物联网中的分布式数据科学计算

韩　鹏

Staff Solution Architect

VMware Greenplum

韩鹏简历

曾任 深圳证券交易所“新一代监察系统MPP项目”项目经理兼架构师。

曾任 深圳证券交易所“新一代海量实时分析系统咨询项目”，首席架构咨询师。

曾任“中国首个银行业PaaS(招商银行Pivotal Cloud Foundry)项目“项目经理及架构师。

曾在中航信(雇员4000+)，担任“大数据平台(GP+HADOOP)”产品经理兼技术经理。

曾领导29人团队(任架构师及项目经理),完成中国首例从Teradata至Greenplum的企业数据仓库迁移。

专注于大数据及数据仓库领域16年，熟悉各类分布式数据平台(MPP、NoSQL、Streaming)，兼具PaaS云平台实施经验。

具备8000余人天项目管理经验，以及开发、运营大型信息系统经验。

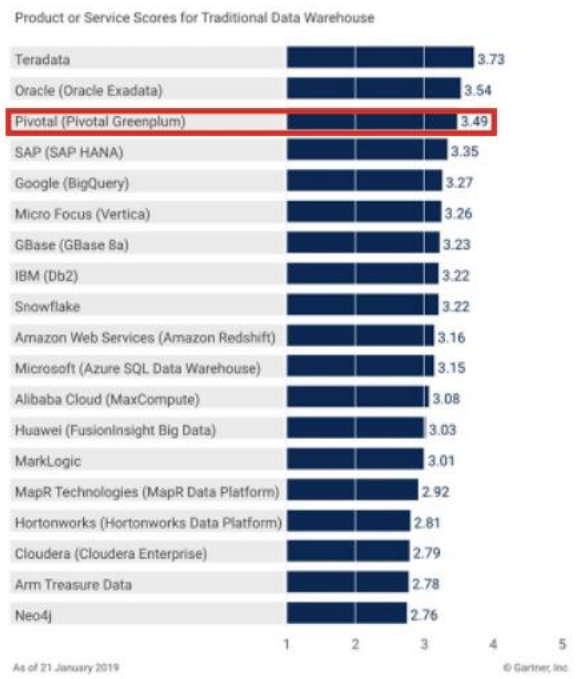
2010年起，持有工信部和人社部颁发的“系统分析师”证书。

吉林大学计算机科学与技术学院“计算机软件与理论”专业硕士，毕业于教育部“符号计算与知识工程”教育部重点实验室。

2019 Gartner《分析型数据管理解决方案核心能力》

传统数仓
(物理和逻辑强耦合高时延)

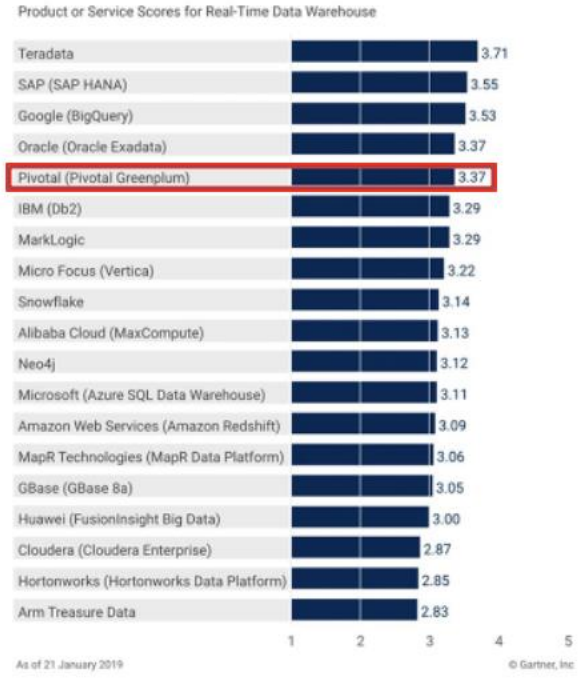
Figure 1. Vendors' Product Scores for Traditional Data Warehouse Use Case



Source: Gartner (March 2019)

实时数仓
(物理逻辑强耦合低时延)

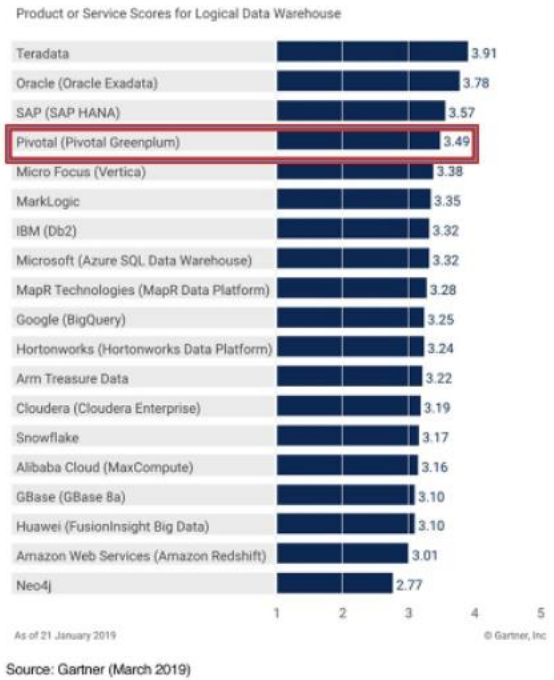
Figure 2. Vendors' Product Scores for Real-Time Data Warehouse Use Case



Source: Gartner (March 2019)

逻辑数仓
(物理松耦合)

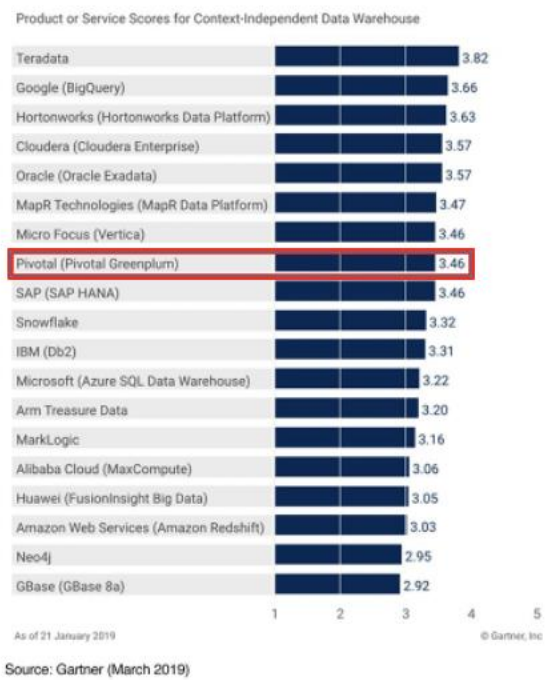
Figure 3. Vendors' Product Scores for Logical Data Warehouse Use Case



Source: Gartner (March 2019)

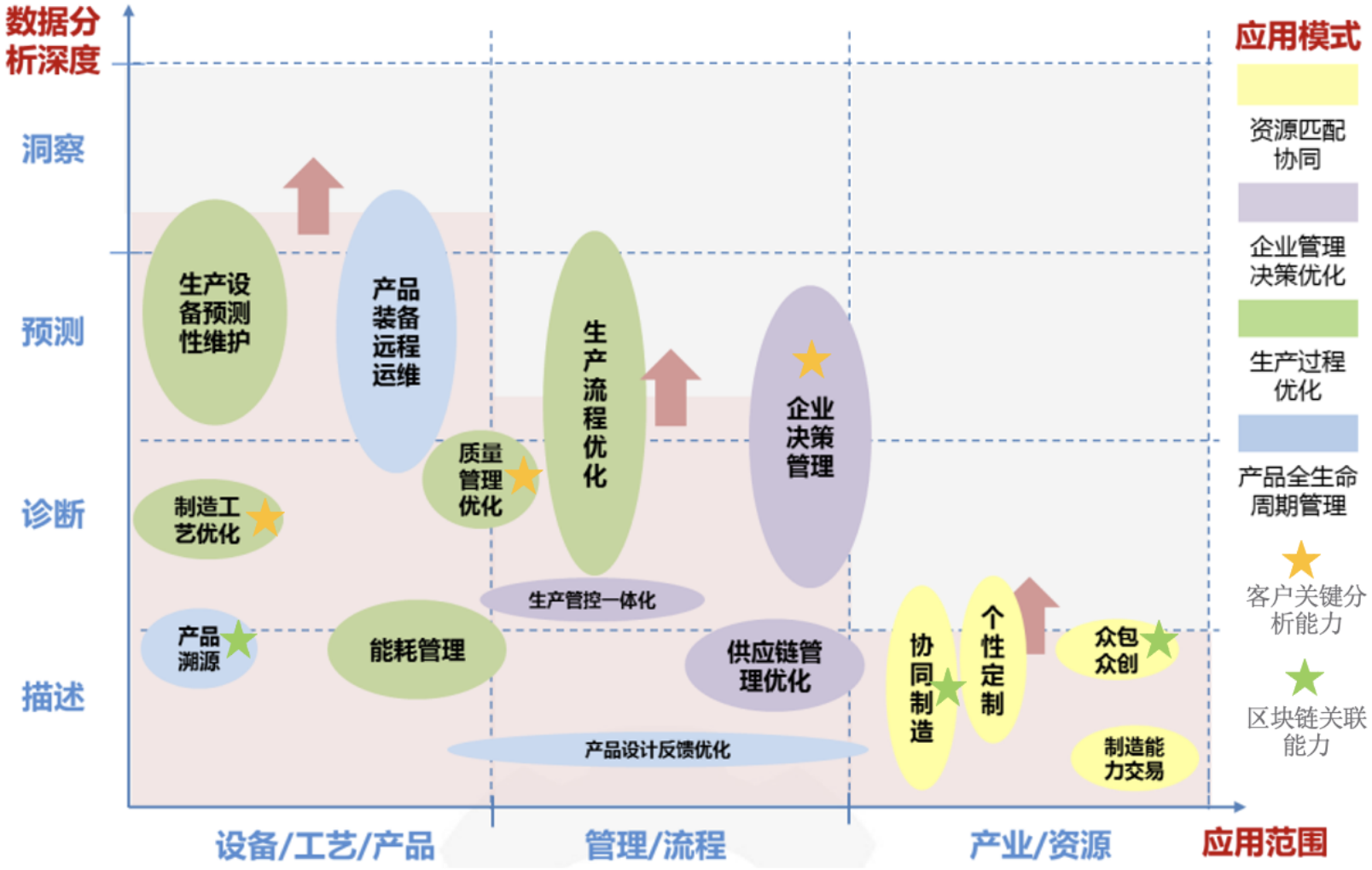
Context-Independent数仓
(逻辑松耦合)

Figure 4. Vendors' Product Scores for Context-Independent Data Warehouse Use Case

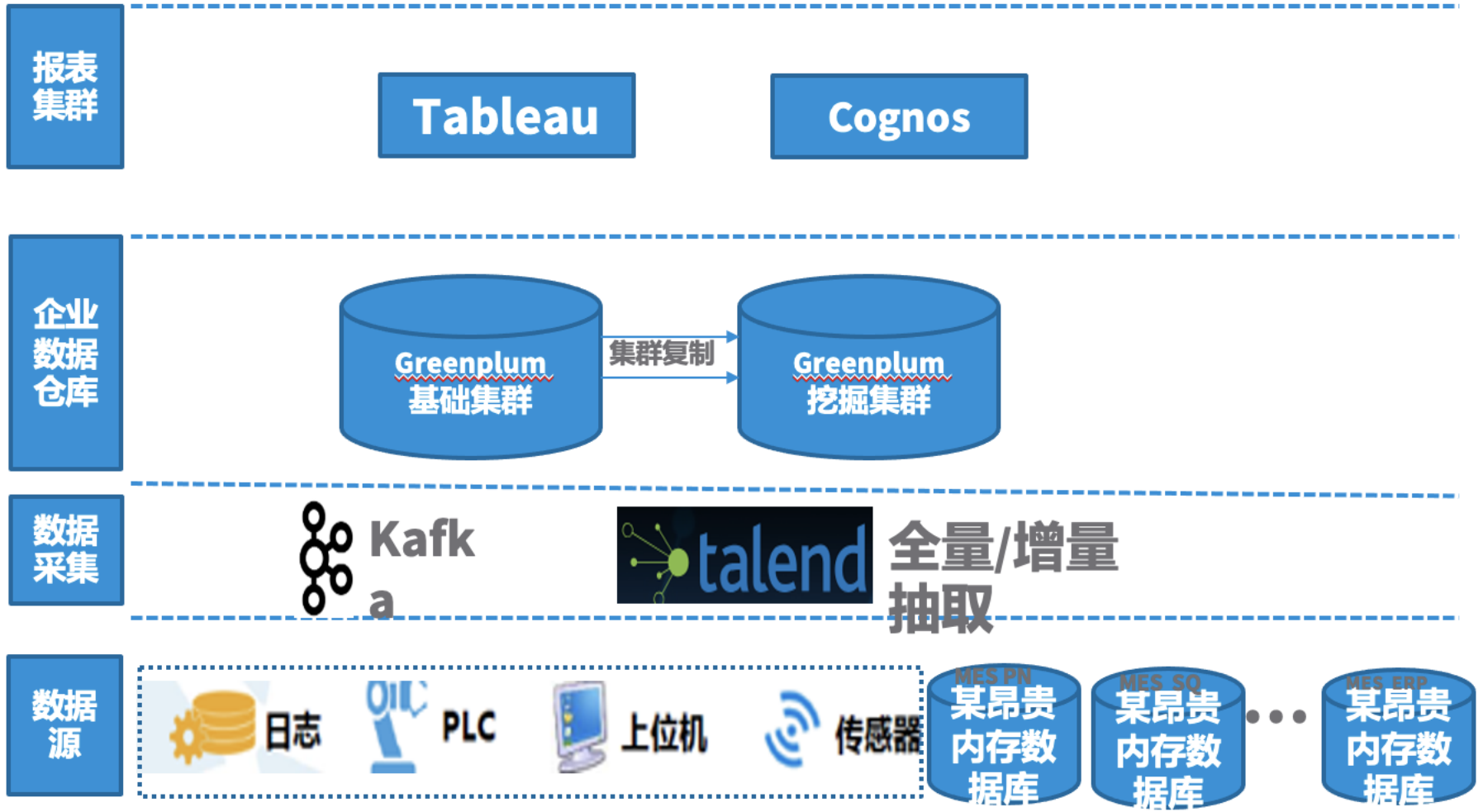


Source: Gartner (March 2019)

物联网及企业大数据分析应用分类图



国内新能源公司: 产能优化



客户挑战

- 某昂贵内存数据库无法支持大数据量的复杂计算。在追溯系统中，某昂贵内存数据库 配置4TB内存，涉及16张表关联，其中两张表为10亿和90亿规模，其他表均为10万级别的小表。运行2小时后内存溢出报错
- 某昂贵内存数据库主要定位OLTP和ERP生态系统，生态封闭，部署规模通常10TB以内。无法作为企业级数据平台。
- 某昂贵内存数据库不能直接对接hadoop、没有库内数据挖掘算法、没有地理信息数据处理能力，没有文本分析和挖掘能力。

解决方案

- 使用MPP架构，搭建企业级数据平台整合数十个某昂贵内存数据库源数据。
- 通过报表预加工处理，原来某昂贵内存数据库上无法支撑的报表查询，在Greenplum上只需要秒级返回。
- Greenplum自带kafka数据加载接口；使用Talend增量抽取某昂贵内存数据库数据。

用机器学习提升半导体智能工厂良品率

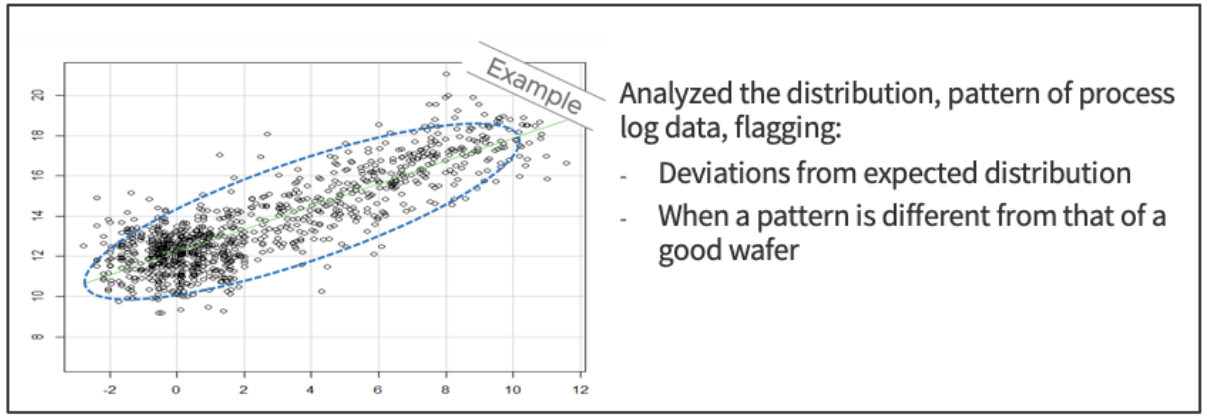
客户

业内领先的芯片厂

挑战

利用数据进一步提高生产质量和数量

在性能和可扩展性方面，足以应对来自工厂产线的多种传感器数据



成果

构建企业智能应用程序，以监控半导体制造输出的质量和产量

开发预测模型和传感器数据的可视化组件，嵌入智能应用程序

影响

产量显著提高，收入大幅增加

360 度全方位了解产线的关键组件和尺寸，包括单个传感器和晶圆级别

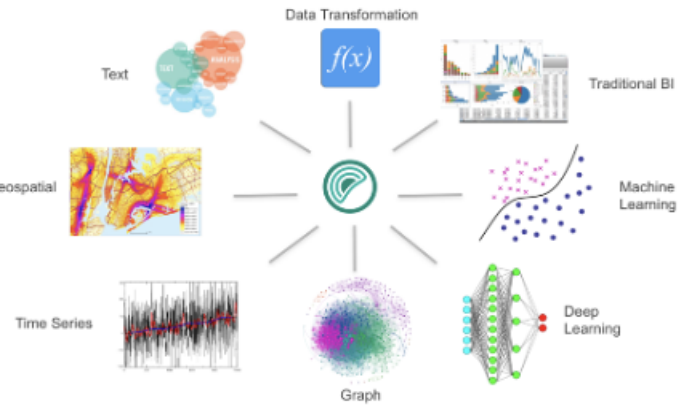
用机器学习提升半导体智能工厂良品率

智能应用流程概览

机器学习模型，预测质量问题 ID 排名

对每个 ID，支持汇总统计和可视化查看，供用户作为支持证据

用户提供反馈，诠释风险评估（例如，不予处理，标记，需要更多调查）

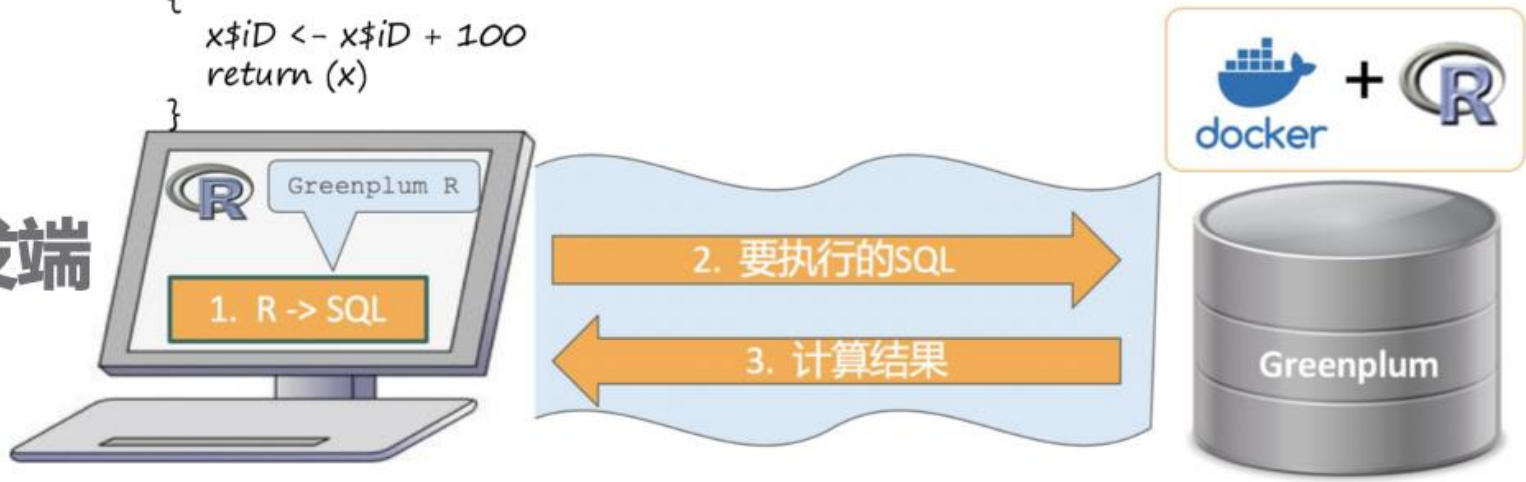


简化数据科学编程接口-并行R

- 开发端编写R代码
- R函数推送代码到GP服务器
- Docker容器中执行R函数
- 结果存在数据库或返给开发端

```
# run fn.inc in Greenplum in parallel
# fn.inc will be transferred to Greenplum and run
in parallel
db.gpApply(data, output.name=<output table>,
FUN=fn.inc, ...)

# a normal R function
fn.inc <- function(x)
{
   x$iD <- x$iD + 100
   return (x)
}
```



- 数据无需离开GP
- 将代码推送到GP

日用高效算法: 机器学习, 图分析, 统计等

MADlib Functions

监督学习

Neural Networks
Support Vector Machines (SVM)
Conditional Random Field (CRF)
Regression Models
- Clustered Variance
- Cox-Proportional Hazards Regression
- Elastic Net Regularization
- Generalized Linear Models
- Linear Regression
- Logistic Regression
- Marginal Effects
- Multinomial Regression
- Naïve Bayes
- Ordinal Regression
- Robust Variance

Tree Methods
- Decision Tree
- Random Forest

非监督学习

Association Rules (Apriori)
Clustering (k-Means)
Principal Component Analysis (PCA)
Topic Modelling (Latent Dirichlet Allocation)

最近邻居

- k-Nearest Neighbors

图分析

All Pairs Shortest Path (APSP)
Breadth-First Search
Hyperlink-Induced Topic Search (HITS)
Average Path Length
Closeness Centrality
Graph Diameter
In-Out Degree
PageRank and Personalized PageRank
Single Source Shortest Path (SSSP)
Weakly Connected Components

工具函数

Columns to Vector
Conjugate Gradient
Linear Solvers
- Dense Linear Systems
- Sparse Linear Systems

Mini-Batching
PMML Export
Term Frequency for Text
Vector to Columns

抽样

Balanced/ Random/ Stratified Sampling

时序分析

- ARIMA

数据类型和转换

Array and Matrix Operations
Matrix Factorization
- Low Rank
- Singular Value Decomposition (SVD)

Norms and Distance Functions
Sparse Vectors
Encoding Categorical Variables
Path Functions
Pivot
Sessionize
Stemming

统计

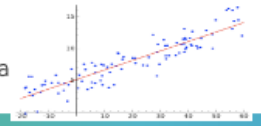
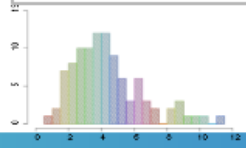
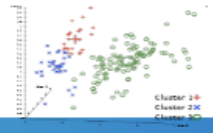
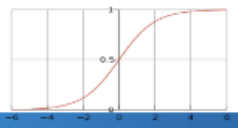
Descriptive Statistics
- Cardinality Estimators
- Correlation and Covariance
- Summary

Inferential Statistics
- Hypothesis Tests

Probability Functions

模型选择

Cross Validation
Prediction Metrics
Train-Test Split


Greenplum的分布式数据科学计算能力
自然语言处理为例

探索及分析

- 语音侦测的构成组件
- 命名实体识别
- 聚类 (利用 MADLib)
- 话题模型 (利用 MADLib)
- 分类/情感分析 (利用 MADlib, Python, R libraries)

识别“特定场景中感兴趣行为或事件”的语言信号

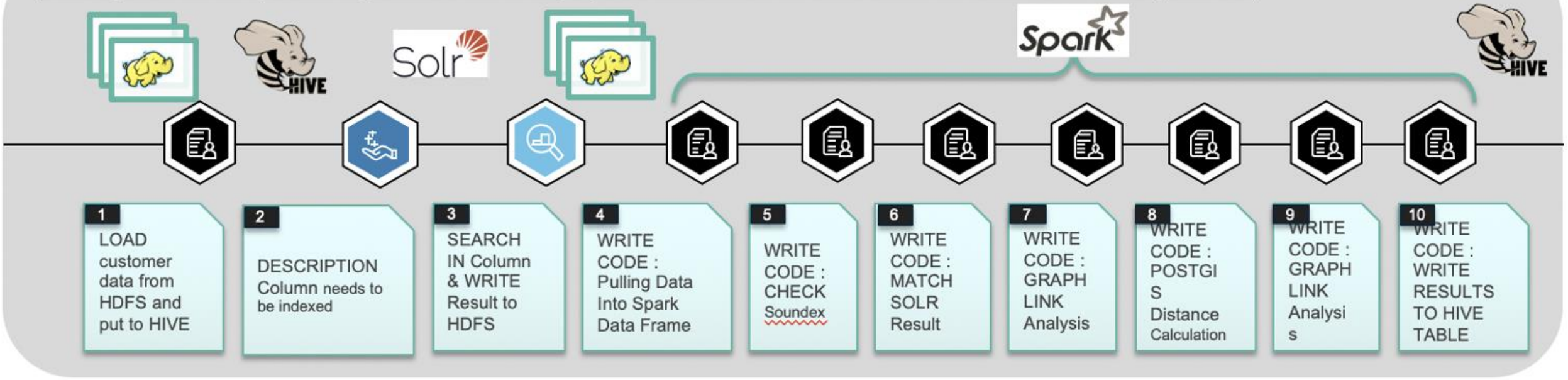
抽取及转换

- 快速文本抽取, 索引/搜索
- 开源分析MPP processing
- 仅保留 索引/存储元数据, 无需ETL
- 类搜索引擎语法
- 更好匹配更相关结果
- 众多数据源和格式, I/O复杂

高级分析场景中百倍于Hadoop的业务生产力

(多平台 自集成 过程/面向对象式编程 人工事务管理 => 单平台 声明式编程:SQL+调库函数 自动事务管理)

Using a Hadoop Ecosystem: 10 steps, 3000+ Lines of code across 4 different systems



Using Greenplum: 1 step, 1 query – 34 Lines of Code

One query – using built-in functions: Soundex (sounds like), NLP (work at same company), Machine Learning MADlib (know directly), Time (yesterday), PostGIS (within 2km)

一个基于SQL的高级分析样例

“找到这样一些人，在VMware工作，且 存在一个中间熟人 介绍可认识，且 名字 听上去类似 ‘张三’ 或 ‘李四’，且 24小时内 在 距离 VMware北京办公室经纬度坐标 最短驾车路径距离 20KM内 一台ATM机上 已 取款金额 > 1千人民币”

Find anyone who works at 'VMware' and know each other 'directly' and whose names sound like 'Zhangsan' or 'Lisi' and have withdrawn an amount > 1000 within 24 hours at an ATM less than 2 KM from reference latitude and longitude.

```
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' , 'vmware', 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 <= 2.0 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(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 <= 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, duration in hours, longtitude, latitude (in question)
select get_people('Pavan','Peter',200,24,103.912680, 1.309432) ;
```

Greenplum Fuzzy String Match function Soundex() to know if people name sounds like 'Zhangsan' or 'Lisi'

GPText.search() function is used to know if both people work at 'VMware'

Amount > 1000

Greenplum POSTGIS functions st_distance_sphere() and st_makepoint() calculate distance between ATM location and reference latitude, longitude < 2 KM

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

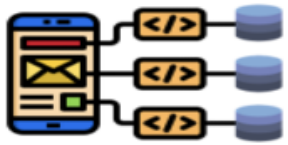
Greenplum Time functions to calculate difference in amount withdrawn time < 24 hours

Greenplum的使命: 便捷处理大数据

避免为计算分离存储

一个海量并行面向对象关系型数据库

一个时序数据库



一个OLTP OLAP混载数据库

一个基于ANSI-SQL的结构化和半结构化数据库

一个云原生数据库

一个企业搜索平台

一个大规模图分析数据库

一个图像识别引擎

一个空间地理数据库

一个R和Python的高性能并行计算集群

Thank You