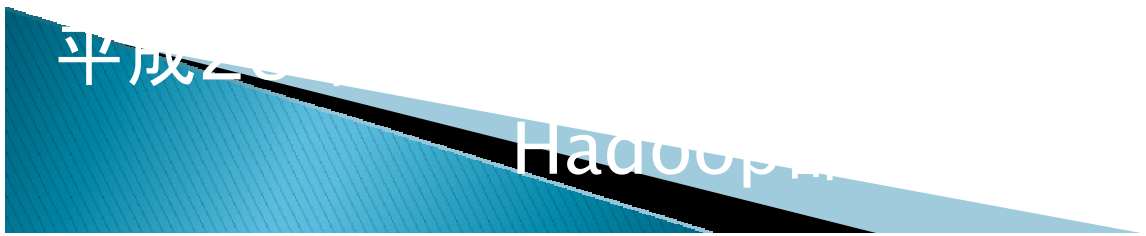
Hadoop？

Hadoopを使って分散処理を実践してみよう。

【部会員】

（部長）	桂	清太郎	モバイルクリエイト（株）
（副部長）	今津	久満	大分シーイーシー（株）
	野口	徹	三井造船システム技研（株）
	高橋	麻衣子	（株）富士通九州システムズ
	早木	建史	モバイルクリエイト（株）
	森	龍也	大分大学大学院
	坂本	親一郎	大分大学大学院

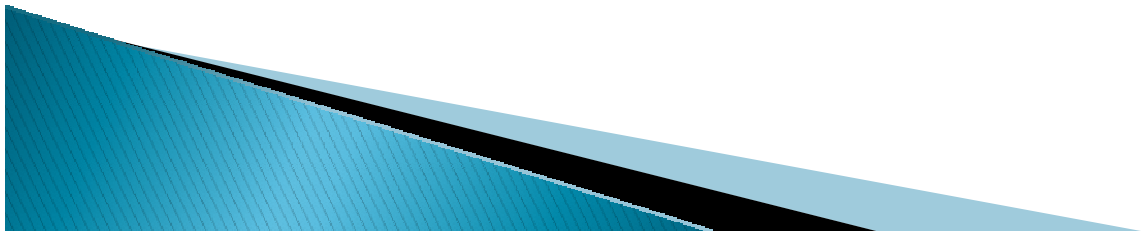


テーマ

Hadoop？

Hadoopを使って分散処理を実践してみよう。

- 現在巷で話題のHadoopについて調査を行う。
- また実際にHadoop環境（ローカルモード、擬似分散モード、完全分散モード）を構築して
- 分散数によって処理速度が上がる事を検証する。
- 今後この技術がどのように活かされていくのか検討する。



Hadoopとは

Hadoopって
なに？？

Hadoop ＝「大規模なデータを処理するための
並列分散処理基盤」

複数のサーバを同時に用いて
短時間に大量に処理することを可能にするプラットフォーム

Hadoopの利用用途

- 主な用途
  - ・ログ解析
  - ・ランキング
  - ・検索
- 利用企業
  - 国内
  - パット

Hadoopを利用する
企業は増加中！！
不可欠な技術になり
そう！！

Hadoopについて

- 由来

・Hadoop創設者のDoug Cuttingの
息子のゾウのぬいぐるみの名前

- 特徴

・Google社の、MapReduce/GFSの論文をもとに実装。

・Apache Software Fundationが開発・公開している
オープンソースソフトウェア（Javaをベースとしている）

メリット、デメリット

- メリット

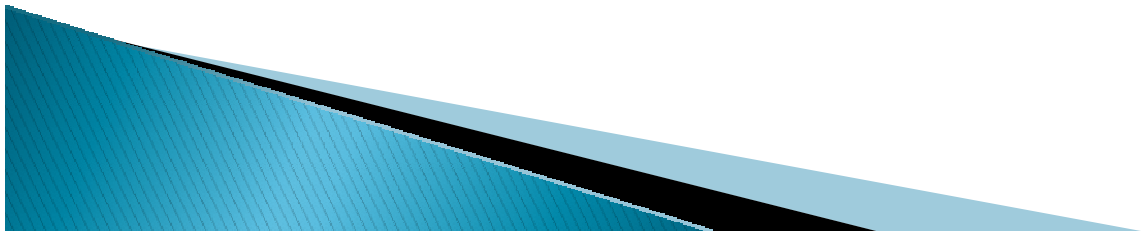
  ・ペタバイトクラスの大量データのバッチ処理を高速に処理することができる。

  ・サーバの台数を増やすことで、スケールアウトを可能にする。

- デメリット

  ・Webのように即座に結果が返ってくるリアルタイムな処理には不向き

  ・管理ノードに障害が起こるとHadoopシステム自体がダウンする。
  ＝SPOF（単一障害点）問題

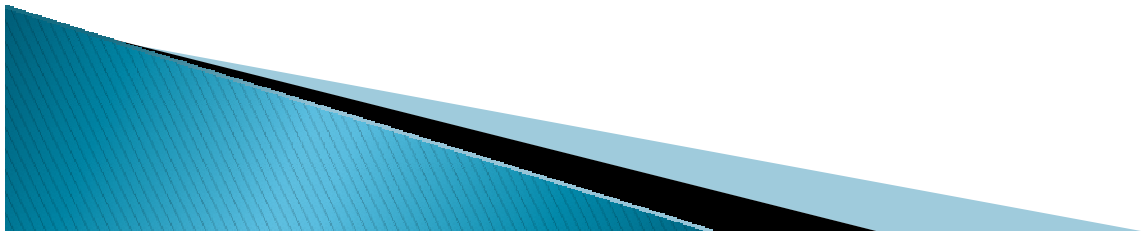


Hadoopの構成

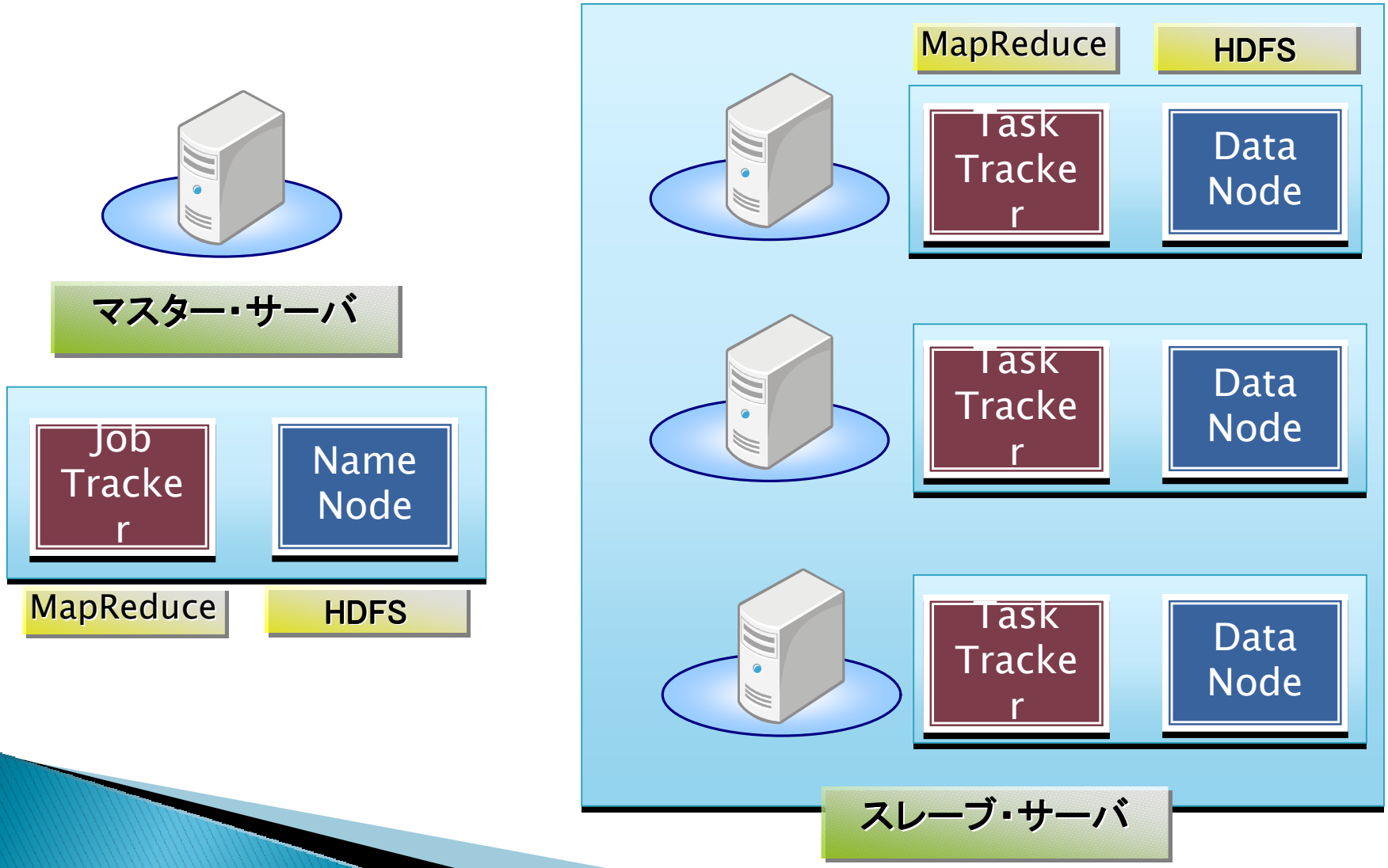
- Hadoop=「HDFS」と「MapReduce」で構成
- HDFS

  Hadoop Distributed File Systemの略で、
  Hadoopの分散ファイルシステム
- MapReduce

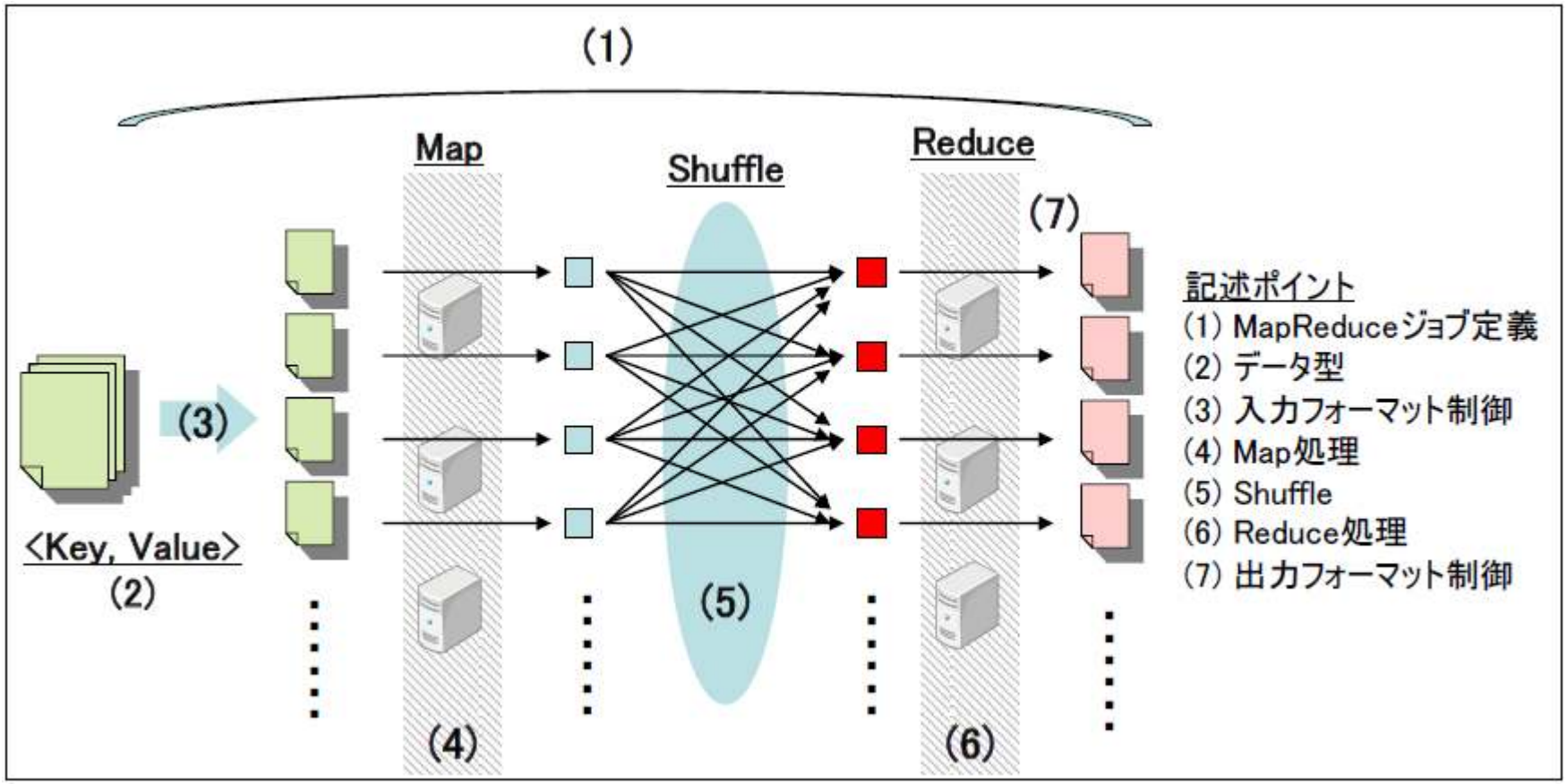
  Googleが開発した、大規模データを並列分散処理するための
  フレームワーク



サーバ構成について


MapReduce
HDFS
Task Tracker
Data Node
Task Tracker
Data Node
Task Tracker
Data Node
スレーブ・サーバ
マスター・サーバ
Job Tracker
Name Node
MapReduce
HDFS

MapReduceについて



MapReduceについて（例）

THE END OF MONEY IS THE END OF LOVE

Key、Valueのペアを作る「Map」

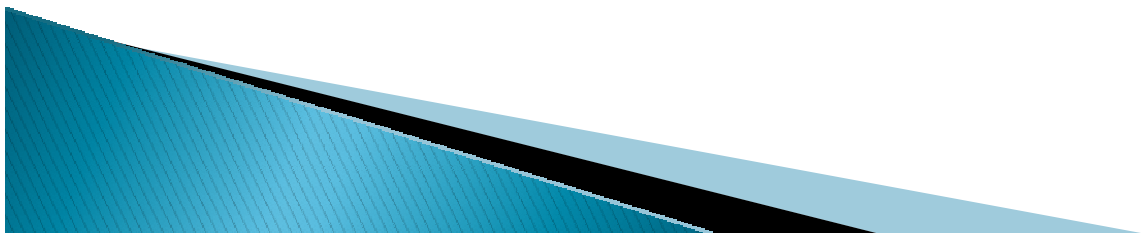
THE	END	OF	MONEY	IS	THE	END	OF	LOVE
1	1	1	1	1	1	1	1	1

同じKeyを持つペアを束ねる「Shuffle」
※キー順にソートし、同じKeyを束ねる

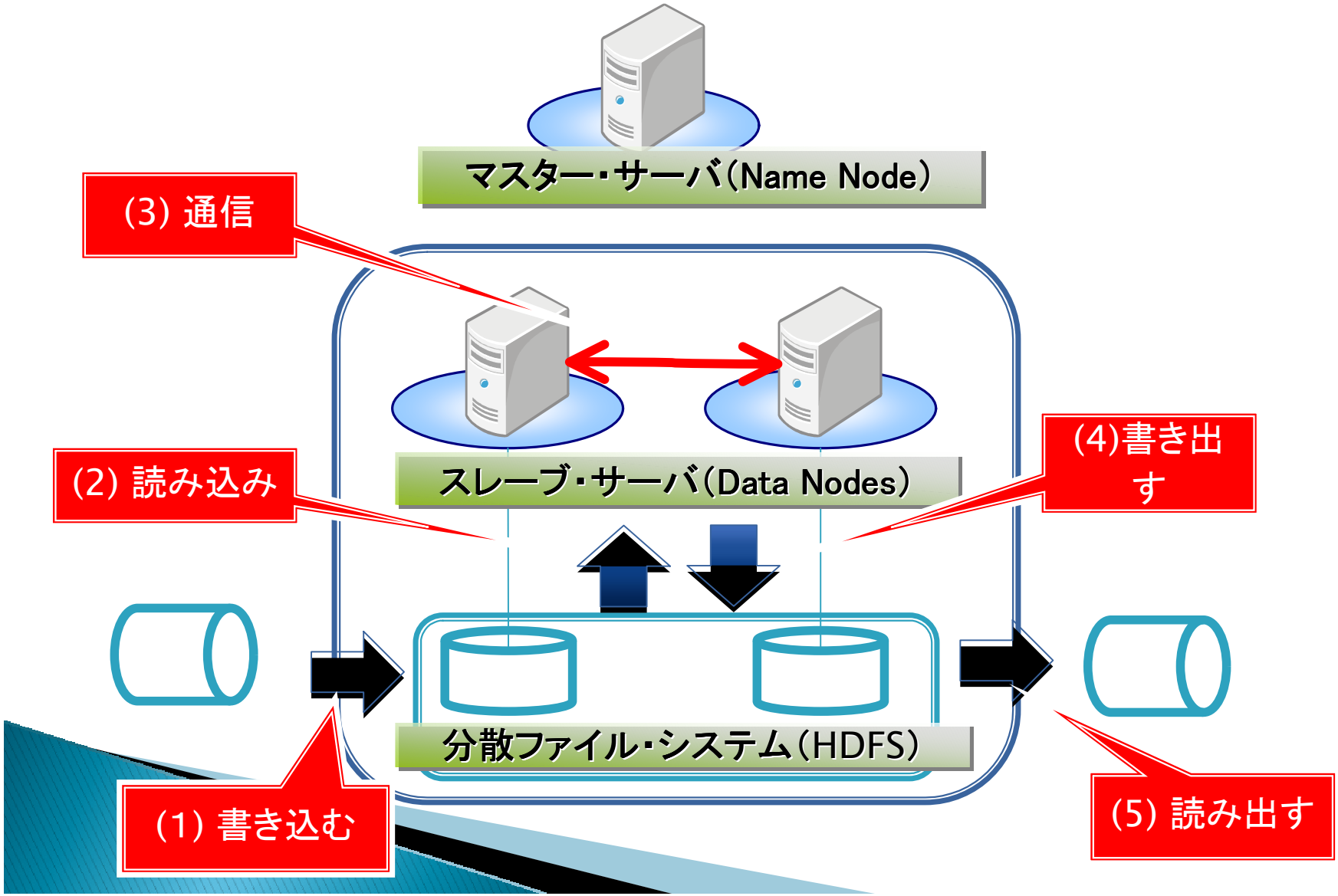
END		IS	LOVE	MONEY	OF		THE	
1	1	1	1	1	1	1	1	1

Shuffleの結果を入力として処理を行う「Reduce」

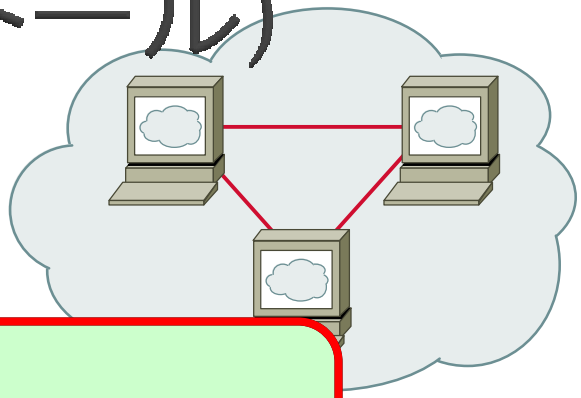
END	IS	LOVE	MONEY	OF	THE
2	1	1	1	2	2



HDFSについて


マスター・サーバ（Name Node）
(3) 通信
スレーブ・サーバ（Data Nodes）
(2) 読み込み
(4)書き出す
分散ファイル・システム（HDFS）
(1) 書き込む
(5) 読み出す

Hadoop環境構築(インストール)



- 動作検証環境

Hadoop 0.21.0

- JAVA: JRE 6
- 仮想OS: Cent OS 6
- 仮想マシン: VMware Player / Parallels Desktop
- ホストOS: Windows / Mac OS

Hadoop環境構築(インストール)

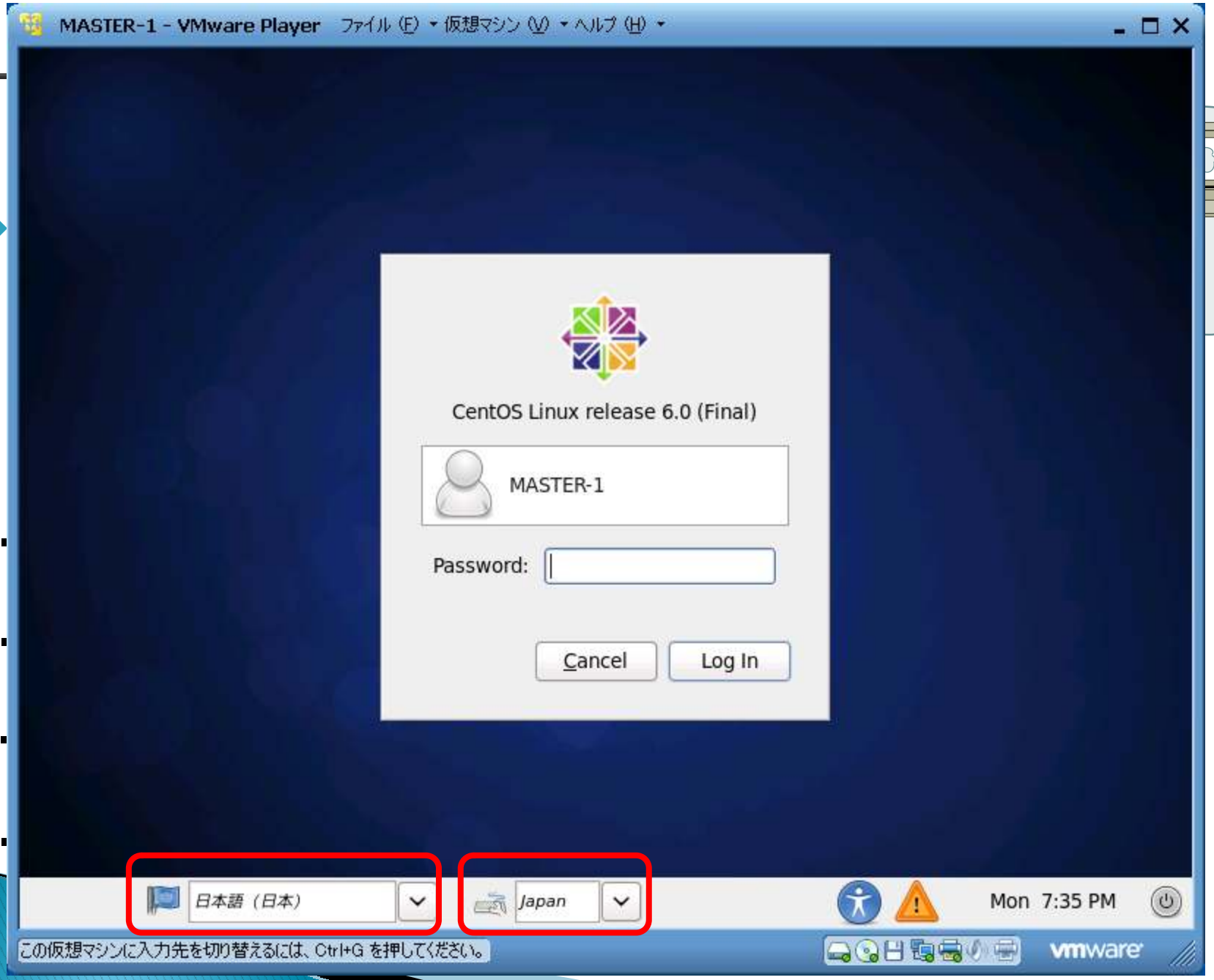
- 動作

・JAV

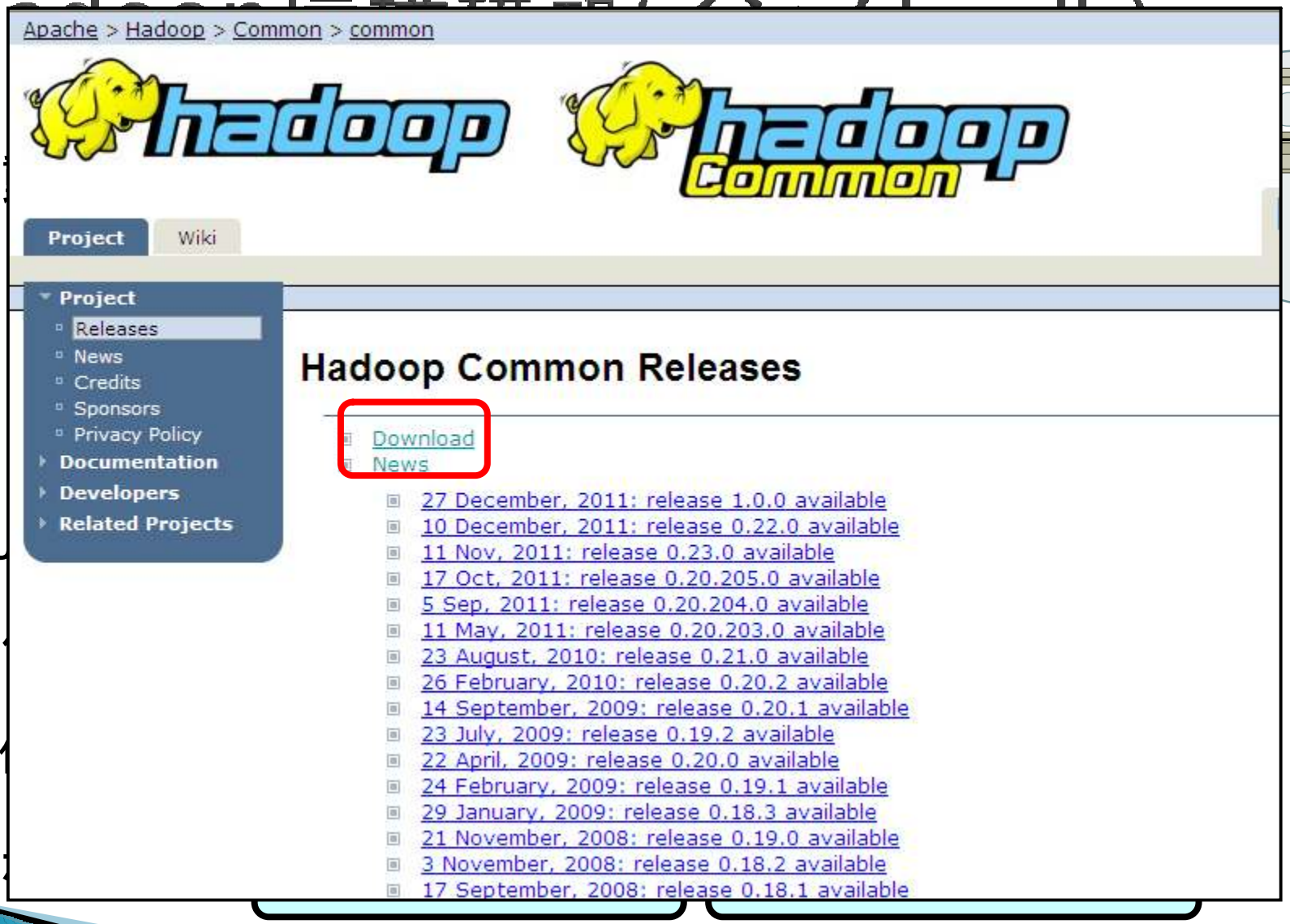
・仮想

・仮想

・ホス


MASTER-1 - VMware Player
ファイル (F)
仮想マシン (V)
ヘルプ (H)
CentOS Linux release 6.0 (Final)
MASTER-1
Password:
Cancel
Log In
日本語（日本）
Japan
Mon 7:35 PM
この仮想マシンに入力先を切り替えるには、Ctrl+G を押してください。
vmware

H


Apache > Hadoop > Common > common
hadoop
hadoop Common
Project
Wiki
Project
Releases
News
Credits
Sponsors
Privacy Policy
Documentation
Developers
Related Projects
Hadoop Common Releases
Download
News
27 December, 2011: release 1.0.0 available
10 December, 2011: release 0.22.0 available
11 Nov, 2011: release 0.23.0 available
17 Oct, 2011: release 0.20.205.0 available
5 Sep, 2011: release 0.20.204.0 available
11 May, 2011: release 0.20.203.0 available
23 August, 2010: release 0.21.0 available
26 February, 2010: release 0.20.2 available
14 September, 2009: release 0.20.1 available
23 July, 2009: release 0.19.2 available
22 April, 2009: release 0.20.0 available
24 February, 2009: release 0.19.1 available
29 January, 2009: release 0.18.3 available
21 November, 2008: release 0.19.0 available
3 November, 2008: release 0.18.2 available
17 September, 2008: release 0.18.1 available

Hadoop環境構築(インストール)

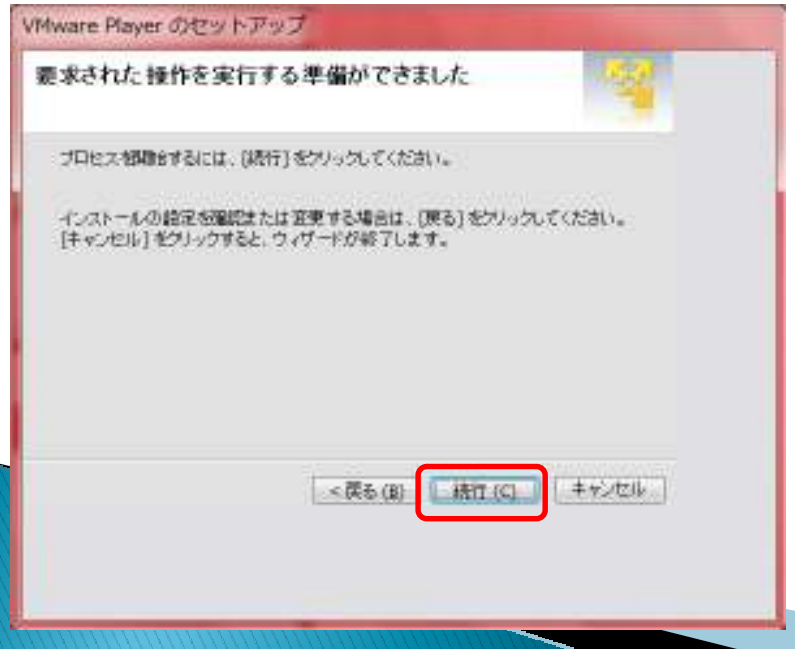
- 環境構築
  ・vmware社VMWarePayer
  インストール
  ①インターネットで
  「VMWarePlayer」で検索する
  ②「ダウンロード」
  を選択する

Hadoop環境構築(インストール)

- 環境構築
  ・vmware社VMWarePayerインストール
  ③「vmware-player.exe」を起動し画面に従い「次へ」を選択する

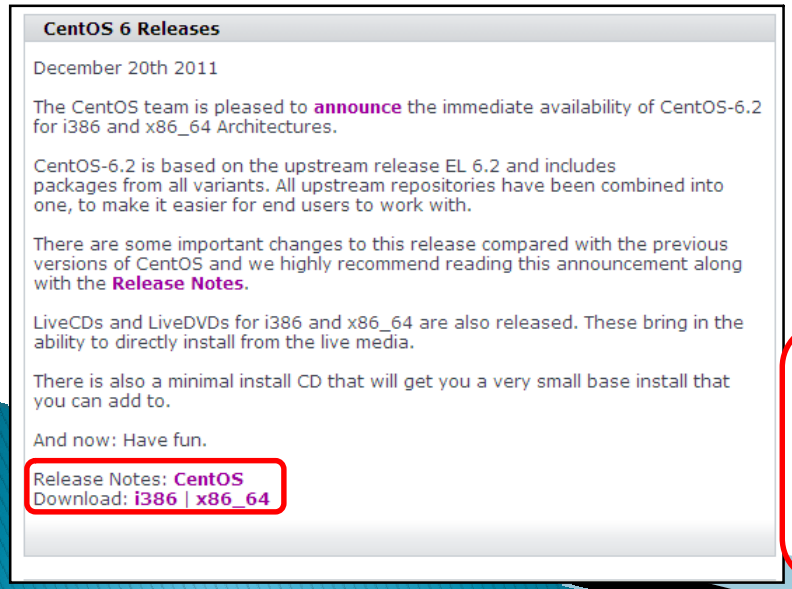
Hadoop環境構築(インストール)

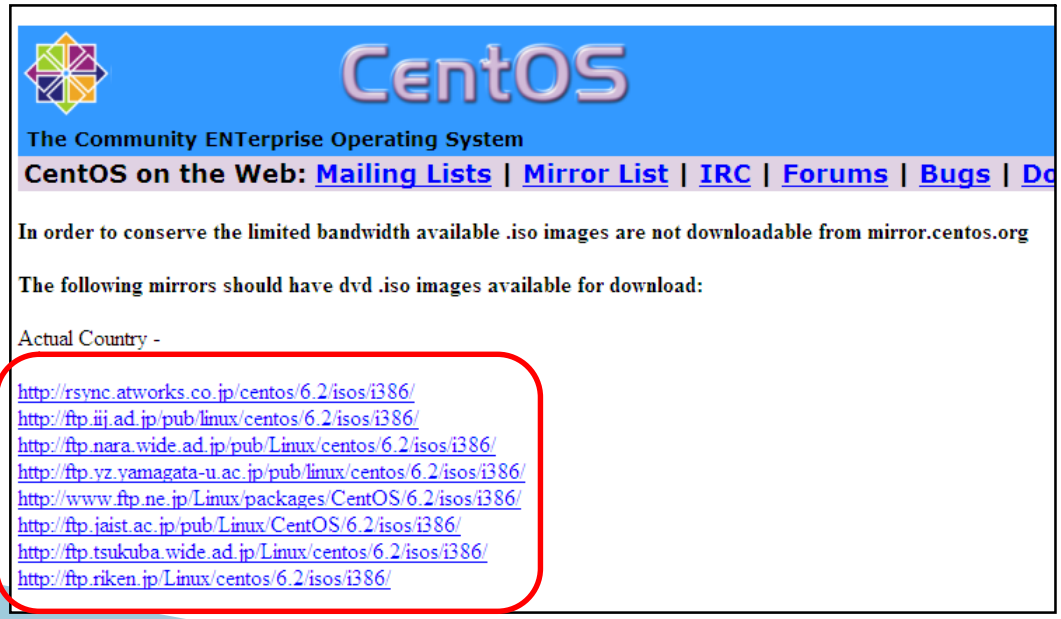
- 環境構築
  ・vmware社VMWarePayerインストール
  ④「続行」を選択するとインストールされる
  ⑤再起動する



Hadoop環境構築(インストール)

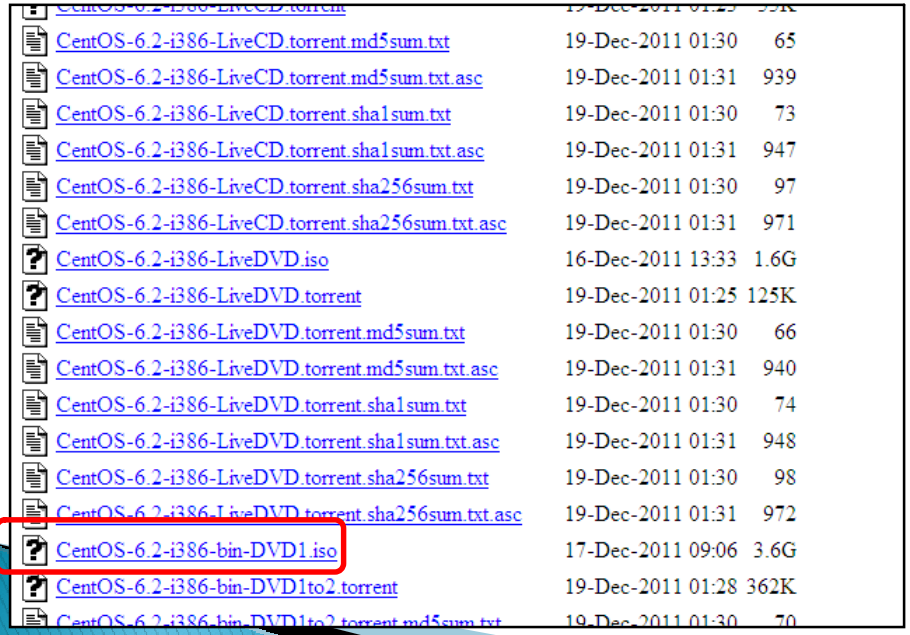
- 環境構築
  ・CentOSインストール
  ①インターネットで「CentOS」で検索
  　http://www.centos.org/
  ②「CentOS 6 Releases」から、
  　「Download i386」を選択する





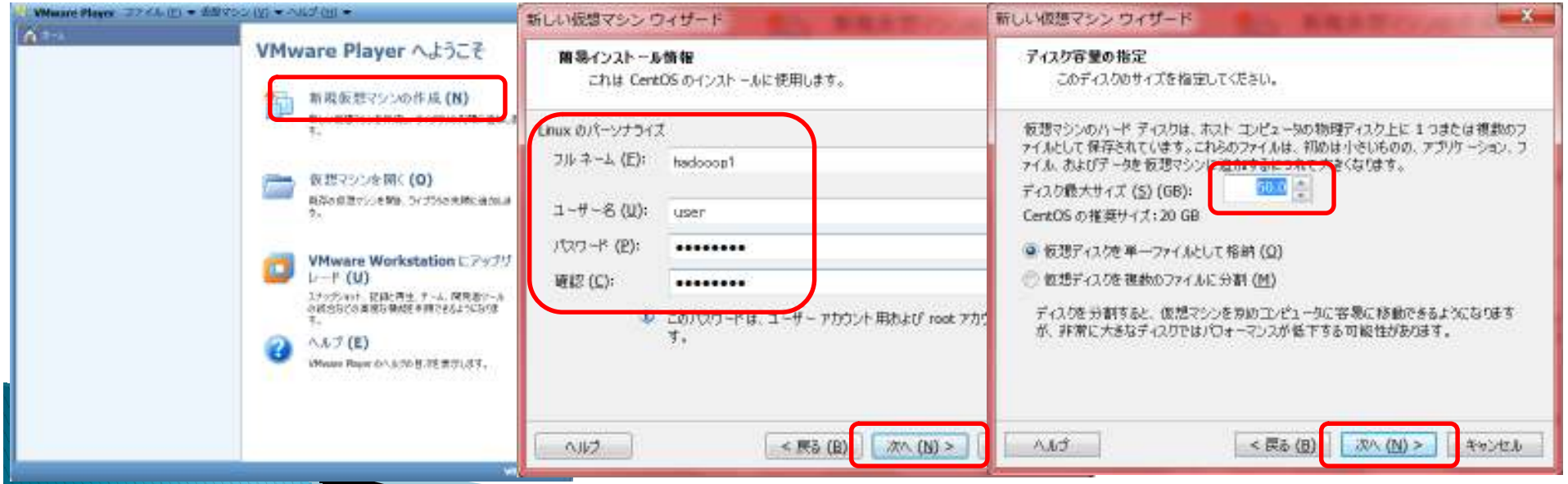
Hadoop環境構築(インストール)

- 環境構築
  ・CentOSインストール
  ③「CentOS-6.2-i386-LiveDVD.iso 」
  　をダウンロードする



Hadoop環境構築(インストール)

- 環境構築
  ・CentOSインストール
  ④VMWarePlayerを起動する
  ⑤「新規仮想マシンの作成」を選択する
  ⑥LINUXパーソナライズを入力し、「次へ」を押す
  ⑦ディスクのサイズを入力し、「次へ」を押す



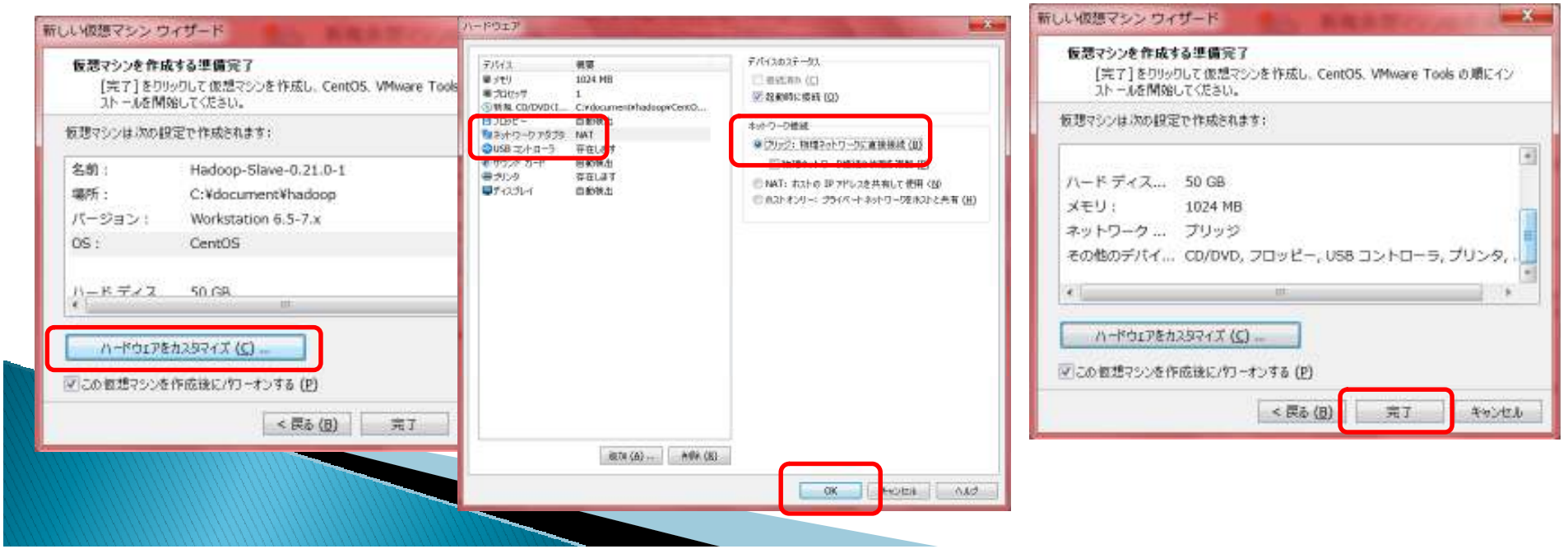
Hadoop環境構築(インストール)

- 環境構築
  ・CentOSインストール

⑧ハードウェアをカスタマイズを選択する

⑨「ネットワークアダプタ」→「ネットワーク接続」
→「ブリッジ」と選択し、「OK」を押す

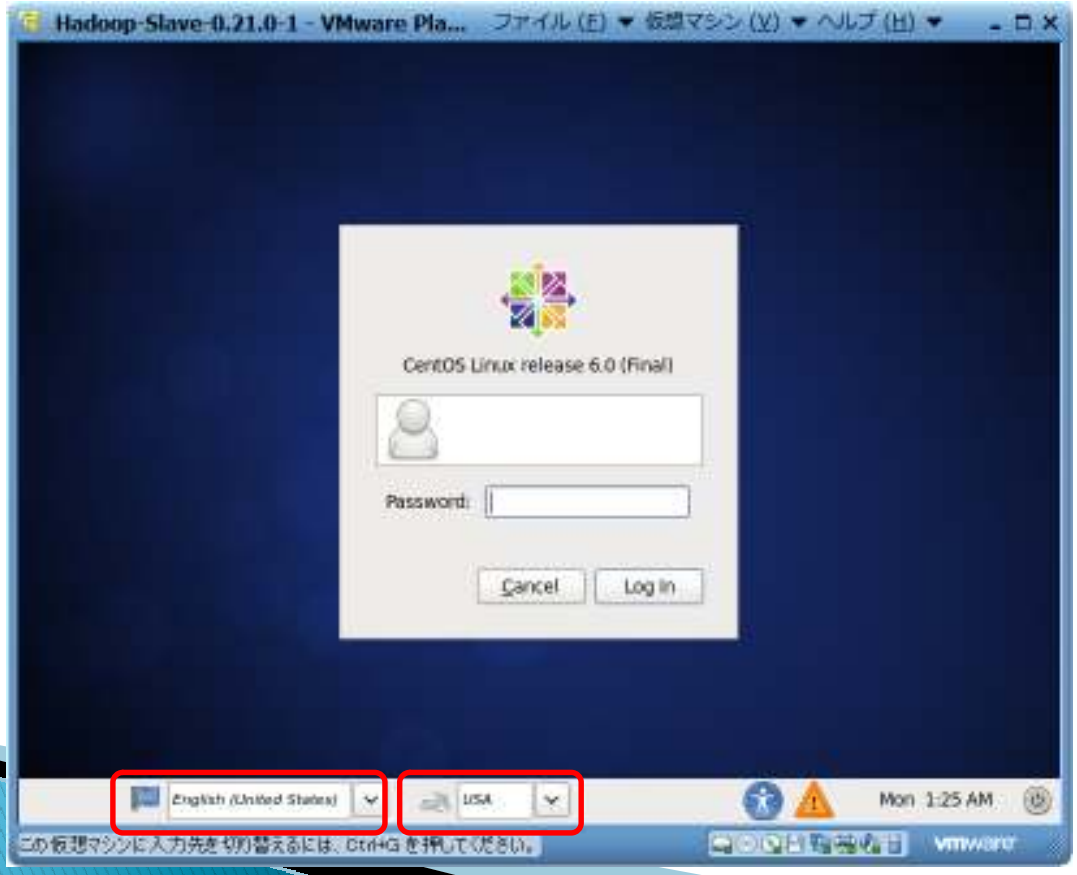
⑩「完了」を押すとインストールが開始される



Hadoop環境構築(インストール)

- 環境構築
  - CentOSインストール

⑪ログイン画面で言語とキーボード設定を
日本語に変更する



Hadoop環境構築(インストール)

- 環境構築
  ・Javaインストール
  ①インターネットで「JRE」検索する
  ②「Linux RPM」をダウンロードする

Solaris

Solaris (32ビット) ファイルサイズ: 25.2 MB	手順	今すぐチェック
Solaris (64ビット)* ファイルサイズ: 11.2 MB + 25.2 MB 32-bit Solaris	手順	Java のインストールが終了したら、お使いの Web ブラウザを再起動して、Java が正しくインストールされていることを確認してください。
Solaris x86 ファイルサイズ: 20.3 MB	手順	
Solaris x64 * ファイルサイズ: 7.5 MB	手順	

* Java アプレットと Java Web Start を使用する場合は、32 ビット バージョンを使用してください。

Linux

Linux RPM ファイルサイズ: 20.2 MB	手順	今すぐチェック
Linux (自己解凍ファイル) ファイルサイズ: 20.7 MB	手順	Java のインストールが終了したら、お使いの Web ブラウザを再起動して、Java が正しくインストールされていることを確認してください。
Linux x64 * ファイルサイズ: 20.3 MB	手順	
Linux x64 RPM * ファイルサイズ: 19.7 MB	手順	

* Java アプレットと Java Web Start を使用する場合は、32 ビット バージョンを使用してください。

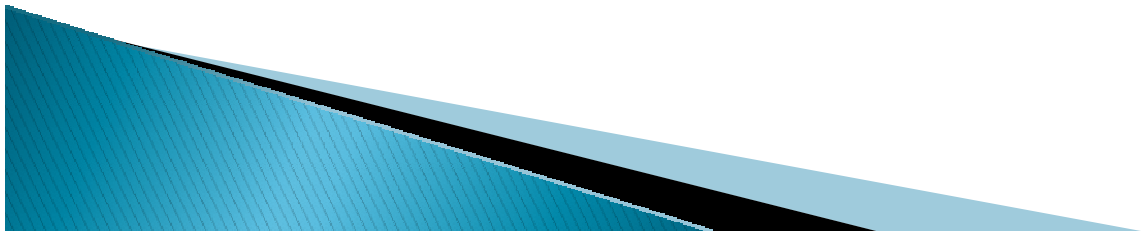
Hadoop環境構築(インストール)

- 環境構築
  ・Javaインストール
  ③コンソールでrootでログインする
  ④JREをインストールする

```
[root@localhost]# chmod 755 jre-6u27-linux-i586.rpm.bin
[root@localhost]# ./jre-6u27-linux-i586-rpm.bin
```

⑤ JAVAのパスを設定する

```
 [root@localhost]# chmod 755 /etc/bashrc
 [root@localhost]# export JAVA_HOME=/usr/java/latest
 [root@localhost]# echo "export JAVA_HOME=/usr/java/latest"
>> /etc/bashrc
 [root@localhost]# tail -n 2 /etc/bashrc
```



Hadoop環境構築(インストール)

- 環境構築
  ・Hadoopインストール
  ①Hadoopユーザー・グループを設定する
  [root@localhost]# groupadd -g 1000 hadoop
  [root@localhost]# useradd -g hadoop -u 1000 -p hadoop hadoop
  ②「Hadoop」で検索する
  ③「Hadoop Common」を選択する
  ④「Download」を選択する

Hadoop環境構築(インストール)

- 環境構築
  ・Hadoopインストール
  ⑤「Download a release now!」を選択する
  ⑥ダウンロード元を選択する

Hadoop環境構築(インストール)

- 環境構築
  - Hadoopインストール

⑦「hadoop-0.21.0」を選択する

⑧「hadoop-0.21.0.tar.gz」をダウンロードする

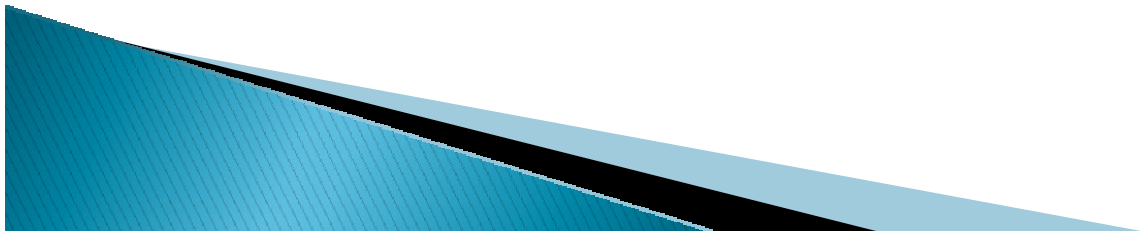
Hadoop環境構築(インストール)

- 環境構築
  ・Hadoopインストール
  ⑨ Hadoopを解凍し・所有ユーザー変更する

```
[root@localhost]# tar xzf hadoop-0.21.0.tar.gz
[root@localhost]# mv hadoop-0.21.0 /usr/local
[root@localhost]# chown -R hadoop:hadoop /usr/local/hadoop-
0.21.0
```

  ⑩ Hadoopのパスを設定する

```
 [root@localhost]# ln -s /usr/local/hadoop-0.21.0
/usr/local/hadoop
 [root@localhost]# export HADOOP_HOME=/usr/local/hadoop
 [root@localhost]# echo "export
HADOOP_HOME=/usr/local/hadoop" >> /etc/bashrc
 [root@localhost]# tail -n 3 /etc/bashrc
```



Hadoop環境構築(インストール)

- 環境構築
  ・Hadoopインストール

⑪設定ファイルを変更する
/usr/local/hadoop/conf
・hdfs-site.xmlを編集

```
<configuration>
  <property>
    <name>dfs.name.dir</name>
    <value>${hadoop.tmp.dir}/dfs/name</value>
  </property>
  <property>
    <name>dfs.data.dir</name>
    <value>${hadoop.tmp.dir}/dfs/data</value>
  </property>
</configuration>
```

NameNodeで使用するディレクトリ

DataNodeで使用するディレクトリ

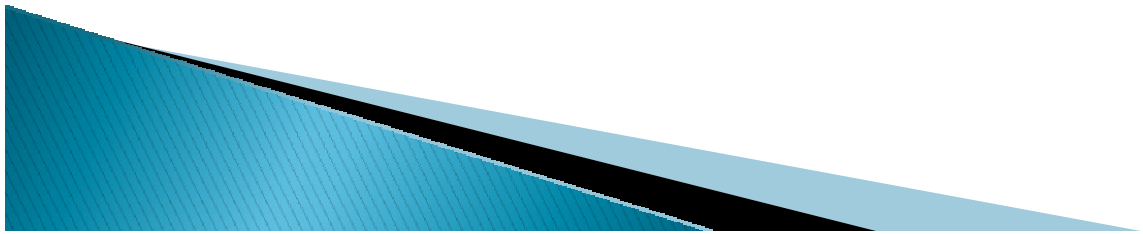
Hadoop環境構築(インストール)

- 環境構築
  - Hadoopインストール

⑩設定ファイルを変更する
/usr/local/hadoop/conf
- hadoop-env.shを編集（Hadoopで使う環境変数）

```
export HADOOP_LOG_DIR=/var/log/hadoop
export HADOOP_PID_DIR=/var/run/hadoop
export HADOOP_IDENT_STRING=sample
```

- ログ保存先
- PIDファイル保存先
- 表示されるインスタンス名

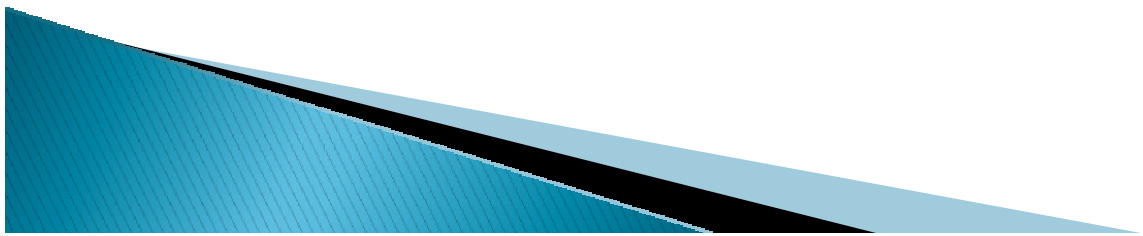


Hadoop環境構築(インストール)

- 環境構築
  ・Hadoopインストール

  ⑪ディレクトリを作成する

```
[root@localhost]# mkdir /hadoop
[root@localhost]# mkdir /var/log/hadoop
[root@localhost]# mkdir /var/run/hadoop
[root@localhost]# chmod 777 /hadoop
[root@localhost]# chown -R hadoop:hadoop /hadoop
[root@localhost]# chown -R hadoop:hadoop /var/log/hadoop
[root@localhost]# chown -R hadoop:hadoop /var/run/hadoop
```



Hadoopの動作モード

- ローカルモード
  デーモンは一切動かさず、単一のJVM上で全て実行する。
- 擬似分散モード
  Hadoopデーモンがローカルマシン上で動作する。
- 完全分散モード
  Hadoopデーモンが複数のマシンからなるクラスタ上で動作する。

モード	サーバー台数	HDFS利用の有無	利用目的
ローカル	1台	無し	デバッグ・開発
擬似分散	1台	有り	小規模なクラスタでのシミュレーション
完全分散	Master1台 Slave n台	有り	実際の運用

設定ファイル	設定箇所	設定値(モード別)		
		ローカル	擬似分散	完全分散
core-site.xml	データ領域に関するローカルディレクトリ	/hadoop	/hadoop	/hadoop
	Namenodeサービスのアドレス	不要	hdfs://localhost:54310/	hdfs://master:54310/
hdfs-site.xml	Namenodeファイル格納ディレクトリ	不要	${hadoop.tmp.dir}/dfs/name	${hadoop.tmp.dir}/dfs/name
	HDFSブロックの格納ディレクトリ	不要	${hadoop.tmp.dir}/dfs/data	${hadoop.tmp.dir}/dfs/data
mapred-site.xml	JobTrackerのアドレス	local	localhost:54311	master:54311
	MapRecudeデータのローカル保存先	不要	${hadoop.tmp.dir}/mapred	${hadoop.tmp.dir}/mapred

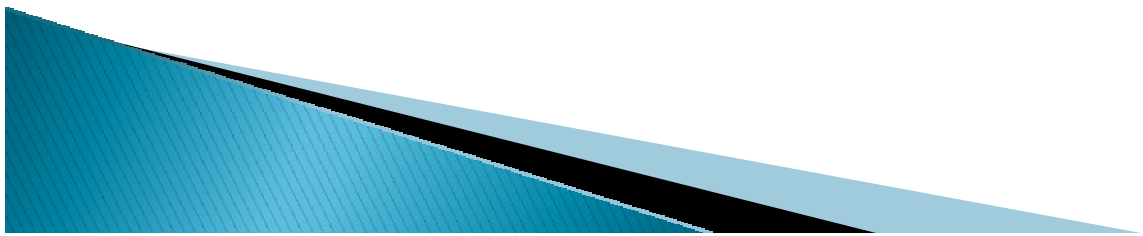
ローカルモード

- サンプルの円周率計算を行う

```
# hadoop jar hadoop-mapred-examples-0.21.0.jar pi 10 10000
```

- 実行結果

36.4秒で計算終了



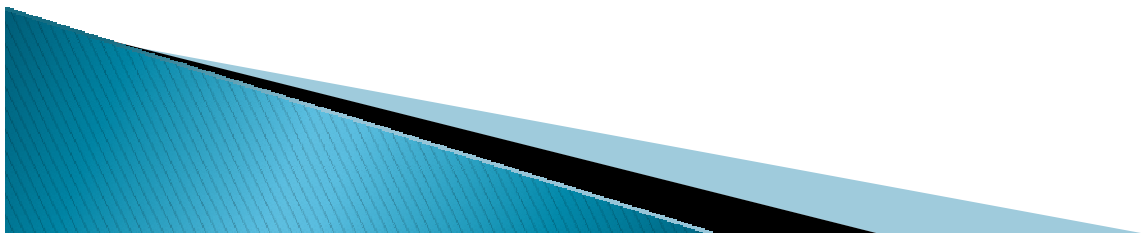
擬似分散モード

- サンプルの円周率計算を行う

```
# hadoop jar hadoop-mapred-examples-0.21.0.jar pi 10 10000
```

- 実行結果

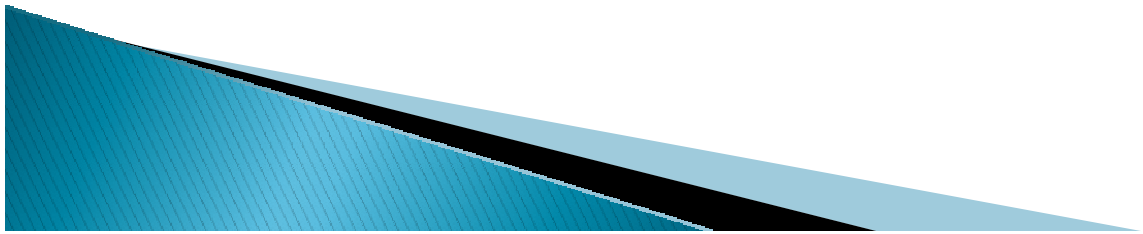
141秒で計算終了



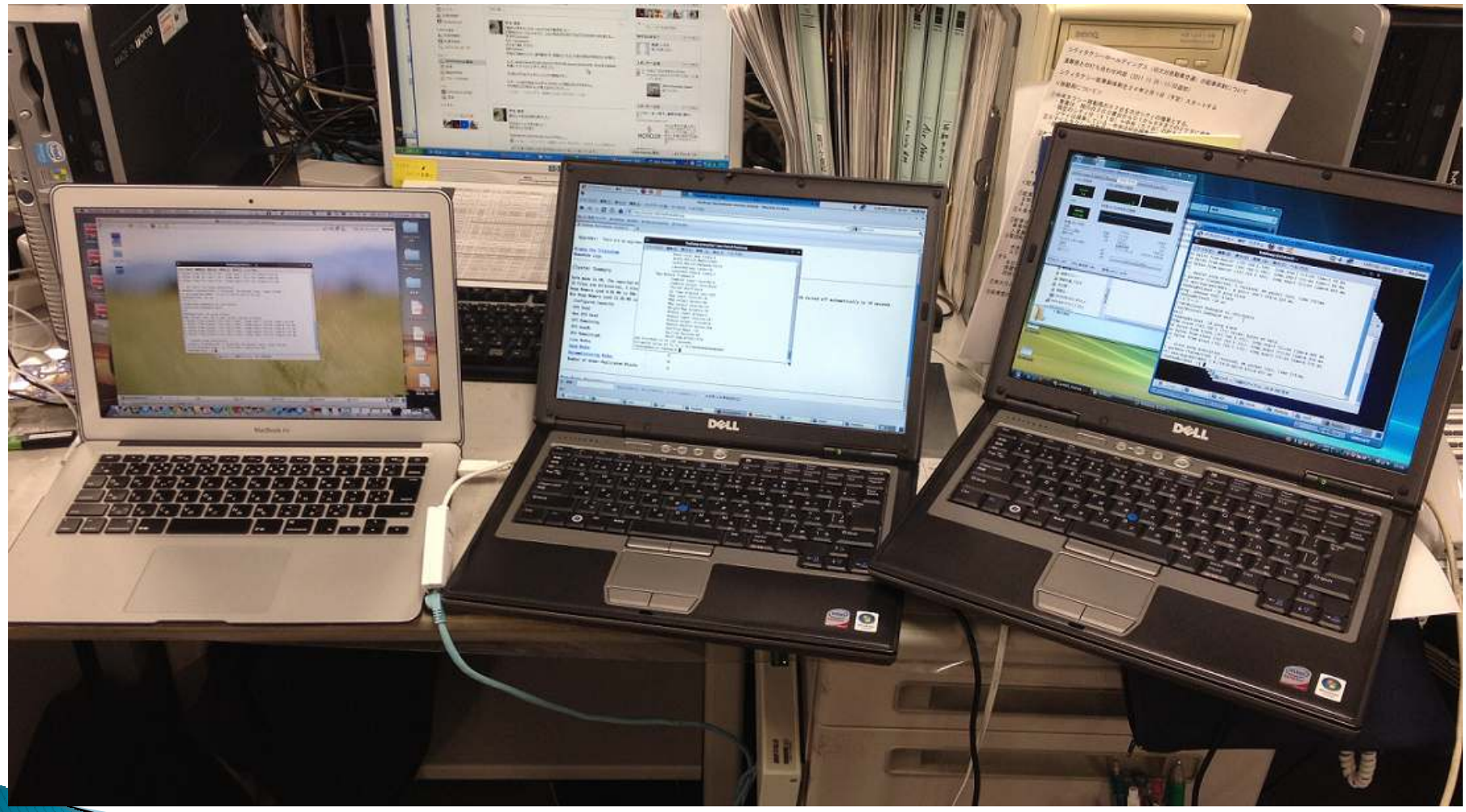
ローカル・擬似分散モードの比較

- ローカルモード ⇒ 35.4秒
- 擬似分散モード ⇒ 141秒

- 擬似分散モードの方が処理時間が長い
  - 通常複数台で行う分散処理を1台で行っているため
  - 使用するファイルシステム
    - ローカルモード：ローカルファイルシステム
    - 擬似分散：HDFS



完全分散モード



完全分散モード

- 完全分散モード　変更点

①hostsファイルの編集(名前解決)

```
# vi /etc/hosts
  192.168.5.101 master master
  192.168.5.102 slave slave
  192.168.5.103 slave2 slave2
```

追加

②ssh公開鍵の設定

⇒master,slave互いに設定しないと上手くいかなかった

(masterで実行)

```
# ssh-keygen -t rsa
# cat .ssh/id_rsa.pub >> .ssh/authorized_keys
# chmod 700 .ssh
# chmod 600 .ssh/*
```

(slaveで実行)

```
# ssh-keygen -t rsa
# scp ~/.ssh/id_rsa.pub hadoop@master:~/.ssh/id_rsa_slave.pub
```

(masterで実行)

```
# cat ~/.ssh/id_rsa_slave.pub >> ~/.ssh/authorized_keys
# scp .ssh/authorized_keys hadoop@slave:~/.ssh/
# ssh hadoop@slave 'chmod 700 ~/.ssh ; chmod 600 ~/.ssh/*'
```

Slave2に対しても
同じように設定

完全分散モード

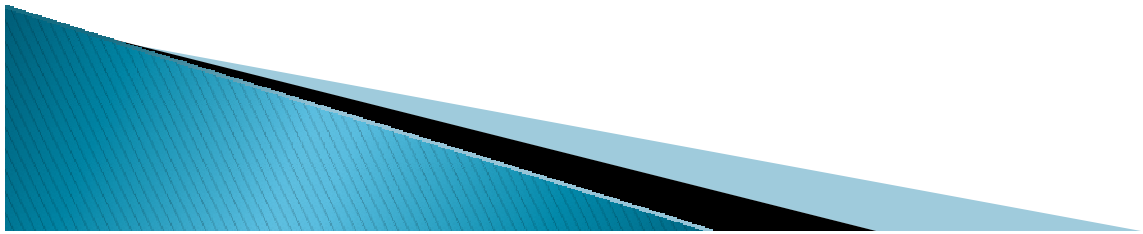
- 完全分散モード　変更点
  ③Hadoop設定ファイルの編集

  core-site.xml (Hadoop共通のプロパティ)

```
<configuration>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>/hadoop</value>
  </property>
  <property>
    <name>fs.default.name</name>
    <value>hdfs://master:54310</value>
  </property>
</configuration>
```

データ領域のマスター

NameNodeサービスのアドレス



完全分散モード

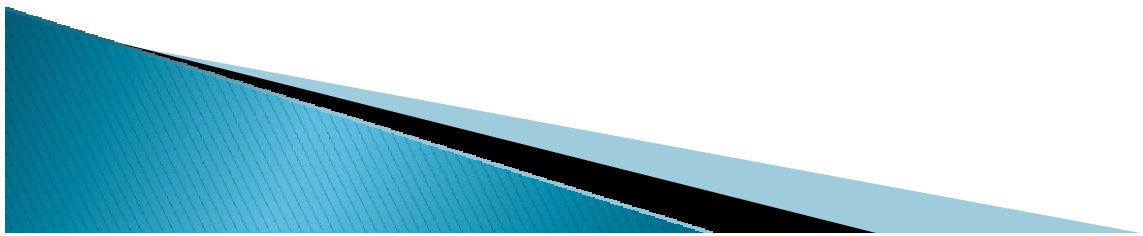
- 完全分散モード　変更点
  ④Hadoop設定ファイルの編集

  hdfs-site.xml (HDFSのプロパティ)

```
<configuration>
  <property>
    <name>dfs.name.dir</name>
    <value>${hadoop.tmp.dir}/dfs/name</value>
  </property>
  <property>
    <name>dfs.data.dir</name>
    <value>${hadoop.tmp.dir}/dfs/data</value>
  </property>
</configuration>
```

NameNode用ファイルの格納場所

DataNode用ファイルの格納場所



完全分散モード

- 完全分散モード　変更点
  ⑤Hadoop設定ファイルの編集

  mapred-site.xml (MapReduceのプロパティ)

```
<configuration>
  <property>
    <name>mapreduce.jobtracker.address</name>
    <value>master:54311</value>
  </property>
  <property>
    <name>mapreduce.cluster.local.dir</name>
    <value>${hadoop.tmp.dir}/mapred</value>
  </property>
</configuration>
```

JobTrackerのアドレス

MapReduceデータの保管場所

完全分散モード

- 完全分散モード　変更点
  ⑥Hadoop設定ファイルの編集

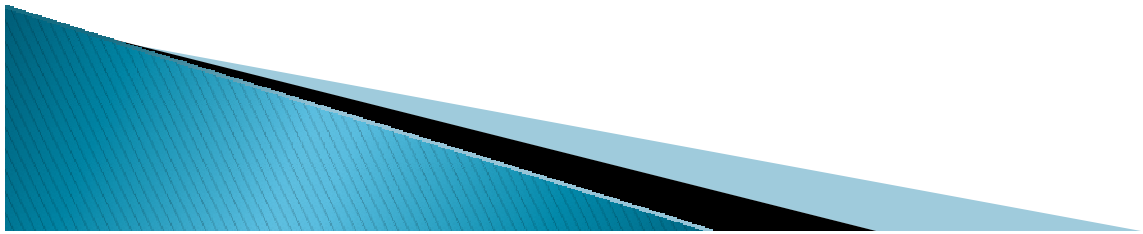
  hadoop-env.sh (Hadoopで使用する環境変数)

  export HADOOP_LOG_DIR=/var/log/hadoop ←ログファイルの保管場所
  export HADOOP_PID_DIR=/var/run/hadoop ←プロセスID
  export HADOOP_IDENT_STRING=sample ←表示されるインターフェース名

  スレーブノードリストの作成

  スレーブサーバのリストファイルにスレーブサーバのホスト名を記述する
  ${HADOOP_HOME}/common/conf/slave
  slave
  slave2

  追加



完全分散モード

- 実行結果

サンプル実行方法

hadoop jar hadoop-mapred-examples-0.21.0.jar サンプル名 引数1 引数2 ・・・

pi (円周率計算)をしてみる

hadoop jar hadoop-mapred-examples-0.21.0.jar pi 10 10000

Master+Slave 1台 ⇒ 50.2秒で実行完了

Master+Slave 2台 ⇒ 31.8秒で実行完了

約1.58倍の性能アップ！！

slaveサーバを増やすことで更に性能アップが見込める

※slaveサーバは異なるスペックで可

完全分散モード

- 実行結果(ノードダウン)

ノードダウン時にそのまま処理が継続されることを検証

①3台繋げた状態でpi (円周率計算)を実行

```
# hadoop jar hadoop-mapred-examples-0.21.0.jar pi 10 10000
```

②slave2をネットワークから分離(LANケーブル抜く)

③masterはノードがダウンしたことを検知して別ノード(ホスト名”slave”)へ作業をふり直す

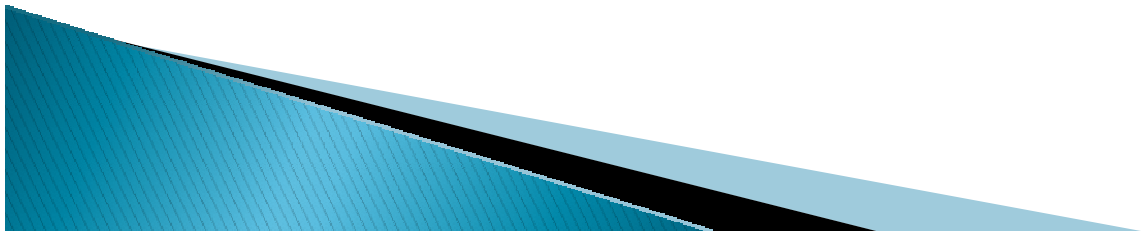
④作業完了

マスターサーバ（Namenode, JobTracker）がノードダウンを検知し、切り離すまでの時間はデフォルトで約10分となっている。

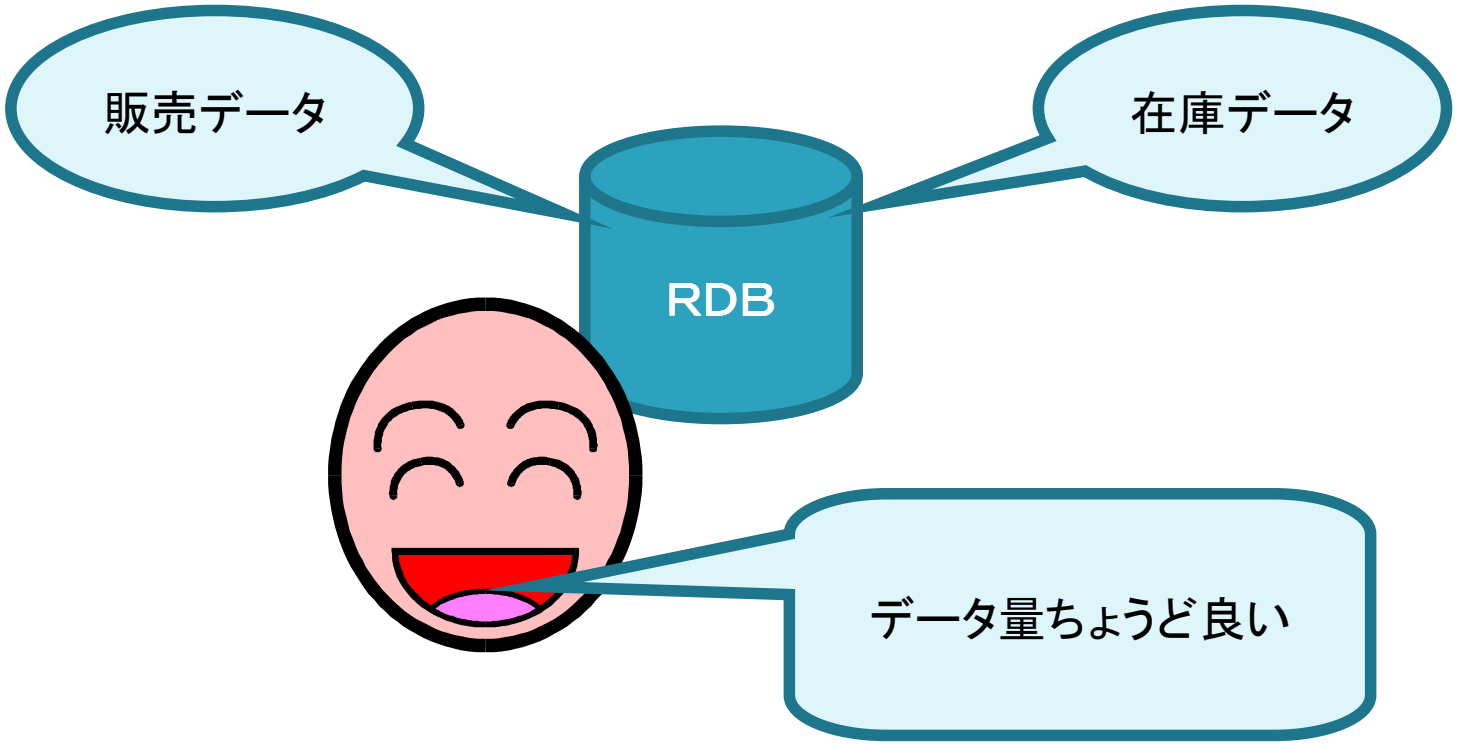
各設定値はcore-site.xmlとmapred-site.xmlで変更可能。

今後この技術はどのように利用されるか。

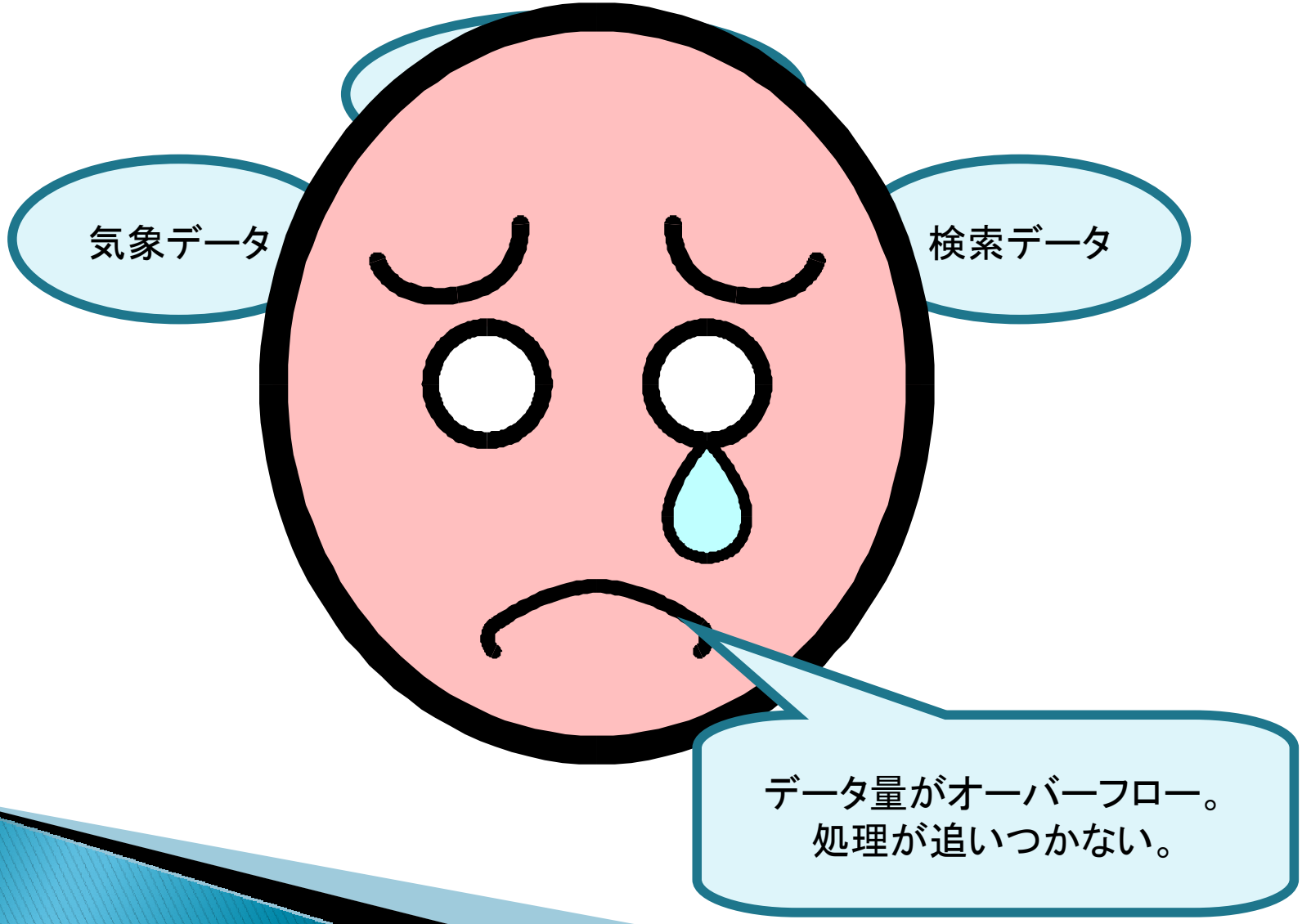
- ビッグデータ



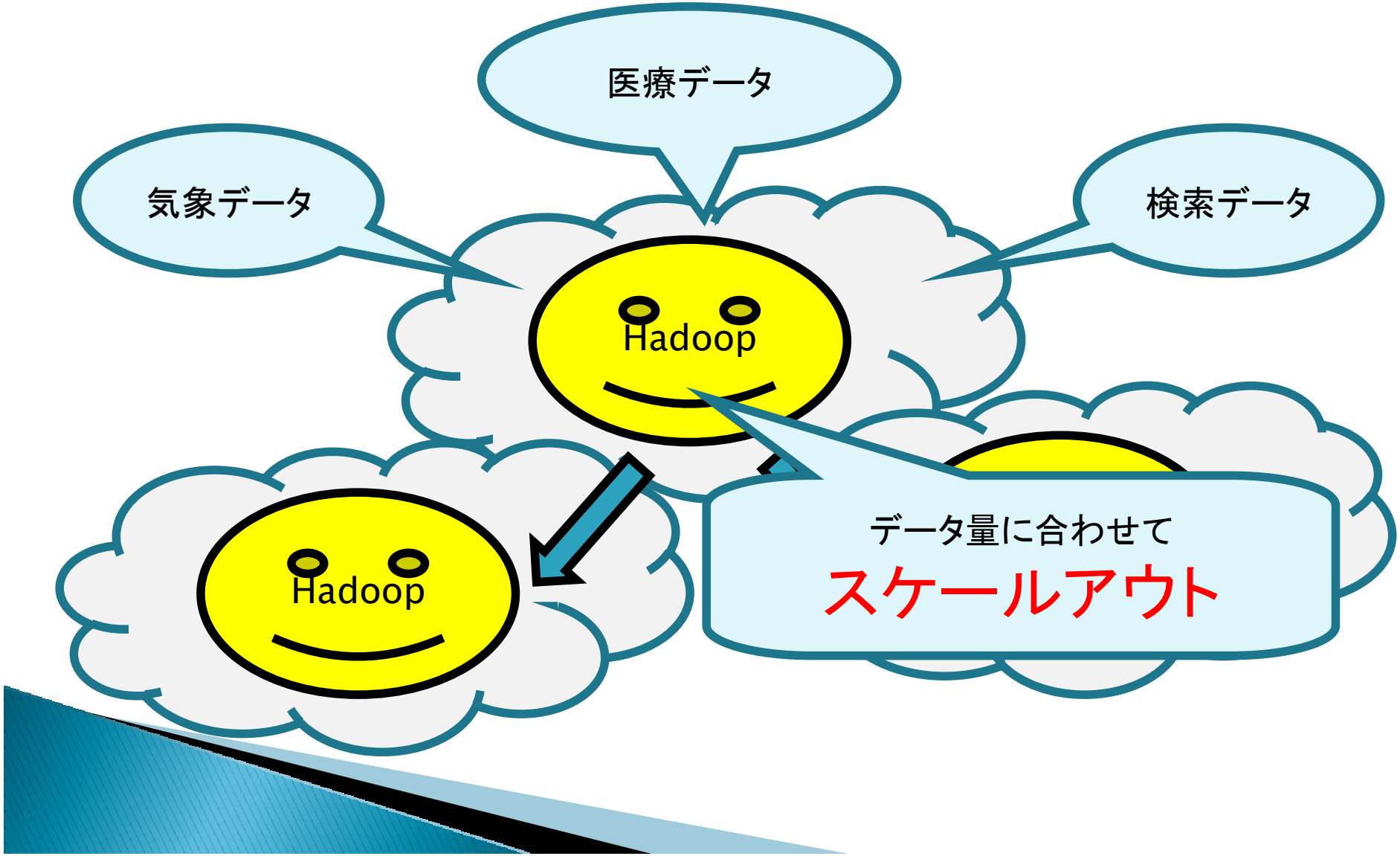
いままでは



最近のニーズ＝＞ビッグデータ

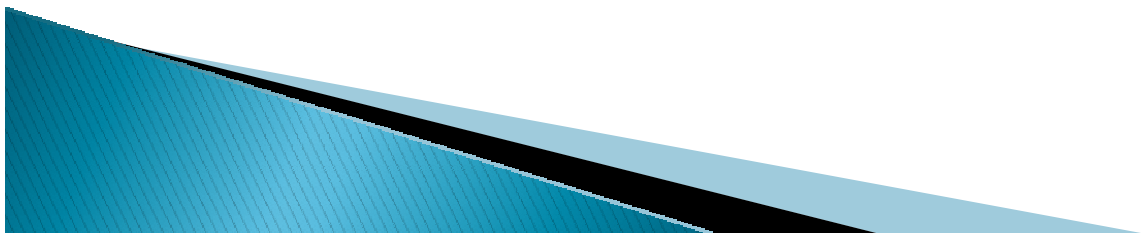


ビッグデータ＝＞Hadoop



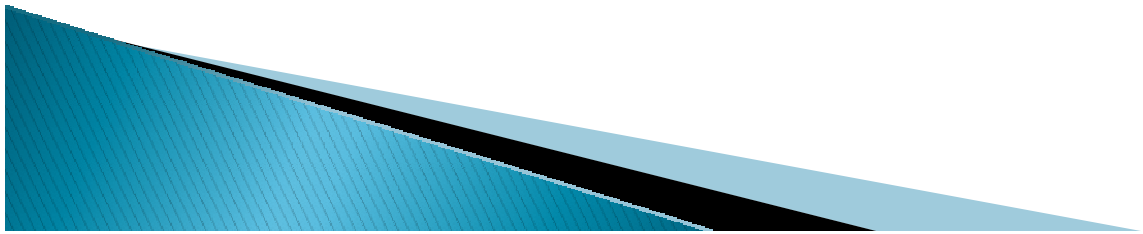
医療データ by Hadoop

- オーダリングや電子カルテをすべてを蓄積する。
- 必要に応じてデータを分析し展開する。
- 障害カルテやナショナルデータベースへの応用に期待。



まとめ

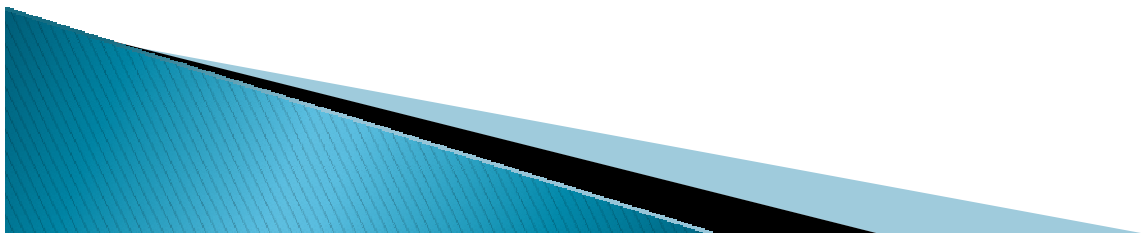
- Googleが2004年8月の一ヶ月間で3ペタバイト以上のデータをMapReduceで処理してから約7年を経過しビッグデータのニーズは年々高まっています。
- 今まで捨てていたさまざまなデータを蓄積し分析展開する事が比較的簡単に実現できるようになります。
- 最近ではHadoopと連携可能なデータベースの登場によってより身近になっています。
- これからのシステムはRDB＋Hadoop等の分散DBで構築する事になっていくでしょう。



Hadoop メジャーリリース

- Hadoop 1.0.0で大きく3つの機能が追加(2011年12月)
  - ・ネットワーク認証プロトコル「Kerberos」によりネットワーク全体にわたってセキュリティを確保
  - ・「WebHDFS RESTful API」によりWebの技術を使って操作することが可能
  - ・「Apache HBase」がフルに利用可能

- 次期メジャーリリース(2012年中頃を予定)
  - ・HDFS（ファイルシステム）の高可用性－ファイルシステムのSPOF（単一障害点）問題を解決
  - ・HDFS Federationにより少なくとも4～5倍、FS ネームシステムを拡大して、より大きなクラスタを構築可能
  - ・次世代MapReduce(YARN)によってより大きなクラスタをサポート可能



参考文献

- 太田 一樹, 下垣 徹, 山下 真一, 猿田 浩輔, 藤井 達朗(2011)『 Hadoop徹底入門』株式会社　翔泳社
- Hadoop公式サイト(http://hadoop.apache.org/)
- Tom White, 玉川竜司, 兼田聖士『Hadoop』
  株式会社オライリー・ジャパン
- Hadoop～Yahoo!JAPANの活用について～
(http://www.slideshare.net/techblogyahoo/hadoopyahoo-japan-5204073)