

株を買うなら、  
三愛UFJフィナンシャル・グループの  
kabu.com証券。

自動売買

シニア割引

子供株

ノーロードファンド

フルオープン系システム基盤による優位性と展開  
～オンメモリDBを生かしたサービス向上～

**kabu.com**  
カブドットコム証券株式会社

## システム構築の考察ポイント

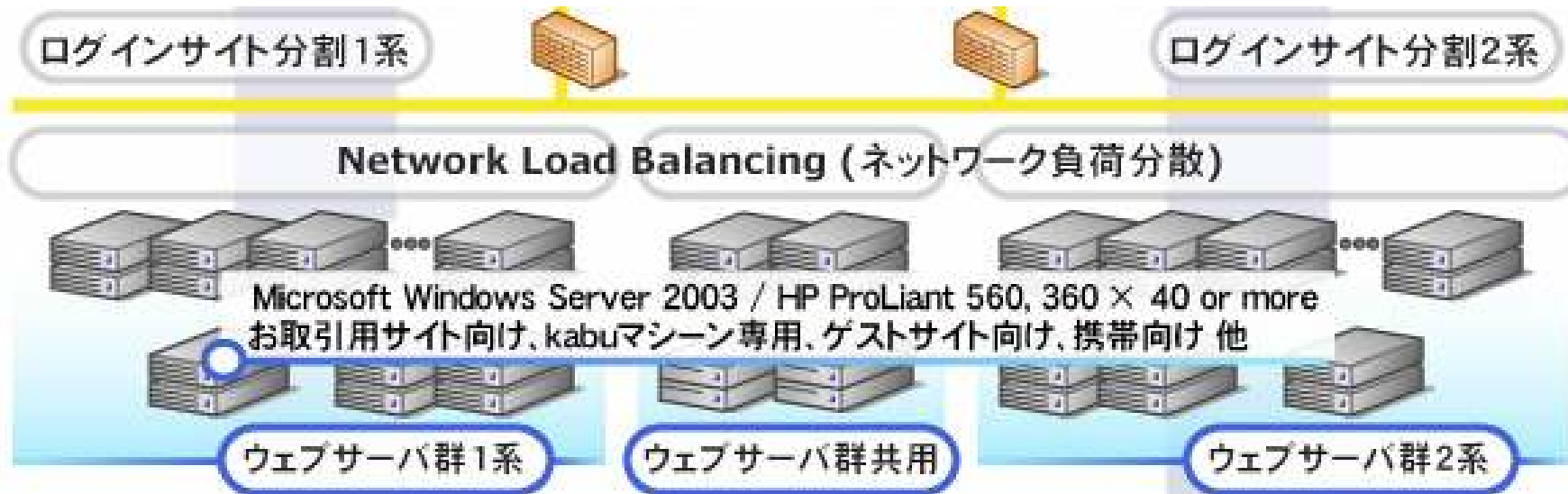
### ■ システムアーキテクチャ

運用のしやすさ、システム全体を止める事なくメンテナンスが可能である事、負荷対策が柔軟に行える事を念頭にアーキテクチャを決定する。



投資と同様、リスク管理が必要

# システム構築の考察ポイント ~Webサーバ



## ■ Webサーバ

1. 負荷分散装置かソフトウェア分散か

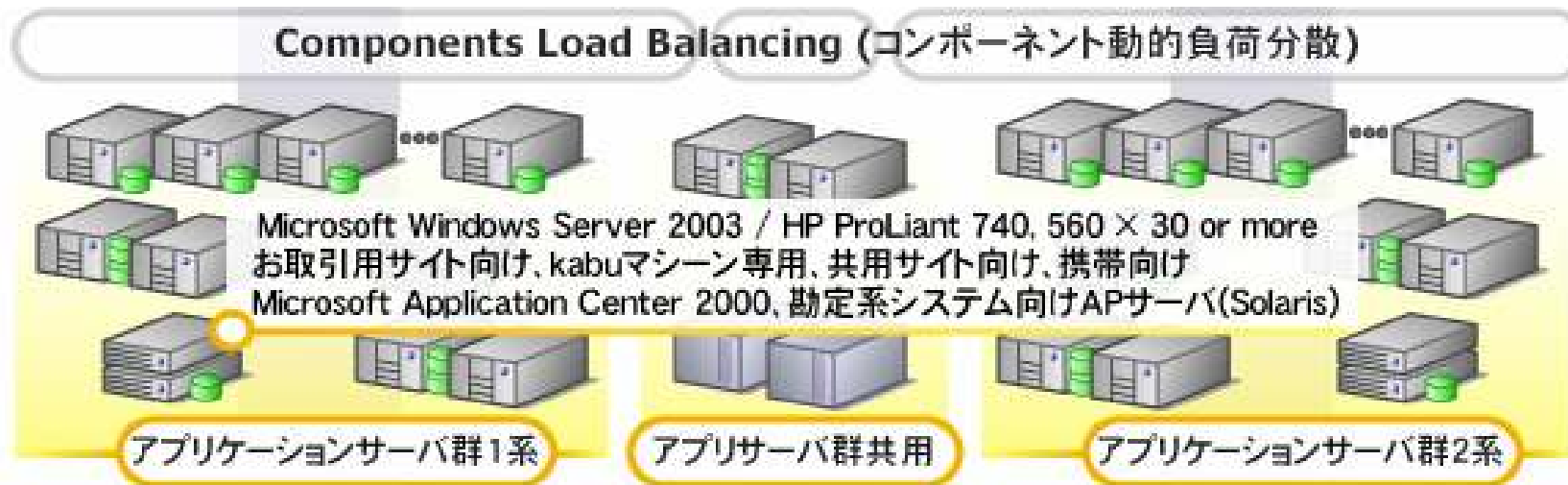
→ 当社ではIIS+Windows NLBを利用しスケールアウト

2. フロント開発者は業務の理解が無くても良い。

→ ビジネスロジックは全てアプリケーションサーバへ(3階層)

→ ActiveServerPagesにおけるフロントデザインへ集中する事による  
圧倒的な開発生産性の向上

# システム構築の考察ポイント ～アプリケーションサーバ



## ■ アプリケーションサーバ

1. 同一構成のハードウェアでスケールアウト  
 サービスパックが部分的に適用しやすいというメリットも。
2. 優秀な外部のコンポーネントを導入する。  
 → **KairosメモリDBを導入**
3. チャンネルに影響されない純粋なビジネスロジックを記述

# システム構築の考察ポイント ～DBサーバ



Microsoft Windows Server 2003 Datacenter Edition 64 Bit ver.  
HP Integrity Superdome 128CPU / UNYSYS ES7000/40 32CPU  
Microsoft SQL Server 2000 64 Bit ver (注文・顧客) / Oracle 9i RAC (勘定)

注文・勘定系データベースサーバ1系、2系 / 顧客データベース

## ■ DBサーバ

基本的にスケールアップの考えだが、当社はスケールアウトとの組み合わせ。CPU増強の限界に対するリスクを考える。

→半導体Disk等を利用する事も検討

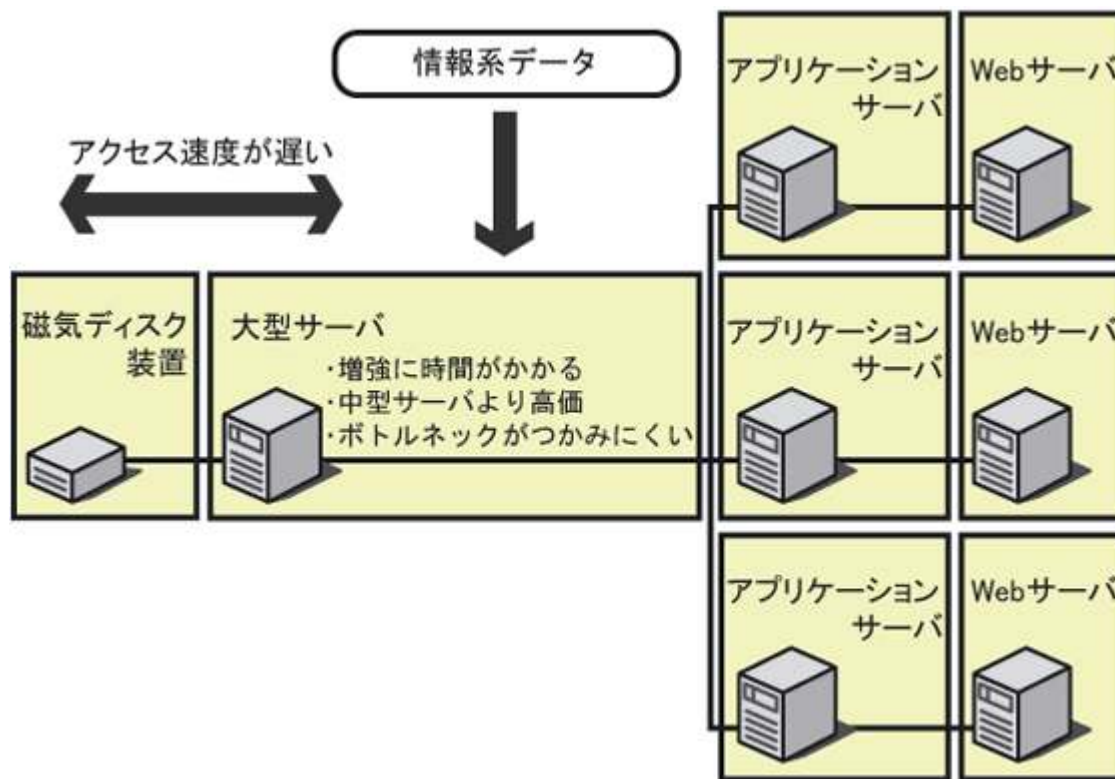
→データがユニークではないモノ(株価等)はフロント～アプリ層へ

## R&D、他社との差別化

- 最新テクノロジーや優秀なパッケージの長所を活かす
- 他社が簡単に真似の出来ないものを構築する
- ROIを意識する(後々無駄にならない)
- 製品のロードマップを重要視する
- コストパフォーマンスには開発/運用の容易さも含まれる

# KairosメモリDBの導入（導入前）

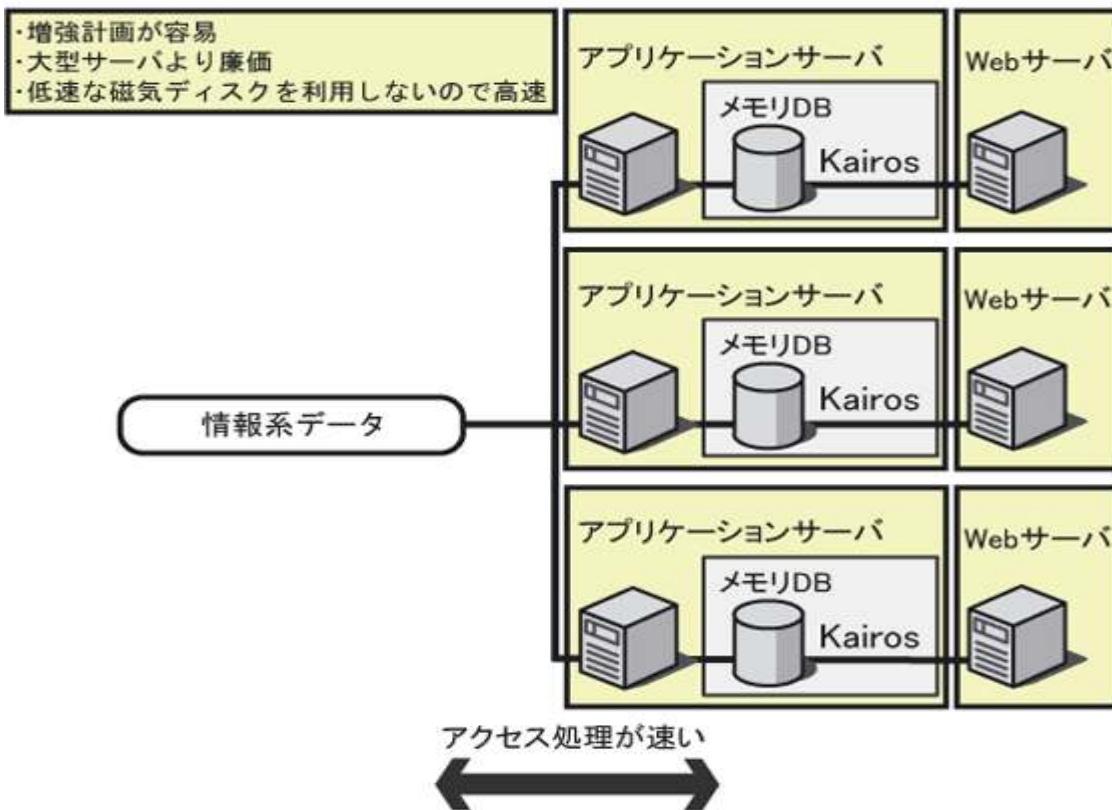
## <メモリDB導入前>



株価というデータ性質上、1箇所でのDBで管理する必要が無い

# KairosメモリDBの導入（導入後）

## <メモリDB導入後>



・ハードウェア費用削減、障害に対する  
リスク分散が実現

# SQLServerとのデータ同期

		構成	説明
MSSQL only		<p>パブリッシャ</p> <p>ディストリビュータ</p> <p>サブスクリバ</p> <p>パブリケーションデータベース トランザクションログ</p> <p>ログリーダーエージェント ディストリビューションデータベース</p> <p>ディストリビューションエージェント サブスクリプションデータベース</p>	<ul style="list-style-type: none"> <li>・パブリッシャ、サブスクリバ方式</li> <li>・複製元テーブルには複製情報管理のための項目追加が必要</li> <li>・複製元データベースのトランザクションログを参照して複製を行う</li> <li>・パブリッシャ/ディストリビューション側に障害が発生した場合の復旧方法が複雑</li> </ul>
		<p>MSSQL</p> <p>Ksync</p> <p>Kairos</p> <p>複製対象テーブル 差分情報テーブル</p> <p>差分情報取込スレッド 差分情報データベース</p> <p>差分情報取込スレッド Kairosデータベース</p>	<ul style="list-style-type: none"> <li>・パブリッシャ、サブスクリバ方式</li> <li>・複製元テーブルには複製情報管理のための項目追加が不要 別途差分情報テーブルが生成される</li> <li>・差分情報は複製対象テーブルに更新トリガが設定され取得される</li> <li>・障害復旧方法は複製対象テーブルを完全同期させるコマンドを提供 復旧手順がシンプル</li> </ul>

# VisualBasicからADOを介したKairosの使い方

```
'ObjectContext
Dim objCtx As Object
Set objCtx = GetObjectContext()

'時価DB接続用変数
Dim adoCnn As New ADODB.Connection
Dim adoRst As New ADODB.Recordset
Dim strCnnStr As String
Dim strSQL As String
Dim objDic As clsDatabaseDictionary

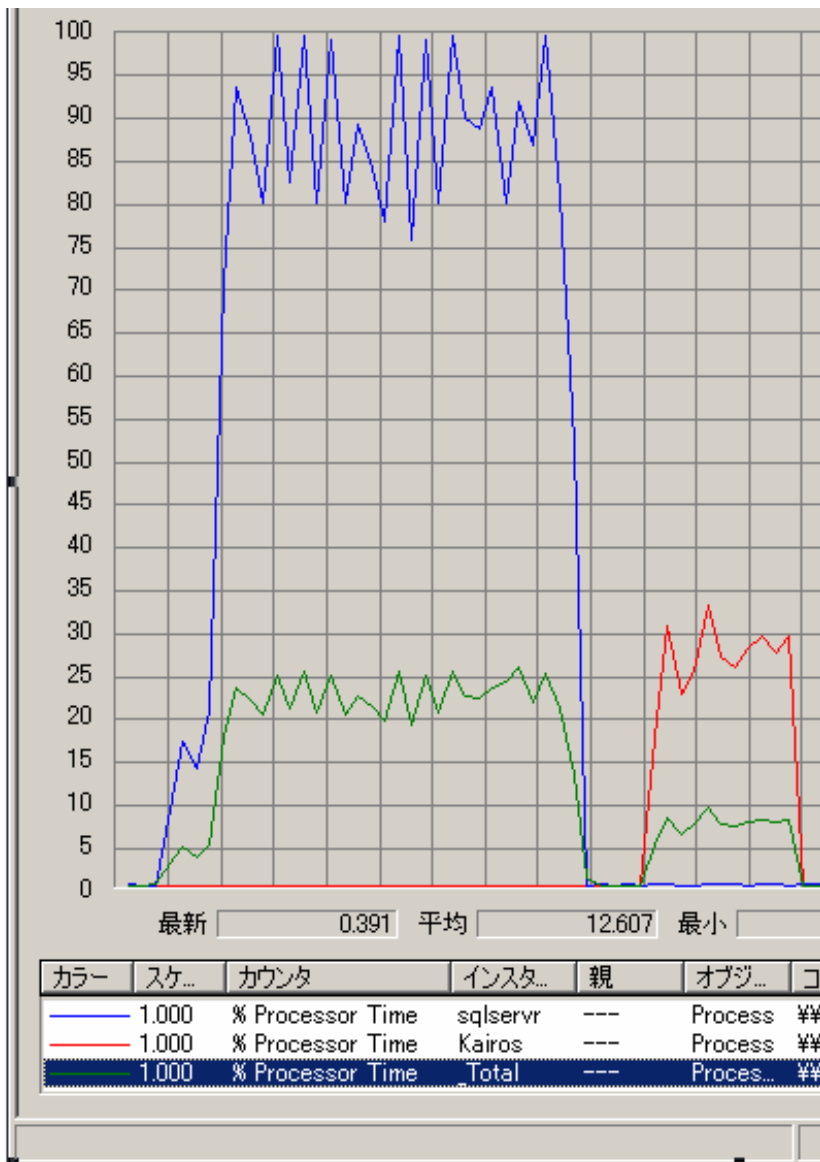
strSQL = "SELECT * FROM VW_StockQuote WHERE cdQuote in(" & strQuoteList & ")"

'DB接続文字列取得
Set objDic = objCtx.CreateInstance("Util.clsDatabaseDictionary")
strCnnStr = objDic.GetDictionaryValue("QuoteDB")

adoCnn.Open strCnnStr
adoRst.CursorLocation = adUseClient
adoRst.Open strSQL, adoCnn, adOpenStatic, adLockOptimistic, adCmdText
Set adoRst.ActiveConnection = Nothing
```



# パフォーマンス実測(当社の株価取得処理で計測)



- 同一筐体にSQLServerとKairosとをセットアップし、SQLServerは2万回の株価取得のクエリをランダムに実行  
Kairosは20万回の株価取得のクエリをランダムに実行
- 各グラフについて  
  - 青色→SQLServer
  - 赤色→Kairos
  - 緑色→TotalProcessorTime
- トータルCPU利用率は20%強から10%未満に。同一条件では1%程度と想定
- Kairos側はSQLServerと比較し、低CPU利用率、短時間で実行を終了

## Kairosへの移行を終えて

- 大型時価DBを完全廃止する事が出来た
- データベース接続文字列の変更(環境設定の変更)のみで、プログラム修正は発生せず、短期間で移行が可能だった。
  - 当社におけるKairos導入にあたり、Kairos側をよりMS製品と近づけるためのカスタマイズが可能だった事が導入の決め手

## Kairosのカスタマイズ

- ADOからのアクセスが可能である事  
→COM+やIISからのアクセスは必須条件
- View(DBにおける複数の表の連結)のアクセスが可能となる事  
→Indexが効率よく利用される事が必須条件
- NTサービスに準拠する事  
→メモリDBが利用可能である前提として、イニシャルデータを取り込み済みの状態で起動される必要があった。



カブドットコム証券

東証 1 部 [8703]

わたしたちは **MUFG** です。   
MUFG