

02. Libertyプロフィールの概要

**WebSphere Application Server V8.5.5** **IBM**

**最新情報ワークショップ**

～次世代WASランタイム Libertyで変わるWebシステム～

**Libertyプロフィールの概要**



© 2013 IBM Corporation

1 WAS V8.5.5 最新情報ワークショップ

02. Libertyプロフィールの概要



### Disclaimer

- この資料は日本アイ・ビー・エム株式会社ならびに日本アイ・ビー・エム システムズ・エンジニアリング株式会社の正式なレビューを受けておりません。
- 当資料は、資料内で説明されている製品の仕様を保証するものではありません。
- 資料の内容には正確を期するよう注意しておりますが、この資料の内容は2013年7月現在の情報であり、製品の新しいリリース、PTFなどによって動作、仕様が変わる可能性があるのでご注意ください。
- 今後国内で提供されるリリース情報は、対応する発表レターなどでご確認ください。
- IBM、IBMロゴおよびibm.comは、世界の多くの国で登録されたInternational Business Machines Corporationの商標です。他の製品名およびサービス名等は、それぞれIBMまたは各社の商標である場合があります。現時点でのIBMの商標リストについては、[www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml)をご覧ください。
- 当資料をコピー等で複製することは、日本アイ・ビー・エム株式会社ならびに日本アイ・ビー・エム システムズ・エンジニアリング株式会社の承諾なしではできません。
- 当資料に記載された製品名または会社名はそれぞれの各社の商標または登録商標です。
- JavaおよびすべてのJava関連の商標およびロゴは Oracleやその関連会社の米国およびその他の国における商標または登録商標です。
- Microsoft, Windows および Windowsロゴは、Microsoft Corporationの米国およびその他の国における商標です。
- Linuxは、Linus Torvaldsの米国およびその他の国における登録商標です。
- UNIXはThe Open Groupの米国およびその他の国における登録商標です。

2

© 2013 IBM Corporation  
WAS V8.5.5 最新情報ワークショップ

### Disclaimer(続き)



- MongoDBは10gen (the MongoDB company)の米国およびその他の国における登録商標です。
- FacebookはFacebook, Inc.の米国およびその他の国における登録商標です。
- TwitterはTwitter, Inc.の米国およびその他の国における登録商標です。
- Yahoo!はYahoo! Inc.の米国およびその他の国における登録商標です。
- GoogleはGoogle Inc.の米国およびその他の国における登録商標です。
- LinkedInはLinkedIn Corporationの米国およびその他の国における登録商標です。
- PayPalはPayPal Inc.の米国およびその他の国における登録商標です。
- NetflixはNetflix, Inc.の米国およびその他の国における登録商標です。
- OpscodeはOpscode, Inc.の米国およびその他の国における登録商標です。

© 2013 IBM Corporation

| 02. Libertyプロファイルの概要                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------|
| 目次(前半)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | IBM                                              |
| <ul style="list-style-type: none"><li>■ WAS V8.5.5とLibertyプロファイル</li><li>■ Libertyプロファイルの基本的なメリット<ul style="list-style-type: none"><li>– メモリー使用量、ディスク使用量、構成情報</li></ul></li><li>■ V8.5.5 Libertyプロファイルの新しいメリット<ul style="list-style-type: none"><li>– Web Profile準拠、Liberty集合 (Collective)、Liberty製品拡張</li><li>– エディションによるフィーチャーの違い</li></ul></li><li>■ V8.5.5 Libertyプロファイルのその他の新機能<ul style="list-style-type: none"><li>– MongoDBサポート、OAuth 2.0サポート</li></ul></li><li>■ Libertyの特徴を再確認<ul style="list-style-type: none"><li>– シンプル、軽量、速い...</li></ul></li><li>■ 世の中では...？<ul style="list-style-type: none"><li>– 「ITシステム」の位置づけの変貌</li></ul></li><li>■ 新しい時代に生きるLibertyプロファイル</li></ul> |                                                  |
| 4                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | © 2013 IBM Corporation<br>WAS V8.5.5 最新情報ワークショップ |

02. Libertyプロファイルの概要

IBM

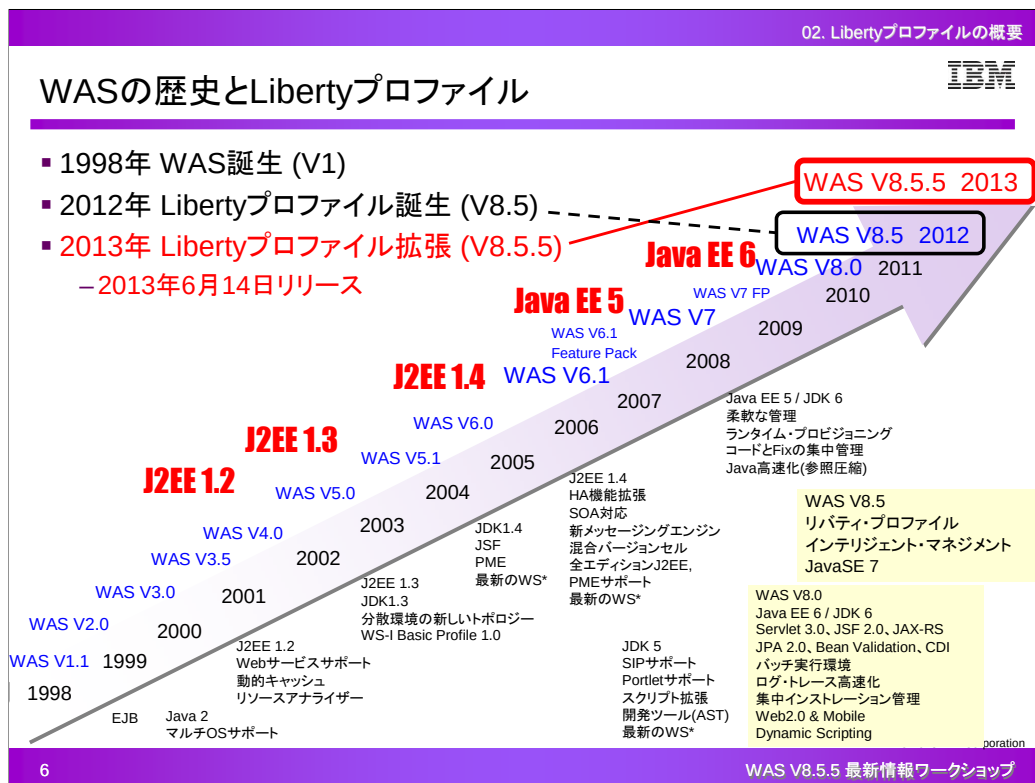
WAS V8.5.5とLibertyプロファイル

© 2013 IBM Corporation

5

WAS V8.5.5 最新情報ワークショップ

## 02. Libertyプロファイルの概要



WASはV1が1998年にリリースされ、今年(2013年)で既に15年の歴史を積み上げてきました。その中で、新しいJava関連標準のサポートと機能の新規追加や更新を重ねてきました。

WAS V8.5は2012年6月にリリースされ、その中で初めてLibertyプロファイルが登場しました。Libertyプロファイルは、新しいタイプのアプリケーション・サーバーです。

WAS V8.5.5は今年(2013年)6月にリリースされ、その中でLibertyプロファイルの機能が大幅に拡張されました。このセッションでは、V8.5.5でのLibertyプロファイル機能の概要を紹介します。

02. Libertyプロファイルの概要

## WAS V8.5.5 とLiberty Coreエディション

IBM

■ V8.5.5の概要

- Liberty Core エディションの追加
- V8.5 に対しては、IBM Installation Managerから上書き適用可能、無償提供 (Refresh Packの位置付け)
- V8.5 系列では、V8.5.0.2 (FixPack 2) の上位FixPackとしての位置付け
  - V8.5.0.2 → V8.5.5.0 → V8.5.5.1 と上げて行く

■ V8.5.5 Liberty Core エディション

- 安価に Libertyプロファイルを提供
- サポートされるフィーチャー (機能単位) は、上位エディションのサブセット

**WAS V8.5 エディション構成**

|                                                                                                                                                  |                                                                                                                           |                                                                                                                             |                                                                                                           |
|--------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|
| <b>WAS for Developers</b><br>Tools +Liberty Edition Profile<br>本番環境のWASで稼動するアプリケーションの迅速な開発をサポート                                                  | <b>WAS Hypervisor Edition</b><br>+Liberty+Intelligent Profile Mgmt<br>VMware, PowerVM, zVM などのサーバー仮想化環境に最適化され、短時間で環境構築が可能 | <b>WAS ND</b><br>Tools +Liberty+Intelligent Edition Profile Mgmt<br>非常に高い可用性、高いパフォーマンス、高度な管理機能を提供。ミッション・クリティカルなアプリケーションに最適。 | <b>WAS for z/OS</b><br>+Liberty+Intelligent Profile Mgmt<br>z/OS Sysplexを活用し、高度なセキュリティ、信頼性を提供。リソース活用を最大化。 |
| <b>WAS (Base)</b><br>Tools +Liberty Edition Profile<br>セキュアでパフォーマンスの高いトランザクション・エンジンを提供。最大で5つのアプリケーション・サーバー・プロファイル間にて、Web層のクラスタリングとフェイルオーバー機能を提供。 |                                                                                                                           |                                                                                                                             |                                                                                                           |
| <b>WAS Express</b><br>+Liberty Profile<br>安価に動的Webアプリケーションを構築可能                                                                                  |                                                                                                                           | <b>WAS CE</b><br>オープンソースベーススモール・フットプリント初期ライセンス・コストが不要                                                                       |                                                                                                           |

**WAS V8.5.5 エディション構成**

|                                                                                                                                                  |                                                                                                                           |                                                                                                                             |                                                                                                                      |
|--------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| <b>WAS for Developers</b><br>Tools +Liberty Edition Profile<br>本番環境のWASで稼動するアプリケーションの迅速な開発をサポート                                                  | <b>WAS Hypervisor Edition</b><br>+Liberty+Intelligent Profile Mgmt<br>VMware, PowerVM, zVM などのサーバー仮想化環境に最適化され、短時間で環境構築が可能 | <b>WAS ND</b><br>Tools +Liberty+Intelligent Edition Profile Mgmt<br>非常に高い可用性、高いパフォーマンス、高度な管理機能を提供。ミッション・クリティカルなアプリケーションに最適。 | <b>WAS for z/OS</b><br>+Liberty+Intelligent+WXS Profile Mgmt Client<br>z/OS Sysplexを活用し、高度なセキュリティ、信頼性を提供。リソース活用を最大化。 |
| <b>WAS (Base)</b><br>Tools +Liberty Edition Profile<br>セキュアでパフォーマンスの高いトランザクション・エンジンを提供。最大で5つのアプリケーション・サーバー・プロファイル間にて、Web層のクラスタリングとフェイルオーバー機能を提供。 |                                                                                                                           |                                                                                                                             |                                                                                                                      |
| <b>WAS Express</b><br>+Liberty Profile<br>安価に動的Webアプリケーションを構築可能                                                                                  |                                                                                                                           | <b>WAS Liberty Core</b><br>Liberty Profile Only (Web Profile)<br>軽量で安価なLiberty Profileに基づくエディション (フル・プロファイルのWASは含まない。)      |                                                                                                                      |

7

© 2013 IBM Corporation  
WAS V8.5.5 最新情報ワークショップ

V8.5.5では新しいエディションが追加されました。Liberty Coreエディションです。Liberty Coreエディションは、従来のWASを含まない、Libertyプロファイルだけのエディションです。位置付けとしては、従来からあるExpressエディションと同様に安価でお手軽に利用できるものです。ただし、Libertyプロファイルの機能としては、他のエディションに含まれるLibertyプロファイルのサブセットの機能を持ちます。エディションにより、Libertyプロファイルの機能に多少の差異があるため、注意が必要です。

エディションによる機能の違いについては、P.21を参照してください。

02. Libertyプロファイルの概要

IBM

Libertyプロファイルの基本的なメリット

© 2013 IBM Corporation

8

WAS V8.5.5 最新情報ワークショップ





V8.5.5のLibertyプロファイルのメリットを図に示しています。たくさんのメリットがありますが、ここでは、赤枠のある3点のメリットについて、確認したいと思います。上から、①メモリ使用量が60MB程度からと小さく、起動時間が5秒程度と短い、②ディスクの使用量も50MB～130MB程度と小さい、③構成がシンプルで、動的構成変更が可能である、の3点です。

次ページ以降で、この3点について詳しく見ていきます。

02. Libertyプロファイルの概要

IBM

## Libertyプロファイルのメモリー使用量は小さい

- Libertyプロファイル自体のメモリー使用量
  - 60MB程度～
- Libertyプロファイルのメモリー使用量が小さい理由
  - 必要な機能単位(フィーチャー)だけを柔軟に組み合わせて利用できるため
- フィーチャーを柔軟に組合せて利用できる理由
  - OSGi フレームワークのサービス・レジストリーを利用するバンドルとして設計されているため
  - バンドルの動的交換が可能のため、**アプリの更新やサーバー構成変更でも再起動が不要**

Libertyプロファイルのフィーチャーはブロックのように組合せて利用することができる。

Libertyプロファイルでのフィーチャー利用

Libertyプロファイルのフィーチャーはサービス・レイヤー(サービス・レジストリー)を利用し、バンドルの動的交換が可能に！

LibertyプロファイルでのOSGi利用イメージ

© 2013 IBM Corporation

10
WAS V8.5.5 最新情報ワークショップ

Libertyプロファイルのメモリー使用量は、WAS自体が利用する部分については、60MB程度からです。このように少しのメモリー使用量で動かせるようになったのは、Libertyプロファイルでは、必要な機能単位(「フィーチャー」と呼びます)だけを柔軟に組み合わせて利用しているからです。

左の図がLibertyプロファイルでのフィーチャー利用のイメージ図です。Libertyのフィーチャーをブロックに喩えて描いています。Libertyでは、まるでブロックを積み上げるがごとく、必要なフィーチャーだけを組み合わせて利用できます。

このようにフィーチャーを組み合わせて利用できるのは、LibertyがOSGiフレームワークを利用しているからです。中でも特に、OSGiのサービス・レイヤー(サービス・レジストリー)を利用していることによる効果が大きいです。OSGiの仕組みは主に3つの要素から構成されています。

右の図にある様に、下から①Moduleレイヤー、②ライフサイクル・レイヤー、③サービス・レイヤーとなっています。Moduleレイヤーは、バンドル(OSGiのモジュールを「バンドル」と呼びます)のバージョン管理、依存関係解決の仕組みを提供します。ライフサイクル・レイヤーは、バンドルの一生における状態(インストール済み→解決済み→開始中→活動中→停止中→解決済み→アンインストール済み)を管理する仕組みを提供します。サービス・レイヤーは、バンドルに含まれるサービス(クラスやインターフェースの単位)を他のバンドルも利用可能なようにするための仕組みを提供します。

従来のWASもOSGiのモジュール・レイヤーやライフサイクル・レイヤーは利用していましたが、サービス・レイヤーは利用していませんでした。Libertyはサービス・レイヤーも利用するため、バンドルの動的交換が可能になりました。このため、アプリケーションの更新やサーバー構成の変更の場合でも、サーバーを再起動する必要がなくなりました。

02. Libertyプロファイルの概要

### Libertyプロファイルのディスク使用量は小さい

- Libertyプロファイルの提供パッケージのサイズ
  - Liberty Core部分(V8.5.5) zip形式: 50MB (展開後: 63MB)
  - Liberty 拡張部分(V8.5.5) zip形式: 30MB (展開後: 34MB)

合計約100MB (提供フィーチャーは今後も追加される可能性が)
- Libertyプロファイルをコンパクトにできる理由
  - 必要なフィーチャーのみを含むパッケージを簡単に作成可能
    - serverコマンドのinclude=minifyオプション
  - 最終的なインストール先では元のサイズよりコンパクトになる

© 2013 IBM Corporation

11WAS V8.5.5 最新情報ワークショップ

Libertyプロファイルが提供しているパッケージは、core部分と拡張部分を足して約100MBです。従来のWASと比べると非常に小さくなっています。ただ、提供されるフィーチャーは今後も追加されてくると予想されます。

しかし、それでもLibertyのディスク使用量は小さく留めることが可能です。それは、Libertyの提供するserverコマンドでは、特定のサーバー構成で利用されているフィーチャーのみをパッケージするオプションが用意されているからです。そのオプションは、`--include=minify` です。minifyは「小さくする」という意味です。

02. Libertyプロファイルの概要

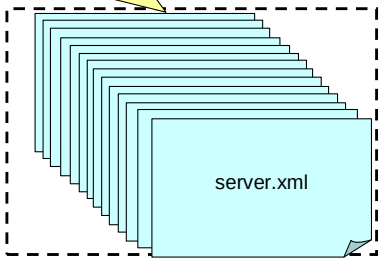
IBM

## Libertyプロファイルの構成情報はシンプル

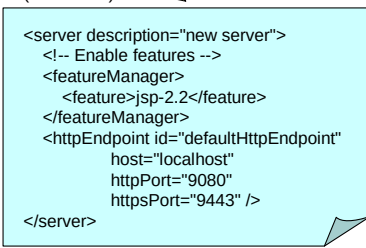
- 起動に必須の構成情報ファイルは server.xml の1個だけ
  - デフォルト値と異なるものだけ記述すればよい(内部デフォルト値を上書きする位置付け)
  - 必要に応じて、他のXMLファイルをincludeすることも可能、他のプロパティファイルで詳細設定も可能(※)  
(※) bootstrap.properties, server.env, jvm.options など、詳細は「Libertyプロファイルによる開発・構成」のセッションで。
- 手作業で編集も可能、開発ツールで生成することも可能
  - 特殊なIDは埋め込まれていない、シンプルなフォーマット
- 構成の動的な更新が可能
  - 構成情報(server.xml)変更後、サーバーの再起動は不要。即時に反映される
  - 前述の通り、OSGiフレームワークのサービス・レジストリーを利用しているため

•構成情報は多数のXMLファイルで構成される  
 •管理コンソール/wsadminツールでのみ変更が可能

•構成情報は1個のXMLファイルで構成が可能  
 •手編集やWDT(開発ツール)での変更が可能



従来WASの構成情報



Libertyプロファイルの構成情報

© 2013 IBM Corporation

12

WAS V8.5.5 最新情報ワークショップ

Libertyプロファイルの構成情報は、1個のXMLファイルで記述することが可能です。それは、server.xmlという名前のファイルで、Libertyが内部に保持しているデフォルト値を上書きしたいものだけを記述すればよい仕組みになっています。右側に例として書かれているserver.xmlファイルは、わずか10行程度のものですが、これはデフォルトで生成されるserver.xmlファイルで、これだけでサーバーを起動することが可能です。もちろん、他のXMLファイルをインクルードしたり、詳細なオプションをプロパティファイル(server.env, bootstrap.properties, jvm.options)で指定することも可能です。

また、Libertyのserver.xmlファイルは、普通のエディターを利用して手作業で編集することもできますし、開発ツール(WDT)で編集することもできます。従来のWASでは、構成情報ファイルは40個以上のXMLファイルで構成され、手作業で編集することはできませんでした。これは、XMLファイルの中にWASが自動で生成する特殊なIDが埋め込まれていたからです。Libertyのserver.xmlファイルには、そのような特殊なIDは使われていません。

また、Libertyの構成情報ファイルの変更後は、その変更が即座にサーバー環境に反映されます。サーバーの再起動は不要です。

02. Libertyプロファイルの概要



V8.5.5 Libertyプロファイルの新しいメリット

© 2013 IBM Corporation

13 WAS V8.5.5 最新情報ワークショップ



前に紹介したメリットの図と同じものですが、赤枠のついている箇所が変えてあります。ここから先は、V8.5.5で追加された機能・更新された機能を中心に、3点を確認していきます。①Web Profile準拠、②Liberty Collective機能、③Liberty製品拡張、の3点です。前に紹介したメリットの図と同じものですが、赤枠のついている箇所が変えてあります。ここから先は、V8.5.5で追加された機能・更新された機能を中心に、3点を確認していきます。①Web Profile準拠、②Liberty Collective機能、③Liberty製品拡張、の3点です。

02. Libertyプロファイルの概要

IBM

## Web Profile準拠 Update

- Web Profile とは
  - Java EE標準中の利用頻度が高い仕様だけを選択したサブセット仕様
  - アプリケーション開発におけるベスト・プラクティスとしての位置付け
  - Web Profile 誕生の背景
    - Java EE標準は約15年の歴史の中で新仕様が追加され続けたが、古い仕様も残された
    - Java EE Full Profileはサイズ(メモリー、ディスク使用量)が大きい

**Java EE 6 Full Profile**

| Java EE 6 Web Profile                                                                                                                |                                                                                                                                | Liberty V8.5.5<br>(coreエディション以外)                                                                                                                                                                      |
|--------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Web App Technologies</b><br>Servlet 3.0<br>JSF 2.0<br>JSPages 2.2 & EL 1.2<br>STL for JSP 1.2<br>Debugging Support 1.0<br>JPA 2.0 | <b>Enterprise App Technologies</b><br>EJB Lite 3.1<br>CD 1.0<br>DI<br>Common Annotations 1.1<br>JTA 1.1<br>Bean Validation 1.0 | JAX-WS 2.2<br>JAX-RS 1.0<br>JMS<br>MDB<br>JNDI(※)<br><br>2.x/1.x CMP/BMP Entity<br>3.0/2.X Remote Home /Component<br>JAX-RPC<br>EJB Timer Service<br>Async Bean Invocation<br>RMI-IIOP interoperation |

Liberty Coreエディションでサポート

Express/Base/NDエディションでサポート

従来WASでサポート

(※)JNDIはWeb Profileには含まれないが、Coreエディションでサポートされる

© 2013 IBM Corporation

WAS V8.5.5 最新情報ワークショップ

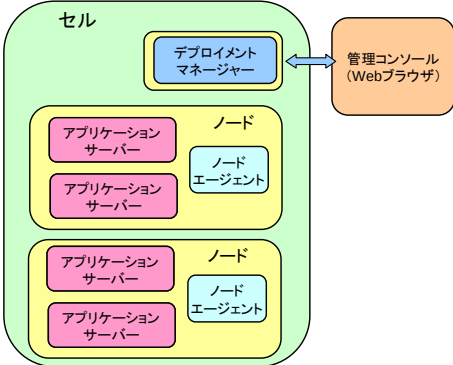
図中の緑色の内側がJava EE 6の全ての仕様です。このうち、黄色の部分がWeb Profileの仕様で、Liberty Coreエディションがサポートしている部分です。Coreエディション以外、つまり、Express/Base/ND等エディションのLibertyプロファイルでは水色の部分の仕様もサポートしています。その外、緑色部分の仕様を利用する場合には、従来WASを利用する必要があります。詳細については、「05. Libertyプロファイルとマイグレーション」を参照してください。

02. Libertyプロファイルの概要

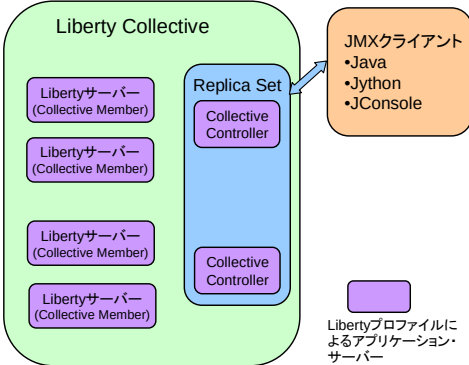
IBM

## Liberty Collective (Liberty集合) New

- WASシステム全体の枠組みは、セルからLiberty Collective (Liberty集合) へ
  - デプロイメント・マネージャーはなくなり、Collective Controllerが各アプリケーション・サーバーを管理する。
    - デプロイメント・マネージャーとは異なり、Collective Controllerはマスター・リポジトリを持たない。 NDのみ
  - ノードの概念はなくなり、ノード・エージェントもない
    - Collective Memberは、ローカルあるいはリモートのCollective Controllerに直接アクセスする。
  - 管理コンソールもない
    - JMXクライアントを作成する (Java、Jython、JConsoleが利用可能)



従来WASのセル構成



V8.5.5 Liberty の Collective構成

© 2013 IBM Corporation

Liberty Collective (Liberty集合)は複数のLibertyサーバーを管理するための仕組みで、従来WASのセルに相当するものです。Collectiveでは、従来のデプロイメント・マネージャーに相当するコンポーネントとして、Collective Controllerがあります。ControllerもLibertyサーバーですが、一般のLibertyサーバーとは異なり、他のメンバーであるLibertyサーバーを管理する役割だけを持ちます。複数のControllerをまとめて、Replica Setとして、高可用性と負荷分散を実現することもできます。Collectiveには、従来のノードやノード・エージェントに相当するものではありません。メンバーであるLibertyサーバーは直接Controllerと通信を行います。V8.5.5のCollectiveには、従来の管理コンソールに相当するコンポーネントは提供されていないため、JMXクライアントを作成する必要があります。JMXクライアントを作成するためには、Java、Jython、Jconsoleが利用可能です。



02. Libertyプロファイルの概要

IBM

### Liberty製品拡張 (1/4) New

- Liberty製品拡張とは
  - 追加フィーチャーをLibertyプロファイルに組み込むこと
    - フィーチャーは自作可能、かつ、Liberty Repository(後述)等からも入手可能
  - wlpディレクトリー構造上に2箇所の拡張ディレクトリーが用意されている

**直接拡張用ディレクトリー**  
wlp/usr/extension/lib/feature上に  
直接フィーチャー置く

**間接拡張用ディレクトリー**  
wlp/etc/extensions上に  
\*.propertiesファイルを置き、  
その中で拡張先ディレクトリーを  
記述する

#### 拡張例

**Liberty Profile install**

**Supergui' install**

**App Services' install**

Liberty製品拡張は、拡張用のディレクトリーとして、直接拡張用ディレクトリーと間接拡張用ディレクトリーの2箇所を持ちます。詳細については、以下リンク先を参照してください。

[http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/cwlp\\_prod\\_ext.html](http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/cwlp_prod_ext.html)

02. Libertyプロファイルの概要

IBM

## Liberty製品拡張(2/4) --- Libertyフィーチャー

- Libertyフィーチャー
  - サーバー構成情報ファイル(server.xml)内で<feature>で指定
- Libertyフィーチャーの構造
  - 1個以上のOSGiバンドルで構成される
    - フィーチャー・マニフェスト・ファイル内で指定される
  - フィーチャー・マニフェスト・ファイル
    - OSGi Enterprise R5 仕様のSubsystem Serviceメタデータ・フォーマット
    - <name>.mf ...製品提供フィーチャーはwlp/lib/features

フィーチャー・マニフェスト・ファイル(<name>.mf)の例

```

Subsystem-ManifestVersion: 1
IBM-ShortName: servlet-3.0
Subsystem-SymbolicName: com.ibm.websphere.appserver.servlet-3.0; visibility:=public
Subsystem-Version: 1.0.0
Subsystem-Content: com.ibm.ws.javaee.servlet.3.0; version="[1,1.0.100]"; location:="dev/api/spec/lib",
com.ibm.websphere.appserver.systemWar-1.0; type="osgi.subsystem.feature",
:
Subsystem-Type: osgi.subsystem.feature
IBM-API-Package: javax.servlet.annotation; type="spec",
javax.servlet.descriptor; type="spec",
:
IBM-Feature-Version: 2
IBM-SPI-Package: com.ibm.wsspi.webcontainer,
com.ibm.wsspi.webcontainer.collaborator,

```

サーバー構成情報ファイル(server.xml)の例

```

<server description="new server">
  <!-- Enable features -->
  <featureManager>
    <feature>servlet-3.0</feature>
    <feature>collectiveMember-1.0</feature>
    <feature>restConnector-1.0</feature>
  </featureManager>

```

(ファイル・フォーマットの詳細は以下リンク先参照)  
[http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/rwlp\\_feat\\_definition.html](http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/rwlp_feat_definition.html)

Subsystem-Content  
構成するOSGiバンドルを指定する

IBM-API-Package  
アプリケーション、他のフィーチャー、Libertyカーネルに公開されるパッケージを指定する

IBM-SPI-Package  
他のフィーチャー、Libertyカーネルに公開されるパッケージを指定する(アプリケーションには公開されない)

18
WAS V8.5.5 最新情報ワークショップ

Libertyのフィーチャーは、フィーチャー・マニフェスト・ファイルにより定義されます。フィーチャーは1個以上のOSGiバンドルで構成され、API、SPIを公開できます。詳細については以下リンク先を参照してください。

[http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/rwlp\\_feat\\_definition.html](http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/rwlp_feat_definition.html)

02. Libertyプロファイルの概要

### Liberty製品拡張 (3/4) --- SPI (System Programming Interface)

- Liberty API と Liberty SPI
  - アプリケーションはAPIだけを利用可能
  - フィーチャーはAPI, SPIともに利用可能
- フィーチャーを作成するメリット
  - SPIの利用が可能
    - ・ Libertyランタイムとの連携が容易になる
  - フィーチャー・マネージャーでの管理が可能
    - ・ server.xmlでのパラメーター設定が可能になる
  - 設定項目を開発ツール(WDT)に公開が可能
    - ・ 開発ツール(WDT)の専用エディターでserver.xmlの設定が可能になる
- Liberty SPIで実現可能なこと
  - セキュリティー機能のカスタマイズ
    - ・ カスタム・ユーザー・レジストリー・・・ユーザー認証のロジック実装が可能
    - ・ トラスト・アソシエーション・インターセプター (TAI)・・・3rdパーティの認証サーバーとのSSO構成が可能
  - 利用可能なサービス／ツールの例
    - ・ ファイル・モニター・・・ファイル・システム上のファイル変化をチェックするサービス
    - ・ Liberty Collective リポジトリー・・・Collective Controller リポジトリーへのデータ公開
    - ・ タイムド・オペレーション・・・サーバー上での実行時間の計測とパターン検出

(参照) <http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.javadoc.liberty.doc/about:blank>

© 2013 IBM Corporation

19
WAS V8.5.5 最新情報ワークショップ

アプリケーションではAPIしか利用できませんが、フィーチャーを作成する場合にはSPIも利用することができます。フィーチャーを作成すると、公開するAPI,SPIを定義できます。

カスタム・ユーザー・レジストリーについては、以下リンク先を参照してください。

[http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.nd.multiplatform.doc/ae/tsec\\_users.html](http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.nd.multiplatform.doc/ae/tsec_users.html)

トラスト・アソシエーション・インターセプターについては、以下リンク先を参照してください。

[http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.nd.multiplatform.doc/ae/twlp\\_dev\\_custom\\_tai.html](http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.nd.multiplatform.doc/ae/twlp_dev_custom_tai.html)

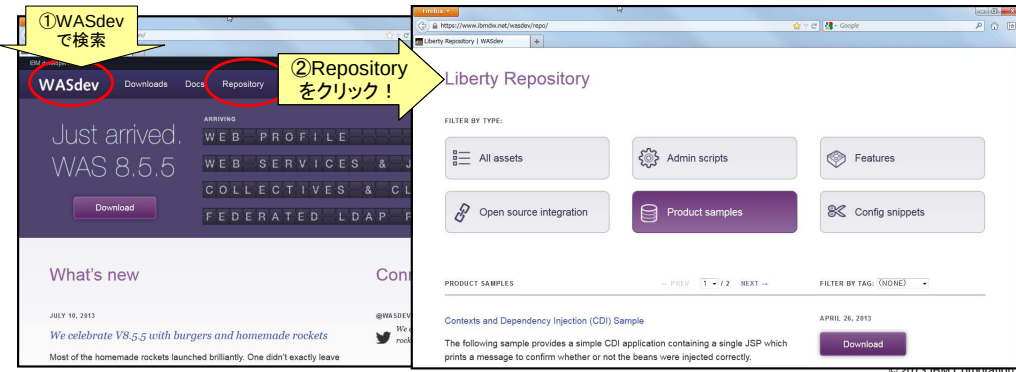
02. Libertyプロファイルの概要

IBM

### Liberty製品拡張(4/4) --- Liberty Repository

- WASdevサイト中のLiberty専用オンライン・リポジトリ  
[https://www.ibmdev.net/wasdev/repo/#category:osi\\_tag:all\\_page:1](https://www.ibmdev.net/wasdev/repo/#category:osi_tag:all_page:1)
  - Libertyサンプル・アプリケーション
  - 管理スクリプト(Jython)のサンプル
  - 構成情報ファイル(server.xml)のサンプル
  - Libertyフィーチャーの個別リリース
  - オープンソース統合
    - Struts, Spring, Hibernate等のオープンソース・ソフトウェア用のサンプル・アプリケーション
  - 拡張サイトWASdev Github --- <http://wasdev.github.io>

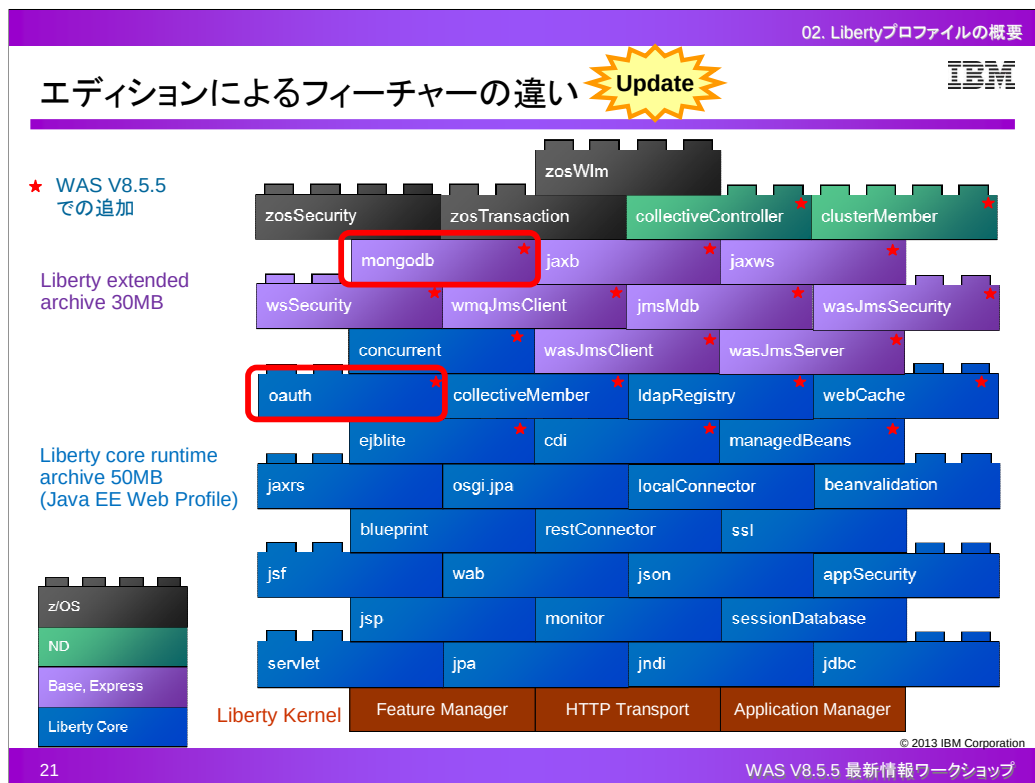
(注意) Liberty Repositoryからダウンロードしたコードには製品サポートはありません。



20WAS V8.5.5 最新情報ワークショップ

WASDevはWASの開発部門がユーザー向けに様々な情報を公開している英語サイトです。この中にLiberty Repositoryのページがあり、Libertyで利用可能なサンプル・アプリ、サンプル・スクリプト等の情報が入手できます。Libertyフィーチャーもダウンロードすることができます。ただし、サポートはありませんので、ご注意ください。

## 02. Libertyプロファイルの概要



図中の一番下の段(茶色)はLibertyカーネルと呼ばれる機能単位で、一般のフィーチャーとは異なります。それ以外の、青色(Core)、紫色(Base, Express)、緑色(ND)、黒色(z/OS)のブロックは、それぞれのエディションで追加されるフィーチャーを示しています。赤い星のついているフィーチャーは、V8.5.5で追加されたものです。赤枠の2個のフィーチャー、monodb, oauthについては、次以降のページで紹介します。

02. Libertyプロファイルの概要

IBM

V8.5.5 Libertyプロファイルのその他の新機能

© 2013 IBM Corporation

22

WAS V8.5.5 最新情報ワークショップ

02. Libertyプロファイルの概要

MongoDBサポート 

IBM

- MongoDBとは  
オープンソース・ソフトウェアで、以下の特徴を持つ
  - NoSQL データベース
    - Not only SQL (SQLだけではない)
    - RDBとは異なり、データ操作にSQL以外が利用可能
  - 文書データベース
    - DB、コレクション、オブジェクト(文書)単位でデータを管理
    - 文書でデータを格納
      - MongoDBでは、文書はJSON形式
      - RDBのようなテーブル形式スキーマを持たない
  - 自動シャーディング
    - 複数サーバーへデータを分散することによる負荷分散
    - 並列処理による検索も可能
  - レプリケーション機能
    - 高可用性構成が容易に実現可能

MongoDBはNoSQL文書データベースのデファクト・スタンダード

- LibertyプロファイルのMongoDBサポート
  - LibertyプロファイルはMongoDBクライアント機能(mongodb-2.0)を提供
  - 詳細は「Libertyプロファイルによる開発・構成」セッションで

Usersコレクションに含まれる文書の例

```
{
  "username": "bob",
  "address": {
    "street": "123 Main Street",
    "city": "Springfield",
    "state": "NY"
  }
}
```

照会の例 `db.users.find({"address.state": "NY"})`

23

© 2013 IBM Corporation  
WAS V8.5.5 最新情報ワークショップ

MongoDBはオープンソース・ソフトウェアであり、上記の4つの特長があります。このため、NoSQL文書データベースのデファクト・スタンダードとされています。V8.5.5のLibertyプロファイル(Coreエディション以外)はMongoDBをサポートしています。MongoDB利用方法の詳細については「03. Libertyプロファイルによる開発・構成」を参照してください。

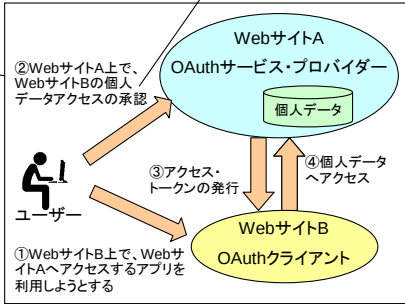
02. Libertyプロファイルの概要

IBM

### OAuth 2.0 サポート New

- OAuthとは
  - Web経由でのAPIアクセスを第三者プログラムに安全に提供するためのセキュリティ認可の仕組みを定めた標準仕様
    - <http://tools.ietf.org/html/draft-ietf-oauth-v2-30>
  - 多くのWebサイト／ネット企業／製品での採用実績
    - Facebook, Twitter, Yahoo!, Google, LinkedIn, Microsoft, Paypal, Netflix, SmartCloud, Lotus Connections 等
  - 従来WASでもOAuth 2.0をサポート
    - V7.0.0.25～、V8.0.0.5～、V8.5.0.1～
  - OAuthの仕組み概要
  - OAuthのメリット
    - 第三者プログラムにはユーザーのアカウント・パスワードを知られずに権限(アクセス・トークン)を付与できる。
- Liberty OAuth-2.0フィーチャー
  - OAuth サービス・プロバイダー (SP) 機能
  - 複数のサービス・プロバイダーが共存可能
  - パーシスタントOAuth 機能をサポート
  - 4 種の標準OAuth 2.0 フローをサポート(※)

(※)①Authorization Code Grant、②Implicit Grant、③Resource Owner Password Credentials Grant、④Client Credentials Grantの各フロー

OAuthの仕組み概要

© 2013 IBM Corporation

OAuthは多くのWebサイト等で既に利用されている標準仕様です。従来WASでも既にOAuth 2.0をサポートしています。V8.5.5のLibertyプロファイルはOAuth 2.0をサポートしています。詳細については、以下リンク先を参照してください。

[http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/cwlp\\_oauth\\_introduction.html](http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.core.doc/ae/cwlp_oauth_introduction.html)



02. Libertyプロファイルの概要

IBM

目次(後半)

- WAS V8.5.5とLibertyプロファイル
- Libertyプロファイルの基本的なメリット
  - メモリー使用量、ディスク使用量、構成情報
- V8.5.5 Libertyプロファイルの新しいメリット
  - Web Profile準拠、Liberty集合 (Collective)、Liberty製品拡張
  - エディションによるフィーチャーの違い
- V8.5.5 Libertyプロファイルのその他の新機能
  - MongoDBサポート、OAuth 2.0サポート
- Libertyの特徴を再確認
  - シンプル、軽量、速い……
- 世の中では……？
  - 「ITシステム」の位置づけの変貌
- 新しい時代に生きるLibertyプロファイル

© 2013 IBM Corporation

25

WAS V8.5.5 最新情報ワークショップ

02. Libertyプロファイルの概要

IBM

Libertyの特徴を再確認

© 2013 IBM Corporation

26

WAS V8.5.5 最新情報ワークショップ

ここからは、どのようなケースでLibertyプロファイルの特徴を生かせるのか、世の中の動向と併せながら見ていきたいと思います。

まずは、Libertyプロファイルの特徴を再確認してみましょう。



Libertyプロファイルには沢山の特徴がありますが、従来のWAS Fullプロファイルと比較して

- ・軽量・高速
- ・構成がシンプルであり、管理が容易
- ・動的な構成変更が可能

という点が際立った特徴と言えます。

では、これがどのような場面、用途で役立つか、幾つか例示して紹介したいと思います。

02. Libertyプロフィールの概要



# 世の中では・・・？

---

## 「ITシステム」の位置づけの変貌

© 2013 IBM Corporation

28

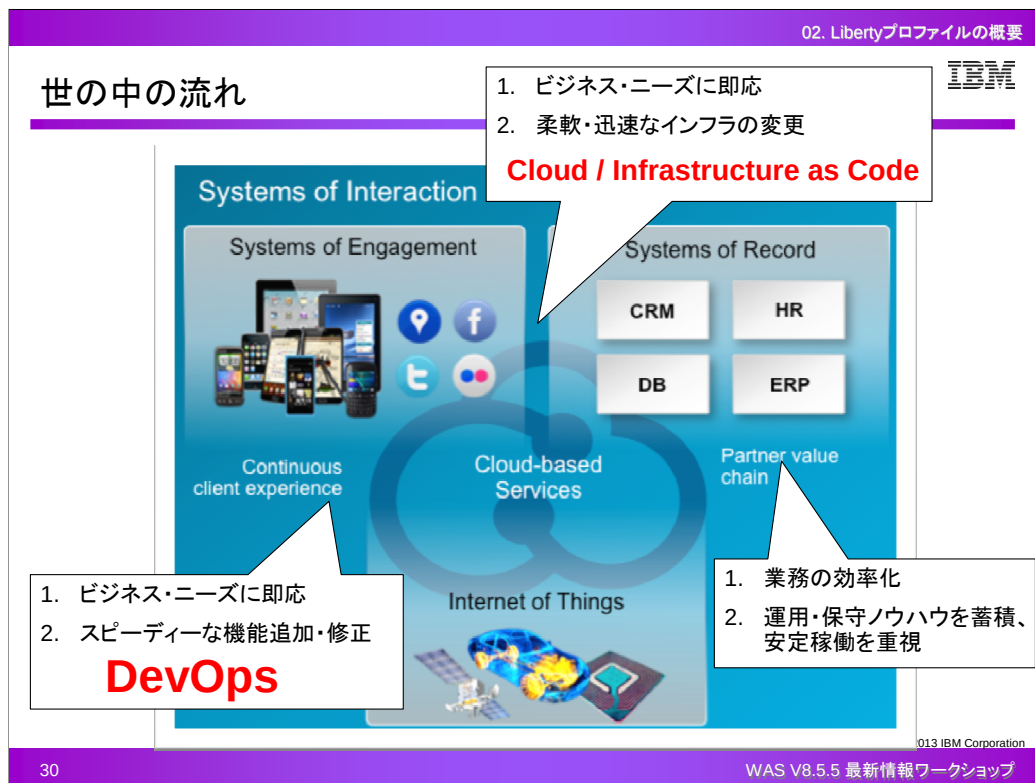
WAS V8.5.5 最新情報ワークショップ

Libertyの特徴が活きる状況・ユースケースを考えるにあたって、まずは世の中ではITシステムがどのように位置づけられているか確認します。



従来のITシステムは、ITを使用して業務を効率化する、という目的で構築・利用されているケースが多くありました。そこでは、データの一貫性や厳密さに主眼が置かれ、運用においては如何に安定稼働を維持するかというのが専らの課題でした。このようなシステムをSystems of Recordと呼ぶ向きもあります。このようなタイプのシステムは既に成熟しており、もはや企業の競争力の源泉とは言えなくなってきました。

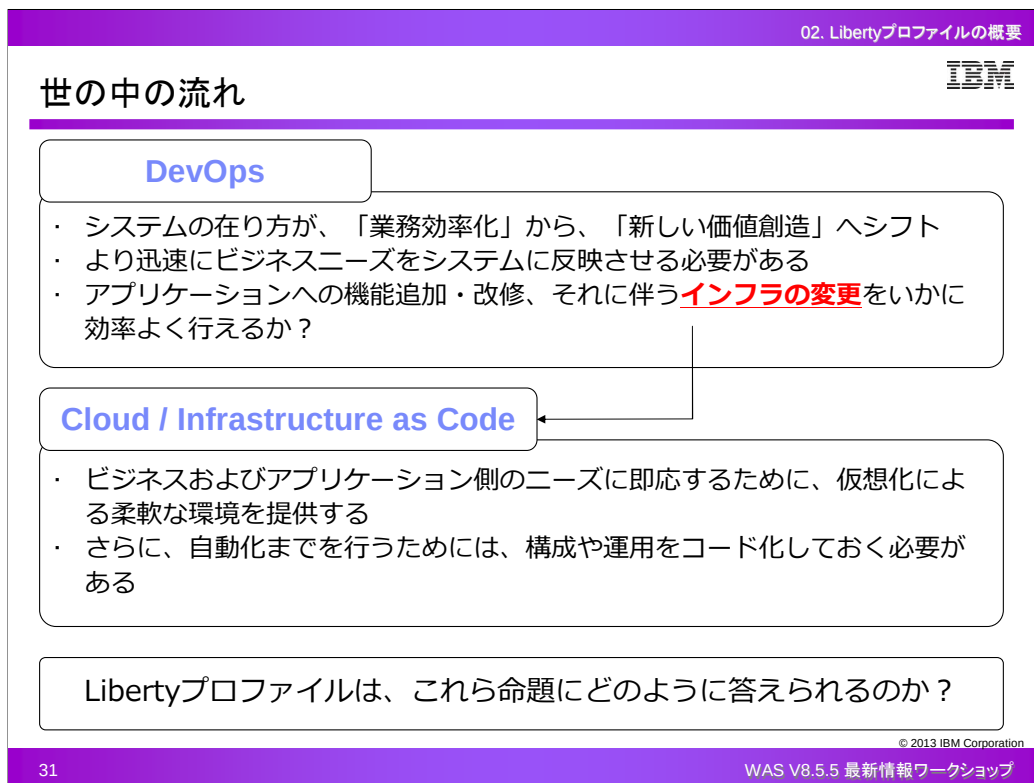
近年ではITが果たす役割は業務の効率化ではなく、新しいテクノロジーを取り込みIT自体をビジネスの武器として適用する、という位置づけに変わってきていると言えます。ITシステム・運用の命題が「如何に安定稼働させるか」ではなく、「如何に新しいビジネスに柔軟に対応できるか」が鍵になると、アプリケーション開発、インフラ構築・運用のあり方も当然変わる必要があります。その基盤として注目されているのがCloudベースのサービスという訳です。



このような流れの中で、アプリケーション開発から運用までを一体化し、短期間の開発サイクルを継続的に実施しようとする、「DevOps」と言う概念が生まれます。(DevOpsと総称していますが、DevOpsには明確な定義はなく、個々の技術の総称をさしたり、開発と運用の担当者間の文化の融合といった意味合いで使われることもあるようです。)

一方で、迅速なアプリケーションの開発、デプロイに対応するためには、インフラもそれに対応できていなくてはなりません。Cloudベースのサービスはまさにその実現に適しています。IaaS、Paasで迅速にインフラ環境を配備し、ビジネス要件・アプリケーション要件に対応します。そのためには、インフラ構築が標準化され、物理的制約から解放される為に仮想化され、更には頻繁な変更に対応できるように自動化されている必要があります。

インフラ構成や構築手順、運用がコードとして管理され、その変更が自動的にインフラ構成に反映させる事ができるという状態にしておく必要がある訳です。このような考え方は「Infrastructure as Code」と呼ばれています。



ここまで、世の中の動向としてITシステムの在り方の変化について記述してきました。上記はそのサマリになります。

では、Libertyプロファイルは、このような新しいシステムにおける命題に、どのように応える事ができるのでしょうか。

02. Libertyプロファイルの概要

IBM

新しい時代に生きるLibertyプロファイル

DevOps

Cloud / Infrastructure as Code

© 2013 IBM Corporation

32

WAS V8.5.5 最新情報ワークショップ



02. Libertyプロファイルの概要



# 新しい時代に生きるLibertyプロファイル

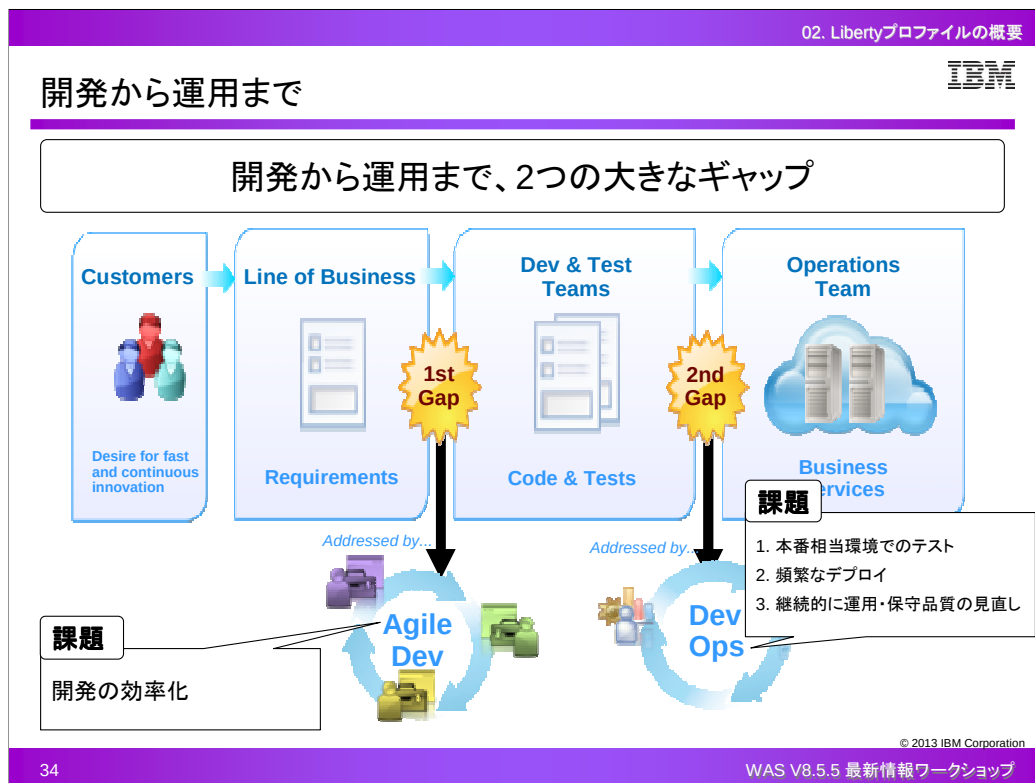
 DevOps

Cloud / Infrastructure as Code

33

© 2013 IBM Corporation  
WAS V8.5.5 最新情報ワークショップ

まずは、DevOpsから見ていきます。



この図は、顧客の要望からシステムへの要件発生、開発・テスト、最終的に運用に乗るまでの流れを模式的に示したものです。この流れの中で、大きなギャップが2つあるといわれています。

1つ目は、要件が挙がってからアプリケーションを開発しテストする流れ、2つ目は、開発・テストが終わってから運用に乗るまでの間です。

1つのめのギャップについてはアジャイル開発が、2つのギャップについてはDevOpsがギャップを埋めるための有効なソリューションであると考えられます。

それぞれのソリューションでは何を解決しなくてはならないのか、を順番に考えてみます。

02. Libertyプロファイルの概要

IBM

## 開発からデリバリーまで・・・解かなくてはならない課題

### アジャイル開発

- 頻繁なコード変更・テスト・デプロイに耐えうる環境の提供
- 分散開発における統一された開発環境の共有

### DevOps

- 本番相当環境でのテスト
  - 開発 / 本番環境等、複数環境の構成管理・同期
  - インフラ環境への変更反映スピード、正確さの確保
- 頻繁なデプロイ
  - 開発されたコードの統合・ビルドの迅速化、正確性の確保
  - デプロイに割くメンテナンス時間の極小化
- 継続的な運用・保守品質の見直し

© 2013 IBM Corporation

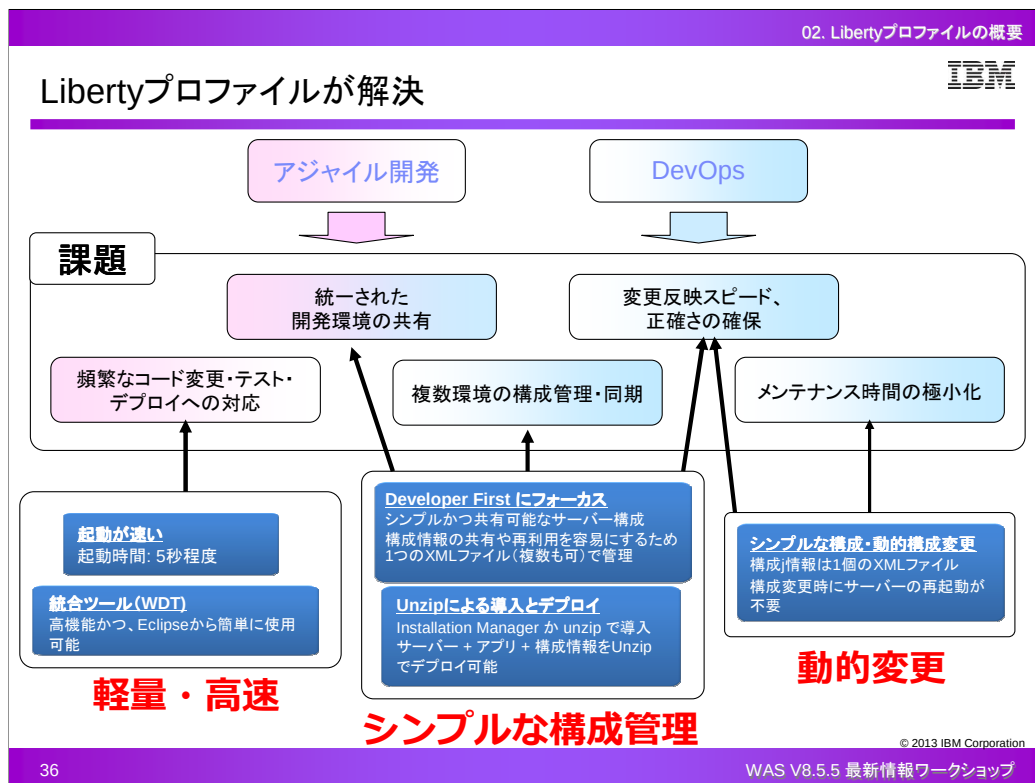
35
WAS V8.5.5 最新情報ワークショップ

まず、アジャイル開発についてですが、(アジャイル開発に限らないかもしれませんが)開発者はコーディング作業をする中で実際に頻繁にコードを変更し、コードの動作確認をします。これまでのWASが提供していたフルプロファイルだと、起動・停止に数分かかっていました。この時間が短縮されると非常に効率が図れる筈です。高速起動・停止は、機能的側面から見れば瑣末かもしれませんが、開発者にとっては非常に嬉しい事でしょう。

また、分散開発を行う際には、統一された開発環境を構築するという点もまた重要な観点です。作成したアプリケーションの実行環境が違えば、コードの実行結果やコードの内容にまで影響を及ぼす可能性もあります。

DevOpsについては、CI(Continuous Integration)やCD(Continuous Delivery)など、コードのビルド・マージからテスト・リリースまでを自動化して効率化・高速化をはかりますが、インフラ環境においても同様に構成変更を自動化する必要があります。ここでのポイントはテストの流れです。単体テスト、結合テスト、統合テストなど、複数環境でのテストを経て本番環境へのリリース(これは多少冗長な流れかもしれませんが)、という流れになります。これらの環境設定に差異があると、当然テスト結果の妥当性にも問題が生じます。従って、複数の環境間で環境設定・パラメータの一貫性を常に保つということは非常に重要な要件となります。

また、実際に本番環境へデプロイを行う場合には、リリースの規模によってはサービス全体を停止させるケースもままあります。この停止時間を最小限に抑えるという事も重要な課題です。



これらの要件・課題に、Libertyプロファイルが持つ特長で応える事が出来ます。

開発者の効率向上のためには、高速起動やEclipseに統合された開発ツールが役に立ちます。

また、サーバー構成込みでランタイム自身をアーカイブ・配布することで構成が完了しますので、統一された開発環境の整備にも役立てることが出来ます。

さらに、サーバー構成が基本的には1つのファイルで完結していることから非常に管理がしやすく自動化ツールとの親和性も高いことから、複数の環境管理にも威力を発揮します。

さらに、フィーチャーの追加など、構成の変更も動的に反映されることから、デプロイや構成変更のサービス停止時間も劇的に短縮できます。

02. Libertyプロファイルの概要

IBM

新しい時代に生きるLibertyプロファイル

DevOps

➡ Cloud / Infrastructure as Code

© 2013 IBM Corporation

37

WAS V8.5.5 最新情報ワークショップ

次に、Cloud環境、および、Infrastructure as code を考えた場合にLibertyがどう生きてくるのか、について記述します。

02. Libertyプロフィールの概要

IBM

時代はオープンクラウドへ？

@IT > Server & Storage > ベンダロックインのないクラウドの第2ステージへ：オープンク...

2013年05月13日 18時53分 更新

ベンダロックインのないクラウドの第2ステージへ：

オープンクラウドに舵を切るIBM、OpenStack/TOSCA  
でIaaS/PaaSの相互運用性を強化

日本IBMは5月13日、同社製品・サービスに対するオープンクラウドの積極的展開に向けた記者説明会を開催した。クラウド間の相互運用性を高めていく狙いだ。

[藤本和彦, @IT]

Atmark IT: <http://www.atmarkit.co.jp/ait/articles/1305/13/news117.html>

TOSCA

⇒

PaaS環境構築のための標準仕様  
<http://docs.oasis-open.org/tosca/TOSCA/v1.0/TOSCA-v1.0.html>

OpenStack

⇒

オープンソースのIaaS環境構築ツール/プロジェクト  
<http://www.openstack.org/>

インフラの構成、構築の手順が標準化され、ツールによって自動化されることになる

© 2013 IBM Corporation

38

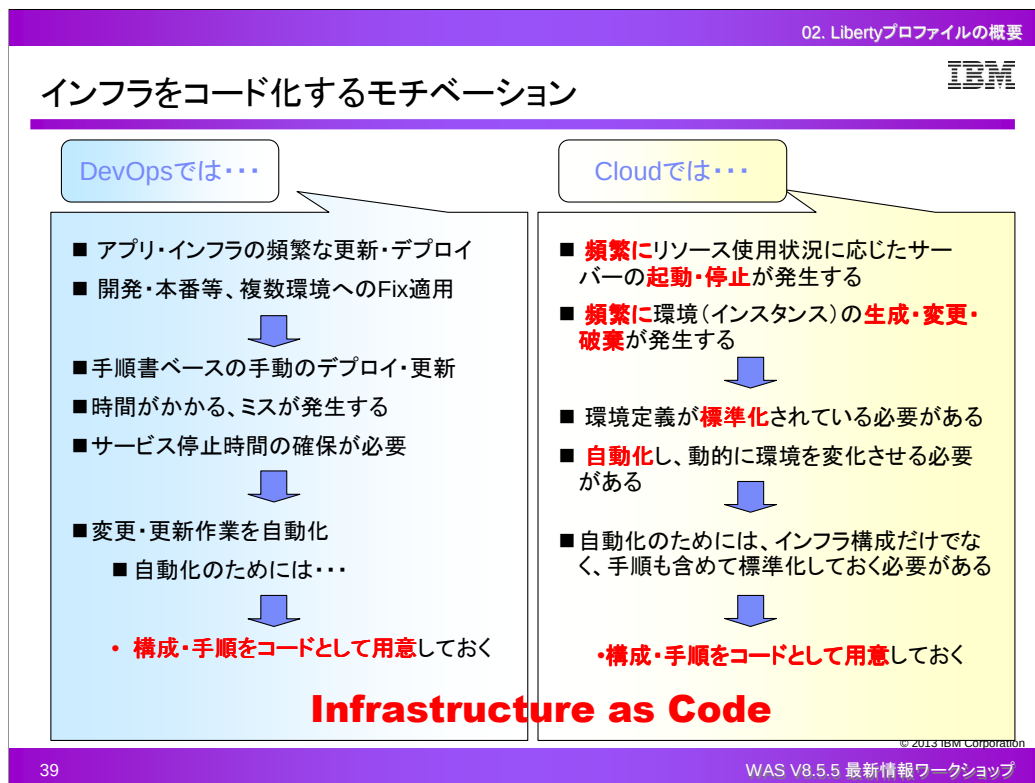
WAS V8.5.5 最新情報ワークショップ

Cloud環境はこれまで各社が独自の製品・サービスをベースにして展開してきましたが、標準化が進められ、これに追従する流れがあります。上記はそれを端的に示す記事です。

オープンソースなどでもCloud環境の構築が進めば、これまでより一層、標準化・仮想化・自動化の流れが加速することになります。

WAS V8.5.5 最新情報ワークショップ

38



そのためには、これまでのインフラの構築・運用のあり方を見直す必要があります。これまでの、手順書を作成し、レビューし、構成変更を行って、エビデンスを取得して・・・という流れを自動化するという発想が生まれます。DevOpsなどを考えた場合、日に数回というリリースがありうることを考えれば、自動化は必須となるでしょう。

そのためには、インフラ構成・運用をコード化してしまう必要があります。

02. Libertyプロファイルの概要

OpscodeとIBMがChefのLibertyプロファイル対応をアナウンス

**OPSCODE**

**抜粋**  
Companies to Deliver Cookbooks For Automating Deployment of  
**WebSphere Application Server Liberty Profile**

[< Previous Post](#) [Next Post >](#)

### Opscode and IBM Join Forces to Bring Open Source Cloud Automation to the Enterprise

Posted on April 25, 2013 by Lucas Welch — No Comments ↓

This morning, we announced an exciting collaboration with our friends at IBM to bring the full power of Opscode Chef to IBM's enterprise customers. From Opscode Chef integration with IBM SmartCloud, to new cookbooks, content and more, we're working closely with IBM to deliver on the promise of DevOps in accelerating the speed of business.

Please read on for the full announcement.

**Opscode and IBM Join Forces to Bring Open Source Cloud Automation to the Enterprise**

*IBM to Support Opscode Chef™ Server In IBM SmartCloud; Opscode Chef to provide automation for configuring IBM AIX; Companies to Deliver Cookbooks For Automating Deployment of WebSphere Application Server Liberty Profile*

**Archives**  
Select Month

**Categories**

- announcements
- awesome chefs
- chef
- chefconf
- cloud
- community
- cookbooks
- culture

OSSの環境構築自動化ツールであるchefがLibertyプロファイル対応  
インフラコード化は確実に進んでいる！？

<http://www.opscode.com/blog/2013/04/25/opscode-and-ibm-join-forces-to-bring-open-source-cloud-automation-to-the-enterprise/>

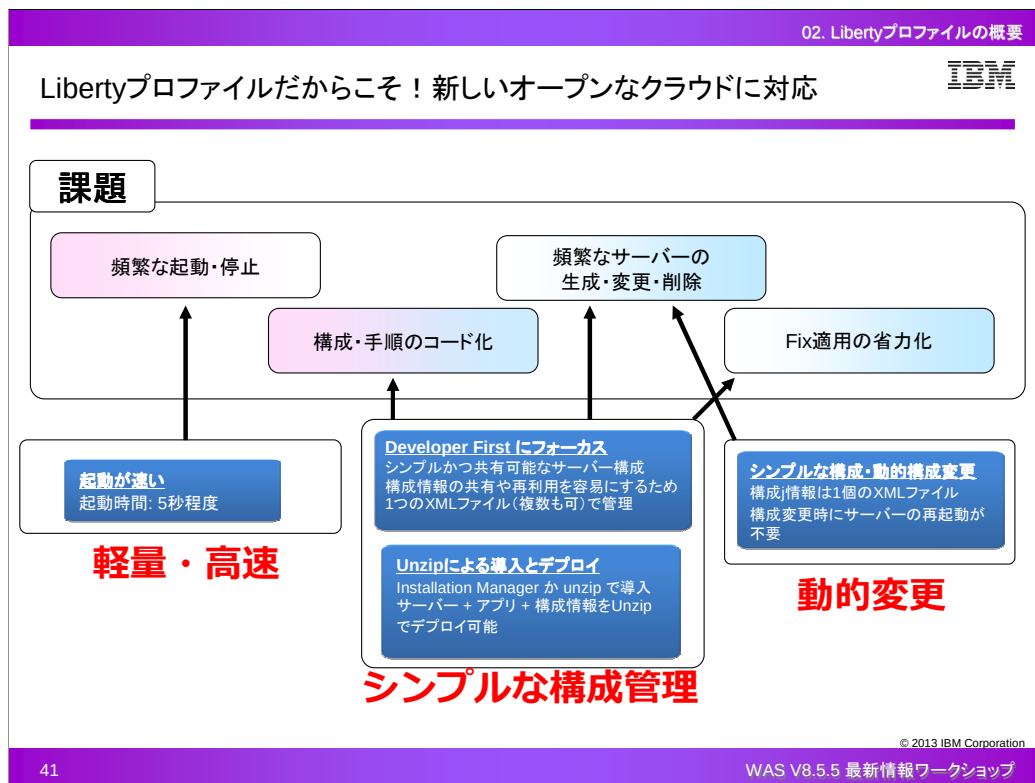
© 2013 IBM Corporation

40

WAS V8.5.5 最新情報ワークショップ

インフラの構築・管理の自動化ツールとして、chefが有名ですが、上記記事にもあるとおり、ChefがLibertyプロファイルに対応するというアナウンスがされています。





このように、クラウド環境を見越して、DevOpsを勧めていく中ではインフラのコード化が重要になってきますが、この点でもLibertyプロファイルは効果的です。

クラウド環境では、負荷などに応じて動的なサーバー追加、起動・停止が発生しますが、軽量・高速であり、可搬性に優れるLibertyプロファイルは正にこのような用途に向いていると言えます。

さらに、構成が数ファイルで完結するという点は、自動化ツールとの親和性が高いという事が言えます。

Libertyプロファイルは、まさに新しい時代のシステムに適合した新しいWASである、という事がいえます。

### まとめ



- Libertyプロファイルは、これからのシステムの基盤として大きな可能性を秘めている。
- 例として、DevOps、Cloud、Infrastructure as Code などとの親和性が挙げられる。
- そこで生きるのは、Libertyプロファイルの「**軽量・高速**」、「**シンプルな構成管理**」、「**動的な変更への対応**」という特徴

02. Libertyプロファイルの概要

IBM

END

© 2013 IBM Corporation

43

WAS V8.5.5 最新情報ワークショップ