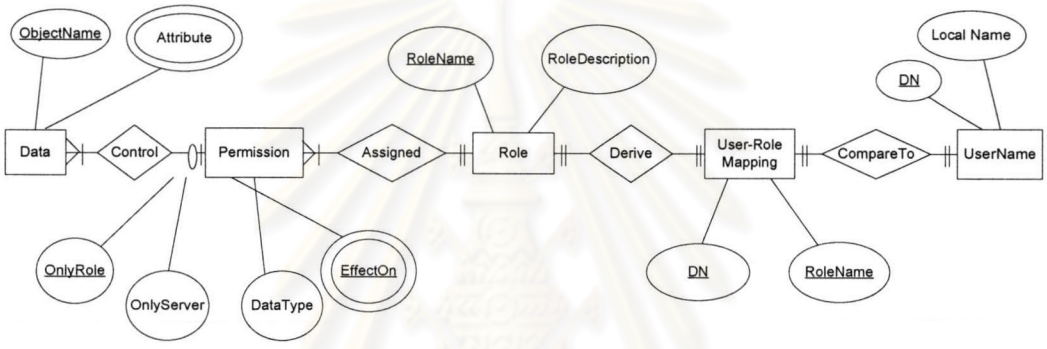
บทที่ 4

ต้นแบบระบบควบคุมการเข้าถึงข้อมูลเชิงบทบาทของเอ็มดีเอส

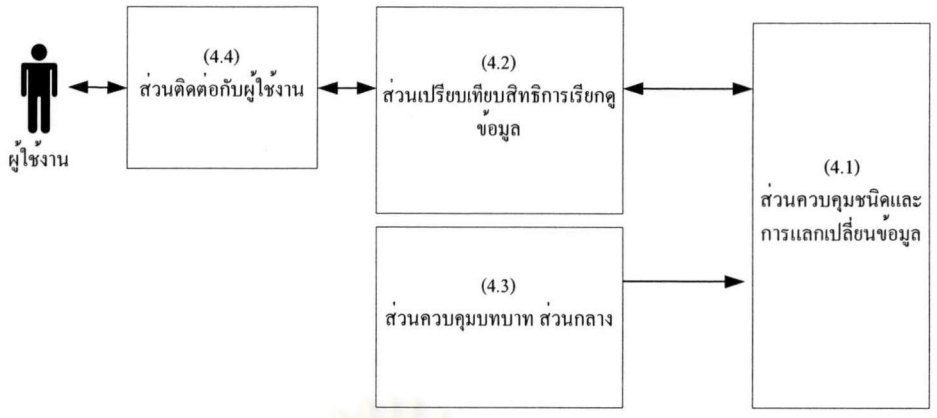
ต้นแบบระบบควบคุมการเข้าถึงข้อมูลเชิงบทบาทของเอ็มดีเอสภายในวิทยานิพนธ์นี้ ได้รับการพัฒนาขึ้นภายใต้ข้อกำหนดของสถาปัตยกรรมโกลบัส [1] รุ่น 2.0 ที่ถูกติดตั้งบน เร็ดแฮ้ดลีนุกซ์ รุ่น 7.3 โดยใช้ภาษาจาวารุ่น 1.4 และเชลล์สคริปต์ (Shell Script) ในการพัฒนาโดยได้มีการเพิ่มข้อมูลควบคุมสิทธิเชิงบทบาทซึ่งมีความสัมพันธ์และรายละเอียดภายในแต่ละข้อมูลแต่ละชนิด (ERD: Entity Relationship Diagram) ดังรูปที่ 4.1



รูปที่ 4.1 แสดงโครงสร้างความสัมพันธ์ระหว่างข้อมูลแต่ละชนิดภายในระบบควบคุมการเข้าถึงข้อมูลเชิงบทบาท

จากรูปที่ 4.1 จะแสดงให้เห็นว่าชื่อผู้ใช้งานแต่ละชื่อจะถูกกำหนดให้มีเพียงบทบาทเดียวเท่านั้นและข้อมูลควบคุมการเรียกดูข้อมูลที่ถูกกำหนดโดยเจ้าของข้อมูลจะสามารถถูกนำไปควบคุมข้อมูลจริงได้หลายชนิดพร้อมๆกัน โดยภายในข้อมูลควบคุมสิทธิจะระบุชื่อของบทบาทที่สามารถเรียกดูข้อมูลที่ถูกควบคุมนี้ได้

นอกจากนี้โครงสร้างของต้นแบบระบบควบคุมการเข้าถึงข้อมูลเชิงบทบาทของเอ็มดีเอสจะมีรายละเอียดดังรูปที่ 4.2 และตารางที่ 4.1



รูปที่ 4.2 โครงสร้างของต้นแบบระบบควบคุมการเข้าถึงข้อมูล

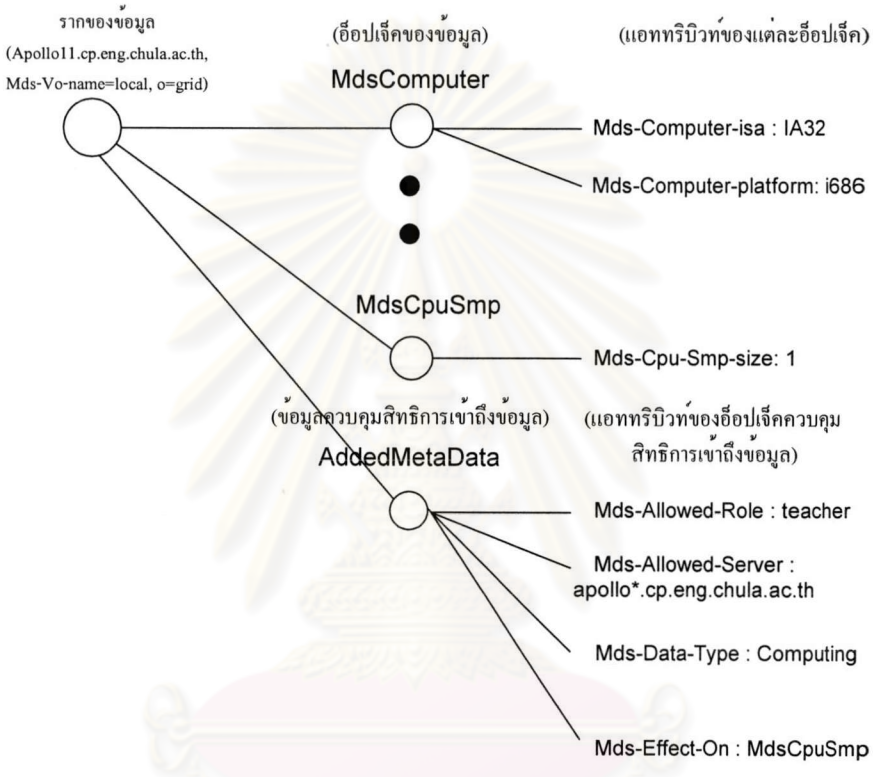
ชื่อองค์ประกอบ	หน้าที่	รายละเอียดในการพัฒนา
4.1 ส่วนควบคุมชนิดและการแลกเปลี่ยนข้อมูล	ทำการแลกเปลี่ยนข้อมูลระหว่างแต่ละเซิร์ฟเวอร์ภายในระบบกริด	ใช้บริการเอ็มดีเอสโดยเพิ่มข้อมูลควบคุมสิทธิการเรียกดูข้อมูล
4.2 ส่วนเปรียบเทียบสิทธิการเรียกดูข้อมูล	ทำการพิจารณาว่าผู้ใช้สามารถเรียกดูข้อมูลใดได้บ้าง	ใช้ภาษาจาวาในการพัฒนา
4.3 ส่วนควบคุมบทบาทส่วนกลาง	ทำหน้าที่เพิ่มเติมข้อมูลบทบาทส่วนกลาง	ใช้บริการเอ็มดีเอสโดยเพิ่มข้อมูลควบคุมบทบาทส่วนกลาง
4.4 ส่วนติดต่อกับผู้ใช้งาน	ทำหน้าที่รับความต้องการและแสดงข้อมูลจริงตามสิทธิของผู้ใช้	ใช้เชลล์สคริปสร้างชุดคำสั่งเลียนแบบชุดคำสั่งเดิมของบริการเอ็มดีเอส

ตารางที่ 4.1 แสดงรายละเอียดในการพัฒนาแต่ละองค์ประกอบภายในระบบควบคุม

จากรูปที่ 4.2 และตารางที่ 4.1 แสดงรายละเอียดการพัฒนาต้นแบบระบบควบคุมการเข้าถึงข้อมูลตามโครงสร้างที่ได้ออกแบบไว้ในส่วนที่ 3 โดยภายในแต่ละส่วนจะมีรายละเอียดและขั้นตอนในการพัฒนาดังต่อไปนี้

4.1 ส่วนควบคุมชนิดและการแลกเปลี่ยนของข้อมูล

ในส่วนควบคุมชนิดและการแลกเปลี่ยนข้อมูลของระบบควบคุมการเข้าถึงข้อมูลจะใช้โครงสร้างการแลกเปลี่ยนข้อมูลของบริการเอ็มดีเอสที่มีอยู่เดิมภายในระบบโกลบัส เพียงแต่จะเพิ่มข้อมูลควบคุมสิทธิในการเรียกดูข้อมูลเข้าไปในกรณีที่เจ้าของข้อมูลต้องการจะควบคุมการเรียกดูข้อมูลของตนเท่านั้น โดยโครงสร้างของข้อมูลจริงและข้อมูลควบคุมภายหลังจากที่ถูกเพิ่มเข้าไปในฐานข้อมูลแอลแดปภายในเซิร์ฟเวอร์ต้นตอของข้อมูล จะมีลักษณะดังรูปที่ 4.3



รูปที่ 4.3 แสดงโครงสร้างของข้อมูลภายหลังจากที่เพิ่มข้อมูลควบคุม

จากรูปที่ 4.3 แสดงให้เห็นถึงโครงสร้างของต้นไม้ข้อมูลซึ่งแต่ละโหนดคืออ็อปเจ็ค (Object) ของข้อมูลในหัวข้อต่างๆ ภายในแต่ละอ็อปเจ็คยังจะประกอบไปด้วยแอททริบิวท์ (Attribute) ที่อธิบายรายละเอียดภายในแต่ละหัวข้ออีกทีหนึ่ง เช่นแอททริบิวท์ของอ็อปเจ็ค "MdsComputer" ก็จะประกอบไปด้วยแอททริบิวท์ "Mds-Computer-Isa" และ "Mds-Computer-Platform" เป็นต้น ข้อมูลควบคุมสิทธิการเรียกดูข้อมูลที่เจ้าของข้อมูลเพิ่มเข้าไปจะอยู่ในรูปของอ็อปเจค ที่มีชื่อว่า "AddedMetaData" ซึ่งจะเป็นโหนดที่เชื่อมต่อกับโหนดรากของเครื่องที่เป็นต้นตอของข้อมูลโดยตรง แอททริบิวท์ของอ็อปเจ็คควบคุมสิทธิการเรียกดูข้อมูลจะประกอบไปด้วย

- ชื่อของบทบาทที่สามารถเรียกดูข้อมูลจริงนี้ได้ (Mds-Allowed-Role)
- ชื่อของเครื่องที่สามารถเรียกดูข้อมูลได้ (Mds-Allowed-Server)
- ข้อมูลอธิบายประเภทของข้อมูล (Mds-Data-Type)
- ชื่ออ็อปเจคของข้อมูลที่ถูกควบคุมโดยข้อมูลควบคุมสิทธินี้ (Mds-Effect-On)

ขั้นตอนการเพิ่มข้อมูลควบคุมสิทธิเข้าไปในส่วนควบคุมชนิดและการแลกเปลี่ยนของข้อมูลจะมีลำดับดังต่อไปนี้

1. ในการที่จะเพิ่มข้อมูลควบคุมสิทธิก่อนอื่นจำเป็นที่จะต้องเพิ่มชนิดของข้อมูลควบคุมสิทธิการเรียกดูข้อมูลเข้าไปไฟล์ควบคุมชนิดของข้อมูลภายในฐานข้อมูลแอลแดปของบริการเอ็มดีเอส (grid-info-resource.schema) เพื่อให้ทุกเซิร์ฟเวอร์สามารถแสดงข้อมูลควบคุมได้อย่างถูกต้อง ซึ่งจะมีรายละเอียดการเพิ่มเติมชนิดของข้อมูลควบคุมสิทธิดังต่อไปนี้

ในการเพิ่มเติมชนิดของแอททริบิวท์เข้าไปในไฟล์ควบคุมชนิดของข้อมูลเราจำเป็นต้องระบุรายละเอียดต่างๆดังต่อไปนี้

- ชื่อของแอททริบิวท์ (Name)
- หมายเลขเฉพาะของแอททริบิวท์
- คำอธิบายของแอททริบิวท์ (Desc)
- รายละเอียดเกี่ยวกับรูปแบบการสืบค้นแอททริบิวท์(Equality, Ordering, Substr)
- ชนิดของแอททริบิวท์ (Syntax) โดยการระบุหมายเลขอธิบายชนิด เช่น 1.3.6.1.4.1.1466.115.121.1.44 หมายถึงเป็นแอททริบิวท์ชนิดสตริง

รายละเอียดของแอททริบิวท์ที่ถูกเพิ่มเข้าไปภายในไฟล์ควบคุมชนิดของข้อมูลภายในฐานข้อมูลแอลแดปจะมีดังต่อไปนี้

```
attributetype ( 1.3.6.1.4.1.3536.2.6.3536.10.1.301        // กำหนดชนิดของแอททริบิวท์
   NAME 'Mds-Allowed-Role'
      DESC 'The role name who can view this information'
   EQUALITY caseIgnoreMatch
   ORDERING caseIgnoreOrderingMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.44
      SINGLE-VALUE
   )
 attributetype ( 1.3.6.1.4.1.3536.2.6.3536.10.1.302
   NAME 'Mds-Allowed-Server'
   DESC 'A single GRIS host name'
   EQUALITY caseIgnoreMatch
   ORDERING caseIgnoreOrderingMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.44
      SINGLE-VALUE
   )
      attributetype ( 1.3.6.1.4.1.3536.2.6.3536.10.1.303
    NAME 'Mds-Data-Type'
   DESC 'A single GIIS name'
   EQUALITY caseIgnoreMatch
   ORDERING caseIgnoreOrderingMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.44
      SINGLE-VALUE
   )
      attributetype ( 1.3.6.1.4.1.3536.2.6.3536.10.1.304
   NAME 'Mds-Effect-On'
   DESC 'A single GIIS host name'
   EQUALITY caseIgnoreMatch
```

```
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.44
 SINGLE-VALUE
)
```

ในการกำหนดชนิดของอ็อปเจ็คของข้อมูลควบคุมสิทธิการเรียกดูข้อมูลเราจำเป็นจะต้องกำหนดรายละเอียดดังต่อไปนี้

- ชื่อของอ็อปเจ็ค (Name)
- หมายเลขเฉพาะประจำตัวของอ็อปเจ็ค
- คำอธิบายของอ็อปเจ็ค (Desc)
- ชนิดของแอททริบิวท์ภายในอ็อปเจ็ค (Structural) โดยสามารถระบุได้ว่าจำเป็นจะต้องประกอบไปด้วยแอททริบิวท์อะไรบ้าง และแอททริบิวท์ใดบ้างที่สามารถละไว้ได้ (Must & May)

ตัวอย่างรายละเอียดของอ็อปเจคควบคุมสิทธิการเรียกดูข้อมูลที่เพิ่มเข้าไปภายในไฟล์ควบคุมชนิดของข้อมูลภายในฐานข้อมูลแอลแดปจะมีลักษณะดังต่อไปนี้

```
   objectclass ( 1.3.6.1.4.1.3536.2.6.3536.10.1.300          // กำหนดชนิดของอ็อปเจค
NAME 'AddedMetaData'
DESC 'A JMS namespace extension to Mds by NM'
SUP 'Mds'
STRUCTURAL
   MUST ( Mds-Allowed-Role $ Mds-Effect-On )   // อธิบายความสัมพันธ์ระหว่างอ็อปเจค
   MAY ( Mds-Allowed-Server $ Mds-Data-Type )    และแอททริบิวท์ที่ได้กำหนดขึ้น
)
```

การเพิ่มอ็อปเจ็คควบคุมสิทธิการเรียกดูข้อมูล (AddedMetaData) ไปพร้อมกับข้อมูลจริงจะทำให้เจ้าของข้อมูลสามารถมั่นใจได้ว่า

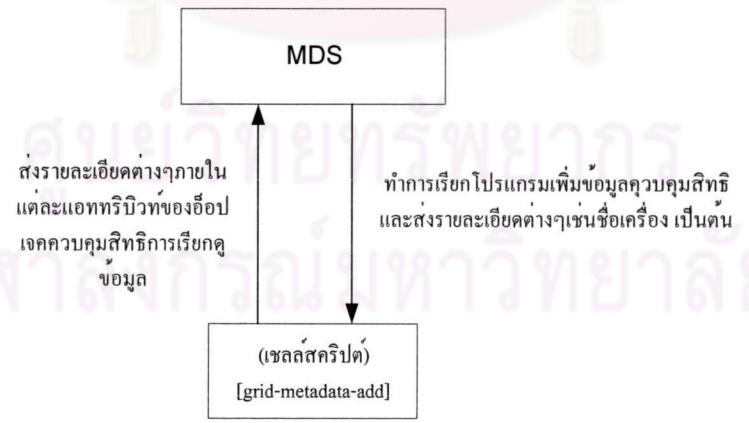
- ผู้ใช้ที่มีบทบาทภายในองค์กรเสมือนตามบทบาทที่เจ้าของข้อมูลเป็นผู้กำหนดเท่านั้นจึงจะสามารถเรียกดูข้อมูลได้ (พิจารณาจาก Mds-Allowed-Role)

- เฉพาะเซิร์ฟเวอร์ที่เจ้าของข้อมูลอนุญาตเท่านั้นจึงจะสามารถแสดงข้อมูลของตนได้ (พิจารณาจาก Mds-Allowed-Server) อนึ่ง แอททริบิวท์นี้สามารถละได้ในกรณีที่เจ้าของไม่สนใจว่าข้อมูลของตนจะถูกคัดลอกส่งต่อไปยังเซิร์ฟเวอร์

- ข้อมูลจริงใดที่จะถูกควบคุมโดยข้อมูลควบคุมสิทธิการเรียกดูข้อมูล (พิจารณาจาก Mds-Effect-On) โดยในอนาคตเราสามารถจะพัฒนาให้ข้อมูลควบคุมสิทธิการเรียกดูใช้ที่กำหนดขึ้นมีผลกับอ็อปเจ็คของข้อมูลจริงได้หลายๆอ็อปเจ็ค ได้โดยการเปลี่ยนชนิดของข้อมูลแอททริบิวท์ "Mds-Effect-On" จากตัวแปรชนิดสตริงเป็นตัวแปรชนิดอาร์เรย์

- สามารถระบุประเภทของข้อมูลจริงที่ถูกควบคุม (พิจารณาจาก Mds-Data-Type) เพื่อในกรณีที่เจ้าของข้อมูลต้องการจะให้ข้อมูลจริงที่ถูกควบคุมสามารถจะถูกนำไปเรียงเป็นหมวดหมู่ตามชื่อประเภทของข้อมูลที่เจ้าของข้อมูลเป็นผู้กำหนดขึ้นมา

2. เขียนโปรแกรมโดยการใช้เชลล์สคริป ให้อยู่ในรูปที่พร้อมจะทำงาน (Executable File) ซึ่งจะทำหน้าที่เพิ่มข้อมูลควบคุมสิทธิการเรียกดูข้อมูลตามความต้องการของเจ้าของข้อมูล โดยส่วนที่ทำหน้าที่เพิ่มข้อมูลภายในโปรแกรมตัวอย่าง (grid-metadata-add) จะมีรูปแบบการทำงานเกี่ยวข้องกับบริการเอ็มดีเอสดังรูปที่ 4.4



รูปที่ 4.4 แสดงรูปแบบการถูกเรียกโดยบริการเอ็มดีเอสของโปรแกรมเพิ่มเติมข้อมูลควบคุมสิทธิ

จากรูปที่ 4.4 แสดงให้เห็นการทำงานของโปรแกรมต้นแบบ (grid-metadata-add) ที่จะคืนค่ารายละเอียดของแต่ละแอททริบิวท์ของข้อมูลควบคุมสิทธิการเรียกดูข้อมูลทุกครั้งที่ถูกเรียกโดยบริการเอ็มดีเอส โดยจะรับแอททริบิวท์จากหน่วยบริการเอ็มดีเอสเกี่ยวกับรายละเอียดต่างๆ ดังเช่น ชื่อเครื่อง, ตำแหน่งของอ็อปเจครากของเครื่องเซิร์ฟเวอร์ที่เป็นต้นกำเนิดข้อมูลในปัจจุบันที่จะถูกส่งมาในลักษณะของอาร์กิวเมนต์โดยบริการเอ็มดีเอส มากำหนดเป็นชื่อเอกเทศ (DN: Distinguished Name) ของข้อมูลควบคุมสิทธิการเรียกดูข้อมูล ก่อนจะเพิ่มรายละเอียดของแอททริบิวท์ตามอ็อปเจคอธิบายชนิดของข้อมูลควบคุม (AddedMetaData) ที่ได้ถูกเพิ่มเข้าไปภายในส่วนที่หนึ่ง โดยส่วนที่ทำหน้าที่เพิ่มรายละเอียดของแต่ละแอททริบิวท์ภายในอ็อปเจคควบคุมสิทธิการเรียกดูข้อมูลของโปรแกรมเพิ่มเติมข้อมูลจะมีรายละเอียดดังต่อไปนี้

```
cat<<EOF
   ${_line_class_dn}dn: Mds-Added-Metadata=Object-MdsCpuSmp, $1
    ${_line_class_oc}objectclass: AddedMetaData
   ${_line_class_av}Mds-Allowed-Role: teacher
   ${_line_class_av}Mds-Allowed-Server: apollo*.cp.eng.chula.ac.th
   ${_line_class_av}Mds-Data-Type: computing
   ${_line_class_av}Mds-Effect-On: MdsCpuSmp
EOF
```

3. แก้ไขไฟล์ควบคุมที่มีหน้าที่จดจำว่าจะต้องไปเรียกโปรแกรมใด, ที่ไหน จึงจะได้ผลลัพท์ตรงกับรายละเอียดของแต่ละอ็อปเจ็คที่ได้ถูกกำหนดไว้ภายในฐานข้อมูลแอลแดป (grid-info-resource-ldif.conf) เพื่อให้ระบบโกลบัสเรียกข้อมูลจากโปรแกรมเพิ่มเติมรายละเอียดของข้อมูลควบคุมสิทธิการเรียกดูข้อมูลที่ได้ถูกพัฒนาขึ้นในส่วนที่สอง (grid-metadata-add) และกำหนดชื่อเอกเทศ (DN: Distinguished Name) ให้แก่ข้อมูลควบคุมที่ถูกเพิ่มเข้าไปภายในต้นไม้ข้อมูลเพื่อให้สามารถนำไปเปรียบเทียบกับชื่อเอกเทศของข้อมูลควบคุมสิทธิที่ได้รับจากโปรแกรมต้นแบบว่าเป็นข้อมูลเดียวกันหรือไม่ โดยข้อมูลที่ถูกเพิ่มเข้าไปภายในไฟล์ควบคุมการเรียกโปรแกรมเพิ่มข้อมูลจะมีรายละเอียดดังต่อไปนี้

```
# generate metadata for MdsCpuSmp object
   dn: Mds-Added-MetaData=Object-MdsCpuSmp, Mds-Host-
hn=apollo11.cp.eng.chula.ac.th, Mds-Vo-name=local, o=grid  // กำหนดชื่อเอกเทศของ
ข้อมูล
   objectclass: GlobusTop
   objectclass: GlobusActiveObject
   objectclass: GlobusActiveSearch
   type: exec   // กำหนดชื่อและที่ตั้งของโปรแกรมต้นแบบที่เป็นต้นตอของข้อมูลควบคุมสิทธิ
    path: /home/g45nsn
    base: grid-metadata-add
   args: -devclassobj -dn Mds-Host-hn=apollo11.cp.eng.chula.ac.th,Mds-Vo-
name=local,o=grid -validto-secs 900 -keepto-secs 900
   cachetime: 30          // กำหนดระยะเวลาในการเก็บรักษาข้อมูล
   timelimit: 20
   sizelimit: 20          // กำหนดขนาดสูงสุดของข้อมูล
```

จากรายละเอียดของไฟล์ควบคุม (grid-info-resource-ldif.conf) ที่แสดงให้เห็นว่า นอกจากจะมีการกำหนดชื่อเอกเทศและระบุโปรแกรมต้นแบบที่เป็นต้นตอของข้อมูล ภายในไฟล์ควบคุมนี้ยังจะสามารถกำหนดรายละเอียดของอาร์กิวเมนท์ของรายละเอียดต่างๆที่จะส่งไปยังโปรแกรมต้นแบบ ระยะเวลาในการเก็บรักษาข้อมูล และขนาดของข้อมูลสูงสุดอีกด้วย

4. แก้ไขไฟล์ควบคุมการส่งข้อมูลระหว่างจีอาร์ไอเอสและจีไอไอเอสของบริการเอ็มดีเอส (grid-info-resource-register.conf) ให้เซิร์ฟเวอร์ส่งข้อมูลควบคุมและข้อมูลจริงที่ถูกเก็บไว้ภายในจีอาร์ไอเอสไปเก็บไว้ยังเซิร์ฟเวอร์อื่นที่ติดตั้งจีไอไอเอสซึ่งทำหน้าที่เก็บรวบรวมข้อมูลต่อไป ซึ่งข้อมูลที่ถูกเพิ่มเติมเข้าไปภายในไฟล์ควบคุมการส่งข้อมูลจะมีรายละเอียดดังต่อไปนี้

```
# add to apollo10
    dn: Mds-Vo-Op-name=register, Mds-Vo-name=apollo10, o=Grid
    regtype: mdsreg2
    reghn: apollo10.cp.eng.chula.ac.th
    regport: 2135
    regperiod: 600
```

type: ldap

hn: apollo11.cp.eng.chula.ac.th

port: 2135

rootdn: Mds-Vo-name=apollo11, o=grid

ttl: 1200

timeout: 20

mode: cachedump

cachettl: 30

จากรายละเอียดของการกำหนดไฟล์ควบคุมให้ส่งข้อมูลภายในจีอาร์ไอเอสไปเก็บไว้ยังเซิร์ฟเวอร์อื่นที่ทำหน้าที่เป็นจีไอไอเอส แสดงให้เห็นว่าเราจำเป็นที่จะต้องระบุชื่อเซิร์ฟเวอร์ที่จะรับทั้งข้อมูลจริงและข้อมูลควบคุมสิทธิการเรียกดูข้อมูล (reghn) , มาตรฐานที่ใช้ในการส่งข้อมูล (type), พอร์ต (port) ที่ใช้, และระยะเวลาที่เกี่ยวข้องกับการเก็บรักษาข้อมูลที่ได้รับจากเซิร์ฟเวอร์ต้นตอนี้ไว้ (ttl, timeout และ cachettl)

5. แก้ไขไฟล์นโยบายที่จะจำว่าเซิร์ฟเวอร์นี้จะยอมรับข้อมูลที่มาจากเซิร์ฟเวอร์ใดบ้าง (grid-info-site-policy.conf) เพื่อให้เซิร์ฟเวอร์ทางฝั่งที่ติดตั้งจีไอไอเอสยอมรับข้อมูลที่ถูกส่งมาจากเซิร์ฟเวอร์ที่เป็นต้นตอของทั้งข้อมูลจริงและข้อมูลควบคุมที่เกี่ยวข้อง ซึ่งจะมีรายละเอียดดังต่อไปนี้

MDS registration policy file

#

example:

objectclass: MdsRegistrationPolicy

policydata: (&(Mds-Service-hn=dc-*.isi.edu)(Mds-Service-port=2135))

#

objectclass: MdsRegistrationPolicy

policydata: (&(Mds-Service-hn=apollo*.cp.eng.chula.ac.th)(Mds-Service-port=2135))

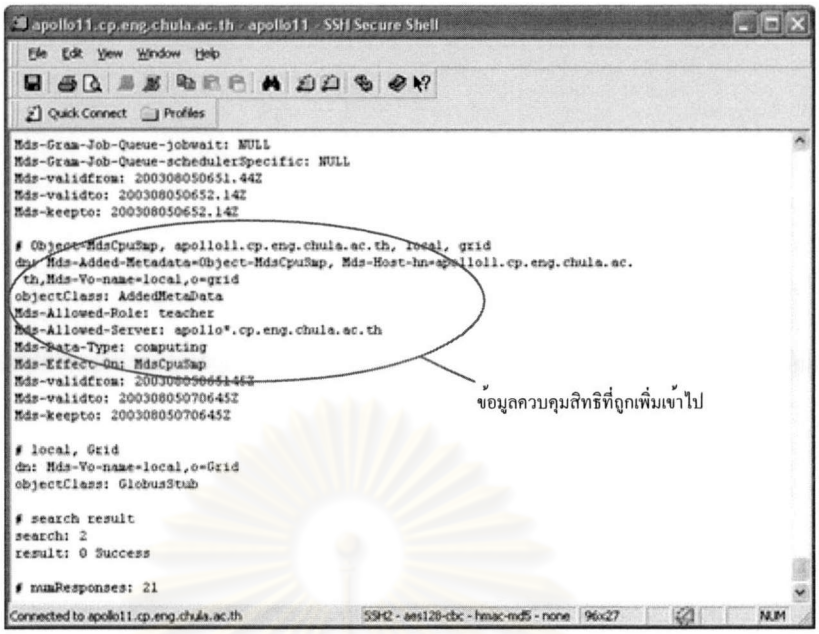
จากรายละเอียดภายในไฟล์ควบคุมนโยบายของเครื่องที่ทำหน้าที่เก็บรวบรวมข้อมูลจะแสดงให้เห็นว่าเครื่องนี้จะรับข้อมูลที่มาเซิร์ฟเวอร์ที่มีชื่อขึ้นต้นด้วย อพอลโล่ห์ (Apollo) และถูกส่งมาทางพอร์ต 2135 เท่านั้น

6. หลังจากที่แก้ไขไฟล์ควบคุมต่างๆที่เกี่ยวข้อง และสร้างโปรแกรมเพิ่มเติมข้อมูลควบคุมตัวอย่างตามขั้นตอนที่ 1-5 แล้ว ก็ทำการเปิดเอ็มดีเอสใหม่อีกครั้งเพื่อให้สิ่งที่ได้แก้ไขไปมีผลขึ้นตามรูปที่ 4.5

รูปที่ 4.5 แสดงการเปิดปิดเอ็มดีเอส

จากรูปที่ 4.5 จะเห็นได้ว่าชุดคำสั่งที่ใช้ในการเปิดปิดบริการของเอ็มดีเอสจำเป็นที่จะต้องเรียกผ่านคอมมานด์ไลน์ (Command Line) โดยผู้ที่มีฐานะเป็นรูท (Root) ของเครื่องเซิร์ฟเวอร์ที่ติดตั้งระบบโกลบัสนี้เท่านั้น

หลังจากที่ได้ทำการแก้ไขตามขั้นตอนหนึ่งถึงหก เมื่อเรียกดูข้อมูลภายในเอ็มดีเอส ผลลัพท์ที่ได้จะมีลักษณะดังรูปที่ 4.6

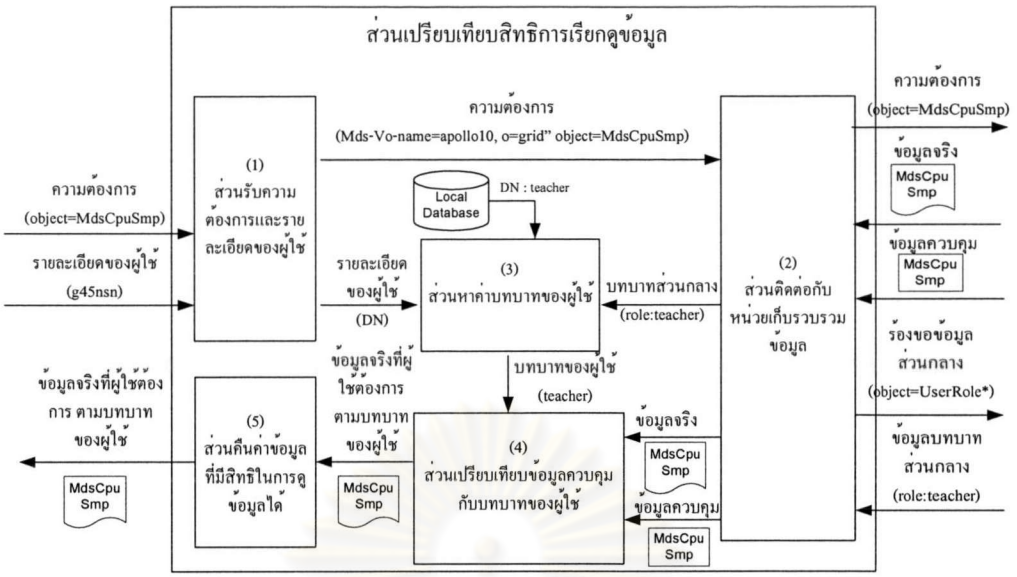


รูปที่ 4.6 แสดงผลลัพท์หลังจากทำการเรียกดูข้อมูล

จากรูปที่ 4.6 แสดงให้เห็นถึงข้อมูลควบคุมสิทธิที่ถูกเพิ่มเข้ามาภายในต้นไม้ข้อมูลตามโครงสร้างของข้อมูลที่ได้แสดงไว้ในตอนแรก

4.2 ส่วนเปรียบเทียบสิทธิการเรียกดูข้อมูลส่วนกลาง

เราได้ออกแบบส่วนเปรียบเทียบสิทธิการเรียกดูข้อมูลส่วนกลางให้อยู่ในรูปของชุดคำสั่งสำเร็จรูปด้วยภาษาจาวาที่จะถูกเรียกโดยส่วนติดต่อกับผู้ใช้อีกทีหนึ่ง ซึ่งเหตุผลที่ออกแบบส่วนเปรียบเทียบสิทธิในลักษณะนี้ อันเนื่องมาจากเราต้องการให้ส่วนเปรียบเทียบการเรียกดูข้อมูลทำงานในลักษณะที่ไม่ขึ้นกับรูปแบบของส่วนติดต่อกับผู้ใช้ ซึ่งจะทำให้ส่วนติดต่อกับผู้ใช้แต่ละแบบจะยังสามารถเรียกใช้ฟังก์ชั่นของส่วนเปรียบเทียบสิทธิอันเดียวกัน โดยรายละเอียดของแต่ละส่วนประกอบย่อยๆจะถูกพัฒนา ดังรูปที่ 4.7 และตารางที่ 4.2



รูปที่ 4.7 แสดงข้อมูลที่ถูกส่งต่อระหว่างแต่ละองค์ประกอบย่อยภายในส่วนเปรียบเทียบสิทธิ

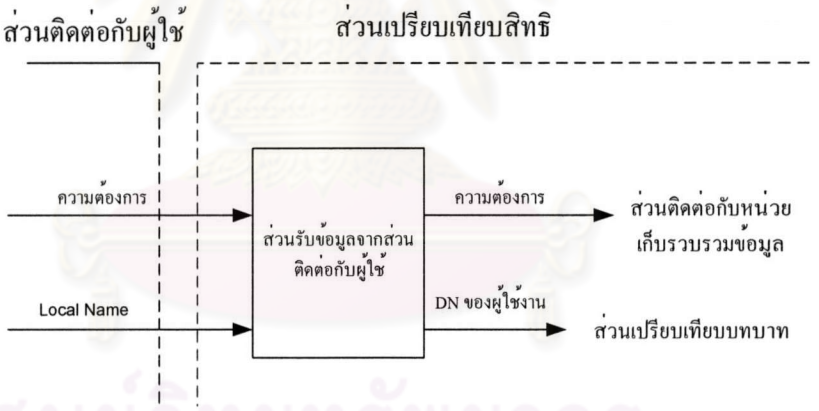
ชื่อองค์ประกอบ	หน้าที่	รายละเอียดในการพัฒนา
1. ส่วนรับความต้องการและรายละเอียดของผู้ใช้	ทำหน้าที่เปลี่ยนความต้องการและชื่อของผู้ใช้ให้อยู่ในรูปแบบที่เหมาะสม	เป็นฟังก์ชั่นภายในคลาส roleMainComparator
2. ส่วนติดต่อกับหน่วยเก็บรวบรวมข้อมูล	ทำหน้าที่ติดต่อกับบริการ เอ็มดีเอส	อยู่ในรูปของคลาส ApolloMDS
3. ส่วนหาค่าบทบาทของผู้ใช้	ทำหน้าที่พิจารณาหาชื่อบทบาทจากชื่อเอกเทศของผู้ใช้	เป็นฟังก์ชั่นภายในคลาส roleMainComparator
4. ส่วนเปรียบเทียบข้อมูลควบคุมกับบทบาทของผู้ใช้	ทำการเปรียบเทียบว่าข้อมูลใดบ้างที่อนุญาตให้บทบาทของผู้ใช้ในขณะนั้นสามารถเรียกดูได้	เป็นฟังก์ชั่นภายในคลาส roleMainComparator
5. ส่วนคืนค่าข้อมูลที่มีสิทธิในการเรียกดู	ทำการแก้ไขข้อมูลที่ผู้ใช้สามารถเรียกดูได้ให้อยู่ในรูปแบบที่เหมาะสม	เป็นฟังก์ชั่นภายในคลาส roleMainComparator

ตารางที่ 4.2 แสดงรายละเอียดในการพัฒนาแต่ละองค์ประกอบภายในส่วนเปรียบเทียบสิทธิ

จากรูปที่ 4.7 และตารางที่ 4.2 แสดงองค์ประกอบและชนิดของข้อมูลที่ถูกแลกเปลี่ยนภายในส่วนเปรียบเทียบสิทธิการเรียกดูข้อมูล ซึ่งแต่ละส่วนย่อยๆจะมีลักษณะการทำงานดังต่อไปนี้

1. ส่วนรับความต้องการและรายละเอียดของผู้ใช้จากส่วนติดต่อกับผู้ใช้ (ฟังก์ชั่น receiveFromUI)

เป็นส่วนรับเอาความต้องการของผู้ใช้ซึ่งจะอยู่ในรูปของชื่ออ็อปเจ็คที่รวบรวมแอททริบิวท์ที่ตรงกับความต้องการของผู้ใช้ เช่น อาจจะระบุ Objectclass=MdsCpuSmp เพื่อหาข้อมูลเกี่ยวกับสภาพการทำงานของซีพียูเป็นต้น หรืออาจจะใช้สัญลักษณ์ไวล์คาร์ด (*) เพื่อความสะดวกในการสืบค้นข้อมูลเช่น Objectclass=* ที่แสดงถึงความต้องการที่จะเอาข้อมูลของทุกอ็อปเจคภายในต้นไม้ข้อมูล โดยรูปแบบการสืบค้นข้อมูลดังกล่าวจะเป็นรูปแบบการประกาศความต้องการเดิมของเอ็มดีเอสภายในระบบโกลบัส โดยในส่วนนี้จะเพิ่มเติมหรือแก้ไขลักษณะการประกาศความต้องการของผู้ใช้เพื่อให้ส่วนติดต่อกับหน่วยเก็บรวบรวมข้อมูลสามารถดึงเอาทั้งข้อมูลจริงและข้อมูลควบคุมที่เกี่ยวข้องออกมาได้ ก่อนที่จะส่งข้อมูลดังกล่าวไปยังส่วนติดต่อกับหน่วยเก็บรวบรวมข้อมูลต่อไปดังรูปที่ 4.8



รูปที่4.8 แสดงการแลกเปลี่ยนข้อมูลกับองค์ประกอบอื่นๆภายในส่วนเปรียบเทียบสิทธิ

จากรูปที่ 4.8 แสดงข้อมูลที่เข้ามาและถูกส่งออกไปจากส่วนรับข้อมูลจากส่วนติดต่อกับผู้ใช้ โดยจะรับเอาชื่อของผู้ใช้งานภายในเซิร์ฟเวอร์นั้น (Local Name) มาเปลี่ยนเป็นชื่อเอกเทศ (DN: Distuighed Name) ที่ผู้ใช้แต่ละคนจะมีชื่อเอกเทศนี้เพียงชื่อเดียวเท่านั้นภายในองค์กรเสมือน โดยการเรียกใช้ฟังก์ชั่นของจาวาค็อกคิต (Java CoG kits) โดยมีขั้นตอนดังต่อไปนี้

- เรียกไฟล์ gridmapfile ที่เก็บชื่อเอกเทศและชื่อของผู้ใช้บนเซิร์ฟเวอร์นั้นๆ (Local Name)
- ใช้คำสั่งของจาวาคอกคิตในการเรียกหาชื่อเอกเทศของผู้ใช้ (จากคำสั่ง getGlobusID)

โดยภายในฟังก์ชั่นส่วนรับข้อมูลจากส่วนติดต่อกับผู้ใช้งาน (receiveFromUI) จะมีรายละเอียดดังต่อไปนี้

```
public void receiveFromUI () {

    // find the DN for user local id
    System.out.print("We know your globusID from using CoGkit: ");
    String globusId;
     GridMap newGrid = new GridMap();        // เรียกฟังก์ชั่นของจาวาคอกคิต
    try {
    newGrid.load("/etc/grid-security/grid-mapfile");
    }
    catch (Exception e) {
    System.out.println("can't load na");
    }

    globusId = newGrid.getGlobusID(this.userLocal);
    System.out.println(""+globusId+"");
    this.userDN = globusId;

    // change from default to object=*
    if (this.wantedInformation.equals("default")) {
    this.wantedInformation = "*";
    }
    }
```

เหตุผลที่จำเป็นต้องเปรียบเทียบหาชื่อเอกเทศของผู้ใช้ในส่วนนี้ ก็เพื่อป้องกันความสับสนที่เกิดจากผู้ใช้แต่ละคนมีชื่อ (Local Name) แตกต่างกันตามเซิร์ฟเวอร์ที่ผู้ใช้คนนั้นไปติดต่อขอเรียกดูข้อมูล โดยเมื่อได้ชื่อเอกเทศของผู้ใช้งานแล้วก็จะส่งข้อมูลดังกล่าวไปยังส่วนหาค่าบทบาทของผู้ใช้ต่อไป

2. ส่วนติดต่อกับหน่วยเก็บรวบรวมข้อมูล

ในส่วนนี้จะนำเอาข้อมูลที่ได้รับจากส่วนที่หนึ่งมาเรียกเอาข้อมูลจริงที่ถูกประกาศโดยผู้ใช้งานและข้อมูลควบคุมที่เกี่ยวข้องโดยจะอาศัยฟังก์ชั่นของจาวาคอกคิตที่จะมีฟังก์ชั่นที่จะเรียกดูข้อมูลที่ถูกเก็บอยู่ภายในฐานข้อมูลแอลแด็ปได้ โดยจะพัฒนาให้อยู่ในรูปของคลาสแยกต่างหาก (ApolloMds.java) ซึ่งจะมีรายละเอียดดังต่อไปนี้

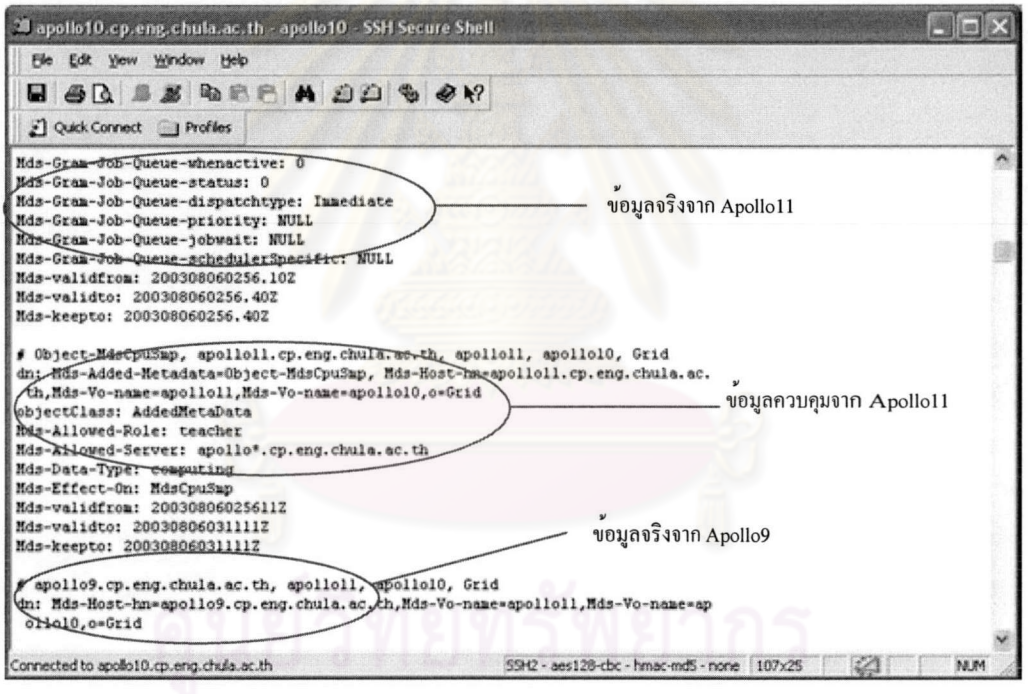
```
public String getResourceInfo(String filter,String [] attr) {
    this.filter = filter;
    Hashtable searchResults = new Hashtable();
    try {
    mds.connect();
    int searchAttributeCount = attr.length;
    String [] searchAttributes = new String[1];
    searchAttributes[0] = attr[0];
                    // เรียกดูข้อมูลภายในบริการเอ็มดีเอสตามอาร์กิวเมนท์ที่เรากำหนด
    searchResults = mds.search(baseDN,filter,searchAttributes,MDS.SUBTREE_SCOPE
);
    mds.disconnect();
    } catch ( org.globus.mds.MDSException mdse ) {
    System.out.println(   "Error from MDS: "
    + mdse.getMessage() );
    }

    StringBuffer resultsBuffer = new StringBuffer();
    Enumeration resultsEnum = searchResults.keys();
```

```
if ( resultsEnum.hasMoreElements() ) {
String dn = (String) resultsEnum.nextElement();
resultsBuffer.append( searchResults.get( dn ).toString() );
}

return resultsBuffer.toString();
}
```

โดยคลาสย่อยที่ทำหน้าที่สืบค้นข้อมูล (ApolloMds.java) จะแสดงข้อมูลทั้งหมดภายในบริการเอ็มดีเอสตามอาร์กิวเมนท์ที่ได้รับ ซึ่งถ้าเราเรียกใช้คลาสนี้โดยตรงผลลัพท์ที่ได้จะมีลักษณะดังรูปที่ 4.9



รูปที่4.9 แสดงผลลัพธ์ที่ได้จากการเรียกใช้ฟังก์ชั่นของหน่วยสืบค้นข้อมูล

จากรูปที่4.9 จะแสดงให้เห็นว่าผลลัพธ์ที่ได้จากการเรียกคลาสย่อย (ApolloMds.java) จะแสดงข้อมูลภายในบริการเอ็มดีเอสโดยใช้ฟังก์ชั่นของ จาวาคอคคิตที่จะแสดงข้อมูลทั้งหมดตามที่ผู้ใช้ต้องการ โดยที่เจ้าของทรัพยากรที่เป็นต้นตอของข้อมูลจะไม่สามารถควบคุมให้ผู้ใช้งานเห็นข้อมูลแค่บางส่วนได้ หลังจากนั้นส่วนติดต่อกับหน่วยเก็บรวบรวมข้อมูลจะ

ส่งข้อมูลทั้งหมดไปยังส่วนเปรียบเทียบข้อมูลควบคุมกับบทบาทของผู้ใช้เพื่อคัดเลือกเฉพาะข้อมูลที่ผู้ใช้สามารถเรียกดูได้ตามข้อมูลควบคุมสิทธิและข้อมูลอธิบายบทบาทของผู้ใช้ต่อไป

3. ส่วนหาค่าบทบาทของผู้ใช้

ส่วนนี้จะเปรียบเทียบหาบทบาทของผู้ใช้ โดยจะใช้ข้อมูลชื่อเอกเทศจากส่วนที่หนึ่ง, ข้อมูลบทบาทส่วนกลางที่ได้รับจากเซิร์ฟเวอร์กระจายบทบาทส่วนกลาง และข้อมูลอธิบายบทบาทของผู้ใช้งานเฉพาะภายในองค์กรจริงนั้นๆที่ถูกเก็บรวมเป็นไฟล์อธิบายบทบาทของผู้ใช้แต่ละคนซึ่งจะมีรายละเอียดดังต่อไปนี้

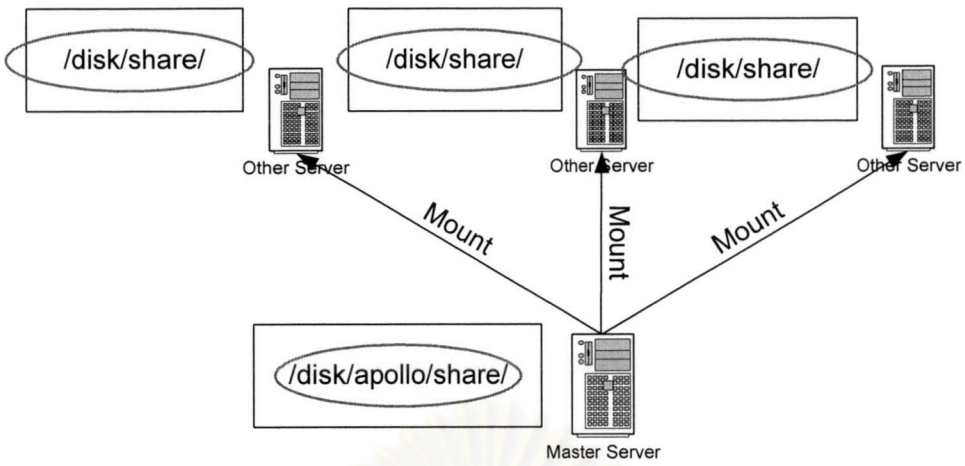
/O=Grid/O=Globus/OU=zeus.cp.eng.chula.ac.th/CN=Natthakrit:student

/O=Grid/O=Globus/OU=cp.eng.chula.ac.th/CN=Natawut:teacher

/O=Grid/O=Chula/OU=cp.eng.chula.ac.th/CN=Rittichai Boonplang:student

/O=Grid/O=Chula/OU=cp.eng.chula.ac.th/CN=Chulayuth Asawaroengchai:student

ภายในไฟล์อธิบายบทบาทของผู้ใช้แต่ละคน จะประกอบไปด้วยชื่อเอกเทศและบทบาทของผู้ใช้แต่ละคนภายในแต่ละองค์กรจริงนั้นๆ โดยผู้ติดตั้งระบบควบคุมการเข้าถึงข้อมูลต้องให้ทุกเซิร์ฟเวอร์ที่ติดตั้งระบบควบคุมการเข้าถึงข้อมูลเชิงบทบาทสามารถเรียกดูข้อมูลอธิบายบทบาทของผู้ใช้นี้ได้ เนื่องจากการทำงานของส่วนหาค่าบทบาทของผู้ใช้เป็นเพียงการเรียกดูข้อมูลเฉพาะภายในขอบเขตขององค์กรจริงเท่านั้น จึงไม่จำเป็นที่จะทำการแลกเปลี่ยนข้อมูลโดยใช้บริการเอ็มดีเอสผ่านทางมาตรฐานแอลแดปโดยอาจจะพัฒนาโดยใช้รูปแบบการแลกเปลี่ยนข้อมูลที่มีอยู่เดิมภายในแต่ละองค์กร เช่นอยู่ในรูปแบบของฐานข้อมูลหรือการเมาท์ไดเร็คทอรีเป็นต้น และเนื่องจากทุกเครื่องที่นำมาทดลองติดตั้งระบบโกลบัสนั้นใช้ระบบปฏิบัติการลินุกซ์หมดทุกเครื่อง เพราะฉะนั้นการพัฒนาส่วนหาค่าบทบาทของผู้ใช้จึงเลือกที่จะใช้การเมาท์ไดเร็คทอรี เพราะจะทำได้ง่ายและไม่จำเป็นจะต้องไปเพิ่มเติมชนิดของข้อมูลที่เกี่ยวข้องกับบทบาทของแต่ละคนภายในฐานข้อมูลของบริการเอ็มดีเอสในปัจจุบัน ซึ่งจะทำให้ข้อมูลที่เก็บรวบรวมรายละเอียดของชื่อเอกเทศกับบทบาทของผู้ใช้ จะถูกเก็บอยู่ภายในไดเร็คทอรีที่ทุกๆเซิร์ฟเวอร์ได้เมาท์ (Mount) ไว้ เพราะฉะนั้นจึงทำให้ทุกเครื่องภายในองค์กรจริงสามารถมองเห็นบทบาทของผู้ใช้แต่ละคน ซึ่งโครงสร้างการกระจายข้อมูลจะมีรายละเอียดดังรูปที่ 4.10

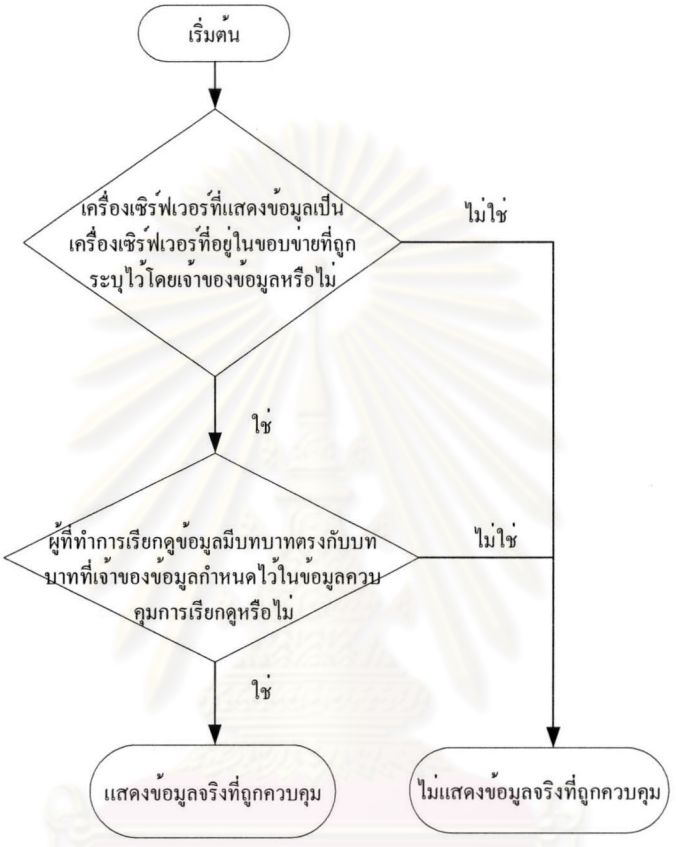


รูปที่4.10 แสดงลักษณะการวางไฟล์อธิบายของบทบาทภายในขอบเขตขององค์กรจริง

จากรูปที่ 4.10 แสดงรูปแบบการกระจายไฟล์อธิบายบทบาทของผู้ใช้แต่ละคนโดยจะเมาท์ (Mount) ไดเร็คทอรี่ (Directory) ของเครื่องต้นตอที่เก็บข้อมูลนี้ไว้ไปยังเครื่องเซิร์ฟเวอร์ย่อยๆ ภายในองค์กรจริงที่ติดตั้งระบบควบคุมการเข้าถึงข้อมูลเพื่อระบบควบคุมสามารถนำเอาข้อมูลดังกล่าวไปพิจารณาหาบทบาทของผู้ใช้แต่ละคนจากชื่อเอกเทศได้ โดยหลังจากที่ได้บทบาทของชื่อเอกเทศที่ถูกส่งมาจากส่วนที่หนึ่ง ส่วนหาค่าบทบาทของผู้ใช้ก็จะส่งชื่อของบทบาทดังกล่าวไปยังส่วนเปรียบเทียบข้อมูลควบคุมกับบทบาทของผู้ใช้ต่อไป

4.ส่วนเปรียบเทียบข้อมูลควบคุมกับบทบาทของผู้ใช้

ในส่วนนี้จะเปรียบเทียบสิทธิการเรียกดูข้อมูลตามบทบาทของผู้ใช้ โดยการดึงเอาข้อมูลควบคุมที่มีอยู่ในกรณีที่ข้อมูลจริงนั้นถูกควบคุมโดยเจ้าของข้อมูล ซึ่งขั้นตอนการพิจารณาจะมีลำดับดังรูปที่ 4.11



รูปที่4.11 แสดงการทำงานของหน่วยประมวลผล

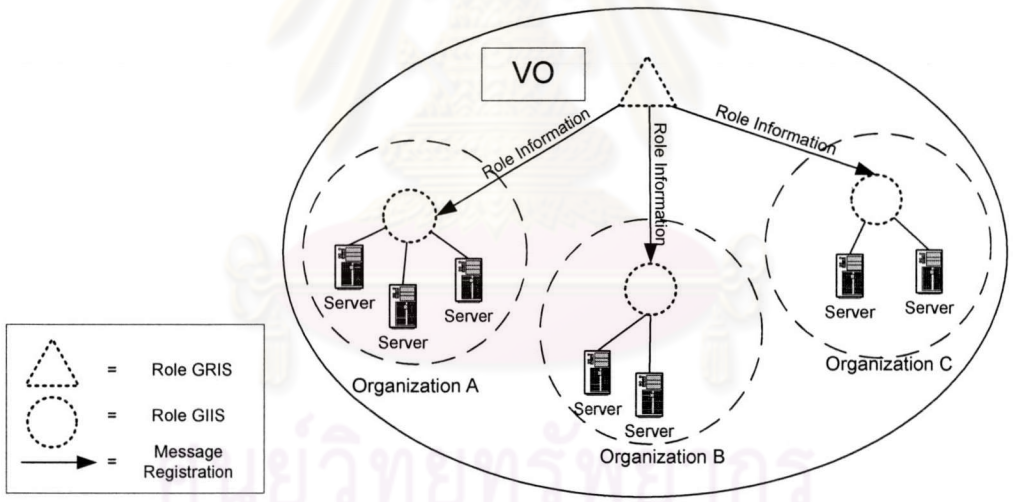
จากรูปที่ 4.11 แสดงลำดับการเปรียบเทียบข้อมูลสิทธิกับข้อมูลบทบาทของผู้ใช้ ซึ่งจะเปรียบเทียบข้อมูลควบคุมประเภทควบคุมเครื่องก่อน ถ้าเครื่องเซิร์ฟเวอร์ที่ผู้ใช้งานนั้นอยู่ในกลุ่มของเซิร์ฟเวอร์ที่เจ้าของข้อมูลอนุญาตให้ดูได้ ก็จะมาพิจารณาทางด้านบทบาทของผู้ใช้ที่ได้รับจากส่วนหาค่าบทบาทของผู้ใช้ว่าตรงกับบทบาทที่อนุญาตให้ดูได้หรือไม่ ถ้าอนุญาตก็จะส่งข้อมูลจริงที่ถูกควบคุมนั้นไปยังส่วนคืนค่าต่อไป

5. ส่วนคืนค่าข้อมูลที่มีสิทธิในการดูข้อมูลได้

หลังจากระบบทำการเปรียบเทียบว่าผู้ใช้ที่มาเรียกดูข้อมูลนั้นมีสิทธิของบทบาทตรงกับข้อมูลอธิบายสิทธิการเรียกดูข้อมูลที่ถูกส่งมาพร้อมกับข้อมูลจริงโดยเจ้าของข้อมูล ระบบจะคืนค่าข้อมูลจริงที่ผู้ใช้สามารถดูข้อมูลได้ กลับไปยังส่วนติดต่อกับผู้ใช้โดยจะให้อยู่ในรูปของตัวแปรชนิดเวคเตอร์เนื่องจากข้อมูลมีจำนวนมากและมีรูปแบบที่ซับซ้อนเกินกว่าจะส่งในรูปแบบของอาร์เรย์

4.3 ส่วนควบคุมบทบาทส่วนกลาง

ในส่วนนี้จะใช้โครงสร้างการแลกเปลี่ยนข้อมูลของมาตรฐานแอลแด็ปซึ่งเป็นโครงสร้างการแลกเปลี่ยนข้อมูลเดิมของของระบบโกลบัส 2.0 เพราะข้อมูลบทบาทส่วนกลางจำเป็นที่จะต้องถูกส่งข้ามเครือข่ายระหว่างแต่ละองค์กรจริงจึงจำเป็นจะต้องใช้มาตรฐานกลางที่ทุกเซิร์ฟเวอร์ที่ติดตั้งระบบโกลบัสสามารถรับข้อมูลบทบาทจากเซิร์ฟเวอร์ส่วนกลางได้ โดยเราจะกระจายข้อมูลควบคุมสิทธิโดยการสร้าง จีไอไอเอสเฉพาะกิจขึ้น ซึ่งจะเก็บเฉพาะข้อมูลบทบาทส่วนกลางไว้บนแต่ละเซิร์ฟเวอร์สำรองภายในแต่ละองค์กรจริง เพื่อแบ่งเบาภาระจากเซิร์ฟเวอร์กลางและลดโอกาสที่ระบบควบคุมจะไม่สามารถทำงานได้อันเนื่องมาจากความผิดพลาดบนเครือข่าย ดังรูปที่4.12



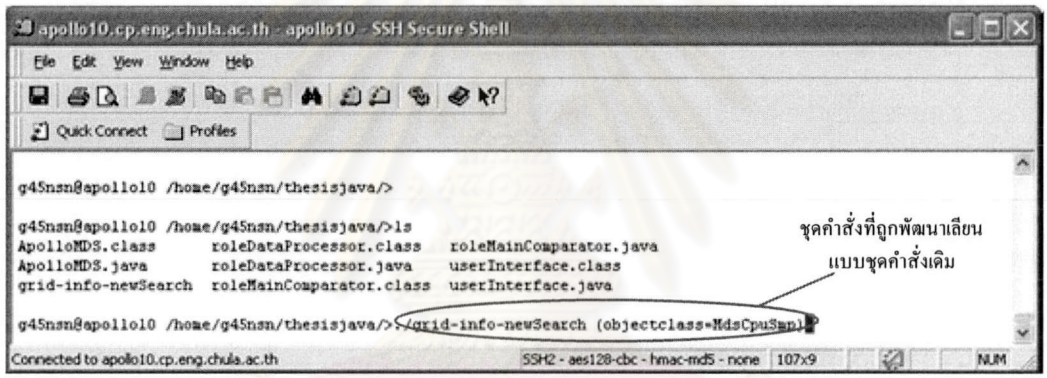
รูปที่ 4.12 แสดงโครงสร้างการกระจายข้อมูลจากเซิร์ฟเวอร์หลักโดยใช้โครงสร้างการเก็บข้อมูลของระบบโกลบัส

จากรูปที่ 4.12 แสดงการกำหนดเครื่องเซิร์ฟเวอร์สำรองที่ทำหน้าที่เป็นจีไอไอเอสเฉพาะกิจเพื่อจะรวบรวมเฉพาะข้อมูลบทบาทที่ได้รับจากเซิร์ฟเวอร์ส่วนกลาง ก่อนที่จะกระจายข้อมูลดังกล่าวไปยังแต่ละเซิร์ฟเวอร์ที่ติดตั้งระบบควบคุมการเข้าถึงข้อมูลภายในขอบเขตขององค์กรจริง

ที่เซิร์ฟเวอร์กระจายบทบาทสำรองนี้ตั้งอยู่ เนื่องจากขั้นตอนในการเพิ่มข้อมูลควบคุมบทบาทส่วนกลางจะทำเช่นเดียวกับการเพิ่มข้อมูลควบคุมสิทธิจึงไม่อธิบายรายละเอียดไว้ในส่วนนี้

4.4 ส่วนติดต่อกับผู้ใช้

ในส่วนนี้เราจะพัฒนาส่วนติดต่อกับผู้ใช้ให้อยู่ในรูปของคำสั่งที่ถูกพัฒนาด้วยภาษาจาวาที่ทำงานเลียนแบบชุดคำสั่งเรียกดูข้อมูลของระบบโกลบัสเดิมที่จากเดิมจะไม่สามารถควบคุมการเรียกดูข้อมูล โดยชุดคำสั่งใหม่ของระบบควบคุมการเข้าถึงข้อมูลจะติดต่อกับตัวแปรแวดล้อม (Environment Variable) เพื่อจะเอาชื่อของผู้ใช้งานเฉพาะเครื่อง (Local Account) แล้วก็จะส่งค่าดังกล่าวไปยังส่วนประมวลผลพร้อมกับความต้องการของผู้ใช้ซึ่งจะอยู่ในรูปของชื่อกิ่งก้านของต้นไม้ข้อมูลที่มีชื่อตรงกับความต้องการของผู้ใช้โดยการเรียกฟังก์ชั่นของส่วนเปรียบเทียบสิทธิการเรียกดูข้อมูล เพื่อรอรับข้อมูลที่ผู้ใช้คนนั้นมีสิทธิในการเรียกดูข้อมูลได้เท่านั้นดังรูปที่ 4.13 และรูปที่ 4.14



รูปที่ 4.13 แสดงลักษณะการเรียกใช้งานส่วนติดต่อกับผู้ใช้งาน

จากรูปที่ 4.13 แสดงรูปแบบการเรียกใช้ระบบควบคุมการเข้าถึงข้อมูลเชิงบทบาทผ่านทางชุดคำสั่งขอสืบค้นข้อมูล (grid-info-newsearch) ที่ถูกพัฒนาจากเชลล์สคริปต์ซึ่งคล้ายกับชุดคำสั่งในการเรียกดูข้อมูลเดิมของบริการเอ็มดีเอสโดยมีรายละเอียดของการใช้ชุดคำสั่งที่ถูกเพิ่มเข้ามาดังต่อไปนี้

ชื่อ

grid-info-newSearch

รูปแบบการใช้งาน

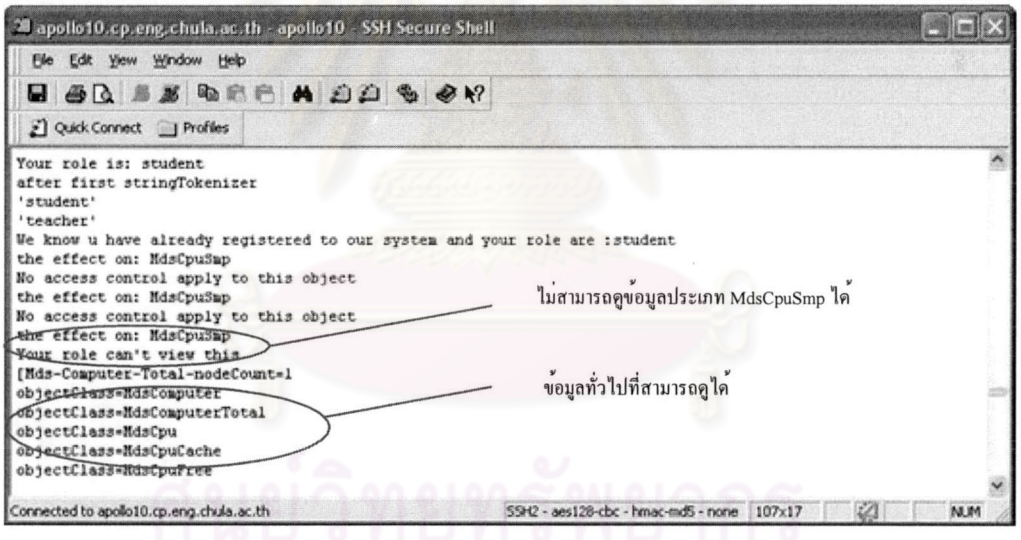
grid-info-newSearch [objectclass ..]

ตัวอย่างการใช้งาน

grid-info-newSearch (objectclass=MdsCpuSmp)

grid-info-newSearch (objectclass=*)

ชุดคำสั่งขอสืบค้นข้อมูลดังกล่าวเป็นเพียงส่วนติดต่อกับผู้ใช้ (User Interface) ที่จะมาเรียกใช้ฟังก์ชั่นของส่วนควบคุมการเข้าถึงข้อมูลอีกทีหนึ่ง จึงทำให้เราสามารถพัฒนาส่วนติดต่อกับผู้ใช้ให้อยู่ในรูปอื่นนอกจากอยู่ในรูปของชุดคำสั่งเชลล์สคริปต์ได้ในอนาคต



รูปที่4.14 แสดงผลลัพท์เมื่อทำการเรียกชุดคำสั่งใหม่ของระบบควบคุมการเข้าถึงข้อมูล

จากรูปที่ 4.14 จะเห็นได้ว่าผู้ใช้ไม่สามารถดูข้อมูลทั้งหมดตามที่ผู้ใช้ต้องการได้ อันเนื่องมาจากข้อจำกัดของบทบาทของผู้ใช้ โดยจะมีการแสดงประโยคอธิบายว่าผู้ใช้ไม่สามารถเรียกดูข้อมูลได้แทนที่ตำแหน่งของข้อมูลเดิม