

OPEN STM Tips

OPEN ACS の応用

1. 背景・目的

本 Tips は、TR-069 の ACS 機能を実装したオープンソースソフトウェアである Open ACS の概要と入手方法、インストール・設定、内部構造、設定スクリプト編集による機能拡張、EJB オブジェクトのリモート化による機能拡張、留意事項について説明する。

2. 対象読者

OPEN STM シリーズのサーバー系プログラム開発者

3. 参考文献・関連文書

Broadband Forum TR-069 Amendment 3 (以降、「TR-069」)

OSSBN-TIPS-13-08-001/02

<http://sourceforge.net/projects/openacs/>

4. その他

本 Tips 中の図表番号につき、参考文献・関連文書からの抜粋には原文の番号をそのまま流用し、独自に作成した図表には”Tips-*”の形式で番号を付与した。

5. 最終更新日

2013 年 10 月 16 日

OPEN STM Tips

OPEN ACS の応用

6. 詳細

6.1 Open ACS の概要と入手方法

Open ACS は、Audrius Valunas が立ち上げた GPLv3 規約に従うオープンソースプロジェクトであり、JBoss 4.2.2 以降、MySQL5.0 以降上で動作する J2EE 標準形式の Web アプリケーションフレームワークである。

Open ACS は、TR-069 が定義する ACS 機能をほぼ全て実装し、デフォルトをカスタマイズしたスクリプトファイルの修正により、制御・管理ロジックを柔軟に追加できる。

以下のサイトから本 Tips 執筆時点で最新版の openacs-bin-0.5.0.3.zip を入手できる。

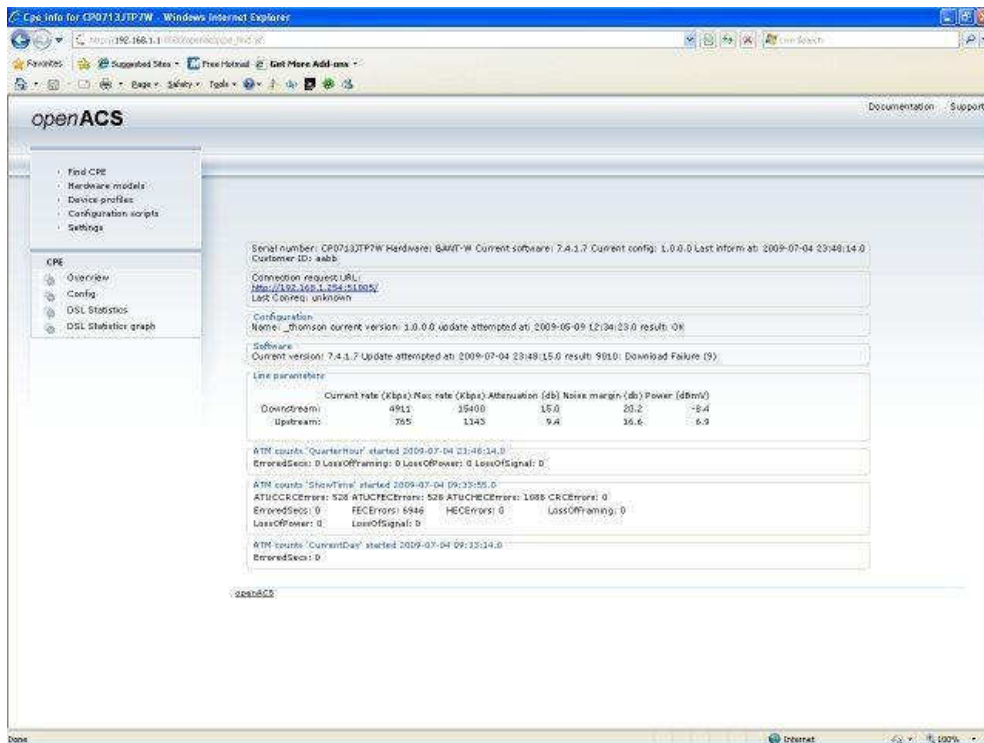
<http://sourceforge.net/projects/openacs/files/openacs-binary/>

Open ACS 実行形式アーカイブの内容を Tips-001 に示す。

ファイル名	内容
acs.ear	Open ACS プログラム本体
openacs-ds.xml	データベース接続設定。MySQL の接続 URL とユーザ ID/パスワード、接続プーリング設定。
openacs-service.xml	JBoss 動作用の MBean 設定。
README.txt	JBoss と JRE のバージョン情報および、簡単なデプロイ&設定方法の説明。
Running.html	上記 README.txt の補完情報 (MySQL の簡単な設定方法及び、XML 設定例に基づく README.txt 説明の重複)。

Tips-001 Open ACS 実行形式アーカイブの内容

SourceForge のスクリーンショットサンプルを Tips-002, Tips-003 に示す。



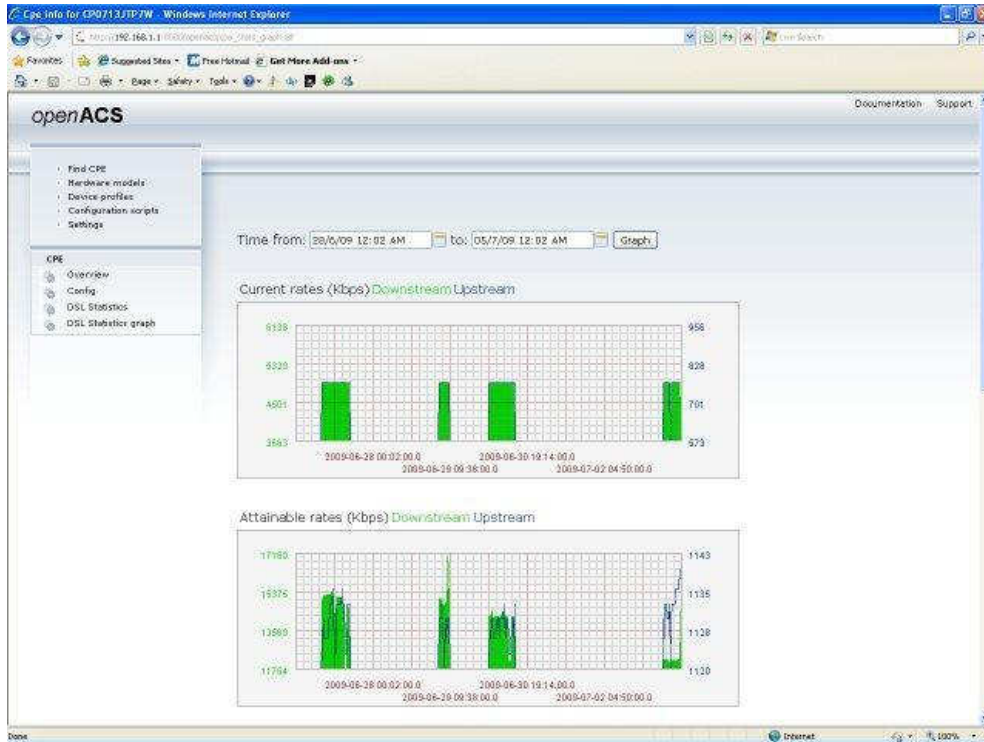
Tips-002 Open ACS スクリーンショットサンプル-1

文書番号: OSSBN-TIPS-13-10-001/02

All Rights Reserved, Copyright © OSS BroadNet 2013

OPEN STM Tips

OPEN ACS の応用



Tips-003 Open ACS スクリーンショットサンプル-2

BBF TR データモデルへの対応状況は、以下の通りである。

TR-069, TR-098, TR-104, TR-106, TR-135, TR-140, TR-143, TR-157, TR-181-1/2,
TR-196, TR-262

詳細は、ソースツリーの/trunk/acs/acs-war/web/WEB-INF/datamodel/配下を参照されたい。

OPEN STM Tips

OPEN ACS の応用

6.2 Open ACS のインストール・設定

Open ACS は JDK1.5, JBoss 4.2.2 環境で開発されているが、いずれのミドルウェアも既にサポートが終了している。このため本 Tips では、保守性と互換性を考慮し、JDK6, JBoss AS5.1GA を前提にインストール・設定方法を説明する。

(1) JDK のインストール

Java SE Development Kit.6u45 を入手・インストールする。

<http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase-6-419409.html#jdk-6u45-oth-JPR>

Linux の yum インストールを使う場合、以下のコマンドを実行する。

```
root@localhost [~]# yum install java-1.6.0-sun-devel
```

(2) JBoss のインストール

JBoss AS 5.1 環境で Open ACS を動作させるには、JBoss 4.2.3 に含まれる一部のファイルが必要である。このため、まずは JBoss 4.2.3 と JBoss AS 5.1 の双方を入手・解凍後、JBoss のインストール・設定を行う。

JBoss 4.2.3 と JBoss AS 5.1 を入手・解凍する。

<http://sourceforge.net/projects/jboss/files/JBoss/JBoss-4.2.3.GA>

```
root@localhost [~]# unzip jboss-4.2.3.GA-jdk6.zip
```

<http://sourceforge.net/projects/jboss/files/JBoss/JBoss-5.1.0.GA/jboss-5.1.0.GA-jdk6.zip/download>

```
root@localhost [~]# unzip jboss-5.1.0.GA-jdk6.zip
```

JBoss 4.2.3 から commons-httpclient.jar を取り出し、JBoss AS 5.1 の同一フォルダにコピーする。

```
root@localhost [~]# cp jboss-4.2.3.GA/server/default/lib/commons-httpclient.jar
```

```
jboss-5.1.0.GA/server/default/lib
```

MySQL用の JDBC ドライバである mysql-connector-java-5.1.26-bin.jar を入手し、JBoss AS 5.1 の lib フォルダにコピーする。

```
root@localhost [~]# cp /my-download-path/mysql-connector-java-5.1.26-bin.jar
```

```
jboss-5.1.0.GA/server/default/lib
```

OPEN STM Tips

OPEN ACS の応用

EJB デプロイ定義ファイルを開き、strictVerifier の値を true から false に変更する。

```
root@localhost [~]# vi jboss-5.1.0.GA/server/default/deployers/ejb-deployer-jboss-beans.xml
```

```
...
```

```
<property name="strictVerifier"><strong>>false</strong></property>
```

```
</bean>
```

```
...
```

JBoss 4.2.3 の/lib/endorsed の内容を JBoss AS 5.1 の同一フォルダにコピーする。

```
root@localhost [~]# cp jboss-4.2.3.GA/lib/endorsed/* jboss-5.1.0.GA/lib/endorsed
```

デフォルトで-Djava.endorsed.dirs オプションパスが未設定の場合、jboss-5.1.0.GA/bin 内の run.conf ファイルの最後に以下の行を追加する。

```
JAVA_OPTS="$JAVA_OPTS -Djava.endorsed.dirs=<YOUR_JBOSS_HOME>/lib/endorsed"
```

※<YOUR_JBOSS_HOME>は、JBoss インストールディレクトリ絶対パスから変更する事。

(3) MySQL のインストール

Linux の yum インストールを使う場合、以下のコマンドを実行する。

```
[root@linux ~]# yum install mysql-server
```

データディレクトリのパスは、/var/lib/mysql に設定される。

/etc/my.cnf をエディタで開き、デフォルト文字コード設定を追加する。

```
[root@linux ~]# vi /etc/my.cnf
```

```
[mysqld]
```

```
...
```

```
# 以下を追記する。
```

```
default-character-set=utf8
```

```
skip-character-set-client-handshake
```

```
...
```

```
[client]
```

```
# 以下を追記する。
```

```
default-character-set=utf8
```

```
...
```

```
[mysql]
```

```
# 以下を追記する。
```

```
default-character-set=utf8
```

```
...
```

文書番号: OSSBN-TIPS-13-10-001/02

All Rights Reserved, Copyright © OSS BroadNet 2013

OPEN STM Tips

OPEN ACS の応用

```
[mysqldump]
```

```
# 以下を追記する。
```

```
default-character-set=utf8
```

```
...
```

自動起動を ON にする。

```
[root@a ~]# chkconfig mysqld on
```

```
[root@a ~]# chkconfig --list mysqld
```

```
mysqld          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

MySQL を起動する。

```
[root@linux ~]# /etc/rc.d/init.d/mysqld start
```

```
MySQL データベースを初期化中: [ OK ]
```

```
MySQL を起動中: [ OK ]
```

ルートユーザにパスワードを設定する。

```
[root@linux ~]# mysqladmin -u root password '設定するパスワードを入力'
```

MySQL に接続する。

```
[root@linux ~]# mysql -u root -p
```

```
Enter password: 'パスワードを入力'
```

MySQL データベースを作成する。

```
mysql> create database ACS DEFAULT CHARACTER SET utf8 ;
```

Open ACS 用のユーザアカウントを作成する。

本例ではユーザ名を'ACS'、パスワードを'ACSpwd'とする。

```
mysql> create user ACS identified by 'ACSpwd';
```

ACS データベースに関する全権限をユーザ ACS に付与する。

```
mysql> grant all on ACS.* to 'ACS' identified by 'ACSpwd';
```

MySQL への接続を終了する。

```
mysql> quit
```

OPEN STM Tips

OPEN ACS の応用

(4) Open ACS のインストールと環境設定

openacs-bin-0.5.0.3.zip を以下の URL から入手・解凍する。

<http://sourceforge.net/projects/openacs/files/openacs-binary/>

解凍されたファイルより、acs.ear と openacs-ds.xml を JBoss AS 5.1 インストールディレクトリの server/default/deploy フォルダにコピー後、openacs-ds.xml を編集し、(3)で行った MySQL の設定内容に合わせる。

```
<?xml version="1.0" encoding="UTF-8"?>
<datasources>
  <local-tx-datasource>
    <jndi-name>ACS</jndi-name>
    <connection-url>jdbc:mysql://localhost/ACS</connection-url>
    <driver-class>com.mysql.jdbc.Driver</driver-class>
    <user-name>ACS</user-name>
    <password>ACSpwd</password>
    <min-pool-size>5</min-pool-size>
    <max-pool-size>20</max-pool-size>
    <idle-timeout-minutes>5</idle-timeout-minutes>
  </local-tx-datasource>
</datasources>
```

解凍されたファイルより、openacs-service.xml を JBoss AS 5.1 インストールディレクトリの server/default/deploy/jms フォルダにコピーする。

JBoss AS 5.1 ルートフォルダの bin に移動後、run.sh を実行し、JBoss AS 5.1 を開始する。

この時-b 0.0.0.0 指定により、利用可能な全インタフェースをリスンする。

```
root@localhost [~]# cd jboss-5.1.0.GA/bin
```

```
root@localhost [~]# ./run.sh -b 0.0.0.0
```

(5) HTML ブラウザから ACS 管理画面への接続

HTML ブラウザは以下の URL により、ACS の Web 管理画面に接続できる。

<http://<ACSサーバーのIPアドレス>:8080/openacs/index.jsf>

OPEN STM Tips

OPEN ACS の応用

(6) CPE から ACS への接続

対象 CPE に、以下の ACS-URL を設定する。

`http://<ACS サーバーの IP アドレス>:8080/openacs/acs`

ACS-URL は、DHCP オプションないしは CPE デフォルトにより設定する。

必要時、CPE に ACS への周期 Inform を設定する。

ACS サーバーは CPE の Inform RPC に応答後、対象 CPE 機器に関する情報をデータベースに記録する (org.openacs.AutoCreateCPE コンテキストパラメータにより設定可能)。

Inform 受信時、ACS サーバーは通知されたソフトウェアと設定ファイルのバージョン情報を照合し、必要時にこれらを更新する設定スクリプトを実行する。

CPE のソフトウェア更新は、イメージファイルのダウンロードにより行われる。イメージファイルは予め/firmware フォルダに配置 (org.openacs.fwbase のコンテキストパラメータにより変更可) し、JBoss 実行ユーザに更新権限を与えておく。

CPE の設定は、ベンダー固有 CPE 設定ファイルのダウンロード、ないしは ACS スクリプト設定のいずれかの方法により行われる。CPE 設定ファイルのダウンロードには Download RPC を使用する。CPE 設定ファイルには、CPE からの Inform 情報と同一のバージョン番号が含まれる。

OPEN STM Tips

OPEN ACS の応用

6.3 Open ACS の内部構造

Open ACS は、2 個のステートレスセッション Bean と 19 個の CMP により構成される。Open ACS の EJB 一覧を Tips-004 に示す。

EJB 名	EJB タイプ	説明
CPEBean	Session, Stateless	Bean 管理トランザクション。
OpenACS		同上。
ScriptBean	Entity, Container	Inform 受信時の ACS 動作の設定スクリプト。
DataModelBean		データモデル名。
HostsBean		ホスト。製造シリアル等の機器固有情報。検出・登録時に ID(PK)を自動生成。
SoftwareBean		(CPE)ソフトウェア名。
SoftwareDetailBean		ソフトウェア名とハードウェアモデルの紐付。
HardwareModelBean		ハードウェアモデル。検出・登録時に ID(PK)を自動生成。
ConfigurationBean		(CPE)設定ファイル名。
DeviceProfileBean		デバイスプロファイル。Inform 間隔等の機器共通設定。
DeviceProfile2SoftwareBean		デバイスプロファイルとソフトウェアの紐付。
DSLStatsBean		DSL 状態情報。
ATMErrorStatsBean		ATM エラー状態情報。
PropertyBean		プロパティ。
ServiceBean		サービス。検出・登録時に ID(PK)を自動生成。
BackupBean		バックアップ。
OuiMapBean		OUI マップ。
HostPropertyBean		ホストプロパティ。
ProfilePropertyBean		プロファイルプロパティ。
ServicePropertyBean	サービスプロパティ。	
Host2ServiceBean	ホストとサービスの紐付。	

Tips-004 Open ACS の EJB 一覧

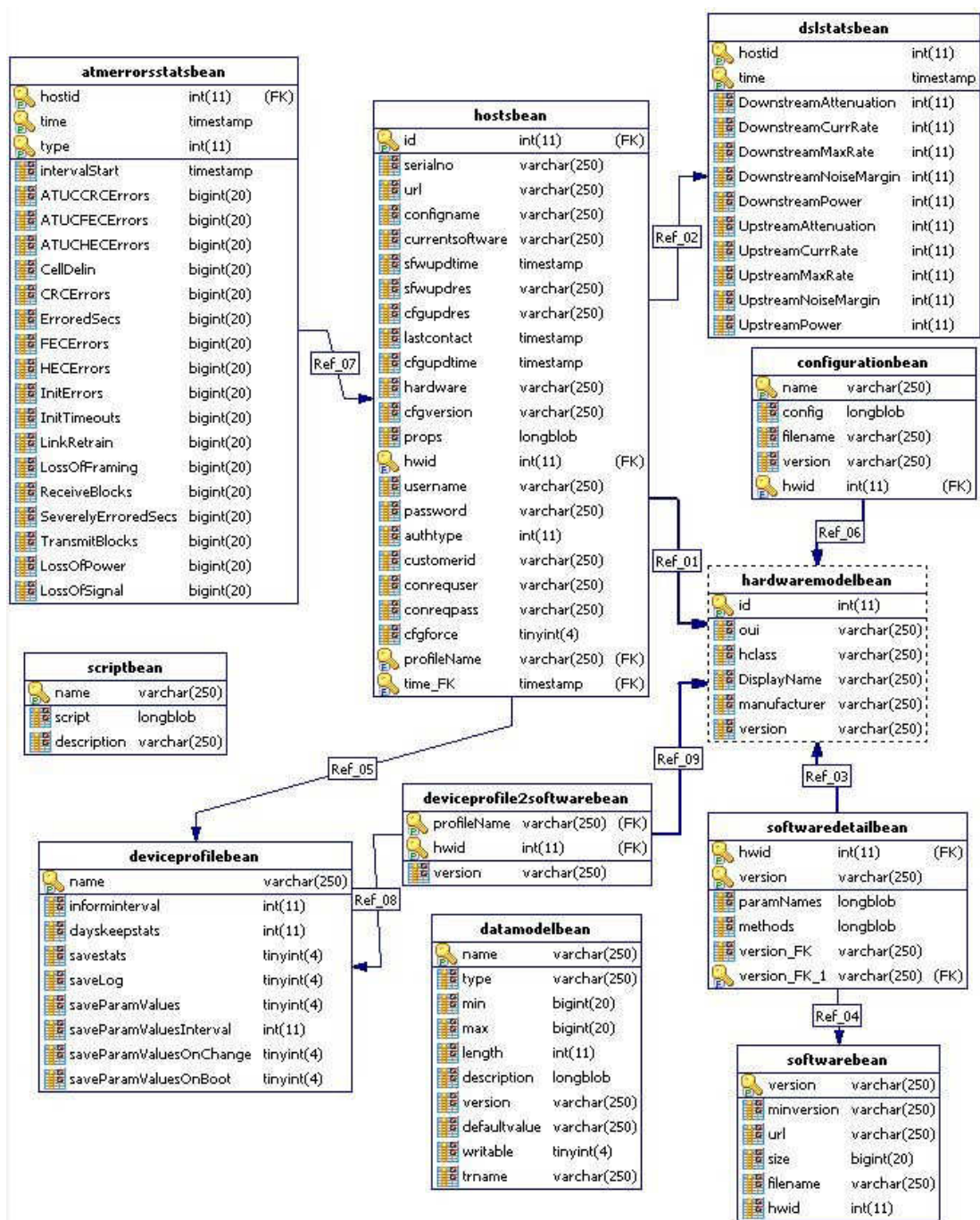
Open ACS では J2EE 仕様に従い、各 Bean の EJB 名、タイプ、属性、構成項目、相互関係等が ejb-jar.xml に定義されている。各エンティティ Bean は、jbosscomp-jdbc.xml のデフォルトである create table = true, alter table = true 設定に従い、EAR ファイルデプロイ時に ejb-jar.xml の定義に従いテーブルオブジェクトとして自動生成される。

上位 OSS/BSS への自動検知 CPE 情報の夜間バッチ取込等、JBoss-MySQL 間の更新タイムラグが許容できる場合、MySQL の SQL 参照が有効である。一方で、自動検知 CPE 情報を外部システムへ迅速に通知する必要がある場合、後述する Inform 受信時の設定スクリプトを応用し、外部へイベント情報を出力する方法が有効である。

OPEN STM Tips

OPEN ACS の応用

MySQL テーブルの ER 図(一部)を Tips-005 に示す。



Tips-005 MySQL テーブルの ER 図 (一部)

OPEN STM Tips

OPEN ACS の応用

DataModel テーブルのレコード例を Tips-006 に示す。

name	type	min	max	length	description	version	defaultvalue	writable	tname
STBService (i).ServiceMonitor	object	-922337203685...	9223372036854	21474836...					0 TR-135
STBService (i).ServiceMonitor.AVStream	string	922337203685...	9223372036854	256			-		0 TR-135
STBService (i).ServiceMonitor.Enable	boolean	-922337203685...	9223372036854	21474836...			-		1 TR-135
STBService (i).ServiceMonitor.Gmin	unsignedInt	-922337203685...	9223372036854	21474836...			-		1 TR-135
STBService (i).ServiceMonitor.ServiceType	string	-922337203685...	9223372036854	21474836...			-		1 TR-135
STBService (i).ServiceMonitor.SevereLossMinDistance	unsignedInt	-922337203685...	9223372036854	21474836...			-		1 TR-135
STBService (i).ServiceMonitor.Status	string	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring	object	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.FetchSamples	unsignedInt	-922337203685...	9223372036854	21474836...			-		1 TR-135
STBService (i).ServiceMonitoring.ForceSample	boolean	-922337203685...	9223372036854	21474836...			-		1 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample	object	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.AudioDecoderStats	object	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.AudioDecoderStats.DecodedFrames	string	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.AudioDecoderStats.DecodingErrors	string	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.AudioDecoderStats.SampleSeconds	string	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.DejitteringStats	object	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.DejitteringStats.EmptyBufferTime	string	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.DejitteringStats.Overruns	string	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.DejitteringStats.SampleSeconds	string	-922337203685...	9223372036854	21474836...			-		0 TR-135
STBService (i).ServiceMonitoring.MainStream (i).Sample.DejitteringStats.Underruns	string	-922337203685...	9223372036854	21474836...			-		0 TR-135

Tips-006 DataModel テーブルのレコード例

OPEN STM Tips

OPEN ACS の応用

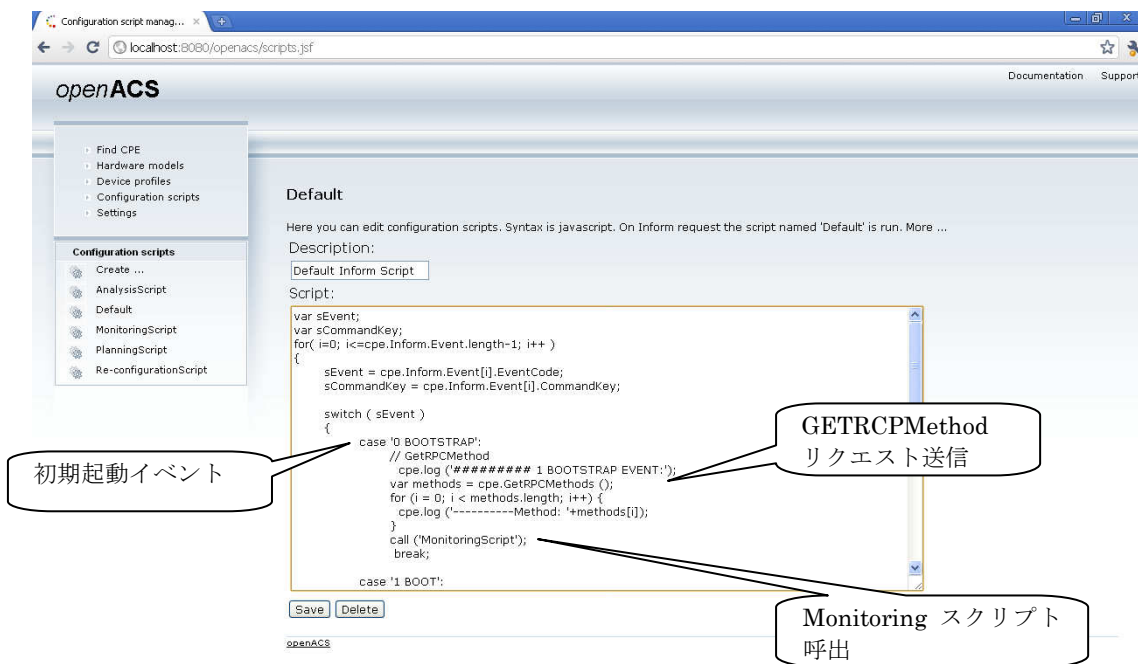
6.4 設定スクリプト編集による機能拡張

(1) デフォルト設定スクリプトの定義画面

Open ACS は、CPE からの Inform メッセージを受信時、デフォルト設定スクリプトを実行する。

デフォルト設定スクリプトへの処理追加とサブスクリプト呼出により、受信 Inform イベントのタイプに応じた様々な個別処理をカスタマイズできる。

Open ACS の設定スクリプト定義画面サンプルを Tips-007 に示す。



Tips-007 Open ACS の設定スクリプト定義画面サンプル

Tips-007 では、デフォルト設定スクリプトの Bootstrap(初期起動)イベント時の動作を表示している。

本例では GETRPCMethods RPC リクエストを送信し、対象機器で操作可能な RPC メソッド一覧を取得後、独自にカスタマイズ作成した Monitoring スクリプトを呼び出して取得状態値で対象機器を監視している。

カスタマイズスクリプトを組み合わせれば、PERIODIC(周期)イベントでカスタマイズ分析スクリプトを呼び出して過去の履歴を検索・分析後、Set RPC により対象機器の設定を最適値に変更する、等の自動設定運用が実現できる。

OPEN STM Tips

OPEN ACS の応用

(2) Inform メッセージの構造

CPE の Inform メッセージ例を Tips-008 に示す。

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cwmp="urn:dslforum-org:cwmp-1-0" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header>
<cwmp:ID>123</cwmp:ID>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
<cwmp:Inform>
<cwmp:DeviceID xsi:type="cwmp:DeviceIdStruct">
<DeviceIdStruct>
<Manufacturer xsi:type="xsd:string(64)"> Cisco</ Manufacturer>
<OUI xsi:type="xsd:string(6)"> 00000C</OUI>
<ProductClass xsi:type="xsd:string(64)">NetDevice</ProductClass>
<SerialNumber xsi:type="xsd:string(64)"> 4231278</SerialNumber>
</DeviceIdStruct>
</cwmp:DeviceID>
<cwmp:Event SOAP-ENV:arrayType="xsd:EventStruct[3]">
<EventStruct>
<EventCode>0 BOOTSTRAP</EventCode>
<CommandKey/>
</EventStruct>
<EventStruct>
<EventCode>1 BOOT</EventCode>
<CommandKey/>
</EventStruct>
</cwmp:Event>
<cwmp:MaxEnvelopes xsi:type="xsd:int">1</cwmp:MaxEnvelopes>
<cwmp:CurrentTime xsi:type="xsd:dateTime">Fri Aug 13 10:42:32 EDT
2010</cwmp:CurrentTime>
<cwmp:RetryCount xsi:type="xsd:int">0</cwmp:RetryCount>
<cwmp:ParameterList SOAP-ENV:arrayType="xsd:ParameterValueStruct[1]">
<ParameterValueStruct>
<Name>InternetGatewayDevice.DeviceSummary</Name>
<Value xsi:type="xsd:string"> InternetGatewayDevice (Baseline:1,
EthernetLAN:1, ADSLWAN:1)</Value>
</ParameterValueStruct>
</cwmp:ParameterList>
</cwmp:Inform>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Tips-008 CPE の Inform メッセージ例（初期起動イベント）

Inform メッセージの送信契機は、単一ないしは2つのイベントコード(EventCode)により表現される。本例ではイベントコードに”0 BOOTSTRAP”、”1 BOOT”が格納されているため、CPE 初回起動時のブートストラップイベントの Inform メッセージである事が分かる。

OPEN STM Tips

OPEN ACS の応用

Inform メッセージの標準イベントコード一覧を Tips-009 に示す。

イベントコード	送信契機
0 BOOTSTRAP	再起動後の最初の ACS とのコンタクト
1 BOOT	リブート
2 PERIODIC	周期インターバル後
3 SCHEDULED	事前予約の日付時刻到来・1 回のみ
4 VALUE CHANGE	更新時 ACS 通知が設定された項目の更新
5 KICKED	N.A.(TR-069 非推奨)
6 CONNECTION REQUEST	ACS からの接続要求受付
7 TRANSFER COMPLETE	Upload/Download の完了
8 DIAGNOSTICS COMPLETE	診断の完了
9 REQUEST DOWNLOAD	CPE による Download 要求
10 AUTONOMOUS TRANSFER COMPLETE	Upload/Download の完了
11 DU STATE CHANGE COMPLETE	DUStateChange の完了
12 AUTONOMOUS DU STATE CHANGE COMPLETE	Request によらず自律的に実行した DUStateChange の完了
M REBOOT	0~10 発生の原因 (必ず他イベントに伴う)
M SCHEDULED INFORM	同上
M DOWNLOAD	同上
M UPLOAD	同上
M DU STATE CHANGE	同上

Tips-009 Inform メッセージの標準イベントコード一覧

OPEN STM Tips

OPEN ACS の応用

(3) デフォルト設定スクリプトの文法構造

基本文法構造は JavaScript に従う。以下に利用可能な関数、変数、メソッドを示す。

(a) ログ出力

以下のログ出力関数により、JBoss の標準ログに出力される。

```
logger ('My message'); // message with default severity WARNING is logged.
```

```
logger ('severe', 'Error message');
```

```
logger ('warning', 'Warning message');
```

```
logger ('info', 'Informational message');
```

(b) 他スクリプトの呼び出し

以下の書式に従う。

```
call ('script_name');
```

(c) cpedb 変数

CPEDB グローバル変数は、データベース中の CPE ビーンの各プロパティ項目への JavaScript によるアクセスを提供する。データベースに値を更新する場合、Save を使用する。

```
var someprop = cpedb.someprop;
```

```
cpedb.someprop = 'new_value_for_someprop';
```

```
cpedb.Save ();
```

(d) cpe 変数

CPE グローバル変数により、TR-069 に定義される全ての CPE 変数と RPC 関数にアクセス可能である。

(e) 使用可能なメソッド

GetRPCMethods ()

CPE がサポートするメソッド配列を返す。

```
var methods = cpe.GetRPCMethods ();
```

```
for (i = 0; i < methods.length; i++) {
```

```
    logger ('Method: '+methods[i]);
```

```
}
```

OPEN STM Tips

OPEN ACS の応用

Download (commandKey, what_to_download, url, username, password, file_size, file_name)

CPE に設定ファイルまたはファームウェアのダウンロード開始を知らせる。

```
var response = cpe.Download ("daCommand", "3 Vendor Configuration  
File","http://192.168.1.1:8080/kkonf", "", "",000,"user.ini");  
logger ("st="+response.StartTime+" ct="+CompleteTime+" status="+response.Status);
```

GetParameterValues (array_of_parameter_names)

プロパティ名・値を含むオブジェクト配列を返す。

```
var parameters = new Array ();  
parameters[0] = 'InternetGatewayDevice.DeviceSummary';  
var response = cpe.GetParameterValues (parameters);  
logger (response[0].name+'='+response[0].value);
```

SetParameterValues (params_array, commandKey)

パラメータを設定する。Params_array はプロパティ名・値を含むオブジェクト配列。

```
var parameters = new Array ();  
parameters[0] = {name: 'InternetGatewayDevice.IPPingDiagnostics.Host', value: '192.168.0.1'};  
// Encode parameter using default type xsd:string  
parameters[1] = {name: 'InternetGatewayDevice.IPPingDiagnostics.NumberOfRepetitions', value: '2'};  
// set encoding type  
parameters[2] = {name: 'InternetGatewayDevice.IPPingDiagnostics.NumberOfRepetitions', value: '2', type:  
'xsd:unsignedInt'};  
cpe.SetParameterValues (parameters, "commandKey");
```

AddObject (top_object_name, parameterKey)

新たなオブジェクトインスタンスを作成する。状態・インスタンス番号のプロパティと一緒にオブジェクトを返す。

DeleteObject (object_name, parameterKey)

オブジェクトを削除する。整数値の結果コードを返す。

OPEN STM Tips

OPEN ACS の応用

SetParameterAttributes (params_array)

パラメータの属性を設定する。返り値なし。

```
var parameters = new Array();
parameters[0]=new Object;
parameters[0].Name='InternetGatewayDevice.IPPingDiagnostics.NumberOfRepetitions';
parameters[0].Notification=0;
parameters[0].NotificationChange=true;
parameters[0].AccessListChange=true;
parameters[0].AccessList= new Array ();
parameters[0].AccessList[0]='subscriber';
cpe.SetParameterAttributes (parameters);
```

GetParameterAttributes (parameter_names_array)

パラメータの属性を取得する。

```
var p=new Array();
p[0]='InternetGatewayDevice.ManagementServer.PeriodicInformEnable';
var r = cpe.GetParameterAttributes (p);
```

Reboot (commandKey)

CPE をリブートする。

```
cpe.Reboot ("commandKey");
```

FactoryReset

CPE を工場出荷設定にリセットする。

```
cpe.FactoryReset ();
```

ScheduleInform (seconds_to_delay, commandKey)

CPE に一度限りの Inform メソッド呼出予約を依頼する。

```
cpe.ScheduleInform (3600, "commandKey"); // schedule inform in one hour
```

SyncParameterValues

CPE の各パラメータ値を確認し、ACS の設定値と同期する。

```
var parameters = new Array ();
parameters[0] = {Name: 'InternetGatewayDevice.ManagementServer.PeriodicInformEnable', Value: '1'};
parameters[1] = {Name: 'InternetGatewayDevice.ManagementServer.PeriodicInformInterval', Value: '300'};
cpe.SyncParameterValues (parameters);
```

OPEN STM Tips

OPEN ACS の応用

(f) db 変数

データベース操作オブジェクトであり、SELECT/INSERT/UPDATE 操作を CMP および MySQL テーブルに対して実行できる。

db.Query (data_source_name, select_statement)

SELECT 相当。項目名を含むオブジェクト配列を返す。

db.Query ("java:ACS", select_statement)

```
try {
    var rs = db.Query ("SELECT id,serialno FROM hostsbean")
    logger ("Rows found = "+rs.length)
    for (i=0; i < rs.length; i++) {
        logger ("id = "+rs[i].id+" serialno= "+rs[i].serialno);
    }
} catch (e) {
    logger ("DS exception: "+e.message)
}
```

db.Update (data_source_name, update_statement)

INSERT または UPDATE 相当。更新行数を返す。

db.Update ("java:ACS", update_statement)

```
try {
    var rs = db.Update ("INSERT id,serialno INTO hostsbean VALUES (NULL,'12345')")
} catch (e) {
    logger ("DS exception: "+e.message)
}
```

OPEN STM Tips

OPEN ACS の応用

6.5 EJB オブジェクトのリモート化による機能拡張

前節に述べた設定スクリプト編集による機能拡張は、CPE からの受信イベントを起点とする処理、具体的には設定ファイル更新による設定変更やファイルダウンロードによるソフトウェア更新、CPE オンライン化の自動検知、定期的な CPE 状態情報や視聴率の収集等に有効である。

しかしながら、ACS 側からの随時操作を起点とするリアルタイム処理、例えば上位システムからの呼出により CPE から特定の設定・状態値を取得・更新して結果を呼出元に返す類の処理(SNMP マネージャの GET/SET 相当)の場合、設定スクリプト編集では上位システム側から対象 CPE との通信セッションに所望の項目・操作を指定する手段がなく不適切である。

上述のようなリアルタイム処理には、Northbound API (上位 API) によるプログラム間連携が有効である。

OSSBN では、Open ACS の一部 EJB メソッドをリモート化し、OPEN STM 等の上位アプリケーション向けの Northbound API を設計・開発している。

Northbound API については、他の関連 Tips を参照されたい。

OPEN STM Tips

OPEN ACS の応用

6.6 留意事項

(1) 未対応 RPC メソッド

Open ACS 0.5.3 は、TR-069 Table 5 に定義される RPC メソッド中、以下の各 Optional メソッドに非対応である。

- ScheduleDownload
- GetAllQueuedTransfers (但し、TR-069 非推奨の GetQueuedTransfer に対応)
- CancelTransfer
- ChangeDUState
- SetVouchers (TR-069 非推奨)
- AutonomousTransferComplete
- DUStateChangeComplete
- AutonomousTransferComplete
- AutonomousDUStateChangeComplete
- RequestDownload
- Kicked (TR-069 非推奨)

TR-069 の Required メソッドのみで業務要件を満たせず、上の Optional メソッドのいずれかが必要となる場合、java.org.openacs.messages 配下に、各 RPC メソッドを表す message クラス追加と、対応する画面・設定スクリプトの改造が必要になる。

なお下線は、JCL SPEC-023 が必須指定しているメソッドを表す。すなわち、JCL SPEC-023 に全て準拠したい場合、プログラム改修が必須となる。

但し STB 運用時、上に示す Open ACS 非対応の機能が全て必要になるとも思えないので、実際のシステム構築に際しては、プログラム改修に先立ち、まずは現実的な業務要件に沿った As Is での運用可否の検討を推奨する。

(2) ACS 管理画面の日本語化

Open ACS は、ACS-WAR 中の WEB フォルダに格納される各 xhtml ファイル内の表示文字列の置換により日本語化が可能だが、文字列リソースがプロパティファイル等の形で分離されていないため、Open ACS のバージョンアップ時に都度修正が必要となる点に注意されたい。

(3) 状態情報の履歴管理

Open ACS には、NMS のように状態履歴を長期間保持・検索・分析する機能はない。

(4) 認証方式

Open ACS はアプリケーションコードでログイン認証、ダイジェスト認証に対応済だが、証明書認証の構成時、JBoss AS への SSL/TLS 設定と暗号鍵の生成・登録が必要になる。

JBoss AS への SSL/TLS 設定方法については、JBoss の関連文書を参照されたい。

以上