

ユーザーの自動登録シナリオ向けのカスタム承認プロセス

1. 概要

このサンプルでは、カスタム承認プロセスを作成し、これを使用してユーザーの自動登録リクエストを承認する方法について説明します。

このサンプルはOracle Identity Manager 11g (11.1.1.3.0) を使用して開発されました。

2. シナリオ

このサンプルでは、ユーザーの自動登録を承認するシナリオについて説明します。リクエスト・レベルの承認タスクは、**SYSTEM ADMINISTRATORS** ロールに割り当てる必要があります。運用レベルの承認タスクは、リクエスト・レベルの承認で提供された組織の値に応じて、組織の管理者に割り当てる必要があります。

3. 要件/前提条件

このユースケースを開始する前に、使用するマシン上に次のソフトウェアをインストールする必要があります。

1. Oracle Identity Manager Managed Server 11g (11.1.1.3.0)
2. Oracle SOA Composite Editor拡張を含むOracle JDeveloper 11g (11.1.1.3.0)

この例に必要なファイルを含むzipファイルは、[こちら](#)からダウンロードできます。

また、Oracle Identity Managerの管理コンソールから次のタスクを実行する必要があります。

1. FINANCEという名前の組織を作成します。
2. FINANCE_APPROVERSという名前のロールを作成します。
3. Danny Craneという名前のエンドユーザーを作成します。
4. Danny CraneにFINANCE_APPROVERSロールを割り当てます。
5. FINANCE組織の管理者ロールに対して、FINANCE_APPROVERSロールを割り当てます。
6. Shirley Schmidtという名前のユーザーを作成します。
7. Shirley SchmidtにSYSTEM ADMINISTRATORSロールを割り当てます。

さらに、Oracle Enterprise Manager (Oracle EM) コンソールから次のタスクを実行する必要があります。

1. Shirley Schmidtの資格証明を資格証明ストア・フレームワーク (CSF) に保存します。これらの資格証明は、カスタム・コンポジットの作成中にJava Embeddingアクティビティによって読み取られます (CSFに資格証明を保存する方法について、詳しくは付録Bを参照してください)。

4. 頭字語

BEAHOME	Oracle Identity Managerインストールでのミドルウェア・ディレクトリのパス
OIMHOME	Oracle Identity Managerホーム・ディレクトリのパス (例: <BEAHOME>/Oracle_IDM1)
SOAHOME	SOAホーム・ディレクトリのパス (例: <BEAHOME>/Oracle_SOA1)
WLS_DOMAIN	WebLogicドメインの名前 (例: base_domain)
SOA_SERVER	SOA管理対象サーバーの名前

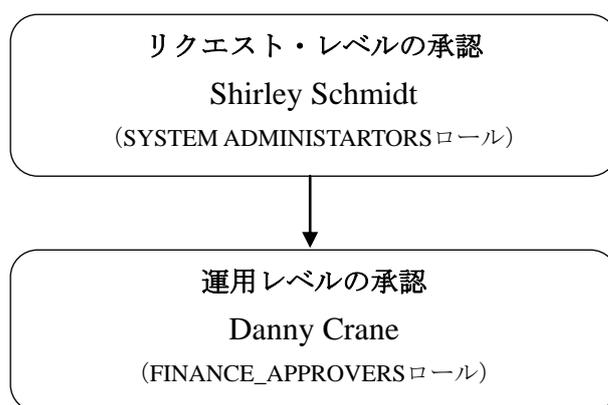
5. カスタム承認プロセスの設計

1. 承認者は誰にすべきか。

運用レベルの承認では、承認者はユーザーの登録先組織の管理者である必要があります。

2. この承認プロセスはどのようなリクエストに対して使用されるか。

ここでのカスタム承認プロセスは、すべてのユーザー自動登録リクエストに対する運用レベルの承認に使用されます。



6. カスタム承認プロセスの開発

運用レベルの承認では、デフォルトで、ユーザーの自動登録タスクがxelsysadmに割り当てられています。ここでのシナリオに従うと、リクエスト・レベルのユーザーの自動登録タスクはすべて、SYSTEM ADMINISTRATORSロールに割り当てる必要があります。また、運用レベルのユーザーの自動登録タスクはすべて、リクエスト・レベルの承認でユーザーに割り当てられた組織の管理者に割り当てる必要があります。

リクエスト・レベルの承認に対しては、Oracle Identity Managerに付属しているDefaultRoleApprovalコンポジットを使用できます。DefaultRoleApproval承認プロセスでは、SYSTEM ADMINISTRATORSロールにタスクが割り当てられます。

運用レベルの承認に対しては、カスタム承認プロセスを作成する必要があります。カスタム承認プロセスを作成するには、次の手順を実行します。

1. SOA複合アプリケーションの作成
2. JDeveloperを使用したSOAコンポジットの変更
3. Oracle SOA管理対象サーバーへのSOAコンポジットの配置
4. Oracle Identity Manager管理対象サーバーへのSOAコンポジットの登録
5. Oracle Identity Managerでの承認ポリシーの作成
6. カスタム承認プロセスのテスト

ここからは、上記の各ステップについて詳しく確認していきます。

注：上記手順の実行中に何らかの間違ひがあり、サンプルを再実行する必要がある場合、付録Cのクリーンアップおよび再実行方法を参照してください。

6.1. SOA複合アプリケーションの作成

Oracle Identity Managerは、カスタムSOAコンポジットを作成するためのヘルパー・ユーティリティを提供しています。このユーティリティは、必要とされるすべての標準に準拠したSOAプロジェクトのテンプレートを作成します。

ヘルパー・ユーティリティを実行して、カスタムSOAコンポジット向けのJDeveloperアプリケーションを作成します。

1. 環境を設定します（Linuxシステムの場合）。
 - `cd <BEAHOME>/wlserver_10.3/server/bin`
 - `bash`
 - `source setWLSEnv.sh`
2. 次のコマンドを実行して、ユーティリティを実行します。
 - `cd <OIMHOME>/server/workflows/new-workflow`

- `ant -f new_project.xml`

3. 次のプロンプトが表示されたら、JDeveloperのアプリケーション名 (SelfRegistrationApprovalApp) を入力します。

Please enter application name

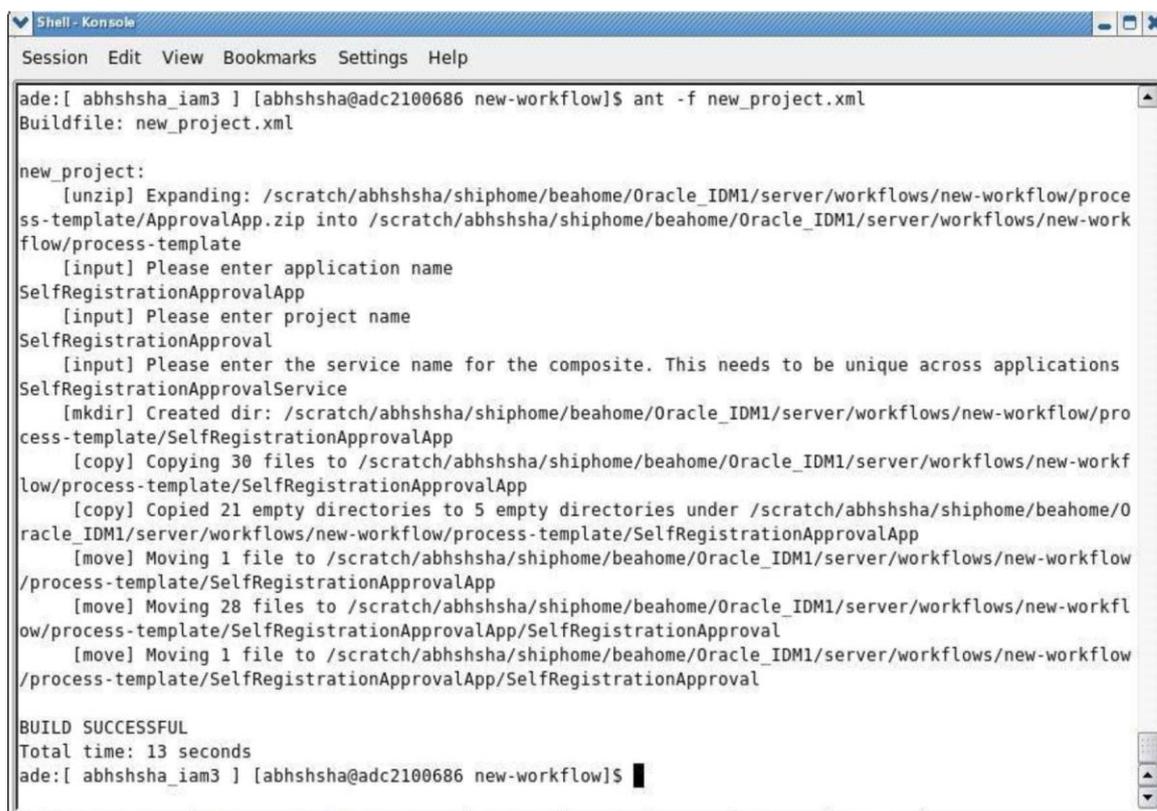
4. 次のプロンプトが表示されたら、JDeveloperのプロジェクト名 (SelfRegistrationApproval) を入力します。

Please enter project name

5. 次のプロンプトが表示されたら、コンポジットのADFバインディング・サービス名 (SelfRegistrationApprovalService) を入力します。

Please enter the service name for the composite. This needs to be unique across applications

次のスクリーンショット (図1) に、SelfRegistrationApprovalAppの作成プロセスを示します。



```
ade:[ abhshsha_iam3 ] [abhshsha@adc2100686 new-workflow]$ ant -f new_project.xml
Buildfile: new_project.xml

new_project:
  [unzip] Expanding: /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template/ApprovalApp.zip into /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template
  [input] Please enter application name
SelfRegistrationApprovalApp
  [input] Please enter project name
SelfRegistrationApproval
  [input] Please enter the service name for the composite. This needs to be unique across applications
SelfRegistrationApprovalService
  [mkdir] Created dir: /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template/SelfRegistrationApprovalApp
  [copy] Copying 30 files to /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template/SelfRegistrationApprovalApp
  [copy] Copied 21 empty directories to 5 empty directories under /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template/SelfRegistrationApprovalApp
  [move] Moving 1 file to /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template/SelfRegistrationApprovalApp
  [move] Moving 28 files to /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template/SelfRegistrationApprovalApp/SelfRegistrationApproval
  [move] Moving 1 file to /scratch/abhshsha/shiphome/beahome/Oracle_IDM1/server/workflows/new-workflow/process-template/SelfRegistrationApprovalApp/SelfRegistrationApproval

BUILD SUCCESSFUL
Total time: 13 seconds
ade:[ abhshsha_iam3 ] [abhshsha@adc2100686 new-workflow]$
```

図1: 新しいSOA複合アプリケーション

新しいアプリケーションは、OIMHOME/server/workflows/new-workflow/process-template/ディレクトリ内に作成されています。次の項では、この新規アプリケーションをJDeveloperで開いて修正します。

6.2. JDeveloperを使用したSOA コンポジットの変更

前項でヘルパー・ユーティリティによって作成されたJDeveloperアプリケーションは、実はSOAアプリケーションです。このアプリケーションに含まれるコンポジットはデフォルトで、ユーザー"xelsysadm"にタスクを割り当てます。ここでは、このSOAコンポジットを変更することで、リクエスト・レベルの承認でユーザーに割り当てられた組織の管理者に対してタスクを割り当てる必要があります。新しく作成したSOAコンポジットを変更するには、次の手順に従います。

1. JDeveloperでアプリケーション
(SelfRegistrationApprovalAppディレクトリ内のSelfRegistrationApprovalApp.jws) を開きます。
2. SelfRegistrationApprovalのSOA Content内で、composite.xmlファイルを開きます。
composite.xml内の次の場所に、bpel.preference.oimurlというプロパティを定義します。

```
<component name="ApprovalProcess">  
<implementation.bpel src="ApprovalProcess.bpel"/>  
<property name="bpel.preference.oimurl">t3://oim_host:oim_port </property>  
</component>
```

このプロパティに格納されるOracle Identity ManagerのURLは、Oracle Identity Managerクライアントを使用してOracle Identity Managerにログインするために、後からコンポジット内のJava Embeddingアクティビティで使用されます。composite.xmlにこのプロパティを追加することで、値をハードコードしなくても、実行時に（Oracle EMコンソールから）変更できるようになります。このサンプルでは、コンポジットのテスト時に実際のoimurl値を設定します。

3. ApprovalProcess.bpelを開き、デザイン・ビューに切り替えます。承認プロセス内の「(x)」をクリックすると、この承認プロセスに含まれる変数が表示されます（図2を参照）。
4. 変数を追加するには、「+」をクリックします（図2を参照）。Nameにoimurlと入力し、Typeで「**Simple Type**」を選択し、Stringに設定します。この変数はステップ2で設定したプロパティからOracle Identity ManagerのURLを読み取る変数であり、Javaコード内で使用できます。
5. orgadminというもう1つの変数を追加し、Typeで「**Simple Type**」を選択し、Stringに設定します。この変数は、組織の管理者データを格納するためにJava Embeddingアクティビティで使用されます。また、この変数の値はOracle Identity ManagerのAPIを使用して取得できます。

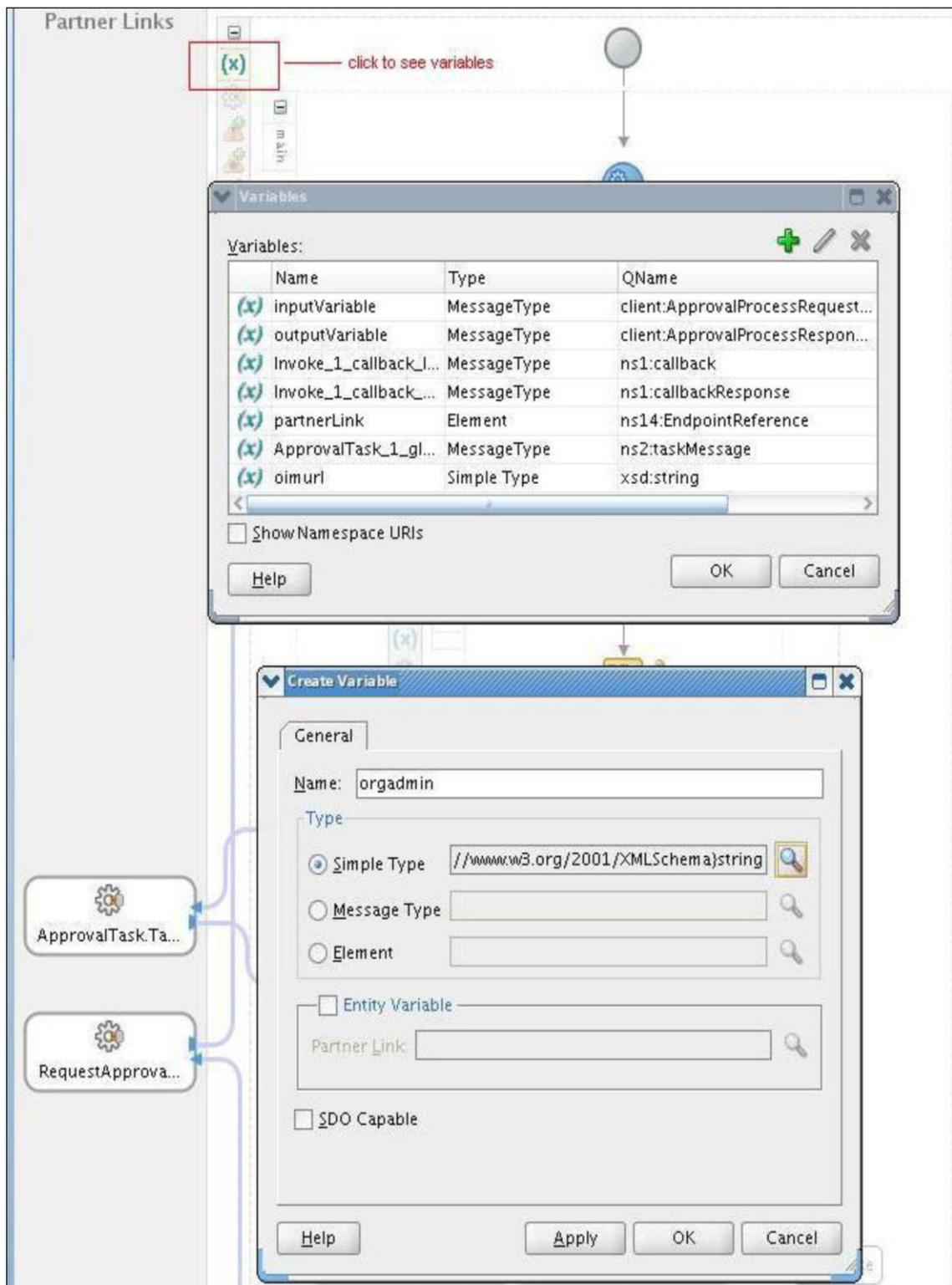


図2：承認プロセスへの変数の追加

- コンポーネント・パレットのBPEL Activitiesから「Assign」アクティビティをドラッグし、receiveInputアクティビティの直下にドロップします（図3を参照）。このアクティビティの名前をAssign_oimurlに変更します。

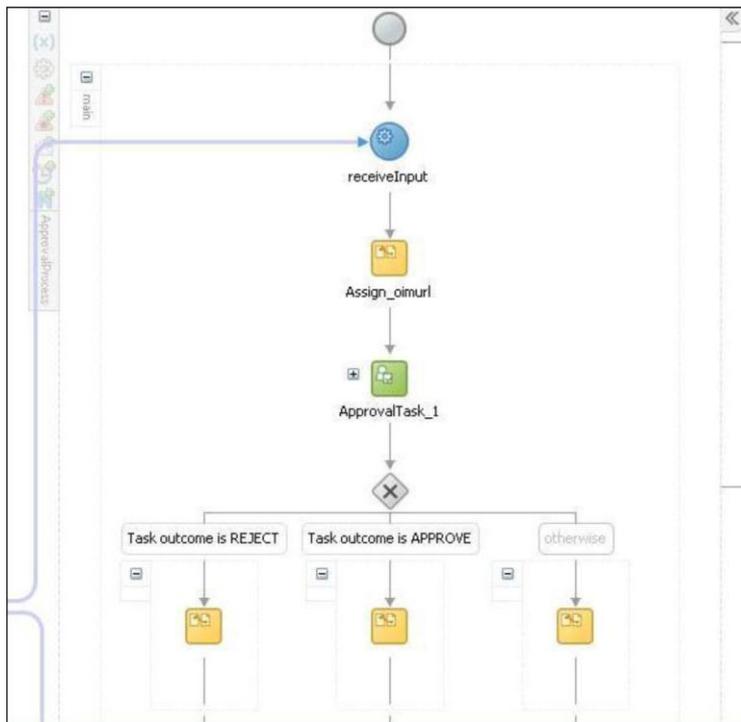


図3 : 承認プロセス内のAssign_oimurlアクティビティ

7. 「Assign_oimurl」アクティビティをダブルクリックします。
「+」記号をクリックし、「Copy Operation」を選択します。Create Copy Operationダイアログが開きます（図4を参照）。

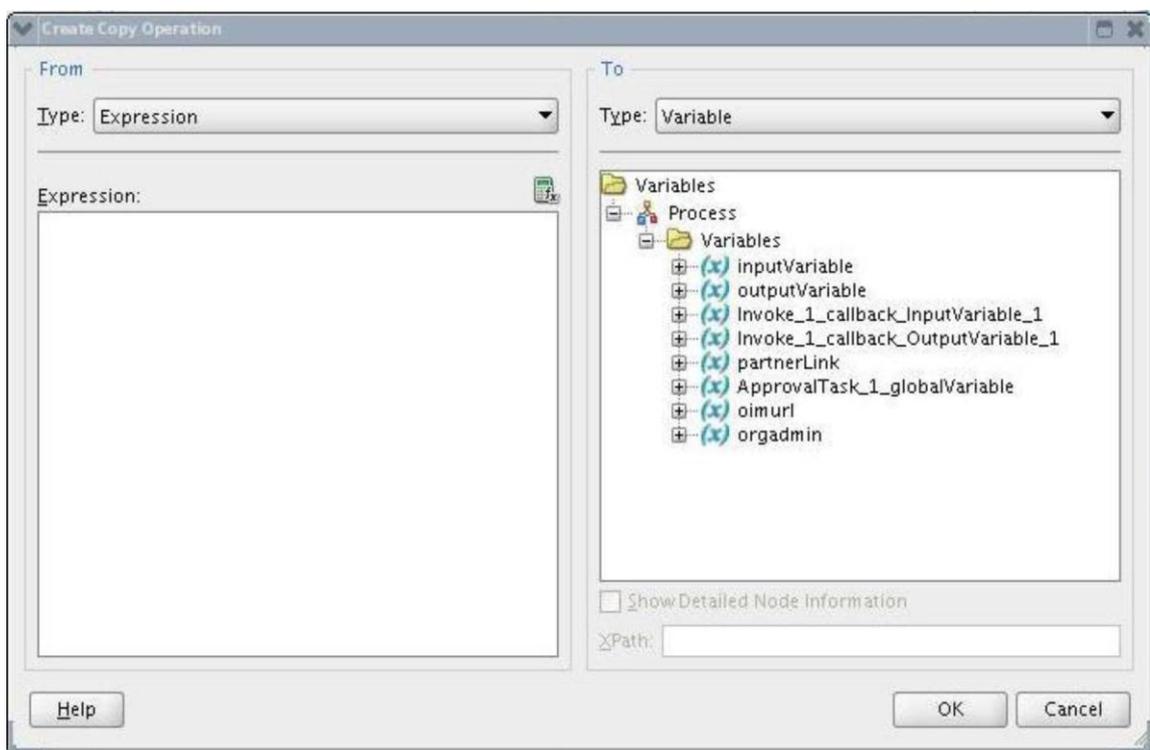


図4 : Create Copy Operationダイアログ

8. FromヘッダーのTypeで「**Expression**」を選択します。「**Expression Builder**」をクリックします。Expression Builderダイアログが開きます（図5を参照）。Functionsヘッダーのドロップダウンから「**BPEL XPath Extension Functions**」を選択します。「**getPreference**」を選択して、「**Insert Into Expression**」をクリックします。この関数の引数として"oimurl"を指定し、「**OK**」をクリックします。これによって、composite.xml内に定義された"bpel.preference.oimurl"プロパティの値が読み取られます。

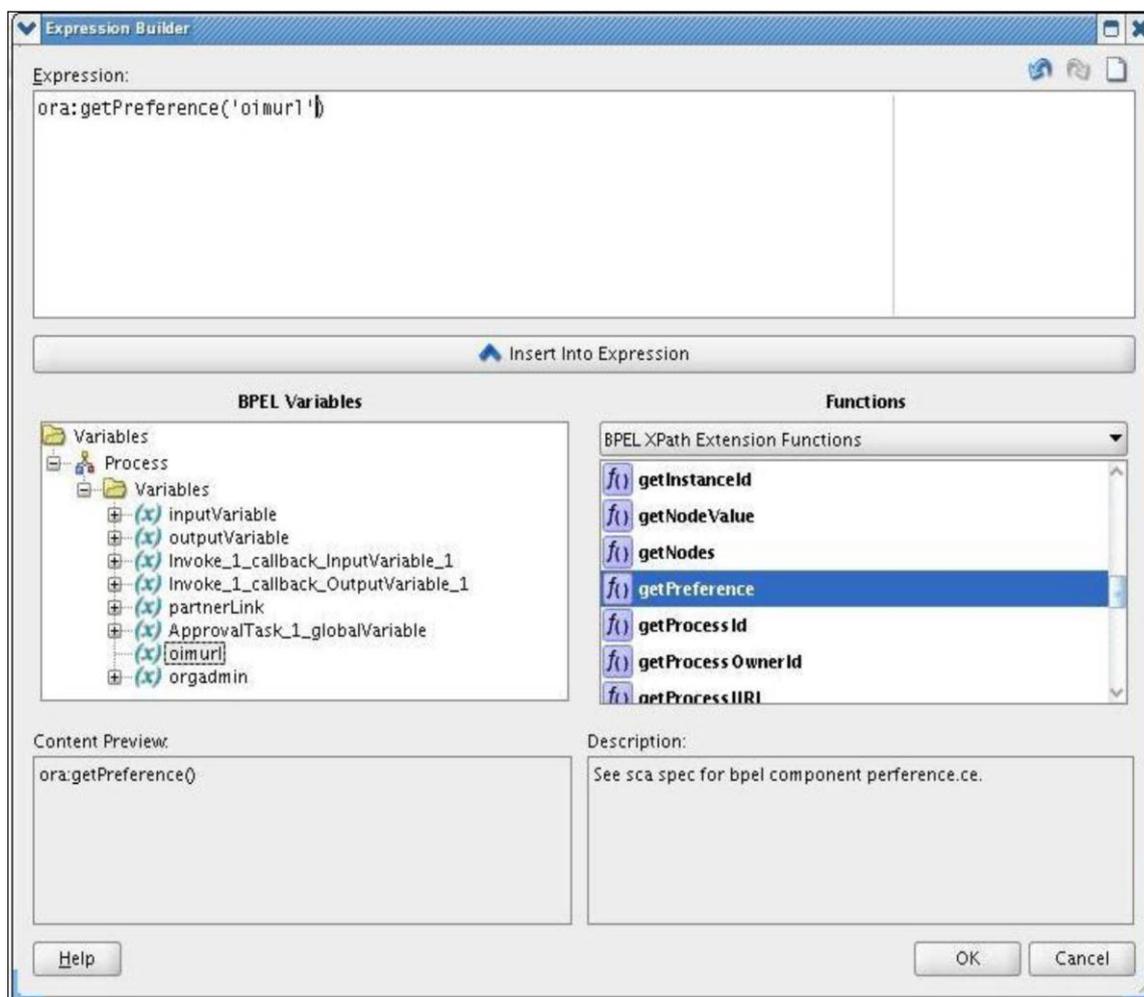


図5 : Expression Builderダイアログ

9. Create Copy Operationダイアログで「**Process**」 → 「**variables**」 → 「**oimurl**」の順に選択し、「**OK**」をクリックします。
10. コンポーネント・パレットのBPEL Activitiesから「**Java Embedding**」アクティビティをドラッグし、Assign_oimurlアクティビティの直下にドロップします（図6を参照）。このアクティビティの名前をGetOrgAdminに変更します。

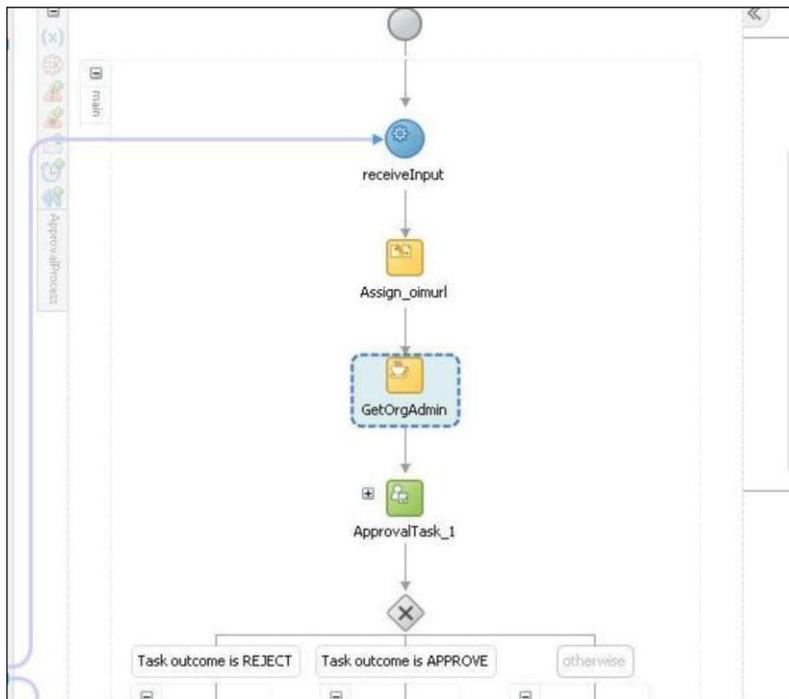


図6：GetOrgAdminアクティビティ

11. 「**GetOrgAdmin**」アクティビティをダブルクリックします。Oracle Identity Manager APIを使用して組織の管理者データを取得するJavaコードをここに記述します（Javaコードとその説明については、付録Aを参照してください）。「**OK**」をクリックします。
12. Oracle Identity Manager APIを使用するには、このコンポジットにoimclient.jarを追加する必要があります。OIMHOME/server/clientからSelfRegistrationApprovalApp/SelfRegistrationApproval/SCA-INF/libへ、oimclient.jarをコピーします。
13. ここでは、Javaコード内で資格証明を取得するため、cwalletも使用します。このため、プロジェクト・ライブラリにjps-manifest.jarを追加する必要があります。「**SelfRegistrationApproval**」プロジェクトを右クリックし、「**Project Properties**」をクリックします。左側のペインから「**Libraries and Classpath**」を選択します。「**Add Jar/Directory**」をクリックし、BEAHOME/oracle_common/modules/oracle.jps_11.1.1/jps-manifest.jarからjps-manifest.jarを追加します。
14. SelfRegistrationApprovalのSOA Content内で、ApprovalTask.taskファイルを開きます。
15. 左側のペインから「**Data**」を選択します。「+」記号をクリックし、「**Add string payload**」を選択します。パラメータ名としてOrganizationAdminを指定します。ステップ11でOracle Identity Manager APIを使用して取得された組織の管理者データが、文字列ペイロードとしてヒューマン・タスクに渡されます。
16. ApprovalProcess.bpelに戻り、「**human task**」ノードを開きます。「**ApprovalTask_1_AssignTaskAttributes**」をダブルクリックします。
17. ペイロードXMLが変数/ns2:initiateTask/task:task/task:payloadにコピーされている行を選択し、クリックして編集します。以下に示すとおり、OrganizationAdmin要素をペイロードに追加します。

```
<payload xmlns="http://xmlns.oracle.com/bpel/workflow/task">
  <RequestID xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
  <RequestModel xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
  <RequestTarget xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
  <url xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
  <RequesterDetails xmlns="http://xmlns.oracle.com/request/RequestDetails"/>
  <BeneficiaryDetails xmlns="http://xmlns.oracle.com/request/RequestDetails"/>
  <ObjectDetails xmlns="http://xmlns.oracle.com/request/RequestDetails"/>
  <OtherDetails xmlns="http://xmlns.oracle.com/request/RequestDetails"/>
  <RequesterDisplayName xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
  <BeneficiaryDisplayName xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
  <Requester xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
  <OrganizationAdmin xmlns="http://xmlns.oracle.com/bpel/workflow/task"/>
</payload>
```

ペイロード内のこのエント리는、ステップ15で定義したOrganizationAdminに相当します。

- 「+」記号をクリックし、「**Copy Operation**」を選択します。Create Copy Operationダイアログが開きます（図7を参照）。Fromヘッダーから「**orgadmin**」変数を選択します。Toヘッダーで、「**initiateTask**」→「**task:task**」→「**task:payload**」の順に選択します。XPathに/ns2:initiateTask/task:task/task:payloadと表示されます。このXPathに対して、/task:OrganizationAdminを追加します。To変数のXPathには、/ns2:initiateTask/task:task/task:payload/task:OrganizationAdminと表示されます。

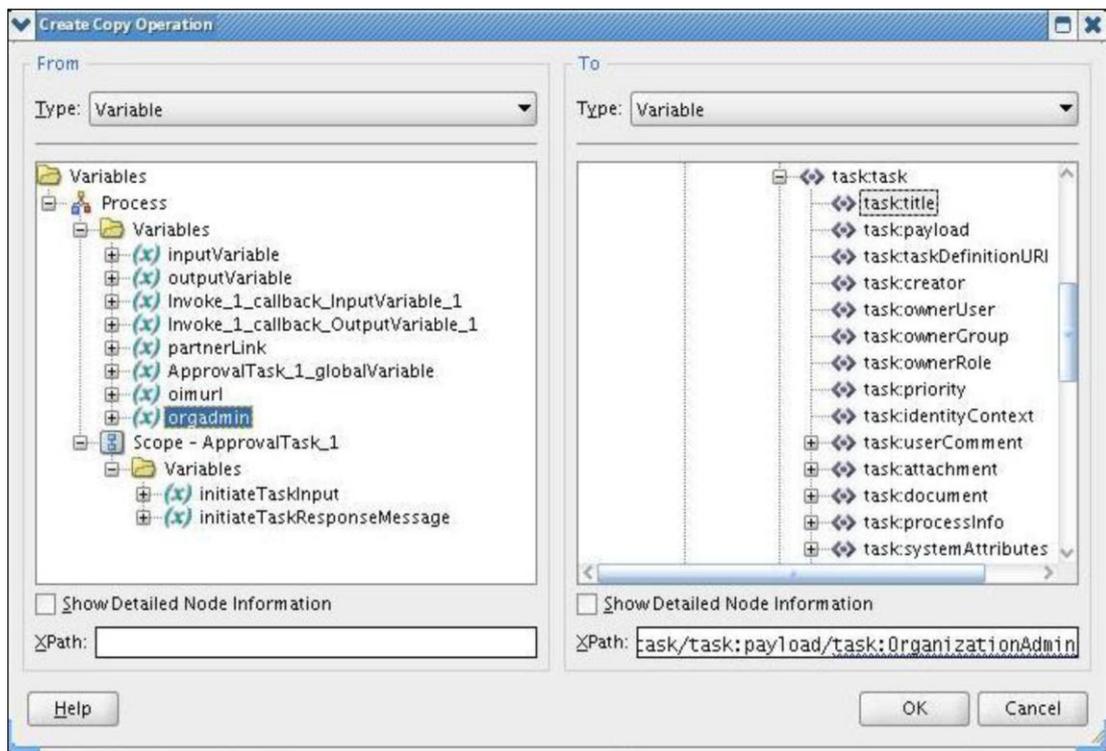


図7 : Create Copy Operationダイアログ

19. ApprovalTask.taskへ移動します。左側のペインから「**Assignment**」をクリックします。
20. 「**Stage1.Participant1**」をダブルクリックします。Edit Participant Typeダイアログが開きます（図8を参照）。Identification TypeにGroupと設定し、Data TypeにBy Expressionと設定します。Value列の「…」 Expression Builderボタンをクリックします。
「**task:task**」 → 「**task:payload**」 → 「**task:OrganizationAdmin**」の順に選択します。

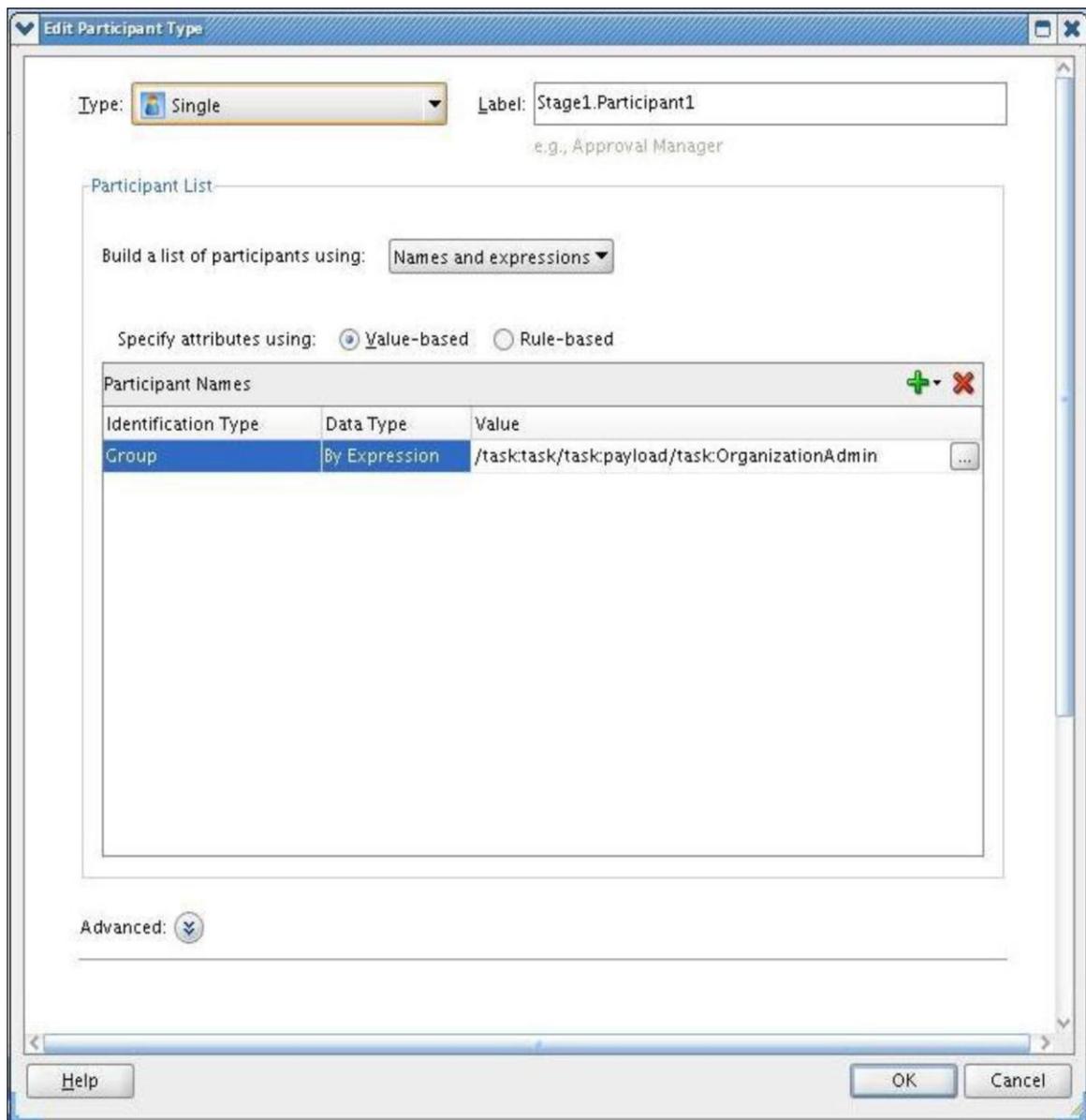


図8 : Edit Participant Typeダイアログ

21. ここまでの作業を保存します。

次の項では、このコンポジットをOracle SOAの管理対象サーバーに配置する方法について説明します。

6.3. Oracle SOA 管理対象サーバーへのSOA コンポジットの配置

SOAコンポジットを配置する前に、Oracle Enterprise Manager Fusion Middleware ControlコンソールのSystem MBean Browserに対して、BpelcClasspathプロパティを設定する必要があります。

1. Oracle Enterprise Manager Fusion Middleware Controlコンソールを開き、WebLogicユーザーとしてログインします。
2. 左側のペインで「**Weblogic Domain**」を開きます。「<WLS_DOMAIN>」を右クリックして、「**System MBean Browser**」を選択します。
3. 「 **Application Defined MBeans** 」 → 「 **oracle.as.soainfra.config** 」 → 「**Server:<SOA_SERVER>**」 → 「**BPELConfig**」 → 「**bpel**」の順に選択します。
4. Attributes列で、「**BpelcClasspath**」をクリックします。oimclient.jarとjps-manifest.jarのフル・パスを入力します。これらのファイルはshiphome内の次の場所にあります。

```
<OIMHOME>/server/client/oimclient.jar
```

```
<BEAHOME>/oracle_common/modules/oracle.jps_11.1.1/jps-manifest.jar
```

注：相当するフル・パスでOIMHOMEとBEAHOMEを置換してください。UNIXの場合はパスをコロン (:) で区切り、Windowsの場合はセミコロン (;) で区切る必要があります。

次の手順に従って、SOAコンポジットをOracle SOAサーバーに配置します。

1. 「**SelfRegistrationApproval**」プロジェクトを右クリックし、「**Deploy**」 → 「**SelfRegistrationApproval**」をクリックします。Oracle SOAサーバーへのコンポジットの配置プロセスを支援するポップアップ・ウィザードが表示されます。
2. Deployment Actionステップで、「**Deploy to Application Server**」を選択します。「**Next**」をクリックします。
3. Deployment Configurationステップで、「**Overwrite any existing composites with the same revision ID**」オプションを選択します（図9を参照）。「**Next**」をクリックします。

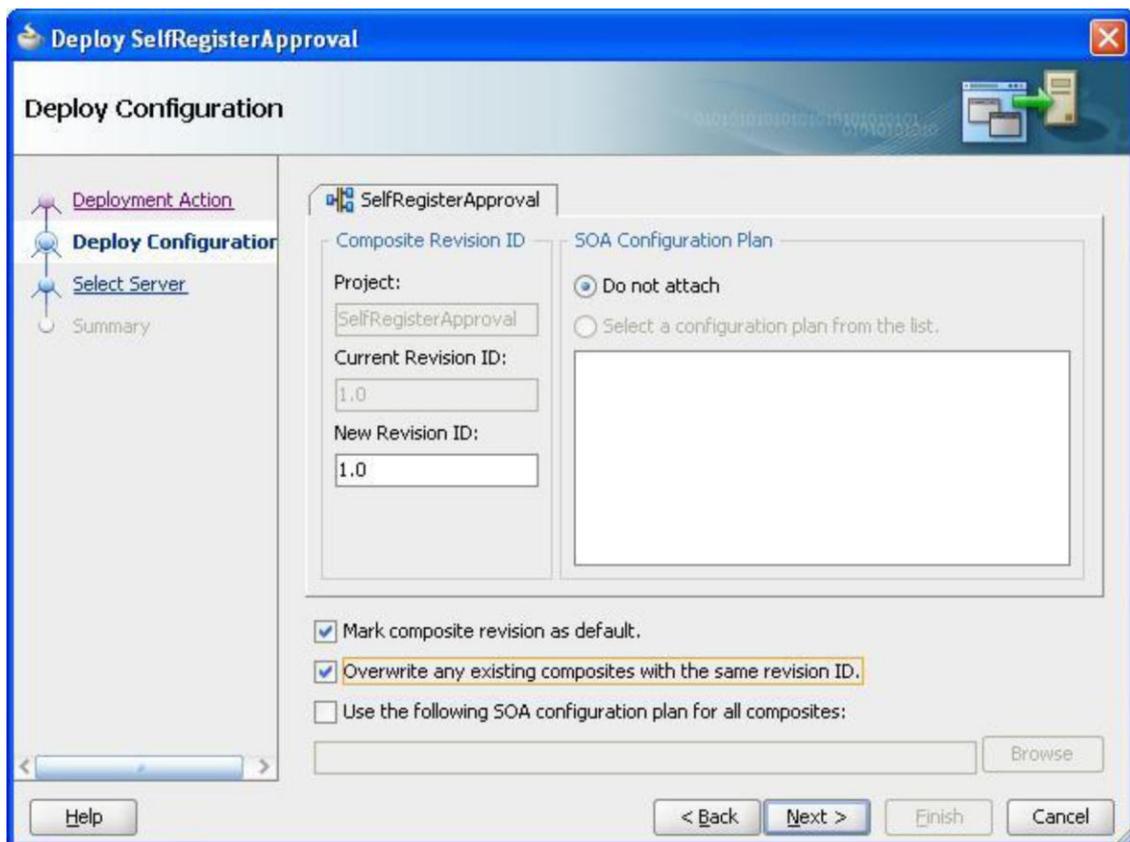


図9 : Deploy SelfRegistrationApprovalウィザード

4. Select Serverステップで、使用するアプリケーション・サーバーへの接続を選択します。接続が存在しない場合は、新しい接続を作成します。「+」記号をクリックすると、「Create Application Server Connection」ポップアップが表示されます。アプリケーション・サーバーへの接続があらかじめ作成されている場合はステップ10へ、それ以外の場合はステップ6へ進みます。
5. "Create Application Server Connection"ポップアップで接続名を入力し、「Next」をクリックします。
6. Authenticationステップで、WebLogicユーザーのパスワードを入力します。「Next」をクリックします。
7. Configurationステップで、管理サーバーのホスト名およびポートと、WebLogicドメインを入力します（図10を参照）。「Next」をクリックします。

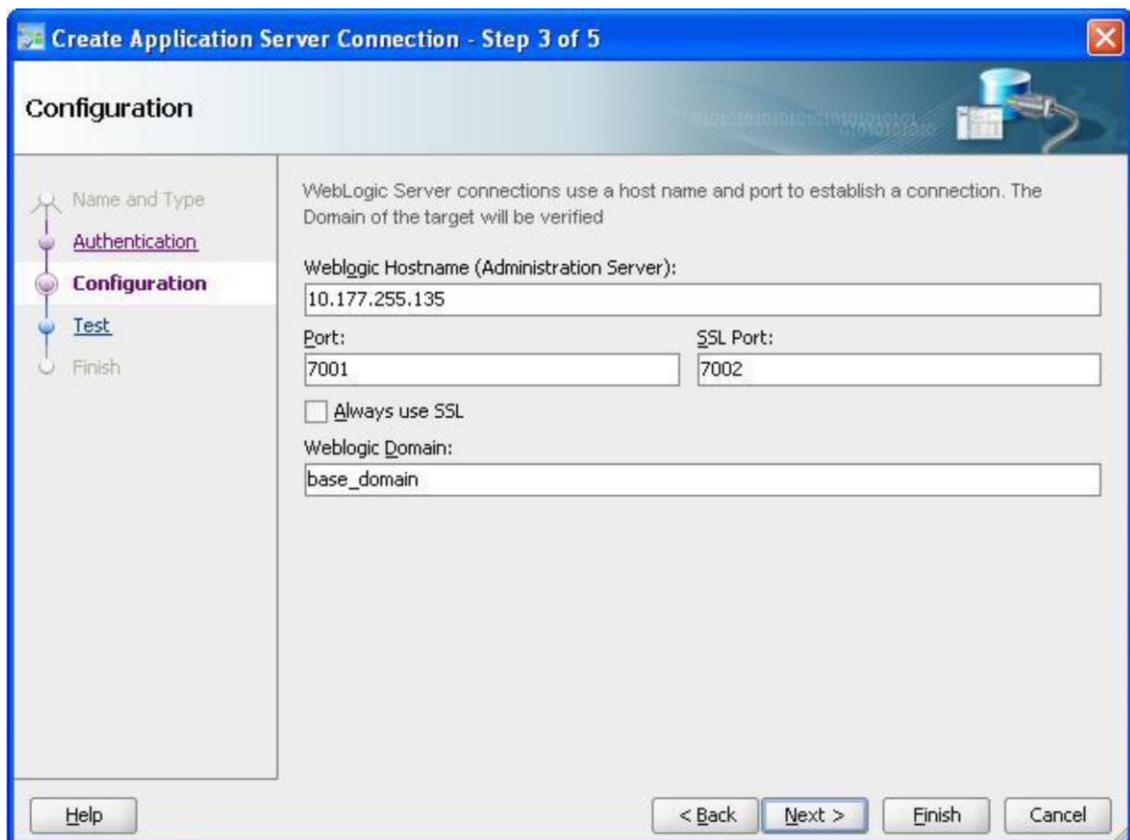


図10 : Create Application Server Connectionポップアップ

8. 「**Test connection**」をクリックします。すべてのテストに成功したら、「**Finish**」をクリックします。
9. 「**Next**」をクリックします。
10. Select SOA serverステップで、SOAサーバーが選択されていることを確認します。「**Finish**」をクリックします。
11. JDeveloperのコンパイラ・ログとデプロイメント・ログを参照して、エラーが発生しているかどうかを確認します。

次の項では、このコンポジットをOracle Identity Managerの管理対象サーバーに登録する方法について説明します。

6.4. Oracle Identity Manager 管理対象サーバーへのSOA コンポージットの登録

SOAコンポージットは、承認プロセスで使用する前に、Oracle Identity Managerに登録する必要があります。SOAコンポージットをOracle Identity Managerに登録するには、次の手順を実行します。

1. 環境設定が済んでいない場合、ここで環境を設定します（Linuxシステムの場合）。
 - `cd <BEAHOME>/wlserver_10.3/server/bin`
 - `bash`
 - `source setWLSEnv.sh`
2. `OIM_HOME/server/workflows/registration/`ディレクトリに`SelfRegistrationApproval.props`プロパティ・ファイルを作成し、次の内容を設定します。

```
name=SelfRegistrationApproval  
  
category=Approval  
  
providerType=BPEL  
  
serviceName=RequestApprovalService  
  
domainName=default  
  
version=1.0  
  
payloadID=payload  
  
operationID=process  
  
listOfTasks=ApprovalTask
```

注：プロパティ・ファイル内に余分な空白が含まれないように注意してください。

3. `OIM_HOME/server/workflows/new-workflow/`ディレクトリから、次のコマンドを実行します。
 - `ant -f registerworkflows-mp.xml register`
4. 次のプロンプトが表示されたら、Oracle Identity Managerの管理者ユーザー名を入力します。
Enter the username
5. 次のプロンプトが表示されたら、Oracle Identity Managerの管理者パスワードを入力します。

Enter the password

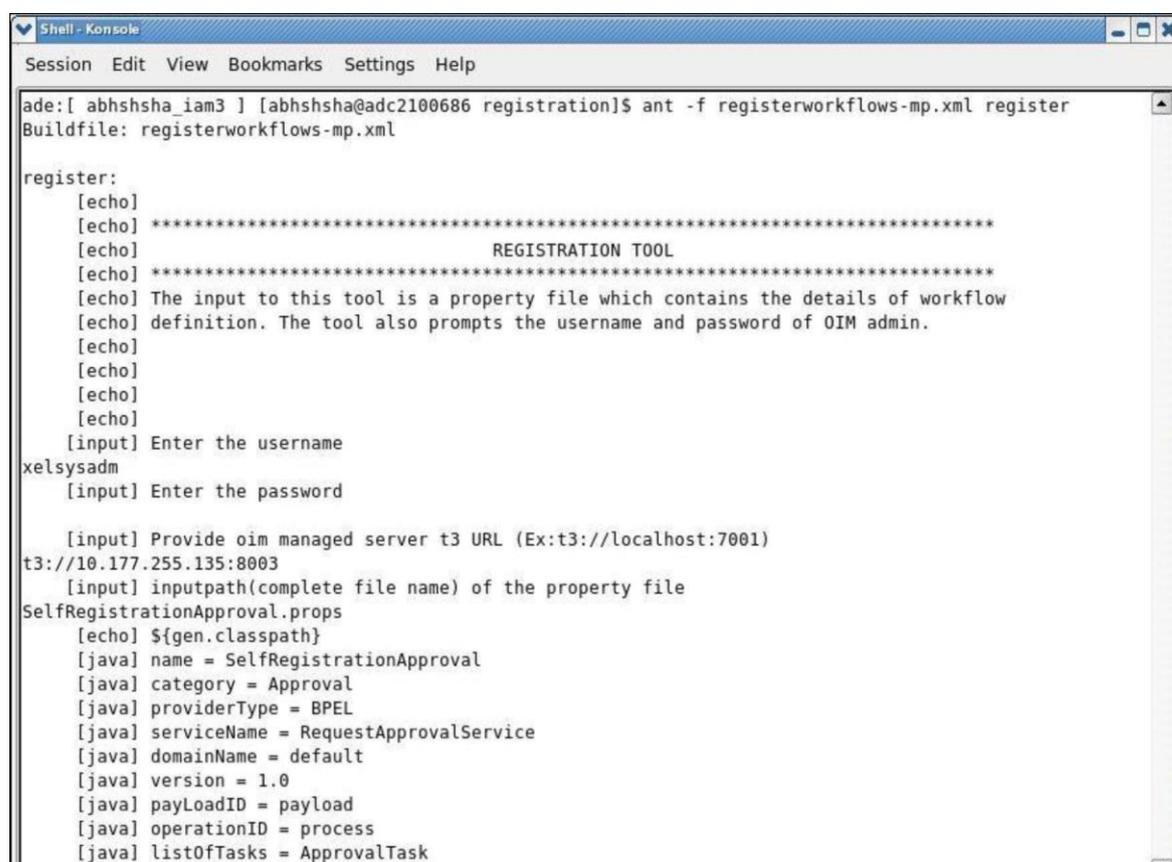
6. 次のプロンプトが表示されたら、Oracle Identity Manager管理対象サーバーt3のURL
(例：t3://10.177.255.135:8003) を入力します。

Provide oim server t3 URL

7. 次のプロンプトが表示されたら、ステップ1で作成したプロパティ・ファイルのパス（絶対パスまたは相対パス）を入力します。

Input path (complete file name) of the property file

次のスクリーンショット（図11）に、SOAコンポジットSelfRegistrationApprovalの登録プロセスを示します。



```
Shell - Konsole
Session Edit View Bookmarks Settings Help
ade:[ abhshsha_iam3 ] [abhshsha@adc2100686 registration]$ ant -f registerworkflows-mp.xml register
Buildfile: registerworkflows-mp.xml

register:
[echo]
[echo] *****
[echo]                                REGISTRATION TOOL
[echo] *****
[echo] The input to this tool is a property file which contains the details of workflow
[echo] definition. The tool also prompts the username and password of OIM admin.
[echo]
[echo]
[echo]
[input] Enter the username
xelsysadm
[input] Enter the password

[input] Provide oim managed server t3 URL (Ex:t3://localhost:7001)
t3://10.177.255.135:8003
[input] inputpath(complete file name) of the property file
SelfRegistrationApproval.props
[echo] ${gen.classpath}
[java] name = SelfRegistrationApproval
[java] category = Approval
[java] providerType = BPEL
[java] serviceName = RequestApprovalService
[java] domainName = default
[java] version = 1.0
[java] payloadID = payload
[java] operationID = process
[java] listOfTasks = ApprovalTask
```

図11：カスタムSOAコンポジットの登録

登録が正しく完了したら、次の項に進みます。次の項では、このコンポジットを呼び出す承認ポリシーの作成方法について説明します。

6.5. Oracle Identity Manager での承認ポリシーの作成

このサンプルでは、2つの承認ポリシーが必要になります。1つは、リクエスト・レベルのユーザー自動登録タスクをSYSTEM ADMINISTRATORSロールに割り当てるための承認ポリシーであり、もう1つは、運用レベルのタスクを組織の管理者に割り当てるための承認ポリシーです。

次に、リクエスト・レベルの承認向けの承認ポリシーを作成する手順を示します。

1. Shirley SchmidtとしてOracle Identity Managerにログインします。
2. Advanced Adminコンソールを開きます。
3. 「Policies」タブをクリックします。
4. 左側のペインで「Create」ボタンをクリックします。
5. ポリシー名（例：SelfRegisterPolicyRL）を入力します。
6. Request typeで「Self-Register User」を選択します。
7. ドロップダウンから、「Request Level of approval」を選択します。
8. Approval Processで「default/DefaultRoleApproval!1.0」を選択します。「Next」をクリックします。
9. ルール名（例：SelfRegistrationRuleRL）を入力します。
10. 「Add Simple Rule」をクリックします。ポップアップが表示されます（図12を参照）。
11. ルールに次の値を指定します（図12を参照）。
 - Entity → Request
 - Attribute → Request Type
 - Condition → Equals
 - Value → Self-Register User
 - Parent Rule Container → Approval Rule

Add Simple Rule

* Indicates required fields.

* Entity Request

* Attribute Request Type

* Condition Equals

* Value Self-Register User

* Parent Rule Container Approval Rule

Save Cancel

図12 : Add Simple Ruleポップアップ

12. 「**Save**」をクリックします。
13. メイン・ページで「**Next**」をクリックします。
14. 「**Finish**」をクリックします。承認ポリシーの作成が成功したことを示すメッセージが表示されます。

次に、リクエスト・レベルの承認向けの承認ポリシーを作成する手順を示します。

1. 左側のペインで「**Create**」ボタンをクリックします。
2. ポリシー名（例：SelfRegisterPolicyOL）を入力します。
3. Request typeで「**Self-Register User**」を選択します。
4. ドロップダウンから、「**Operation Level of approval**」を選択します。
5. 「**All Scope**」を選択します。
6. Approval Processで、新しく作成したコンポジット（default/SelfRegistrationApproval!1.0）を選択します。「**Next**」をクリックします。
7. ルール名（例：SelfRegistrationRuleOL）を入力します。
8. 「**Add Simple Rule**」をクリックします。ポップアップが表示されます（図12を参照）。
9. ルールに次の値を指定します（図12を参照）。
 - Entity → Request
 - Attribute → Request Type

- Condition → Equals
- Value → Self-Register User
- Parent Rule Container → Approval Rule

10. 「**Save**」をクリックします。
11. メイン・ページで「**Next**」をクリックします。
12. 「**Finish**」をクリックします。承認ポリシーの作成が成功したことを示すメッセージが表示されます。
13. Oracle Identity Managerをログアウトします。

次の項では、カスタム承認プロセスのテスト方法について説明します。

6.6. カスタム承認プロセスのテスト

SOAコンポジットを呼び出す前に、SOAコンポジットのcomposite.xmlに含まれるプロパティ・セットに有効なOracle Identity ManagerのURLを設定する必要があります。この値は、Oracle EMコンソールから設定できます。以下の手順を実行します。

1. Oracle EMコンソールを開き、WebLogicユーザーとしてログインします。
2. 左側のペインで「**Weblogic Domain**」を開きます。「<WLS_DOMAIN>」を右クリックして、「**System MBean Browser**」を選択します。
3. 「**Application Defined MBeans**」 → 「**oracle.soa.config**」 → 「**Server:<SOA_SERVER>**」 → 「**SCAComposite**」 → 「**SelfRegistrationApproval[1.0]**」 → 「**SCAComposite.SCAComponent**」 → 「**ApprovalProcess**」の順に選択します。
4. Attributes列で、「**Properties**」 → 「**Element_0**」をクリックします。"value"キーに対して、有効なOracle Identity ManagerのURLを入力します（例：t3://10.177.255.135）。

承認プロセスをテストするには、次の手順を実行します。

1. Oracle Identity Managerのログイン・ページを開きます。
2. 「**Register**」リンクをクリックします。
3. Basic Informationステップでユーザーの詳細情報を入力します。「**Next**」をクリックします。
4. ユーザー確認用の質問を入力します。「**Register**」をクリックします。
5. 登録追跡リクエスト番号を書き留めておきます。「**Back to login**」をクリックします。
6. Shirley Schmidtとしてログインします。
7. 「**Search Approval Tasks**」をクリックします。同じリクエストIDを持つ自動登録タスクが表示されることを確認します。

8. 登録追跡リクエスト番号と同じリクエスト番号を持つタスク行を選択し、「**Open task details**」をクリックします。
9. **Organization**に"FINANCE"と入力します。
10. 「**Approve**」をクリックします。タスクが運用レベルの承認へと移行します。
11. Oracle Identity Managerをログアウトします。
12. Danny Craneとしてログインします（このユーザーはFINANCE_APPROVERSグループのメンバーであるため、同じタスクが表示されるはずです）。
13. 「**Search Approval Tasks**」をクリックします。同じリクエストIDを持つ自動登録タスクが表示されることを確認します。
14. 「**Approve**」をクリックします。タスクが無事承認され、Oracle Identity Managerにユーザーが作成されました。
15. Oracle Identity Managerをログアウトします。
16. 「**Track Registration**」をクリックします。追跡IDを入力し、「**Submit**」をクリックします。登録リクエスト・ステータスがcompletedになっていることを確認します。

付録A

Java コード

```
try {

    System.out.println("Prototype for invoking an OIM API from a SOA Composite");
    System.out.println("RTM Usecase: Self Registration Approval by Organization
                        Administrator");
    String oimUserName = "";
    String oimPassword = "";
    String oimURL = "";

    //get system administrator's credentials
    oracle.security.jps.JpsContext ctx =
        oracle.security.jps.JpsContextFactory.getContextFactory().getContext();

    final oracle.security.jps.service.credstore.CredentialStore cs =
        (oracle.security.jps.service.credstore.CredentialStore)ctx.getServiceInstance(
            oracle.security.jps.service.credstore.CredentialStore.class);

    oracle.security.jps.service.credstore.CredentialMap cmap =
        cs.getCredentialMap("oracle.oim.sysadminMap");

    oracle.security.jps.service.credstore.Credential cred = cmap.getCredential("sysadmin");

    if (cred instanceof oracle.security.jps.service.credstore.PasswordCredential) {

        oracle.security.jps.service.credstore.PasswordCredential pcred =
            (oracle.security.jps.service.credstore.PasswordCredential)cred;

        char[] p = pcred.getPassword();
        oimUserName = pcred.getName();
        oimPassword = new String(p);
    }

    //get oimurl
    Object obj = getVariableData("oimurl");
    oimURL = obj.toString();
    System.out.println("oimurl=" + oimURL);

    // set the initial context factory
```

```

String oimInitialContextFactory = "weblogic.jndi.WLInitialContextFactory";

// set up the environment for making the OIM API invocation
java.util.Hashtable env = new java.util.Hashtable();
env.put(oracle.iam.platform.OIMClient.JAVA_NAMING_FACTORY_INITIAL,
        oimInitialContextFactory);
env.put(oracle.iam.platform.OIMClient.JAVA_NAMING_PROVIDER_URL, oimURL);

oracle.iam.platform.OIMClient client = new oracle.iam.platform.OIMClient(env);
client.login(oimUserName, oimPassword.toCharArray());
System.out.println("Login Successful");

// get the RequestService to get details of the request
oracle.iam.request.api.RequestService reqSvc =
        (oracle.iam.request.api.RequestService)client.getService(
                oracle.iam.request.api.RequestService.class);

Object reqIdXMLElem = getVariableData("inputVariable",
        "payload",
        "/ns3:process/ns4:RequestID");

String reqId = ((oracle.xml.parser.v2.XMLElement)reqIdXMLElem).getText();
System.out.println("The request ID is "+reqId);

// invoke the getBasicRequestData() method on the RequestService API
oracle.iam.request.vo.Request req = reqSvc.getBasicRequestData(reqId);
String act_key = "";
java.util.List< oracle.iam.request.vo.RequestEntity> targetEntities =
        req.getTargetEntities();

for( oracle.iam.request.vo.RequestEntity reqEntity: targetEntities){
    java.util.List< oracle.iam.request.vo.RequestEntityAttribute> attributes =
        reqEntity.getEntityData();
    for( oracle.iam.request.vo.RequestEntityAttribute attribute : attributes){
        if(attribute.getName().equalsIgnoreCase("Organization")){
            act_key = attribute.getValue().toString();
        }
    }
}

System.out.println("Organization Key is "+act_key);

```

```

if(act_key != "" && act_key != " ") {

    Thor.API.Operations.tcOrganizationOperationsIntf orgAPI =
        (Thor.API.Operations.tcOrganizationOperationsIntf)client.getService(
            Thor.API.Operations.tcOrganizationOperationsIntf.class);

    Thor.API.tcResultSet rset= orgAPI.getAdministrators(Long.parseLong(act_key));

    StringBuffer sb = new StringBuffer();

    for (int i = 0; i < rset.getRowCount();i++){
        rset.goToRow(i);
        sb.append(rset.getStringValue("Groups.Group Name"));
        if(i >= 0 && i < (rset.getRowCount()-1)){
            sb.append(",");
        }
    }
    String grpNames = sb.toString();
    System.out.println("Groups="+grpNames);
    setVariableData("orgadmin",grpNames);
}
else{
    setVariableData("orgadmin","SYSTEM ADMINISTRATORS");
}
Object obj1 = getVariableData("orgadmin");
System.out.println("OrganizationAdmins = " + obj1.toString());
} catch (Exception e){
    System.out.println("-----");
    e.printStackTrace();
    System.out.println("-----");
}
}

```

ブロック・レベルの説明

Oracle Identity Manager APIを使用して組織の管理者データを取得するには、Javaコードで次のステップを実行します。

1. システム管理者の資格証明を取得します。
2. システム管理者としてログインします。
3. Oracle Identity Manager APIを使用して、組織の管理者をデータ取得します。

ここからは、それぞれのステップについて説明します。

1. システム管理者の資格証明を取得します。

システム管理者の資格証明は資格証明ストア (*credential*) に保存されています。はじめに資格証明ストアを取得してから、資格証明マップを取得し、キーを使用して資格証明を取得します。以下に、そのコードを示します。

```
//get Credential store

oracle.security.jps.JpsContext ctx =
    oracle.security.jps.JpsContextFactory.getContextFactory().getContext();

final oracle.security.jps.service.credstore.CredentialStore cs =
    (oracle.security.jps.service.credstore.CredentialStore)ctx.getServiceInstance(
        oracle.security.jps.service.credstore.CredentialStore.class);

//get Credential
oracle.security.jps.service.credstore.CredentialMap cmap =
    cs.getCredentialMap("oracle.oim.sysadminMap");

oracle.security.jps.service.credstore.Credential cred = cmap.getCredential("sysadmin");
```

2. システム管理者としてログインします。

はじめに環境を設定してから、システム管理者としてOracle Identity Managerにログインします。以下に、そのコードを示します。

```
//setup the environment
String oimInitialContextFactory = "weblogic.jndi.WLInitialContextFactory";
java.util.Hashtable env = new java.util.Hashtable();

env.put(oracle.iam.platform.OIMClient.JAVA_NAMING_FACTORY_INITIAL,
    oimInitialContextFactory);
env.put(oracle.iam.platform.OIMClient.JAVA_NAMING_PROVIDER_URL, oimURL);

//login to OIM
oracle.iam.platform.OIMClient client = new oracle.iam.platform.OIMClient(env);
client.login(oimUserName, oimPassword.toCharArray());
System.out.println("Login Successful");
```

3. Oracle Identity Manager APIを使用して、組織の管理者データを取得します。

リクエスト・オブジェクトの取得には、*oracle.iam.request.api.RequestService* APIが使用されています。また、組織の管理者データを取得するために、*Thor.API.Operations.tcOrganizationOperationsIntf* APIが使用されています。

付録B

CSFへの資格証明の保存

CSFに資格証明を保存するには、次の手順を実行します。

1. Oracle EMコンソールを開き、WebLogicとしてログインします。
2. 左側のペインで「Weblogic Domain」を開きます。
3. 「<WLS_DOMAIN>」を右クリックします。「Security」→「Credentials」を選択します。
4. 「Create Map」ボタンをクリックします。マップ名として"oracle.oim.sysadminMap"を入力します。「OK」をクリックします。
5. 「Create Key」ボタンをクリックします（図13を参照）。以下の情報を設定します。
 - Select Map : oracle.oim.sysadminMap
 - Key : sysadmin
 - Type : Password
 - Username : <Shirley SchmidtのログインID>
 - Password : <Shirley Schmidtのパスワード>「OK」をクリックします。

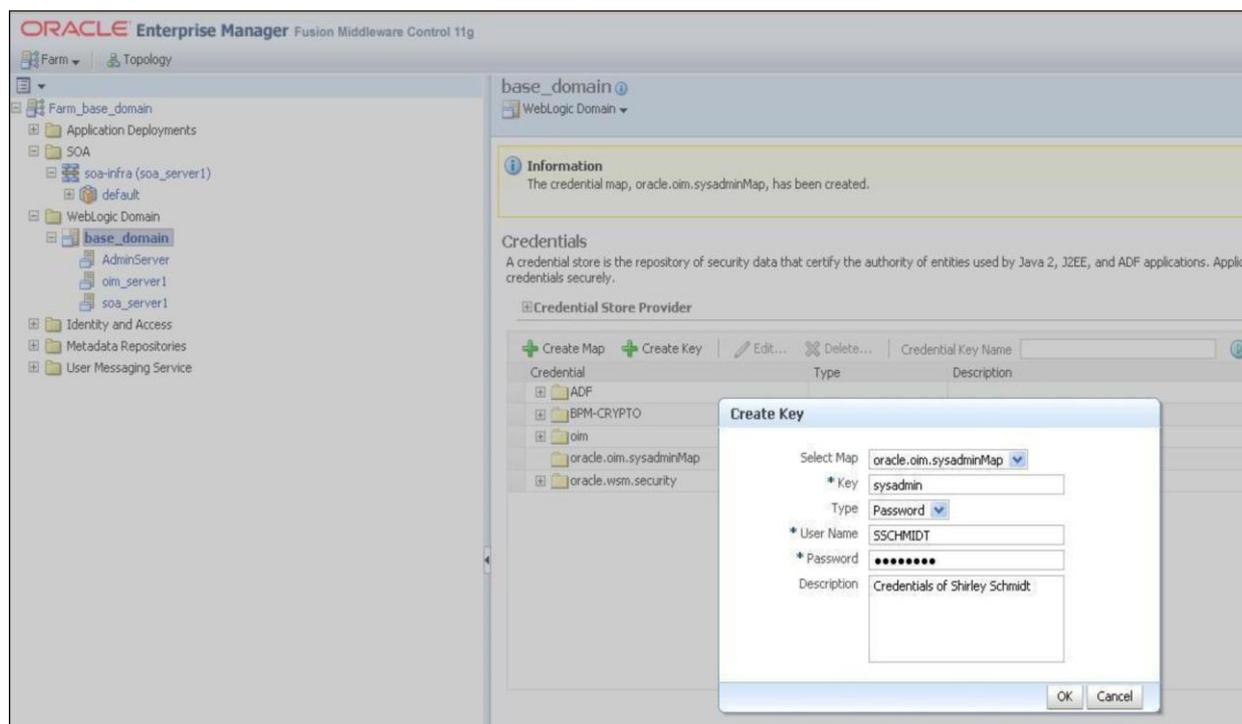


図13 : Oracle Identity Managerシステム管理者の資格証明

付録C :

クリーンアップ

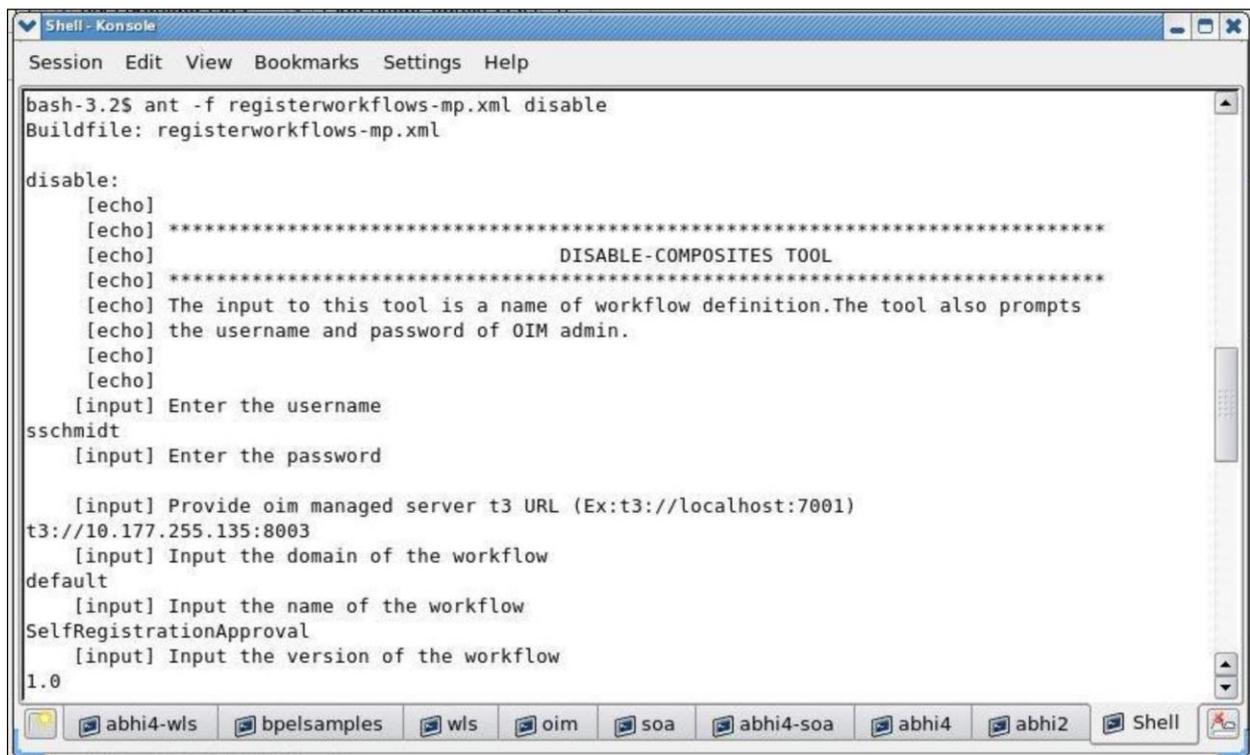
Advanced Adminコンソールから次の承認ポリシーを削除します（ポリシーが存在する場合のみ）。

1. SelfRegisterPolicyRL
2. SelfRegisterPolicyOL

承認プロセスを登録している場合（第6.4項）、これを無効化する必要があります。Oracle Identity Managerで承認プロセスを無効化するには、次の手順を実行します。

1. 環境設定が済んでいない場合、ここで環境を設定します（Linuxシステムの場合）。
 - `cd <BEAHOME>/wlserver_10.3/server/bin`
 - `bash`
 - `source setWLSEnv.sh`
2. <OIMHOME>/server/workflows/registration/ディレクトリから、次のコマンドを実行します。
 - `ant -f registerworkflows-mp.xml disable`
3. 次のプロンプトが表示されたら、Oracle Identity Managerの管理者ユーザー名を入力します。
Enter the username
4. 次のプロンプトが表示されたら、Oracle Identity Managerの管理者パスワードを入力します。
Enter the password
5. 次のプロンプトが表示されたら、Oracle Identity Manager管理対象サーバーt3のURL（例：t3://10.177.255.135:8003）を入力します。
Provide oim managed server t3 URL
6. 次のプロンプトが表示されたら、ワークフローのドメインを入力します。
Input the domain of the workflow
7. 次のプロンプトが表示されたら、ワークフローの名前（SelfRegistrationApproval）を入力します。
Input the name of the workflow
8. 次のプロンプトが表示されたら、ワークフローのバージョン（1.0）を入力します。
Input the version of the workflow

次のスクリーンショット（図14）に、SOAコンポジットSelfRegistrationApprovalの無効化プロセスを示します。



```
bash-3.2$ ant -f registerworkflows-mp.xml disable
Buildfile: registerworkflows-mp.xml

disable:
  [echo]
  [echo] *****
  [echo]                                     DISABLE-COMPOSITES TOOL
  [echo] *****
  [echo] The input to this tool is a name of workflow definition.The tool also prompts
  [echo] the username and password of OIM admin.
  [echo]
  [input] Enter the username
sschmidt
  [input] Enter the password

  [input] Provide oim managed server t3 URL (Ex:t3://localhost:7001)
t3://10.177.255.135:8003
  [input] Input the domain of the workflow
default
  [input] Input the name of the workflow
SelfRegistrationApproval
  [input] Input the version of the workflow
1.0
```

図14 : SelfRegistrationApprovalの無効化

Oracle SOAサーバーからコンポジットをアンデプロイするには、次の手順に従います。

1. 環境設定が済んでいない場合、ここで環境を設定します (Linuxシステムの場合)。
 - cd <BEAHOME>/wlserver_10.3/server/bin
 - bash
 - source setWLSEnv.sh
2. <SOAHOME>/bin/ディレクトリから、次のコマンドを実行します。
 - ant -f ant-sca-deploy.xml undeploy -DserverURL=server.url -DcompositeName=compoite.name -Drevision=revision.id -Duser=user -Dpassword=password -Dpartition=partition.name

それぞれの変数に指定する内容は次のとおりです。

- serverURL - soa-infraアプリケーションをホストするサーバーのURL (例 : http://10.177.255.135:8001)
- compositeName - コンポジット名 (SelfRegistrationApproval)
- revision - コンポジットのリビジョンID (1.0)
- user/password - WebLogicのユーザーとパスワード
- partition - コンポジットが配置されているパーティションの名前 (省略可。デフォルト値は"default")

サンプルの再実行

事前に、前述のクリーンアップを必ず実行してください。その後、第6章で説明した手順を実行します。

注：このサンプルを以前に実行した際に承認プロセスを登録していた場合、第6.4項のステップを実行しようとする、次の例外が発生する可能性があります。

```
oracle.iam.platform.workflowservice.exception.IAMWorkflowException:The workflow definition with name default/SelfRegistrationApproval!1.0 already exists in OIM.
```

このような場合、次の代替手順を実行して承認プロセスを有効化します。

1. 環境設定が済んでいない場合、ここで環境を設定します（Linuxシステムの場合）。
 - `cd <BEAHOME>/wlserver_10.3/server/bin`
 - `bash`
 - `source setWLSEnv.sh`
2. <OIMHOME>/server/workflows/registration/ディレクトリから、次のコマンドを実行します。
 - `ant -f registerworkflows-mp.xml enable`
3. 次のプロンプトが表示されたら、Oracle Identity Managerの管理者ユーザー名を入力します。

Enter the username
4. 次のプロンプトが表示されたら、Oracle Identity Managerの管理者パスワードを入力します。

Enter the password
5. 次のプロンプトが表示されたら、Oracle Identity Manager管理対象サーバーt3のURL（例：t3://10.177.255.135:8003）を入力します。

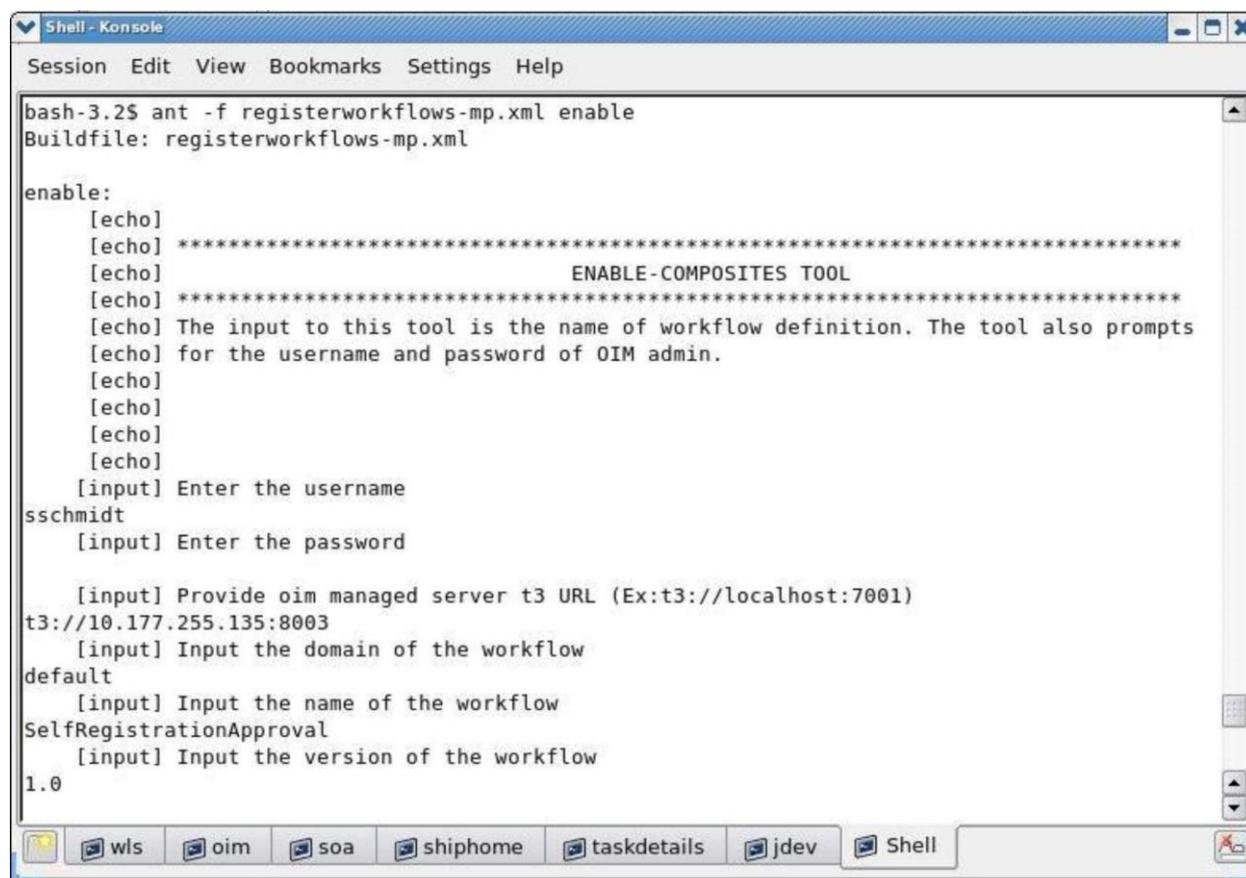
Provide oim managed server t3 URL
6. 次のプロンプトが表示されたら、ワークフローのドメインを入力します。

Input the domain of the workflow
7. 次のプロンプトが表示されたら、ワークフローの名前（SelfRegistrationApproval）を入力します。

Input the name of the workflow
8. 次のプロンプトが表示されたら、ワークフローのバージョン（1.0）を入力します。

Input the version of the workflow

次のスクリーンショット（図15）に、SOAコンポジットSelfRegistrationApprovalの有効化プロセスを示します。



```
bash-3.2$ ant -f registerworkflows-mp.xml enable
Buildfile: registerworkflows-mp.xml

enable:
  [echo]
  [echo] *****
  [echo]                               ENABLE-COMPOSITES TOOL
  [echo] *****
  [echo] The input to this tool is the name of workflow definition. The tool also prompts
  [echo] for the username and password of OIM admin.
  [echo]
  [echo]
  [echo]
  [input] Enter the username
sschmidt
  [input] Enter the password

  [input] Provide oim managed server t3 URL (Ex:t3://localhost:7001)
t3://10.177.255.135:8003
  [input] Input the domain of the workflow
default
  [input] Input the name of the workflow
SelfRegistrationApproval
  [input] Input the version of the workflow
1.0
```

図15 : SelfRegistrationApprovalの有効化