

บันทึกวิจัย

การติดตั้ง iBroker, iService และ Web Application ให้อยู่ในสภาพแวดล้อมของกริดสารสนเทศ

นัยนา สหเวชภัณฑ์ และคำรณ อรุณเรื่อ

โครงการกริดสารสนเทศ

หน่วยปฏิบัติการวิจัยการจำลองขนาดใหญ่
ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

กันยายน 2551

สารบัญ

1	บทนำ.....	4
2	การติดตั้ง Information Broker (iBroker).....	6
3	การติดตั้ง Information Service (iService).....	7
4	การลงทะเบียน iService กับ iBroker.....	9
5	การติดตั้ง Web Server ให้อยู่ในสภาพแวดล้อมของกริด	10
6	การติดต่อกับ iBroker	12
7	การติดต่อกับ iService	12

สารบัญรูปภาพ

รูปที่ 2-1 การให้บริการของ iBroker.....	6
รูปที่ 3-1 การให้บริการของ iService.....	8
รูปที่ 4-1 deployTemplate.xml	9
รูปที่ 5-1 Tomcat Home.....	10
รูปที่ 5-2 Authentication Required (Tomcat Manager Application)	11
รูปที่ 5-3 หน้าจอแสดงถึงความสำเร็จในการติดตั้ง Web Server บน Globus	11
รูปที่ ก-1 หน้าจอแสดงถึงความสำเร็จในการติดตั้ง web server	16

1 บทนำ

ระบบกริดสารสนเทศ[1] คือระบบซอฟต์แวร์ที่สนับสนุนการบูรณาการข้อมูลจากแหล่งข้อมูลต่างๆ บนพื้นฐานของเทคโนโลยีกริดคอมพิวเตอร์ โดยระบบกริดสารสนเทศมีองค์ประกอบที่สำคัญคือ (1) เอกสาร MDL – มาตรฐานโครงสร้างข้อมูลหนึ่งๆบนพื้นฐานของภาษา MDL (Marker Directory Language) (2) Marker Directory Service (MDS)– ระบบซอฟต์แวร์ที่ทำหน้าที่ในการจัดเก็บเอกสาร MDL และให้บริการการสืบค้นเอกสาร MDL (3) Information Broker (iBroker) – ระบบซอฟต์แวร์ที่ทำหน้าที่เป็นนายหน้าในการให้บริการข้อมูล ตลอดจนการบูรณาการข้อมูลจากแหล่งข้อมูลต่างๆ และ (4) Information Service (iService) -- ระบบซอฟต์แวร์ที่ทำหน้าที่ในการเข้าถึงข้อมูลของแหล่งข้อมูลหนึ่งๆ กล่าวโดยสรุป iBroker และ iService เป็นองค์ประกอบหลักที่ทำหน้าที่เกี่ยวข้องกับข้อมูลโดยตรง ในขณะที่เอกสาร MDL ทำหน้าที่ในการสร้างมาตรฐานให้กับแหล่งข้อมูลต่างๆ

ทั้งนี้ การสร้าง “แหล่งข้อมูลเสมือน” ภายใต้ระบบกริดสารสนเทศมีความจำเป็นอย่างยิ่ง เพื่ออำนวยความสะดวกให้แก่ผู้ใช้บริการข้อมูลในการเข้าถึงแหล่งข้อมูลใดๆ โดยเทคโนโลยีที่นำมาใช้ในการสร้างแหล่งข้อมูลเสมือน คือ เทคโนโลยีกริดคอมพิวเตอร์ ดังนั้นในการพัฒนาระบบกริดสารสนเทศเพื่อให้บริการสร้างแหล่งข้อมูลเสมือน จึงจำเป็นต้องทำการติดตั้ง iBroker และ iService ให้อยู่ในสภาพแวดล้อมของกริด ตลอดจนการติดตั้ง Web Application ซึ่งพัฒนาบนพื้นฐานของระบบกริดสารสนเทศ

รายงานบันทึกวิจัยฉบับนี้ จึงขอนำเสนอถึงขั้นตอนการติดตั้ง iBroker, iService และ Web Application ให้อยู่ในสภาพแวดล้อมของกริด รวมถึงขั้นตอนในการติดตั้งซอฟต์แวร์พื้นฐานที่จำเป็นต่อการติดตั้ง iBroker, iService และ Web Application นอกจากนี้ยังได้นำเสนอถึงขั้นตอนการลงทะเบียน iService กับ iBroker เพื่อให้ iBroker รู้จัก iService ที่มาทำการลงทะเบียนด้วย ทั้งนี้เพื่อสนับสนุนให้ iBroker นั้นๆ สามารถทำการบูรณาการข้อมูลจากแหล่งข้อมูลต่างๆผ่านทางชุดของ iServices ที่มาลงทะเบียนด้วย ทั้งนี้การนำเสนอขั้นตอนการติดตั้งดังกล่าวได้อยู่บนพื้นฐานของซอฟต์แวร์ที่ได้จัดเตรียมไว้ให้ในแผ่นซีดี ซึ่งประกอบด้วย 3 ไดรกเทอรีที่สำคัญดังนี้

1. ไดรกเทอรี requiredSw บรรจุซอฟต์แวร์ในการติดตั้งซอฟต์แวร์พื้นฐาน ซึ่งประกอบด้วย
 - Tool for Software Build Process apache-ant-1.6.5-bin.zip
 - Java SE Development Kit (JDK) jdk1.5.0.tar
 - Globus-Container ws-core-4.0.5-src.zip
 - Application Server jakarta-tomcat-5.0.30.tar
2. ไดรกเทอรี iBrokerSW บรรจุซอฟต์แวร์ในการติดตั้ง iBroker ซึ่งประกอบด้วย
 - iBroker WS deploy batch file ไฟล์ iBrokerWSDeploy.bat
 - iBroker Web Service ไฟล์ org_igrid_informationbroker_wsrf.gar¹
 - iBroker API ไฟล์ iBroker.jar
 - iBroker configuration folder \iBroker
 - iService stub ไฟล์ org_igrid_isource_wsrf_stubs.jar
 - iService deploy file ไฟล์ deployTemplate.xml

¹ org_igrid_informationbroker_wsrf.gar ประกอบด้วย org_igrid_ibroker_wsrf.jar, org_igrid_ibroker_wsrf_stubs.jar, wsdl, wdd, jndi

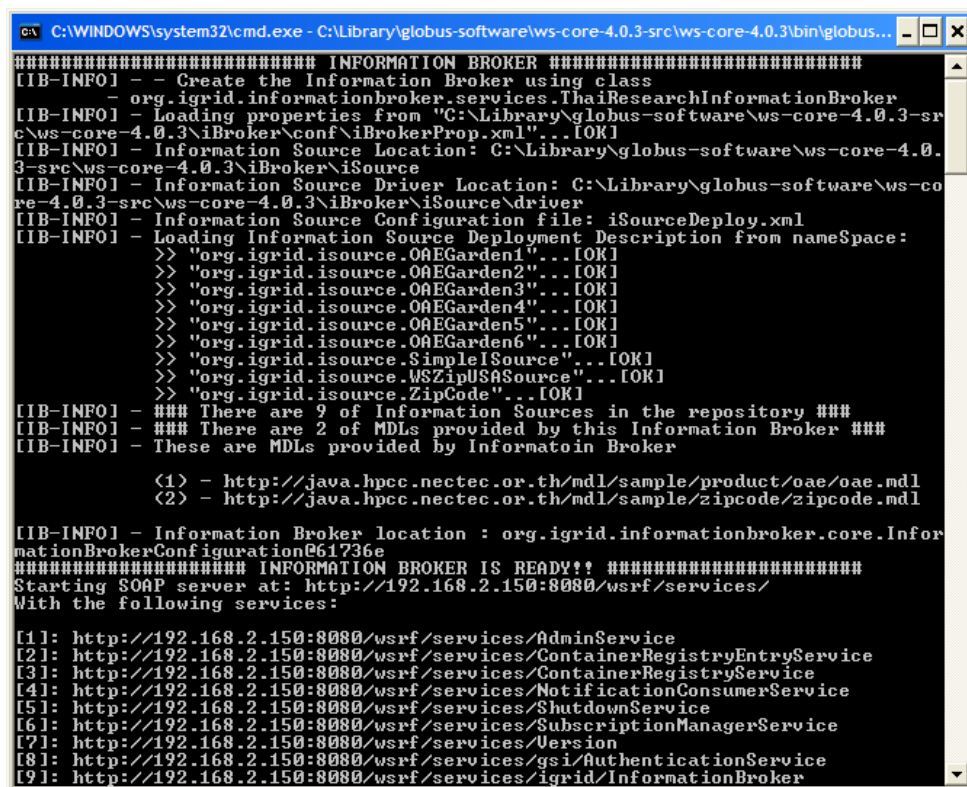
- example of iService deploy file ไฟล์ deployExample.xml
3. ไดรกเทอร์รี่ iServiceSW บรรจขอพต์แเวร์ในการติดตั้ง iService ซึ่งประกอบด้วย
- iService WS deploy batch file ไฟล์ iServiceWSDeploy.bat
 - iService Web Service ไฟล์ org_igrid_isource_wsrf.gar²
 - iService framework ไฟล์ igrid_informationsource_framework.jar
4. ไดรกเทอร์รี่ InformationBrokerClient บรรจขอพต์แเวร์ในการทดสอบการติดต่อกับ iBroker และ iService ซึ่งประกอบด้วย
- Source code folder \src
 - Binary code foler \bin

²org_igrid_isource_wsrf.gar ประกอบด้วย org_igrid_isource_wsrf.jar, org_igrid_isource_wsrf_stubs.jar, server-deploy.wsdd, InformationSource.wsdl, jndi-config-deploy.xml

2 การติดตั้ง Information Broker (iBroker)

การติดตั้ง Information Broker (iBroker) มีขั้นตอนดังนี้

1. ติดตั้งซอฟต์แวร์พื้นฐาน Tool for Software Build Process, Java SE Development Kit (JDK) และ Globus Container โดยมีรายละเอียดใน “ภาคผนวก ก. การติดตั้งซอฟต์แวร์พื้นฐาน”
2. Deploy iBroker ใน globus container โดย
 - รันไฟล์ d:\iBrokerSW\iBrokerWSDeploy.bat เพื่อ
 - deploy iBroker WS “org_igrid_iBroker_wsrf.gar ”
 - สำเนาไฟล์ d:\iBrokerSW\iBroker.jar ไปยัง %GLOBUS_LOCATION%\lib
 - สำเนา folder d:\iBrokerSW\iBroker ไปยัง %GLOBUS_LOCATION%
 - สำเนาไฟล์ d:\iBrokerSW\org_igrid_isource_wsrf_stubs.jar ไปยัง %GLOBUS_LOCATION%\lib
3. ทดสอบการติดตั้ง Information Broker โดย
 - รันไฟล์ %GLOBUS_LOCATION%\bin\globus-start-container.bat -noses -p <port number> เพื่อเป็นการ start container และ การให้บริการของ Information Broker ที่ port <port number> ดังแสดงในรูปแบบที่ 2-1



```
##### INFORMATION BROKER #####
[IIB-INFO] - - Create the Information Broker using class
- org.igrid.informationbroker.services.ThaiResearchInformationBroker
[IIB-INFO] - Loading properties from "C:\Library\globus-software\ws-core-4.0.3-src\ws-core-4.0.3-src\ws-core-4.0.3\iBroker\conf\iBrokerProp.xml"...[OK]
[IIB-INFO] - Information Source Location: C:\Library\globus-software\ws-core-4.0.3-src\ws-core-4.0.3\iBroker\iSource
[IIB-INFO] - Information Source Driver Location: C:\Library\globus-software\ws-core-4.0.3-src\ws-core-4.0.3\iBroker\iSource\driver
[IIB-INFO] - Information Source Configuration file: iSourceDeploy.xml
[IIB-INFO] - Loading Information Source Deployment Description from nameSpace:
  >> "org.igrid.isource.OAEGarden1"...[OK]
  >> "org.igrid.isource.OAEGarden2"...[OK]
  >> "org.igrid.isource.OAEGarden3"...[OK]
  >> "org.igrid.isource.OAEGarden4"...[OK]
  >> "org.igrid.isource.OAEGarden5"...[OK]
  >> "org.igrid.isource.OAEGarden6"...[OK]
  >> "org.igrid.isource.SimpleISource"...[OK]
  >> "org.igrid.isource.WSZipUSASource"...[OK]
  >> "org.igrid.isource.ZipCode"...[OK]
[IIB-INFO] - ### There are 9 of Information Sources in the repository ###
[IIB-INFO] - ### There are 2 of MDLs provided by this Information Broker ###
[IIB-INFO] - These are MDLs provided by Informatoin Broker
  (1) - http://java.hpcc.nectec.or.th/mdl/sample/product/oea/oea.mdl
  (2) - http://java.hpcc.nectec.or.th/mdl/sample/zipcode/zipcode.mdl
[IIB-INFO] - Information Broker location : org.igrid.informationbroker.core.InformationBrokerConfiguration@61736e
##### INFORMATION BROKER IS READY!! #####
Starting SOAP server at: http://192.168.2.150:8080/wsrf/services/
With the following services:
[1]: http://192.168.2.150:8080/wsrf/services/AdminService
[2]: http://192.168.2.150:8080/wsrf/services/ContainerRegistryEntryService
[3]: http://192.168.2.150:8080/wsrf/services/ContainerRegistryService
[4]: http://192.168.2.150:8080/wsrf/services/NotificationConsumerService
[5]: http://192.168.2.150:8080/wsrf/services/ShutdownService
[6]: http://192.168.2.150:8080/wsrf/services/SubscriptionManagerService
[7]: http://192.168.2.150:8080/wsrf/services/Version
[8]: http://192.168.2.150:8080/wsrf/services/ysi/AuthenticationService
[9]: http://192.168.2.150:8080/wsrf/services/igrid/InformationBroker
```

รูปที่ 2-1 การให้บริการของ iBroker

3 การติดตั้ง Information Service (iService)

การติดตั้ง Information Service (iService) มีขั้นตอนดังนี้

- ติดตั้งซอฟต์แวร์พื้นฐาน Tool for Software Build Process, Java SE Development Kit (JDK) และ Globus Container ดังรายละเอียดใน“ภาคผนวก ก. การติดตั้งซอฟต์แวร์พื้นฐาน” ทั้งนี้
 - ในกรณีที่ทำการติดตั้ง iBroker และ iService ไว้ในเครื่องคอมพิวเตอร์เดียวกัน ให้ทำการติดตั้ง globus container สำหรับ iService ใน path ที่แตกต่างจากการติดตั้ง globus container สำหรับ iBroker นอกจากนี้ set environment variable ของ GLOBUS_LOCATION ให้สอดคล้องกับการติดตั้ง globus container นั้นๆเมื่อมีการอ้างอิงถึง iBroker หรือ iService
 - ในกรณีที่ทำการติดตั้ง Information Service มากกว่า 1 service ไว้ในเครื่องคอมพิวเตอร์เดียวกัน ให้ทำการติดตั้ง globus container สำหรับ iService ใดๆใน path ที่แตกต่างกัน ตลอดจนการ set environment variable ของ GLOBUS_LOCATION ให้สอดคล้องกับการติดตั้ง globus container สำหรับ iService นั้นๆ
- Deploy iService ใน globus container โดย
 - รันไฟล์ d:\iServiceSW\iServiceWSDeploy.bat เพื่อ
 - deploy iService WS “org_igrid_isource_wsrf.gar”
 - สำเนาไฟล์ d:\iServiceSW\igrid_informationsource_framework.jar ไปยัง %GLOBUS_LOCATION%\lib
 - สำเนา iService ไปยัง %GLOBUS_LOCATION%\lib
 - ในกรณีที่ใช้ *GiSTool* (สร้าง iService ที่ชื่อ iServiceInstance.jar): ให้สำเนาไฟล์ iServiceInstance.jar ไปยัง %GLOBUS_LOCATION%\lib หรือ
 - ในกรณีที่พัฒนาเองตาม *Information Service Framework* (สร้าง iService ที่ชื่อ iServiceInstance.jar): ให้สำเนาไฟล์ iServiceInstance.jar ไปยัง %GLOBUS_LOCATION%\lib
- ปรับปรุง เพื่อให้ WSRP รู้จัก class หลักใน iServiceInstance.jar ที่ implement ฟังก์ชัน getInformation()
 - เปิดไฟล์ %GLOBUS_LOCATION%\etc\org_igrid_isource_wsrf\server-config.wsdd
 - แก้ไข value ใน parameter name ที่มีค่าเท่ากับ “iSource” ให้เป็น absolute path ของ class หลักใน iService ที่ implement ฟังก์ชัน getInformation()
 - ในกรณีที่ใช้ *GiSTool*: ให้ค่า value เท่ากับ informationSource.DatabaseISource

```
<parameter name="iSource" value="informationSource.DatabaseISource"/>
```

หรือ

- ในกรณีที่พัฒนาเองตาม *Information Service Framework* : ให้ค่า value เท่ากับ absolute path ของ class หลักใน *iService* ที่ implement ฟังก์ชัน *getInformation()* ตามที่ได้พัฒนาไว้

4. ทดสอบการติดตั้ง *Information Service* โดย

- รันไฟล์ `%GLOBUS_LOCATION%\bin\globus-start-container.bat -nosec -p <port number>` เพื่อเป็นการ start container และ การให้บริการของ *i Service* ที่ port `<port number>` (สามารถเปลี่ยนเป็น port number ใดๆ โดยที่ยังไม่มีการใช้งานได้) ดังแสดงในรูปที่ 3-1

```

C:\WINDOWS\system32\cmd.exe - C:\Library\globus-software\ws-core-4.0.3-src\ws-core-4.0.3-iSource\bi...
InformationSource Created
Starting SOAP server at: http://192.168.2.150:1978/wsrf/services/
With the following services:
[1]: http://192.168.2.150:1978/wsrf/services/AdminService
[2]: http://192.168.2.150:1978/wsrf/services/ContainerRegistryEntryService
[3]: http://192.168.2.150:1978/wsrf/services/ContainerRegistryService
[4]: http://192.168.2.150:1978/wsrf/services/NotificationConsumerService
[5]: http://192.168.2.150:1978/wsrf/services/ShutdownService
[6]: http://192.168.2.150:1978/wsrf/services/SubscriptionManagerService
[7]: http://192.168.2.150:1978/wsrf/services/Uversion
[8]: http://192.168.2.150:1978/wsrf/services/gsi/authenticationService
[9]: http://192.168.2.150:1978/wsrf/services/igrid/isource/InformationSourceService

```

รูปที่ 3-1 การให้บริการของ *iService*

หมายเหตุ:

- ในกรณีที่มีการปรับปรุง *iService* ให้ดำเนินการตั้งแต่ขั้นตอนที่ 2 จากส่วน “สำเนา *iService*” .เข้าสู่ระบบเป็นต้นไป
- ในกรณีที่ใช้ *GiSTool*: ให้ทำการสำเนา Library ของ *GiSTool* มายัง `%GLOBUS_LOCATION%\lib` ของ *iService* นั้นๆ เพื่อให้สามารถใช้งาน *iService* ได้อย่างสมบูรณ์ ทั้งนี้ เนื่องจากในการทำงานของ *iService* นั้น ต้องการใช้ *Hibernate driver* และ *JDBC driver* ซึ่งถ้าไม่ดำเนินการสำเนานั้นจะเกิด error เมื่อมีการเรียกใช้งาน *iService*

no class def found error org/hibernate/criterion

4 การลงทะเบียน iService กับ iBroker

การลงทะเบียน iService กับ iBroker เป็นหน้าที่ที่ผู้ดูแล iBroker จะต้องดำเนินการ โดยมีขั้นตอนดังนี้

1. สร้าง folder <ชื่อ folder ที่ต้องการ> ใน %GLOBUS_LOCATION%\iBroker\iSource
2. ปรับปรุงไฟล์ deploy.xml เพื่อให้ Information Broker รู้จัก Information Service
 - สำเนาไฟล์ d:\iBrokerSW\deployTemplate.xml ไปยัง %GLOBUS_LOCATION%\iBroker\iSource\<ชื่อ folder ที่ต้องการ> โดยมีข้อมูล ดังนี้

```
<source-deploy>
  <source name=..... type=...>
    <provider> </provider>
    <location> ... </location>
    <description> ... </description>
    <mdl-list>
      <uri> ... </uri>
    </mdl-list>
  </source>
  <parameter name="className" value="...">
  <parameter name="allowMDL" value="*">
  <parameter name="wsdl" value="...">
</source-deploy>
```

รูปที่ 4-1 deployTemplate.xml

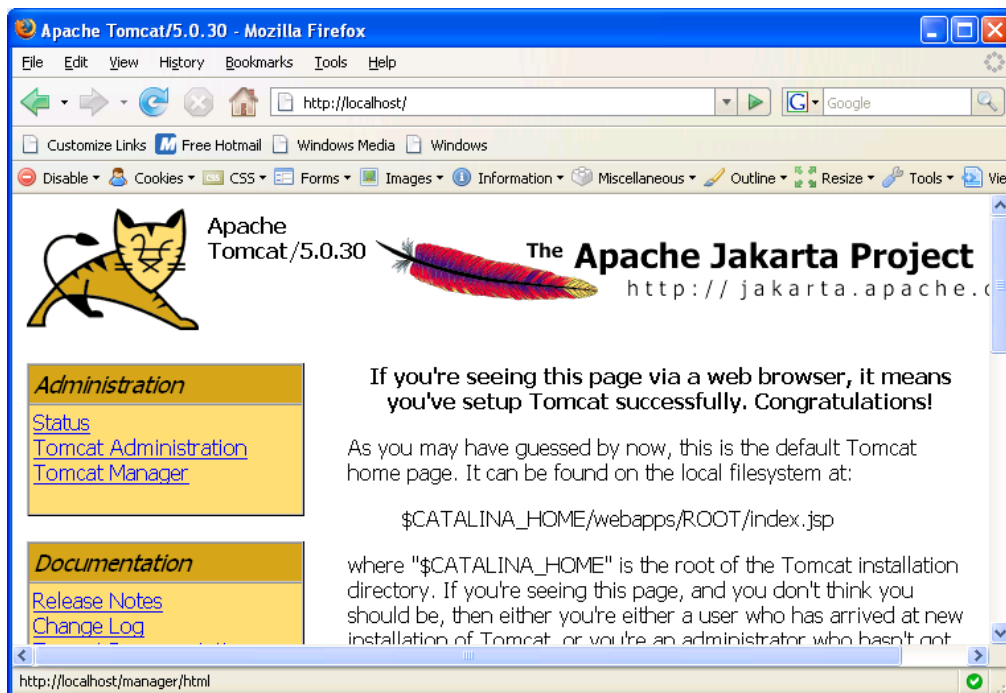
- เปลี่ยนชื่อ deployTemplate.xml เป็น deploy.xml
 - บันทึกข้อมูลต่างๆดังนี้ ให้ครบถ้วน (ศึกษาจากตัวอย่างใน d:\iBrokerSW\deployExample.xml)
 - ชื่อ iService และประเภท iService ใน <source name="....." type="...."> (type สามารถเป็น REMOTE เมื่อ iBroker และ iService อยู่คนละ server กัน หรือ LOCAL เมื่อ iBroker และ iService อยู่ server เดียวกัน)
 - uri ของ เอกสาร MDL ที่ iService ให้บริการ ใน <mdl-list> <uri> ...
 - ชื่อ class หลักใน iService ที่ implement ฟังก์ชัน getInformation() ใน <parameter name="className" value="....."/> (ในกรณีที่ใช้ GiSTool สร้าง iService นั้น value จะมีค่าเป็น informationSource.DatabaseISource)
 - endpoint ของ iService ใน <parameter name="wsdl" value="....." /> (สามารถดู endpoint ของ iSource ได้จากการ start globus container ของ iSource นั้นๆ ดังแสดงในรูปที่ 3-1 โดย แสดงอยู่ในลำดับที่ 9 (endpoint ในลำดับที่ 1-8 เป็นของ globus)
3. รันไฟล์ %GLOBUS_LOCATION%\bin\globus-start-container.bat -p <port number> เพื่อเป็นการ start container และ การให้บริการของ iBroker ที่ port <port number> อีกครั้ง ทั้งนี้เพื่อให้ iBroker รู้จัก iService ที่มาลงทะเบียนใหม่

5 การติดตั้ง Web Server ให้อยู่ในสภาพแวดล้อมของกริด

Web Application ที่พัฒนาบนพื้นฐานของ Information Grid จะมีการเรียกใช้งาน iBroker ซึ่งตั้งอยู่บนสภาพแวดล้อมของกริด ดังนั้นจึงจำเป็นต้องทำการติดตั้ง Web Application ให้อยู่ในสภาพแวดล้อมของกริดด้วย โดยมีขั้นตอนดังนี้ (ในที่นี้ใช้ Tomcat เป็น Application Server)

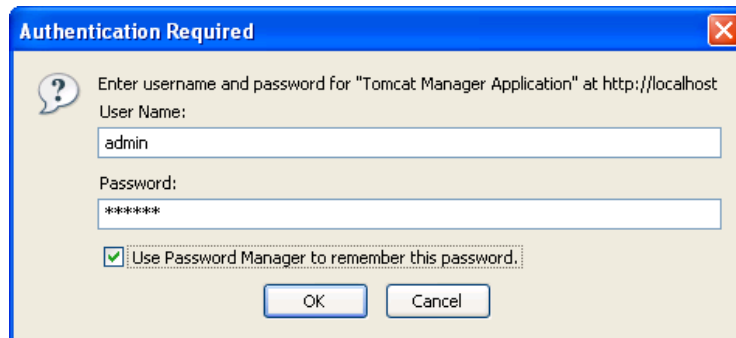
เปิด service ของ Information Broker, Information Source และ tomcat ก่อนที่จะทำการรัน

1. ติดตั้ง Web Server ดังรายละเอียดใน “ภาคผนวก ก. การติดตั้งซอฟต์แวร์พื้นฐาน”
2. ติดตั้ง iBroker แต่ไม่ต้อง start container
3. ติดตั้ง globus บน Web Server
 - cd %GLOBUS_LOCATION% ของ iBroker
 - รันไฟล์ `ant -f share\globus_wsrf_common\tomcat\tomcat.xml deployTomcat -Dtomcat.dir=%TOMCAT_HOME%`
4. ลบ iBroker เพื่อไม่ให้ iBroker มารันที่ Web Server ทั้งนี้ เนื่องจากต้องการเพียงสภาพแวดล้อม และ Library ของ globus เท่านั้น
 - `Rmdir /s /q %TOMCAT_HOME%\webapps\wsrf\WEB_INF\etc\org_igrid_informationbroker_wsrf`
5. ทดสอบการใช้งาน globus บน Tomcat โดย
 - นำไฟล์ test.jsp ไปยัง %TOMCAT_HOME%\webapps\wsrf
 - รันไฟล์ `%TOMCAT_HOME%\bin\catalina.bat run` เพื่อเริ่มต้นการใช้งาน Tomcat
 - เปิด browser เพื่อไปยัง `http://localhost` จะปรากฏหน้าดังรูปที่ 5-1



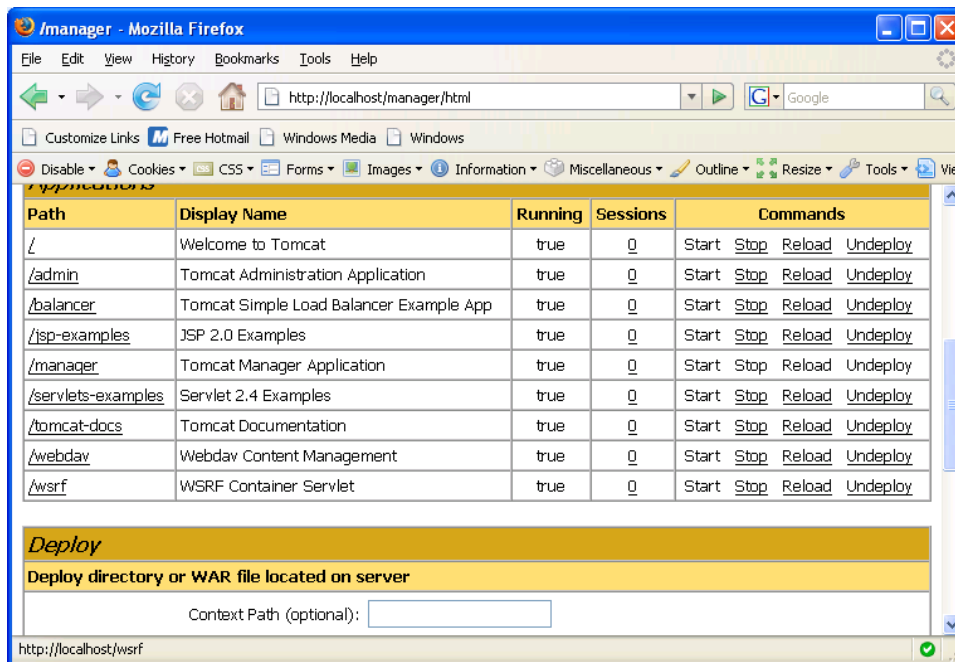
รูปที่ 5-1 Tomcat Home

- เลือกเมนู “Tomcat Manager” โดยหน้าจอจะปรากฏดังรูปที่ 5-2



รูปที่ 5-2 Authentication Required (Tomcat Manager Application)

- ให้บันทึก user name = admin และ password = tomcat และกดปุ่ม “OK” โดยหน้าจอจะปรากฏดังรูปที่ 5-3 ซึ่งถ้าเห็น path “/wsrf” จะแสดงถึงความสำเร็จในการติดตั้ง Web Server บน globus



รูปที่ 5-3 หน้าจอแสดงถึงความสำเร็จในการติดตั้ง Web Server บน Globus

6 การติดต่อกับ iBroker

1. สำเนา d:\InformationBrokerClient มายัง <path ที่ต้องการ>
2. เปิด Dos Console และ start globus container ของ iBroker ที่จะติดต่อกับ
3. ปรับแก้ uri สำหรับ iBroker ที่ต้องการติดต่อ ในโปรแกรม SimpleClient.java (อยู่ใน package org.igrid.informationbroker.client) ซึ่งเป็น โปรแกรม client ที่ใช้ในการติดต่อกับ iBroker

String factoryURI =

```
"http://10.226.39.105:8080/wsrf/services/igrid/InformationBrokerFactoryService";
```

4. เปิด Dos Console เพื่อรัน โปรแกรม SimpleClient.java
 - รันไฟล์ %GLOBUS_LOCATION%\etc\globus-dev-env-bat
 - รันไฟล์ <path ที่ติดตั้ง>\bin\java org.igrid.informationbroker.client.SimpleClient

7 การติดต่อกับ iService

1. import project InformationBrokerClient ใน d:\InformationBrokerClient เข้าสู่ Eclipse
2. เปิด Dos Console และ start globus container ของ iService ที่จะติดต่อกับ
3. ปรับแก้ uri สำหรับ iService ที่ต้องการติดต่อ ในโปรแกรม Main.java (อยู่ใน package org.igrid.isource.client) ซึ่งเป็น โปรแกรม client ที่ใช้ในการติดต่อกับ iService

```
String uri = "http://10.226.39.105:8001/wsrf/services/igrid/isource/InformationSourceService";
```

4. รัน โปรแกรม Main.java จาก Eclipse

ภาคผนวก ก. การติดตั้งซอฟต์แวร์พื้นฐาน

ซอฟต์แวร์พื้นฐานที่จำเป็นต่อการติดตั้ง Information Broker, Information Source และ Web Application ประกอบด้วย

- Tool for Software Build Process apache-ant-1.6.5-bin.zip
- Java SE Development Kit (JDK) jdk1.5.0.tar
- Globus-Container ws-core-4.0.5-src.zip
- Application Server jakarta-tomcat-5.0.30.tar

ก.1 การติดตั้ง Tool for Software Build Process

การติดตั้ง Tool for Software Build Process (ในที่นี้ใช้ Apache Ant) มีขั้นตอนดังนี้

1. unzip ไฟล์ d:\requiredSW\apache-ant-1.6.5-bin.zip ไปยัง <path ที่ต้องการติดตั้ง> ซึ่งเมื่อทำการ unzip เสร็จจะได้ “<path ที่ต้องการติดตั้ง>\apache-ant-1.6.5”
2. set variable environment โดยให้ variable “ANT_HOME” มีค่าเท่ากับ path ของ java build process “<path ที่ต้องการติดตั้ง>\apache-ant-1.6.5”
3. set variable environment โดยปรับปรุง variable “PATH” ให้มีค่า path ของ java build process “;%ANT_HOME%\bin” เพิ่ม

ก.2 การติดตั้ง Java SE Development Kit (JDK)

การติดตั้ง Java SE Development Kit (JDK) มีขั้นตอนดังนี้

1. untar ไฟล์ d:\requiredSW\jdk1.5.0.tar ไปยัง <path ที่ต้องการติดตั้ง> ซึ่งเมื่อทำการ untar เสร็จจะได้ “<path ที่ต้องการติดตั้ง>\jdk1.5.0”
2. set variable environment โดยให้ variable “JAVA_HOME” มีค่าเท่ากับ path ของ java library “<path ที่ต้องการติดตั้ง>\jdk1.5.0”
3. set variable environment โดยปรับปรุง variable “PATH” ให้มีค่า path ของ java library “;%JAVA_HOME%\bin” เพิ่ม

ก.3 การติดตั้ง globus-container

การติดตั้ง globus-container มีขั้นตอนดังนี้

1. unzip ไฟล์ d:\requiredSW\ws-core-4.0.5-src.zip ไปยัง <path ที่ต้องการติดตั้ง> ซึ่งเมื่อทำการ unzip เสร็จจะได้ “<path ที่ต้องการติดตั้ง>\ws-core-4.0.3”
2. set variable environment โดยให้ variable “GLOBUS_LOCATION” มีค่าเท่ากับ path ของ globus-container “<path ที่ต้องการติดตั้ง>\ws-core-4.0.3”
3. cd %GLOBUS_LOCATION%

4. รันไฟล์ ant เพื่อสร้าง globus container ซึ่งเมื่อสร้างสำเร็จจะมีข้อความ “Build Successful” ปรากฏ ทั้งนี้สามารถทดสอบการ start และ stop globus container โดย

- run ไฟล์ %GLOBUS_LOCATION%\bin\globus-start-container.bat –nosed เพื่อเป็นการ start container
- run ไฟล์ %GLOBUS_LOCATION%\bin\globus-stop-container.bat –nosed เพื่อเป็นการ stop container

หมายเหตุ: ควร set variable environment สำหรับ GLOBUS_LOCATION ก่อนทุกครั้ง มิฉะนั้น เมื่อ run ant แล้วเสร็จ ทุกไฟล์ที่สร้างขึ้นใหม่จะอยู่ภายใต้ folder “install”

ก.4 การติดตั้ง Application Server

การติดตั้ง application server (ในที่นี้ใช้ Apache Tomcat) มีขั้นตอนดังนี้

1. untar ไฟล์ d:\requiredSW\apache-tomcat-5.5.25.tar ไปยัง <path ที่ต้องการติดตั้ง> ซึ่งเมื่อทำการ untar เสร็จจะได้ “<path ที่ต้องการติดตั้ง>\jakarta-tomcat-5.5.25”
2. set variable environment โดยให้
 - variable “CATALINA_HOME” มีค่าเท่ากับ path ของ web server “<path ที่ต้องการติดตั้ง>\jakarta-tomcat-5.5.25”
 - variable “TOMCAT_HOME” มีค่าเท่ากับ path ของ web server “<path ที่ต้องการติดตั้ง>\jakarta-tomcat-5.5.25”
3. ปรับปรุงข้อมูลผู้ใช้งานเว็บเพจ เพื่อให้สามารถบริหารจัดการข้อมูลในเว็บเพจได้
 - cd %TOMCAT_HOME%
 - เปิดไฟล์ %TOMCAT_HOME%\conf\tomcat-users.xml โดยมีข้อมูลดังนี้

```
<tomcat-users>
  <user name="tomcat" password="tomcat" roles="tomcat" />
  <user name="role1" password="tomcat" roles="role1" />
  <user name="both" password="tomcat" roles="tomcat, role1" />
</tomcat-users>
```

- ปรับปรุงข้อมูลผู้ใช้งานเว็บเพจในไฟล์ %TOMCAT_HOME%\conf\tomcat-users.xml ให้เป็นดังนี้ (ไม่ควรก๊อปปี้ เพราะอาจจะทำให้โปรแกรมผิดพลาดได้)

```
<?xml version='1.0' encoding=utf-8'?>
<tomcat-users>
  <role rolename="manager" />
  <role rolename="admin" />
  <role rolename=" tomcat" />
  <role rolename=" role1" />
  <user name="tomcat" password="tomcat" roles="tomcat" />
  <user name="role1" password="tomcat" roles="role1" />
  <user name="both" password="tomcat" roles="tomcat, role1" />
  <user name="admin" password="tomcat" roles="tomcat,
role1,manager,admin" />
</tomcat-users>
```

หมายเหตุ: ควรสำเนาไฟล์ tomcat-users.xml ไว้ เนื่องจากไฟล์นี้จะถูกลบ เมื่อทำการ undeploy jakarta-tomcat-5.5.25

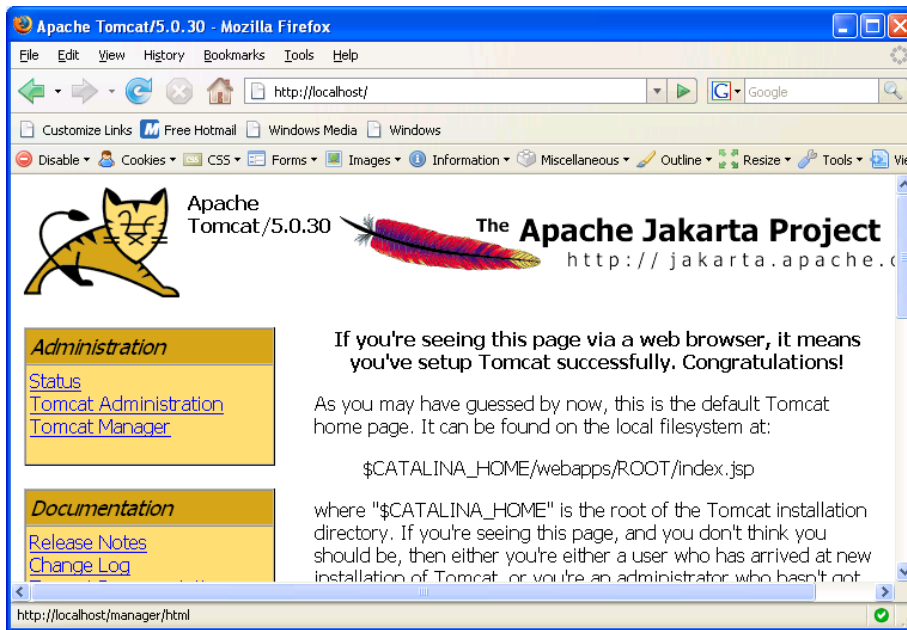
4. ปรับปรุงข้อมูล http port เพื่อไม่ให้ซ้ำกับ port ที่มีการเปิดให้บริการอยู่แล้ว ทั้งนี้เพื่อให้เป็น port สำหรับการใช้งาน internet
 - เปิดไฟล์ %TOMCAT_HOME%\conf\server.xml โดยมีข้อมูลดังนี้

```
<!--Define a non-SSL Coyote HTTP/1.1 Connector on port 8080 -->
<Connector port="8080"
  maxThread="150" minSpareThread="25" maxSpareThreads="75"
  enableLookup="false" redirectPort="8443" acceptCount="100"
  debug="0" connectionTimeout="20000" disableUploadTimeout="true" />

<Connector port="8443"
  maxThread="150" minSpareThread="25" maxSpareThreads="75"
  enableLookup="false" disableUploadTimeout="true" acceptCount="100" debug="0"
  scheme="https" secure="true" clientAuth="false" sslProtocol="TLS" />
.....
```

- เปลี่ยน port number ของ connector จาก “8080” เป็น port number อื่นที่ยังไม่ได้ถูกใช้งาน ทั้งนี้เนื่องจาก port number “8080”
5. ทดสอบความถูกต้องของการติดตั้ง web server โดย
 - cd %TOMCAT_HOME%\bin
 - รันไฟล์ catalina.bat run เพื่อ เริ่มต้นการ ใช้งาน web server

- เปิด browser เพื่อไปยัง `http://localhost:<port ที่ตั้งไว้>` ซึ่งถ้าหน้าจอปรากฏดังรูปที่ ก-1 แสดงว่าการติดตั้ง web server นั้นสมบูรณ์



รูปที่ ก-1 หน้าจอแสดงถึงความสำเร็จในการติดตั้ง web server