SAS® GLOBALFORUM 2015

The Journey Is Yours

SAS and Hadoop – 4th S.O.T.U

Paul Kent, SAS

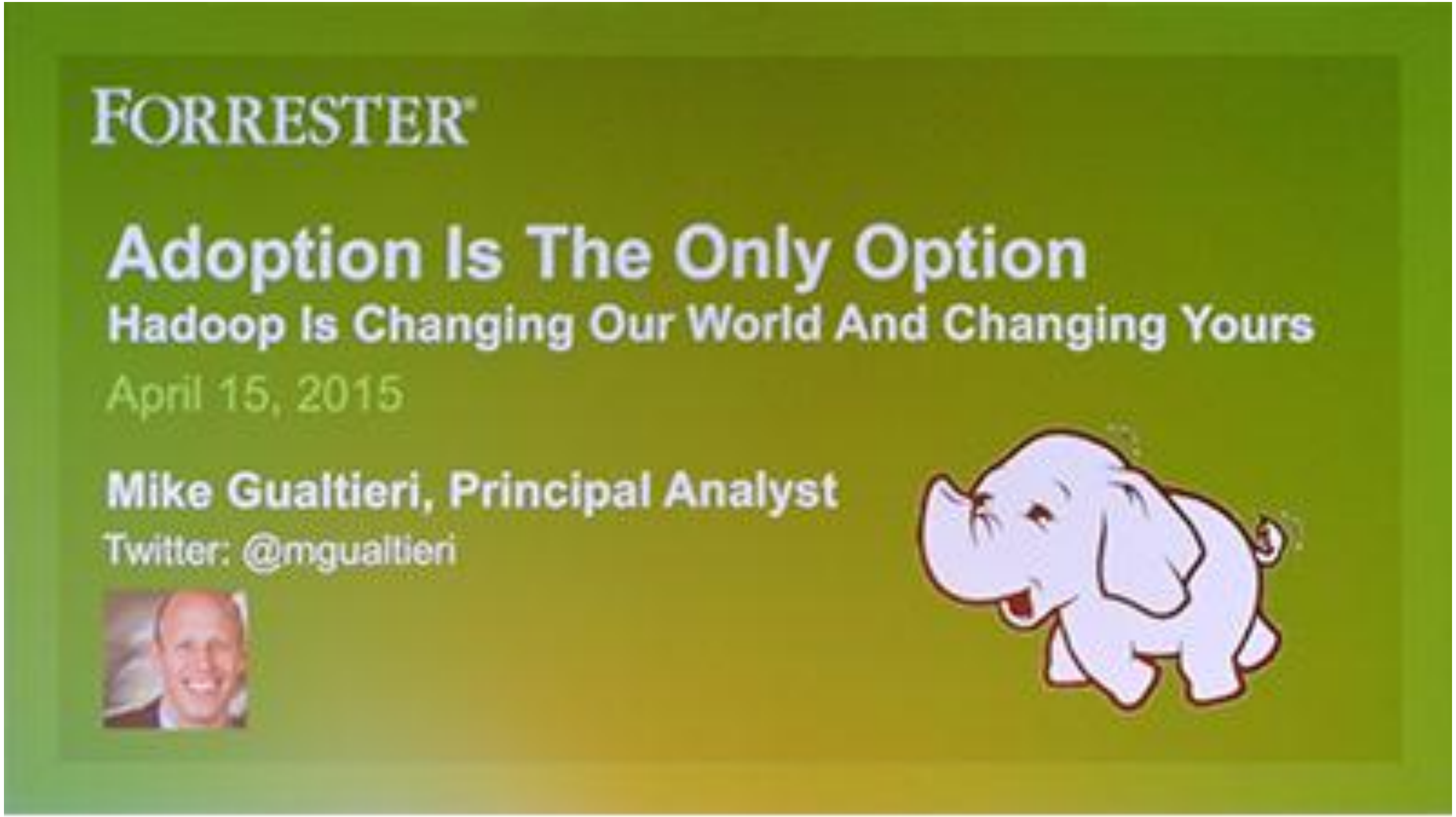
SAS GLOBALFORUM

SAS and Hadoop :: the BIG Picture

SAS and Hadoop are made for each other

This talk explains some of the reasons why.

Examples are drawn from the customer community to illustrate how SAS is a good addition to your Hadoop Cluster.



FORRESTER®

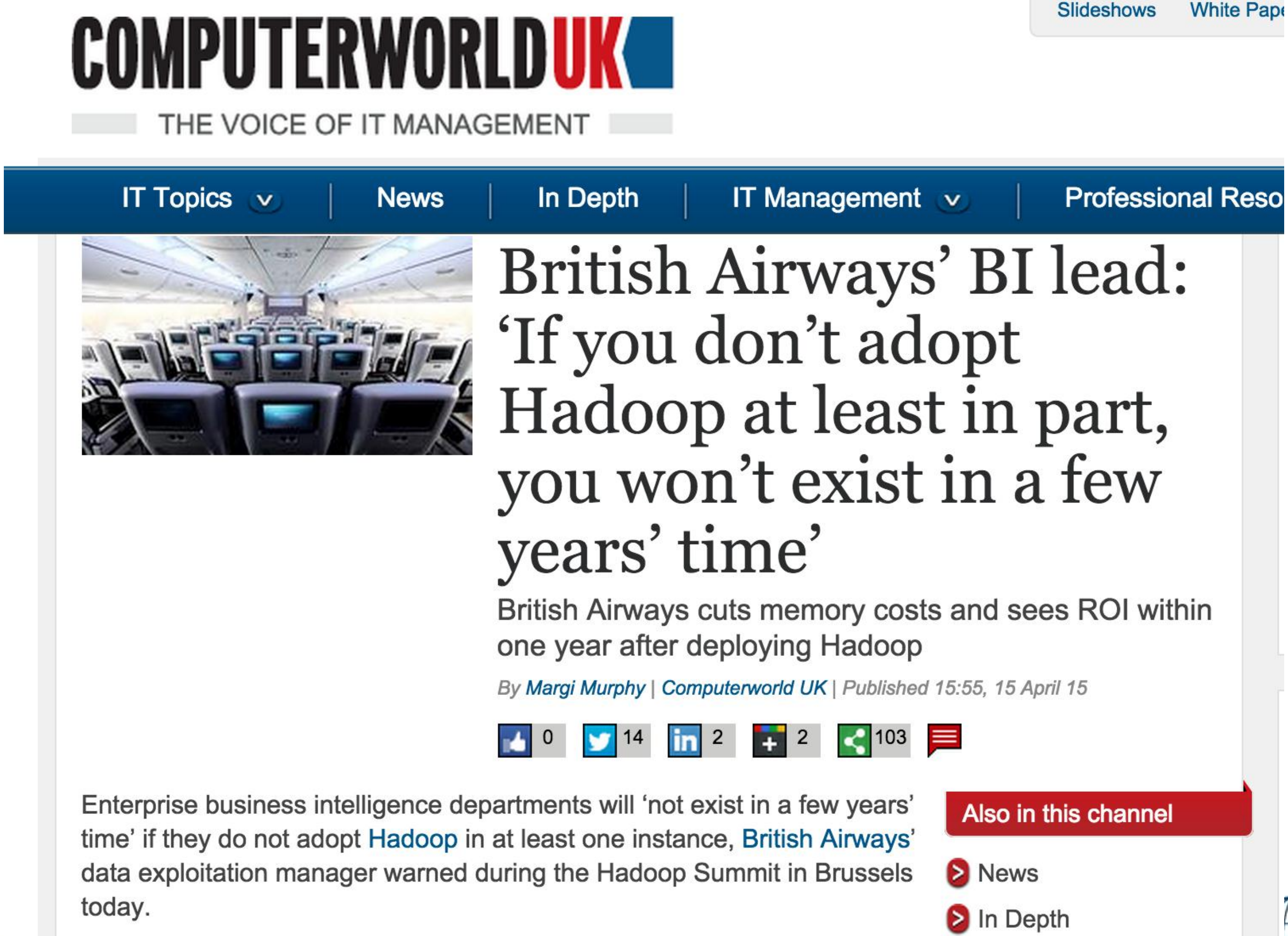
Adoption Is The Only Option

Hadoop Is Changing Our World And Changing Yours

April 15, 2015

Mike Gualtieri, Principal Analyst

Twitter: @mgualtieri



British Airways' BI lead: 'If you don't adopt Hadoop at least in part, you won't exist in a few years' time'

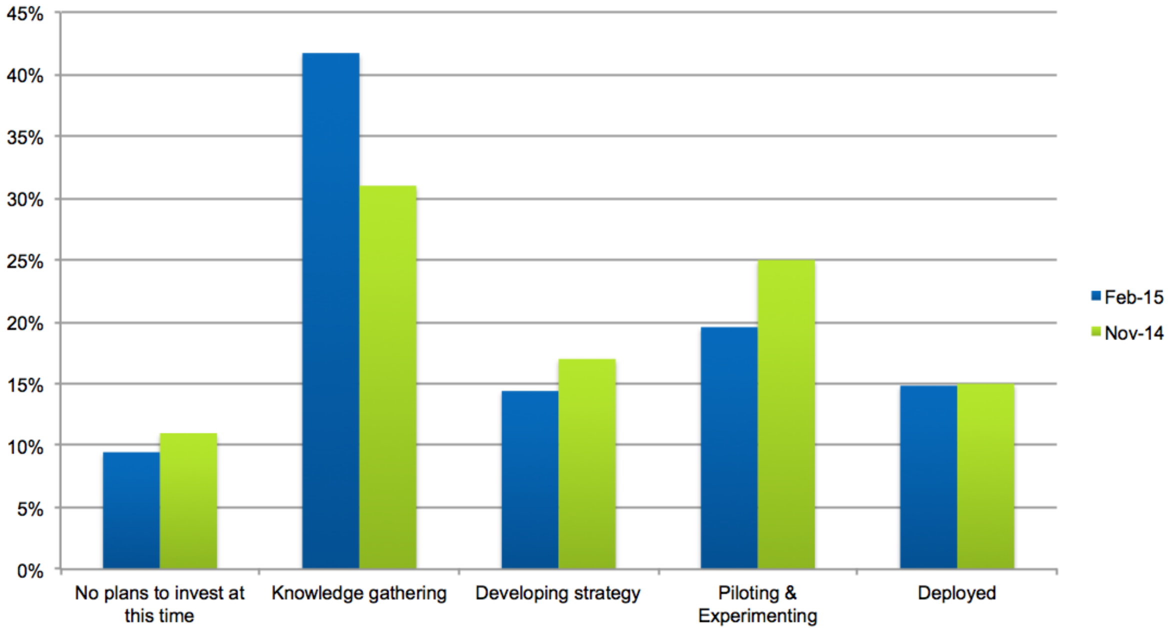
British Airways cuts memory costs and sees ROI within one year after deploying Hadoop

By Margi Murphy | Computerworld UK | Published 15:55, 15 April 15

Enterprise business intelligence departments will 'not exist in a few years' time' if they do not adopt Hadoop in at least one instance, British Airways' data exploitation manager warned during the Hadoop Summit in Brussels today.

SAS GLOBAL FORUM

Are YOU Hadooping Yet?



n=465 (Feb-15), 504 (Nov-14)

The Stages of the Relationship

1. Connecting (Getting to know each other)
   - What exactly is Hadoop?
   - Base SAS connections to Hadoop
2. Dating
   - SAS Access to Hadoop
   - Pig Storage extensions from SAS
   - SAS SerDes for Hive
3. Engaged
   - Scoring Accelerator
   - SPDS Server for Hadoop
   - Grid Manager for Hadoop
4. Committed
   - Data Loader for Hadoop
   - SAS High Performance Procedures and the LASR Analytic Server

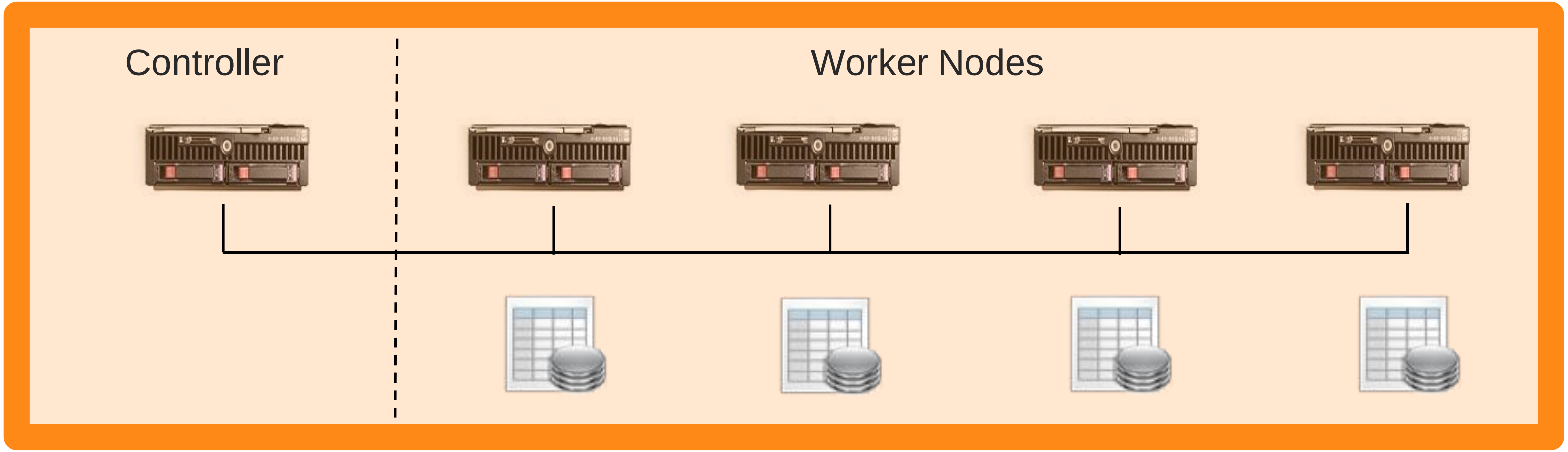
1. Introductions

Apache Hadoop - Background

The project includes these subprojects:

- Hadoop Common: The common utilities that support the other Hadoop subprojects.
- Hadoop Distributed File System (HDFS™): A distributed file system that provides high-throughput access to application data.
- Hadoop MapReduce: A software framework for distributed processing of large data sets on compute clusters.

Hadoop – Simplified View



- MPP (Massively Parallel) hardware running database-like software
- A single logical table is stored in parts across multiple worker nodes
- “work” operates in parallel on the different parts of the table

Idea #1 - HDFS. Never forgets!

Head Node	Data 1	Data 2	Data 3	Data 4…
MYFILE.TXT				
..block1 ->	block1copy1			
..block2 ->		block2copy2		
..block3 ->			block3copy3	

Idea #1 - HDFS. Never forgets!

Head Node	Data 1	Data 2	Data 3	Data 4…
MYFILE.TXT				
..block1 ->	block1copy1		block1copy2	
..block2 ->		block2copy2		block2copy2
..block3 ->	block3copy2		block3copy3	

Idea #1 - HDFS. Never forgets!

Head Node	Data 1	Data 2	Data 3	Data 4…
MYFILE.TXT				
..block1 ->	block1copy1		block1copy2	
..block2 ->		block2copy2		block2copy2
..block3 ->	block3copy2		block3copy3	

Idea #2 - MapReduce

- We Want the Minimum Age in the Room

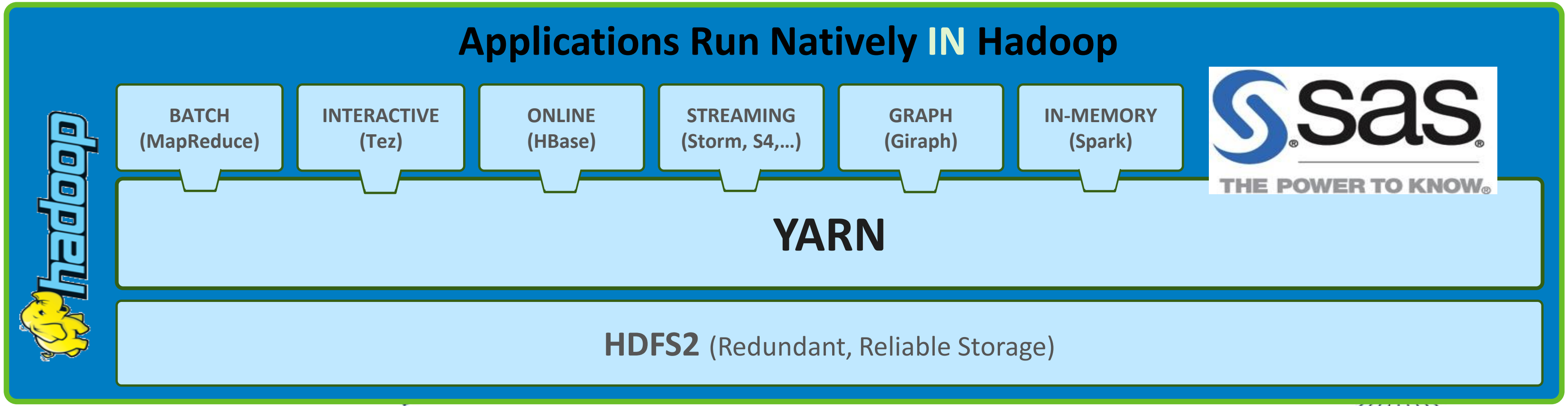
- Each Row in the audience is a data node
- I’ll be the coordinator
  - From outside to center, accumulate MIN
  - Sweep from back to front. Youngest Advances

Idea #3 YARN: Share Hadoop

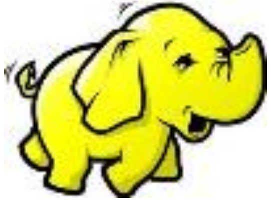
Store ALL DATA in one place…

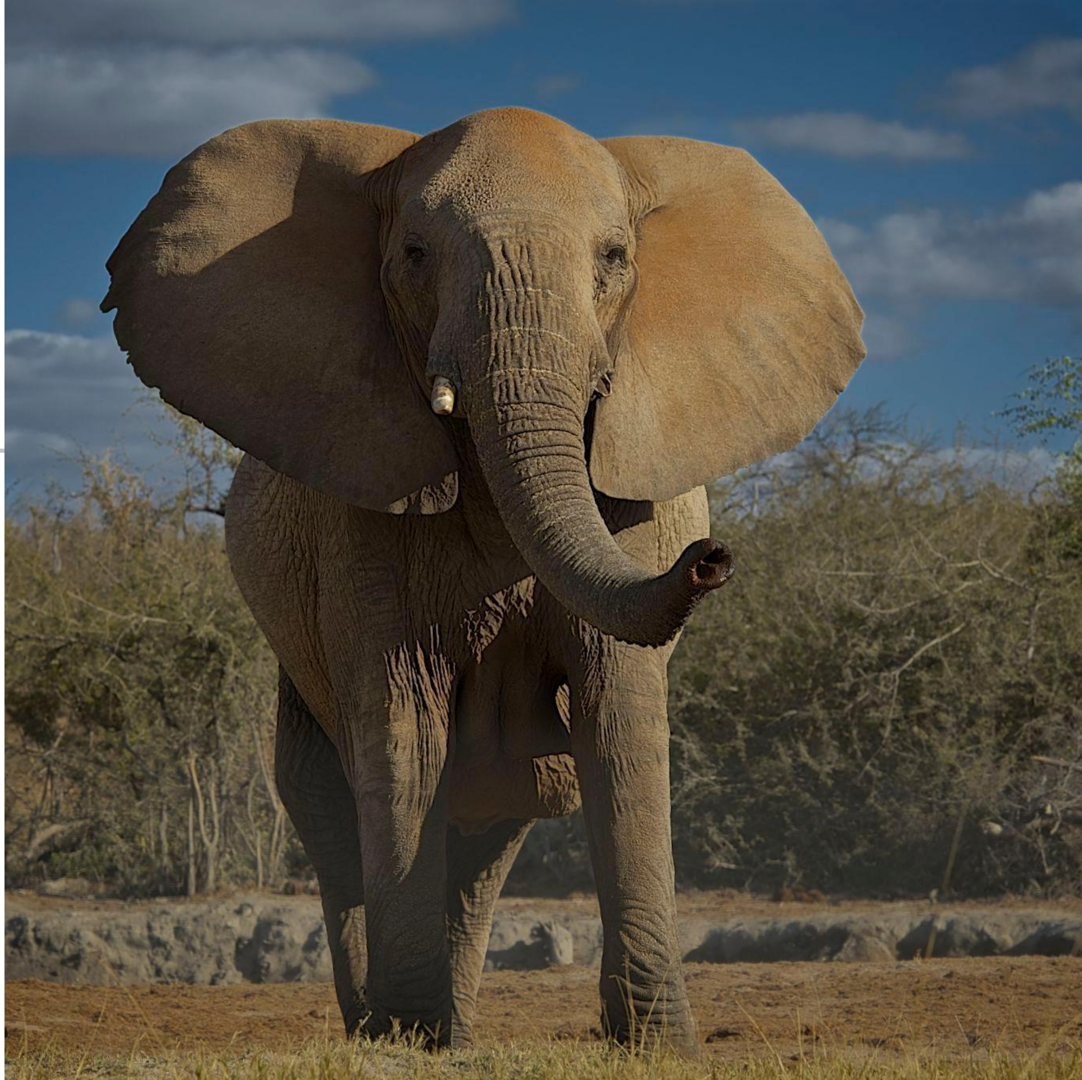
Interact with that data in MULTIPLE WAYS

with Predictable Performance and Quality of Service



Connecting





FILENAME xxx HADOOP

```
FILENAME paul HADOOP
   “/users/kent/mybigfile.txt”
   CONFIG=“/etc/hadoop.cfg” USER=“kent” PASS=“sekrit”;

DATA MYFILE;
   INFILE paul;
   INPUT name $ age sex $ height weight;
   RUN;
```

/etc/hadoop.cfg ?

```
<configuration>
<property>
  <name>fs.default.name</name>
  <value>hdfs://exa.unx.sas.com:8020</value>
</property>
<property>
  <name>mapred.job.tracker</name>
  <value>exa.unx.sas.com:8021</value>
</property>
</configuration>
```

Different Hadoop Versions?

```
options set=SAS_HADOOP_JAR_PATH="/u/kent/jars/cdh4/”;
```

- OpenSource Apache 2.0
- Cloudera CDH4 and CDH5
- Hortonworks 1.3.2 and 2.x (including DDN and Teradata OEM editions)
- Pivotal HD (was Greenplum)
- MAPR
- IBM BigInsights

PROC HADOOP Syntax

```
PROC HADOOP CFG=CFG … [VERBOSE];

HDFS <hdfs commands>;

MAPREDUCE <mapreduce options>;

PIG <pig options>;

RUN;
```

HDFS statement

```
hdfs mkdir='/tmp/rick/mydir';

hdfs copyfromlocal='myfile.txt' out='/tmp/rick/mydir/myfile.txt';

hdfs copytolocal='/tmp/rick/mydir/myfile.txt' out='myfile2.txt';

hdfs delete='/tmp/rick/mydir/myfile.txt';
```

HDFS Statement: new in 9.4m3

new and improved

```
hdfs ls='/tmp/rick';

hdfs ls='/tmp/rick' out=lsfile;

hdfs cat='/tmp/rick/testfile.txt';

hdfs cat='/tmp/rick/*.txt out=catfile;
```

SAS GLOBAL FORUM

Proc SQOOP

new and improved

- Allows users to submit commands to Apache Sqoop from within a SAS session
- Moves data to and from the database
- Imports data into either HDFS or Hive tables
- Uses Apache Oozie to execute commands using a RESTful API, so no jars are required on the user’s client machine

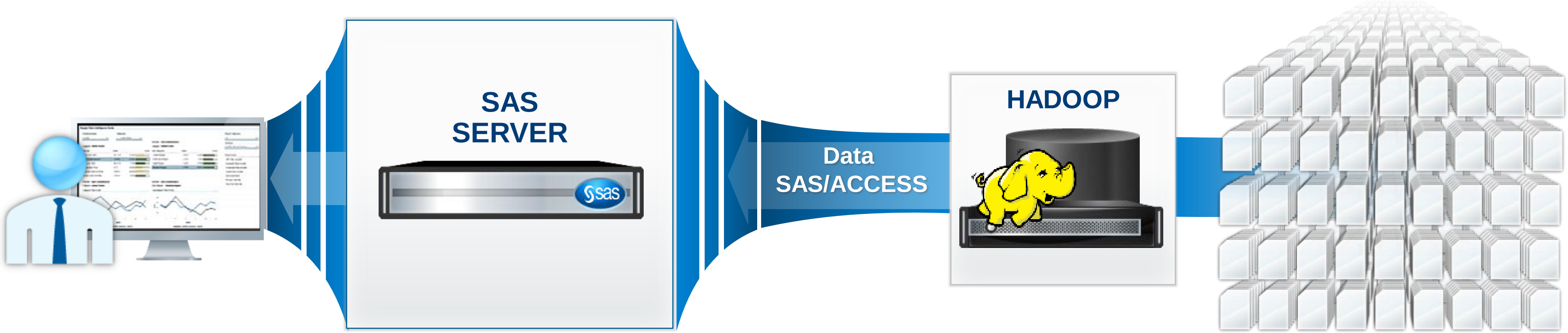
2. Dating

SAS Learns Hadoop Tables

Hadoop Learns SAS Tables

SAS / ACCESS TO HADOOP

1 SAS/Access to Hadoop or Impala - Push *some* of SAS' processing to Hadoop



SAS/Access to Hadoop
SAS/Access to Impala
SAS/Access to Hawq (M3)

LIBNAME xxx HADOOP

```
LIBNAME olly HADOOP
    SERVER=olly.mycompany.com
    USER=“kent” PASS=“sekrit”;


PROC DATASETS LIB=OLLY;
    RUN;
```

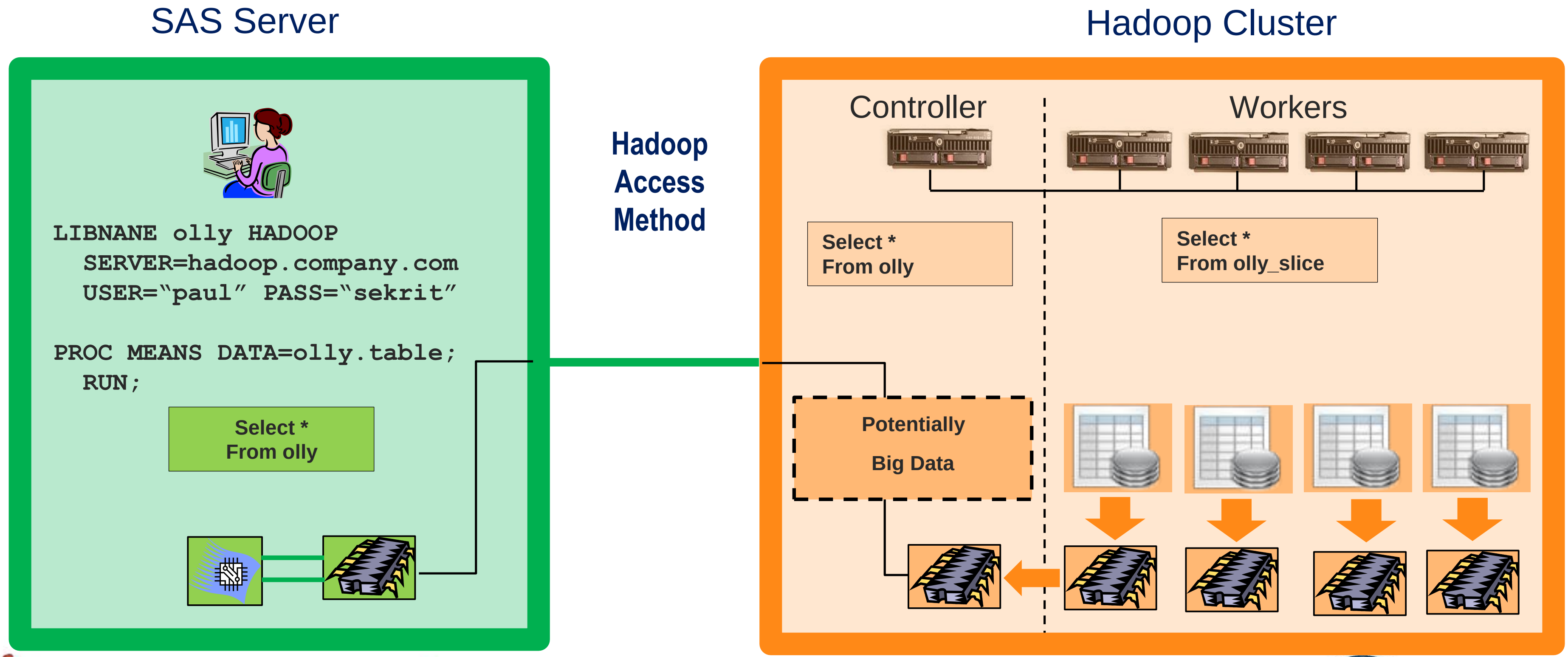
LIBNAME xxx HADOOP

- Cool! I don’t have to repeat the INPUT statement in every program that I want to access my files!!
- Thanks to Apache HIVE
  - supplies the metadata that projects a relational view of several underlying file types.
  - Provides SQL with relational primitives like JOIN and GROUP BY

Not only HIVE. Cloudera Impala

Not only HIVE. Pivotal Hawk

Hadoop LIBNAME Statement



SAS Server

```
LIBNANE olly HADOOP
   SERVER=hadoop.company.com
   USER=“paul” PASS=“sekrit”

PROC MEANS DATA=olly.table;
   RUN;
```

Select *
From olly

Hadoop
Access
Method

Hadoop Cluster

Controller

Workers

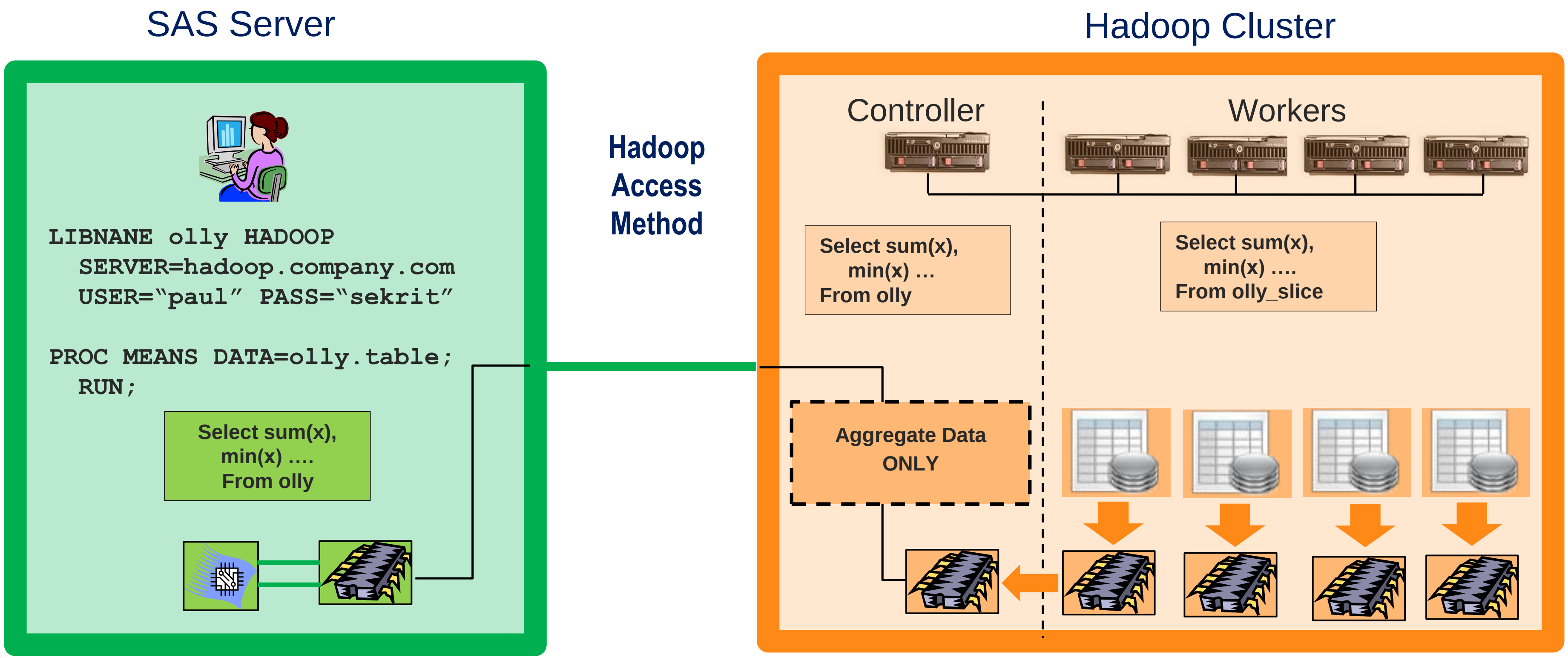
Select *
From olly

Select *
From olly_slice

Potentially

Big Data

Hadoop LIBNAME Statement – with SQL Pasthru



SAS Server

```
LIBNANE olly HADOOP
   SERVER=hadoop.company.com
   USER=“paul” PASS=“sekrit”

PROC MEANS DATA=olly.table;
   RUN;
```

Select sum(x),
min(x) ….
From olly

Hadoop
Access
Method

Hadoop Cluster

Select sum(x),
min(x) …
From olly

Select sum(x),
min(x) ….
From olly_slice

Aggregate Data
ONLY

HADOOP LIBNAME Statement

- PROC SQL explicit SQL is supported

- This sends the SQL exactly as you typed it down into the HIVE processor

- One way to move the work (joins, group by) down onto the cluster

HADOOP LIBNAME Statement

new and improved

- PROC SQL implicit SQL is supported
- Base Procedure pushdown
- More ways to move the work down onto the cluster

More Implicit Passthru

```
options sqlgeneration=dbms;

proc rank data=x.cake out=order descending ties=low;

   var present taste;

   ranks PresentRank TasteRank;

run;
```

```
IMPALA_25: Prepared: on connection 2
SELECT `table0`.`name`, `table0`.`present`, `table0`.`taste`, `table1`.`rankalias0` AS
`PresentRank`, `table2`.`rankalias1` AS `TasteRank` FROM ( SELECT `name` AS `name`, `present` AS
`present`, `taste` AS `taste` FROM `cake` ) AS `table0` LEFT JOIN ( WITH subquery0 AS (SELECT
`present`, `tempcol0` AS `rankalias0` FROM ( SELECT `present`, MIN( `tempcol1` ) OVER ( PARTITION
BY `present` ) AS `tempcol0` FROM( SELECT `present`, CAST( ROW_NUMBER() OVER ( ORDER BY `present`
DESC ) AS DOUBLE ) AS `tempcol1` FROM ( SELECT `name` AS `name`, `present` AS `present`, `taste`
AS `taste` FROM `cake` ) AS `subquery2` WHERE ( ( `present` IS NOT NULL ) ) ) AS `subquery1` ) AS
`subquery0` ) SELECT DISTINCT `present`, `rankalias0` FROM subquery0 ) AS `table1` ON ( (
`table0`.`present` = `table1`.`present` ) ) LEFT JOIN ( WITH subquery3 AS (SELECT `taste`,
`tempcol2` AS `rankalias1` FROM ( SELECT `taste`, MIN( `tempcol3` ) OVER ( PARTITION BY `taste` )
AS `tempcol2` FROM( SELECT `taste`, CAST( ROW_NUMBER() OVER ( ORDER BY `taste` DESC ) AS DOUBLE )
AS `tempcol3` FROM ( SELECT `name` AS `name`, `present` AS `present`, `taste` AS `taste` FROM
`cake` ) AS `subquery5` WHERE ( ( `taste` IS NOT NULL ) ) ) AS `subquery4` ) AS `subquery3` )
SELECT DISTINCT `taste`, `rankalias1` FROM subquery3 ) AS `table2` ON ( ( `table0`.`taste` =
`table2`.`taste` ) )
```

Hadoop (PIG) Learns SAS Tables

```
register pigudf.jar, sas.lasr.hadoop.jar, sas.lasr.jar;
/* Load the data from a CSV in HDFS */
A = load '/user/kent/class.csv'
    using PigStorage(',')
    as (name:chararray, sex:chararray,
          age:int, height:double, weight:double);
```

(continued…)

Hadoop (PIG) Learns SAS Tables

```
Store A into '/user/kent/class'
        using com.sas.pigudf.sashdat.pig.SASHdatStoreFunc(
                ’bigcdh01.unx.sas.com',
                '/user/kent/class_bigcdh01.xml');
```

SPD *Engine* with Hadoop

new and improved

- Support for running on MapR 4.0.2
- Support for Code Accelerator
- Enhanced WHERE pushdown: AND, OR, NOT, parenthesis, range operators and in-lists
- Parallel write support can improve write performance up to 40%
- Optionally uses Apache Curator/Zookeeper as a distributed lock server. No more physical lock files.

```
libname spdat spde '/user/dodeca' hdfshost=default;
```

Hive SerDe for SPDE data

new and improved

- Provides direct read access to SPDE data from the Hadoop ecosystem
- Access SPDE data from Hive, MapReduce and Pig
- Java tool provided to populate the Hive metastore with SPDE metadata

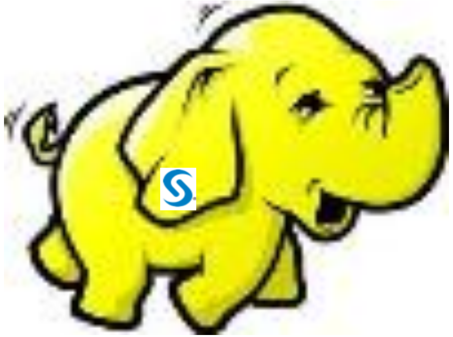
SAS Hadoop Engine – Anyfile Reader

- Consistent with Hadoop and its laissez-faire approach to schema “on Need”
- PROC HDMD describes as much of the file as you want to see as a table
- Libname engine matches metadata (HDMD) with datafile and returns “rowbuffers” to calling SAS procedure
- Imagine exporting INFILE/INPUT statement to Hadoop to be run there (at hadoop scale)

TEACH HADOOP (MAP/REDUCE) ABOUT SAS

```
/* Create HDMD file */
proc hdmd name=gridlib.people
        format=delimited
        sep=tab
        file_type=custom_sequence
        input_format='com.sas.hadoop.ep.inputformat.sequence.PeopleCustomSequenceInputFormat'
        data_file='people.seq';

    COLUMN name   varchar(20) ctype=char;
    COLUMN sex    varchar(1)  ctype=char;
    COLUMN age    int         ctype=int32;
    column height double      ctype=double;
    column weight double      ctype=double;
run;
```

3. Engaged

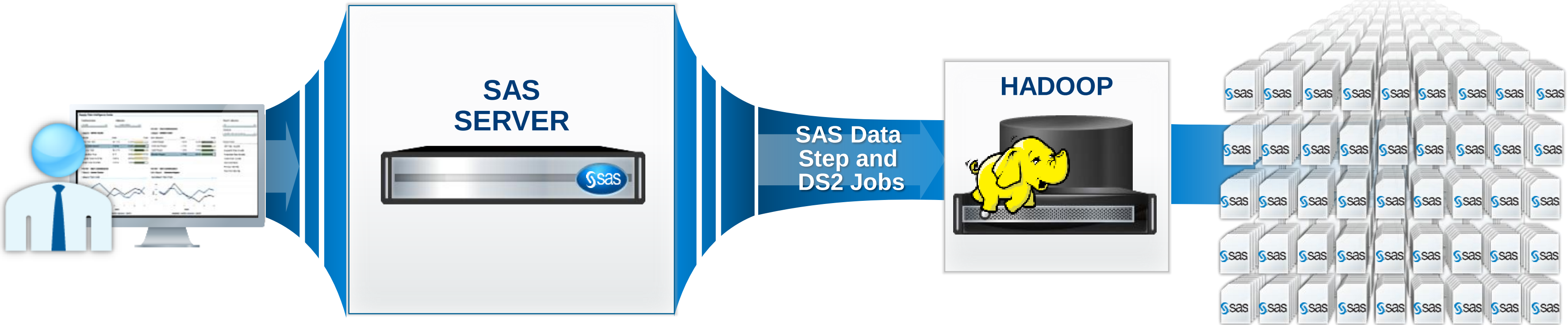


SAS Embedded Process

DATA LOADER, EP & ACCELERATORS

new and improved

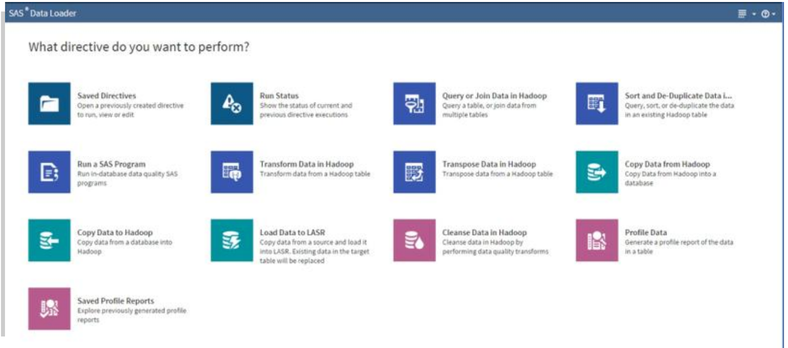
2 Push SAS ETL/ELT processing to Hadoop with MapReduce



SAS Data Loader for Hadoop:

- *SAS Code Accelerator for Hadoop*
- *SAS Data Quality Accelerator for Hadoop*

SAS Scoring Accelerator for Hadoop



What directive do you want to perform?

Saved Directives Open a previously created directive to run, view or edit

Run Status Show the status of current and previous directive executions

Query or Join Data in Hadoop Query a table, or join data from multiple tables

Sort and De-Duplicate Data i... Query, sort, or de-duplicate the data in an existing Hadoop table

Run a SAS Program Run in-database data quality SAS programs

Transform Data in Hadoop Transform data from a Hadoop table

Transpose Data in Hadoop Transpose data from a Hadoop table

Copy Data from Hadoop Copy Data from Hadoop into a database

Copy Data to Hadoop Copy data from a database into Hadoop

Load Data to LASR Copy data from a source and load it into LASR. Existing data in the target table will be replaced

Cleanse Data in Hadoop Cleanse data in Hadoop by performing data quality transforms

Profile Data Generate a profile report of the data in a table

Saved Profile Reports Explore previously generated profile reports

DS2 Merge on Hadoop

new and improved

- DS2 Code Accelerator generates Hive SQL to emulate classic DATA Step merge
- Merged data flows into the DS2 PDV in BY-groups ready for post – processing
  - must use a BY statement for predictable results in parallel execution
  - any “colliding” columns must have compatible data types.
- IN= is supported
- FIRST. / LAST. supported

Ex: *merge t1(in=int1) t2(in=int2) t3(in=int3); by x;*

Hadoop Code Accelerator

new and improved

- Supports new file types :
  - SPDE
  - Via HCATALOG: ORC, Parquet, Avro
- Supports Hive partitioned tables
- Supports SQL in the DS2 SET statement
  - Example: *set { select * from t1 inner join t2 on t1.id = t2.id };*
  - SQL is passed to Hive to create the input for the DS2 EP
- Supports Multiple Tables in the SET statement.
  - Emulated in Hive using a SQL UNION ALL
- Supports the new MERGE statement
  - Emulated in Hive SQL

New Hadoop Embedded Process

new and improved

- Fully integrated with YARN Resource Manager
- Runs in the same process as the Map-Reduce JVM
- EP Proxy Java and TKTS C share native memory allocated by TK
- Native buffers are allocated outside of JVM heap space
- TK Journal writes directly to MR job log
- Supports the new Analytic Store scoring model from Factory Miner
- Supports reading from HCatalog SerDes for the Scoring Accelerator and HPA/LASR parallel load.

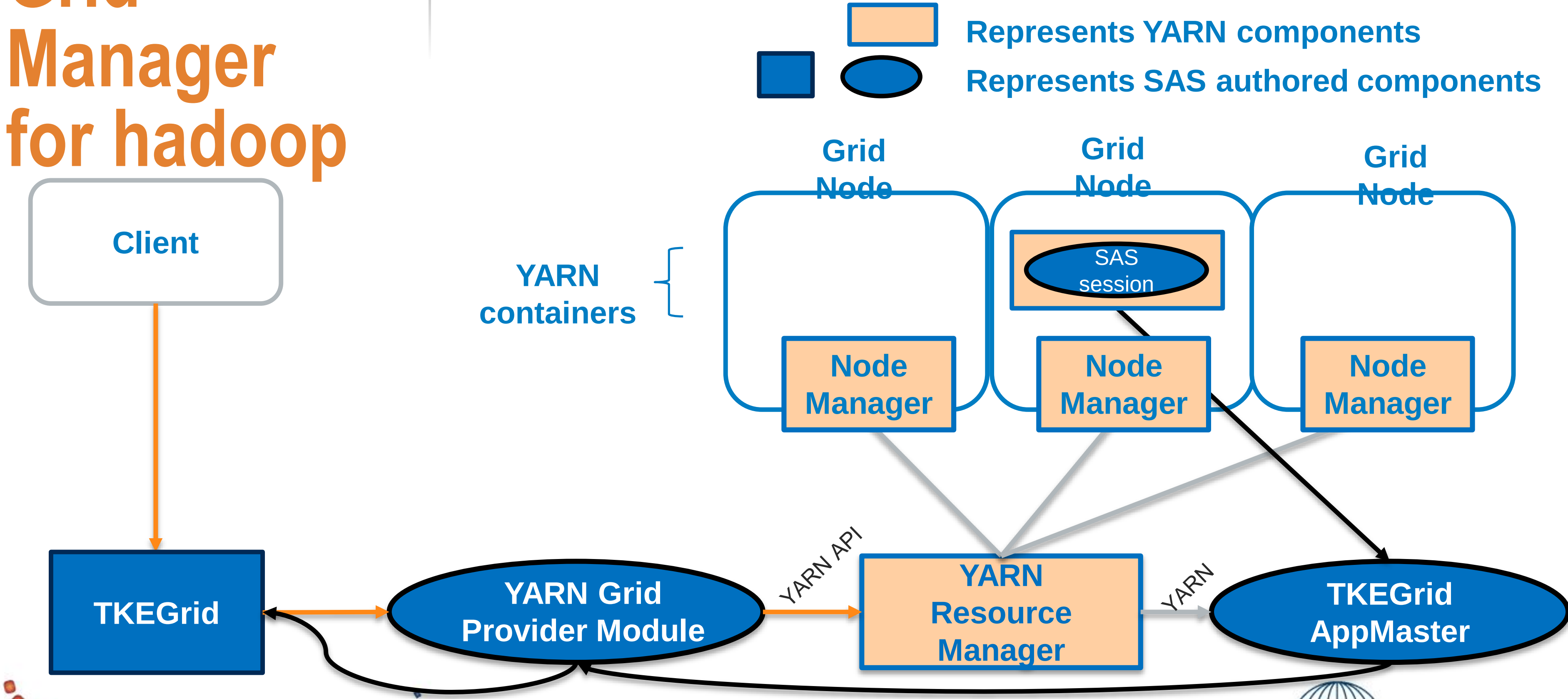
New Hadoop EP Install Process

new and improved

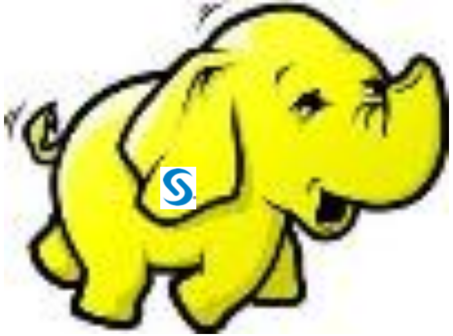
- Does *not* require installing Access to Hadoop to get the EP zip.
- Does *not* require a Linux user to run
- Does *not* create Linux service under initd.d
- Does *not* start the spawner deamon process
- Does *not* need to be copied to Hadoop lib folder or Hadoop native lib folder
- Does *not* require root access
- Does *not* require two way SSH keys setup

Grid Manager for hadoop

TKEGRID INTEGRATION WITH YARN



4. Committed



Data Loader for Hadoop

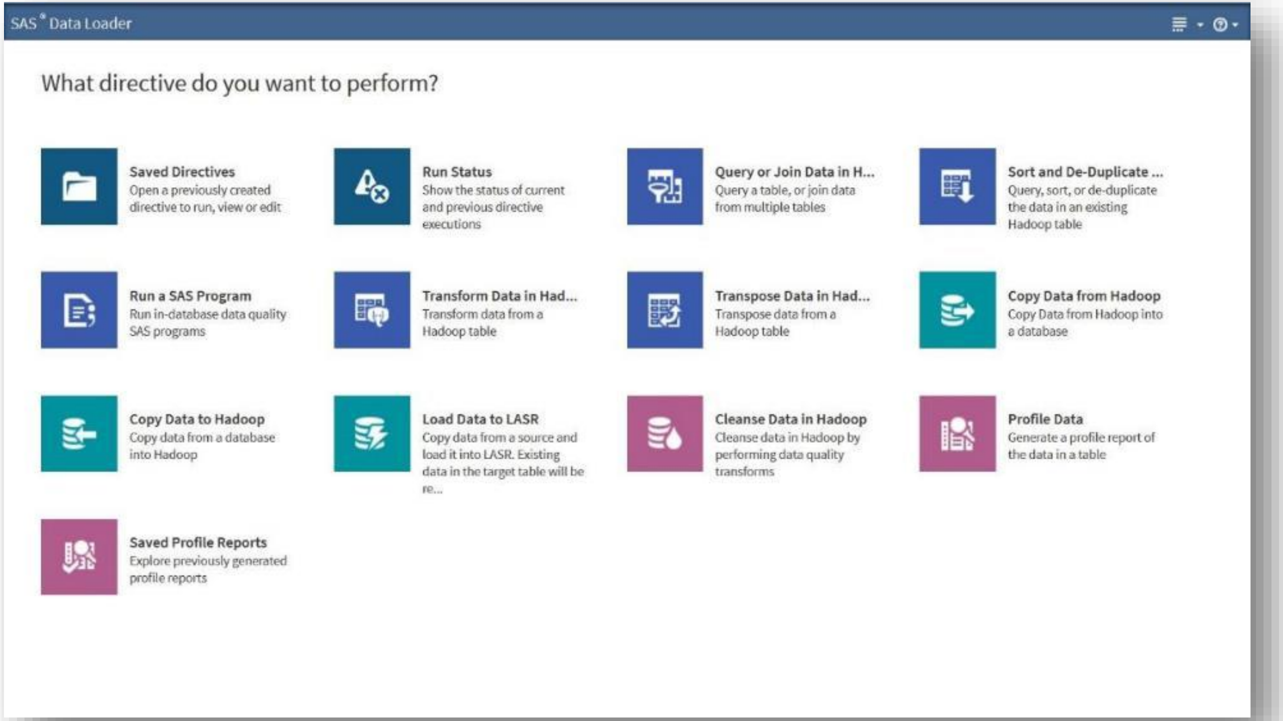
SAS HPA and VA on Hadoop

SAS GLOBAL FORUM

Introducing

SAS DATA LOADER FOR HADOOP

Self-service big data preparation for business users



SAS® Data Loader

What directive do you want to perform?

Saved Directives
Open a previously created directive to run, view or edit

Run Status
Show the status of current and previous directive executions

Query or Join Data in H...
Query a table, or join data from multiple tables

Sort and De-Duplicate ...
Query, sort, or de-duplicate the data in an existing Hadoop table

Run a SAS Program
Run in-database data quality SAS programs

Transform Data in Had...
Transform data from a Hadoop table

Transpose Data in Had...
Transpose data from a Hadoop table

Copy Data from Hadoop
Copy Data from Hadoop into a database

Copy Data to Hadoop
Copy data from a database into Hadoop

Load Data to LASR
Copy data from a source and load it into LASR. Existing data in the target table will be re...

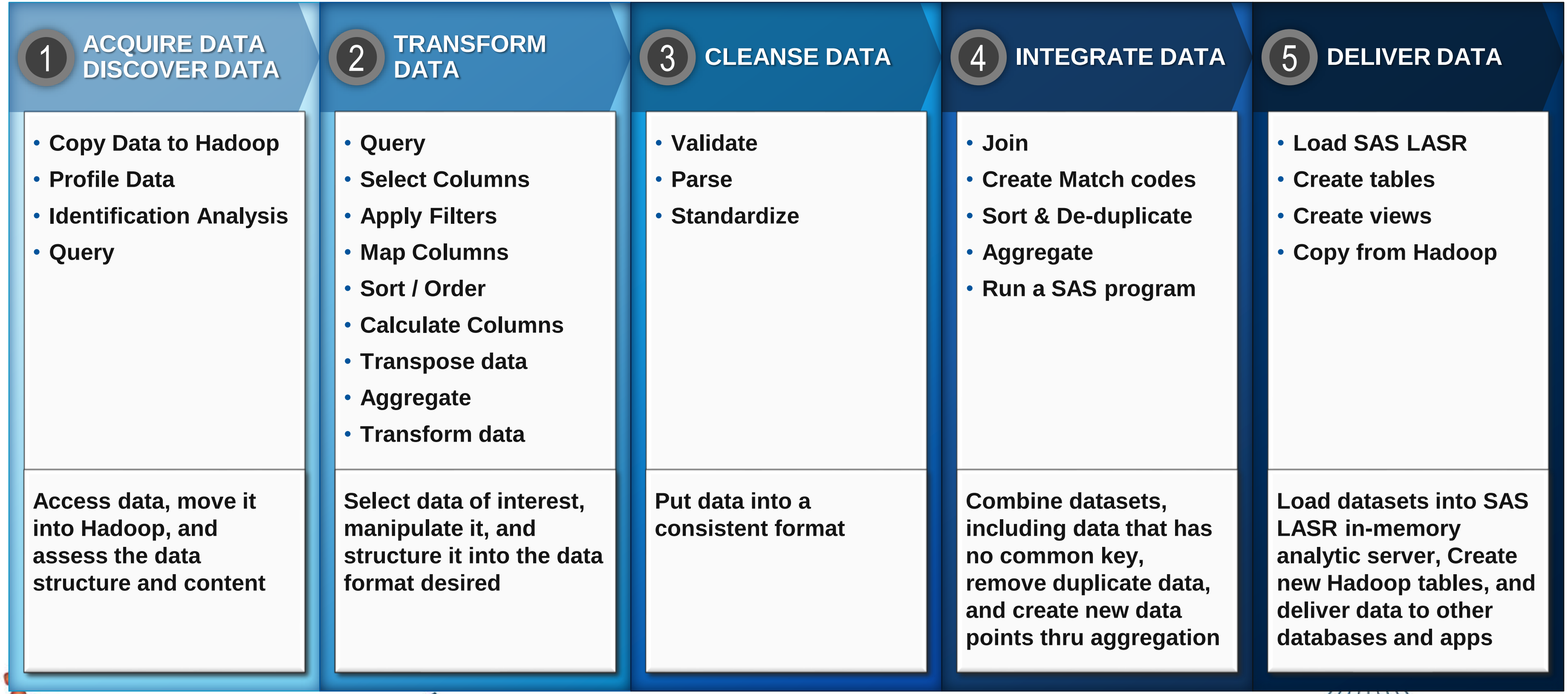
Cleanse Data in Hadoop
Cleanse data in Hadoop by performing data quality transforms

Profile Data
Generate a profile report of the data in a table

Saved Profile Reports
Explore previously generated profile reports

Certified by Hortonworks and Cloudera

Capabilities - SAS Data Loader for Hadoop



1 ACQUIRE DATA DISCOVER DATA

- Copy Data to Hadoop
- Profile Data
- Identification Analysis
- Query

Access data, move it into Hadoop, and assess the data structure and content

2 TRANSFORM DATA

- Query
- Select Columns
- Apply Filters
- Map Columns
- Sort / Order
- Calculate Columns
- Transpose data
- Aggregate
- Transform data

Select data of interest, manipulate it, and structure it into the data format desired

3 CLEANSE DATA

- Validate
- Parse
- Standardize

Put data into a consistent format

4 INTEGRATE DATA

- Join
- Create Match codes
- Sort & De-duplicate
- Aggregate
- Run a SAS program

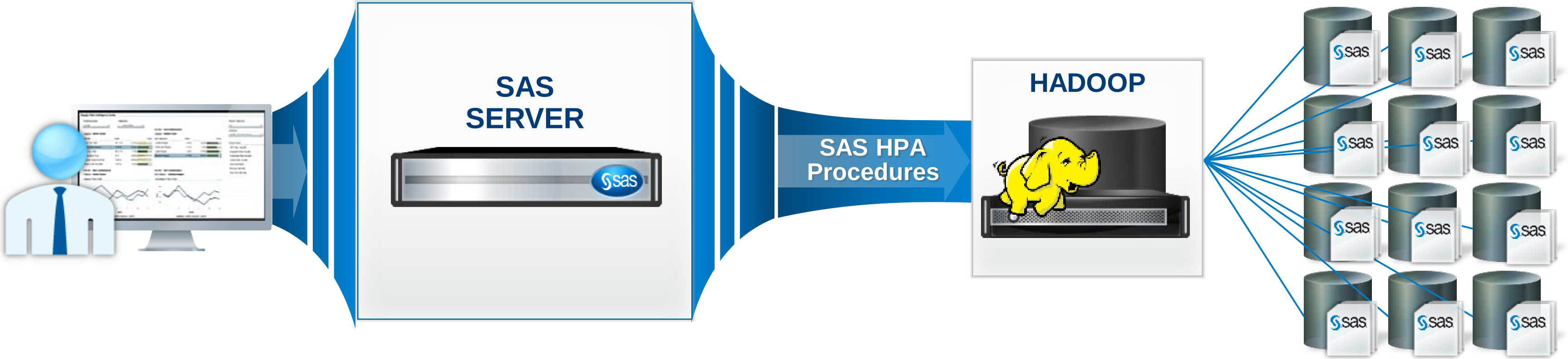
Combine datasets, including data that has no common key, remove duplicate data, and create new data points thru aggregation

5 DELIVER DATA

- Load SAS LASR
- Create tables
- Create views
- Copy from Hadoop

Load datasets into SAS LASR in-memory analytic server, Create new Hadoop tables, and deliver data to other databases and apps

SAS / High Performance Analytics



SAS High-Performance Statistics
SAS High-Performance Data Mining
SAS High-Performance Text Mining
SAS High-Performance Econometrics
SAS High-Performance Forecasting
SAS High-Performance Optimization

SAS / High Performance Analytics

Prepare
Explore / Transform
Model

Prepare

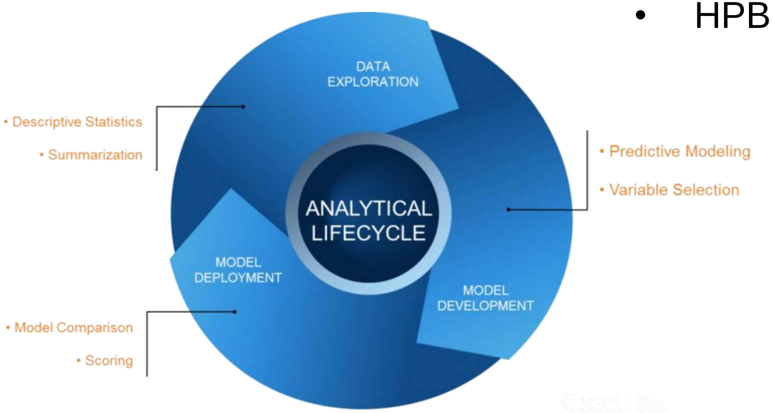
- HPDS2
- HPDMDB
- HPSAMPLE

Explore / Transform

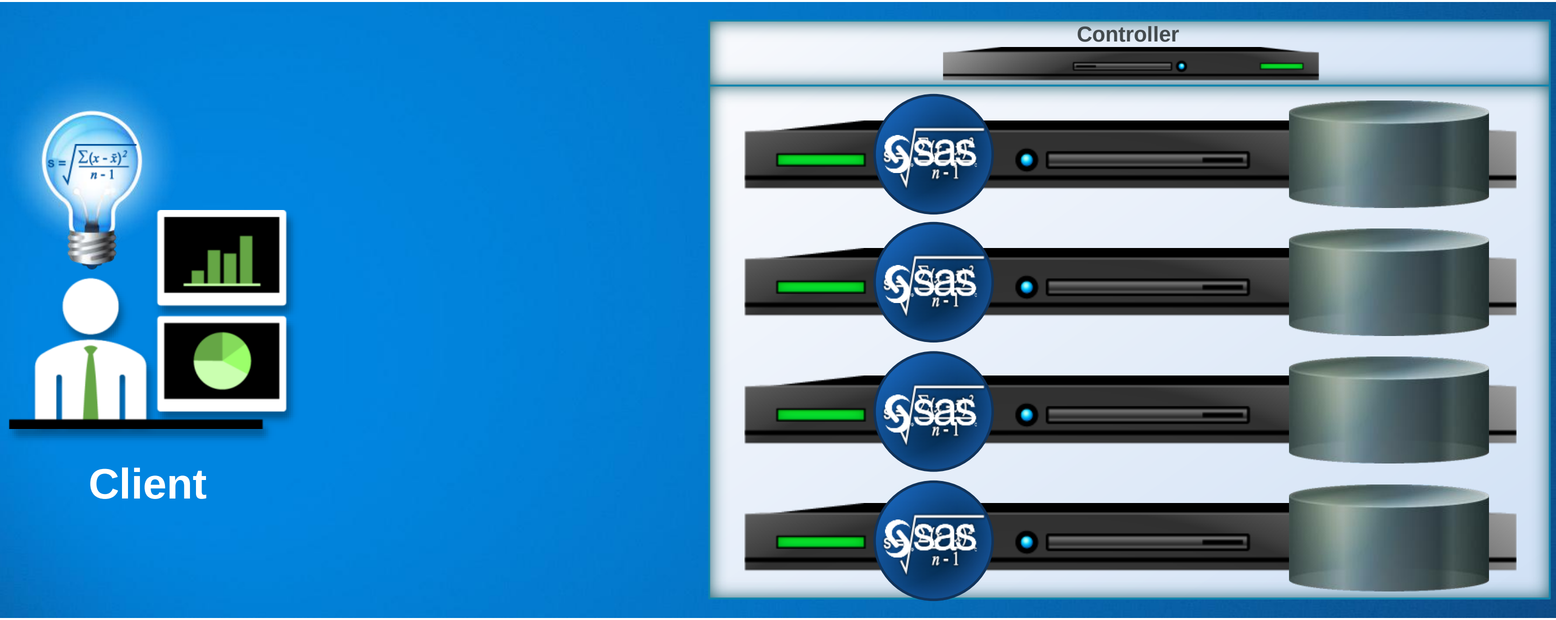
- HPSUMMARY
- HPCORR
- HPREDUCE
- HPIMPUTE
- HPBIN

Model

- HPLOGISTIC
- HPREG
- HPNEURAL
- HPNLIN
- HPCOUNTREG
- HPMIXED
- HPSEVERITY
- HPFOREST
- HPSVM
- HPDECIDE
- HPQLIM
- HPLSO
- HPSPLIT
- HPTMINE
- HPTMSCORE



SAS / High Performance Analytics

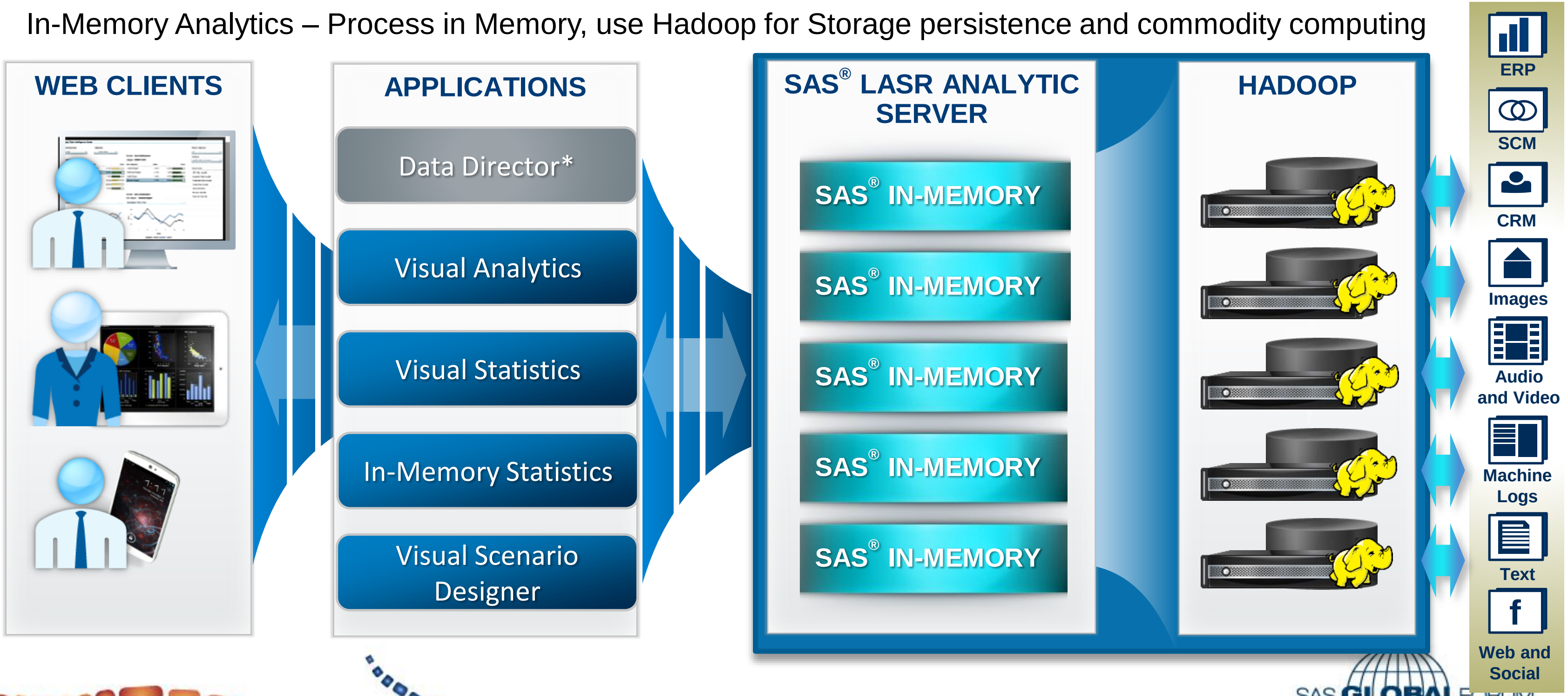


IN-MEMORY(LASR BASED) SOLUTIONS ON HADOOP

4

SAS ANALYTIC HADOOP ENVIRONMENT

In-Memory Analytics – Process in Memory, use Hadoop for Storage persistence and commodity computing



WEB CLIENTS

APPLICATIONS

Data Director*

Visual Analytics

Visual Statistics

In-Memory Statistics

Visual Scenario Designer

SAS® LASR ANALYTIC SERVER

SAS® IN-MEMORY

SAS® IN-MEMORY

SAS® IN-MEMORY

SAS® IN-MEMORY

SAS® IN-MEMORY

HADOOP

ERP

SCM

CRM

Images

Audio and Video

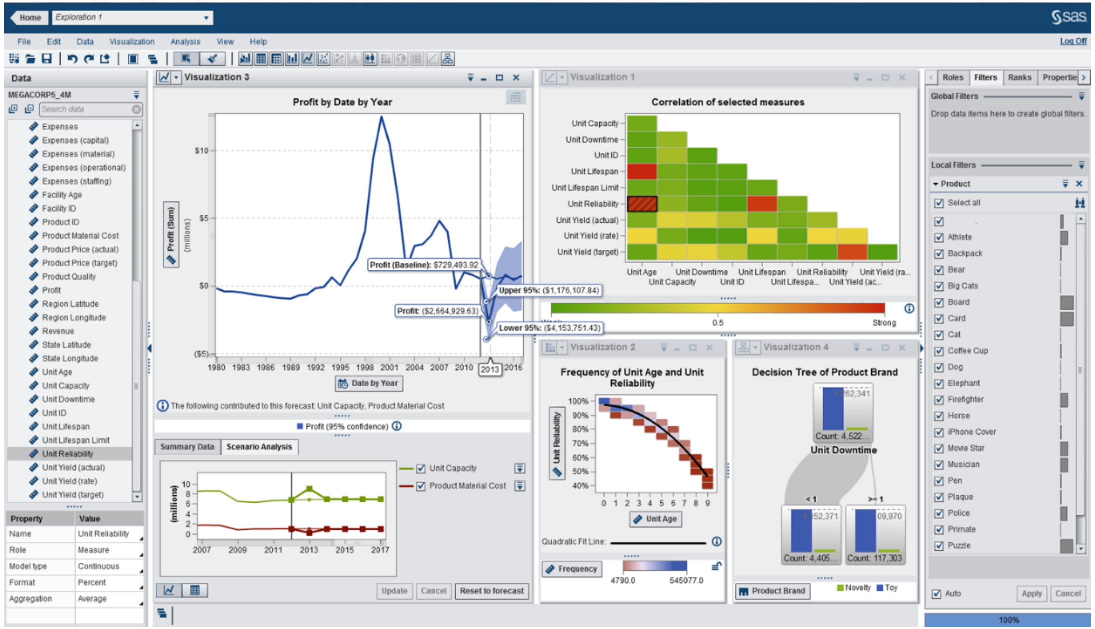
Machine Logs

Text

Web and Social

SAS VISUAL ANALYTICS

- Interactive exploration, dashboards and reporting
- Auto-charting automatically picks the best graph
- Forecasting, scenario analysis, Decision Trees and other analytic visualizations
- Text analysis and content categorization
- Feature-rich mobile apps for iPad® and Android

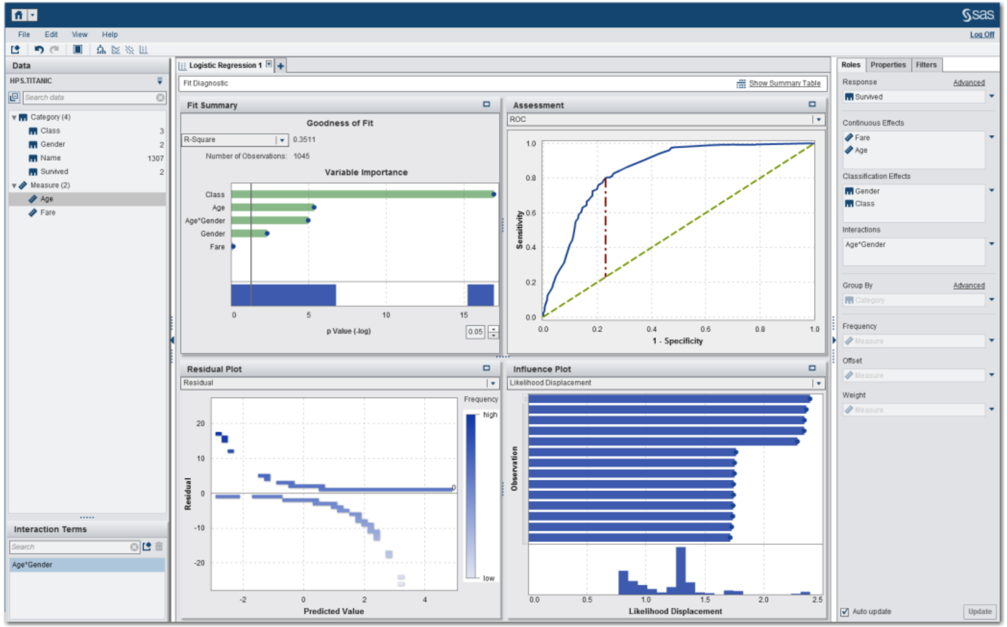


The following contributed to this forecast: Unit Capacity, Product Material Cost

Property	Value
Name	Unit Reliability
Role	Measure
Model type	Continuous
Format	Percent
Aggregation	Average

VA NOW WITH VISUAL STATISTICS

new and improved



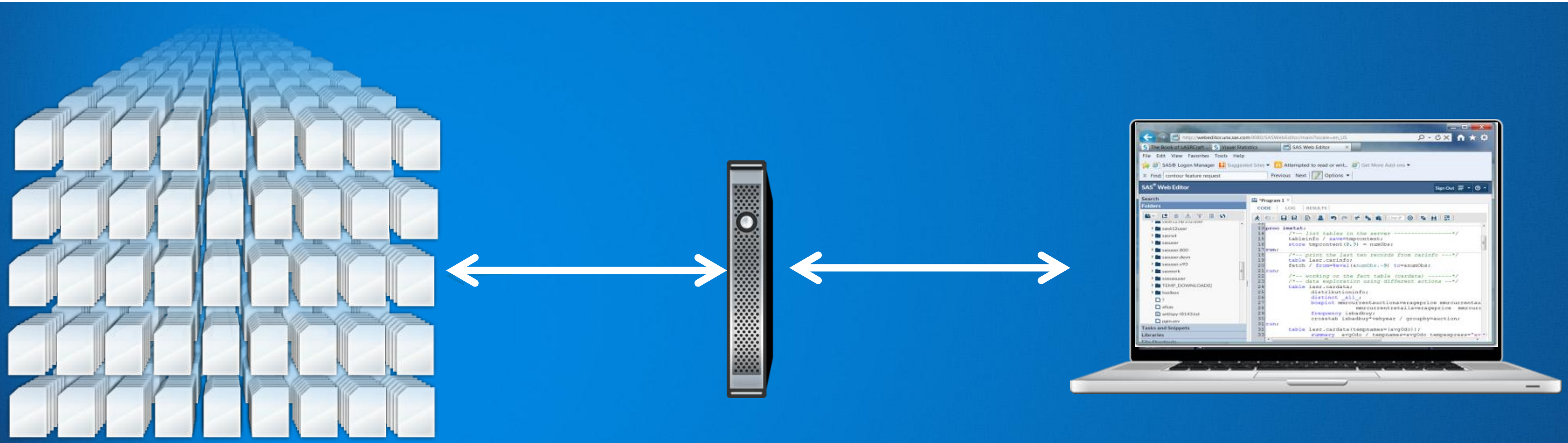
- Interactive, visual application for statistical modeling and classification
- Multiple methods:
  - logistic, Regression, GLM, Trees, Forest, Clustering and more…
- Model comparison and assessment
- Group BY Processing

SAS In-Memory Statistics (IMSTAT) for Hadoop

- Data Prep in Hadoop
  - Parallel Data Step
  - Read / Write Text & HDFS Files
  - Text Processing in Hadoop
  - Structure and Prepare Data For Analysis
- Play nicely in the Hadoop Ecosystem
- Play nicely with SAS Users
  - Libname, Procs & SAS Code

- In-Memory Analytics
  - Interactive “Modern” Analytic Methods
    - » Descriptive Statistics
    - » Statistics
    - » Forecasting
    - » Classification
    - » Text Mining
    - » Optimization
    - » Collaborative Filtering
  - Group By Processing

INTERACTIVE ANALYSIS AT SCALE …



```
proc imstat;
        /*-- list tables in the server ------------------*/
        tableinfo / save=tmpcontent;
        store tmpcontent(2,3) = numObs;
run;
        /*-- print the last ten records from carinfo ---*/
        table lasr.carinfo;
        fetch / from=%eval(&numObs.-9) to=&numObs;
run;
        /*-- working on the fact table (cardata) -------*/
        /*-- data exploration using different actions --*/
        table lasr.cardata;
              distributioninfo;
              distinct _all_;
              boxplot mmrcurrentauctionaverageprice mmrcurrentau
                        mmrcurrentretailaverageprice  mmrcurr
              frequency isbadbuy;
              crosstab isbadbuy*vehyear / groupby=auction;
run;
        table lasr.cardata(tempnames=(avgOdo));
              summary  avgOdo / tempnames=avgOdo tempexpress="av
```

LASR Analytic Server on Hadoop

SAS Server
~ BASE, ODS, STAT, Access, LASR (IMSTAT, RECOMMEND Etc..)

SAS Studio, HTML 5 new Modern Coding Environment

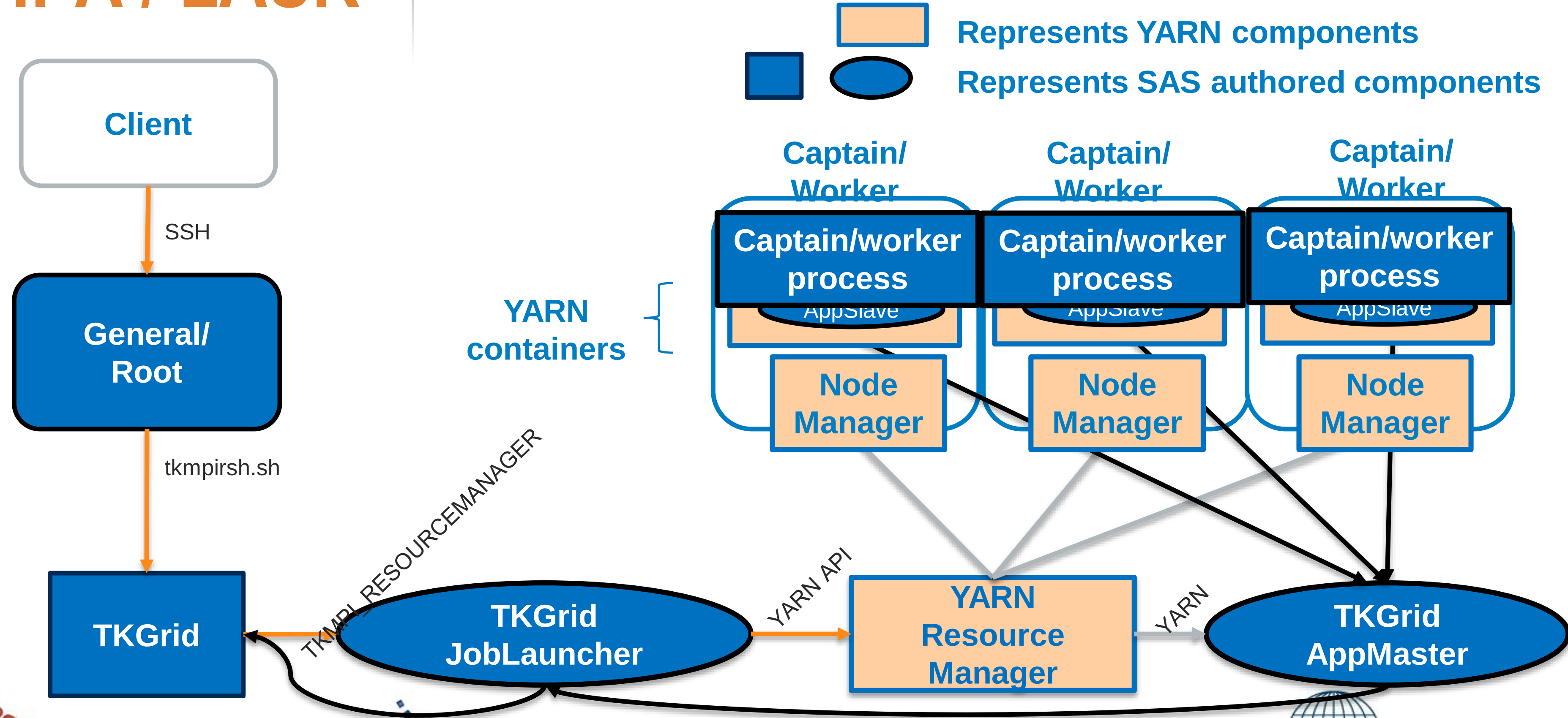
SAS In-Memory Statistics for HAdoop

```
proc imstat;
    /*-- list tables in the server ------------------*/
    tableinfo / save=tmpcontent;
    store tmpcontent(2,3) = numObs;
run;
    /*-- print the last ten records from carinfo ---*/
    table lasr.carinfo;
    fetch / from=%eval(&numObs.-9) to=&numObs;
run;
    /*-- working on the fact table (cardata) -------*/
    /*-- data exploration using different actions --*/
    table lasr.cardata;
         distributioninfo;
         distinct _all_;
         boxplot mmrcurrentauctionaverageprice
            mmrcurrentauctioncleanprice
            mmrcurrentretailaverageprice
            mmrcurrentretailcleanprice /
            groupby=auction;
         frequency isbadbuy;
         crosstab isbadbuy*vehyear / groupby=auction;
run;
    table lasr.cardata(tempnames=(avgOdo));
         summary  avgOdo / tempnames=avgOdo
            tempexpress="avgOdo = vehodo /
            (year(purchdate)-vehyear);";
```

```
proc recommend port=&lasrport;
add / item=movieid
user=userid
rating=rating;

addtable mylasr.ml100k / type=rating
vars=(movieid userid rating);

method svd / factors=10 maxiter=20
maxfeval=100 tech=lbfgs seed=1234
function=L2 lamda=0.1 label="svdlbfgs"
details;

method knn / similarity=pc positive k=20
label="knn1";

method ensemble / details label="em1"
maxiter=10 seed=4321;
run;
```

HPA / LASR

TKGRID INTEGRATION WITH YARN



Version Soup

Baseline support matrix @ 9.4M2

9.4M2	Hive	Impala	Score Accl	Code Accl	DQ Accl	Data Ldr	SPDE
Cloudera 4.5	🟩	🟩	🟩	▨	▨	▨	▨
Cloudera 5.0	🟩	🟩	🟩	🟩 New 9.4M2	🟩 New 9.4M2	🟩 New 9.4M2	🟩
HortonWorks 1.3.2	🟩	▨	🟩	▨	▨	▨	▨
HortonWorks 2.0	🟩	▨	🟩	🟩 New 9.4M2	🟩 New 9.4M2	🟩 New 9.4M2	🟩
Pivotal HD 1.1	🟩	▨	▨	▨	▨	▨	▨
Pivotal HD 2.0	▨	▨	▨	▨	▨	▨	▨
IBM BigInsights 2.1	🟩	▨	▨	▨	▨	▨	▨
MapR V3	🟩	▨	▨	▨	▨	▨	▨

🟩 Supported at 9.4M2

▨ Not Supported

* Minus Yarn Integration

Net New Product

Baseline support matrix – on track for 94m3

			EP					New @M3
9.4M3	Hive	Impala	Score Accl	**Code Accl**	**DQ Accl**	**Data Ldr**	SPDE	**SAS Grid**
Cloudera 4.x	🟩	🟩	🟩	⬜	⬜	⬜	⬜	⬜
Cloudera 5.x	🟩	🟩	🟩	🟩	🟩	🟩	🟩	M3
HortonWorks 1.x	🟩	⬜	🟩	⬜	⬜	⬜	⬜	⬜
HortonWorks 2.x	🟩	⬜	🟩	🟩	🟩	🟩	🟩	M3
Pivotal HD 2.x	M3	⬜	M3	M3	M3	*	M3	-
IBM BigInsights 3.x	🟩	⬜	**	**	**	**	**	**
MapR V4.x	🟩	LA - M3	M3	M3	M3	M3	M3	M3

🟩	Supported
🟨	On Track for 9.4M3
⬜	No Plans
🟦	Research Needed for M3
	** IBM Issues
	Net New Product
	** MapR support for Impala

* PivotalHD is behind on Yarn, MR2 and Hive integration necessary for support – currently it is a stretch to make M3 with Data Loader.
** Limitations with IBM BigInsights 3.x - doesn't support MR2, Yarn. Parallel feeds supported.
** Note: SAS/Access to Impala is Limited Availability

In-memory support matrix

		LASR Analtyic Server 2.4			
9.4M2	HPA	VA	IMSTAT	**VS**	VSD
Cloudera 4.5	🟩	🟩	🟩	🟩	🟩
Cloudera 5.0	🟩	🟩	🟩	🟩	🟩
HortonWorks 1.3.2	🟩	🟩	🟩	🟩	🟩
HortonWorks 2.0	🟩	🟩	🟩	🟩	🟩
Pivotal HD 1.1	▨	🟩	🟩	🟩	🟩
Pivotal HD 2.0	▨	🟩	🟩	🟩	🟩
IBM BigInsights 2.1	▨	🟩 *	🟩 *	🟩 *	🟩 *
MapR V3	▨	▨	▨	▨	▨

🟩 Supported at 9.4M2

▨ Not Supported

* Minus Yarn Integration

Net New Product

In-memory support matrix on track for 94m3

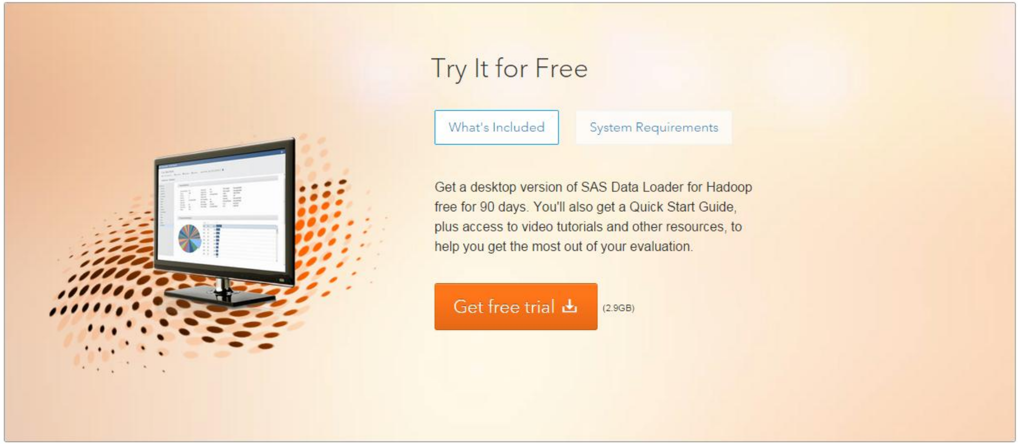
		LASR Analtyic Server 2.x			
9.4M3	HPA	VA	IMSTAT	**VS**	VSD
Cloudera 4.x	🟩	🟩	🟩	🟩	🟩
Cloudera 5.x	🟩	🟩	🟩	🟩	🟩
HortonWorks 1.x	🟩	🟩	🟩	🟩	🟩
HortonWorks 2.x	🟩	🟩	🟩	🟩	🟩
Pivotal HD 2.x	🟨 M3	🟩	🟩	🟩	🟩
IBM BigInsights 3.x	🟨 M3	🟩 **	🟩 **	🟩 **	🟩 **
MapR V4.x	🟨 M3	🟨 M3	🟨 M3	🟨 M3	🟨 M3

- 🟩 Supported
- 🟨 On Track for 9.4M3
- ▨ No Plans
- 🟦 Research Needed for M3
- ** IBM Issues
- **Net New Product**
- ** MapR support for Impala

** Limitations with IBM BigInsights 3.x - doesn’t support MR2. Parallel feeds supported.

GIVE IT A TRY YOURSELF ! DATA LOADER FOR HADOOP TRIAL

- SAS Data Loader for Hadoop & 90 Day Trial - Launched February ’15
  - » Transpose, Filter, Query, Aggregate, Sort, Join
  - » Sqoop Integration
  - » In-Hadoop Data Quality
  - » In-Hadoop Data Profiling
- *What you’ll need to get started*
  - » *VMware Player*
  - » *Cloudera QuickStart VM CDH 5.3 ;*
  - » *Hortonworks HDP Sandbox 2.2*
  - » *Data Loader Vapp 2.2*
- What's available to help
  - » Online help + phone support
  - » Online doc
  - » Dedicated group
  - » Tons of How-do- style quick videos



Try It for Free

Get a desktop version of SAS Data Loader for Hadoop free for 90 days. You'll also get a Quick Start Guide, plus access to video tutorials and other resources, to help you get the most out of your evaluation.

https://communities.sas.com/groups/sas-data-loader

Thank You!

4 short years and…

- Hadoop is on the opening session, 2 Hadoop CEO’s share the panel discussion at Exec Track
- 50% of the people attending an executive conference talk have an active Hadoop project
- SAS has built a family of applications that take full advantage of Hadoop as an analytics platform

Paul.Kent@SAS.com

@hornpolish

paulmkent

SPD ***Server*** with Hadoop

new and improved

- SPDS 5.2 server on Linux
- Support for running on Cloudera 5.2
- Only server and libname parameter files need to change
- Read/write/update support
- Kerberos support
- Limited WHERE pushdown
- Parallel read support without a WHERE clause