What is Vitess?

Cloud Native Database

Massively Scalable

HA

Based on MySQL

Vitess

CONGRATS

CLASS OF 2019

Vitess

Love, CNCF

Vitess Stats

Started
2010

Marquee
Adopters

100+
Contributors

8,000
+
Stars

17,000
+
Commits

1000+
Slack
Members

1000
+
Forks

Vitess

Key Adopters

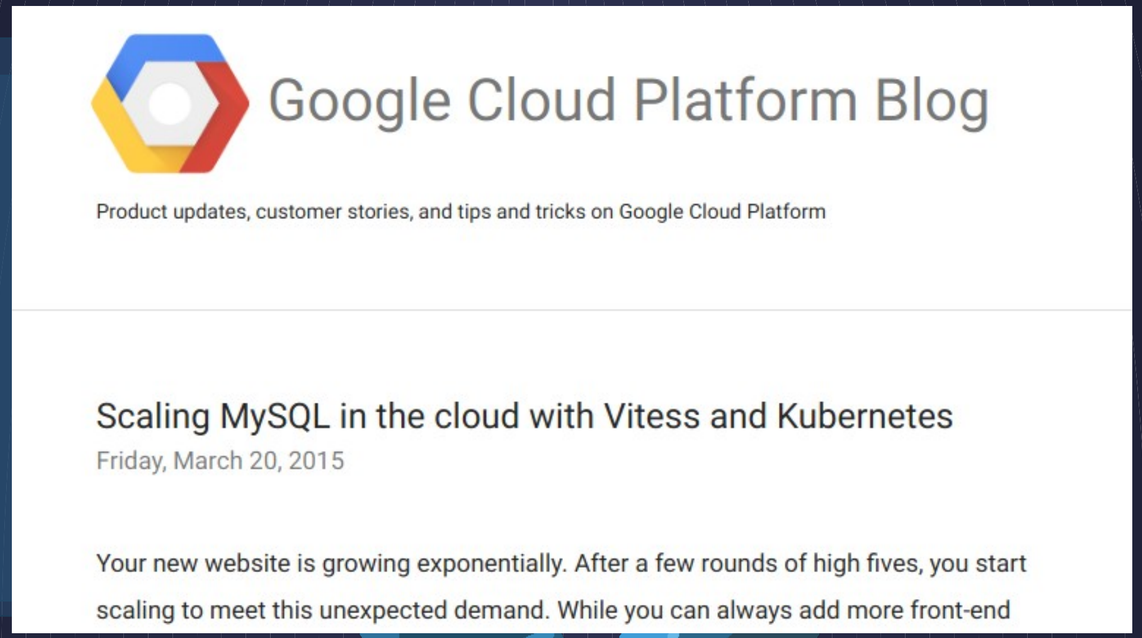
YouTube in 2010

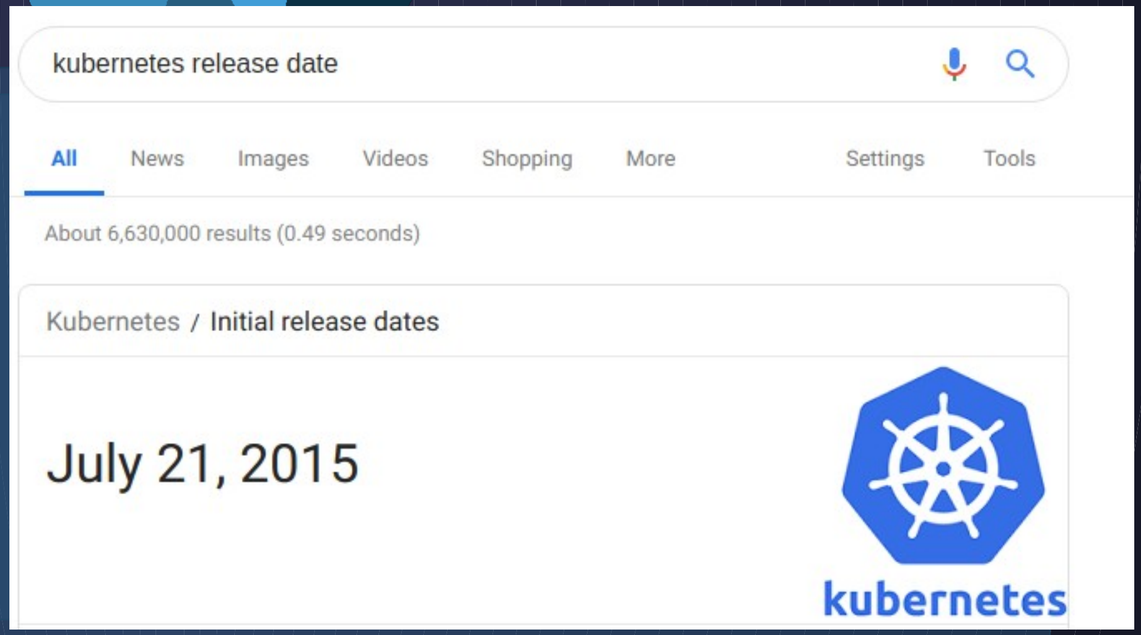
2013

Storage Engine running as Stateless Application

Stateless Storage

2015


Google Cloud Platform Blog
Product updates, customer stories, and tips and tricks on Google Cloud Platform
Scaling MySQL in the cloud with Vitess and Kubernetes
Friday, March 20, 2015
Your new website is growing exponentially. After a few rounds of high fives, you start scaling to meet this unexpected demand. While you can always add more front-end


kubernetes release date
All
News
Images
Videos
Shopping
More
Settings
Tools
About 6,630,000 results (0.49 seconds)
Kubernetes / Initial release dates
July 21, 2015
kubernetes

Vitess

Kubernetes Workloads

Oldest, since 2016

Hundreds of keyspaces

10,000+ Tablets

Migrate from cloud to cloud
MySQL 8.0

Earlier in 2019

Kelsey Hightower ✓
@kelseyhightower

I'm always going to recommend people exercise extreme caution when running stateful workloads on Kubernetes. Most people who are asking "can I run stateful workloads on Kubernetes" don't have much experience with Kubernetes and often times the workload they are asking about.

12:10 AM · Mar 24, 2019 · Twitter Web Client

283 Retweets **914** Likes

Kelsey Hightower ✓ @kelseyhightower · Mar 24
Replying to @kelseyhightower
Some people believe that rubbing Kubernetes on a stateful workload turns it into a fully managed database offering rivaling RDS. This is false. Maybe with enough effort, and additional components, and an SRE team, you can build RDS on top of Kubernetes.

18 149 494

Sugu Sougoumarane @ssougou · Mar 24
Replying to @kelseyhightower
@vitessio fills this gap. @HubSpot and @JD_Corporate have been running stateful workloads on it for over two years now. But it will be nice if Kubernetes put more thought into this problem area :).

2 1 10

Vitess

One does not simply…

move MySQL to Kubernetes…

Vitess

Later in 2019

Kelsey Hightower ✓
@kelseyhightower

Replying to @kelseyhightower @chrislovecnm and 2 others

I challenge anyone to believe that Kubernetes alone will deliver RDS like database management out of the box. If people believe that then I'm not sure they understand how RDS works.

3:53 PM · Oct 8, 2019 · Twitter Web App

2 Likes

Chris Love @chrislovecnm · Oct 8

@kelseyhightower u "recommend people exercise extreme caution when running stateful workloads on K8s". Which I agree with. You need to be an expert at Kafka and k8s to run it k8s. I think you in the past recommended not doing it at all? Just doing some fact-checking for a preso.

1

Kelsey Hightower ✓
@kelseyhightower

Replying to @chrislovecnm

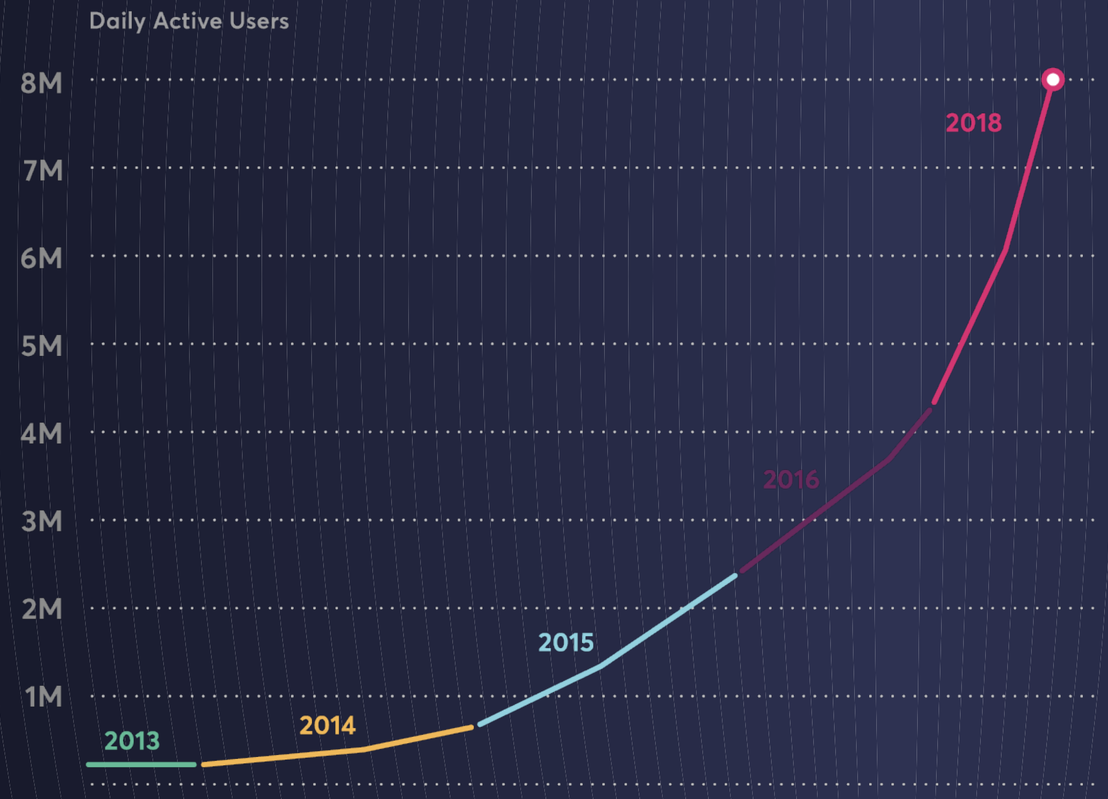
I can't suggest people never do it, but I stand by Kubernetes is not enough. You'll need tooling and help from the stateful system. @cockroachdb is a good example of a database that meets Kubernetes part way.

For others such as mysql, then something like @vitessio helps.

3:42 PM · Oct 8, 2019 · Twitter Web App

Vitess

Slack Growth



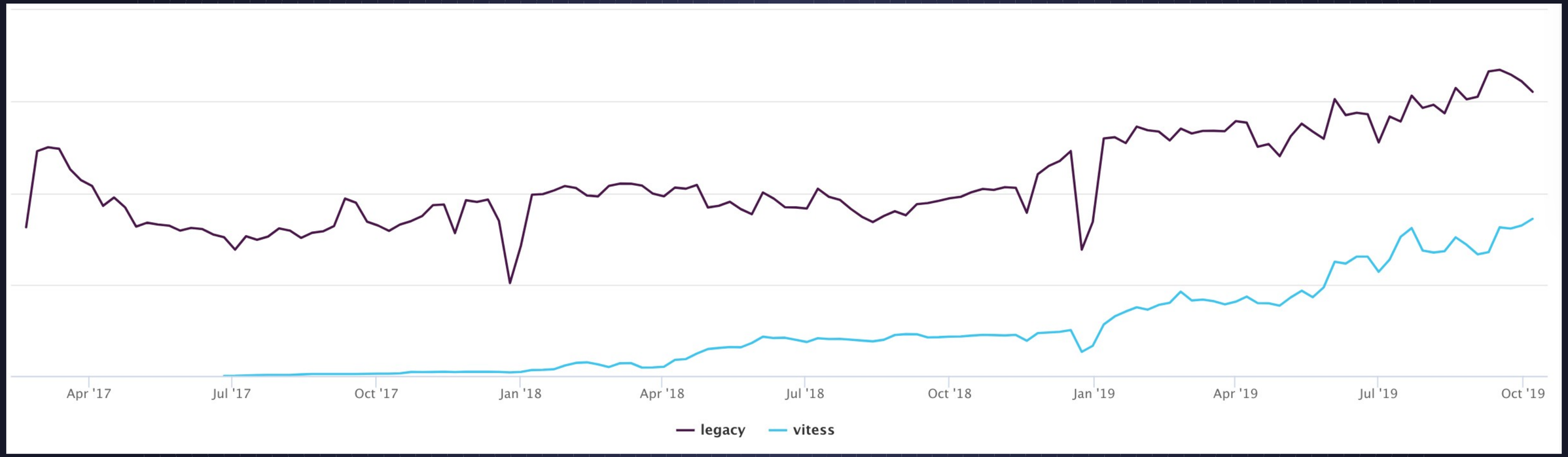
8M+ daily active users

4M+ of users outside of US

500K+ organizations use Slack in more than 100 countries

Vitess

Vitess @Slack



Vitess

Slack Vitess Usage

35%
Migrated

500K
Peak QPS

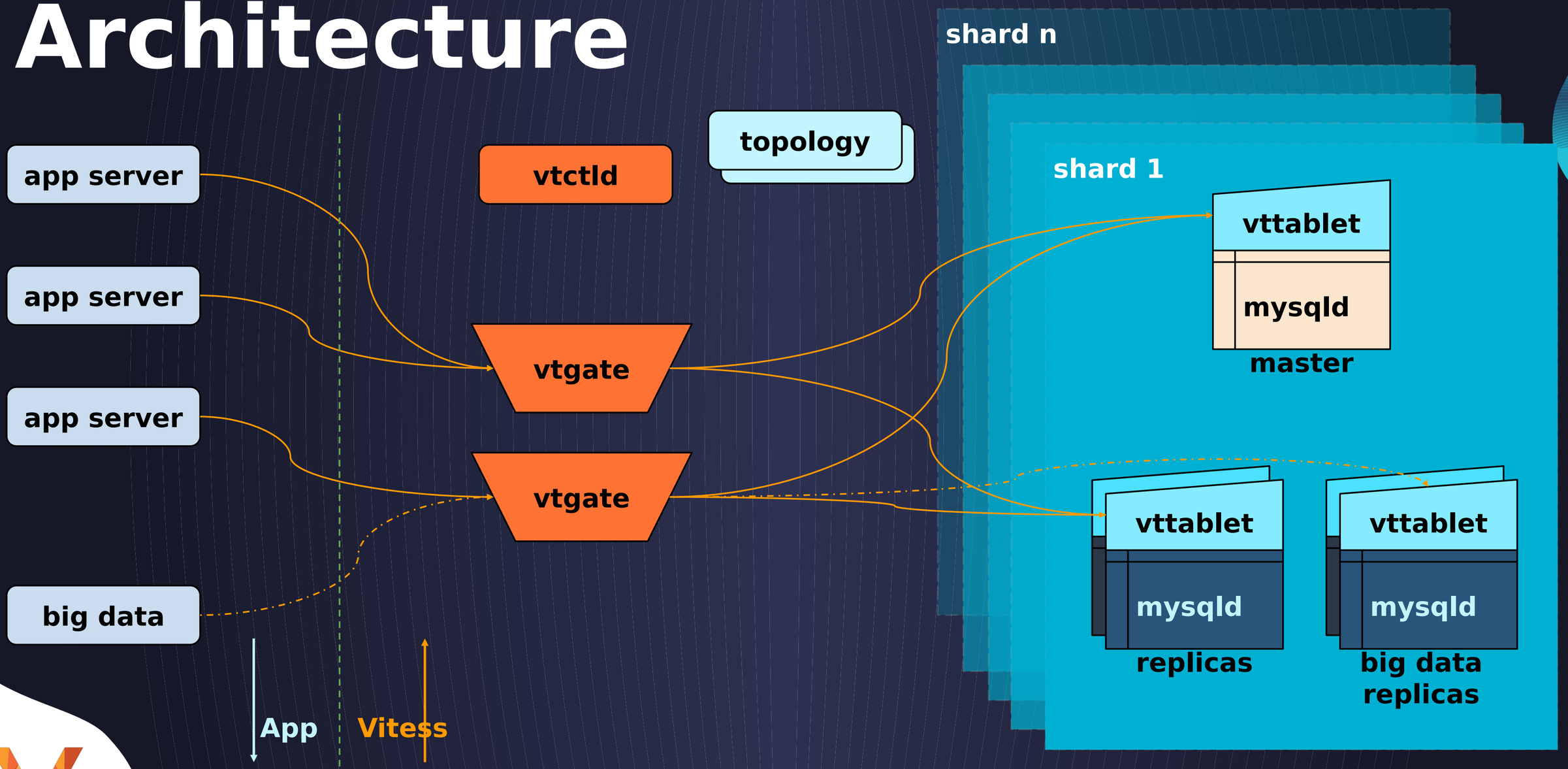
10B
Queries per day

1ms
Added latency

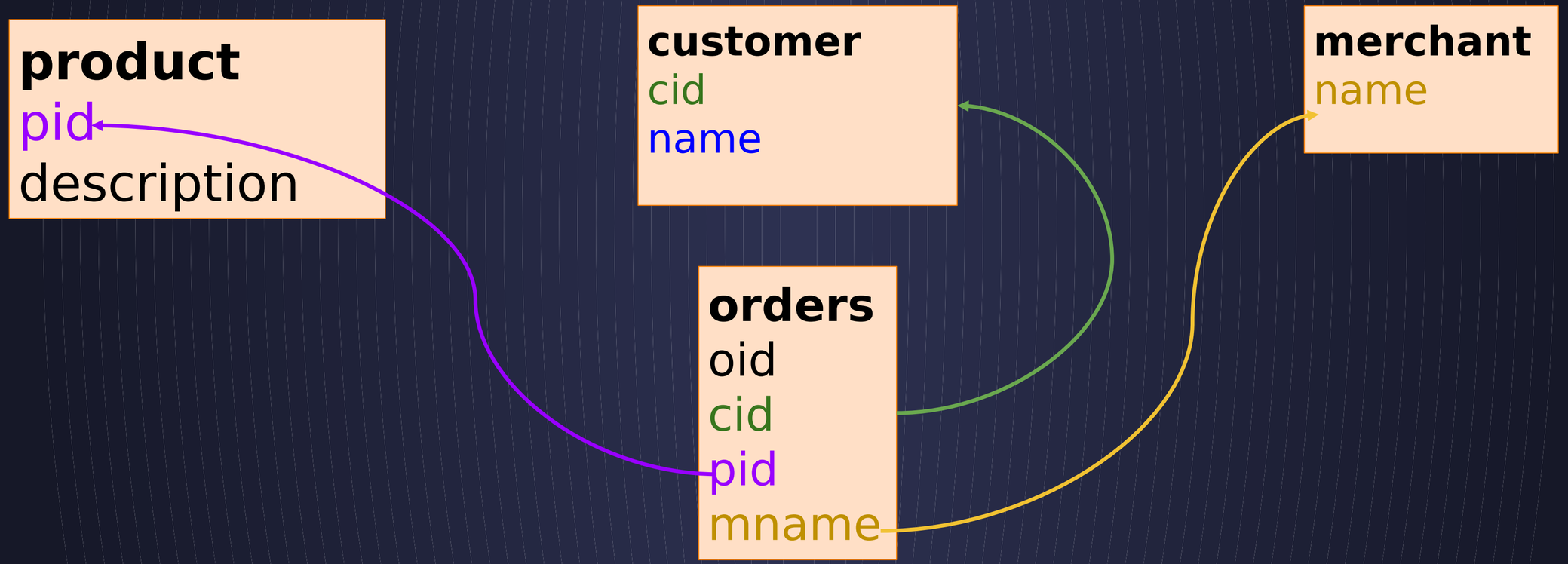
“Our goal is that all MySQL at Slack is run behind Vitess. There’s no other bet we’re making in terms of storage in the foreseeable future.”

Michael Demmer, Principal Engineer, Slack

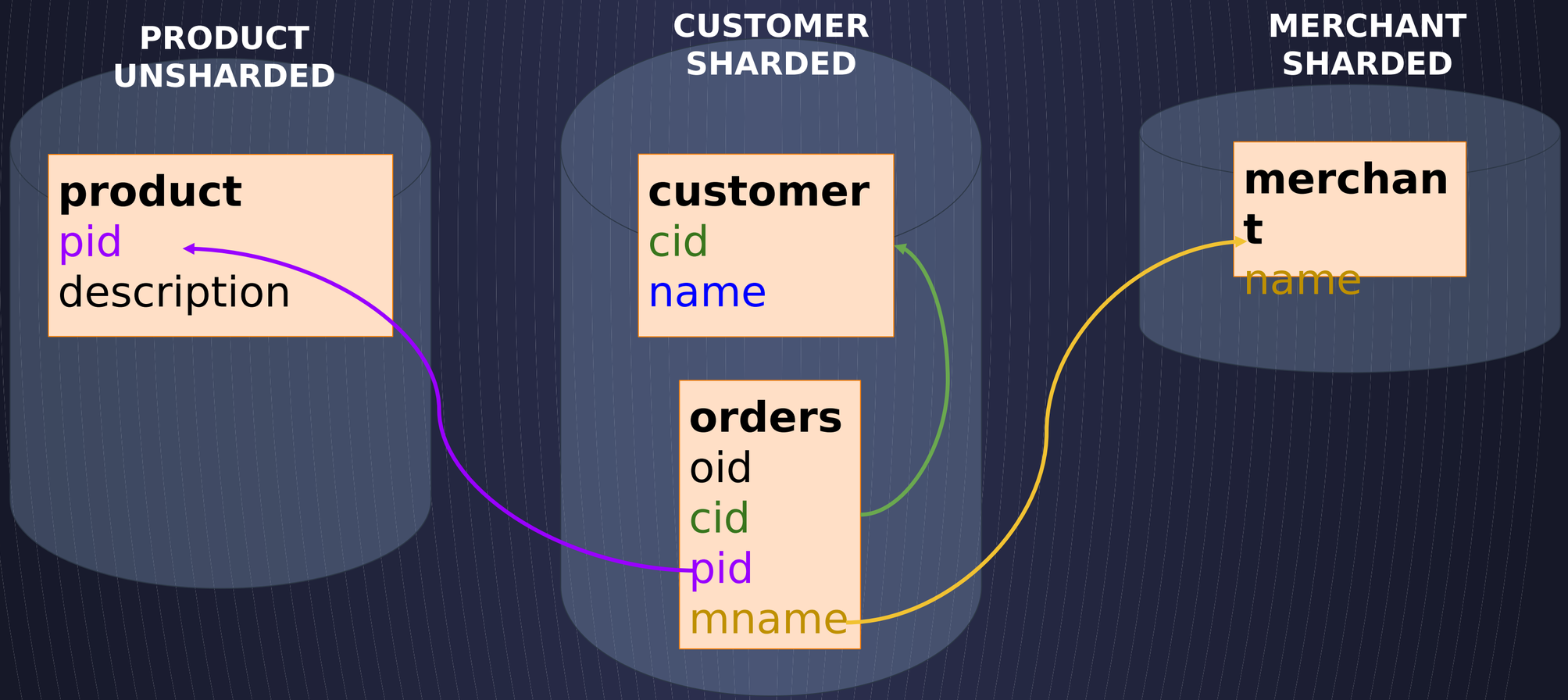
Architecture



Marketplace Schema



Sharded


PRODUCT
UNSHARDED
product
pid
description
CUSTOMER
SHARDED
customer
cid
name
orders
oid
cid
pid
mname
MERCHANT
SHARDED
merchan
t
name

Vitess VReplication Demo

product

product 0

pid	description

customer

customer -80

cid	name	balance

customer 80-

cid	name	balance

orders -80

oid	cid	mname	pid	price

orders 80-

oid	cid	mname	pid	price

merchant

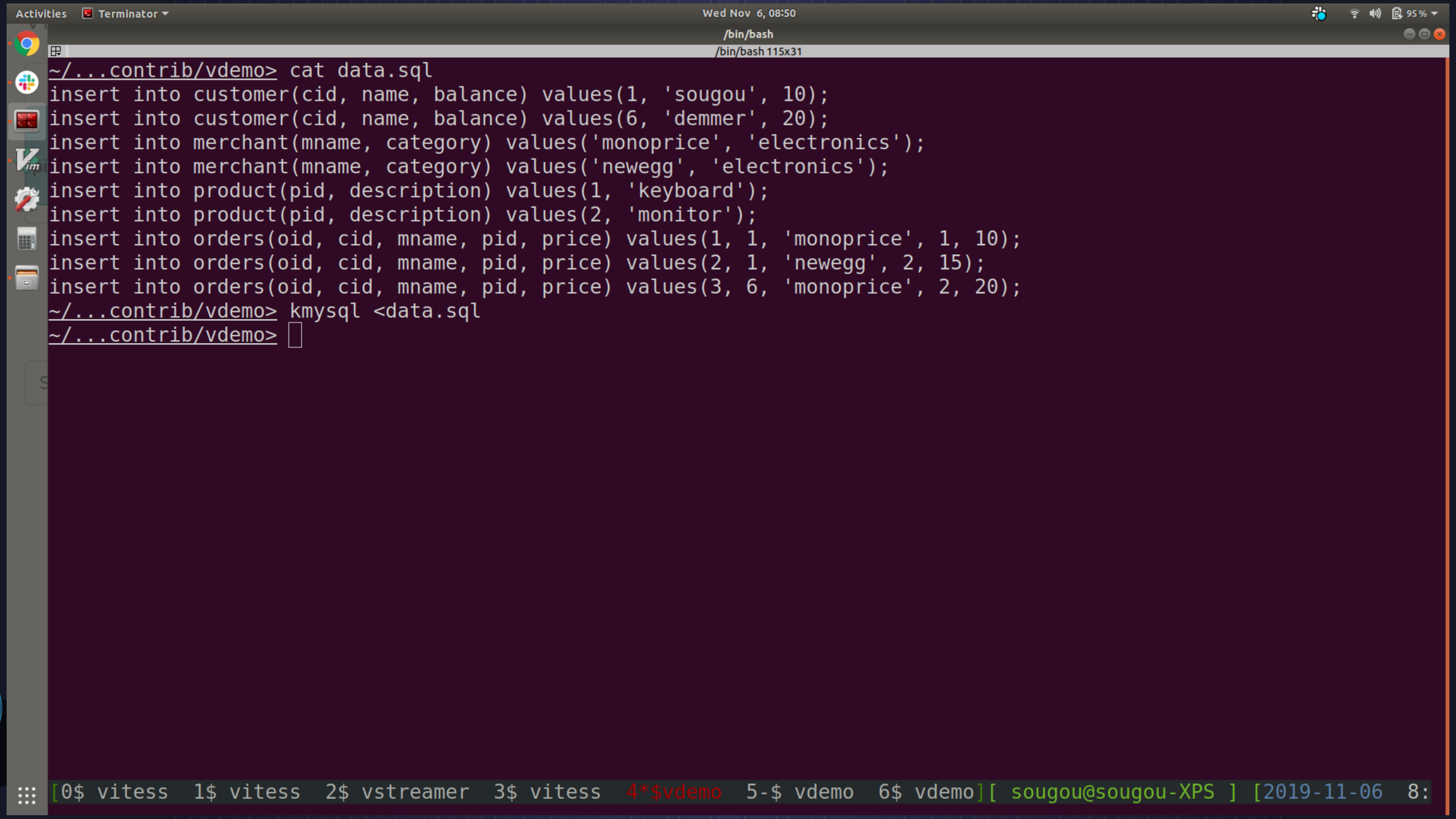
merchant -80

mname	category

merchant 80-

mname	category

Samples ▾ Query


Activities
Terminator
Wed Nov 6, 08:50
95 %
/bin/bash
/bin/bash 115x31
~/...contrib/vdemo> cat data.sql
insert into customer(cid, name, balance) values(1, 'sougou', 10);
insert into customer(cid, name, balance) values(6, 'demmer', 20);
insert into merchant(mname, category) values('monoprice', 'electronics');
insert into merchant(mname, category) values('newegg', 'electronics');
insert into product(pid, description) values(1, 'keyboard');
insert into product(pid, description) values(2, 'monitor');
insert into orders(oid, cid, mname, pid, price) values(1, 1, 'monoprice', 1, 10);
insert into orders(oid, cid, mname, pid, price) values(2, 1, 'newegg', 2, 15);
insert into orders(oid, cid, mname, pid, price) values(3, 6, 'monoprice', 2, 20);
~/...contrib/vdemo> kmysql <data.sql
~/...contrib/vdemo>
[0$ vitess 1$ vitess 2$ vstreamer 3$ vitess 4*$vdemo 5-$ vdemo 6$ vdemo][sougou@sougou-XPS] [2019-11-06 8:

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

Samples ▾ Query

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

Samples ▾ select * from product

result

pid	description
1	keyboard
2	monitor

Executed Queries

product: select * from product limit 10001

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

Samples ▾ select * from customer

result

cid	name	balance
6	demmer	20
1	sougou	10

Executed Queries

customer:80-: select * from customer limit 10001
customer:-80: select * from customer limit 10001

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

Samples ▾

```
select name, oid, mname from customer c join orders o on c.cid = o.cid
```

result

name	oid	mname
demmer	3	monoprice
sougou	1	monoprice
sougou	2	newegg

Executed Queries

customer:80-: select name, oid, mname from customer as c join orders as o on c.cid = o.cid limit 10001
customer:-80: select name, oid, mname from customer as c join orders as o on c.cid = o.cid limit 10001

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

Samples ▾

```
select c.name, p.description from customer c join orders o on c.cid = o.cid join product p on o.pid = p.pid
```

result

name	oid	mname
demmer	3	monoprice
sougou	1	monoprice
sougou	2	newegg

Executed Queries

customer:80-: select name, oid, mname from customer as c join orders as o on c.cid = o.cid limit 10001
customer:-80: select name, oid, mname from customer as c join orders as o on c.cid = o.cid limit 10001

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

Samples ▾

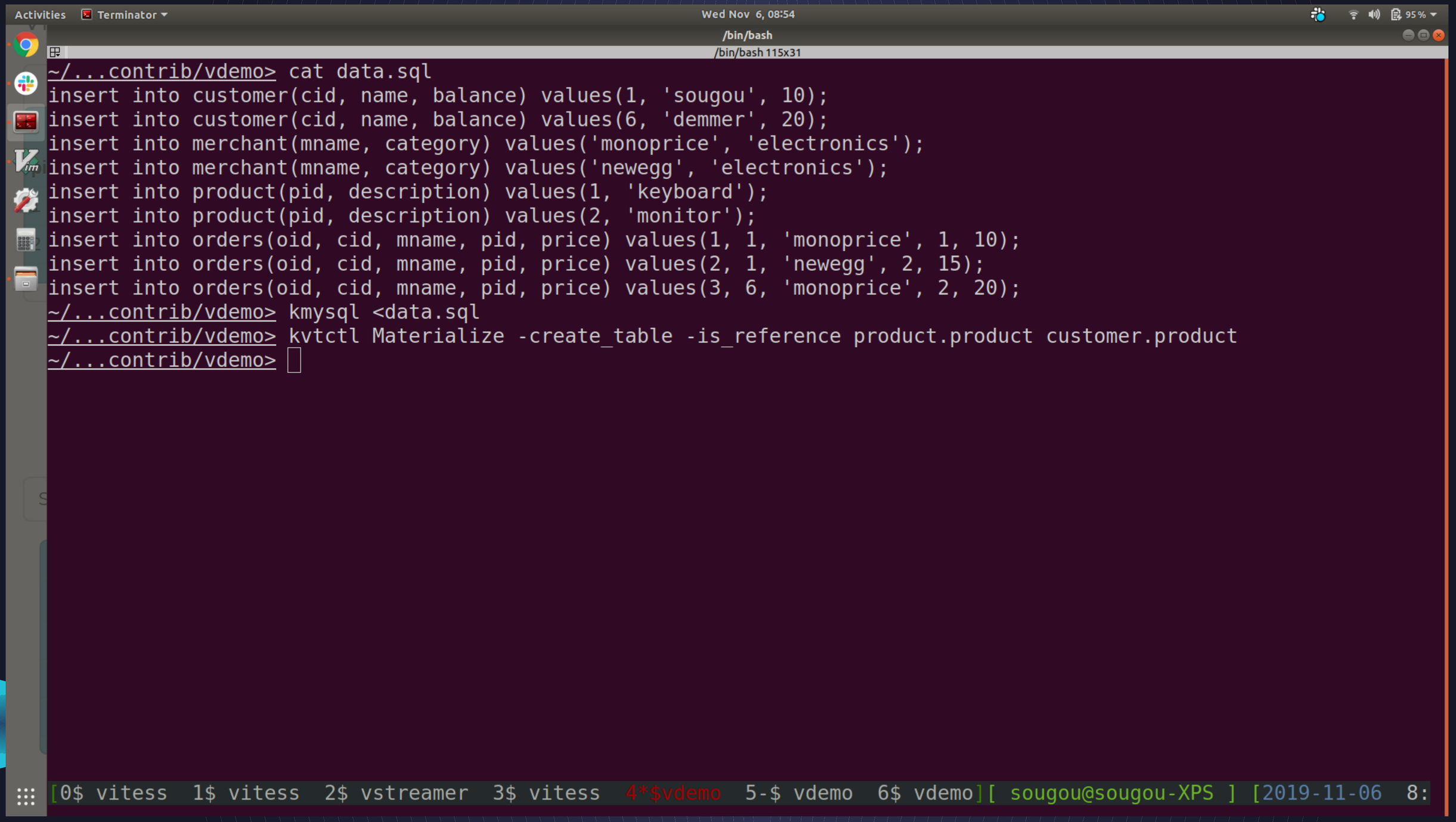
```
select c.name, p.description from customer c join orders o on c.cid = o.cid join product p on o.pid = p.pid
```

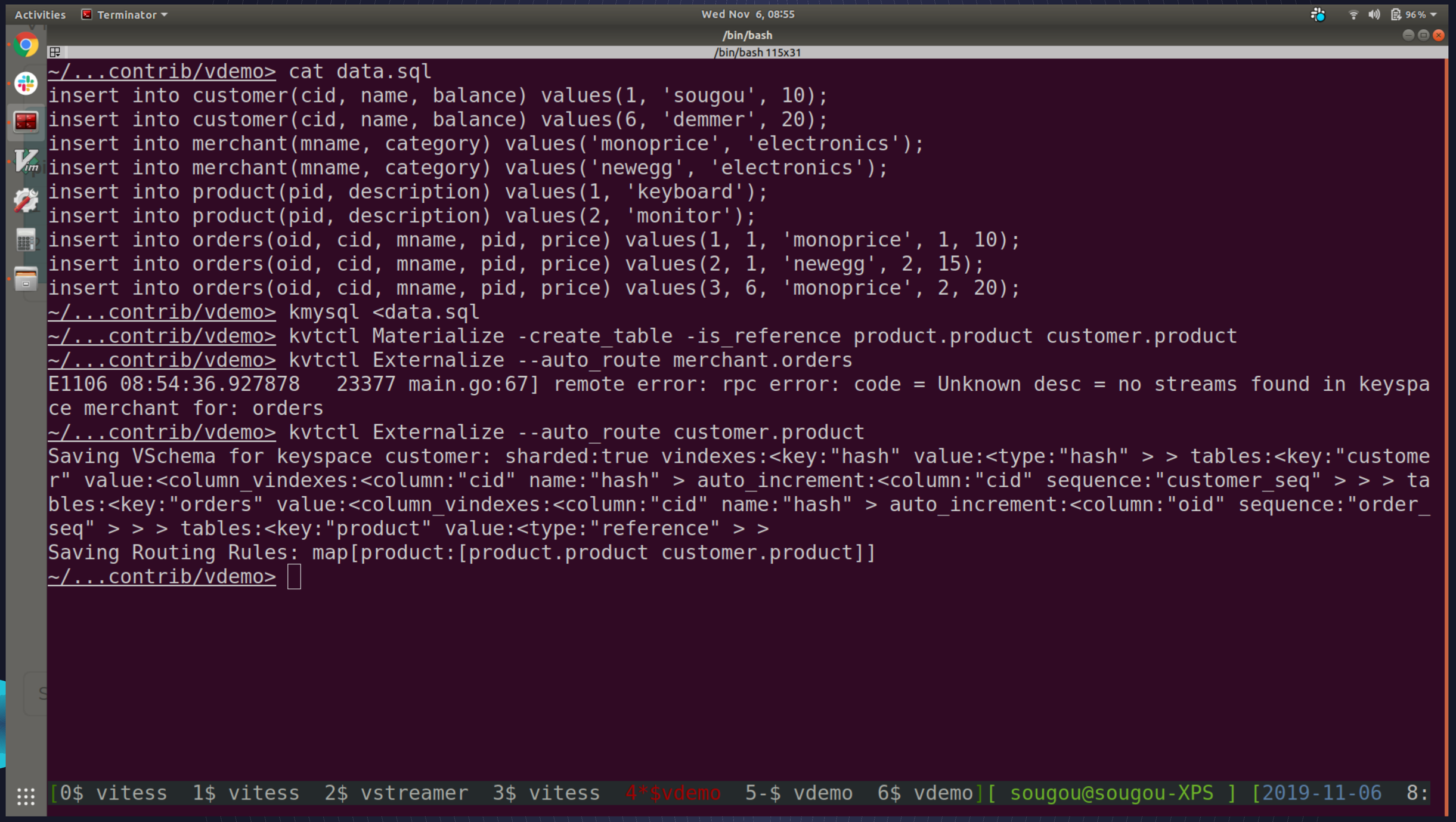
result

name	description
demmer	monitor
sougou	keyboard
sougou	monitor

Executed Queries

customer:80-: select c.name, o.pid from customer as c join orders as o on c.cid = o.cid limit 10001
customer:-80: select c.name, o.pid from customer as c join orders as o on c.cid = o.cid limit 10001
product: select p.description from product as p where p.pid = 2 limit 10001
product: select p.description from product as p where p.pid = 1 limit 10001
product: select p.description from product as p where p.pid = 2 limit 10001


Activities
Terminator
Wed Nov 6, 08:54
95%
/bin/bash
/bin/bash 115x31
~/...contrib/vdemo> cat data.sql
insert into customer(cid, name, balance) values(1, 'sougou', 10);
insert into customer(cid, name, balance) values(6, 'demmer', 20);
insert into merchant(mname, category) values('monoprice', 'electronics');
insert into merchant(mname, category) values('newegg', 'electronics');
insert into product(pid, description) values(1, 'keyboard');
insert into product(pid, description) values(2, 'monitor');
insert into orders(oid, cid, mname, pid, price) values(1, 1, 'monoprice', 1, 10);
insert into orders(oid, cid, mname, pid, price) values(2, 1, 'newegg', 2, 15);
insert into orders(oid, cid, mname, pid, price) values(3, 6, 'monoprice', 2, 20);
~/...contrib/vdemo> kmysql <data.sql
~/...contrib/vdemo> kvtctl Materialize -create_table -is_reference product.product customer.product
~/...contrib/vdemo>
[0$ vitess 1$ vitess 2$ vstreamer 3$ vitess 4*$vdemo 5-$ vdemo 6$ vdemo][sougou@sougou-XPS] [2019-11-06 8:


Activities
Terminator
Wed Nov 6, 08:55
96%
/bin/bash
/bin/bash 115x31
~/...contrib/vdemo> cat data.sql
insert into customer(cid, name, balance) values(1, 'sougou', 10);
insert into customer(cid, name, balance) values(6, 'demmer', 20);
insert into merchant(mname, category) values('monoprice', 'electronics');
insert into merchant(mname, category) values('newegg', 'electronics');
insert into product(pid, description) values(1, 'keyboard');
insert into product(pid, description) values(2, 'monitor');
insert into orders(oid, cid, mname, pid, price) values(1, 1, 'monoprice', 1, 10);
insert into orders(oid, cid, mname, pid, price) values(2, 1, 'newegg', 2, 15);
insert into orders(oid, cid, mname, pid, price) values(3, 6, 'monoprice', 2, 20);
~/...contrib/vdemo> kmysql <data.sql
~/...contrib/vdemo> kvtctl Materialize -create_table -is_reference product.product customer.product
~/...contrib/vdemo> kvtctl Externalize --auto_route merchant.orders
E1106 08:54:36.927878 23377 main.go:67] remote error: rpc error: code = Unknown desc = no streams found in keyspace merchant for: orders
~/...contrib/vdemo> kvtctl Externalize --auto_route customer.product
Saving VSchema for keyspace customer: sharded:true vindexes:<key:"hash" value:<type:"hash" > > tables:<key:"customer" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"cid" sequence:"customer_seq" > > > tables:<key:"orders" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"oid" sequence:"order_seq" > > > tables:<key:"product" value:<type:"reference" > >
Saving Routing Rules: map[product:[product.product customer.product]]
~/...contrib/vdemo>
[0$ vitess 1$ vitess 2$ vstreamer 3$ vitess 4*$vdemo 5-$ vdemo 6$ vdemo][sougou@sougou-XPS] [2019-11-06 8:

pid	description
1	keyboard
2	monitor

cid	name	balance
1	sougou	10

cid	name	balance
6	demmer	20

mname	category
monoprice	electronics

mname	category
newegg	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

product -80

pid	description
1	keyboard
2	monitor

product 80-

pid	description
1	keyboard
2	monitor

Samples ▾ select c.name, p.description from customer c join orders o on c.cid = o.cid join product p on o.pid = p.pid

result

name	description
demmer	monitor
sougou	keyboard
sougou	monitor

Executed Queries

customer:80-: select c.name, p.description from customer as c join orders as o on c.cid = o.cid join product as p on o.pid = p.pid limit 10001

customer:-80: select c.name, p.description from customer as c join orders as o on c.cid = o.cid join product as p on o.pid = p.pid limit 10001

product

product 0

pid	description
1	keyboard
2	monitor
3	mouse

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

product -80

pid	description
1	keyboard
2	monitor
3	mouse

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

merchant

merchant -80

mname	category
monoprice	electronics

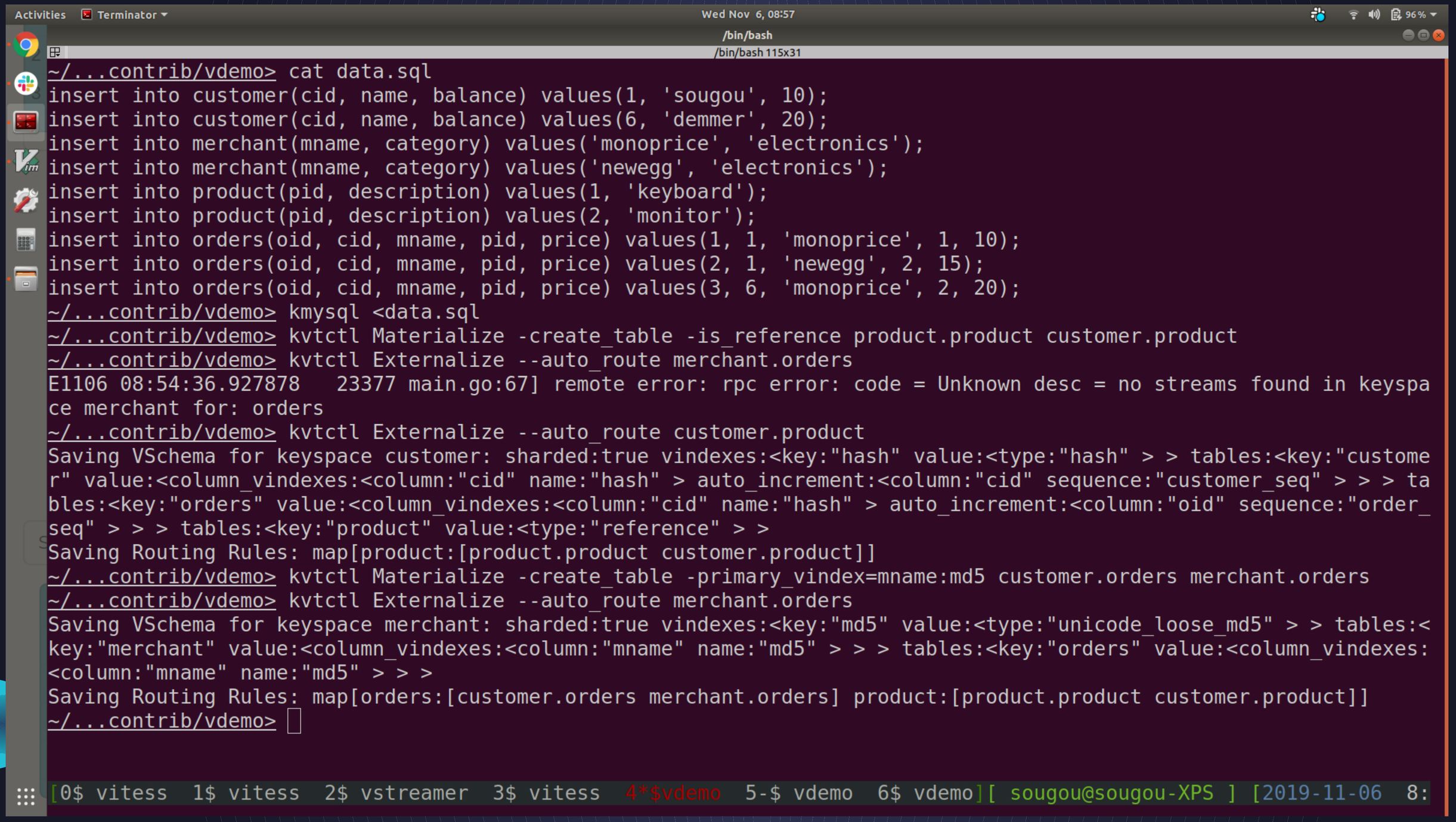
merchant 80-

mname	category
newegg	electronics

Samples ▾ insert into product(pid, description) values(3, 'mouse')

Executed Queries

product: insert into product(pid, description) values (3, 'mouse')


Activities
Terminator
Wed Nov 6, 08:57
96%
/bin/bash
/bin/bash 115x31
~/...contrib/vdemo> cat data.sql
insert into customer(cid, name, balance) values(1, 'sougou', 10);
insert into customer(cid, name, balance) values(6, 'demmer', 20);
insert into merchant(mname, category) values('monoprice', 'electronics');
insert into merchant(mname, category) values('newegg', 'electronics');
insert into product(pid, description) values(1, 'keyboard');
insert into product(pid, description) values(2, 'monitor');
insert into orders(oid, cid, mname, pid, price) values(1, 1, 'monoprice', 1, 10);
insert into orders(oid, cid, mname, pid, price) values(2, 1, 'newegg', 2, 15);
insert into orders(oid, cid, mname, pid, price) values(3, 6, 'monoprice', 2, 20);
~/...contrib/vdemo> kmysql <data.sql
~/...contrib/vdemo> kvtctl Materialize -create_table -is_reference product.product customer.product
~/...contrib/vdemo> kvtctl Externalize --auto_route merchant.orders
E1106 08:54:36.927878 23377 main.go:67] remote error: rpc error: code = Unknown desc = no streams found in keyspace merchant for: orders
~/...contrib/vdemo> kvtctl Externalize --auto_route customer.product
Saving VSchema for keyspace customer: sharded:true vindexes:<key:"hash" value:<type:"hash" > > tables:<key:"customer" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"cid" sequence:"customer_seq" > > > tables:<key:"orders" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"oid" sequence:"order_seq" > > > tables:<key:"product" value:<type:"reference" > >
Saving Routing Rules: map[product:[product.product customer.product]]
~/...contrib/vdemo> kvtctl Materialize -create_table -primary_vindex=mname:md5 customer.orders merchant.orders
~/...contrib/vdemo> kvtctl Externalize --auto_route merchant.orders
Saving VSchema for keyspace merchant: sharded:true vindexes:<key:"md5" value:<type:"unicode_loose_md5" > > tables:<key:"merchant" value:<column_vindexes:<column:"mname" name:"md5" > > > tables:<key:"orders" value:<column_vindexes:<column:"mname" name:"md5" > > >
Saving Routing Rules: map[orders:[customer.orders merchant.orders] product:[product.product customer.product]]
~/...contrib/vdemo>
[0$ vitess 1$ vitess 2$ vstreamer 3$ vitess 4*$vdemo 5-$ vdemo 6$ vdemo][sougou@sougou-XPS] [2019-11-06 8:

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor
3	mouse

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

product -80

pid	description
1	keyboard
2	monitor
3	mouse

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

Samples

```
select m.mname, m.category, o.oid from merchant m join orders o on m.mname = o.mname
```

1	keyboard
2	monitor
3	mouse

1	sougou	10

6	demmer	20

monoprice	electronics

newegg	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

product -80

pid	description
1	keyboard
2	monitor
3	mouse

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

Samples ▾ select m.mname, m.category, o.oid from merchant m join orders o on m.mname = o.mname

result

mname	category	oid
monoprice	electronics	1
monoprice	electronics	3
newegg	electronics	2

Executed Queries

merchant:80-: select m.mname, m.category, o.oid from merchant as m join orders as o on m.mname = o.mname limit 10001

merchant:-80: select m.mname, m.category, o.oid from merchant as m join orders as o on m.mname = o.mname limit 10001

```
insert into product(pid, description) values(2, 'monitor');
insert into orders(oid, cid, mname, pid, price) values(1, 1, 'monoprice', 1, 10);
insert into orders(oid, cid, mname, pid, price) values(2, 1, 'newegg', 2, 15);
insert into orders(oid, cid, mname, pid, price) values(3, 6, 'monoprice', 2, 20);
~/...contrib/vdemo> kmysql <data.sql
~/...contrib/vdemo> kvtctl Materialize -create_table -is_reference product.product customer.product
~/...contrib/vdemo> kvtctl Externalize --auto_route merchant.orders
E1106 08:54:36.927878   23377 main.go:67] remote error: rpc error: code = Unknown desc = no streams found in keyspa
ce merchant for: orders
~/...contrib/vdemo> kvtctl Externalize --auto_route customer.product
Saving VSchema for keyspace customer: sharded:true vindexes:<key:"hash" value:<type:"hash" > > tables:<key:"custome
r" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"cid" sequence:"customer_seq" > > > ta
bles:<key:"orders" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"oid" sequence:"order_
seq" > > > tables:<key:"product" value:<type:"reference" > >
Saving Routing Rules: map[product:[product.product customer.product]]
~/...contrib/vdemo> kvtctl Materialize -create_table -primary_vindex=mname:md5 customer.orders merchant.orders
~/...contrib/vdemo> kvtctl Externalize --auto_route merchant.orders
Saving VSchema for keyspace merchant: sharded:true vindexes:<key:"md5" value:<type:"unicode_loose_md5" > > tables:<
key:"merchant" value:<column_vindexes:<column:"mname" name:"md5" > > > tables:<key:"orders" value:<column_vindexes:
<column:"mname" name:"md5" > > >
Saving Routing Rules: map[orders:[customer.orders merchant.orders] product:[product.product customer.product]]
~/...contrib/vdemo> kvtctl ApplySchema -sql='create table sales(pid int, kount int, amount int, primary key(pid))'
product
~/...contrib/vdemo> kvtctl Materialize 'select pid, count(*) as kount, sum(price) as amount from customer.orders gr
oup by pid' product.sales
~/...contrib/vdemo> kvtctl Externalize product.sales
Saving VSchema for keyspace product: tables:<key:"customer_seq" value:<type:"sequence" > > tables:<key:"order_seq"
value:<type:"sequence" > > tables:<key:"product" value:<> > tables:<key:"sales" value:<> >
Saving Routing Rules: map[orders:[customer.orders merchant.orders] product:[product.product customer.product]]
~/...contrib/vdemo>
```

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor
3	mouse

sales 0

pid	kount	amount
1	1	10
2	2	35

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

product -80

pid	description
1	keyboard
2	monitor
3	mouse

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

Samples

```
select product.pid, description, amount from product join sales on product.pid = sales.pid
```

pid	description
1	keyboard
2	monitor
3	mouse

sales 0

pid	kount	amount
1	1	10
2	2	35

cid	name	balance
1	sougou	10

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

product -80

pid	description
1	keyboard
2	monitor
3	mouse

cid	name	balance
6	demmer	20

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

mname	category
monoprice	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20

mname	category
newegg	electronics

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

Samples ▾ select product.pid, description, amount from product join sales on product.pid = sales.pid

result

pid	description	amount
1	keyboard	10
2	monitor	35

Executed Queries

product: select product.pid, description, amount from product join sales on product.pid = sales.pid limit 10001

product 0

pid	description
1	keyboard
2	monitor
3	mouse

sales 0

pid	kount	amount
1	1	10
2	2	35

customer -80

cid	name	balance
1	sougou	10

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

product -80

pid	description
1	keyboard
2	monitor
3	mouse

customer 80-

cid	name	balance
6	demmer	20

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

merchant -80

mname	category
monoprice	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20

merchant 80-

mname	category
newegg	electronics

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

Samples ▾ select description, kount, amount from product join sales on product.pid = sales.pid order by amount desc limit

result

description	kount	amount
monitor	2	35

Executed Queries

product: select description, kount, amount from product join sales on product.pid = sales.pid order by amount desc limit 1

product

product 0

pid	description
1	keyboard
2	monitor
3	mouse

sales 0

pid	kount	amount
1	2	60
2	2	35

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20
4	6	monoprice	1	50

product -80

pid	description
1	keyboard
2	monitor
3	mouse

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20
4	6	monoprice	1	50

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

Samples ▾

```
insert into orders(oid, cid, mname, pid, price) values(4, 6, 'monoprice', 1, 50)
```

Executed Queries

customer:80-: insert into orders(oid, cid, mname, pid, price) values (4, 6, 'monoprice', 1, 50) /* vtgate:: keyspace_id:f098480ac4c4be71 */

product 0

pid	description
1	keyboard
2	monitor
3	mouse

sales 0

pid	kount	amount
1	2	60
2	2	35

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
2	1	newegg	2	15

orders 80-

oid	cid	mname	pid	price
3	6	monoprice	2	20
4	6	monoprice	1	50

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20
4	6	monoprice	1	50

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

product -80

pid	description
1	keyboard
2	monitor
3	mouse

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

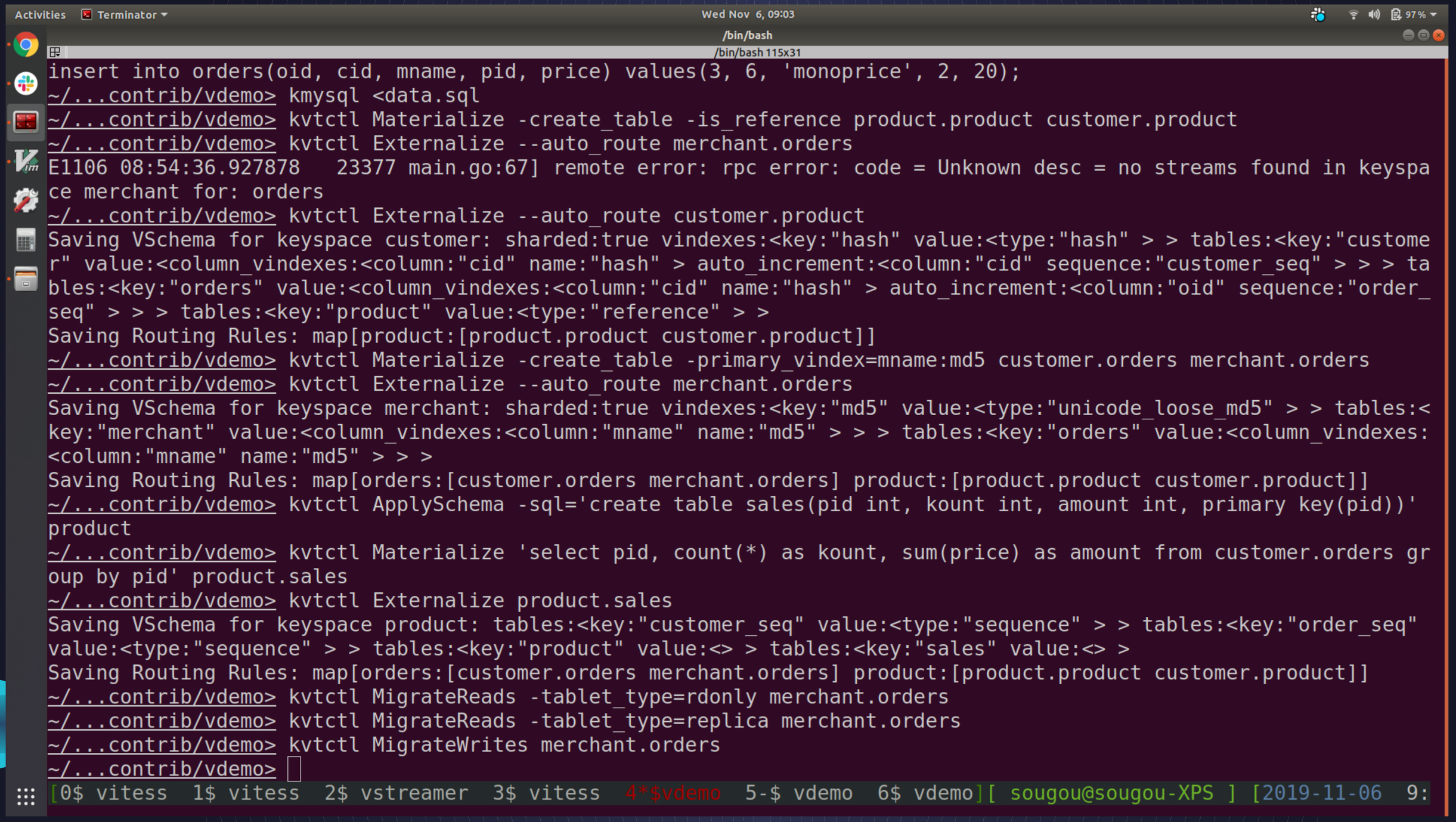
Samples ▾ select description, kount, amount from product join sales on product.pid = sales.pid order by amount desc limit

result

description	kount	amount
keyboard	2	60

Executed Queries

product: select description, kount, amount from product join sales on product.pid = sales.pid order by amount desc limit 1


Activities
Terminator
Wed Nov 6, 09:03
97%
/bin/bash
/bin/bash 115x31
insert into orders(oid, cid, mname, pid, price) values(3, 6, 'monoprice', 2, 20);
~/...contrib/vdemo> kmysql <data.sql
~/...contrib/vdemo> kvtctl Materialize -create_table -is_reference product.product customer.product
~/...contrib/vdemo> kvtctl Externalize --auto_route merchant.orders
E1106 08:54:36.927878 23377 main.go:67] remote error: rpc error: code = Unknown desc = no streams found in keyspace merchant for: orders
~/...contrib/vdemo> kvtctl Externalize --auto_route customer.product
Saving VSchema for keyspace customer: sharded:true vindexes:<key:"hash" value:<type:"hash" > > tables:<key:"customer" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"cid" sequence:"customer_seq" > > > tables:<key:"orders" value:<column_vindexes:<column:"cid" name:"hash" > auto_increment:<column:"oid" sequence:"order_seq" > > > tables:<key:"product" value:<type:"reference" > >
Saving Routing Rules: map[product:[product.product customer.product]]
~/...contrib/vdemo> kvtctl Materialize -create_table -primary_vindex=mname:md5 customer.orders merchant.orders
~/...contrib/vdemo> kvtctl Externalize --auto_route merchant.orders
Saving VSchema for keyspace merchant: sharded:true vindexes:<key:"md5" value:<type:"unicode_loose_md5" > > tables:<key:"merchant" value:<column_vindexes:<column:"mname" name:"md5" > > > tables:<key:"orders" value:<column_vindexes:<column:"mname" name:"md5" > > >
Saving Routing Rules: map[orders:[customer.orders merchant.orders] product:[product.product customer.product]]
~/...contrib/vdemo> kvtctl ApplySchema -sql='create table sales(pid int, kount int, amount int, primary key(pid))' product
~/...contrib/vdemo> kvtctl Materialize 'select pid, count(*) as kount, sum(price) as amount from customer.orders group by pid' product.sales
~/...contrib/vdemo> kvtctl Externalize product.sales
Saving VSchema for keyspace product: tables:<key:"customer_seq" value:<type:"sequence" > > tables:<key:"order_seq" value:<type:"sequence" > > tables:<key:"product" value:<> > tables:<key:"sales" value:<> >
Saving Routing Rules: map[orders:[customer.orders merchant.orders] product:[product.product customer.product]]
~/...contrib/vdemo> kvtctl MigrateReads -tablet_type=rdonly merchant.orders
~/...contrib/vdemo> kvtctl MigrateReads -tablet_type=replica merchant.orders
~/...contrib/vdemo> kvtctl MigrateWrites merchant.orders
~/...contrib/vdemo>
[0$ vitess 1$ vitess 2$ vstreamer 3$ vitess 4*$vdemo 5-$ vdemo 6$ vdemo][sougou@sougou-XPS] [2019-11-06 9:

Vitess VReplication Demo

product

product 0

pid	description
1	keyboard
2	monitor
3	mouse

sales 0

pid	kount	amount
1	2	60
2	2	35

customer

customer -80

cid	name	balance
1	sougou	10

customer 80-

cid	name	balance
6	demmer	20

uorder0: (1105, 'vtgate: http://sougou-XPS:15001/: target: customer.-80.master, used tablet: test-200 (sougou-XPS): vttablet: rpc error: code = FailedPrecondition desc = disallowed due to rule: enforce blacklisted tables (CallerID:)')

uorder1: (1105, 'vtgate: http://sougou-XPS:15001/: target: customer.80-.master, used tablet: test-300 (sougou-XPS): vttablet: rpc error: code = FailedPrecondition desc = disallowed due to rule: enforce blacklisted tables (CallerID:)')

product -80

pid	description
1	keyboard
2	monitor
3	mouse

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

merchant

merchant -80

mname	category
monoprice	electronics

merchant 80-

mname	category
newegg	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20
4	6	monoprice	1	50

orders 80-

oid	cid	mname	pid	price
2	1	newegg	2	15

1	keyboard
2	monitor
3	mouse

sales 0

pid	kount	amount
1	3	110
2	2	35

1	sougou	10

uorder0: (1105, 'vtgate: http://sougou-XPS:15001/: target: customer.-80.master, used tablet: test-200 (sougou-XPS): vttablet: rpc error: code = FailedPrecondition desc = disallowed due to rule: enforce blacklisted tables (CallerID:)')

product -80

pid	description
1	keyboard
2	monitor
3	mouse

6	demmer	20

uorder1: (1105, 'vtgate: http://sougou-XPS:15001/: target: customer.80-.master, used tablet: test-300 (sougou-XPS): vttablet: rpc error: code = FailedPrecondition desc = disallowed due to rule: enforce blacklisted tables (CallerID:)')

product 80-

pid	description
1	keyboard
2	monitor
3	mouse

monoprice	electronics

orders -80

oid	cid	mname	pid	price
1	1	monoprice	1	10
3	6	monoprice	2	20
4	6	monoprice	1	50
5	6	monoprice	1	50

newegg	electronics

orders 80-

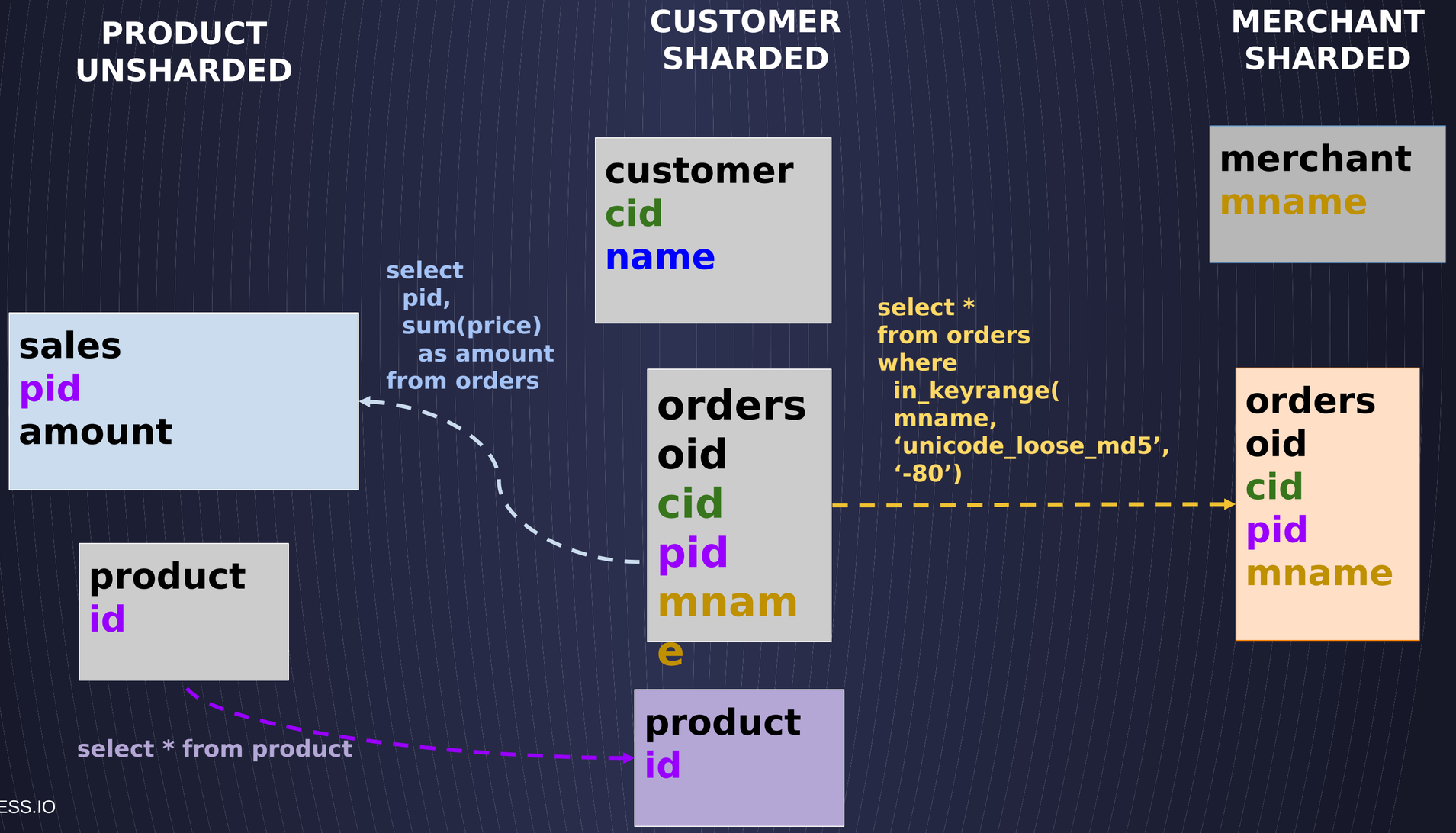
oid	cid	mname	pid	price
2	1	newegg	2	15

Samples ▾ insert into orders(oid, cid, mname, pid, price) values(5, 6, 'monoprice', 1, 50)

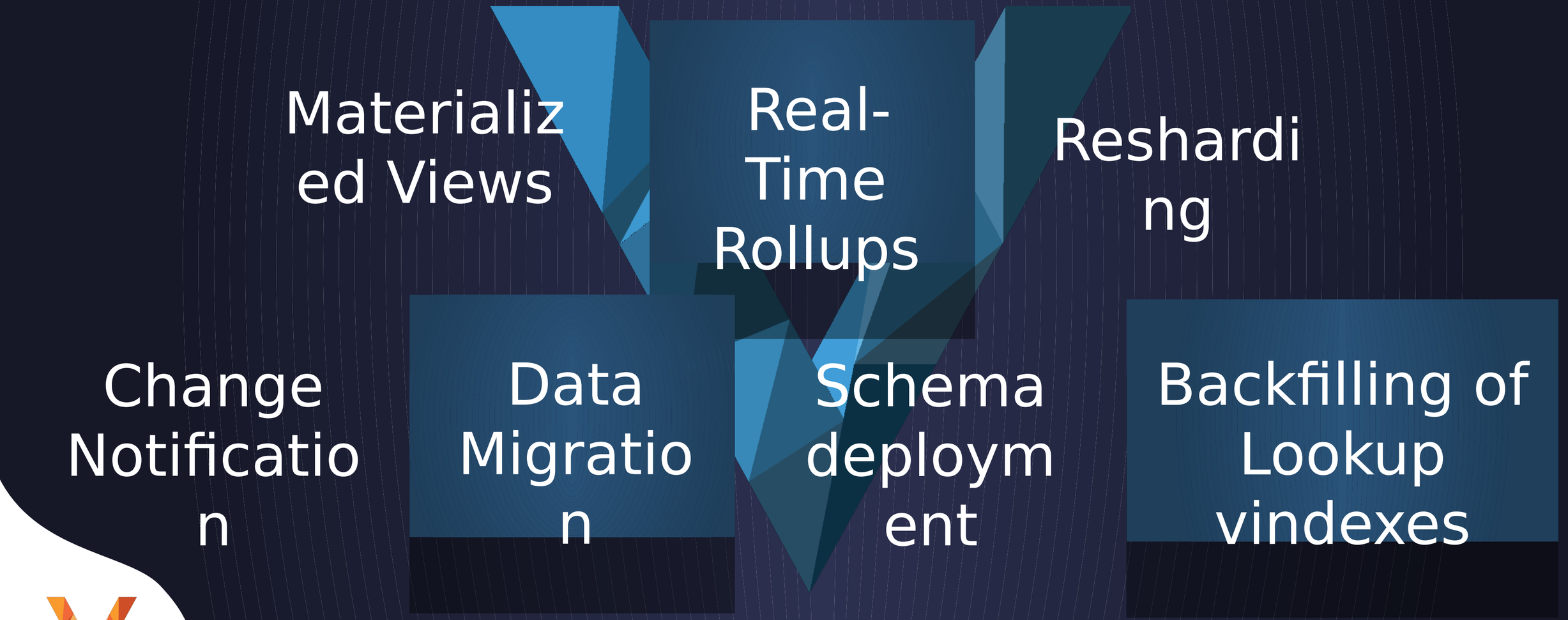
Executed Queries

merchant:-80: insert into orders(oid, cid, mname, pid, price) values (5, 6, 'monoprice', 1, 50) /* vtgate:: keyspace_id:7416746d4309a1bbb73e1817a482aa95 */

VReplication



Use Cases


Materialized Views
Real-Time Rollups
Resharding
Change Notification
Data Migration
Schema deployment
Backfilling of Lookup vindexes

What's Next

vitess.io

Do the tutorial

Join Vitess Slack

console.planetscale.com

Bring up a cluster