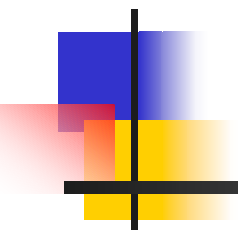
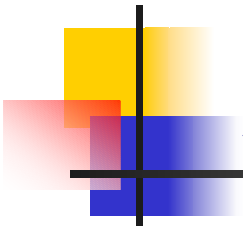


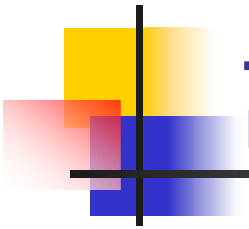
MySQLによる ミッションクリティカルサービスの 設計と運用

飯坂　剛一　/ Office ISK
渡辺　隆志　/（株）Infoseek



進行予定

- 自己紹介と項目説明　3分
  - 「場」になれるための時間。リラックス～
- MySQLとInfoseek 3分
- 設計関連　12分
- 運用関連　15分
- チューニング関連　12分
- 質疑応答　5分

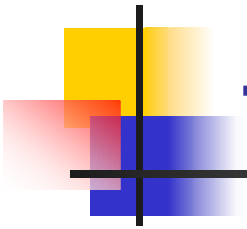


設計関連（TOC）

- ソフトウェア構造
- Barista　/ Tea
- Java からのMySQLアクセス mm.mysql
- Auto Reconnectionの設定方法
- Connection Poolingについて
  - DBConnectionBroaker
- DB構造
- ロールバック

運用関連（TOC）

- 機材選定
- OSの選択
- クラスタ
- 設備投資の内訳
- 機材構成概要
- バックアップ
- アクセス集中について



チューニング（TOC）

- Linux
- Java
- MySQL

MySQL と infoseek

- プロフィール
- メール
- ニュース検索
- 不動産検索
- スクール検索

MySQLを使った理由

- MySQLの長所にある
  - コスト
  - サーバサイドでのData Replication
  - ハイパフォーマンス （速い）
  - 保守がしやすい

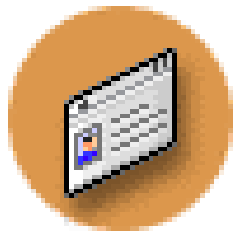
MySQLに不満はある？

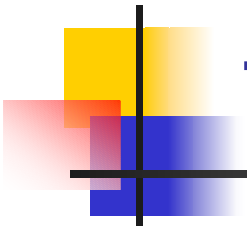
- 短所もやはり…
  - Cursor や View などがなくSQL仕様で劣る
    - 結局は使い方で回避できる
    - それを実装することで遅くなるのなら不要！
    - Larry Wallも言っている

      “There’s More Than One Way To Do It!”
  - Raw I/O ができない
    - やっぱりORACLEは良い

infoseekではどうしたか

- こうしたことをふまえて、
  プロフィールを例にとり、
  どう対応していったかを話そう



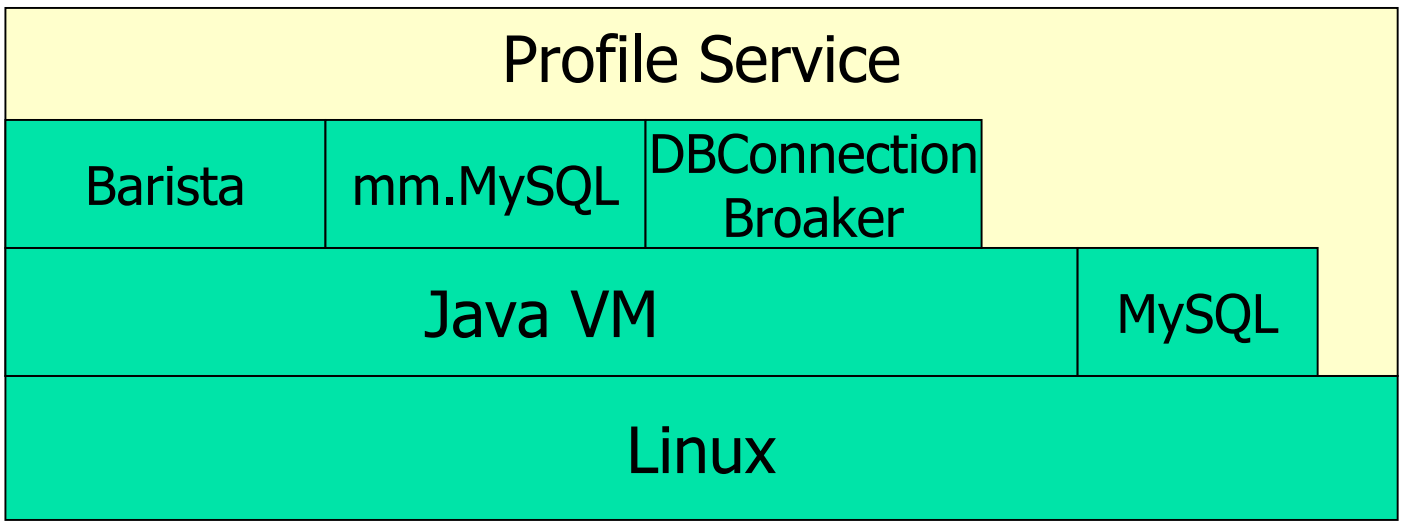


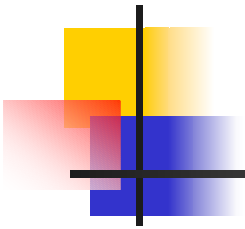
設計関連（TOC）

- ソフトウェア構造
- Barista / Tea
- Java からのMySQLアクセス mm.mysql
- Auto Reconnectionの設定方法
- Connection Poolingについて
  - DBConnectionBroaker
- DB構造
- ロールバック

ソフトウェア構造

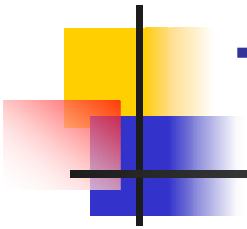
- プロフィールサービス





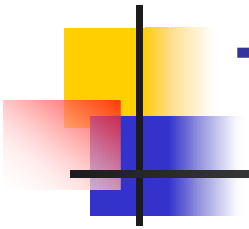
Barista

- WDIG(旧Go.com)が開発した、Pure Java Servlet Engine (Servlet 2.3 対応)
- 強力なLog機能（Trove log API ）
  - http://opensource.go.com/
- 高パフォーマンス
- Servlet EngineごとIDE上で起動できるので、Servletの開発が容易
- 残念な事に、外部には非公開。



Tea (1)

- WDIG(旧Go.com)が開発した、JavaVM用Template言語
- ESPN.com, Movies.com, ABCNews.com and Disney.com.等で使用されている。
- 現在は、オープンソースで配布
  - http://opensource.go.com/
- O'REILLY Java Servlet Programming 2nd Editionで、一章を割いて解説されている。



Tea (2)

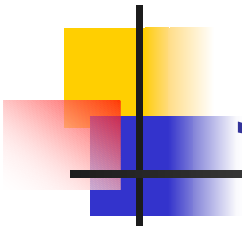
- 特徴
  - 平易な文法
  - JAVA BEANSにより、データを分離できる
  - class ファイルへ、直接Compileされる
  - IDEが用意されている（日本語では不具合あり）
  - Servlet以外にも、汎用のTemplate言語として使用可能

Tea (3)

- コード例

```
<% template HelloSample() *>
<HTML><BODY>
<%
request = getRequest(“EUC_JP”)
myName = request.parameters(“name”) %>

Hello <%myName%> <HR>
<%
if myName == “OSDN”{
        “Nice !!(^^)”
}
%>
</BODY></HTML>
```



Java からのMySQLアクセス

- MM.MySQL
  - GLPLな、Type4 (Pure Java) JDBC1/2 Driver
  - 入手先
    - http://mmmysql.sourceforge.net/

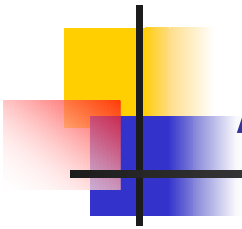
    *最新版は、このページには書いていないので注意

MM.MySQLの設定方法

- 設定方法
  - 配布アーカイブ内の、README内に記述
  - 日本語を使う際には
    - useUnicode
    - characterEncoding

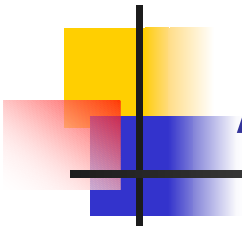
    の指定が重要
- JDBC URLの例

  jdbc:mysql://dbhost:3306/test?user=isj&password=jsi&useUnicode=true&characterEncoding=EUC_JP



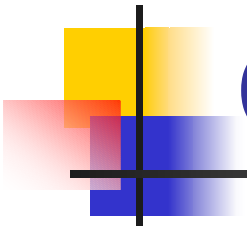
Auto Reconnection機能

- MM.MySQLには、Auto Reconnection機能がある。
- Statementを実行する際などに、Connectionが切れていた場合、自動的に、Connectionの再接続を行う。



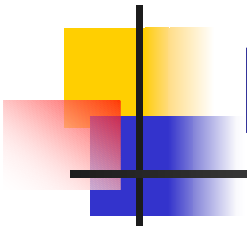
Auto Reconnectionの設定方法

- JDBC URL パラメータで指定
  - autoReconnect
    - Auto Reconnection を行うかどうか (true or false)
  - maxReconnects
    - 何回Reconnectを試行するか
  - initialTimeout
    - Reconnectを試行する間隔 [sec]



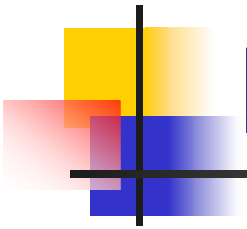
Connection Pooling

- Infoseek では、DbConnectionBroker (http://javaexchange.com/) を使用。
- MM.MySQL単体でも、ver 2.0.9より PooledConnectionDataSource で、ConnectionPoolingを行うことが出来る。(Infoseekでは、使用を検討中)



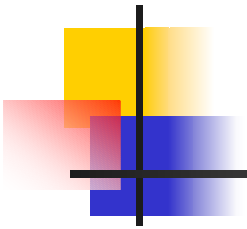
DBConnectionBroaker

- 汎用の、Connection Poolingライブラリ
- 長所
  - フリーである
  - 設定が簡単
  - Auto Reconnection機能
- 欠点
  - JDBC 2.0 Optional Packageとは異なるAPI



DB構造

- KEYの設定は重要
  - パフォーマンスに非常に影響する
- デフォルト値を利用する
- Mod Timeには特性（癖）がある

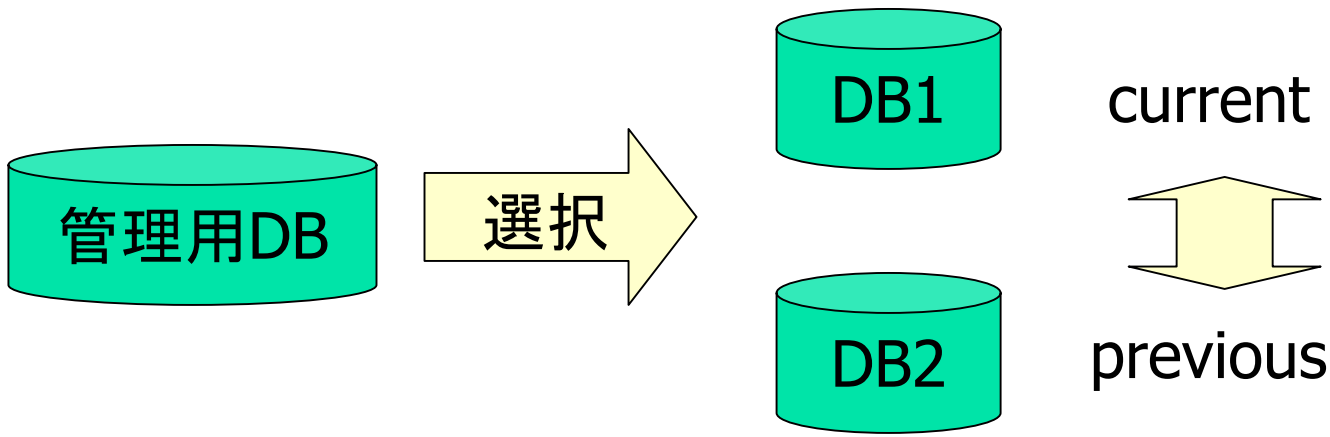


ロールバック (1)

- プロフィールではやっていないが…
- データ更新を意識して利用すべき
  - 元データが「いつも正しい」とは限らない
  - ダンプによりデータを戻すのは非現実的

ロールバック (2)

- 同じ構造のDBを複数定義

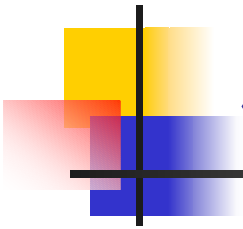


設計で気をくばった点

- プロフィールサービスについて
  - JOINを避けるために、一覧表示に必要なデータは、On MemoryでCache
  - Lockを減らす為に、頻繁に追加、更新がかかるデータ(Counter, FootPrint等)は別Tableに移す
  - 不必要なデータは、On timeでなく、Off timeでBatch処理により削除

運用関連（TOC）

- 機材選定
- OSの選択
- クラスタ
- 設備投資の内訳
- 機材構成概要
- バックアップ
- アクセス集中について



機材選定 (1)

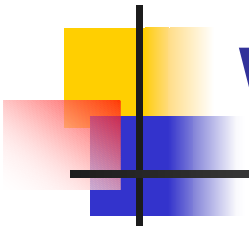
- 機材選定はとても重要
- 速いものは良い
- 速くても不安定は困る
- パフォーマンス VS コスト
- 誰も保守できないのは困る
- 誰もが知っているのも困る

機材選定 (2)

- マシンが落ちると信用も落ちる
  - 常時接続の普及
  - サービスはますます落とせない
- 連続稼動率
  - １年で30分のサービス停止しかないとき？
  - 1年＝525,600分
  - 99.9942%

OSの選択

- SysAdmin　２００１年７号より
- メール処理能力は:
  - Linux > Solaris > FreeBSD > Windows2000
- ファイルのリード・ライトでの処理能力
  - Linux > Windows2000 > FreeBSD > Solaris



Web ApplicationとLinux

- 低い投資コスト
- 非常に相性がよい
- パフォーマンスが良い
- セキュリティには気を配る必要あり
  - 必要なデーモンのみ起動
  - プライベートネットワークに押し込む

並列Cluster System (1)

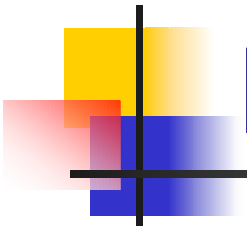
- 長所
  - スケラービリティー
  - ハイパフォーマンス
  - 冗長性が高い
- 短所
  - 占有スペース
  - 保守が煩雑
  - 性能を引き出すためにはMPI

並列Cluster System (2)

- Linuxもなかなかやる
  - Clusters@Top500

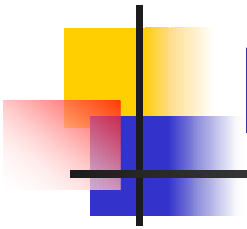
    http://clusters.top500.org/
- Dolphin Interconnect LLC社
  - WulKit3
- SCALI社：ScaMPI
  - ノード間300MB/sec超のインターコネクト

HA Cluster System (1)



- HA = High Availability
  - フォルトトレーランス
- High Availability Linux Project
  - http://linux-ha.org/

HA Cluster System (2)



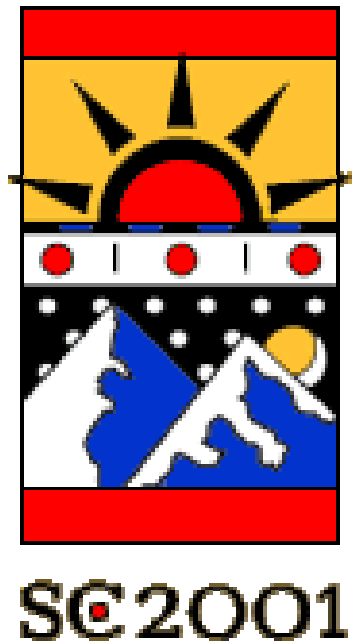
- Open Sources
  - Ultra Monkey
  - LVS
- コマーシャルプロダクト
  - SteelEye: LifeKeeper
  - Mission Critical Linux: Convolo
  - SGI： FailSafe
  - SUN: Sun Enterprise Cluster
  - Fujitsu: SafeCluster

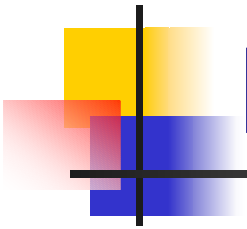
Blade Cluster System

- 収容性
  - 32 Server in 3U
- 保守性
  - 交換が容易
  - ダイナミックリソース管理
- 運用コスト
  - 低い電源消費量

Blade Cluster System

- SC2001でも存在感
  - これからの大きな流れに？
- 増えるベンダー
  - HP
  - Compaq
  - NEXCOM
  - ・・・・・





Load Balancing (1)

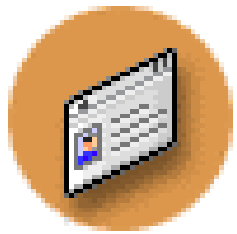
- ロードバランスを行う利点
  - 負荷分散
  - 冗長性
  - 保守性
  - スケーラビリティー
  - 段階的な設備投資

Load Balancing (2)

- ロードバランスでの欠点は？
  - 追加機材が必要
    - 投資コストの増大
    - 保守管理の対象が増加　＝　運用コスト大

infoseekではどうしたか

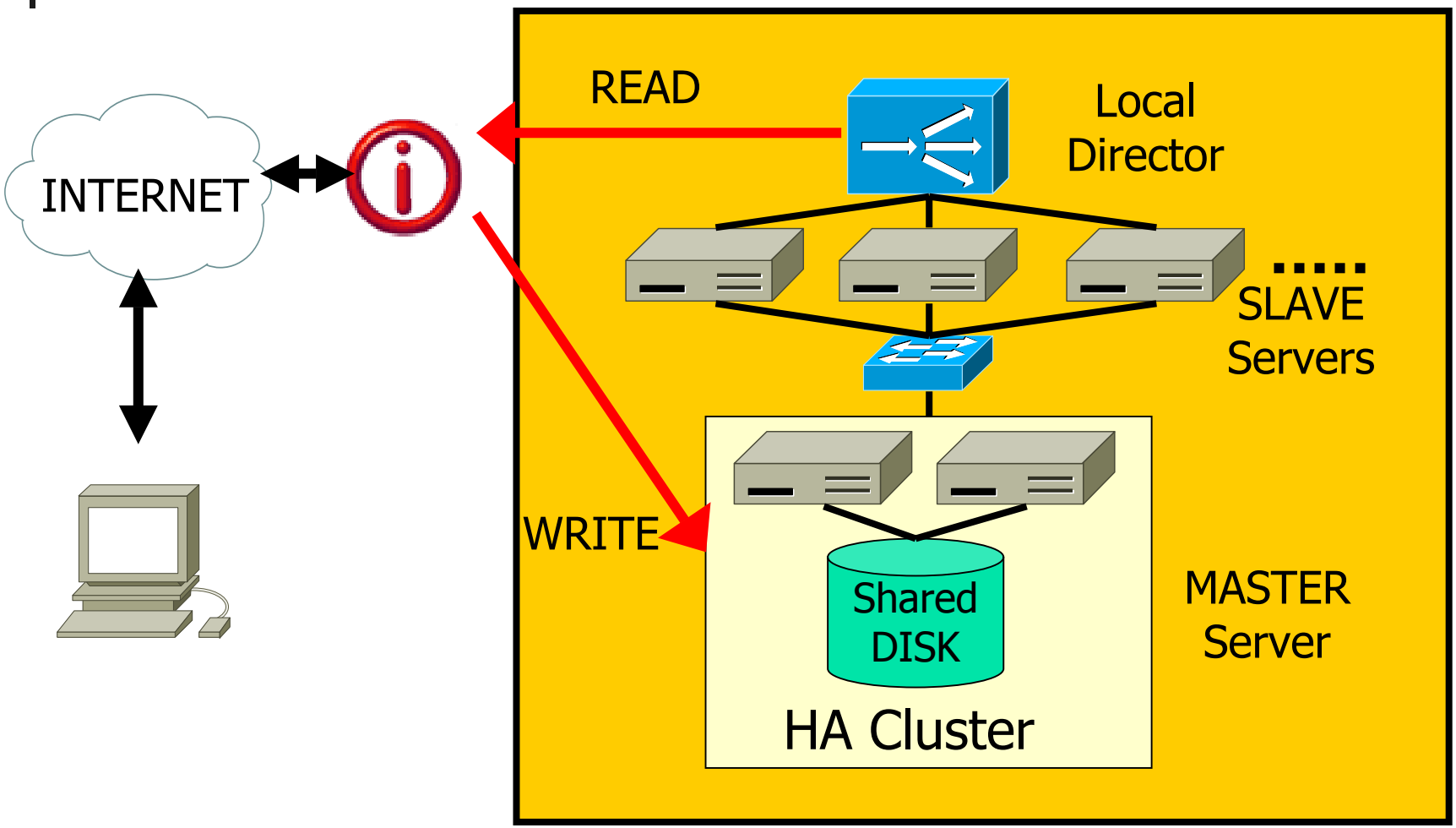
- こうしたことをふまえて、
  プロフィールを例にとり、
  どう対応していったかを話そう



設備投資の内訳

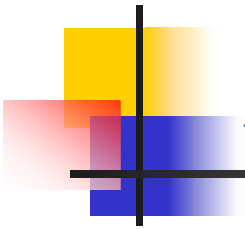
- 設備投資額
  - HW + SW + 保守 + 運営 = TotalCost
- 陳腐化の速度
  - HW > SW
- リソース管理
  - 需要予測と設備投資との同調
- 最高性能は不要
  - 現実的な性能をターゲット

機材構成概要


READ
Local
Director
INTERNET
SLAVE
Servers
WRITE
Shared
DISK
MASTER
Server
HA Cluster

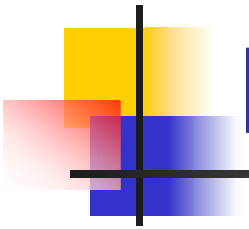
LifeKeeper

- LifeKeeper
  - ダウンロードして評価できる
  - サポートOSが多い
    - Linux, WindowsNT/2000, Solaris(Intel)
  - サポートアプリケーションが多い
  - 投資コストが低い
  - 日本での代理店
    - テンアートニー
      http://www.10artni.co.jp



バックアップ

- バックアップの対象
  - 機材
  - データ
- MySQLのData Replicationを利用



MySQLのData Replication (1)

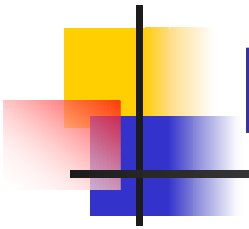
- マスター側の設定 (my.cnf)

  [mysqld]

  log-bin=/var/spool/mysql/profilebinlog

  server-id=xxxx

  bin-log=on



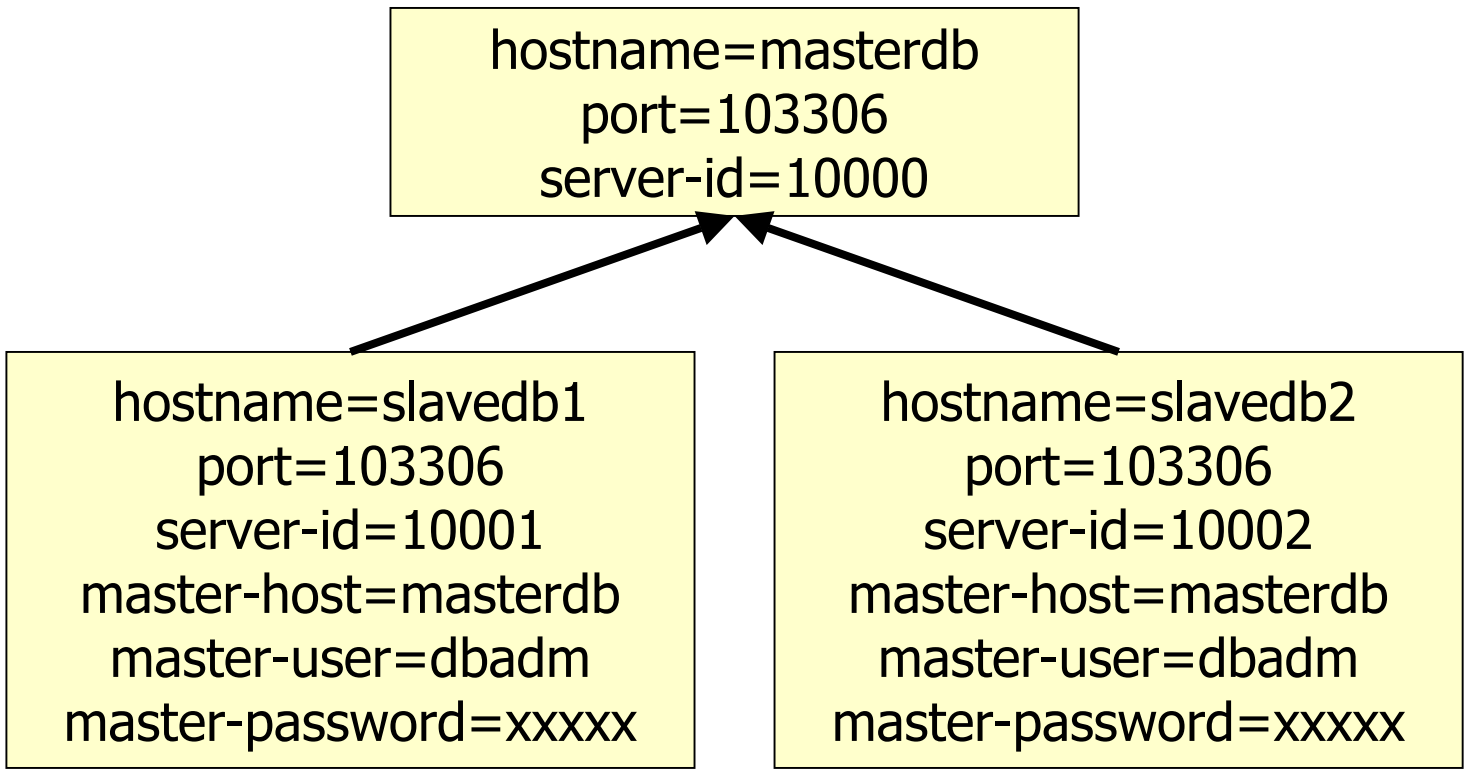
MySQLのData Replication (2)

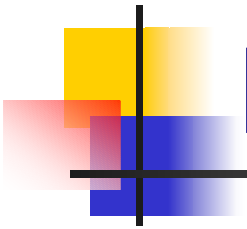
- スレーブ側の設定 (my.cnf)

[mysqld]
server-id=xxxx
master-host=マスターサーバのホスト名
master-user=Replicationを行うMySQLユーザ名
master-password=Master-userのパスワード
master-port=マスターサーバのポート番号

MySQLのData Replication (3)

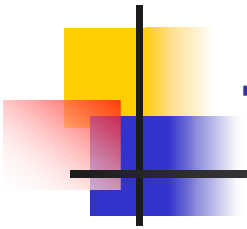
- 設定例





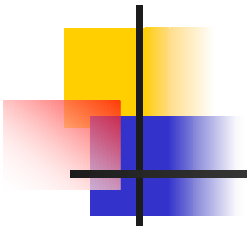
MySQLのData Replication（4）

- 注意すべき点
  - Replication開始時点でマスターとスレーブのDBの内容を同期させる必要あり
  - Bin-logのケアが必要
    - max-binlog-size で設定した値でローテーション
    - ローテーションさせたくない場合はlog-binで指定するファイル名に拡張子以降を記述しない
    - 1KB < max-binlog-size < 1GB （1GB default)
  - RAND()は同じにならない



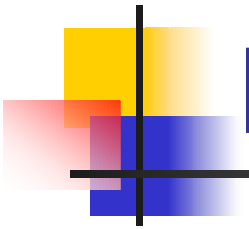
チューニング（TOC）

- Linux
- Java
- MySQL



Linuxをチューニング

- Mountの方法
- ファイルシステム
  - tune2fs
  - ジャーナルファイルシステム



Mount (1)

- /etc/fstab にDISK Labelを書かない
  - 便利ではあるが該当するDISKを探すことに
  - 起動の際の時間短縮

Mount (2)

- /etc/fstabにDISKラベルを書いた例

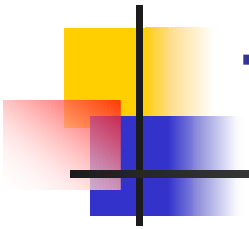
```
LABEL=/          /           ext3    defaults          1 1
none             /dev/pts    devpts  gid=5,mode=620    0 0
LABEL=/home      /home       ext3    defaults          1 2
LABEL=/opt       /opt        ext3    defaults          1 2
none             /proc       proc    defaults          0 0
none             /dev/shm    tmpfs   defaults          0 0
LABEL=/tmp       /tmp        ext3    defaults          1 2
LABEL=/var       /var        ext3    defaults          1 2
/dev/hda3        swap        swap    defaults          0 0
/dev/cdrom       /mnt/cdrom  iso9660 noauto,owner,kudzu,ro 0 0
/dev/cdrom1      /mnt/cdrom1 iso9660 noauto,owner,kudzu,ro 0 0
/dev/fd0         /mnt/floppy auto    noauto,owner,kudzu 0 0
```

ファイルシステム (1)

- Maximum mount counts
  - ext2などではパーティションやDISKが複数マウントしているときは、それぞれのこの値を異なる数値に変える
  - 起動時のfsckの時間短縮

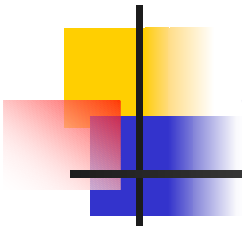
```
# tune2fs -l  /dev/hda1 | grep -i Mount
Last mounted on:          <not available>
Last mount time:          Tue Feb  5 11:48:05 2002
Mount count:              4
Maximum mount count:      -1

# tune2fs –c 7 /dev/hda1
```



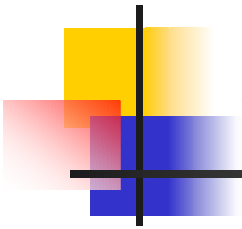
ファイルシステム (2)

- ジャーナルファイルシステム
  - Write Fast, Read Slow
  - fsckが不要
  - RedHat 7.2ではext3が標準
  - SGIのXFSも“使える”
    - http://www.oss.sgi.com/projects/xfs/



JavaのVMをチューニング(1)

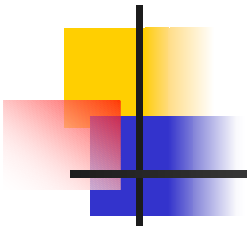
- JavaVMの起動パラメタ
  - メモリ割り当てプールの設定は必須
    - サーバアプリケーションにおいては、デフォルトの最大値である64MBでは、小さすぎる。



JavaのVMをチューニング(2)

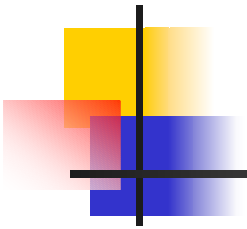
- JavaVMの起動パラメタ
  - インクリメンタルガーベジコレクタの使用の是非、一概には言えないので、実際に試してから決定するのが良い。

    Infoseekでは、
    - 使用：ニュース検索, 株価検索　など
    - 未使用：プロフィール など



MySQLをチューニング (1)

- MySQLのパフォーマンスに依存するもの
  - CPU性能
  - メモリサイズ
  - DISK性能（転送速度）
  - mysqldのオプションパラメタ



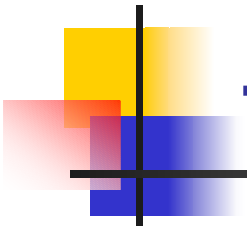
MySQLをチューニング (2)

- メモリに関するもの
  - key_buffer_size
  - table_cache
  - record_buffer
  - sort_buffer

MySQLのチューニングを評価

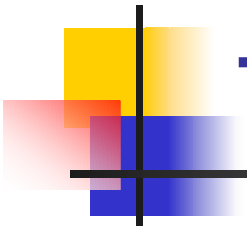
- SHOW STATUSでヒット率を評価
  - 100-((Key_reads/Key_read_requests)*100)
  - 90以上が目標

```
mysql> SHOW STATUS LIKE 'Key%';
+-------------------------+-------------------------+
| Variable_name           | Value                   |
+-------------------------+-------------------------+
| Key_blocks_used         | 7793                    |
| Key_read_requests       | 1589396597              |
| Key_reads               | 356247                  |
| Key_write_requests      | 6675835                 |
| Key_writes              | 6502631                 |
+-------------------------+-------------------------+
```



チューニングでの注意点

- テストサーバで十分評価する
  - MySQL 3.x ではデーモンの再起動が必要
    - サービスが停止することになる
  - MySQL 4.x ではダイナミックに変更できる
- １度にいくつもパラメタを変更しない
- パラメタが与える影響を確かめる

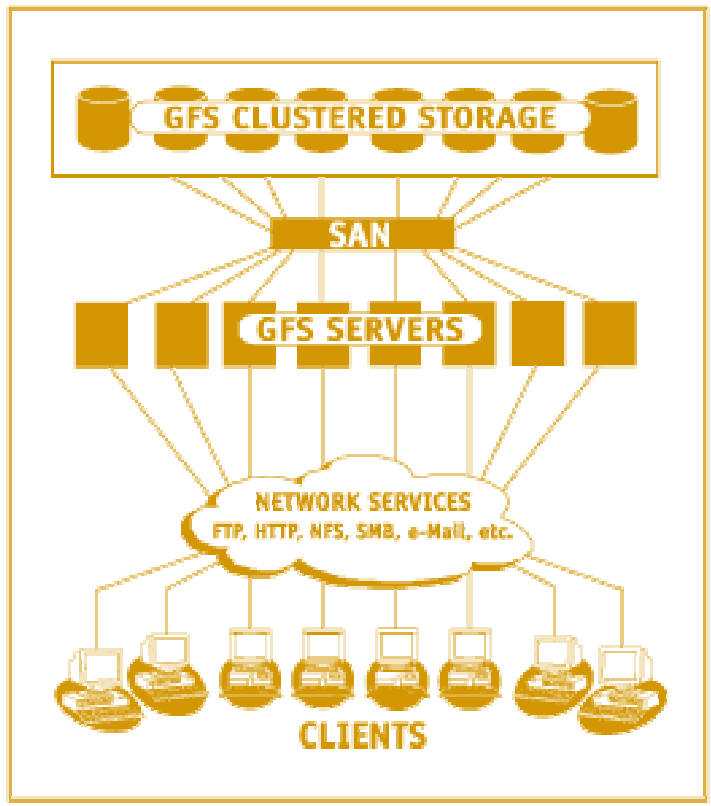


TODO

- SAN
- MySQL 4.x

SAN(Storage Area Network)

- GFS: Global File System
  - Open Source (だった)
  - DISK増強が容易
- ミネソタ大学が母体
  - Sistina Software
    1313 Fifth Street Southeast
    Suite 111
    Minneapolis, MN 55414



參考資料 (1)

- Which OS is Fastest for High-Performance Network Applications?

  Jeffrey B. Rothman and John Buckman

  SysAdmin - July 2001 Volume 10 Number 07

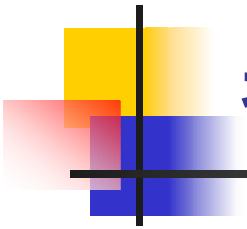
  http://www.samag.com/documents/s=1148/sam0107a/0107a.htm

- Which OS is Fastest -- FreeBSD Follow-Up

  Jeffrey Rothman and John Buckman

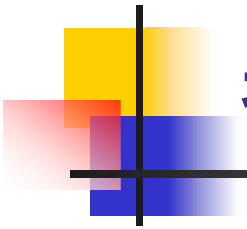
  SysAdmin - August 2001 Volume 10 Number 8

  http://www.samag.com/documents/s=1147/sam0108q/0108q.htm



參考資料 (2)

- High Availability Linux Project:
  http://linux-ha.org/
- Dolphin Interconnect LLC
  - http://www.dolphinics.com/
- SCALI
  - http://www.scali.com/
- SteelEye : LifeKeeper
  - http://www.steeleye.com/
- SC2001
  - http://www.sc2001.org/



参考資料 (3)

- MySQL Manual:

  Tuning Server Parameters

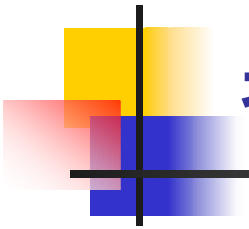
  http://www.mysql.com/doc/S/e/Server_parameters.html
- MySQL Manual:

  SHOW VARIABLES

  http://www.mysql.com/doc/S/H/SHOW_VARIABLES.html
- MySQL Manual:

  SHOW STATUS

  http://www.mysql.com/doc/S/H/SHOW_STATUS.html



參考資料 (4)

- mytop:

  A “TOP” Clone for MySQL

  http://public.yahoo.com/~jzawodn/mytop