



ORACLE®

WLST 入門

日本オラクル株式会社 オラクルユニバーシティ
岡田 大輔
2012年07月12日

以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント(確約)するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

OracleとJavaは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

自己紹介

- 氏名: 岡田 大輔 (おかだ だいすけ)
 - 所属: オラクルユニバーシティビジネス推進部
 - Fusion Middleware, Java (+Solaris)研修・資格推進担当
 - Email: daisuke.x.okada@oracle.com
- WebLogic Serverの使用歴: 10年以上
 - 数年前までFMW系研修のセンセイでした
 - Java (SE, EE), WLS, ADF, Coherence, SOA Suite, OSB, OBPM ...
 - WebLogic Server は 5.1 から 12c までさわったことがあります!!

Quiz.

- 次の中でWebLogic Serverのドメインの設定を変更することができるものはどれ？
 1. ドメイン構成ウィザード
 2. Administration Console
 3. weblogic.Deployer
 4. WLST

Answer. 全て変更可能

- 次の中でWebLogic Serverのドメインの設定を変更することができるものはどれ？
 1. ドメイン構成ウィザード
 2. Administration Console
 3. weblogic.Deployer
 4. WLST

皆さん、使い分けできますか？

Agenda

- WLSTとは？
- WLSTコマンド
- MBeanの調べ方
- WLSTスクリプティング

WLSTとは

- WebLogic Server Scripting Tool
 - WebLogic ドメインの作成、管理、監視に使用できるコマンドライン・スクリプト環境 (WebLogic Server 9.x ~)
 - WLST オンライン・WLSTオフライン
 - 対話モード・スクリプトモード・組み込みモード
 - Built on **Jython** (100% Pure Java の Python実装)



WLSTの実行方法

- 前提: WLST実行前にCLASSPATHを設定
⇒ \$WL_HOME/server/bin/setWLSEnv.sh (cmd)
- 対話モード
 - > java weblogic.WLST
- スクリプトモード (-i: スクリプト終了後に対話モード)
 - > java weblogic.WLST xxx.py
 - > java weblogic.WLST -i xxx.py
 - 対話モードで > execfile('xxx.py') でもスクリプト実行可能
- 組み込みモード
 - weblogic.management.scriping.utils.WLSTInterpreter
 - exec('command_str')メソッドで実行

WLSTの機能



複数のツールで実行していた
WebLogic Serverの管理作業を
WLSTひとつに集約できる

```

コマンドプロンプト - java weblogic.WLST
C:\u01\work\domains\12c_domain>java weblogic.WLST
WebLogic Scripting Tool (WLST) を初期化しています...
WebLogic Server Administration Scripting Shellへようこそ
使用可能なコマンドに関するヘルプを表示するには、help()と入力してください

wls:/offline> connect('weblogic', 'weblogic1', 't3://localhost:7001')
ユーザー ID weblogicでt3://localhost:7001に接続しています...
to Admin Server 'AdminServer' that belongs to domain '12c_domain' が正常に接続されました

警告: サーバーへの接続に安全でないプロトコルが使用
されました。通信セキュリティを確保するには、かわりにSSLポートまたは
管理ポートを使用する必要があります。

wls:/12c_domain/serverConfig> ls()
dr-- AdminConsole
dr-- AppDeployments
dr-- BridgeDestinations
dr-- Clusters
dr-- CoherenceClusterSystemResources
dr-- CoherenceServers
dr-- CustomResources
    
```

WLST



```

コマンドプロンプト
C:\u01\work\domains\12c_domain>java weblogic.Deployer -adminurl t3://localhost:7001 -username weblogic -password weblogic1 -deploy -name zMFR -targets cluster1 -source c:\u01\work\apps\zMFR.war
weblogic.Deployerがオプション -adminurl t3://localhost:7001 -username weblogic -deploy -name zMFR -targets cluster1 -source c:\u01\work\apps\zMFR.warを指定して呼び出されました。
(2012/07/08 11時12分21秒 JST) (Info) <J2EE Deployment SP1> (BE1-260121) <アプリケーション zMFR [アーカイブ: c:\u01\work\apps\zMFR.war]のdeploy操作をcluster1に初期化しています。
タスクが開始されました: [Deployer:149026]cluster1上のアプリケーション zMFRをデプロイ。
タスク4完了: [Deployer:149026]cluster1上のアプリケーション zMFRをデプロイ。
ターゲットの状態: クラスター cluster1でdeploy完了
C:\u01\work\domains\12c_domain>
    
```

weblogic.Deployer

WLSTとMBean

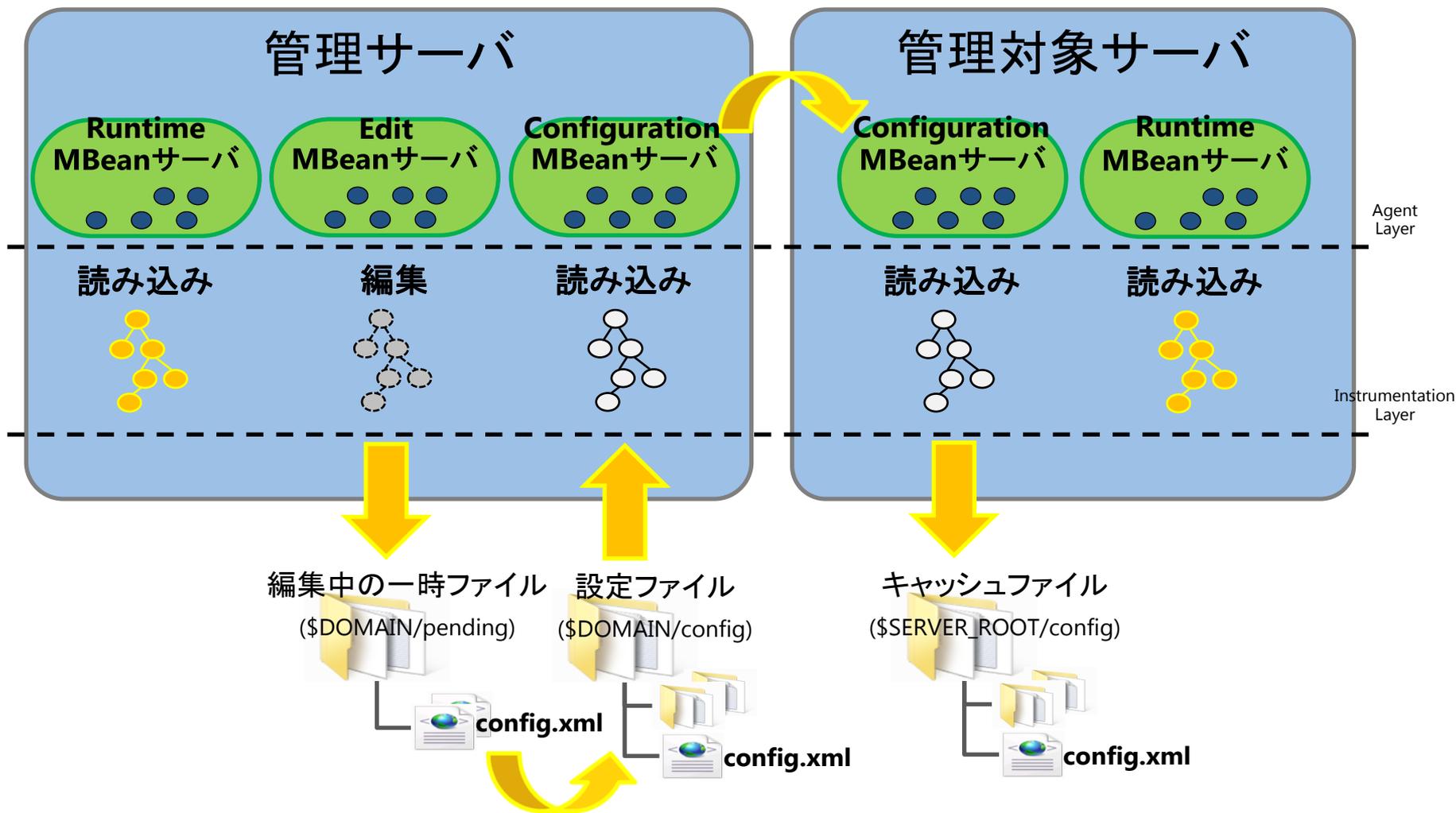
- WebLogic Serverのリソース(構成情報、監視情報)はJMXベースで管理されている
 - JMX管理対象オブジェクトを**MBean**と呼ぶ
 - WLSTではMBeanをファイルシステムのように操作する
 - MBeanサーバ: ツリー(ドライブ)
 - MBean: ディレクトリ
 - MBeanの属性: ファイル

コマンド	ツリー
serverConfig()	ServerConfigurationMBean
serverRuntime()	ServerRuntimeMBean
domainRuntime()	DomainRuntimeMBean
edit()	EditMBean (管理サーバのみ)
jndi()	JNDI (注: MBeanサーバではない)

3種類のMBean

MBean	説明	MBeanの場所
ConfigurationMBean	ドメイン構成情報を表すMBean \$DOMAIN/config配下の情報	管理サーバ・管理対象サーバ
RuntimeMBean	サーバの実行時情報を表すMBean Administration Consoleの[監視]タブの情報	管理サーバ・管理対象サーバ
EditMBean	ドメイン構成情報の編集のために使われるMBean Administration Consoleの編集ロック	管理サーバ

構成管理アーキテクチャ



対話モードの操作

コマンド実行例

- 対話モードでの操作はUnix環境のディレクトリ操作に似ている
 - cdで移動
 - get/setで設定
 - lsで属性一覧

注意点:

- コマンドは大文字小文字を区別する
- 文字列は引用符(' or ")で囲む
 - cd ('/Servers')
 - × cd (/Servers)

```
wls:/ > connect(user, password,
adminServer)
wls:/ > cd('/Servers/AdminServer')
wls:/ > get('ListenPort')
7001
wls:/ > ls()
dr--    COM
dr--    CandidateMachines
dr--    Cluster
dr--    CoherenceClusterSystemResource
dr--    DataSource
dr--    DefaultFileStore
dr--    ExecuteQueues
dr--    FederationServices
dr--    IIOP
...
```

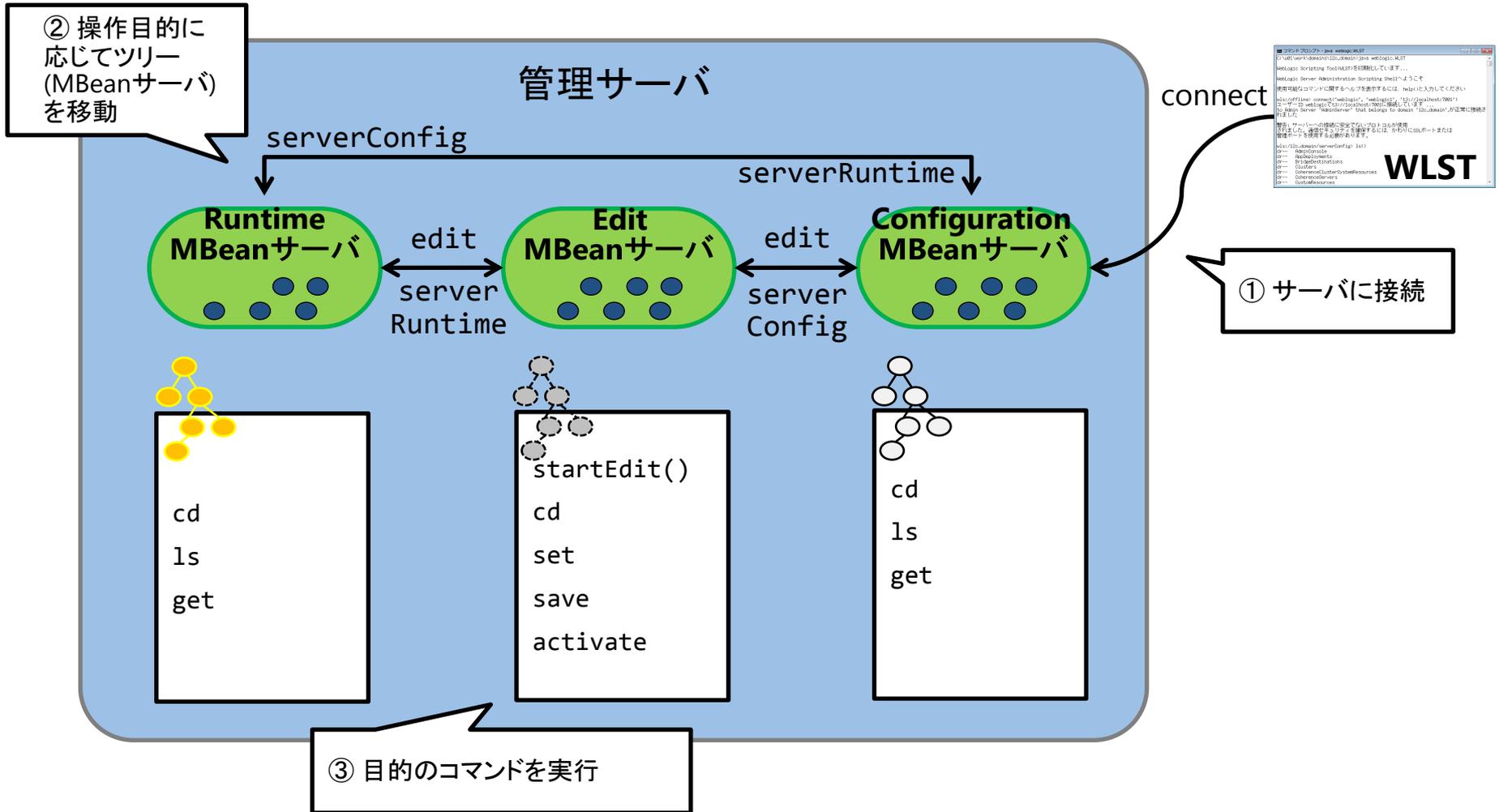
まとめ

- WLSTはWebLogicドメインの作成、管理、監視に使用できる
 - 構成情報、監視情報をあらわすMBeanを直接操作可能
 - 構成情報: ConfigurationMBean / EditMBean
 - 監視情報: RuntimeMBean
 - Jythonベースのツール
 - 用途に応じた起動モード
 - 対話モード/スクリプトモード/組み込みモード

Agenda

- WLSTとは？
- WLSTコマンド
- MBeanの調べ方
- WLSTスクリプティング

WLST操作の基本的な流れ



WLST操作

```
コマンドプロンプト - java weblogic.WLST

C:\u01\work\domains\12c_domain>java weblogic.WLST
WebLogic Scripting Tool(WLST)を初期化しています...
WebLogic Server Administration Scripting Shellへようこそ
使用可能なコマンドに関するヘルプを表示するには、help()と入力してください

wls:/offline> connect('weblogic','weblogic1','t3://localhost:7001')
ユーザー ID weblogic で t3://localhost:7001 に接続しています...
to Admin Server 'AdminServer' that belongs to domain '12c_domain' が正常に
安全でないプロトコルが使用
リディを確保するには、かわりにSSLポートまたは
があります。

wls:/12c_domain/serverConfig> edit()
ロケーションが編集ツリーに移動しました。これは、DomainMBeanをルートとする
書き込み可能なツリーです。変更するには、startEdit()を使用して
編集セッションを開始する必要があります。
詳細は、help(edit)を使用してください

wls:/12c_domain/edit> startEdit()
編集セッションを開始しています...
編集セッションが開始されました。変更が完了したら、必ず保存してアクティブ化
してください。

wls:/12c_domain/edit !> od('/Servers')
wls:/12c_domain/edit/Servers !> create('Server4','Server')
Server4 という名前のタイプ Server の MBean が正常に作成されました。
wls:/12c_domain/edit/Servers !> od('Server4')
wls:/12c_domain/edit/Servers/Server4 !> set('ListenAddress','localhost')
wls:/12c_domain/edit/Servers/Server4 !> cmo.setListenPort(7017)
wls:/12c_domain/edit/Servers/Server4 !> validate()
変更を検証しています...
変更が正常に検証されました
wls:/12c_domain/edit/Servers/Server4 !> save()
すべての変更を保存しています...
すべての変更が正常に保存されました。
wls:/12c_domain/edit/Servers/Server4 !> activate()
すべての変更をアクティブ化しています。しばらく時間がかかる場合があります...
アクティブ化が完了すると、この編集セッションに関連付けられた
編集ロックが開放されます。
アクティブ化が完了しました
wls:/12c_domain/edit/Servers/Server4> _
```

② 操作目的に応じてツリー (MBeanサーバ) を移動

① サーバに接続

- ③ 目的のコマンドを実行
- a. ロックして編集
 - b. 管理対象サーバ作成
 - c. 保存
 - d. アクティブ化

WLSTコマンド

- 制御・参照・情報 -

コマンド	説明	構文
connect (オフライン)	任意のサーバに接続する	connect([username, password], [url], [timeout]) timeout: milisec指定
disconnect (オンライン)	接続中のサーバから接続解除する	disconnect()
cd (オンライン オフライン)	MBean階層を移動する	cd(mbeanName)
ls (オンライン オフライン)	指定した管理オブジェクトの属性、操作、子オブジェクトを表示する	ls([a c o] [moPath]) a=属性のみ、c=子オブジェクトのみ、o=メソッドのみ
find (オンライン オフライン)	現在の階層内でMBeanを検索する	find([name], [type], [searchInstancesOnly]) name =Mbean名、type=MBeanType
threadDump (オフライン)	指定されたサーバのスレッド・ダンプを取得する	threadDump([writeToFile], [fileName], [serverName]) writeToFile: trueならファイル出力
exit (オンライン オフライン)	WLSTを終了する	exit([defaultAnswer], [exitcode]) exitcode: 終了コード (デフォルトは0)

WLSTコマンド

- 編集 -

コマンド	説明	構文
startEdit (オンライン)	編集セッションを開始する	startEdit()
save (オンライン)	変更された編集内容を保存する	save()
activate (オンライン オフライン)	編集セッションをアクティブ化する	activate()
stopEdit (オンライン)	編集セッションを停止して編集ロックを開放する	stopEdit() ※ 編集セッション中の変更は破棄されない
cancelEdit (オンライン)	編集セッションを取り消してロックを開放する	cancelEdit() ※ 編集セッション中の変更は破棄される
validate (オンライン)	保存前の変更の検証する	validate()
undo (オンライン)	保存されていない変更を破棄する	undo()

WLSTコマンド

- 編集 -

コマンド	説明	構文
set (オンライン オフライン)	MBeanの属性値を設定する	set(attrName, value) ※ cmo.setattrName(value)と等価
get (オンライン オフライン)	MBeanの属性値を取得する	get(attrName)
create (オンライン オフライン)	MBeanを作成する	create(name, childMBeanType, [baseProviderType]) childMBeanType: 作成するMBeanタイプ
delete (オンライン オフライン)	MBeanを削除する	delete(name, childMBeanType)
assign (オンライン)	リソースを割り当てる	assign(sourceType, sourceName, destinationType, destinationName) sourceType: 割り当てるMbeanタイプ desitinationType:宛先タイプ(Target,Cluster)
encrypt (オンライン)	指定された文字列を暗号化する	encrypt(obj, [domainDir]) domainDir: 省略時は接続中のドメイン用の暗号化

WLSTコマンド

- ライフサイクル・ノードマネージャ -

コマンド	説明	構文
start (オンライン)	NodeManagerを使用してサーバを起動	start(name, [type], [url], [block]) type: 'Server'か'Cluster'を指定 block: 'false'を指定すると起動完了まで制御をブロックしない
shutdown (オンライン)	実行中のサーバ・クラスタを停止	shutdown([name], [entityType], [ignoreSessions], [timeOut], [force], [block]) entityType: ServerかClusterを指定 ignoreSessions: trueの場合HTTPセッションを直ちに中止 force: force='true'の場合強制停止
nmConnect (オフライン)	NodeManagerに接続	nmConnect([username, password], [host], [port], [domainName], [domainDir] [nmType], [verbose]) nmType: NodeManagerタイプ (ssl plain ssh rsh)
nmKill (オンライン オフライン)	NodeManagerで起動したサーバを強制停止	nmKill([serverName], [serverType]) serverType: WebLogicかCoherenceを指定
nmStart (オフライン)	NodeMangerを使用してサーバを起動	nmStart([serverName], [domainDir], [props], [writer], [serverType]) props: makePorpertiesObject('name=value')で指定可能 (要 <i>from java.util import Properties</i>)

(補足) start と nmStart

- nmStart / start
 - nmStart: NodeManagerに対して配下のサーバを起動することを指定するコマンド
 - 管理サーバが停止していても起動可能
 - start: 管理サーバに対してドメイン内のサーバを起動することを指定するコマンド
 - startは複数台のサーバ連続で起動する際は便利
 - block= 'false' と typeに'Cluster' を指定可能
 - 両者は特長を把握して状況に応じて使い分ける
- 停止は **shutdown** (nmKill → kill -9)

WLST グローバル変数

- WLSTで使用できる定義済み変数 -

変数名	説明
connected	WLSTオンラインかどうか
domainName	接続中のドメイン名
serverName	接続中のサーバ名
isAdminServer	接続中のサーバが管理サーバかどうか
recording	WLST記録中かどうか
username	接続中のユーザ名
version	WebLogic Serverのバージョン
cmo	現在の管理オブジェクト

cmoの使用例

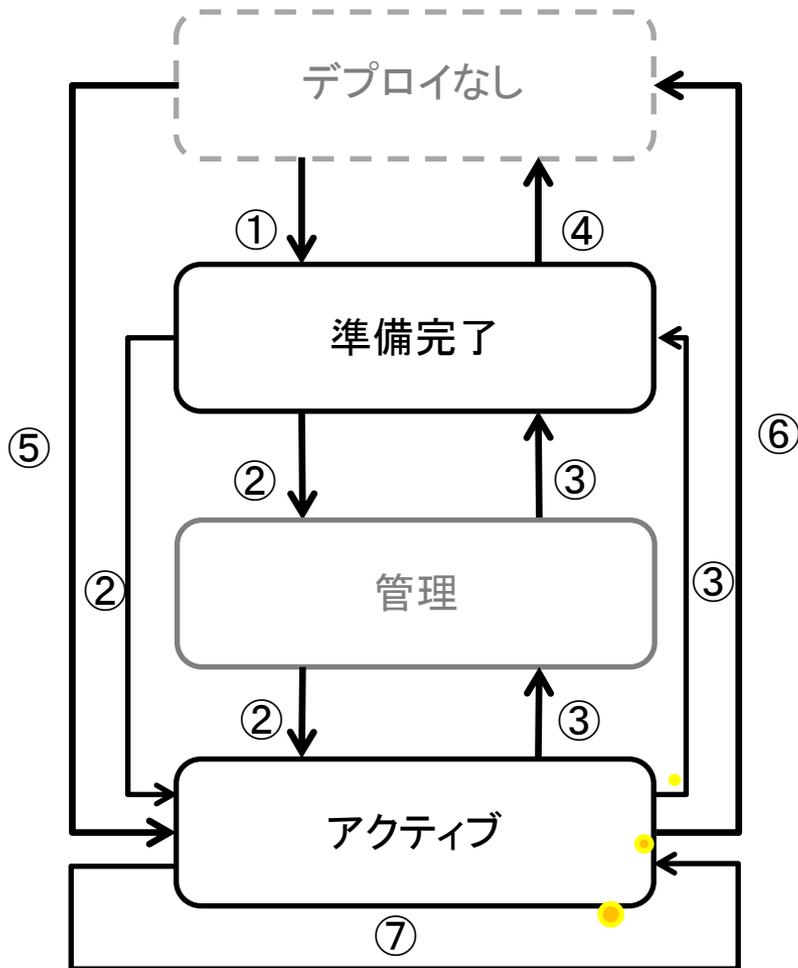
```
wlst:/domain/ServerConfig> serverRuntime() ← cmoはServerRuntimeMBean  
wlst:/domain/ServerRuntime> cmo.getState()  
wlst:/domain/ServerRuntime> svrRuntime = cmo
```

WLSTコマンド

- デプロイメント -

コマンド	説明	構文
deploy (オンライン)	アプリケーションをデプロイ	deploy(appName, path, [targets], [stageMode], [planPath], [options]) appName: デプロイメント名
undeploy (オンライン)	アプリケーションをアンデプロイ	undeploy(appName, [targets], [options])
redeploy (オンライン)	アプリケーションを再デプロイ	redeploy(appName, [planPath], [options])
startApplication (オンライン)	アプリケーションを起動	startApplication(appName, [options]) option: adminMode='true'を指定すると管理モードで起動
stopApplication (オンライン)	アプリケーションを停止	stopApplication(appName, [options])
distributeApplication (オンライン)	アプリケーションを分散	distributeApplication(appPath, [planPath], [targets], [options]) ※ appNameは appPath のファイル名が使われる
listApplications (オンライン)	デプロイ済みアプリケーションの一覧	listApplications()

コマンドラインデプロイメント



配布資料の矢印の向きが反対でした...
m(_)m

コマンドラインデプロイメントはWLST、または weblogic.Deployer で実行可能

* コマンドは管理サーバに対して実行

	WLST	weblogic.Deployer
①	distributeApplication()	-distribute
②	startApplication()	-start
③	stopApplication()	-stop
④	undeploy()	-undeploy
⑤	deploy()	-deploy
⑥	undeploy()	-undeploy
⑦	redeploy()	-redeploy

管理モード起動時は管理ポートを有効化する必要がある (主にプロダクション再デプロイメント用)

WLSTでSSL接続を行う場合

- 管理ポートを有効化した場合、全ての管理トラフィックは管理ポート(SSL)を経由してアクセスする
 - 管理コンソール (**`https://<hostname>:<admin_port>/console`**)
 - コマンドラインツール (WLST, weblogic.Deployer ...)
 - 管理対象サーバ起動時の管理サーバ指定
 - `$ startManagedWebLogic.sh <sever_name> t3s://<hostname>:<admin_port>`
- コマンドラインツールでSSL接続を行う場合は起動オプションでキーストアを指定する必要がある
 - `-Dweblogic.security.TrustKeyStore=<KeyStore>`
 - キーストアの指定 (デモ用証明書の場合: DemoTrust)
 - `-Dweblogic.security.ssl.ignoreHostnameVerification=true`
 - ホスト名検証の無効化 (デモ用証明書使用時のみ)

SSLを使用するWLST接続

```
コマンドプロンプト - java -Dweblogic.security.TrustKeyStore=DemoTrust -Dweblogic.sec...
C:\u01\work\domains\12c_domain>java -Dweblogic.security.TrustKeyStore=DemoTrust
-Dweblogic.security.ssl.ignoreHostnameVerification=true weblogic.WLST

wls:/offline> connect()
ユーザー名を入力してください:weblogic
パスワードを入力してください:
サーバーのURLを入力してください[t3://localhost:7001]。t3s://localhost:9002
ユーザーID weblogic@t3s://localhost:9002に接続しています ...
<2012/07/09 19時29分02秒 JST> <Info> <Security> <BEA-090905> <起動パフォーマンス
を向上するためにCryptoJ JCEプロバイダ自己整合性チェックを無効にしています。この
チェックを有効にするには、-Dweblogic.security.allowCryptoJDefaultJCEVerification
=trueを指定します>
<2012/07/09 19時29分02秒 JST> <Info> <Security> <BEA-090906> <RSA CryptoJのデフ
ォルトの乱数ジェネレータをECDRBGからFIPS186PRNGに変更しています。この変更を無効
にするには、-Dweblogic.security.allowCryptoJDefaultPRNG=trueを指定します>
<2012/07/09 19時29分02秒 JST> <Info> <Security> <BEA-090908> <デフォルトのWebLog
ic SSLホスト名検証実装を使用しています。>
to Admin Server 'AdminServer' that belongs to domain '12c_domain'.が正常に接続さ
れました
```

カスタムキーストアの場合の指定例:

- Dweblogic.security.TrustKeyStore=CustomTrust ¥
- Dweblogic.security.CustomTrustKeyStoreFileName=/home/wluser/cert/myTrust.jks ¥
- Dweblogic.security.CustomTrustKeyStoreType=jks ¥
- Dweblogic.security.SSL.trustedCAKeyStore=/home/wluser/cert/myTrust.jks

そ

れを入力してください

管理ポートを指定

参考: 管理ポートの設定

```
コマンドプロンプト - java weblogic.WLST
wls:/12c_domain/serverConfig> edit()
ロケーションが編集ツリーに移動しました。これは、DomainMBeanをルートとする
書き込み可能なツリーです。変更するには、startEdit()を使用して
編集セッションを開始する必要があります。

詳細は、help(edit)を使用してください

wls:/12c_domain/edit> startEdit()
編集セッションを開始しています ...
編集セッションが開始されました。変更が完了したら、必ず保存してアクティブ化
してください。
wls:/12c_domain/edit !> omo.setAdministrationPortEnabled(true)
wls:/12c_domain/edit !> save()
すべての変更を保存しています ...
すべての変更が正常に保存されました。
wls:/12c_domain/edit !> activate()
すべての変更をアクティブ化しています。しばらく時間がかかる場合があります ...
アクティブ化が完了すると、この編集セッションに関連付けられた
編集ロックが開放されます。
Traceback (innermost last):
  File "<console>", line 1, in ?
  File "<iostream>", line 376, in ad
  File "<iostream>", line 1847, in r
WLSTException: Error occured while p
エラーが発生しました。 : java.rmi.RemoteException: User 'principals=[weblogic, A
```

Activateすると管理ポートが有効になるため、管理ポ
ート経由で接続していない現在のWLSTセッションは
エラーになる
対応策: disconnect後に管理ポート経由でconnect

参考: easeSyntax()

- 対話モードの入力を簡易構文で -

```
CA: コマンドプロンプト - java weblogic.WLST
wls:/12c_domain/serverConfig> easeSyntax()

一部のWLSTコマンドの構文の簡易化が選択されました。
ただし、簡易構文は、必ず相互モードで
使用してください。スクリプト・モードやループで使用すると、
簡易構文は正しく機能しません。簡易構文が
選択されましたが、まだ正規のJython構文を使用
できます。
easeSyntaxを使用して、これを無効にしてください。
簡易構文をサポートするコマンドについては、help(eas
wls:/12c_domain/serverConfig> serverRuntime
ロケーションがserverRuntimeツリーに移動しました。こ
ートとする読取り専用のツリーです。
詳細は、help(serverRuntime)を使用してください

wls:/12c_domain/serverRuntime> cd /JVMRuntime/AdminServer
wls:/12c_domain/serverRuntime/JVMRuntime/AdminServer> get HeapSizeCurrent
288030720L
wls:/12c_domain/serverRuntime/JVMRuntime/AdminServer> ls

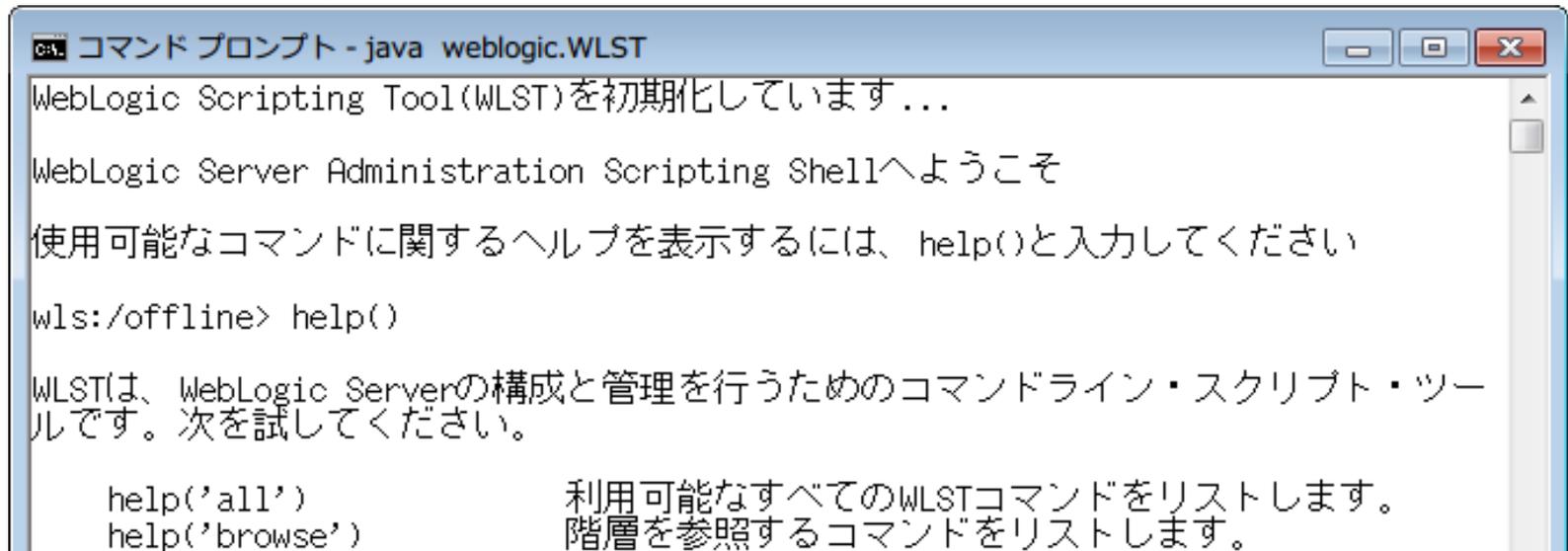
-r--   HeapFreeCurrent           108700208
-r--   HeapFreePercent          62
-r--   HeapSizeCurrent          288030720
-r--   HeapSizeMax              477233152
-r--   JavaVMVendor             Oracle Corporation
```

簡易構文を有効化するとUnixシェルと
同様に操作できる
全てのコマンドで簡易構文が使用でき
るわけではないので注意

まとめ

- コマンド使用方法の調べ方

1. WLST helpコマンド



```
コマンドプロンプト - java weblogic.WLST
WebLogic Scripting Tool (WLST)を初期化しています...
WebLogic Server Administration Scripting Shellへようこそ
使用可能なコマンドに関するヘルプを表示するには、help()と入力してください
wls:/offline> help()
WLSTは、WebLogic Serverの構成と管理を行うためのコマンドライン・スクリプト・ツ
ールです。次を試してください。
help('all')          利用可能なすべてのWLSTコマンドをリストします。
help('browse')       階層を参照するコマンドをリストします。
```

2. WebLogic Scripting Tool コマンド・リファレンス

- http://docs.oracle.com/cd/E28613_01/web.1211/b65931/toc.htm

Agenda

- WLSTとは？
- WLSTコマンド
- MBeanの調べ方
- WLSTスクリプティング

MBeanの一覧表示

- WLST lsコマンド -

```
コマンドプロンプト - java weblogic.WLST
wls:/12c_domain/serverConfig> ls()
dr--  AdminConsole
dr--  AppDeployments
dr--  Clusters
dr--  DeploymentConfiguration
dr--  Deployments
dr--  EmbeddedLDAP

-r--  AdminServerName                AdminServer
-r--  AdministrationMBeanAuditingEnabled  false

-rw-  AdminServerName
-rw-  AdministrationMBeanAuditingEnabled
-rw-  AdministrationPort
-rw-  AdministrationPortEnabled
-rw-  AdministrationProtocol

-r-x  setName(String)
-r-x  setProperty(String, Object)
-r-x  unSet                          Void : String(propertyName)

wls:/12c_domain/serverConfig>
```

子MBean

属性名と値

Q. MBean属性に w が表示されるのはどのMBeanツリー?

A. **EditMBean**ツリー (edit()で移動)

メソッド

MBeanの一覧表示

- JRockit Mission Control -

The screenshot shows the MBean Browser (MBeanブラウザ) interface. On the left, there is a tree view under 'MBeanツリー' with a filter box. The tree view shows a hierarchy of MBeans, with 'AdminServer' selected. On the right, there is a table titled 'MBeanの機能' showing the properties of the selected MBean. The table has two columns: '名前' (Name) and '値' (Value). The table is currently displaying the properties of the 'AdminServer' MBean. Below the table, there are buttons for '検索...' (Search), '視覚化...' (Visualize), '更新...' (Refresh), and 'リフレッシュ' (Refresh).

名前	値
ActivationTime	1341718664993
AdminServer	true
AdminServerHost	localhost
AdminServerListenPort	7001
AdminServerListenPortSecure	false
AdministrationPort	9002
AdministrationPortEnabled	false
AdministrationURL	t3://localhost:7001
ApplicationRuntimes	javax.management.ObjectName[13]
AsyncReplicationRuntime	null
ClusterMaster	false
ClusterRuntime	null
ConnectorServiceRuntime	com.bea.Type=ConnectorServiceRuntime, Serv...
CurrentDirectory	C:¥u01¥work¥domains¥12c_domain¥.
CurrentMachine	
DefaultExecuteQueueRuntime	null
DefaultURL	t3://localhost:7001
EntityCacheCumulativeRuntime	null
EntityCacheCurrentStateRuntime	null
EntityCacheHistoricalRuntime	null

MBeanの調べ方 (ConfigurationMBean)

- config.xmlから推測 -

```
<?xml version="1.0" encoding="UTF-8"?>
<domain xsi:schemaLocation="http://xmlns.oracle.com/weblogic/security/xacml http://xmlns.oracle.com/weblogic/security/xacml/1.0/xacml.xsd
http://xmlns.oracle.com/weblogic/security/providers/passwordvalidator
http://xmlns.oracle.com/weblog
http://xmlns.oracle.com/weblog
http://xmlns.oracle.com/weblog
instance" xmlns:wls="http://xmlns.oracle.com"
xmlns="http://xmlns.oracle.com">
```

複合型子要素=ConfigurationMBean, 単純型子要素=該当MBeanの属性

```
<name>12c_domain</name>
<domain-version>12.1.1.0</domain-version>
+ <security-configuration>
+ <server>
+ <cluster>
<production-mode-enabled>true</production-mode-enabled>
+ <embedded-ldap>
<administration-port-enabl
<configuration-version>12
+ <app-deployment>
+ <library>
+ <machine>
+ <jms-server>
+ <migratable-target>
+ <jdbc-store>
+ <jms-system-resource>
<admin-server-name>Adn
+ <jdbc-system-resource>
+ <restful-management-serv
</domain>
```

```
- <server>
  <name>AdminServer</
  <jms-thread-pool-size>
- <ssl>
  <enabled>true</enabled>
</ssl>
<listen-address>localhost</listen-address>
<xml-registry xsi:nil="true"/>
<xml-entity-cache xsi:nil="true"/>
<jms-default-connection-factories-enabled>false</jms-default-connection-factories-enabled>
- <server-diagnostic-config>
  <wldf-diagnostic-volume>Low</wldf-diagnostic-volume>
</server-diagnostic-config>
- <transaction-log-jdbc-store>
  <data-source>derbyStoreDataSource</data-source>
  <enabled>true</enabled>
</transaction-log-jdbc-store>
</server>
```

ConfigurationMBeanは属性値にMBean(子MBean)を持つものがある

Administration Consoleのチャートとグラフ

The screenshot displays the Oracle WebLogic Server Administration Console 12c interface. The main content area is titled 'ホーム・ページ' (Home Page) and contains several sections:

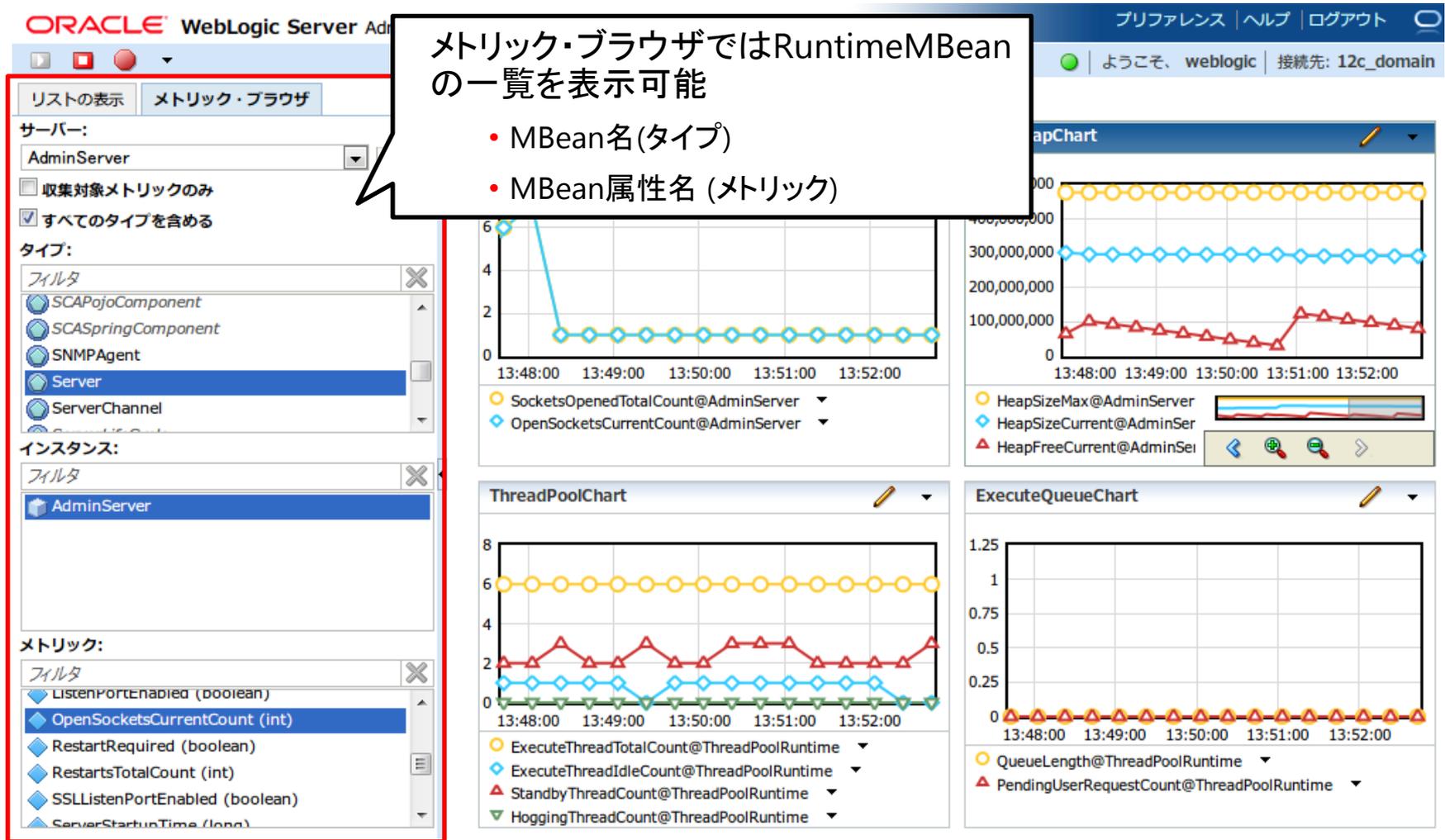
- 情報とリソース** (Information and Resources): Includes links for 'アプリケーションの構成', 'GridLink for RACデータ・ソースの構成', '最近のタスク・ステータス', and 'コンソールのプリファレンス設定'.
- ドメイン構成** (Domain Configuration): Lists 'ドメイン' (Domain).
- 環境** (Environment): Lists 'サーバー', 'クラスタ', '仮想ホスト', '移行可能なターゲット', 'Coherenceサーバー', 'Coherenceクラスタ', 'マシン', 'ワーク・マネージャ', and '起動クラスと停止クラス'.
- サービス** (Services): Lists 'メッセージング' (with sub-items: JMSサーバー, ストア・アンド・フォワード・エージェント, JMSモジュール, バス・サービス, ブリッジ), 'データ・ソース', '永続ストア', 'XMLレジストリ', 'XMLエンティティ・キャッシュ', '外部JNDIプロバイダ', 'ワーク・コンテキスト', 'jCOM', 'メール・セッション', 'FileT3', and 'JTA'.
- 相互運用性** (Interoperability): Lists 'WTCサーバー' and 'Jolt接続プール'.
- 診断** (Diagnosis): Lists 'ログ・ファイル', '診断モジュール', '診断イメージ', 'リクエスト・パフォーマンス', 'アーカイブ', 'コンテキスト', and 'SNMP'.
- チャートとグラフ** (Charts and Graphs): This section is highlighted with a red box and contains the link '監視ダッシュボード' (Monitoring Dashboard).

On the left side, there are several utility panels:

- チェンジ・センター** (Change Center): Includes '変更と再起動の表示' and buttons for 'ロックして編集' and '構成の解放'.
- ドメイン構造** (Domain Structure): A tree view showing the hierarchy of '12c_domain' including '環境', 'デプロイメント', 'サービス', 'セキュリティ・レルム', '相互運用性', and '診断'.
- 操作ガイド** (Operation Guide): Lists tasks like '構成の検索', 'チェンジ・センターの使用', 'WLSTスクリプトの記録', 'コンソールのプリファレンスを変更', 'コンソール拡張の管理', and 'サーバーの監視'.
- システム・ステータス** (System Status): Shows '実行中のサーバーのヘルス' (Health of running servers) with a bar chart and counts for 'Failed (0)', 'Critical (0)', 'Overloaded (0)', 'Warning (0)', and 'OK (1)'.

MBeanの調べ方 (RuntimeMBean)

- 監視ダッシュボード -



MBeanの調べ方 (全MBean)

- WebLogic Server MBean Reference -

The screenshot shows the Oracle WebLogic Server MBean Reference documentation. The left sidebar contains a tree view of MBeans, with 'ApplicationRuntimeMBean' selected under 'Runtime MBeans'. The main content area displays the 'ApplicationRuntimeMBean' page, which includes navigation links for 'Overview', 'Related MBeans', 'Attributes', and 'Operations'. The 'Overview' section provides a description of an application as a J2EE Enterprise application packaged in an EAR file or exploded directory, and lists three states: UNPREPARED, PREPARED, and ACTIVATED. A 'Security roles' section at the bottom indicates that all non-encrypted attributes and operations are accessible to all roles.

Contents | View Bookmarks | Bookmark Current Topic

ApplicationRuntimeMBean

[Overview](#) | [Related MBeans](#) | [Attributes](#) | [Operations](#)

Overview

An application represents a J2EE Enterprise application packaged in an EAR file or EAR exploded directory. The EAR file or directory contains a set of components such as WAR, EJB, and RAR connector components, each of which can be deployed on one or more targets. A target is a server or a cluster. Modules in the application can have one of the following states:

- UNPREPARED - Indicates that none of the modules in this application are currently prepared or active.
- PREPARED - Indicates that none of the modules in this application are currently prepared or active.
- ACTIVATED - Indicates at least one module in this application is currently active.

ApplicationRuntimeMBean encapsulates runtime information about a deployed Enterprise application.

[Security roles](#) | The following roles have read, write, and invoke permission for all non-encrypted attributes and operations in this MBean:

まとめ

- WLSTを使いこなすにはMBeanの理解が重要なポイント
 - MBeanを操作するのでMBeanの属性や階層関係を把握しておく必要がある
 - MBeanの属性や階層関係は、製品ドキュメントなどを参照
 - どこに情報があるのかを把握して必要に応じて参照する

Agenda

- WLSTとは？
- WLSTコマンド
- MBeanの調べ方
- WLSTスクリプティング

Jython構文の基本

- Javaパッケージをインポート可能

```
from java.lang import * #WLSTは標準で実行
import time
```

- メソッドの実行

```
print System.currentTimeMillis()
```

- 文字列変換

```
str(5)
```

- ループ文や分岐文

```
for i in range(1,10)
    print i
```

```
if(flag):
    print 'yes'
else:
    print 'no'
```

WLSTスクリプティングではインデントに注意!!

- Jythonではコードブロックをインデントで表現する ({}などは使わない)

```
if(true):  
    print 'a'  
else:  
    print 'b'  
print 'c'
```



```
if(true):  
    print 'a'  
else:  
    print 'b'  
print 'c'
```

- 空ブロック

```
if(true):  
    print 'a'  
else:  
    pass
```

例1: ロックしているユーザを確認

```
from weblogic.management.mbeanservers.edit import NotEditorException

### define variables
username = 'weblogic'
password = 'welcome1'
adminServer = 't3://localhost:7001'

### connect to the Administration Server
connect(username, password, adminServer)

### obtain the ConfigurationManagerMBean
cfgManager = getConfigManager()
try:
    cfgManager.getChanges()
    print '===> Currently there is a Session'
    if cfgManager.isEditor() == true:
        ### you are making changes!!!
        print '===> Looks like you started that session'
        print '===> You can check the console for any pending changes'
        print '===> Try rerunning this script after you release or commit the pending
changes'
    sys.exit()
```

例1: ロックしているユーザを確認

```
except NotEditorException, e:
    if cfgManager.getCurrentEditor() is None:
        ### no session
        print 'Currently there is no Session'
    else:
        ### someone else is making changes
        userWithSession = cfgManager.getCurrentEditor().replace(' ', '')
        print '==> Currently there is a Session'
        print '==> User ¥"' + userWithSession + '¥" is making the changes'
        print '==> Wait until ¥"' + userWithSession + '¥" complete the current session'
        sys.exit()
except Exception:
    ### other Errors
    print '==> Error, see log for more info'
    sys.exit(1)
```

Q. 他のユーザがロックを取得している場合のロック解除方法は？

A. `cancelEdit()`

※Adminロールのユーザで実行

例2: ThreadPoolRuntimeMBeanの監視

```
### method for obtaining the information of ThreadPoolRuntimeMBean
def getThreadPoolInfo(threadPoolRuntime):
    ## get attribute value
    executeThreadTotalCount = threadPoolRuntime.getExecuteThreadTotalCount()
    executeThreadIdleCount = threadPoolRuntime.getExecuteThreadIdleCount()
    standbyThreadCount = threadPoolRuntime.getStandbyThreadCount()
    activeThreadCount = executeThreadTotalCount - executeThreadIdleCount -
        standbyThreadCount

    ## convert to String
    strTime = str(System.currentTimeMillis()) + ','
    strExecuteThreadTotalCount = str(executeThreadTotalCount) + ','
    strExecuteThreadIdleCount = str(executeThreadIdleCount) + ','
    strStandbyThreadCount = str(standbyThreadCount) + ','
    strActiveThreadCount = str(activeThreadCount) + '¥n'

    ## return as array
    return [strTime, strExecuteThreadTotalCount, strExecuteThreadIdleCount,
        strStandbyThreadCount, strActiveThreadCount]
```

例2: ThreadPoolRuntimeMBeanの監視

```
### method for writing to a file
def writeToFile(oList):
    f = open(fileName, 'a')

    for v in oList:
        print v
        f.write(v)

    f.close()

### define variables (for Connection)
username = 'weblogic'
password = 'welcome1'
admin_url = 't3://localhost:7001'

### define variables (for Scripting)
fileName = 'c:\tmp\threadmonitor.log'
delimiter1 = ','
delimiter2 = '\n'
```

例2: ThreadPoolRuntimeMBeanの監視

```
### connect to server
connect(username, password, admin_url)

### change drive to serverRuntime
serverRuntime()

### obtain ThreadPoolRuntimeMbean
threadPoolRuntime = getMBean('ThreadPoolRuntime/ThreadPoolRuntime')

while(true):
    try:
        ## call method
        threadPoolInfoList = getThreadPoolInfo(threadPoolRuntime)
        writeToFile(threadPoolInfoList)

        ## sleep 5 secs
        Thread.sleep(5000)
    except:
        print '<<<error has been occurred>>>'
```

Administration ConsoleのWLSTスクリプト記録

ホーム ログアウト プリファレンス 記録 ヘルプ ようこそ、weblogic 接続先: 12c_domain
ホーム > 12c_domain > サーバーのサマリー > AdminServer > プリファレンス > WLSTスクリプトの記録

プリファレンス

ユーザー・プリファレンス 共有プリファレンス 拡張 WLSTスクリプトの記録

全般 制御

保存

このページでは、WLSTスクリプトの記録のプリファレンスを指定します。ユーザーごとに独自の記録のプリファレンスを指定できます。

基本スクリプト・ディレクトリ: C:\u01\work\domains\12c_domain

スクリプト・ファイル名: Script1341830476259.py

ファイルの末尾に追加

ロックが必要

自動記録

ツールバーから記録する前にプロンプトを表示

プリスクリプトとポストスクリプトの生成

TIPS. WLST コマンドでもスクリプト記録可能

```
wls:/ > startRecording('script_filename')  
wls:/ > stopRecording()
```

まとめ

- WLSTスクリプティングは定型化された管理タスクを実行する際に効果を発揮
 - Jython構文に基づく
 - 設定変更の際は整合性に注意
 - WLST validateコマンドなどを活用
 - Administration Consoleのスクリプト記録で作成できるテンプレートを活用
 - サンプルスクリプトも参照可能
 - WLST オンラインサンプル
 - `$WL_HOME/samples/server/examples/src/examples/wlst/online`
 - WLST オフラインサンプル
 - `$WL_HOME/common/templates/scripts/wlst`

今回のまとめ

- WLSTはWebLogicドメインの管理をサポートする強力なコマンドライン・スクリプティングツールです
 - ある目的に対して対応するツールは複数存在します。状況に応じて最も効率が良いと思われるツールを選択しましょう。
- WLSTではMBeanをダイレクトに操作するのでMBeanの理解を深める必要があります
 - MBeanの構造が理解できるとWebLogicドメインの設定作業は格段にわかりやすくなります

効率的な運用管理のためにWLSTも活用ください!!

Reference:

- Oracle WebLogic Scripting Tool
 - http://docs.oracle.com/cd/E28613_01/web.1211/b65932/toc.htm
- WebLogic Scripting Tool コマンド・リファレンス
 - http://docs.oracle.com/cd/E28613_01/web.1211/b65931/toc.htm
- Oracle WebLogic Server MBean Reference
 - http://docs.oracle.com/cd/E28613_01/apirefs.1211/e24403/core/index.html
- Oracle WebLogic Server MBean Javadoc
 - http://docs.oracle.com/cd/E28613_01/apirefs.1211/e24395/toc.htm
- The Jython Project
 - <http://www.jython.org/>

Oracle Universityからのお知らせ

- WLSTによるサーバ管理をはじめとしたWebLogic Serverの管理方法を学習したい方に最適な研修コースをご提供しています。
 - Classroomトレーニングだけでなく、Live Virtual Class、『**Oracle トレーニング・オンデマンド**』など多様な受講形態から選択いただけます。

The screenshot shows the Oracle University website interface. At the top left is the Oracle logo. Below it are navigation tabs: 製品 / サービス, ダウンロード, 価格/ライセンス, and サポート. On the right is a search bar with the text 'オラクル 研修' and a magnifying glass icon. Below the navigation is the 'Oracle University' header. On the left is a sidebar menu with links: 100% Student Satisfaction, オラクル認定資格制度, コーススケジュール, Help, Learning Credits, ラーニングパス, キャンペーン, 受講形態, and トレーニング・オンデマンド. The main content area features two promotional banners. The top banner is red and white, titled 'Oracle トレーニング・オンデマンド' and describes a 1-year on-demand training service. The bottom banner is grey and white, titled 'Java SE 7 対応研修 登場!' and features an image of a character holding a red sphere, with text about the release of Java SE 7 and a link to 'Java SE 7 New Features 詳細'.

Oracle WebLogic Server 11g: 管理

Oracle Application Gridの基盤を支える

Oracle WebLogic Server 11gの管理コース！



このコースでは、Web管理者がOracle WebLogic Server 11gのインストールおよび設定する方法について説明します。Web管理者が管理コンソールやコマンドライン、およびスクリプトツール(WLST)などを使用して、Java EEアプリケーションをOracle WebLogic Server 11gにデプロイする方法についても説明します。

その他に、Oracle WebLogic Server のWebインタフェースとしてOracle HTTP Serverを設定する方法を解説し、またOracle WebLogic Serverクラスタを設定してアプリケーションのフェイルオーバーとロードバランシングをサポートする方法を学習します。また、WebLogic Server管理者の管理タスクの概要について説明します。

コース内容	<ul style="list-style-type: none"> ■ Oracle Fusion Middleware の概要 ■ Oracle WebLogic Serverのインストール ■ WebLogic Server ドメインのコンフィグレーション ■ アプリケーションのデプロイ ■ JMS アプリケーションのコンフィグレーション ■ Oracle HTTP Server のコンフィグレーション ■ バックアップおよびリカバリの管理 	<ul style="list-style-type: none"> ■ WebLogic Serverのアーキテクチャ ■ 管理コンソールおよび他の管理ツールの概要 ■ Oracle WebLogic Server の管理およびログgingsの使用 ■ データソース、JDBCドライバ、接続プールの設定 ■ WebLogic Serverの基本セキュリティのコンフィグレーション ■ Oracle WebLogic クラスタのコンフィグレーション ■ 全体バックアップ、増分バックアップ
受講前提条件	<ul style="list-style-type: none"> ・Linux の基本コマンドおよびデスクトップのナビゲーション ・クライアント/サーバーの概念における TCP/IP ネットワークに関する基本的な知識 ・Java EE の基礎知識(サーブレットや JSP など) ※推奨 	
対象者	<ul style="list-style-type: none"> ・Oracle WebLogic Server 11g/12c管理者 ・Javaアプリケーション開発者 	
コース日程	5日間 日程の詳細は Oracle University Webサイト にてご確認ください。	次回開催日程 ■7/9(月) - 7/13(金) トレーニングキャンパス青山 ■8/27(月) - 8/31(金) 三田(芝浦)会場
受講料	定価 ¥363,825(税込) ※Oracle PartnerNetwork会員様は、パートナー割引価格で受講いただけます。	

Hardware and Software Engineered to Work Together

ORACLE®