "state": "ca"} { "prod_id": [0, 8, 170, 173, 1, 124, 46, 764, 30, 711, 0, 3, 25], "purch_flag": "false"}
:15957	"device": "IOS7", "state": "sc"} { "prod_id": [628], "purch_flag": "false"}
:20677	"device": "IOS7", "state": "ny"} { "prod_id": [], "purch_flag": "false"}
:23599	"device": "IOS5", "state": "ri"} { "prod_id": [1, 0, 87, 170, 445, 6, 0, 2, 7, 7, 110], "purch_flag": "false"}
:16122	"device": "IOS5", "state": "fl"} { "prod_id": [26, 504, 1, 47, 31, 116, 167, 71, 20, 305, 554, 0, 189, 384], "purch_flag": "false"}
:15342	"device": "IOS5", "state": "ms"} { "prod_id": [15, 0], "purch_flag": "false"}
:16996	"device": "AOS42", "state": "oh"} { "prod_id": [40, 499, 15], "purch_flag": "true"}
:21402	"device": "IOS5", "state": "nm"} { "prod_id": [], "purch_flag": "false"}
:15344	"device": "IOS7", "state": "ny"} { "prod_id": [27], "purch_flag": "false"}
:16841	"device": "AOS42", "state": "mo"} { "prod_id": [181, 0, 33, 708, 92], "purch_flag": "false"}
:15492	"device": "AOS42", "state": "mi"} { "prod_id": [242, 0, 2, 3, 166, 26, 0, 1], "purch_flag": "false"}

© 2015 MapR Technologies 51



Complex Query Examples

```
SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date` FROM
hive.products as prod,
(SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id, clk.`date` FROM maprdb.customers as cust, (SELECT cast(clicks.user_info.cust_id as bigint) as cust_id,
cast(FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id,
clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd') as `date`
FROM dfs.clicks.`/clicks@clicks.json` as clicks) as clk
WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false') as cust_clk
WHERE cust_clk.prod_id = prod.prod_id
and cust_clk.`date` between '2014-01-01' and '2014-03-31';
```

© 2015 MapR Technologies 52





Complex Query Examples

```
SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date`  
FROM hive.products as prod,  
WHERE cust_clk.prod_id = prod.prod_id  
and cust_clk.`date` between '2014-01-01' and '2014-03-31';
```

Query

```
SELECT cast(clicks.user_info.cust_id as bigint) as cust_id,  
cast(FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id, clicks.trans_info.purch_flag as  
purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd') as `date`  
FROM dfs.clicks.`/clicks/clicks.json` as clicks
```

The screenshot shows the Apache Drill interface with the URL 127.0.0.1:8047/query. The results of the query are displayed in a table:

prod_id	date	cust_id	purch_flag
174	2014-01-26T00:00:00.000-08:00	22,526	false
2	2014-01-26T00:00:00.000-08:00	22,526	false
582	2014-01-10T00:00:00.000-08:00	21,449	false
700	2014-01-10T00:00:00.000-08:00	22,526	false

© 2015 MapR Technologies 53



Complex Query Examples

```
(SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id, clk.`date`  
FROM maprdb.customers as cust,  
WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false') as  
cust_clk
```

Query

```
select cast(cust.personal.name as varchar(40)) as cust_name  
from maprdb.customers as cust, dfs.clicks.`/clicks/clicks.json` as clk  
where cast(cust.row_key as bigint) = cast(clk.user_info.cust_id as bigint)  
and cast(clk.trans_info.purch_flag as varchar(20)) = 'false' and  
clk.trans_info.prod_id[0] is not null limit 10
```

The screenshot shows the Apache Drill interface with the URL 127.0.0.1:8047/query. The results of the query are displayed in a table:

Show: 10 : entries	Search:	Show / hide columns
cust_name		
'Corrine Mecham'		
'James Fowler'		
'Thomas Massie'		

© 2015 MapR Technologies 54



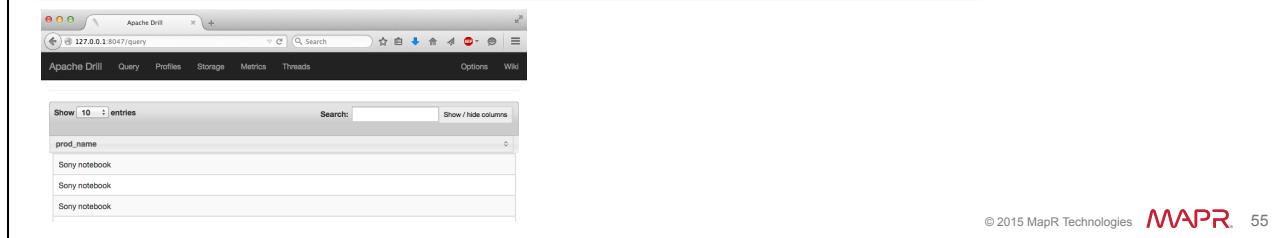


Complex Query Examples

```
SELECT cast(clicks.user_info.cust_id as bigint) as cust_id,
       cast(FLATTEN(clicks.trans_info.prod_id) as bigint) as prod_id,
       clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`, 'yyyy-mm-dd') as `date`
  FROM dfs.clicks.`/clicks/clicks.json` as clicks as clk
```

Query

```
select prod.name as prod_name
  from hive.products as prod, dfs.clicks.`/clicks/clicks.json` as clk
 where cast(clk.trans_info.prod_id[0] as bigint) = prod.prod_id and cast(clk.trans_info.purch_flag
 as varchar(20)) = 'false' and clk.trans_info.prod_id[0] is not null limit 10
```



The screenshot shows the Apache Drill interface running on port 8047. The query results are displayed in a table with one column labeled 'prod_name'. The results show three rows: 'Sony notebook', 'Sony notebook', and 'Sony notebook'. The interface includes a top navigation bar with tabs for Apache Drill, Query, Profiles, Storage, Metrics, Threads, Options, and Wiki. Below the table, there is a search bar and a 'Show / hide columns' button.

© 2015 MapR Technologies  55



Knowledge Check

Drill can join which of the following types into a single query?:

- Structured table data
- Semi structured table data
- Structured text files
- Semi structured text files
- Any combination of these data types

© 2015 MapR Technologies  56



Views

Drill Explorer

Browse SQL

View Definition SQL:

```
SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date` FROM
hive.products as prod,
(SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id,
clk.`date` FROM maprdb.customers as cust, (SELECT cast(clicks.user_info.cust_id
as bigint) as cust_id, cast(FLATTEN(clicks.trans_info.prod_id) as bigint)
as prod_id, clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`,
'yyyy-mm-dd') as `date` FROM dfs.clicks.`/clicks/clicks.json` as clicks) as clk
WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false')
as cust_clk WHERE cust_clk.prod_id = prod.prod_id
and cust_clk.`date` between '2014-01-01' and '2014-03-31';
```

Total Number of Records: 47,629

	cust_name	date	prod_name
1	"Jerry Rackley"	2014-01-29T00:00:00.000-08:00	"Sony VAIO Duo 13"
2	"Francis Diaz"	2014-01-29T00:00:00.000-08:00	"Sony VAIO Duo 13"
3	"Lucio Goods"	2014-01-05T00:00:00.000-08:00	"Sony VAIO Duo 13"
4	"Thelma Mcnair"	2014-01-04T00:00:00.000-08:00	"HP Spectre 13 X2"
5	"Kayla Jones"	2014-01-22T00:00:00.000-08:00	"HP Spectre 13 X2"
6	"Carmen Polk"	2014-01-30T00:00:00.000-08:00	"Lenovo ThinkPad S431 Ultrabook"
7	"Regina Moriarty"	2014-01-03T00:00:00.000-08:00	"Lenovo ThinkPad S431 Ultrabook"
8	"Lisa Sprague"	2014-01-05T00:00:00.000-08:00	"Alienware 14"
9	"Tony Hendrix"	2014-01-30T00:00:00.000-08:00	"Alienware 14"
10	"Andre Jude"	2014-01-29T00:00:00.000-08:00	"Alienware 14"

© 2015 MapR Technologies 57

Views

Drill Explorer

Browse SQL

View Definition SQL:

```
SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date` FROM
hive.products as prod,
(SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id,
clk.`date` FROM maprdb.customers as cust, (SELECT cast(clicks.user_info.cust_id
as bigint) as cust_id, cast(FLATTEN(clicks.trans_info.prod_id) as bigint)
as prod_id, clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`,
'yyyy-mm-dd') as `date` FROM dfs.clicks.`/clicks/clicks.json` as clicks) as clk
WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false')
as cust_clk WHERE cust_clk.prod_id = prod.prod_id
and cust_clk.`date` between '2014-01-01' and '2014-03-31';
```

© 2015 MapR Technologies 58



 Views

Drill Explorer

Browse SQL

View Definition SQL:

```
SELECT cust_clk.cust_name, prod.name as prod_name, cust_clk.`date` FROM
hive.products as prod,
(SELECT cast(cust.personal.name as varchar(40)) as cust_name, clk.prod_id,
clk.`date` FROM maprdb.customers as cust, (SELECT cast(clicks.user_info.cust_id
as bigint) as cust_id, cast(FLATTEN(clicks.trans_info.prod_id) as bigint)
as prod_id, clicks.trans_info.purch_flag as purch_flag, to_date(clicks.`date`,
'yyyy-mm-dd') as `date` FROM dfs.clicks`./clicks.clicks.json` as clicks) as clk
WHERE cast(cust.row_key as bigint) = clk.cust_id and clk.purch_flag = 'false')
as cust_clk WHERE cust_clk.prod_id = prod.prod_id
and cust_clk.`date` between '2014-01-01' and '2014-03-31';
```

Total Number of Records: 47,629

	cust_name	date	prod_name
1	"Jerry Rackley"	2014-01-29T00:00:00.000-08:00	"Sony VAIO Duo 13"
2	"Francis Diaz"	2014-01-29T00:00:00.000-08:00	"Sony VAIO Duo 13"
3	"Lucio Goods"	2014-01-05T00:00:00.000-08:00	"Sony VAIO Duo 13"
4	"Thelma Mcnair"	2014-01-04T00:00:00.000-08:00	"HP Spectre 13 X2"
5	"Kayla Jones"	2014-01-22T00:00:00.000-08:00	"HP Spectre 13 X2"
6	"Carmen Polk"	2014-01-30T00:00:00.000-08:00	"Lenovo ThinkPad S431 Ultrabook"
7	"Regina Moriarty"	2014-01-03T00:00:00.000-08:00	"Lenovo ThinkPad S431 Ultrabook"
8	"Lisa Sprague"	2014-01-05T00:00:00.000-08:00	"Alienware 14"
9	"Tony Hendrix"	2014-01-30T00:00:00.000-08:00	"Alienware 14"
10	"Andre Jude"	2014-01-29T00:00:00.000-08:00	"Alienware 14"

Preview Create As...

© 2015 MapR Technologies  59

 Views

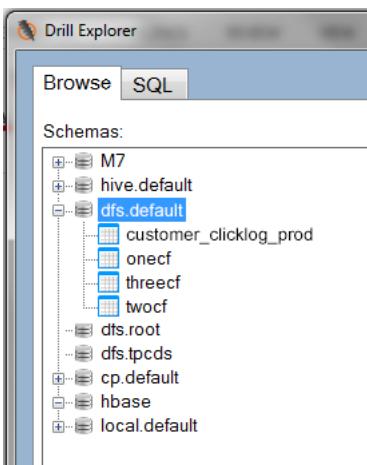
Create As...

Schema: **dfs.default** View Name: **customer_clicklog_prod**

Copy Save

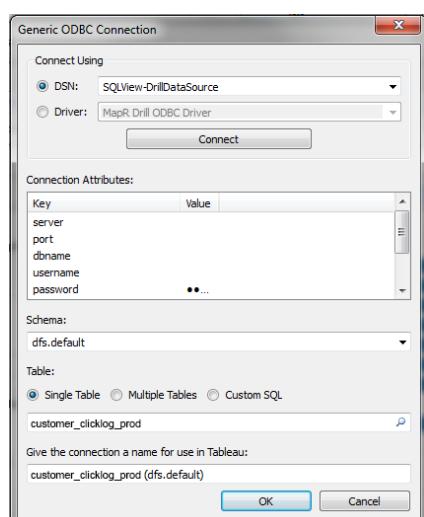
© 2015 MapR Technologies  60





Views

© 2015 MapR Technologies **MAPR** 61



Views

© 2015 MapR Technologies **MAPR** 62



Views

+ Tableau

MicroStrategy

SAS THE POWER TO KNOW.

© 2015 MapR Technologies 63

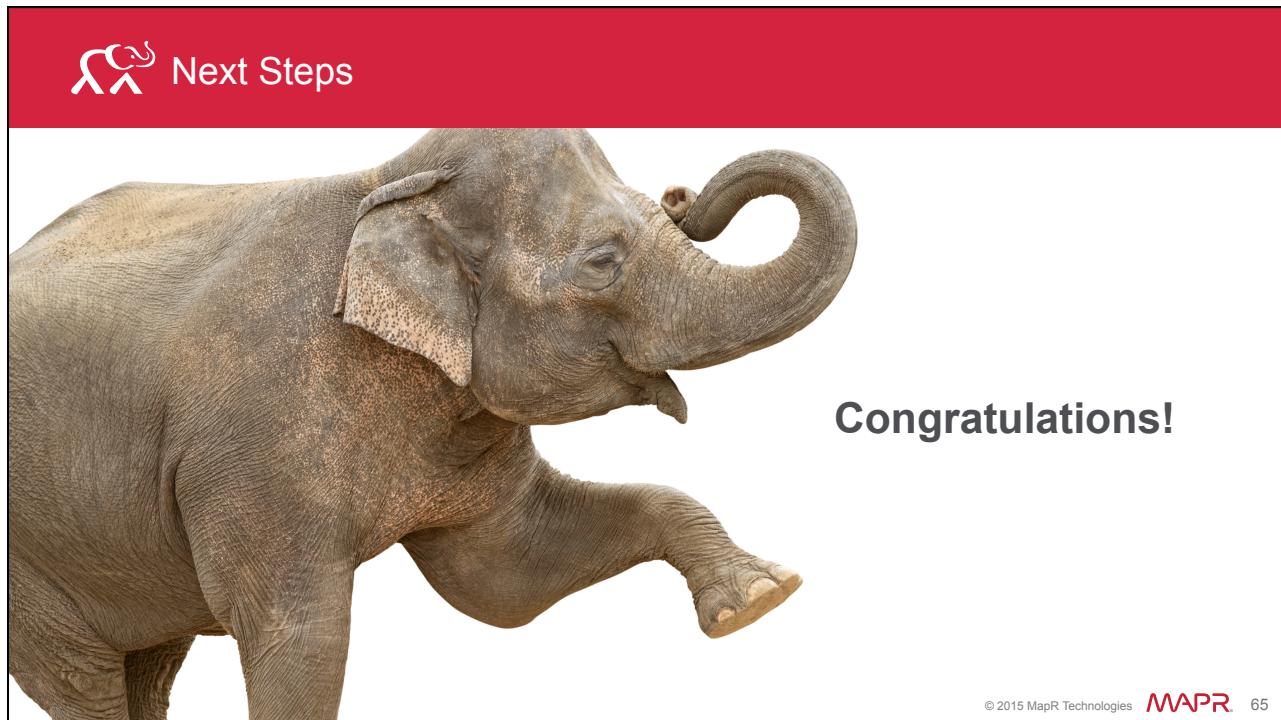
Knowledge Check

Drill Explorer views can be visualized with BI tools using:

- The ODBC interface
- The JDBC interface
- Either the ODBC or JDBC interface

© 2015 MapR Technologies 64





Next Steps

Congratulations!

© 2015 MapR Technologies **MAPR** 65



PROPRIETARY AND CONFIDENTIAL INFORMATION
©2015 MapR Technologies, Inc. All Rights Reserved