# Work Flow Engine

**Related MIS Term project,**
**Based on [http://www.uengine.org](http://www.uengine.org),**
**'The Workflow Management Coalition Specifiacation(wfMC)'**

with uEngine
the universal engine for e-Business

Lee, Yong Ki (blue1130@snu.ac.kr)

Digital Interactions Lab.(http://di.snu.ac.kr)

Dept. of IE, SNU

# Content

Content is..

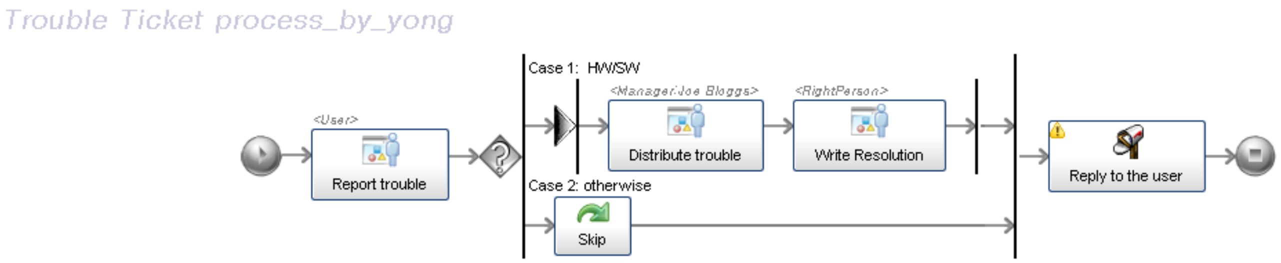
- Switch Activity
- Event-driven Sub-process
- Database Activity
- URLform Application
- ScopeActivity and EventHandler

# Flow control with Switch Activity

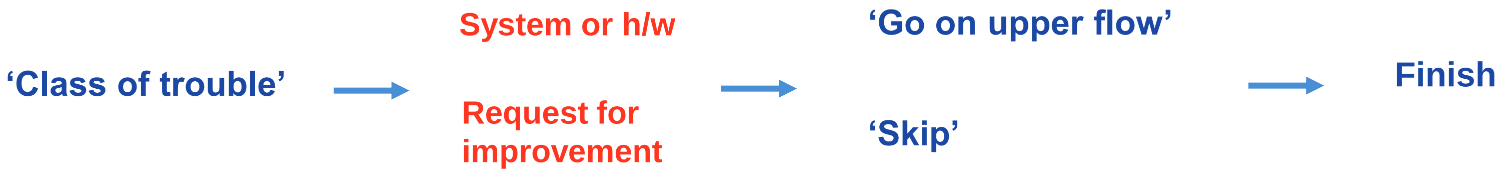
## Switch Activity

We will learn about Switch Activity that can run different activities with their conditions.

The example what we want to make is as below.



That is..



‘Class of trouble’

System or h/w

Request for improvement

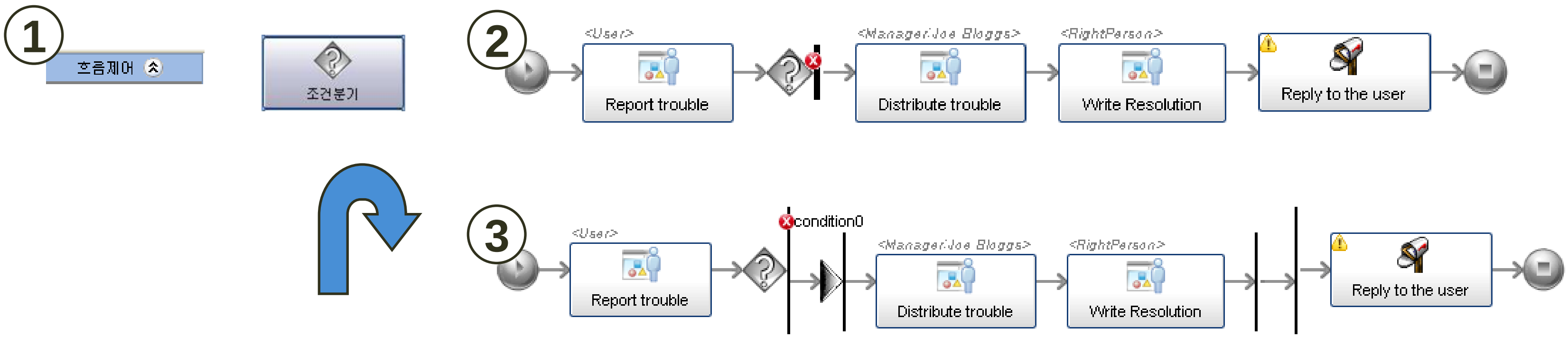
‘Go on upper flow’

‘Skip’

Finish

# Flow control with Switch Activity

## Switch Activity(cont’d)



**Step 1.**

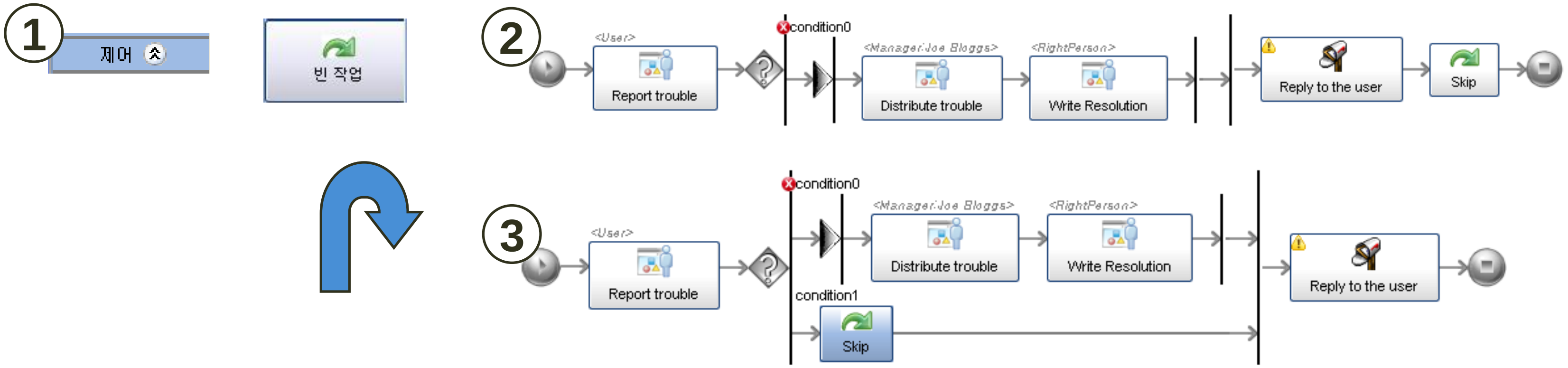
**Click the button ‘흐름제어’->‘조건분기’, and drag & drop the question mark to location which you want to locate.**

**Step 2.**

**And then input activities into the region of question mark.**

# Flow control with Switch Activity

## Switch Activity(cont'd)



**Step 3.**

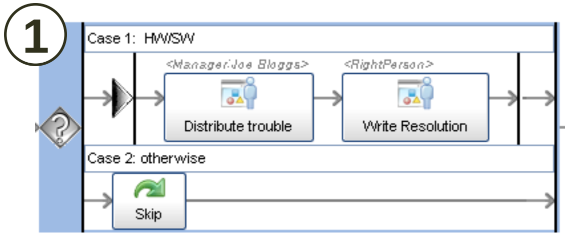
Iterate step 1~2.

# Flow control with Switch Activity

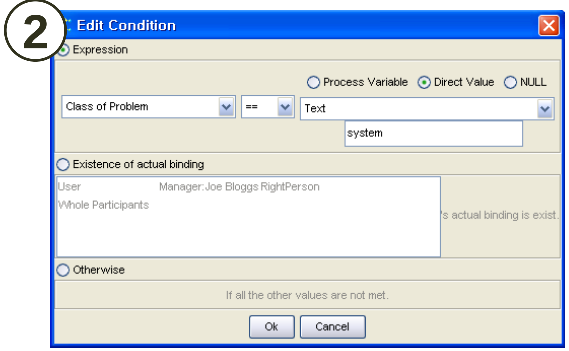
## Switch Activity(cont’d)

### Conditions Setting.

1



2



Edit Condition

● Expression

○ Process Variable ● Direct Value ○ NULL

Class of Problem == Text

system

○ Existence of actual binding

User    Manager:Joe Bloggs RightPerson

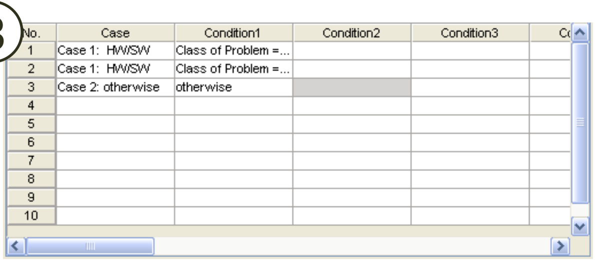
Whole Participants

's actual binding is exist.

○ Otherwise

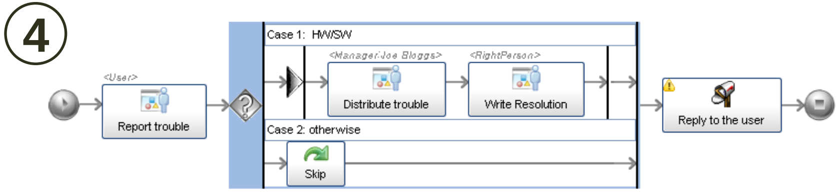
If all the other values are not met.

3



| No. | Case | Condition1 | Condition2 | Condition3 | Co |
|---|---|---|---|---|---|
| 1 | Case 1:  HW/SW | Class of Problem =... | | | |
| 2 | Case 1:  HW/SW | Class of Problem =... | | | |
| 3 | Case 2: otherwise | otherwise | | | |
| 4 | | | | | |
| 5 | | | | | |
| 6 | | | | | |
| 7 | | | | | |
| 8 | | | | | |
| 9 | | | | | |
| 10 | | | | | |

4



Step 4.

Input the name of conditions and Edit the box of conditions.

and then figure 3 is created.

If you want to confirm this process(conditional branch), deploy this model or simulate it.

# Content

Content is..

- Switch Activity
- Event-driven Sub-process
- Database Activity
- URLform Application
- ScopeActivity and EventHandler

# Sub Process

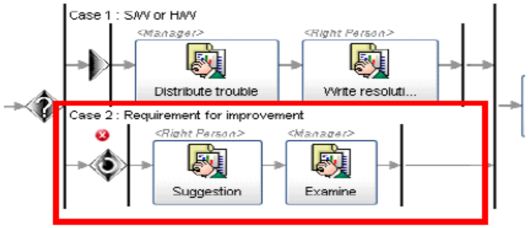
## Definition

### Suppose that..

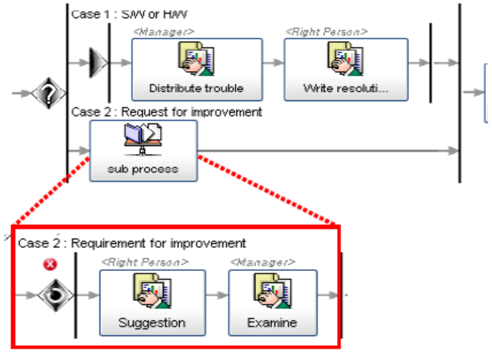
You are willing to add new activities into the second flow with functions that can make a suggestion for improvement.

You can make it as type A or as type B

A



B



### The merit of Sub-process

- It is resuable in other main processes.
- Large process can be divided into sub processes to be easy when were monitored

# Sub Process

What we want to make is ‘Suggestion for improvement’ process for ‘request for improvement

1

| Role name(ID) | Display name |
|---|---|
| Drafter | Drafter |
| Manager | Manager |

2

| Variable name | Display name | Type | Inputter |
|---|---|---|---|
| Contents | Contents | Text | TextAreaInput(80,5) |
| Approved | Approved | Yes/No | |

3

| Activity name | Role | Parameter |
|---|---|---|
| Suggestion | Drafter | Contents |
| Examine | Manager | Contents(in), approved |

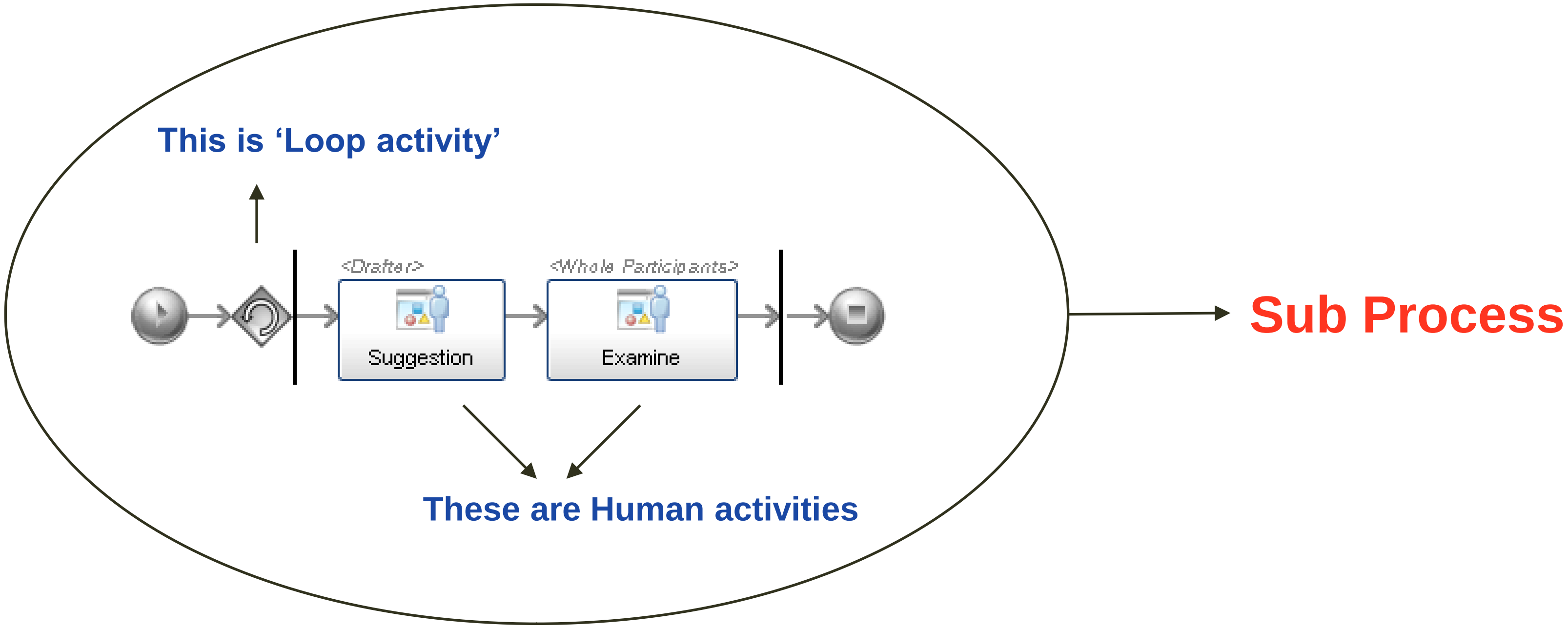
4

| Variable | Logic | Value |
|---|---|---|
| Approved | == | No(Direct value) |

1. Participants role definition
2. Variable definition
3. Human work activity property
4. Loop activity condition

# Sub Process

## Sub process



This is ‘Loop activity’

These are Human activities

Sub Process

After finished, deploy it and see the result. Do not forget executing by **‘Set as production’**

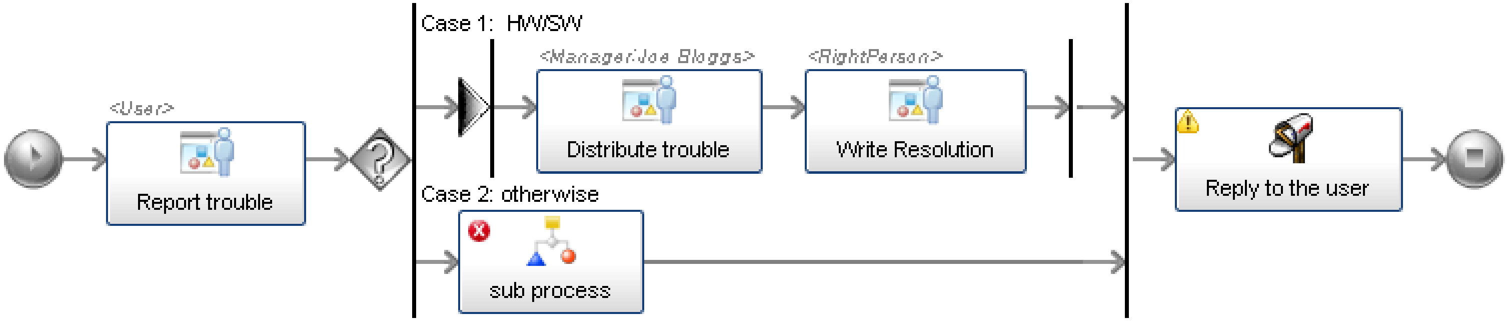
# Sub Process

## Set up ‘Sub process’

**First, Delete current ‘Skip’ activity from Second flow.**

**Second, Click ‘Sub process’ on the activity pallet and set it in the second block.**

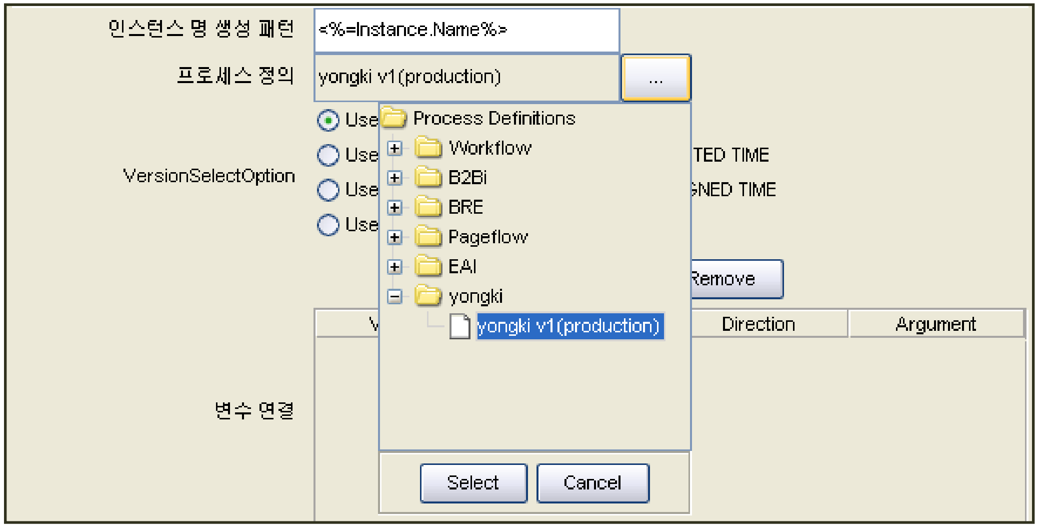
**The feature is illustrated as below.**



# Sub Process

## Set up ‘Sub process’(cont’d)

**To set up ‘Sub process’ property, click ‘definition ID’ in sub process property window and select what you want from process definition list as below feature.**



인스턴스 명 생성 패턴 <%=Instance.Name%>

프로세스 정의 yongki v1(production) ...

VersionSelectOption
● Use
○ Use
○ Use
○ Use

TED TIME
GNED TIME

변수 연결

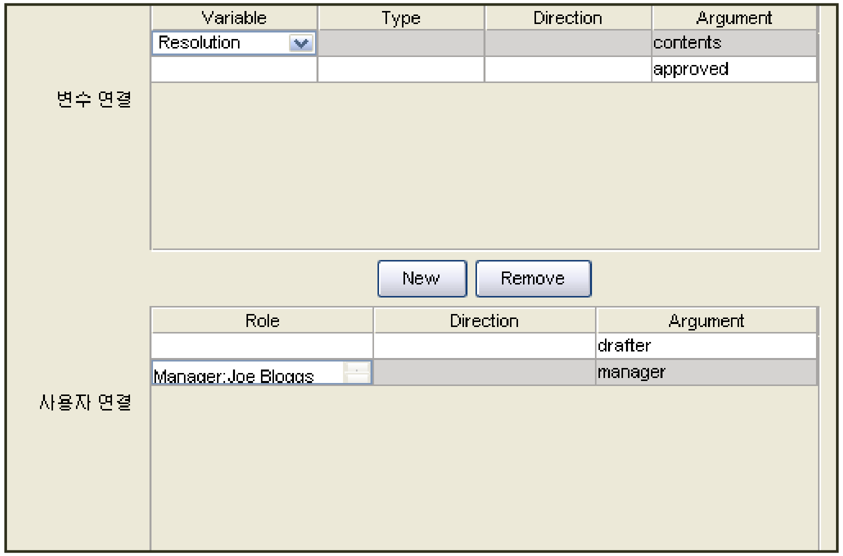
# Sub Process

## Set up ‘Sub process’(cont’d)

To use ‘Sub process’ we have to bind the variables and roles between two processes.

So..

1. Select ‘Version Select Option’ (recommend ‘Use the current production version’)
2. Variable and User definition as below feature



변수 연결

| Variable | Type | Direction | Argument |
|---|---|---|---|
| Resolution | | | contents |
| | | | approved |

사용자 연결

| Role | Direction | Argument |
|---|---|---|
| | | drafter |
| Manager:Joe Bloggs | | manager |

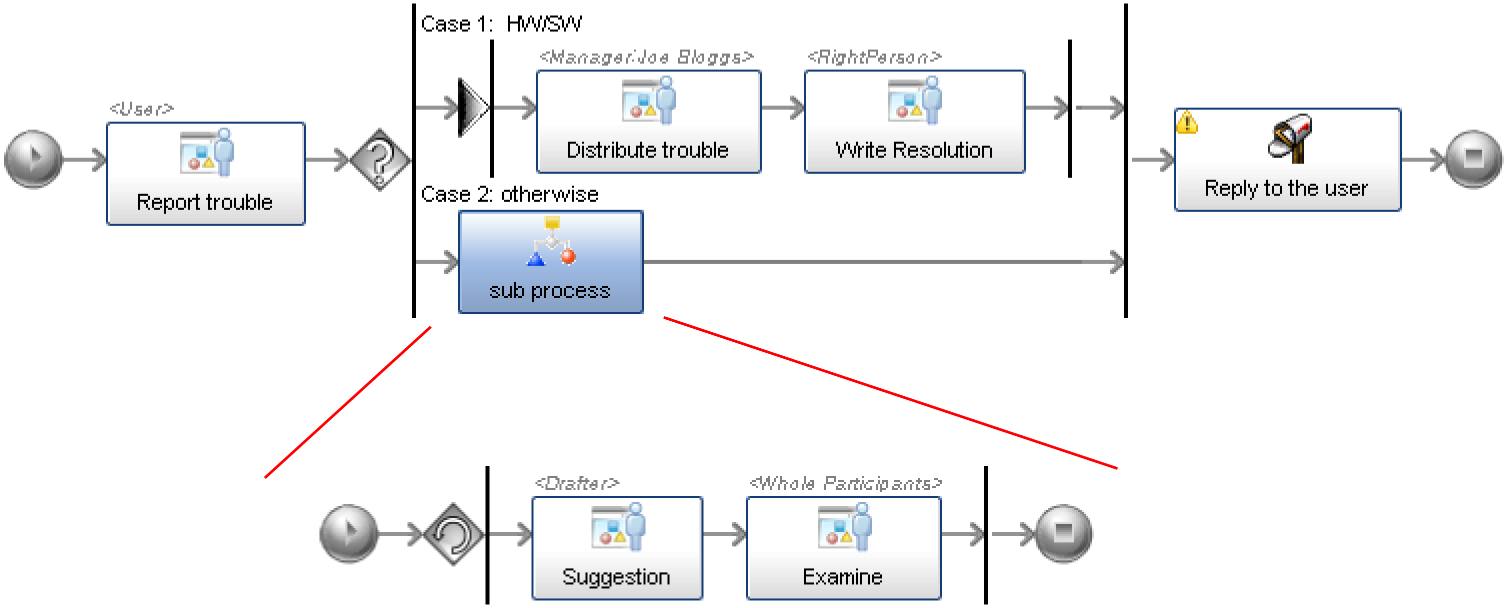
Now deploy main process

and let’s see how it works.

# Sub Process

## Set up ‘Sub process’(cont’d)

Now deploy main process and let’s see how it works.



# Content

Content is..

- Switch Activity
- Event-driven Sub-process
- Database Activity
- URLform Application
- ScopeActivity and EventHandler

# Database Activity

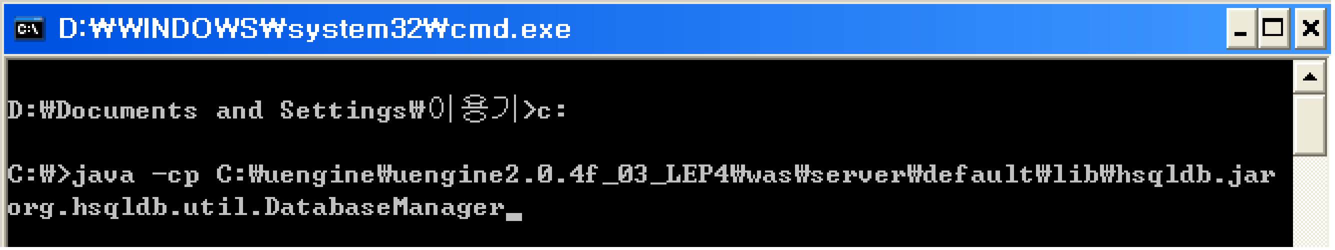
## Hypersonic DB

We will learn about how to make and use database linkage

uEngine can connect to several types of database, but we will use **default uEngine DB-HyperSonic Database**

Run Hypersonic Client by command as below.

java –cp[path(hsqldb.jar]hsql.jar org.hsqldb.util.DatabaseManager



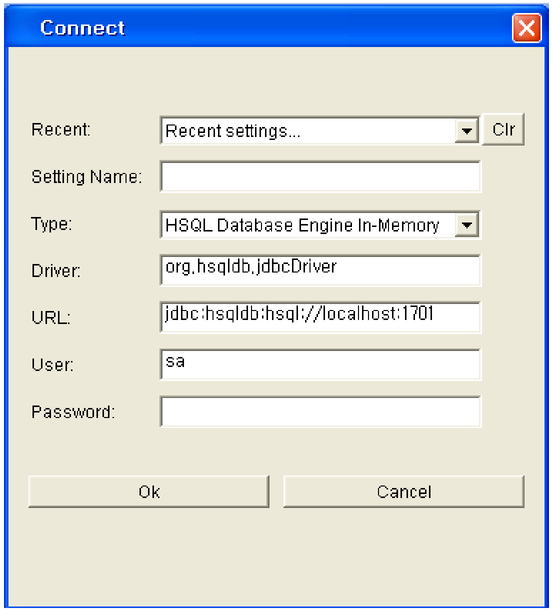
```
D:₩Documents and Settings₩이홍기>c:

C:₩>java -cp C:₩uengine₩uengine2.0.4f_03_LEP4₩was₩server₩default₩lib₩hsqldb.jar
org.hsqldb.util.DatabaseManager_
```

# Database Activity

## Hypersonic DB

Input ‘jdbc:hsqldb:hsql://localhost:1701’ into URL text box and connect.



Connect

Recent: Recent settings...

Setting Name:

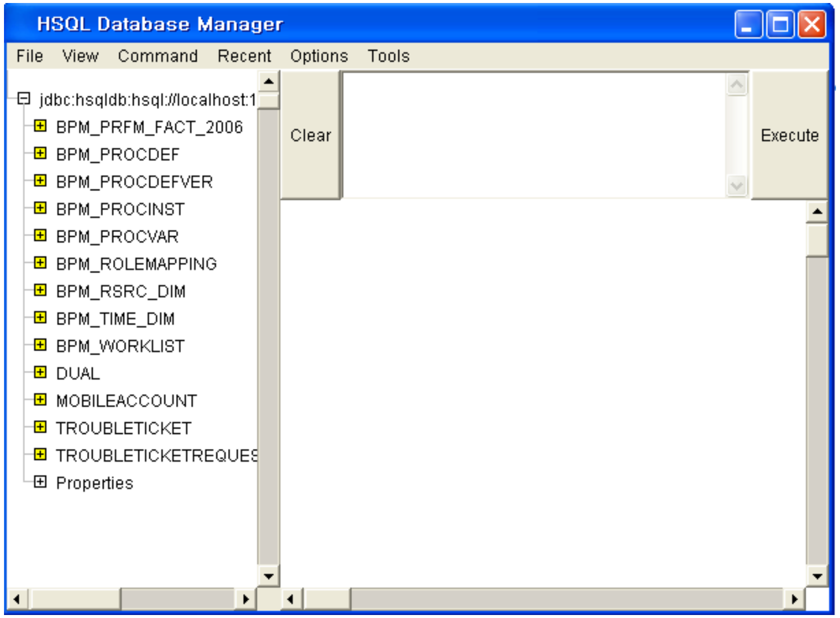
Type: HSQL Database Engine In-Memory

Driver: org.hsqldb.jdbcDriver

URL: jdbc:hsqldb:hsql://localhost:1701

User: sa

Password:



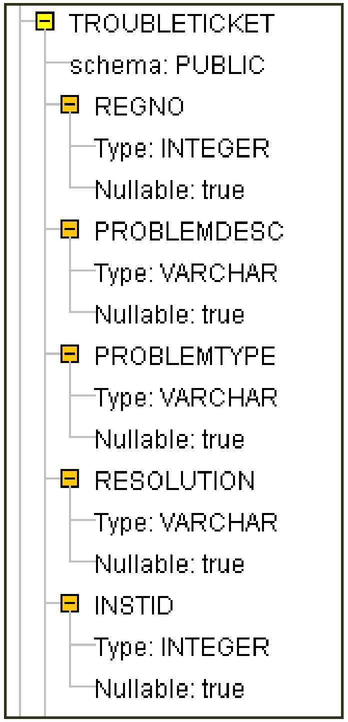
HSQL Database Manager

```
⊟ jdbc:hsqldb:hsql://localhost:1
   ⊞ BPM_PRFM_FACT_2006
   ⊞ BPM_PROCDEF
   ⊞ BPM_PROCDEFVER
   ⊞ BPM_PROCINST
   ⊞ BPM_PROCVAR
   ⊞ BPM_ROLEMAPPING
   ⊞ BPM_RSRC_DIM
   ⊞ BPM_TIME_DIM
   ⊞ BPM_WORKLIST
   ⊞ DUAL
   ⊞ MOBILEACCOUNT
   ⊞ TROUBLETICKET
   ⊞ TROUBLETICKETREQUES
   ⊞ Properties
```

# Database Activity

## Hypersonic DB

Here we will use ‘Trouble Ticket’ table for holding process results and it consists of fields as chart below.



```
⊟ TROUBLETICKET
  schema: PUBLIC
  ⊟ REGNO
     Type: INTEGER
     Nullable: true
  ⊟ PROBLEMDESC
     Type: VARCHAR
     Nullable: true
  ⊟ PROBLEMTYPE
     Type: VARCHAR
     Nullable: true
  ⊟ RESOLUTION
     Type: VARCHAR
     Nullable: true
  ⊟ INSTID
     Type: INTEGER
     Nullable: true
```

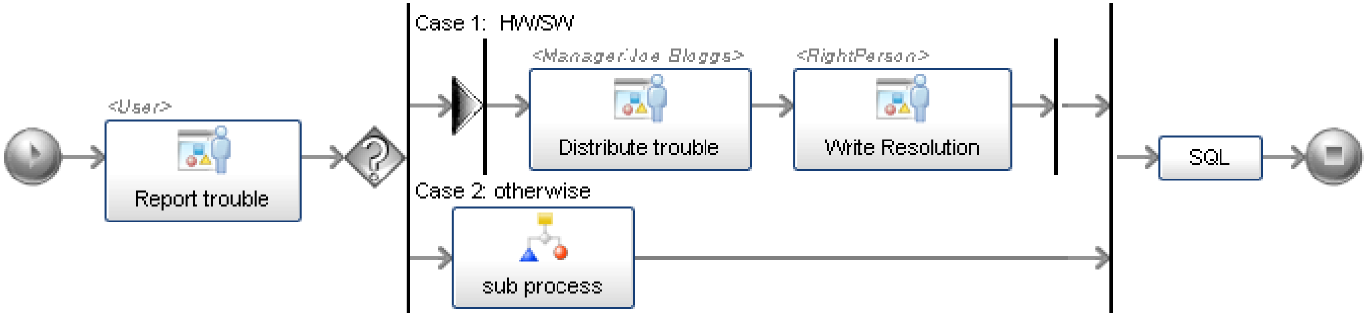
After deploying process,

you can watch its progress in Hypersonic DB here.

# Database Activity

## SQL Activity

**First, Call 'SQL Activity' from pallet and change its name to 'send to database' as below feature.**



**Second, Input SQL query as below into the text box in property window.**

```
Insert into troubleticket (regno, problem type, problem desc)
values(<%=Instance.InstanceId%>, ?,?)
```

**Finally, Set the parameters for sending to database. (We have to save two variables**

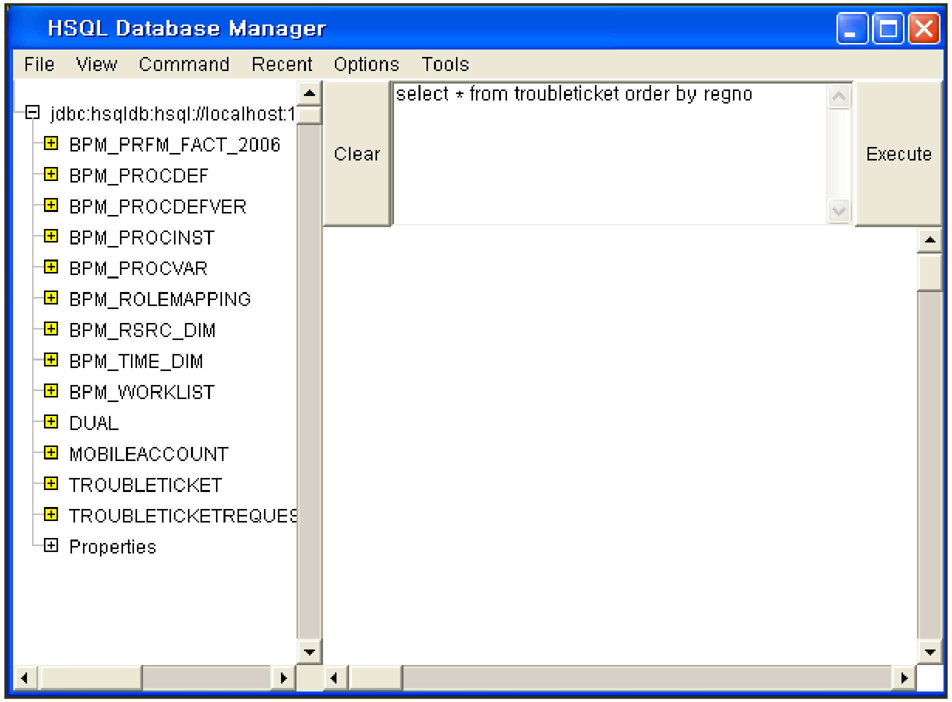
**– 'Class of trouble' and 'Trouble description'**

# Database Activity

## SQL Activity(cont'd)

Deploying trouble ticket process, open table and see what changed.

Check if new fields added to the table as you inputted as below feature.



```
select * from troubleticket order by regno
```

# Content

Content is..

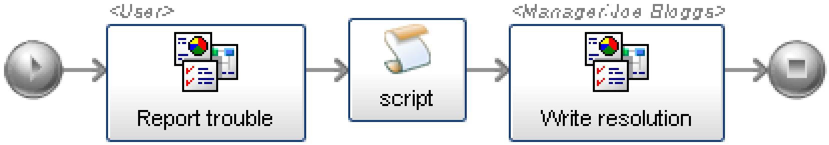
- Switch Activity
- Event-driven Sub-process
- Database Activity
- URLform Application
- ScopeActivity and EventHandler

# URL Form Application

## Web Application Activity

Web Application Activity can make us **reuse existing Web application** when you make user workitem handler.

The example for Web application is simplified form of Trouble Ticket and consists of two URL Application activities. It will show external linkage with **JSP program acts ‘Report Trouble’ and ‘Write Resolution’** as below feature.



# URL Form Application

## Web Application Activity(cont'd)

JSP files are as below.

Location :
\was\server\default\deploy\ext.ear\portal-web-complete.war\html\uengine-Web\sample_url_applications\troubleticket

Contents

- Register_index.jsp
- Register_summit.jsp
- write_resolution_index.jsp
- wirte_resolution_submit.jsp
- View.jsp

# URL Form Application

## Web Application Activity(cont’d)

### Role and Variable Definition

| Role name(ID) | Display name | Property |
|---|---|---|
| User | User | Contents |
| Manager | Manager | Roll mapping |

| Variable name | Display name | Type |
|---|---|---|
| Regno | Regno | Text |
| resultURL | resultURL | Text |

Variable ‘regno’ is identifier for saving trouble report to database.

Variable ‘resultURL’ is to save URL address from the result URL.

# URL Form Application

## Web Application Activity(cont’d)

### URL linked activity properties

| Factor | Description |
|---|---|
| URL | Web page address to call |
| URL when success | Web page address to move on to next activity<br>(This activity finishes when ‘URL when success’ page shows up) |
| Result URLPV | Variable for URL Address at the finished point |

# URL Form Application

## Web Application Activity(cont'd)

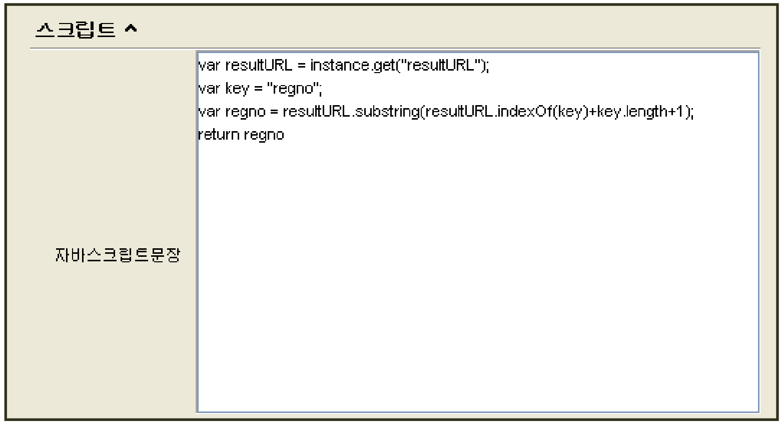
To control URL form, we need to translate result URL address into usable shape with 'Script activity'

That is..(in the Example)

After finishing previous actiity, variable 'resultURL' contains URL address in it.

And URL address has value of 'regno' so that we can parse it from URL form to refer again in the next process continuously.

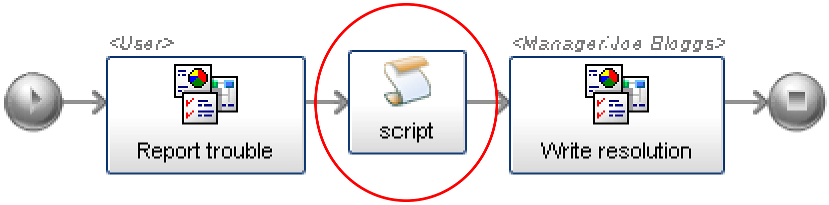
So 'Script activity' will get regno from the 'resultURL'.



스크립트 ^

자바스크립트문장

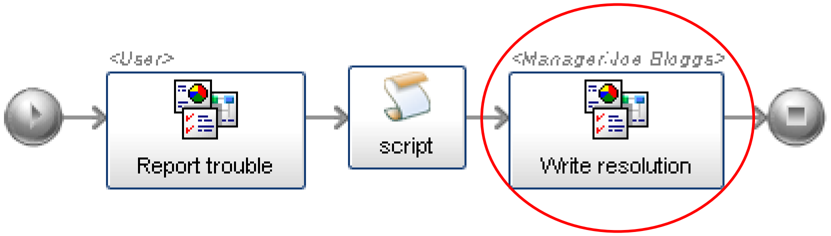
```
var resultURL = instance.get("resultURL");
var key = "regno";
var regno = resultURL.substring(resultURL.indexOf(key)+key.length+1);
return regno
```



# URL Form Application

## Web Application Activity(cont'd)

The next step is to make activity that writes resolution and saves.



That is..

Data flow(1)

Register_index.jsp → Get 'Class of trouble' Get 'Trouble description' → View.jsp

Data flow(2)

Get 'regno and resultURL' through Script → Write_resolution_index.jsp → View.jsp

# Content

Content is..

- Switch Activity
- Event-driven Sub-process
- Database Activity
- URLform Application
- ScopeActivity and EventHandler

# Scope Activity and Event Handler

## Scope Activity and Event Handler

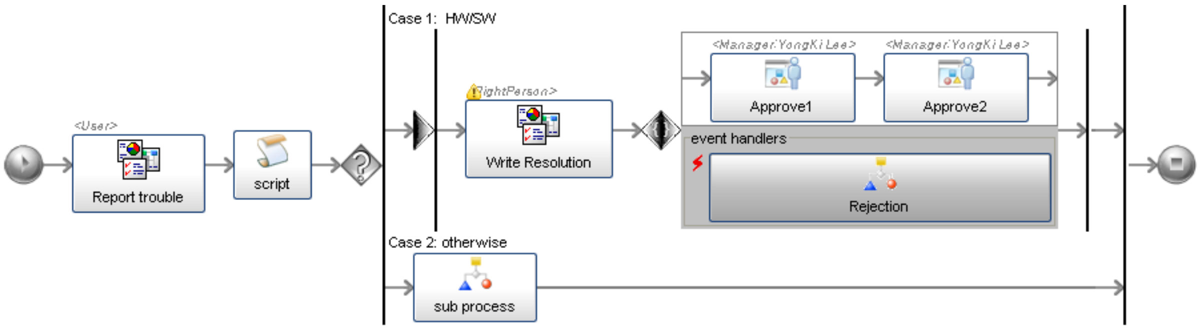
Scope Activity makes it possible to control activities we made as a group.

And the Activities will have common event handler that can stop main process and run sub process whenever called.

**That is..**

The trouble resolution report should pass two stage approvals.

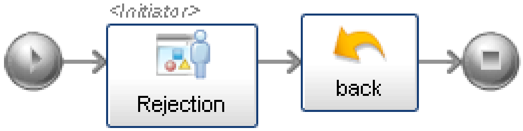
In case of rejection, Right person should make resolution report again.



# Scope Activity and Event Handler

## Scope Activity and Event Handler(cont’d)

‘Back Activity’ makes rejection to Right person.



Set activity property as below.

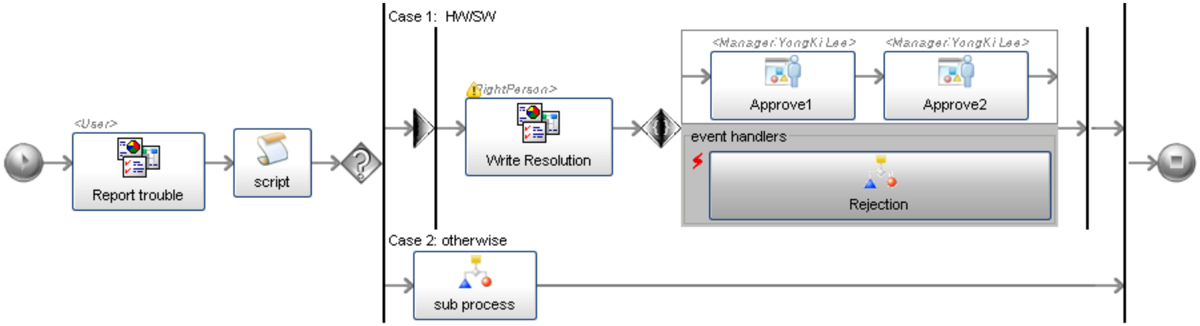
| Factor | Value |
|---|---|
| TargetActivityPV | |
| TargetActivity/InstanceID | <%=Instance.MainProcessInstanceID%> |
| TargetActivity/TracingTag | Follow ‘sub process number’ |

TracingTag is a pointer that indicates where to go back by Back activity.

# Scope Activity and Event Handler

## Scope Activity and Event Handler(cont’d)

Make sub process for the event(each other) and deploy it.



# Thank You !