HITACHI
Inspire the Next

SQL on Hadoopのホントのところ

Impala vs Hive on Tez vs Drill

2017/09/09

株式会社 日立製作所 OSSソリューションセンタ

木下 翔伍

講演者

HITACHI
Inspire the Next

木下 翔伍 ／ Kinoshita Shogo

- エンタープライズ向け**ビッグデータ**関連ソリューション検討・開発
  Hadoopエコシステム(Spark, Hive 等)の技術検証含む

例えば、

スマートメーター(デジタル電力計)1,000万台のデータを扱うユースケースで
Sparkの性能検証

- 検証結果の一部が書籍に

  Apache Spark ビッグデータ性能検証
  (ISBN 9784295001126)

- 今日はSQL on Hadoop
  クエリエンジンの話をします

HITACHI
Inspire the Next

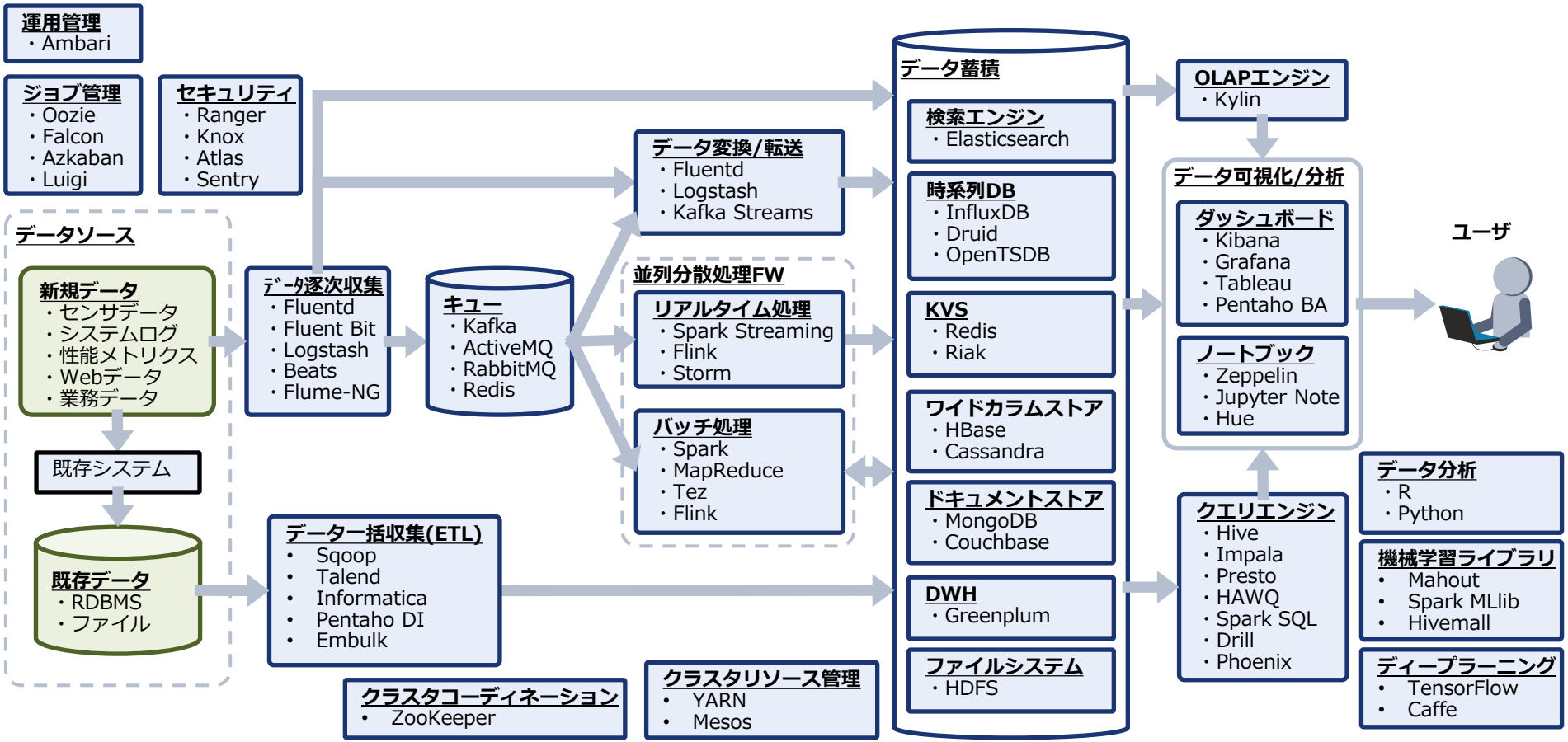
Contents

HITACHI
Inspire the Next

1. Motivation

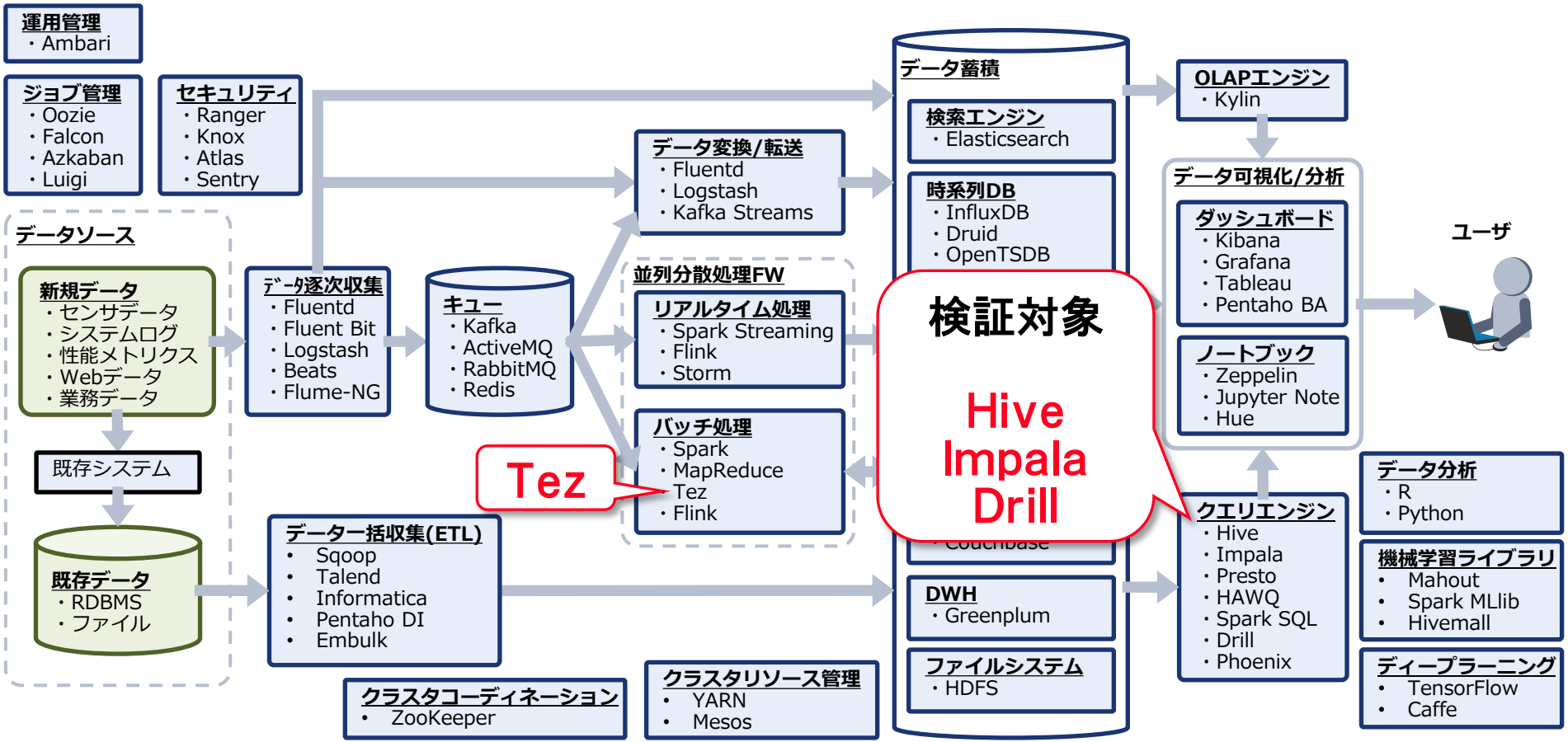
1-1 ビッグデータ処理基盤はOSSの組み合わせが一般的

HITACHI
Inspire the Next



1-1 ビッグデータ処理基盤はOSSの組み合わせが一般的

HITACHI
Inspire the Next

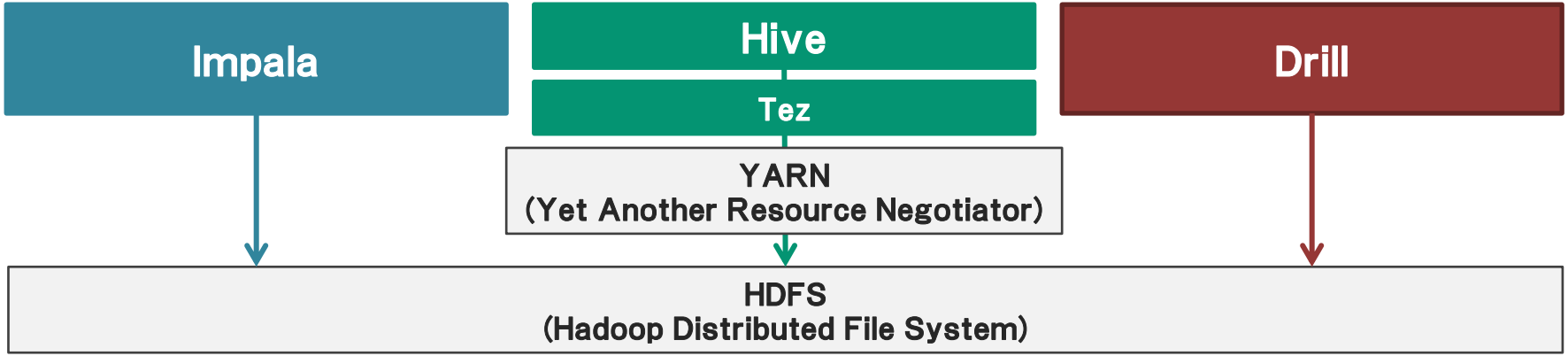


1-2 Hadoop上のSQLクエリエンジン

HITACHI
Inspire the Next

◆ クエリエンジンとは

データを操作する指示を(主にSQLで)受け、それに応じたデータ処理機能を提供



Hadoop向けのネイティブな分析データベース

- HDFSに直接アクセス
- インメモリ処理

データ処理アプリケーションのフレームワーク

- HDFSアクセス頻度低減
- コンテナを一定時間保持
- 既存Hiveアプリは改修不要

スキーマフリーなSQLクエリエンジン

- 非構造化データも取扱い
- 多様なデータソース(クラウド/オブジェクトストレージ)に対応

1-3 困りごと

何を基準にクエリエンジンを選んでよいかわからない

実際に自分で試して確認する！

HITACHI
Inspire the Next

2. 検証内容の検討

2-1 検証内容

HITACHI
Inspire the Next

◆ 目的

どのクエリエンジンを選べばよいかわかるようにする

方針	クエリ処理性能がより高いクエリエンジンを選ぶ

- クエリエンジンどうしで明らかな性能差はあるのか
- 同じクエリエンジンでもデータ量で性能は変わるのか

◆ 検証内容

検証項目	検証内容	比較対象
クエリエンジンの性能差	クエリエンジンの間に処理性能の差があるかどうか検証する	クエリ処理時間
処理性能の安定性 (データ量によらない性能)	データ量を変動させて処理性能に低下・向上があるかどうか検証する	スループット[データ量／時間]

2-2 実験内容

HITACHI
Inspire the Next

◆ クエリの概要

TPC-DS	意思決定支援(Decision Support)ソリューション向けのベンチマーク ユースケースに基づいて99個の処理(クエリ)が定義

- 意思決定支援はビッグデータ利活用のひとつ
- 定義されたクエリを実行すれば、あるシナリオに沿った分析処理をしたことになる

たとえばQuery 3 の場合・・・特定メーカーのブランドのアイテムごとに、ある年の特定の月における合計販売金額を算出する

◆ 使用するSQL

- RDB含めてクエリエンジンごとに差異が大きい
- あるSQLを別エンジンで実行するには改修が必要となることもしばしば
- 本検証ではImpala SQLとHiveQLの2種類のSQLを使用

(※)本検証で活用したSQL

https://github.com/cloudera/impala-tpcds-kit/tree/master/queries

https://github.com/hortonworks/hive-testbench/tree/hive14/sample-queries-tpcds

2-3 処理は3種類に分類

HITACHI
Inspire the Next

◆ 特徴による処理(クエリ)の分類

分類	クエリの特徴	本検証で用いたTPC-DSクエリの番号
interactive	ファクトテーブル1つのみのスタースキーマを使った処理	3, 12, 15, 19, 26, 43, 52, 55, 82, 84, 91, 96
data mining	BI, ETLツールと連携を前提に大量データを返す処理	34, 73, 98
deep reporting	複数ファクトテーブルや大きな中間データセットを扱うなど複雑な処理	20, 21, 40, 45, 46, 49, 50, 58, 66, 68, 76, 79, 89, 93, 97

http://hortonworks.com/blog/benchmarking-apache-hive-13-enterprise-hadoop/ を参考に編集

- TPC-DS 全99クエリのうち上記クエリをImpala SQLとHiveQLで実行(計60クエリ)

スタースキーマとは・・・DWH(データウェアハウス)でよく用いられるスキーマ(データモデル)
主要データ(ファクト)を集めたファクトテーブルを中心にして、ファクトの詳細なレコードを格納するディメンションテーブルから成る

2-4 検証環境（物理構成）

HITACHI
Inspire the Next

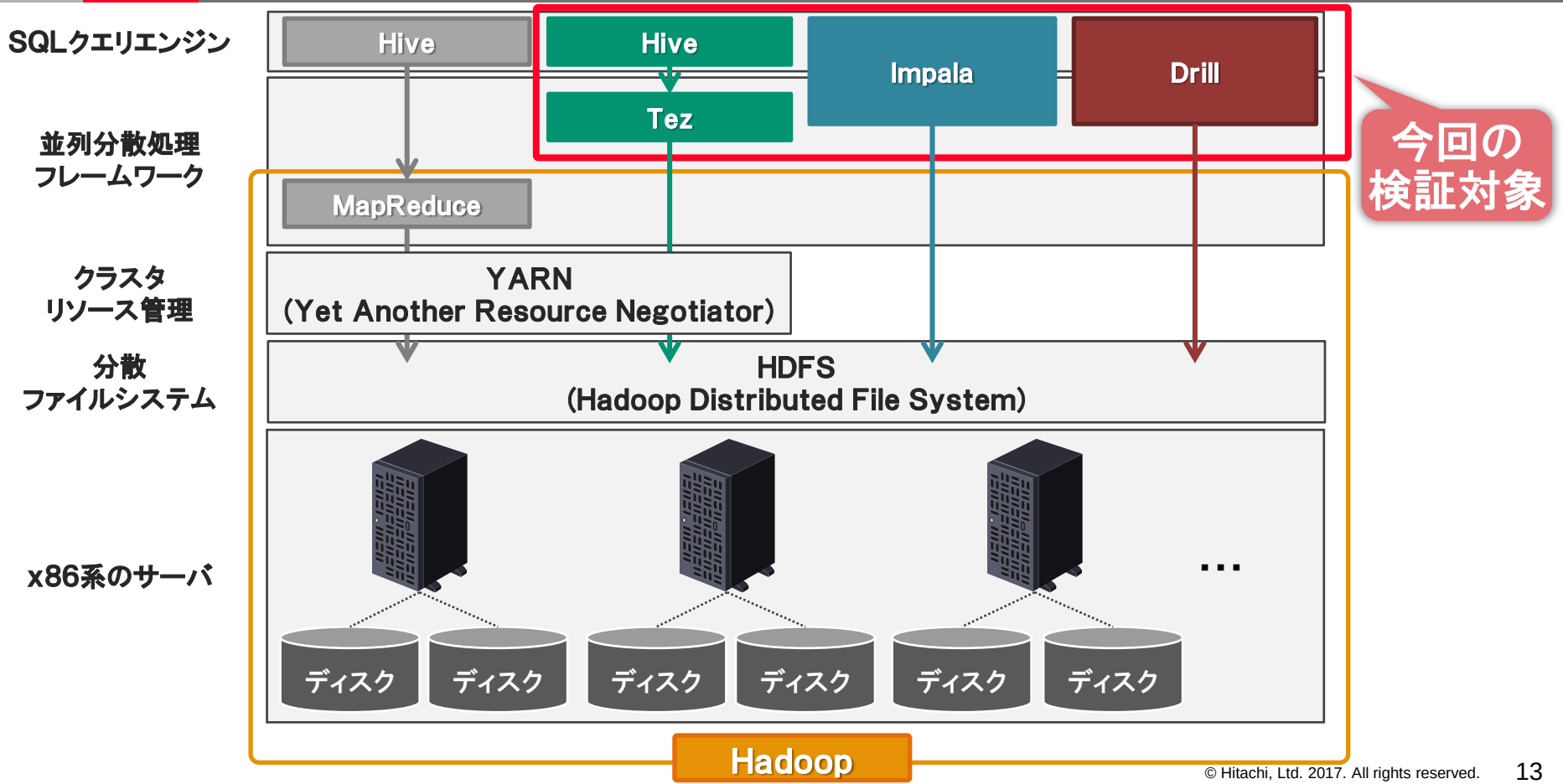
◆ マシン一覧

	スペック
CPUコア数	2 コア
メモリ容量	16 GB
ディスク台数	1 台
1ディスク容量	160 GB

	1ノード	6ノード合計
CPUコア数	40 コア	240 コア
メモリ容量	384 GB	2, 304 GB
ディスク台数	10 台	100 台
1ディスク容量	1, 200 GB	ー
ディスク合計容量	12 TB （12, 000GB）	72 TB （72, 000GB）

2-5 検証環境(論理構成)

HITACHI
Inspire the Next



2-6 主なパラメータ設定

HITACHI
Inspire the Next

◆ Impala

- CDH5.9
- 管理ソフトの初期設定を活用
- mem_limit = -1（無制限）

```
yarn.nodemanager.resource.cpu-vcores = 40
yarn.nodemanager.resource.memory-mb = 248030
yarn.scheduler.minimum-allocation-mb = 1024
yarn.scheduler.maxmum-allocation-mb = 248030
yarn.scheduler.mimimum-allocation-vcores = 1
yarn.scheduler.maximum-allocation-vcores = 40
```

◆ Hive on Tez

- HDP2.5.3
- マニュアルインストール
- hive.execution.engine = tez

```
yarn.nodemanager.resource.cpu-vcores = 35
yarn.nodemanager.resource.memory-mb = 294919
yarn.scheduler.minimum-allocation-mb = 1024
yarn.scheduler.maxmum-allocation-mb = 294919
yarn.scheduler.mimimum-allocation-vcores = 1
yarn.scheduler.maximum-allocation-vcores = 35
```

Hive on Tezでは設定ファイルが空白であったため、
本検証前にパラメータチューニングを実施し設定値を求めた

◆ Apache Drill1.9

- Hive on Tez検証環境に追加構築（追加パラメータは次の2つでその他は同じ）
- DRILL_MAX_HEAP = 4GB
- DRILL_MAX_DIRECT_MEMORY = 8GB

HITACHI
Inspire the Next

3. 結果と考察

3-1 本検証の取り組み内容

HITACHI
Inspire the Next

◆ 目的

どのクエリエンジンを選べばよいかわかるようにする

方針	クエリ処理性能がより高いクエリエンジンを選ぶ

◆ 検証内容

検証項目	検証内容	比較対象
クエリエンジンの性能差	クエリエンジンの間に処理性能の差があるかどうか検証する	クエリ処理時間
処理性能の安定性 (データ量によらない性能)	データ量を変動させて処理性能に低下・向上があるかどうか検証する	スループット[データ量／時間]

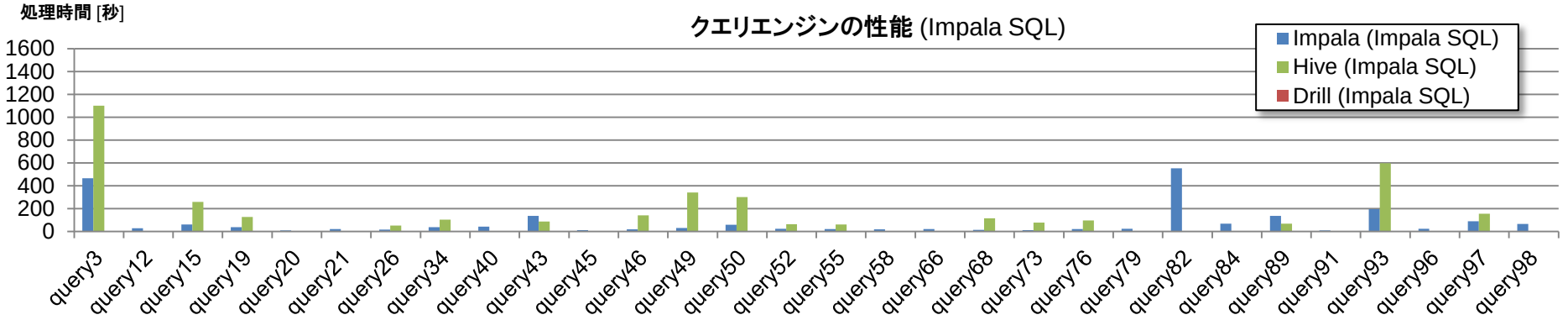
◆ 処理と実験の内容

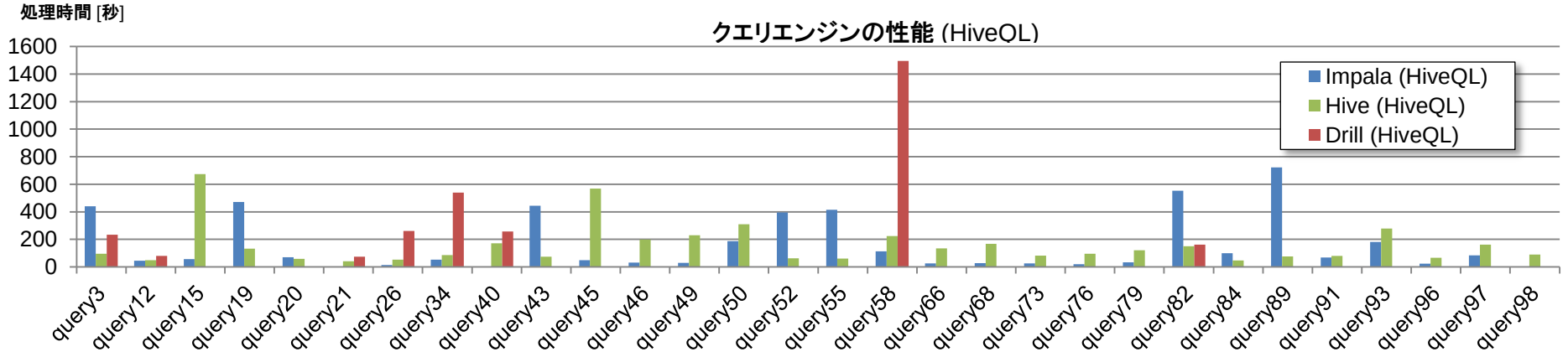
テキスト形式1，000 GBのデータを **TPC-DS** のクエリ

Impala SQLとHiveQLで計60個実行して、所要時間を計測

3-2 結果(クエリの処理時間)

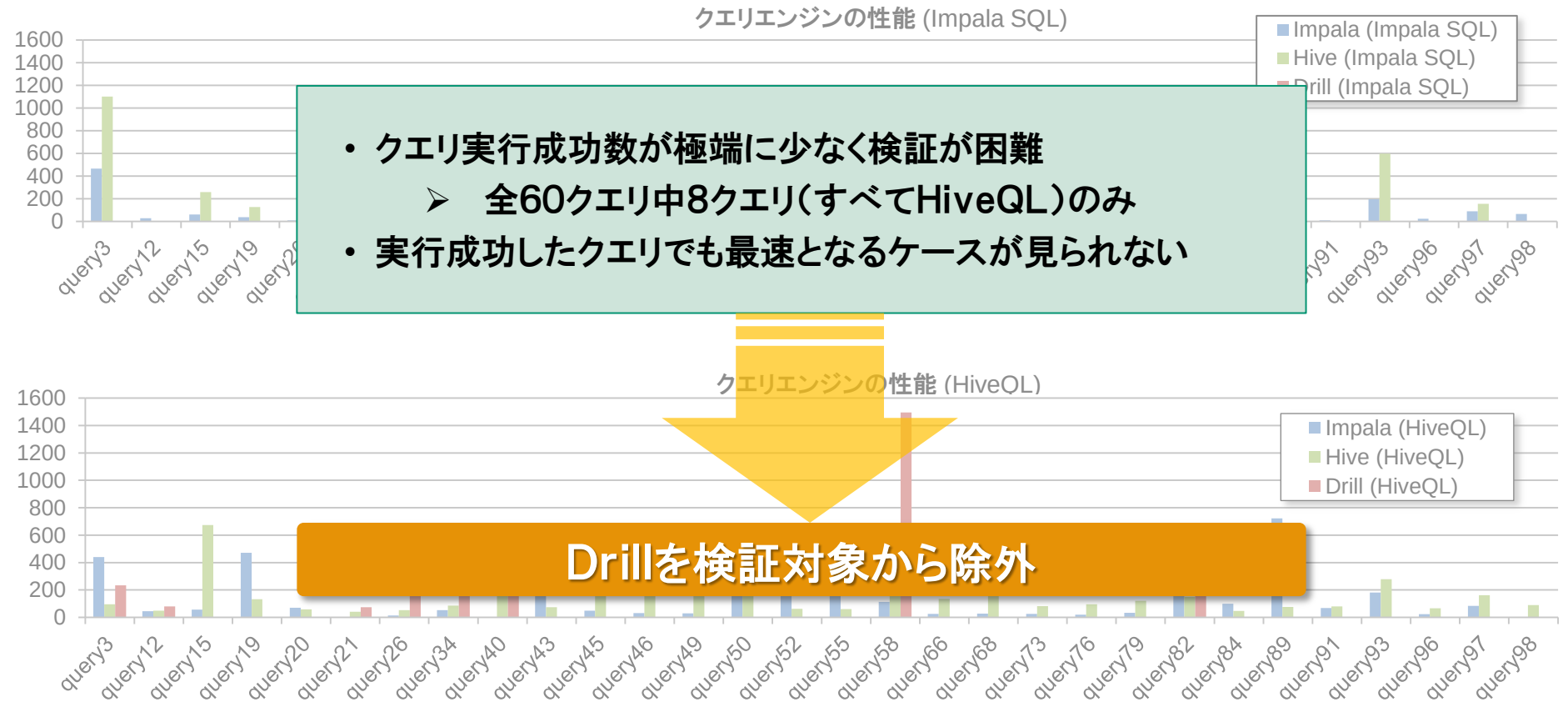
HITACHI
Inspire the Next





3-2 結果(クエリの処理時間)

HITACHI
Inspire the Next



- クエリ実行成功数が極端に少なく検証が困難
  - ➢ 全60クエリ中8クエリ(すべてHiveQL)のみ
- 実行成功したクエリでも最速となるケースが見られない

Drillを検証対象から除外

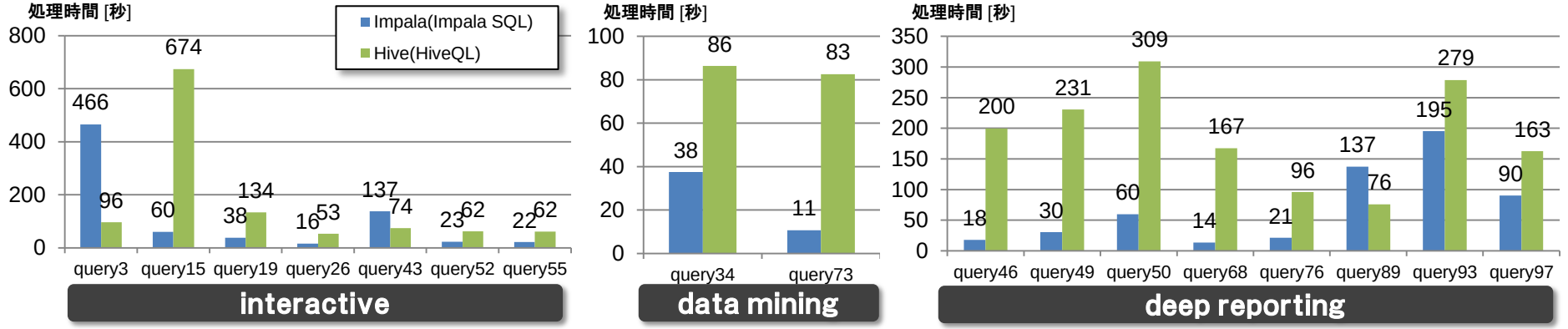
3-3 検証1. クエリエンジンの性能差

HITACHI
Inspire the Next

目的	各クエリエンジンがどのような処理に適しているか検証する
実施内容	各処理(クエリ)の処理に要した時間を計測し比較する
検証条件	• TPC−DS 1, 000 GB • テキストファイル

◆ 所要時間 Impala(Impala SQL) vs Hive(HiveQL)

※ 値は小さいほうが良い



クエリエンジンどうしを比較するとHiveよりもImpalaのほうが高速処理できる傾向がある

3-4 傾向にあてはまらないクエリを処理するとき何が起きているのか

HITACHI
Inspire the Next

◆ ImpalaよりもHiveが高性能だったクエリ

- クエリ番号3, 43 [Impala SQL]　　interactive
- クエリ番号89 [Impala SQL]　　deep reporting

◆ Impalaでクエリ実行時に時間を費やした処理の傾向

クエリ番号 Impala SQL	分類	最も時間を要した処理 （a）	2番目に時間を要した処理	処理時間全体に対する（a）の割合
Query3	interactive	HASH JOIN	EXCHANGE	50%
Query43	interactive	HASH JOIN	HASH JOIN	60%
Query89	deep reporting	HASH JOIN	SCAN HDFS	75%

HASH JOIN（テーブルの結合）に著しく時間を要している

3-4 傾向にあてはまらないクエリを処理するとき何が起きているのか

HITACHI
Inspire the Next

◆ Impalaのクエリ実行計画

```
Operator          #Hosts   Avg Time   Max Time   #Rows  Est. #Rows   Peak Mem  Est. Peak Mem  Detail
------------------------------------------------------------------------------------------------------------------
11:MERGING-EXCHANGE      1   35.763us   35.763us    100         100         0       -1.00 B  UNPARTITIONED
06:TOP-N                 1  537.728us  537.728us    100         100   36.00 KB        8.40 KB
10:AGGREGATE             1    2.702ms    2.702ms    112          -1    2.36 MB      128.00 MB  FINALIZE
09:EXCHANGE              1  156.805us  156.805us    112          -1         0               0  HASH(s_store_name,s_store_id)
05:AGGREGATE             1    1s522ms    1s522ms    112          -1    9.62 MB      128.00 MB  STREAMING
04:HASH JOIN             1    1s604ms    1s604ms  29.62M         -1    9.08 MB        2.00 GB  INNER JOIN, BROADCAST
|--08:EXCHANGE           1   17.228us   17.228us    224          -1         0               0  BROADCAST
|  02:SCAN HDFS          1    3.797ms    3.797ms    224          -1  321.00 KB       32.00 MB  tpcds_text_100.store
03:HASH JOIN             1    8s172ms    8s172ms  54.43M         -1    4.35 GB        2.00 GB  INNER JOIN, BROADCAST
|--07:EXCHANGE           1    1s150ms    1s150ms  54.43M         -1         0               0  BROADCAST
|  01:SCAN HDFS          6  861.031ms  886.501ms  54.43M         -1    1.15 GB        6.88 GB  tpcds_text_100.store_sales
00:SCAN HDFS             1   27.376ms   27.376ms     364         -1   18.06 MB       48.00 MB  tpcds_text_100.date_dim
```

Query43 (Impala SQL, interactive)クエリ実行計画の例(抜粋)

03:HASH JOIN は処理時間全体の約60%

				処理時間全体に対する(a)の割合
				50%
Query43	interactive	HASH JOIN	HASH JOIN	60%
Query89	deep reporting	HASH JOIN	SCAN HDFS	75%

見積りメモリ量を最大メモリ量(実使用量)が上回っている

3-5 傾向にあてはまるクエリを処理するとき何が起きているのか

HITACHI
Inspire the Next

◆ Impalaでクエリ実行時に時間を費やした処理の傾向（抜粋）

クエリ番号 Impala SQL	分類	最も時間を要した処理 （a）	2番目に時間を要した処理	処理時間全体に対する（a）の割合
Query29	interactive	SCAN HDFS	HASH JOIN	10%
Query55	interactive	SCAN HDFS	HASH JOIN	39%
Query34	data mining	HASH JOIN	SCAN HDFS	15%
Query97	deep reporting	AGGREAGATION	AGGREAGATION	41%

3-5 傾向にあてはまるクエリを処理するとき何が起きているのか

HITACHI
Inspire the Next

◆ Impalaでクエリ実行時に時間を費やした処理の傾向(抜粋)

クエリ番号 Impala SQL	分類	最も時間を要した処理 (a)	2番目に時間を要した処理	処理時間全体に対する(a)の割合
Query29	interactive	SCAN HDFS	HASH JOIN	10%
Query55	interactive	SCAN HDFS	HASH JOIN	39%
Query34	data			
Query97	deep re			

03:HASH JOIN は処理時間全体の約37%

01:SCAN HDFS は処理時間全体の約39%

◆ Impalaのクエリ実行計画

```
Operator          #Hosts   Avg Time   Max Time   #Rows  Est. #Rows   Peak Mem  Est. Peak Mem  Detail
------------------------------------------------------------------------------------------------------------------
06:TOP-N               1  301.315us  301.315us     100         100   20.00 KB       2.64 KB
10:AGGREGATE           1    1.880ms    1.880ms     550          -1    2.35 MB     128.00 MB  FINALIZE
09:EXCHANGE            1  352.791us  352.791us     550          -1          0             0  HASH(i_brand,i_brand_id)
05:AGGREGATE           1    8.364ms    8.364ms     550          -1    1.59 MB     128.00 MB  STREAMING
04:HASH JOIN           1  309.251ms  309.251ms  82.76K          -1    1.15 MB       2.00 GB  INNER JOIN, BROADCAST
|--08:EXCHANGE         1   79.318us   79.318us   1.88K          -1          0             0  BROADCAST
|  02:SCAN HDFS        1   78.030ms   78.030ms   1.88K          -1   40.05 MB     128.00 MB  tpcds_text_100.item
03:HASH JOIN           1    1s545ms    1s545ms   8.80M          -1  801.06 MB       2.00 GB  INNER JOIN, BROADCAST
|--07:EXCHANGE         1  304.554ms  304.554ms   8.80M          -1          0             0  BROADCAST
|  01:SCAN HDFS        6    1s739ms    2s458ms   8.80M          -1    1.04 GB       6.88 GB  tpcds_text_100.store_sales
00:SCAN HDFS           1   26.227ms   26.227ms      30          -1   10.04 MB      48.00 MB  tpcds_text_100.date_dim
```

Query55 (Impala SQL, interactive)クエリ実行計画の例(抜粋)

見積りメモリ量の範囲内で最大メモリ量(実使用量)が収まっている

3-6 処理内容によって向き不向きがある

HITACHI
Inspire the Next

分類ごとに要する1クエリあたりの処理時間の平均一覧

クエリエンジン	SQL	クエリ処理平均時間（interactive）	クエリ処理平均時間（data mining）	クエリ処理平均時間（deep reporting）
Impala	Impala SQL	109 秒	24 秒	71 秒
Hive on Tez	HiveQL	238 秒	84 秒	328 秒

- エンジンによって平均的に短時間で処理できる分類が異なる
  - Impalaでは　data mining ＜ deep reporting ＜ interactive
  - Hive on Tezでは　data mining ＜ interactive ＜ deep reporting
- エンジンによらず分類の処理平均時間が同じならば、その分類に時間を要する/要しない処理が集まっていたと考えられる

クエリエンジンには得意な（向いている）処理がある

3-7 処理によって適したクエリエンジンが異なる

HITACHI
Inspire the Next

HiveよりもImpalaのほうが高性能な傾向があるが、

- 傾向にあてはまらないクエリがある
- クエリエンジンによって得意な処理内容がある

クエリエンジンの性能特性

Impalaの特性	• 複雑な処理（複数回のJOIN等）を比較的短時間で処理できる • メモリ量が十分でないとき、著しく性能低下
Hive on Tezの特性	• 簡素な処理（検索や数値集約等）について比較的短時間で処理できる

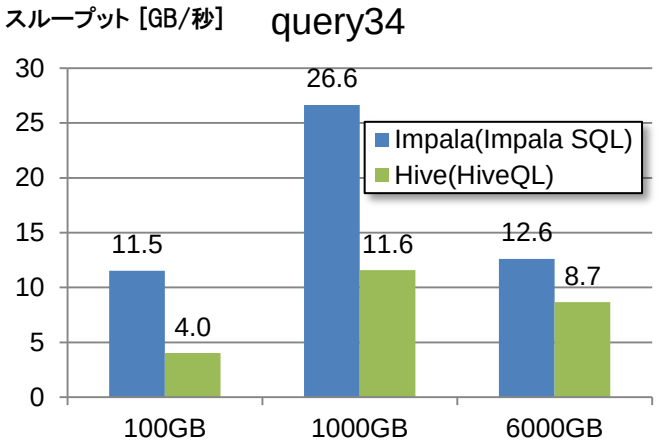
3-8 検証2. 処理性能の安定性

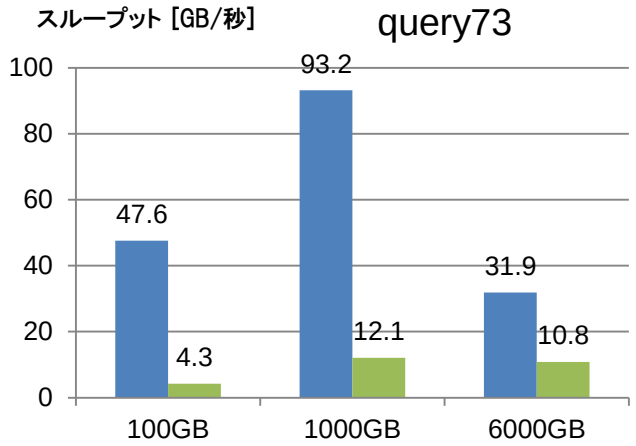
data mining

HITACHI
Inspire the Next

目的	データ量によらず安定した処理性能であるかどうかを検証する
実施内容	処理時間を基に算出したスループット[データ量(GB) / 秒]を比較する
検証条件	• TPC−DS 100GB / 1, 000 GB / 6, 000 GB • テキストファイル

◆ スループット Impala(Impala SQL) vs Hive(HiveQL)





※ 値は大きいほうが良い

3-8 検証2. 処理性能の安定性

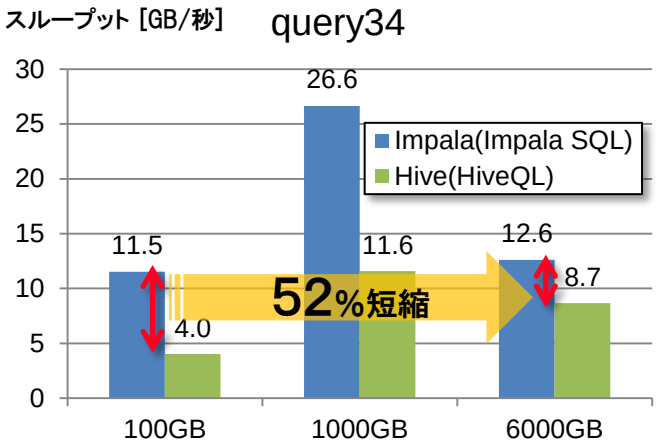
data mining

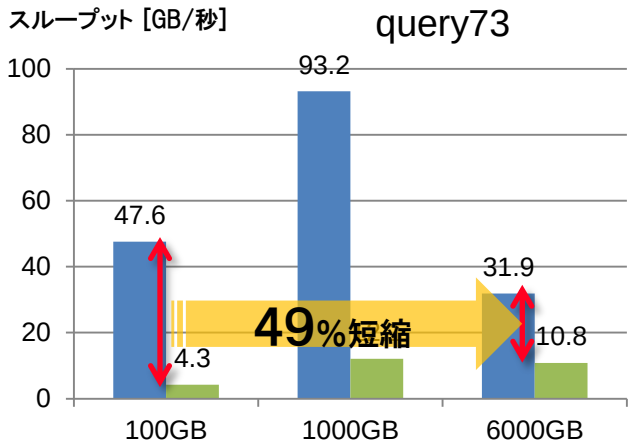
HITACHI
Inspire the Next

目的　データ量によらず安定した処理性能であるかどうかを検証する

data miningでは
- HiveよりもImpalaのほうがスループットが高い
- データ量100GBに比べて6, 000GBでは両者のスループットの差が縮小
  - Query34では約52%短縮
  - Query73では約49%短縮

◆ 結



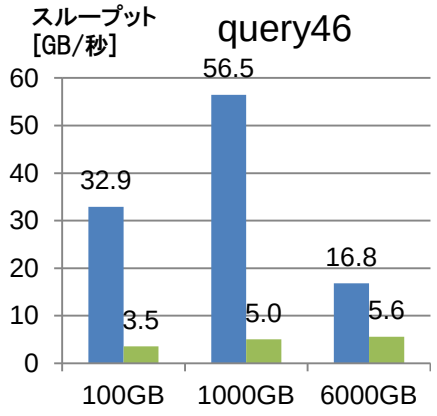


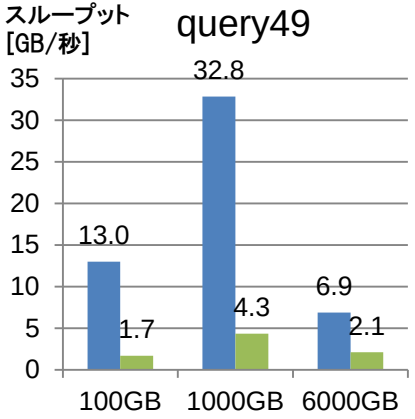
※ 値は大きいほうが良い

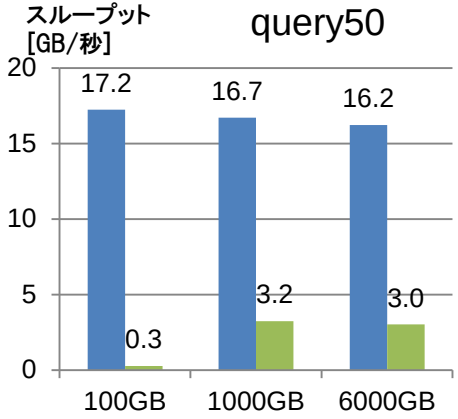
3-8 検証2. 処理性能の安定性

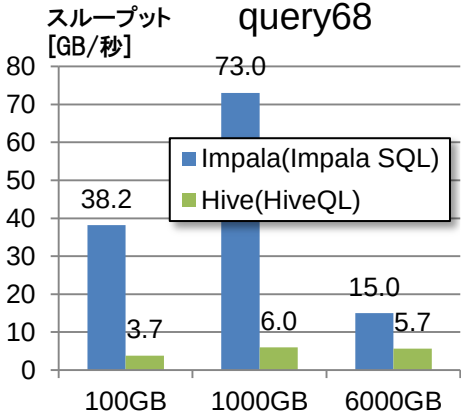
deep reporting

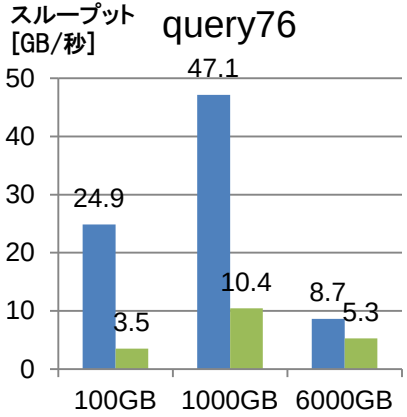
HITACHI
Inspire the Next

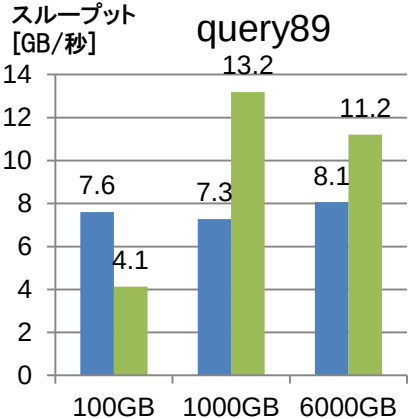


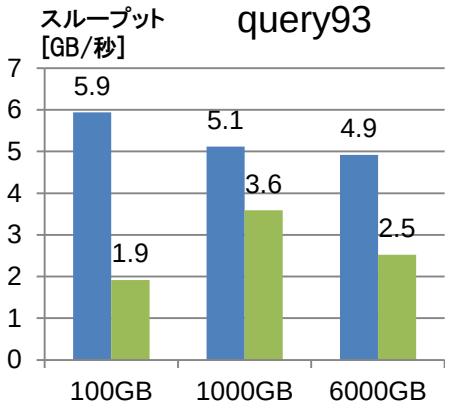


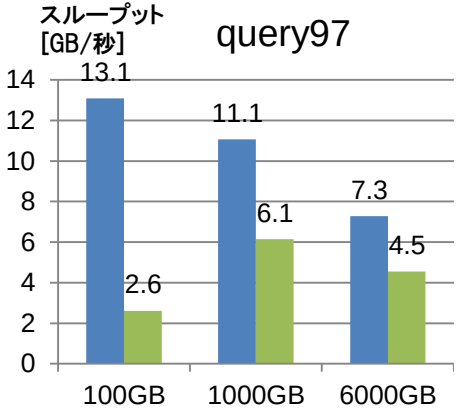










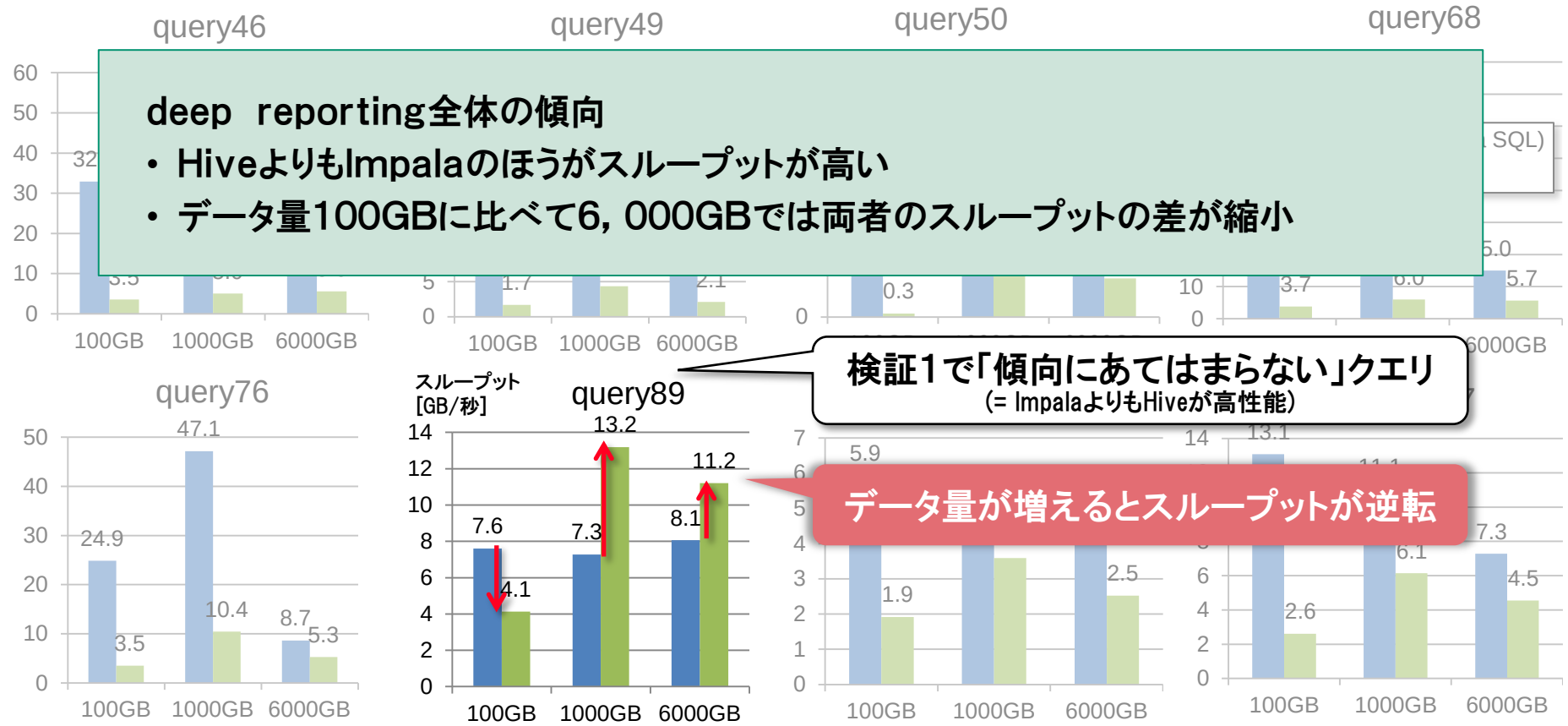


※ 値は大きいほうが良い

3-8 検証2. 処理性能の安定性

deep reporting

HITACHI
Inspire the Next



deep reporting全体の傾向

- HiveよりもImpalaのほうがスループットが高い
- データ量100GBに比べて6,000GBでは両者のスループットの差が縮小

検証1で「傾向にあてはまらない」クエリ
(= ImpalaよりもHiveが高性能)

データ量が増えるとスループットが逆転

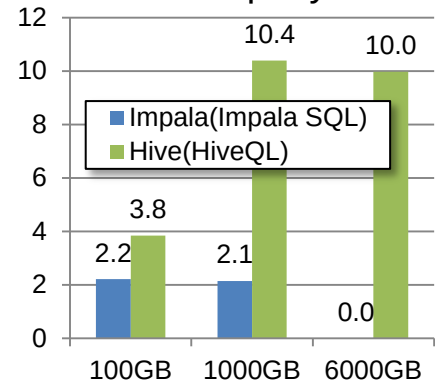
※ 値は大きいほうが良い

3-8 検証2. 処理性能の安定性

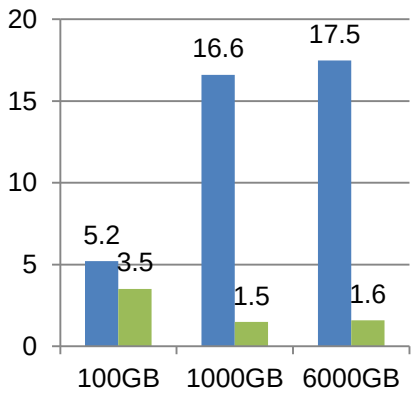
inteactive

HITACHI
Inspire the Next

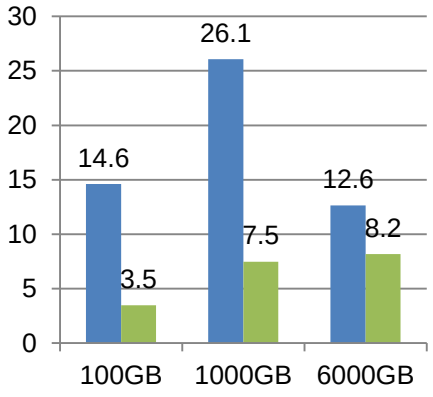
スループット [GB/秒] query3



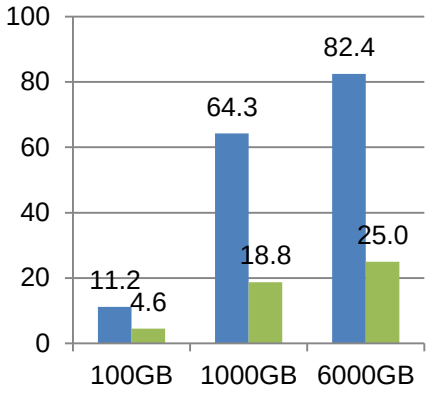
スループット [GB/秒] query15



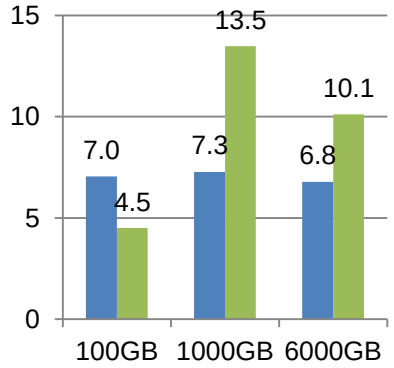
スループット [GB/秒] query19



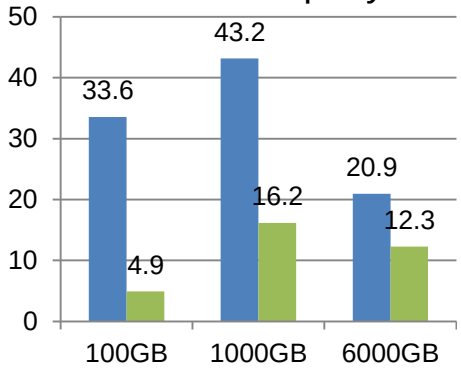
スループット [GB/秒] query26



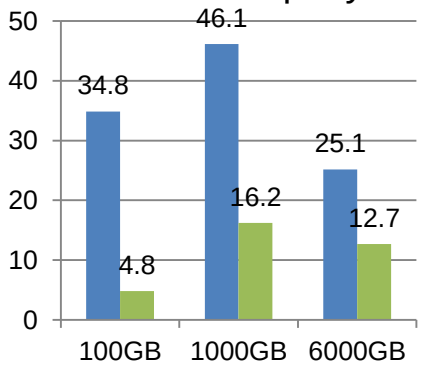
スループット [GB/秒] query43



スループット [GB/秒] query52



スループット [GB/秒] query55

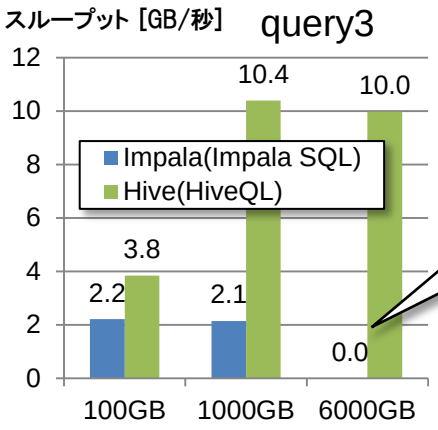


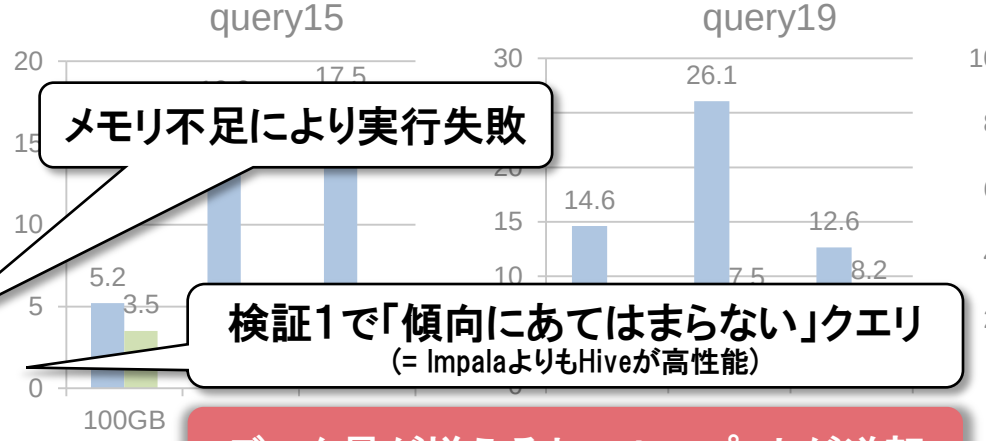
※ 値は大きいほうが良い

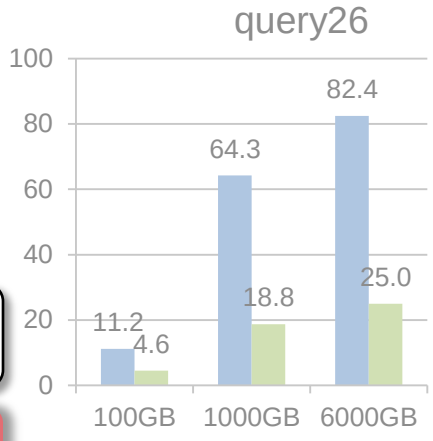
3-8 検証2. 処理性能の安定性

inteactive

HITACHI
Inspire the Next

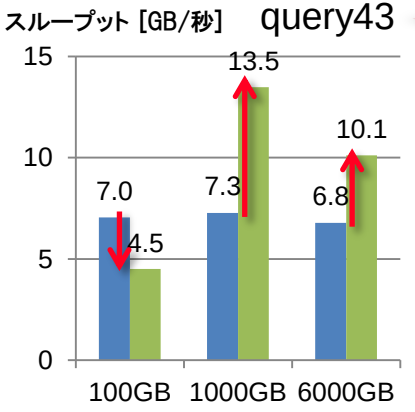






メモリ不足により実行失敗

検証1で「傾向にあてはまらない」クエリ
(= ImpalaよりもHiveが高性能)



データ量が増えるとスループットが逆転

inteactive全体の傾向

- HiveよりもImpalaのほうがスループットが高い
- データ量100GBに比べて6, 000GBでは両者のスループットの差が縮小

※ 値は大きいほうが良い

3-9 データ規模によって適したクエリエンジンが異なる

HITACHI
Inspire the Next

◆ 検証2からわかる傾向

- Impalaが高スループット
- データ量を増やすとImpalaとHiveのスループットの差は縮まる

◆ スループットの変化

- データ量1，000GBの時点でImpalaよりもHiveが高スループットだったクエリは、検証1で「傾向にあてはまらない」クエリ
- 検証1で、Impalaの特性として「処理データ量に対してメモリが小さいと性能低下」の可能性
- Query3(6，000GB)をImpalaで実行するとメモリ不足が原因で失敗した

データ量が増えるとメモリを多く消費するので、
インメモリ処理方式のImpalaは性能低下（ジョブ失敗）した

**メモリ量を上回る（TB規模の）データ量の処理にはHive
そうでない（GB規模の）処理にはImpala**

HITACHI
Inspire the Next

4. 追加検証　性能向上施策

パフォーマンスチューニング

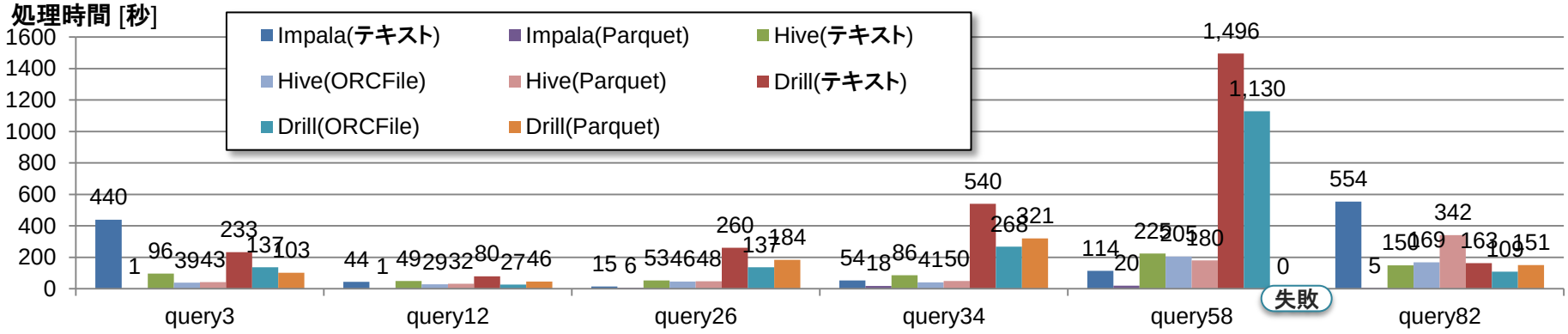
4-1 追加検証1. ファイルフォーマットによる性能差

HITACHI
Inspire the Next

検証目的	パフォーマンスチューニングの一環でより良いファイルフォーマットを検証する
検証内容	ファイルフォーマットを変えたときの処理に要した時間を比較する
検証条件	• TPC−DS 1, 000 GB • テキストファイル，ORCFile，Parquet

◆ 結果

※ 値は小さいほうが良い



Impala ＋ Parquet，　Hive ＋ ORCFile　がよりよい組合せ

4-2 追加検証2. 割当メモリ量による性能差

HITACHI
Inspire the Next

検証目的	パフォーマンスチューニングの一環でより良いメモリの割り当て方を検証する
検証内容	クエリエンジンへの割当メモリ量を変えたときの処理に要する時間を比較する
検証条件	• TPC−DS 1, 000 GB • テキストファイル • 割当メモリ量 32GB, 64GB, 170GB, 256GB

◆ 本検証のメモリ初期設定値(確認)

- Impala　mem_limit = -1 (無制限)
- Hive on Tez　yarn.nodemanager.resource.memory-mb = 294919 (約282GB)
  yarn.scheduler.maxmum-allocation-mb = 294919 (約282GB)
- Drill　DRILL_MAX_DIRECT_MEMORY = 8GB

◆ 結果

以降のスライドでクエリエンジンごとに検証する

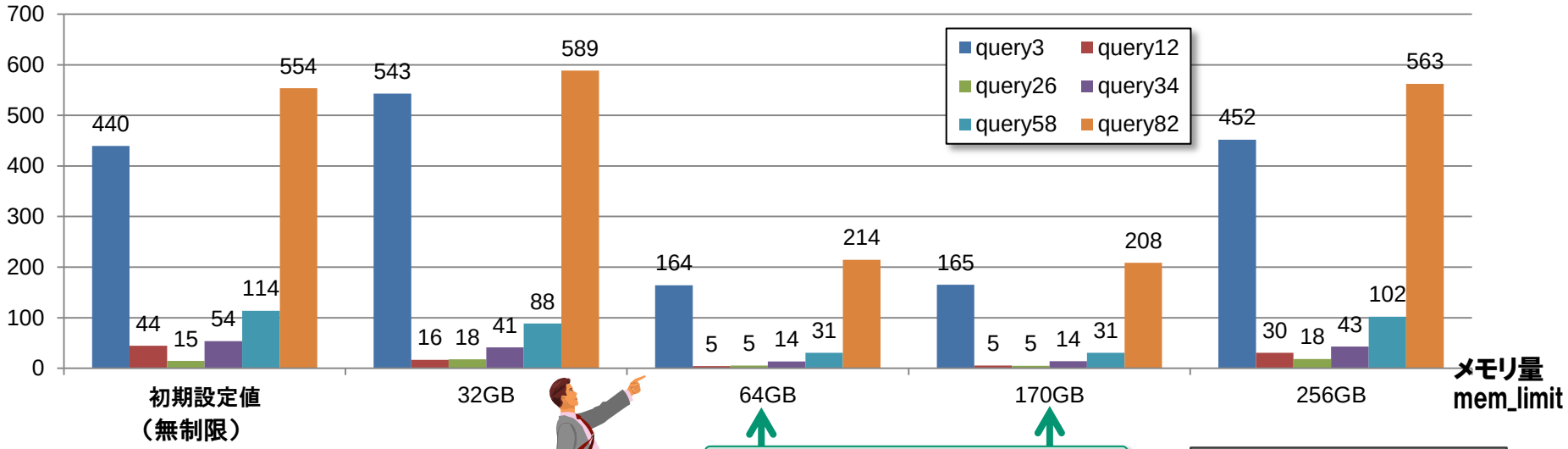
4-3 Impalaのメモリチューニングの結果

HITACHI
Inspire the Next

処理時間 [秒]

Impalaのメモリチューニング結果

※ 値は小さいほうが良い



平均約4倍の性能向上

性能低下
（初期設定と同程度）

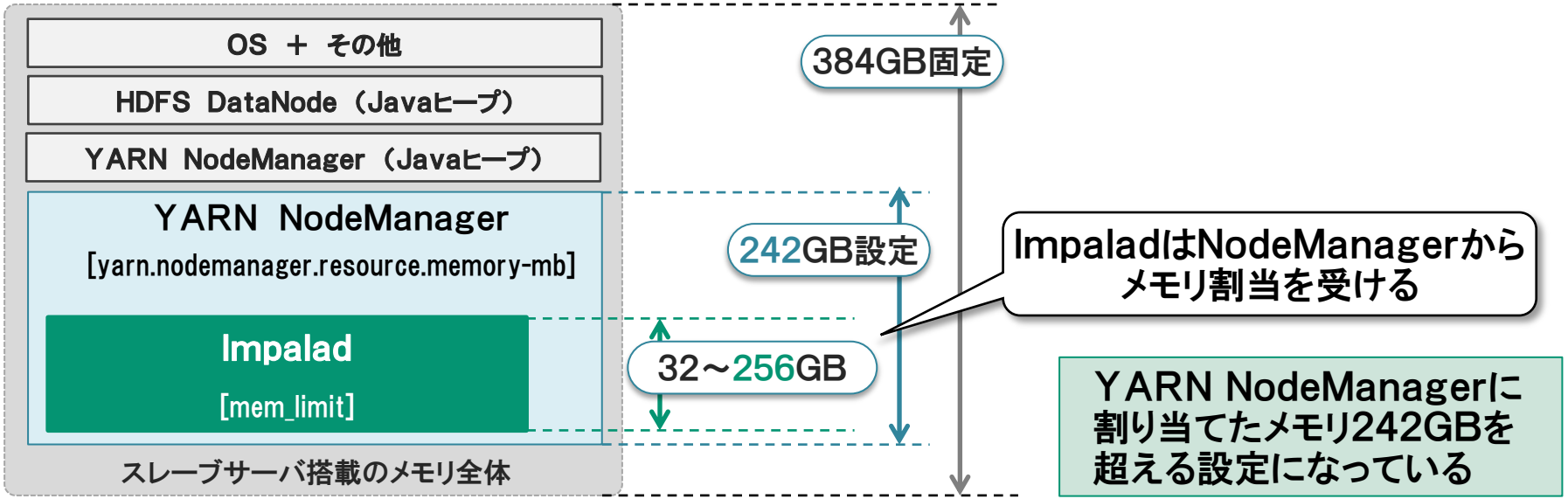
◆ 初期値との比較

- 170GBまでは、割当量を増やすほど処理性能が向上する傾向
- 256GBでは、初期設定と同程度まで処理性能が低下

4-4 ImpalaはYARN NodeManagerからメモリを割り当てる

HITACHI
Inspire the Next

◆ Impalaのメモリ管理方式と検証時の設定



Impalaのメモリ割当（mem_limit）はNodeManagerへの割当メモリ量の範囲内で大きく設定すべき

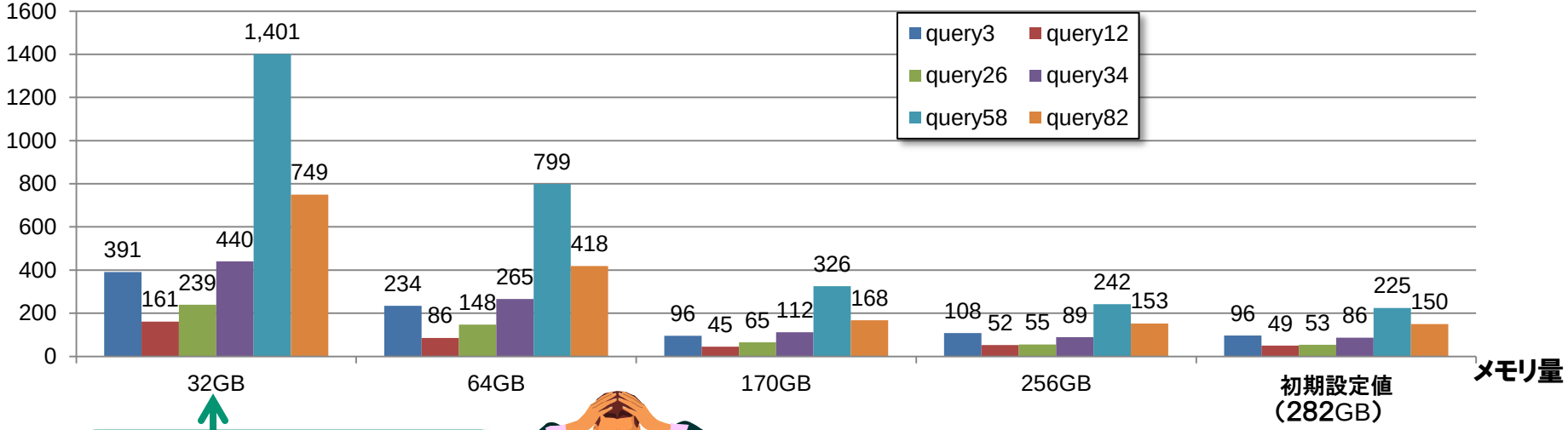
4-5 Hive on Tezのメモリチューニングの結果

HITACHI
Inspire the Next

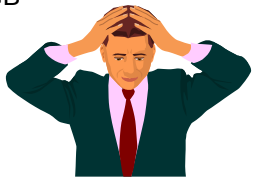
処理時間 [秒]

Hive on Tezのメモリチューニング結果

※ 値は小さいほうが良い



約5.2倍の性能低下



メモリ量設定パラメータ
yarn.scheduler.maxmum-allocation-mb
yarn.nodemanager.resource.memory-mb

◆ 初期値との比較

- メモリ割当を減らすほど性能も低下する傾向
- 初期設定値(282GB)が最も性能が高い

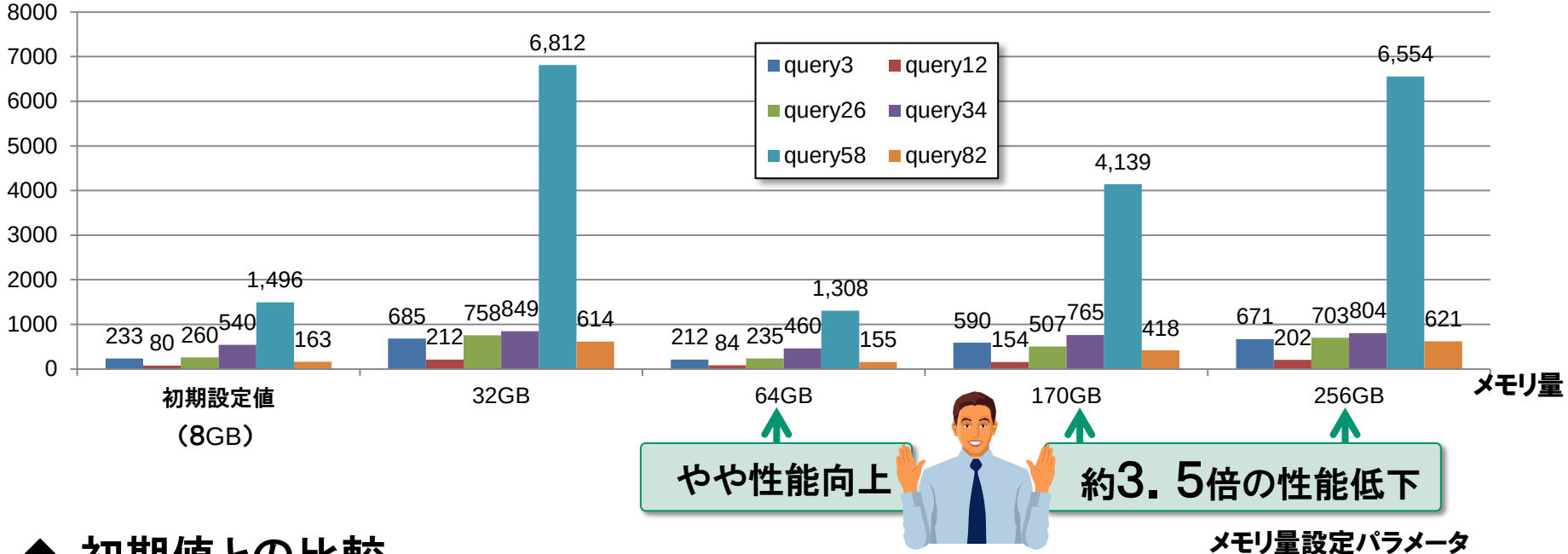
4-6 Drillのチューニングの結果

HITACHI
Inspire the Next

処理時間 [秒]

Drillのメモリチューニング結果

※ 値は小さいほうが良い



メモリ量設定パラメータ
DRILL_MAX_DIRECT_MEMORY

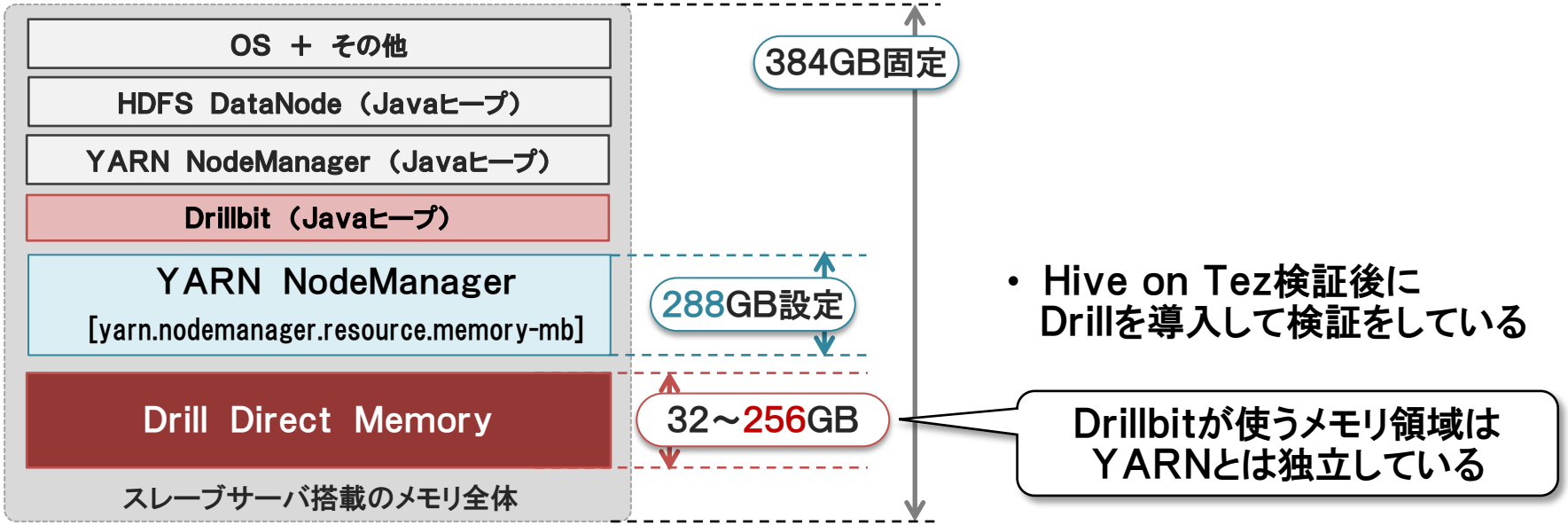
◆ 初期値との比較

- 64GBでは、処理性能が向上
- 256GBまでは、処理性能が低下する傾向

4-7 DrillとYARNのメモリ管理は独立している

HITACHI
Inspire the Next

◆ Drillのメモリ管理方式と検証時の設定



- Hive on Tez検証後に
  Drillを導入して検証をしている

DrillとYARNで確保したメモリ量がサーバ搭載のメモリ量384GBを超える設定になっている

Drillダイレクトメモリ領域に割り当てる容量を予め空けておくべき

HITACHI
Inspire the Next

5. ふりかえり

5-1 検証のふりかえり

HITACHI
Inspire the Next

◆ 検証1　クエリエンジンの性能差

HiveよりもImpalaのほうが高性能な傾向があり、得意な処理がある

Impala	• 複雑な処理（複数回のJOIN等）に強み • メモリ量が十分でないとき、著しく性能低下
Hive on Tez	• 簡素な処理（検索や数値集約等）に強み

◆ 検証2　処理性能の安定性

HiveよりもImpalaのほうが高スループットだが、データ量を増やすとその差が縮まる傾向がある

Impala	• メモリ量の範囲で収まる（GB規模の）データ処理に適する • メモリ量以上のデータ処理で、クエリ実行に失敗することがある
Hive on Tez	• メモリ量を上回る（TB規模の）データ処理に適する

5-2 SQL on Hadoopのまとめ

HITACHI
Inspire the Next

項目	Impala	Hive on Tez	Drill
推奨用途	データサイエンティスト等によるアドホックな分析	バッチ処理による大量データ処理(レポーティング等)	複数データストアを同時に使う処理
性能特性	• 比較的高性能 • メモリに処理データが載らないとき、処理が中断(失敗)することがある	• データ量が増えるほどスループットの観点で有利 • 処理内容による極端な性能劣化や処理中断(失敗)が見られない	• 本検証では確認できなかった
得意な処理	• 複数ファクトテーブルを含むスキーマを扱い、結合を複数含むような複雑な処理	• 単一ファクトテーブルのスキーマや、値の集約など比較的簡素な処理	• 本検証では確認できなかった
メモリ量の考え方	• 処理データ量以上の容量を割り当てる • YARN NodeManagerへの割当量より小さく設定	• YARN NodeManagerへの割当量はマシン搭載メモリの65～85%の範囲内で調整	• Drillダイレクトメモリ領域、YARNやOS、その他デーモンを含めたメモリ割当量の総和が、マシン搭載メモリ量以内になるよう調整

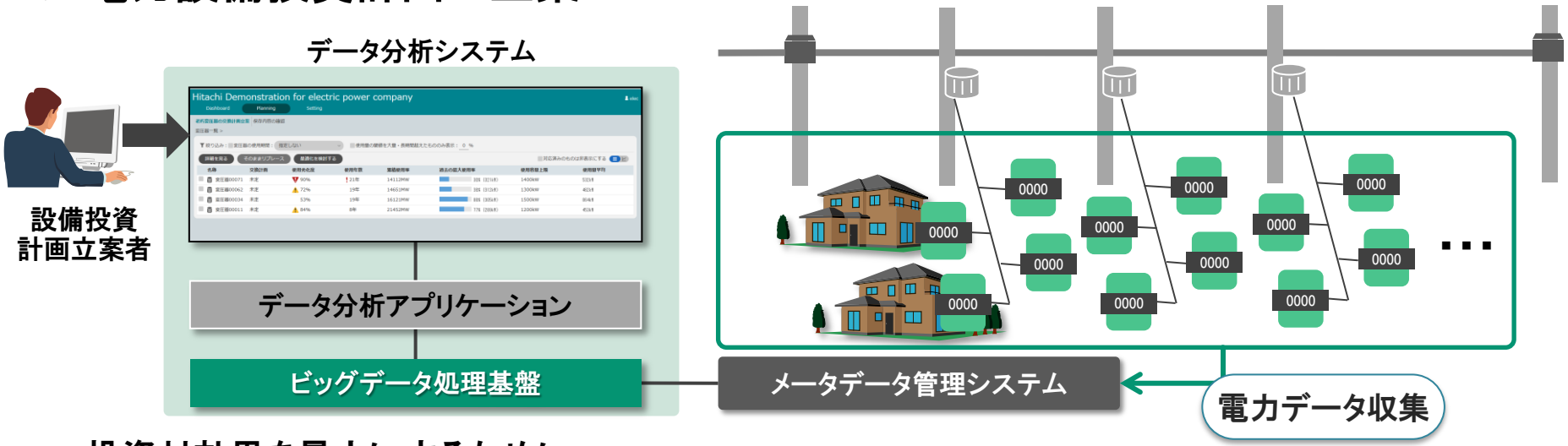
HITACHI
Inspire the Next

Appendix

Appendix データ分析の例

HITACHI
Inspire the Next

◆ 電力設備投資計画の立案



投資対効果を最大にするために

- 仮説を立てる
- 裏付けをとる(検証する)ため実績(収集した電力データ)を多角的に分析する
- 修正を繰り返して設備投資計画をつくる

分析処理は速やかに実行したい

Appendix 分析向けのデータモデル

HITACHI
Inspire the Next

◆ スタースキーマ

- ファクトテーブルとディメンションテーブルで構成されるスキーマ(データモデル)
- DWH(データウェアハウス)でよく用いられる

◆ ファクトテーブル

- スタースキーマの中心であるが、複数あってもよい
- ディメンションテーブルに対する外部キーをカラムに含む
- ファクトテーブルとディメンションテーブルは多対1のリレーション

◆ ディメンションテーブル

- ファクトの詳細な(主に年月日時分秒のような時間別に)レコード情報を格納する

[参考]
https://docs.oracle.com/cd/E16338_01/server.112/b56309/schemas.htm
https://www.ibm.com/support/knowledgecenter/ja/SSEPGG_9.5.0/com.ibm.dwe.cubeserv.doc/topics/c_cube-starschemas.html
http://support.pb.com/help/spectrum/9.3/webhelp/ja/EnterpriseDataIntegrationGuide/EnterpriseDataIntegrationGuide/source/Introduction/StarSchemaConcept.html

Appendix 検証 実行可能なSQLクエリ

HITACHI
Inspire the Next

検証目的	分析アプリケーションを実装するときのSQLは何がよいか検証する
検証内容	クエリエンジンごとに実行成功したSQLクエリの数を比較する
検証条件	• TPC−DS 1, 000 GB • テキストファイル

◆ 結果

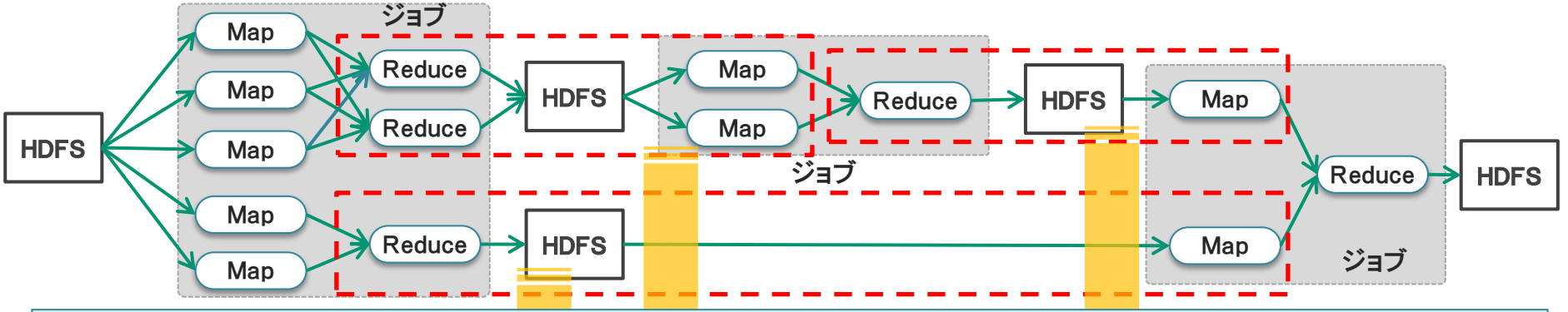
クエリエンジン	Impala		Hive on Tez		Drill		合計
	成功数	成功率[%]	成功数	成功率[%]	成功数	成功率[%]	成功率[%]
Impala	33	100	17	52	0	0	51
HiveQL	30	91	33	100	8	24	72
合計	63	96	50	79	8	12	64

HiveQLは汎用性が高いといえる ※本検証の範囲(Impala SQLとHiveQL)の結果である点に注意

Appendix TezはHDFSのI／Oを効率化した処理方式

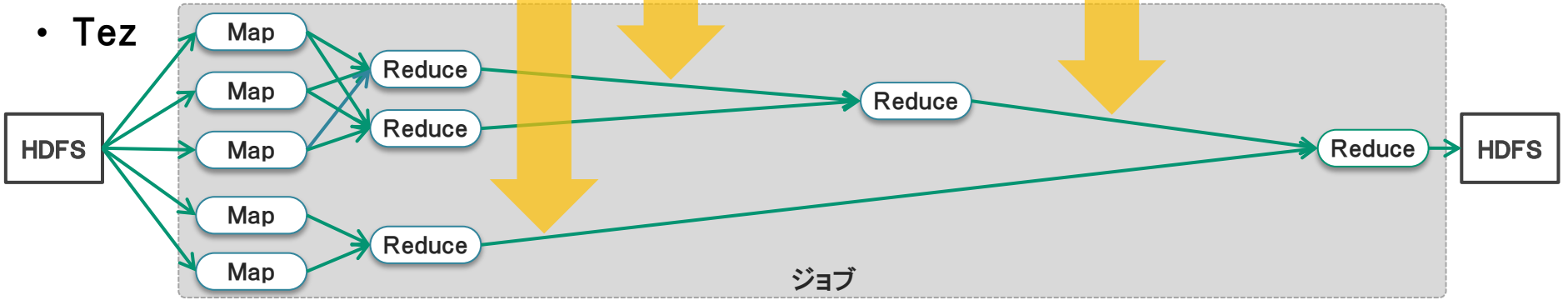
HITACHI
Inspire the Next

- MapReduce



Map処理とReduce処理を柔軟に組合せることでジョブ間のHDFSアクセスとジョブ全体を最適化

- Tez



HITACHI
Inspire the Next

END

SQL on Hadoopのホントのところ

Impala vs Hive on Tez vs Drill

2017/09/09

株式会社 日立製作所 OSSソリューションセンタ

木下 翔伍

他社商品名、商標等の引用に関する表示

HITACHI
Inspire the Next

- HITACHIは、株式会社 日立製作所の商標または登録商標です。

- Apache Hadoop, Apache Drill, Apache Hive, Apache Impala, Apache Tez, Apache ZooKeeperは、Apache Software Foundationの米国およびその他の国における登録商標または商標です。
- ClouderaおよびCDHは、Cloudera Inc. の米国およびその他の国における登録商標もしくは商標です。
- HortonworksおよびHortonworks Data Platformは、Hortonworks Inc. の米国およびその他の国における登録商標または商標です。
- OracleとJavaは、Oracle Corporation及びその子会社、関連会社の米国およびその他の国における登録商標です。

- その他記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

HITACHI

Inspire the Next