

Hadoopによる バッチ処理検証

日本ヒューレット・パカード株式会社
テクノロジーサービス事業統括
テクノロジーコンサルティング統括本部
2011年11月

目次

- 検証内容
- 検証構成
 - 検証環境構成概要図
 - 検証環境詳細
 - Hadoopの主な設定情報
- 検証結果
- まとめ
- コンサルティングサービス
 - Hadoop導入コンサルティングサービス
 - オープンソース・エキスパートサービス
- Appendix
 - チューニング項目一覧
 - MapReduce処理時のCPU使用率推移
 - MapReduce処理時のWrite/Read量推移
 - MapReduce処理時のDisk %Util推移
 - MapReduceにおけるシャッフルとソート



検証内容



検証内容

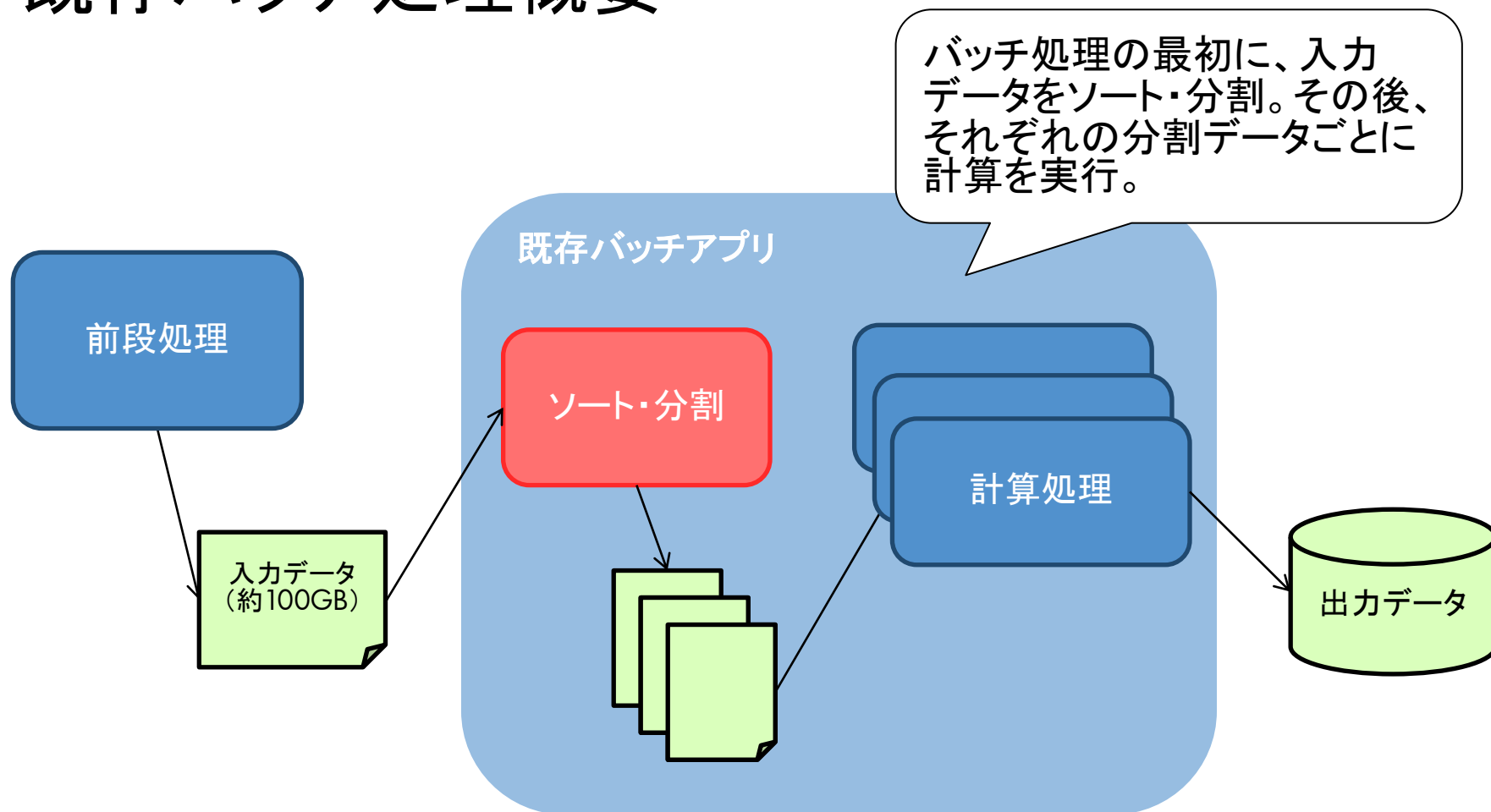
- 検証の背景

- データ量増加に伴い、既存バッチシステムが性能の限界に直面している。
- バッチ処理におけるデータのソートおよび分割処理が、既存のアプリケーションのボトルネックになっているが、既存のアーキテクチャを踏襲して、それを改修するためには多くのコストがかかるため、それを最低限にしつつ性能改善をしたい。

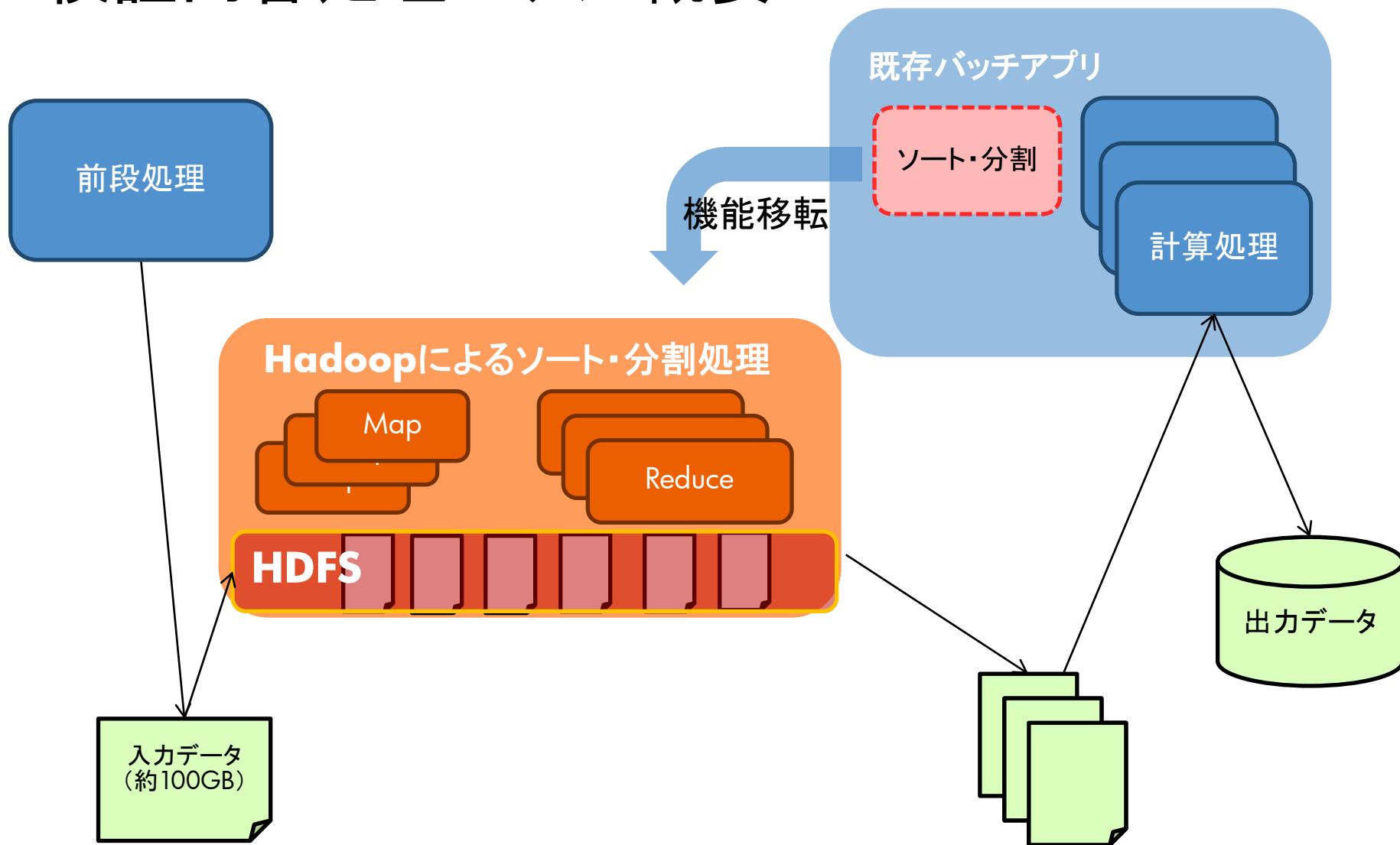
- 検証の目的と範囲

- 既存バッチ処理のうちのデータのソートおよび分割の部分にHadoopを適用した場合の効果を実証する。
- Hadoopを使用し、疑似データのソートおよび分割を行い、それにかかる実行時間の調査を行う。また、サーバー台数を増加させることにより、どれくらい性能増加するのかを確認し、Hadoopとしてボトルネックとなりがちな点を分析し、Hadoopの仕組みを詳しく理解する。
- Hadoop導入・運用に対する効果・影響に関する考察を行う。

既存バッチ処理概要



検証内容処理モデル概要



Hadoop実行処理

起動コマンド:

hadoop ...

- D map.output.key.field.separator=. ¥
- D num.key.fields.for.partition=1 ¥
- D mapred.reduce.tasks=<分割ファイル数N> ¥
- partitioner org.apache.hadoop.mapred.lib.KeyFieldBasedPartitioner ¥
- reducer /bin/cat
- ...

Map処理は左下吹き出しのスクリプト。
Reduce処理はOSコマンドの/bin/catのみ。
ソート・分割処理は、Hadoop Jobのオプション指定のみで実現

Key1、Key2、Key3で
ソート・分割処理

HDFS上のディレクトリ

結果1

結果2

結果N

Hadoop
(HDFS)

Job1

Map
(複数)

Reduce1

Reduce2

⋮

ReduceN

Key1でグルーピング

Mapスクリプト

```
for line in sys.stdin:
    line = line.rstrip()
    Key1 = line[i:j]
    Key2 = line[k:l]
    Key3 = line[m:n]
    print '%s.%s.%s\t%s' % (Key1, Key2, Key3 line)
```

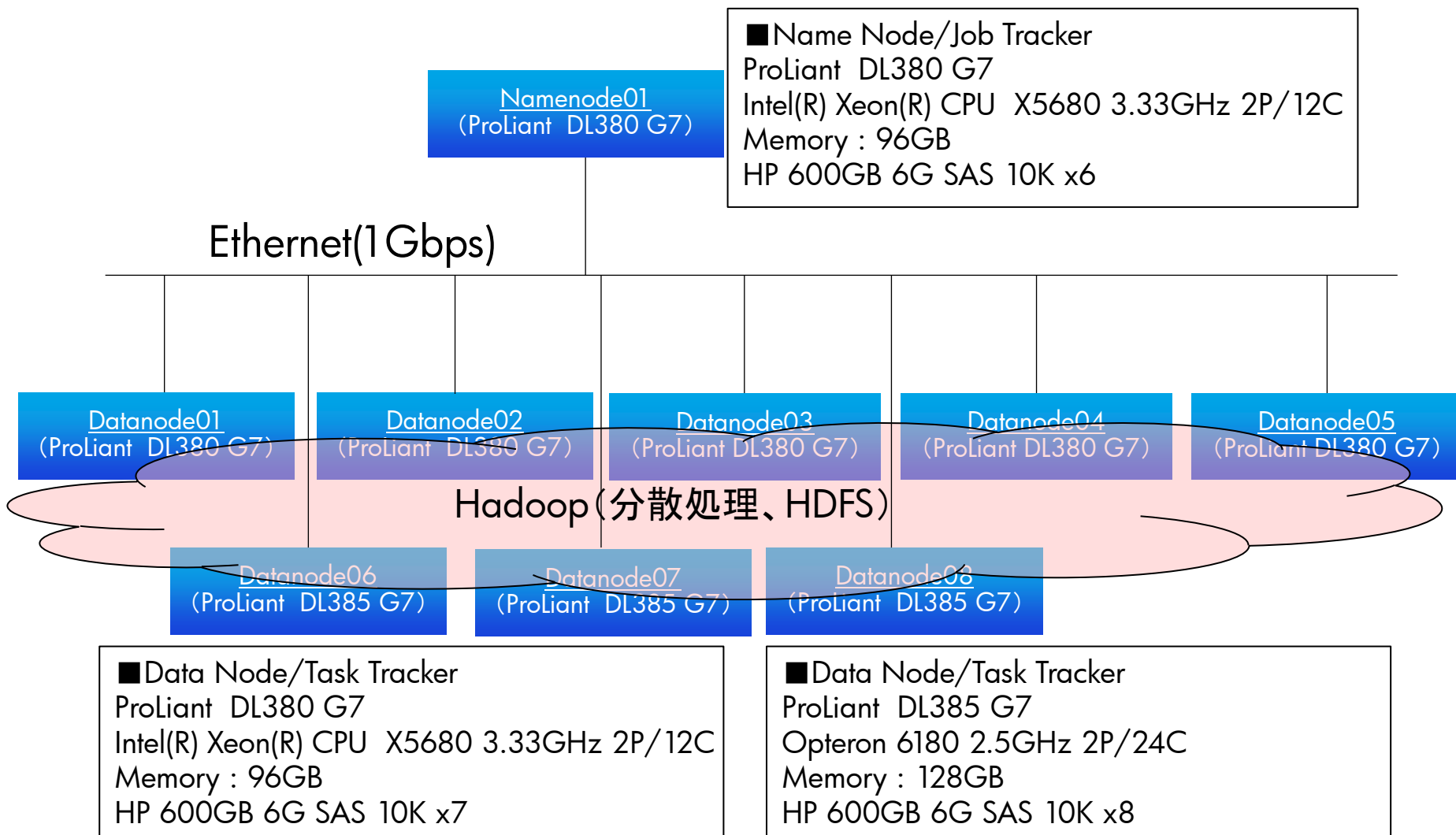
※i-nは数字

Key 1-1	xxxx
Key 1-1	xxxx
Key 1-4	xxxx
Key 1-4	xxxx
Key 1-4	xxxx
...	

検証構成



検証環境構成概要図



検証環境詳細(1/2)

ホスト名	Namenode01	Datanode01 - 05
サーバー	HP ProLiant DL380 G7	HP ProLiant DL380 G7
Hadoop	CDH3u1 (Cloudera's Distribution for Hadoop)	CDH3u1 (Cloudera's Distribution for Hadoop)
OS	RHEL 5.6 x86_64	RHEL 5.6 x86_64
CPUタイプ	Intel(R) Xeon(R) CPU X5680 3.33GHz	Intel(R) Xeon(R) CPU X5680 3.33GHz
CPU数	2P(12core) HT=on	2P(12core) HT=on
メモリ	96GB	96GB
オンボードNIC	HP NC382i PCIe Dp Gigabit Svr Adptr	HP NC382i PCIe Dp Gigabit Svr Adptr
内蔵ディスク	HP 600GB 6G SAS 10K HDD x6	HP 600GB 6G SAS 10K HDD x7
ディスク コントローラー	Smart アレイP812 / 1G FBWC コント ローラー	Smart アレイP812/ 1G FBWC コントローラー
ディスク構成	RAID10(6本)	OS領域がRAID0(1本)、残りは1本ずつRAID0(6 本) ※Hadoop側でディスクに対してラウンドロビンアク セス構成



検証環境詳細(2/2)

ホスト名	Datanode06 - 08
サーバー	HP ProLiant DL385 G7
Hadoop	CDH3u1 (Cloudera's Distribution for Hadoop)
OS	RHEL 5.6 x86_64
CPUタイプ	Opteron 6180 2.5GHz
CPU数	2P(24core)
メモリ	128GB
オンボードNIC	HP NC382i PCIe Dp Gigabit Svr Adptr
内蔵ディスク	HP 600GB 6G SAS 10K HDD x8
ディスク コントローラー	Smart アレイP812/ 1G FBWC コントローラー
ディスク構成	OS領域がRAID1(2本)、残りは1本ずつRAID0(6本) ※Hadoop側でディスクに対してラウンドロビンアクセス 構成



Hadoopの主な設定情報

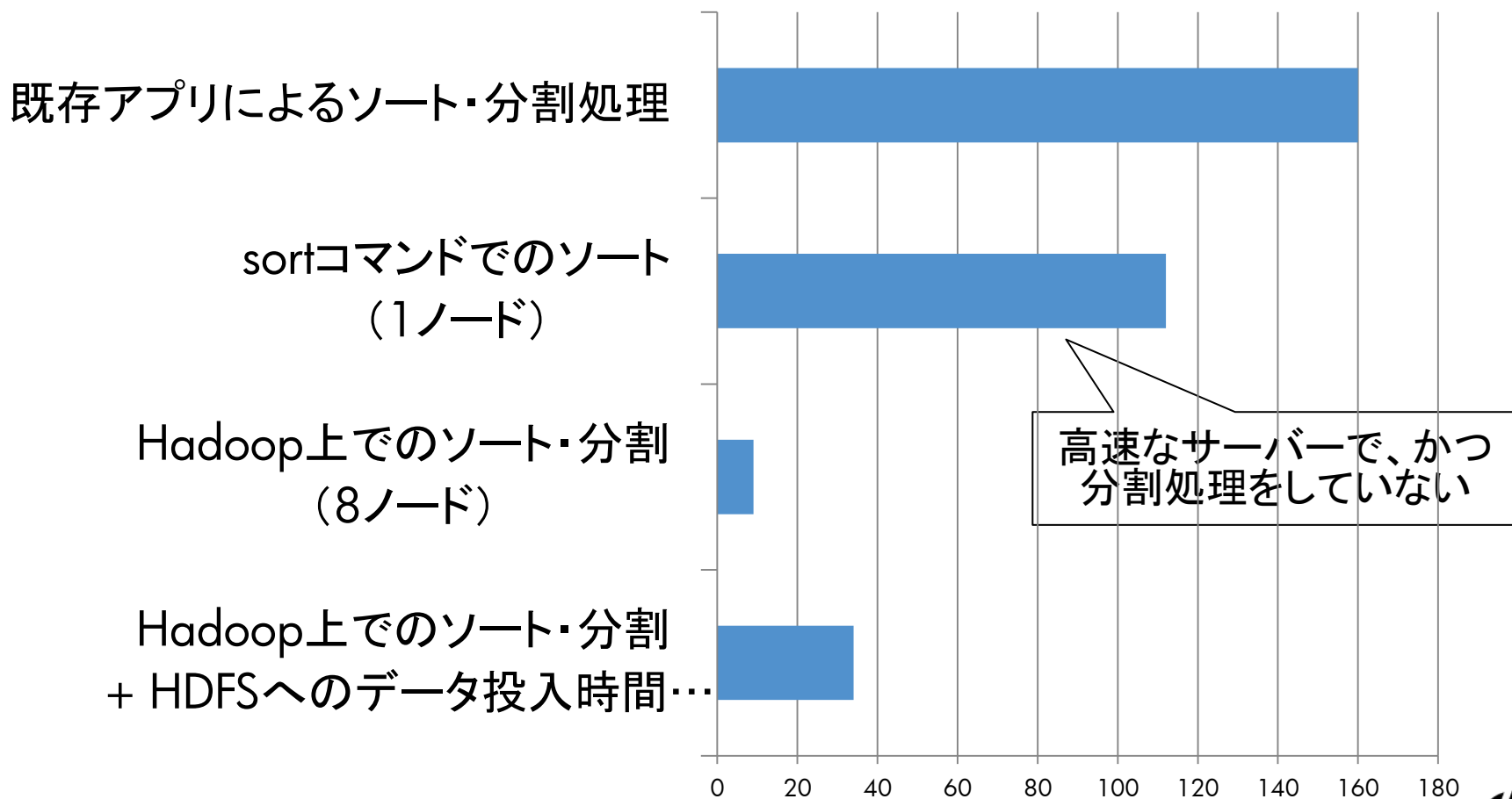
項目	既定値	設定値	備考
mapred.child.java.opts	-Xmx200m	-Xmx1024m	TaskTrackerの子プロセスに渡されるJavaオプション
dfs.block.size	67108864	134217728	新規ファイルのブロックサイズ(byte)
io.sort.mb	100	512	Mapper上でファイルソートを行う際に、マージストリームに割当てられるバッファメモリ量(MB)。タスクのJVMアロケーションによる
dfs.replication	3	2	可用性をどこまで担保するかによる。速度優先であれば2を推奨
mapred.reduce.tasks	1	84	ジョブあたりのReduceタスク数
mapred.tasktracker.map.tasks.maximum	2	16	TaskTrackerによって同時稼働可能なMapタスクの最大数。HTを使用している場合は、搭載コア数の1.2～1.5倍が目安。
mapred.tasktracker.reduce.tasks.maximum	2	16	TaskTrackerによって同時稼働可能なMapタスクの最大数。HTを使用している場合は、搭載コア数の1.2～1.5倍が目安。

検証結果

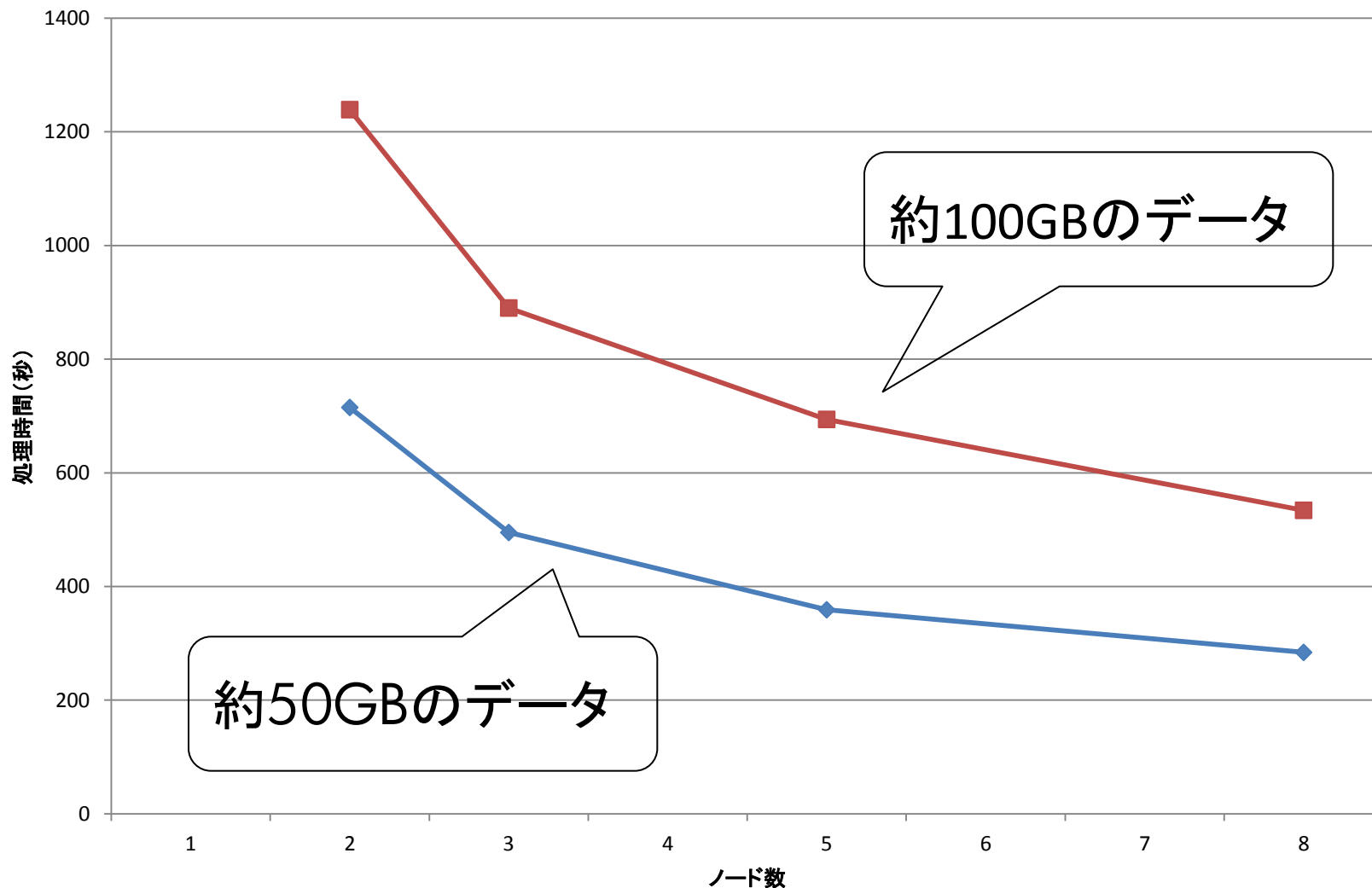


既存バッチシステムとHadoop処理の処理時間比較

約100GBデータの処理時間比較

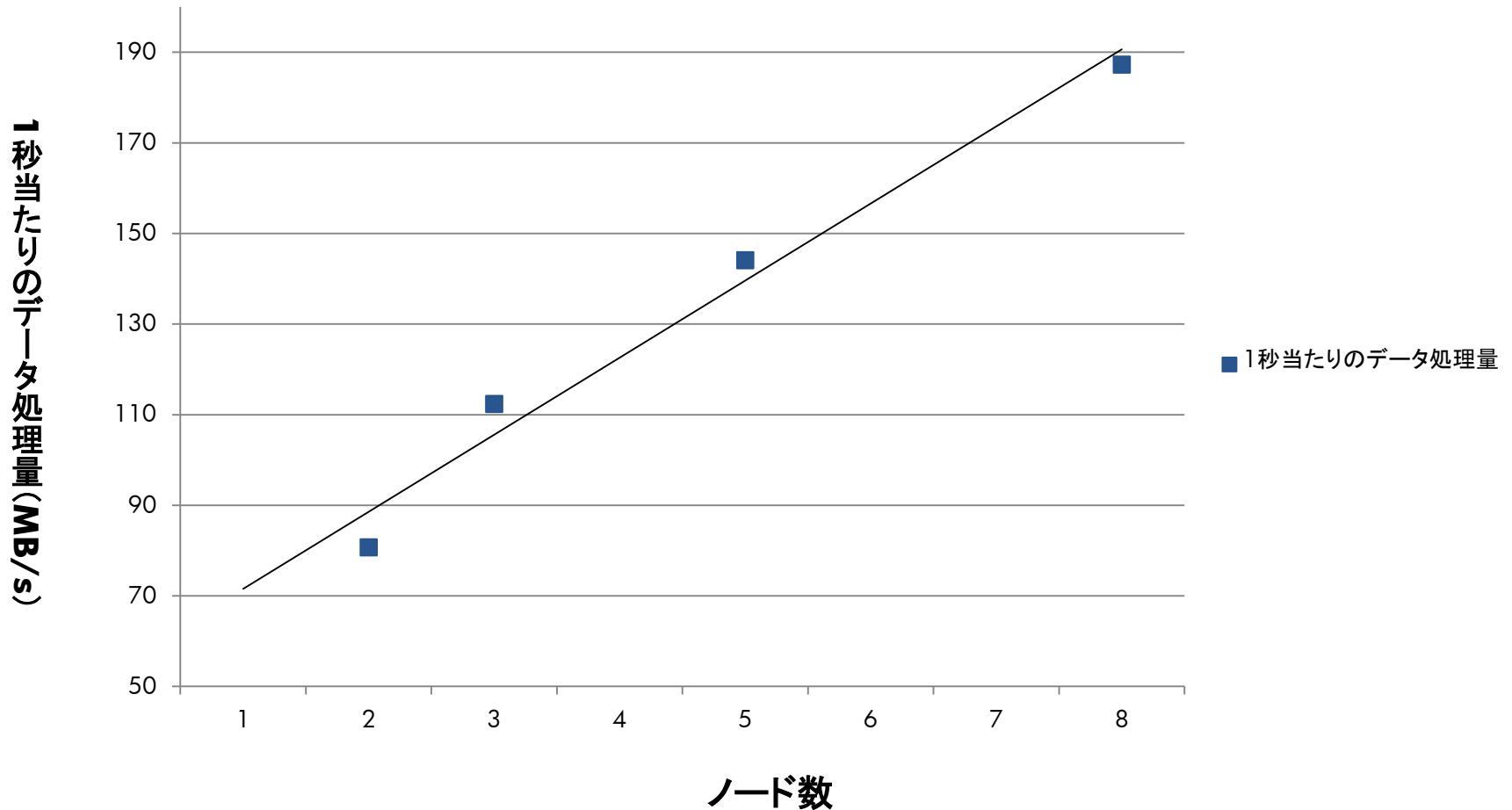


Hadoop処理実行時間



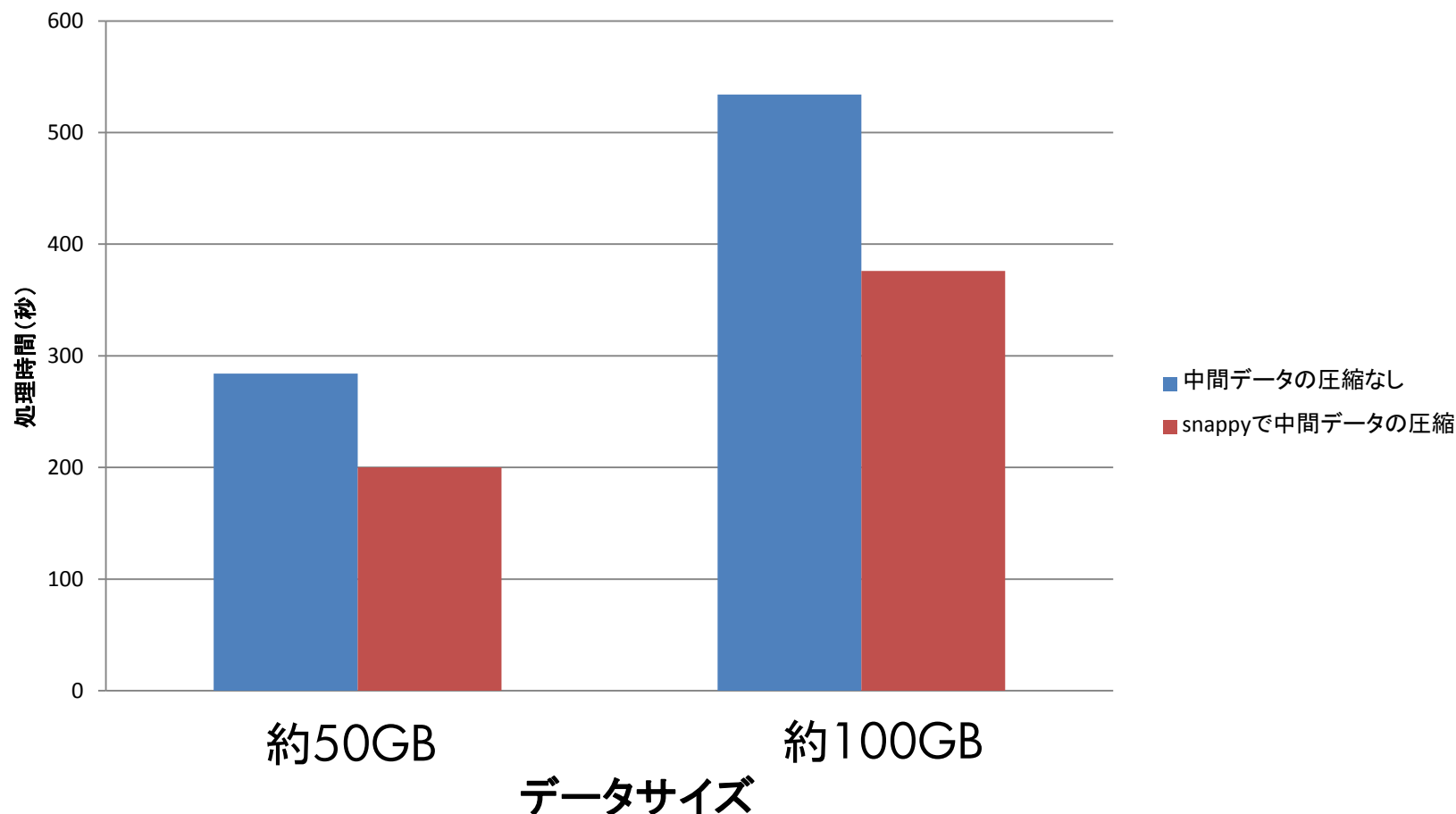
Hadoop処理性能

1秒当たりのデータ処理量(約100GBのデータ)

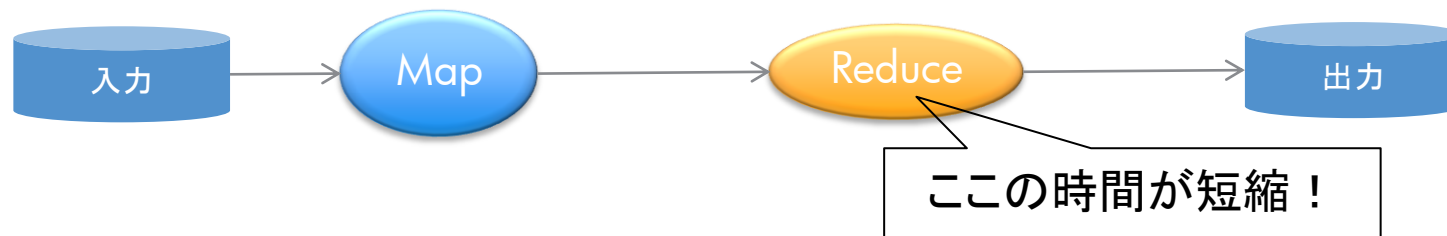


Hadoop処理実行時間(中間データの圧縮)

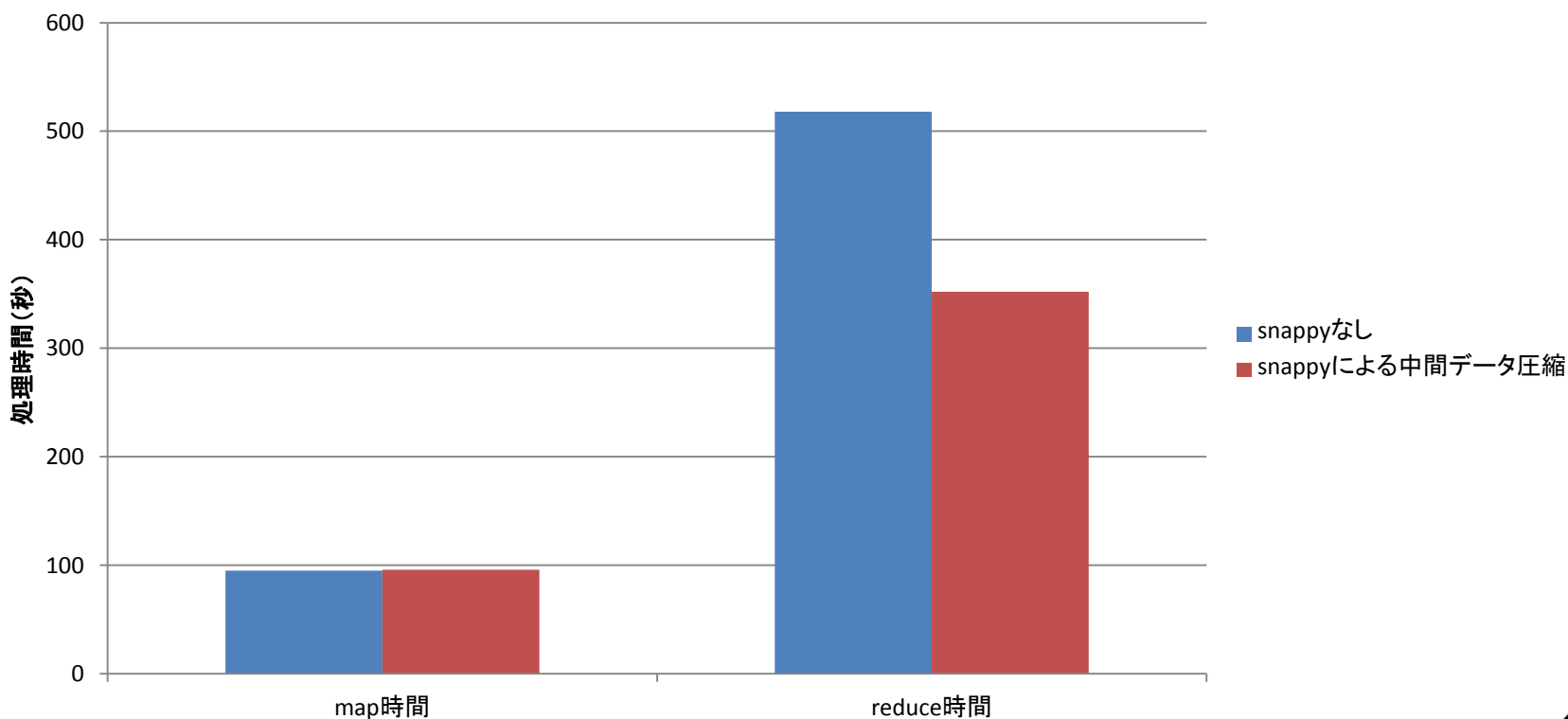
Job全体での効果



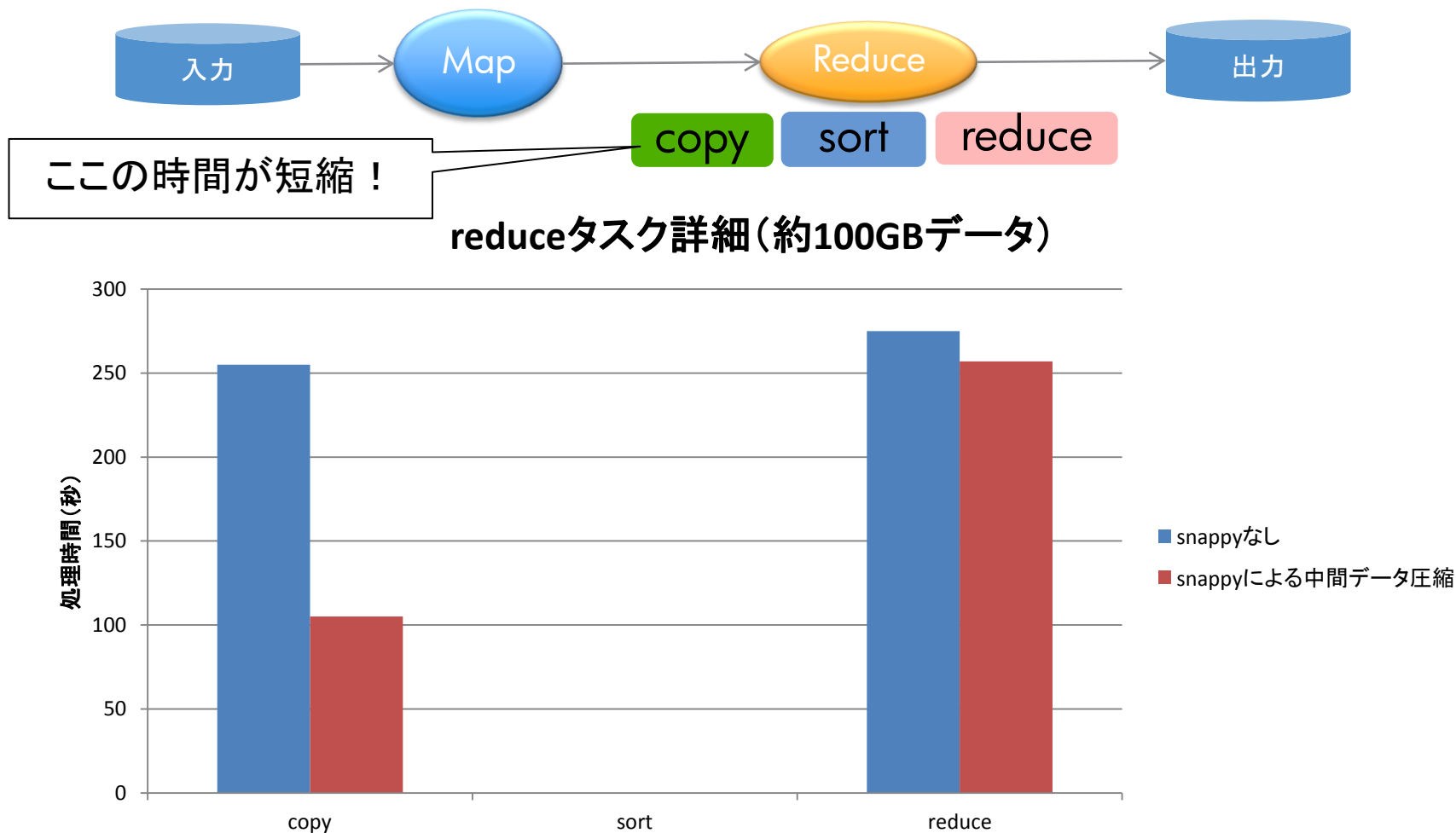
Hadoop処理実行時間(中間データの圧縮)



map/reduce別の時間変化(約100GBデータ)



Hadoop処理実行時間(中間データの圧縮)



※copyフェーズ、sortフェーズ、reduceフェーズの時間は、
10秒置きの画面キャプチャから判断しているため、あくまでも参考値

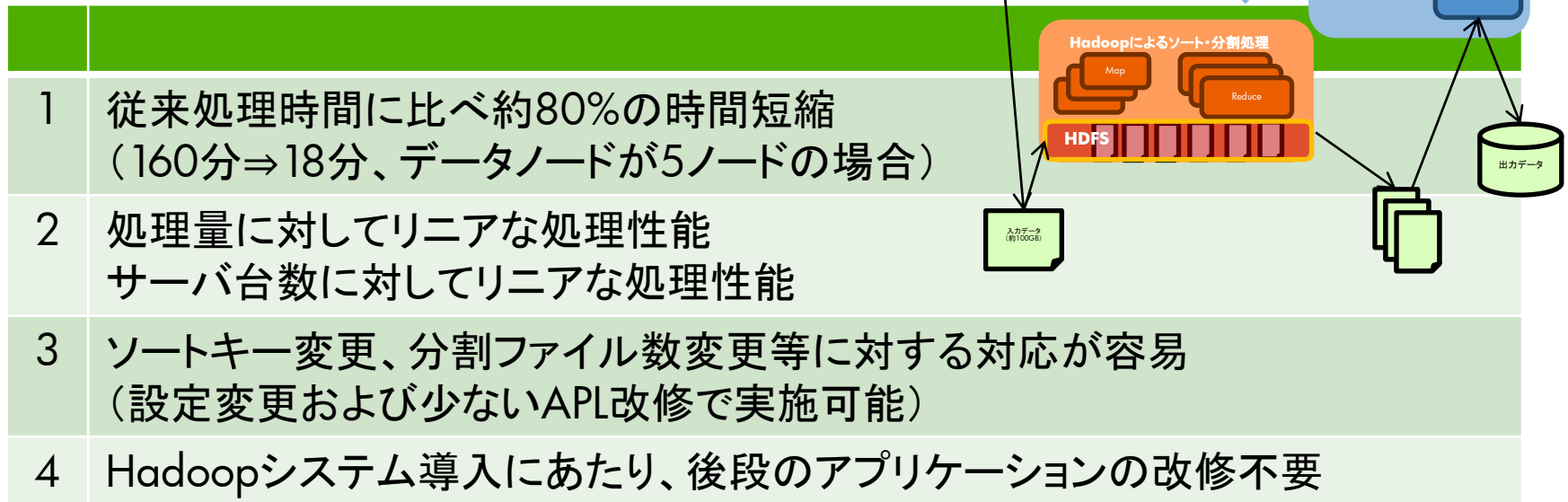
まとめ



まとめ

1	ノード数増加によるHadoop性能の増加を確認
2	map処理はノード数増加により大きく性能が増加 → map処理はCPUの使用率が100%近くで推移しており、Diskの%UTILも同様に100%近くまで上昇している。そのため、多くのプロセスや書き込み待ちが発生しており、ノード数増加により、プロセスや書き込み待ちが解消され性能向上が大きくなっていると分析できる
3	reduce処理もノード数増加により性能が向上 → iostatの結果から、HDFSへの書き込みがボトルネックになっており、ノード数を増やすことで負荷分散が可能となったためと分析できる
4	中間データのsnappy圧縮により、中間データの転送量が少なくなり、reduceタスクのコピーフェーズが大幅に時間短縮されることが確認
5	JavaのMapReduceでも実装したが、パフォーマンスの変化は特になし → 一概にHadoop Streamingが遅いとは言えないという結果
6	50GB～100GBの範囲のデータでも、Hadoopを使うことで十分な効果を得ることができることを実証
7	Hadoopの仕組みを最大限に利用することで、非常に短いコードで既存バッチの高速化を図ることが可能

導入・運用上のメリット



- 一般的にソートや分割を行う処理はHadoopの導入効果が期待できるエリアであり、今回の試験結果からも効果がある事が判明
- Hadoopは、本バッチ処理以外でも類似処理への展開・有効活用が可能と思われるので、適応可能箇所の検討は有効

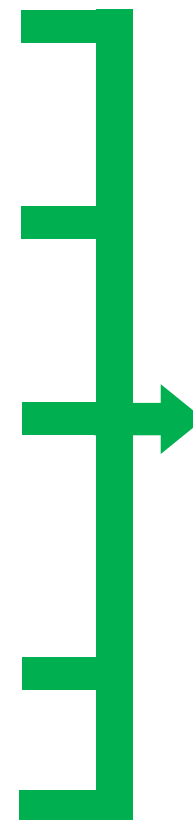
Hadoop導入コンサルティング サービス



HP Hadoop導入サービス

お客様環境にHadoopクラスタを導入するサービスです

フェーズ	サービス概要
検討	• 適用領域、既存システムの連携についてのヒアリングを実施 • 必要に応じて、導入前検証(PoC)をHPソリューションセンターで実施
設計	• ヒアリング内容に基づき、サーバ構成、ネットワーク構成を設計 • Hadoopクラスタ全体アーキテクチャの設計 • 運用設計
構築	• 設計内容に基づいたHadoopクラスタの構築 • Hadoopへのデータ投入、データ解析用サンプルスクリプトの作成 • 既存システムやBIツールへのデータ連携 • HP CMUおよびZabbix/Gangliaを使ったクラスタ運用基盤の構築 • MapReduceアプリケーション開発支援
テスト	• 性能測定・チューニング • MapReduceアプリケーションのテスト支援 • 運用テスト
運用	• Hadoopクラスタ運用支援 • 保守サービスの提供 • Hadoop活用時に使用する各種オープンソースソフトウェアを1つの窓口で技術支援



検討から運用支援までサービスを提供

HP Hadoop 導入前検証(PoC)サービス

本格導入前の検証をHPソリューションセンターで実施いたします

HPソリューションセンターを使った導入前検証

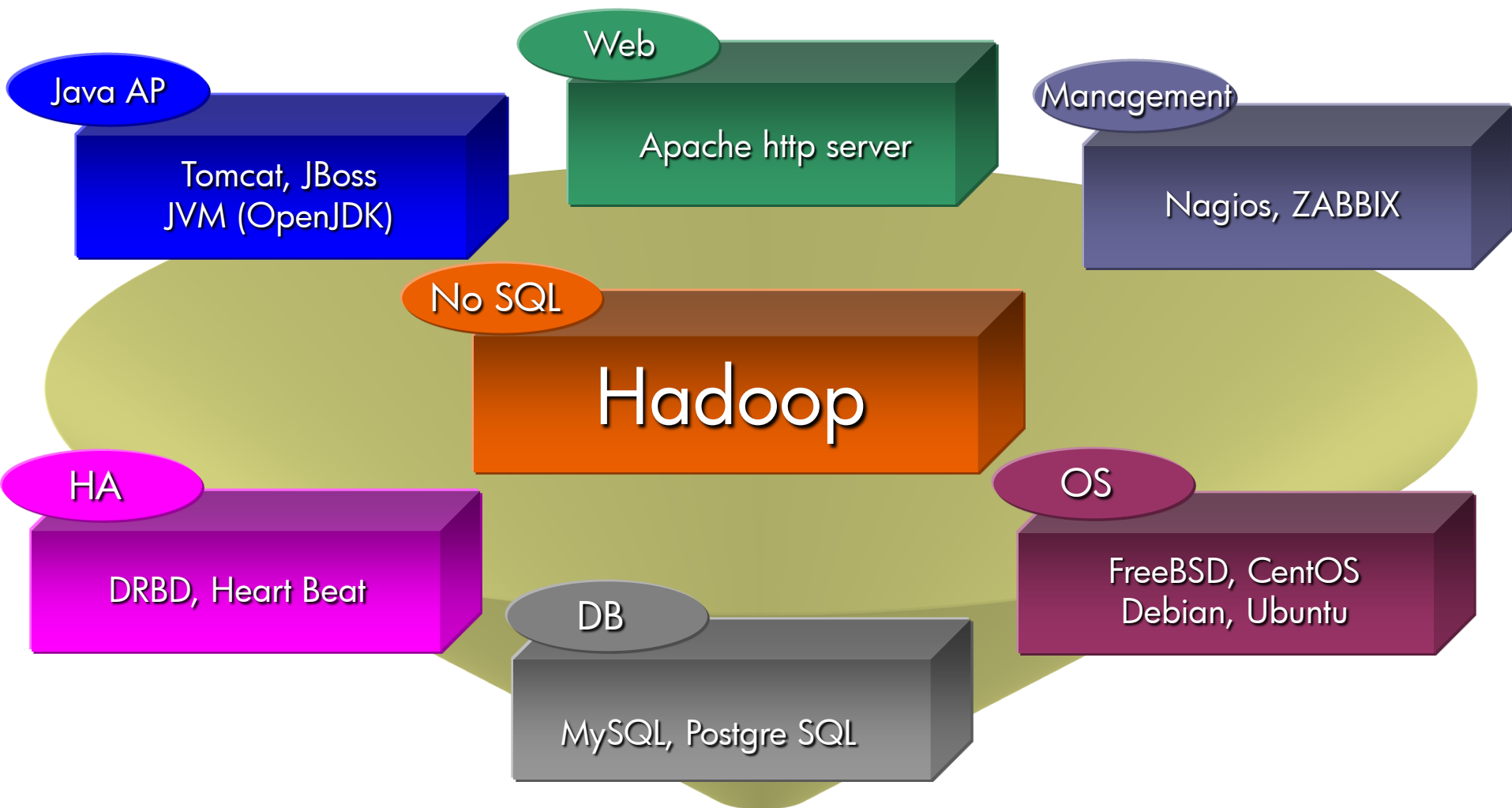
- HPソリューションセンターの機器を使用し、お客様データを使用した導入前検証です。
- VPN接続でお客様先からも作業可能。
- 設定情報、使用スクリプト等のベストプラクティスをお持ち帰りいただけます。

お客様環境での実施も可能

- お客様環境で既存サーバ、他社製サーバを使った構築にも対応いたします。



Hadoopだけでなく、関連するオープンソースソフトウェアを含めた
トータルサポート



ワンストップで、トータルサポートを提供

オープンソース・エキスパート サービス



オープンソース・エキスパート・サービス

オープンソースを用いたシステム構築において以下のようなことでお困りではないでしょうか？

開発

OSS製品の仕様や、バージョン間の違や他製品との違いを調査したいが、調査方法などが分からず困っている。

開発時にアプリケーションのコーディング方法について、作成中のアプリケーションを含めて相談したいが、適切な相談先が見つからない。

保守

古いOSS製品や、最新のOSS製品を使いたいが、保守サービスがないので困っている。

対 障害 応 害

障害が発生し、いろいろな窓口に問い合わせしてみたが、障害原因を特定することができずに困っている。

**オープンソース・エキスパート・サービスが
これらのお悩みを解決いたします。**

オープンソース・エキスパート・サービス サービス仕様

概要

本サービスは、導入・開発・試験・運用・障害発生などのあらゆるフェーズにおいて、様々なオープンソースソフトウェア(OSS)製品に関するご質問を1つの窓口で統一してお受けして回答するサービスです。
対象となるOSS製品はLinuxなどのオペレーティングシステムからJBossやPostgreSQLなどのミドルウェア製品まで広範囲に渡り、複数のOSS製品が関連したご質問にも対応いたします。

対象顧客

複数のOSS製品をお使いで、それらOSS製品の包括的な問い合わせ窓口が必要なお客様

メリット

- 複数のOSS製品の問い合わせ先を一本化することができます。
- OSS製品の調査や障害対応などに掛かる時間やコストを削減できます。

契約体系・価格

本サービスでは、お客様のご要望に合う最適なサービスを提供させて頂くために個別に契約内容を相談させて頂きます。例えば以下のような内容にて契約させて頂きます。

- 対応製品: 「サービス対象OSS製品」の中から5製品を選択して頂きます。
- 契約単位: システム環境ごとの契約となります。
- 契約期間と稼働時間は例えば以下のような組み合わせで契約可能です。
契約稼働時間を超えた場合は、追加見積にて対応させて頂きます。
 - 12ヶ月/360時間、6ヶ月/180時間、3ヶ月/90時間
(各月の最大稼働時間は30時間までとさせて頂きます。)

前提

- サービス対象のOSS製品であれば、それ自体が動作しているOSがサービス対象のOSでなくても対応可能です。
- ソースコードが公開されているOSS製品であれば古いバージョンから最新バージョンの製品まで対応させて頂きます。
- 本サービスは委任契約の技術支援サービスとなりますので、お客様のところで発生した障害などの解決をお約束するものではありません。

サービス内容

- お問い合わせ方法
 - メール (電話でのお問い合わせについてはご相談ください。)
 - お問い合わせ頂ける担当者は3人までとさせて頂きます。
- お問い合わせ時間
 - 平日8.45 - 17.30 (祝祭日、年末年始を除く)
- 一次応答時間の目安
 - AM受付は当営業日中、PM受付は翌営業日AM中
- お問い合わせ頂ける質問数には制限がございません
- お問い合わせ単位もしくは月次で弊社担当者の稼働時間を報告いたします。
- お問い合わせ頂ける内容
 - インストール方法、パラメータ調査、設定方法、仕様確認
 - 複数OSS製品の連携・接続方法
 - 障害調査
 - 障害原因のログ解析、回避方法の提案
 - 過去事例の調査、コミュニティからの情報収集
 - ソースコードの調査
 - 既存パッチに関する情報提供
- オプションでの対応
 - お客様先でのオンサイト作業は本サービスでは対象外となりますが、別途対応可能ですのでご相談ください。
 - 場合によっては、保守契約のある別製品に関する問い合わせ窓口の代行も可能ですのでご相談下さい。

サービス問合せ窓口

本サービスのお見積、ご質問に関しては、jpn_tc_solution_desk@hp.comへお問い合わせください。



サービス対象OSS製品

OS(オペレーティングシステム)

- CentOS, Debian GNU/Linux, Ubuntu Server, openSUSE, FreeBSD
- Red Hat Enterprise Linux
- その他Linuxはご相談下さい。
- ネットワーク、webサーバー
 - BIND
 - Samba
 - vsftp
 - Apache HTTP Server
 - mod_jk、mod_proxy、mod_alias、mod_rewrite、mod_jk2
 - Squid
 - Postfix, Sendmail
 - procmail, Dovecot
 - OpenLDAP
 - OpenSSL
 - OpenSSH
 - memcached
- ミドルウェア
 - Hadoop **new**
 - memcached

- データベース
 - MySQL
 - PostgreSQL
- アプリケーションサーバ
 - Tomcat
 - JBoss Application Server
- Java
 - OpenJDK
 - Struts
 - Spring
 - JBoss ESB
 - JBoss jBPM
 - JBoss Hibernate
 - JBoss Drools
 - JBoss Seam
- 開発
 - Eclipse、CVS、Subversion
 - bash, cshなどのシェル
 - Perl
- 管理・監視
 - ZABBIX

※ リストに載っていない製品はご相談下さい。



Appendix



チューニング項目一覧(HDFSパラメータ)

項目	既定値	推奨値	設定値	備考
dfs.block.size	67108864	134217728 or 268435456	134217728	新規ファイルのブロックサイズ(byte)
dfs.datanode.handler.count	3	3-5	5	DataNode上でブロックリクエストを処理するスレッド数
dfs.replication	3	2	2	可用性をどこまで担保するかによる。速度優先であれば2を推奨。

チューニング項目一覧(コアパラメータ)

項目	既定値	推奨値	設定値	備考
io.file.buffer.size	4096	65536	131072	読込・書込時にバッファされるデータ量(byte)



チューニング項目一覧(MapReduceパラメータ)

項目	既定値	推奨値	設定値	備考
mapred.child.java.opts	-Xmx200m	クラスタ構成に依存。	-Xmx1024m	TaskTrackerの子プロセスに渡されるJavaオプション
mapred.job.tracker.handler.count	10	全TaskTrackerノード数の4%	10	JobTracker用のサーバスレッド数。多ければ多いほど、多数のTaskTrackerからのリクエストをハンドルし易い
mapred.reduce.parallel.copies	5	Mapスロット数の半数	8	ReduceがCopy段階で使うパラレル転送数
mapred.reduce.slowstart.completed.maps	0.05	0.5から0.8	0.5	ReduceがJobのスケジュールをする前に終わっていないMapの割合
mapred.reduce.tasks	1	クラスタ中のReduceタスク数の90%程度	84	ジョブあたりのReduceタスク数
mapred.tasktracker.map.tasks.maximum	2	搭載コア・RAM量で処理可能な数	16	TaskTrackerによって同時稼働可能なMapタスクの最大数。HTを使用している場合は、搭載コア数の1.2～1.5倍が目安。
mapred.tasktracker.reduce.tasks.maximum	2	搭載コア・RAM量で処理可能な数	16	TaskTrackerによって同時稼働可能なMapタスクの最大数。HTを使用している場合は、搭載コア数の1.2～1.5倍が目安。

チューニング項目一覧(MapReduceパラメータ)

項目	既定値	推奨値	設定値	備考
mapred.jobtracker.taskScheduler	org.apache.hadoop.mapred.JobQueueTaskScheduler	org.apache.hadoop.mapred.FairScheduler	org.apache.hadoop.mapred.JobQueueTaskScheduler	タスクスケジュール用クラス
tasktracker.http.threads	40	20台以上のノードで構成されたクラスタの場合、60または80	40	HTTPサーバに許可されたWorker Thread数（中間Map出力をフェッチするためにReducerが利用）
io.sort.mb	100	256 or 512	512	Mapper上でファイルソートを行う際に、マージストリームに割当てられるバッファメモリ量(MB)。タスクのJVMアロケーションによる
io.sort.factor	10	64	64	ファイルソートを行う際に、一度にマージ可能なストリーム数。オープンしたファイルハンドル数を決定
mapred.job.reduce.input.buffer.percent	0.0	0.5 - 0.8	0.5	reduceの過程でmapの出力を保持するために使われるバッファの、総ヒープサイズに対する比率
mapred.job.reuse.jvm.num.tasks	1	-1	-1	タスク JVM の再利用を有効にする設定

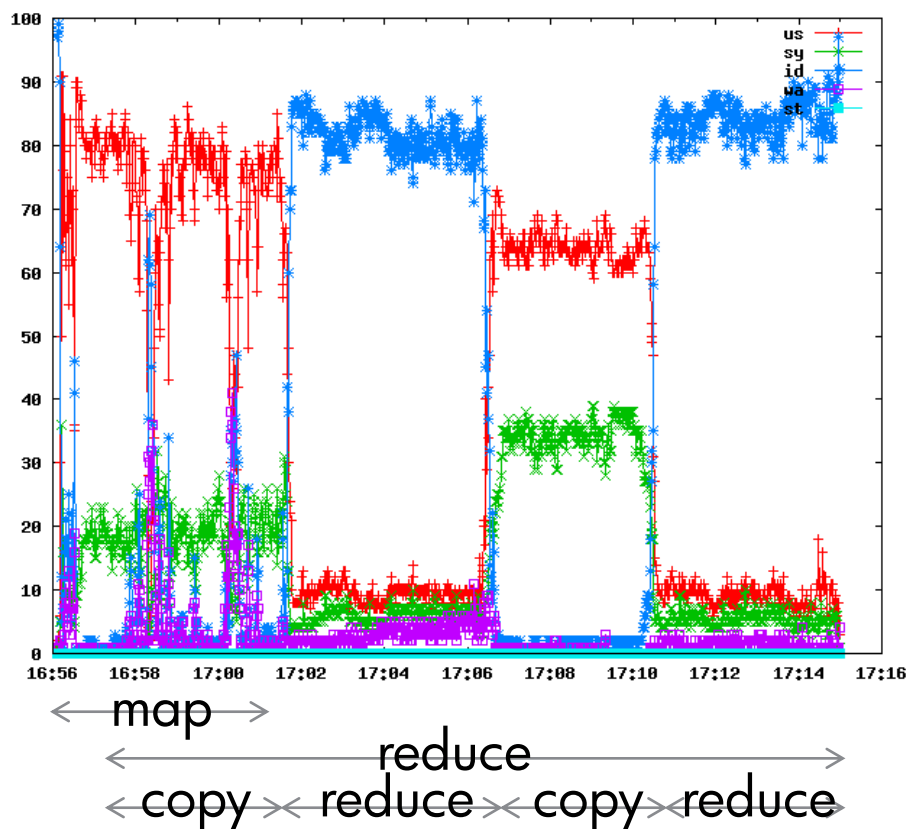
チューニング項目一覧(その他)

- noatimeオプションでディスクをマウント
 - ファイルの読み込み時に最終アクセス時刻の情報を書かないようにするもの
- システムのswappinessを減らす
 - /etc/sysctl.conf内のvm.swappinessを0に設定
- RAID構成より、Hadoop側でラウンドロビンの読み書きをするJBOD構成
 - RAID0の読込・書込は、RAIDアレイ内で最も遅いディスクの速度に合わせて行われる
 - Yahooが行なった調査によれば、JBODのパフォーマンスはRAID0と比較して10 – 30%良好
- snappyにより、MapReduceの中間データを圧縮
 - 中間データのsnappy圧縮はCDH3u1から使用可能

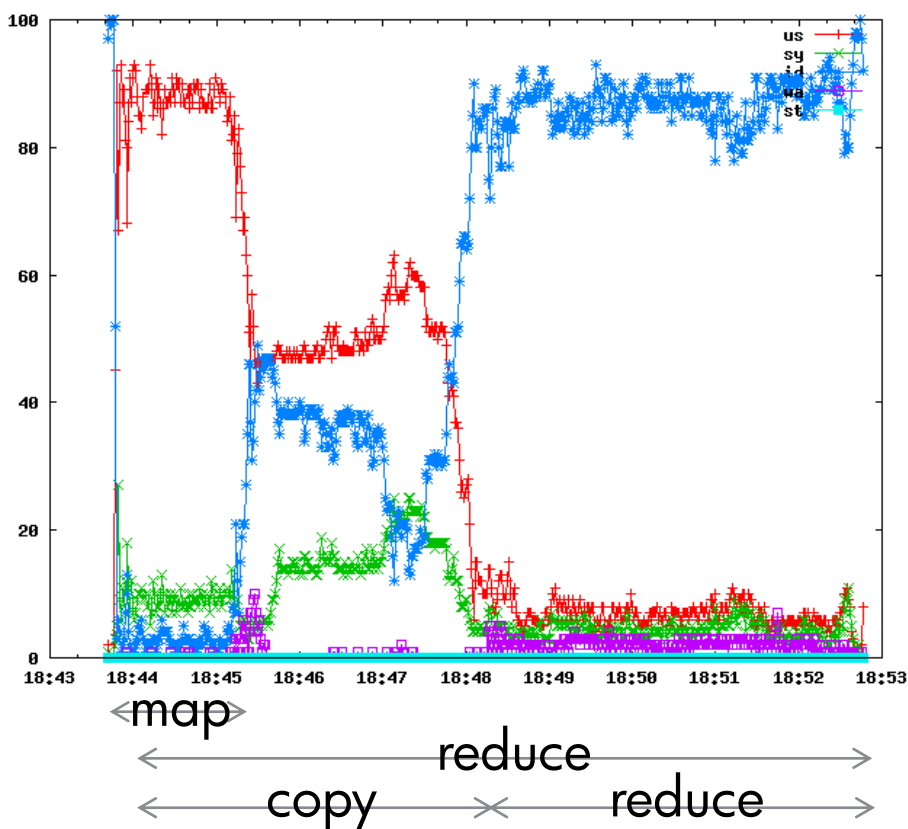


MapReduce処理時のCPU使用率推移

DataNode3ノード構成時の
DataNode1

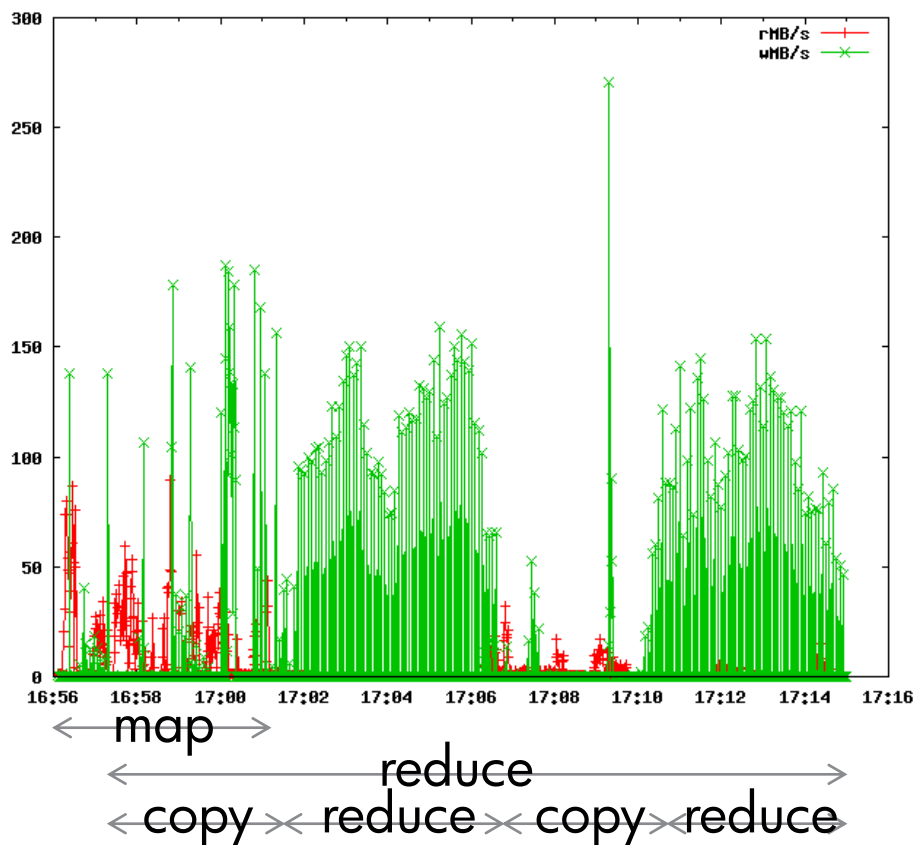


DataNode8ノード構成時の
DataNode1

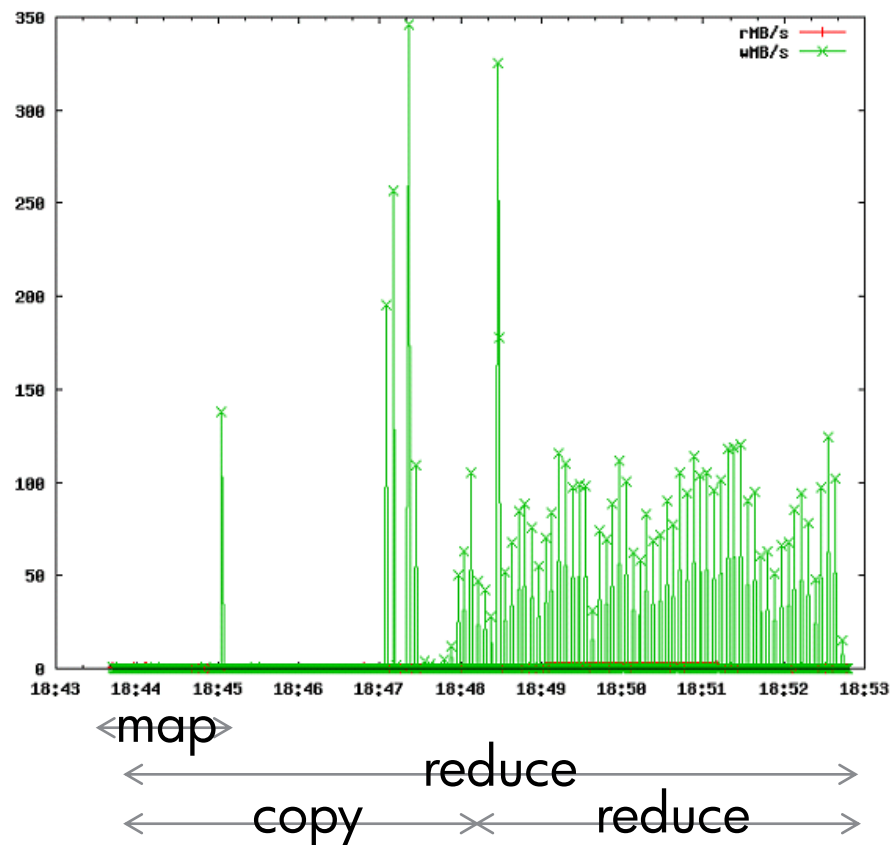


MapReduce処理時のWrite/Read量推移

DataNode3ノード構成時の
DataNode1

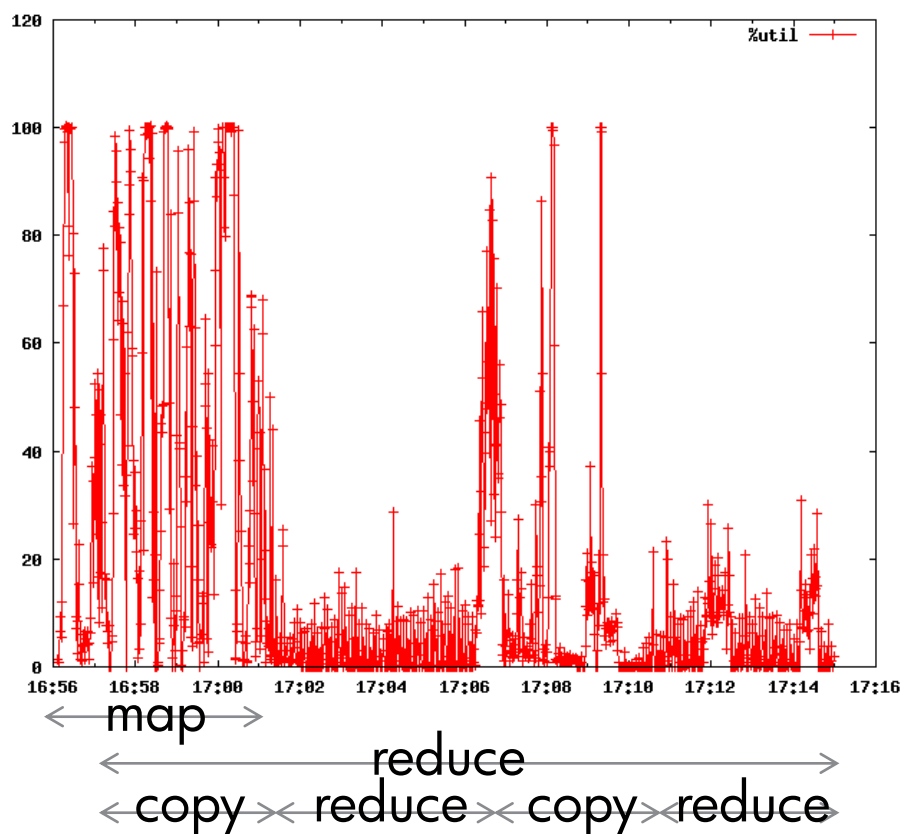


DataNode8ノード構成時の
DataNode1

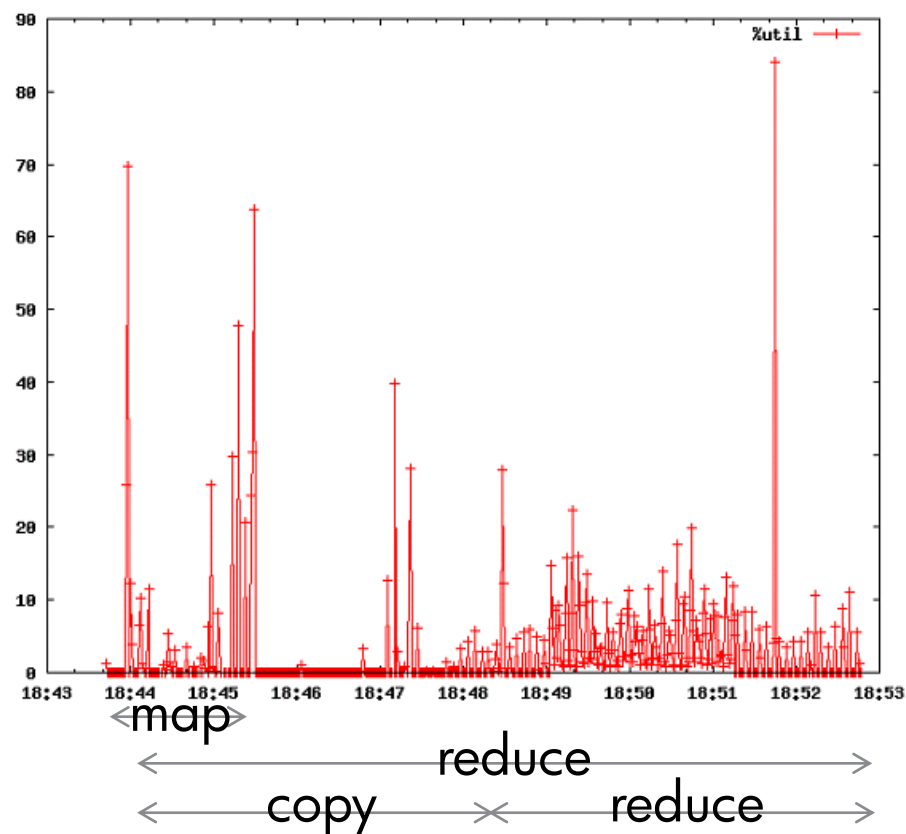


MapReduce処理時のDisk %Util推移

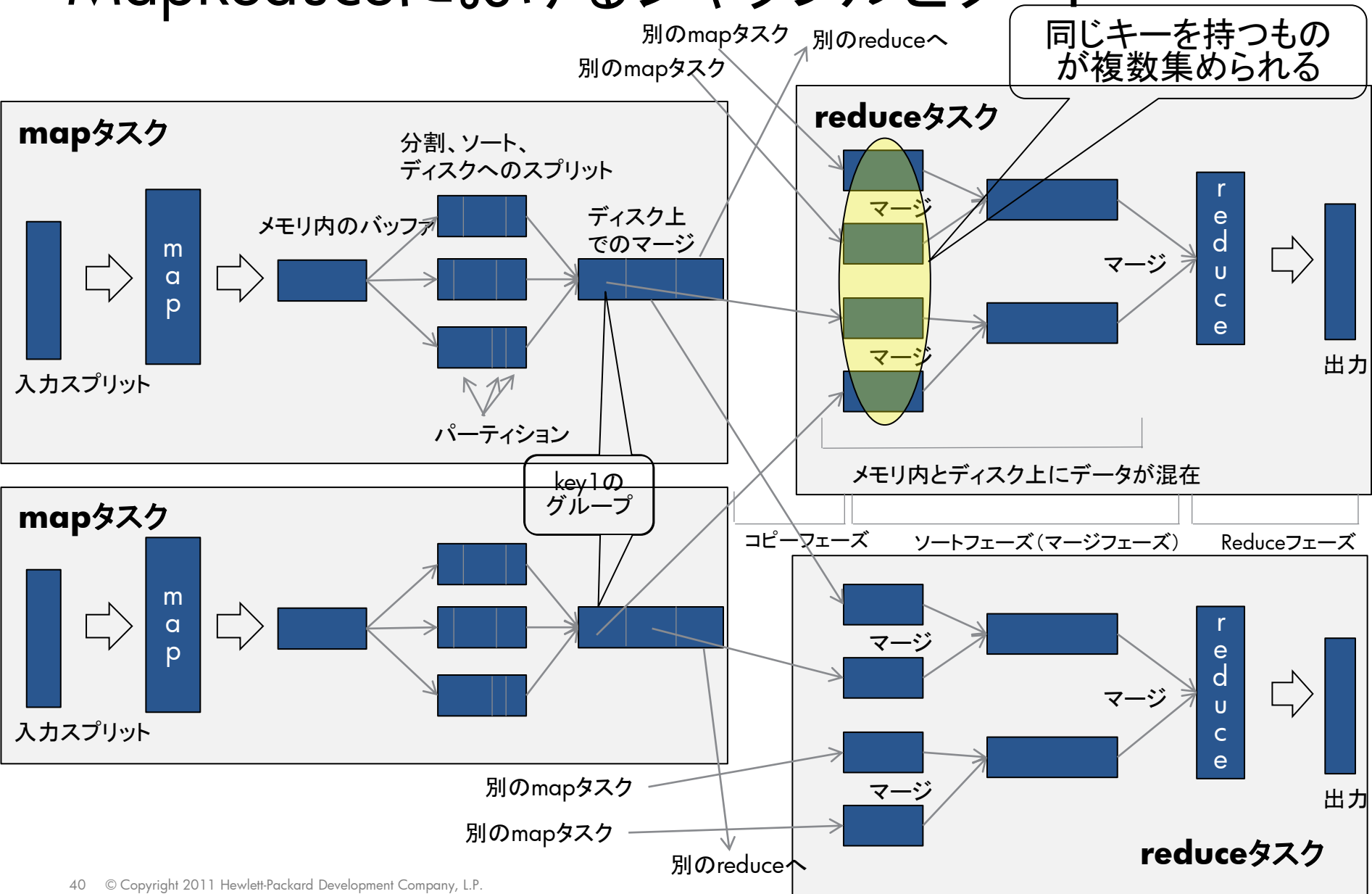
DataNode3ノード構成時の
DataNode1



DataNode8ノード構成時の
DataNode1



MapReduceにおけるシャッフルとソート



Thank you

