
VIOPS-2 Workshop
パネルディスカッション資料

R&D Lab兼
データセンター事業部
エキスパート

國武 功一

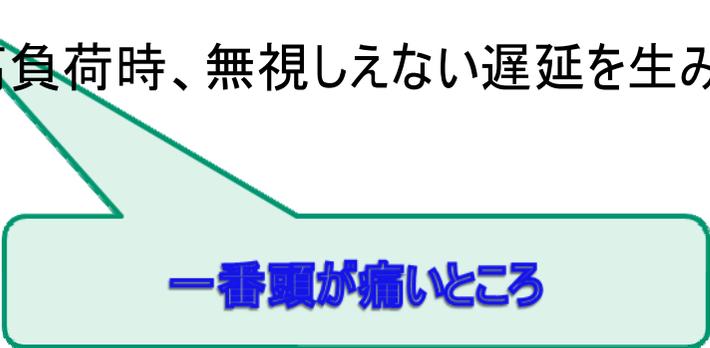


Webサイト、やっとできました www.benc.jp - IPv4/IP6 dual stack – 仮想マシンです

 **What** なにを分かち合うの？

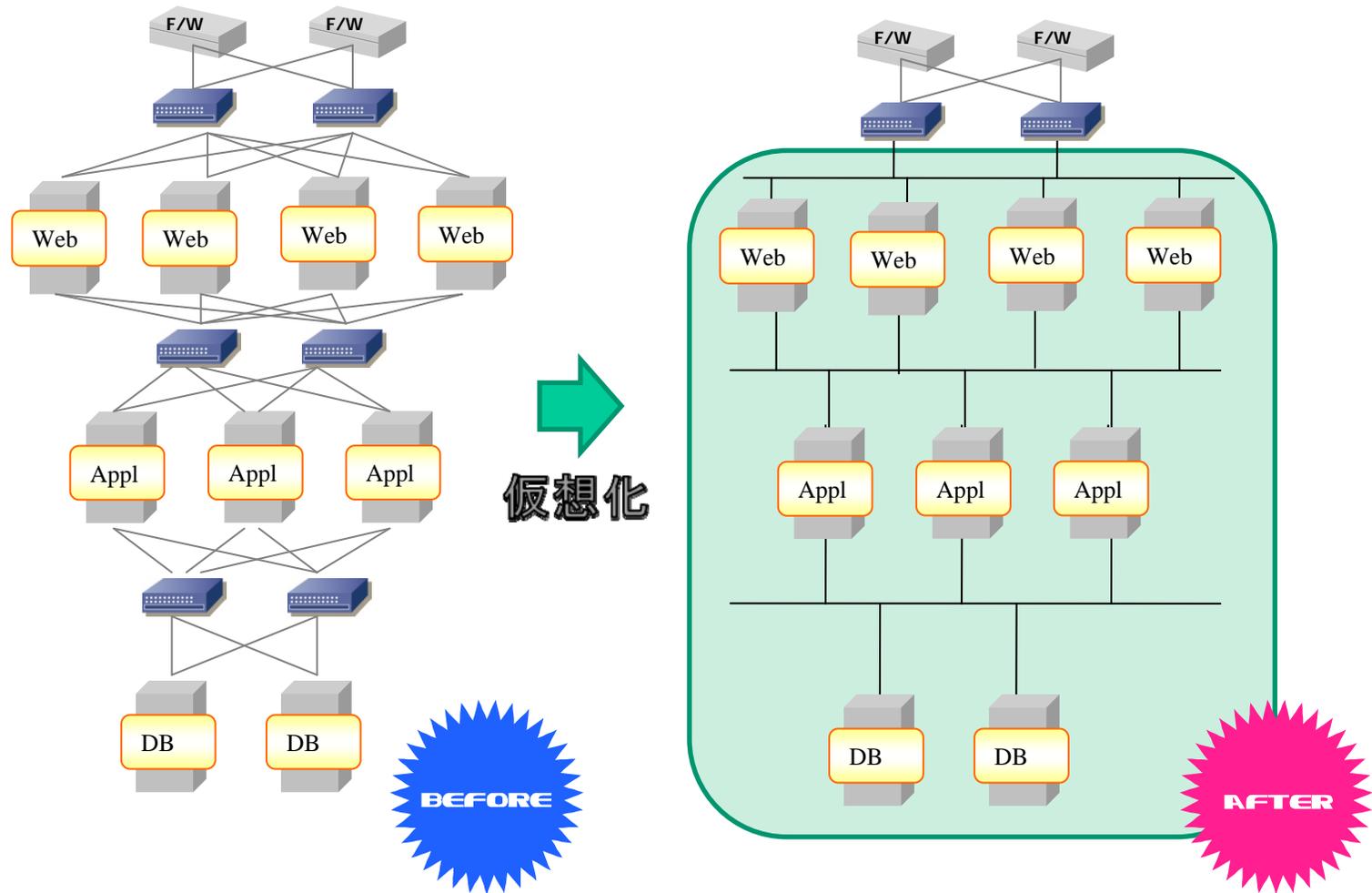
- CPU
- メモリ
- ストレージ容量
- I/O帯域
 - ネットワーク
 - ディスク
- 予算？

- CPU
 - おおむねオーバーヘッドは無視できる。
- メモリ
 - オーバーコミット状態で動かさなければ、おおむねオーバーヘッドは無視できる
- ストレージ容量
 - ファイルシステムによっては、動的に容量を変更できる
- I/O
 - 高負荷時、無視しえない遅延を生み出す。

A light green callout box with a dark green border and rounded corners. A line extends from the top-left corner of the box, pointing towards the 'I/O' item in the list above. The text inside the box is in blue.

一番頭が痛いところ

まあそれはさておき、一般的な構成



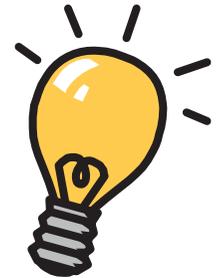
 仮想化

- Webサーバ
 - CPU:あまり消費しない
 - メモリ:キャッシュとして多めに割り当て
- Appサーバ
 - CPU:多く消費
 - メモリ:それなりに。動的コンテンツを取り扱う
- DBサーバ
 - CPU:ピークでは多く消費
 - メモリ:DB自身もキャッシュとして多く利用
 - データ更新時にディスクアクセス
 - メモリで処理できなければ、一時ファイルを作ってソートなど

Thin file
Thin Provisioningが
使えそう？

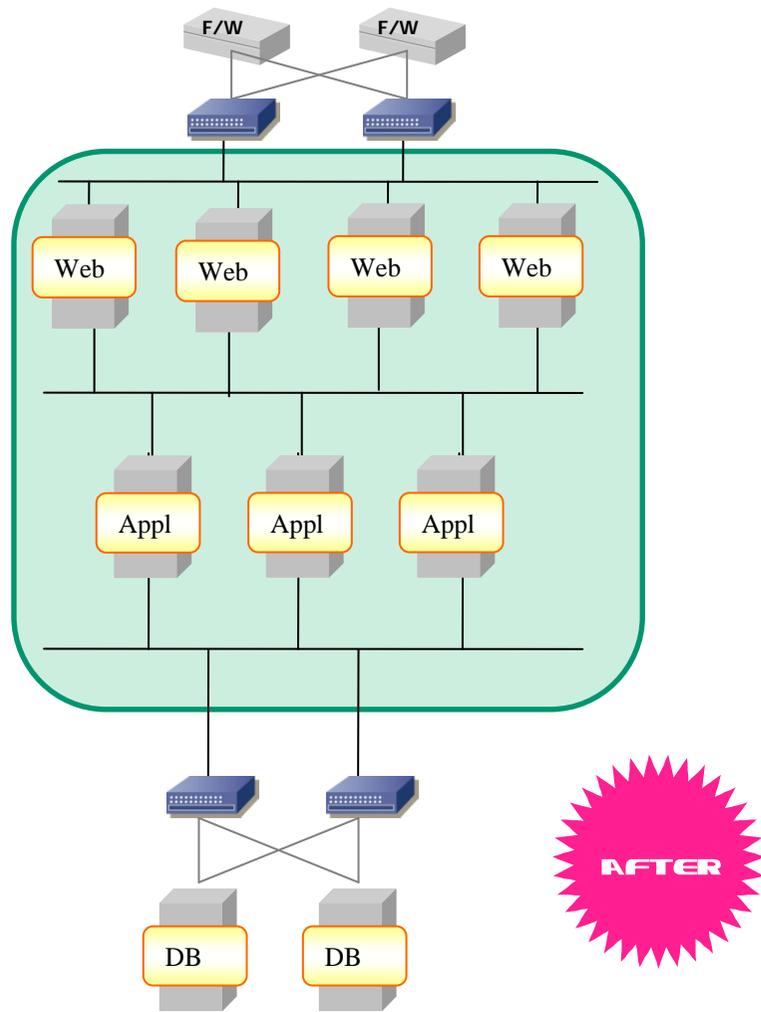
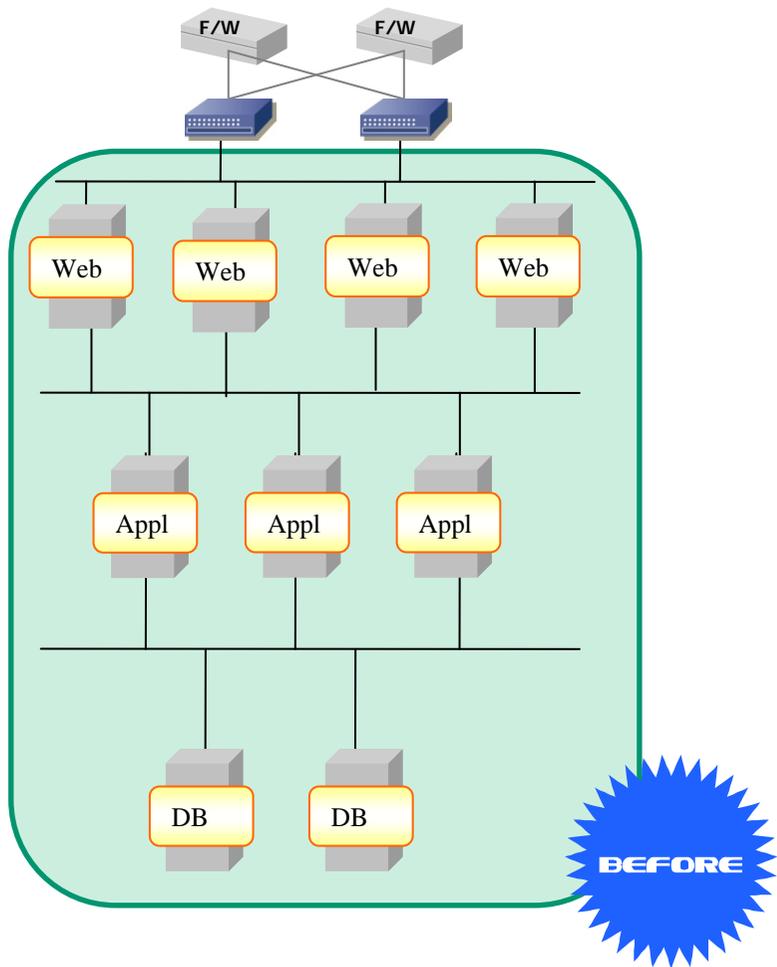
使ってほしくないリソースを一番消費する

- サーバへのアクセスは、一般的に
Read : Write = 8 : 2
程度と言われており、圧倒的にReadが多い
– ならいっそ、ディスクアクセスさせなければ！？



結論！
コンテンツ、DBのデータサイズよりも
大きなメモリ領域を割り当てる！

いっそDBの仮想化をあきらめてみる



□ 仮想化

- ベンチマーク以外でそこまで求められる？
 - SQLの最適化（インデックスを適切に使おう！）
 - これだけで数百倍速くなることも
 - データベースエンジンの適切な選択
 - テーブルロックでスケールする？
 - エージング後のデータベース再チューニング
 - やはりスケールアウトが基本
 - 仮想サーバを増やすことはスケールアウトになってます？
（仮想I/Oは魅力的で、サーバ⇔ストレージ間の帯域を心配しなくてよさそうだが、実際にストレージ内では？
ちょっとモヤモヤ）

- ファイルシステムをガリガリするアプリ
 - コンテンツ改竄検知
 - rsync
 - locateファイルの更新

サーバ屋さんの対処は限られる

- I/Oアクセスを削減
 - 割り当てメモリを増やす
 - mountオプション変更 (noatime)
- ピークをずらす
 - Cronによる起動時間の変更