บทที่ 3

แนวความคิดและทฤษฎีที่เกี่ยวข้อง

3.1 ความหมายของไฮเพอร์เท็กซ์

Vannever Bush เป็นผู้ออกแบบระบบไฮเพอร์เท็กซ์เป็นคนแรกเมื่อปี 1945 โดยต้องการให้ไฮเพอร์เท็กซ์เป็นเพียงสภาพแวดล้อมสำหรับเชื่อมข้อมูลที่อยู่ตามตู้เก็บเอกสารต่างๆ เพื่อความง่ายต่อการค้นคว้า การวิจัย การแลกเปลี่ยนข้อมูลสำหรับทำโปรเจคต่างๆ เท่านั้น แต่ไม่ได้ตั้งชื่อเป็นไฮเพอร์เท็กซ์เช่นในปัจจุบันนี้ [วิโรจน์, 2534] ต่อมาในช่วงกลางของทศวรรษ 1960 Theodor Holm Nelson ได้สร้างและให้คำจำกัดความ "ไฮเพอร์เท็กซ์" ว่า "ไฮเพอร์เท็กซ์ คือ งานที่เขียนโดยไม่ต่อเนื่องกันเป็นลำดับ (nonsequential writing)" [Berk and Delvin, 1991]

ในปัจจุบันด้วยความก้าวหน้าทางด้านคอมพิวเตอร์ และเทคโนโลยีในการเก็บข้อมูล "ไฮเพอร์เท็กซ์" จึงเป็นทั้งเทคโนโลยี โครงสร้างข้อมูล และ ตัวเชื่อมโยงกับผู้ใช้ในการเขียนและการอ่านเอกสาร [Berk and Delvin, 1991] หรือเป็นแนวความคิดของการเข้าถึงและใช้ข้อมูลในลักษณะที่ไม่ใช่เชิงเส้น (Non-linear) โดยใช้ซอฟต์แวร์เชื่อมโยงระหว่างส่วนประกอบของเอกสาร [McGraw and McDaniel, 1989] จึงเห็นได้ว่าไฮเพอร์เท็กซ์เป็นข้อความ หรือเอกสารที่มีลักษณะพิเศษที่สามารถเข้าถึงข้อมูลผ่านทางไฮเพอร์สเปซ ซึ่งเป็นลักษณะเรขาคณิตหลายมิติ ด้วยเหตุนี้ไฮเพอร์เท็กซ์จึงเป็นเอกสารที่มีหลายมิตินั่นเอง [Bonura, 1994] ทั้งหมดที่ได้กล่าวมานี้ ทำให้กล่าวได้ว่าไฮเพอร์เท็กซ์มีความหมายหลายนัยทั้งนี้ขึ้นอยู่กับการมองว่ามองในระดับใด [วิลาส วุวงศ์, ปัญญาพล หอระตะ, 2533]

3.1.1 ความหมายในระดับนามธรรม

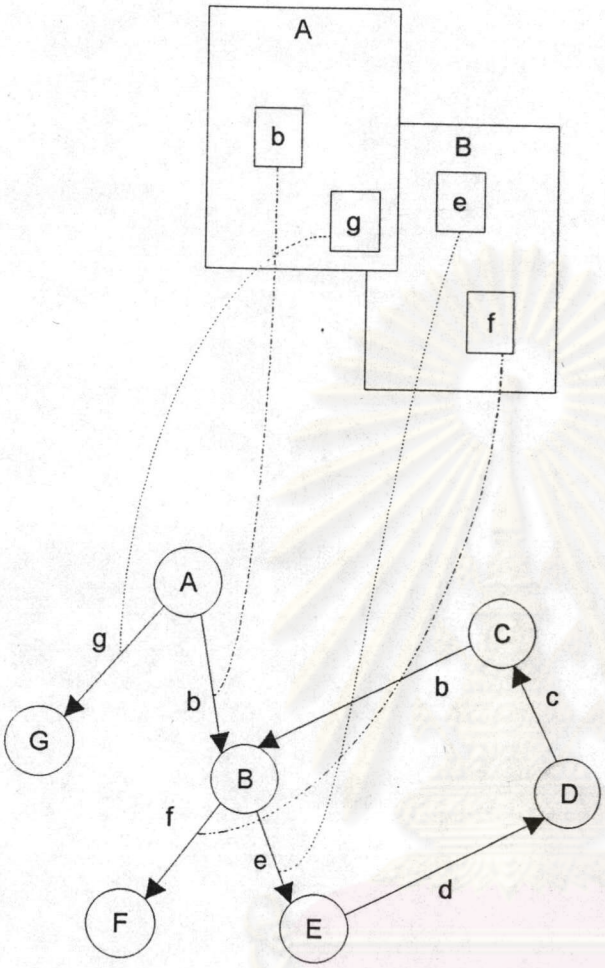
ในระดับนามธรรม ไฮเพอร์เท็กซ์หมายถึงการสร้าง (creation) การแสดง (representation) และการกำหนดความสัมพันธ์ของข้อมูล (define relationship) โดยที่ความสัมพันธ์ของข้อมูลเป็นการเชื่อมโยงระหว่างข้อมูลหลายชิ้น ข้อมูลอาจเป็นข้อความ (text) กราฟ (graphics) ภาพ (image) เสียง (audio) และ/หรือวีดีโอ ในแง่นี้ไฮเพอร์เท็กซ์เป็นรูปแบบการแสดงความรู้อย่างหนึ่งที่ใช้กันในการสร้างฐานข้อมูลของระบบสารสนเทศ

3.1.2 ความหมายในระดับรูปธรรม

ในระดับที่เป็นรูปธรรมมากขึ้นไฮเพอร์เท็กซ์เป็นระบบซอฟต์แวร์ที่ช่วยเก็บและค้นหา สารสนเทศ หรือข้อมูล โดยข้อมูลแต่ละชิ้นจะถูกเก็บเป็นโหนด และโหนดจะถูกเชื่อมโยงกันเป็นเครือข่ายด้วยเส้นเชื่อม การแสดงสารสนเทศของโหนดจะแสดงบนหน้าต่าง (window) 1 บานต่อโหนด การสืบค้นสารสนเทศในโหนดอื่น ๆ ทำได้โดยการวิ่งตามเส้นเชื่อมในเครือข่ายของโหนด

การเรียกโหนดอื่นขึ้นมาแสดงบนจอภาพ ทำได้โดยการเลื่อนเคอร์เซอร์หรือพอยเตอร์ไปยังคำหรือสัญลักษณ์ ที่ทำหน้าที่เป็นตัวเชื่อมบนหน้าต่าง ที่กำลังแสดงสารสนเทศของโหนดปัจจุบันอยู่ หลังจากนั้นถึงกดปุ่มบนคีย์บอร์ดหรือเม้าส์ดังแสดงในรูปที่ 3.1

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 3.1 แสดงความสัมพันธ์ระหว่างเครือข่ายของโหนดในฐานข้อมูลกับหน้าต่างบนจอภาพ

3.1.3 ความหมายในระดับของการประยุกต์ใช้

เมื่อมองไฮเพอร์เท็กซ์จากการประยุกต์ใช้ไฮเพอร์เท็กซ์ อาจเป็นโปรแกรมประมวลผลคำ (word processor) ชนิดหนึ่งที่อนุญาตให้ผู้เขียน หรือผู้พิมพ์บันทึกข้อมูลทั้งที่เป็นข้อความ รูปภาพ และ/หรือ เสียงได้ ซึ่งนำไปใช้ในงานเขียน โดยผู้เขียนไม่จำเป็นต้องเขียนเนื้อเรื่องต่อเนื่องเป็นแบบเชิงเส้น ผู้เขียนสามารถแบ่งเรื่องที่จะเขียนหรือความคิดที่จะเขียนออกเป็นหน่วยย่อยๆ ขนาด 1 ย่อหน้า แล้วบันทึกข้อมูล 1 ย่อหน้านั้นเข้าไปในโหนด โหนดเหล่านี้อาจไม่จำเป็นต้องต่อเนื่องกันเป็นลำดับ แต่อาจจะถูกเชื่อมโยงกันเป็นเครือข่าย นอกจากนั้นยังสามารถจัดลำดับการเชื่อมโยงกันใหม่ได้อีกด้วย [วิลาส วุวงศ์, ปัญญาพล หอระตะ, 2533]

จากทั้งหมดที่กล่าวมานี้ จะสรุปได้ว่าสารสนเทศไฮเพอร์เท็กซ์ มีความแตกต่างจากข้อความทั่วไปตรงที่มีความยืดหยุ่นและโครงสร้างซึ่งมีรายละเอียดฝังอยู่ในข้อความ สามารถเปลี่ยนจากเรื่องทั่วๆ ไปที่กล่าวถึงอย่างกว้างๆ ไปยังเรื่องเฉพาะที่ตรงจุดและมีรายละเอียดมากขึ้น

3.2 ส่วนประกอบของไฮเพอร์เท็กซ์

3.2.1 โหนด (Node) คือ หน่วยที่เล็กที่สุดของสารสนเทศในเอกสารไฮเพอร์เท็กซ์ หรือส่วนหนึ่งของเอกสารที่ครอบคลุมเนื้อหา 1 เรื่อง 1 รายการ หรือ 1 หัวข้อ โดย 1 โหนดอาจจะพอดีกับ 1 หน้าจอภาพ หรือมีขนาดเท่ากับ 1 คำ หรือมีขนาดเท่ากับหนังสือทั้งเล่ม โหนดหนึ่งๆ อาจจะเป็นสารสนเทศที่ประกอบด้วย ข้อความ, รูปภาพ และ/หรือข้อมูลในรูปแบบอื่น [Berk and Delvin, 1991] โหนดแต่ละโหนดเชื่อมโยงกันด้วยเส้นเชื่อม (Link) หรือตัวชี้ (Pointer) ไปยังโหนดอื่น หรือเนื้อหาที่ต้องการในเอกสารไฮเพอร์เท็กซ์ ในบางครั้งอาจจะใช้คำว่า article, topic, page, card, unit, chunk แทนโหนด

3.2.2 เส้นเชื่อม (Link) ซึ่งในบางครั้งใช้คำว่าตัวชี้ (Pointer) และในบางระบบเส้นเชื่อมคือคุณสมบัติ (Attribute) ของโหนด [Woodhead, 1991] นิยามของเส้นเชื่อมคือตัวชี้ (pointer) จากโหนดหนึ่งไปยังอีกโหนดหนึ่งซึ่งเก็บข้อมูลที่มีความสัมพันธ์เกี่ยวข้องกัน [Berk and Delvin, 1991] ถ้าเปรียบเทียบโหนดเป็นหน้ากระดาษในหนังสือเล่มหนึ่งๆ เส้นเชื่อมนี้อาจจะเปรียบได้กับข้อความเชิงอรรถที่อยู่บรรทัดล่างสุดของแต่ละหน้าเพื่อชี้แนะว่าถ้าต้องการอ่านเรื่องนั้นเรื่องนี้เพิ่มเติม ให้พลิกไปยังหน้า ... เส้นเชื่อมไม่ได้แสดงรายละเอียดของสารสนเทศ แต่เป็นเครื่องชี้นำว่า ถ้าจะอ่านรายละเอียดเรื่องนี้จะต้องกระโดดไปอ่านที่จุดใด ตำแหน่งใด [วิโรจน์, 2534] การเดินทางผ่านเส้นเชื่อมนี้จะทำได้โดยเลื่อนเคอร์เซอร์ไปยังคำหรือวลีที่มีการเน้นให้เป็นจุดเด่น, ปุ่ม ,ไอคอน หรือรูปภาพ ซึ่งเป็นจุดเริ่มต้นของเส้นเชื่อมแล้วคลิกเมาส์ หรือเคาะปุ่ม 'Enter' บนคีย์บอร์ดเป็นต้น

ความสามารถของระบบไฮเพอร์เท็กซ์จะขึ้นอยู่กับการเชื่อมโยงโหนดในระบบเข้าด้วยกัน ระบบไฮเพอร์เท็กซ์จะบรรลุจุดมุ่งหมายดังที่ผู้สร้างต้องการหรือไม่ ส่วนใหญ่ปัจจัยสำคัญจึงขึ้นอยู่กับการออกแบบเส้นเชื่อม

เส้นเชื่อมแบ่งได้เป็น 2 ประเภท คือ [วิโรจน์, 2534]

ก) เส้นเชื่อมทางอ้อม (Implicit Link) การเดินทางผ่านเส้นเชื่อมประเภทนี้ ไม่สามารถชี้ไปที่ข้อความนั้นแล้วเลือกโดยตรง จะต้องมีการกดปุ่มประกอบมากกว่า 1 ปุ่มขึ้นไปเสมอ ซึ่งโดยมากมักจะเป็นปุ่ม F1 หรือไม่ก็เป็นการคลิกเมาส์ตรงตำแหน่งนั้นอีกที เช่น ในระบบ Help ของเวิร์ดราชวิถี เมื่อต้องการดูว่า KR ในราชวิถีเอาไว้ทำอะไร ก็จะต้องกด Ctrl-K แล้วตามด้วยการกด F1 ซึ่งจะทำให้ Help เฉพาะปุ่ม Ctrl-K ปรากฏขึ้นมา เป็นต้น

ข) เส้นเชื่อมทางตรง (Explicit Link) การเดินทางผ่านเส้นเชื่อมประเภทนี้ สามารถชี้ไปที่ข้อความนั้นแล้วเลือกโดยตรง เช่นในระบบ Help ของซอฟต์แวร์ภาษาตระกูล Turbo ของ Borland และซอฟต์แวร์ ของ Microsoft เป็นต้น เส้นเชื่อมประเภทนี้ยังมีการแบ่งออกเป็นอีก 2 ประเภท คือ

1) เส้นเชื่อมทางตรงธรรมดา (Normal Explicit Links) ซึ่งสามารถเข้าถึงข้อมูลในโหนดต่างๆ ในระบบได้ไม่จำกัด และสามารถเข้าถึงได้อย่างไม่เป็นระเบียบ จึงมีความคล่องตัวสูง ความสัมพันธ์โดยมากเป็น one-to-many ข้อเสียคือ ถ้ามีโหนดจำนวนมาก จะทำให้ความเร็วในการเข้าถึงข้อมูลหรือเข้าถึงโหนดช้าลงอย่างมาก และต้องเปลืองเนื้อที่สำหรับเก็บค่าสตริงหรือตัวอักษรที่เก็บความสัมพันธ์ของแต่ละโหนด

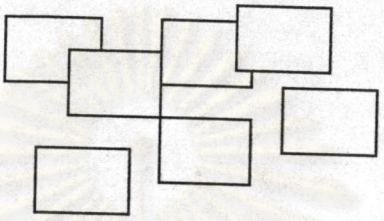
2) เส้นเชื่อมทางตรงที่จำกัดในท้องถิ่น (Local Explicit Links) ซึ่งจำกัดท้องถิ่นในการเข้าออกจากโหนดหนึ่งๆ เส้นเชื่อมประเภทนี้มักจะมีข้อกำหนดที่เข้มงวดในการเข้าออกจากโหนดหนึ่งไปยังอีกโหนดหนึ่ง แต่นั่นก็ทำให้ระบบรู้ตำแหน่งที่แน่นอนของโหนด ผลก็คือการเข้าถึงข้อมูลเป็นไปด้วยความเร็วสูงมากกว่า ที่สำคัญก็คือทำให้ประหยัดเนื้อที่สำหรับเก็บค่าสตริงหรือตัวอักษรอีกด้วย

3.3 โครงสร้างของไฮเพอร์เท็กซ์

ระบบไฮเพอร์เท็กซ์มีประโยชน์สำหรับงานจัดเก็บเอกสารที่มีความสัมพันธ์ต่อเนื่องกัน ซึ่งการจัดทำระบบดังกล่าวให้ได้ผลดีนั้น นอกจากจะต้องมีระบบซอฟต์แวร์และฮาร์ดแวร์ที่ดีแล้ว ต้องมีการกำหนดความสัมพันธ์ระหว่างโหนดโดยผ่านเส้นทางต่างๆ ที่จะสามารถนำเสนอและสื่อให้ผู้อ่านเข้าใจในเนื้อหาอย่างดีด้วย ดังนั้นโครงสร้างการจัดลำดับของโหนดจึงต้องนำมาพิจารณาในการพัฒนาระบบ

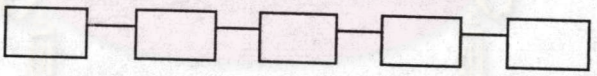
โครงสร้างของไฮเพอร์เท็กซ์สามารถแบ่งได้เป็น 3 ประเภทดังนี้ [บัณฑิต, 2534]

3.3.1 โครงสร้างฐานข้อมูลไฮเพอร์เท็กซ์ที่มีการจัดลำดับโหนดแบบสุ่ม (Random Hypertext or Unstructured Hypetext) เป็นโครงสร้างที่โหนดสามารถปรับปรุงเปลี่ยนแปลงตำแหน่งและลำดับได้อย่างรวดเร็ว โดยการใช้เทคนิคการจัดลำดับที่เป็นมาตรฐาน เช่นสูตรการคำนวณ เป็นต้น



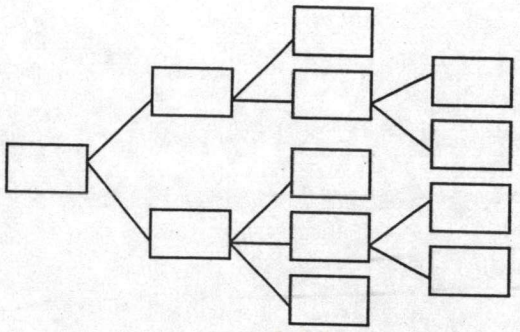
รูปที่ 3.2 แสดงโครงสร้างฐานข้อมูลไฮเพอร์เท็กซ์ที่มีการจัดลำดับโหนดแบบสุ่ม

3.3.2 โครงสร้างฐานข้อมูลไฮเพอร์เท็กซ์ที่มีการจัดลำดับโหนดแบบอนุกรม (Sequential Hypertext) เป็นโครงสร้างที่โหนดมีการจัดลำดับเป็นอนุกรมเรียงจากต้นไปท้าย โดยไม่มีการแตกกิ่งก้านสาขาออกไป และมักจะไม่เปลี่ยนแปลง ตัวอย่างเช่น เรียงความ ซึ่งจะต้องอ่านจากต้นไปจนจบ



รูปที่ 3.3 แสดงโครงสร้างฐานข้อมูลไฮเพอร์เท็กซ์ที่มีการจัดลำดับโหนดแบบอนุกรม

3.3.3 โครงสร้างฐานข้อมูลไฮเพอร์เท็กซ์ที่มีการจัดลำดับโหนดแบบลำดับขั้น (Hierarchical Hypertext) เป็นโครงสร้างที่โหนดมีการจัดลำดับเป็นรูปร่างคล้ายกับกิ่งก้านสาขาของต้นไม้ ซึ่งทำให้บางครั้งเรียกโครงสร้างนี้ว่า "โครงสร้างแบบ Tree" โครงสร้างแบบนี้ข้อมูลจะถูกเรียงลำดับและตามหัวข้อ โดยในแต่ละหัวข้อก็สามารถแบ่งย่อยออกไปได้อีกทีหนึ่ง ตัวอย่างเช่น คู่มือทางการแพทย์



รูปที่ 3.4 แสดงโครงสร้างฐานข้อมูลไฮเพอร์เท็กซ์ที่มีการจัดลำดับโหนดแบบลำดับขั้น

3.4 สถาปัตยกรรมของไฮเพอร์เท็กซ์

3.4.1 การจัดระบบไฮเพอร์เท็กซ์ (Hypertext Organization)

ระบบไฮเพอร์เท็กซ์โดยทั่วไปเริ่มต้นจากโครงสร้างพื้นฐาน 3 ประเภทที่ได้กล่าวมาแล้ว ใน 3.3 ในกรณีที่ใช้เอกสารที่มีอยู่แล้วเป็นแหล่งที่มา โครงสร้างของระบบไฮเพอร์เท็กซ์ จะขึ้นอยู่กับเอกสารต้นกำเนิด ดังนั้นผู้พัฒนาจะสามารถเปลี่ยนแฟ้มเอกสารธรรมดาไปเป็นแฟ้มเอกสารไฮเพอร์เท็กซ์ได้ โดยคงโครงสร้างทางตรรกศาสตร์ (logical pattern) ของความสัมพันธ์หลัก และความสัมพันธ์ย่อยระหว่างส่วนต่างๆ ในเอกสารไว้ตามเดิม ดังนั้นงานที่ต้องทำคือกำหนดปริมาณข้อมูลที่จะบรรจุลงใน 1 โหนด อย่างไรก็ดี เมื่อใช้สารสนเทศประเภทไม่มีโครงสร้างเป็นแหล่งที่มา ผู้พัฒนาจะต้องออกแบบโครงสร้างใหม่โดยเพิ่มเส้นเชื่อมระหว่างโหนด เพื่อสร้างความสัมพันธ์ระหว่างแต่ละโหนดขึ้นมา

3.4.2 การท่องไปในระบบไฮเพอร์เท็กซ์ (Hypertext System Navigation)

วิธีที่ผู้ใช้ระบบจะท่องไปในระบบไฮเพอร์เท็กซ์ ทำได้โดยใช้ระบบดรรชนีหรือระบบสืบค้นข้อมูลที่ระบบมีอยู่ เทคนิคการสืบค้นข้อมูลส่วนใหญ่ใช้บัญชีศัพท์ หรือรายการคำหลัก (keyword list) ซึ่งไฮเพอร์เท็กซ์โหนดแต่ละโหนดจะถูกกำหนดคำที่ใช้เป็นดรรชนี หรือคำหลัก บางระบบจะใช้รูปแบบของเส้นเชื่อมทั้งชนิดเส้นเชื่อมทั่วไปและเส้นเชื่อมที่จำกัดท้องถิ่น ซึ่งระบบหลังนี้จะให้ความยืดหยุ่นมากกว่า

I17478510

3.4.3 โปรแกรมเลือกอ่าน (Hypertext Browsers)

โปรแกรมเลือกอ่านเป็นเครื่องมือที่ช่วยในการท่องไปในระบบไฮเพอร์เท็กซ์นี้มีประสิทธิภาพ เพราะทำให้ผู้ใช้ระบบเห็นโครงสร้างของระบบที่ใช้และทำให้การเข้าถึงข้อมูลทำได้สะดวกรวดเร็ว ไม่หลงทาง

โปรแกรมเลือกอ่านอาจจะประกอบด้วยฟอร์มรายการโหนดที่เคยผ่านหรือแผนที่ของส่วนประกอบในระบบซึ่งทำให้เห็นภาพรวมของระบบเป็นต้น

3.4.4 โปรแกรมกรองข้อมูล (Information Filters)

โปรแกรมกรองข้อมูลทำหน้าที่ต่างจากโปรแกรมเลือกอ่านเล็กน้อย คือจำกัดข้อมูลที่ต้องการสืบค้น จุดประสงค์ของโปรแกรมกรองข้อมูลคือทำให้รูปแบบการสืบค้นใกล้เคียงกับความต้องการของผู้ใช้ระบบ นั่นคือผู้ใช้สามารถจำกัดหรือขยายวงการสืบค้นได้โดยใช้ระบบกลไกการกรองข้อมูล

3.4.5 โปรแกรมจำตำแหน่ง (Bookmarks)

โปรแกรมจำตำแหน่งหรือโปรแกรมคั่นหนังสือนี้ เป็นอีกเครื่องมือที่ใช้ช่วยในการท่องไปในระบบไฮเพอร์เท็กซ์ คือช่วยจำตำแหน่งในเอกสารอิเล็กโทรนิกส์ซึ่งต้องการมา "เยี่ยม" บ่อยๆ แต่อย่างไรก็ดีการที่ใช้ที่คั่นหนังสือนี้มากเกินไป จะทำให้ความสำคัญของที่คั่นหนังสือนี้ลดลงได้ เนื่องจากทำให้ผู้ใช้จำตำแหน่งได้แต่อาจจะสับสนว่าต้องการจำตำแหน่งไว้เพื่ออะไร เนื่องจากมีตำแหน่งที่ต้องจำมากเกินไปนั้นเอง [Bielawski, 1996]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.5 ลักษณะในการเข้าถึงข้อมูล

การเข้าถึงข้อมูลเป็นกิจกรรมที่ผู้ใช้ระบบติดตามสารสนเทศที่ต้องการโดยใช้กลไกการเชื่อมโยงเป็นตัวช่วย

ซอฟต์แวร์ที่ใช้ในการสร้างเอกสารไฮเพอร์เท็กซ์ต้องมีคุณสมบัติในการสร้างดรรชนีของเรื่อง หรือรูปภาพ และสามารถสร้างตัวเชื่อมเมื่อมีการเพิ่มข้อมูล [Martin, 1990] นอกจากนี้ยังมีคุณสมบัติในการเป็นเครื่องมือในการเข้าถึงข้อมูลที่ต้องการ

3.5.1 การอ่าน (reading) คือการอ่านแบบธรรมดาบรรทัดต่อบรรทัด จากหน้าแรกถึงหน้าสุดท้าย

3.5.2 การเลือกอ่าน (browsing) คือการอ่านข้ามจากหน้าหนึ่งไปยังอีกหน้าหนึ่ง โดยจะอ่านเพียงบางส่วนในหน้าหนึ่งๆเท่านั้น

3.5.3 การสืบค้น (searching) คือการเข้าถึงข้อมูลที่ต้องการโดยรู้ตำแหน่งที่ข้อมูลนั้นเก็บอยู่ [Rada, 1991]

3.6 แฟ้มข้อมูล Rich Text Format

แฟ้มข้อมูล Rich Text Format หรือ RTF File จัดอยู่ในประเภทเอกสารที่มีข้อความแทรกด้วยคำสั่งพิเศษ (text markup language) ซึ่งจะทำให้ทั้งข้อความและรูปลักษณ์ของตัวข้อความถูกแทนที่ด้วยตัวอักษร ASCII ที่เรียงต่อ ๆ กัน (ไม่เหมือนกับเอกสารเวิร์ดโปรเซสเซอร์ ซึ่งจะประกอบด้วยข้อความที่มีการสอดแทรกด้วยข้อมูลในการควบคุมที่เป็นเลขฐาน 2) ซึ่งอักษรที่เรียงต่อๆ กันนั้นประกอบด้วยคำสั่งพิเศษผสมกับข้อความทั่ว ๆ ไป หลาย ๆ สิ่งในแฟ้มจำเป็นต้องถูกตัดทิ้งไป ไม่ว่าจะเป็นเส้น, ย่อหน้า และการแบ่งหน้า จะถูกควบคุมด้วยคำสั่งของ RTF เหมือนกับการกำหนดรูปแบบตัวอักษร, สี, กั้นหน้า-กั้นหลัง เป็นต้น คำสั่ง RTF จะขึ้นต้นด้วยตัวอักษร " \ " เสมอ และกลุ่มข้อความที่จะกำหนดรูปแบบพิเศษจะอยู่ในเครื่องหมายปีกกา " { " และ " } " ขอยกตัวอย่าง เช่นประโยคต่อไปนี้ "The last word of this paragraph sentence is in **bold face**." จะถูกแทนด้วยรูปแบบ RTF ด้วยอักษรที่มาเรียงต่อกันดังนี้ " \pard\plain\f4\fs20 The last word of this paragraph sentence is in {\f4\b bold face}. \par "

โครงสร้างทั้งหมดของแฟ้มข้อมูล RTF นี้มีลักษณะง่าย ๆ คือ เริ่มจากส่วนหัว (header) ของแฟ้มซึ่งเป็นคำสั่งพิเศษของ RTF ตามด้วยข้อความ (body) ในแฟ้มข้อมูล

ส่วนหัว ประกอบด้วยตัวระบุว่าเป็นแฟ้มข้อมูล RTF คือ "\rtf1" ตามด้วยองค์ประกอบต่าง ๆ เช่น ชุดตัวอักษรที่ใช้ในแฟ้ม ตารางตัวอักษรและตารางสี ซึ่งเป็นความสัมพันธ์ที่แทนการผสมสีแบบ RGB และอาจจะมีตารางของรูปแบบอื่น ๆ อีกต่อจากตารางสี ตัวอย่างเช่น {\rtf1\ansi \deff0\deflang1024

```
{\fontbl
{\f0\froman Times New Roman;}
{\f1\froman Symbol;}
{\f2\fswiss Arial;}
{\f3\fswiss Helvetica;}
{\f4\fswiss Hel;}
}
{\colortbl;
\red0\green0\blue0;
\red0\green0\blue255;
\red0\green255\blue0;
\red255\green0\blue0;
\red255\green255\blue255;
}
```

ส่วนข้อความ ประกอบด้วยคำสั่งและข้อความที่มาต่อ ๆ กัน และลงท้ายด้วยเครื่องหมายปีกกา การใช้คำสั่งกำหนดรูปแบบใด ๆ ที่จะประกาศอยู่ในส่วนนี้ จะขึ้นต้นและลงท้ายด้วยเครื่องหมายปีกกา การประกาศรูปแบบใดๆ ที่ไม่ได้ระบุไว้ระหว่างเครื่องหมายปีกกา จะเป็นการกำหนดรูปแบบให้กับข้อความแรกที่พบ จนถึงคำสั่งที่เปลี่ยนกำหนดรูปแบบใหม่

RTF นั้นมีการจัดรูปแบบตัวอักษรที่เพียงพอ แต่ไม่ได้ให้โครงสร้างและการจัดระเบียบในเอกสารแบบ SGML (Standard Generize Markup Language) หรืออีกนัยหนึ่งก็

คือ RTF ไม่ได้คำนึงถึงเนื้อหาในเอกสาร และไม่สามารถอธิบายถึงความสัมพันธ์ระหว่างออบเจกต์ที่ต่างกันในเอกสาร [Duncan, 1995]

คำสั่งพิเศษที่ใช้ในแฟ้มข้อมูล RTF สามารถแบ่งได้เป็น

3.6.1 ชุดตัวอักษร (Character Set)

กลุ่มคำสั่งพิเศษกำหนดชนิดตัวอักษรที่สามารถใช้ได้ในแฟ้มข้อมูล เช่น Windows ANSI เป็นต้น

3.6.2 ตารางตัวอักษร (Font Table)

กลุ่มคำสั่งพิเศษกำหนดกลุ่มตัวอักษรทั้งหมดที่ใช้ในแฟ้มข้อมูล

3.6.3 ตารางสี (Color Table)

กลุ่มคำสั่งพิเศษกำหนดสีทั้งหมดที่ใช้ในแฟ้มข้อมูล

3.6.4 รูปแบบของเอกสาร (Document Formatting)

กลุ่มคำสั่งพิเศษกำหนดรูปแบบของเอกสาร เช่น ความกว้างและความยาวของกระดาษ ช่วงเว้นห่างระหว่างขอบกระดาษ เป็นต้น

3.6.5 รูปแบบของส่วนย่อยในเอกสาร (Section Formatting)

กลุ่มคำสั่งพิเศษกำหนดรูปแบบของส่วนย่อยในเอกสาร (กลุ่มของย่อหน้า) เช่น จำนวนคอลัมน์ ช่องห่างระหว่างคอลัมน์ รูปแบบของเลขที่หน้า เบรคโค้ด เป็นต้น

3.6.6 รูปแบบของย่อหน้า (Paragraph Formatting)

กลุ่มคำสั่งพิเศษกำหนดรูปแบบของย่อหน้า เช่นการจัดย่อหน้า (ชิดซ้าย ชิดขวา หรืออยู่ตรงกลาง) การวางแท็บ เป็นต้น

3.6.7 รูปแบบทั่วไป (General Formatting)

กลุ่มคำสั่งพิเศษกำหนดกลุ่มว่าเป็น กลุ่มเชิงอรรถ กลุ่มตาราง กลุ่มรูปภาพ เป็นต้น

3.6.8 รูปแบบตัวอักษร (Character Formatting)

กลุ่มคำสั่งพิเศษกำหนดตัวอักษรหนา เอียง ขีดเส้นใต้ ขีดเส้นทับ เป็นต้น [Andlerigh, 1996]

ในการสร้างเส้นเชื่อมภายในเอกสารอิเล็กทรอนิกส์ไฮเพอร์เท็กซ์ต้องใช้คำสั่งพิเศษต่อไปนี้คือ

- **Hot Spots** เป็นกลุ่มคำสั่งพิเศษที่ประกาศบริเวณที่ผู้ใช้สามารถเลือก (เช่นใช้เมาส์คลิก) เพื่อกระโดดไปยังหัวข้ออื่น หรือแสดงหัวข้ออื่นในหน้าต่างป๊อปอัพ

รูปแบบ {\uldb *hot-spot text* }{\v *context-string*}

\uldb คือคำสั่งกำหนดรูปแบบตัวอักษรข้อความที่อยู่ตามมา ให้มีเส้นใต้ 2 เส้น

hot-spot text คือข้อความหรือรูปภาพที่ผู้ใช้สามารถเลือกเพื่อกระโดดไปยังหัวข้ออื่น หรือจัดว่าเป็นต้นทางของเส้นเชื่อม

\v คือคำสั่งกำหนดรูปแบบตัวอักษรข้อความที่อยู่ตามมา ให้เป็นข้อความที่ไม่ต้องแสดง (hidden text)

context-string คือข้อความที่เป็นเชิงอรรถประเภท context string ของหัวข้อที่เป็นปลายทางของเส้นเชื่อม

- **Context Strings** เป็นคำสั่งพิเศษประเภทเชิงอรรถ ในแต่ละหัวข้อในแฟ้มข้อมูลจะมี context string ที่ไม่ซ้ำกันเป็นค่าเฉพาะสำหรับแต่ละหัวข้อ เพื่อประโยชน์ในการระบุปลายทางของเส้นเชื่อม

รูปแบบ #{\footnote context string}

คือสัญลักษณ์ของเชิงอรรถประเภท context string

\footnote คือคำสั่งกำหนดกลุ่มเชิงอรรถ

context string คือกลุ่มตัวอักษรตั้งแต่ 1 - 128 อักษร ประกอบด้วยอักษร A - Z ตัวเลข 0 - 9 และ จุด (.) หรือเส้นใต้ (_) โดยเขียนติดต่อกันไม่เว้นช่องว่าง