การเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุ

วิเชียร ชุติมาสกุล [1] และ ศิริพร กองอำนวยสุข [2]
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี บางมด ทุ่งครุ กรุงเทพฯ 10140

บทคัดย่อ

การเพิ่มประสิทธิภาพการเข้าถึงข้อมูลมีความสำคัญมากในระบบฐานข้อมูลเชิงวัตถุ เนื่องจากความต้องการลดจำนวนข้อมูลที่โอนย้ายและเวลา วัตถุประกอบด้วยส่วนข้อมูลและฟังก์ชันที่เกี่ยวข้อง การเข้าถึงข้อมูลสามารถกระทำโดยผ่านฟังก์ชันที่กำหนดล่วงหน้าสำหรับวัตถุนั้นๆ บทความนี้แสดงถึงความเป็นไปได้ในการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุในสองระดับ การเพิ่มประสิทธิภาพการเข้าถึงข้อมูลในระดับบนเป็นการพิจารณาคำถามหรือการสอบถามเพื่อให้ได้วัตถุที่ต้องการจากการกำหนดเงื่อนไขบนวัตถุอื่นๆ ที่เกี่ยวข้อง โดยความสัมพันธ์ระหว่างวัตถุทั้งสองเป็นแบบหลายวัตถุต่อหลายวัตถุ ค่าใช้จ่ายในการค้นหาข้อมูลจากวัตถุที่กำหนดเป็นเงื่อนไขและเชื่อมไปยังวัตถุที่ต้องการมีค่าน้อยกว่าค่าใช้จ่ายในการค้นหาข้อมูลจากวัตถุที่ต้องการไปยังวัตถุที่กำหนดเป็นเงื่อนไข การเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุได้พิจารณาจากเส้นทางและค่าใช้จ่ายที่เกิดขึ้น ประเด็นถัดมาเป็นการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลในระดับล่าง กระทำโดยการวิเคราะห์นิพจน์ต่างๆ ที่สมมูลกัน และการพิจารณาประเมินค่าใช้จ่ายจากการนำข้อมูลเข้าและข้อมูลชั่วคราวที่เกิดขึ้น การคำนวณตามกฎพีชคณิตเชิงวัตถุจะสามารถระบุนิพจน์ที่สามารถเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุได้

คำสำคัญ : วัตถุ / การสอบถามข้อมูลเชิงวัตถุ / การเพิ่มประสิทธิภาพ

---

[1] *รองศาสตราจารย์ คณะเทคโนโลยีสารสนเทศ*

[2] *นักวิจัย คณะเทคโนโลยีสารสนเทศ*

Object Query Optimization

Wichian Chutimasakul [1] **and Siriporn Kong-Umnauysuk** [2]

King Mongkut's University of Technology Thonburi, Bangmod, Toongkru, Bangkok 10140

Abstract

An object query optimization is important for an object-oriented database system because it is necessary to improve the system by reducing the transferred data and time. The only way to access data is by using its pre-defined operations. This research is constructed to find out the possibility for optimizing an object query. The high-level object query optimization is to access a constraint object in order to get a required object. The relationship between both objects is a many-to-many. The cost of this method is less than the cost of querying a required object before a constraint object. Furthermore, the work emphasizes an object query optimization by regarding the path and cost of each query. The selection of equivalent expressions is the second consideration, which is also known as the low-level object query optimization. This method can be achieved by analyzing input and temporary data of each expression in order to optimize an object query.

Keywords : Object / Object Query **Language** / Optimization

---

[1] *Associate Professor, School of Information Technology.*

[2] *Researcher, School of Information Technology.*

บทนำ

เทคโนโลยีเชิงวัตถุ (Object-Oriented Technology) มีลักษณะหลายประการที่สามารถประยุกต์กับงานที่มีความซับซ้อน อาทิ การกำหนดคลาส (class) และวัตถุ (object) การนิยามนามธรรม (abstraction) การถ่ายทอดคุณสมบัติ (inheritance) เอ็นแคบซูเลชัน (encapsulation) และโพลิมอร์ฟิซึม (polymorphism) [1] บทความนี้เน้นการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุ วัตถุประกอบด้วยข้อมูลและฟังก์ชันที่เกี่ยวข้อง การกำหนดวัตถุทำให้มีชนิดข้อมูลเพิ่มเติมจากชนิดข้อมูลพื้นฐาน เช่น ตัวเลข (integer) หรือ ตัวอักษร (characters) จากลักษณะต่างๆ ของเทคโนโลยีเชิงวัตถุดังกล่าว สามารถจำลองระบบที่ซับซ้อนได้ใกล้เคียงกับโลกความเป็นจริง และสามารถพัฒนาระบบได้อย่างมีประสิทธิภาพ การเข้าถึงข้อมูลในฐานข้อมูลเชิงวัตถุเพื่อให้ได้ข้อมูลที่ต้องการอย่างรวดเร็วเพื่อการตัดสินใจมีความสำคัญ ดังนั้นการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุเป็นหัวข้อการวิจัยที่ต้องการได้มาซึ่งการส่งผ่านข้อมูลเท่าที่ต้องการใช้งานจริง เพื่อใช้เวลาน้อยในการดึงข้อมูล โดยการเข้าถึงข้อมูลนั้นต้องใช้ฟังก์ชันที่กำหนดไว้ในวัตถุนั้นล่วงหน้าแล้วเท่านั้น งานวิจัยการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลแบบย้อนกลับของระบบแบบจำลองเชิงโปรตีน (An Object-based Protein Modeling System) [2] และอัลกอริทึมกำหนดแผนปฏิบัติงานในระบบการจัดการฐานข้อมูลเชิงวัตถุ [3] เป็นการเพิ่มประสิทธิภาพการสอบถามข้อมูลและการแยกพิจารณาเงื่อนไขและอัลกอริทึมทำให้ขาดความคล่องตัว บทความนี้นำเสนอวิธีการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุ ซึ่งประกอบด้วยการค้นหาวัตถุที่ต้องการจากวัตถุเงื่อนไข และการเพิ่มประสิทธิภาพโดยการเลือกนิพจน์ที่มีค่าใช้จ่ายน้อยกว่าจากนิพจน์ที่สมมูลกัน หัวข้อถัดไปกล่าวถึงการเข้าถึงข้อมูลเชิงวัตถุ การเพิ่มประสิทธิภาพโดยค้นหาวัตถุที่ต้องการจากวัตถุเงื่อนไข การเพิ่มประสิทธิภาพโดยเลือกนิพจน์พีชคณิตเชิงวัตถุ และสรุปผลงานวิจัย

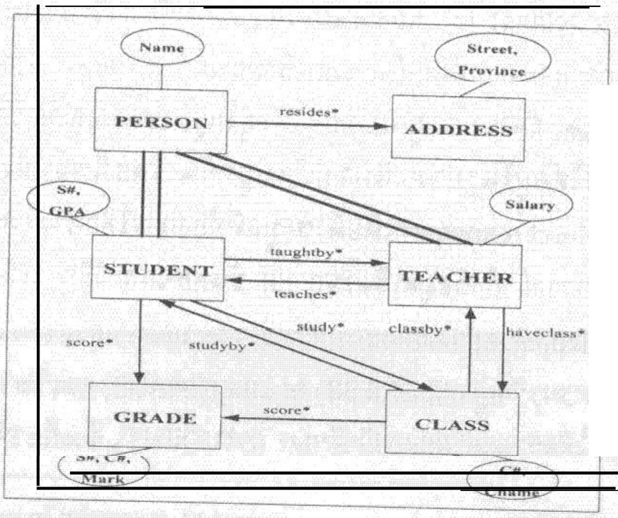
การเข้าถึงข้อมูลเชิงวัตถุ

วัตถุ (object) ประกอบด้วยข้อมูลและฟังก์ชันที่เกี่ยวข้อง การเข้าถึงข้อมูลภายในวัตถุสามารถทำได้โดยผ่านฟังก์ชันที่กำหนดแล้วล่วงหน้าดังกล่าว วัตถุที่มีลักษณะคล้ายกันจะรวมกันเป็นชนิดวัตถุ (object type) หรือคลาส (class) [4] ขั้นตอนการสอบถามข้อมูลในระบบการจัดการฐานข้อมูลเชิงวัตถุคล้ายกับขั้นตอนการสอบถามข้อมูลในระบบเชิงสัมพันธ์ [5] ภาษาการสอบถามข้อมูลเชิงวัตถุ (Object Query Language: OQL) จะถูกเปลี่ยนให้อยู่ในรูปพีชคณิตเชิงวัตถุ (object algebra) และสามารถเปลี่ยนต่อไปให้อยู่ในรูปของนิพจน์ที่สมมูลกัน โดยนิพจน์ดังกล่าวถูกเรียกใช้โดยตัวจัดการข้อมูลเชิงวัตถุ (object manager) เพื่อดึงข้อมูลที่ต้องการได้อย่างรวดเร็วและมีค่าใช้จ่ายต่ำเป็นการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุ [3] ขั้นตอนและวิธีการเข้าถึงข้อมูลเชิงวัตถุจากคำถามของผู้ใช้ได้อธิบายเชิงยกตัวอย่างของระบบการเรียนการสอน คลาสของระบบการเรียนการสอนประกอบด้วยบุคคล (PERSON) ที่อยู่ (ADDRESS) คะแนน (GRADE) และวิชา (CLASS) คลาส*บุคคล* ประกอบด้วยชื่อ (Name) ใช้สร้างคลาสย่อย*นักศึกษา* (STUDENT) ซึ่งเก็บรหัสนักศึกษา (S#) และ

คะแนนเฉลี่ยสะสม (GPA) และคลาสย่อย*อาจารย์* (TEACHER) ใช้เก็บเงินเดือน (Salary) คลาส*ที่อยู่* ประกอบด้วยชื่อถนน (Street) และชื่อจังหวัด (Province) ส่วนคลาส*คะแนน*ประกอบด้วยรหัสวิชา (C#) รหัสนักศึกษา (S#) และเกรด (Mark) และคลาสวิชาประกอบด้วยรหัสวิชา (C#) และชื่อวิชา (Cname)

รูปที่ 1 แสดงความสัมพันธ์ของคลาสต่างๆ ในระบบการเรียนการสอน ความสัมพันธ์ของคลาสบุคคลประกอบด้วยความสัมพันธ์จากบุคคลไปยังที่อยู่คือที่พักอาศัย (resides*) ความสัมพันธ์ของคลาสนักศึกษาได้แก่ความสัมพันธ์จากนักศึกษาไปยังอาจารย์คืออาจารย์ผู้สอน (taughtby*) ความสัมพันธ์จากนักศึกษาไปยังวิชาคือวิชาเรียน (study*) และความสัมพันธ์จากนักศึกษาไปยังคะแนนคือเกรดที่ได้ (score*) ความสัมพันธ์ของคลาสอาจารย์ได้แก่ความสัมพันธ์จากอาจารย์ไปยังนักศึกษาคือนักศึกษาที่อาจารย์สอน (teaches*) และความสัมพันธ์จากอาจารย์ไปยังวิชาคือวิชาสอน (haveclass*) ความสัมพันธ์ของคลาสวิชาได้แก่ความสัมพันธ์จากวิชาไปยังนักศึกษาคือนักศึกษาที่เรียนวิชานั้น (studyby*) ความสัมพันธ์จากวิชาไปยังอาจารย์คืออาจารย์ที่สอนในวิชานั้น (classby*) และความสัมพันธ์จากวิชาไปยังคะแนนคือเกรดในวิชานั้น (score*) สัญลักษณ์เส้นคู่แสดงการถ่ายทอดคุณสมบัติไปยังคลาสย่อย ลูกศรแสดงเมททอดหรือฟังก์ชัน เมททอดที่มี '*' หมายถึง ข้อมูลที่เป็นเซตของค่าวัตถุเอกลักษณ์ [8] รายละเอียดของคลาสต่างๆ ในระบบการเรียนการสอนแสดงได้ดังรูปที่ 2 ซึ่งแต่ละคลาสประกอบด้วย

- ADT (Abstract Data Type) คือ การประกาศชื่อชนิดวัตถุหรือคลาส
- Supertype คือ การประกาศว่า ADT นั้นรับการถ่ายทอดคุณสมบัติมาจากคลาสอื่น
- Extent คือ การประกาศชื่อของกลุ่ม (collection) วัตถุที่มีอยู่ใน ADT
- Private คือ การประกาศชื่อตัวแปรที่เป็นข้อมูลส่วนตัว
- Public คือ การประกาศฟังก์ชันใช้ร่วมกันของคลาส
- Link คือ การประกาศฟังก์ชันความสัมพันธ์ระหว่างวัตถุ ฟังก์ชันนี้เก็บค่าวัตถุเอกลักษณ์อื่นที่สัมพันธ์กันและเข้าถึงข้อมูลที่อ้างอิงโดยเรียกใช้ฟังก์ชันการเข้าถึงข้อมูลของวัตถุที่อ้างถึงนั้น



รูปที่ 1 ความสัมพันธ์ระหว่างคลาสของระบบการเรียนการสอน

ADT	PERSON	ADT	ADDRESS
Extent	PERSONS	Extent	ADDRESSED
Private	Name : string	Private	Street : string
Public	Name ()		Province* : string
Link	resides* [ADDRESS]	Public	Street ()
			Province* ()
ADT	STUDENT	ADT	CLASS
Supertype	PERSON	Extent	CLASSES
Extent	STUDENTS	Private	C# : string
Private	S# : integer		Name : string
	GPA : real	Public	C# 0
Public	S# ()		Name 0
	GPA ()	Link	study by* [STUDENT]
Link	taughtby* [TEACHER]		score* [GRADE]
	score* [GRADE]		class by* [TEACHER]
	study* [CLASS]		
ADT	TEACHER	ADT	GRADE
Supertype	PERSON	Extent	GRADES
Extent	TEACHERS	Private	S# : string
Prirate	Salary : real		C# : integer
Public	Salary ()		Mark : real
Link	haveclass* [CLASS]	Public	S# ()
	teaches* [STUDENT]		C# ()
			Mark ()

รูปที่ 2 รายละเอียดของคลาสในระบบการเรียนการสอน

ขั้นตอนการเข้าถึงข้อมูล

ขั้นตอนการเข้าถึงข้อมูลสำหรับคำถามของผู้ใช้จะถูกแปลงเป็นนิพจน์พีชคณิตเชิงวัตถุ (Object Algebra Expression) ซึ่งสามารถแปลงได้หลายนิพจน์ที่สมมูลกันโดยใช้คุณสมบัติทางคณิตศาสตร์ พีชคณิตเชิงวัตถุ (object algebra) ได้กำหนดตัวปฏิบัติการเชิงวัตถุพื้นฐาน 5 อย่าง ดังนี้คือ [7][8]

- **Union** $(P \cup Q)$: เซตของวัตถุที่อยู่ใน P หรือ Q หรือทั้งสอง
- **Difference** (P - Q) : เซตของวัตถุที่อยู่ใน P และไม่อยู่ใน Q
- **Select** $(P\ \sigma_F < Q_1 ... Q_k >)$: เซตของวัตถุที่อยู่ใน P ซึ่งสอดคล้องเงื่อนไข F และมี $Q_1, ..., Q_k$ เป็น พารามิเตอร์
- **Generate** $(Q_1\ \gamma^T_F < Q_2 ... Q_k >)$: ตัวปฏิบัติการให้ผลลัพธ์เป็นวัตถุ T กำหนดเป็นตัวแปรเป้าหมาย ซึ่งไม่ใช่เป็นค่าที่อยู่ในเซตของวัตถุเข้า โดย T สอดคล้องตามเงื่อนไข F และมี $Q_1, ..., Q_k$ เป็นพารามิเตอร์
- **Map** $(Q_1 \mapsto_{mlist} < Q_2 ... Q_k >)$: mlist คือ รายชื่อเมททอดของรูปแบบ $m_1, ..., m_m$ Map ใช้กับลำดับของเมททอดใน mlist ไปยังแต่ละวัตถุ $q_1 \in Q_1$ โดยวัตถุใน $< Q_2 ... Q_k >$ เป็นพารามิเตอร์ให้กับเมททอดใน mlist ผลลัพธ์ คือ เซตของวัตถุที่ได้จากลำดับของการทำงาน

ตัวอย่างการประยุกต์พีชคณิตเชิงวัตถุกับการค้นหาข้อมูลจากฐานข้อมูลเชิงวัตถุมีดังนี้คือ

- การ **Select** "จงหานักศึกษาที่มีเกรดเฉลี่ยไม่ต่ำกว่า 3.00" พีชคณิตเชิงวัตถุคือ

  (STUDENTS) σ_F < > where $F \equiv$ (S.GPA >= "3.00")

- การ **Generate** "จงหาอาจารย์ที่สอนนักศึกษาศิริพร" พีชคณิตเชิงวัตถุคือ
  (STUDENTS) γ^{T}_{F} < > where F ≡ (S.Name = "ศิริพร" Λ T = S.taughtby*.Name)
- การ Map "จงหานักศึกษาที่อยู่ในความดูแลของอาจารย์สมชาย" พีชคณิตเชิงวัตถุในกรณีที่มีเซตวัตถุอาจารย์สมชาย (P) อยู่แล้ว และมีฟังก์ชัน teaches* คือ
  (T.Name = "สมชาย") $|\text{-->}_{teaches*}$ < >

ภาษาการสอบถามข้อมูลเชิงโครงสร้าง (Structured Query Language: SQL) เป็นภาษาพื้นฐานของการสอบถามข้อมูลเชิงสัมพันธ์ซึ่งมีรูปแบบที่เข้าใจง่าย ภาษาการสอบถามข้อมูลเชิงวัตถุ (Object Query Language: OQL) มีรูปแบบการสอบถามข้อมูลคล้ายรูปแบบการสอบถามข้อมูลเชิงสัมพันธ์ [9] ตัวอย่างการค้นหาชื่อจังหวัดที่นักศึกษา "ศิริพร" อาศัยอยู่ สามารถกำหนดในรูปของ OQL ได้ดังนี้คือ

```
SELECT    S.resides*.Province
FROM      STUDENT S in STUDENTS
WHERE     S.Name  =  "ศิริพร"
```

และสามารถเขียนในรูปพีชคณิตเชิงวัตถุคือ

(STUDENTS) γ^{T}_{F} < > where F ≡ (S.Name = "ศิริพร" Λ T = S.resides*.Province)

ความหมายของรูปแบบคำถามคือ ต้องการ S.resides*.Province ในประโยค SELECT โดยที่ S คือวัตถุนักศึกษา (STUDENTS) ในประโยค FROM โดยมีเงื่อนไขว่านักศึกษาชื่อ "ศิริพร" โดยฟังก์ชัน Name ที่ใช้ในประโยค WHERE เป็นฟังก์ชันใช้ร่วมกันของ PERSON ซึ่ง STUDENT สืบทอดมา สังเกตว่าไม่มีการพิจารณาค่าใช้จ่ายในส่วนนี้เนื่องจากเป็นการรับคุณสมบัติการถ่ายทอด (Inheritance) ส่วน resides* เป็นฟังก์ชันของ PERSON ที่เชื่อมโยงไปยัง ADDRESS เมื่อต่อท้ายด้วย ".Province" หมายถึงการเรียกฟังก์ชันการเข้าถึงชื่อของวัตถุ ADDRESSED

การเพิ่มประสิทธิภาพโดยค้นหาวัตถุที่ต้องการจากวัตถุเงื่อนไข

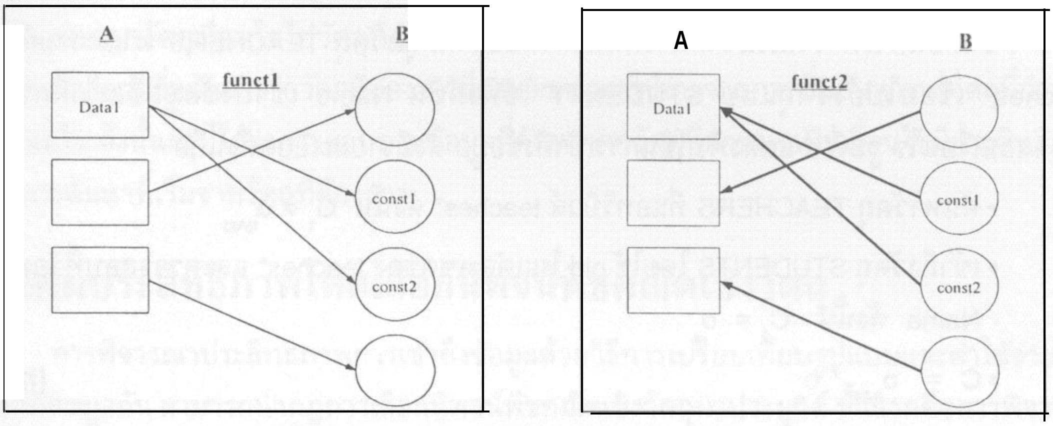
การสอบถามข้อมูลวัตถุสามารถเขียนในรูปแบบของ "SELECT - FROM - WHERE" ได้ 2 แบบ คือ (ก) การค้นหาวัตถุที่ต้องการจากวัตถุที่ต้องการและตรวจสอบเงื่อนไขของแอทริบิวส์ที่เรียกฟังก์ชันของวัตถุที่เป็นเงื่อนไข และ (ข) การค้นหาวัตถุที่ต้องการจากวัตถุที่กำหนดให้เป็นเงื่อนไขและตรวจสอบที่วัตถุเงื่อนไขก่อนจึงใช้ฟังก์ชันในวัตถุเงื่อนไขนั้นเชื่อมโยงไปสู่วัตถุที่ต้องการ ตัวอย่างคำถามเช่น จงหาค่าของ a เมื่อค่า *b="const1"* โดยที่ *a* มีฟังก์ชัน *funct1* ที่เชื่อมไปหา *b* และ *b* มีฟังก์ชัน *funct2* ที่เชื่อมไปหา *a* รูปแบบการสอบถามข้อมูลมี 2 แบบคือ

ก)
```
SELECT a
FROM   a in As
WHERE  a.funct1 = "const1"
```

ข)
```
SELECT  b.funct2
FROM    b in Bs
WHERE   b = "const1"
```

1. เส้นทางการค้นหาวัตถุ

การค้นหาที่เริ่มจากวัตถุที่ต้องการและตรวจสอบวัตถุเงื่อนไข เปรียบเสมือนการตรวจสอบเงื่อนไขอยู่ในขั้นตอนท้ายของการค้นหาข้อมูลทำให้งานค้นหาข้อมูลมีจำนวนมาก สำหรับวิธีการค้นหาคำตอบ โดยเริ่มค้นหาจากกลุ่มวัตถุที่กำหนดเป็นเงื่อนไขและเข้าถึงวัตถุที่ต้องการโดยใช้ความสัมพันธ์ที่เชื่อมโยงระหว่างวัตถุโดยใช้ค่าเอกลักษณ์ของวัตถุ เปรียบเสมือนการตรวจสอบเงื่อนไขอยู่ในขั้นตอนแรกของการค้นหาข้อมูลทำให้ได้รับข้อมูลที่ขนาดเล็กและทราบตำแหน่งของข้อมูลที่ต้องการชัดเจน ไม่ต้องโอนย้ายข้อมูลจำนวนมาก ช่วยประหยัดเวลา ส่งผลให้การสอบถามข้อมูลมีประสิทธิภาพดีขึ้น เส้นทางการค้นหาวัตถุแสดงดังรูปที่ 3 รูป ก) เริ่มค้นหาจากวัตถุที่ต้องการ และ ข) เริ่มค้นหาจากวัตถุที่กำหนดเป็นเงื่อนไข



ก) การค้นหาจากวัตถุที่ต้องการ ข) การค้นหาจากวัตถุเงื่อนไข

รูปที่ 3 เส้นทางการค้นหาข้อมูล

2. ค่าใช้จ่ายในการค้นหาข้อมูลเชิงวัตถุ

ค่าใช้จ่ายในการสอบถามข้อมูลเชิงวัตถุประกอบด้วย 4 ส่วนหลัก คือ ค่าใช้จ่ายของหน่วยประมวลผลกลาง (ซีพียู) ค่าใช้จ่ายในการติดต่อกับอุปกรณ์ภายนอก ค่าใช้จ่ายในการส่งผ่านข้อความ และค่าใช้จ่ายในการถ่ายโอนข้อมูล โดยค่าใช้จ่าย 2 ส่วนแรกเป็นค่าใช้จ่ายที่คงที่ในสภาพแวดล้อมหนึ่งๆ ค่าใช้จ่าย 2 ส่วนถัดมาขึ้นอยู่กับการกระจายของข้อมูลในระบบเชิงกระจาย งานวิจัยนี้เป็นการวิเคราะห์การลดค่าใช้จ่ายในการถ่ายโอนข้อมูลเพื่อการเพิ่มประสิทธิภาพจากการเข้าถึงข้อมูลเชิงวัตถุ

การค้นหาข้อมูลเชิงวัตถุ ระบบจัดการฐานข้อมูลเชิงวัตถุมีการอ่านวัตถุต่างๆ ของแต่ละคลาสในลักษณะการค้นหาแบบเส้นตรง (linear search) ค่าใช้จ่ายในการค้นหาและเข้าถึงข้อมูล (Cost Function: C) คือ จำนวนของวัตถุ (o_i) เมื่อ *i* คือชนิดวัตถุหนึ่งๆ การวิเคราะห์ค่าใช้จ่ายของการค้นหาวัตถุที่ตรวจสอบเงื่อนไขการคำนวณในกรณีที่ไม่ดีสุด (worst case) คือ วัตถุที่ต้องการไม่พบหรือพบเป็นวัตถุท้ายสุด ซึ่งต้องค้นหาวัตถุทั้งหมดในกลุ่มวัตถุนั้น ค่าใช้จ่ายของการค้นหาวัตถุทั้งหมด (C) เท่ากับผลคูณของ $o_{i(All)}$ โดยที่ *All* คือจำนวนชนิดวัตถุทั้งหมดหรือคลาสที่เกี่ยวข้อง

3. ตัวอย่างการค้นหาวัตถุ

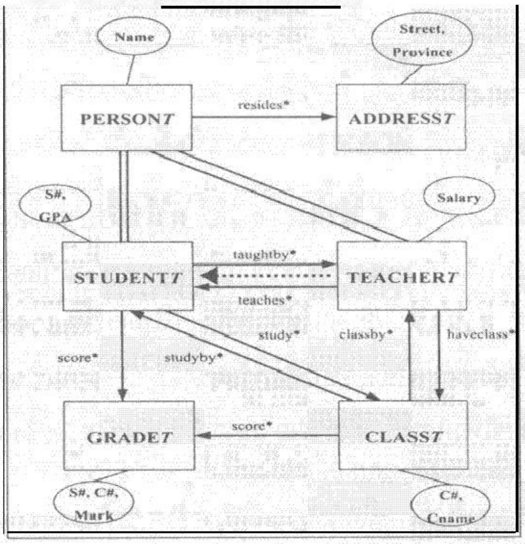
ตัวอย่างคำถามที่สามารถเข้าถึงข้อมูลได้ 2 รูปแบบ อาทิ จงหาชื่ออาจารย์ที่สอนนักศึกษา "ศิริพร" วัตถุที่ต้องการคืออาจารย์ และวัตถุที่เป็นเงื่อนไขคือนักศึกษา สมมุติให้ o_s คือจำนวนวัตถุนักศึกษา o_c คือจำนวนวัตถุวิชา และ o_t คือจำนวนวัตถุอาจารย์ และ C_i คือค่าใช้จ่ายในขั้นตอน i รูปแบบแสดงได้ดังนี้

1) การเริ่มค้นหาจากวัตถุที่ต้องการ คือเริ่มค้นหาจากวัตถุอาจารย์

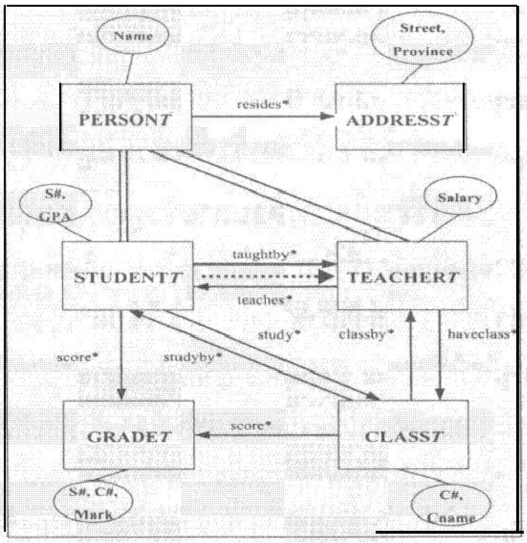
```
SELECT  T.Name
FROM    TEACHER S in TEACHERS
WHERE   T.teaches*.Name = "ศิริพร"
```

ขั้นตอนการทำงานเริ่มจากการค้นหาคำตอบจากกลุ่มวัตถุ TEACHERS ตามความสัมพันธ์ teaches* เชื่อมไปยังวัตถุแบบ STUDENT*T* ใช้ฟังก์ชัน Name เข้าถึงข้อมูลชื่อนักศึกษาและตรวจสอบเงื่อนไข รูปที่ 4 แสดงฟังก์ชันการเข้าถึงข้อมูล ดังมีรายละเอียดดังนี้คือ

- ค้นหาวัตถุ TEACHERS ที่แอทริบิวส์ teaches* ดังนั้น $C_1 = o_{t(All)}$
- เข้าถึงวัตถุ STUDENTS โดยใช้ oid ในแต่ละเซตของ teaches* และตรวจสอบที่แอทริบิวส์ Name ดงนน $C_2 = o_s$
- $C = o_{t(All)} * o_s$ (ก)



รูปที่ 4 ฟังก์ชันการเข้าถึงข้อมูลจากวัตถุที่ต้องการ



รูปที่ 5 ฟังก์ชันการเข้าถึงข้อมูลจากวัตถุเงื่อนไข

2) การเริ่มค้นหาจากวัตถุเงื่อนไข คือเริ่มค้นหาจากวัตถุนักศึกษา

```
SELECT  S.taughtby*.Name
FROM    STUDENT S in STUDENTS
WHERE   S.Name = "ศิริพร"
```

ขั้นตอนการทำงานเริ่มจากการค้นหาคำตอบจากกลุ่มวัตถุ STUDENTS ตรวจสอบเงื่อนไข และเชื่อมกับวัตถุแบบ STUDENT*T* โดยความสัมพันธ์ taughtby* ซึ่งค่าฟังก์ชันเป็นวัตถุรูปแบบ TEACHER*T* และสามารถใช้ฟังก์ชัน Name เข้าถึงข้อมูลชื่ออาจารย์ที่ต้องการ รูปที่ 5 แสดงฟังก์ชันการเข้าถึงข้อมูล ซึ่งมีขั้นตอนดังนี้คือ

- ค้นหาวัตถุ STUDENTS ที่แอทริบิวส์ Name มีค่าเท่ากับ "ศิริพร" ดังนั้น $C_1 = o_s$
- เข้าถึงวัตถุ TEACHERS โดยใช้ oid ในเซตของ taughtby* ดังนั้น $C_2 = o_t$
- $C = o_s + o_t$ **(ข)**

การสอบถามวัตถุที่ต้องการและวัตถุเงื่อนไขสัมพันธ์กันแบบหลายวัตถุต่อหลายวัตถุ (many-to-many) รูปแบบการเข้าถึงข้อมูลวัตถุที่ต้องการสามารถเริ่มค้นหาได้ทั้งจากวัตถุที่ต้องการไปยังวัตถุเงื่อนไขและจากวัตถุเงื่อนไขไปวัตถุที่ต้องการ เมื่อเปรียบเทียบค่าใช้จ่ายแบบ (ก) และ (ข) พบว่าค่าใช้จ่ายจากการเริ่มค้นหาจากวัตถุที่ต้องการมีค่ามากกว่าค่าใช้จ่ายจากการเริ่มค้นหาจากวัตถุที่กำหนดเป็นเงื่อนไข ดังนั้นสรุปได้ว่าการสอบถามข้อมูลที่ค้นหาจากวัตถุที่กำหนดเป็นเงื่อนไขมีประสิทธิภาพดีกว่าการค้นหาที่เริ่มจากวัตถุที่ต้องการ

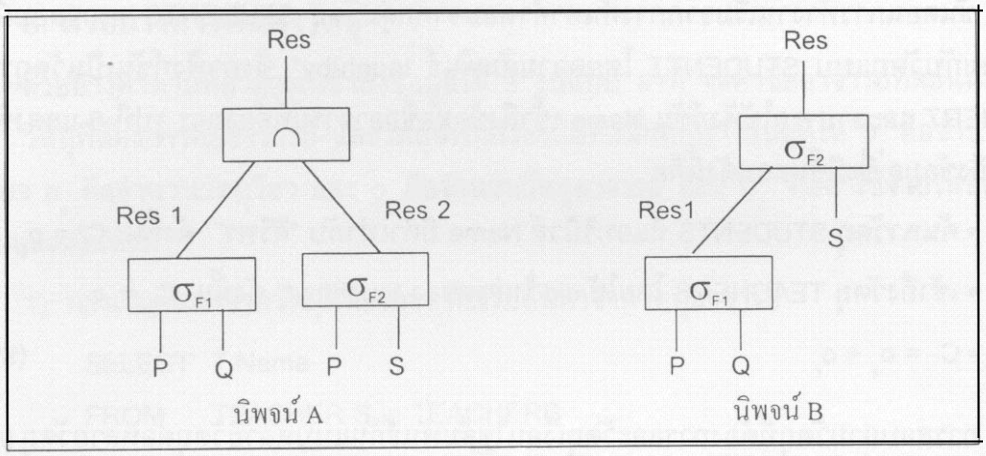
การเพิ่มประสิทธิภาพโดยเลือกนิพจน์พีชคณิตเชิงวัตถุ

การพิจารณาประสิทธิภาพการเข้าถึงข้อมูลด้วยวิธีการเปรียบเทียบรูปแบบและค่าใช้จ่ายของนิพจน์ที่สมมูลกัน สามารถนำกฎการเลือกนิพจน์พีชคณิตเชิงวัตถุมาประยุกต์ ขั้นแรกคือการพิจารณารูปแบบพีชคณิตเชิงวัตถุเทียบกับกฎที่จะอธิบายในหัวข้อถัดไป การตรวจเช็ครูปแบบที่มีอยู่ จำนวนวัตถุเข้าและวัตถุชั่วคราวที่เกิดขึ้นระหว่างการประมวลผล ความสัมพันธ์กันในลักษณะเข้าใกล้จำนวนวัตถุทั้งหมดสามารถวิเคราะห์นิพจน์ที่มีประสิทธิภาพดีกว่าได้ แต่ถ้าจำนวนวัตถุต่างๆ ไม่มีความสัมพันธ์ตามเงื่อนไขในแต่ละกรณีแล้วให้เปรียบเทียบเทอมที่กำหนดมาให้แทน ข้อกำหนดในการวิเคราะห์นิพจน์เชิงวัตถุแสดงในภาคผนวก ก. ส่วนวิธีการวิเคราะห์กฎการสมมูลสามารถค้นหาได้ในหัวข้อวิจัยเรื่องการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุของ ศิริพร กองอำนวยสุข [10] บทความนี้เน้นกฎการสมมูลกันทางพีชคณิตซึ่งแบ่งเป็น 3 กลุ่ม กลุ่มแรกสามารถระบุนิพจน์ที่มีประสิทธิภาพได้ทันที กลุ่มที่สองและสามเป็นการระบุนิพจน์ที่มีประสิทธิภาพได้จากการวิเคราะห์จำนวนผลลัพธ์เข้าและจำนวนผลลัพธ์ชั่วคราวที่เกิดขึ้น ตามลำดับ

1. Factorization of Cascaded Select

$$(P\ \sigma_{F1}\ Q) \cap (P\ \sigma_{F2}\ S) \Leftrightarrow (P\ \sigma_{F1}\ Q)\ \sigma_{F2}\ S$$

นิพจน์พีชคณิตเชิงวัตถุ ด้านบนเมื่อพิจารณา Select 2 ครั้งที่แยกกัน และ การทำ Intersection จะสมมูลกับ การทำพีชคณิตเชิงวัตถุ Select 2 ครั้ง [8] โดยที่นิพจน์ในรูปแบบของ $(P\ \sigma_{F1}\ Q) \cap (P\ \sigma_{F2}\ S)$ มีประสิทธิภาพน้อยกว่า $(P\ \sigma_{F1}\ Q)\ \sigma_{F2}\ S$ แผนภาพโครงสร้างนิพจน์แสดงดังรูปที่ 6 กำหนดให้ Res คือผลลัพธ์ (result) และ Res1 และ Res2 คือผลลัพธ์ชั่วคราว (temporary result)



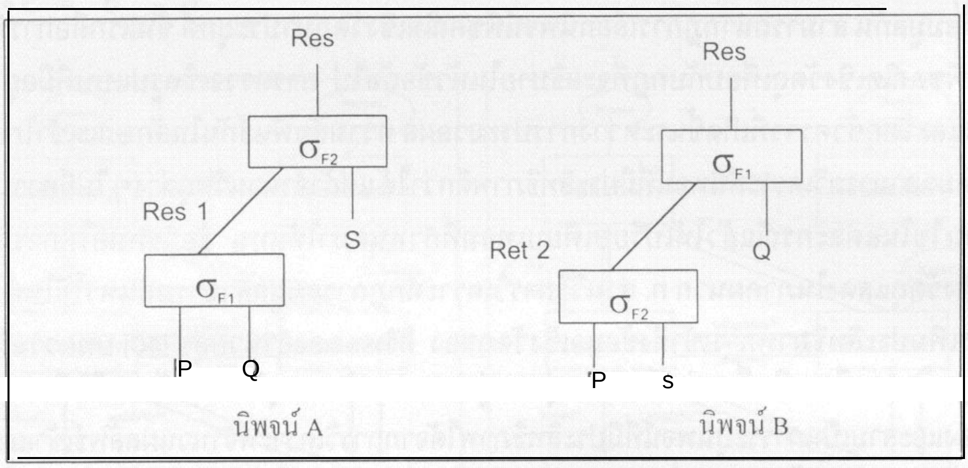
รูปที่ 6 แผนภาพโครงสร้างนิพจน์ Factorization of Cascaded Select

2. นิพจน์สมมูลโดยคุณสมบัติการสลับที่ (Commutativity) [10]

ก. Commutativity of Select

$$(P\,\sigma_{F1}\,Q)\,\sigma_{F2}\,S \Leftrightarrow (P\,\sigma_{F2}\,S)\,\sigma_{F1}\,Q$$

นิพจน์พีชคณิตเชิงวัตถุ Select 2 ครั้งแต่ลำดับต่างกัน สามารถแสดงโครงสร้างนิพจน์ดังรูปที่ 7

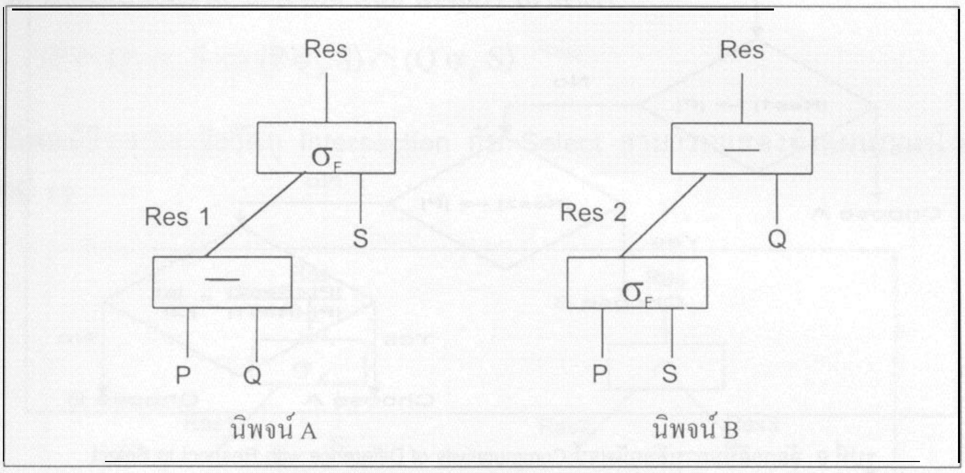


รูปที่ 7 แผนภาพโครงสร้างนิพจน์ Commutativity of Select

ข. Commutativity of Difference with Respect to Select

$$(P - Q)\,\sigma_F\,S \Leftrightarrow (P\,\sigma_F\,S) - Q$$

นิพจน์พีชคณิตเชิงวัตถุ Difference และ Select แผนภาพโครงสร้างนิพจน์ แสดงดังรูปที่ 8



รูปที่ 8 แผนภาพโครงสร้างนิพจน์ Commutativity of Difference with Respect to Select

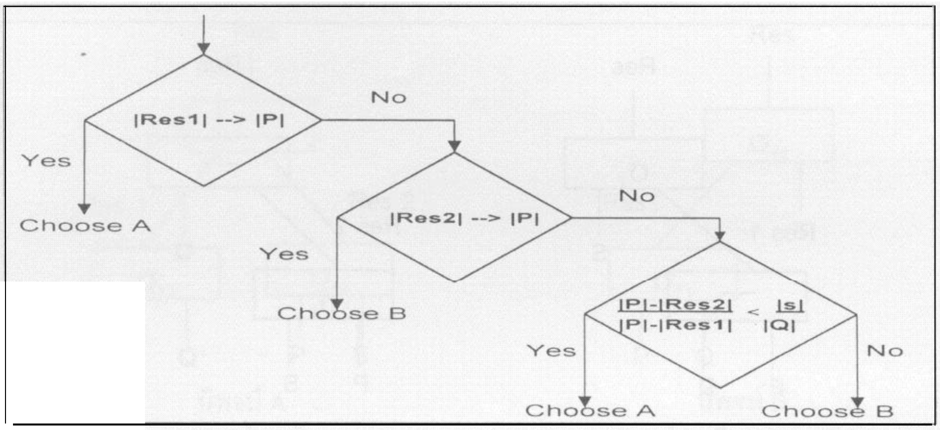
การเลือกนิพจน์ที่มีประสิทธิภาพสำหรับคุณสมบัติการสลับที่

การเลือกนิพจน์ที่มีประสิทธิภาพสำหรับคุณสมบัติการสลับที่มีขั้นตอนดังต่อไปนี้ [10] และสามารถดังแสดงรูปที่ 9

- วิเคราะห์ความสัมพันธ์ของค่าหรือจำนวนวัตถุเอกลักษณ์ (Object Identity) นำเข้า หรือผลลัพธ์ชั่วคราวเป็นอันดับแรก ถ้าจำนวนวัตถุเอกลักษณ์ของผลลัพธ์ชั่วคราวของนิพจน์ซ้าย (A) มีค่าเข้าใกล้จำนวนวัตถุเอกลักษณ์ของเซตที่จะเลือก นิพจน์ขวา (B) จะมีประสิทธิภาพดีกว่า ในทางกลับกันถ้าจำนวนวัตถุเอกลักษณ์ของผลลัพธ์ชั่วคราวของนิพจน์ขวา (B) มีค่าเข้าใกล้จำนวนวัตถุเอกลักษณ์ของเซตที่จะเลือก นิพจน์ซ้าย (A) จะมีประสิทธิภาพดีกว่า
- ในกรณีจำนวนวัตถุเอกลักษณ์ไม่มีความสัมพันธ์ข้างต้นแล้ว ต้องคำนึงถึงขนาดหรือจำนวนวัตถุในเซตโดยวิเคราะห์ความสัมพันธ์ระหว่าง (|P| - |Res 2|) / (|P| - |Res 1|) กับ |S| / |Q| ถ้าผลลัพธ์มีค่าน้อยกว่า นิพจน์ซ้าย (A) จะมีประสิทธิภาพดีกว่านิพจน์ขวา (B) กรณีมีค่ามากกว่า นิพจน์ขวา (B) จะมีประสิทธิภาพดีกว่านิพจน์ซ้าย (A)

ตัวอย่าง กำหนดให้ |P| = 100, |Q| = 100, |S| = 10

กรณีที่ 1	|Res1|—>|P|	: |Res1| = 95, |Res2| = 10	นิพจน์ B มีประสิทธิภาพดีกว่า A
กรณีที่ 2	|Res2|—>|P|	: |Res1| = 10, |Res2| = 95	นิพจน์ A มีประสิทธิภาพดีกว่า B
กรณีที่ 3	|Res1|, |Res2| -/->|P|	: |Res1| = 50, |Res2| = 40	แทนค่าเพื่อการวิเคราะห์ข้างต้น
	(|P| - |Res2|) / (|P| - |Res1|) กับ |S| / |Q| —> 6/5 > 1/10		นิพจน์ B มีประสิทธิภาพดีกว่า A



รูปที่ 9 อัลกอลิทึมการเลือกนิพจน์ Commutativity of Difference with Respect to Select

การเปรียบเทียบค่าใช้จ่ายของนิพจน์ A และ B

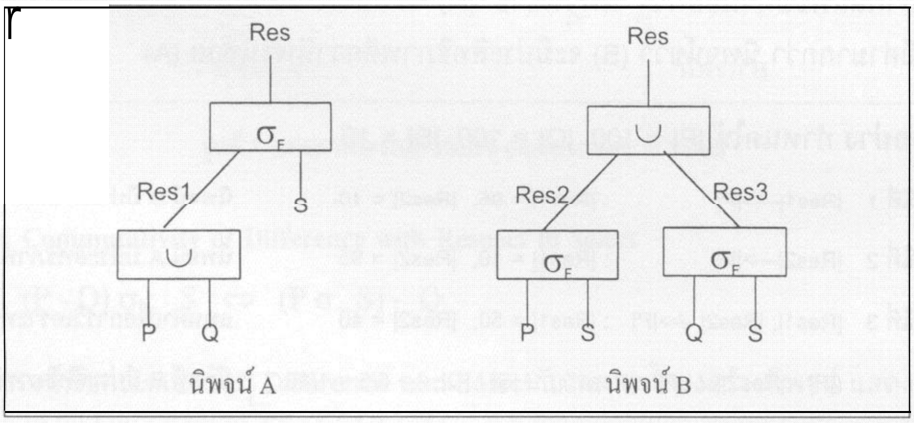
กรณี	A	B	เลือก
ค่าใช้จ่ายรวม	(|P|* |Q|) + (|Res1|* |S|)	(|P|* |S|) + (|Res2|* |Q|)	
1	(100*100)+ (95*10)= 10.950	(l00*10)+(10*100)= 2,000	B
2	(100*100)+(10*10)= 10,100	(100*10)+ (95*10) =)10.500	A
3	(100*100)+(50*10)= 10,500	(100*10)+(40*100)=5,000	B

3 นิพจน์สมมูลโดยคุณสมบัติการกระจาย (Distributivity) [10]

ก. Distributivity of Union with Respect to Select

$$(P \cup Q)\ \sigma_F\ S \Leftrightarrow (P\ \sigma_F\ S) \cup (Q\ s_F\ S)$$

นิพจน์พีชคณิตเชิงวัตถุ Union กับ Select สามารถแสดงดังแผนภาพโครงสร้างนิพจน์ใน รูปที่ 10

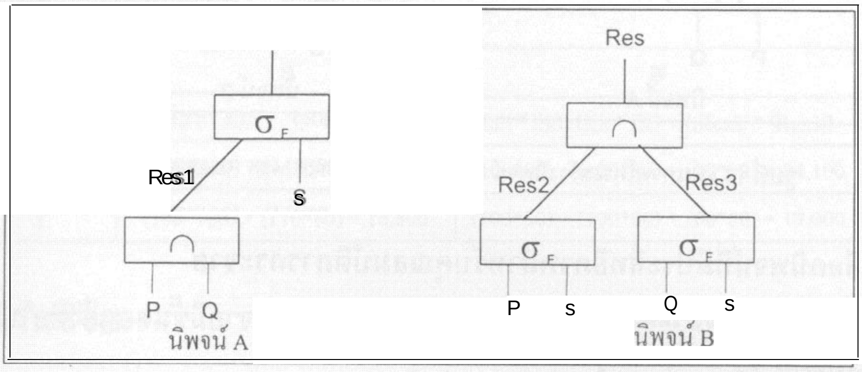


รูปที่ 10 แผนภาพโครงสร้างนิพจน์ Distributivity of Union with Respect to Select

ข. Distributivity of Intersect with Respect to Select

$$(P \cap Q)\ \sigma_F\ S \Leftrightarrow (P\ \sigma_F\ S) \cap (Q\ \sigma_F\ S)$$

นิพจน์พีชคณิตเชิงวัตถุ Intersection กับ Select สามารถแสดงดังแผนภาพโครงสร้างนิพจน์รูปที่ 11

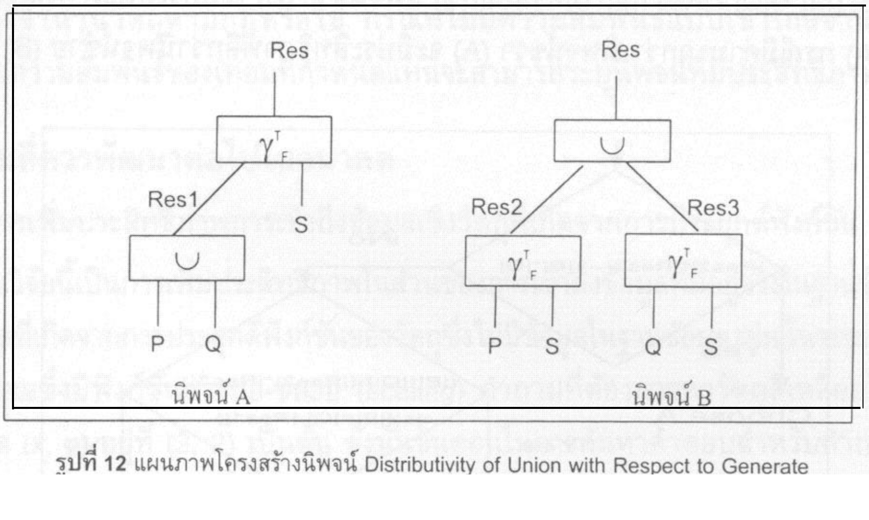


รูปที่ 11 แผนภาพโครงสร้างนิพจน์ Distributivity of Intersect with Respect to Select

ค. Distributivity of Union with Respect to Generate

$$(P \cup Q)\ \gamma^T_F\ \ S \Leftrightarrow (P\ \gamma^T_F\ S) \cup (Q\ \gamma^T_F\ S)$$

นิพจน์พีชคณิตเชิงวัตถุ Union กับ Generate สามารถแสดงดังแผนภาพโครงสร้างนิพจน์รูปที่ 12

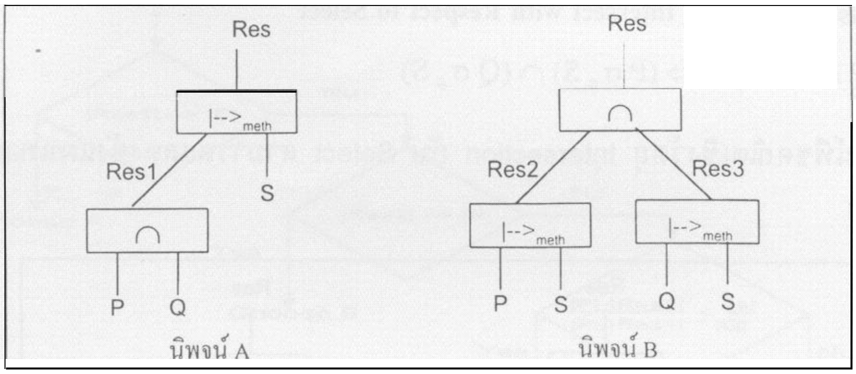


รูปที่ 12 แผนภาพโครงสร้างนิพจน์ Distributivity of Union with Respect to Generate

ง. Distributivity of Union with Respect to Map

$$(P \cap Q)\ |\text{-->}_{meth}\ S \Leftrightarrow (P\ |\text{-->}_{meth}\ S) \cap (Q\ \text{-->}_{meth}\ S)$$

นิพจน์พีชคณิตเชิงวัตถุ Intersection กับ Map สามารถแสดงดังแผนภาพโครงสร้างนิพจน์รูปที่ 13

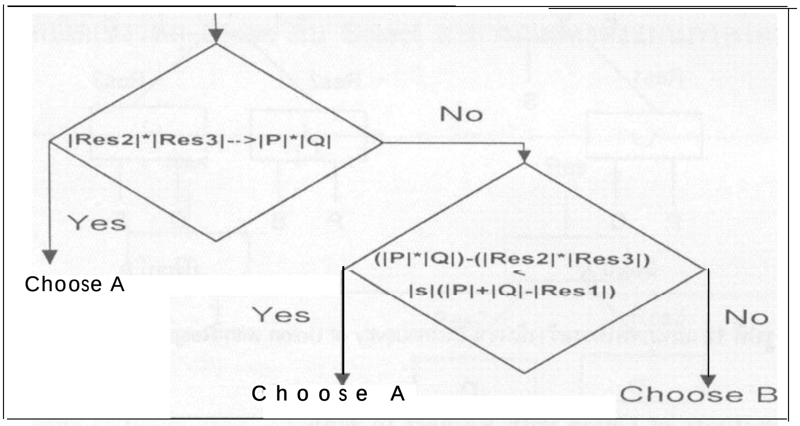


รูปที่ 13 แผนภาพโครงสร้างนิพจน์ Distributivity of Union with Respect to Map

การเลือกนิพจน์ที่มีประสิทธิภาพสำหรับคุณสมบัติการกระจาย

การเลือกนิพจน์ที่มีประสิทธิภาพสำหรับคุณสมบัติการกระจายมีขั้นตอนดังต่อไปนี้ [10] และสามารถแสดงในรูปที่ 14

- วิเคราะห์ความสัมพันธ์ของจำนวนวัตถุเอกลักษณ์นำเข้าหรือผลลัพธ์ชั่วคราว ถ้าผลคูณจำนวนวัตถุเอกลักษณ์ของผลลัพธ์ชั่วคราวของนิพจน์ขวา (B) มีค่าเข้าใกล้ผลคูณจำนวนเอกลักษณ์ของวัตถุเข้าที่จะเลือก นิพจน์ซ้าย (A) จะมีประสิทธิภาพดีกว่า
- ในกรณีจำนวนวัตถุเอกลักษณ์ไม่มีความสัมพันธ์ข้างต้นแล้ว ต้องพิจารณาถึงขนาดหรือจำนวนวัตถุในเซตโดยวิเคราะห์ความสัมพันธ์ระหว่าง (|P| * |Q|) - (|Res2| * |Res3|) กับ |S| (|P| + |Q| - |Res1|) ถ้ามีค่าน้อยกว่า นิพจน์ซ้าย (B) จะมีประสิทธิภาพดีกว่านิพจน์ขวา (A) กรณีมีค่ามากกว่า นิพจน์ขวา (A) จะมีประสิทธิภาพดีกว่านิพจน์ซ้าย (B)



รูปที่ 14 อัลกอลิทึมการเลือกนิพจน์ Distributivity of Union with Respect to Map

ตัวอย่าง กำหนดให้ |P| = 100, |Q| = 100, |S| = 80, |Res1| = 110

กรณีที่ 1 (|Res2|*|Res3|) → |P|*|Q| : |Res2| = 90, |Res3| = 90 นิพจน์ A มีประสิทธิภาพดีกว่า B

กรณีที่ 2 : |Res2| = 60, |Res3| = 50 แทนค่าเพื่อการวิเคราะห์ข้างต้น

(|P|*|Q|)-(|Res2|*|Res3|) กับ |S|(|P|+|Q|-|Res1|) → 7,000 < 7,200 นิพจน์ A มีประสิทธิภาพดีกว่า B

การเปรียบเทียบค่าใช้จ่ายของนิพจน์ A และ B

กรณี	A	B	เลือก
ค่าใช้จ่ายรวม	(|P| * |Q|) + (|Res1| * |S|)	(|P| * |S|)+(|Q| * |S|) + (|Res2| * |Res3|)	
1	(100*100)+(110*80)=18,800	(100*80)+ (100*80)+ (90*90)=24,100	A
2	(100*100)+(110*80)= 18.800	(100*80)+ (100*80)+(60*50)= 19.000	A

สรุปผลงานวิจัย

แนวทางการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุในระดับบนเป็นการพิจารณาคำถามที่ต้องการวัตถุหนึ่งจากการกำหนดเงื่อนไขของอีกวัตถุหนึ่งและความสัมพันธ์ระหว่างวัตถุที่ต้องการกับวัตถุเงื่อนไขเป็นแบบหลายวัตถุต่อหลายวัตถุ การเข้าถึงวัตถุที่ต้องการโดยเริ่มค้นหาจากวัตถุเงื่อนไขมีประสิทธิภาพดีกว่าการเข้าถึงข้อมูลโดยเริ่มค้นหาจากวัตถุที่ต้องการ เนื่องจากการเข้าถึงข้อมูลที่ต้องการโดยเริ่มค้นหาจากวัตถุเงื่อนไขมีค่าใช้จ่ายในการค้นหาน้อยกว่าการเข้าถึงข้อมูลโดยเริ่มค้นหาจากวัตถุที่ต้องการ และแนวทางการเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุระดับล่าง เป็นการเลือกนิพจน์ที่มีประสิทธิภาพโดยวิเคราะห์ความสัมพันธ์ของจำนวนผลลัพธ์ชั่วคราวมีความสัมพันธ์แบบเข้าใกล้จำนวนวัตถุตามกฎหรือไม่ กรณีที่ไม่มีความสัมพันธ์แบบเข้าใกล้จำนวนวัตถุ ให้เปรียบเทียบความสัมพันธ์ของเทอมที่กำหนดแทนจะสามารถระบุนิพจน์ที่มีประสิทธิภาพดีกว่าได้

งานที่ควรพัฒนาต่อไปในอนาคต

• การเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุที่เกิดจากการประยุกต์ฟังก์ชัน

งานวิจัยนี้เป็นการเพิ่มประสิทธิภาพในส่วนของการเข้าถึงข้อมูลที่มีอยู่จริงในฐานข้อมูลเชิงวัตถุ สำหรับข้อมูลที่เกิดจากการประยุกต์ฟังก์ชันของวัตถุซึ่งไม่มีข้อมูลในฐานข้อมูล เช่น ในระบบภาพสามมิติ มีวัตถุสี่เหลี่ยมซึ่งมีฟังก์ชันการย่อ-ขยาย (scaling) คำถามที่ต้องการหาวัตถุสี่เหลี่ยมใดที่ขยาย 2 เท่าแล้วมีจุด (x, y) อยู่ที่ (2, 2) เป็นต้น ดังนั้นขั้นตอนในการค้นหาคำตอบสำหรับคำถามนี้ซับซ้อนและยุ่งยากกว่าการค้นหาข้อมูลที่มีอยู่จริงในระบบ เนื่องจากต้องทำการคำนวณตามฟังก์ชันของวัตถุนั้นก่อนจึงสามารถให้คำตอบได้

• การเข้าถึงข้อมูลเชิงวัตถุสำหรับระบบแบบกระจาย

ระบบกระจาย (distributed system) ข้อมูลวัตถุจะถูกเก็บอยู่ในหลายๆ ที่ ดังนั้นการเข้าถึงวัตถุที่ต้องการต้องผ่านขั้นตอนต่างๆ การเข้าถึงข้อมูลที่กระจายอยู่หลายที่ ได้แก่ การส่งข้อความ การเรียกฟังก์ชันของวัตถุนั้นเพื่อเข้าถึงข้อมูลของวัตถุนั้น และการถ่ายโอนข้อมูลหรือวัตถุที่ต้องการ

กลับมา ขั้นตอนเหล่านี้ทำให้เกิดค่าใช้จ่ายในการส่งข้อความ และค่าใช้จ่ายในการถ่ายโอนข้อมูล นอกจากนี้การทำงานหรือประมวลผลในระบบการกระจายเป็นแบบขนาน (parallelism) ซึ่งหมายถึงค่าใช้จ่ายด้านเวลาในการเข้าถึงข้อมูลอาจลดลง เนื่องจากสามารถเข้าถึงข้อมูลหลายๆ ที่ ในเวลาพร้อมๆ กันได้ ดังนั้น การเปรียบเทียบประสิทธิภาพในเชิงเวลาของการเข้าถึงข้อมูลดังกล่าวจะมีความซับซ้อนขึ้น

เอกสารอ้างอิง

1. Coad, P. ***and*** Yourdon, E., 1991, ***Object-Oriented*** Analysis, 2nd ed., Prentice Hall.
2. Gray, M.D.P., Kulkami, G.K. and Paton, W.N., 1992, ***Object-Oriented Database: A Semantic Data Mode/ Approach,*** Edited by Hoare, C.A.R., New York, Prentice Hall, pp. 157-170.
3. Straube, D. and özsu, M., 1995, "Query Optimization and Execution Plan Generation in Object-Oriented Data Management System," IEEE Transaction on ***Knowledge and Data Engineering,*** Vol.l(2): 210-227.
4. Tkach, D. and Puttick, R.,1994, ***Object Technology in Application Development,*** The Benjamin/Cumming.
5. özsu, M. and Straube, D., ***1989, Issues in Query Model Design in Object-Oriented Database*** Systems, The Department of Computer Science, University of Alberta.
6. Kim, W., 1989, "A Model of Queries for Object-Oriented Database," ***Proceedings of the 15th International Conference on Very Large Databases,*** pp. 423-432.
7. özsu, M. and Blakeley, A.J., 1995, Query ***Processing*** in Object-Oriented ***Database System,*** in Modem Database Systems: The Object Model, Interoperability, and Beyond, Addison-Wesley, pp. 146-174.
8. Straube, D. and özsu, M.,1990, ***Queries and Query Processing in Object-Oriented Database Systems,*** The Department of Computer Science, University of Alberta, TR90-33, pp. 31-33.
9. Chamberlin, D., 1976, "SEQUEL 2: A Unified Approach to Data Definition, Manipulation and Control," ***IBM Journal of Research and Development,*** pp. 278-293.
10. ศิริพร กองอำนวยสุข, 2540, "การเพิ่มประสิทธิภาพการเข้าถึงข้อมูลเชิงวัตถุ," *วิทยานิพนธ์ปริญญาวิทยาศาสตรมหาบัณฑิต* คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี.

ภาคผนวก

ข้อกำหนดในการวิเคราะห์นิพจน์เชิงวัตถุ

1. กำหนดให้ตัวจัดการข้อมูลเชิงวัตถุ (Object Manager: OM) มีตัวปฏิบัติการ 4 รูปแบบ โดย $[i_n]$ และ [o] แสดงกลุ่มวัตถุเอกลักษณ์ (Object Identity : oid) ที่เป็นข้อมูลวัตถุเข้าและออกตามลำดับ

- Stream Union มีรูปแบบสมการคือ $OM \cup \{| i_1 |, | i_2 |, | o |\}$
- Stream Difference มีรูปแบบสมการคือ $OM_{diff} \{| i_1 |, | i_2 |, | o |\}$
- Stream Intersection มีรูปแบบสมการคือ $OM \cap \{| i_1 |, | i_2 |, | o |\}$
- Atom Evaluation มีรูปแบบสมการคือ $OM_{eval} \{| i_1 |, | i_2 |, | o |$, meth, pred$\}$

ตัวปฏิบัติการนี้ประยุกต์เมทธอด meth กับแต่ละสมาชิกของ $[i_1]x[i_2]$ และสร้างกลุ่มวัตถุเอกลักษณ์ชั่วคราว ผลลัพธ์เป็นเซตของ oid คือ o, meth หมายถึง เมทธอดที่เฉพาะเจาะจง และ pred หมายถึง เงื่อนไขของวัตถุข้อมูลเข้าหรือผลที่ได้จากเมทธอด

2. ตัวจัดการข้อมูลเชิงวัตถุมีขั้นตอนเริ่มต้นในการนำวัตถุเข้าเพื่อปฏิบัติงานเท่ากับ $|i_1|x|i_2|$

3. P, Q และ S เป็นเซตเอกลักษณ์ของวัตถุข้อมูลเข้าใดๆ ซึ่งอาจเป็นเซตวัตถุที่มาจากคลาสเดียวกันหรือต่างคลาสกัน และ Res เป็นเซตเอกลักษณ์ของวัตถุข้อมูลที่ได้จากการประมวลผล

4. Res i โดยที่ $i = 1, 2$ เป็นเซตวัตถุของผลลัพธ์ชั่วคราวในขณะประมวลผล

5. |Object-Name| แทนจำนวนของวัตถุในเซตนั้นๆ เช่น |P| แทนจำนวนของวัตถุในเซต P และ |Res1| แทนจำนวนของวัตถุในเซต Res1