

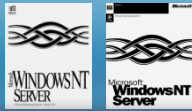
マイクロソフトの次世代
Webテクノロジー 徹底解説
Windows Server 2008 / IIS7.0 /
FastCGIで変わるPHP環境

マイクロソフト株式会社
IT Proエバンジェリスト
奥主 洋 (おくぬしひろし)

<http://blogs.technet.com/hirookun>

IIS - 色々な過去

1996



V1&V2をリリース

1997



V4をOption Packとしてリリース

2000



V5を既定インストール

2001

Code Red & Nimda

2002



V5.1をリリース、後にSP2で強化

2003



V6.0をリリース、後にSP1で再強化

2007



V7.0を開発環境リリース

200x



V7.0を運用環境リリース

SP: サービスパックの略でマイクロソフトの修正プログラムパッケージ

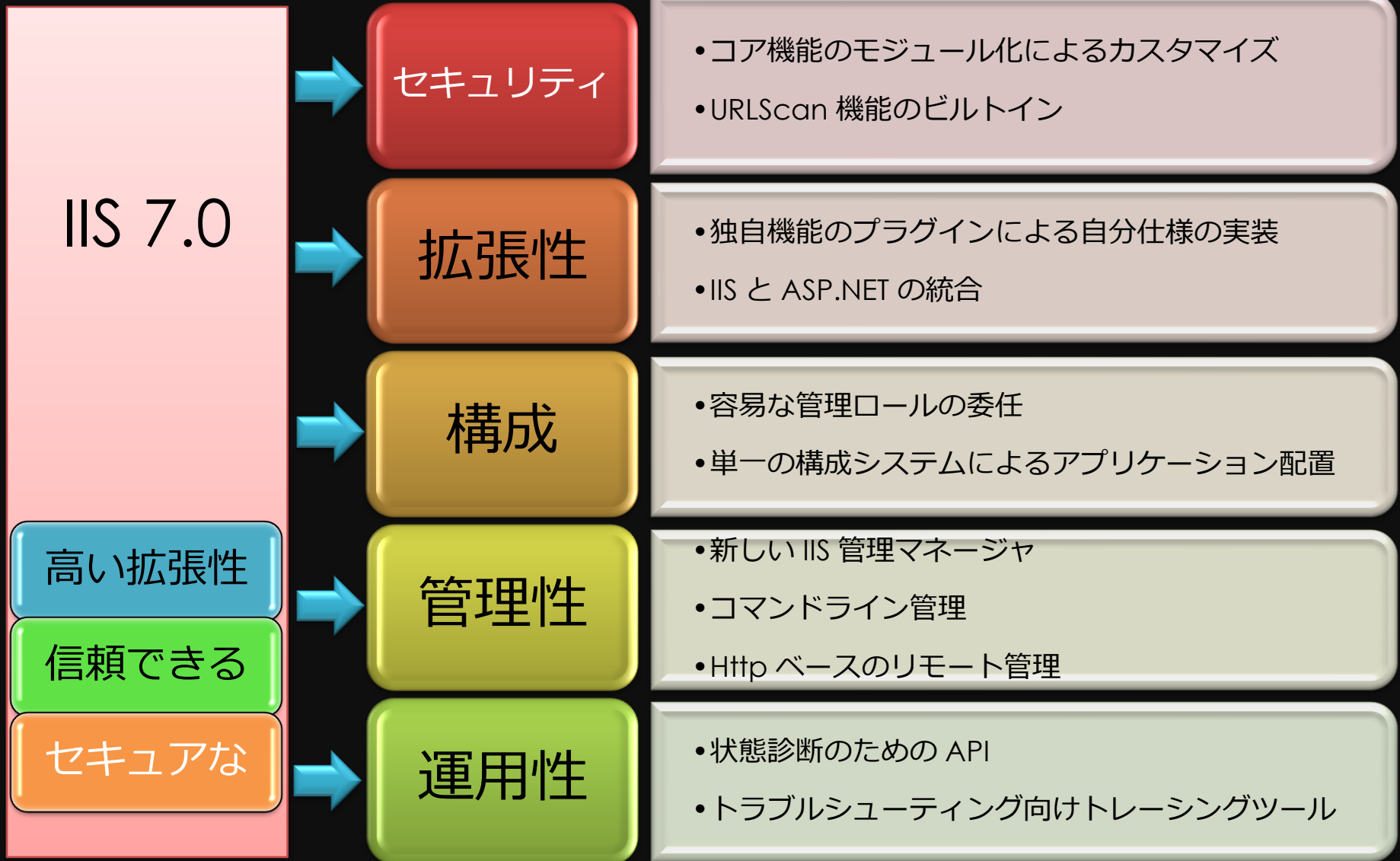
IIS 6.0 : 証明済のプラットフォーム

- 証明されたスケーラビリティ
 - MySpace – 230億 ページビュー/月
 - Microsoft.com – 1万 リクエスト/秒、30万 同時接続
 - Match.com – 3千万 ページビュー/日
- 証明されたセキュリティ
 - 出荷以来 緊急 レベルの IIS 6.0 修正プログラムなし
- 証明された信頼性
 - フォーチュン1000 企業の 55% が IIS をインターネット企業トップWebサイトに使用
(<http://www.port80software.com>)

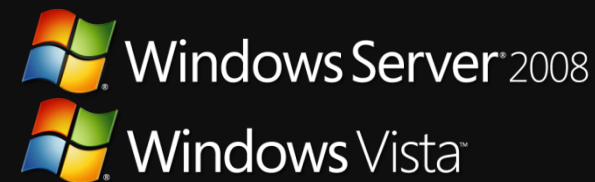


安心してお使いいただけるWebプラットフォームです！

IIS 7.0 の柱 次のレベルへ



IIS 7.0の提供形態



	役割数	機能制限	セットアップ	同時接続数
Windows Vista Home Basic	-	IISなし	-	3
Windows Vista Home Premium	-	FTPなし、認証一部なし、ログなし	プログラムと機能 Pkgmgr	3
Windows Vista Business/Enterprise/Ultimate	-	リモート管理なし	プログラムと機能 Pkgmgr	10
Windows Server 2008 Data Center/Enterprise/Standard	17	全機能	サーバーマネージャ ServerManagerCmd Pkgmgr	制限なし
Windows Server 2008 Data Center/Enterprise/Standard Server Core	17	GUI、.NETなし	Pkgmgr	制限なし
Windows Web Server 2008	2	全機能	サーバーマネージャ ServerManagerCmd Pkgmgr	制限なし
Windows Web Server 2008 Server Core	2	GUI、.NETなし	Pkgmgr	制限なし

①



Internet Information Services (IIS)は
セキュリティが心配だ。

だからインターネット環境はダメ



本当にそうなのかなあ??

IIS6のセキュリティパッチ実情

<http://www.microsoft.com/japan/technet/security/current.aspx>

IISのバージョン	件数
IIS4.0 (Windows NT 4.0)	42件
IIS5.0 (Windows 2000)	31件
IIS5.1 (Windows XP)	7件
IIS6.0 (Windows 2003)	2件
IIS7.0 (Windows Vista)	??
IIS7.0 (Windows 2008)	??

製品/テクノロジー: Internet Information Services 6.0

サービスパック: すべて

深別度: 緊急 (Critical) 重要 (Important) 警告 (Moderate) 注意 (Low)

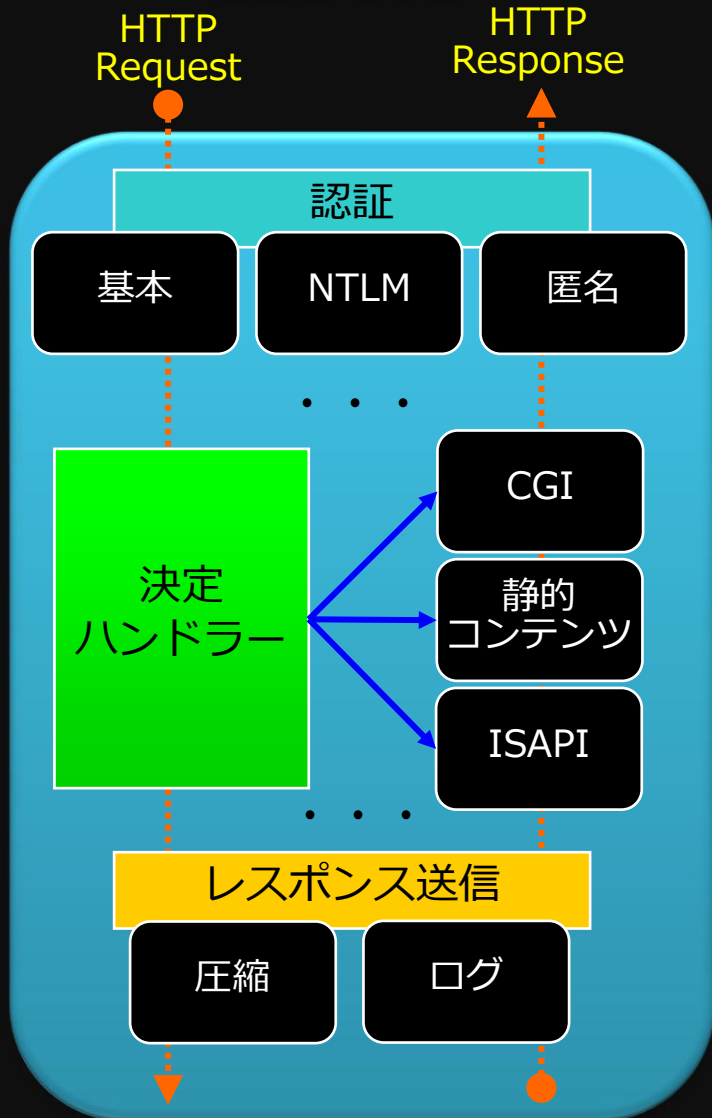
日付	タイトル	影響を受けるサービスパック	深別度
2006/07/12	MS06-034: Active Server Pages を使用した Internet Information Services (IIS) の脆弱性により、リモートでコードが実行される (917537)	Windows 2000 Service Pack 4, Windows XP Service Pack 1, Windows XP Service Pack 2, Windows XP 64-Bit 初期出荷版, Windows Server 2003 初期出荷版, Windows Server 2003 SP1, Windows Server 2003 for Itanium-based Systems 初期出荷版, Windows Server 2003 for Itanium-based Systems SP 1, Windows Server 2003 x64 初期出荷版, Internet Information Server 6.0 初期出荷版, Internet Information Server 5.1 初期出荷版, Internet Information Server 5.0 初期出荷版	重要
2004/10/13	MS04-030: WebDav XML Message ハンドラの脆弱性によりサービス拒否が起こる (824151)	Windows 2000 Service Pack 3, Windows 2000 Service Pack 4, Windows Server 2003 初期出荷版, Windows XP 初期出荷版, Windows XP Service Pack 1, Windows XP 64-Bit SP1, Windows XP 64-Bit Version 2003 初期出荷版, Windows Server 2003 x64 初期出荷版	重要

2007年8月31日現在

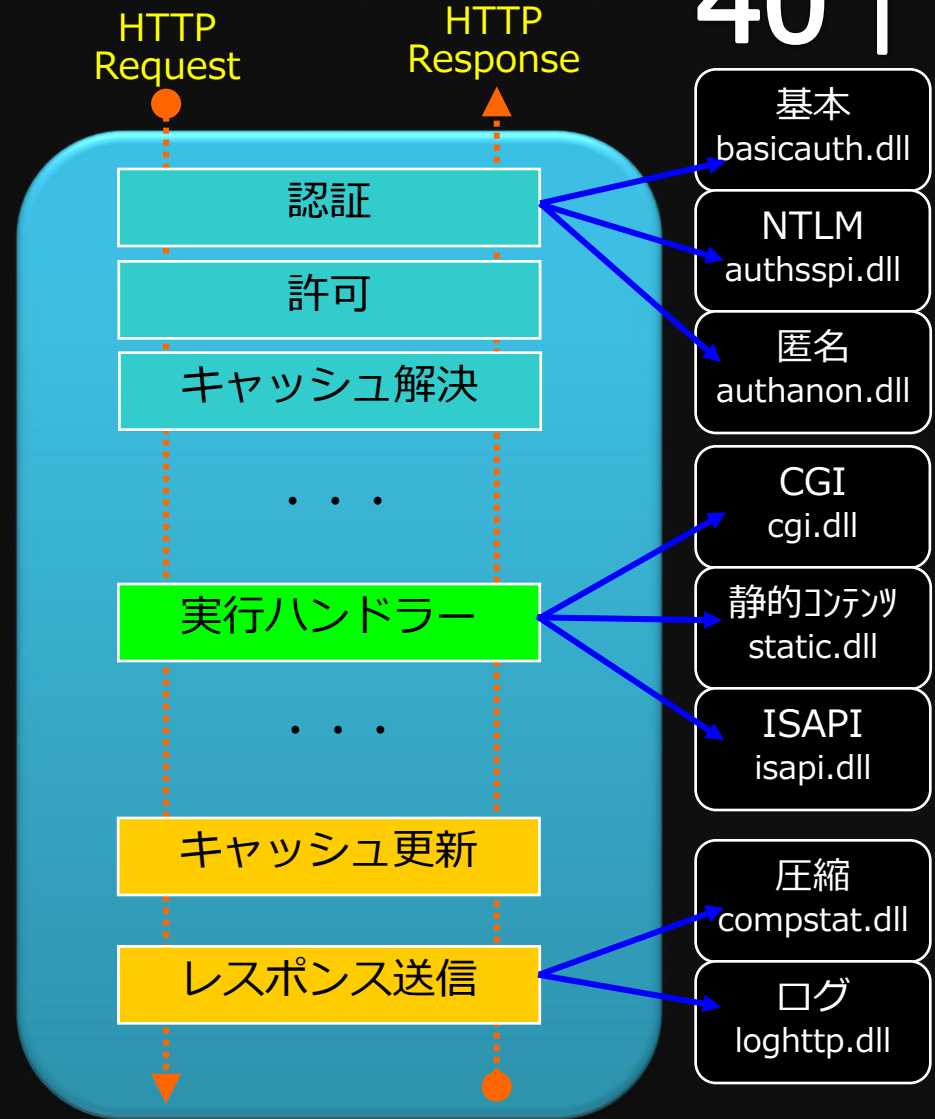
重要が2件

アーキテクチャの変更

IIS6.0



IIS7.0 (統合モード)



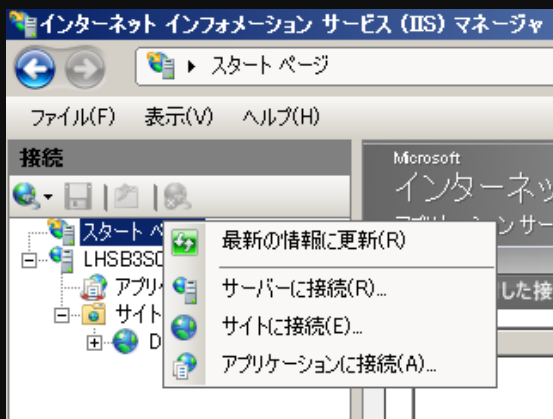
セキュリティ



モジュール化！
必要なものだけ
＝攻撃対象の軽減



認証機能の統合
＝.NETの認証機能を
他のコンテンツにも



サイト/アプリレベル
での管理委任
＝管理者の制御強化

リクエスト
フィルタリング



<http://site.com/bin> ❌

<http://site.com/binary> ✅

```
<configuration>
  <system.webServer>
    <security>
      <requestFiltering>
        <hiddenNamespaces>
          <add hiddenDirectory="BIN"/>
        </hiddenNamespaces>
      </requestFiltering>
    </security>
  </system.webServer>
</configuration>
```

新機能: HiddenNamespace
指定URLにアクセス＝404.8

```
<configuration>
  <system.webServer>
    <security>
      <requestFiltering>
        <fileExtensions allowUnlisted="true" >
          <add fileExtension=".asp" allowed="false"/>
        </fileExtensions>
      </requestFiltering>
    </security>
  </system.webServer>
</configuration>
```

IIS6で登場したURLScan
IIS7ではモジュールで実装

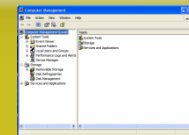
Windows Server 2008 の設計思想

サーバー サービス (その他役割)



Shell

Microsoft
.net
Server Core Plus



MMC



ネットワーク

DNS
DHCP
File

+IIS7.0



セキュリティ



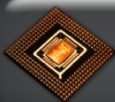
Active Directory

Server Core



管理

WMI
RPC
Agents
Anti-virus



ホットプラグ可能なハードウェア



注意: IIS7.0 が Server Core に搭載されたのは JuneCTP というビルドからで、Beta3 には入っていません。



Internet Information Services (IIS)は
ブラックボックスだ。

だから問題解析に困る、運用しづらい



本当にそうなのかなあ??

作業中 - Microsoft Internet Explorer

ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)

戻る 検索 お気に入り

アドレス(D) http://localhost/

作業中

表示しようとしているページは構成中の可能性があります。

後でもう一度このサイトの管理者に連絡してください。

Web サイト管理者が、ヘルプにある「動的コンテンツ」...

IIS ヘルプにアクセス

1. **【スタート】** をクリックします。
2. **【名前】** ボックスに入ります。
3. **【ヘルプ】** メニューを開きます。
4. **【インターネット】** をクリックします。

ページが表示されました

ページが見つかりません - Microsoft Internet Explorer

ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)

戻る 検索 お気に入り

アドレス(D) http://localhost/index.php

移動 リンク >>

ページが見つかりません

検索中のページは、削除されたか、名前が変更されたか、または一時的に利用できない可能性があります。

対処方法

- ブラウザのアドレスバーに表示された Web サイト アドレスの綴りと形式が正しいことを確認してください。
- リンクをクリックしてこのページに到達した場合は、Web サイトの管理者に連絡して、リンクの形式が正しくないことを通知してください。
- 別のリンクへアクセスするには、**【戻る】** をクリックしてください。

HTTP エラー 404 - ファイルまたはディレクトリが見つかりません。
インターネット インフォメーション サービス (IIS)

技術情報 (サポート担当者用)

- 「[Microsoft 製品サポート サービス](#)」で、「HTTP」および「404」の文字列によりタイトルを検索してください。
- IIS マネージャ (inetmgr) からアクセスできる IIS の **【ヘルプ】** を開き、「Web サイトのセットアップ」、「一般的な管理タスク」、および「カスタム エラー メッセージについて」というトピックを検索してください。

ページが表示されました

イントラネット

Localhost でブラウザーに戻るエラー

アプリケーション "Default Web Site" でのサーバー エラー

インターネット インフォメーション サービス 7.0

エラーの概要

HTTP エラー 404.0 - Not Found
探しているリソースは削除されたか、名前が変更されたか、または一時的に使用不可能になっています。

エラー情報の詳細

モジュール	IIS Web Core	要求された URL	http://localhost:80/arimasen.htm
通知	MapRequestHandler	物理パス	C:\inetpub\wwwroot\arimasen.htm
ハンドラー	StaticFile	ログイン方法	匿名
エラー コード	0x80070002	ログイン ユーザー	匿名

可能性のある原因:

- 指定されたディレクトリまたはファイルは Web サーバーに存在しません。
- URL に誤字が含まれています。
- URLScan などのカスタム フィルタやモジュールによってファイルへのアクセスが制限されています。

対処方法:

- Web サーバーにコンテンツを作成します。
- ブラウザの URL を確認します。
- この HTTP 状態コードに対して失敗した要求を診断するトレース規則を作成し、SetStatus を呼び出しているモジュールを確認します。失敗した要求のトレース規則の作成の詳細については、[ここ](#)をクリックします。



エラーの概要

エラー情報の詳細

可能性のある原因

対処方法

- 再構築されたカスタムエラー
 - サーバー外のクライアントと "Localhost" で相違
- 詳細なエラー結果情報
 - 時刻、URL、実行中のモジュール、ステータス etc

Runtime Status and Control Data

実行時のステータス・制御データの取得

- 実行時ステータスをAPIで取得、制御する
- 最新状態・詳細な実行時の状況を取得する方法を提供する

WMIでの利用

```
' connect to provider
set oService = GetObject("winmgmts:root¥WebAdministration")

for Each oProcess in oService.InstancesOf("WorkerProcess")
' the PID of the process is just the Id property
WScript.Echo "---W3WP DETAILS---"
WScript.Echo " W3WP PID : " & oProcess.PID
WScript.Echo " W3WP AppPoolId : " & oProcess.ApplicationPool
```

Next

APPCMDでの利用も
MWAでの利用も
もちろん可能！

※MWA:Microsoft.Web.Administration

IISマネージャでの利用

接続

スタート ページ
W2K8IDS3 (W2K8IDS3¥)
アプリケーション プール
サイト

要求

ワーカー プロセスの現在の要求を表示します

現在、'0' 秒を超えて動作している、アプリケーション プール 'DefaultAppPool' に関連付けられているプロセス ID '3172' の要求を表示しています

フィルタ: 検索(G) すべて表示(A)

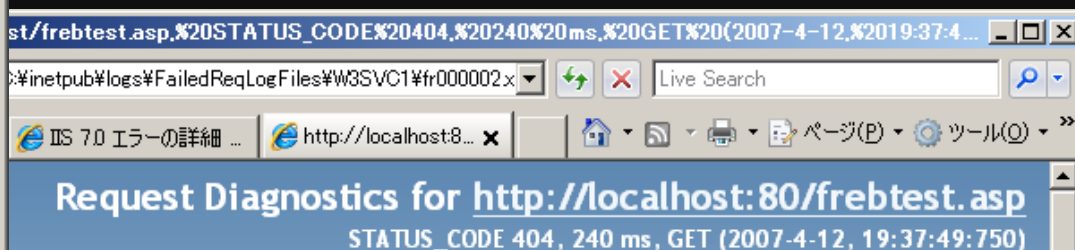
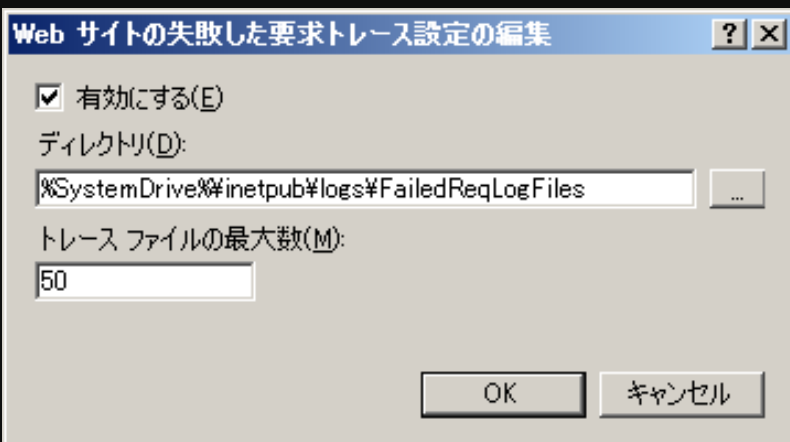
	URL	動詞	クライ...	状況	モジュール名
1	/test.asp	GET	:1	ExecuteRequestHa...	IsapiModule

操作

ヘルプ
オンライン ヘルプ

失敗した要求トレース 機能

Automatic Failed Request Tracing あるいは FREB



fr999999.xml

+

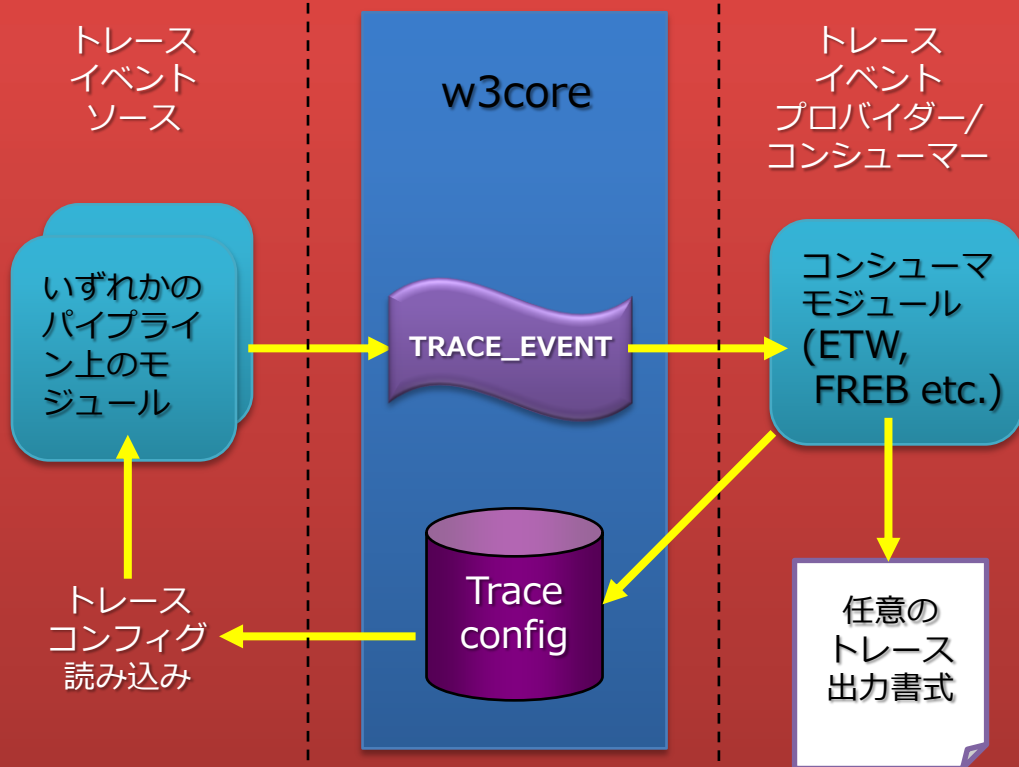
freb.xml



ETW機能との統合

Event Tracing for Windows

トレースの共通フレームワーク



- ETWを利用した出力はETLファイルへ
- ETWは IIS6.0 SP1でも出力可能
- ETWはKernel Modeによるトレース出力
- Vista以降ではOS管理ツールと統合





③

Internet Information Services (IIS)は
台数が多いと管理が大変だ。

だからホスティング環境では使えない



本当にそうなのかなあ??

新しいサーバー構成の仕組み

¥Windows¥Microsoft.NET¥Framework¥vX.X.XXXXX¥config

.NET Framework
の構成情報
machine.config



ASP.NET 既定
の構成情報
web.config



継承構造

アプリケーション
の構成情報
web.config
*.php



¥Windows¥System32¥inetsrv¥config

IIS7.0
の構成情報



applicationHost.config

他にも

- administration.config
IISツールUI関連構成、IISユーザー情報
- redirection.config
シェアドコンフィグレーション設定



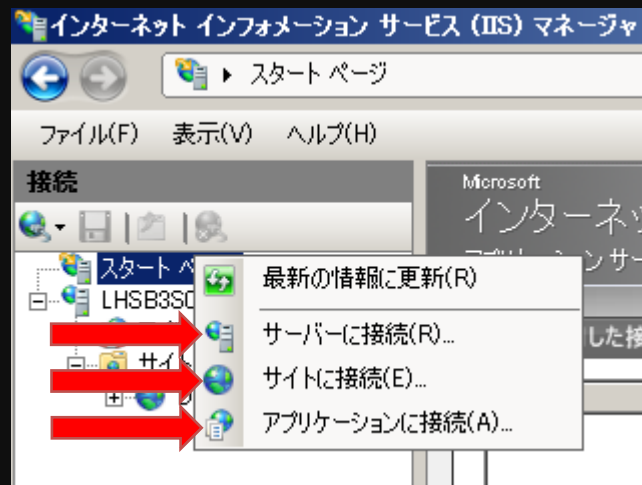
サーバー構成も
アプリで指定可能に

IIS6.0ではmetabase.xml、しかも実際の構成情報はメモリ上が最新
IIS7.0では新configファイルが本当の意味でのマスターファイル

スキーマ定義ファイル : %windir%¥system32¥inetsrv¥config¥schema¥
IIS_schema.xml, ASPNET_schema.xml , FX_schema.xml

サーバー管理者視点での構成管理

- システム管理者はグローバルおよび特定設定が可能
- サイト/アプリを委任されたユーザーは自身のサイト/アプリの設定をすることが可能



```
<sectionGroup name="system.webServer">
  <section name="asp" overrideModeDefault="Deny" />
  <section name="caching" overrideModeDefault="Deny" />
  <section name="cgi" overrideModeDefault="Deny" />
  <section name="defaultDocument" overrideModeDefault="Allow" />
  <section name="directoryBrowse" overrideModeDefault="Allow" />
  <section name="globalModules" allowDefinition="AppHostOnly" overrideModeDefault="Deny" />
</sectionGroup>
```

applicationHost.config

- セクションをアンロックするとセクション全体をサイト/アプリの管理者に開放
- 粒度の小さいロックも特定の元素/アトリビュートに対して可能
- lockAttributes、lockAllAttributesExcept、lockElements (ほか)

実際の運用では標準を決める！
⇒ アプリをXCOPY配布可能

豊富な管理手法

GUI : IISマネージャ



Scripting : WMI

```
Set oService = GetObject("winmgmts:root¥WebAdministration")
```

' Create binding for site

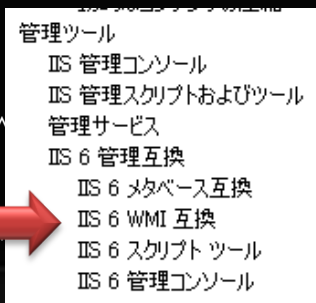
```
Set oBinding = oService.Get("BindingElement").SpawnInstance_  
oBinding.BindingInformation = "*:80:www.site.com"  
oBinding.Protocol = "http"
```

' Create site

```
oService.Get("Site").Create _  
"NewSite", array(oBinding), "C:¥inetpub¥www"
```

' Create application

```
oService.Get("Application").Create _  
"/foo", "NewSite", "C:¥inetpub¥www¥root¥root"
```



管理機能は全部WS-Managementベースでの設計のため、通信は
HTTP/HTTPS ベースで行われるため、ファイアウォールを超えた管理も容易

CUI : appcmd+winrs

```
C:¥> appcmd list sites  
SITE "Default Web Site" (id:1,bindings:HTTP/*:80:,state:Started)  
SITE "Site1" (id:2,bindings:http/*:81:,state:Started)  
SITE "Site2" (id:3,bindings:http/*:82:,state:Stopped)
```

```
C:¥> appcmd list requests  
REQUEST "fb0000008000000e" (url:GET /wait.aspx?time=10000,time:4276  
msec,client:localhost)
```

```
C:¥> appcmd list requests /apppool.name:DefaultAppPool
```

```
C:¥> appcmd list requests /wp.name:3567
```

```
C:¥> appcmd list requests /site.id:1
```

App : マネージドコード 例 : Microsoft.Web.Administration

```
ServerManager iisManager = new ServerManager();
```

```
foreach(WorkerProcess w3wp in iisManager.WorkerProcesses) {  
Console.WriteLine("W3WP {0}", w3wp.ProcessId);
```

```
foreach(Request request in w3wp.GetRequests(0)) {  
Console.WriteLine("{0} - {1},{2},{3}",  
request.Url,  
request.ClientIPAddr,  
request.TimeElapsed,  
request.TimeInState);  
}  
}
```

PowerShellとの組合せは強力!

Windows PowerShell

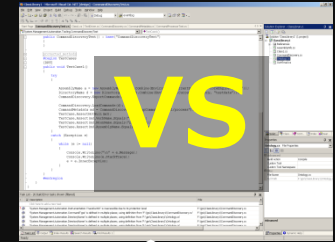
～ 強力な次世代型 Windows スクリプティング環境 ～

<PowerShell とは?>

- BASH / KSH のようなシェル
- Perl / Ruby のようなプログラミング
- TCL のような埋め込み
- AS400 CL / VMS DCL のような生産指向
- ファイルシステムのように、簡単にデータストアにアクセス

<一例>

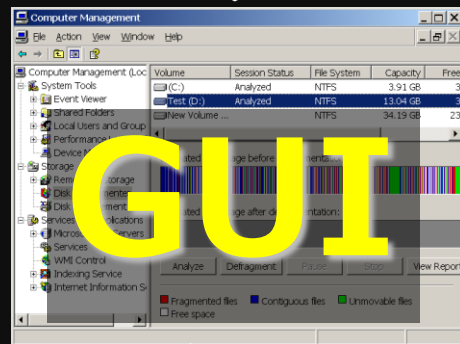
管理機能をコマンドレットとして実装し、コマンドラインと管理コンソールから同等にアクセスできるようにする



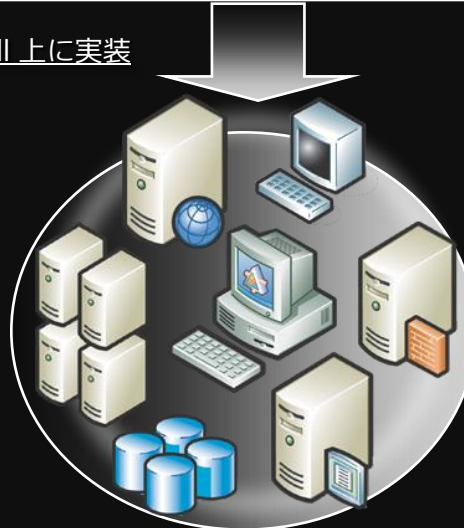
PowerShell Engine



GUI のツールをPowerShell 上に実装



ユーザーフレンドリー、初期設定等



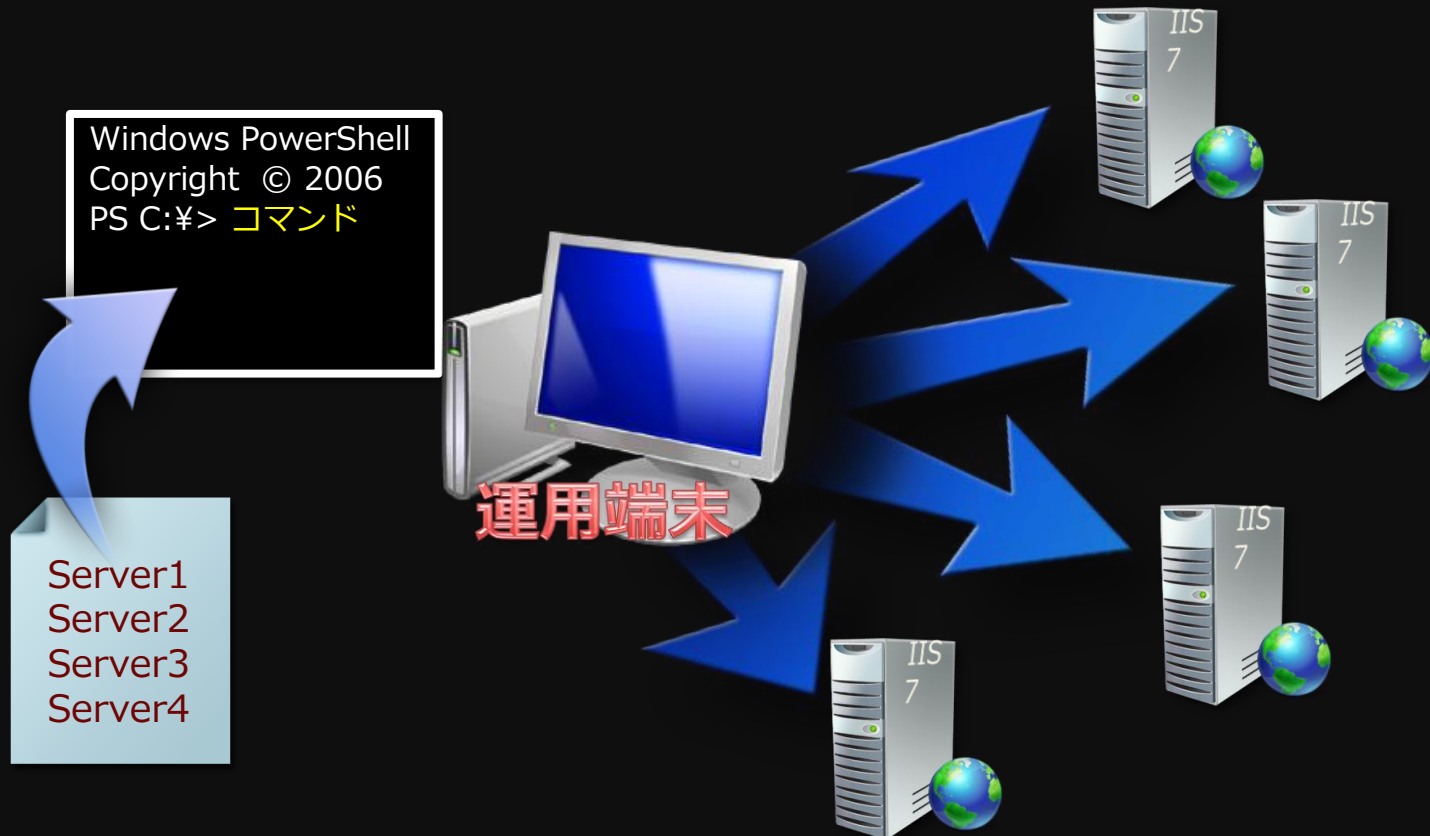
管理対象



バッチ、スクリプトで自動化

複数サーバーの同時管理

- Windows PowerShell + Microsoft.Web.Administration
 - 事前ロードするコマンドレット（専用コマンド）も開発でき、幅広い可能性を提供している
- WinRSとAppCmdの組合せもあるが、連続操作になる



IISとPowerShell連携の例

アプリケーションの配布

```
# file: Deploy-Application.ps1
$sourceMachine = "DemoServer1"
$farmList = get-content '.¥RestOfFarm.txt'
$filesToCopy = get-content '.¥AppManifest.txt'
foreach ($targetMachine in $farmList)
{
  foreach ($file in $filesToCopy)
  {
    $sourcePath = "¥¥" + (join-path $sourceMachine $file)
    $destPath = "¥¥" + (join-path $targetMachine $file)
    write-host -for yellow "$targetMachine : Copying files
from
  $sourcePath"
    copy-item $sourcePath $destPath
    -recurse -force
  }
}
```

Server1
Server2
Server3
Server4

C:¥PHP
C:¥App1
C:¥App2
D:¥App3

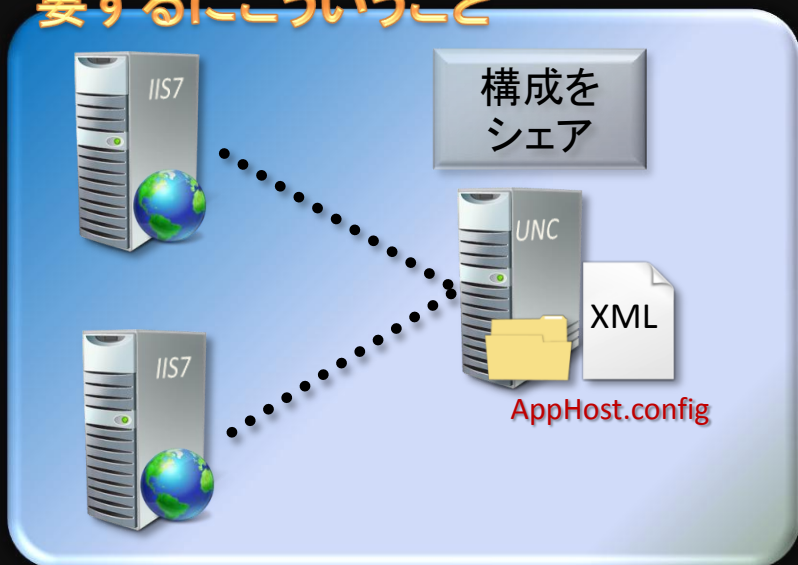
XML

web.config

PS C:¥> .¥Deploy-Application.ps1

シェアドコンフィグレーション

要するにこういうこと



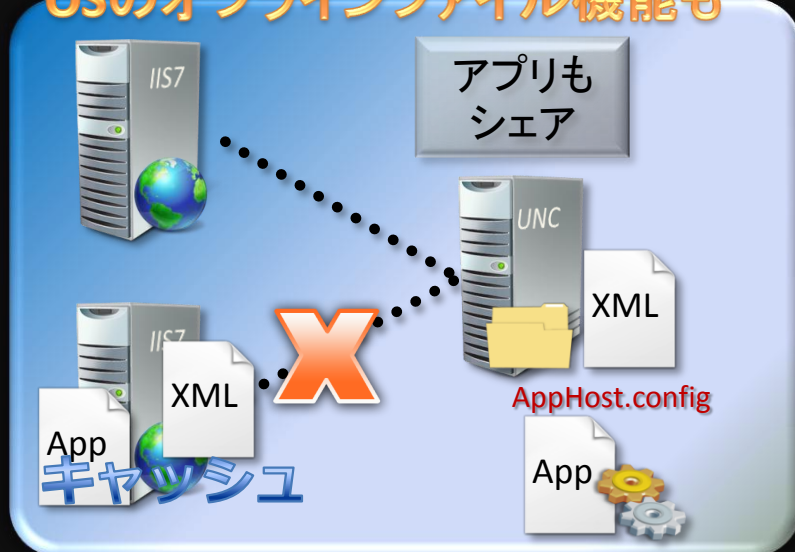
構成だけでなくアプリも



ステージング構成でVersion保持



OSのオフラインファイル機能も



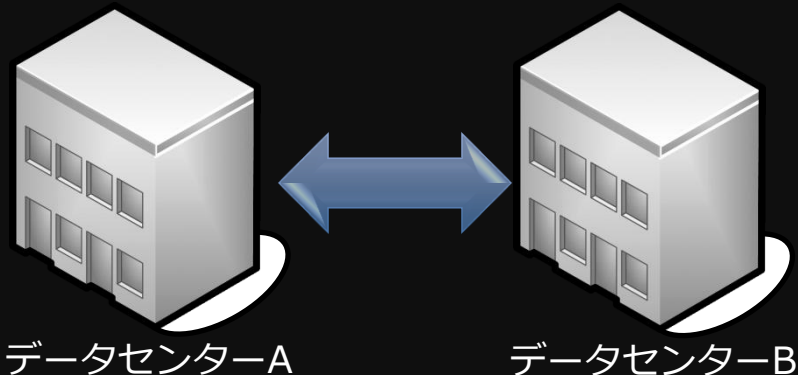
Windows Server 2008 / IIS 7.0 展開

Microsoft®

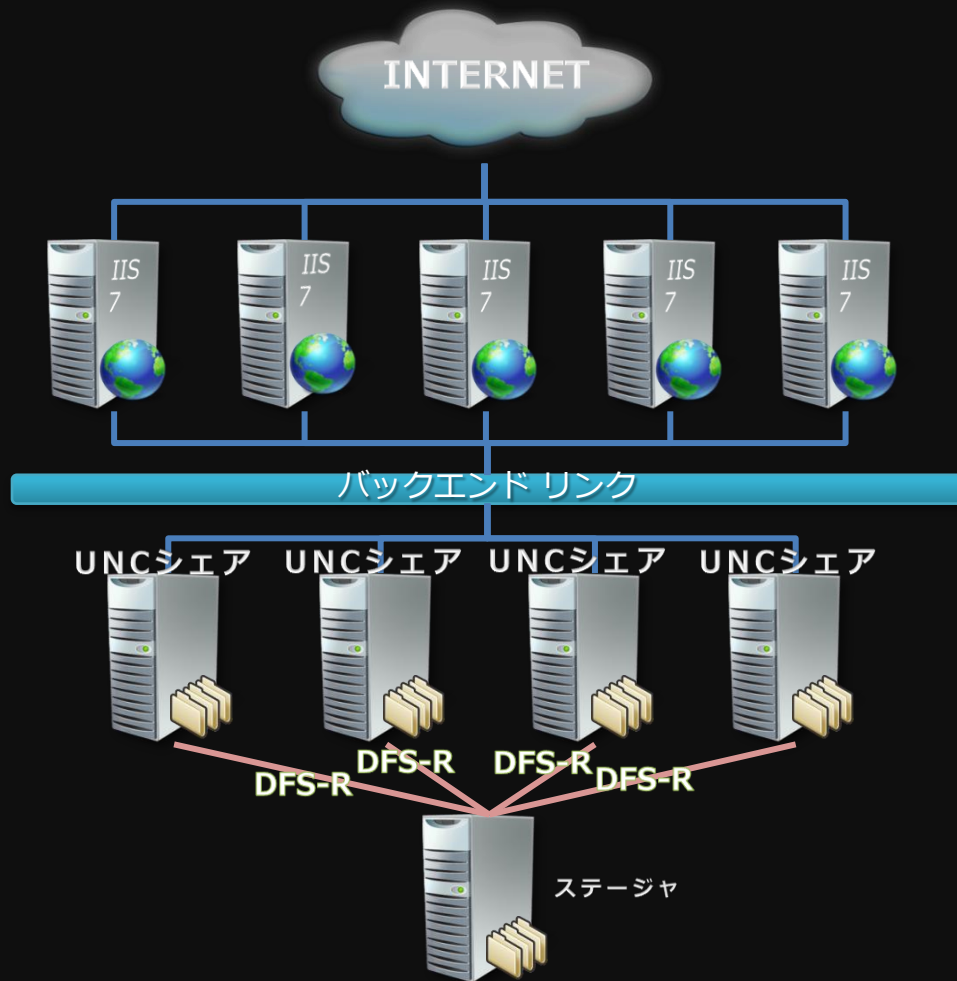
UNC コンテンツストア

- より簡単になったコンテンツ同期
- H/W のベースロードが軽く
- 業界共通の痛み

TCP/IP スタック 改良



広帯域を使ったファイル複製、
コンテンツ同期、ピア・ツー・
ピア通信、ログ SHIPPING



ファイル複製サービスの改良

- 遅延気味の長距離リンクは40倍
(12Mbps 対 425Mbps ↑)
- リモート地域を跨いだホスティングもより現実的に



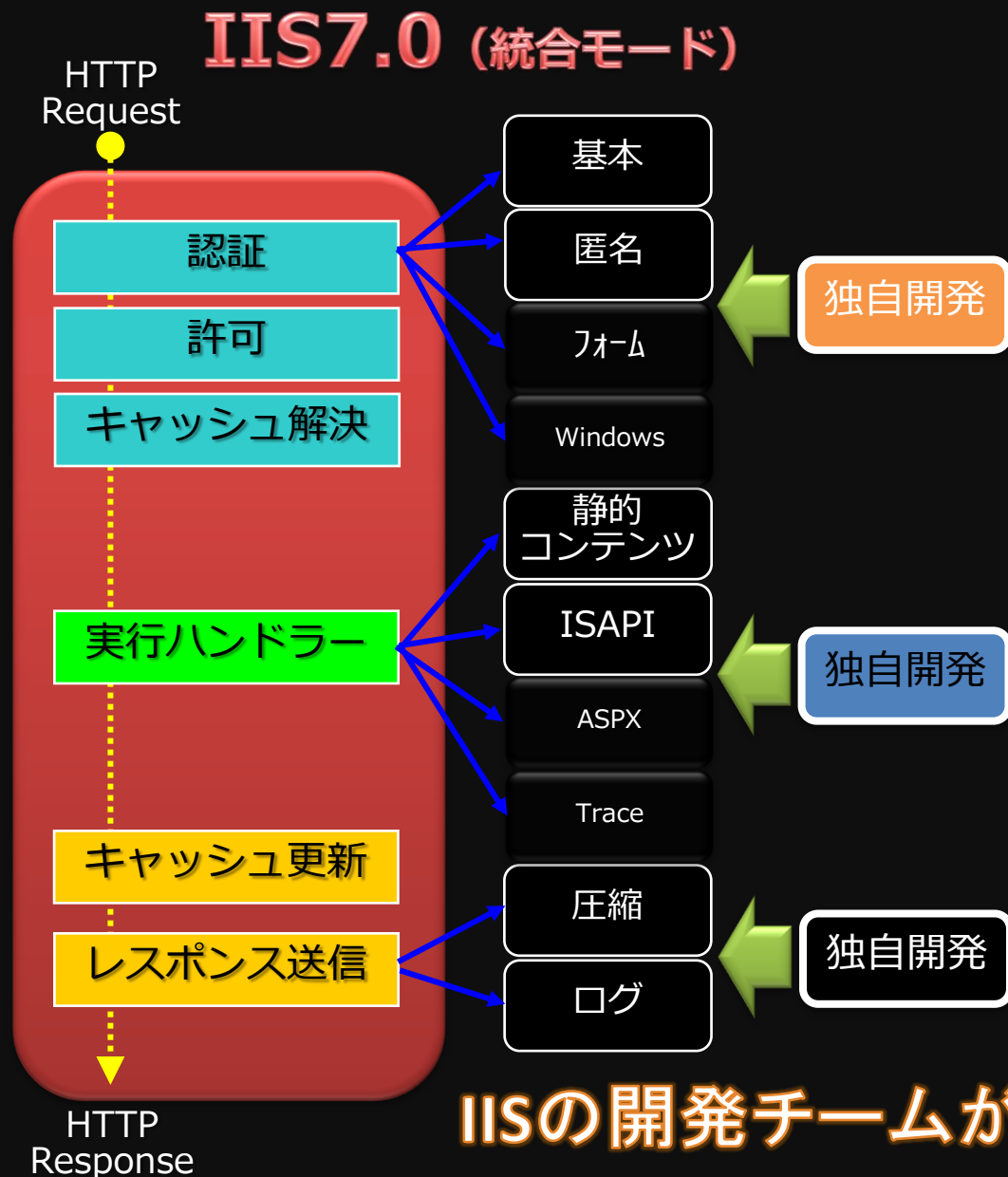
Internet Information Services (IIS)は
機能を拡張するのが大変だ。

だから思うように動かせない



本当にそうなのかなあ??

IIS7.0におけるサーバー機能開発



• モジュール開発


- ISAPIの互換性を残し、新強力APIでの開発
- 統一された機能の管理
- 粒度が小さい機能
- 全リクエスト対象

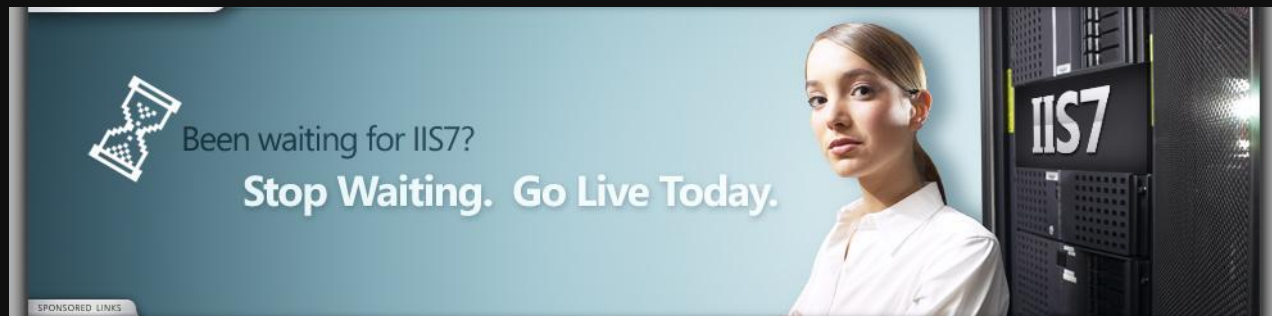
• 統合モードの影響

- 実行時の変更
 - IISネイティブのスレッドプールを使用
- パフォーマンス影響
 - プリコンディション
 - リシャッフル、マージ
- IHttpModule イベント順序
- コンフィギュレーション

IISの開発チームが使っているAPIを公開!

IIS7情報サイト（英語）

- IIS7だけじゃない！ IIS5・IIS5.1・IIS6の情報も
- <http://www.iis.net> 
- 広告ベース運営のコミュニティサイト
- IIS7製品開発チームの技術記事
- バーチャルラボでIIS7を体験できる！





Internet Information Services (IIS)は
.NETで開発しないと性能が出ない。

だから一緒にASP.NETの採用も検討必須



本当にそうなのかなあ??

動的コンテンツの出力キャッシュ活用

Microsoft®

- バックエンドに依存した呼び出しを抑える
- パフォーマンス面で飛躍的な向上
- 簡単な WCAT (Web Capacity Analysis Tool) による
ストレステストをかなり実施

www.microsoft.com/en/us/default.aspx

出力キャッシュ	テスト期間	RPS	CPU%
無効	1時間	346	100%
有効	1時間	3305	100%

- すべてのアプリケーションに適切な訳ではない
※パーソナライズ度の非常に高いコンテンツは効果があまり見込めないかもしれない

FastCGI & PHP



- PHP on IIS

- これまで：CGI（スピード出ず）、ISAPI（スレッド対応要）
- これから：FastCGI PHP5.2.1でテストも実施
 - ※今までのCGI動作の10倍速いパフォーマンス
 - ※Windows 2008は標準、Vista・2003・XPはダウンロード

- FastCGI対応による安定・高速化

- CGIプロセスをプールして再利用する構造
- シングルスレッド動作したまま性能向上：ISAPI版よりも安定度向上

- IIS7のプロセスモデルによりPHPの複数バージョンをサイド・バイ・サイドで動作可能

- アプリケーションプールで分離

- Internet標準のサポート

- FastCGIの活用はPHPだけでなく、Python・Ruby・Perlも

+ Zend Technologies 様との協業

Zend Core / Zend Platform によるさらなるパフォーマンス向上！！

Microsoft® TechNet バーチャルラボ 「はじめてのIIS7.0」 近日公開！

◎日本オリジナル！ Windows Server 2008, Windows Web Server 2008, Server Core を実際にバーチャルラボで体験してください。使用ビルドはJune CTPです！
IIS7.0ラボはこれから順次拡充していきますのでお楽しみに！！

<http://www.microsoft.com/japan/technet/traincert/virtuallab/default.mspix>

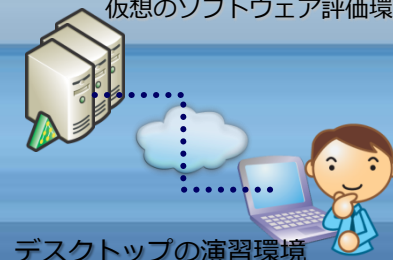
Microsoft® TechNet フォーラム 「IISフォーラム」 開設！

◎日本語で投稿できるIIS5&IIS6フォーラムとIIS7フォーラムを開設しました。
講師もディスカッションに参加いたしますので是非奮って投稿をお願いします！

<http://forums.microsoft.com/technet-ja/default.aspx?siteid=36>

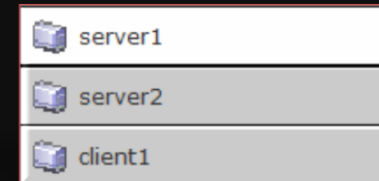
TechNet バーチャル ラボ

<http://www.microsoft.com/japan/technet/traincert/virtuallab/>



準備するのは、Web ブラウザとインターネット接続環境だけ

複数マシン環境でも場所の確保不要

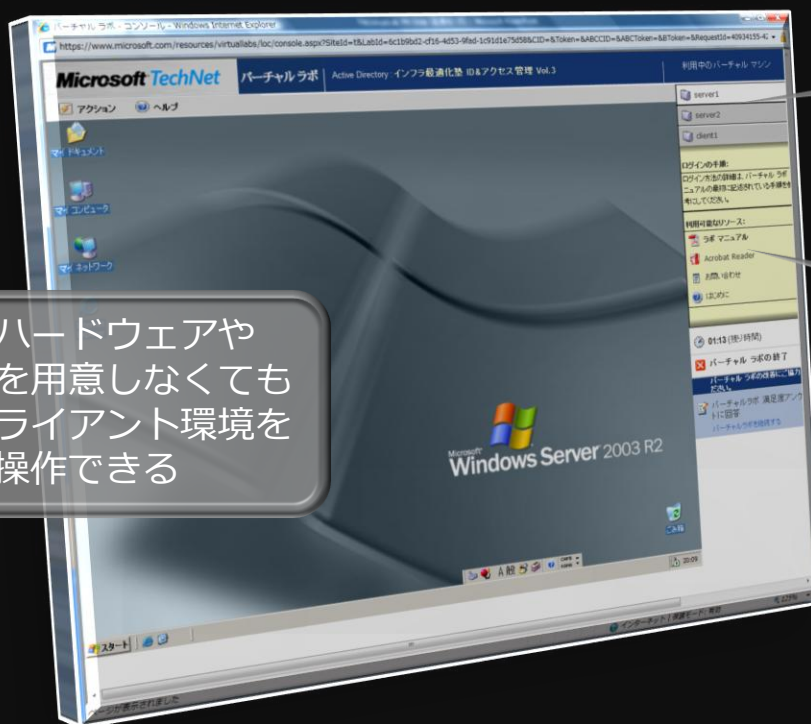


マニュアル(90分シナリオ)付き
新しいシナリオも続々登場！！

利用可能なリソース:

- ラボ マニュアル
- Acrobat Reader
- お問い合わせ
- はじめに

テスト用にハードウェアやソフトウェアを用意しなくてもサーバーやクライアント環境を実際に操作できる



バーチャル ラボは、マイクロソフトがホスティングする仮想のソフトウェア評価環境に、デスクトップ上の Web ブラウザからアクセスし、製品評価やトレーニングができる 無償のオンラインサービス です。最新のラインナップ(ラボのシナリオ)は、TechNet バーチャル ラボ サイト、TechNet Flash ニュースレターにて紹介します。

ありがとうございました！

<http://blogs.technet.com/hirookun>

hiroshi.okunushi@microsoft.com

それではボタンタッチします！