

WAS最新情報

Agenda

1. WAS V9 発表概要
2. WAS V9 traditional 新機能ハイライト
3. WAS 7 / 8, Java 6 EOS
4. WAS関連ソリューション

参考情報

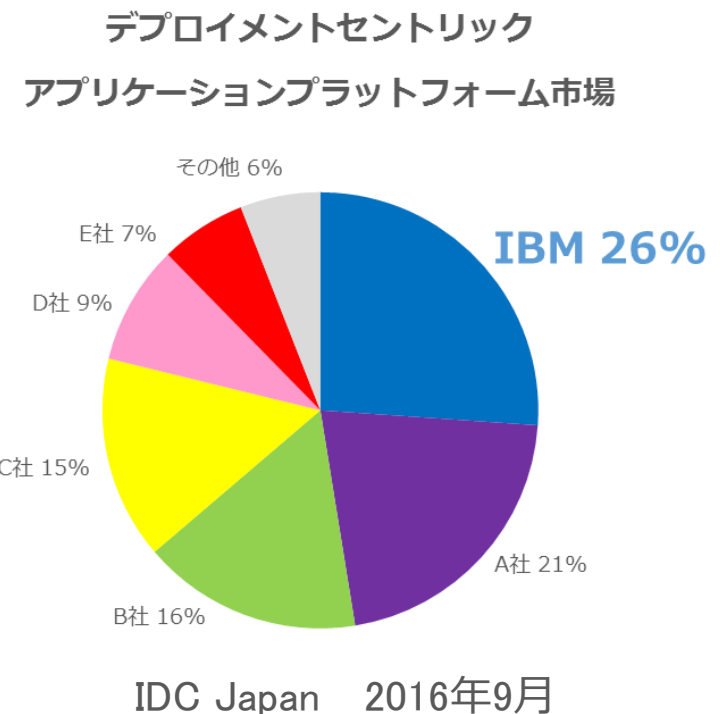
1. WAS V9 発表概要

WebSphere Application Serverとは

- WebSphere Application Server (WAS)は,
Java EE 仕様に従って作成されたエンタープライズ・アプリケーション
を実行するプラットフォーム

◆ 8年連続国内シェア No.1 !!

- 業界標準技術への対応
- 信頼性・管理機能の強化
- 製品戦略に基づく一貫した機能拡張



*出典 : IDC Japan JPJ40590016

2016年9月「国内アプリケーションデプロイメント／構造化データ管理ソフトウェア市場シェア、2015年：激化するパブリッククラウドの覇権争い」

WAS V9 ～18年の歴史と進化～

■ WAS V9登場！！

- ◆ 2016/6/7 発表
 - 発表レター: <https://ibm.biz/Bd4Nr3>
- ◆ 2016/6/24 ダウンロード開始

Java EE 7

WAS V9

WAS V8.5.5.6

2016

WAS V8.5.5

2015

WAS V8.5

2014

2013

2012

2011

2010

2009

2008

2007

2006

2005

2004

2003

2002

2001

2000

1999

1998

Java EE 5

WAS V7

WAS V6.1
Feature Pack

2009

WAS V8.0

Java EE 6 / JDK 6
バッチ実行環境
ログ・トレース高速化
集中インストレーション管理

WAS V7

Java EE 5 / JDK 6
柔軟な管理
ランタイム・プロビジョニング
コードとFixの集中管理
Java高速化(参照圧縮)

WAS V6.0

J2EE 1.4
HA機能拡張
SOA対応
新Messaging Engine
最新のWS*

WAS V6.1

JDK 5
SIPサポート
開発ツール(AST)
最新のWS*

WAS V5.1

JDK1.4
JSF
PME
最新のWS*

J2EE 1.4

WAS V6.0

WAS V5.1

J2EE 1.3

WAS V5.0

WAS V5.0

J2EE 1.3
JDK1.3
分散環境の新トポロジー
WS-I Basic Profile 1.0

WAS V4.0

J2EE 1.2
Webサービスサポート
動的キャッシュ
リソースアナライザー

WAS V3.0

EJB
Java 2
マルチOSサポート

J2EE 1.2

WAS V4.0

WAS V3.5

WAS V3.0

WAS V2.0

WAS V1.1

WAS V1.0

WAS V9 登場！！

Java EE 7, Java 8対応
API機能強化
API Connectを同梱
クラウド対応強化

WAS V8.5.5

Liberty Core Edition提供
WXSをBASE/NDに同梱
WebサーバーPluginでの
インテリジェント管理

WAS V8.5.5.6

WAS Liberty
Java EE 7対応

WAS V8.5

Libertyプロファイル
インテリジェント管理
JavaSE 7

WAS が提供する 2 つのランタイム

- “traditional” と “Liberty” の2つのランタイムを提供
 - ◆ “traditional” はこれまでの “フル・プロファイル”と同じ

WAS エディション	提供されるWASランタイム	
WebSphere Application Server Liberty Core	WAS Liberty Java EE7 Web profile	
WebSphere Application Server Base	WAS Liberty Java EE7 完全対応	and WAS traditional Java EE7 完全対応
WebSphere Application Server Network Deployment (& z/OS)	WAS Liberty Java EE7 完全対応 + 拡張管理	and WAS traditional Java EE7 完全対応 + 拡張管理

用途に応じたランタイムの選択

- WAS traditional

- ◆ **既存資産の活用**を目的としたランタイム
 - ◆ 旧WASでおこなっていた**従来の運用を継続**したいお客様
- ◆ WAS Libertyで対応していない
 - 旧API**を使用しているアプリケーションの実行環境として
 - ◆ JAX-RPCやEntity Bean, CommonJ など

- WAS Liberty

- ◆ モダンなアプリケーション開発・サーバー運用に対応した**新時代のランタイム**
 - ◆ 軽量さを活かしたAgile開発やCD(継続的デリバリー)
 - ◆ ツールによる運用の自動化・Platform as a Code
- ◆ **クラウド環境**での使用やリソースの限定されたIoT環境での使用

WAS traditional の Java SE / EEのサポート

WAS V6.1	J2EE 1.4 Servlet 2.4/JSP 2.0 EJB 2.1				J2SE 5.0
WAS V7.0	J2EE 1.4 Servlet 2.4/JSP 2.0 EJB 2.1	Java EE 5 Servlet 2.5/JSP 2.1 EJB 3.0			Java SE 6
WAS V8.0	J2EE 1.4 Servlet 2.4/JSP 2.0 EJB 2.1	Java EE 5 Servlet 2.5/JSP 2.1 EJB 3.0	Java EE 6 Servlet 3.0/JSP 2.2 EJB3.1 / JAX-RS1.0		Java SE 6
WAS V8.5	J2EE 1.4 Servlet 2.4/JSP 2.0 EJB 2.1	Java EE 5 Servlet 2.5/JSP 2.1 EJB 3.0	Java EE 6 Servlet 3.0/JSP 2.2 EJB3.1 / JAX-RS1.0		Java SE 6/7/8 (*1)
WAS V9.0		Java EE 5 Servlet 2.5/JSP 2.1 EJB 3.0	Java EE 6 Servlet 3.0/JSP 2.2 EJB3.1 / JAX-RS1.0	Java EE 7 Servlet 3.1/JSP 2.3 EJB3.2 / JAX-RS2.0	Java SE 8

(*1) Java SE 8のサポートはWAS V8.5.5.9以降

変更点: Edition構成 これまでの提供

WAS for Developer



アプリを効率よく開発するためのランタイム開発者向けに無償版を提供。Eclipseアダプターも提供。

WAS Family Edition

WAS ND



ミッション・クリティカルなアプリ向けに、可用性、高いパフォーマンス、高度な運用管理機能を提供。

WXSの全機能が利用可能

WAS for z/OS



z/OSのシスプレックスの機能を活用して、高いセキュリティ、高信頼性、優れたリソース活用を実現

WXS z/OS クライアント機能が利用可能

WAS Hypervisor Edition



Pure Application System, VMware, PowerVM, zVMやその他の仮想化環境上で稼動するために最適化されたWAS ND環境を提供。

WAS (Base)



Web層のクラスタリングと、セッション・フェイルオーバー機能の提供により、ある程度の規模の環境において、高いセキュリティと高パフォーマンスを提供するトランザクション・エンジン。

WXSのHTTP セッション・キャッシングと動的キャッシュが利用可能

WAS Liberty Core

Libertyプロファイル
(Web Profile only)

軽量で低コストのLibertyプロファイル・ベースの製品。Java EEの全機能が不要なWebアプリケーションの稼動環境を迅速に構築。

WAS Express



動的Webアプリケーションを稼動するために、低コストですぐに使える環境を提供。PVU数に制限あり。

Expressエディション / Hypervisorエディションの廃止

- Expressエディション
 - ◆ 32bit版のみの提供だったExpressエディションは役割を終えV9からは提供されません
 - ◆ WAS Baseへのトレードアップが可能
 - ◆ WAS V8.5.5は継続して提供するため、WAS V8.5.5のExpressは利用可能
- Hypervisorエディション
 - ◆ VMwareの仮想マシンイメージとして提供されていたWAS ND版
 - ◆ PureApplicationなどの最新の構築製品への移行が必要

WAS V9 のEdition構成

WAS Family Edition

WAS ND



ミッション・クリティカルなアプリ向けに、可用性、高いパフォーマンス、高度な運用管理機能を提供。

WXSの全機能が利用可能

WAS for z/OS



z/OSのシスプレックスの機能を活用して、高いセキュリティ、高信頼性、優れたリソース活用を実現

WXS z/OS クライアント機能が利用可能

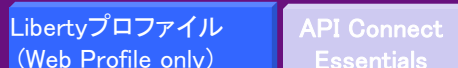
WAS (Base)



Web層のクラスタリングと、セッション・フェイルオーバー機能の提供により、ある程度の規模の環境において、高いセキュリティと高パフォーマンスを提供するトランザクション・エンジン。

WXSのHTTP セッション・キャッシングと動的キャッシュが利用可能

WAS Liberty Core

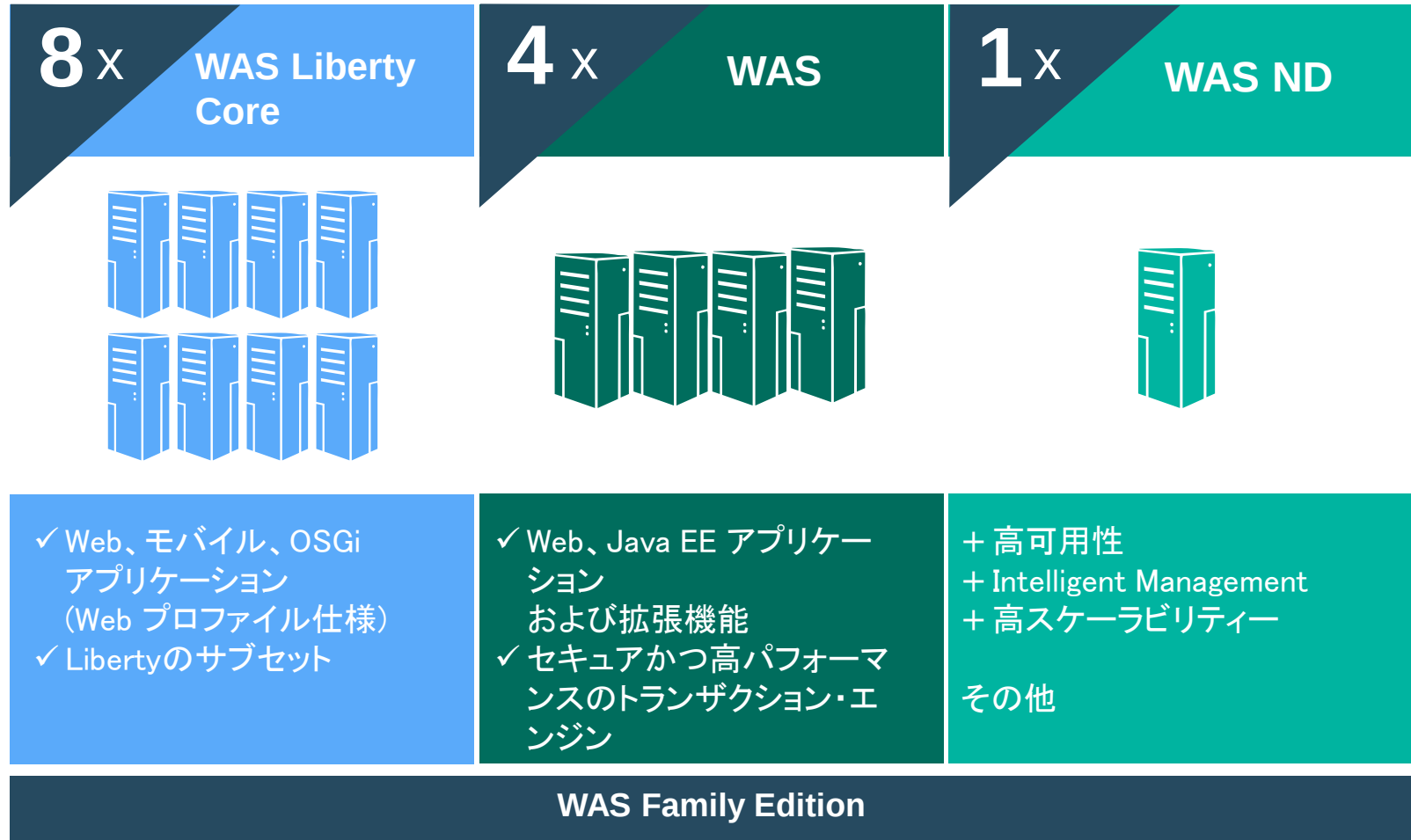


軽量で低コストのLibertyプロファイル・ベースの製品。Java EEの全機能が不要なWebアプリケーションの稼動環境を迅速に構築。



WAS Family Edition

- ◆ (日本語) <https://ibm.biz/BdH8Fi>
- ◆ (英語) <https://ibm.biz/BdHPNy>



Family Editionの1PVUは次に相当: 1 PVU ND *または* 4 PUV Base *または* 8 PVU Liberty Core
または いずれかの組み合わせ
後に新たな組み合わせを再デプロイすることも可能

新しい月額課金モデル（仮想コア・ベース:VPC）の登場

- 2016/07/22 発表レター「 IBM WebSphere Application Server は、新たにハイブリッド・クラウド用の、月別の料金設定オプションを提供します」
 - ◆ http://www-01.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/5/760/JAJPJP16-0385/index.html&lang=jp&request_locale=ja
- 発表内容サマリー
 - ◆ 仮想コア単位での月額課金モデル
 - WAS ND、WAS (base)、Liberty Core、WAS Family全てのエディションが対象
 - 通常の 買切り型の 課金モデルは従来通り
 - 契約期間は、1ヶ月～60ヶ月まで
 - ◆ メリット
 - パブリック・クラウドなど、物理コアが把握不可能な環境でもライセンス適用が可能
 - 物理コアが把握可能な環境の場合、仮想コアの総和と物理コアを比較し、より少ないコア数分のライセンスを購入すればOK
 - 例えば 物理8コアのサーバー環境を、仮想コア 32コアとしてオーバーコミットして利用している場合、8コア分の月額課金ライセンスでOK
 - ILMTが必須ではない
 - ILMT 環境を構築するためのリソースを別途確保しなくてよく、小規模環境でも適用しやすい
 - お客様による管理は必要になります
 - サポートは PA Perpetual Software Licenseと同様 (PA Technical Support)

その他変更点

- WAS Tools Editionは、2016/4/12で営業活動終了
 - ◆ Rational Application Developer for WebSphere Software(RAD) がバンドルされたライセンス
 - ◆ 移行先
 - WASは、同等PVUのWASのライセンス
 - RADは、バンドル50PVUごとに1許可ユーザーの比率の代替ライセンス
- V9から提供されない主な機能
 - ◆ 32bit版 traditional WAS
 - V9では、64bit版のみ提供
 - ◆ IBM Java 6, 7
 - V9では、IBM Java 8 のみ提供
 - ◆ Web2.0 Mobile and Toolkit
 - ◆ Edgeコンポーネント
 - Load Balancer for IPv4 (Legacy Load Balancer)
 - ◆ Assembly & Deploy Tools
 - Eclipseのプラグイン、WAS Developer Tools for eclipseのご利用が可能

2. WAS V9 traditional 新機能ハイライト

WAS V9のテーマ : ハイブリッド・クラウド

CREATE

開発者が高速に開発することを可能にする

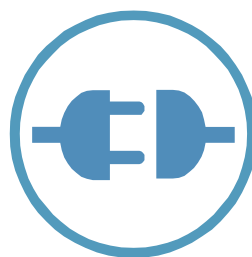
- ・ Java EE7 マーケットのリードとオープンソースのサポート
- ・ マイクロサービス化、API化を加速する
- ・ 効率的&継続的なデリバリのためのDevOpsツールとの完全統合



CONNECT

新規アプリ / 既存アプリを簡単にクラウドに接続する

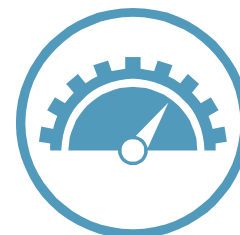
- ・ APIを作成し、公開する
- ・ アプリから外部のAPIに接続する
- ・ オンプレミス、クラウドを問わず、どこにでもデプロイできる



OPTIMIZE

ミッション・クリティカル・システムのスマートな管理

- ・ クラウド利用も含めたインフラの最適化
- ・ 高可用性: 自動スケールリング、動的ルーティング、ヘルス管理、問題分析
- ・ クラウドとモバイル対応のセキュリティ
- ・ JavaとNode.jsの企業向け管理



WAS traditional 新機能ハイライト(1)



- Java EE 7 対応
 - ◆ Libertyですでに実績がある共通コード
- Java SE 8 対応
 - ◆ IBM Java SDK 8 が同梱かつ唯一の前提Java
 - (補足) WAS 8.5.5.9 フル・プロファイル (2016年3月提供)
/ Liberty 8.5.5.5 (2015年3月提供) でJava SE 8を追加サポート

新Java EE/SE仕様対応以外(構成管理など)は、
V8.5.5から大きな変更なし

信頼と実績のWAS V9 traditional

WAS traditional 新機能ハイライト(2)

- V8.5
 - ◆ インテリジェント管理
 - 旧Virtual EnterpriseのNDへの統合
- V8.5.5
 - ◆ Webサーバーインテリジェント管理
 - ODRで提供される機能のほとんどがIHSのプラグインで実現可能に
 - ◆ インメモリー・キー・バリュー・データストア(KVS)
 - WXS (WebSphere eXtreme Scale) の同梱
 - ND版は、フル機能利用可能
 - Base版は、HTTPセッションと動的キャッシュの保管のみ可能

V8.0とV9.0を比較すると、さまざまな機能追加

WAS traditional 新機能ハイライト(3)

■ API 対応

- ◆ API Connect Essentialsを同梱
 - WASのバージョンに関わらず、利用可能
- ◆ API Discovery **9.0.0.1 new**
 - REST API の活用を促進するための機能
 - SwaggerでAPIドキュメンテーションを公開

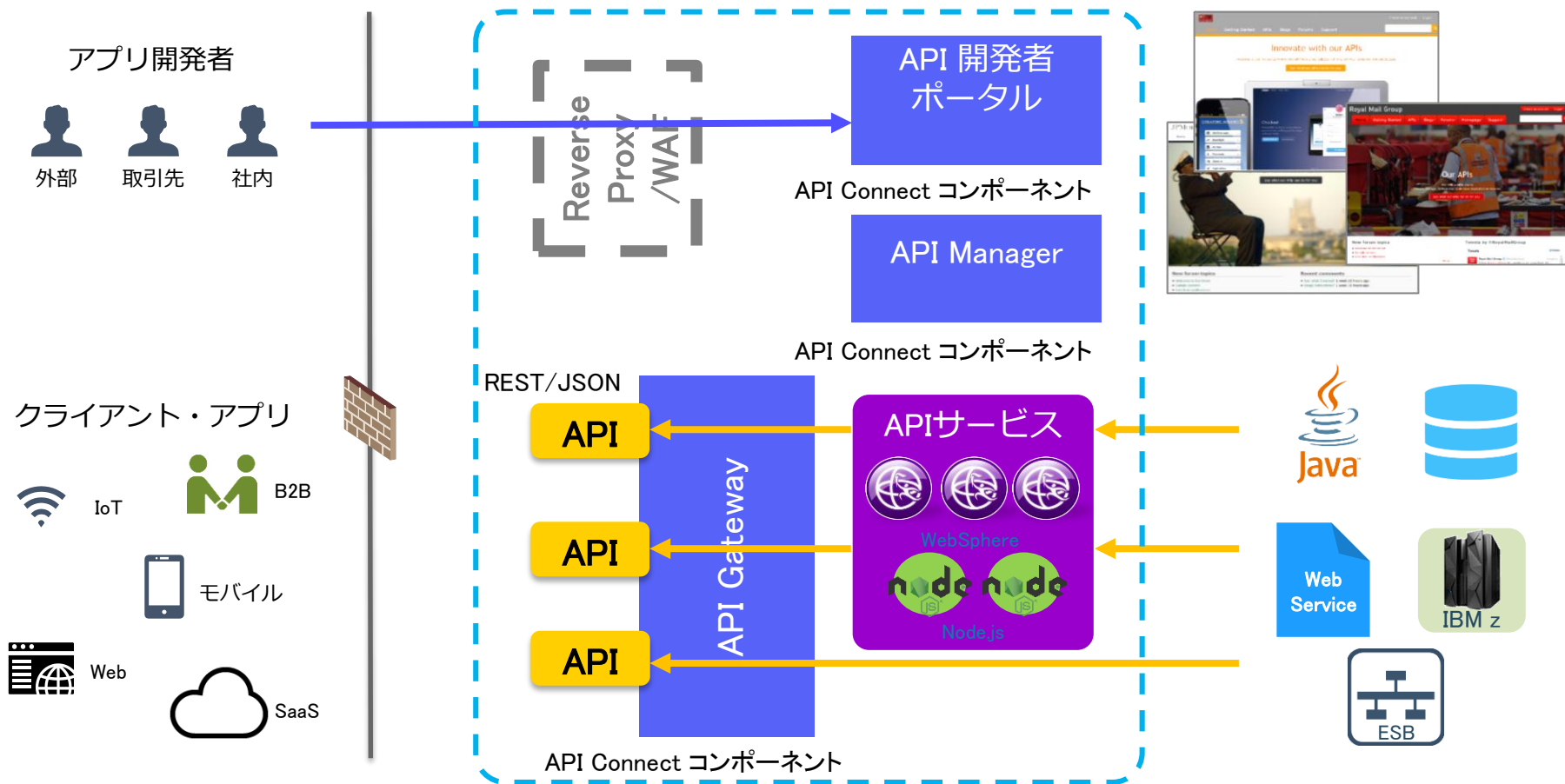


■ クラウド対応

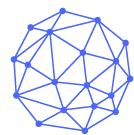
- ◆ WAS for Bluemix
 - WAS V9 traditional も提供
 - シングル・テナント・オプション
- ◆ WAS V9 traditional Dockerイメージ提供



API Connect のコンポーネントとアーキテクチャー



WebSphere Connect と API Connect の違い



IBM API Connect

WebSphere Connect

- 本番利用可能
- IBM サポートあり
- 500 × WAS PVU数 / 月 (SS&S契約のあるPVU数)
- シングル構成
- Micro ゲートウェイ
- 分析
- マイクロサービス・アプリケーション開発 (Node.js/Java)



API Connect Essentials

- 開発者向け
- 無償 / フォーラムサポート
- 5万APIコール / 月
- シングル構成
- Micro ゲートウェイ
- 分析
- マイクロサービス・アプリケーション開発 (Node.js/Java)

API Connect Professional

- 小規模プロジェクト向け
- 有償 / IBM サポートあり
- 500万APIコール / 月 or PVU課金
- クラスター構成(2 or 3台)
- Micro ゲートウェイ (HA)
- 分析
- マイクロサービス・アプリケーション開発 (Node.js/Java)

API Connect Enterprise

- 大規模プロジェクト向け
- 有償 / IBM サポートあり
- 2500万APIコール / 月 or PVU課金
- クラスター構成(3台以上)
- Micro ゲートウェイ (HA) と DataPower ゲートウェイ (HA)
- 高度な分析
- マイクロサービス・アプリケーション開発 (Node.js/Java)

WAS for Bluemix

■ WAS traditional V9は、WAS for Bluemix上でも提供

サービス

アプリケーション・サービス

新しい Web アプリとモバイル・アプリのデリバリー。



WebSphere Application Server

Bluemix 上にホストされたクラウド環境にある事前構成済みの **WebSphere Application Server** インスタンス



```

ejbdeploy.sh
enableProvisioning.sh
endptEnabler.sh
eventbusctl.sh
eventcatalog.sh
eventemit.sh
eventpurgepool.sh
eventpurge.sh
eventquery.sh
eventUpgradeDB2.sh
eventUpgradeDB2ZOS.sh
eventUpgradeOracle.sh
executePath.sh
executeQuery.sh
executeSLT.sh
Extractor.sh
FindJBTimers.sh
genHistoryReport.sh
genPluginCfg.sh
genVersionReport.sh
hadgrAdd.sh
hadgrRemove.sh
-hadgr-4.1M pld
/opt/IBM/WebSphere/Profiles/DefaultAppSrv01/bin
-bash-4.1M ./serverStatus.sh -all
ADN001161: Tool information is being logged in file
/opt/IBM/WebSphere/Profiles/DefaultAppSrv01/logs/serverStatus.log
ADN001281: Starting tool with the DefaultAppSrv01 profile
ADN005031: Retrieving server status for all servers
ADN005051: Servers found in configuration:
ADN005061: Server name: server1
ADN005081: The Application Server "server1" IS STARTED
-bash-4.1M
  
```

プラン	フィーチャー	料金
WAS Liberty Coreプラン	WebSphere Application Server Liberty Core. Red Hat Linux ゲスト。	¥ 22.00 JPY/Hour
✓ WAS Base プラン	従来型 WebSphere Application Server Base. Red Hat Linux ゲスト。	¥ 32.00 JPY/Hour
このプランでは、ホストされた Red Hat Linux オペレーティング・システム上に従来型 WebSphere Application Server Base をプロビジョンします。示されている料金は、2GB の RAM と 12GB の HD が搭載された 1vCPU として定義されている 1つのインスタンスの 1時間の料金を反映しています。課金は、インスタンス・サイズに基づいて増加します。		
WAS ND プラン	WebSphere Application Server Network Deployment. セル、集合、またはスタンドアロン。 Red Hat Linux ゲスト。	¥ 73.00 JPY/Hour



WebSphere Integrated Solutions Console

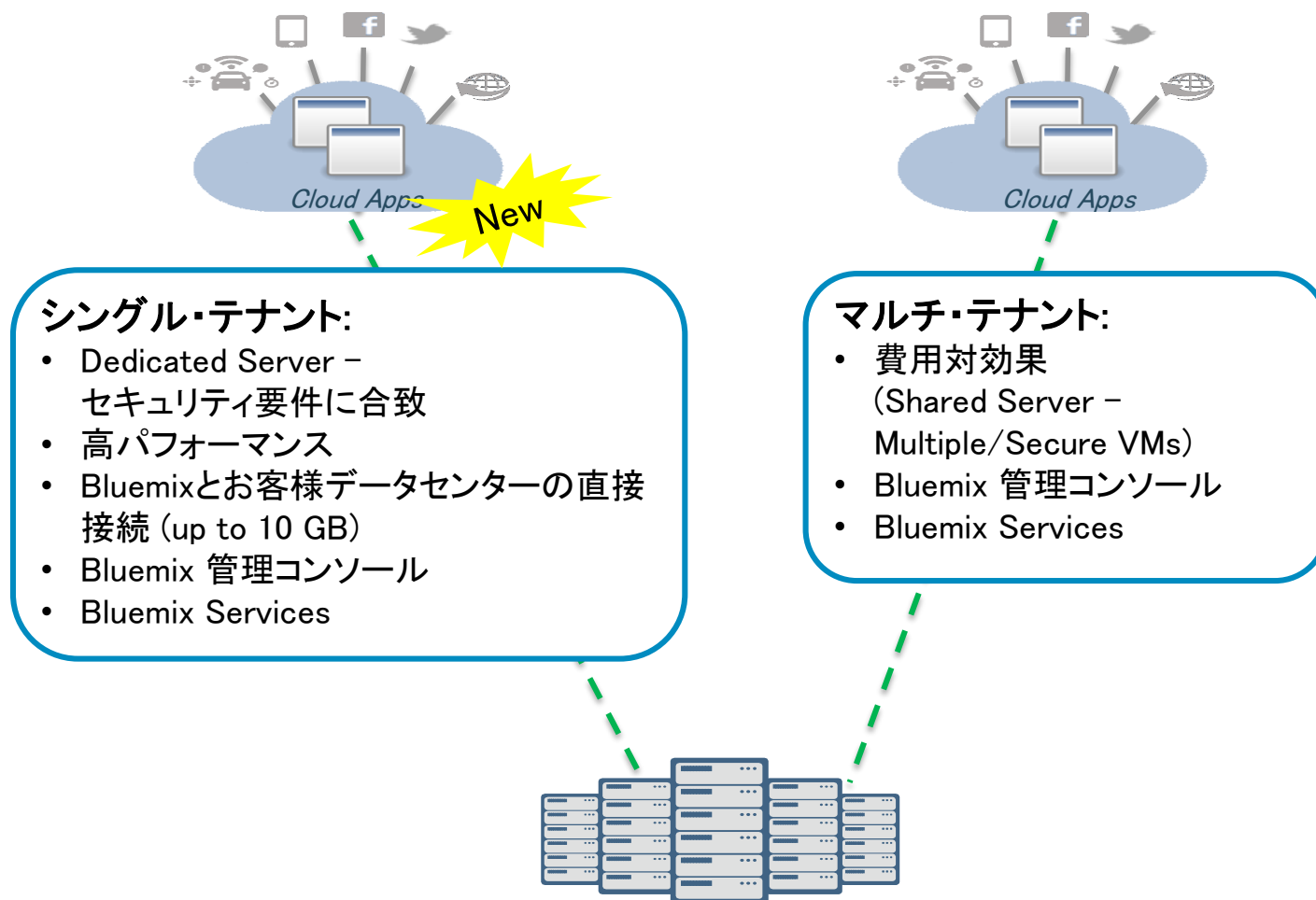
User ID:

Password:

IBM

Licensed Materials - Property of IBM (c) Copyright IBM Corp. 1997, 2011 All Rights Reserved. IBM, the IBM logo, IBM.com and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [Copyright and trademark information](#)

WAS for Bluemixシングル・テナント



WAS traditional Dockerイメージの提供

- WAS traditionalとIHSのDockerイメージを提供

WebSphere traditional and IBM HTTP Server on Docker Hub



KAVISURESH / JUNE 15, 2016 / 2 COMMENTS

We are pleased to announce that pre-built Docker images containing [IBM WebSphere Application Server for Developers V8.5.5.9 traditional](#) and [IBM HTTP Server](#) are now available on Docker Hub.

This enables you to get WebSphere Application Server traditional and IBM HTTP Server up and running quickly in your Docker environment with only a single command.

Running WebSphere Application server traditional is as simple as executing:

```
docker run --name test -h test -p 9043:9043 -p 9443:9443 -d ibmcom/websphere-traditional
```

RECENT POSTS

[WebSphere traditional and IBM HTTP Server on Docker Hub](#)

[Beta: WebSphere Liberty and tools \(June 2016\)](#)

[Putting the 'Micro' into Microservices with Raspberry Pis](#)

[Running a multi-container application using Docker Compose](#)

[Deploying applications to WAS traditional and IBM HTTP Server under Docker](#)

WASdev「WebSphere traditional and IBM HTTP Server on Docker Hub」

<https://developer.ibm.com/wasdev/blog/2016/06/15/websphere-traditional-ibm-http-server-docker-hub/>

3. WAS 7 / 8, Java 6 EOS

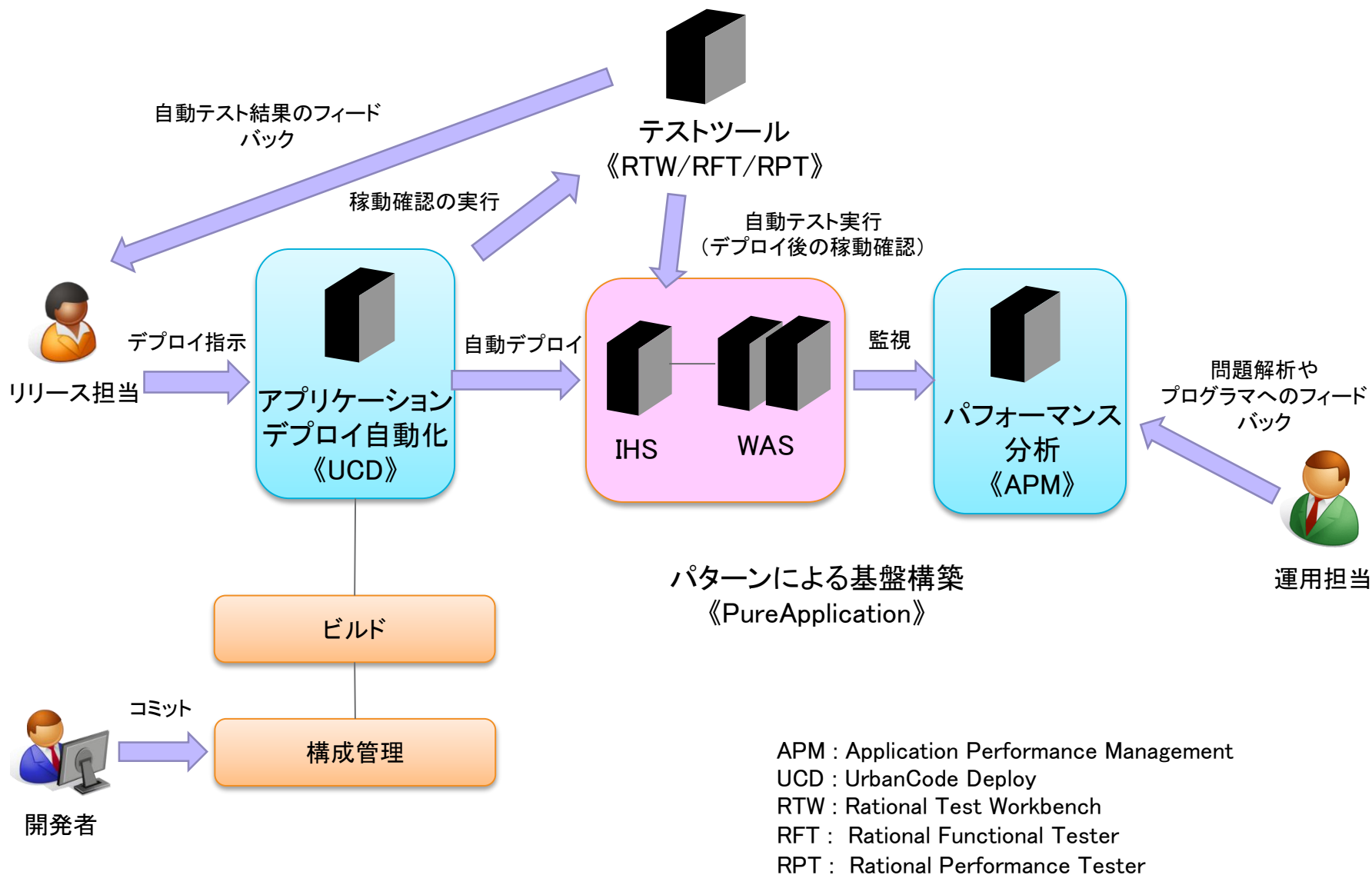
WAS 7 / 8 通常サポート終了とJDKサポート終了

- WAS 7 / 8 は2018年4月に通常サポートが終了します
- WAS 8.5については, JDK 6は2018年4月以降, JDK 7は2019年9月以降は新規の修正は作成されません

2016	2017	2018	2019	2020
WAS V7.0		2018/4	2019/9	
JDK 6				
WAS V8.0				
JDK 6				
WAS V8.5				
JDK 6				
JDK 7				
JDK 8				

4. WAS関連ソリューション

WAS関連ソリューション全体像



参考情報

- developerWorks WebSphere Application Server カテゴリー
 - ◆ <http://www.ibm.com/developerworks/jp/websphere/category/was/>
- WAS 9 Knowledge Center (オンラインマニュアル)
 - ◆ http://www.ibm.com/support/knowledgecenter/ja/SSAW57_9.0.0/as_ditamaps/was900_welcome_n_dmp.html
- WAS V9 アナウンスメント・セミナー資料
 - ◆ http://www.ibm.com/developerworks/jp/websphere/library/was/was9_ws/
- WebSphere Application Server traditional V9 導入ガイド
 - ◆ http://www.ibm.com/developerworks/jp/websphere/library/was/twas9_install/
- WebSphere Application Server V9.0へのマイグレーションガイド
– WAS traditional編 –
 - ◆ http://www.ibm.com/developerworks/jp/websphere/library/was/was9_twas_migration/
- WebSphere Application Server V9.0 パラメーター・シート
 - ◆ http://www.ibm.com/developerworks/jp/websphere/library/was/was9nd_parameter/
- WAS V9 Liberty基盤設計セミナー資料
 - ◆ http://www.ibm.com/developerworks/jp/websphere/library/was/liberty_v9_infradesign/
- WASdev.net
 - ◆ <https://developer.ibm.com/wasdev/>

ワークショップ、セッション、および資料は、IBMまたはセッション発表者によって準備され、それぞれ独自の見解を反映したものです。それらは情報提供の目的のみで提供されており、いかなる参加者に対しても法律的またはその他の指導や助言を意図したものではなく、またそのような結果を生むものでもありません。本講演資料に含まれている情報については、完全性と正確性を期するよう努力しましたが、「現状のまま」提供され、明示または暗示にかかわらずいかなる保証も伴わないものとします。本講演資料またはその他の資料の使用によって、あるいはその他の関連によって、いかなる損害が生じた場合も、IBMは責任を負わないものとします。本講演資料に含まれている内容は、IBMまたはそのサプライヤーやライセンス交付者からいかなる保証または表明を引きだすことを意図したものでも、IBMソフトウェアの使用を規定する適用ライセンス契約の条項を変更することを意図したものでもなく、またそのような結果を生むものでもありません。

本講演資料でIBM製品、プログラム、またはサービスに言及していても、IBMが営業活動を行っているすべての国でそれらが使用可能であることを暗示するものではありません。本講演資料で言及している製品リリース日付や製品機能は、市場機会またはその他の要因に基づいてIBM独自の決定権をもっていつでも変更できるものとし、いかなる方法においても将来の製品または機能が使用可能になると確約することを意図したものではありません。本講演資料に含まれている内容は、参加者が開始する活動によって特定の販売、売上高の向上、またはその他の結果が生じると述べる、または暗示することを意図したものでも、またそのような結果を生むものでもありません。パフォーマンスは、管理された環境において標準的なIBMベンチマークを使用した測定と予測に基づいています。ユーザーが経験する実際のスループットやパフォーマンスは、ユーザーのジョブ・ストリームにおけるマルチプログラミングの量、入出力構成、ストレージ構成、および処理されるワークロードなどの考慮事項を含む、数多くの要因に応じて変化します。したがって、個々のユーザーがここで述べられているものと同様の結果を得られると確約するものではありません。

記述されているすべてのお客様事例は、それらのお客様がどのようにIBM製品を使用したか、またそれらのお客様が達成した結果の実例として示されたものです。実際の環境コストおよびパフォーマンス特性は、お客様ごとに異なる場合があります。

IBM、IBM ロゴ、ibm.com、WebSphere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をご覧ください。

Windowsは Microsoft Corporationの米国およびその他の国における商標です。

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。