DTCC

2017第八届中国数据库技术大会

DATABASE TECHNOLOGY CONFERENCE CHINA 2017

原生SQL on Hadoop引擎—Apache HAWQ 2.X 最新技术解密

马丽丽　2017.5.13

提纲

- Apache HAWQ 历史
- 系统架构
- 最新功能介绍
- 展望与未来

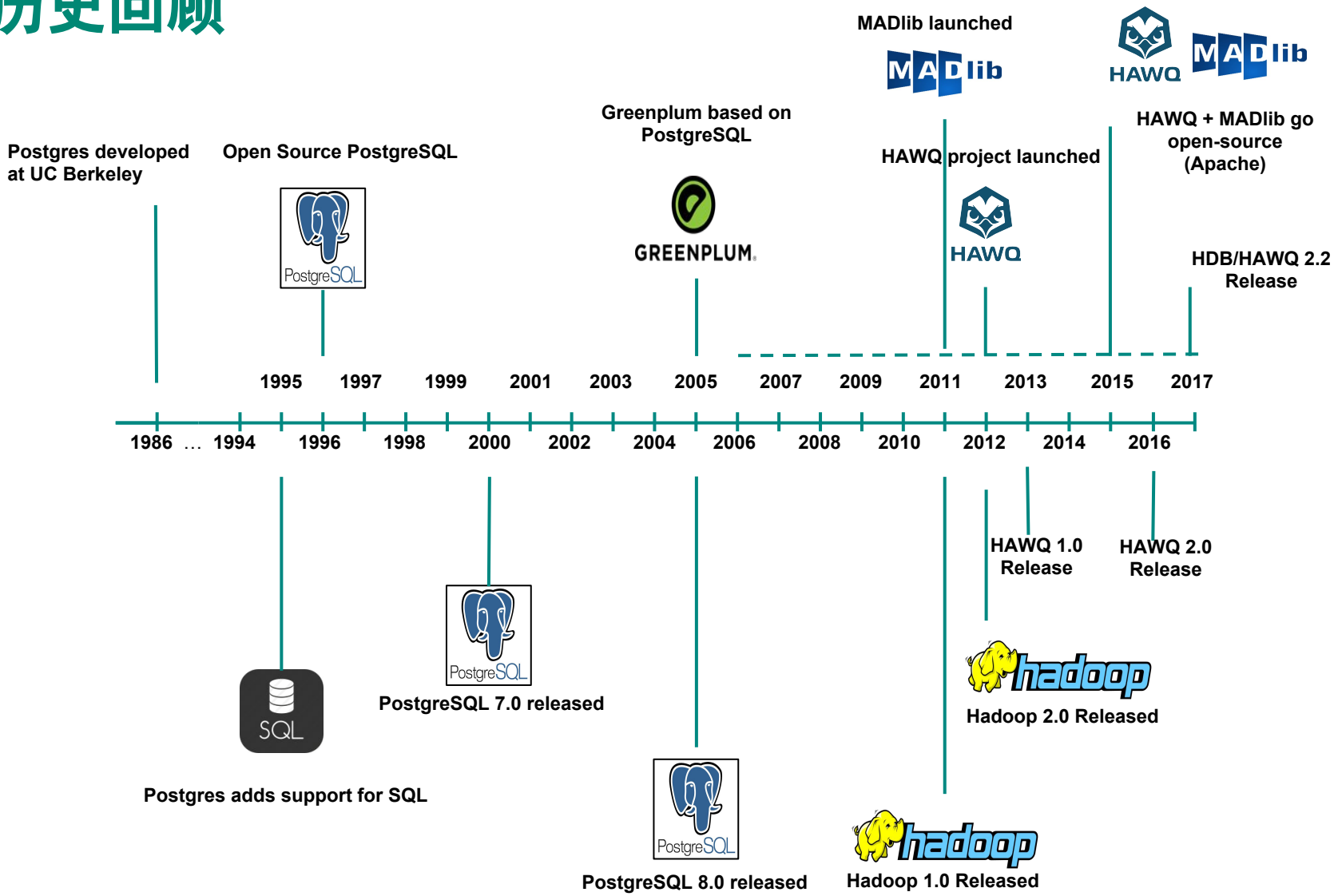
HAWQ 是什么？？？

Hadoop-native SQL query engine and advanced analytics MPP database that offers high-performance interactive query execution and machine learning to Data Analysts & Data Scientists who want to find insights in large/complex datasets.

Pivotal HDB

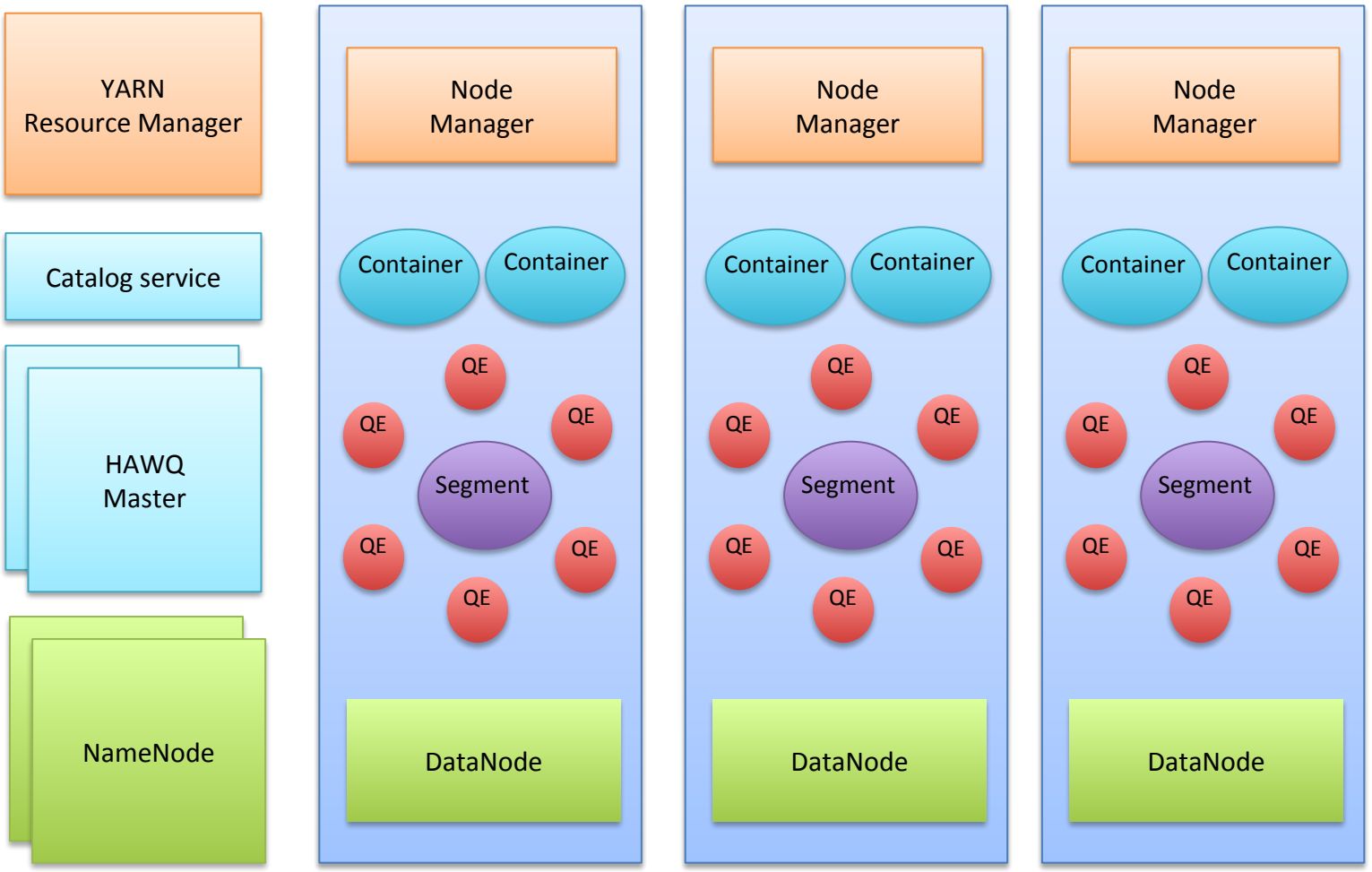
Pivotal

历史回顾



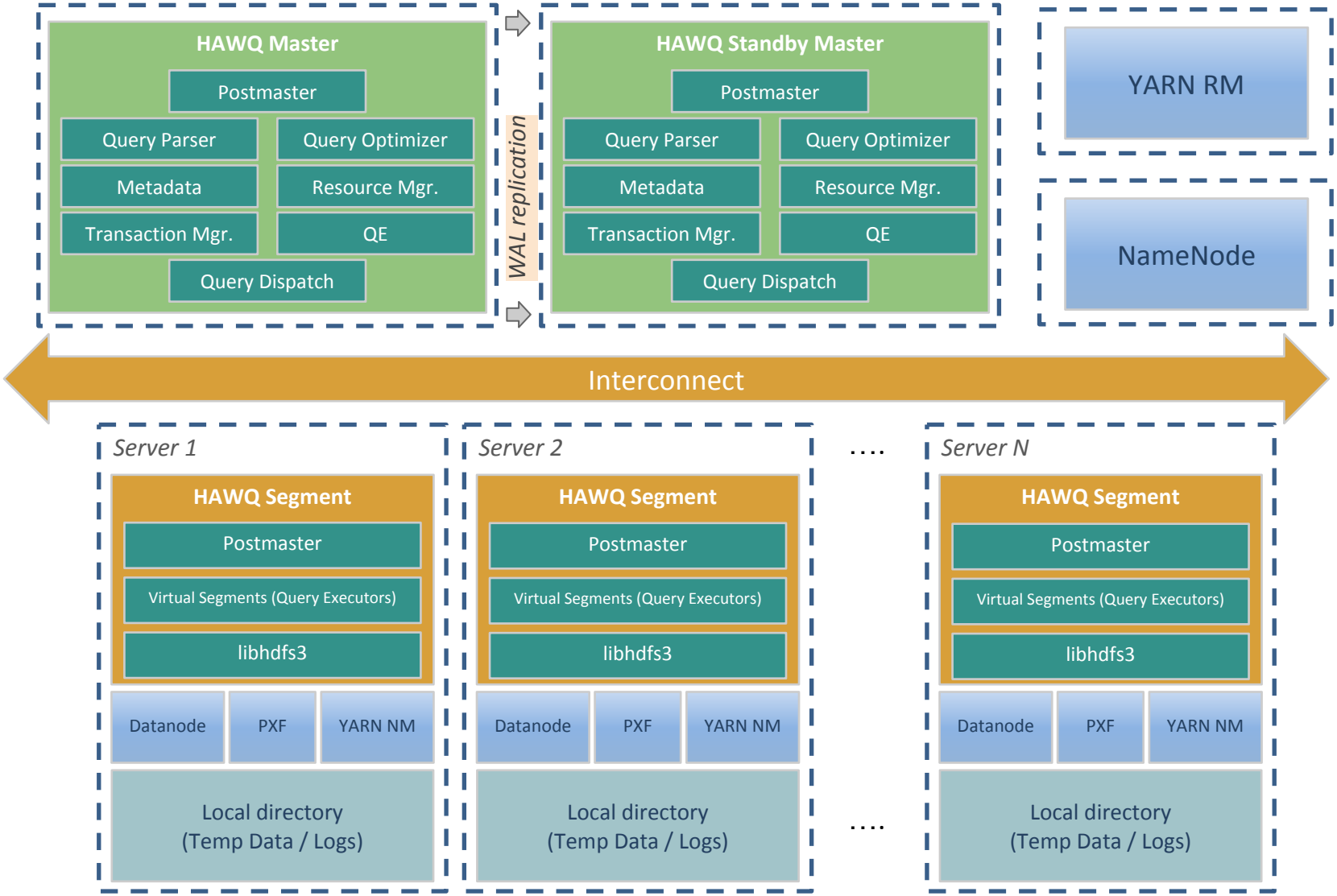
Pivotal

HAWQ架构



Pivotal

HAWQ 部件图

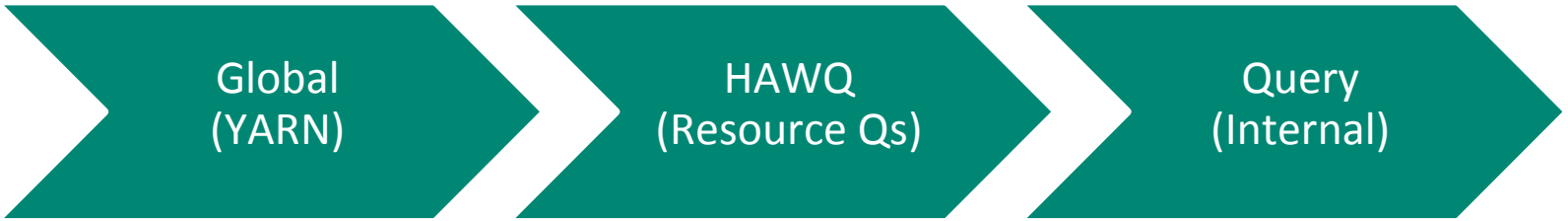


Pivotal

HAWQ 2.0概览

Areas of Enhancement	New Features
Elastic & Scalable Architecture	Elastic Runtime for Query Execution
	Per Table Directory storage (user friendly)
Hadoop-Native Integrations	Block-level Storage
	New Dispatcher + Fault Tolerance Service
Performance & Optimizations	Dynamic Cluster Expansion (no redistribute)
	YARN-Integrated 3-Tier Resource Mgmt
Simplified User Experience	HCatalog integration - Read Access
	Simpler Management via Ambari and CLI
Cloud-Readiness	HDFS Catalog Cache

分层资源管理

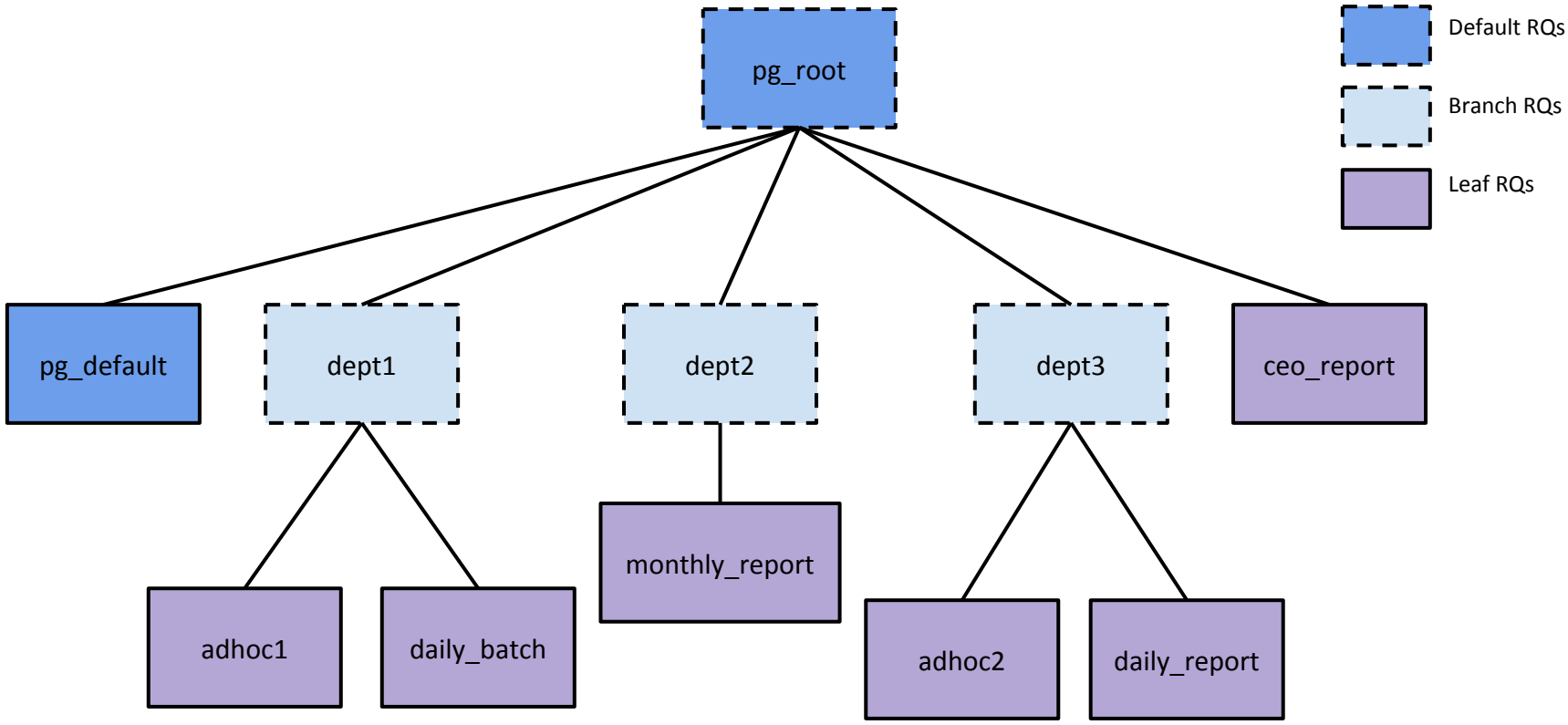


Global (YARN)	HAWQ (Resource Qs)	Query (Internal)
Cluster level	HAWQ Internal	Query level
Cluster-Admin defined	HAWQ-Admin defined	System defined
Hardware efficiency	Multi-tenancy	Query Optimization
Share with MR/Hive/+	Workload prioritization	Operator prioritization
Defined in XML	Defined in DDL	Dynamic

资源管理器

- **Responsibility**
  - Responsible for acquiring & returning CPU/Mem resources from/to YARN
  - Responsible for resource allocation among HAWQ users and queries
- **Master resource manager process**
  - Resource negotiation with YARN and resource allocation
  - Manage and maintain the resources in resource pool
  - Handle resource allocation/return RPC requests from QD (query dispatcher)
  - Fault tolerance service are in the same process
- **Segment resource manager process**
  - One HAWQ RM on each Segment
  - Negotiation with Master resource manager (for resource enforcement)
  - Fault tolerance service: Heartbeat sender

层级资源队列



创建资源队列示例

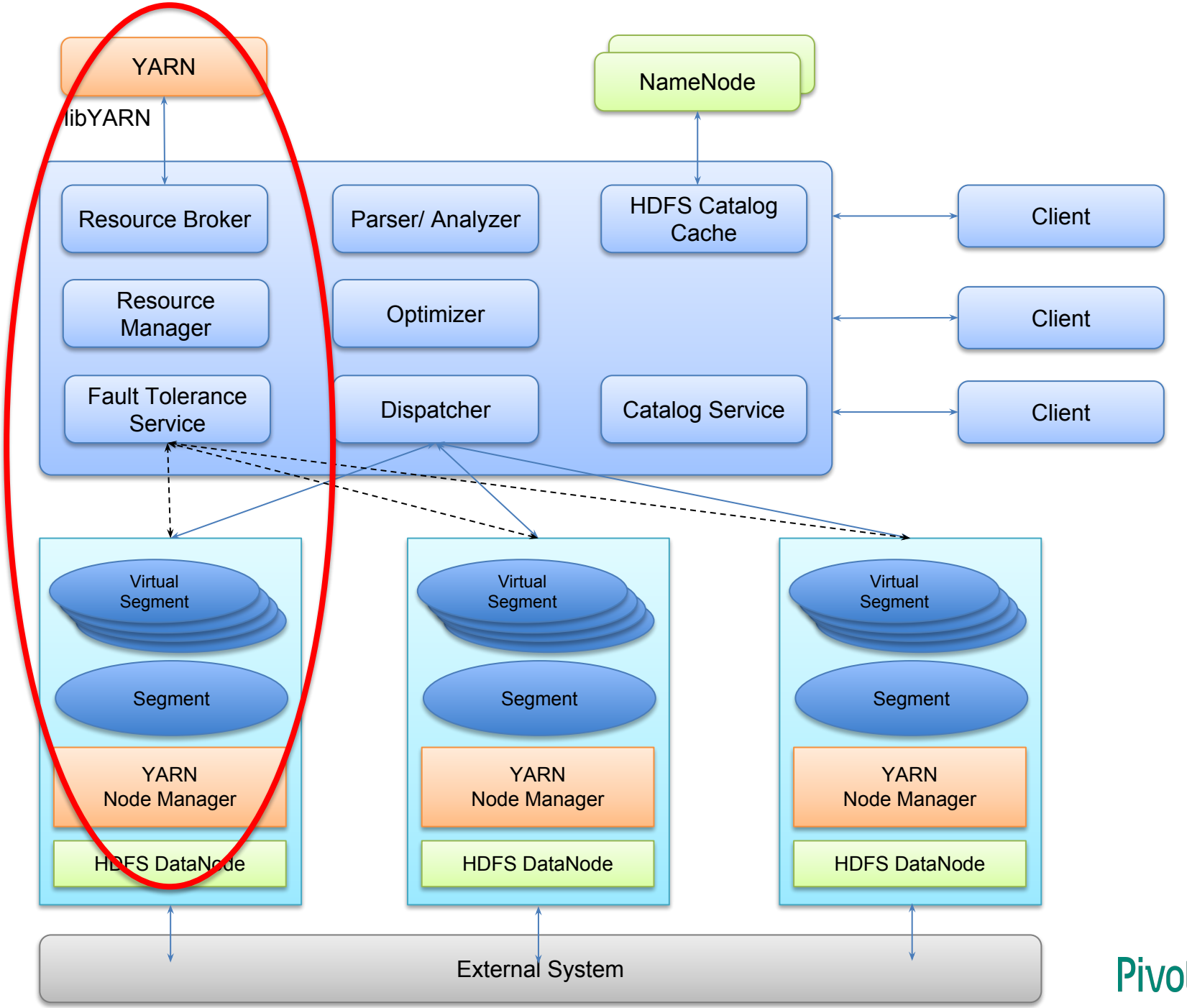
CREATE RESOURCE QUEUE name WITH (queue_attribute=value [, ...])

where queue_attribute can be:

PARENT='queue_name'
ACTIVE_STATEMENTS=integer
MEMORY_LIMIT_CLUSTER=percentage
CORE_LIMIT_CLUSTER=percentage
SEGMENT_RESOURCE_QUOTA='mem:memory_units'
RESOURCE_OVERCOMMIT_FACTOR=factor

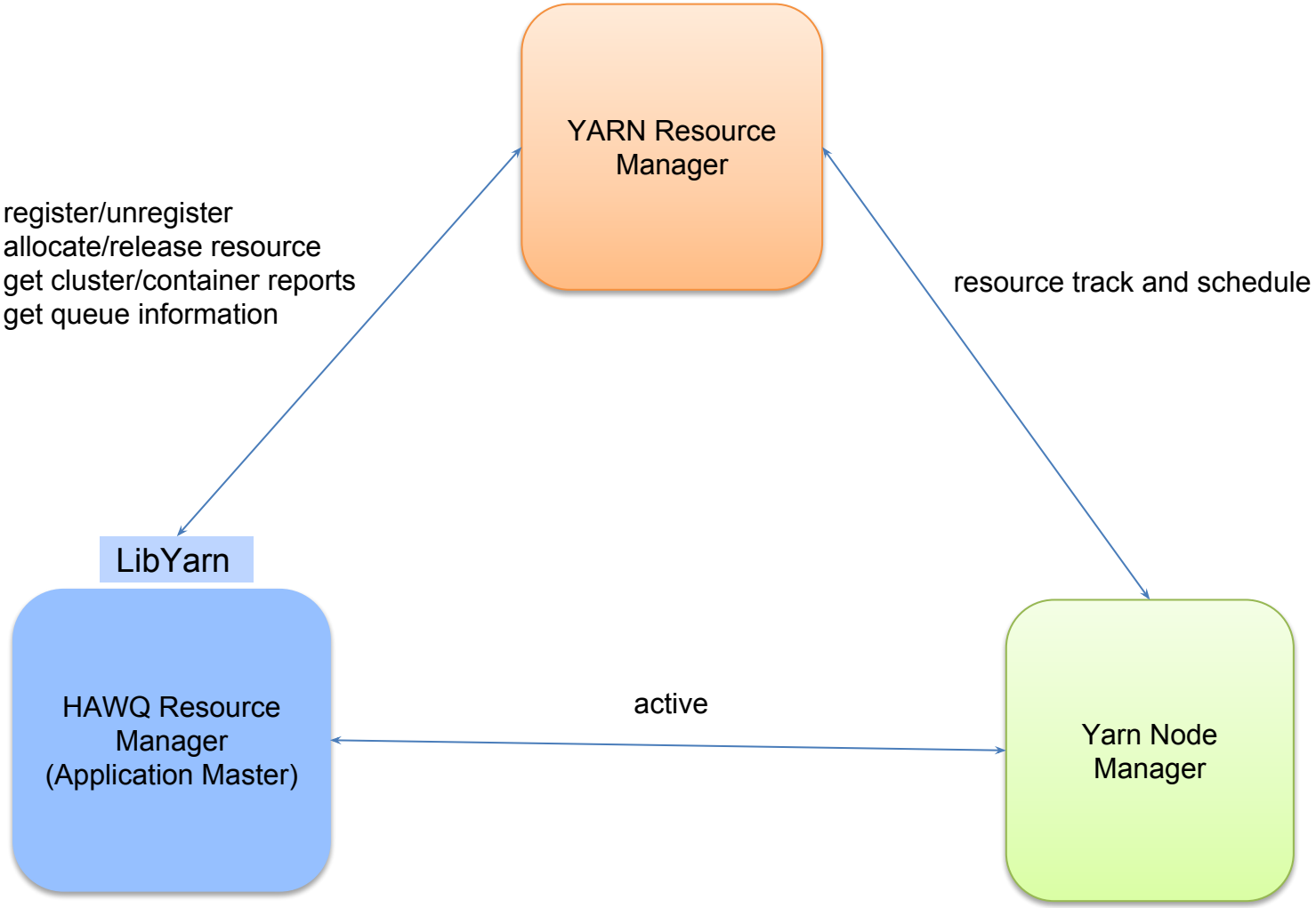
memory_units ::= {64mb|128mb|256mb|1gb|2gb}
percentage ::= integer %

Example: create resource queue test_queue_1 with (parent='pg_root', memory_limit_cluster=50%, core_limit_cluster=50%);


YARN
libYARN
NameNode
Resource Broker
Parser/ Analyzer
HDFS Catalog Cache
Client
Resource Manager
Optimizer
Client
Fault Tolerance Service
Dispatcher
Catalog Service
Client
Virtual Segment
Segment
YARN Node Manager
HDFS DataNode
Virtual Segment
Segment
YARN Node Manager
HDFS DataNode
Virtual Segment
Segment
YARN Node Manager
HDFS DataNode
External System

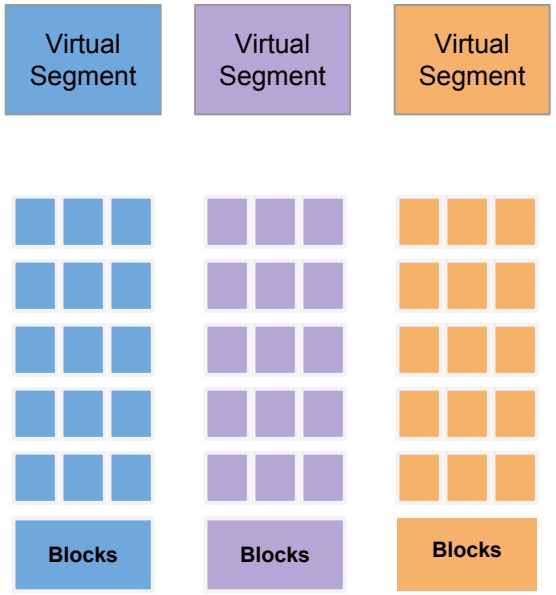
Pivotal

RM与Yarn的交互

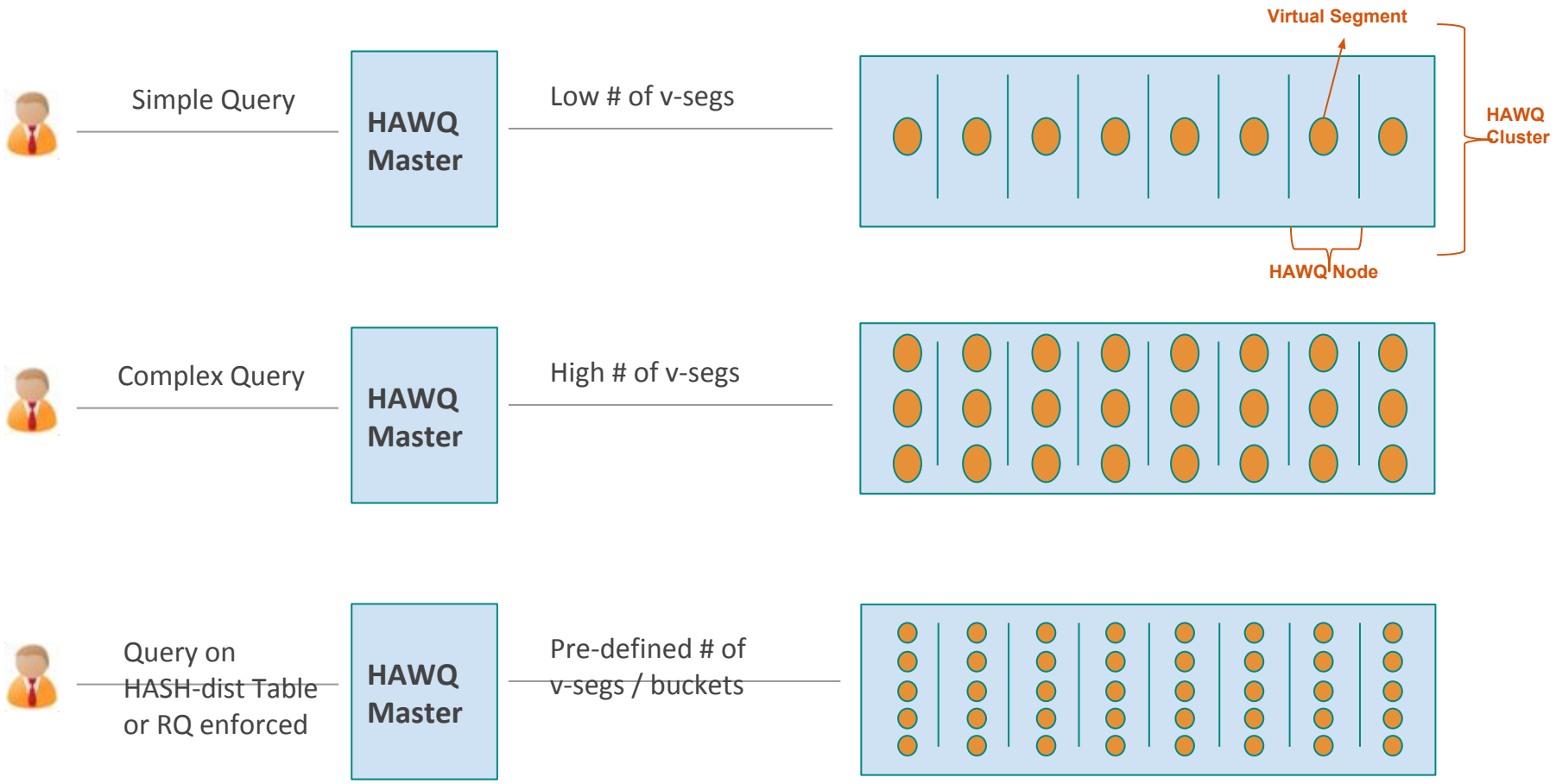


弹性查询执行

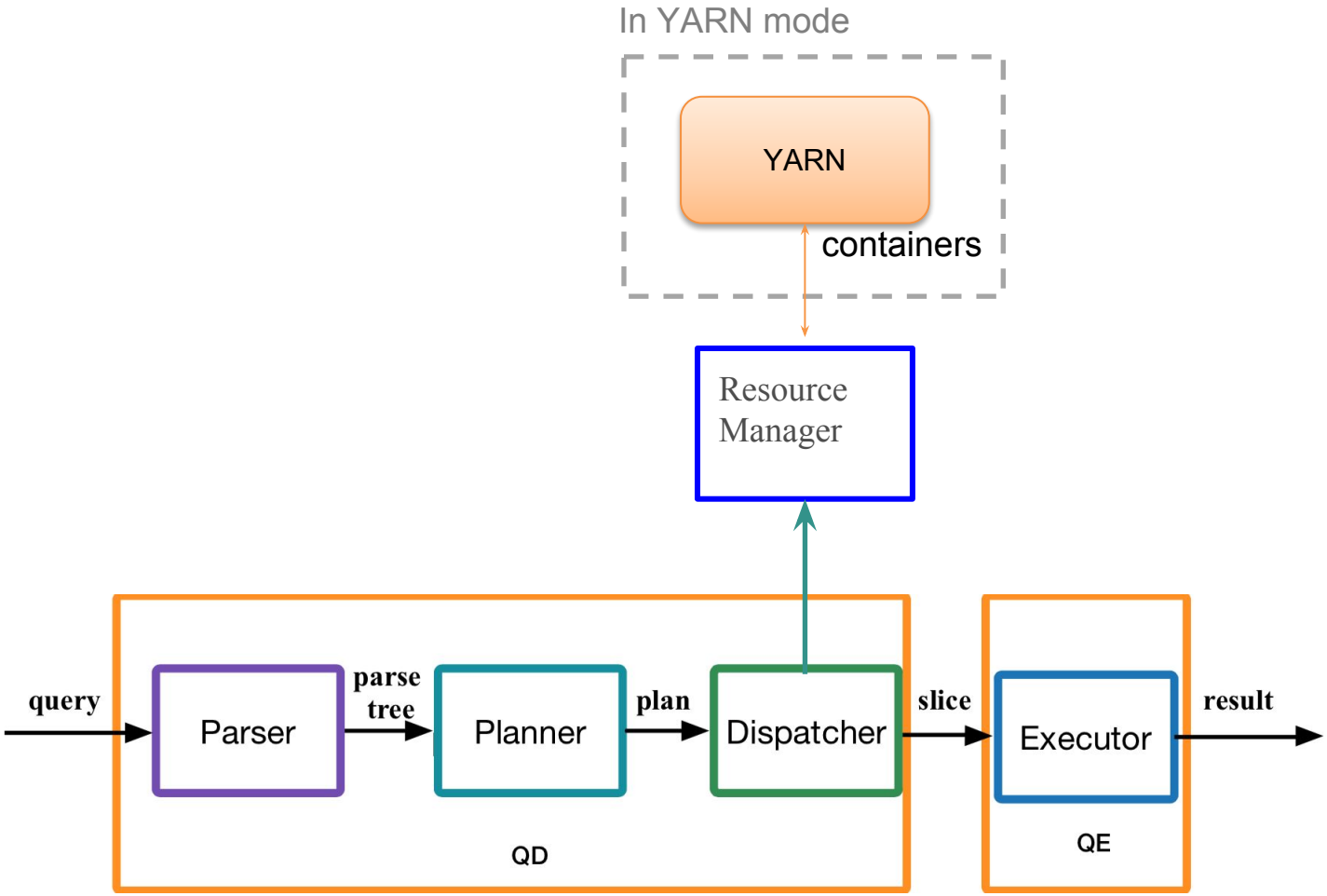
- **Query execution is dynamic & flexible**
  - Allows Scale-up/down
  - Allows Scale-in/out
  - Smart & efficient use of resources
  - More adapted to shared or cloud environments
- **How it works: “block level storage” and “virtual segments”**
  - **Block level storage support**
    - AO and Parquet
    - Scanners read granular blocks (vs files)
    - More control on task granularity
  - **Plan/Task scheduling**
    - Choose nodes that have data close
    - Dispatch query to nodes with available resources
    - Start virtual segments on demands



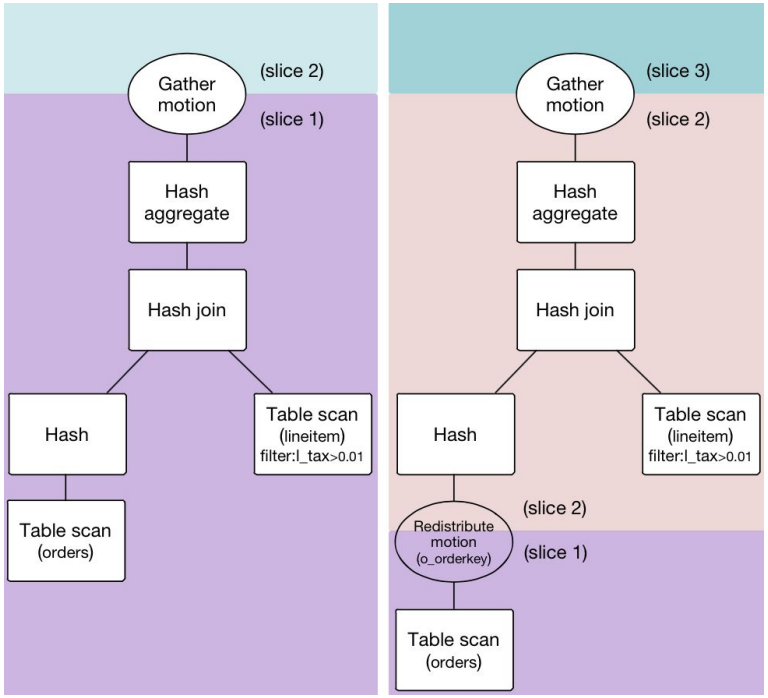
虚拟Segment



查询执行流程图



查询计划



Query Plan

- Relational operators: scans, joins, etc
- Parallel 'motion' operators

Parallel Motion Operators:

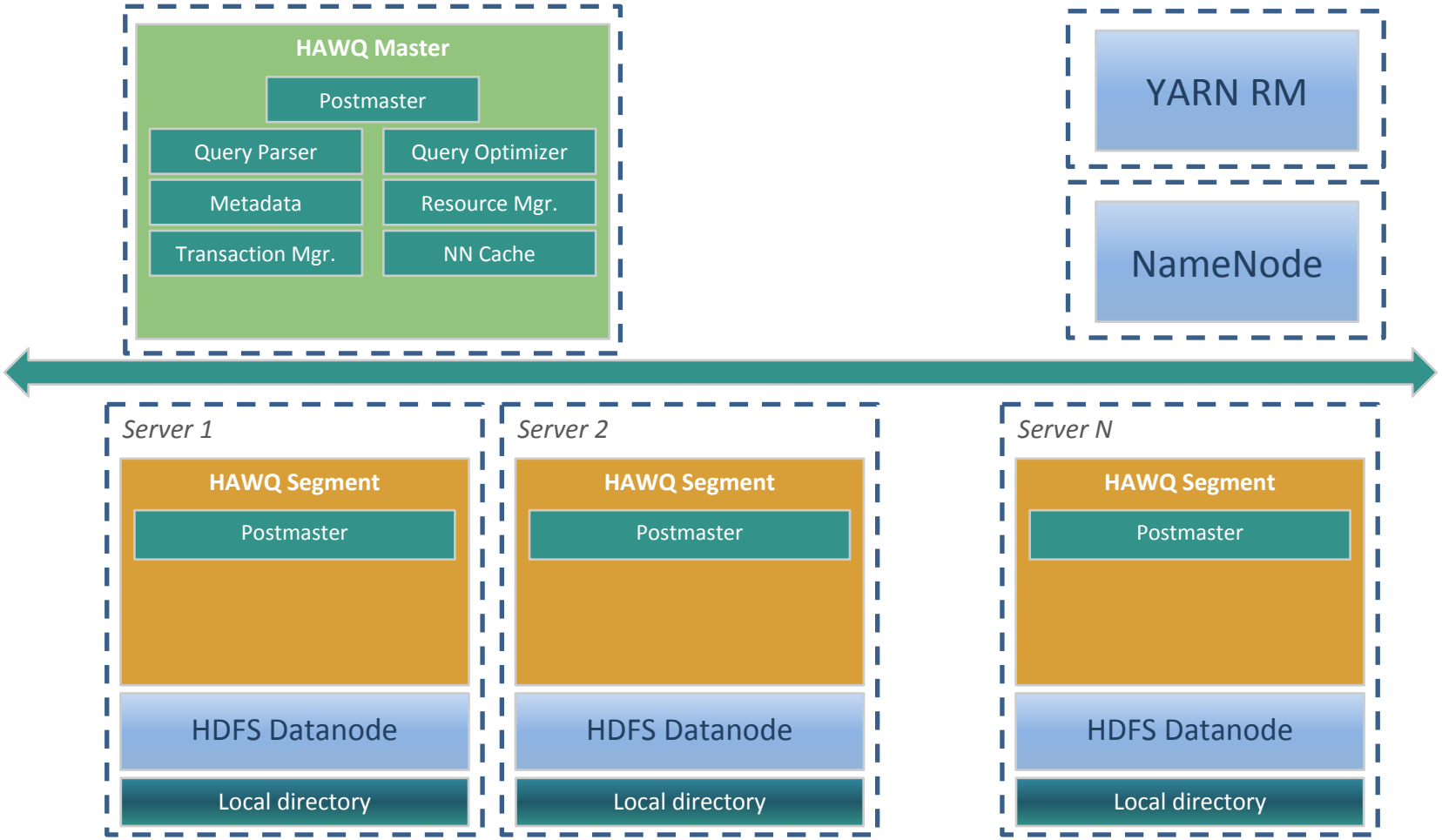
- Broadcast: Every segment sends the input tuples to all other segments
- Redistribution: Every segment rehashes tuples on a column and redistributes to the appropriate segments
- Gather: Every segment sends the input tuples to a single segment (i.e. the master)

*SELECT l_orderkey, count(l_quantity) FROM **lineitem, orders***

WHERE l_orderkey=o_orderkey AND l_tax>0.01

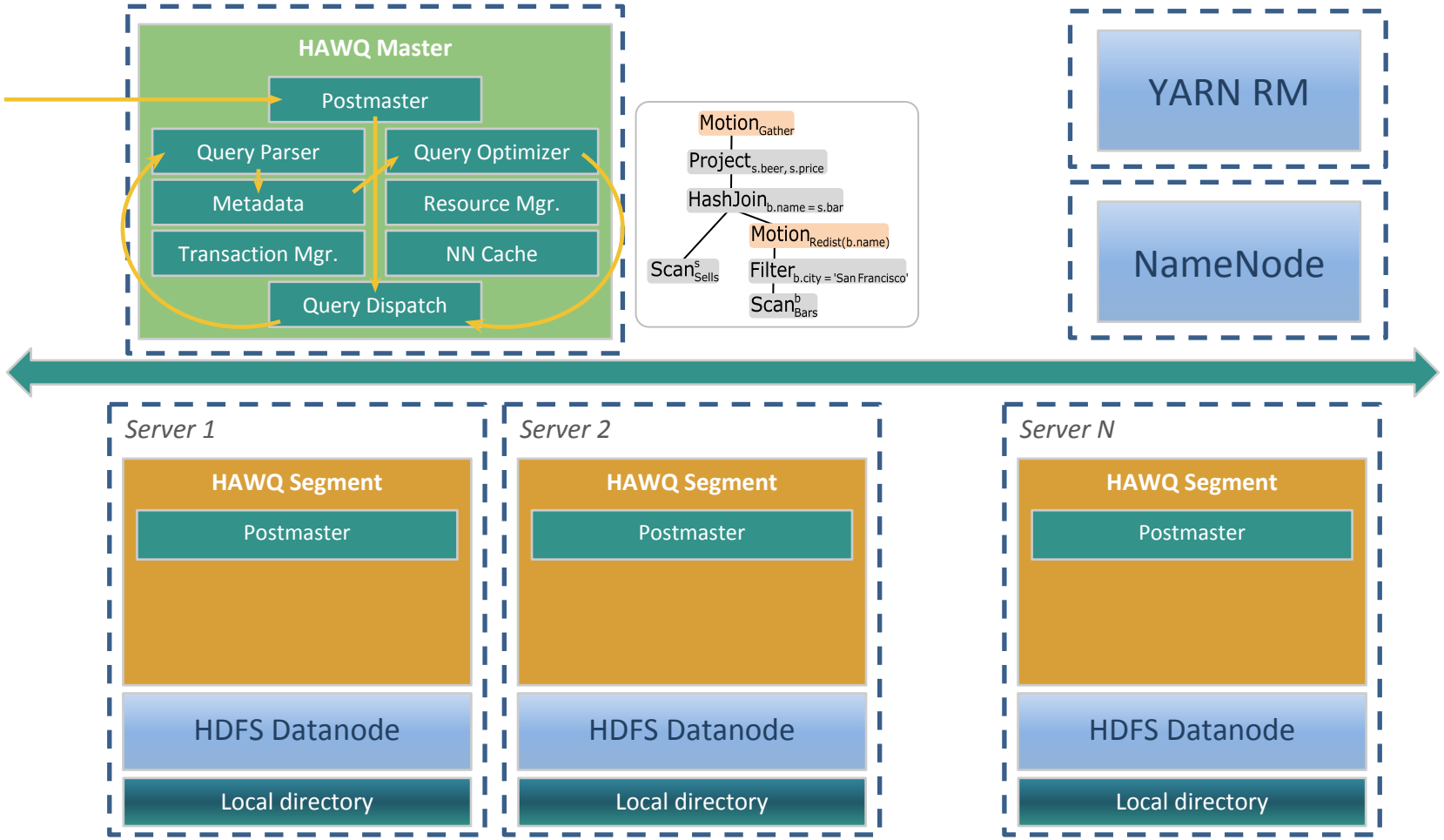
GROUP BY l_orderkey;

查询执行示例



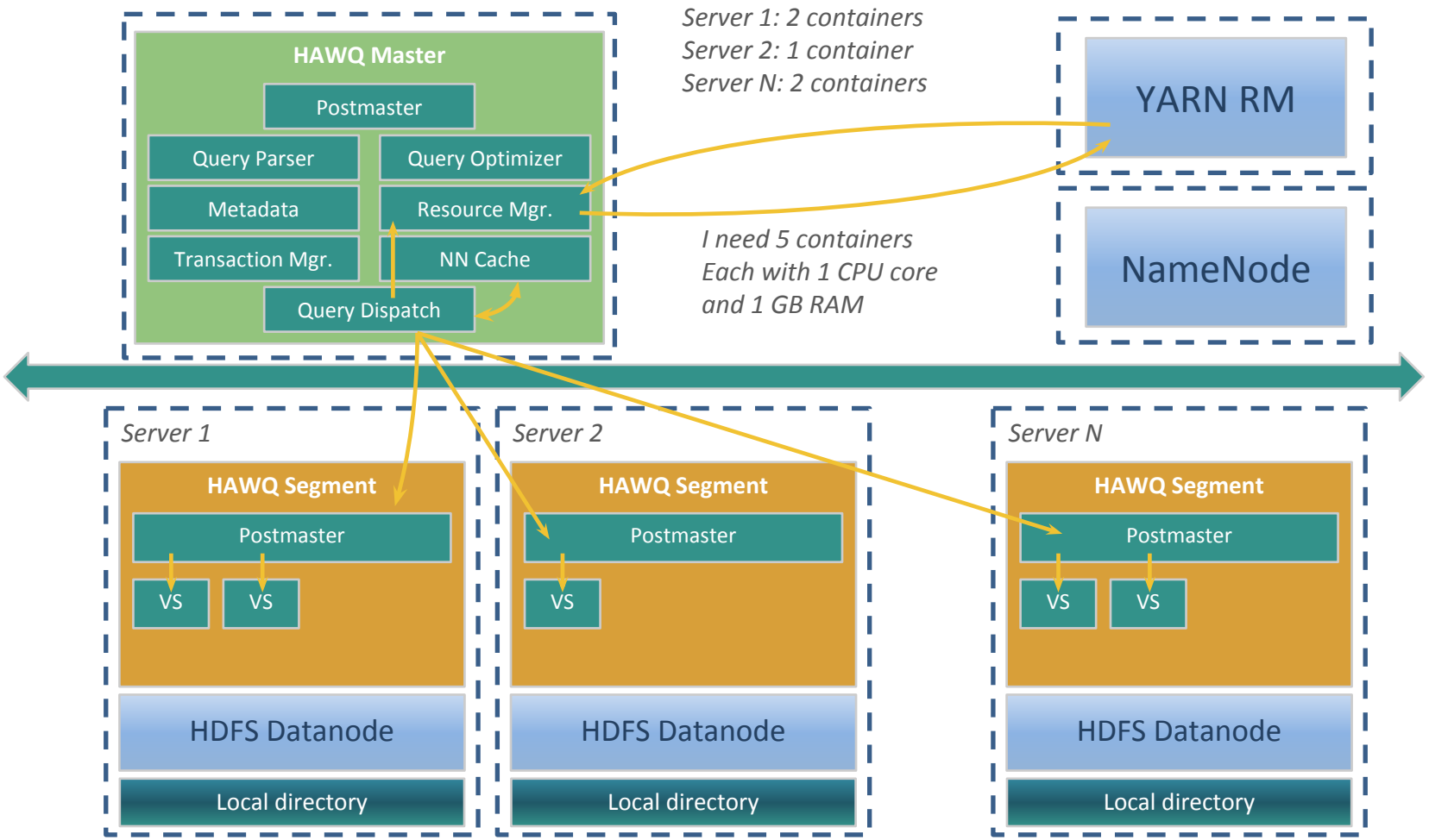
Pivotal

查询执行示例 — 计划生成



Pivotal

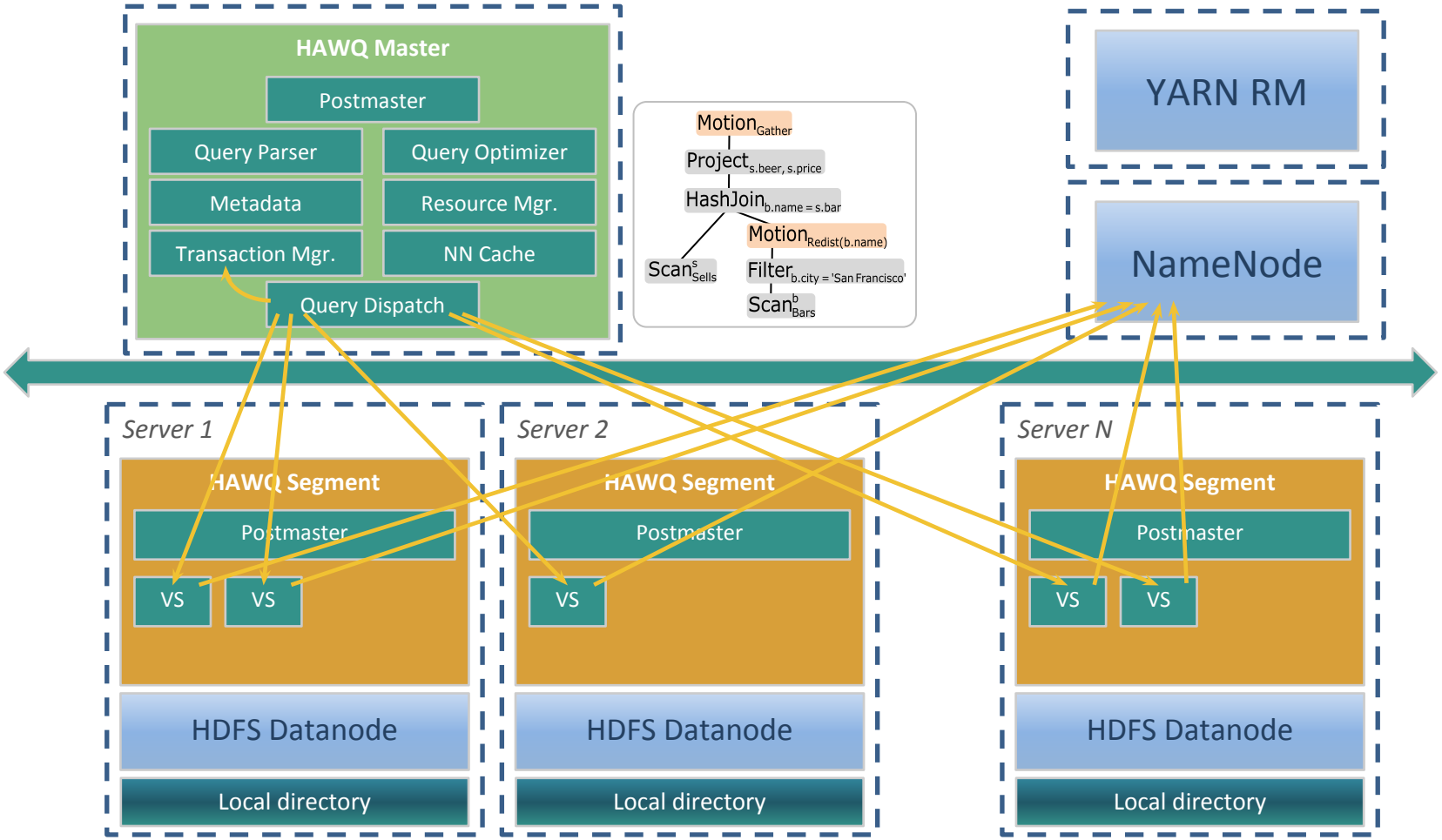
查询执行示例 – 资源申请



VS = Virtual Segment (container for Query Executors)
# of QEs in a v-seg = # of slices in a query

Pivotal

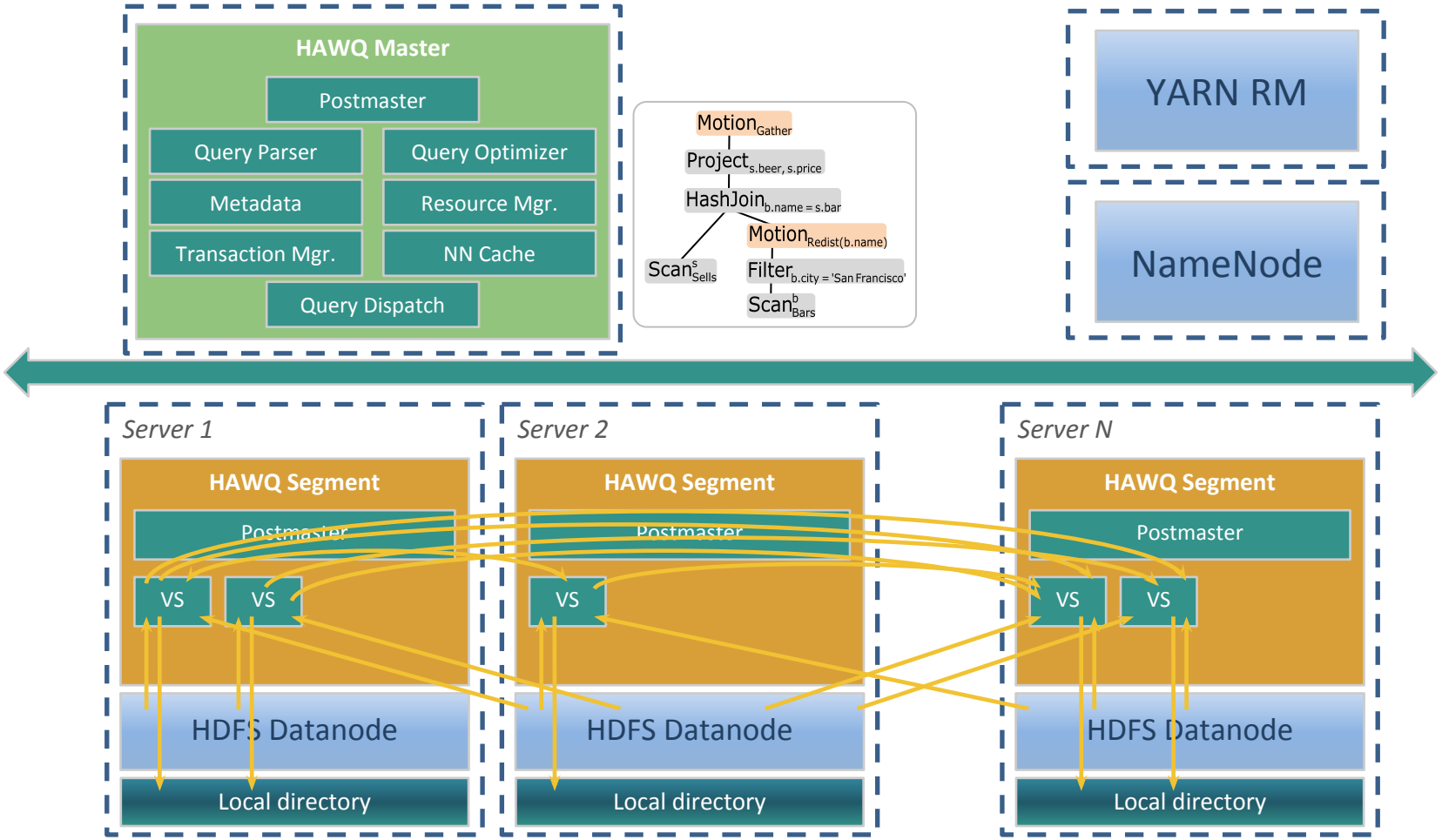
查询执行示例 —准备执行



VS = Virtual Segment (container for Query Executors)
# of QEs in a v-seg = # of slices in a query

Pivotal

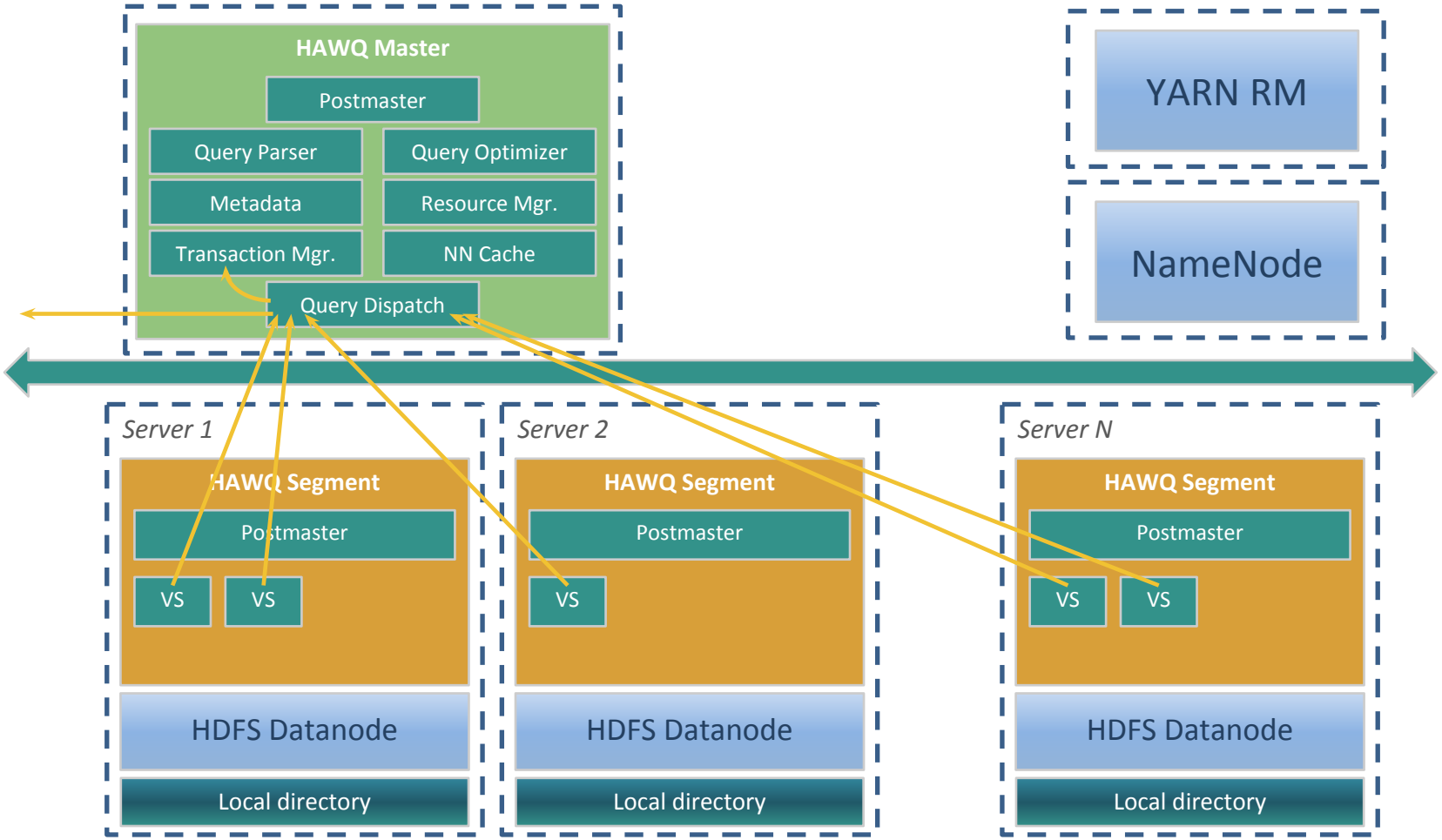
查询执行示例 – 执行



VS = Virtual Segment (container for Query Executors)
# of QEs in a v-seg = # of slices in a query

Pivotal

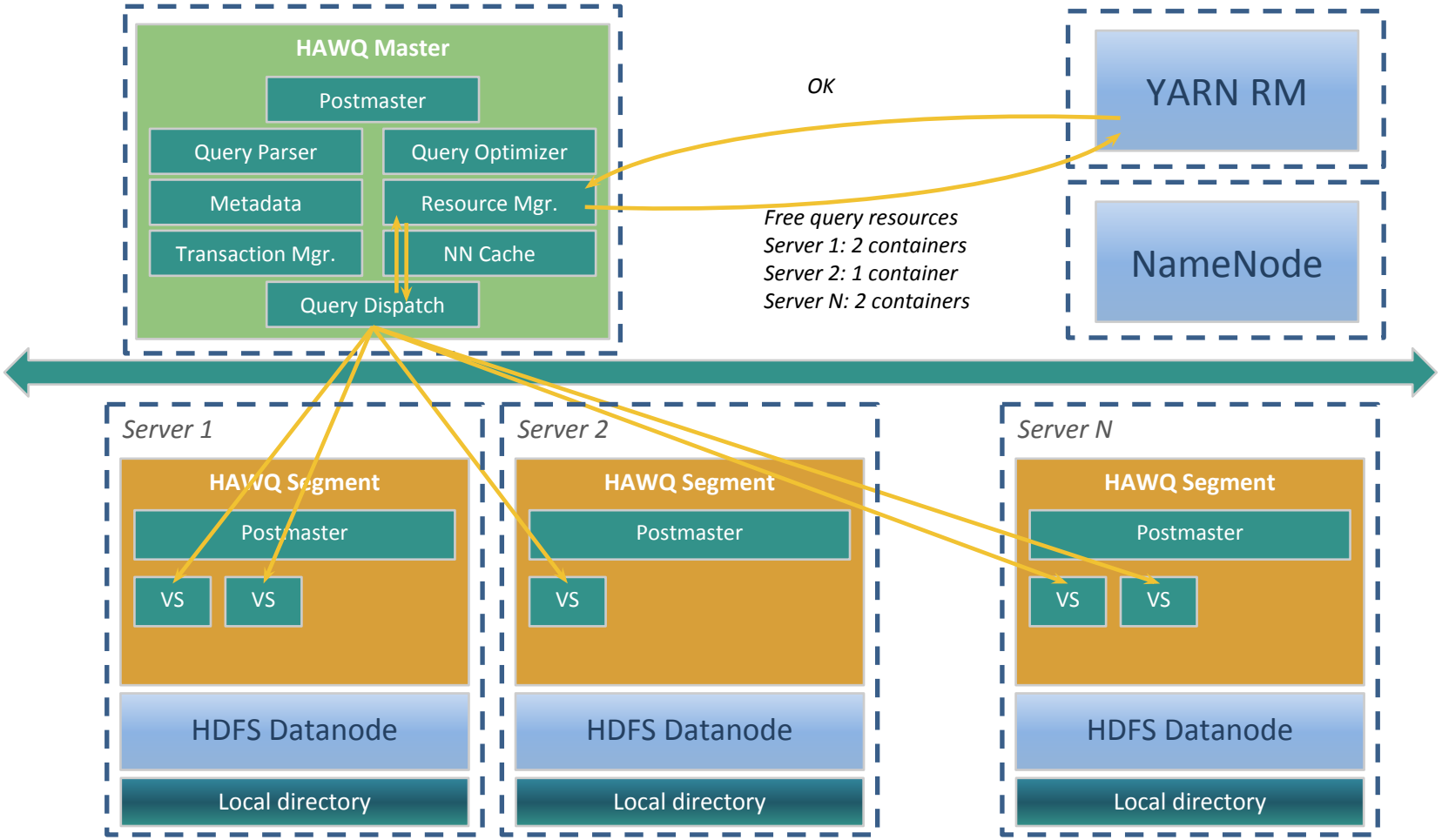
查询执行示例 – 结果返回



VS = Virtual Segment (container for Query Executors)
# of QEs in a v-seg = # of slices in a query

Pivotal

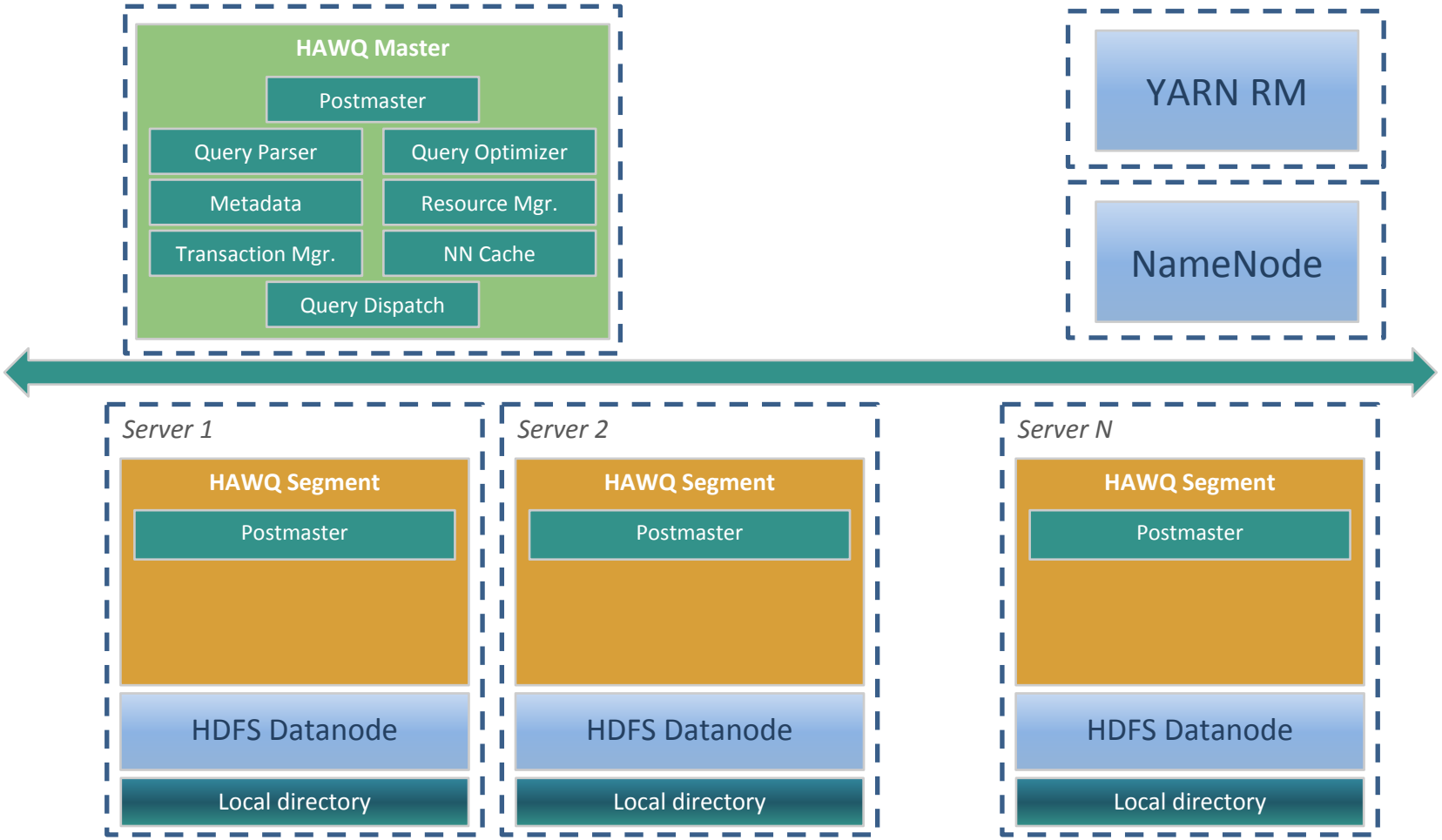
查询执行示例 – 清理



VS = Virtual Segment (container for Query Executors)
# of QEs in a v-seg = # of slices in a query

Pivotal

查询执行示例 — 清理



Pivotal

HDB/HAWQ 2.2.0.0最新功能

- HAWQ Register
- HAWQ Ranger 集成
- PXF ORC Profile
- RHEL-7 Support

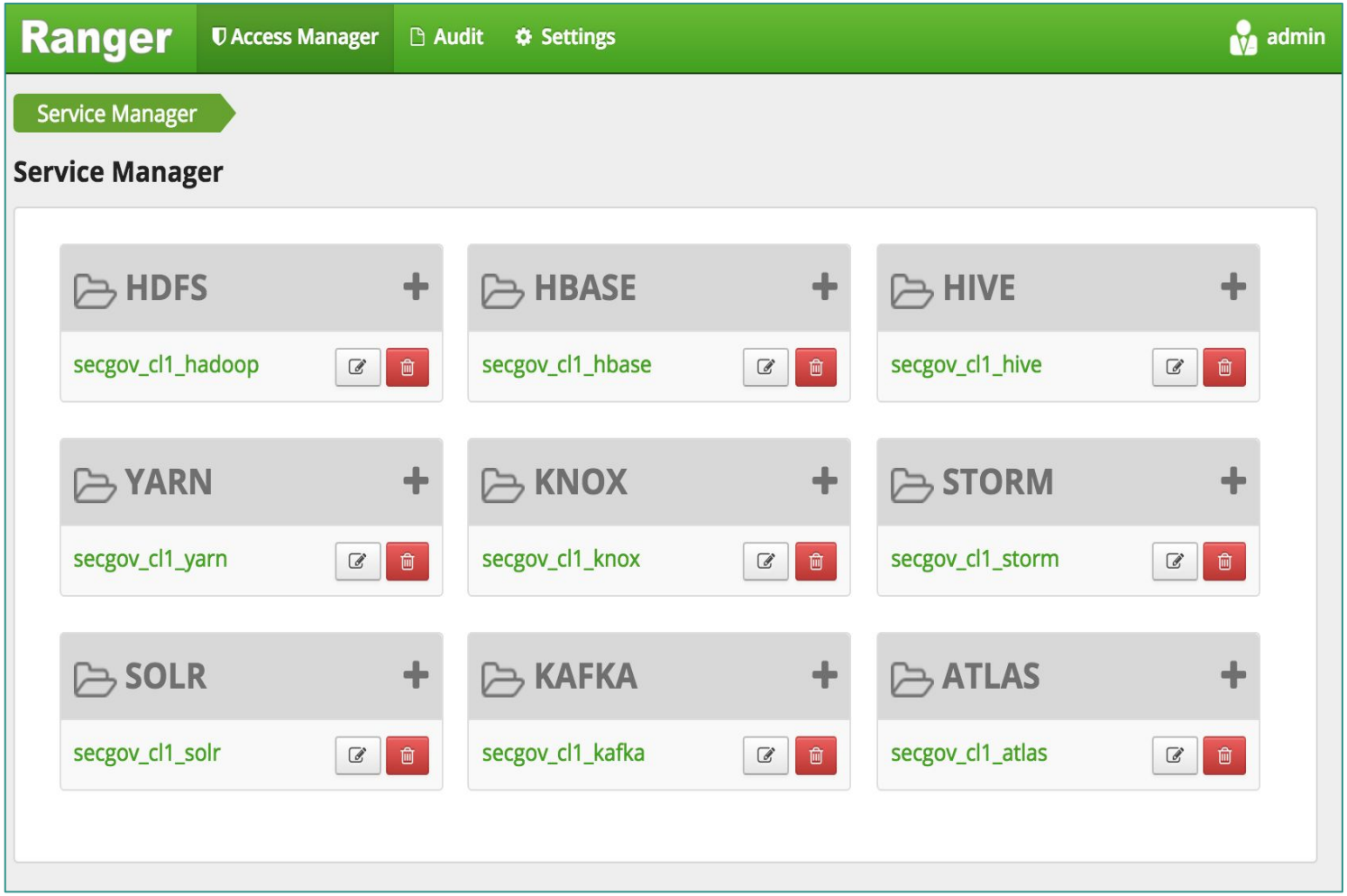
HAWQ Extract/HAWQ Register

- HAWQ Extract
  - Extract out metadata & HDFS file location for the table to yaml configuration file
  - Yaml configuration can be used by HAWQInputFormat
  - Usage *hawq extract [-h hostname] [-p port] [-U username] [-d database] [-o output_file] [-W] <tablename>*
- HAWQ Register
  - Register existing files on HDFS directly to HAWQ internal table
  - Scenario
    - Register other systems generated data
    - HAWQ cluster migration
  - Usage
    - *hawq register [-h <hostname>] [-p <port>] [-U <username>] -d <databasename> -f <hdfspath> <tablename>*
    - *hawq register [-h <hostname>] [-p <port>] [-U <username>] -d <databasename> -c <configFile> <tablename>*

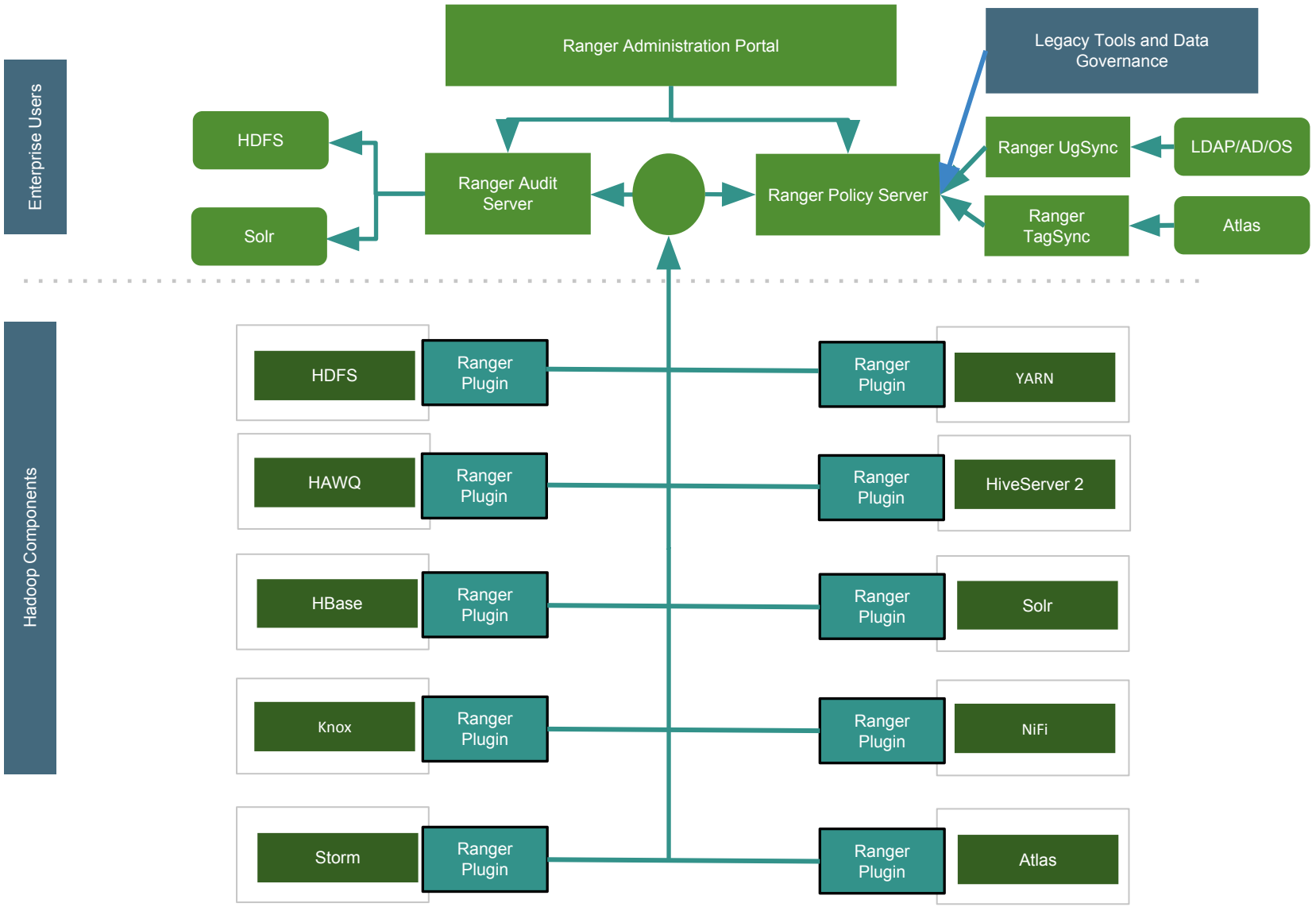
HAWQ与Ranger集成

- Ranger: A Global User Authorization Tool in Hadoop eco-system
  - Can support multiple systems such as HDFS, Hive, HBase, Knox, etc.
  - Provides a central UI for user to defining policies for different systems
  - Provide a base Java Plugin thus feasible for other products to define its own plugin to be controlled by Ranger
- HAWQ Current ACL
  - Implement through Grant/Revoke SQL Command
  - Current ACL is controlled by catalog table, which is stored in HAWQ master
- HAWQ needs to keep align with hadoop eco-systems, so we need integrate with Ranger ACL
  - Provide a GUC specifying whether enable ranger as ACL check
  - Once ranger is configured, move all the ACL check to Ranger side
  - Define all the policies in Ranger

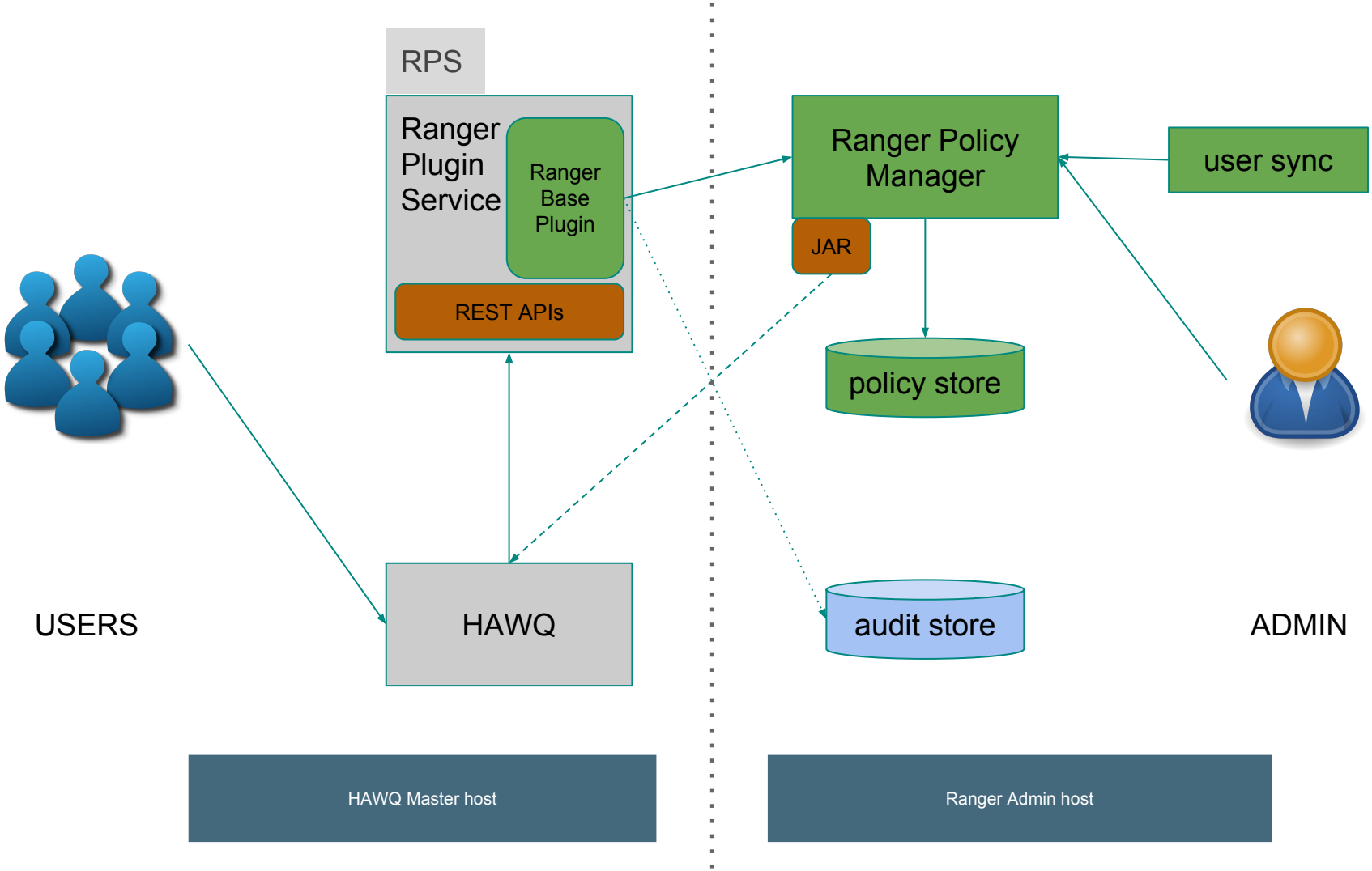
Apache Ranger: 集中化权限管理工具



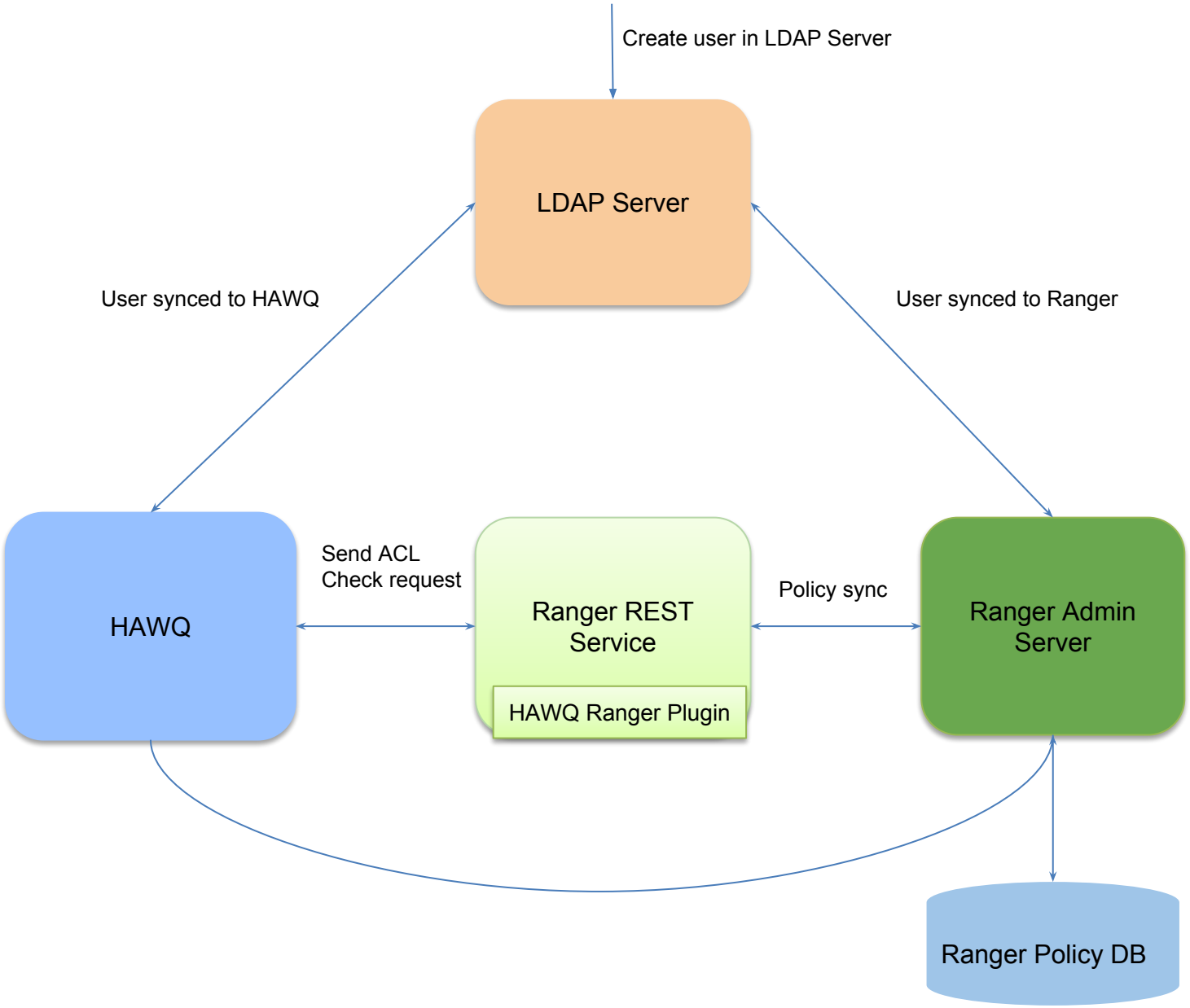
Apache Ranger 架构



HAWQ与Ranger集成

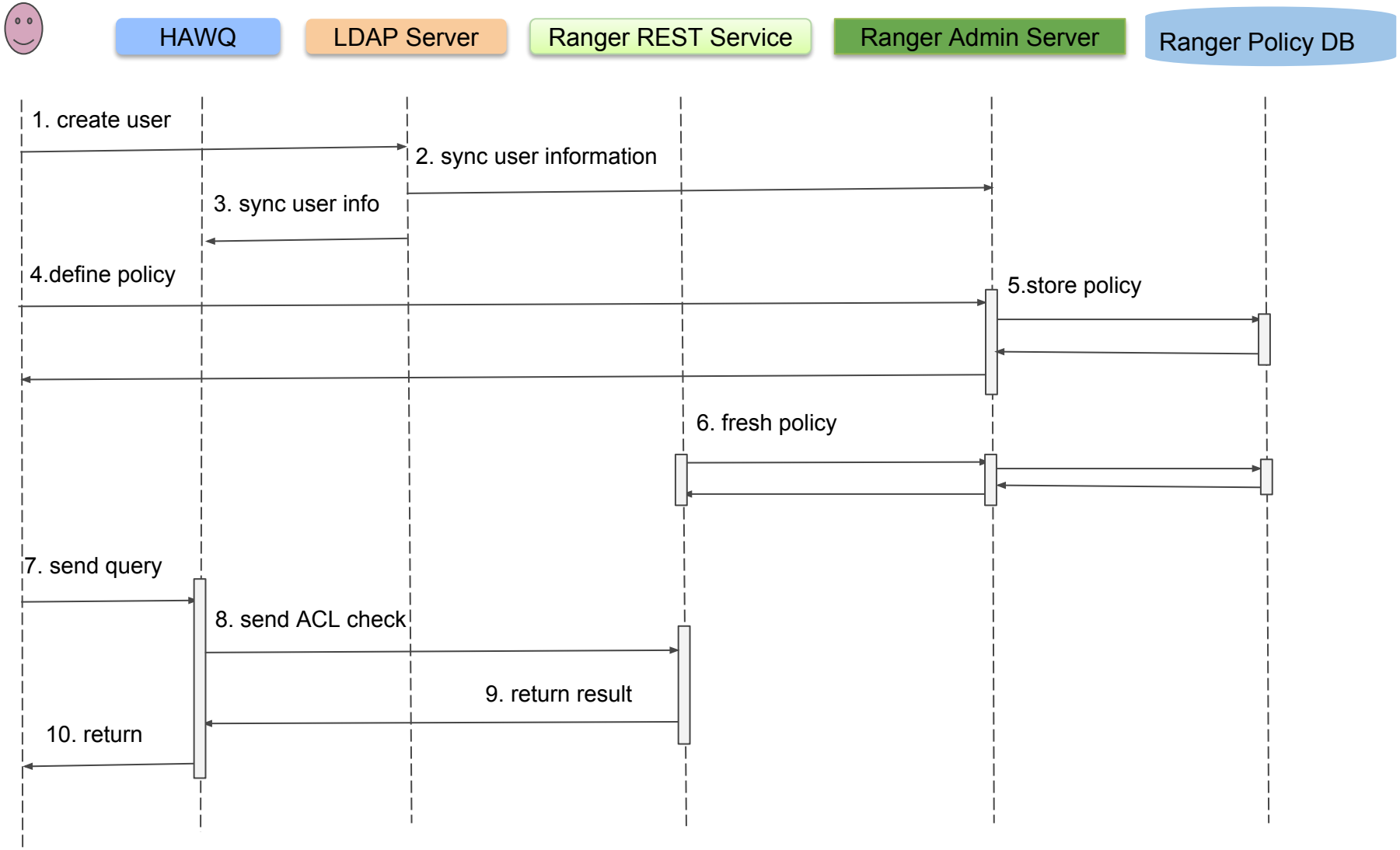


用户管理典型场景



Pivotal

HAWQ Ranger工作序列图



Pivotal

未来工作

TDE(透明数据加密) 支持

- TDE: HDFS implements transparent, end-to-end encryption
  - Data is transparently encrypted and decrypted without requiring changes to user application code
  - Data can only be encrypted and decrypted by the client
  - HDFS never stores or has access to unencrypted data or unencrypted data encryption keys
- HAWQ Enhancement
  - Modify libhdfs3 to add support for TDE

Parquet 格式升级

- Parquet 2.0 Enhancement
  - Add more Converted Type: Enum, Decimal, Date, Timstamp
  - Add more statistics in DataPageHeader: including max/min/null count, distinct count
  - Add Dictionary Page
  - Add sorting column information in Rowgroup meta
  - …
- HAWQ Upgrade to Parquet 2.0 support
  - Bring performance improvement by leveraging statistics information
  - Become more compatible with other systems which have supported Parquet 2.0

欢迎加入Apache HAWQ社区

- 贡献方式
  - Document／Wiki Enrich
  - Bug Report／Fix
  - 新功能开发
- 联系我们
  - Website: http://hawq.incubator.apache.org/
  - Wiki: https://cwiki.apache.org/confluence/display/HAWQ
  - Repo: https://github.com/apache/incubator-hawq.git
  - JIRA: https://issues.apache.org/jira/browse/HAWQ
  - Mailing lists: dev/user@hawq.incubator.apache.org

HAWQ官方纯技术讨论群

THANKS

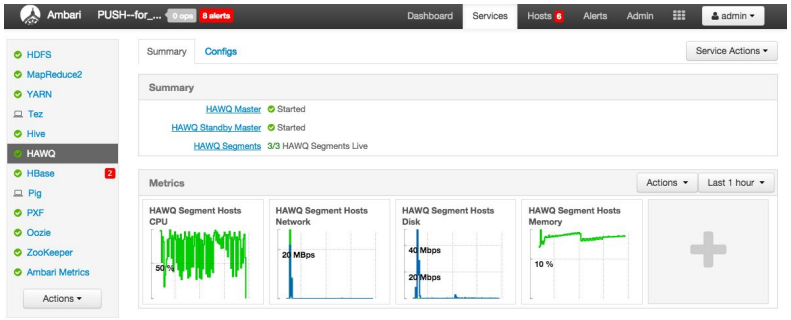
SequeMedia 盛拓传媒 IT168 ITPUB ChinaUnix

Hadoop-Native Administration via Ambari

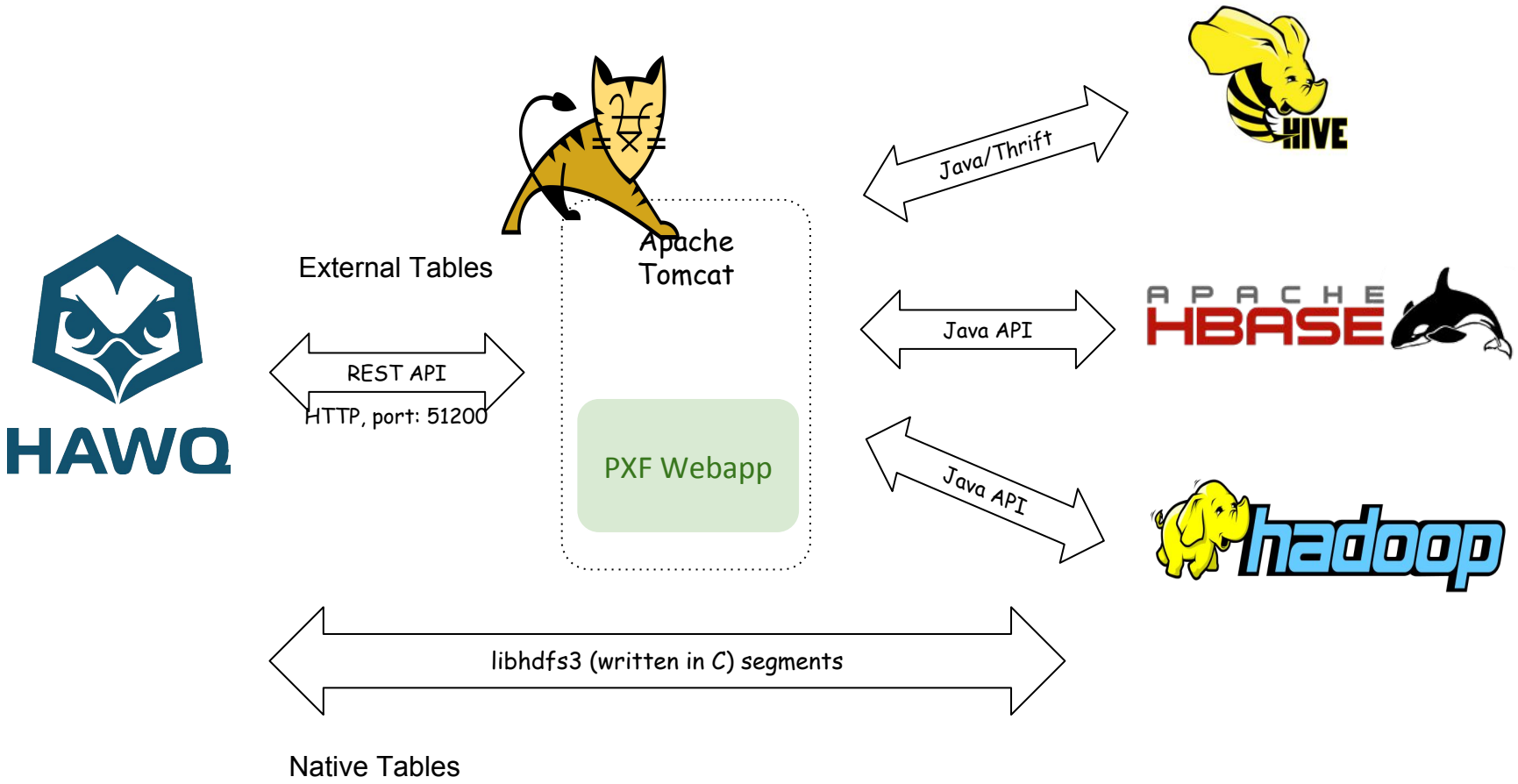
Manage HDB Alongside Hadoop Services

- Installation & Configuration
  - Use standard Ambari interface
  - Install HDB with just a few mouse clicks
  - Wizard-based experience
  - Stack Advisor enhancements
  - Proactive user warnings
  - Service Checks
- Kerberos & High Availability Support
- HAWQ Master > Standby Failover
- Cluster Expansion Support
- Visual Widgets on System Resources
- Service & Component Alerts

Apache Ambari

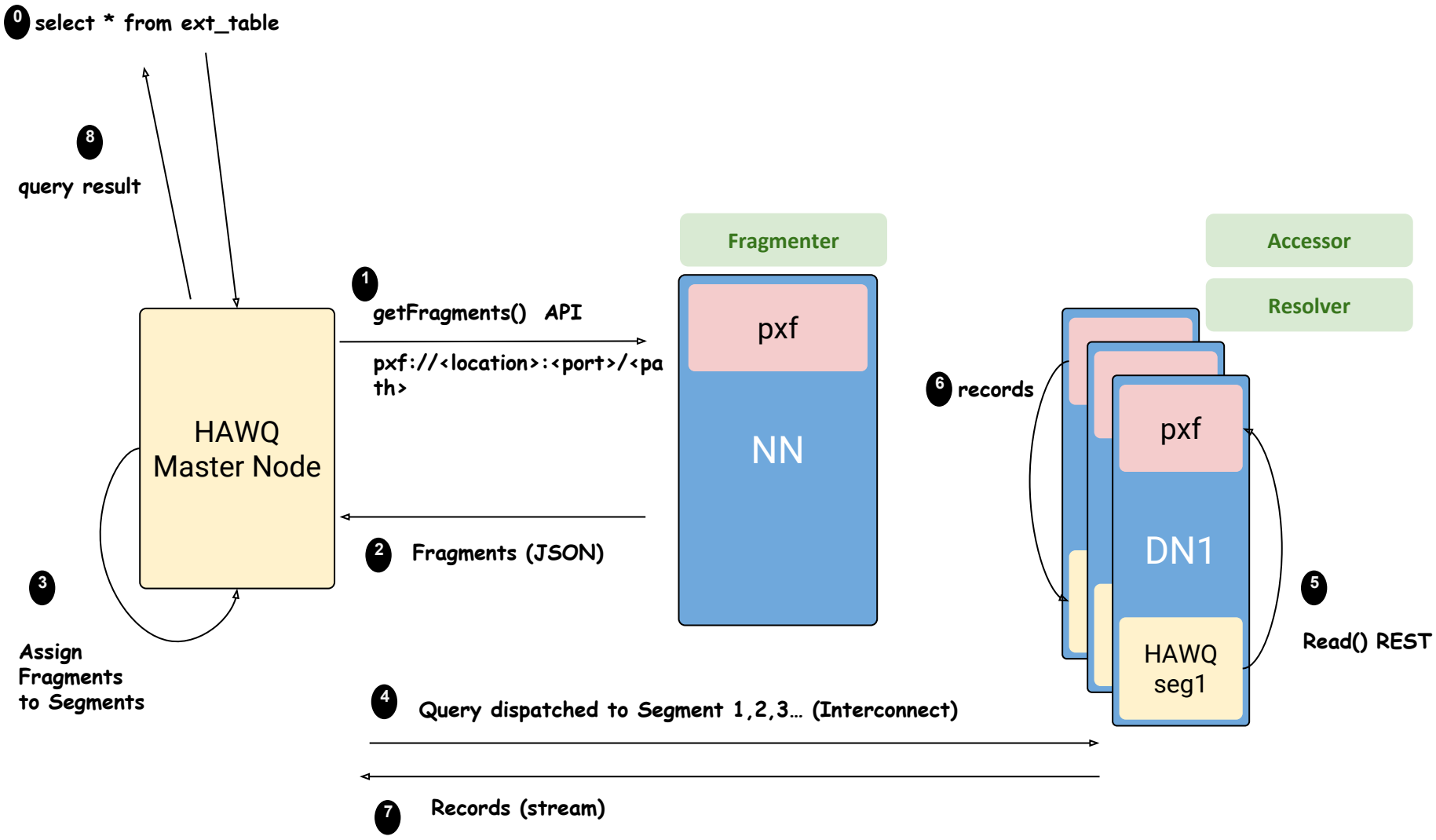


PXF Framework



Pivotal

Architecture - Read Data Flow



Pivotal

New HCatalog Integration

Simplified interoperability on external data

Now, HAWQ can read the schema automatically from HCatalog

SELECT * FROM hcatalog.default.weblogs
WHERE ts between '2015-09-01' and '2015-09-30';