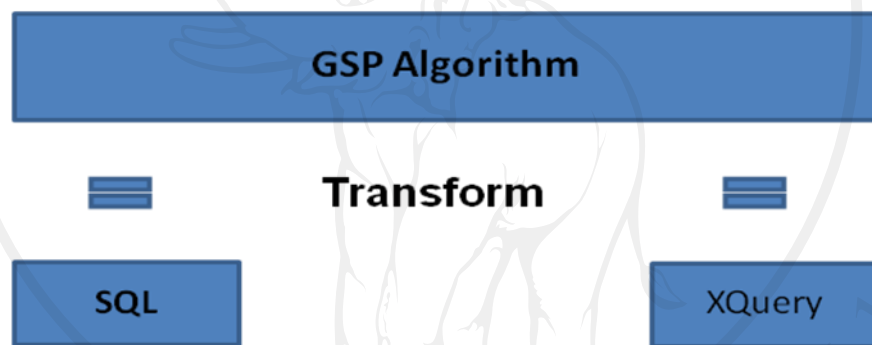


บทที่ 3

แนวคิดในการแก้ไขปัญหและขั้นตอนการพัฒนา

ในบทนี้จะกล่าวถึงแนวคิดที่ใช้ในการแก้ไขปัญหในการอิมพลีเมนต์ ขั้นตอนวิธี GSP บนฐานข้อมูลเชิงเวลา โดยการแปลงข้อมูลเชิงเวลาในรูปแบบของฐานข้อมูลเชิงสัมพันธ์ ให้อยู่ในรูปแบบของฐานข้อมูลเอ็กซ์เอ็มแอล และสอบถามข้อมูลดังกล่าวผ่านภาษาสำหรับการสอบถามเอ็กซ์คิวรี โดยได้นำเอาทฤษฎีต่างๆในบทที่ 2 มาประยุกต์ใช้ ดังที่แสดงในรูปภาพที่ 3.1



รูปที่ 3.1 แสดงแนวคิดในการอิมพลีเมนต์ขั้นตอนวิธี GSP บนฐานข้อมูลเชิงเวลา

โดยนำข้อมูลที่อยู่ในรูปแบบ Temporal Database มาแปลงให้อยู่ในรูปแบบของ XML และจัดเก็บลงบนฐานข้อมูล XML ของ IBM DB2 จากนั้นทำการอิมพลีเมนต์อัลกอริทึม GSP มาอิมพลีเมนต์ 2 ลักษณะ คือ

1. ใช้ภาษา SQL ตามรูปแบบของฐานข้อมูลเชิงสัมพันธ์ทั่วไป
2. อิมพลีเมนต์ GSP ผ่าน XQuery สำหรับโครงสร้างข้อมูลที่ถูกจัดเก็บในรูปแบบของฐานข้อมูลเอ็กซ์เอ็มแอล

ซึ่งผลลัพธ์ของข้อมูลที่ได้จากการหา รูปแบบลำดับของข้อมูล ทั้งสองจะต้องได้เหมือนกัน ถึงจะเป็นข้อมูลที่ต้องการ ดังที่แสดงให้เห็นในรูปภาพที่ 3.2

Customer ID	Transaction Time	Items Bought
1	2011-06-25	30
1	2011-06-30	90
2	2011-06-31	10,20
2	2011-06-11	30
2	2011-06-16	20
2	2011-06-18	30
2	2011-06-20	40,60,70
3	2011-06-25	30,50,70
4	2011-06-25	30
4	2011-06-30	40,70
4	2011-07-25	90
5	2011-06-25	90

```

<Bought_Item start="2011-01-01" end="2011-07-31">
  <customer start="2011-06-10" end="2011-06-20">
    <id start="2011-06-10" end="2011-06-20">2</id>
    <item_bought start="2011-06-10" end="2011-06-10">10,20</item_bought>
    <item_bought start="2011-06-15" end="2011-06-18">30</item_bought>
    <item_bought start="2011-06-16" end="2011-06-16">20</item_bought>
    <item_bought start="2011-06-20" end="2011-06-20">40,60,70</item_bought>
  </customer>
  .
  .
  .
</Bought_Item>

```

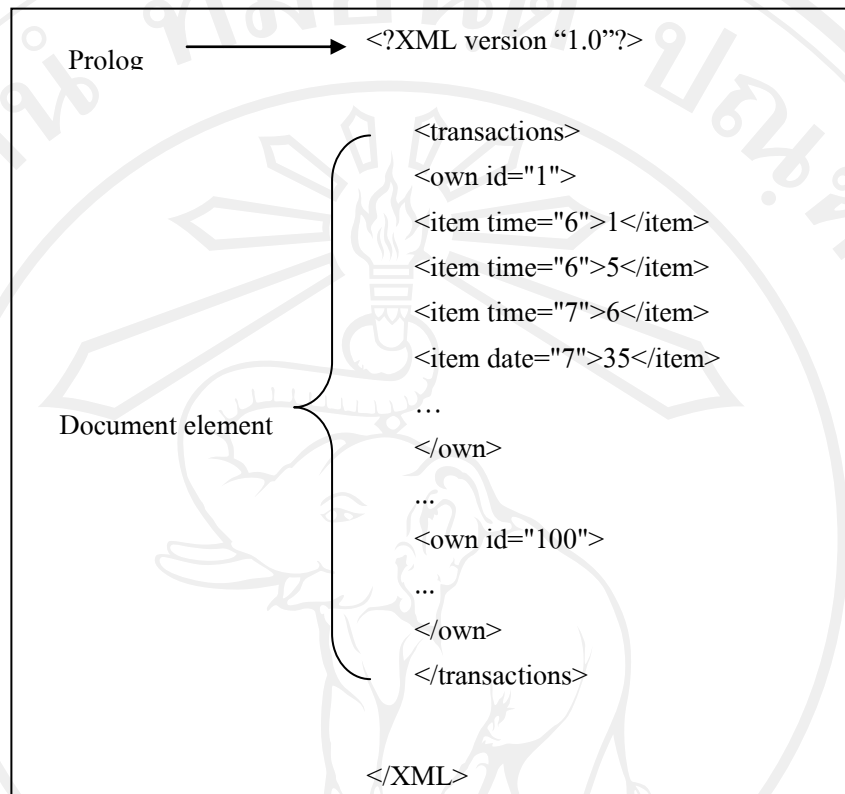


รูปที่ 3.2 แสดงผลลัพธ์จากการสอบถามข้อมูลผ่านภาษาสำหรับการสอบถามเอ็กซ์คิวรี และเอสคิวแอล

โดยเปรียบเทียบประสิทธิภาพ จากเวลาที่ใช้ในการประมวลผล ทั้งนี้ในการทดลอง จะมีการแปรผันตัวแปรต้นต่างๆ เช่น ค่า min gap ค่า max gap ค่า sliding window ค่า support หรือแม้แต่ลักษณะการอิมพลีเมนต์บนฐานข้อมูล นอกจากจะพิจารณาผลลัพธ์ในแง่เวลาในการประมวลผลแล้ว ยังจะพิจารณา Query Plan ของทั้งสองแนวทางด้วย ว่าทำงานอย่างไร รวมไปถึงการปรับแต่งฐานข้อมูลเชิงกายภาพด้วย

จากนั้นจะนำเวลาที่ใช้ในการประมวลผลที่ได้มาเปรียบเทียบประสิทธิภาพในการหาความสัมพันธ์ของข้อมูล บนฐานข้อมูลแต่ละแบบ

3.1 การแปลงข้อมูลให้อยู่ในรูปแบบโครงสร้างเอ็กซ์เอ็มแอล



รูปที่ 3.3 โครงสร้างเอ็กซ์เอ็มแอล

โครงสร้างของเอกสารเอ็กซ์เอ็มแอล ดังที่แสดงให้เห็นในรูปที่ 3.3 ประกอบด้วย

1. ส่วนแรก (Prolog) ในส่วนนี้ จะประกอบด้วย XML declaration เป็นการประกาศให้รู้ว่าเอกสารดังกล่าวคือ XML และเป็นการประกาศเวอร์ชันของ XML การใส่ค่า XML declaration จะประกาศหรือไม่ประกาศก็ได้ แต่ควรมีข้อกำหนดนี้ในเอกสาร โดยบรรทัดว่างในเอกสารเอ็กซ์เอ็มแอลตัวประมวลผลของเอ็กซ์เอ็มแอล (XML Processor) จะข้ามและไม่นำบรรทัดว่างเหล่านั้นมาประมวลผล

2. ส่วนที่สอง (Document element) ในส่วนที่สองเรียกว่า Document element หรือ Root element ซึ่งสามารถบรรจุอีลิเมนต์ (Element) เพื่อเติม ในเอกสาร XML ได้ ในเอกสาร XML นั้น Element จะแสดงลักษณะโครงสร้างของเอกสารและแสดงส่วนประกอบของเนื้อหาเอกสารอยู่ภายใน จากรูปที่ 3.3 เป็นตัวอย่างของการเก็บข้อมูลรายการเปลี่ยนแปลงของการซื้อสินค้า Transaction Element ซึ่งประกอบด้วย OWN และ ITEM

จากรูปที่ 3.3 แสดงให้เห็นว่าสินค้าหลายๆ ชิ้นที่อยู่ในรายการซื้อเดียวกัน และข้อมูลผู้ซื้อสินค้าในรายการเปลี่ยนแปลงนั้น จะถูกกำหนดโดยค่าของแอตทริบิวต์ที่สามารถแสดงตัวตน

โดยไม่ซ้ำใคร ดังที่แสดงให้เห็นในแถบป้ายบอกข้อมูล (Tag) OWN ที่มีการระบุค่า ID ซึ่งเป็นรหัสของผู้ซื้อสินค้า และ แถบป้ายบอกข้อมูลITEM ที่มีการระบุค่า TIME ซึ่งเป็นช่วงเวลาของการซื้อสินค้านั้น ยกตัวอย่างเช่น รายการซื้อสินค้าของลูกค้าที่มี ID เท่ากับ 1 ได้ทำการซื้อสินค้ารหัส 1, 5 โดยทำการซื้อในวันที่ 7 และซื้อสินค้ารหัส 6, 35 โดยการซื้อสินค้าในวันที่ 7

โดยการแปลงข้อมูลสำหรับการทดลองในครั้งนี้ได้ทำการแปลงข้อมูลจากฐานข้อมูลเชิงสัมพันธ์ให้อยู่ในรูปแบบโครงสร้างข้อมูลเอ็กซ์เอ็มแอล ที่มีลักษณะการจัดเก็บข้อมูลต่างกันสองดังที่จะแสดงให้เห็นในรูปที่ 3.4 และรูปที่ 3.5

```
<?XML version "1.0"?>
  <buyinfo>
    <item itemid="1">
      <buydate date="1">
        <CID>16</CID>
        <CID>39</CID>
      </buydate>
      ...
      <buydate date="31">
        <CID>9</CID>
        <CID>23</CID>
        <CID>41</CID>
        <CID>54</CID>
        <CID>72</CID>
        <CID>76</CID>
      </buydate>
    </item>
    ...
  </buyinfo>
</XML>
```

รูปที่ 3.4 โครงสร้างการจัดเก็บข้อมูลการซื้อสินค้า ในรูปแบบเอ็กซ์เอ็มแอล

จากรูปที่ 3.4 แสดงให้เห็นถึงรูปแบบการแปลงข้อมูลจากฐานข้อมูลเชิงสัมพันธ์ ให้อยู่ในรูปแบบของโครงสร้างเอ็กซ์เอ็มแอล ที่จัดเก็บ DB2 เพื่อใช้ในการทดสอบการอิมพลีเมนต์ขั้นตอนวิธี GSP เพื่อหารูปแบบลำดับการซื้อสินค้า จะเห็นได้ว่าการจัดเก็บข้อมูลการซื้อสินค้าจะถูกเก็บในลักษณะของการสร้างอิลิเมนต์ ของรหัสสินค้าที่มีการซื้อเดือนนั้น ๆ ซึ่งภายในจะเก็บข้อมูลวันที่ โดยจะสร้างอิลิเมนต์ของวันที่แต่ละวันในเดือนนั้น ๆ โดยมี DATE เป็นตัวระบุวันที่ ที่มีการซื้อสินค้าซึ่งภายในอิลิเมนต์ดังกล่าวก็จะถูกจัดเก็บ ข้อมูลรหัสของผู้ซื้อสินค้าชิ้นนั้นๆ ภายในวันที่ดังกล่าวไว้ด้วย

```
<?XML version "1.0"?>
  <buyinfo>
    <cid id="1">
      <item date="6">1</item>
      <item date="6">5</item>
      <item date="6">20</item>
      <item date="6">24</item>
      ...
    </cid>
    ...
    <cid id="100">
      <item date="4" tid="125">20</item>
      <item date="4" tid="125">39</item>
      <item date="4" tid="125">40</item>
      <item date="4" tid="125">80</item>
      <item date="4" tid="125">85</item>
      ...
    </cid>
  </buyinfo>
</XML>
```

รูปที่ 3.5 โครงสร้างการจัดเก็บข้อมูลการซื้อสินค้า ในรูปแบบเอ็กซ์เอ็มแอล แบบที่ 2

จากรูปที่ 3.5 แสดงให้เห็นถึงรูปแบบการแปลงข้อมูลจากฐานข้อมูลเชิงสัมพันธ์ ให้อยู่ในรูปแบบของโครงสร้างเอ็กซ์เอ็มแอล ที่จัดเก็บ DB2 แบบที่ 2 เพื่อใช้ในการทดสอบการอิมพลีเมนต์ขั้นตอนวิธี GSP เพื่อหารูปแบบลำดับการซื้อสินค้า จะเห็นได้ว่าการจัดเก็บข้อมูลการซื้อสินค้าจะถูกเก็บในลักษณะของการสร้างอีลิเมนต์ ของรหัสลูกค้าที่มีการซื้อเดือนนั้น ๆ โดยมี ID เป็นตัวระบุตัวตนของผู้ซื้อสินค้า ซึ่งภายในจะเก็บข้อมูลของสินค้า โดยจะสร้างอีลิเมนต์ของการซื้อสินค้าของลูกค้ารายดังกล่าว โดยมี DATE เป็นตัวระบุวันที่ ที่มีการซื้อสินค้าของสินค้าชิ้นดังกล่าว

ซึ่งการแปลงข้อมูลที่อยู่ในรูปแบบของฐานข้อมูลเชิงสัมพันธ์ ให้อยู่ในรูปแบบของฐานข้อมูลเอ็กซ์เอ็มแอลทั้งสองรูปแบบที่กล่าวไปนั้น เพื่อเปรียบเทียบว่าลักษณะของการเก็บข้อมูลของเอ็กซ์เอ็มแอลมีผลกับประสิทธิภาพในสอบถามข้อมูลหรือไม่ เมื่อนำมาใช้ในการอิมพลีเมนต์บนขั้นตอนวิธี GSP

3.2 การอิมพลีเมนต์ขั้นตอนวิธี GSP

สำหรับขั้นตอนวิธี GSP ที่จะนำมาอิมพลีเมนต์ ทฤษฎีดังกล่าวถูกกล่าวไว้ในบทที่ 2 ดังนั้นในบทนี้จะนำเสนอเฉพาะส่วนที่เกี่ยวข้องกับการอิมพลีเมนต์ ที่ต้องใช้ภาษาสำหรับการสอบถามเอ็กซ์คิวรี มาเกี่ยวข้องเพื่อใช้ในการสอบถามข้อมูลจากฐานข้อมูลเอ็กซ์เอ็มแอล นั่นคือส่วนของ k -length candidate ซึ่งเป็นส่วนที่ใช้ในการนับค่าสนับสนุนซึ่งเป็นส่วนหนึ่งในกระบวนการในการค้นหารูปแบบลำดับของข้อมูลบนขั้นตอนวิธี GSP ดังที่จะแสดงให้เห็นในรูปที่ 3.6

```

Let  $C_k$  denote the set of candidate  $k$ -itemsets.
Let Owner denote all transaction owners.
for each ( $c \in C_k$ ) {
  for each owner in Owner
    Parse XML element by the white spaces.
    XQuery1: Determine the first time stamp of supporting transactions.
    for each time stamp from XQuery1
      for each XML element
        XQuery2: Determine the element with minimum time stamp.
        if the element satisfies the time constraints
          Increase the support count for the candidate.

```

รูปที่ 3.6 GSP k -length candidate verification

โดยขั้นตอนวิธี GSP ในส่วนนี้จะใช้ในการ อิมพลีเมนต์เพื่อใช้ในการหารูปแบบลำดับของข้อมูล บนฐานข้อมูลเอ็กซ์เอ็มแอลทั้งสองแบบ ซึ่งจะแตกต่างกันที่การเขียนภาษาสำหรับการสอบถามเอ็กซ์คิวรี ซึ่งมีความแตกต่างกัน เนื่องจากลักษณะของการจัดเก็บข้อมูลที่แตกต่างกัน ดังที่จะแสดงให้เห็นในรูปที่ 3.7 และรูปที่ 3.8

<pre>xquery count(fn:distinct-values(for \$info in db2fn:xmlcolumn("DBT2000XMLV6.INFOGSP")/buyinfo/cid let \$resultXML := \$info where item = patternArr[0] return \$resultXML));</pre> a) XQuery 1	<pre>xquery count(for \$info in db2- fn:xmlcolumn("DBT2000XMLV3.INFOGSP")/buyinfo/tid where \$info/item = patternArr[p] and \$info/cid = own and \$info/buydate >= startdate and \$info/buydate <= (startdate + windowsize) return \$info/buydate)</pre> b) XQuery 2
---	--

รูปที่ 3.7 ภาษาสำหรับการสอบถามข้อมูล ในรูปแบบเอ็กซ์เอ็มแอล แบบที่ 1

<pre>xquery fn:distinct-values(for \$info in db2- fn:xmlcolumn("DBT2000XMLV6.INFOGSP")/ buyinfo/cid[@id="1"] return \$info/item[.="20"]/@time)</pre> a) XQuery 1	<pre>xquery fn:min(for \$info in db2-fn: xmlcolumn("DBT2000XMLV6.INFOGSP") /buyinfo/cid[@id = own] return \$info/item[.=""] + patternArr[p] + "\" and @time >= \" + window_time + \" and @time <= \" + (window_time+ max_gap) + \""/@time)</pre> b) XQuery 2
---	--

รูปที่ 3.8 ภาษาสำหรับการสอบถามข้อมูล ในรูปแบบเอ็กซ์เอ็มแอล แบบที่ 2