บทที่ 1

บทนำ

1.1 ความเป็นมา

ในปัจจุบันนี้ ระบบคลังข้อมูลเป็นสิ่งที่มีความสำคัญและจำเป็นมากขึ้นสำหรับองค์กร เพื่อนำมาใช้ในการสนับสนุนการตัดสินใจของผู้บริหาร โดยจะมีการดึงข้อมูลหรือสารสนเทศจากคลังข้อมูลขึ้นมาใช้งาน ซึ่งในการดึงข้อมูลขึ้นมาใช้งานแต่ละครั้ง ส่วนมากมักจะเป็นลักษณะของการสอบถามข้อมูลแบบ Ad-hoc (Ad-hoc Query) กล่าวคือ จะไม่ทราบว่าผู้ใช้จะสอบถามข้อมูลอะไรบ้าง จึงต้องมีการรวบรวมข้อมูลจากหลายแหล่งมาไว้ที่ศูนย์กลาง และเก็บบันทึกข้อมูลทุกอย่างไว้ในคลังข้อมูลทั้งข้อมูลในอดีตและปัจจุบัน ขอบเขตของข้อมูลที่เก็บไว้จึงกว้างขวางครอบคลุมทุกแง่มุมที่สำคัญขององค์กรนั้น ๆ เพื่อให้พร้อมใช้งานในการสอบถามข้อมูลทุกรูปแบบ ดังนั้นคลังข้อมูลจะมีขนาดใหญ่โตมาก อาจจะมีขนาดเป็นกิกะไบต์หรือเทอราไบต์ก็ได้ ทั้งนี้ขึ้นอยู่กับปริมาณข้อมูลที่อยู่ในคลังข้อมูล

เมื่อคลังข้อมูลมีขนาดใหญ่โตมากและมีการสอบถามข้อมูลแบบซับซ้อน การค้นหาข้อมูลก็ย่อมใช้เวลามากขึ้น โดยทั่วไปแล้วถ้าเราต้องการลดเวลาในการค้นหาข้อมูลในคลังข้อมูล (เพิ่มประสิทธิภาพ) เราสามารถทำได้หลายวิธี เช่น โดยการ

- เพิ่มสมรรถนะของเครื่อง เช่น เพิ่มหน่วยความจำ เพิ่มฮาร์ดแวร์ใหม่เข้าไป [1, 2]
- ประมวลผลแบบคู่ขนาน (Parallel Processing) [5, 6]
- ทำดัชนี (Indexing) [9, 11]

ใน 2 วิธีแรกนั้นเราต้องเสียค่าใช้จ่ายในการเพิ่มประสิทธิภาพของการค้นหาข้อมูล แต่สำหรับการทำดัชนีนั้นเป็นการเพิ่มประสิทธิภาพโดยไม่ต้องเสียค่าใช้จ่ายเพิ่ม เพราะการทำดัชนีเราไม่ต้องเพิ่มอุปกรณ์ฮาร์ดแวร์ แต่เป็นการกรองข้อมูลให้เล็กลง ซึ่งการทำดัชนีนั้นก็มีหลายแบบด้วยกัน และแต่ละแบบก็เหมาะกับลักษณะข้อมูลที่แตกต่างกัน

ในงานวิทยานิพนธ์ชิ้นนี้ เป็นการพัฒนาขั้นตอนวิธี (Algorithm) เพื่อสร้างดัชนีสำหรับการค้นหาข้อมูลในคลังข้อมูล ซึ่งอยู่บนหลักการของการทำดัชนีแบบบิตแมป ที่เหมาะกับข้อมูลที่มีคาร์ดินอลิตี้ (คือ จำนวนค่าที่เป็นไปได้ของแอทริบิวต์ที่นำมาสร้างดัชนี แทนด้วย C) ต่ำ มีค่าแตกต่างไม่มากนัก และเหมาะกับข้อมูลที่ไม่เปลี่ยนแปลง ซึ่งพบบ่อยในคลังข้อมูล ทั้งนี้เพื่อประหยัดพื้นที่ในการจัดเก็บดัชนี นอกจากนี้การทำดัชนีแบบบิตแมปยังมีคุณสมบัติของการดำเนินการระดับบิต (Bit Operation) ระหว่างบิตแมปเวกเตอร์ก่อนดึงข้อมูลจริง ทำให้ใช้เวลาน้อยลงในการค้นหาข้อมูล [9, 11]

1.2 การตรวจเอกสารและงานวิจัยที่เกี่ยวข้อง

1.2.1 การตรวจเอกสาร

ในระบบคลังข้อมูล การทำดัชนีบิตแมปมีความสำคัญต่อการเพิ่มประสิทธิภาพโดยไม่ต้องเสียค่าใช้จ่ายในการเพิ่มอุปกรณ์ฮาร์ดแวร์ เพื่อดึงข้อมูลหรือสารสนเทศขึ้นมาใช้งาน ซึ่งส่วนมากมักมีลักษณะการสอบถามข้อมูลแบบซับซ้อนและแบบทันทีทันใด นอกจากนี้ดัชนีบิตแมปยังมีคุณสมบัติในการดำเนินการระดับบิตระหว่างบิตแมปเวกเตอร์ก่อนดึงข้อมูลจริง ทำให้ใช้เวลาน้อยลงในการค้นหาข้อมูล

1.2.2 งานวิจัยที่เกี่ยวข้อง

An Overview of Data Warehousing and OLAP Technology

งานวิจัยนี้ [14] ได้กล่าวถึงคุณลักษณะ ความสำคัญ และสถาปัตยกรรมของคลังข้อมูล การเปรียบเทียบความแตกต่างระหว่างฐานข้อมูลดำเนินการและคลังข้อมูล การออกแบบคลังข้อมูลเชิงแนวคิด เครื่องมือที่ใช้ในการสกัดข้อมูล ล้างทำความสะอาดข้อมูล และโหลดข้อมูลเข้าสู่คลังข้อมูล เครื่องมือในการสอบถามและวิเคราะห์ข้อมูล การเพิ่มประสิทธิภาพของการสอบถามข้อมูล เครื่องมือในการจัดการคลังข้อมูล และงานวิจัยในสาขานี้

An Efficient Bitmap Encoding Scheme for Selection Queries

งานวิจัยนี้ [30] ได้กล่าวถึงประโยชน์ของดัชนีบิตแมปที่มีการนำไปใช้ในการประมวลผลการสอบถามข้อมูลแบบซับซ้อนและแบบทันทีทันใด เพื่อใช้ในระบบการสนับสนุนการตัดสินใจ การสร้างดัชนีบิตแมปแบบพื้นฐาน แบบ Range และได้มีการนำเสนอดัชนีบิตแมปแบบช่วงขึ้นมา

Encoded Bitmap Indexing for Data Warehouses

งานวิจัยนี้ [10] ได้กล่าวถึงความเป็นมาของงานวิจัยว่า เนื่องจากระบบคลังข้อมูลมีขนาดที่ใหญ่โต ชนิดการสอบถามข้อมูลมักจะเป็นแบบซับซ้อน มีอัตราการอ่านข้อมูลที่สูงมาก ทำให้เทคนิคการสร้างดัชนีที่ออกแบบและปรับแต่งสำหรับระบบฐานข้อมูลแบบดั้งเดิม (ฐานข้อมูลดำเนินการ) ที่มีอยู่นั้น ไม่เหมาะสมสำหรับคลังข้อมูล จึงได้มีการนำเสนอการสร้างดัชนีบิตแมปแบบเข้ารหัสสำหรับคลังข้อมูล เพื่อพัฒนาประสิทธิภาพของดัชนีบิตแมปแบบพื้นฐาน ในกรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีคาร์ดินอลิตี้สูง ๆ

1.3 วัตถุประสงค์

เพื่อหาเทคนิคใหม่ในการสร้างดัชนีบิตแมปที่สามารถลดพื้นที่ในการจัดเก็บดัชนี และเวลาที่ใช้ในการค้นหาข้อมูลในคลังข้อมูล

1.4 วิธีการดำเนินการวิจัย

1. ศึกษาแนวคิดที่เกี่ยวข้องกับระบบคลังข้อมูล และดัชนีบิตแมปทั้ง 4 ชนิด ได้แก่ ดัชนีบิตแมปแบบพื้นฐาน แบบ Range แบบช่วง และแบบเข้ารหัส

2. วิเคราะห์และออกแบบเทคนิคการสร้างดัชนีสำหรับการค้นหาข้อมูลในคลังข้อมูล ในส่วนของดัชนีบิตแมปแบบใหม่ ซึ่งเรียกว่า ดัชนีบิตแมปแบบกระจาย

3. กำหนดรูปแบบการประเมินเทคนิคของดัชนีที่สร้างขึ้นใหม่ เปรียบเทียบกับดัชนีบิตแมปทั้ง 3 ชนิด คือ ดัชนีบิตแมปแบบพื้นฐาน แบบช่วง และแบบเข้ารหัส ในกรณีที่เป็นการสอบถามข้อมูลแบบค่าเท่ากัน (Equality Query) และแบบความเป็นสมาชิก (Membership Query)

4. ทำการวิเคราะห์และศึกษาข้อมูลจากการวัดเปรียบเทียบสมรรถนะ TPC-H [34] ในเรื่องข้อมูลทดสอบและการสอบถามข้อมูล

5. ติดตั้งโปรแกรมสำหรับรันผลการสร้างข้อมูลทดสอบจากการวัดเปรียบเทียบสมรรถนะ TPC-H บนระบบปฏิบัติการ Linux Red Hat 9.0 โดยจัดเก็บฐานข้อมูลทดสอบ ในรูปแบบของ Flat File และการสอบถามข้อมูล (ดูรายละเอียดจากภาคผนวก ก หัวข้อ ก.1 และ ก.2 ตามลำดับ)

6. จัดเตรียมข้อมูลทดสอบ โดยการกรองข้อมูลเพื่อเลือกเฉพาะแอทริบิวต์ที่จะนำมาสร้างดัชนี โดยการใช้คำสั่ง awk (ดูรายละเอียดจากภาคผนวก ก หัวข้อ ก.3) บนระบบปฏิบัติการ Linux

7. แยกค่าของแอทริบิวต์ที่ได้จากข้อ 6. ออกมาเป็นแฟ้มข้อมูล สำหรับการค้นหาข้อมูลของดัชนีบิตแมปทั้ง 4 ชนิด บนระบบปฏิบัติการ Linux

8. พัฒนาโปรแกรมเพื่อจัดเก็บดัชนีบิตแมปทั้ง 4 ชนิด ในรูปแบบของบิต โดยใช้ตัวแปลภาษาซีบนระบบปฏิบัติการ Windows XP

9. พัฒนาโปรแกรมเพื่อการค้นหาข้อมูล และประเมินประสิทธิภาพการค้นหาข้อมูลของดัชนีบิตแมปทั้ง 4 ชนิด

10. ตรวจสอบความถูกต้องของการค้นหาข้อมูล โดยการจัดเก็บข้อมูลที่ได้จากการค้นหาไว้ในแฟ้มข้อมูล แล้วนำแฟ้มข้อมูลต้นฉบับและแฟ้มข้อมูลที่ได้จากการค้นหามาเปรียบเทียบว่าเหมือนกันหรือไม่ โดยการใช้คำสั่ง diff ใน Linux ดังนี้

diff [option] From-File To-File

11. สรุปผลจากการพัฒนา การทดสอบ และการประเมินเปรียบเทียบดัชนีบิตแมปทั้ง 4 ชนิด

1.5 ขอบเขตของงานวิจัย

1. ศึกษาดัชนีบิตแมปแบบเดิมที่เคยมีอยู่ ได้แก่ ดัชนีบิตแมปแบบพื้นฐาน แบบ Range แบบช่วง และแบบเข้ารหัส

2. วิเคราะห์และออกแบบเทคนิคเพื่อสร้างดัชนีบิตแมปแบบใหม่สำหรับการค้นหาข้อมูลในคลังข้อมูล ซึ่งเรียกว่า ดัชนีบิตแมปแบบกระจาย

3. ประเมินประสิทธิภาพของเทคนิคการสร้างดัชนีบิตแมปเดิมที่เคยมีอยู่ ได้แก่ ดัชนีบิตแมปแบบพื้นฐาน แบบช่วง และแบบเข้ารหัส กับดัชนีบิตแมปแบบใหม่ ซึ่งเรียกว่า ดัชนีบิตแมปแบบกระจาย สำหรับการค้นหาข้อมูลในคลังข้อมูล ในกรณีที่เป็นการสอบถามข้อมูลแบบค่าเท่ากันและแบบความเป็นสมาชิก

1.6 ขั้นตอนการดำเนินงาน

1. ศึกษางานวิจัยและเอกสารที่เกี่ยวข้อง และกำหนดขอบเขตของปัญหาให้ชัดเจน

2. วิเคราะห์และออกแบบเทคนิคการสร้างดัชนีบิตแมปสำหรับการค้นหาข้อมูลในคลังข้อมูล

3. กำหนดรูปแบบการประเมินเทคนิคของดัชนีบิตแมปที่สร้างขึ้นใหม่ ซึ่งเรียกว่าดัชนีบิตแมปแบบกระจาย และดัชนีบิตแมปแบบเดิมที่เคยมีอยู่ทั้ง 3 ชนิด คือ ดัชนีบิตแมปแบบพื้นฐาน แบบช่วง และแบบเข้ารหัส

4. ศึกษาและวิเคราะห์หาเครื่องมือสำหรับใช้ประเมินดัชนีบิตแมปที่สร้างขึ้นใหม่และดัชนีบิตแมปแบบเดิมที่เคยมีอยู่

5. พัฒนาดัชนีบิตแมปสำหรับการค้นหาข้อมูล ตามที่ได้ทำการออกแบบไว้

6. ดำเนินการประเมินดัชนีบิตแมปที่สร้างขึ้นใหม่กับดัชนีบิตแมปแบบเดิมที่เคยมีอยู่

7. สรุปผลจากการพัฒนา การทดสอบ และการประเมินเปรียบเทียบดัชนีบิตแมปทั้ง 4 ชนิด

8. จัดทำเอกสารประกอบการวิจัย

1.7 ระยะเวลาการดำเนินงาน

ตุลาคม พ.ศ. 2547 – กุมภาพันธ์ พ.ศ. 2549

ตาราง 1-1 ตารางแสดงระยะเวลาดำเนินงาน

ขั้นตอน	พ.ศ. 2547			พ.ศ. 2548												พ.ศ. 2549	
	ตค.	พย.	ธค.	มค.	กพ.	มีค.	เมย.	พค.	มิย.	กค.	สค.	กย.	ตค.	พย.	ธค.	มค.	กพ.
1.	←	→															
2.		←	→														
3.			←	→													
4.				←			→										
5.							←			→							
6.									←						→		
7.												←				→	
8.		←														→	

1.8 เครื่องมือและอุปกรณ์ที่ใช้

1. ฮาร์ดแวร์ จำนวน 2 เครื่อง

- เครื่องคอมพิวเตอร์ สำหรับเตรียมข้อมูลจากการวัดเปรียบเทียบสมรรถนะ TPC-H มีสมรรถนะดังนี้

CPU : Pentium Celeron 1.69 GHz

HDD : 40 GB.

RAM : 256 MB.

- เครื่องคอมพิวเตอร์ สำหรับพัฒนาและประเมินประสิทธิภาพของดัชนีบิตแมป มีสมรรถนะดังนี้

CPU : Pentium Celeron 1.69 GHz

HDD : 40 GB.

RAM : 384 MB.

2. ซอฟต์แวร์

- ระบบปฏิบัติการ : Linux Red Hat 9.0 สำหรับเตรียมข้อมูลจากการวัดเปรียบเทียบสมรรถนะ TPC-H
- ระบบปฏิบัติการ : Microsoft Windows XP สำหรับพัฒนาและประเมินประสิทธิภาพของดัชนีบิตแมป

- ซอฟต์แวร์ในการพัฒนาระบบ : ตัวแปลภาษาซี (C Compiler) สำหรับพัฒนาและประเมินประสิทธิภาพของดัชนีบิตแมป

1.9 ประโยชน์ที่คาดว่าจะได้รับ

ได้เทคนิคใหม่สำหรับการสร้างดัชนีบิตแมปที่สามารถลดพื้นที่ในการจัดเก็บดัชนี และลดเวลาที่ใช้ในการค้นหาข้อมูลในคลังข้อมูล