บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะอธิบายถึงชุดคำสั่ง 2 ชนิด คือ เอกซ์วิว (XVIEW) และฟิกส์ เอกซ์วิวจะใช้ในการทำตัวประสานกับผู้ใช้บนระบบเอกซ์วินโดว์ ส่วนฟิกซ์เป็นชุดคำสั่งกราฟิกที่ใช้ในการทำวิจัยนี้

เอกซ์วิว (XVIEW - X Window-System-Based Visual/Integrated Environment for Workstations) (Heller, 1991)

เอกซ์วิว คือ เครื่องมือที่ถูกพัฒนาเพื่อใช้ในการสร้างตัวประสานกับผู้ใช้ (User Interface) แบบโอเพนลุกของระบบเอกซ์วินโดว์ โดยมีเอกซ์ลิบซึ่งเป็นชุดคำสั่งขั้นต่ำสุด (Lowest Level Library) ของวินโดว์เป็นฐาน

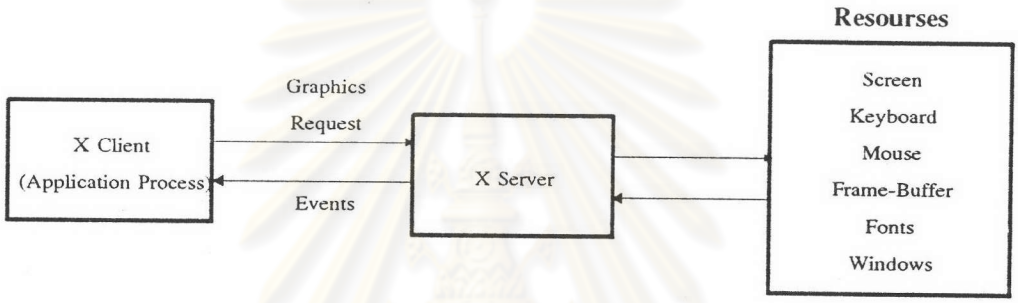
1. ระบบเอกซ์วินโดว์

ระบบเอกซ์วินโดว์หรือเอกซ์ เป็นระบบวินโดว์แบบไคลแอนท์เซิฟเวอร์ (Client-Server) ที่ใช้ในการควบคุมการแสดงผลบนจอภาพ เอกซ์จะแบ่งจอภาพออกเป็นส่วนนำเข้าและส่วนนำออกหลายส่วน เรียกว่า วินโดว์ แต่ละส่วนจะทำหน้าที่เป็นเครื่องปลายทางเสมือน (Virtual Terminal) ที่ไม่ขึ้นแก่กัน เครื่องปลายทางเสมือนเหล่านี้สามารถใช้กับโปรแกรมประยุกต์ที่แตกต่างกันได้ เช่น ตัวหนึ่งอาจใช้กับโปรแกรมประยุกต์ที่เป็นข้อความ อีกตัวหนึ่งอาจใช้กับโปรแกรมประยุกต์ที่เป็นกราฟิก เอกซ์จะรับข้อมูลเข้าบนจอภาพโดยอาศัยตัวชี้ (Pointer) ซึ่งมักถูกควบคุมโดยเมาส์ (Mouse) หรืออาจเป็นอุปกรณ์อื่นๆก็ได้ และสามารถรับข้อมูลเข้าจากแผงแป้นอักขระ (Keyboard) ได้เช่นกัน โปรแกรมประยุกต์หนึ่งของเอกซ์อาจใช้เพียง 1 วินโดว์ หรือหลายวินโดว์พร้อมๆ กันได้ และแต่ละวินโดว์ของเอกซ์อาจมีวินโดว์ของตัวเองอยู่ภายในอีกก็ได้ โดยจะเรียกวินโดว์แรกว่าเป็นวินโดว์แม่ (Parent Window) และวินโดว์ที่อยู่ภายในวินโดว์แม่เรียกว่า วินโดว์ลูก (Child Window)

เอกซ์ เป็นวินโดว์ที่ทำงานบนระบบเครือข่าย (Network) ได้ โดยมีโปรโตคอลของระบบเครือข่ายที่เรียกว่า เอกซ์โปรโตคอล (X Protocol) ทำให้เอกซ์สามารถทำงานบนเครื่องชนิดต่างๆ หรือระบบปฏิบัติการต่างๆ ได้ ซึ่งทำให้เอกซ์มีลักษณะที่ไม่ขึ้นกับอุปกรณ์ (Device-Independent) เอกซ์โปรโตคอลใช้ติดต่อระหว่างกระบวนการ (Process) 2 กระบวนการของเอกซ์

กระบวนการหนึ่ง คือ โปรแกรมประยุกต์ซึ่งเรียกว่า ไคลแอนต์ และอีกกระบวนการหนึ่งเรียกว่า เซิฟเวอร์ ซึ่งใช้ในการควบคุมทรัพยากร (Resource) ต่างๆของเอกซ์ เช่น จอภาพ แผงแป้นอักขระ และตัวชี้ เป็นต้น

เอกซ์เซิฟเวอร์จะต้องให้บริการกับไคลแอนต์จำนวนมากได้ โดยการแปลและปฏิบัติตามข้อความที่ถูกส่งมาจากไคลแอนต์ ข้อความนี้จะเรียกว่า รีเควสต์ (Request) เมื่อมีข้อมูลเข้ามา เอกซ์เซิฟเวอร์จะต้องส่งข้อความไปบอกไคลแอนต์ ข้อความนี้เรียกว่า อีเวนต์ (Event) และเนื่องจากโครงสร้างข้อมูลในเอกซ์มีจำนวนมาก ข้อมูลเหล่านี้จึงต้องมีเลขประจำตัวด้วย



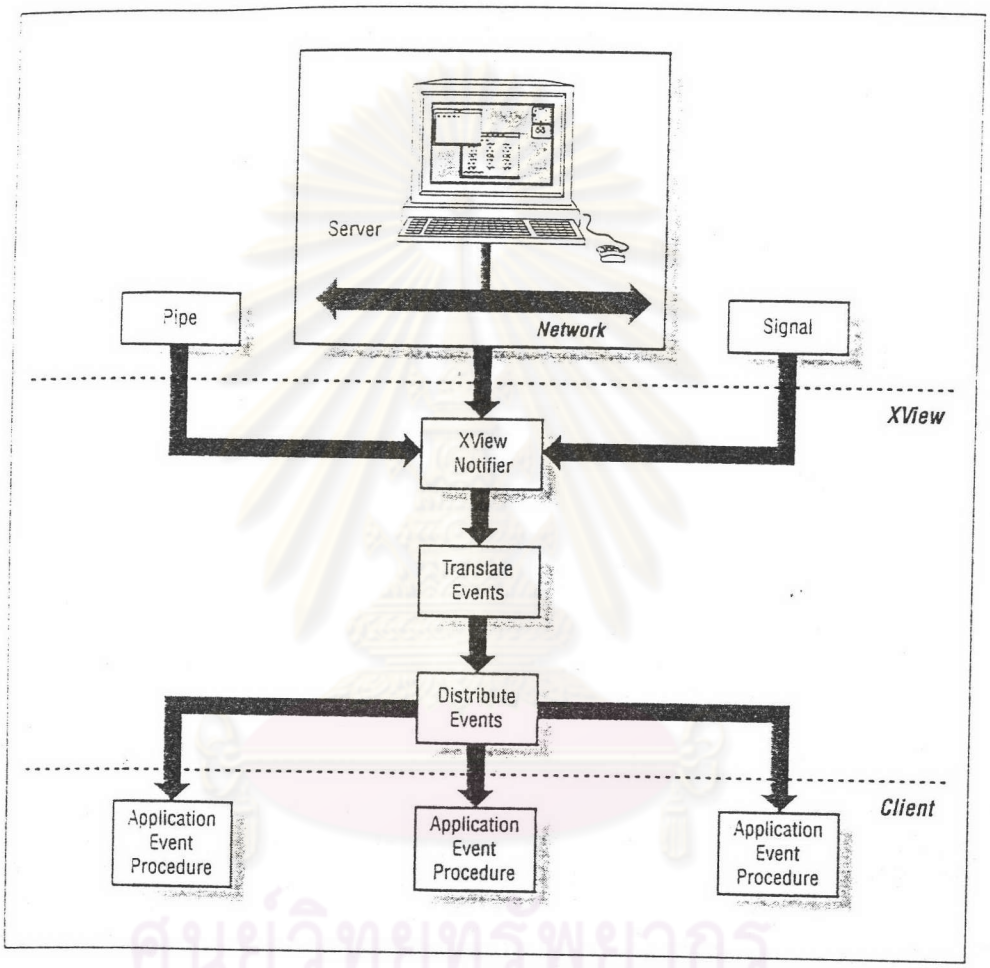
รูป 2.1 ความสัมพันธ์ระหว่างเอกซ์เซิร์ฟเวอร์กับโปรแกรมประยุกต์ (Mansfield, 1993)

เอกซ์ลิบ เป็นชุดคำสั่งของภาษาซีขั้นต่ำสุดของเอกซ์ ซึ่งทำหน้าที่แปลโครงสร้างข้อมูลและฟังก์ชันของภาษาซีไปเป็นเอกซ์โปรโตคอล เนื่องจากการใช้เอกซ์ลิบจะทำให้โปรแกรมใหญ่และยุ่งยาก จึงได้มีการพัฒนาทูลคิตเพื่อช่วยให้การสร้างตัวประสานกับผู้ใช้ของโปรแกรมประยุกต์ง่ายขึ้น ซึ่งทำโดยการจัดกลุ่มการทำงานพื้นฐานที่ประกอบด้วยฟังก์ชันของเอกซ์ลิบหลายฟังก์ชันให้เป็น 1 ฟังก์ชันของทูลคิต เอกซ์วิวก็เป็นทูลคิตหนึ่งในจำนวนหลายๆ ทูลคิตของเอกซ์

การติดต่อระหว่างโปรแกรมประยุกต์กับอุปกรณ์ (Device) จะต้องผ่านการควบคุมจากเอกซ์เซิร์ฟเวอร์ (X Server) เมื่อมีข้อมูลจากอุปกรณ์เข้ามา เอกซ์เซิฟเวอร์จะส่งข้อความที่เรียกว่าอีเวนต์ ไปให้โปรแกรมประยุกต์ เอกซ์อีเวนต์ (X Event) เป็นโครงสร้างข้อมูลที่ใช้เก็บข้อมูลสำคัญของเหตุการณ์ที่เกิดขึ้น อีเวนต์อาจเกิดจากระบบวินโดว์ ระบบปฏิบัติการ ข้อมูลที่ผู้ใช้ป้อนให้ หรือโปรแกรมประยุกต์ก็ได้ เช่น การกดปุ่มบนเมาส์จะทำให้เกิดอีเวนต์ขึ้น หรือการเลื่อนวินโดว์ไปมาบนจอภาพก็จะมีอีเวนต์เกิดขึ้นเช่นเดียวกัน เอกซ์เซิฟเวอร์จะมีหน้าที่ดูแลและแจกจ่ายอีเวนต์เหล่านี้ไปยังส่วนต่างๆ ที่เกี่ยวข้อง

ระหว่างเซิฟเวอร์กับโปรแกรมประยุกต์ของเอกซ์วิว จะต้องผ่านส่วนที่จัดการกับอีเวนต์ซึ่งเรียกว่า โนติฟายเออร์ (Notifier) เมื่อมีการสร้างวัตถุในเอกซ์วิวขึ้นมา วัตถุนั้นจะมีอีเวนต์ที่

เกี่ยวข้องมากมาย วิธีการที่ง่ายที่สุดในการจัดการกับอีเวนต์เหล่านี้ คือ ส่งการควบคุมอีเวนต์ไปให้เอกซ์วิว หลังจากนั้นเอกซ์วิวโนติฟายเออร์จะทำการแจกจ่ายอีเวนต์ไปยังวัตถุที่เหมาะสม ดังรูปที่ 2.2



รูป 2.2 ลักษณะการทำงานโดยอาศัยโนติฟายเออร์ (Heller, 1991)

ตัวประสานกับผู้ใช้แบบกราฟิกของระบบเอกซ์วินโดว์ที่นิยมใช้กันอย่างแพร่หลายมี 2 ชนิด คือ โอเอสเอฟ/โมทิฟ (OSF/Motif) ของโอเพน ซอฟแวร์ ฟาวด์เดชัน (Open Software Foundation) และโอเพนลุกของบริษัทซันไมโครซิสเต็มกับเอทีแอนด์ที (AT&T) สำหรับโอเพนลุกเป็นตัวประสานกับผู้ใช้แบบกราฟิกที่มีใช้ในระบบโอเพนวินโดว์ (Open Windows) ซึ่งเป็นระบบ

เอกซ์วินโดว์ของบริษัทซันไมโครซิสเต็ม โอเพนลุกมีทูลคิตที่สำคัญ 3 ชนิด คือ โอลิต (OLIT - OPEN LOOK Intrinsics Toolkit) ทีเอ็นที (tNt) และเอกซ์วิว

2. เอกซ์วิว

เอกซ์วิว เป็นทูลคิตที่ถูกพัฒนาขึ้นโดยบริษัทซันไมโครซิสเต็ม เป็นการพัฒนาขึ้นมาจากทูลคิตเดิมของบริษัทซันไมโครซิสเต็มที่ชื่อว่า ซันวิว (Sun View) วัตถุประสงค์แรกของการพัฒนาเอกซ์วิว คือ เพื่อให้ผู้ใช้ระบบซันวินโดว์ (Sun Window) ซึ่งเป็นระบบวินโดว์ของบริษัทซันไมโครซิสเต็ม สามารถเคลื่อนย้ายโปรแกรมไปยังระบบเอกซ์วินโดว์ได้ ในปัจจุบันเอกซ์วิวได้กลายเป็นส่วนหนึ่งของมาตรฐานของระบบเอกซ์วินโดว์แล้ว

เอกซ์วิวถูกใช้เพื่อทำให้การพัฒนาโปรแกรมภายใต้ระบบเอกซ์วินโดว์ง่ายขึ้น โดยที่เอกซ์วิวจะจัดเตรียมองค์ประกอบต่างๆ ของตัวประสานกับผู้ใช้ไว้ให้เรียกใช้งานได้ องค์ประกอบเหล่านี้จะมีลักษณะเป็นแบบโอเพนลุก ซึ่งเป็นตัวประสานกับผู้ใช้แบบกราฟิกชนิดหนึ่ง

2.1 ลักษณะทั่วไปของเอกซ์วิว

เอกซ์วิว เป็นทูลคิตที่มีลักษณะเชิงวัตถุ (Object-Oriented) โดยแต่ละส่วนของตัวประสานกับผู้ใช้จะถูกมองว่าเป็นวัตถุตัวหนึ่งของแพคเกจ (Package) ซึ่งแพคเกจแต่ละตัวจะมีชุดของคุณสมบัติที่ใช้ในการกำหนดวัตถุ ดังนั้นจึงสามารถสร้างตัวประสานกับผู้ใช้ได้โดยการเลือกใช้วัตถุจากแพคเกจที่มีอยู่

ลักษณะเชิงวัตถุของเอกซ์วิวที่สำคัญมี 4 อย่าง คือ

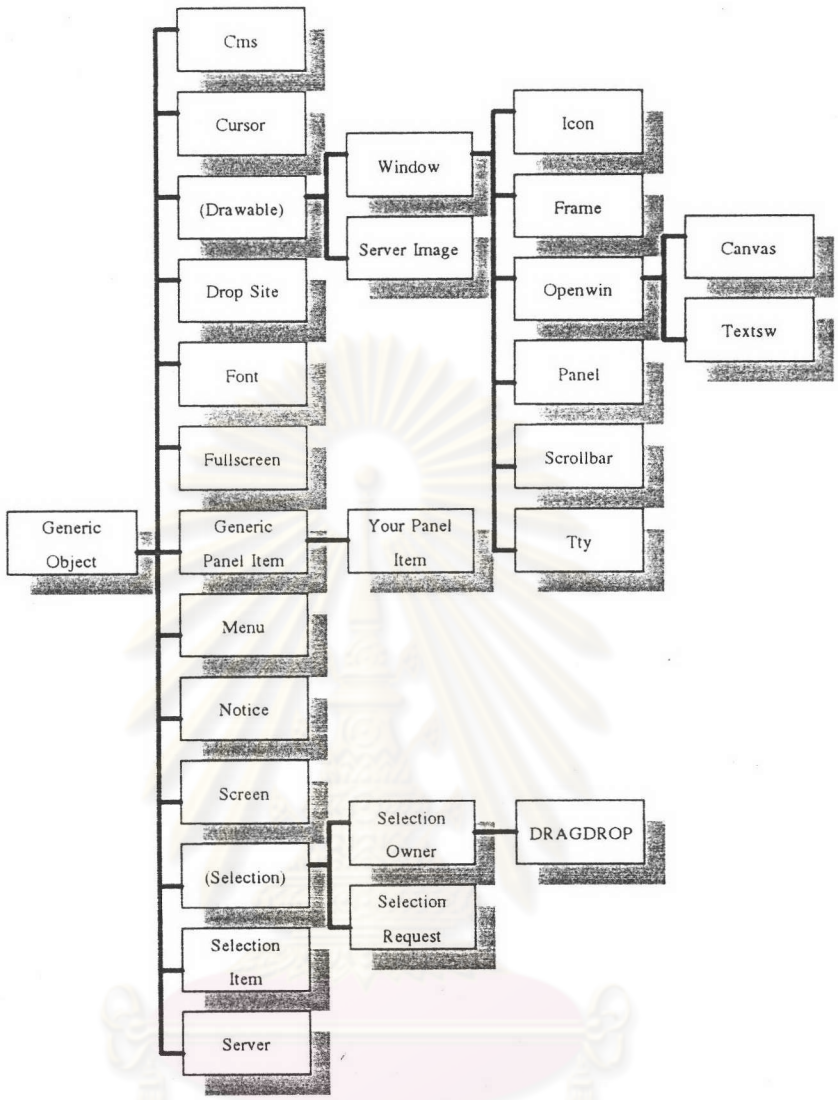
ก. วัตถุจะมีความสัมพันธ์กันในแบบลำดับชั้นของคลาส (Class Hierarchy)

ข. ชนิดของข้อมูลของวัตถุจะถูกปิดบัง (Opaque) ไว้

ค. ลักษณะประจำ (Attribute) ของวัตถุ จะถูกกำหนดได้โดยการส่งข้อความ (Message) ไปยังฟังก์ชัน

ง. การเรียกให้วัตถุทำงานจะต้องผ่านกระบวนการของอีเวนต์

เอกซ์วิวจะกำหนดคลาสของวัตถุแต่ละตัวในลักษณะลำดับชั้นรูปต้นไม้ (Tree Hierarchy) เช่น เฟรม จะเป็นคลาสย่อยของวินโดว์ซึ่งเป็นคลาสย่อยของดรอเอเบิล (Drawable) อีกที ดังรูปที่ 2.3



รูป 2.3 ความสัมพันธ์ของคลาสในเอกซ์วิว (Heller, 1991)

คลาสแต่ละคลาสจะมีลักษณะเฉพาะตัวที่ทำให้แตกต่างไปจากคลาสอื่น สำหรับเอกซ์วิวแล้วจะเรียกคลาสว่าแพคเกจ ซึ่งจะหมายถึง กลุ่มขององค์ประกอบของฟังก์ชันที่สัมพันธ์กัน อย่างไรก็ตามเอกซ์วิวจะมีแพคเกจบางตัวที่ไม่เป็นสมาชิกของลำดับชั้นของคลาส เช่น โนติฟายเออร์ แพคเกจ (Notifier Package)

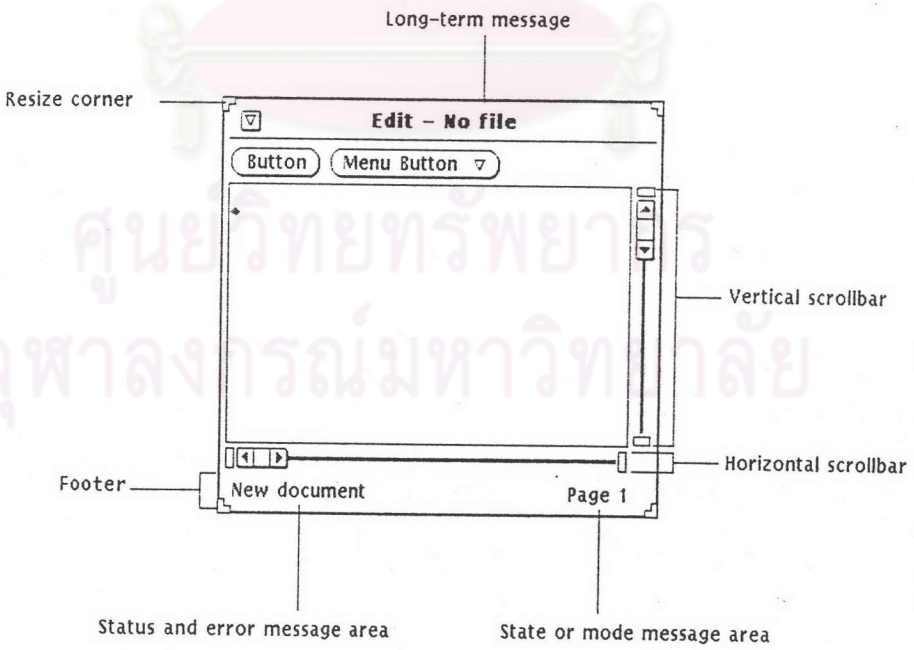
วัตถุของเอกซ์วิวมีทั้งชนิดที่มองเห็นได้ เช่น วินโดว์ แถบเลื่อน (Scrollbar) พาเนล (Panel) เป็นต้น และชนิดที่มองไม่เห็น เช่น เซิฟเวอร์ สกรีน (Screen) เป็นต้น วัตถุที่มองไม่เห็นเหล่านี้จะมีข้อมูลที่มีประโยชน์ต่อวัตถุที่มองเห็นได้ เช่น สกรีนจะมีข้อมูลขนาดของอุปกรณ์ พิกัดของจอภาพ เป็นต้น

วัตถุในเอกซ์วิวสามารถสืบทอดคุณสมบัติแบบลูกโซ่ (Chained Inheritance) ได้ กล่าวคือ วัตถุของแต่ละคลาสสืบทอดคุณสมบัติจากคลาสแม่ (Parent Class) หรือที่เรียกว่า ซุปเปอร์คลาส (Superclass) ได้

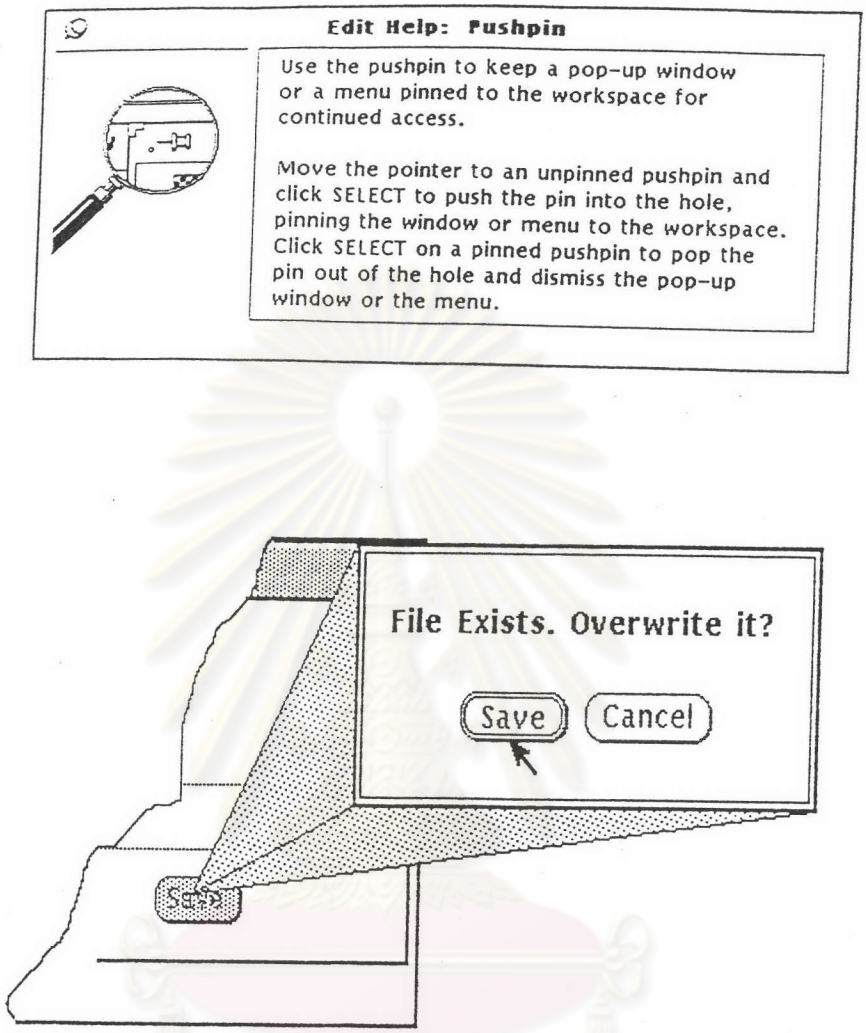
เมื่อมีการสร้างวัตถุขึ้น ฟังก์ชันที่ใช้สร้างวัตถุนั้นจะส่งค่าควบคุม (Handle) ของวัตถุกลับมา ซึ่งเปรียบเสมือนชื่อหรือตัวชี้ที่ชี้ไปยังวัตถุนั้น ค่าควบคุมนี้จะใช้ในการอ้างอิงถึงวัตถุเมื่อต้องการทำงานกับวัตถุนั้น เนื่องจากชนิดข้อมูลของเอกซ์วิวมีลักษณะของการซ่อนข้อมูล (Information Hiding) ดังนั้นค่าควบคุมจึงถูกปิดบังไว้ ทำให้ไม่สามารถมองเห็นข้อมูลที่แท้จริงในโครงสร้างข้อมูลของวัตถุได้

2.2 เฟรม (Frame) และ ซับเฟรม (Subframe)

เฟรม เป็นวินโดว์หลักที่ใช้บรรจุวินโดว์อื่น วินโดว์ที่อยู่ในเฟรมจะเรียกว่า ซับวินโดว์ (Subwindow) และมีเฟรมนั้นเป็นเจ้าของ (Owner) เฟรมมี 2 ชนิด คือ เฟรมฐาน (Base Frame) และเฟรมแบบผุดขึ้น (Pop-Up Frame) โปรแกรมประยุกต์จะใช้เฟรมฐานสำหรับการทำงานหลักของโปรแกรม เช่น เฟรมสำหรับวาดภาพ เฟรมสำหรับบรรณาธิกรณ์ข้อความ (Text Editor) เป็นต้น สำหรับเฟรมแบบผุดขึ้นจะใช้ในการทำงานเฉพาะอย่าง และจะหายไปเมื่อการทำงานนั้นจบลง เช่น เฟรมสำหรับการยืนยันการกดปุ่ม เฟรมช่วยเหลือ (Help Frame) สำหรับแสดงข้อความช่วยเหลือ เป็นต้น



รูป 2.4 เฟรมฐาน (Heller, 1991)



รูป 2.5 เฟรมแบบผุดขึ้น (Heller, 1991)

ซับเฟรม คือ เฟรมที่มีเจ้าของเป็นเฟรมอีกเฟรมหนึ่ง ซับเฟรมจะถูกควบคุมการทำงานโดยเฟรมที่เป็นเจ้าของ เช่น เมื่อปิดเฟรมที่เป็นเจ้าของแล้ว ซับเฟรมของเฟรมนั้นก็จะหายไปด้วย ข้อแตกต่างระหว่างซับเฟรมกับซับวินโดว์ คือ ซับเฟรมสามารถแสดงผลนอกขอบเขตของเฟรมที่เป็นเจ้าของได้ ในขณะที่ซับวินโดว์แสดงผลได้เฉพาะในขอบเขตของเฟรมเจ้าของเท่านั้น

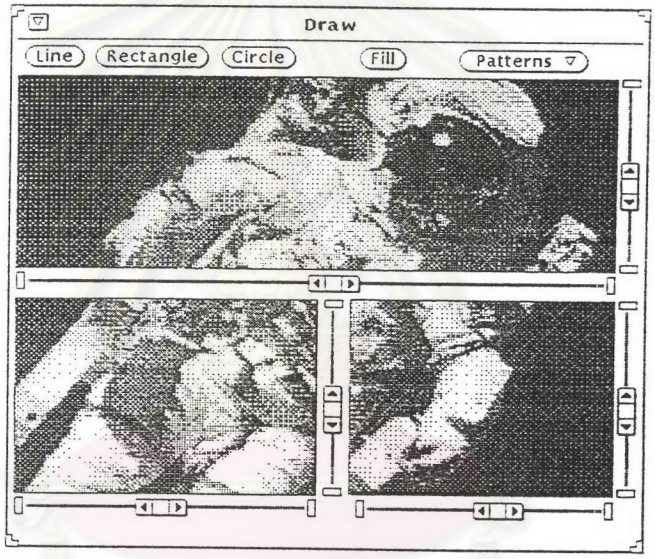
2.3 ซับวินโดว์

ซับวินโดว์จะต้องมีเจ้าของเสมอ ซึ่งเจ้าของนั้นอาจเป็นเฟรมหรือวินโดว์อื่นๆ แต่ซับวินโดว์จะไม่สามารถเป็นเจ้าของเฟรมได้ และซับวินโดว์จะมีขอบเขตการแสดงผลที่ถูกจำกัดอยู่

ในขอบเขตของเฟรมที่เป็นเจ้าของเท่านั้น ซึ่งแตกต่างจากซับเฟรม ซับวินโดว์มีหลายชนิด เช่น แคนวาส (Canvas) พาเนล (Panel) เป็นต้น

2.3.1 แคนวาส

แคนวาส (คือ ดิสเพล (Display) ในเอกซ์ลิบ) เป็นซับวินโดว์ของเฟรมซึ่งเป็นคลาสย่อยของแพคเกจชื่อโอเพนวิน (Openwin) โอเพนวินเป็นคลาสย่อยของวินโดว์ซึ่งมีลักษณะเป็นคลาสที่ถูกปิดบัง (Hidden Class) ซึ่งมีลักษณะสำคัญ คือ สามารถแบ่งแยกวิวได้ (Splittable Views) ทำให้แคนวาสสามารถแสดงวิว (View) ที่ต่างกันของภาพเดียวกันได้ ดังรูปที่ 2.6



รูป 2.6 การแบ่งแยกวิวของแคนวาส (Heller, 1991)

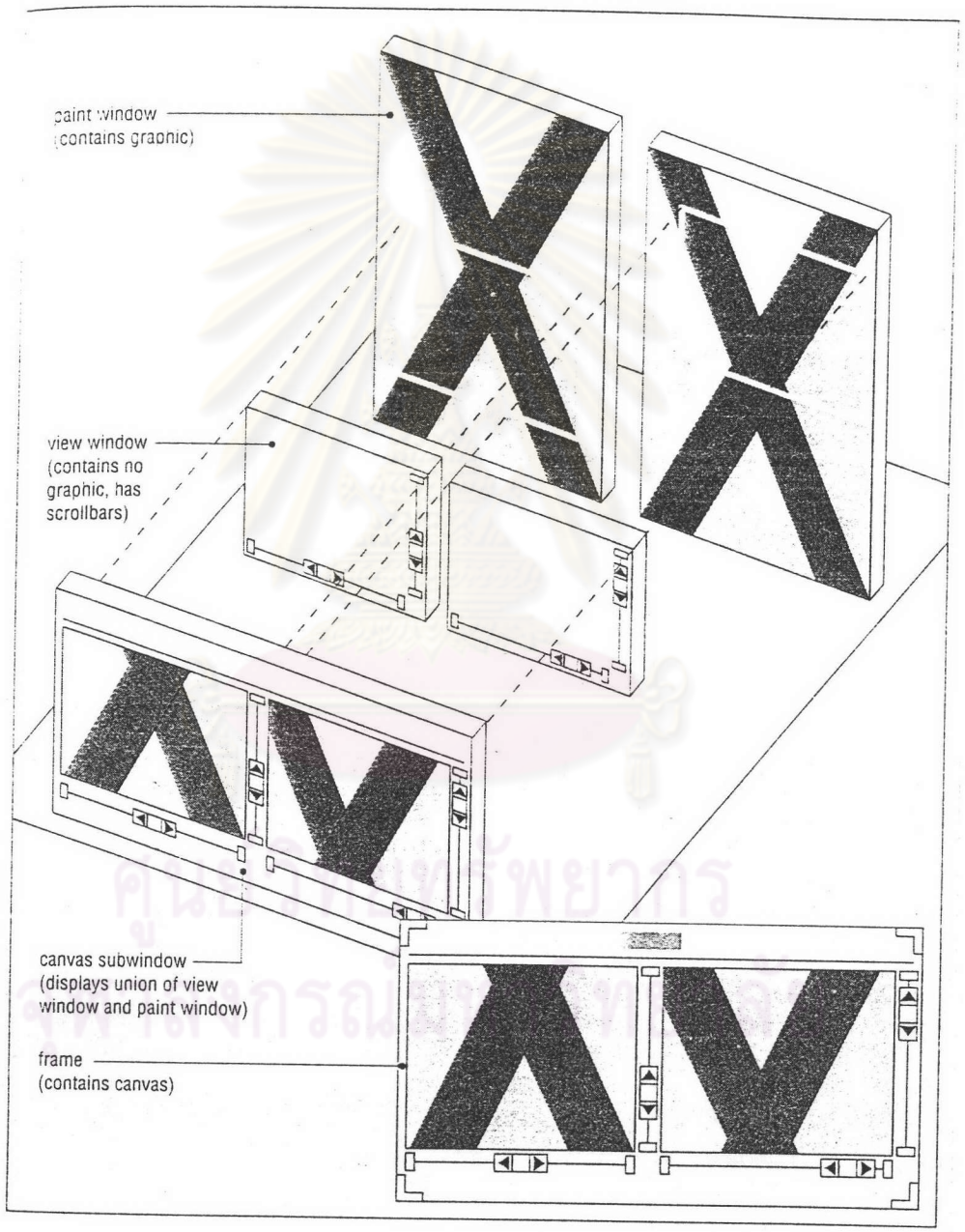
แคนวาสประกอบด้วยวินโดว์ 3 ชนิด คือ

2.3.1.1 แคนวาสซับวินโดว์ (Canvas Subwindow) จะมีเฟรมเป็นเจ้าของและเป็นอาณาเขตของการแสดงผลบนเฟรมนั้น

2.3.1.2 วิววินโดว์ (View Window) ใช้เป็นพื้นผิวของการแสดงภาพหรือบางส่วนของภาพที่เก็บอยู่ในเพ้นต์วินโดว์ (Paint Window) วิววินโดว์จะต้องอยู่ภายในขอบเขตของแคนวาสซับวินโดว์เท่านั้น ในแคนวาสซับวินโดว์หนึ่งอาจมีวิววินโดว์ได้มากกว่า 1 วิววินโดว์

2.3.1.3 เพ้นต์วินโดว์ ใช้เป็นที่เก็บข้อมูลภาพของวิววินโดว์ โดยวิววินโดว์แต่ละตัวจะมีเพ้นต์วินโดว์ได้เพียง 1 วินโดว์เท่านั้น การใช้เพ้นต์วินโดว์ทำให้ภาพจริงที่เก็บไว้สามารถมีขนาดใหญ่กว่าวิววินโดว์ได้ อีเวนต์ต่างๆ เกี่ยวกับรูปภาพจะเกิดขึ้นที่เพ้นต์วินโดว์นี้

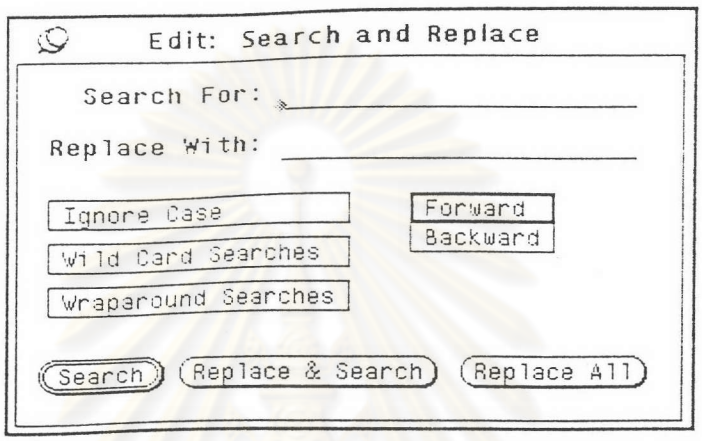
ความสัมพันธ์ของวินโดว์ทั้ง 3 ชนิดนี้สามารถแสดงให้เห็นได้ดังรูปที่ 2.7



รูป 2.7 ความสัมพันธ์ของวินโดว์ทั้ง 3 ชนิดของแคนวาส (Heller, 1991)

2.3.2 พาเนล

พาเนลเป็นซับวินโดว์ที่ใช้เป็นส่วนควบคุม (Control Area) โดยพาเนลจะเป็นสถานที่ที่ใช้แสดงตัวควบคุมต่างๆ เช่น ปุ่ม (Button) สไลเดอร์ (Slider) เป็นต้น ตัวควบคุมเหล่านี้เรียกว่า พาเนล ไอเท็ม (Panel Item)



รูป 2.8 ลักษณะของพาเนลและพาเนล ไอเท็ม (Heller, 1991)

2.3.3 รายการเลือก (Menu)

รายการเลือกโดยตัวเองแล้วไม่มีมีลักษณะเป็นวินโดว์ รายการเลือกจะถูกแสดงบนวินโดว์เฉพาะช่วงที่ถูกสั่งให้แสดงบนจอภาพเท่านั้น เอกซ์วิวมีรายการเลือก 3 ชนิด คือ

2.3.3.1 รายการเลือกแบบผุดขึ้น (Pop-Up Menu) จะแสดงเมื่อผู้ใช้กดปุ่มรายการเลือก (Menu Button) ในวินโดว์

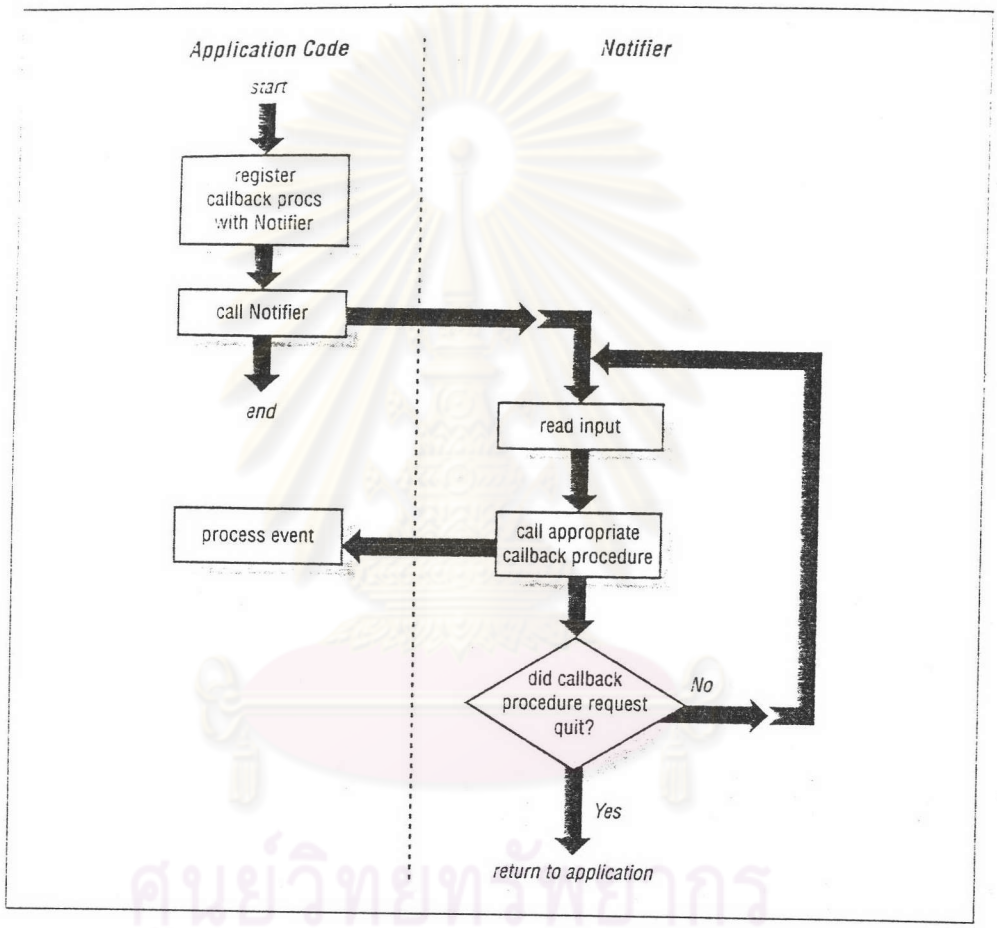
2.3.3.2 รายการเลือกแบบดึงขวา (Pull-Right Menu) จะแสดงรายการทางด้านขวาของรายการเลือก มักใช้เป็นรายการเลือกของรายการที่เรียกใช้

2.3.3.3 รายการเลือกแบบดึงลง (Pull-Down Menu) จะแสดงใต้ปุ่มรายการเลือกของพาเนล

2.4 โนติฟายเออร์

เอกซ์วิวทำงานในลักษณะโนติฟิเคชัน เบส (Notification-Based) เมื่อมีการรับข้อมูลเข้ามาจากระบบ โนติฟายเออร์จะแปลงข้อความนั้นเป็นอีเวนต์ในระดับที่สูงขึ้น และทำการแจกจ่ายไปยังแต่ละวัตถุของเอกซ์วิวที่เกี่ยวข้องกับข้อมูลเข้านี้

ในโปรแกรมประยุกต์ปกติ ส่วนควบคุมหลักจะอยู่ในตัวโปรแกรมประยุกต์เอง แต่การทำงานแบบโนติฟิเคชัน เบสนั้น ส่วนควบคุมหลักจะอยู่ในโนติฟายเออร์ ซึ่งโนติฟายเออร์จะอ่านอีเวนต์ที่เกิดขึ้นและแจ้งให้ทราบ (Notify) หรือเรียกกลับ (Call Out) ไปยังแต่ละกระบวนงาน (Procedure) ในโปรแกรมประยุกต์ที่ได้เรจิสเตอร์ (Register) ไว้กับโนติฟายเออร์ ดังรูปที่ 2.9



รูป 2.9 การทำงานแบบโนติฟิเคชันเบส (Heller, 1991)

โนติฟายเออร์ในเอกซ์วิว มี 2 ลักษณะ คือ

2.4.1 กระบวนงานแจ้งให้ทราบ (Notify Procedure) จะใช้กับวัตถุประเภทพาเนล เมื่อมีอีเวนต์ที่เกี่ยวข้องกับวัตถุนั้น โนติฟายเออร์จะเรียกกระบวนงานแจ้งให้ทราบของวัตถุนั้นมาตอบสนองต่ออีเวนต์นั้น เช่น เมื่อมีการกดปุ่มบันทึกข้อมูลบนพาเนล โนติฟายเออร์จะเรียกกระบวนงานที่ใช้บันทึกข้อมูลของโปรแกรมประยุกต์นั้นให้ทำงาน เป็นต้น

2.4.2 กระบวนงานอีเวนต์ (Event Procedure) จะใช้กับวัตถุประเภทแคนวาส ในกระบวนงานอีเวนต์นี้จะมีอีเวนต์ต่างๆ ผ่านเข้ามามากมาย เอกซ์วิวจะปล่อยให้โปรแกรมประยุกต์เลือกตอบสนองต่ออีเวนต์ที่ต้องการเอง

ฟิกส์ (PHIGS -Programmer's Hierarchical Interactive Graphics System)

ในหัวข้อนี้จะอธิบายถึงเรื่องของฟิกส์แต่พอสังเขป โดยแบ่งเป็น 3 ส่วน คือ ลักษณะทั่วไปของฟิกส์ แนวคิดหลักของฟิกส์ และความสัมพันธ์ของซันฟิกส์รุ่น 2.0 กับระบบเอกซ์วินโดว์ คำว่า "ฟิกส์" ที่กล่าวไว้ในการวิจัยนี้ หมายถึง ฟิกส์ซึ่งถูกกำหนดเป็นมาตรฐานโดยไอเอสโอ สำหรับคำว่า "ซันฟิกส์รุ่น 2.0" หมายถึง ชุดคำสั่งซันฟิกส์รุ่น 2.0 ของบริษัทซันไมโครซิสเต็ม ส่วนคำว่า "การทำให้เกิดผลของฟิกส์" คือ การนำเอาฟิกส์มาใช้สร้างชุดคำสั่งจริง เช่น ซันฟิกส์รุ่น 2.0 หรือเดคฟิกส์ (DEC PHIGS) เป็นต้น

1. ลักษณะทั่วไปของฟิกส์

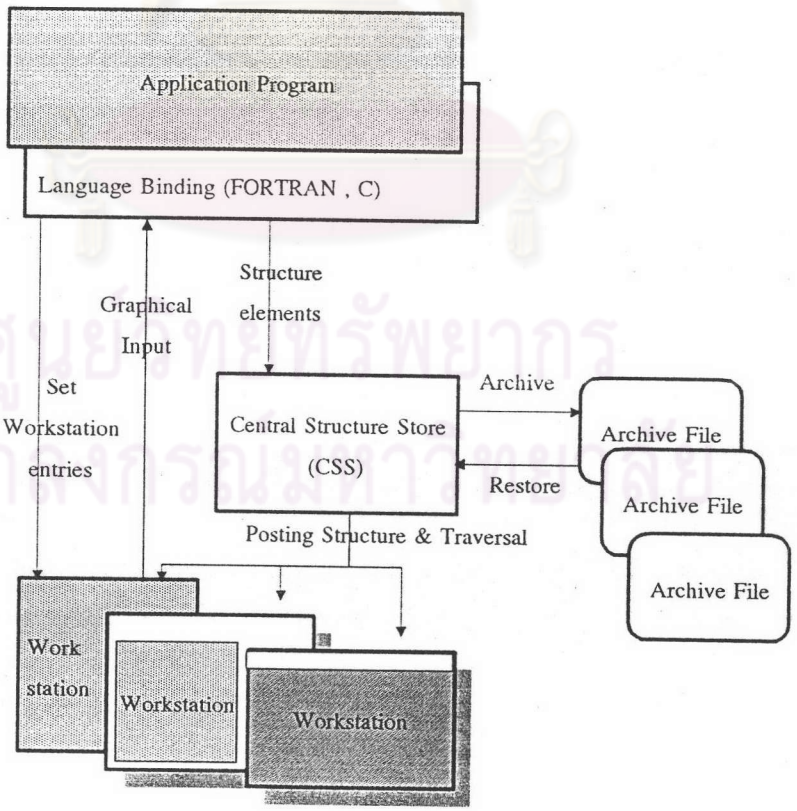
ฟิกส์เป็นชุดคำสั่งทางกราฟิกซึ่งถูกกำหนดเป็นมาตรฐานโดยไอเอสโอ ในปี 1988 และส่วนเพิ่มของฟิกส์ ที่เรียกว่า ฟิกส์พลัส (PHIGS PLUS) ถูกกำหนดเป็นมาตรฐานในปี 1991 คำสั่งมาตรฐานของฟิกส์ จะถูกกำหนดเป็นคำสั่งที่ไม่ขึ้นกับภาษาโปรแกรมใดๆ เลย สำหรับชุดคำสั่งของฟิกส์ที่มีใช้กันทั่วไป จะเป็นการนำฟิกส์มาตรฐานนี้ไปผนวกกับภาษาโปรแกรม เช่น ซี (C) ฟอร์แทรน (FORTRAN) เป็นต้น

ความสามารถทางกราฟิกของฟิกส์มีตั้งแต่การแสดงภาพรูปทรงกราฟิกง่ายๆ เช่น เส้นตรงหรือพื้นที่ และรูปทรงที่ซับซ้อนขึ้น เช่น เส้นโค้งบีสไปลน์แบบนอนยูนิฟอร์ม เรชันนัล (Non-Uniform Rational B-spline ย่อว่า NURB) ไปจนถึงการให้แสงและเชดสี ข้อมูลกราฟิกต่างๆ จะถูกเก็บลงในฐานข้อมูลกราฟิกของฟิกส์ ซึ่งเรียกว่า หน่วยเก็บโครงสร้างส่วนกลาง (Central Structure Store ย่อว่า CSS) ดังนั้นฟิกส์จึงเหมาะกับงานกราฟิกประเภทดิสเพลย์ ลิสต์ (Display List) เช่น งานประเภทแคด (CAD) มากกว่าประเภทภาวะทันที (Immediate Mode) เช่น ไฟลต์ซิมูเลเตอร์ (Flight Simulator)

เนื่องจากฟิกส์มีคุณสมบัติใช้ได้กับหลายระบบ (Portable) และไม่ขึ้นกับอุปกรณ์ (Device-Independent) ดังนั้นการกำหนดข้อมูลของฟิกส์ที่เกี่ยวข้องกับอุปกรณ์ มักจะเป็นข้อมูลที่เป็นกลาง เช่น การใช้นอร์มัลไลซ์ โปรเจกชัน โคออดิเนต (Normalized Projection Coordinate ย่อว่า NPC) กำหนดขนาดการแสดงภาพในสถานที่แสดงภาพ เป็นต้น และการอ้างถึงอุปกรณ์ต่างๆ จะเป็นแบบอุปกรณ์เชิงตรรกะ (Logical Device) เช่น อุปกรณ์นำเข้าของฟิกส์ประเภทโลเคเตอร์

(Locator) สโตรก (Stroke) และพิก (Pick) อาจใช้อุปกรณ์จริง (Real Device) เป็นเมาส์ตัวเดียวก็ได้ หรืออุปกรณ์เกี่ยวกับการแสดงภาพของฟิกส์ซึ่งเรียกว่า เวิร์กสเตชัน (Workstation) โดยที่อุปกรณ์จริงอาจเป็นจอภาพ วินโดว์ในระบบเอกซ์วินโดว์ เครื่องพิมพ์ (Printer) หรือพล็อตเตอร์ (Plotter) ก็ได้

การสร้างโปรแกรมประยุกต์โดยใช้ฟิกส์นั้น ทำได้โดยใช้การยึดเหนี่ยวของภาษา (Language Binding) ซึ่งในซันฟิกส์รุ่น 2.0 จะใช้ได้ 2 ภาษา คือ ภาษาฟอร์แทรนและภาษาซี เมื่อโปรแกรมประยุกต์ได้เรียกใช้งานฟิกส์แล้ว ฟิกส์จะเก็บข้อมูลต่างๆ ลงในหน่วยเก็บโครงสร้างส่วนกลาง หลังจากนั้นเมื่อโปรแกรมประยุกต์ได้สร้างเวิร์กสเตชัน ฟิกส์จะนำข้อมูลจากตารางเวิร์กสเตชันและหน่วยเก็บโครงสร้างส่วนกลางมาใช้ในการแสดงผล โดยที่โปรแกรมประยุกต์แต่ละโปรแกรมสามารถมีเวิร์กสเตชันได้หลายเวิร์กสเตชัน แต่จะมีหน่วยเก็บโครงสร้างส่วนกลางได้เพียง 1 หน่วยต่อ 1 เครื่องเท่านั้น เวิร์กสเตชันของฟิกส์จะใช้ติดต่อกับอุปกรณ์ต่างๆ คือ จะมีการรับข้อมูลจากอุปกรณ์นำเข้า และแสดงภาพจากข้อมูลในหน่วยเก็บโครงสร้างส่วนกลาง โดยการทำทราเวอร์ซัลให้กับโครงสร้าง หรือสามารถเก็บข้อมูลจากหน่วยเก็บโครงสร้างส่วนกลางลงในแฟ้ม ข้อมูลเป็นแบบแฟ้มเก็บถาวร (Archive File)



รูป 2.10 สถาปัตยกรรมเชิงตรรกะของฟิกส์ (Blake, 1993)

ฟิกส์มีข้อจำกัดหลายอย่าง เช่น ไม่สามารถทำการตามรังสีแสงและการกระจายแสง (Radiosity) ได้ เนื่องจากฟิกส์ไม่สามารถคำนวณผลกระทบของวัตถุหนึ่งต่อวัตถุอื่นในฉากได้ ฟิกส์ไม่สามารถทำการแปลงส่งลักษณะพื้นผิว ภาพเบลอที่เกิดจากการเคลื่อนไหว (Motion Blur) และหมอกคล้ายจริง (Realistic Fog) ได้ ฟิกส์จึงไม่เหมาะกับงานสร้างภาพเหมือนจริง และเนื่องจากฟิกส์ทำงานแบบดิสเพลย์ ลิส ดังนั้นฟิกส์จึงไม่เหมาะกับงานประเภทภาพเคลื่อนไหว (Animation) ด้วย สำหรับข้อจำกัดนี้อาจถูกแก้ไขในการทำให้เกิดผลของฟิกส์ (PHIGS's Implementation) บางตัว แต่จะทำให้การทำให้เกิดผลของฟิกส์ตัวนั้น สูญเสียคุณสมบัติใช้ได้หลายระบบไป

เอ็มไอทีเอกซ์ คอนซอร์เตียม (M.I.T.X Consortium) ได้ทำการพัฒนาโปรโตคอลของฟิกส์และเอกซ์วินโดว์ ที่เรียกว่า เพกซ์โปรโตคอล (PEX - PHIGS Extended to X Protocol) ทำให้ฟิกส์สามารถแสดงภาพกราฟิกบนเครื่องต่างๆ ในระบบเครือข่ายได้ สำหรับเอกซ์วินโดว์ที่จะรู้จักเพกซ์โปรโตคอลจะต้องเป็นรุ่น 5 ขึ้นไป

การทำให้เกิดผลของฟิกส์ส่วนใหญ่มักมีจุดประสงค์ในเชิงการค้า ยกเว้นชุดคำสั่งเพกซ์เอสไอ (PEX-SI) ซึ่งถูกพัฒนาโดยเอ็มไอทีเอกซ์ คอนซอร์เตียม ซึ่งเป็นผู้พัฒนาฟิกส์ด้วยเช่นกัน การทำให้เกิดผลของฟิกส์อื่นๆ จึงยึดเพกซ์ เอสไอเป็นหลัก เพกซ์ เอสไอถูกพัฒนาเพื่อใช้กับระบบเอกซ์วินโดว์ จึงทำให้การทำให้เกิดผลของฟิกส์ส่วนมากต้องทำงานร่วมกับระบบเอกซ์วินโดว์

2. <u>แนวคิดหลักของฟิกส์</u>

2.1 ขั้นตอนการทำงานของฟิกส์ ซึ่งสามารถแสดงให้เห็นได้ดังนี้

2.1.1 การเรียกใช้งานฟิกส์เป็นขั้นตอนแรกของการทำงาน ซึ่งหลังจากขั้นตอนนี้จะสามารถเรียกใช้คำสั่งต่างๆ ของฟิกส์ได้ ถ้าข้ามขั้นตอนนี้ไปคำสั่งต่างๆ ของฟิกส์จะไม่สามารถใช้งานได้

2.1.2 การกำหนดข้อมูลต่างๆ ของแบบจำลอง (Model) ที่ต้องการแสดงภาพลงในฐานข้อมูลกราฟิกของฟิกส์ ที่เรียกว่า หน่วยเก็บโครงสร้างส่วนกลาง ในหน่วยเก็บโครงสร้างส่วนกลางนี้สามารถจัดกลุ่มของข้อมูลที่เกี่ยวข้องกันไว้เป็นกลุ่มๆ ได้ ซึ่งจะเรียกกลุ่มเหล่านี้ว่าโครงสร้าง (Structure) ในโครงสร้างนี้จะมีองค์ประกอบ (Element) มากมาย แต่องค์ประกอบที่สำคัญมี 2 ชนิดคือ รูปทรงพื้นฐาน (Primitive) และลักษณะประจำ โดยรูปทรงพื้นฐานจะเป็นตัวกำหนดรูปทรงของภาพที่ปรากฏ เช่น วงกลม เส้นตรง สี่เหลี่ยม เป็นต้น ส่วนลักษณะประจำจะกำหนดลักษณะต่างๆ ของภาพที่ปรากฏ เช่น สี ความหนาของเส้น เป็นต้น

2.1.3 การสร้างเวิร์กสเตชันสำหรับแสดงภาพ โดยที่คำว่า เวิร์กสเตชันของฟิกส์มิได้หมายถึง เครื่องคอมพิวเตอร์สมรรถนะสูงดังที่ใช้กันอยู่ทั่วไป แต่หมายถึงสถานที่สำหรับแสดงภาพของแบบจำลองที่กำหนดไว้ในหน่วยเก็บโครงสร้างส่วนกลาง ซึ่งเวิร์กสเตชันในที่นี้อาจเป็นจอภาพ เครื่องพิมพ์ หรือพล็อตเตอร์ ก็ได้ สำหรับขั้นตอนที่ 2 และ 3 จะทำขั้นตอนใดก่อนก็ได้

2.1.4 การเชื่อมโยงแบบจำลองเข้ากับเวิร์กสเตชัน เป็นเพียงการบอกให้เวิร์กสเตชันรู้ว่าจะต้องแสดงภาพของแบบจำลองใด ในขั้นตอนนี้จะไม่มีการนำข้อมูลของแบบจำลองในหน่วยเก็บโครงสร้างส่วนกลางไปใส่ในเวิร์กสเตชันข้อมูลยังคงเก็บอยู่ในหน่วยเก็บโครงสร้างส่วนกลาง

2.1.5 การแสดงภาพ โดยที่ภาพของแบบจำลองจะถูกแสดงบนเวิร์กสเตชันได้ด้วยการปรับ (Update) เวิร์กสเตชัน ดังนั้นถ้าต้องการแสดงภาพของแบบจำลองจำเป็นจะต้องมีขั้นตอนที่ 1-5

2.1.6 การแก้ไขแบบจำลอง ซึ่งขั้นตอนนี้อาจมีหรือไม่มีก็ได้ ขั้นตอนนี้เป็นการแก้ไขส่วนต่างๆ ของแบบจำลองให้เป็นไปตามความต้องการโดยการเปลี่ยนแปลงองค์ประกอบภายในโครงสร้างนั้น หลังจากแก้ไขข้อมูลของแบบจำลองในหน่วยเก็บโครงสร้างส่วนกลางแล้ว ไม่จำเป็นต้องส่งโครงสร้างของแบบจำลองไปยังเวิร์กสเตชันอีก เมื่อทำการแก้ไขแล้วสามารถกำหนดให้เวิร์กสเตชันแสดงภาพที่เปลี่ยนไปโดยทันทีหรือไม่ก็ได้

2.1.7 การปิดเวิร์กสเตชันเมื่อไม่ต้องการใช้งาน จะต้องปิดเวิร์กสเตชันเสมอ

2.1.8 การยกเลิกการใช้งานฟิกส์ ซึ่งหลังจากขั้นตอนนี้คำสั่งต่างๆ ของฟิกส์จะไม่สามารถใช้งานได้อีก

จากขั้นตอนต่างๆ ในข้างต้นสามารถแสดงได้ดังโปรแกรมภาษาซี (Gaskins, 1991) ต่อไปนี้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
#include  <phigs/phigs.h>
#define   WORKSTATION_ID  (Pint) 1
#define   STRUCTURE_ID  (Pint) 1

main( )
{
/* กำหนดจุดต่างๆ บนโพลีไลน์ */
    Ppoint_list3   point_list;
```

```
    static Ppoint3   points[] = {{.2, .2, 0}, {.4, .7, 0}, {.6, .3, 0}, {.8, .8, 0}};

    point_list.num_points = 4;
    point_list.points = points;

    popen_phigs(PDEF_ERR_FILE, PDEF_MEM_SIZE);      /* เรียกใช้งานฟิกส์ */

/* นิยามแบบจำลองให้แก่ฐานข้อมูลฟิกส์ */
    popen_struct (STRUCTURE_ID);            /* สร้างโครงสร้าง */
            pset_linetype (PLINE_DASH);  /* กำหนดลักษณะประจำเป็นเส้นประ */
            ppolyline3 (&point_list);       /* กำหนดรูปทรงพื้นฐานเป็นโพลีไลน์ */
    pclose_struct ();                       /* ปิดโครงสร้าง */

/* สร้างและกำหนดเวิร์กสเตชันของฟิกส์แบบเอกซ์ทูลเพื่อใช้ในการแสดงผล */
    popen_ws (WORKSTATION_ID,NULL,phigs_ws_type_x_tools);

/* เชื่อมโยงโครงสร้างเข้ากับเวิร์กสเตชัน */
    ppost_struct (WORKSTATION_ID,STRUCTURE_ID,(Pfloat)1.0);

/* สั่งให้เวิร์กสเตชันแสดงภาพแบบจำลองนั้น */
    pupd_ws (WORKSTATION_ID,PFLAG_PERFORM);
    sleep (10);         /* หยุดรอ 100 วินาที */

/* เริ่มต้นแก้ไของค์ประกอบในโครงสร้าง */
    pset_edit_mode (PEDIT_REPLACE);
    popen_struct (STRUCTURE_ID);
            pset_elem_ptr (1);
            pset_linetype (PLINE_SOLID); /* เปลี่ยนลักษณะประจำเป็นเส้นทึบ */
    pclose_struct ();
    pset_edit_mode (PEDIT_INSERT);
```

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
/* สั่งให้เวิร์กสเตชันแสดงภาพแบบจำลองนั้น */
    pupd_ws (WORKSTATION_ID,PFLAG_PERFORM);
    sleep (10);          /* หยุดรอ 100 วินาที */
    pclose_ws (WORKSTATION_ID);          /* ปิดเวิร์กสเตชัน */
    pclose_phigs ();                     /* จบการใช้งานฟิกส์ */
}
```

ในโปรแกรมนี้จะเป็นการสร้างภาพโพลีไลน์ที่มีลักษณะเป็นเส้นประ และหลังจากนั้นจะกลายเป็นเส้นโพลีไลน์เดิมแต่มีลักษณะเป็นเส้นทึบ

2.2 ลักษณะการทำงานของฟิกส์ (Blake, 1993)

ฟิกส์มีสถานะ (State) ของการทำงานที่สำคัญ 4 ชนิด ซึ่งแต่ละชนิดก็จะมี 2 สถานะ คือ เปิด (OPEN ย่อว่า OP) และปิด (CLOSED ย่อว่า CL) ดังนี้

2.2.1 สถานะของฟิกส์ แทนด้วย "PH" ใช้บอกว่าขณะนี้มีการเรียกใช้ฟิกส์หรือไม่ ถ้ามีการเรียกใช้จะมีสถานะเป็น "เปิด" หรือแทนด้วย "PHOP" และถ้าไม่มีการเรียกใช้จะมีสถานะเป็น "ปิด" หรือแทนด้วย "PHCL"

2.2.2 สถานะของเวิร์กสเตชัน แทนด้วย "WS" ถ้ามีการเปิดใช้งานจะมีสถานะ "เปิด" หรือแทนด้วย "WSOP" และถ้าไม่มีการใช้จะมีสถานะ "ปิด" หรือแทนด้วย "WSCL" ในกรณีสถานะ "WSOP" จะหมายความว่า มีเวิร์กสเตชันเปิดใช้งานอยู่แล้วอย่างน้อย 1 เวิร์กสเตชัน แต่ถ้าเป็นสถานะ "WSCL" แสดงว่าไม่มีการเปิดใช้งานเวิร์กสเตชันใดๆ เลย

2.2.3 สถานะของโครงสร้าง แทนด้วย "ST" เมื่อมีการเปิดโครงสร้างเพื่อใส่ข้อมูล จะมีสถานะเป็น "เปิด" หรือแทนด้วย "STOP" และเมื่อปิดโครงสร้างก็จะมีสถานะเป็น "ปิด" หรือแทนด้วย "STCL" ในเวลาหนึ่งๆ สามารถเปิดโครงสร้างได้เพียง 1 โครงสร้างเท่านั้น

2.2.4 สถานะของแฟ้มเก็บถาวร แทนด้วย "AR" เนื่องจากหน่วยเก็บโครงสร้างส่วนกลางเก็บข้อมูลไว้ในหน่วยความจำ (Memory) เท่านั้น เมื่อจบโปรแกรมข้อมูลต่างๆ จะสูญหายไปหมด เราสามารถเก็บข้อมูลเหล่านี้ไว้ในแฟ้ม (File) ได้ ซึ่งเรียกว่า แฟ้มเก็บถาวร เมื่อเปิดแฟ้ม จะมีสถานะเป็น "เปิด" หรือแทนด้วย "AROP" และจะมีสถานะเป็น "ปิด" หรือแทนด้วย "ARCL" เมื่อปิดแฟ้ม ในขณะเดียวกันสามารถเปิดแฟ้มเก็บถาวรได้มากกว่า 1 แฟ้ม

2.3 รูปทรงพื้นฐาน (Gaskins, 1991)

2.3.1 รูปทรงพื้นฐานของฟิกส์

รูปทรงพื้นฐานใช้กำหนดลักษณะรูปทรงของวัตถุหรือแบบจำลอง โดยข้อมูลที่ใช้

กำหนดลักษณะอาจจะเป็น จุดยอด (Vertex) หรือจุดควบคุม (Control Point) เป็นต้น ทั้งนี้ขึ้นอยู่กับชนิดของรูปทรงพื้นฐานที่ใช้ รูปทรงพื้นฐานของฟิกส์อาจแบ่งได้เป็น 4 ชนิด คือ แบบเส้น (Line) แบบพื้นที่ (Area) แบบข้อความ (Text) หรือแบบพิเศษ โดยมีรายละเอียดแต่ละชนิดดังนี้

2.3.1.1 แบบเส้น (Line Primitive)

โพลีไลน์ (Polyline) คือ กลุ่มของส่วนของเส้นตรงที่เชื่อมต่อกัน ซึ่งมีทั้งแบบ 2 มิติ และ 3 มิติ

2.3.1.2 แบบพื้นที่ (Area Primitive)

2.3.1.2.1 พื้นที่เติมเต็ม (Fill Area) คือ โพลีกอน (Polygon) แบบแบน มีทั้ง 2 มิติ และ 3 มิติ

2.3.1.2.2 กลุ่มของพื้นที่เติมเต็ม (Fill Area Set) คือ กลุ่มของโพลีกอน ที่อาจทับซ้อนกัน (Overlap) หรือแยกออกจากกัน (Disjoint) ส่วนที่ทับซ้อนกันจะกลายเป็นช่องบนโพลีกอน

สำหรับกลุ่มของพื้นที่เติมเต็มจะมีเส้นที่ใช้บอกอาณาเขตของพื้นที่ได้ เรียกว่า ขอบ (Edge) ซึ่งพื้นที่เติมเต็มจะไม่มี

2.3.1.3 แบบตัวข้อความ (Text Primitive)

2.3.1.3.1 ข้อความ คือ สายของอักขระ ซึ่งมีทั้งแบบ 3 มิติ และ 2 มิติ

2.3.1.3.2 ข้อความแอนโนเตชัน (Annotation Text) คือ สายของอักขระ ซึ่งจะแสดงบนเพลน (Plane) ของ Screen ทำให้สามารถอ่านได้ง่าย ถึงแม้จะเปลี่ยนวิวไปอย่างไรก็ตาม ตัวอักษรก็จะแสดงอยู่บนเพลนของ Screen เสมอ ซึ่งแตกต่างจากข้อความ เมื่อมีการเปลี่ยนมุมมองไปอาจจะทำให้ไม่สามารถอ่านได้ นอกจากนี้ข้อความแอนโนเตชันสามารถมีเส้นที่ลากเชื่อมต่อกับจุดใดจุดหนึ่งบนแบบจำลองได้

2.3.1.4 แบบอื่นๆ

2.3.1.4.1 โพลีมาร์กเกอร์ (Polymarker) คือ กลุ่มของสัญลักษณ์ต่างๆ เช่น *, o, x เป็นต้น ซึ่งสัญลักษณ์เหล่านี้มักใช้ในการแสดงกลุ่มของจุด

2.3.1.4.2 แถวลำดับของเซลล์ (Cell Array) เป็นกริด (Grid) ของสี่เหลี่ยมจตุรัสหรือสี่เหลี่ยมด้านขนาน ซึ่งสามารถกำหนดสีลงไปในแต่ละช่องได้ มักใช้ในการแสดงภาพประเภทแผนที่บิต (Bitmap) มีทั้งแบบ 2 มิติ และ 3 มิติ

2.3.1.4.3 รูปทรงพื้นฐานเจนเนอร์รัลไลซ์ ดรออิง (Generalized Drawing Primitive ย่อว่า GDP) เป็นกลวิธีที่ใช้เพื่อเพิ่มชนิดของรูปทรงพื้นฐาน รูปทรงพื้นฐาน

เจนเนอร์รัลไลซ์ ดรออิงอาจเป็น ทรงกลม ลูกบาศก์ วงกลม เป็นต้น ซึ่งจะขึ้นอยู่กับการทำให้เกิดผลของฟีกส์

2.3.2 รูปทรงพื้นฐานของฟีกส์พลัส

ในส่วนของฟีกส์พลัสยังมีรูปทรงพื้นฐานมากกว่าที่กล่าวข้างต้นอีกหลายชนิด และมักจะมีคำว่า กำหนดด้วยข้อมูล (with data) ต่อท้าย ซึ่งหมายถึง สามารถกำหนดข้อมูลสีไว้ที่แต่ละจุดยอดได้ มักใช้ในการทำเชด (Shade) สีบนรูปทรงพื้นฐานนั้นๆ รูปทรงพื้นฐานต่างๆ ของฟีกส์พลัสมีดังนี้

2.3.2.1 แบบเส้น

2.3.2.1.1 กลุ่มของโพลีไลน์ที่กำหนดด้วยข้อมูล (Polyline Set with Data) คือ กลุ่มของโพลีไลน์ ซึ่งแต่ละเส้นจะไม่เชื่อมต่อกัน

2.3.2.1.2 เส้นโค้งบีสไปลน์ (B-spline Curve) คือ เส้นโค้งสไบลน์ (Spline Curve) ชนิดหนึ่ง อาจเรียกอีกอย่างว่า เส้นโค้งบีสไปลน์แบบนอนยูนิฟอร์ม เรชันนัล มีทั้งแบบ เรชันนัล (Rational) และนอนเรชันนัล (Non-rational)

2.3.2.2 แบบพื้นที่

2.3.2.2.1 กลุ่มของพื้นที่เติมเต็มที่กำหนดด้วยข้อมูล (Fill Area Set with Data) มีลักษณะเหมือนกลุ่มของพื้นที่เติมเต็ม แต่สามารถกำหนด ข้อมูลสี ทิศทาง และ นอร์มัล เวกเตอร์ (Normal Vector) เพื่อใช้ในการทำผลกระทบของแสง (Light Effect) ได้

2.3.2.2.2 ชุดของกลุ่มของพื้นที่เติมเต็มที่กำหนดด้วยข้อมูล (Set of Fill Area Set with Data) คือ กลุ่มของพื้นที่เติมเต็มที่กำหนดด้วยข้อมูล ซึ่งเป็นรูปทรงพื้นฐานแบบพื้นที่ที่ใช้มาก มักใช้ในการแสดงภาพพื้นผิวของวัตถุ

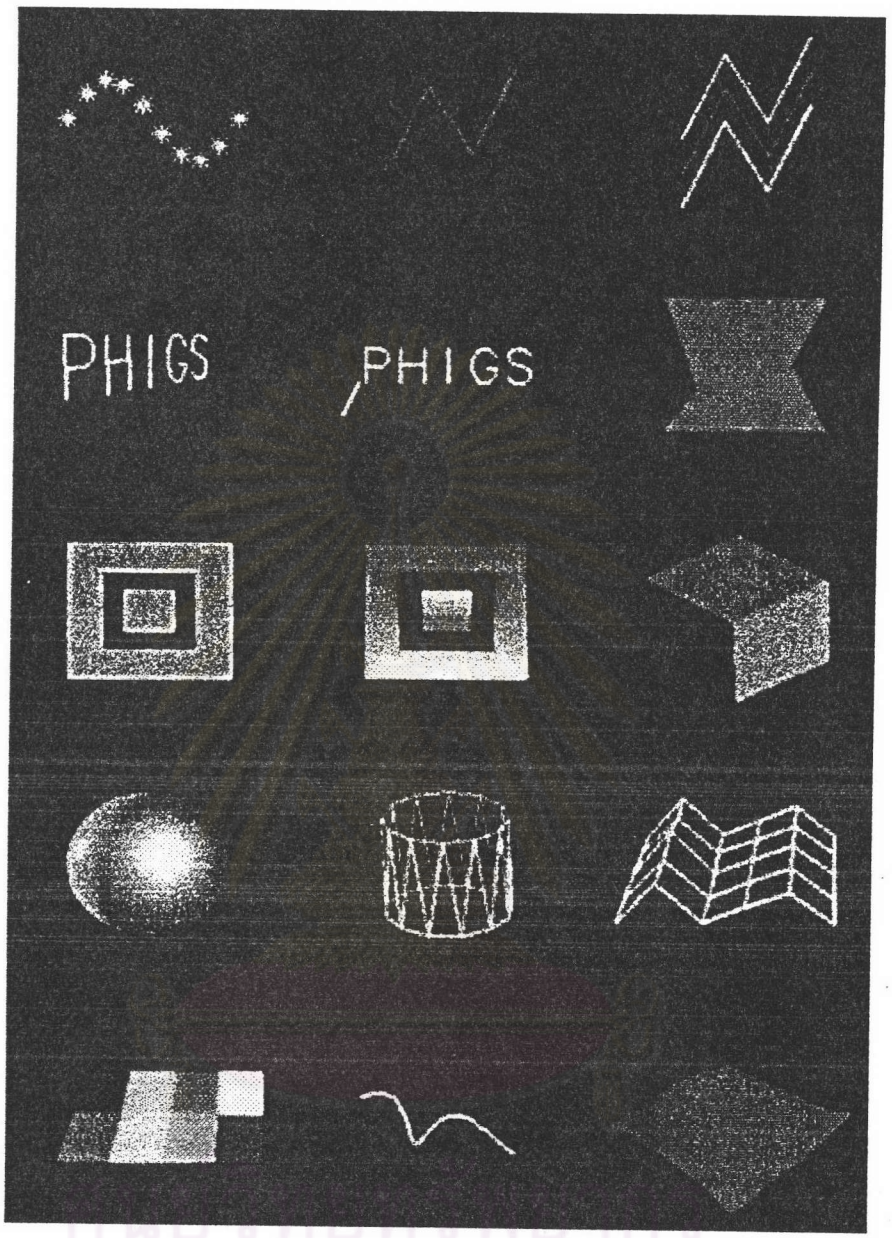
2.3.2.2.3 ตารางสามเหลี่ยมที่กำหนดด้วยข้อมูล (Triangle Strip with Data) คือ กลุ่มของรูปทรงสามเหลี่ยมที่เชื่อมต่อกัน

2.3.2.2.4 ตารางสี่เหลี่ยมที่กำหนดด้วยข้อมูล (Quadrilateral Mesh with Data) คือ กลุ่มของรูปทรงสี่เหลี่ยมที่เชื่อมต่อกัน

2.3.2.2.5 พื้นผิวแบบบีสไปลน์ (B-spline Surface) เป็น พื้นผิวแบบสไปลน์ (Spline-surface) ชนิดที่นิยมใช้มาก อาจเรียกอีกอย่างหนึ่งว่า พื้นผิวบีสไปลน์แบบนอนยูนิฟอร์ม เรชันนัล (Non-Uniform Rational B-spline Surface)

2.3.2.3 แบบพิเศษ

2.3.2.3.1 แถวลำดับของเซลล์แบบฟีกส์พลัส (Cell Array PLUS) มีลักษณะเหมือนแถวลำดับของเซลล์ แต่สามารถกำหนดข้อมูลสีโดยตรงได้



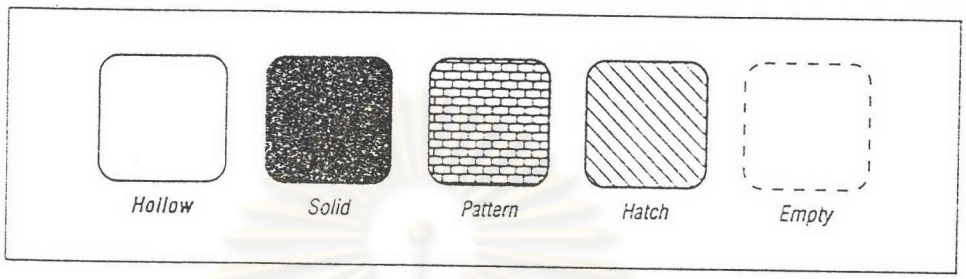
โพลีมาร์เกอร์	โพลีไลน์	โพลีไลน์ที่กำหนดด้วยข้อมูล
ข้อความ	ข้อความแอนโนเตชัน	พื้นที่เติมเต็ม
กลุ่มของพื้นที่เติมเต็ม	กลุ่มของพื้นที่เติมเต็มที่กำหนดด้วยข้อมูล	ชุดของกลุ่มของพื้นที่เติมเต็มที่กำหนดด้วยข้อมูล
เจนเนอร์รัลไลซ์ ดรออิง	ตารางสามเหลี่ยมที่กำหนดด้วยข้อมูล	ตารางสามเหลี่ยมที่กำหนดด้วยข้อมูล
แถวลำดับของเซลล์	เส้นโค้งบีสไปลน์	พื้นผิวแบบบีสไปลน์

รูป 2.11 รูปทรงพื้นฐานชนิดต่างๆ (Gaskins, 1991)

2.4 ลักษณะประจำ

สิ่งที่ใช้ควบคุมลักษณะที่ปรากฏของรูปทรงพื้นฐาน เรียกว่า ลักษณะประจำ ซึ่งรูปทรงพื้นฐาน แต่ละชนิดจะมีลักษณะประจำ เป็นของตัวเอง เช่น โพลีไลน์จะมีลักษณะประจำ

คือ ความหนาของเส้น สีและรูปแบบของเส้น พื้นที่เติมเต็มจะมีลักษณะประจำ คือ รูปแบบภายใน (Interior Style) สีของพื้นผิว ส่วนข้อความจะมีลักษณะประจำเป็น ความสูงของตัวอักษร สี และแบบอักษร (Font) เป็นต้น



รูป 2.12 ลักษณะประจำชนิดรูปแบบภายในของรูปทรงพื้นฐานประเภทพื้นที่ (Gaskins, 1991)

2.5 ทราเวอร์ซัล (Traversal) (Gaskins, 1991)

การเก็บข้อมูลของแบบจำลองลงในหน่วยเก็บโครงสร้างส่วนกลาง ทำได้โดยการเปิดโครงสร้างแล้วบันทึกข้อมูลลงในโครงสร้าง ข้อมูลหน่วยที่เล็กที่สุดในโครงสร้างเรียกว่า องค์ประกอบ ซึ่งมีหลายชนิด เช่น ลักษณะประจำ รูปทรงพื้นฐาน ป้าย (Label) เป็นต้น

เมื่อต้องการแสดงภาพฟิกส์จะอ่านข้อมูลในโครงสร้างตั้งแต่ องค์ประกอบ ที่ 1 เรียงตามลำดับไปจนถึงองค์ประกอบสุดท้าย ถ้าองค์ประกอบนั้นเป็นลักษณะประจำ ฟิกส์จะนำค่าของลักษณะประจำนั้นไปใช้กับรูปทรงพื้นฐานที่เกี่ยวข้องที่อยู่ในองค์ประกอบลำดับต่อมา เมื่ออ่านพบองค์ประกอบที่เป็นรูปทรงพื้นฐาน ฟิกส์จะทำการวาดภาพของรูปทรงพื้นฐานนั้น ด้วยลักษณะประจำที่ถูกกำหนดไว้ก่อนหน้านี้ กระบวนการที่กล่าวมาทั้งหมดนี้เรียกว่า ทราเวอร์ซัล

เนื่องจากการทราเวอร์ซัลเป็นการทำงานแบบเรียงลำดับ คือ ตั้งแต่องค์ประกอบแรกจนถึงองค์ประกอบสุดท้าย และจะทำการวาดภาพเมื่อพบองค์ประกอบที่เป็นรูปทรงพื้นฐาน ดังนั้นค่าของลักษณะประจำที่ถูกกำหนดในองค์ประกอบที่อยู่ลำดับต่อมาจากรูปทรงพื้นฐานนั้น ก็จะไม่ถูกนำไปใช้กับรูปทรงพื้นฐานที่อยู่ลำดับก่อนหน้านี้

ขั้นตอนการทราเวอร์ซัลจะเกิดขึ้นเมื่อมีการสั่งให้ปรับเวิร์กสเตชัน ซึ่งอาจสั่งโดยตรงหรือกำหนดให้เป็นแบบอัตโนมัติก็ได้ การทราเวอร์ซัลจะทำกับโครงสร้างที่ถูกส่งมายังเวิร์กสเตชันนั้นเท่านั้น

2.6 สี

สีจัดเป็นลักษณะประจำชนิดหนึ่ง และเป็นลักษณะประจำที่ถูกเรียกใช้มาก

สำหรับฟิกส์นั้นสามารถแบ่งชนิดของการกำหนดข้อมูลสีได้เป็น 2 ชนิดใหญ่ๆ คือ

2.6.1 การกำหนดข้อมูลสีทางอ้อม (Indirect Color) เป็นการเรียกใช้สีที่อยู่ในตารางสีของเวิร์กสเตชัน โดยใช้ ดรรชนีของตารางสี

2.6.2 การกำหนดข้อมูลสีโดยตรง (Direct Color) เป็นการกำหนดข้อมูลของสีโดยตรง ข้อมูลเหล่านี้จะแตกต่างกันไปในแต่ละแบบจำลองของสี แบบจำลองของสีที่ฟิกส์ใช้ได้มี 4 ชนิด คือ

2.6.2.1 แบบแดง เขียว ฟ้า (Red-Green-Blue)

2.6.2.2 แบบฮิวส์ เซตตูเรชัน แวลู (Hue-Saturation-Value)

2.6.2.3 แบบฮิวส์ ไลต์เนส เซตตูเรชัน Hue-Lightness-Saturation

2.6.2.4 แบบซีไออี แอลยูวี (CIE-LUV)

2.7 โครงสร้างและการแก้ไขโครงสร้าง (Structure and Structure Editing)

หลังจากที่ได้บันทึกข้อมูลของแบบจำลองลงในหน่วยเก็บโครงสร้างส่วนกลางแล้ว ยังสามารถทำการเปลี่ยนแปลง เพิ่มเติม หรือลบข้อมูลเหล่านั้นได้อีก ข้อมูลที่เกี่ยวข้องกันจะถูกจัดเป็นกลุ่มๆ เรียกว่า โครงสร้าง และส่วนที่เล็กที่สุดในโครงสร้าง เรียกว่า องค์ประกอบ ซึ่งแต่ละ องค์ประกอบจะมีตัวเลขที่ใช้อ้างถึงลำดับที่ในโครงสร้าง เรียกว่า ตัวชี้องค์ประกอบ (Element Pointer) โดยตัวชี้องค์ประกอบที่ 0 หมายถึง ตำแหน่งก่อนหน้าองค์ประกอบแรก 1 ตำแหน่ง และตำแหน่งที่ N หมายถึง องค์ประกอบสุดท้ายของโครงสร้าง

โครงสร้างเป็นส่วนสำคัญที่ช่วยในการจัดการกับฐานข้อมูลกราฟิก การจัดกลุ่มข้อมูลสามารถทำได้โดยจัดกลุ่มของข้อมูลที่เกี่ยวข้องกันไว้ในโครงสร้างเดียวกัน โดยนิยามข้อมูลต่างๆ ลงในองค์ประกอบของโครงสร้าง เมื่อต้องการอ้างถึงข้อมูลกลุ่มใด จะสามารถอ้างถึงได้โดยการเรียกใช้โครงสร้างนั้น เช่น ข้อมูลของรถยนต์ อาจประกอบด้วยข้อมูลของล้อ ตัวถัง เครื่องยนต์ เป็นต้น เมื่อต้องการใช้ข้อมูลเหล่านี้ก็เพียงแต่อ้างถึงโครงสร้างของรถยนต์เท่านั้น

ด้วยวิธีการนี้ทำให้สามารถแสดงภาพของวัตถุชิ้นเดียวกัน ในจำนวนมากกว่า 1 ชิ้น ที่ตำแหน่งต่างๆ ณ เวลาเดียวกันได้ โดยการบันทึกข้อมูลเพียงครั้งเดียว

ภาวะ (Mode) ของการแก้ไขมี 2 ชนิด คือ การแทรก (INSERT) และการแทนที่ (REPLACE) ในกรณีของการแทรก ฟิกส์จะทำการเพิ่มองค์ประกอบใหม่ถัดจากตำแหน่งขององค์ประกอบปัจจุบันไป 1 ตำแหน่ง แล้วเลื่อนตัวชี้องค์ประกอบไปยังตำแหน่งขององค์ประกอบใหม่ ส่วนในกรณีของการแทนที่ ฟิกส์จะนำองค์ประกอบใหม่มาใส่แทนองค์ประกอบเดิม และไม่มีการเลื่อนตัวชี้องค์ประกอบ การแก้ไขขององค์ประกอบในโครงสร้างจึงทำได้โดยการระบุตำแหน่งของตัวชี้องค์ประกอบที่ต้องการแก้ไข แล้วทำการแก้ไข เพิ่มเติม หรือลบองค์ประกอบนั้นๆ

2.8 การปรับ

การเปลี่ยนแปลงข้อมูลใดๆ ในโครงสร้างเป็นเพียงการเปลี่ยนแปลงข้อมูลในหน่วยเก็บโครงสร้างส่วนกลางเท่านั้น โดยจะไม่ทำให้เกิดการเปลี่ยนแปลงใดๆ บนเวิร์กสเตชันที่แสดงภาพเลย ดังนั้นเมื่อต้องการแสดงภาพของแบบจำลอง จะต้องสั่งให้มีการปรับเวิร์กสเตชันก่อน จึงจะมีภาพปรากฏออกมา การปรับนี้สามารถสั่งงานได้โดยตรงหรือสั่งให้ปรับโดยอัตโนมัติก็ได้ ฟิกส์มีส่วนสำคัญเกี่ยวกับการปรับ อยู่ 3 ส่วน คือ

2.8.1 สถานะวิชัว รีพรีเซนเตชัน (State of Visual Representation ย่อว่า SVR) เป็นค่าที่ใช้บอกถึงสถานะปัจจุบันของรายการสถานะเวิร์กสเตชัน (Workstation State List) และโครงสร้างทั้งหมดที่เกี่ยวข้องกับเวิร์กสเตชันนั้น เมื่อมีการแก้ไขโครงสร้าง ค่านี้จะถูกกำหนดเป็นดิฟเฟอร์ (DEFER) หรือซิมูเลต (SIMULATED) ทั้งนี้ขึ้นอยู่กับวิธีการปรับ และภาวะดิฟเฟอร์รัล (Deferal Mode) และเมื่อเวิร์กสเตชันถูกปรับแล้ว ค่าจะเปลี่ยนเป็น ถูกต้อง (CORRECT)

2.8.2 ภาวะดิฟเฟอร์รัลใช้เพื่อบอกให้ฟิกส์รู้ว่าเมื่อใดที่จะต้องทำการปรับ เช่น ทำการปรับทันที หรือรอจนกว่าจะมีสัญญาณนำเข้าจึงจะทำการปรับ

2.8.3 ภาวะโมดิฟิเคชัน (Modification Mode) ใช้เพื่อบอกให้ฟิกส์รู้ว่า ต้องทำการปรับอย่างไร เช่น ปรับโดยไม่ต้องมีการทำทราเวอร์ซัลใหม่

2.9 ทรานส์ฟอร์เมชัน ไพป์ไลน์ (Transformation Pipeline) (Gaskins, 1991)

การแสดงภาพของรูปทรงพื้นฐานจะต้องผ่านขั้นตอนของลำดับการ ทรานส์ฟอร์เมชันชุดหนึ่ง เรียกว่า ทรานส์ฟอร์เมชัน ไพป์ไลน์ซึ่งประกอบด้วย 3 ขั้นตอน ดังนี้

2.9.1 ขั้นตอนโมเดลลิง (Modeling Stage) เป็นการแปลงรูปทรงพื้นฐานทั้งหมดให้อยู่ในพิกัด (Coordinate) โลก

ข้อมูลพิกัดของรูปทรงพื้นฐานจะต้องถูกกำหนดลงบนระบบพิกัดโลก (World Coordinate) แต่ละรูปทรงพื้นฐานสามารถมีพิกัดเป็นของตัวเอง ซึ่งเรียกว่า พิกัดของแบบจำลอง (Modeling Coordinate) เมื่อต้องการจะสร้างแบบจำลองสามารถทำได้โดยนำ รูปทรงพื้นฐานต่างๆ มารวมกันในพิกัดของแบบจำลองของแบบจำลองที่เรียกว่า พิกัดโลก ซึ่งจะต้องมีการแปลงค่าตำแหน่งของรูปทรงพื้นฐานจากพิกัดของแบบจำลองให้มาอยู่ในตำแหน่งที่ต้องการบนพิกัดโลก เช่น แบบจำลองของรถยนต์ ประกอบด้วย ตัวถังกับล้อ การสร้างล้อทำได้โดยการวาดวงกลม มีจุดศูนย์กลางอยู่ที่ (0,0) ซึ่งเป็นจุดกำเนิด (Origin) ในพิกัดของแบบจำลองของล้อ เมื่อต้องการสร้างแบบจำลองของรถยนต์ ต้องทำการแปลงตำแหน่งของวงกลมนี้ ให้ไปอยู่ในจุดที่ถูกต้องบนแบบจำลอง

2.9.2 ขั้นตอนวิว (Viewing Stage) ขั้นตอนนี้เป็นการแปลง (Transform) เพื่อกำหนดลักษณะภาพที่จะปรากฏออกมา เช่น ลักษณะเพอร์สเปกทีฟ (Perspective) หรือลักษณะมองภาพจากด้านข้าง เป็นต้น ซึ่งสามารถกำหนดได้ด้วยการแปลงวิว (Viewing Transform) และคลิปปิง เพลน (Clipping plane)

การแปลงวิว ประกอบด้วย 2 ส่วนใหญ่ๆ คือ

2.9.2.1 วิว โอเรียนเตชัน (View Orientation) ใช้ระบุทิศทางและความสัมพันธ์ของผู้มองกับแบบจำลอง ขั้นตอนนี้จะเป็นการแปลงพิกัดโลกให้เป็นพิกัดอ้างอิงของวิว (View Reference Coordinate) แกนวาย (Y) ของพิกัดอ้างอิงของวิวจะเป็นแกนตั้ง และแกนเอกซ์ (X) จะเป็นแกนนอนของจอภาพ ส่วนแกนแซด (Z) จะตั้งฉากกับจอภาพ โดยค่าบวกของแกนแซดจะพุ่งตรงเข้าสู่สายตาเรา การกำหนดพิกัดอ้างอิงของวิวสามารถทำได้โดยกำหนดจุดอ้างอิงของวิว (View Reference Point) ซึ่งเป็นจุดกำเนิดของพิกัดอ้างอิงของวิวด้วยค่าของพิกัดโลก และวิว เพลน นอร์มัล (View Plane Normal) จะใช้กำหนดทิศทางแกนแซดของพิกัดอ้างอิงของวิว ส่วนทิศทางของแกนวายจะถูกกำหนดด้วยวิว อัป เวคเตอร์ (View Up Vector) ส่วนแกนเอกซ์จะถูกกำหนดด้วยกฎของมือขวา

2.9.2.2 วิว แมปปิง (View Mapping) ใช้ระบุลักษณะภาพที่ปรากฏ เช่น เพอร์สเปกทีฟ เป็นต้น ในขั้นตอนนี้จะเป็นการแปลงพิกัดอ้างอิงของวิวให้เป็นระบบพิกัดแบบนอร์มัลไลซ์ โปรเจกชัน โคออดิเนต ข้อมูลที่ต้องกำหนดในขั้นนี้คือ ลักษณะการโปรเจกชัน และปริมาตรของวิว (View Volume)

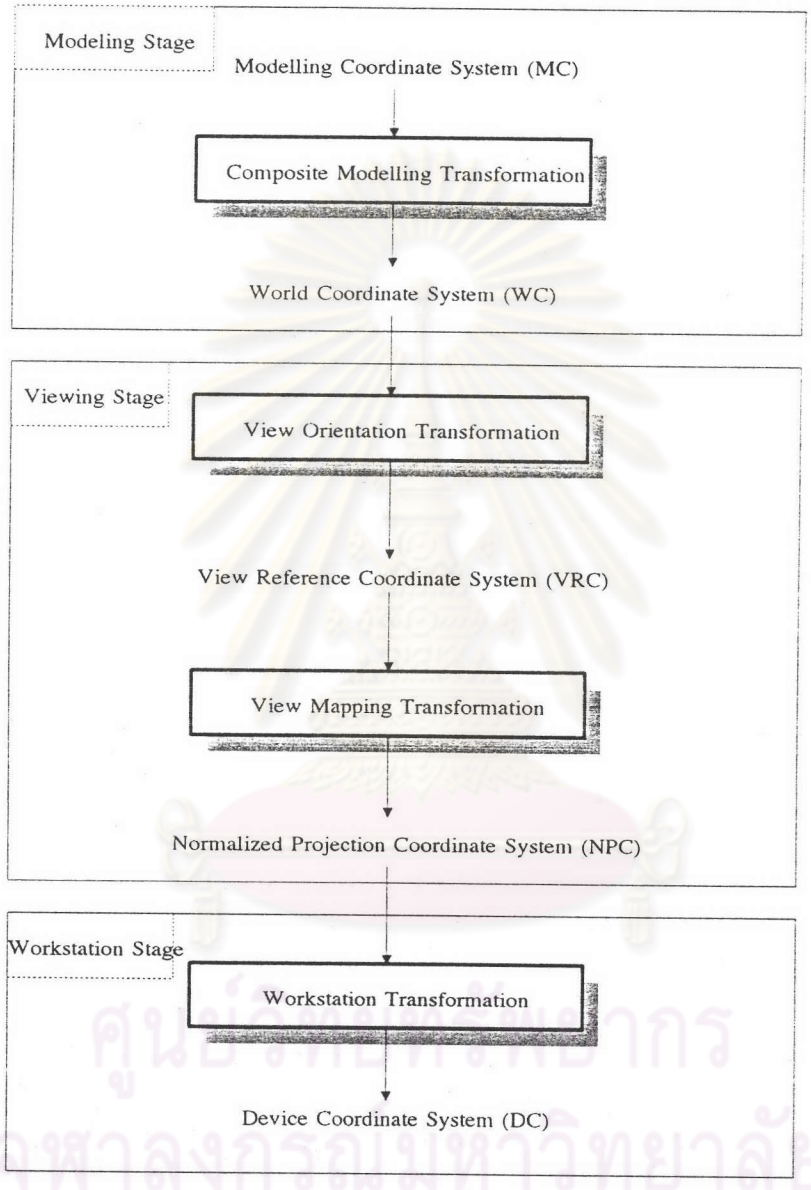
ส่วนคลิปปิง เพลนใช้ระบุว่าส่วนใดของแบบจำลอง คือ ส่วนที่ต้องการมอง และตัดส่วนที่ไม่ต้องการมองออกไป

ภาพของแบบจำลองนั้นสามารถแสดงในวิวที่ต่างกัน พร้อมๆ กันได้ ฟิกส์สามารถแบ่งเวิร์กสเตชันออกเป็นรีเจียน (Region) และแต่ละรีเจียนสามารถแสดงภาพของแบบจำลองในวิวต่างๆ ได้ด้วย รีเจียนเหล่านี้จะถูกกำหนดด้วยระบบพิกัดแบบนอร์มัลไลซ์ โปรเจกชัน โคออดิเนต ซึ่งเป็นพิกัดที่มีลักษณะเป็นทรงลูกบาศก์ขนาด 1 หน่วย (Unit Cube) ที่มีความยาวทุกด้าน ด้านละ 1 หน่วย การใช้นอร์มัลไลซ์ โปรเจกชัน โคออดิเนตจะช่วยให้สามารถทำงานแบบไม่ขึ้นกับอุปกรณ์ได้ เนื่องจากอุปกรณ์แสดงภาพอาจมีระบบพิกัดที่ต่างกัน

2.9.3 ขั้นตอนเวิร์กสเตชัน (Workstation Stage)

เป็นขั้นตอนที่ทำการแปลงค่าจากนอร์มัลไลซ์ โปรเจกชัน โคออดิเนต ให้เป็นพิกัดของอุปกรณ์ (Device Coordinate) ซึ่งแต่ละอุปกรณ์แสดงภาพก็จะมีพิกัดต่างกัน เช่น

เอกซ์วินโดว์นั้น เวิร์กสเตชันจะใช้หน่วยของจุดภาพ (Pixel) และมีจุดกำเนิดอยู่ที่มุมซ้ายบนของวินโดว์



รูป 2.13 ขั้นตอนต่างๆ ของทรานส์ฟอร์เมชัน ไพป์ไลน์

2.10 เครือข่ายของโครงสร้าง (Structure Network) (Gaskins, 1991)

โครงสร้างหนึ่งๆ สามารถเรียกใช้โครงสร้างอื่นๆ ได้ เช่น โครงสร้างของรถยนต์อาจประกอบด้วยโครงสร้างของตัวถังและโครงสร้างของล้อ การเรียกใช้โครงสร้างมักทำร่วมกับการทำโมเดลลิง ทรานส์ฟอร์เมชัน (Modeling Transformation) เมื่อใช้ความสามารถทั้ง 2 อย่างนี้

จะช่วยให้การสร้างแบบจำลองทำได้ง่ายขึ้น เช่น การสร้างแบบจำลองของรถยนต์ ทำได้โดยการเรียกโครงสร้างของตัวถัง และเรียกโครงสร้างของล้อรถยนต์ 4 ครั้ง แล้วทำโมเดลลิง ทรานส์ฟอร์เมชันเพื่อให้ล้ออยู่ในตำแหน่งที่ถูกต้อง

ความสัมพันธ์ของการเรียกใช้โครงสร้างจะมีลักษณะเป็นแบบเชิงลำดับชั้น (Hierarchical) เช่น โครงสร้างของร่างกาย จะมีความสัมพันธ์กับโครงสร้างของนิ้ว โดยผ่านโครงสร้างของแขนและมือตามลำดับ โดยมีตำแหน่งและทิศทางสัมพันธ์กับตำแหน่งและทิศทางของโครงสร้างแรก ฟิกส์เรียกความสัมพันธ์ของโครงสร้างในลักษณะนี้ว่า เครือข่ายโครงสร้าง

เนื่องจากความสัมพันธ์ของเครือข่ายโครงสร้างเป็นแบบเชิงลำดับชั้น จึงทำให้มีการสืบทอดลักษณะประจำจากโครงสร้างแม่ (Parent Structure) ไปยังโครงสร้างลูก (Child Structure) ในขณะที่ลักษณะประจำของโครงสร้างลูกจะไม่มีผลต่อโครงสร้างแม่

2.11 อุปกรณ์นำเข้า (Blake, 1993)

อุปกรณ์นำเข้าของฟิกส์มีลักษณะเป็นอุปกรณ์นำเข้าเชิงตรรกะ (Logical Input Device) จึงทำให้มีลักษณะไม่ขึ้นกับอุปกรณ์ ซึ่งจะขึ้นอยู่กับการใช้อุปกรณ์จริงให้เหมาะสมกับอุปกรณ์นำเข้าของฟิกส์อย่างไร

2.11.1 ชนิดของอุปกรณ์นำเข้า

อุปกรณ์นำเข้าของฟิกส์มีด้วยกัน 6 ชนิด คือ

2.11.1.1 โลเคเตอร์ (Locator) ใช้ในการเลือกจุด หรือใช้บอกตำแหน่งของพิกัดโลกบนเวิร์กสเตชันของฟิกส์มักใช้กับอุปกรณ์ประเภทเมาส์

2.11.1.2 สโตรก มีลักษณะเหมือนโลเคเตอร์ แต่ใช้บอกกลุ่มของจุด มักใช้ในการวาดภาพแบบฟรีแฮนด์ (Freehand)

2.11.1.3 พิก ใช้ในการเลือกรูปทรงพื้นฐานที่แสดงอยู่บนเวิร์กสเตชัน

2.11.1.4 ชอยซ์ (Choice) มีลักษณะเหมือนรายการเลือก

2.11.1.5 แวลูเอเตอร์ (Valuator) ใช้เลือกค่าอิงดรรชนี (Floating point) ที่อยู่ในขอบเขตที่กำหนด มีลักษณะเหมือนสไลเดอร์ ในเอกซ์วิว หรือไดแอล บอกซ์ (Dial Box) ในทูลคิตของวินโดว์

2.11.1.6 สายอักขระ (String) ใช้รับสายของอักขระ

ค่าปัจจุบันของอุปกรณ์นำเข้าเชิงตรรกะเรียกว่า เมเชอร์ (Measure) และเรียกวิธีการเปลี่ยนค่าของเมเชอร์ว่า ทริกเกอร์ (Trigger) เช่น เมเชอร์ของโลเคเตอร์ คือ จุดหนึ่งบนพิกัดโลก และการเลือกค่าเมเชอร์ใหม่ โดยการเลื่อนตัวชี้ตำแหน่ง (Cursor) ไปตำแหน่งใหม่ แล้วกดปุ่มบนเมาส์ ซึ่งจะเรียกปุ่มของเมาส์ว่า ทริกเกอร์

อุปกรณ์นำเข้าแต่ละตัวสามารถอ้างถึงได้ โดยใช้เบอร์ของอุปกรณ์ (Device Number) เนื่องจากเวิร์กสเตชันเดียวกันสามารถมีอุปกรณ์นำเข้าได้มากกว่า 1 ชนิด และมีอุปกรณ์นำเข้าชนิดเดียวกันได้มากว่า 1 ตัว

2.11.2 ภาวะโอเปอร์เรชัน (Operation Mode) ของอุปกรณ์นำเข้า

ภาวะโอเปอร์เรชันของอุปกรณ์นำเข้ามี 3 ภาวะ คือ

2.11.2.1 ภาวะการร้องขอ (Request Mode) เมื่อมีการเรียกใช้ภาวะนี้ กระบวนการเมเซอร์ (Measure Process) จะเริ่มทำงานโดยการแสดงให้ผู้ใช้รู้ว่ามีการรอรับข้อมูลเข้า โดยการแสดงตัวพร้อม (Prompt) ของอุปกรณ์ และทำการเอคโค (Echo) เมื่อมีการรับข้อมูลที่ถูกต้อง ซึ่งจะมีเพียง 2 วิธี ที่จะกลับสู่การทำงานปกติของโปรแกรม คือ การส่งทริกเกอร์ ว่าได้รับข้อมูลที่ถูกต้อง (Valid Data) หรือการหยุด (Break) เท่านั้น

2.11.2.2 ภาวะการสุ่ม (Sample Mode) ในภาวะนี้ จะไม่มีการบล็อก (Block) การทำงานของโปรแกรม เมื่ออุปกรณ์ถูกสุ่มแล้ว กระบวนการเมเซอร์ของอุปกรณ์ ก็จะส่งค่าปัจจุบันของเมเซอร์กลับทันที

2.11.2.3 ภาวะอีเวนต์ (Event Mode) ในภาวะนี้ การกระทำใดๆ ของอุปกรณ์นำเข้า จะถูกบันทึกลงบนคิว (Queue) ของข้อมูลเข้า เมื่อโปรแกรมต้องการข้อมูลเข้าจะทำได้โดยการอ่านดูในคิวของข้อมูลเข้าว่ามีข้อมูลเข้าที่ต้องการหรือไม่ ถ้ามีก็นำข้อมูลในคิวออกมา

2.12 เวิร์กสเตชัน (Sun Microsystems, 1991)

เวิร์กสเตชันของฟิกส์จะเป็นตัวประสานเชิงตรรกะ (Logical Interface) ให้กับโปรแกรมประยุกต์เพื่อใช้ควบคุมอุปกรณ์จริง ฟิกส์แบ่งประเภทของเวิร์กสเตชันออกเป็น 5 ลักษณะ คือ

2.12.1 แบบรับข้อมูลเข้าอย่างเดียว (Input Only) เวิร์กสเตชันนี้จะต้องมีอุปกรณ์นำเข้าเชิงตรรกะอย่างน้อย 1 อุปกรณ์

2.12.2 แบบแสดงผลอย่างเดียว (Output Only) จะใช้สำหรับแสดงภาพเท่านั้น

2.12.3 แบบนำเข้าและแสดงผล (Input and Output) เป็นเวิร์กสเตชันที่มีลักษณะร่วมกันของ 2 แบบแรก

2.12.4 แบบนำเข้าเป็นเมต้าไฟล์ (Metafile Input)

2.12.5 แบบแสดงผลเป็นเมต้าไฟล์ (Metafile Output)

สำหรับซันฟิกส์รุ่น 2.0 มีเวิร์กสเตชันให้ใช้ 2 ชนิด คือ แบบแฟ้มแผนที่บิต (Bitmap File) ที่เรียกว่า คอมพิวเตอร์ กราฟิก เมต้าไฟล์ (Computer Graphics Metafile ย่อว่า

CGM) และแบบวินโดว์ของ เอกซ์วินโดว์ ซึ่งเป็นแบบที่นิยมใช้กันมาก และสำหรับเวิร์กสเตชันแบบวินโดว์นี้ ยังแบ่งออกเป็น 2 ลักษณะ คือ

2.12.1 แบบเอกซ์ทูล (X Tool) สามารถเป็นได้ทั้งเวิร์กสเตชันแบบนำเข้าอย่างเดียว แบบแสดงผลอย่างเดียว และแบบนำเข้าและแสดงผล การทำงานทั้งหมดของเวิร์กสเตชันชนิดนี้จะถูกจัดการโดยฟิกส์

2.12.2 แบบเอกซ์ ดรอเอเบิล (X Drawable) เป็นเวิร์กสเตชันแบบแสดงผลอย่างเดียว เวิร์กสเตชันแบบนี้จะอนุญาติให้โปรแกรมประยุกต์สามารถจัดการกับเอกซ์วินโดว์ได้โดยตรง เนื่องจากอุปกรณ์นำเข้าของฟิกส์จะไม่สามารถใช้งานได้ในเวิร์กสเตชันแบบนี้ อุปกรณ์นำเข้าต่างๆ จึงต้องผ่านการจัดการจากเอกซ์วินโดว์

ที่เก็บข้อมูลของเวิร์กสเตชัน เรียกว่า รายการสถานะของเวิร์กสเตชัน (Workstation State List ย่อว่า WSL) แต่ละเวิร์กสเตชันจะมีรายการสถานะเวิร์กสเตชันเป็นของตัวเอง รายการ สถานะเวิร์กสเตชันประกอบด้วยข้อมูลหลายอย่าง เช่น ตารางสีของเวิร์กสเตชัน (Workstation Color Table) ตารางวิว (View Table) ภาวะการทำงาน รายการของโครงสร้างที่ถูกส่งมา เป็นต้น

2.13 การเก็บถาวร (Archive) (Gaskins, 1991)

ข้อมูลต่างๆ ของหน่วยเก็บโครงสร้างส่วนกลางสามารถเก็บลงในแฟ้มได้ ซึ่งแฟ้ม ประเภทนี้เรียกว่า แฟ้มเก็บถาวร การบันทึกข้อมูลสามารถเลือกบันทึกข้อมูลของหน่วยเก็บโครงสร้างส่วนกลางที่ต้องการได้ โดยการอ้างถึงโครงสร้างแล้ว การเก็บถาวรจะทำการบันทึกข้อมูลทั้งหมดของ โครงสร้างที่ระบุลงในแฟ้ม

```
ARF_BEGARF "simple.phar";
ARF_ARFDESC "@(#)SunPHIGS2.0 Created on Fri Oct 18 13:30:00 1991";
ARF_ ARFVERSION 2;

ARF_BEGSTRUCT 1;
ARF_LINE3
    (0.2, 0.2, 0)
    (0.4, 0.7, 0)
    (0.6, 0.3, 0)
ARF_ENDSTRUCT;
ARF_ENDARF;
```

รูป 2.14 ข้อมูลในแฟ้มเก็บถาวร (Gaskins, 1991)

2.14 ฟังก์ชันการสอบถาม (Inquiry Function)

ฟิกส์มีแหล่งเก็บข้อมูลที่สำคัญ 4 ชนิด คือ

2.14.1 รายการสถานะของฟิกส์ (PHIGS State List) ใช้เก็บข้อมูลครอบคลุม (Global) ของสถานะฟิกส์ (PHIGS State) เช่น รายการของเวิร์กสเตชันที่ถูกเปิดใช้งาน ตัวระบุ

โครงสร้าง (Structure Identifier) ที่ถูกสร้างขึ้น หรือภาวะการแก้ไข (Edit Mode) ปัจจุบัน เป็นต้น

2.14.2 รายการสถานะของเวิร์กสเตชัน (Workstation State List) ใช้เก็บข้อมูลปัจจุบันของเวิร์กสเตชันนั้นๆ เช่น รายการของโครงสร้างที่ถูกส่งมา สถานะของการปรับภาพ เป็นต้น

2.14.3 ตารางอธิบายข้อมูลฟิกส์ (PHIGS Description Table) ใช้เก็บข้อมูลความสามารถ (Capability) ของฟิกส์ เช่น จำนวนที่มากที่สุดของเวิร์กสเตชันที่ถูกเปิดใช้งานได้พร้อมกัน ชนิดของเวิร์กสเตชันที่มีให้ใช้ เป็นต้น

2.14.4 ตารางอธิบายข้อมูลเวิร์กสเตชัน (Workstation Description Table) ใช้เก็บข้อมูลเกี่ยวกับความสามารถ และข้อจำกัดของเวิร์กสเตชัน เช่น แบบจำลองสีที่สนับสนุนชนิดของเส้นที่สามารถแสดงได้ เป็นต้น

การเข้าถึงข้อมูลต่างๆ เหล่านี้ ทำได้โดยใช้ฟังก์ชันการสอบถาม ซึ่งอาร์กิวเมนต์ (Argument) ในฟังก์ชันจะมี 2 ส่วน คือ ลักษณะของข้อมูลที่ต้องการรู้ และตัวชี้บอกความผิดพลาด (Error Indicator) เพื่อใช้บอกความผิดพลาดที่เกิดขึ้น เช่น กรณีไม่มีค่าของข้อมูลนั้นอยู่ในรายการ เป็นต้น

2.15 ฟังก์ชันเอสเคป (Escape Function)

ในกรณีที่การทำให้เกิดผลของฟิกส์ต้องการเพิ่มฟังก์ชันให้กับฟิกส์ สามารถทำได้โดยผ่านฟังก์ชันเอสเคป ซึ่งฟังก์ชันเอสเคปนี้มักใช้ในการแก้ปัญหาเฉพาะของฟิกส์ในบางกรณี เช่น เมื่อใช้เวิร์กสเตชันแบบเอกซ์ ดรอเอเบิลจะไม่สามารถใช้อุปกรณ์นำเข้าของฟิกส์ได้ เนื่องจากระบบ พิกัดของเอกซ์วินโดว์เป็นคนละชนิดกับพิกัดโลกของฟิกส์ จึงต้องมีฟังก์ชันเอสเคป เพื่อใช้ในการแปลงข้อมูล

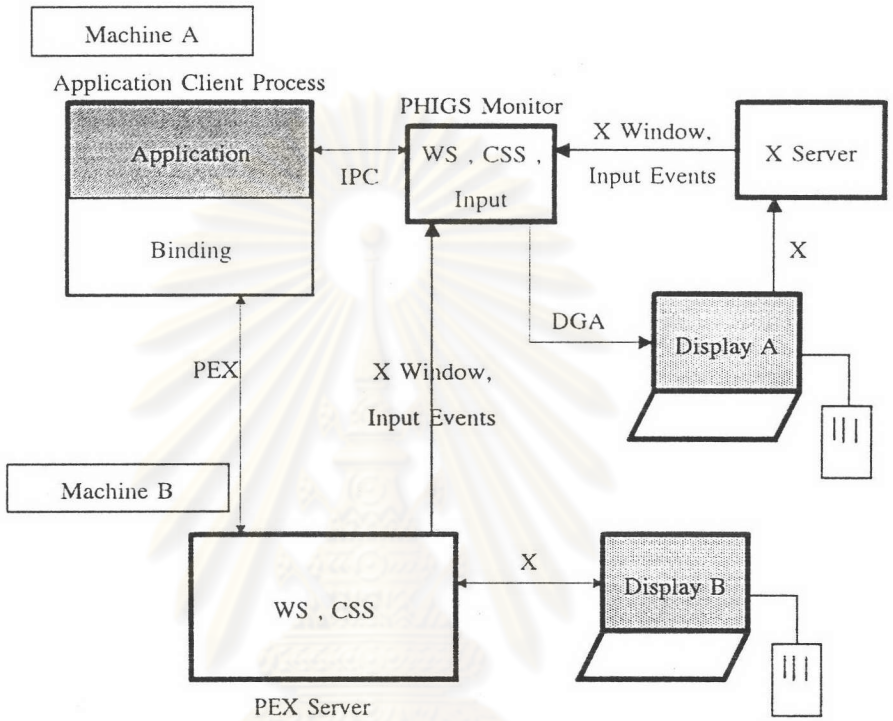
3. ความสัมพันธ์ระหว่างซันฟิกส์รุ่น 2.0 กับระบบเอกซ์วินโดว์ (Sun Microsystems, 1991)

ซันฟิกส์รุ่น 2.0 มีเวิร์กสเตชันอยู่ 2 ชนิดที่ต้องทำงานเกี่ยวข้องกับระบบเอกซ์วินโดว์ คือ แบบเอกซ์ทูล และ แบบเอกซ์ดรอเอเบิล

3.1 เอกซ์ทูล

โปรแกรมประยุกต์ที่มีการใช้งานเอกซ์ทูลจะทำให้เกิดกระบวนการ 2 กระบวนการ กระบวนการแรกจะเป็นกระบวนการของโปรแกรมประยุกต์นั้นเอง ส่วนกระบวนการที่สองจะเรียกว่า ฟิกส์มอนิเตอร์ (PHIGS monitor) ซึ่งใช้ควบคุมการติดต่อกับระบบเอกซ์วินโดว์ทั้งหมด กระบวนการของฟิกส์มอนิเตอร์จะเกิดขึ้นบนเครื่องเดียวกันกับกระบวนการของโปรแกรมประยุกต์เสมอ

โปรแกรมประยุกต์จะไม่สามารถติดต่อกับฟิกส์มอนิเตอร์โดยตรงได้ การติดต่อกับฟิกส์มอนิเตอร์จะทำได้โดยผ่านฟังก์ชันของซันฟิกส์รุ่น 2.0 ที่ได้ผูกติดกับภาษาซีหรือฟอร์แทรนไว้



รูป 2.15 การทำงานของซันฟิกส์รุ่น 2.0 แบบที่ใช้เอกซ์ทูล (Sun Microsystems, 1991)

จากรูป 2.15 กระบวนการทั้งสองจะติดต่อกันได้ โดยใช้การติดต่อระหว่างกระบวนการ (Interprocess Communication ย่อว่า IPC) สำหรับเอกซ์ทูลที่ทำงานบนเครื่องเดียวกันกับกระบวนการของฟิกส์มอนิเตอร์ ฟิกส์มอนิเตอร์จะจัดการทั้งส่วนของข้อมูลเข้าและออกส่วนหน่วยเก็บโครงสร้างส่วนกลาง รวมทั้งควบคุมการติดต่อกับระบบเอกซ์วินโดว์ ฟิกส์มอนิเตอร์จะไม่ส่งข้อมูลออกที่เป็นรูปกราฟิกไปให้เอกซ์ เซิร์ฟเวอร์ แต่จะส่งไปยังส่วนอุปกรณ์แสดงผลโดยตรงด้วยการเข้าถึงโดยตรงแบบกราฟิก (Direct Graphics Access ย่อว่า DGA) ฟิกส์มอนิเตอร์จะติดต่อกับเอกซ์เซิร์ฟเวอร์เพื่อรับอีเวนต์ของเอกซ์วินโดว์และแปลงให้เข้ากับอุปกรณ์นำเข้าเชิงตรรกะของฟิกส์

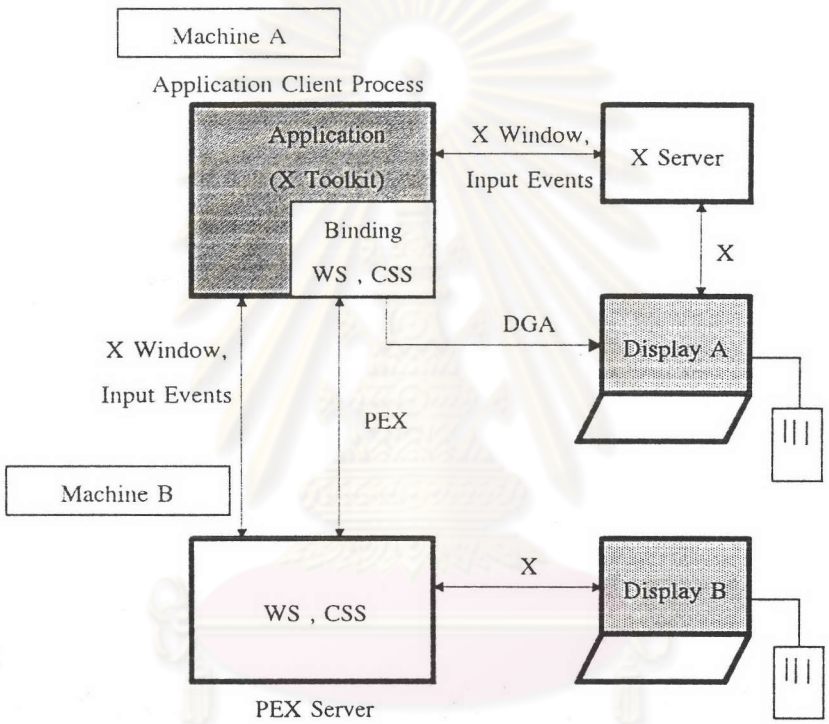
กรณีที่เอกซ์ทูลไม่ได้ทำงานบนเครื่องเดียวกันกับฟิกส์มอนิเตอร์ บนเครื่องที่เอกซ์ทูลทำงานอยู่นั้นจำเป็นต้องมีเพกซ์เซิร์ฟเวอร์อยู่ด้วย ในกรณีนี้ฟิกส์มอนิเตอร์จะจัดการกับข้อมูล

I17520988

เข้าที่ได้จากอีเวนต์ของเอกซ์วินโดว์ให้ ส่วนหน่วยเก็บโครงสร้างส่วนกลางและข้อมูลออกของโปรแกรมประยุกต์นั้นจะถูกจัดการโดยเพกซ์ เซิร์ฟเวอร์ โดยใช้เพกซ์ โปรโตคอลในการติดต่อกัน

3.2 เอกซ์ ดรอเอเบิล (X Drawable)

การใช้เวิร์กสเตชันชนิดนี้จะมีกระบวนการเกิดขึ้นเพียง 1 กระบวนการ คือ กระบวนการของโปรแกรมประยุกต์นั้น การจัดการส่วนของข้อมูลออกและส่วนของหน่วยเก็บโครงสร้างส่วนกลางจะอยู่ภายในกระบวนการนี้ด้วย



รูป 2.16 การทำงานของซันฟิกส์รุ่น 2.0 แบบที่ใช้เอกซ์ดรอเอเบิล (Sun Microsystems, 1991)

โปรแกรมประยุกต์จะถูกสร้างขึ้นมาด้วยเอกซ์ลิบหรือเอกซ์ทูลคิตร่วมกับซันฟิกส์รุ่น 2.0 สำหรับเวิร์กสเตชันแบบเอกซ์ดรอเอเบิลที่ทำงานบนเครื่องเดียวกัน ข้อมูลออกที่เป็นรูปกราฟิกจะถูกส่งตรงไปยังส่วนอุปกรณ์ โดยไม่ผ่านเอกซ์เซิร์ฟเวอร์ด้วยการเข้าถึงโดยตรงแบบกราฟิก ส่วนเอกซ์ดรอเอเบิลที่ไม่ได้ทำงานบนเครื่องเดียวกัน ข้อมูลออกและหน่วยเก็บโครงสร้างส่วนกลางจะถูกจัดการโดยเพกซ์ เซิร์ฟเวอร์ สำหรับข้อมูลเข้าทั้งในกรณีที่ทำงานบนเครื่องเดียวกันหรือคนละเครื่องจะถูกจัดการโดยเอกซ์ลิบหรือเอกซ์ทูลคิตที่ใช้