Pivotal Summit 2019

Bringing Cloud Databases On-Premises with Greenplum and Kubernetes

November 19, 2019

Choon Sam Park
Director, Data Business Korea/Japan

Part 1: Why

AI and Ease of Use


“Software Ate The World,
Now AI Is Eating Software”

AI is eating software – AI at the core of new startups

BenchSci blog: since November 2017, listed 158 startups using machine learning to research and develop drugs

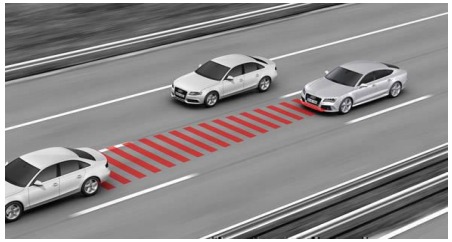
AI-powered code generation tools like TabNine, TypeSQL and BAYOU

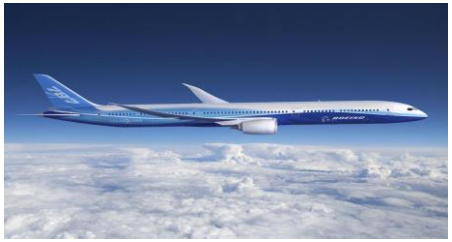
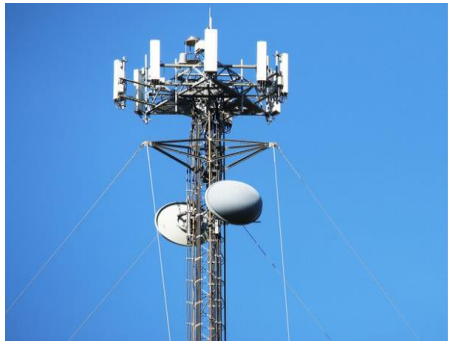
Getting answers to any question about your medical data, from natural language to AI generated SQL (Question-to-SQL)

LabTwin

- *Source: https://www.forbes.com/sites/cognitiveworld/2019/08/29/software-ate-the-world-now-ai-is-eating-software/#388361a85810*

Customers Bring AI to their core business w/ Greenplum

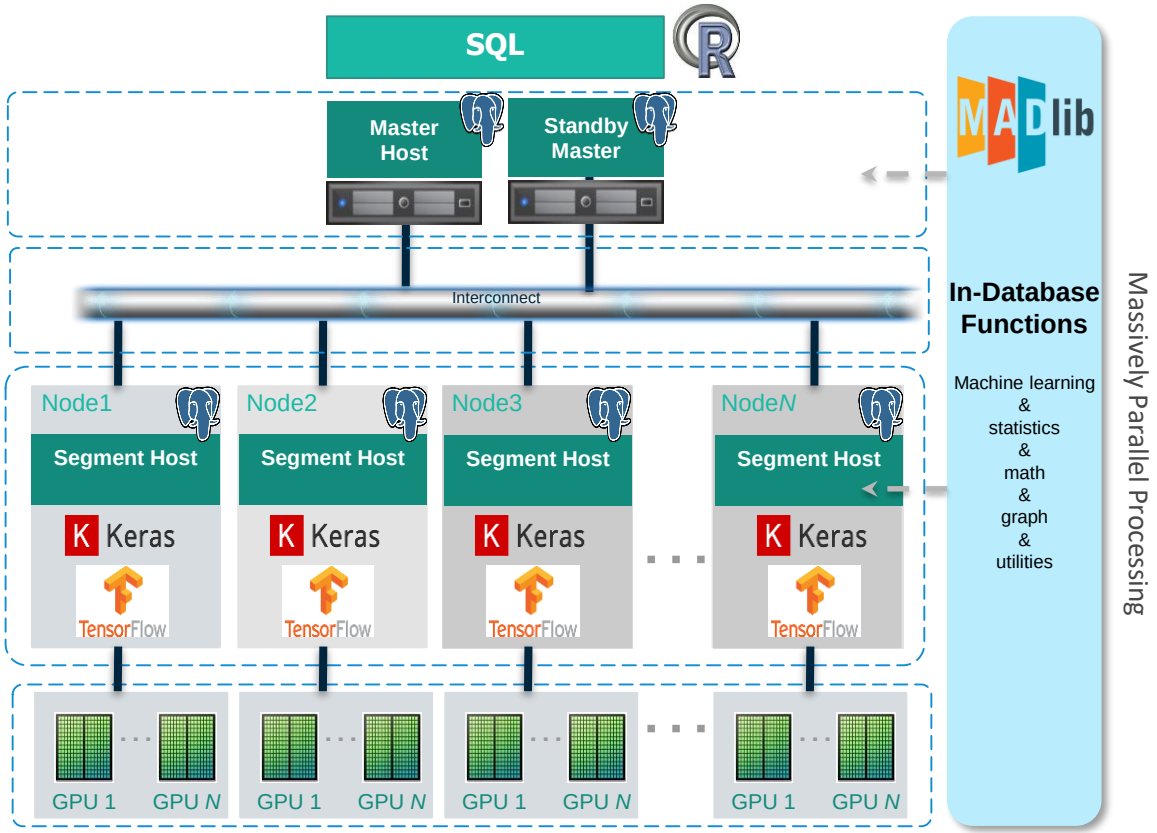


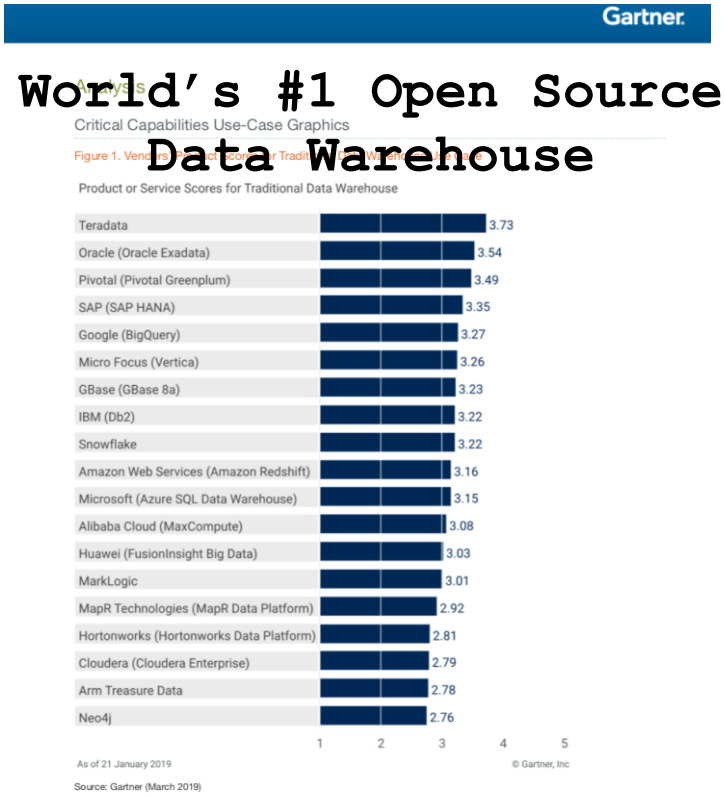



Greenplum : Integrated Analytic Platform

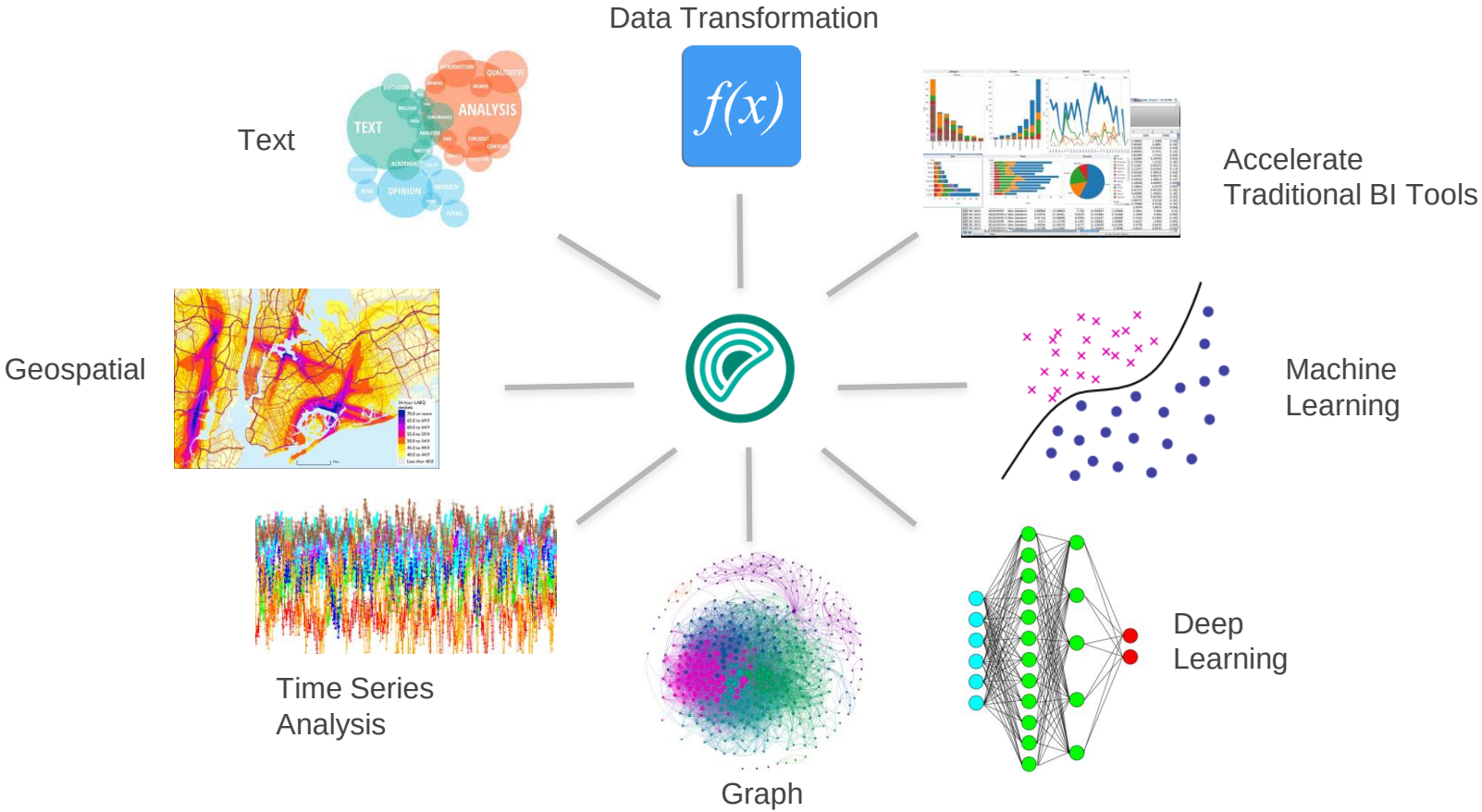
- Started Greenplum in California since 2003
- Acquired by EMC in 2010
  - Launched the Data Computing Appliance
- Pivotal Created in 2013
  - Greenplum, Gemfire, Pivotal Labs, Spring, Cloud Foundry
- Greenplum is Opensource based Analytic Platform
  - Distribution Architecture, ML/AI, Anywhere, K8s

Greenplum = Massively Parallel Postgres for Analytics





Greenplum = A massively parallel Postgres for AI



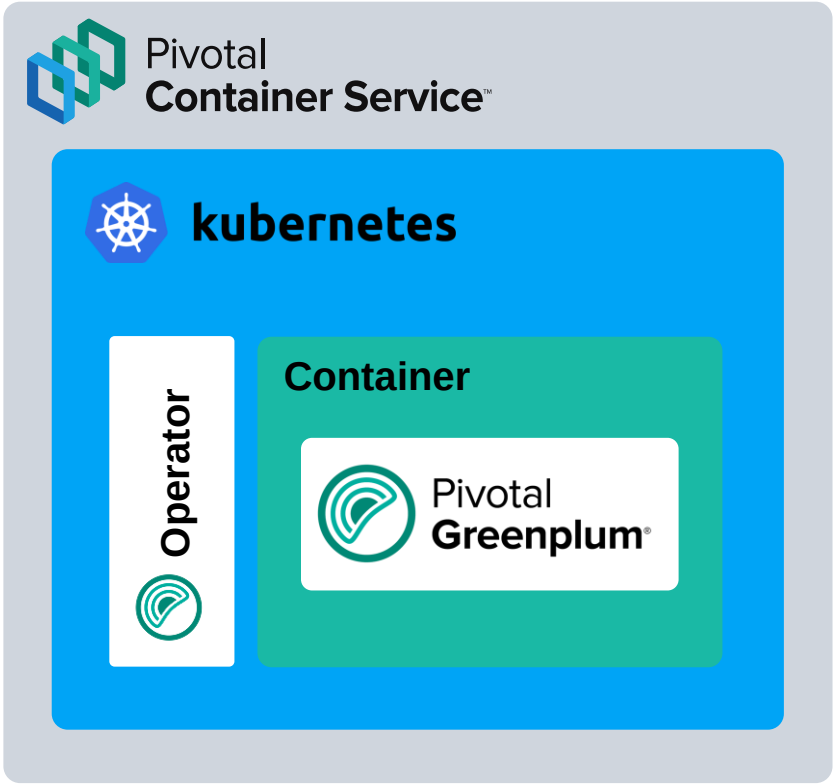
Bringing Cloud Databases On-Premises

1. Greenplum is embedded in containers for portability and dependency management

1. Each container is managed by Kubernetes for higher availability & elasticity

1. Kubernetes operator is used for automation

1. PKS for multi-cloud and day-2 operations with full-stack support



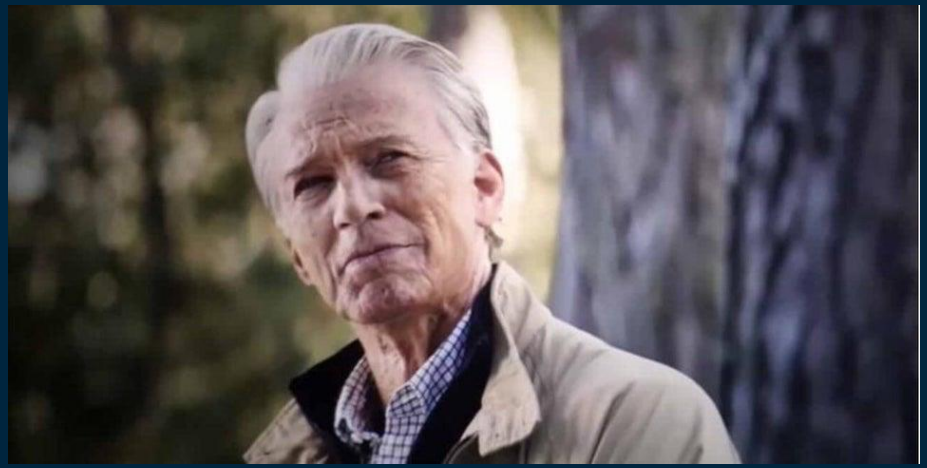
Part 2: Scenario

A day in the Life of a Data Scientist

This is (almost) a real scenario

https://github.com/pnagula/Greenplum-Super-Query


Where is Captain America?



1. I need an AI Platform

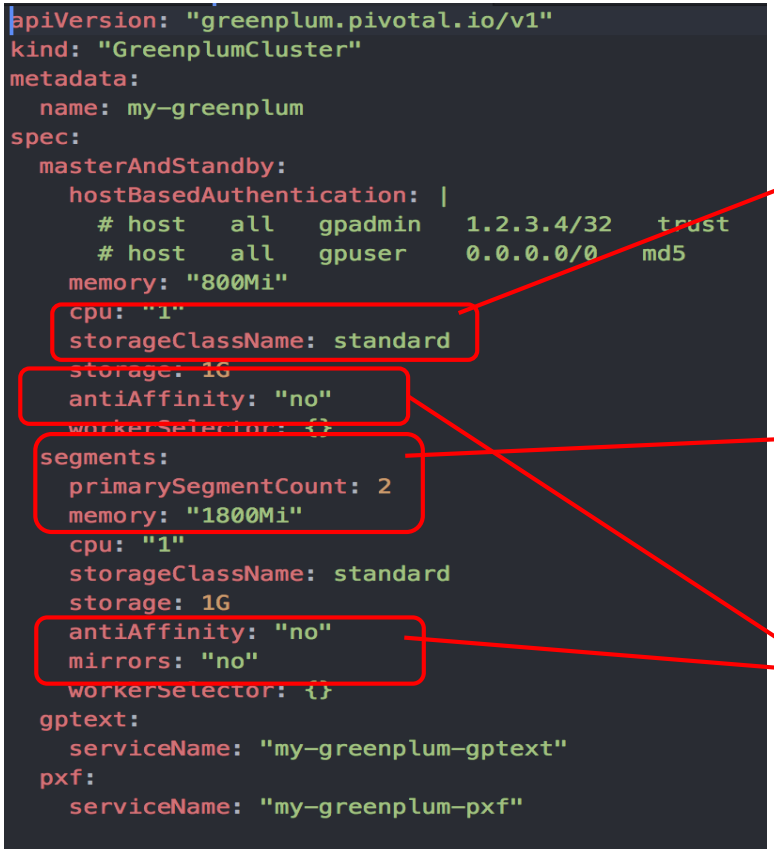
Pivotal

My friendly Ops Team has done some "One-Time Setup" for me.

K8s Cluster Ready
Operator Ready

- Downloaded Greenplum for K8s
- Uploaded images to registry
- Created K8s cluster
- Deployed Greenplum Operator
- Prepared instance manifest
  - add extensions
  - adjust storage, sizing, etc.

I get to decide what options to use!



```yaml
apiVersion: "greenplum.pivotal.io/v1"
kind: "GreenplumCluster"
metadata:
  name: my-greenplum
spec:
  masterAndStandby:
    hostBasedAuthentication: |
      # host    all    gpadmin    1.2.3.4/32    trust
      # host    all    gpuser     0.0.0.0/0     md5
    memory: "800Mi"
    cpu: "1"
    storageClassName: standard
    storage: 1G
    antiAffinity: "no"
    workerSelector: {}
  segments:
    primarySegmentCount: 2
    memory: "1800Mi"
    cpu: "1"
    storageClassName: standard
    storage: 1G
    antiAffinity: "no"
    mirrors: "no"
    workerSelector: {}
  gptext:
    serviceName: "my-greenplum-gptext"
  pxf:
    serviceName: "my-greenplum-pxf"
```

For Best Performance:

- Backed by a local SSD
- XFS filesystem, using `readahead` cache

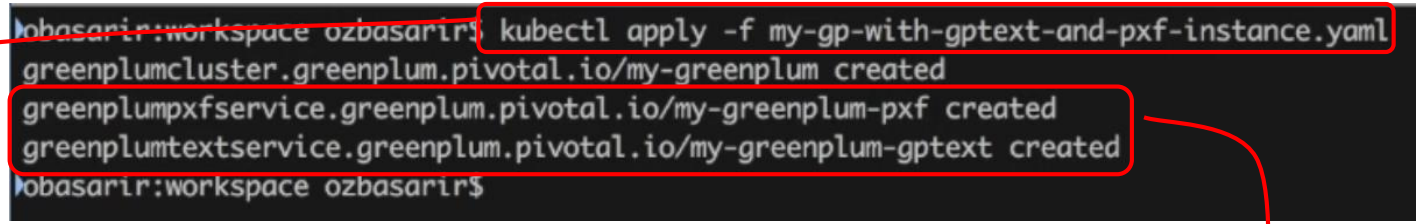
- Only 2 Segments to get started
- 1 GB each because we are in Dev.

.

- No Mirrors we are in Dev.
- AntiAffinity turned off with no mirrors

Same Command

- Initialize Greenplum Workbench
- Update Configuration
- Upgrade Minor Versions
- Apply Patches



Kubectl apply -f my-gp.yaml

- Options installed automatically

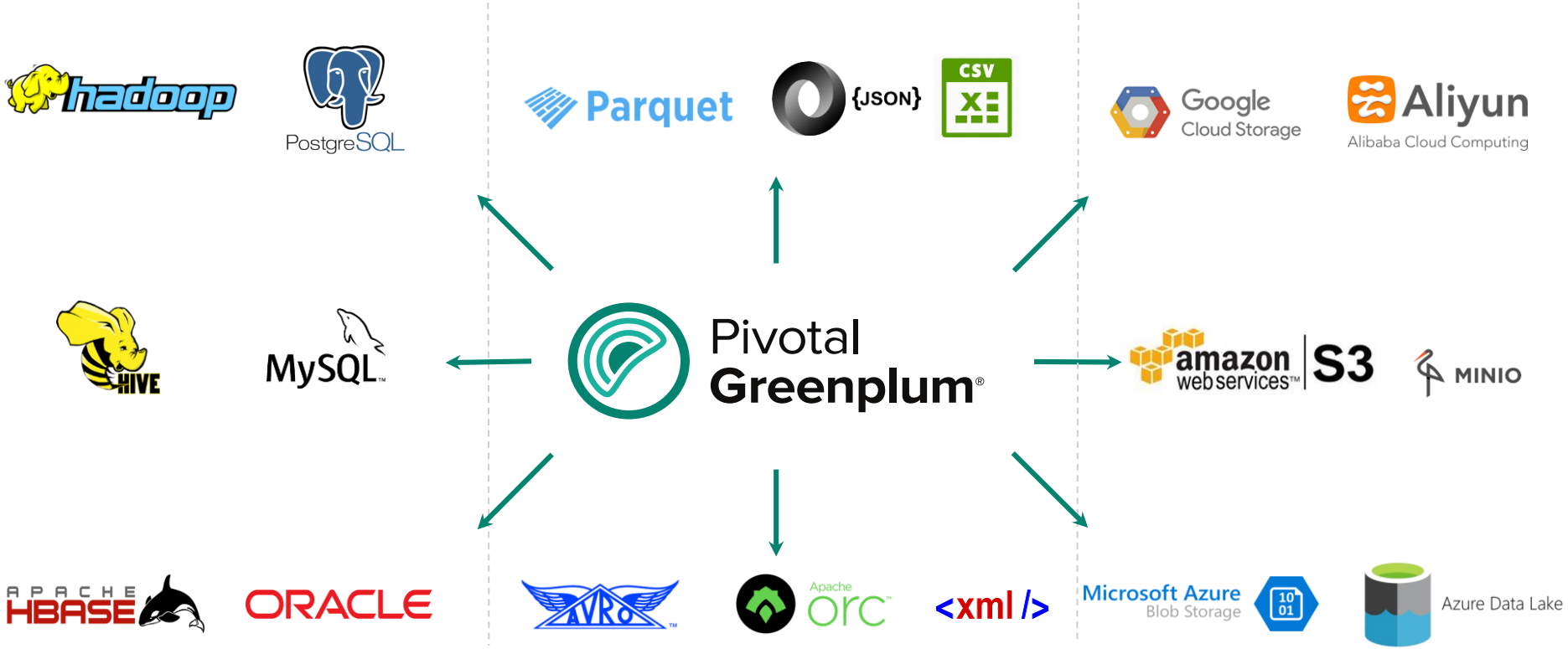
Ready for User Queries in 94 seconds

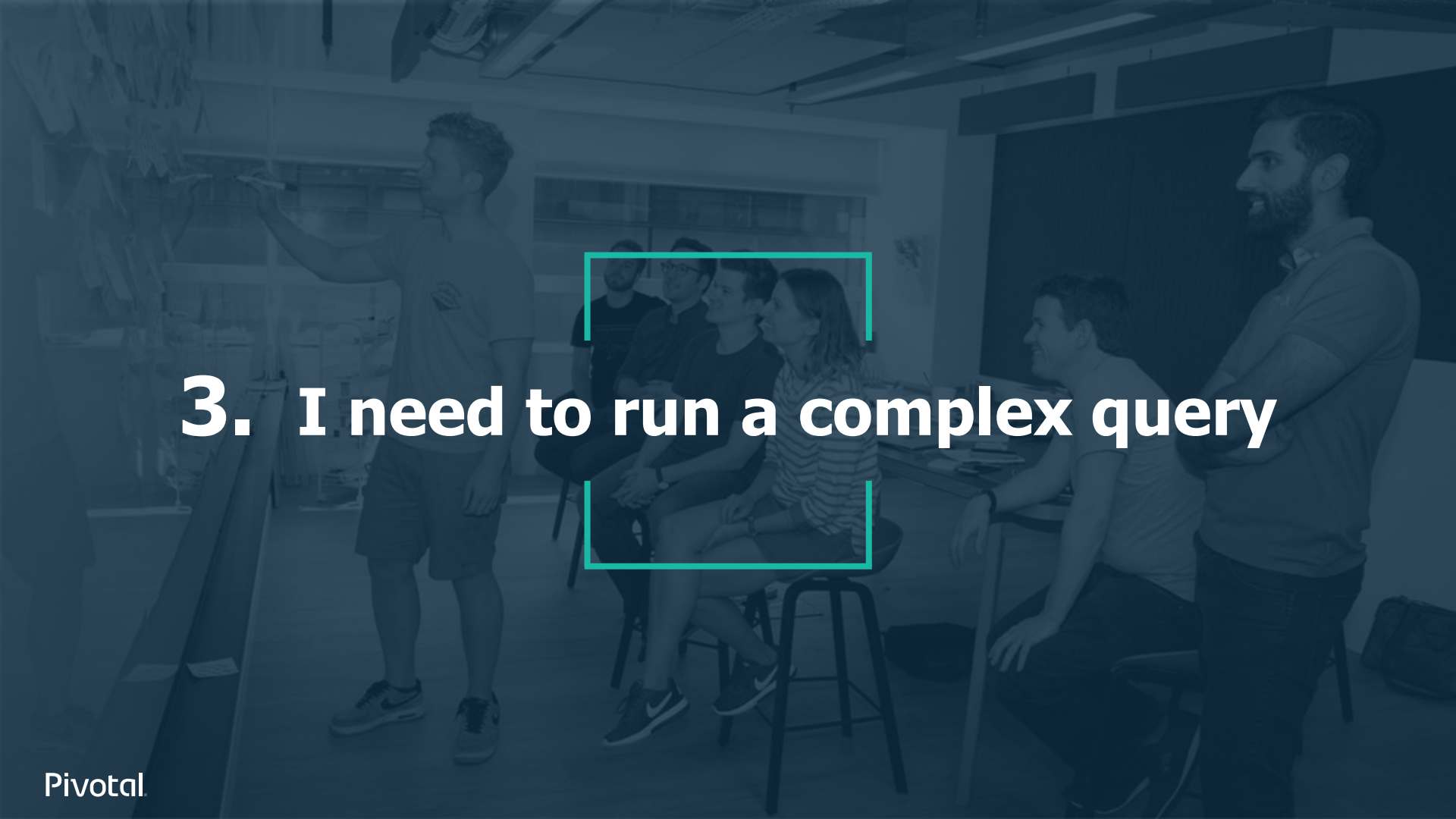
```
NAME                                                  STATUS    AGE
greenplumcluster.greenplum.pivotal.io/my-greenplum    Running   94s

NAME                                                                AGE
greenplumtextservice.greenplum.pivotal.io/my-greenplum-gptext       93s

NAME                                                          AGE
greenplumpxfservice.greenplum.pivotal.io/my-greenplum-pxf     94s

NAME                                    READY   STATUS    RESTARTS   AGE
pod/greenplum-operator-7fbffdcf64-w6vzw 1/1     Running   0          2d7h
pod/master-0                            1/1     Running   0          90s
pod/master-1                            1/1     Running   0          90s
pod/my-greenplum-gptext-solr-0          1/1     Running   0          93s
pod/my-greenplum-gptext-zookeeper-0     1/1     Running   0          93s
pod/my-greenplum-gptext-zookeeper-1     1/1     Running   0          77s
pod/my-greenplum-gptext-zookeeper-2     1/1     Running   0          53s
pod/my-greenplum-pxf-d5489784b-rhgts    1/1     Running   0          93s
pod/my-greenplum-pxf-d5489784b-sst9n    1/1     Running   0          93s
pod/segment-a-0                         1/1     Running   0          90s
pod/segment-a-1                         1/1     Running   0          90s
```

2. I need to load up some data

Pivotal

Greenplum can access it all.





3. I need to run a complex query

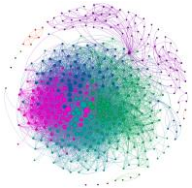
Pivotal

User question

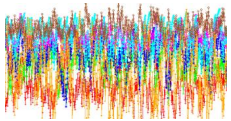
“Find anyone whose names sound like ‘Steve’ or “Peggy” and who were at WW2 and knows each other directly and have withdrawn at least $20 after 1945 less than 20 KM from a reference latitude and longitude (Peggy’s parents)”

An interesting Challenge!!!

Language Analytics
Are these the same words?



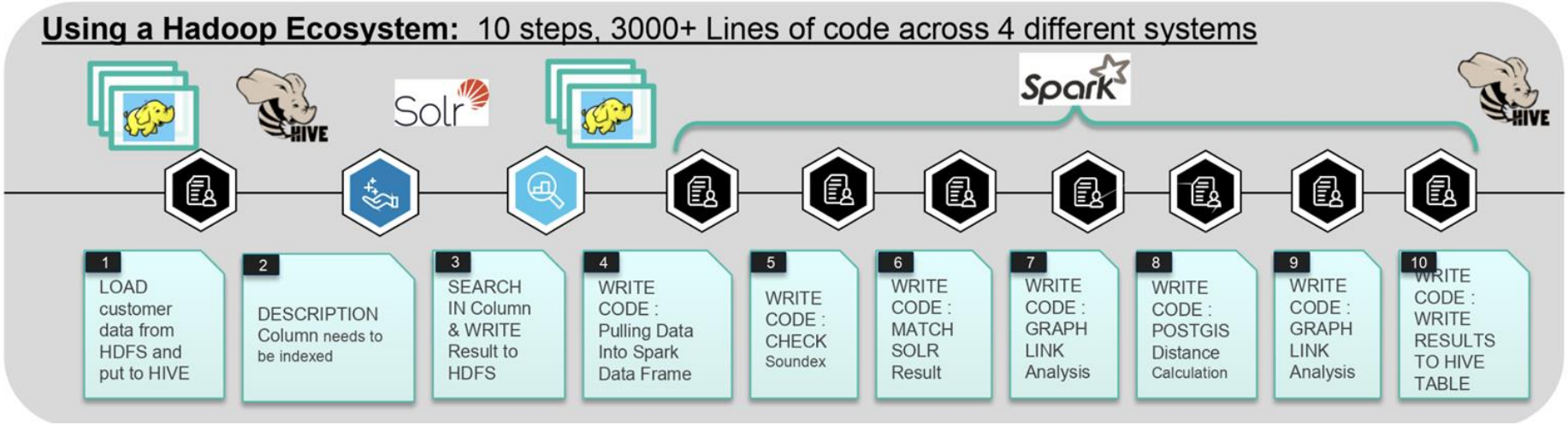
Graph Analytics
Who do they know?



Time Series
When did it happen?

Geospatial Analytics
Where are they?

We have Legacy Data Lake/Swamp



32 Lines one query!!!

Pivotal

“Find anyone whose names *sound like* ‘Steve’ or ‘Peggy’ and who were at WW2

Knows each other *directly*

GPText and Greenplum

Extract and Transform

- Fast text extraction, indexing/search
- Open source analytics with MPP processing
- Index/store metadata only, avoid data ETL
- Search-engine like syntax
- Better matching for more relevant results
- Many sources and formats, w/o complexity

Explore and Analyze

- Part of Speech Detection
- Named Entity Recognition
- Categorization (via MADLib)
- Topic Modeling (via MADLib)
- Classification/Sentiment (via MADlib, Python, R libraries)

Identify language that signals interesting behaviors and events for the use case

I really love my Ops - GPText Installed by Default

```
NAME                                                  STATUS    AGE
greenplumcluster.greenplum.pivotal.io/my-greenplum    Running   94s

NAME                                                            AGE
greenplumtextservice.greenplum.pivotal.io/my-greenplum-gptext   93s

NAME                                                          AGE
greenplumpxfservice.greenplum.pivotal.io/my-greenplum-pxf     94s

NAME                                    READY   STATUS    RESTARTS   AGE
pod/greenplum-operator-7fbffdcf64-w6vzw 1/1     Running   0          2d7h
pod/master-0                            1/1     Running   0          90s
pod/master-1                            1/1     Running   0          90s
pod/my-greenplum-gptext-solr-0          1/1     Running   0          93s
pod/my-greenplum-gptext-zookeeper-0     1/1     Running   0          93s
pod/my-greenplum-gptext-zookeeper-1     1/1     Running   0          77s
pod/my-greenplum-gptext-zookeeper-2     1/1     Running   0          53s
pod/my-greenplum-pxf-d5489784b-rhgts    1/1     Running   0          93s
pod/my-greenplum-pxf-d5489784b-sst9n    1/1     Running   0          93s
pod/segment-a-0                         1/1     Running   0          90s
pod/segment-a-1                         1/1     Running   0          90s
```

- Installed by automatically
- Scale GPText resources independently of GPDB
- Running 3 instances.

"Find anyone whose names sound like 'Steve' or "Peggy" and who were at WW2 and knows each other directly and have withdrawn at least $20 after 1945 less than 20 KM from a reference latitude and longitude (Peggy's parents)"

Greenplum Fuzzy String Match function **Soundex()** to know if people name sounds like 'Steve' or 'Peggy'

GPText.search() function is used to know if both people were at WW2

```
drop function if exists get_people(text,text,integer,integer,float,float);
CREATE FUNCTION get_people(text,text,integer,integer,float,float) RETURNS integer
AS $$
declare
linkchk integer; v1 record; v2 record;
begin
 execute 'truncate table results;';
 for v1 in select distinct a.id,a.firstname,a.lastname,amount,tran_date,c.lat,c.lng,address,a.description,d.score from people a,transactions b,location c,
    (SELECT w.id, q.score FROM people w, gptext.search(TABLE(SELECT 1 SCATTER BY 1), 'gpadmin.public.people' , 'World War 2', null) q
     WHERE (q.id::integer) = w.id order by 2 desc) d
     where soundex(firstname)=soundex($1) and a.id=b.id and amount > $3 and (extract(epoch from tran_date) - extract(epoch from now()))/3600 < $4

and st_distance_sphere(st_makepoint($5, $6),st_makepoint(c.lng, c.lat))/1000.0 <= 20.0 and b.locid=c.locid and a.id=d.id
 loop
   for v2 in select distinct a.id,a.firstname,a.lastname,amount,tran_date,c.lat,c.lng,address,a.description,d.score from people a,transactions b,location c,
    (SELECT w.id, q.score FROM people w, gptext.search(TABLE(SELECT 1 SCATTER BY 1), 'gpadmin.public.people' , 'Pivotal', null) q
     WHERE (q.id::integer) = w.id order by 2 desc) d
     where soundex(firstname)=soundex($2) and a.id=b.id and amount > $3 and extract(year from tran_date) > $4
    and st_distance_sphere(st_makepoint($5, $6),st_makepoint(c.lng, c.lat))/1000.0 <= 2.0 and b.locid=c.locid and a.id=d.id
 loop
     execute 'DROP TABLE IF EXISTS out, out_summary;';
     execute  'SELECT madlib.graph_bfs(''people'',''id'',''links'',NULL,'||v1.id||',''out'');'  ;
     select 1 into linkchk from out where dist=1 and id=v2.id;
     if linkchk is not null  then
          insert into results values (v1.id,v1.firstname,v1.lastname,v1.amount,v1.tran_date,v1.lat,v1.lng,v1.address,v1.description,v1.score);
          insert into results values (v2.id,v2.firstname,v2.lastname,v2.amount,v2.tran_date,v2.lat,v2.lng,v2.address,v2.description,v2.score);
      end if;
    end loop;
 end loop;
 return 0;
end
$$ LANGUAGE plpgsql;
--                person1 , person 2, amount, year, longtitude, latitude (in question)
select get_people('Steve','Peggy',20, 1945, 37.926868, -78.024902) ;
```

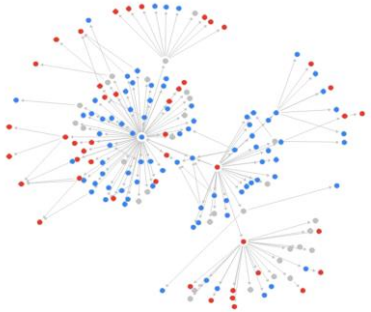
Pivotal

Graph Analytics - finding networks.

Social Network

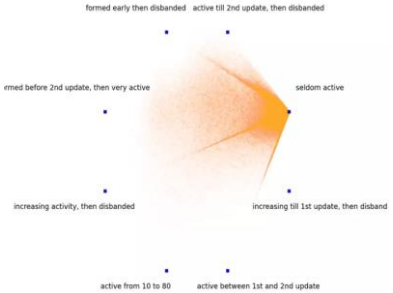
* Grandjean, M. (2016)

Epidemiology



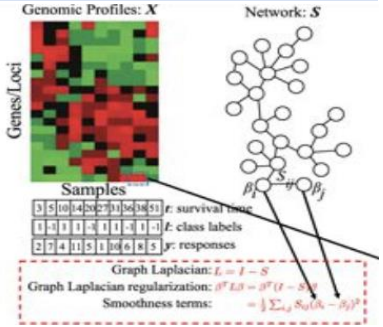
* http://www.netminer.com/community

MMO Role-Playing Game



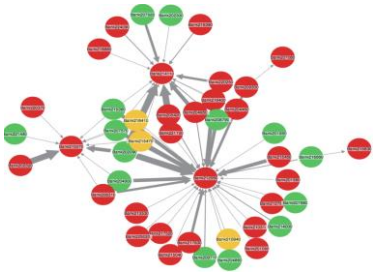
* www.researchgate.net

Chemistry



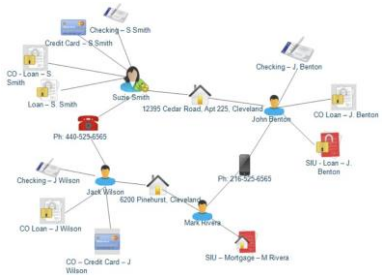
* https://www.nature.com/articles/

Bank Risk



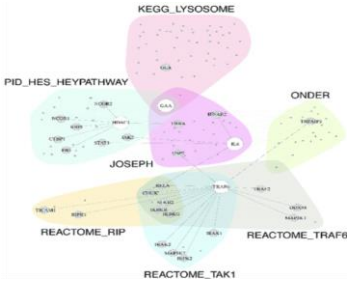
* https://cambridge-intelligence.com

1st Party Fraud



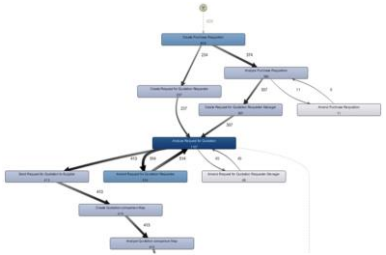
* www.infoglide.com

Gene



* www.researchgate.net

Manufacturing



* https://blog.trifinance.com

"Find anyone whose names sound like 'Steve' or "Peggy" and who were at WW2 and knows each other directly and have withdrawn at least $20 after 1945 less than 20 KM from a reference latitude and longitude (Peggy's parents)"

Greenplum Fuzzy String Match function **Soundex()** to know if people name sounds like 'Steve' or 'Peggy'

GPText.search() function is used to know if both people were at WW2

```
drop function if exists get_people(text,text,integer,integer,float,float);
CREATE FUNCTION get_people(text,text,integer,integer,float,float) RETURNS integer
AS $$
declare
linkchk integer; v1 record; v2 record;
begin
 execute 'truncate table results;';
 for v1 in select distinct a.id,a.firstname,a.lastname,amount,tran_date,c.lat,c.lng,address,a.description,d.score from people a,transactions b,location c,
    (SELECT w.id, q.score FROM people w, gptext.search(TABLE(SELECT 1 SCATTER BY 1), 'gpadmin.public.people' , 'World War 2', null) q
     WHERE (q.id::integer) = w.id order by 2 desc) d
     where soundex(firstname)=soundex($1) and a.id=b.id and amount > $3 and (extract(epoch from tran_date) - extract(epoch from now()))/3600 < $4

and st_distance_sphere(st_makepoint($5, $6),st_makepoint(c.lng, c.lat))/1000.0 <= 20.0 and b.locid=c.locid and a.id=d.id
 loop
   for v2 in select distinct a.id,a.firstname,a.lastname,amount,tran_date,c.lat,c.lng,address,a.description,d.score from people a,transactions b,location c,
    (SELECT w.id, q.score FROM people w, gptext.search(TABLE(SELECT 1 SCATTER BY 1), 'gpadmin.public.people' , 'Pivotal', null) q
     WHERE (q.id::integer) = w.id order by 2 desc) d
     where soundex(firstname)=soundex($2) and a.id=b.id and amount > $3 and extract(year from tran_date) > $4
    and st_distance_sphere(st_makepoint($5, $6),st_makepoint(c.lng, c.lat))/1000.0 <= 2.0 and b.locid=c.locid and a.id=d.id
 loop
     execute 'DROP TABLE IF EXISTS out, out_summary;';
     execute  'SELECT madlib.graph_bfs(''people'',''id'',''links'',NULL,'||v1.id||',''out'');'  ;
     select 1 into linkchk from out where dist=1 and id=v2.id;
     if linkchk is not null  then
          insert into results values (v1.id,v1.firstname,v1.lastname,v1.amount,v1.tran_date,v1.lat,v1.lng,v1.address,v1.description,v1.score);
          insert into results values (v2.id,v2.firstname,v2.lastname,v2.amount,v2.tran_date,v2.lat,v2.lng,v2.address,v2.description,v2.score);
      end if;
    end loop;
 end loop;
 return 0;
end
$$ LANGUAGE plpgsql;
--                person1 , person 2, amount, year, longtitude, latitude (in question)
select get_people('Steve','Peggy',20, 1945, 37.926868, -78.024902) ;
```

Greenplum and **Apache MADlib BFS** search to know if there are direct or indirect links between people

Pivotal

withdrawn an amount > $20 after 1945
less than 20 KM from a reference latitude and longitude(Peggy’s parents)

*geocourse_gpdb - QGIS
Browser
nyc_census_blocks
nyc_census_blocks
nyc_census_sociodata
nyc_homicides
nyc_homicides
nyc_neighborhoods
nyc_neighborhoods
nyc_streets
nyc_streets
nyc_subway_stations
nyc_subway_stations
raster_overviews
spatial_ref_sys
test1
test1
test1
Layers
nyc_subway_stations
nyc_streets
Greenplum is a Geospatial Database
Type to locate (⌘K)
Coordinate 549664,4512082
Scale 0:1
Magnifier 100%
Rotation 0.0 °
Render
EPSG:4326

"Find anyone whose names sound like 'Steve' or "Peggy" and who were at WW2 and knows each other directly and have withdrawn at least $20 after 1945 less than 20 KM from a reference latitude and longitude (Peggy's parents)"

```
drop function if exists get_people(text,text,integer,integer,float,float);
CREATE FUNCTION get_people(text,text,integer,integer,float,float) RETURNS integer
AS $$
declare
linkchk integer; v1 record; v2 record;
begin
 execute 'truncate table results;';
 for v1 in select distinct a.id,a.firstname,a.lastname,amount,tran_date,c.lat,c.lng,address,a.description,d.score from people a,transactions b,location c,
    (SELECT w.id, q.score FROM people w, gptext.search(TABLE(SELECT 1 SCATTER BY 1), 'gpadmin.public.people' , 'World War 2', null) q
     WHERE (q.id::integer) = w.id order by 2 desc) d
     where soundex(firstname)=soundex($1) and a.id=b.id and amount > $3 and (extract(epoch from tran_date) - extract(epoch from now()))/3600 < $4

and st_distance_sphere(st_makepoint($5, $6),st_makepoint(c.lng, c.lat))/1000.0 <= 20.0 and b.locid=c.locid and a.id=d.id
 loop
   for v2 in select distinct a.id,a.firstname,a.lastname,amount,tran_date,c.lat,c.lng,address,a.description,d.score from people a,transactions b,location c,
    (SELECT w.id, q.score FROM people w, gptext.search(TABLE(SELECT 1 SCATTER BY 1), 'gpadmin.public.people' , 'Pivotal', null) q
     WHERE (q.id::integer) = w.id order by 2 desc) d
     where soundex(firstname)=soundex($2) and a.id=b.id and amount > $3 and extract(year from tran_date) > $4
    and st_distance_sphere(st_makepoint($5, $6),st_makepoint(c.lng, c.lat))/1000.0 <= 2.0 and b.locid=c.locid and a.id=d.id
 loop
     execute 'DROP TABLE IF EXISTS out, out_summary;';
     execute  'SELECT madlib.graph_bfs(''people'',''id'',''links'',NULL,'||v1.id||',''out'');'  ;
     select 1 into linkchk from out where dist=1 and id=v2.id;
     if linkchk is not null  then
          insert into results values (v1.id,v1.firstname,v1.lastname,v1.amount,v1.tran_date,v1.lat,v1.lng,v1.address,v1.description,v1.score);
          insert into results values (v2.id,v2.firstname,v2.lastname,v2.amount,v2.tran_date,v2.lat,v2.lng,v2.address,v2.description,v2.score);
      end if;
    end loop;
 end loop;
 return 0;
end
$$ LANGUAGE plpgsql;
--                person1 , person 2, amount, year, longtitude, latitude (in question)
select get_people('Steve','Peggy',20, 1945, 37.926868, -78.024902) ;
```

Greenplum Fuzzy String Match function **Soundex()** to know if people name sounds like 'Steve' or 'Peggy'

Amount > $20

Greenplum and **Apache MADlib BFS** search to know if there are direct or indirect links between people

Greenplum **Time functions** to calculate amount withdrawn time after the year1945

Greenplum POSTGIS functions **st_distance_sphere() and st_makepoint()** calculate distance between bank location and reference latitude, longitude < 20 KM

Pivotal

4. I need seamless Day 2 operations

Pivotal

Node Fails - GPDB Auto-recovers

No manual recovery needed;
Just re-run the query!

Master dies and is recovered in 34s.
Same process applies to segments.

Even if its host dies, the master (or segment) will recover on another host because of compute-storage separation.

If you use remote storage then mirrors are not required for auto-recovery.

```
gpadmin@master-0:~$ psql
psql (8.3.23)
Type "help" for help.

gpadmin=# select * from foo;
 i
---
 1
 3
 2
(3 rows)

gpadmin=# command terminated with exit code 137
```

```
NAME                                  READY   STATUS    RESTARTS   AGE
pod/greenplum-operator-795f848569-vn9c7   1/1     Running   0          48m
pod/master-0                          1/1     Running   0          34s
pod/master-1                          1/1     Running   0          47m
pod/segment-a-0                       1/1     Running   0          47m
```

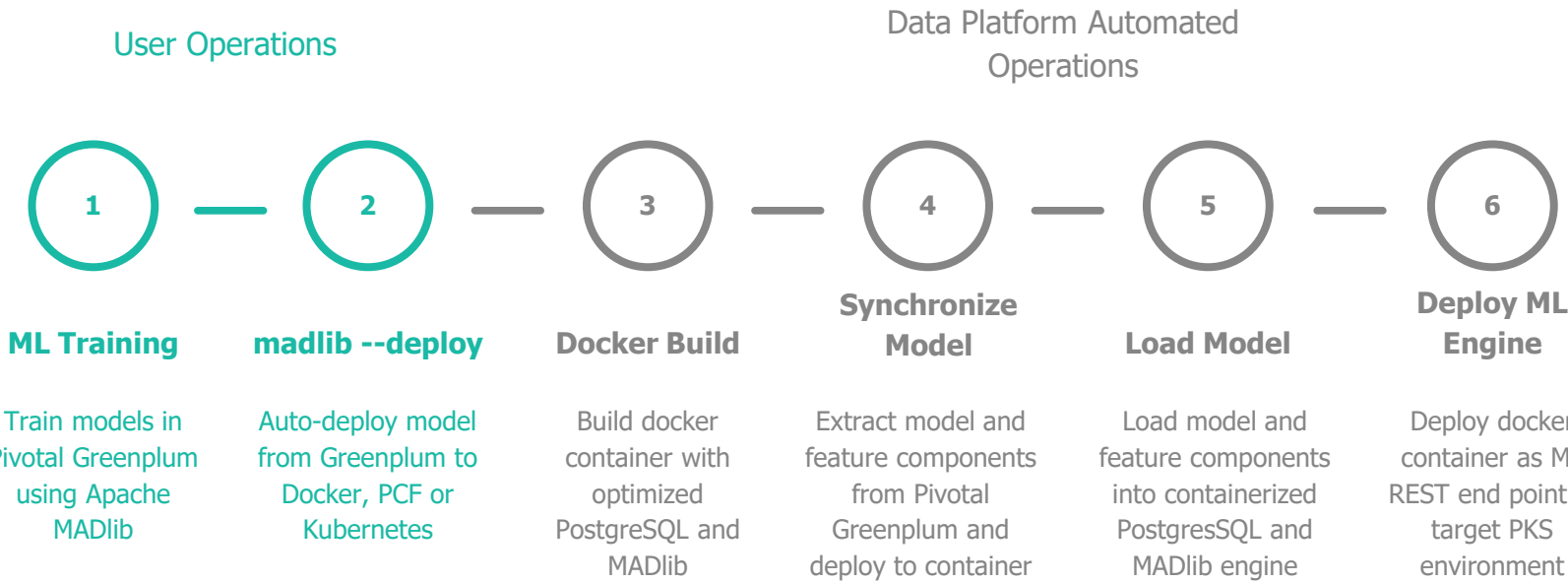
```
gpadmin@master-0:~$ psql
psql (8.3.23)
Type "help" for help.

gpadmin=# select * from foo;
 i
---
 2
 1
 3
(3 rows)

gpadmin=#
```

If I had to go into Production - Not Today :)

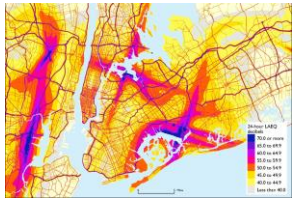
Real Time Scoring For Apache MADlib

Single command to deploy a MADlib trained model from Pivotal Greenplum / Postgres to Docker, PCF or Kubernetes

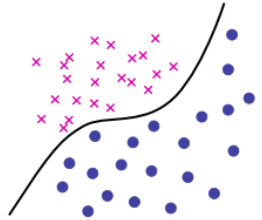
```
$ madlib --deploy
```

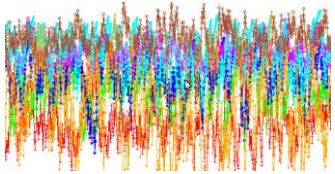


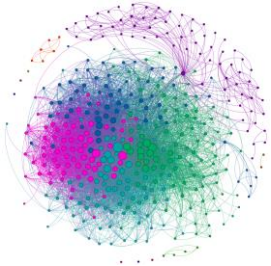
Pivotal

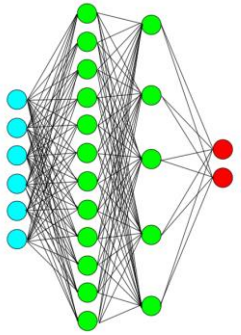


Pivotal
Greenplum®









Transforming How The World Builds Software