รายการอ้างอิง

ภาษาอังกฤษ

Bozdoc, M. 2001. The history of CAD[Online]. Available from: http://www.bozdoc.f2s.com/CAD-History.htm[2001, November 13].

Parasolid Technical Support Group. 2001. Parasolid V13.2 Functional Description. [Electronic book]. pp. 49-117. Cambridge: Unigraphics Solutions Inc.

Parasolid Technical Support Group. 2001. Parasolid XT Format. [Electronic book]. p.2. Cambridge: Unigraphics Solutions Inc.

Queensland Manufacturing Institute Ltd. 2001. Introduction to CAD/CAM for SMEs[Online]. Available from: http://www.qmi.asn.au/services/techassist/cad_brief.html[2001, June 26]

van Leeuwen, J. P., Wagter, H., Oxman, R. M. 1995. A Feature based approach to modelling Architectural Information. In: Modeling of Buildings through their Life-cycle. InFischer Law and Luiten. (eds.) Proceedings of CIB W78 Workshop.

บรรณานุกรม

ภาษาอังกฤษ

Beveridge, J., Wiener, R. 1996. Multithreading Applications in Win32: The complete guide to threads. (n.p.): Addisom Wesley Developers Press.

Chapman, D. 1998. Sams teach yourself Visual C++ 6 in 21 Days. Indianapolis IN: Sams Publishing.

Draxler, L. B. 2000. Windows programming under the hood of MFC: With a quick tour of Visual C++ tools. New Jersey: Prentice Hall.

Eddon, G., Eddon, H. 1999. Inside Microsoft COM+ base services. Redmond WA: Microsoft Press.

Ezzell, B. 2000. Mastering™ Windows 2000 programming with Visual C++. Alameda CA: SYBEX.

Gordon, A. 2000. The COM and COM+ programming premier. New Jersey: Prentice Hall.

Gregory, K. 1998. Special edition using Visual C++ 6. Indianapolis IN: QUE.

Hill, F.S., Jr. 2001. Computer graphics using OpenGL. 2nd ed. New Jersey: Prentice Hall.

Kirtland, M. 1999. Designing component-based applications. Redmond WA: Microsoft Press.

Kruglinski, J., Shepherd, G. and Wingo, S. 1998. Programming Microsoft Visual C++. 5th ed. Redmond WA: Microsoft Press.

McConnell, S. 1993. Code complete. Redmond WA: Microsoft Press.

Norman, R. J. 1996. Object-oriented systems analysis and design. New Jersey: Prentice Hall.

Parasolid Technical Support Group. 2001. Parasolid V13.2 Downward Interfaces. [Electronic book] Cambridge: Unigraphics Solutions Inc.

Parasolid Technical Support Group. 2001. Parasolid V13.2 Getting Started With Parasolid. [Electronic book] Cambridge: Unigraphics Solutions Inc.

Parasolid Technical Support Group. 2001. Parasolid V13.2 PK Interface Programming Reference Manual Part 1. [Electronic book] Cambridge: Unigraphics Solutions Inc.

บรรณานุกรม (ต่อ)

Parasolid Technical Support Group. 2001. Parasolid V13.2 PK Interface Programming Reference Manual Part 2. [Electronic book] Cambridge: Unigraphics Solutions Inc.

Tapadiya, P. 2001. COM+ programming: A practical guide using Visual C++ and ATL. New Jersey: Prentice Hall.

Woo, M., Neider, J., Davis, T., and Shreiner, D. 2000. OpenGL programming guide. 3rd ed. Delhi: Addison Wesley.

ภาคผนวก ก.

ภาคผนวก ก.

วิธีใช้งานซอฟแวร์ซียู-โซลิด (CU-Solid)

ก.1. การติดตั้งซียู-โซลิด

ก.1.1. ความต้องการของระบบในการติดตั้งซียู-โซลิด ซียู-โซลิด เหมือนกับซอฟแวร์คอมพิวเตอร์ช่วยในการออกแบบทั่วไปที่ต้องการอุปกรณ์คอมพิวเตอร์ที่มีประสิทธิภาพสูง เพื่อช่วยให้การใช้งานเป็นไปได้อย่างสะดวกและราบรื่น ก่อนการติดตั้งซียู-โซลิดจึงควรจัดเตรียมอุปกรณ์คอมพิวเตอร์ให้มีความพร้อมในการใช้งาน ดังนี้

ก.1.1.1. ซีพียู (CPU) ควรมีความเร็วไม่น้อยกว่า 200 เมกกะเฮิร์ท

ก.1.1.2. แรม (RAM) อย่างน้อย 64 เมกกะไบท์ (ควรจะเป็น 256 เมกกะไบท์เพื่อความราบรื่นในการใช้งาน)

ก.1.1.3. ฮาร์ดดิสก์ (Hard disk) มีที่ว่างอย่างน้อย 50 เมกกะไบท์

ก.1.1.4. วีดีโอการ์ด ที่มีความสามารถในการแสดงผลสามมิติสำหรับโอเพนจีเอล (OpenGL) มีหน่วยความจำไม่น้อยกว่า 8 เมกกะไบท์

ก.1.1.5. เมาส์ ควรเป็นชนิด 3 ปุ่ม

ก.1.1.6. ระบบปฏิบัติการวินโดว์ 2000 (Windows 2000)

ก.1.2. ขั้นตอนการติดตั้งซียู-โซลิด มีดังนี้

ก.1.2.1. เรียกโปรแกรมติดตั้งซียู-โซลิด “CU-Solid.msi”

ก.1.2.2. เมื่อปรากฏกรอบข้อความต้อนรับเข้าสู่การติดตั้งให้กดปุ่ม “Next”

ก.1.2.3. เลือกไดเรกทอรีที่ต้องการติดตั้งซียู-โซลิดแล้วกดปุ่ม “Next”

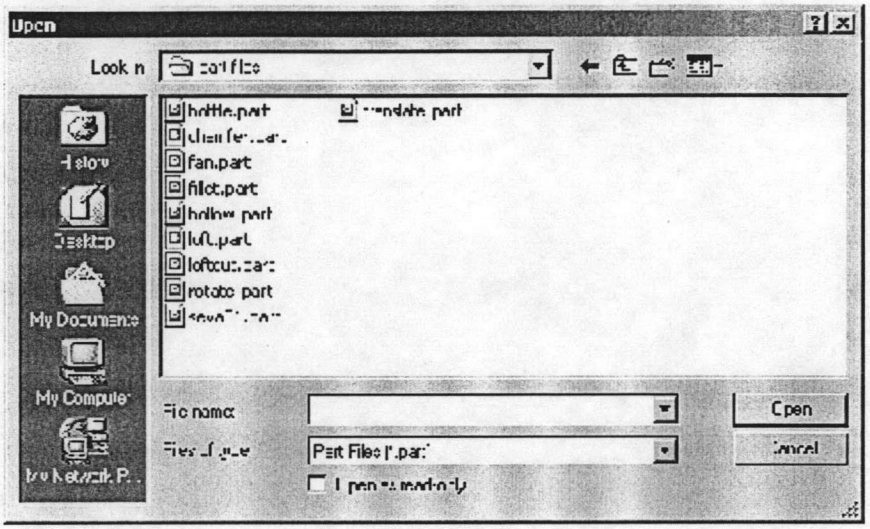
ก.1.2.4. เมื่อปรากฏกรอบโต้ตอบเพื่อยืนยันการติดตั้งให้กดปุ่ม “Next” จากนั้นโปรแกรมติดตั้งจะติดตั้งซียู-โซลิดในไดเรกทอรีที่เลือกเมื่อเสร็จแล้วให้กดปุ่ม “Finish” เป็นอันเสร็จสิ้นการติดตั้ง

ก.1.2.5. ในระหว่างการติดตั้งถ้าต้องการยกเลิกการติดตั้งก็สามารถทำได้โดยกดปุ่ม “Cancle”

ก.2. การใช้งานซียูโซลิดเบื้องต้น

ก.2.1. <u>การสร้างเอกสารแบบจำลองโซลิดใหม่</u> ที่เมนู "File" เลือกคำสั่ง "New" เพื่อสร้างเอกสารเปล่าสำหรับแบบจำลองโซลิด

ก.2.2. <u>การเปิดแฟ้มข้อมูลแบบจำลองโซลิด</u> ที่เมนู "File" เลือกคำสั่ง "Open" จะปรากฏกรอบโต้ตอบสำหรับเลือกแฟ้มข้อมูล ดังรูปที่ ก.1 เลือกแฟ้มข้อมูลที่ต้องการเปิด ซึ่งจะต้องเป็นแฟ้มข้อมูลที่มีนามสกุลเป็น "part" เท่านั้นแล้วกดปุ่มตกลง

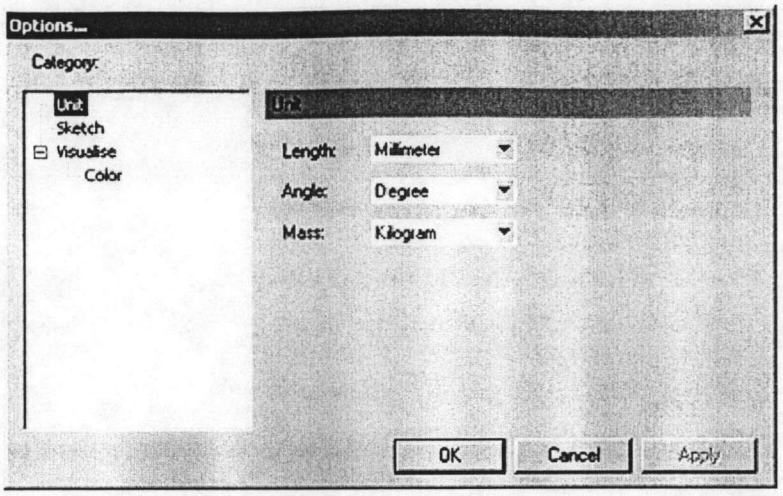


รูปที่ ก.1 แสดงกรอบโต้ตอบสำหรับเปิดแฟ้มข้อมูลแบบจำลองโซลิด

ก.2.3. <u>การบันทึกแฟ้มข้อมูลแบบจำลองโซลิด</u> ที่เมนู "File" เลือกคำสั่ง "Save" หรือ "Save as..." เพื่อบันทึกแบบจำลองโซลิดที่สร้าง

ก.2.4. <u>การส่งออกและนำเข้าแฟ้มข้อมูลชนิดอื่น ๆ</u> ที่เมนู "File" เลือกคำสั่ง "Import" เมื่อต้องการนำเข้า และเลือกคำสั่ง "Export" เมื่อต้องการส่งออกแบบจำลองโซลิดในรูปแบบแฟ้มข้อมูลชนิดอื่น ๆ นอกเหนือจากแฟ้มข้อมูลของซียู-โซลิด สำหรับเวอร์ชัน 1.0 ในส่วนตัวโปรแกมมาตรฐานนี้รองรับเพียงการส่งออกและนำเข้าแฟ้มข้อมูลชนิดพาราโซลิด XT เท่านั้น

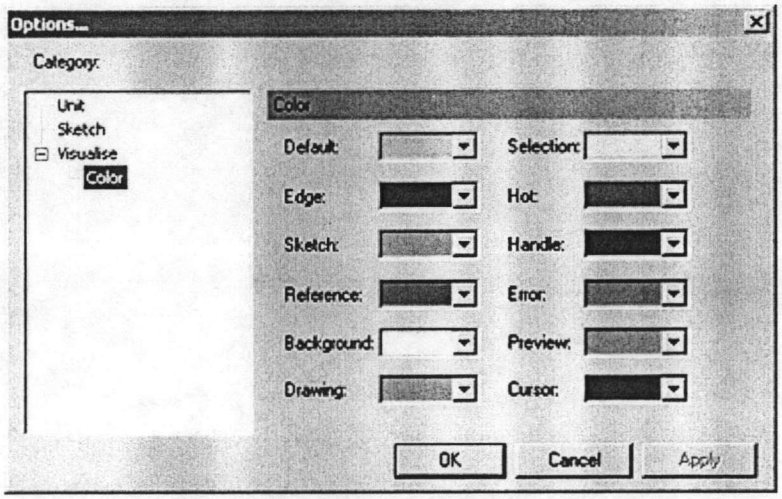
ก.2.5. <u>การกำหนดหน่วย</u> ที่เมนู "Tools" เลือกคำสั่ง "Option" จะปรากฏกรอบโต้ตอบจากนั้นเลือก "Unit" ที่ทรีด้านซ้ายของกรอบโต้ตอบ จะปรากฏคอมโบสำหรับการเปลี่ยนหน่วยดังรูปที่ ก.2 เปลี่ยนหน่วยที่ใช้ตามต้องการ โดยหน่วยที่สามารถกำหนดได้มี 3 ประเภท คือ ความยาว มุม และมวล ส่วนพื้นที่ และปริมาตรจะมีหน่วยสัมพันธ์กับหน่วยของความยาว



รูปที่ ก.2 แสดงกรอบโต้ตอบที่ใช้เปลี่ยนหน่วยที่ใช้สร้างแบบจำลองโซลิด

ก.2.6. <u>การปรับแต่งการแสดงผล</u>

ก.2.6.1. การปรับแต่งสี ที่เมนู "Tools" เลือกคำสั่ง "Options" จะปรากฏกรอบโต้ตอบ ให้เลือก "Visualize / Color" จะปรากฏชนิดของสีที่ต้องการปรับค่าดังรูปที่ ก.3 สำหรับซียู-โซลิดเวอร์ชัน 1.0 นี้สามารถปรับสีได้ทั้งหมด 12 ชนิดด้วยกันคือ

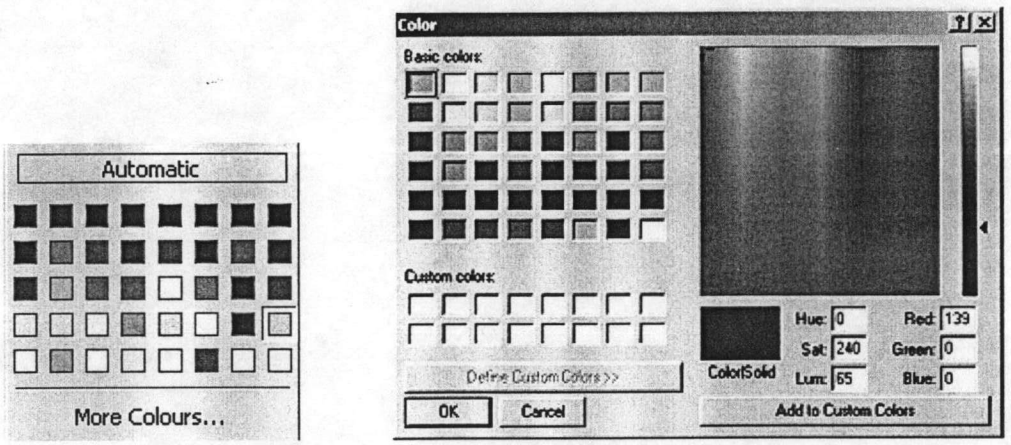


รูปที่ ก.3 แสดงกรอบโต้ตอบในการเปลี่ยนสีของการแสดงผล

1. สีของแบบจำลองโซลิดที่ไม่ได้กำหนดวัสดุ (Default)
2. สีขอบ (Edge) เป็นสีที่จะแสดงขอบของแบบจำลองโซลิดในการแสดงผลแบบมีแสงเงาพร้อมแสดงเส้นขอบ (Shade with edge)
3. สีแบบร่างบนระนาบ (Sketch) เป็นสีที่ใช้แสดงเส้นแบบร่างบนระนาบที่สร้างเสร็จแล้ว

4. สีระนาบอ้างอิง (Reference)
5. สีพื้นหลัง (Background)
6. สีเซกเมนต์ของแบบร่างบนระนาบ (Drawing) เป็นสีที่แสดงเซกเมนต์ของแบบร่างบนระนาบที่ถูกสร้างในโหมดการสร้างแบบร่างบนระนาบ
7. สีองค์ประกอบที่ถูกเลือก (Selection)
8. สีองค์ประกอบที่ถูกชี้ด้วยเมาส์ (Hot)
9. สีวัตถุและสัญลักษณ์ (Handle)
10. สีองค์ประกอบที่เกิดความผิดพลาด (Error)
11. สีพรีวิวของแบบจำลองที่กำลังสร้าง (Preview)
12. สีเคอร์เซอร์ในโหมดการสร้างแบบร่างบนระนาบ (Cursor)

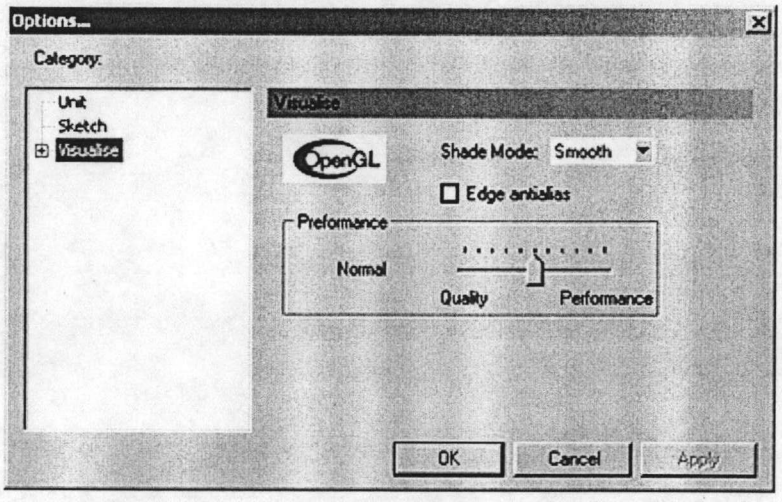
เมื่อต้องการเปลี่ยนสีของส่วนประกอบใดให้คลิ๊กปุ่มที่มีลูกศรชี้ลงในช่องสีที่ต้องการเปลี่ยน จะปรากฏกรอบแสดงแถบสีต่าง ๆ ดังรูปที่ ก.4 ถ้าต้องการใช้ค่าสีเริ่มต้นของซียู-โซลิดให้กดปุ่ม “Automatic” หรือถ้าต้องการเลือกสีอื่น ๆ นอกเหนือจากแถบสีที่ปรากฏให้กดปุ่ม “More Colors...”



รูปที่ ก.4 แสดงกรอบสีที่ใช้เปลี่ยนสีการแสดงผล

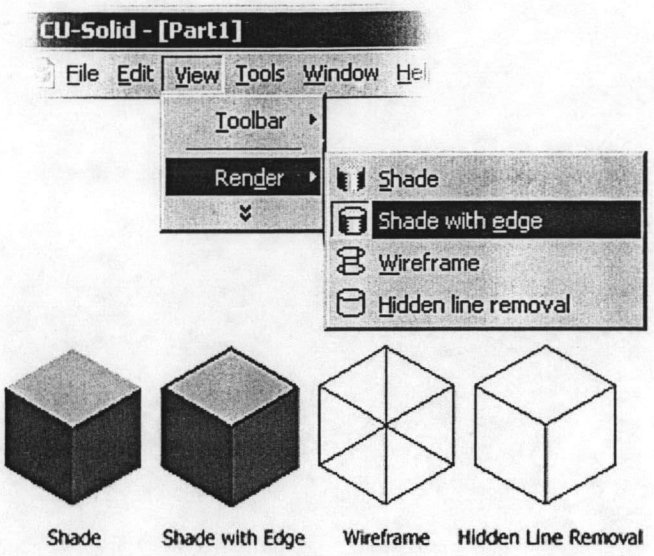
ก.2.6.2. การปรับแต่งประสิทธิภาพการแสดงผล ที่เมนู Tools เลือกคำสั่ง Options จะปรากฏกรอบโต้ตอบ ให้เลือก Visualize จะปรากฏแถบเลื่อนปรับประสิทธิภาพดังรูปที่ ก.5 จากนั้นปรับค่าประสิทธิภาพการแสดงผลโดยการเลื่อนแถบเลื่อน (Slider bar) ไปมาระหว่าง “Quality” และ “Performance” ถ้าเลื่อนให้ใกล้ค่า Quality คุณภาพของภาพสามมิติของ

แบบจำลองที่ได้จะสูง แต่ความเร็วของการแสดงผลจะลดลง ในขณะที่ถ้าเลื่อนแถบไปทาง "Performance" ความเร็วในการแสดงผลจะสูงแต่คุณภาพของภาพแบบจำลองสามมิติที่ได้จะต่ำ ในการปรับค่านี้ปรับเพื่อให้เหมาะสมกับประสิทธิภาพของเครื่องคอมพิวเตอร์ และวีดีโอการ์ดที่ใช้กับซียู-โซลิด



รูปที่ ก.5 แสดงการปรับแต่งประสิทธิภาพการแสดงผล

ก.2.6.3. การเปลี่ยนชนิดการแสดงผล ที่เมนู View -> Render เลือกชนิดของการแสดงผลสามมิติที่ต้องการ โดยซียู-โซลิดสามารถแสดงผลได้ 4 แบบคือ การแสดงผลแบบมีแสงเงา (Shade) การแสดงผลแบบมีแสงเงาพร้อมแสดงเส้นขอบ (Shade with Edge) การแสดงผลแบบโครงลวด (Wireframe) และการแสดงผลแบบโครงลวดที่ซ่อนเส้นที่ถูกทับ (Hidden line removal)



รูปที่ ก.6 การแสดงผลแบบต่าง ๆ ด้วยซียู-โซลิด

ก.2.7. <u>การเปลี่ยนมุมมอง</u>

ก.2.7.1. การหมุน (Rotate) เลือกคำสั่งหมุน จากแถบเครื่องมือ หรือจากเมนู View ก็ได้ เมื่อต้องการหมุน ให้กดเมาส์ปุ่มซ้ายค้างไว้แล้วเลื่อนเมาส์ไปในทิศที่ตั้งฉากกับทิศของแกนที่ต้องการหมุนรอบ

ก.2.7.2. การเลื่อนตำแหน่ง (Pan) เลือกคำสั่งเลื่อนตำแหน่ง จากแถบเครื่องมือ หรือจากเมนู View ก็ได้ เมื่อต้องการเลื่อนตำแหน่งมุมมองไปที่ใด ให้กดปุ่มซ้ายของเมาส์ค้างไว้แล้วเลื่อนเมาส์ไปยังตำแหน่งที่ต้องการ โดยตำแหน่งการมองจะเคลื่อนที่สวนทางกับการเคลื่อนที่ของเมาส์

ก.2.7.3. การซูม (Zoom) เลือกคำสั่งซูม จากแถบเครื่องมือ หรือจากเมนู View ก็ได้ เมื่อต้องการซูมให้กดปุ่มซ้ายของเมาส์ค้างไว้แล้วเลื่อนขึ้นหรือลง เพื่อซูมเข้าและออกตามลำดับ

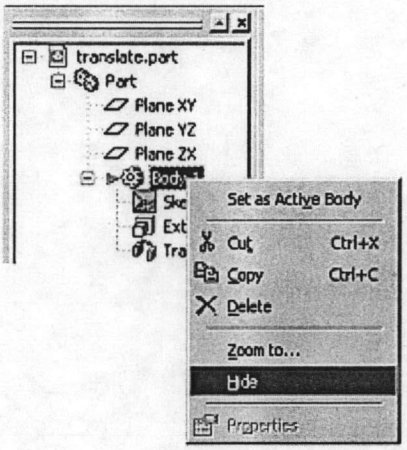
ก.2.7.4. การซูมพอดีจอ (Fit to Screen) เลือกคำสั่งซูมพอดีจอ จากแถบเครื่องมือ ซียู-โซลิดจำเปลี่ยนมุมมองให้แบบจำลองโซลิดที่มีอยู่มาอยู่ตรงกลางจอ และแสดงขนาดที่ใหญ่พอดีกับจอ

ก.2.7.5. การซูมหน้าต่าง (Zoom Window) เลือกคำสั่งซูมหน้าต่าง จากแถบเครื่องมือ จากนั้นลากเมาส์ให้หน้าต่างสี่เหลี่ยมล้อมรอบบริเวณที่ต้องการซูม เมื่อปล่อยเมาส์ซียู-โซลิดจะซูมไปยังบริเวณที่เลือก

นอกจากการเลือกคำสั่งการเปลี่ยนมุมมองจากแถบเครื่องมือแล้ว ผู้ใช้สามารถเปลี่ยนมุมมองได้จากลำดับการกดปุ่มของเมาส์ โดยเริ่มต้นถ้ากดปุ่มกลางค้างไว้ แล้วเลื่อนเมาส์ไปมา จะ

เป็นการเลื่อนตำแหน่ง ในขณะที่กดปุ่มกลางค้างไว้ให้กดปุ่มซ้ายมือตาม จะเป็นการหมุน เมื่อปล่อยปุ่มซ้าย ขณะที่ปุ่มกลางยังคงกดค้างอยู่จะกลายเป็นการซูม

ก.2.8. การซ่อนแบบจำลอง (Hide) ที่ทรีของลักษณะจำเพาะ คลิ๊กเมาส์ขวาบนแบบจำลองที่ต้องการซ่อนจะปรากฏเมนูป๊อบอัพ ให้เลือกคำสั่งซ่อน (Hide) และเมื่อต้องการให้กลับมาปรากฏคืนให้ทำซ้ำเดิมอีกครั้ง



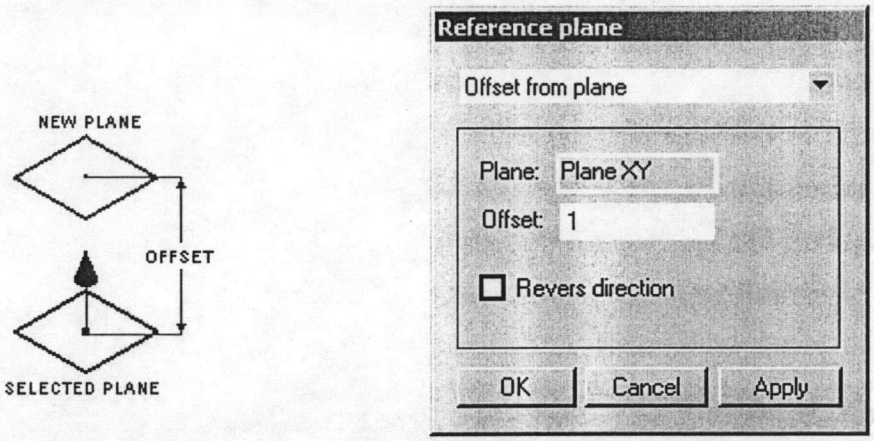
รูปที่ ก.7 แสดงการซ่อนภาพแบบจำลองโซลิด

ก.3. การสร้างระนาบอ้างอิง (Reference Plane)

เมื่อเริ่มเอกสารใหม่ ซียู-โซลิด จะสร้างระนาบอ้างอิงสามระนาบ ซึ่งเป็นระนาบอ้างอิงมาตรฐาน คือ ระนาบ XY ระนาบ YZ และระนาบ ZX ใช้เพื่อระบบแกนอ้างอิงรวม (World Coordinate System) ของแบบจำลองที่จะสร้าง เมื่อต้องการสร้างระนาบอ้างอิงเพิ่มเพื่อใช้ช่วยในการสร้างแบบจำลองสามารถทำได้โดย เริ่มต้นจากเลือกคำสั่งสร้างระนาบอ้างอิง (Reference Plane) จากแถบเครื่องมือ จากนั้นจะปรากฏกรอบโต้ตอบ สำหรับกำหนดวิธีการสร้างระนาบอ้างอิง และพารามิเตอร์ที่ต้องการ วิธีการสร้างระนาบอ้างอิงด้วยซียู-โซลิดนี้มีหลายวิธีคือ

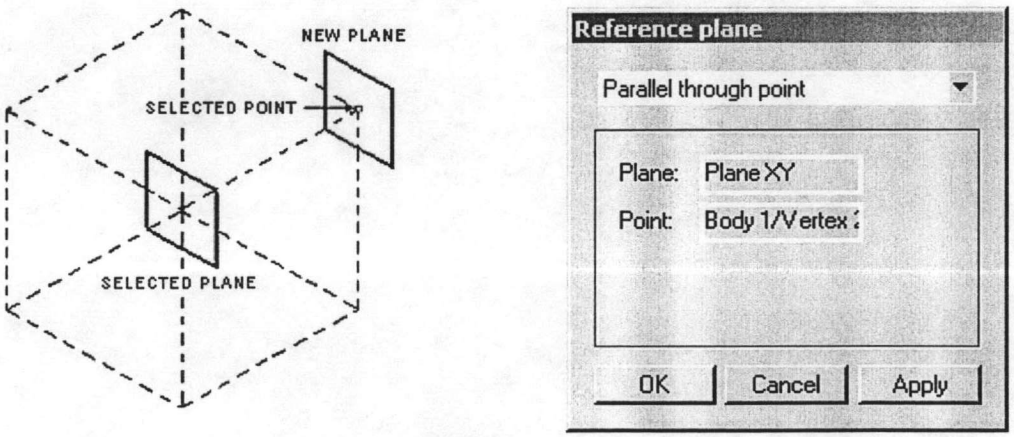
ก.3.1. ออฟเซตจากระนาบ (Offset From Plane) เมื่อต้องการสร้างระนาบอ้างอิงโดยการออฟเซตจากระนาบที่มีอยู่เดิม เมื่อปรากฏกรอบโต้ตอบสำหรับการสร้างระนาบอ้างอิง ที่กรอบคอมโบ (Combo box) ให้เลือกเป็น "Offset from plane" จะได้กรอบโต้ตอบดังรูปที่ ก.8 เลือกระนาบที่ต้องการนำมาทำออฟเซต ซึ่งอาจเป็นระนาบอ้างอิงหรือ เป็นองค์ประกอบทางโทโปโลยีชนิดหน้าที่เป็นระนาบก็ได้ เมื่อเลือกระนาบได้แล้ว ในช่อง Plane จะปรากฏชื่อของระนาบที่เลือก จากนั้นกำหนดค่าออฟเซตในช่อง Offset ซึ่งจะต้องเป็นค่าบวกที่มากกว่าศูนย์เสมอ และถ้าต้องการกลับทิศการออฟเซต สามารถทำได้โดยเลือก Revers

direction ให้มีเครื่องหมายถูกปรากฏอยู่ในช่องสี่เหลี่ยม สุดท้ายกดปุ่มตกลง (OK) เพื่อจบการสร้างระนาบอ้างอิง



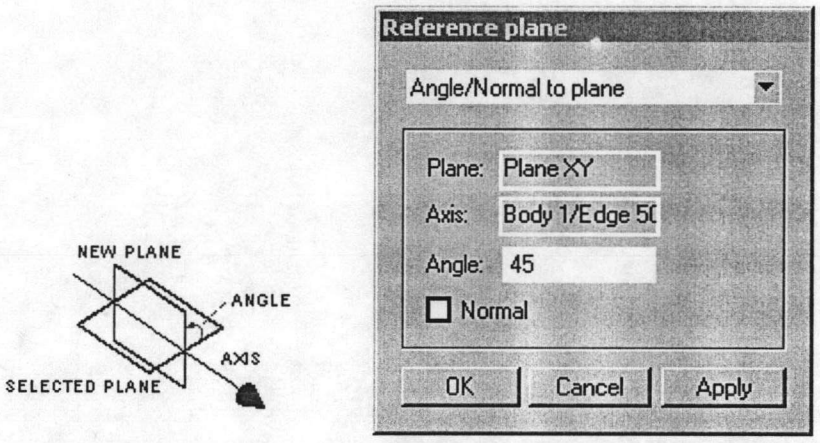
รูปที่ ก.8 แสดงการสร้างระนาบอ้างอิงด้วยการออฟเซตจากระนาบ

ก.3.2. ขนานกับระนาบที่จุด (Parallel Through Point) วิธีนี้เป็นการสร้างระนาบอ้างอิงโดยการกำหนดระนาบที่ขนานกับระนาบอ้างอิงที่ต้องการสร้าง และกำหนดจุดศูนย์กลางระนาบด้วยองค์ประกอบทางโทโปโลยีชนิดจุดยอด เมื่อปรากฏกรอบโต้ตอบสำหรับการสร้างระนาบอ้างอิง ที่กรอบคอมโบให้เลือกเป็น “Parallel through point” จะได้กรอบโต้ตอบดังรูปที่ ก.9 คลิ๊กที่ช่อง Plane เพื่อบอกซียู-โซลิดว่าต้องการเลือกระนาบที่ขนานกับระนาบอ้างอิงที่ต้องการสร้าง ระนาบนี้อาจเป็นระนาบอ้างอิงหรือ เป็นองค์ประกอบทางโทโปโลยีชนิดหน้าที่เป็นระนาบก็ได้ เมื่อเลือกระนาบได้แล้ว ในช่อง Plane จะปรากฏชื่อของระนาบที่เลือก จากนั้นให้คลิ๊กที่ช่อง “Point” และกำหนดจุดที่ต้องการสร้างระนาบอ้างอิงโดยการเลือกองค์ประกอบโทโปโลยีที่เป็นจุดยอด ณ ตำแหน่งที่ต้องการสร้างระนาบอ้างอ้าง สุดท้ายกดปุ่มตกลง (OK) เป็นการจบการสร้างระนาบอ้างอิง



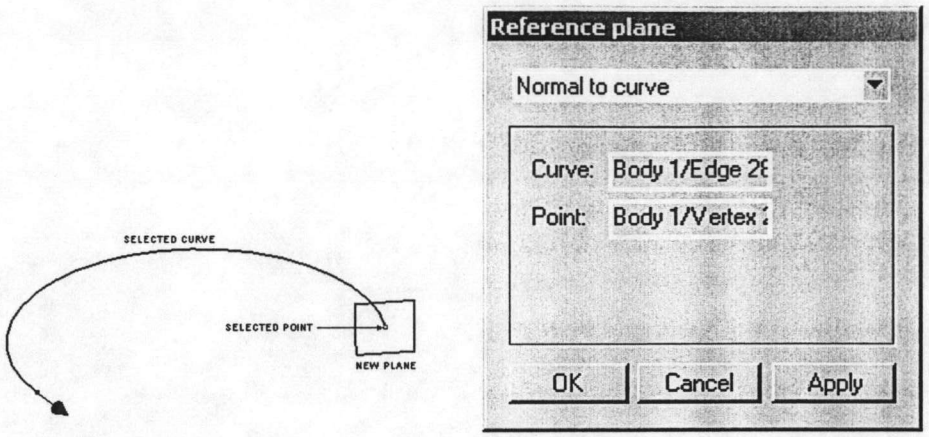
รูปที่ ก.9 แสดงการสร้างระนาบอ้างอิงที่ขนานกับระนาบใด ๆ ณ จุดที่กำหนด

ก.3.3. <u>ทำมุมใด ๆ หรือตั้งฉากกับระนาบ</u> (Angle/Normal to Plane) วิธีนี้เป็นการสร้างระนาบอ้างอิงที่ทำมุมกับระนาบใด ๆ เมื่อปรากฏกรอบโต้ตอบสำหรับการสร้างระนาบอ้างอิงแล้ว ให้เลือกกรอบคอมโบด้านบนของกรอบโต้ตอบเป็น "Angle/Normal to plane" จะได้กรอบโต้ตอบดังรูปที่ ก.10 คลิ๊กเมาส์ที่ช่อง "Plane" แล้วกำหนดระนาบที่ต้องการใช้เป็นตัวเทียบมุมกับระนาบอ้างอิงที่ต้องการสร้าง ระนาบที่เลือกนี้อาจเป็นระนาบอ้างอิง หรือองค์ประกอบทางโทโปโลยีชนิดหน้าที่เป็นระนาบก็ได้ เมื่อเลือกระนาบได้แล้วให้คลิ๊กเมาส์ที่ช่อง "Axis" แล้วเลือกแกนหมุนจากนั้นกำหนดมุมที่ต้องการในช่อง "Angle" หรือเลือกช่อง "Normal" ให้เป็นเครื่องหมายถูกเมื่อต้องการสร้างระนาบอ้างอิงที่ตั้งฉากกับระนาบที่เลือกในช่อง "Plane" สุดท้ายกดปุ่มตกลง (OK) เพื่อจบการสร้างระนาบอ้างอิง



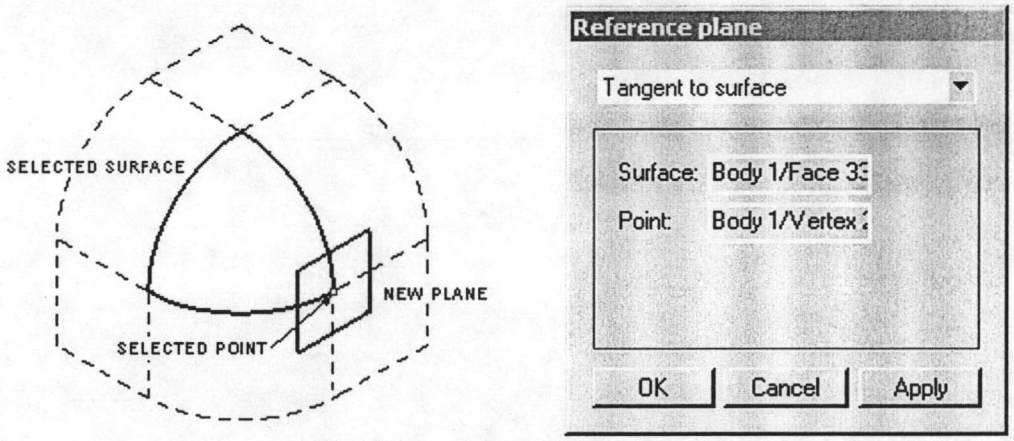
รูปที่ ก.10 แสดงการสร้างระนาบอ้างอิงที่ทำมุมหรือตั้งฉากกับระนาบ

ก.3.4. <u>ตั้งฉากกับเส้นโค้ง</u> (Normal to Curve) วิธีนี้เป็นการสร้างระนาบอ้างอิงที่ตั้งฉากกับเส้นโค้ง ณ จุดยอดใด ๆ ที่อยู่บนเส้นโค้ง ในการสร้างเมื่อปรากฏกรอบโต้ตอบให้เลือกกรอบคอมโบด้านบนให้เป็น "Normal to curve" จะได้กรอบโต้ตอบดังรูปที่ ก.11 คลิ๊กเมาส์ที่ช่อง "Curve" เพื่อเลือกเส้นโค้งที่ต้องการ เส้นโค้งที่เลือกนี้อาจเป็นองค์ประกอบทางโทโปโลยีชนิดขอบของแบบจำลองโซลิด หรือเป็นเซกเมนต์ของแบบร่างบนระนาบก็ได้ เมื่อเลือกเส้นโค้งได้แล้ว ให้คลิ๊กเมาส์ที่ช่อง "Point" เพื่อเลือกจุดยอด ณ ตำแหน่งที่ต้องการสร้างระนาบอ้างอิง สุดท้ายกดปุ่มตกลง (OK) เพื่อสิ้นสุดการสร้างระนาบอ้างอิง



รูปที่ ก.11 แสดงการสร้างระนาบอ้างอิงที่ตั้งฉากกับเส้นโค้ง ณ จุดบนเส้นโค้ง

ก.3.5. <u>สัมผัสกับพื้นผิว</u> (Tangent to Surface) วิธีการสร้างระนาบวิธีนี้จะสร้างระนาบที่ขนานกับพื้นผิวที่จุดยอดบนพื้นผิวที่กำหนด ในขั้นแรกเมื่อปรากฏกรอบโต้ตอบ ให้เลือกกรอบคอมโบเป็น “Tangent to surface” จะได้กรอบโต้ตอบดังรูปที่ ก.12 แล้วคลิ๊กเมาส์ที่ช่อง “Surface” เพื่อเลือกพื้นผิวที่ต้องการสร้างระนาบอ้างอิง โดยพื้นผิวที่เลือกนี้จะเลือกจากองค์ประกอบทางโทโปโลยีชนิดหน้าของแบบจำลองที่สร้าง จากนั้นคลิ๊กเมาส์ที่ช่อง “Point” แล้วเลือกจุดยอดในตำแหน่งที่ต้องการสร้างระนาบอ้างอิง สุดท้ายกดปุ่มตกลง (OK) เพื่อสิ้นสุดการสร้างระนาบอ้างอิง



รูปที่ ก.12 แสดงการสร้างระนาบอ้างอิงที่สัมผัสกับพื้นผิว

ก.4. การสร้างแบบร่างสองมิติบนระนาบ (Sketch)

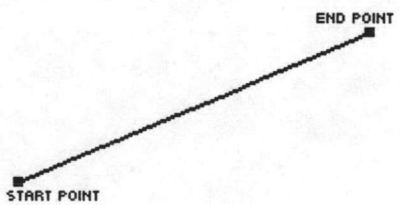
ซียู-โซลิดสร้างแบบจำลองโซลิดด้วยการเริ่มต้นจากการสร้างแบบร่างสองมิติบนระนาบ แล้วจึงนำไปสร้างเป็นแบบจำลองโซลิดด้วยคำสั่งอื่น ๆ ซียู-โซลิดสามารถสร้างแบบร่างสองมิติบนระนาบได้โดยมีขั้นตอนดังนี้

1. เลือกคำสั่งเริ่มสร้างแบบร่าง (Sketcher) จากแถบเครื่องมือ
2. ใช้เมาส์คลิกที่ระนาบที่ต้องการวาดแบบร่าง ระนาบที่เลือกนี้อาจเป็นระนาบอ้างอิง หรือเป็นหน้าของโซลิดที่เป็นระนาบก็ได้ เมื่อเลือกระนาบได้แล้ว ซียู-โซลิดจะเปลี่ยนมุมมองให้ตั้งฉากกับระนาบที่เลือก และเปลี่ยนแถบเครื่องมือเป็นชุดแถบเครื่องมือสำหรับการสร้างเซกเมนต์ของแบบร่างบนระนาบ
3. เลือกคำสั่งสร้างเซกเมนต์ของแบบร่างบนระนาบแบบต่าง ๆ ตามต้องการ
4. เมื่อต้องการสิ้นสุดการสร้างแบบร่างบนระนาบให้กดปุ่มสิ้นสุดการสร้างแบบร่างบนระนาบ (Exit Sketcher) จากแถบเครื่องมือ

ในการสร้างเซกเมนต์ของแบบร่างบนระนาบซียู-โซลิดสามารถสร้างได้หลายชนิด ซึ่งแต่ละชนิดมีวิธีการสร้างดังนี้

ก.4.1. <u>การสร้างเส้นตรง</u> เป็นการสร้างเซกเมนต์ของแบบร่างบนระนาบที่เป็นเส้นตรง มีวิธีการสร้างสองวิธีด้วยกันคือ

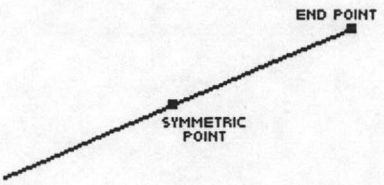
ก.4.1.1. การสร้างเส้นตรงด้วยจุดต้นและจุดปลาย (Line 2-Points) เป็นการสร้างเส้นตรงที่ผ่านจุดสองจุดที่กำหนด ในขั้นแรกเลือกคำสั่งสร้างเส้นตรงด้วยจุดสองจุด จากแถบเครื่องมือ แล้วใช้เมาส์คลิ๊กไปบนตำแหน่งที่ต้องการสองจุดเพื่อกำหนดจุดเริ่มต้น และจุดปลายของเส้นตรง



รูปที่ ก.13 แสดงการสร้างเส้นตรงด้วยจุดต้น และจุดปลาย

ก.4.1.2. การสร้างเส้นตรงด้วยจุดสมมาตรและจุดปลาย (Line Symmetry) ในการสร้างเส้นตรงด้วยวิธีนี้ในขั้นแรกให้เลือกปุ่มคำสั่งสร้างเส้นตรงด้วยวิธีการ

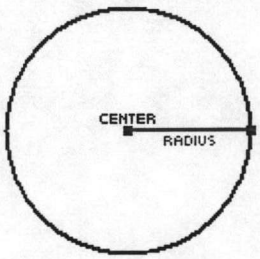
กำหนดจุดสมมาตรและจุดปลาย จากแถบเครื่องมือ แล้วคลิ๊กเมาส์ในตำแหน่งที่ต้องการสร้างจุดสมมาตร จากนั้นเลื่อนเมาส์ไปยังจุดที่ต้องการสร้างจุดปลายแล้วคลิ๊กเมาส์อีกครั้ง



รูปที่ ก.14 แสดงการสร้างเส้นตรงด้วยจุดสมมาตรและจุดปลาย

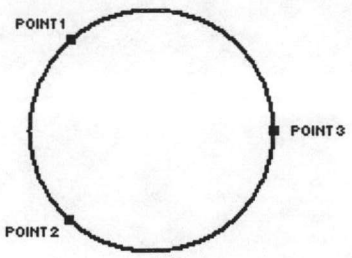
ก.4.2. <u>การสร้างวงกลม</u> เป็นการสร้างเซกเมนต์ของแบบร่างบนระนาบที่เป็นวงกลม มีวิธีการสร้าง 3 วิธีคือ

ก.4.2.1. การสร้างวงกลมด้วยจุดศูนย์กลางและรัศมี (Circle Center-Radius) เริ่มจากการเลือกคำสั่งสร้างวงกลมด้วยจุดศูนย์กลางและรัศมี จากแถบเครื่องมือ แล้วกำหนดจุดศูนย์กลางของวงกลมโดยคลิ๊กเมาส์ ณ ตำแหน่งที่ต้องการ จากนั้นคลิ๊กเมาส์อีกครั้งในตำแหน่งที่อยู่บนเส้นรอบวง



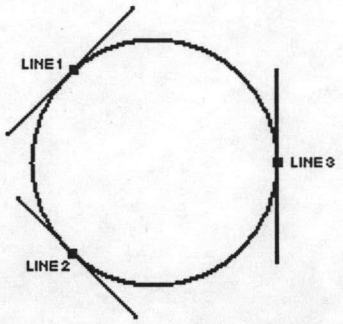
รูปที่ ก.15 แสดงการสร้างวงกลมด้วยจุดศูนย์กลางและรัศมี

ก.4.2.2. การสร้างวงกลมด้วยจุดบนเส้นรอบวง 3 จุด (Circle 3-Points) ในขั้นแรกเลือกคำสั่งสร้างวงกลมด้วยจุดบนเส้นรอบวง 3 จุด จากแถบเครื่องมือ แล้วกำหนดจุดบนเส้นรอบวง 3 จุดโดยใช้เมาส์คลิ๊กไปบนตำแหน่งที่ต้องการ



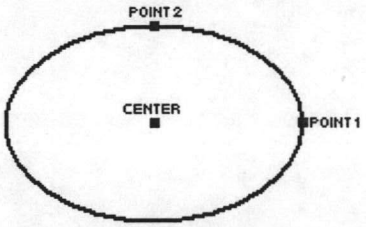
รูปที่ ก.16 แสดงการสร้างวงกลมด้วยจุดบนเส้นรอบวง 3 เส้น

ก.4.2.3. การสร้างวงกลมด้วยเส้นสัมผัสวงกลม 3 เส้น (Circle 3-Tangent) เริ่มจากการเลือกคำสั่งสร้างวงกลมด้วยเส้นสัมผัสวงกลม 3 เส้น จากแถบเครื่องมือ แล้วเลือกเส้นตรง 3 เส้นที่สัมผัสกับวงกลมที่ต้องการสร้าง โดยมีเงื่อนไขว่าเส้นตรงทั้งสามจะขนานกันได้ไม่เกิน 1 คู่ และคู่ที่ขนานกันนี้จะต้องไม่ซ้อนทับอยู่ในแนวเดียวกัน



รูปที่ ก.17 แสดงการสร้างวงกลมด้วยเส้นสัมผัส 3 เส้น

ก.4.3. <u>การสร้างวงรี</u> (Ellipse) เป็นการสร้างเซกเมนต์ของแบบร่างบนระนาบที่เป็นวงรีโดยการกำหนดจุดศูนย์กลางของวงรี จุดที่อยู่บนวงรีที่ตำแหน่งแกนเอก และจุดบนวงรีที่ตำแหน่งแกนโท ในขั้นแรกให้เลือกคำสั่งสร้างวงรี จากแถบเครื่องมือ แล้วกำหนดจุดศูนย์กลางของวงรี จุดที่อยู่บนวงรีที่ตำแหน่งแกนเอก และจุดบนวงรีที่ตำแหน่งแกนโท ตามลำดับโดยการคลิ๊กเมาส์ในตำแหน่งที่ต้องการ

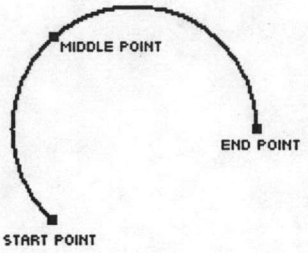


รูปที่ ก.18 แสดงการสร้างวงรี

ก.4.4. <u>การสร้างส่วนโค้งของวงกลม</u> เป็นการสร้างเซกเมนต์ของแบบร่างบนระนาบที่เป็นส่วนโค้งของวงกลม มีวิธีการสร้าง 3 วิธีคือ

ก.4.4.1. การสร้างส่วนโค้งของวงกลมด้วยจุดเริ่มต้น จุดระหว่างจุดเริ่มต้นกับจุดปลาย และจุดปลาย (Arc Start-Mid-End) การสร้างส่วนโค้งของวงกลมด้วยวิธีนี้

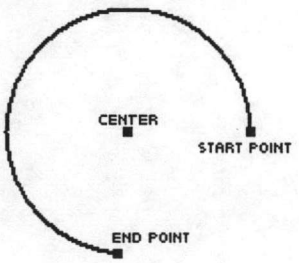
เริ่มจากการเลือกคำสั่งสร้างส่วนโค้งของวงกลมแบบจุดเริ่มต้น จุดระหว่างจุดเริ่มต้นกับจุดปลาย และจุดปลาย จากแถบเครื่องมือจากนั้นกำหนดจุดทั้ง 3 ตามลำดับโดยการคลิ๊กเมาส์ในตำแหน่งที่ต้องการ



รูปที่ ก.19 แสดงการสร้างส่วนโค้งของวงกลมด้วยจุดบนส่วนโค้ง 3 จุด

ก.4.4.2. การสร้างส่วนโค้งของวงกลมด้วยจุดเริ่มต้น จุดปลาย และจุดระหว่างจุดเริ่มต้นกับจุดปลาย (Arc Start-End-Mid) วิธีนี้จะเหมือนกับวิธีการสร้างส่วนโค้งของวงกลมในหัวข้อ 4.4.1 แต่ลำดับของจุดที่กำหนดจะต่างกัน โดยวิธีนี้จะกำหนดจุดเริ่มต้น จุดปลาย และจุดระหว่างจุดเริ่มต้นกับจุดปลายตามลำดับ ในการสร้างจะเรียกคำสั่งสร้างส่วนโค้งของวงกลมด้วยจุดเริ่มต้น จุดปลาย และจุดระหว่างจุดเริ่มต้นกับจุดปลาย จากแถบเครื่องมือ

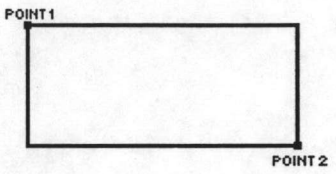
ก.4.4.3. การสร้างส่วนโค้งของวงกลมด้วยจุดเริ่มต้น จุดศูนย์กลางความโค้ง และจุดปลาย (Arc Start-Center-End) ในขั้นแรกเรียกคำสั่งสร้างส่วนโค้งของวงกลมด้วยจุดเริ่มต้น จุดศูนย์กลางความโค้ง และจุดปลาย จากแถบเครื่องมือ แล้วกำหนดจุดทั้ง 3 ตามลำดับโดยการใช้เมาส์คลิ๊กในตำแหน่งที่ต้องการ



รูปที่ ก.20 แสดงการสร้างส่วนโค้งของวงกลมด้วยจุดเริ่มต้น จุดศูนย์กลาง และจุดปลาย

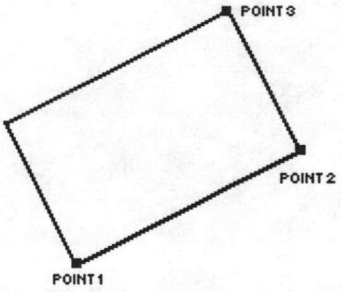
ก.4.5. <u>การสร้างสี่เหลี่ยม</u> เป็นการสร้างเซกเมนต์ของแบบร่างบนระนาบที่เป็นเส้นตรงสี่เส้นที่ต่อกันเป็นสี่เหลี่ยม มีวิธีการสร้าง 3 วิธีคือ

ก.4.5.1. การสร้างสี่เหลี่ยมผืนผ้าด้วยจุดสองจุด (Rectangle) วิธีนี้จะสร้างสี่เหลี่ยมผืนผ้าที่มีความยาวอยู่ในแนวนอน และความสูงอยู่ในแนวตั้ง โดยกำหนดจุดที่มุมของสี่เหลี่ยมผืนผ้า 2 จุดที่เป็นจุดของมุมที่อยู่ตรงกันข้าม สามารถเรียกคำสั่งสร้าง ได้จากแถบเครื่องมือ



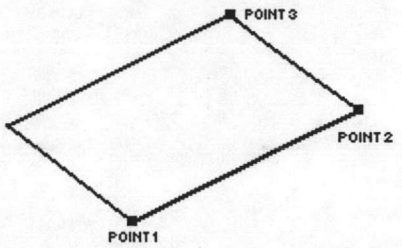
รูปที่ ก.21 แสดงการสร้างสี่เหลี่ยมผืนผ้าด้วยจุด 2 จุด

ก.4.5.2. การสร้างสี่เหลี่ยมผืนผ้าด้วยจุดสามจุด (Oriented Rectangle) วิธีนี้จะสร้างสี่เหลี่ยมผืนผ้าด้วยการกำหนดมุมของสี่เหลี่ยมผืนผ้าที่ต้องการสร้าง 3 มุม เริ่มจากการเรียกคำสั่งการสร้างสี่เหลี่ยมผืนผ้าด้วยจุดสามจุด จากแถบเครื่องมือจากนั้นกำหนดมุมทั้งสามของสี่เหลี่ยมผืนผ้าที่ต้องการสร้างโดยการคลิ๊กเมาส์ในตำแหน่งที่ต้องการ



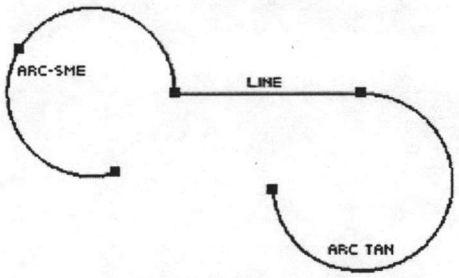
รูปที่ ก.22 แสดงการสร้างสี่เหลี่ยมผืนผ้าใด ๆ ด้วยจุด 3 จุด

ก.4.5.3. การสร้างสี่เหลี่ยมด้านขนานด้วยจุด 3 จุด (Parallelogram) เป็นการสร้างสี่เหลี่ยมด้านขนานโดยกำหนดมุมของสี่เหลี่ยมด้านขนาน 3 มุม ในการสร้างเริ่มต้นด้วยการเลือกคำสั่งสร้างสี่เหลี่ยมด้านขนานด้วยจุด 3 จุด จากแถบเครื่องมือ จากนั้นกำหนดมุมทั้งสามของสี่เหลี่ยมด้านขนานโดยการคลิ๊กเมาส์ในตำแหน่งที่ต้องการ



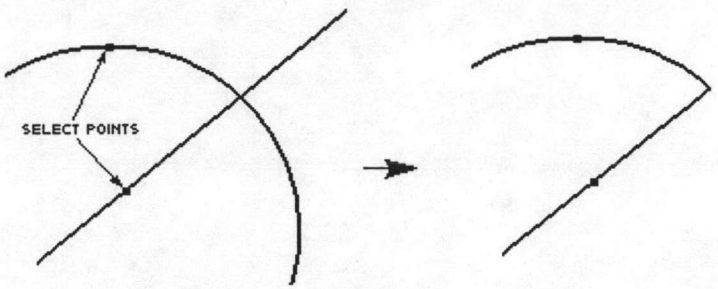
รูปที่ ก.23 แสดงการสร้างสี่เหลี่ยมด้านขนานด้วยจุด 3 จุด

ก.4.6. <u>การสร้างเส้นต่อเนื่อง</u> (Contour) เป็นการสร้างเซกเมนต์ของแบบร่างบนระนาบชนิดเส้นตรง และส่วนโค้งของเส้นตรงที่ต่อเนื่องกัน โดยเซกเมนต์ที่เป็นเส้นตรงจะสร้างด้วยวิธีการกำหนดจุดต้นและจุดปลาย ส่วนเซกเมนต์ที่เป็นส่วนโค้งของวงกลมสามารถเลือกสร้างได้ 2 วิธีคือ วิธีกำหนดจุดเริ่มต้น จุดระหว่างจุดเริ่มต้นกับจุดปลาย และจุดปลาย กับวิธีการสร้างส่วนโค้งของวงกลมให้สัมผัสกับเส้นต่อเนื่องที่สร้างก่อนหน้านี้ที่ตำแหน่งจุดเริ่มต้นของส่วนโค้งของวงกลม ในการใช้งาน เริ่มแรกเรียกคำสั่งสร้างเส้นต่อเนื่อง จากแถบเครื่องมือจากนั้นจะปรากฏปุ่มสร้างเส้นต่อเนื่องแบบต่าง ๆ บนแถบเครื่องมือ “Sketch Tools” เมื่อต้องการสร้างเส้นต่อเนื่องชนิดเส้นตรงให้กดปุ่ม (Add line) ถ้าต้องการสร้างเส้นต่อเนื่องชนิดส่วนโค้งของวงกลมด้วยวิธีกำหนดจุดเริ่มต้น จุดระหว่างจุดเริ่มต้นกับจุดปลาย และจุดปลาย ให้กดปุ่ม (Add arc) และถ้าต้องการสร้างเส้นต่อเนื่องชนิดส่วนโค้งของวงกลมที่สัมผัสกับเส้นต่อเนื่องก่อนหน้านี้ให้กดปุ่ม (Add tangent arc)



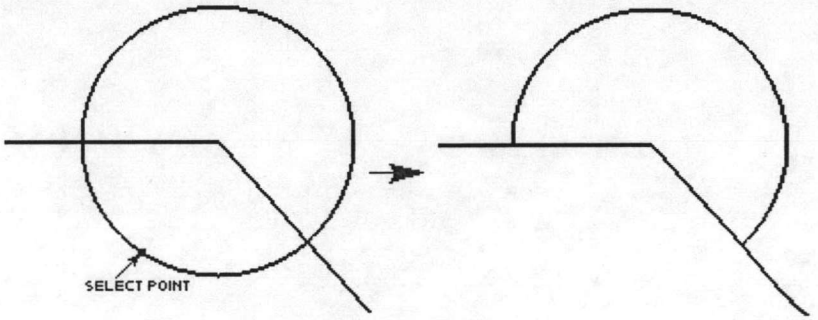
รูปที่ ก.24 แสดงการสร้างเส้นต่อเนื่องแบบต่าง ๆ

ก.4.7. <u>การทริมเส้นส่วนเกิน</u> (Trim) เป็นการลบเส้นส่วนเกินที่ไม่ต้องการที่เกิดจากการตัดกันของเซกเมนต์สองเส้นโดยเลือกคำสั่งทริม จากแถบเครื่องมือ จากนั้นเลือกเส้นส่วนที่ต้องการเก็บไว้บนเส้นที่ตัดกันสองเส้น



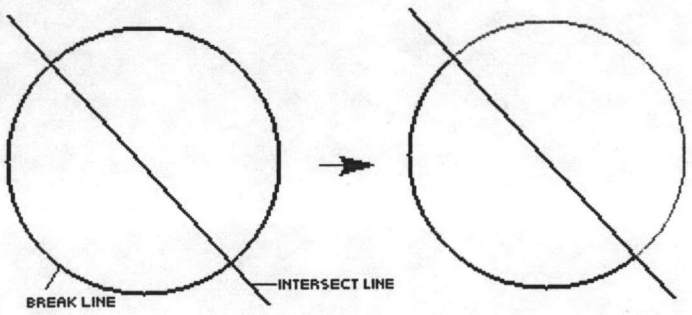
รูปที่ ก.25 แสดงการทริมเส้นส่วนเกิน

ก.4.8. <u>การลบส่วนของเส้นที่ไม่ต้องการ</u> (Erase) เป็นคำสั่งที่ใช้ลบส่วนของเส้นที่ไม่ต้องการโดยใช้เมาส์คลิ๊กไปบนส่วนของเส้นที่ไม่ต้องการนั้น การเรียกใช้คำสั่งนี้ ให้กดที่ปุ่ม จากแถบเครื่องมือ



รูปที่ ก.26 แสดงการลบส่วนของเส้นที่ไม่ต้องการ

ก.4.9. <u>การแบ่งเส้นออกเป็นส่วนย่อย ๆ ด้วยเส้นที่มาตัดกัน</u> (Break) เป็นการแบ่งเส้นบนระนาบออกเป็นเส้นย่อย ๆ ด้วยเส้นที่มาตัดกับเส้นที่จะถูกแบ่งนี้ เริ่มจากการเรียกคำสั่งแบ่งเส้น จากแถบเครื่องมือ จากนั้นเลือกเส้นที่ต้องการแบ่งออกเป็นเส้นย่อย ๆ โดยใช้เมาส์คลิ๊กบนเส้นที่ต้องการ แล้วเลือกเส้นที่มาตัดกับเส้นที่จะถูกแบ่ง



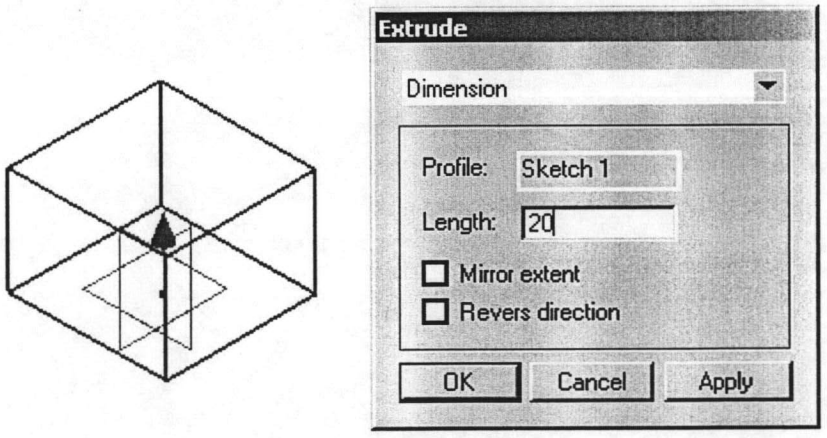
รูปที่ ก.27 แสดงการแบ่งเส้นออกเป็นส่วนย่อย ๆ ด้วยเส้นที่มาตัดกัน

ก.5. การสร้างแบบจำลองโซลิด

ก.5.1. การขึ้นรูปแบบจำลองโซลิด เป็นการสร้างแบบจำลองโซลิดจากแบบร่างบนระนาบ แบ่งออกเป็น 8 วิธีคือ

ก.5.1.1. การพอกด้วยการยืดในแนวเส้นตรง (Extrusion) เป็นการยืดแบบร่างบนระนาบไปตามแนวเส้นตรงที่ตั้งฉากกับระนาบของแบบร่างเป็นแบบจำลองโซลิด ซึ่งแบบร่างที่นำมาใช้จะต้องเป็นแบบร่างบนระนาบที่เป็นวงปิดเท่านั้น จากนั้นนำแบบจำลองที่ได้ไปกระทำบูลลีนแบบรวมกับลำตัวที่อยู่ในสถานะเอคทีฟ เริ่มจากเรียกคำสั่งการพอกด้วยการยืดในแนวเส้นตรง จากแถบเครื่องมือจะปรากฏกรอบโต้ตอบ การสร้างแบบจำลองโซลิดโดยการพอกด้วยการยืดในแนวเส้นตรงสามารถกำหนดขอบเขตของการยืดได้ดังนี้

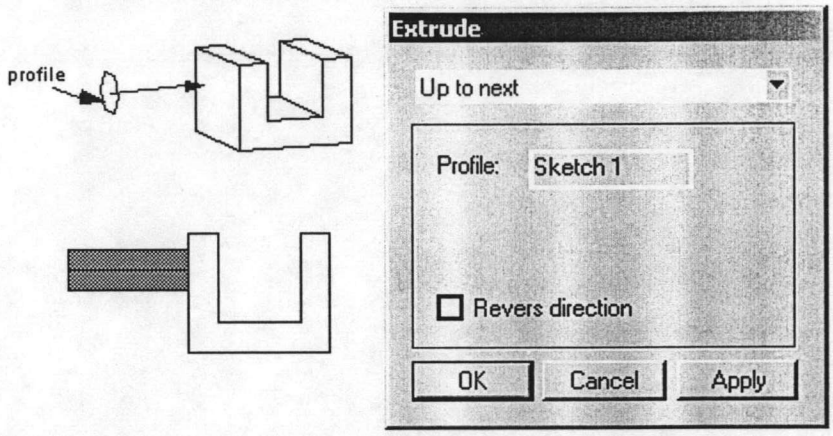
ก.5.1.1.1. *การกำหนดของเขตด้วยระยะการยืด (Dimension)* เมื่อกรอบโต้ตอบปรากฏให้เลือกกรอบคอมโบด้านบนเป็น “Dimension” จะได้กรอบโต้ตอบดังรูปที่ ก.28 เลือกแบบร่างบนระนาบที่ต้องการนำมาทำแบบจำลองโซลิด จากนั้นกำหนดค่าระยะการยืดในช่อง “Length” ซึ่งจะต้องมากกว่า 0 เสมอ ถ้าต้องการกลับทิศการยืดให้เลือก “Reverse direction” เป็นเครื่องหมายถูก หรือถ้าต้องการยืดออกทั้งสองด้านพร้อมกันก็ให้เลือก “Mirror extent” เป็นเครื่องหมายถูก เมื่อกำหนดค่าเสร็จให้กดตกลงเพื่อสิ้นสุดการทำงาน



รูปที่ ก.28 แสดงการยืดตามแนวเส้นตรงโดยการกำหนดขอบเขตด้วยระยะยืด

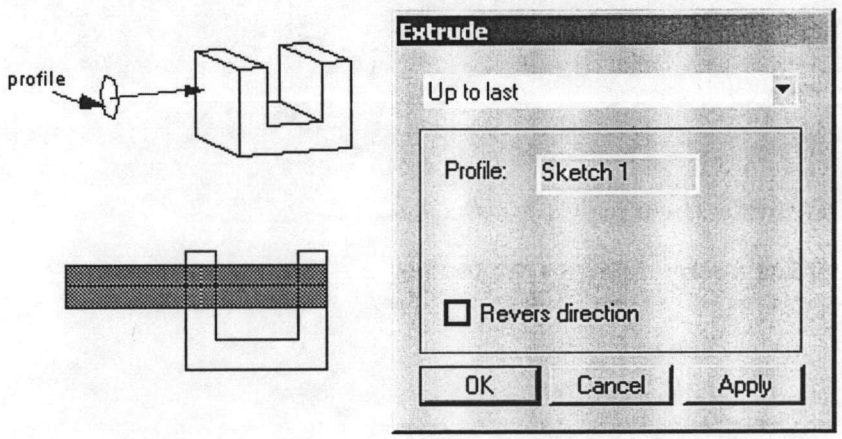
ก.5.1.1.2. *การกำหนดขอบเขตด้วยหน้าของแบบจำลองที่อยู่ถัดไป (Up to Next)* เมื่อกรอบโต้ตอบปรากฏขึ้นให้เปลี่ยนกรอบคอมโบด้านบนให้เป็น “Up to next” จะได้กรอบโต้ตอบดังรูปที่ ก.29 เลือกแบบร่างบนระนาบที่ต้องการนำมาทำการยืดตามแนวเส้นตรง ถ้า

ต้องการกลับทิศการยืดให้เลือก "Reverse direction" เป็นเครื่องหมายถูก เสร็จแล้วกดปุ่มตกลง เพื่อสร้างแบบจำลองโซลิดด้วยวิธีการนี้



รูปที่ ก.29 แสดงการยืดตามแนวเส้นตรงที่กำหนดของเขตด้วยหน้าของแบบจำลองที่อยู่ถัดไป

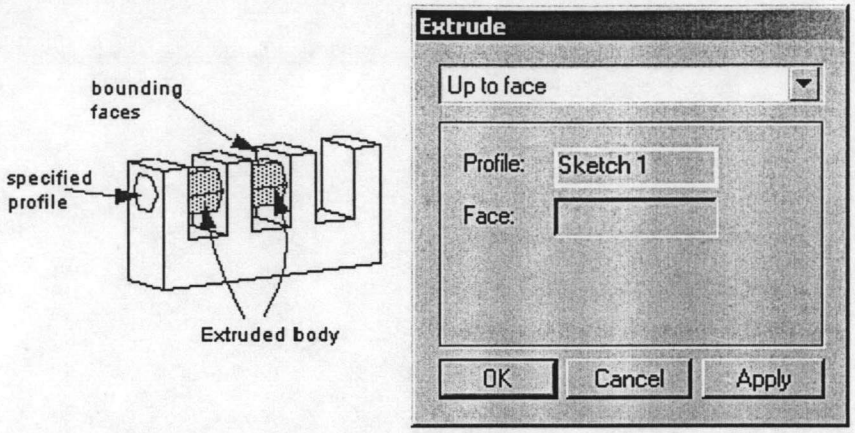
ก.5.1.1.3. *การกำหนดขอบเขตด้วยหน้าสุดท้ายของแบบจำลอง (Up to last)* เมื่อกรอบโต้ตอบปรากฏขึ้นให้เปลี่ยนกรอบคอมโบด้านบนเป็น "Up to last" จะได้กรอบโต้ตอบดังรูปที่ ก.30 เลือกแบบร่างบนระนาบที่ต้องการจะนำมายืดตามแนวเส้นตรง ถ้าต้องการกลับทิศการยืดให้เลือก "Reverse direction" เป็นเครื่องหมายถูก เสร็จแล้วกดปุ่มตกลงเพื่อสร้างแบบจำลองโซลิดด้วยวิธีการนี้



รูปที่ ก.30 แสดงการยืดในแนวเส้นตรงที่กำหนดขอบเขตด้วยหน้าสุดท้ายของแบบจำลองโซลิด

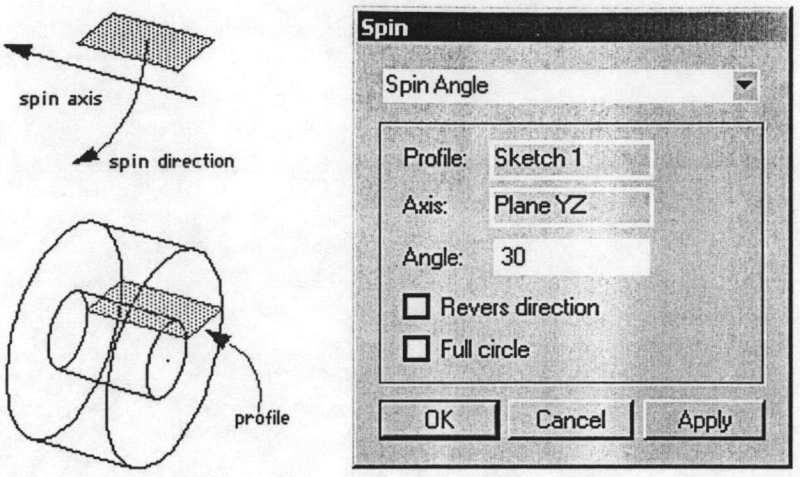
ก.5.1.1.4. *การกำหนดขอบเขตด้วยหน้าใด ๆ ของแบบจำลองที่ตัดกับแนวการยืด (Up to Face)* เมื่อกรอบโต้ตอบปรากฏขึ้นให้เปลี่ยนกรอบคอมโบด้านบนเป็น "Up to face" จะได้กรอบโต้ตอบดังรูปที่ ก.31 เลือกแบบร่างที่ต้องการนำมายืด จากนั้นคลิ๊กที่ช่อง

"Face" เพื่อเลือกหน้าที่ต้องการใช้เป็นขอบเขตการยืด ซึ่งหน้าที่ใช้จะต้องตัดผ่านแนวการยืดด้วย สุดท้ายกดปุ่มตกลงเพื่อสร้างแบบจำลองโซลิด



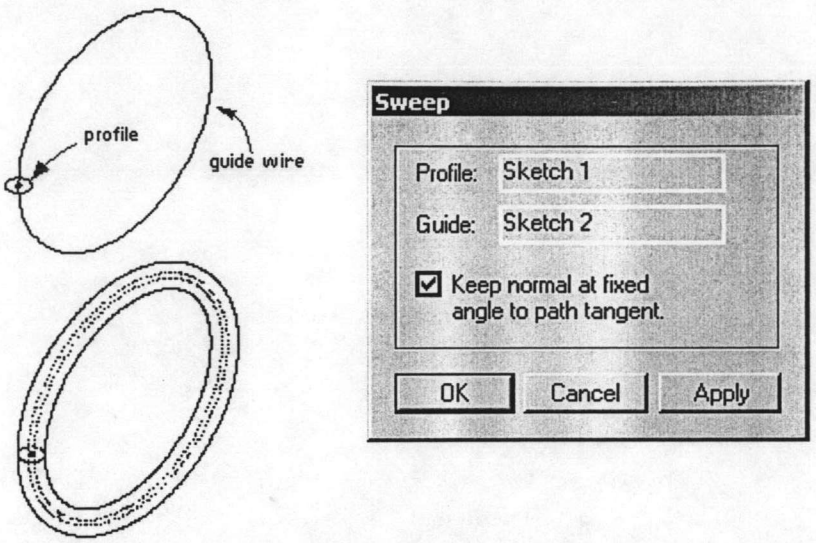
รูปที่ ก.31 แสดงการยืดในแนวเส้นตรงที่กำหนดของเขตด้วยหน้าใด ๆ ของแบบจำลองโซลิด

ก.5.1.2. การพอกด้วยการกวาดเชิงมุม (Revolution) เป็นการสร้างแบบจำลองโซลิดด้วยการกวาดแบบร่างบนระนาบที่เป็นวงปิด รอบแกนหมุนเป็นค่ามุมต่าง ๆ แล้วนำแบบจำลองที่ได้ไปกระทำบูลลีนแบบรวมกับแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ ในการใช้งานจะเริ่มด้วยการเรียกคำสั่งสร้างแบบจำลองโซลิดด้วยการกวาดเชิงมุม จากแถบเครื่องมือ จากนั้นจะปรากฏกรอบโต้ตอบเพื่อกรอกพารามิเตอร์ในการสร้างดังรูปที่ ก.32 คลิ๊กที่ช่อง "Profile" เพื่อเลือกแบบร่างบนระนาบที่ต้องการนำมาใช้ เมื่อเลือกระนาบได้แล้วให้คลิ๊กที่ช่อง "Axis" เพื่อเลือกแกนหมุน โดยแกนหมุนที่เลือกอาจเป็นขอบของแบบจำลองโซลิดที่เป็นเส้นตรง หรือเป็นระนาบอ้างอิงก็ได้ ถ้าเป็นระนาบอ้างอิงจะใช้แกนที่ตั้งฉากกับระนาบอ้างอิงนั้นที่จุดศูนย์กลางระนาบเป็นแกนหมุน และทิศทางการกวาดเชิงมุมจะเป็นไปตามกฎมือขวา หลังจากเลือกแกนหมุนแล้ว กำหนดค่ามุมการกวาดในช่อง "Angle" ถ้าต้องการกลับทิศการกวาดให้เลือกช่อง "Reverse direction" เป็นเครื่องหมายถูก หรือถ้าต้องการให้กวาดครบรอบเป็นวงกลมก็สามารถทำโดยเลือกช่อง "Full circle" ให้เป็นเครื่องหมายถูก



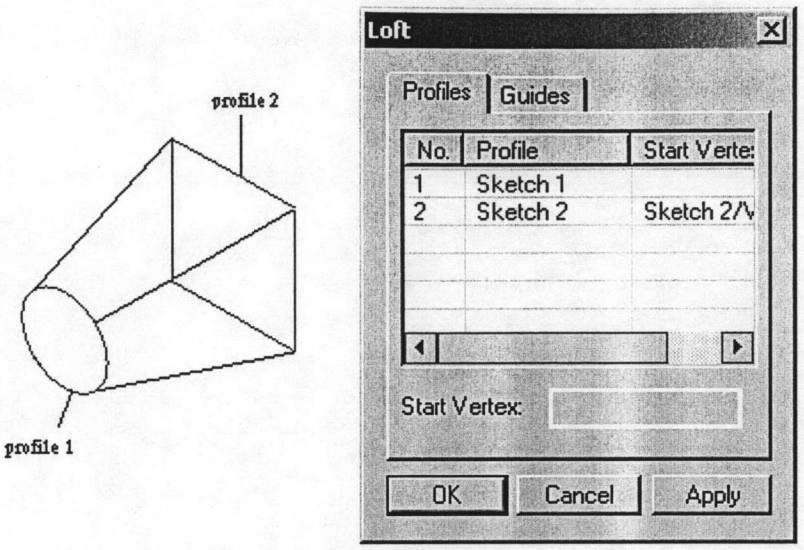
รูปที่ ก.32 แสดงการกวาดเชิงมุม

ก.5.1.3. การพอกด้วยการยึดตามเส้นนำ (Sweep along guide) เป็นการสร้างแบบจำลองโซลิดโดยการยึดแบบร่างบนระนาบที่เป็นวงปิด ไปตามเส้นนำซึ่งเป็นแบบร่างบนระนาบเหมือนกันแต่อาจเป็นวงปิด หรือเปิดก็ได้แล้วนำแบบจำลองที่ได้มากระทำบูลลีนแบบรวมกับแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ ในการสร้างแบบจำลองโซลิดด้วยวิธีนี้เริ่มด้วยการเลือกคำสั่งสร้างแบบจำลองโซลิดด้วยการพอกโดยการยึดตามเส้นนำ จากแถบเครื่องมือ จากนั้นจะปรากฏกรอบโต้ตอบเพื่อกำหนดพารามิเตอร์ในการสร้างดังรูปที่ ก.33 คลิ๊กที่ช่อง "Profile" แล้วเลือกแบบร่างบนระนาบที่เป็นวงปิดที่ต้องการนำมาทำการพอกด้วยการยึดตามเส้นนำ เมื่อเลือกได้แล้วให้คลิ๊กที่ช่อง "Guide" เพื่อกำหนดเส้นนำโดยการเลือกจากแบบร่างบนระนาบเช่นกัน ในการยึดตามเส้นนำสามารถกำหนดมุมของระนาบของแบบร่างที่นำมายึดให้คงที่ตลอดการยึดเทียบกับเส้นนำ หรือให้เทียบกันระบบแกนอ้างอิงรวมก็ได้ โดยถ้าต้องการให้คงที่เทียบกับเส้นนำให้เลือกช่อง "Keep normal at fixed angle to path tangent" เป็นเครื่องหมายถูก แต่ถ้าต้องการรักษาระนาบของแบบร่างให้คงที่เทียบกับระบบแกนอ้างอิงรวมก็ให้เลือกเป็นช่องว่างแทน



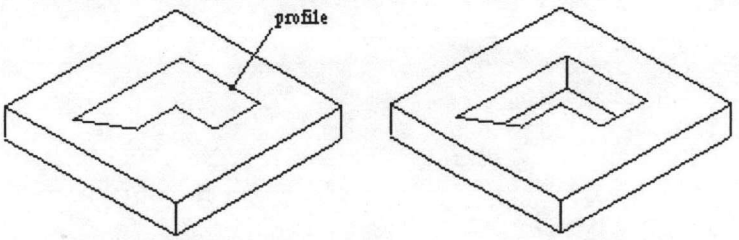
รูปที่ ก.33 แสดงการกวาดตามเส้นนำ

ก.5.1.4. การพอกด้วยลอฟท์ (Loft) เป็นการสร้างแบบจำลองโซลิดโดยการกำหนดหน้าตัดที่ตำแหน่งต่าง ๆ ของแบบจำลองโซลิดด้วยกลุ่มของแบบร่างบนระนาบที่เป็นวงปิดแล้วนำแบบจำลองที่ได้มากระทำบูลลีนแบบรวมกับแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ ในการสร้างขั้นแรกให้เลือกคำสั่งสร้างแบบจำลองโซลิดจากการพอกด้วยลอฟท์ ที่อยู่บนแถบเครื่องมือ จากนั้นจะปรากฏกรอบโต้ตอบดังรูป ก.34 จากนั้นให้เลือกแบบร่างบนระนาบที่เป็นวงปิดที่จะใช้เป็นหน้าตัดของแบบจำลองโซลิดที่ต้องการสร้างเรียงตามลำดับก่อนหลังให้ถูกต้อง เมื่อเลือกแบบร่างบนระนาบแต่ละอันแล้วชื่อของแบบร่างบนระนาบที่เลือกจะปรากฏอยู่ในกรอบโต้ตอบ สำหรับแบบร่างบนระนาบที่มีจุดยอด (ไม่เป็นวงกลม หรือวงรี) ต้องมีการกำหนดจุดอ้างอิงระว่างหน้าตัดด้วย โดยคลิ๊กที่ชื่อของแบบร่างบนระนาบที่ต้องการกำหนดจุดอ้างอิง จากนั้นคลิ๊กที่ช่อง “Start vertex” แล้วเลือกจุดยอดที่ต้องการจากระนาบอ้างอิงที่เลือก เมื่อกำหนดค่าพารามิเตอร์ต่าง ๆ ครบถ้วนแล้วกดปุ่มตกลง (OK) เพื่อสิ้นสุดการทำงาน



รูปที่ ก.34 แสดงการลอฟท์

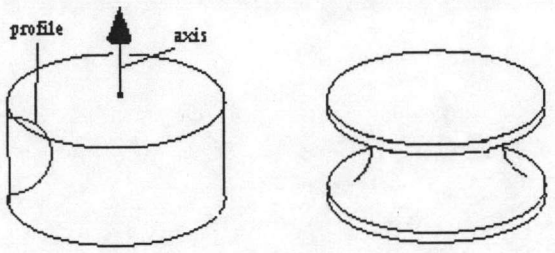
ก.5.1.5. การตัดด้วยการยืดในแนวเส้นตรง (Extrusion Cut) เป็นการสร้างแบบจำลองโซลิดโดยยืดแบบร่างบนระนาบที่เป็นวงปิดในแนวเส้นตรงที่ตั้งฉากกับระนาบของแบบร่างที่นำมายืด จากนั้นนำแบบจำลองที่ได้ไปกระทำบูลลีนแบบลบกับแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ โดยเริ่มต้นจากเรียกคำสั่งการยืดในแนวเส้นตรง จากแถบเครื่องมือจะปรากฏกรอบโต้ตอบสำหรับกำหนดพารามิเตอร์ในการสร้าง ซึ่งมีขั้นตอนในการกำหนดค่าพารามิเตอร์เหมือนกับการพอกด้วยการยืดในแนวเส้นตรง



รูปที่ ก.35 แสดงการตัดด้วยการยืดในแนวเส้นตรง

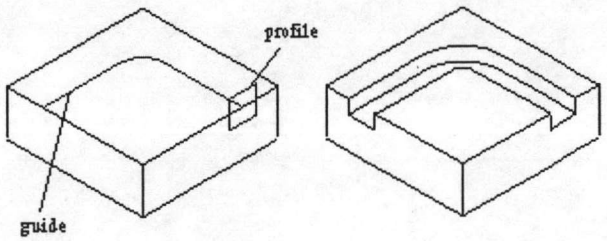
ก.5.1.6. การตัดด้วยการกวาดเชิงมุม (Revolution Cut) เป็นการสร้างแบบจำลองโซลิดโดยการกวาดแบบร่างบนระนาบที่เป็นวงปิดรอบแกนหมุนด้วยมุมการหมุนใด ๆ แล้วนำแบบจำลองที่เกิดขึ้นมากระทำบูลลีนแบบลบกับแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ ในการสร้างแบบจำลองโซลิดด้วยวิธีนี้เริ่มจากการเรียกคำสั่งการตัดด้วยการกวาดเชิงมุม จาก

แถบเครื่องมือแล้วจะปรากฏกรอบโต้ตอบเพื่อกำหนดพารามิเตอร์ในการสร้าง ซึ่งมีขั้นตอนการกำหนดพารามิเตอร์เหมือนกับการพอกด้วยการกวาดเชิงมุม



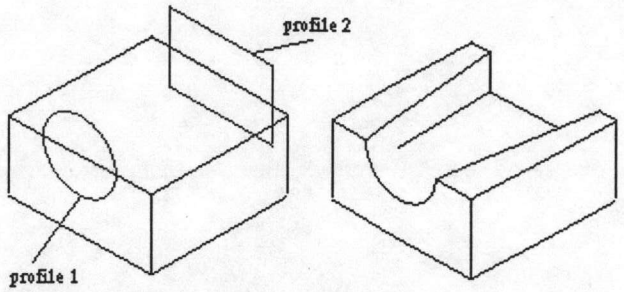
รูปที่ ก.36 แสดงการตัดด้วยการกวาดเชิงมุม

ก.5.1.7. การตัดด้วยการยืดตามเส้นนำ (Swept Cut) เป็นการสร้างแบบจำลองโซลิดโดยการยืดแบบร่างบนระนาบที่เป็นวงปิดไปตามแนวของเส้นนำแล้วนำแบบจำลองโซลิดที่เกิดขึ้นมากระทำบูลลีนแบบลบกับแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ ในการสร้างทำโดยเลือกคำสั่งการตัดด้วยการยืดตามเส้นนำ จากแถบเครื่องมือ จากนั้นจะปรากฏกรอบโต้ตอบเพื่อกำหนดพารามิเตอร์ในการสร้าง ซึ่งมีขั้นตอนเหมือนกับการพอกด้วยการยืดตามเส้นนำ



รูปที่ ก.37 แสดงการตัดด้วยการยืดตามเส้นนำ

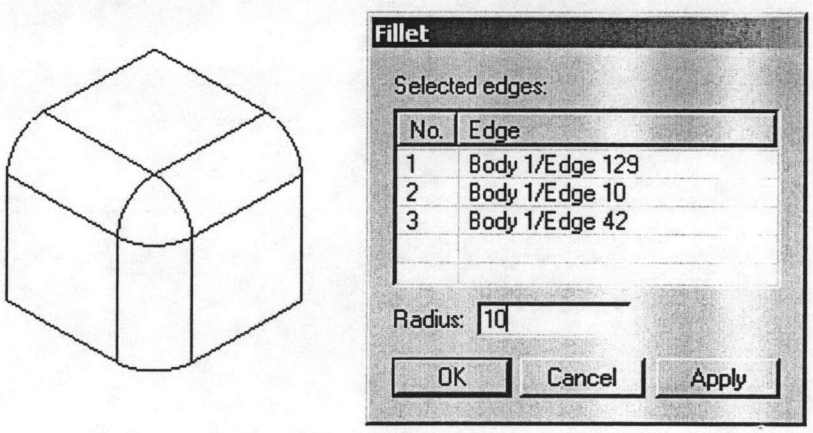
ก.5.1.8. การตัดด้วยลอฟท์ (Loft Cut) เป็นการสร้างแบบจำลองโซลิดด้วยการกำหนดหน้าตัดของโซลิดที่จะสร้างด้วยแบบร่างบนระนาบที่เป็นวงปิดจากนั้นนำแบบจำลองโซลิดที่ได้มากระทำบูลลีนแบบลบกับแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ เริ่มจากการเลือกคำสั่งการตัดด้วยลอฟท์ จากแถบเครื่องมือจากนั้นจะปรากฏกรอบโต้ตอบเพื่อกำหนดพารามิเตอร์ซึ่งมีขั้นตอนเหมือนกับการพอกด้วยลอฟท์



รูปที่ ก.38 แสดงการตัดด้วยลอฟท์

ก.5.2. <u>การตกแต่งแบบจำลองโซลิด</u> (Dress up) เป็นการปรับแต่งแบบจำลองโซลิดที่สร้างขึ้นด้วยกระบวนการขึ้นรูปในหัวข้อ 5.1 ซียู-โซลิดมีคำสั่งที่ใช้ตกแต่งแบบจำลองโซลิด 3 คำสั่งด้วยกันคือ

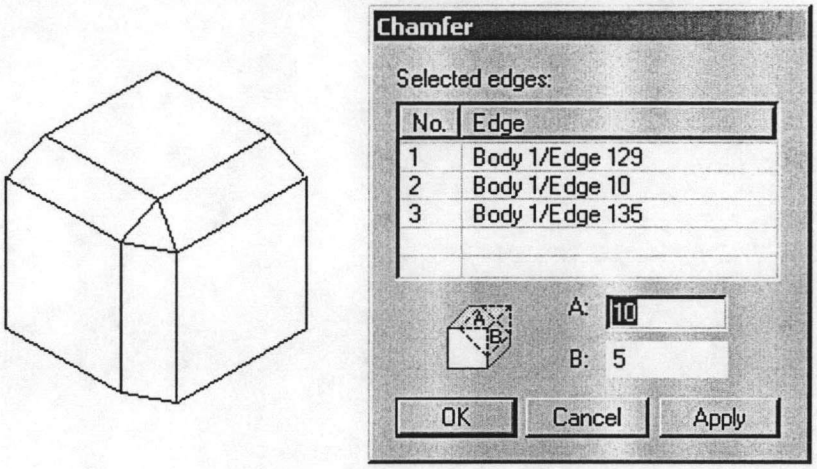
ก.5.2.1. การทำฟิลเลต (Fillet) เป็นการลบขอบของแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟด้วยผิวโค้งของทรงกระบอก ในการทำฟิลเลตเริ่มจากเรียกคำสั่งการทำฟิลเลต จากแถบเครื่องมือ จากนั้นจะปรากฏกรอบโต้ตอบเพื่อกำหนดขอบที่ต้องการลบและรัศมีของผิวทรงกระบอกที่จะใช้ทำฟิลเลตดังรูปที่ ก.39 เลือกขอบที่ต้องการทำฟิลเลตโดยใช้เมาส์คลิ๊กลงบนขอบที่ต้องการ หรืออาจเลือกเป็นหน้าของแบบจำลองโซลิดก็ได้ ซึ่งถ้าเลือกเป็นหน้าขอบทั้งหมดที่ปิดล้อมหน้าที่เลือกจะถูกทำฟิลเลต สุดท้ายกำหนดค่ารัศมีการทำฟิลเลตที่ต้องการในช่อง “Radius” แล้วกดปุ่มตกลง



รูปที่ ก.39 แสดงการทำฟิลเลต

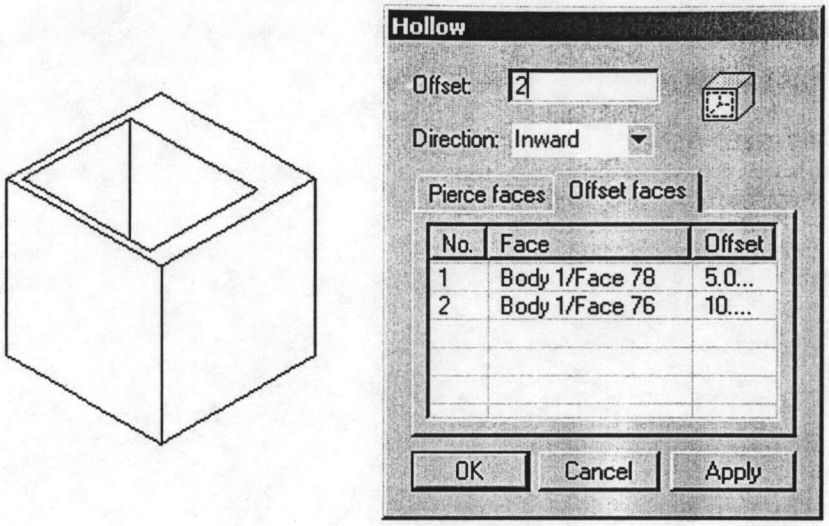
ก.5.2.2. การทำแชมเฟอร์ (Chamfer) เป็นการลบขอบของแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟด้วยผิวของระนาบเอียง ในการทำแชมเฟอร์เริ่มจากเรียกคำสั่งการทำ

แชมเฟอร์ จากแถบเครื่องมือ แล้วกรอบโต้ตอบจะปรากฏขึ้นเพื่อให้กำหนดพารามิเตอร์ที่ใช้ทำแชมเฟอร์ดังรูปที่ ก.40 เลือกขอบที่ต้องการทำแชมเฟอร์โดยใช้เมาส์คลิ๊กลงบนขอบหรือหน้าที่ต้องการ ซึ่งถ้าเลือกเป็นหน้า ขอบทั้งหมดที่ปิดล้อมหน้าที่เลือกจะถูกทำแชมเฟอร์ สุดท้ายกำหนดค่าลาดเอียงทั้งสองด้านของขอบที่เลือกในช่อง "A" และ "B" แล้วกดปุ่มตกลง



รูปที่ ก.40 แสดงการทำแชมเฟอร์

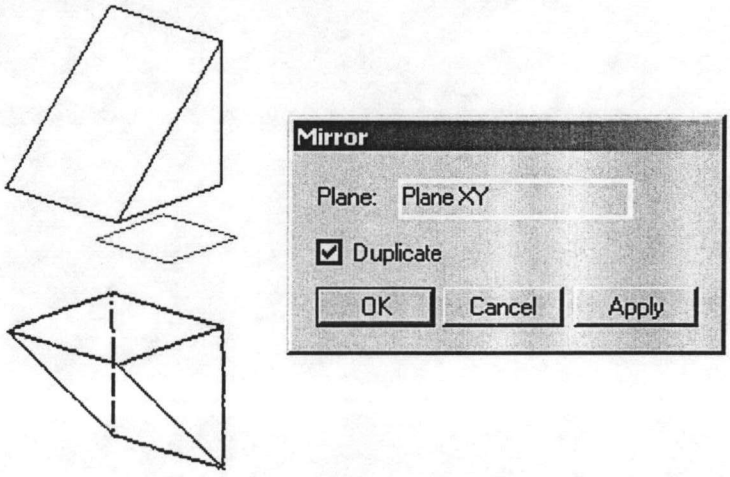
ก.5.2.3. การทำโซลิดกลวง (Hollow) เป็นตัดเนื้อโซลิดภายในแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟออกกลายเป็นโซลิดกลวง เริ่มจากการเลือกคำสั่งการทำโซลิดกลวง จากแถบเครื่องมือจะปรากฏกรอบโต้ตอบดังรูปที่ ก.41 เลือกแท็บ "Pierce faces" เพื่อกำหนดหน้าที่ต้องการเปิดออกให้บริเวณที่ว่างภายในโซลิดกลวงเชื่อมต่อกับที่ว่างภายนอก แล้วเลือกหน้าที่ต้องการด้วยเมาส์ จากนั้นกำหนดค่าความหนาของโซลิดกลวงที่จะสร้างในช่อง "Offset" ในกรณีที่แต่ละหน้าของโซลิดกลวงมีความหนาไม่เท่ากันสามารถกำหนดได้โดยเลือกแท็บ "Offset faces" แล้วเลือกหน้าที่ต้องการกำหนดความหนาทีละหน้าพร้อมกับกำหนดค่าความหนาที่ต้องการ เมื่อกำหนดพารามิเตอร์ที่ต้องการครบแล้วให้กดปุ่มตกลง



รูปที่ ก.41 แสดงการทำโซลิดกลวง

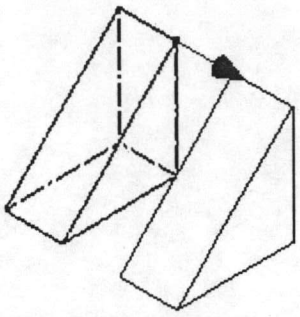
ก.5.3. <u>การปรับตำแหน่งการวางตัวของแบบจำลองโซลิด</u> (Transform) เป็นการเปลี่ยนตำแหน่งของแบบจำลองโซลิดเทียบกับระบบแกนอ้างอิงรวม (World Coordinate) ซึ่งมีวิธีการปรับตำแหน่ง 3 วิธีคือ

ก.5.3.1. การทำมิเรอร์ (Mirror) เป็นการทำแบบจำลองโซลิดที่เป็นภาพสะท้อนจากระนาบมิเรอร์ของแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟ ในการทำมิเรอร์เริ่มจากการเรียกคำสั่งการทำมิเรอร์ จากแถบเครื่องมือจะปรากฏกรอบโต้ตอบดังรูปที่ ก.42 จากนั้นเลือกระนาบที่จะใช้ทำมิเรอร์โดยการเลือกจากระนาบอ้างอิงหรือหน้าของแบบจำลองโซลิดที่เป็นระนาบก็ได้ แล้วกดปุ่มตกลง



รูปที่ ก.42 แสดงการทำมิเรอร์

ก.5.3.2. การเลื่อนตำแหน่งในแนวเส้นตรง (Translate) เป็นการเปลี่ยนตำแหน่งของแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟในแนวเส้นตรงเทียบกับระบบแกนอ้างอิงรวม ซึ่งเรียกคำสั่งการเลื่อนตำแหน่งในแนวเส้นตรง จากแถบเครื่องมือแล้วจะปรากฏกรอบโต้ตอบสำหรับกำหนดพารามิเตอร์ ในขั้นตอนนี้สามารถกำหนดพารามิเตอร์ได้ 3 วิธีคือ



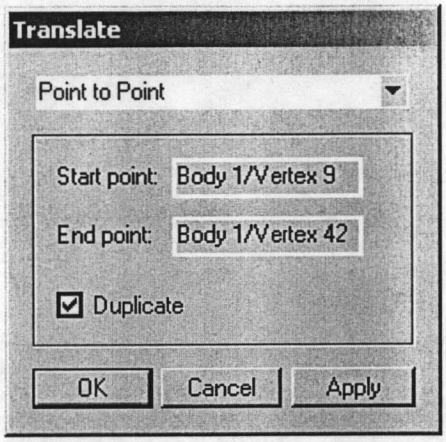
รูปที่ ก.43 แสดงการเลื่อนตำแหน่งเชิงเส้น

ก.5.3.2.1. *ทิศทางและขนาดการกระจัด (Direction and Displacement)* ในตอนบนของกรอบโต้ตอบให้เลือกกรอบคอมโบเป็น "Direction and Displacement" จะได้กรอบโต้ตอบดังรูปที่ ก.44 จากนั้นคลิ๊กในช่อง "Direction" เพื่อกำหนดทิศทางของการเลื่อนตำแหน่งโดยเลือกจากของที่เป็นเส้นตรง หรือจากระนาบอ้างอิงก็ได้ โดยถ้าเลือกจากระนาบอ้างอิงทิศทางของการเลื่อนตำแหน่งจะถูกกำหนดด้วยเวกเตอร์ตั้งฉาก (Normal vector) ของระนาบอ้างอิง สุดท้ายกำหนดระยะการเลื่อนตำแหน่งที่ต้องการ แล้วกดปุ่มตกลง

รูปที่ ก.44 แสดงกรอบโต้ตอบของการเลื่อนตำแหน่งในแนวเส้นตรงด้วยการกระจัด

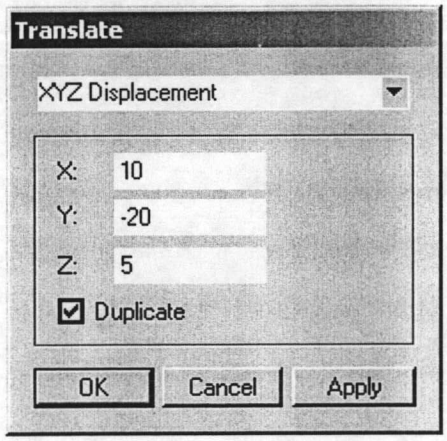
ก.5.3.2.2. *จุดอ้างอิง 2 จุด (Point to Point)* ในกรอบคอมโบด้านบนของกรอบโต้ตอบเลือกเป็น "Point to Point" จะได้กรอบโต้ตอบดังรูปที่ ก.45 คลิ๊กที่ช่อง "Start

point" แล้วเลือกจุดอ้างอิงแรก คลิ๊กที่ช่อง "End point" แล้วเลือกจุดอ้างอิงอีกจุดหนึ่ง แล้วกดปุ่มตกลง ในการเลื่อนตำแหน่งด้วยวิธีนี้ แบบจำลองโซลิดจะเลื่อนไปตามเวกเตอร์ที่เกิดจากจุดปลายลบจุดเริ่มต้น



รูปที่ ก.45 แสดงกรอบโต้ตอบสำหรับการเลื่อนตำแหน่งด้วยจุดอ้างอิง 2 จุด

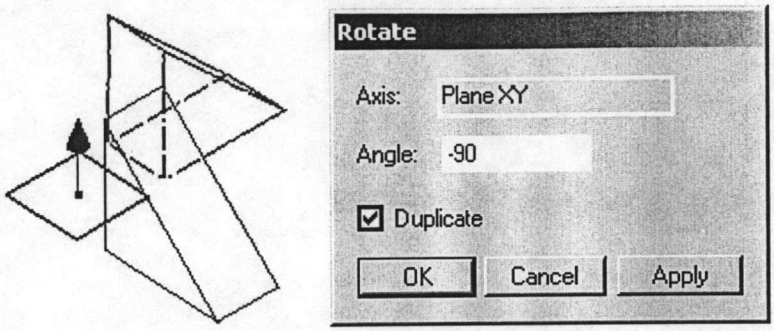
ก.5.3.2.3. *ระยะการกระจัดในแนวแกนอ้างอิง XYZ (XYZ Displacement)* ในกรอบคอมโบด้านบนเลือกเป็น "XYZ Displacement" จะได้กรอบโต้ตอบดังรูปที่ ก.46 กำหนดค่าการกระจัดในแนวแกน X Y และ Z ในช่อง "X" "Y" และ "Z" ตามลำดับ แล้วกดปุ่มตกลง



รูปที่ ก.46 แสดงกรอบโต้ตอบสำหรับการเลื่อนตำแหน่งด้วยระยะการกระจัดในแนว XYZ

ก.5.3.3. การเลื่อนตำแหน่งเชิงมุม (Rotate) เป็นการหมุนแบบจำลองโซลิดที่อยู่ในสภาวะแอคทีฟรอบแกนหมุนด้วยมุมใด ๆ เรียกคำสั่งการเลื่อนตำแหน่งเชิงมุม จาก

แถบเครื่องมือ จะปรากฏกรอบโต้ตอบดังรูปที่ ก.47 กำหนดแกนหมุนโดยเลือกจากขอบที่เป็นเส้นตรงของแบบจำลองโซลิด หรือจากระนาบอ้างอิงก็ได้ ถ้าเลือกแกนหมุนด้วยระนาบอ้างอิง แบบจำลองโซลิดจะหมุนรอบแกนตั้งฉาก (Normal vector) ของระนาบอ้างอิงที่จุดศูนย์กลางระนาบ แล้วกำหนดมุมการหมุน กดปุ่มตกลงเพื่อจบการทำงาน

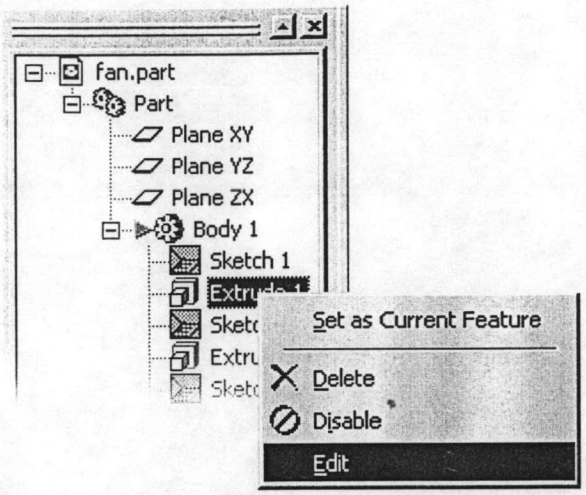


รูปที่ ก.47 แสดงการเลื่อนตำแหน่งเชิงมุม

ในการปรับตำแหน่งการวางตัวของแบบจำลองโซลิดด้วยวิธีต่าง ๆ ที่ได้กล่าวมานี้สามารถทำสำเนาแบบจำลองโซลิดที่เป็นต้นแบบเอาไว้ก่อนได้โดยเลือกช่อง "Duplicate" ให้เป็นเครื่องหมายถูก แบบจำลองโซลิดที่ได้จะเป็นการกระทำบูลลีนแบบรวมระหว่างแบบจำลองโซลิดต้นแบบกับแบบจำลองโซลิดที่ปรับตำแหน่งแล้ว

ก.6. การแก้ไขแบบจำลองโซลิด

ก.6.1. <u>การแก้ไขพารามิเตอร์ในการสร้างแบบจำลองโซลิด</u> เมื่อต้องการแก้ไขแบบจำลองโซลิดที่สร้างสามารถทำได้โดยการคลิ๊กเมาส์ขวาบนลักษณะจำเพาะที่ต้องการแก้ไขบนทรีของลักษณะจำเพาะ จากนั้นเลือกคำสั่งแก้ไข (Edit) จะปรากฏกรอบโต้ตอบที่ใช้สร้างแบบจำลองโซลิดด้วยวิธีที่เลือก จากนั้นแก้ไขพารามิเตอร์ตามต้องการ แล้วกดปุ่มตกลง (OK) เพื่อสิ้นสุดการแก้ไข ในกรณีเมื่อต้องการแก้ไขแบบร่างบนระนาบก็สามารถทำได้ด้วยวิธีเดียวกัน เมื่อเลือกคำสั่ง "Edit" ซียู-โซลิดจะเข้าสู่โหมดการสร้างแบบร่างบนระนาบ เมื่อแก้ไขแบบร่างเสร็จก็กดปุ่มออกจากโหมดการสร้างแบบร่างบนระนาบ เมื่อเสร็จสิ้นการแก้ไขซียู-โซลิด



รูปที่ ก.48 แสดงเมนูป๊อบอัพสำหรับการแก้ไขแบบจำลองโซลิด

ก.6.2. <u>การลบลักษณะจำเพาะ</u> สามารถทำได้โดยคลิ๊กเมาส์ขวาบนลักษณะจำเพาะที่ต้องการลบในทรีของลักษณะจำเพาะจะปรากฏเมนูป๊อบอัพ เลือกคำสั่งลบ (Delete) ลักษณะจำเพาะที่เลือกจะถูกลบออกไปพร้อมกับที่ซียู-โซลิดจะปรับปรุงแบบจำลองโซลิดใหม่

ก.6.3. <u>การซ่อนการทำงานของลักษณะจำเพาะ</u> คล้ายกับเป็นการลบลักษณะจำเพาะที่สร้างขึ้นชั่วคราวโดยการคลิ๊กเมาส์ขวาบนลักษณะจำเพาะที่ต้องการซ่อนในทรีของลักษณะจำเพาะจะปรากฏเมนูป๊อบอัพ เลือกคำสั่งซ่อนการทำงาน (Disable) ลักษณะจำเพาะที่ถูกซ่อนสามารถเรียกกลับมาให้ทำงานใหม่ได้โดยทำตามขั้นตอนเหมือนกับการซ่อนการทำงาน

ก.6.4. <u>การเปลี่ยนลำดับการทำงานของลักษณะจำเพาะ</u> ทำโดยใช้เมาส์ลากลักษณะจำเพาะที่ต้องการย้ายตำแหน่งไปปล่อยในตำแหน่งใหม่ที่ต้องการ

ก.6.5. <u>การแทรกลักษณะจำเพาะ</u> เป็นการสร้างลักษณะจำเพาะแทรกระหว่างลักษณะจำเพาะที่สร้างขึ้นมาก่อนหน้านี้ สามารถทำโดยในขั้นแรกคลิ๊กขวาที่ลักษณะจำเพาะลำดับก่อนลักษณะจำเพาะที่ต้องการแทรกจะปรากฏเมนูป๊อบอัพให้เลือก “Set as Current Feature” เพื่อกำหนดให้เป็นลักษณะจำเพาะสุดท้ายที่ทำงาน แบบจำลองโซลิดจะกลับไปยังสภาวะที่ลักษณะจำเพาะที่เลือกทำงานเป็นลำดับสุดท้าย จากนั้นสร้างแบบจำลองโซลิดตามต้องการเมื่อเสร็จแล้วก็ให้คลิ๊กขวาที่ลักษณะจำเพาะลำดับสุดท้ายในทรีแสดงลักษณะจำเพาะของลำตัวที่กำลังทำงานอยู่ แล้วเรียกคำสั่ง “Set as Current Feature” ด้วยวิธีเดียวกันกับในข้างต้น

ภาคผนวก ข.

ภาคผนวก ข.

วิธีสร้างซอฟแวร์ส่วนประกอบเสริมสำหรับซียู-โซลิด

ข.1. การติดตั้งตัวช่วยสร้างซอฟแวร์ส่วนประกอบเสริมสำหรับซียู-โซลิด

ข.1.1. ความต้องการของระบบในการติดตั้ง ก่อนการติดตั้งตัวช่วยสร้างซอฟแวร์ส่วนประกอบเสริมควรเตรียมคอมพิวเตอร์ที่จะติดตั้งให้พร้อมใช้งานเสียก่อน มิฉะนั้นตัวช่วยสร้างซอฟแวร์ส่วนประกอบเสริมอาจไม่สามารถทำงานได้ สิ่งระบบต้องการก่อนการติดตั้งมีดังนี้

ข.1.1.1. ซียู-โซลิด เวอร์ชัน 1.0

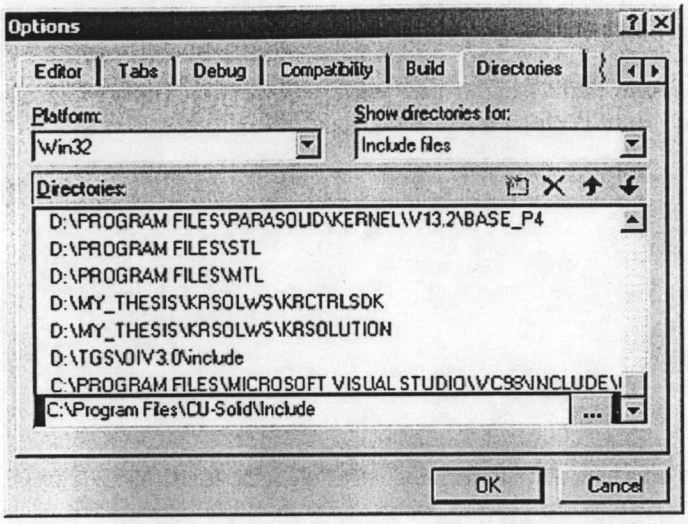
ข.1.1.2. ไมโครซอฟวิชชวลซีพลัสพลัสเวอร์ชัน 6.0 (Microsoft Visual C++ 6.0) หรือสูงกว่า ต้องรองรับยูนิโค้ด (Unicode) ด้วย

ข.1.1.3. ระบบปฏิบัติการวินโดว์ 2000

ข.1.2. ขั้นตอนการติดตั้ง โดยปกติแล้วตัวช่วยสร้างซอฟแวร์ส่วนประกอบเสริมนี้จะถูกติดตั้งพร้อมกับการติดตั้งซียู-โซลิด แต่ยังไม่สามารถใช้งานได้ ต้องมีการปรับแต่งเสียก่อนดังนี้

ข.1.2.1. ทำสำเนาไฟล์ “CUSOLID.awx” จากไดเรกทอรีย่อยที่ชื่อ “appwizard” ซึ่งอยู่ในไดเรกทอรีที่ติดตั้งซียู-โซลิดอีกทีหนึ่ง มาไว้ในไดเรกทอรีเทมเพลทของไมโครซอฟวิชชวลซีพลัสพลัส

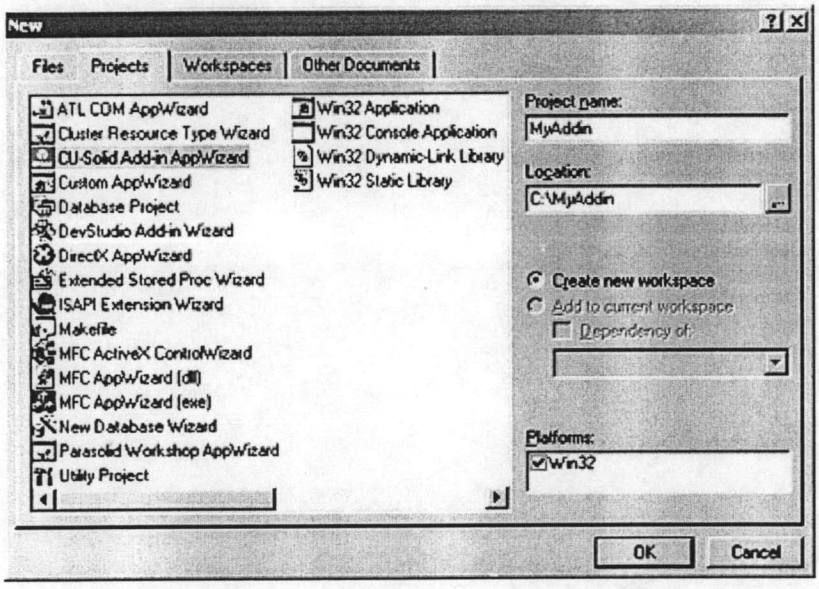
ข.1.2.2. เพิ่มไดเรกทอรี “Include” ที่อยู่ในไดเรกทอรี่ที่ติดตั้งซียู-โซลิดให้กับไมโครซอฟวิชชวลซีพลัสพลัส โดยเลือก “Tools->Options...->Directories” เลือกช่อง “Show directories for” เป็น “Include files” จากนั้นจึงพิมพ์ชื่อไดเรกทอรีที่ต้องการเพิ่มลงในแถบรายชื่อ แล้วกดปุ่มตกลง (OK)



รูปที่ ข.1 แสดงการเพิ่มไดเรกทอรีอินครูดให้กับวิซชวลซีพลัสพลัส

ข.2. การสร้างโปรเจกซอฟแวร์ส่วนประกอบเสริมสำหรับซียู-โซลิด

ในการสร้างโปรเจกซอฟแวร์ส่วนประกอบเสริมสำหรับซียู-โซลิด จะใช้ ซียู-โซลิด แอดอินแอพวิซซาร์ด (CU-Solid Add-in AppWizard) เพื่อสร้างโครงสร้างหลักของซอฟแวร์ส่วนประกอบเสริม ซึ่งมีขั้นตอนดังนี้



รูปที่ ข.2 แสดงกรอบโต้ตอบที่ใช้สร้างโปรเจกใหม่

ข.2.1. ในเมนู "File" ของวิซชวลซีพลัสพลัส เลือก "New" จะปรากฏกรอบโต้ตอบให้เลือกแถบด้านบนเป็น "Projects"

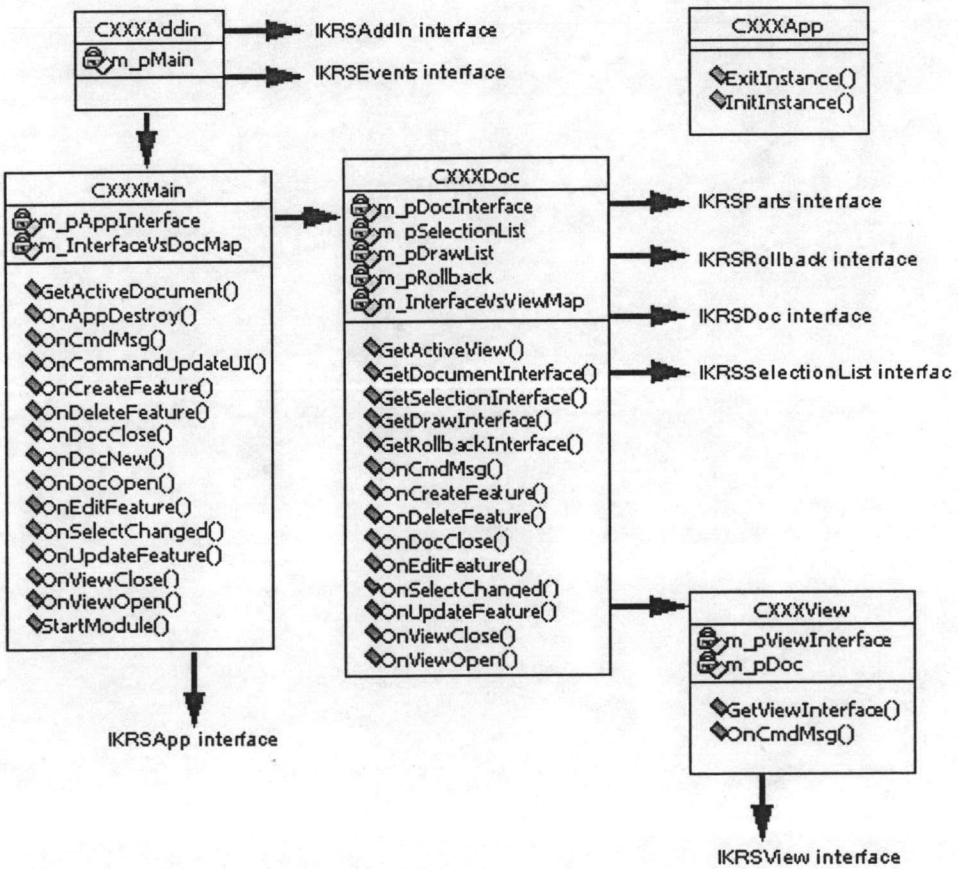
ข.2.2. เลือก "CU-Solid Add-in AppWizard"

ข.2.3. ใส่ชื่อโปรเจกในช่อง “Project name” และตำแหน่งไดเรกทอรีในช่อง “Location” เสร็จแล้วกดปุ่มตกลง

ข.2.4. ทำการยืนยันการสร้างโปรเจกใหม่เมื่อกรอบโต้ตอบสำหรับยืนยันการสร้างโปรเจกใหม่ปรากฏขึ้นโดยกดปุ่มตกลง

ข.3. โครงสร้างของโมดูล

เมื่อสร้างโปรเจกใหม่ตามขั้นตอนข้างต้นแล้วซียู-โซลิดแอดอินแอพวิซซาร์ดจะสร้างคลาสสำหรับภาษาซีขึ้น 5 คลาส และตัวต่อประสาน 1 ตัว ทั้งคลาสและตัวต่อประสานที่สร้างขึ้นมานี้จะมีชื่อขึ้นต้นเป็นชื่อของโปรเจก (สมมติให้โปรเจกชื่อ “XXX”) โดยโมดูลหลักจะต้องมีการสืบทอดมาจากตัวต่อประสาน IKRSAddIn และ IKRSEvents. คลาสและตัวต่อประสานที่สร้างขึ้นได้แก่



รูปที่ ข.3 แสดงโครงสร้างของซอฟแวร์ส่วนประกอบเสริม

ข.3.1. คลาส CXXXAddin เป็นคลาสแบบ ATL มีการสืบทอดมาจากตัวต่อประสาน IXXXAddin IKRSAddIn และ IKRSEvents ซึ่งจะมีหน้าที่เป็นตัวเก็บข้อมูลของซอฟแวร์ส่วนประกอบเสริมที่สร้าง เช่น ชื่อ ลิขสิทธิ์ สถานะการเชื่อมต่อกับซียู-โซลิด นอกจากนี้ยังเป็นตัวรับอีฟเว่นจากซียู-โซลิดเพื่อส่งต่อไปยังคลาส CXXXMain ซึ่งถูกเอนแคปซูเลสอยู่ภายในคลาสนี้

ข.3.2. คลาส CXXXApp เป็นคลาสหลักสำหรับซอฟแวร์ส่วนประกอบเสริม สืบทอดมาจากคลาส CWinApp สร้างขึ้นโดยตัวสร้างโมดูลของ MFC ประกอบด้วยฟังก์ชันดังตารางที่ ข.1

ตารางที่ ข.1 ฟังก์ชันภายในคลาส CXXXApp

ฟังก์ชัน	คำอธิบาย
BOOL InitInstance()	ทำหน้าที่ดำเนินการเริ่มต้นการทำงานของซอฟแวร์ส่วนประกอบเสริม
int ExitInstance()	ทำหน้าที่ล้างหน่วยความจำที่ไม่ต้องการใช้แล้วเมื่อจบการทำงานของซอฟแวร์ส่วนประกอบเสริม

ข.3.3. คลาส CXXXMain เป็นคลาสส่วนที่ควบคุมการทำงาน และเก็บข้อมูลหลักของซอฟแวร์ส่วนประกอบเสริมที่ใช้ร่วมกันในทุก ๆ เอกสาร โดยติดต่อกับซียู-โซลิดผ่านทางตัวต่อประสาน IKRSApp คลาส CXXXMain ประกอบด้วยฟังก์ชันดังนี้

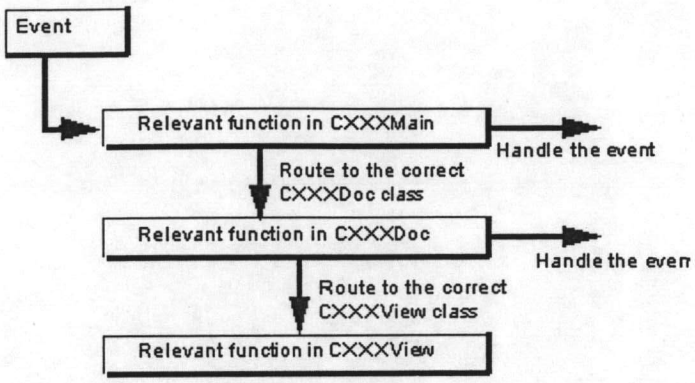
ข.3.4. คลาส CXXXDoc เป็นคลาสที่ควบคุมการทำงาน และเก็บข้อมูลของแต่ละเอกสาร ซึ่งจะติดต่อกับซียู-โซลิดผ่านทางตัวต่อประสาน IKRSDoc

ข.3.5. คลาส CXXXView เป็นคลาสที่ทำงานควบคุมมุมมองและการแสดงผล โดยติดต่อกับซียู-โซลิดผ่านทางตัวต่อประสาน IKRSView

ข.3.6. ตัวต่อประสาน IXXXAddin เป็นตัวต่อประสานที่สืบทอดมาจากตัวต่อประสาน IDispatch อีกทีหนึ่ง ตัวต่อประสานนี้จะถูกเรียกใช้โดยซียู-โซลิด

ข.4. การจัดการอีฟเว่นภายในโมดูล

ข.4.1. <u>การจัดการแมสเซจทั่วไป</u> การที่โมดูลจะรับอีฟเว่นจากซียู-โซลิดได้นั้นจำเป็นต้องมีการสืบทอดมาจากตัวต่อประสาน IKRSEvents แต่ถ้าโมดูลภายในที่ไม่ได้สืบทอดมาจาก IKRSEvents ก็สามารถรับอีฟเว่นผ่านทาง CXXXMain อีกทีหนึ่งก็ได้ แต่ถ้าสร้างโปรเจคด้วยซียู-โซลิดแอดอินแอพวิซซาร์ด คลาส CXXXAddin จะมีการสืบทอดจาก IKRSEvents อยู่แล้วพร้อมทั้งเอนแคปซูเลส CXXXMain ไว้ด้วย



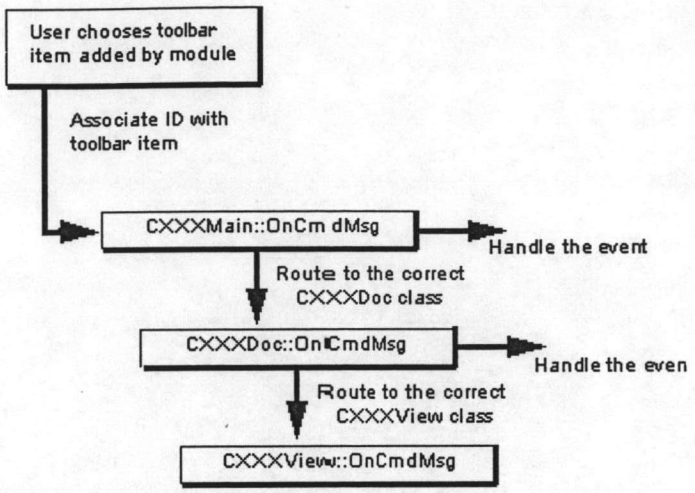
รูปที่ ข.4 แสดงการส่งผ่านอีฟเว่นแมสเซจทั่วไป

เมื่อโมดูลหลัก (CXXXAddin) ได้รับอีฟเว่นจากซียู-โซลิดจะจ่ายอีฟเว่นแมสเซจทั้งหมดต่อไปยัง CXXXMain ซึ่งอีฟเว่นที่ส่งมานี้ส่วนหนึ่งจะถูกจัดการด้วยฟังก์ชันภายในของ CXXXMain อีกส่วนหนึ่งจะถูกส่งไปยัง CXXXDoc ในคลาส CXXXDoc นี้ก็จะมีฟังก์ชันภายในส่วนหนึ่งที่จัดการกับแมสเซจที่ส่งผ่านมา และถ้ามีแมสเซจใดที่ยังไม่ได้รับการจัดการ CXXXDoc จะส่งแมสเซจที่เหลือต่อไปให้กับ CXXXView ซึ่งเป็นตัวสุดท้ายที่จะรับอีฟเว่น สำหรับโปรเจกที่สร้างด้วยซียู-โซลิดแอดอินแอพวิซซาร์ดมีการสร้างฟังก์ชันภายในเพื่อรองรับอีฟเว่นแมสเซจดังตารางที่ ข.2

ตารางที่ ข.2 แสดงฟังก์ชันรองรับอีฟเว่นในคลาสที่สร้างขึ้นโดยซียู-โซลิดแอออินแอพวิซซาจ

Class	Available message handlers
CXXXMain	OnCmdMsg OnDocNew OnDocOpen OnDocClose OnViewOpen OnViewClose OnAppDestroy OnSelectTopols OnCreateFeature OnDeleteFeature OnUpdateFeature OnEditFeature
CXXXDoc	OnCmdMsg OnDocClose OnViewOpen OnViewClose OnSelectTopols OnCreateFeature OnDeleteFeature OnUpdateFeature OnEditFeature
CXXXView	OnCmdMsg

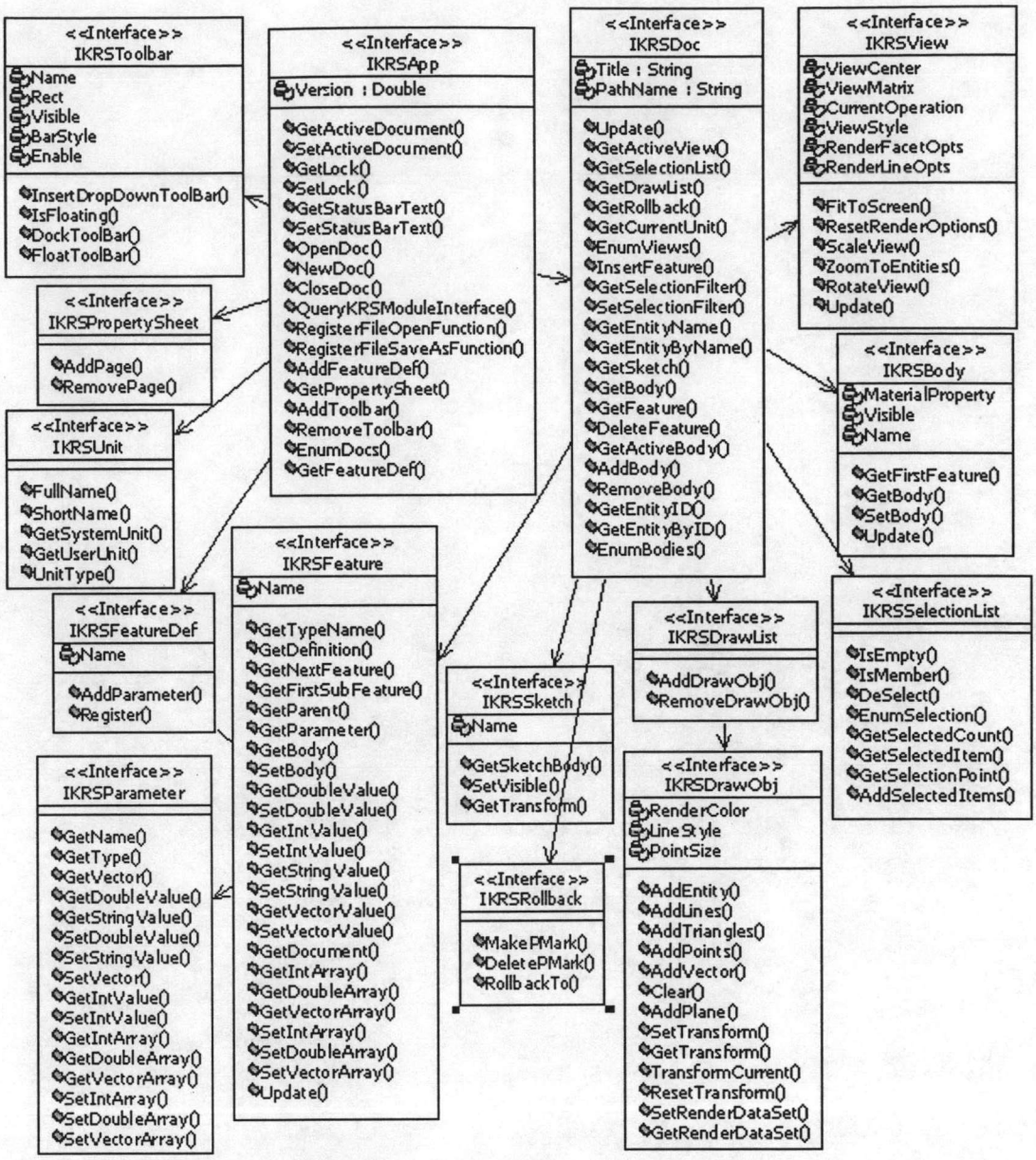
ข.4.2. <u>การจัดการแมสเซจคำสั่งจากแถบเครื่องมือ</u> เป็นอีฟเว่นแมสเซจที่ส่งโดยแถบเครื่องมือที่สร้างจากโมดูลของซอฟแวร์ส่วนประกอบเสริม โดยแมสเซจจะส่งผ่านฟังก์ชันจัดการอีฟเว่นชื่อ OnCmdMsg



รูปที่ ข.5 แสดงการส่งผ่านอีฟเว่นแมสเซจคำสั่งจากแถบเครื่องมือ

ฟังก์ชัน OnCmdMsg นอกจากจัดการอีฟเว่นคำสั่งจากแถบเครื่องมือแล้วยังจัดการอีฟเว่นแมสเซจที่ส่งมาเมื่อชนิดของแฟ้มข้อมูลที่ลงทะเบียนไว้มีการเปิด หรือบันทึกด้วย

ข.5. ตัวต่อประสานกับซียู-โซลิด



รูปที่ ข.6 แสดงตัวต่อประสานต่าง ๆ ของซียู-โซลิด

ข.5.1. IKRSAddIn เป็นตัวต่อประสานที่เก็บข้อมูลรายละเอียดของซอฟแวร์ส่วนประกอบเสริม ได้แก่ ชื่อซอฟแวร์ส่วนประกอบเสริม ชื่อผู้ผลิต ลิขสิทธิ์ นอกจากนี้ยังทำหน้าที่เริ่มต้น และหยุดการใช้งานซอฟแวร์ส่วนประกอบเสริมด้วย ตัวต่อประสาน IKRSAddIn ประกอบด้วยฟังก์ชันและพารามิเตอร์ต่างๆ ดังต่อไปนี้

ข.5.1.1. ฟังก์ชัน OnConnection เรียกเพื่อโหลดโมดูลซอฟแวร์ส่วนประกอบเสริม มีรูปแบบฟังก์ชันดังนี้

```
HRESULT OnConnection (
        IDispatch* pIKRSApp,
        IDispatch* pIKRSAddIn,
        KRSCONNECTMODE dwConnectMode
);
```

พารามิเตอร์

pIKRSApp

[in] พารามิเตอร์นี้จะส่งตัวต่อประสาน IKRSApp ที่ซียู-โซลิดส่งผ่านเข้ามาเมื่อมีการเรียกใช้งานตัวต่อประสานนี้

pIKRSAddIn

[in] พารามิเตอร์นี้จะส่งตัวต่อประสาน IKRSAddIn ในกรณีที่ซอฟแวร์ส่วนประกอบเสริมนี้ถูกเรียกใช้ด้วยตัวต่อประสานอื่นแทนการถูกเรียกใช้โดยซียู-โซลิด

dwConnectMode

[in] เป็นพารามิเตอร์ที่บอกถึงวิธีการเชื่อมต่อระหว่างซอฟแวร์ส่วนประกอบเสริมนี้กับซียู-โซลิด โดยมีค่าดังตารางที่ ข.3

ตารางที่ ข.3 แสดงค่าของตัวแปรชนิด KRSCONNECTMODE

KRSCONNECTMODE	ความหมายของค่าตัวแปร
KRSCONNECT_ATSTARTUP	โมดูลถูกโหลดตอนเริ่มต้นใช้งานซียู-โซลิด
KRSCONNECT_BYUSER	โมดูลถูกโหลดด้วยคำสั่งจากผู้ใช้โดยตรง
KRSCONNECT_EXTERNALLY	โมดูลถูกโหลดด้วยโปรแกรมอื่น ๆ ภายนอกซียู-โซลิด

ค่าส่งกลับ (Return Values)

S_OK ถ้าการเชื่อมต่อสำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.1.2. ฟังก์ชัน OnDisconnection เรียกใช้เพื่อตัดการติดต่อระหว่างซียู-โซลิดและโมดูลซอฟแวร์ส่วนประกอบเสริม มีรูปแบบฟังก์ชันดังนี้

```
HRESULT OnDisconnection (
        KRSDISCONNECTMODE dwDisconnectMode
);
```

พารามิเตอร์

dwDisconnectMode

[in] เป็นพารามิเตอร์ที่บอกถึงวิธีการตัดการเชื่อมต่อระหว่างซอฟแวร์ส่วนประกอบเสริมนี้กับซียู-โซลิด โดยมีค่าดังตารางที่ ข.4

ตารางที่ ข.4 ตารางค่าของตัวแปร KRSDISCONNECTMODE

KRSDISCONNECTMODE	ความหมายของค่าตัวแปร
KRSDISCONNECT_ATSHUTDOWN	โมดูลถูกตัดการเชื่อมต่อตอนจบการใช้ซียู-โซลิด
KRSDISCONNECT_BYUSER	โมดูลถูกตัดการเชื่อมต่อด้วยคำสั่งจากผู้ใช้
KRSDISCONNECT_EXTERNALLY	โมดูลถูกตัดการเชื่อมต่อด้วยโปรแกรมอื่น ๆ ภายนอกซียู-โซลิด

ค่าส่งกลับ

S_OK ถ้าตัดการเชื่อมต่อสำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.1.3. ฟังก์ชัน IsConnected ใช้เพื่อตรวจสอบสถานะการเชื่อมต่อระหว่างซียู-โซลิดกับโมดูลซอฟแวร์ส่วนประกอบเสริม มีรูปแบบฟังก์ชันดังนี้

```
HRESULT IsConnected(
        KRSBOOL *pConnected
);
```

พารามิเตอร์

pConnected

[out] เป็นบัฟเฟอร์ที่จะใช้รับค่าแสดงการเชื่อมต่อ ถ้าเป็น KRS_TRUE แสดงว่าโมดูลนี้กำลังอยู่ในสถานะเชื่อมต่อกับซียู-โซลิด แต่ถ้าเป็น KRS_FALSE แสดงว่าไม่ได้เชื่อมต่อกับซียู-โซลิด

ค่าส่งกลับ

S_OK ถ้าฟังก์ชันถูกเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.1.4. คุณสมบัติ Name เป็นข้อมูลที่แสดงชื่อของโมดูลซอฟแวร์ส่วนประกอบเสริม เป็นคุณสมบัติแบบอ่านอย่างเดียว (prop get) มีรูปแบบดังนี้

```
HRESULT Name (
        BSTR *pVal
);
```

พารามิเตอร์

pVal

[out, retval] เป็นบัฟเฟอร์ใช้รับชื่อของโมดูลซอฟแวร์ส่วนประกอบเสริม

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.1.5. คุณสมบัติ CopyRight เป็นข้อความที่แสดงลิขสิทธิ์ของโมดูลซอฟแวร์ส่วนประกอบเสริม ซึ่งมีรูปแบบดังนี้

```
HRESULT CopyRight (
        BSTR *pVal
);
```

พารามิเตอร์

pVal

[out, retval] เป็นบัฟเฟอร์ใช้รับข้อความที่ใช้แสดงลิขสิทธิ์ของโมดูลซอฟแวร์ส่วนประกอบเสริม

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.1.6. คุณสมบัติ InfoURL เป็นข้อความแสดง URL ของเวบไซด์ที่แสดงข้อมูลของซอฟแวร์ส่วนประกอบเสริม ซึ่งมีรูปแบบดังนี้

```
HRESULT InfoURL (
        BSTR *pVal
);
```

พารามิเตอร์

pVal

[out, retval] เป็นบัฟเฟอร์ใช้รับข้อความที่ใช้ URL ของเวบไซต์ที่เกี่ยวข้องกับโมดูลซอฟแวร์ส่วนประกอบเสริม

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2. IKRSEvents เป็นตัวต่อประสานที่ทำหน้าที่รับอีฟเว่น และคำสั่งจากซียู-โซลิด ซึ่งอีฟเว่นที่รับได้แก่ การสร้างเอกสารใหม่ การเปิดเอกสาร การปิดเอกสาร การสร้างมุมมองภาพใหม่ การปิดมุมมองภาพ การเปลี่ยนแปลงการเลือกองค์ประกอบ และการกดปุ่มคำสั่งจากแถบเครื่องมือที่สร้างจากซอฟแวร์ส่วนประกอบเสริมนั้น IKRSEvents ประกอบด้วยฟังก์ชันดังนี้

ข.5.2.1. ฟังก์ชัน OnCommand เป็นฟังก์ชันที่ถูกเรียกเพื่อส่งอีฟเว่นแมสเซจคำสั่งจากแถบเครื่องมือ หรือเมื่อต้องการเปิดและบันทึกแฟ้มข้อมูลชนิดที่ลงทะเบียนไว้ มีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnCommand (
    long CommandID,
    void* wparam,
    void* lparam
);
```

พารามิเตอร์

CommandID

[in] รหัสของแมสเซจที่ส่งมายังฟังก์ชัน

wparam

[in] พารามิเตอร์ที่ส่งมาพร้อมกับแมสเซจ

lparam

[in] พารามิเตอร์ที่ส่งมาพร้อมกับแมสเซจ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.2. ฟังก์ชัน OnCommandUpdateUI เป็นฟังก์ชันที่จะรับแมสเซจการปรับปรุงแถบเครื่องมือ ซึ่งมีรูปของฟังก์ชันดังนี้

```
HRESULT OnCommandUpdateUI (
        long CommandID,
        long* CommandFlags,
        BSTR* ButtonItemText,
        long* BitmapID
);
```

พารามิเตอร์

CommandID

[in] รหัสคำสั่งของปุ่มบนแถบเครื่องมือที่ต้องการปรับปรุง

CommandFlags

[in, out] เป็นตัวแปรที่บ่งบอกถึงสถานะที่ต้องการปรับปรุง มีค่าดังตารางที่ ข.5

ตารางที่ ข.5 ตารางค่าตัวแปรชนิด KRSCMDUI

KRSCMDUI	ความหมายของค่าตัวแปร
KRSCMDUI_ENABLE	กำหนดสถานะพร้อมใช้งาน
KRSCMDUI_DISABLE	กำหนดสถานะที่ไม่สามารถใช้งานได้
KRSCMDUI_CHECKED	ปุ่มอยู่ในสถานะถูกกดอยู่
KRSCMDUI_UNCHECKED	ยกเลิกสถานะของปุ่มที่ถูกกด
KRSCMDUI_CHANGETEXT	เปลี่ยนตัวข้อความบนปุ่ม
KRSCMDUI_USEBMP	เปลี่ยนรูปภาพบนปุ่ม

ButtonItemText

[out] บัฟเฟอร์สำหรับส่งข้อความที่เปลี่ยนแปลงกลับเมื่อตัวแปร CommandFlags = KRSCMDUI_CHANGETEXT

BitmapID

[out] บัฟเฟอร์สำหรับส่งค่ารหัสรีซอสของภาพบนปุ่มที่ต้องการเปลี่ยนเมื่อ CommandFlags = KRSCMDUI_USEBMP

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.3. ฟังก์ชัน OnDocClose เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจสำหรับปิดเอกสารแบบจำลองโซลิด มีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnDocClose (
        IKRSDoc* pDoc
);
```

พารามิเตอร์

pDoc

[in] ตัวต่อประสานของเอกสารที่จะปิด

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.4. ฟังก์ชัน OnDocNew เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อมีการสร้างเอกสารแบบจำลองโซลิดใหม่ มีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnDocNew (
        IKRSDoc* pDoc
);
```

พารามิเตอร์

pDoc

[in] ตัวต่อประสานของเอกสารที่สร้างขึ้นใหม่

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.5. ฟังก์ชัน OnDocOpen เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อมีการเปิดเอกสารแบบจำลองโซลิดจากแฟ้มข้อมูล ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnDocOpen (
        IKRSDoc* pDoc
);
```

พารามิเตอร์

pDoc

[in] ตัวต่อประสานของเอกสารที่ถูกเปิด

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.6. ฟังก์ชัน OnSelectChanged เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อเปลี่ยนแปลงองค์ประกอบที่ถูกเลือกโดยผู้ใช้ ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnSelectChanged ();
```

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.7. ฟังก์ชัน OnViewClose เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อมีการปิดหน้าต่างมุมมองแบบจำลองโซลิด ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnViewClose (
    IKRSDoc* pDoc,
    IKRSView* pView
);
```

พารามิเตอร์

pDoc

[in] ตัวต่อประสานของเอกสารของมุมมองที่ถูกปิด

pView

[in] ตัวต่อประสานของมุมมองที่ถูกปิด

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.8. ฟังก์ชัน OnViewOpen เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อมีการเปิดหน้าต่างมุมมองแบบจำลองโซลิด ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnViewOpen (
        IKRSDoc* pDoc,
        IKRSView* pView
);
```

พารามิเตอร์

pDoc

[in] ตัวต่อประสานของเอกสารของมุมมองที่ถูกเปิด

pView

[in] ตัวต่อประสานของมุมมองที่ถูกเปิด

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.9. ฟังก์ชัน OnUpdateFeature เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อมีการปรับปรุงค่าพารามิเตอร์ของลักษณะจำเพาะ ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnUpdateFeature (
        IKRSFeature *pFeatureInterface
);
```

พารามิเตอร์

pFeatureInterface

[in] ตัวต่อประสานของลักษณะจำเพาะที่ต้องการปรับปรุง

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.10. ฟังก์ชัน OnEditFeature เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อผู้ใช้ต้องการแก้ไขพารามิเตอร์ของลักษณะจำเพาะ ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnEditFeature (
        IKRSFeature *pFeatureInterface
);
```

พารามิเตอร์

pFeatureInterface

[in] ตัวต่อประสานของลักษณะจำเพาะที่ต้องการแก้ไข

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.11. ฟังก์ชัน OnCreateFeature เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อมีการสร้างลักษณะจำเพาะขึ้นใหม่ ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnCreateFeature (
    IKRSFeature *pFeatureInterface
);
```

พารามิเตอร์

pFeatureInterface

[in] ตัวต่อประสานของลักษณะจำเพาะใหม่ที่สร้างขึ้น

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.2.12. ฟังก์ชัน OnDeleteFeature เป็นฟังก์ชันที่เรียกใช้เพื่อส่งอีฟเว่นแมสเซจเมื่อมีการลบลักษณะจำเพาะ ซึ่งมีรูปแบบของฟังก์ชันดังนี้

```
HRESULT OnDeleteFeature (
    IKRSFeature *pFeatureInterface
);
```

พารามิเตอร์

pFeatureInterface

[in] ตัวต่อประสานของลักษณะจำเพาะที่จะถูกลบ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3. IKRSApp เป็นตัวต่อประสานหลัก มีหน้าที่ติดต่อกับข้อมูลทั่วไปของตัวโปรแกรมหลัก ได้แก่ ข้อมูลชนิดของลักษณะจำเพาะ แถบเครื่องมือ ชนิดของไฟล์ที่สามารถเปิดหรือบันทึกได้ รวมถึงการสร้างเอกสารใหม่ การเปิดเอกสารที่มีอยู่ การบันทึกเอกสาร การเรียกไปยังซอฟแวร์ส่วนประกอบเสริมอื่น ๆ ที่ได้ลงทะเบียนไว้ และการนิยามชนิดของลักษณะจำเพาะใหม่ ตัวต่อประสาน IKRSApp ประกอบด้วยฟังก์ชัน และคุณสมบัติ ดังนี้

ข.5.3.1. ฟังก์ชัน GetActiveDocument เป็นฟังก์ชันใช้เรียกเอกสารที่ผู้ใช้กำลังทำงานอยู่ ณ ขณะที่เรียกฟังก์ชัน มีรูปแบบดังนี้

```
HRESULT GetActiveDocument (
        IKRSDoc** ppDocInterface
);
```

พารามิเตอร์

ppDocInterface

[out, retval] ตัวต่อประสานของเอกสารที่อยู่ในสภาวะแอคทีฟ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.2. ฟังก์ชัน SetActiveDocument เป็นฟังก์ชันใช้กำหนดเอกสารที่ต้องการให้อยู่ในสภาวะแอคทีฟ มีรูปแบบดังนี้

```
HRESULT SetActiveDocument (
        IKRSDoc* pDocInterface
);
```

พารามิเตอร์

pDocInterface

[in] ตัวต่อประสานของเอกสารที่ต้องการให้อยู่ในสภาวะแอคทีฟ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.3. ฟังก์ชัน GetLock เป็นฟังก์ชันที่ใช้อ่านค่าการปิดกั้นการเรียกใช้คำสั่งจากแถบเครื่องมือ มีรูปแบบดังนี้

```
HRESULT GetLock (
        KRSBOOL*    pLock
);
```

พารามิเตอร์

pLock

[out, retval] บัฟเฟอร์รองรับค่าที่บอกสภาวะการปิดกั้นการเรียกใช้งานคำสั่งจากแถบเครื่องมือ ถ้ามีค่าเป็น KRS_TRUE แสดงว่าแถบเครื่องมือถูกปิดกั้นการเรียกคำสั่ง แต่ถ้าเป็น KRS_FALSE แสดงว่าสามารถเรียกใช้งานได้

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.4. ฟังก์ชัน SetLock เป็นฟังก์ชันที่ใช้กำหนดค่าการปิดกั้นการเรียกใช้คำสั่งจากแถบเครื่องมือ มีรูปแบบดังนี้

```
HRESULT SetLock (
        KRSBOOL     bLock
);
```

พารามิเตอร์

bLock

[in] ค่าที่กำหนดสภาวะการปิดกั้นการเรียกใช้งานคำสั่งจากแถบเครื่องมือ ถ้ามีค่าเป็น KRS_TRUE แสดงว่าแถบเครื่องมือถูกกำหนดให้ปิดกั้นการเรียกคำสั่ง แต่ถ้าเป็น KRS_FALSE แสดงว่ากำหนดให้เรียกใช้งานได้

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.5. ฟังก์ชัน GetStatusBarText เป็นฟังก์ชันที่ใช้อ่านข้อความที่แสดงอยู่บนแถบแสดงสถานะ (Status bar) มีรูปแบบดังนี้

```
HRESULT GetStatusBarText (
        int iPane,
        BSTR* pVal
);
```

พารามิเตอร์

iPane

[in] ช่องของแถบแสดงสถานะที่ต้องการอ่านค่าข้อความ

pVal

[out, retval] บัฟเฟอร์รองรับข้อความบนแถบแสดงสถานะที่ส่งกลับ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.6. ฟังก์ชัน SetStatusBarText เป็นฟังก์ชันที่ใช้กำหนดข้อความที่แสดงอยู่บนแถบแสดงสถานะ (Status bar) มีรูปแบบดังนี้

```
HRESULT SetStatusBarText (
        int iPane,
        BSTR newVal
);
```

พารามิเตอร์

iPane

[in] ช่องของแถบแสดงสถานะที่ต้องการกำหนดค่าข้อความ

newVal

[in] ข้อความที่ต้องการให้แสดงบนแถบแสดงสถานะ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.7. คุณสมบัติ Version (propget) เป็นคุณสมบัติที่แสดงค่าเวอร์ชันของซียู-โซลิด มีรูปแบบการเรียกใช้ดังนี้

```
HRESULT Version (
        double* pVersion
);
```

พารามิเตอร์

pVersion

[out, retval] บัฟเฟอร์รองรับค่าเวอร์ชันที่ส่งกลับ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.8. ฟังก์ชัน OpenDoc เป็นฟังก์ชันเรียกใช้เพื่อเปิดเอกสารจากแฟ้มข้อมูลแบบจำลองโซลิดที่มีอยู่ มีรูปแบบดังนี้

```
HRESULT OpenDoc (
        BSTR fileName,
        IKRSDoc** ppDoc
);
```

พารามิเตอร์

fileName

[in] ชื่อแฟ้มข้อมูลที่ต้องการเปิด

ppDoc

[out, retval] บัฟเฟอร์รองรับตัวต่อประสานของเอกสารที่ถูกเปิดด้วยฟังก์ชัน

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.9. ฟังก์ชัน NewDoc เป็นฟังก์ชันเรียกใช้เพื่อเปิดเอกสารแบบจำลองโซลิดใหม่ มีรูปแบบดังนี้

```
HRESULT NewDoc (
        IKRSDoc** ppDoc
);
```

พารามิเตอร์

ppDoc

[out, retval] บัฟเฟอร์รองรับตัวต่อประสานของเอกสารใหม่ที่ถูกสร้าง

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.10. ฟังก์ชัน CloseDoc เป็นฟังก์ชันเรียกใช้เพื่อปิดเอกสารแบบจำลองโซลิด มีรูปแบบดังนี้

```
HRESULT CloseDoc (
    IKRSDoc* pDoc
);
```

พารามิเตอร์

pDoc

[in] ตัวต่อประสานของเอกสารที่ต้องการปิด

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.11. ฟังก์ชัน QueryKRSModuleInterface เป็นฟังก์ชันเรียกใช้เพื่อถามหาตัวต่อประสานของซอฟแวร์ส่วนประกอบเสริมที่ลงทะเบียนไว้กับซียู-โซลิด มีรูปแบบดังนี้

```
HRESULT QueryKRSModuleInterface (
    GUID* riid,
    IUnknown** ppUnknown
);
```

พารามิเตอร์

riid

[in] เลขรหัสประจำคอมโพเนนท์ที่ต้องการ

ppUnknown

[out, retval] บัฟเฟอร์รองรับตัวต่อประสานอันโนนของตัวต่อประสานที่ต้องการ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.12. ฟังก์ชัน RegisterFileOpenFunction เป็นฟังก์ชันเรียกใช้เพื่อลงทะเบียนชนิดของแฟ้มข้อมูลแบบจำลองโซลิดที่สามารถเปิดด้วยซอฟแวร์ส่วนประกอบเสริม มีรูปแบบดังนี้

```
HRESULT RegisterFileOpenFunction (
        IKRSAddIn* pAddIn,
        BSTR description,
        long CommandID
);
```

พารามิเตอร์

pAddIn

[in] ตัวต่อประสานของซอฟแวร์ส่วนประกอบเสริมที่จะเป็นตัวเปิดชนิดของแฟ้มข้อมูลแบบจำลองโซลิดที่ลงทะเบียน

description

[in] เป็นค่าที่ใช้กำหนดตัวกรองของกรอบโต้ตอบที่ใช้เปิดเลือกแฟ้มข้อมูลแบบจำลองโซลิด จะมีสองส่วน ส่วนแรกเป็นคำอธิบาย หรือชื่อชนิดของแฟ้มข้อมูลแบบจำลองโซลิด อีกส่วนเป็นนามสกุลที่จะใช้เป็นตัวกรอง ทั้งสองส่วนจะเขียนต่อกันและคั่นกลางด้วยสัญลักษณ์ "|" และใช้คั่นเมื่อตอนจบข้อความด้วย เช่น "SETP Files|(*.stp)|"

CommandID

[in] เป็นรหัสคำสั่งที่จะส่งมาพร้อมกับอีฟเว่น OnCommand เมื่อมีการเรียกเปิดแฟ้มข้อมูลแบบจำลองโซลิดชนิดที่ลงทะเบียนไว้

ค่าส่งกลับ

S_OK ถ้าการลงทะเบียนสำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.13. ฟังก์ชัน RegisterFileSaveAsFunction เป็นฟังก์ชันเรียกใช้เพื่อลงทะเบียนชนิดของแฟ้มข้อมูลแบบจำลองโซลิดที่สามารถบันทึกด้วยซอฟแวร์ส่วนประกอบเสริม มีรูปแบบดังนี้

```
HRESULT RegisterFileSaveAsFunction (
        IKRSAddIn* pAddIn,
        BSTR description,
        long CommandID
);
```

พารามิเตอร์

pAddIn

[in] ตัวต่อประสานของซอฟแวร์ส่วนประกอบเสริมที่จะเป็นตัวบันทึกชนิดของแฟ้มข้อมูลแบบจำลองโซลิดที่ลงทะเบียน

description

[in] เป็นค่าที่ใช้กำหนดตัวกรองของกรอบโต้ตอบที่ใช้เลือกแฟ้มข้อมูลแบบจำลองโซลิดที่จะบันทึก จะมีสองส่วน ส่วนแรกเป็นคำอธิบาย หรือชื่อชนิดของแฟ้มข้อมูลแบบจำลองโซลิด อีกส่วนเป็นนามสกุลที่จะใช้เป็นตัวกรอง ทั้งสองส่วนจะเขียนต่อกันและคั่นกลางด้วยสัญลักษณ์ "|" และใช้คั่นเมื่อตอนจบข้อความด้วย เช่น "SETP Files|(*.stp)|"

CommandID

[in] เป็นรหัสคำสั่งที่จะส่งมาพร้อมกับอีฟเว่น OnCommand เมื่อมีการเรียกคำสั่งบันทึกแฟ้มข้อมูลแบบจำลองโซลิดชนิดที่ลงทะเบียนไว้

ค่าส่งกลับ

S_OK ถ้าการลงทะเบียนสำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.14. ฟังก์ชัน AddFeatureDef เป็นฟังก์ชันเรียกใช้เพื่อนิยามลักษณะจำเพาะชนิดใหม่ มีรูปแบบดังนี้

```
HRESULT AddFeatureDef (
        BSTR szName,
        IKRSAddIn* pAddInInterface,
        UINT uiBitmap,
        COLORREF clrMask,
        void* hResource,
        IKRSFeatureDef** ppFeatureDefInterface
);
```

พารามิเตอร์

szName

[in] ชื่อของลักษณะจำเพาะที่ต้องการนิยาม

pAddInInterface

[in] ตัวต่อประสานของซอฟแวร์ส่วนประกอบเสริมที่จะทำหน้าที่ปรับปรุงและแก้ไขแบบจำลองโซลิดด้วยลักษณะจำเพาะที่นิยามขึ้น

uiBitnap

[in] รหัสรีซอสของรูปที่จะใช้แสดงแทนลักษณะจำเพาะในทรีของลักษณะจำเพาะ ภาพที่ใช้จะต้องมีขนาด กว้าง 16 พิกเซล และยาว 15 พิกเซล เท่านั้น

clrMask

[in] เป็นสีของรูปภาพที่กำหนดโดย uiBitmap ที่ต้องการให้โปร่งใส

hResource

[in] แฮนเดิลของรีซอสที่เก็บรูปภาพที่กำหนดโดย uiBitmap

ppFeatureDefInterface

[out, retval] บัฟเฟอร์รองรับตัวต่อประสานนิยามของลักษณะจำเพาะที่นิยามขึ้น

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.15. ฟังก์ชัน GetPropertySheet เป็นฟังก์ชันที่ใช้ตัวต่อประสานที่ติดต่อกับกรอบกำหนดคุณสมบัติ (Property Sheet) มีรูปแบบดังนี้

```
HRESULT GetPropertySheet (
    IKRSPropertySheet** ppPropSheet
);
```

พารามิเตอร์

ppPropSheet

[out, retval] บัฟเฟอร์รองรับตัวต่อประสานกรอบกำหนดคุณสมบัติ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.16. ฟังก์ชัน AddToolbar เป็นฟังก์ชันเรียกใช้เพื่อแถบเครื่องมือ มีรูปแบบดังนี้

```
HRESULT AddToolbar (
        BSTR szName,
        long ResourceID,
        long hResource,
        IKRSAddIn* pAddInInterface,
        IKRSToolbar** ppToolbarInterface
);
```

พารามิเตอร์

szName

[in] ชื่อของลักษณะจำเพาะที่ต้องการนิยาม

ResourceID

[in] รหัสรีซอสของแถบเครื่องมือที่ต้องการสร้าง โดยปุ่มบนแถบเครื่องมือจะต้องมีขนาดกว้าง 16 พิกเซล ยาว 15 พิกเซล

hResource

[in] แฮนเดิลของรีซอสที่แถบเครื่องมือที่กำหนดด้วย ResourceID

pAddInInterface

[in] ตัวต่อประสานของซอฟแวร์ส่วนประกอบเสริมที่รับอีฟเว่นคำสั่งจากแถบเครื่องมือที่สร้าง

ppToolbarInterface

[out, retval] บัฟเฟอร์รองรับตัวต่อประสานของแถบเครื่องมือที่สร้าง

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.17. ฟังก์ชัน RemoveToolbar เป็นฟังก์ชันที่ใช้ลบแถบเครื่องมือที่สร้างขึ้นจากคำสั่ง AddToolbar มีรูปแบบดังนี้

```
HRESULT RemoveToolbar (
        IKRSToolbar* pToolbarInterface
);
```

พารามิเตอร์

pToolbarInterface

[in] ตัวต่อประสานของแถบเครื่องมือที่ต้องการลบ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.3.18. ฟังก์ชัน EnumDocs เป็นฟังก์ชันที่ใช้เรียกเอกสารทั้งหมดที่กำลังใช้งานอยู่ในรูปตัวต่อประสานชนิดอีนัม (Enum) มีรูปแบบดังนี้

```
HRESULT EnumDocs (
        IUnknown** ppEnumDocs
);
```

พารามิเตอร์

ppEnumDocs

[in] บัฟเฟอร์รองรับตัวต่อประสานอันโนนของตัวต่อประสานอีนัมที่เก็บค่าตัวต่อประสานของเอกสารทั้งหมดที่เปิดใช้งานอยู่

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

หมายเหตุ

หลังจากเรียกใช้ฟังก์ชันนี้สำเร็จ ต้องลบหน่วยความจำที่เก็บตัวต่อประสานอีนัมเมื่อไม่ได้ใช้งานแล้วด้วยการเรียกใช้ฟังก์ชัน Release ของตัวต่อประสานอีนัมที่ต้องการลบทิ้ง

ข.5.3.19. ฟังก์ชัน GetFeatureDef เป็นฟังก์ชันที่ใช้เรียกนิยามของลักษณะจำเพาะที่นิยามไว้ มีรูปแบบดังนี้

```
HRESULT GetFeatureDef (
        BSTR bszName,
        IKRSFeatureDef** ppFeatureDefInterface
);
```

พารามิเตอร์

bszName

[in] ชื่อนิยามของลักษณะจำเพาะที่ต้องการ

ppEnumDocs

[in] บัฟเฟอร์รองรับตัวต่อประสานนิยามลักษณะจำเพาะที่ต้องการ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.4. IKRSDoc มีทำหน้าที่ติดต่อกับตัวต่อประสานอื่นๆ ที่ใช้จัดการข้อมูลภายในเอกสารให้กับซอฟแวร์ส่วนประกอบเสริม เช่น IKRSBody IKRSFeature IKRSDrawObj IKRSSketch IKRSUnit IKRSDrawList IKRSSelectionList IKRSView IKRSRollback นอกจากนี้ยังใช้ติดต่อกับส่วนจัดการข้อมูลแบบจำลองโซลิดในเอกสารอีกด้วย

ข.5.5. IKRSView เป็นตัวต่อประสานที่ใช้ติดต่อ และจัดการกับมุมมองของแบบจำลอง โดยจะมีฟังก์ชันการหมุน การเลื่อนตำแหน่ง การซูม การเปลี่ยนทรานสฟอร์มเมชันเมตริกของมุมมอง การซูมเต็มจอ และการเปลี่ยนค่าตัวเลือกเรนเดอร์สำหรับพาราโซลิดด้วย

ข.5.6. IKRSSelectionList เป็นตัวต่อประสานที่ทำหน้าที่ติดต่อระหว่าซอฟแวร์ส่วนประกอบเสริมกับส่วนจัดการการเลือกองค์ประกอบโดยผู้ใช้ของซียู-โซลิด โดยสามารถอ่านข้อมูลองค์ประกอบที่ถูกเลือกโดยผู้ใช้ได้ มีฟังก์ชันใช้งานดังนี้

ข.5.6.1. ฟังก์ชัน IsEmpty ใช้ตรวจสอบว่ามีองค์ประกอบที่ถูกเลือกอยู่หรือไม่ มีรูปแบบฟังก์ชันดังนี้

```
HRESULT IsEmpty (
        KRSBOOL* pEmpty
);
```

พารามิเตอร์

pEmpty

[out, retval] บัฟเฟอร์ค่าตรวจสอบถ้า เป็น KRS_TRUE แสดงว่าไม่มีองค์ประกอบที่ถูกเลือกอยู่ แต่ถ้าเป็น KRS_FALSE แสดงว่ามีองค์ประกอบที่ถูกเลือก

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.6.2. ฟังก์ชัน IsMember ใช้ตรวจสอบว่ามีองค์ประกอบว่าถูกเลือกอยู่หรือไม่ มีรูปแบบฟังก์ชันดังนี้

```
HRESULT IsMember (
    PK_ENTITY_t Entity,
    KRSBOOL* pMember
);
```

พารามิเตอร์

Entity

[in] องค์ประกอบที่ต้องการตรวจสอบ

pMember

[out, retval] บัฟเฟอร์รองรับค่าตรวจสอบที่ส่งกลับ ถ้าเป็น KRS_TRUE แสดงว่าองค์ประกอบนี้ถูกเลือกโดยผู้ใช้ แต่ถ้าเป็น KRS_FALSE แสดงว่าไม่ได้ถูกเลือก

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.6.3. ฟังก์ชัน DeSelect ใช้ยกเลือกองค์ประกอบที่ถูกเลือกโดยผู้ใช้ทั้งหมด มีรูปแบบดังนี้

```
HRESULT DeSelect ();
```

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.6.4. ฟังก์ชัน EnumSelection รับค่าองค์ประกอบที่ถูกเลือกทั้งหมดในรูปตัวต่อประสานอีนัม มีรูปแบบฟังก์ชันดังนี้

```
HRESULT EnumSelection (
        IUnknown** pUnknown
);
```

พารามิเตอร์

pUnknown

[out, retval] บัฟเฟอร์รองรับตัวต่อประสานอันโนนของตัวต่อประสานอีนัมที่เก็บองค์ประกอบที่ถูกเลือกทั้งหมด

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

หมายเหตุ

เมื่อเลือกใช้งานตัวต่อประสานอีนัม จะต้องลบออกจากหน่วยความจำด้วยฟังก์ชัน Release ของตัวต่อประสานอีนัม

ข.5.6.5. ฟังก์ชัน GetSelectedCount ใช้นับองค์ประกอบทั้งหมดที่ถูกเลือกโดยผู้ใช้ มีรูปแบบฟังก์ชันดังนี้

```
HRESULT GetSelectedCount (
        long* pCount
);
```

พารามิเตอร์

pCount

[out, retval] บัฟเฟอร์รองรับค่าจำนวนองค์ประกอบทั้งหมดที่ถูกเลือกโดยผู้ใช้

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.6.6. ฟังก์ชัน GetSelectedItemt ใช้อ่านค่าองค์ประกอบที่ถูกเลือกโดยผู้ใช้ ซึ่งต้องกำหนดลำดับที่ถูกเลือกด้วย มีรูปแบบฟังก์ชันดังนี้

```
HRESULT GetSelectedItem (
        long index,
        PK_ENTITY_t* pEntity
);
```

พารามิเตอร์

index

[in] ลำดับการเลือกขององค์ประกอบที่ต้อง

pEntity

[out, retval] บัฟเฟอร์รองรับองค์ประกอบที่ถูกเลือกในลำดับที่ต้องการ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.7. IKRSDrawList เป็นตัวต่อประสานที่ทำหน้าที่ติดต่อกับข้อมูลวัตถุภาพสามมิติที่วาดโดยซอฟแวร์ส่วนประกอบเสริม ซึ่งวัตถุนี้จะเป็นภาพที่แสดงวัตถุตัวอย่าง หรือภาพสัญลักษณ์ไม่เกี่ยวข้อกับภาพของแบบจำลองที่สร้าง มีฟังก์ชันใช้งานดังนี้

ข.5.7.1. ฟังก์ชัน AddDrawObj ใช้เพิ่มตัวต่อประสาน IKRSDrawObj ให้กับเอกสาร มีรูปแบบการใช้งานดังนี้

```
HRESULT AddDrawObj (
        IKRSDrawObj** ppDrawObj
);
```

พารามิเตอร์

ppDrawObj

[in] บัฟเฟอร์รองรับตัวต่อประสาน IKRSDrawObj ที่เพิ่มให้กับเอกสาร

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.7.2. ฟังก์ชัน RemoveDrawObj ใช้ลบตัวต่อประสาน IKRSDrawObj ออกจากเอกสาร มีรูปแบบการใช้งานดังนี้

```
HRESULT RemoveDrawObj (
        IKRSDrawObj* pDrawObj
);
```

พารามิเตอร์

ppDrawObj

[in] ตัวต่อประสาน IKRSDrawObj ที่ต้องการลบ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.8. IKRSPropertySheet เป็นตัวต่อประสานที่ใช้ติดต่อกับกรอบกำหนดคุณสมบัติ (Property Sheet) ของซียู-โซลิด เพื่อสร้างหน้าที่ใช้กำหนดคุณลักษณะ (Property Page) โดยซอฟแวร์ส่วนประกอบเสริม มีฟังก์ชันใช้งานดังนี้

ข.5.8.1. ฟังก์ชัน AddPage ใช้เพิ่มหน้ากำหนดคุณสมบัติ (Property Page) ลงในกรอบกำหนดคุณสมบัติ มีรูปแบบการใช้งานดังนี้

```
HRESULT AddPage (
        void* pPropPage
);
```

พารามิเตอร์

pPropPage

[in] พอยเตอร์ของคลาสหน้ากำหนดคุณสมบัติที่ต้องการเพิ่ม คลาสหน้ากำหนดคุณสมบัติจะต้องมีการสืบทอดมาจากคลาส CPropertyPage

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.8.2. ฟังก์ชัน RemovePage ใช้ลบหน้ากำหนดคุณสมบัติออกจากกรอบกำหนดคุณสมบัติ มีรูปแบบการใช้งานดังนี้

```
HRESULT RemovePage (
        void* pPropPage
);
```

พารามิเตอร์

pPropPage

[in] พอยเตอร์ของคลาสหน้ากำหนดคุณสมบัติที่ต้องการลบ คลาสหน้ากำหนดคุณสมบัติจะต้องมีการสืบทอดมาจากคลาส CPropertyPage

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.9. IKRSBody เป็นตัวต่อประสานที่ใช้ติดต่อกับข้อมูลของลำตัวที่สร้างขึ้น ซึ่งข้อมูลที่สามารถติดต่อได้คือ ชื่อของลำตัว คุณสมบัติของวัสดุ ค่าลำตัวจากพาราโซลิด การมองเห็น และลักษณะจำเพาะที่กระทำต่อลำตัวนี้

ข.5.10. IKRSSketch เป็นตัวต่อประสานที่ใช้ติดต่อกับข้อมูลแบบร่างบนระนาบ ข้อมูลที่สามารถเข้าถึงได้คือ ชื่อของแบบร่างบนระนาบ ระนาบที่ใช้สร้างแบบร่าง และข้อมูลลำตัวของแบบร่างบนระนาบจากพาราโซลิด มีรูปแบบฟังก์ชัน และคุณสมบัติดังนี้

ข.5.10.1. คุณสมบัติ Name เป็นคุณสมบัติแสดงถึงชื่อของแบบร่างบนระนาบ (propget)

```
HRESULT Name (
    BSTR* bszName
);
```

พารามิเตอร์

bszName

[out, retval] บัฟเฟอร์รองรับชื่อของแบบร่างบนระนาบ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.10.2. ฟังก์ชัน GetSketchBody เป็นฟังก์ชันใช้อ่านข้อมูลลำตัวเส้นที่เกิดจากแบบร่างบนระนาบ

```
HRESULT GetSketchBody (
    PK_BODY_t* pSktBody
);
```

พารามิเตอร์

pSktBody

[out, retval] บัฟเฟอร์รองรับลำตัวเส้นที่เกิดจากแบบร่างบนระนาบ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.10.3. ฟังก์ชัน SetVisible เป็นฟังก์ชันใช้กำหนดการมองเห็นของแบบร่างบนระนาบ

```
HRESULT SetVisible (
    KRSBOOL* pVisible
);
```

พารามิเตอร์

pVisible

[in] ถ้าเป็น KRS_TRUE จะเป็นการกำหนดให้แบบร่างบนระนาบสามารถมองเห็นได้ แต่ถ้าเป็น KRS_FALSE จะเป็นการซ่อนแบบร่างบนระนาบ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.10.4. ฟังก์ชัน GetTransform เป็นฟังก์ชันใช้อ่านค่าทรานส์ฟอร์มเมชันของระนาบที่แบบร่างบนระนาบวางตัวอยู่

```
HRESULT GetTransform (
    KRSMATRIX* pMatrix
);
```

พารามิเตอร์

pSktBody

[out, retval] บัฟเฟอร์รองรับเมตริกทรานส์ฟอร์มเมชันของระนาบที่แบบร่างบนระนาบวางตัวอยู่

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.11. IKRSDrawObj เป็นตัวต่อประสานที่ใช้ติดต่อกับข้อมูลภาพวาดโดยซอฟแวร์ส่วนประกอบเสริม ที่เป็นภาพแสดงสถานะ แบบจำลองตัวอย่าง หรือสัญลักษณ์ของแบบจำลอง ไม่เกี่ยวข้องกับภาพแบบจำลองที่สร้างจริง

ข.5.12. IKRSUnit เป็นตัวต่อประสานที่ใช้จัดการหน่วยที่ใช้สร้างแบบจำลอง โดยจะเปลี่ยนกลับไปกลับมาระหว่างหน่วยที่ใช้สร้างแบบจำลองซึ่งเป็นหน่วยเมตริก (Metric) และหน่วยที่ผู้ใช้กำหนด (User Unit) มีฟังก์ชันใช้งาน และคุณสมบัติดังนี้

ข.5.12.1. คุณสมบัติ FullName เป็นคุณสมบัติที่แสดงชื่อเต็มของหน่วยที่กำหนดด้วยตัวต่อประสานนี้

```
HRESULT FullName (
        BSTR* pFullName
);
```

พารามิเตอร์

pFullName

[out, retval] บัฟเฟอร์รองรับชื่อเต็มของหน่วยที่ใช้

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.12.2. คุณสมบัติ ShortName เป็นคุณสมบัติที่แสดงชื่อย่อของหน่วยที่กำหนดด้วยตัวต่อประสานนี้

```
HRESULT ShortName (
        BSTR* pShortName
);
```

พารามิเตอร์

pShortName

[out, retval] บัฟเฟอร์รองรับชื่อย่อของหน่วยที่ใช้

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.12.3. คุณสมบัติ UnitType เป็นคุณสมบัติที่แสดงชนิดของหน่วยที่ใช้ ซึ่งมี 3 แบบคือ ความยาว มุม และน้ำหนัก

```
HRESULT UnitType (
        KRSUNITTYPE * pType
);
```

พารามิเตอร์

pType

[out, retval] บัฟเฟอร์รองรับชนิดของหน่วยที่ใช้ มีค่าดังตารางที่ ข.6

ตารางที่ ข.6 ตารางค่าตัวแปรชนิด KRSUNITTYPE

KRSUNITTYPE	ความหมายของค่าตัวแปร
KRSUNIT_LENGTH	หน่วยวัดความยาว
KRSUNIT_ANGULAR	หน่วยวัดมุม
KRSUNIT_MASS	หน่วนวัดมวล

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.12.4. ฟังก์ชัน GetSystemUnit ใช้ปรับหน่วยที่ผู้ใช้กำหนดเข้ามาทางส่วนติดต่อกับผู้ใช้ ให้เป็นหน่วยมาตรฐานที่ใช้สร้างแบบจำลองด้วยคำสั่งของพาราโซลิด มีรูปแบบดังนี้

```
HRESULT GetSystemUnit (
        double UserUnit
        double* pSystemUnit
);
```

พารามิเตอร์

UserUnit

[in] ค่าที่กำหนดโดยผู้ใช้ที่ต้องการเปลี่ยนให้อยู่ในหน่วยที่ใช้สร้างแบบจำลอง

pSystemUnit

[out, retval] บัฟเฟอร์รองรับค่าที่ปรับเปลี่ยนให้อยู่ในหน่วยมาตรฐาน

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.12.5. ฟังก์ชัน GetUserUnit ใช้ปรับหน่วยมาตรฐานที่ใช้สร้างแบบจำลองเพื่อแสดงผลในรูปหน่วยที่ผู้ใช้กำหนด มีรูปแบบดังนี้

```
HRESULT GetUserUnit (
        double SystemUnit
        double* pUserUnit
);
```

พารามิเตอร์

SystemUnit

[in] ค่าในหน่วยมาตรฐานที่ต้องการเปลี่ยน

pUserUnit

[out, retval] บัฟเฟอร์รองรับค่าที่ปรับเปลี่ยนให้อยู่ในหน่วยของผู้ใช้

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.13. IKRSFeatureDef เป็นตัวต่อประสานที่ใช้ติดต่อกับข้อมูลชนิดของลักษณะจำเพาะที่นิยามไว้ ข้อมูลเหล่านี้คือ ชื่อชนิดของลักษณะจำเพาะ ชนิดของพารามิเตอร์ทั้งหมดที่ใช้โดยลักษณะจำเพาะชนิดนี้ ตัวต่อประสาน IKRSFeatureDef มีฟังก์ชัน และคุณสมบัติดังนี้

ข.5.13.1. คุณสมบัติ Name เป็นคุณสมบัติที่แสดงชื่อของนิยามลักษณะจำเพาะ มีรูปแบบเรียกใช้ดังนี้

```
HRESULT Name (
        BSTR* pName
);
```

พารามิเตอร์

pName

[out, retval] บัฟเฟอร์รองรับชื่อของนิยามลักษณะจำเพาะ

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.13.2. ฟังก์ชัน AddParameter เป็นฟังก์ชันที่ใช้เพิ่มชนิดของตัวแปรสำหรับการนิยามลักษณะจำเพาะ การเรียกใช้จะต้องเรียกก่อนการเรียกใช้ฟังก์ชัน Register มีรูปแบบดังนี้

```
HRESULT AddParameter (
    BSTR szParamName,
    KRSVARTYPE Type
);
```

พารามิเตอร์

szParamName

[in] ชื่อของตัวแปรที่จะเพิ่ม

Type

[in] ชนิดของตัวแปรที่ต้องการเพิ่ม มีค่าดังตารางที่ ข.7

ตารางที่ ข.7 ตารางค่าตัวแปรชนิด KRSVERTYPE

KRSVARTYPE	ความหมายของค่าตัวแปร
KRSVARTYPE_INT	จำนวนเต็ม
KRSVARTYPE_DOUBLE	จำนวนจริง
KRSVARTYPE_STRING	ข้อความ
KRSVARTYPE_VECTOR	เวกเตอร์
KRSVARTYPE_INTARRAY	อาร์เรย์ของจำนวนเต็ม
KRSVARTYPE_DOUBLEARRAY	อาร์เรย์ของจำนวนจริง
KRSVARTYPE_VECTORARRAY	อาร์เรย์ของเวกเตอร์

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.13.3. ฟังก์ชัน Register เป็นฟังก์ชันใช้ลงทะเบียนนิยามลักษณะจำเพาะที่กำหนด เมื่อเรียกใช้แล้วจะไม่สามารถใช้ฟังก์ชัน AddParameter ได้อีก ก่อนที่นิยามลักษณะจำเพาะจะนำมาใช้ได้นั้นก็ต้องผ่านการเรียกฟังก์ชันนี้ก่อนเสมอ

HRESULT Register ();

ค่าส่งกลับ

S_OK ถ้าการเรียกใช้สำเร็จ และ E_FAIL ถ้าไม่สำเร็จ

ข.5.14. IKRSFeature เป็นตัวต่อประสานที่ทำหน้าที่ติดต่อกับข้อมูลของลักษณะจำเพาะที่ใช้สร้างแบบจำลองสามมิติ ได้แก่ ชื่อชนิดของลักษณะจำเพาะที่นิยามไว้ ชื่อของลักษณะจำเพาะ ข้อมูลพารามิเตอร์ของลักษณะจำเพาะ ลำตัวที่ถูกกระทำด้วยลักษณะจำเพาะ และลักษณะจำเพาะที่กระทำต่อลำตัวเดียวกันนี้ในลำดับก่อนและหลังลักษณะจำเพาะนี้

ข.5.15. IKRSParameter เป็นตัวต่อประสานที่ทำหน้าที่ติดต่อกับข้อมูลพารามิเตอร์ภายในลักษณะจำเพาะที่ใช้สร้างแบบ

ข.5.16. IKRSToolbar เป็นตัวต่อประสานที่ใช้ติดต่อกับแถบเครื่องมือที่สร้างโดยซอฟแวร์ส่วนประกอบเสริม

ข.6. การสร้างแถบเครื่องมือในซียู-โซลิด

การสร้างแถบเครื่องมือคำสั่งให้แสดงในซียู-โซลิดทำได้โดยใช้ฟังก์ชัน IKRSApp::AddToolbar โดยทั่วไปจะเรียกใช้ฟังก์ชันนี้ภายในฟังก์ชัน CXXXMain::StartModule ซึ่งถูกเรียกใช้งานเมื่อซอฟแวร์ส่วนประกอบเสริมถูกโหลดขึ้นใช้งาน การสร้างแถบเครื่องมือในซียู-โซลิดมีขั้นตอนการสร้างดังนี้

ข.6.1. สร้างรูปของปุ่มบนแถบเครื่องมือที่ต้องการโดยใช้ตัวสร้างรีซอส (Resource Editor) ของวิชชวลซีพลัสพลัส

ข.6.2. กำหนดคลาสที่จะรองรับการทำงานของคำสั่งจากแถบเครื่องมือที่จะสร้าง โดยอาจกำหนดให้เป็น CXXXMain CXXXDoc หรือ CXXXView ก็ได้ตามความเหมาะสม

ข.6.3. สร้างฟังก์ชันรองรับการทำงานของคำสั่งจะแถบเครื่องมือในคลาสที่เลือกตามข้อ 6.2 ซึ่งฟังก์ชันที่สร้างนี้จะต้องส่งค่าชนิด HRESULT กลับ

ตัวอย่าง

```
HRESULT CXXXDoc:: OnExtrude()
{
        :
}
```

ข.6.4. เพิ่มรหัสคำสั่งของแถบเครื่องมือลงในฟังก์ชัน OnCmdMsg ในคลาสที่เลือกพร้อมกับเชื่อมต่อไปยังฟังก์ชันรองรับการทำงานของคำสั่งที่สร้างขึ้นตามข้อ 6.3

ตัวอย่าง

```
HRESULT CXXXDoc::OnCmdMsg(UINT nID, void* wparam, void* lpparm )
{
HRESULT hr = S_FALSE;
// either handle the event here...
switch( nID )
        {
case ID_SOLID_EXTRUDE:
OnExtrude();
hr = S_OK;
break;
        default:
break;
         }
        // check if the message has already been handled
        if ( hr == S_FALSE)
        {
                // try routing this to the active view
                CAddonView* pView = GetActiveView();
                if ( pView )
                        hr = pView->OnCmdMsg( nID, lpparm );
        }
return hr;
}
```

ข.6.5. ในคลาส CXXXMain สร้างตัวแปรชนิด CComPtr<IKRSToolbar> เพื่อรองรับแถบเครื่องมือใหม่ที่จะสร้างขึ้น

ตัวอย่าง

```
class CXXXMain {
        :
        :
CComPtr<IKRSToolbar>  m_pToolbar;
}
```

ข.6.6. ในฟังก์ชัน CXXXMain::StartModule เรียกคำสั่งสร้างแถบเครื่องมือ และกำหนดค่าแถบเครื่องมือที่สร้างขึ้นให้กับตัวแปรที่สร้างขึ้นตามข้อ 6.5

ตัวอย่าง

```
HRESULT CXXXMain::StartModule(IKRSAddIn* pAddIn)
{
// Add solid toolbar.
CString csString = _T("Solid");
BSTR bsStr = csString.AllocSysString();
        m_pAppInterface->AddToolbar(bsStr, (long)IDR_SOLID,
                                    (long)theApp.m_hInstance, pAddIn,
                                    &m_ pToolbar);
SysFreeString(bsStr);
}
```

ข.6.7. ในกรณีที่ต้องการสร้างปุ่มดรอพดาวน์สามารถทำได้โดยสร้างปุ่มคำสั่งด้วยตัวสร้างรีซอสของวิชชวลซีพลัสพลัส (ชนิดแถบเครื่องมือ) และเรียกคำสั่งแทรกปุ่มดรอพดาวน์ (InsertDropDownToolBar) จากตัวแปรที่สร้างในข้อ 6.5 ใต้คำสั่ง AddToolbar ในข้อ 6.6

ข.6.8. ในฟังก์ชัน CXXXMain::OnAppDestroy ให้ลบแถบเครื่องมือที่สร้างทิ้งด้วยคำสั่ง IKRSApp::RemoveToolbar

ตัวอย่าง

```
HRESULT CXXXMain::OnAppDestroy()
{
        :
m_pAppInterface->RemoveToolbar(m_ pToolbar);
        :
```

```
return S_OK;
}
```

ข.7. การลงทะเบียนชนิดของแฟ้มที่สามารถเปิด และบันทึกได้

เมื่อต้องการสร้างซอฟแวร์ส่วนประกอบเสริมให้สามารถเปิดอ่าน และบันทึกข้อมูลแบบจำลองโซลิดในรูปแฟ้มข้อมูลชนิดอื่น ๆ นอกเหนือจากแฟ้มข้อมูลของซียู-โซลิด สามารถทำได้โดยการลงทะเบียนชนิดของแฟ้มข้อมูลที่ต้องการติดต่อด้วย ในการลงทะเบียนจะเรียกผ่านฟังก์ชัน IKRSApp::RegisterFileOpenFunction และ IKRSApp:: RegisterFileSaveAsFunction เพื่อลงทะเบียนชนิดของแฟ้มที่จะเปิดอ่าน และบันทึกตามลำดับ ซึ่งการเรียกฟังก์ชันการลงทะเบียนควรเรียกภายในฟังก์ชัน CXXXMain::StartModule อีกทีหนึ่ง

ข.8. วัตถุที่ใช้แสดงภาพ (The Draw Object)

เมื่อซอฟแวร์ส่วนประกอบเสริมต้องการแสดงภาพวัตถุ หรือสัญลักษณะต่าง ๆ นอกเหนือจากภาพแบบจำลองโซลิดที่สร้างขึ้น เช่น เมื่อต้องการสร้างภาพพรีวิว สามารถกระทำผ่านตัวต่อประสาน IKRSDrawObj ซึ่งจะทำหน้าที่รับข้อมูลภาพที่ต้องการวาด แล้วนำไปสร้างเป็นภาพสามมิติบนจอ

ข.8.1. <u>การกำหนดข้อมูลเพื่อนำไปสร้างเป็นภาพสามมิติ</u>

ข.8.1.1. องค์ประกอบทางโทโปโลยีของพาราโซลิด เมื่อกำหนดข้อมูลองค์ประกอบทางโทโปโลยีของพาราโซลิดให้กับ IKRSDrawObj ผ่านทางฟังก์ชัน IKRSDrawObj::AddEntity ข้อมูลนี้จะถูกเปลี่ยนเป็นข้อมูลสำหรับโอเพนจีเอลแล้วจึงนำไปแสดงผลต่อไป

ข.8.1.2. ข้อมูลเส้นตรง เป็นการกำหนดพารามิเตอร์ของเส้นตรงซึ่งประกอบด้วยกลุ่มข้อมูลตำแหน่งของจุดปลายทั้งสองของเส้นตรง และวิธีการเชื่อมต่อระหว่างจุดยอด แบ่งเป็น เส้นตรงที่ใช้ข้อมูล 2 จุด เส้นตรงต่อเนื่อง (Lins Strip) และเส้นตรงต่อเนื่องเป็นวงปิด (Line Loop) การกำหนดข้อมูลเส้นตรงนี้จะกำหนดผ่านทางฟังก์ชัน IKRSDrawObj::AddLines

ข.8.1.3. ข้อมูลที่เป็นพื้นผิวที่ปิดล้อมด้วยสามเหลี่ยม ในการสร้างภาพสามมิติของพื้นผิวต่าง ๆ จะต้องสร้างเป็นพื้นผิวสามเหลี่ยมย่อย ๆ ต่อกัน ซึ่งจะต้องกำหนดตำแหน่งจุดยอดทั้งสามของสามเหลี่ยมพร้อมกับเวกเตอร์ตั้งฉากกับพื้นผิวที่จะสร้าง ณ ตำแหน่งจุดยอดด้วย การกำหนดข้อมูลชนิดนี้จะกระทำผ่านทางฟังก์ชัน IKRSDrawObj:: AddTriangles

ข.8.1.4. ข้อมูลที่เป็นระนาบ เป็นการกำหนดพารามิเตอร์ของระนาบเพื่อสร้างภาพสามมิติของระนาบ โดยเรียกผ่านฟังก์ชัน IKRSDrawObj::AddPlane

ข.8.1.5. ข้อมูลเวกเตอร์ในสามมิติ เป็นการสร้างภาพเวกเตอร์แสดงทิศทางในสามมิติ ในการกำหนดค่าเวกเตอร์จะกระทำผ่านฟังก์ชัน IKRSDrawObj::AddVector

ข.8.2. <u>การกำหนดคุณสมบัติของภาพสามมิติ</u> การกำหนดคุณสมบัตินี้สามารถกำหนดได้สองวิธีคือ การนิยามคุณสมบัติต่าง ๆ เอง หรือเลือกจากค่าคุณสมบัติมาตรฐานที่นิยามขึ้นโดยซียู-โซลิดกก็ได้ ซึ่งคุณสมบัติมาตรฐานนี้จะประกอบไปด้วย สี ชนิดของเส้น ขนาดของเส้น และจุด ในการกำหนดคุณสมบัติด้วยคุณสมบัติมาตรฐานจะกระทำผ่านฟังก์ชัน IKRSDrawObj:: SetRenderDataSet ด้วยพารามิเตอร์ชนิดของคุณสมบัติมาตรฐาน ดังตารางที่ ข.8

ตาราง ข.8 ตารางค่าตัวแปรชนิด KRSRENDERTYPE

รหัสชนิดคุณสมบัติ	คำอธิบาย
KRSRENDER_DEFINE	เป็นการกำหนดให้อ่านค่าพารามิเตอร์จากแอทริบิวต์ของพาราโซลิดที่นิยามติดไปกับองค์ประกอบทางโทโปโลยีที่ใช้เป็นข้อมูลในการสร้างภาพสามมิติ
KRSRENDER_DEFAULT	เป็นคุณสมบัติใช้กับองค์ประกอบที่ไม่ได้นิยามแอททริบิวต์สีและการสะท้อน
KRSRENDER_EDGE	เป็นคุณสมบัติที่ใช้แสดงขอบในการแสดงผลแบบมีแสงเงาพร้อมกับแสดงขอบ (Shade with Edge)
KRSRENDER_HOT	เป็นคุณสมบัติที่ใช้กับองค์ประกอบที่ถูกชี้ด้วยเมาส์
KRSRENDER_SEL	เป็นคุณสมบัติที่ใช้กับองค์ประกอบที่ถูกเลือก
KRSRENDER_REFERENCE	เป็นคุณสมบัติที่ใช้กับระนาบอ้างอิง
KRSRENDER_SKETCH	เป็นคุณสมบัติที่ใช้กับแบบร่างบนระนาบ
KRSRENDER_SKESEG	เป็นคุณสมบัติที่ใช้กับเซกเมนต์ของแบบร่างบนระนาบในโหมดการสร้างแบบร่างบนระนาบ
KRSRENDER_PREVIEW	เป็นคุณสมบัติที่ใช้พรีวิวแบบจำลองที่กำลังสร้าง
KRSRENDER_ERROR	เป็นคุณสมบัติที่ใช้กับองค์ประกอบที่เกิดข้อผิดพลาด
KRSRENDER_HANDLE	เป็นคุณสมบัติที่ใช้กับสัญลักษณะต่าง ๆ

ถ้าต้องการนิยามคุณสมบัติต่าง ๆ เองก็สามารถทำได้ซึ่งคุณสมบัติเหล่าที่จะกำหนดนี้แบ่งย่อยออกเป็น 3 ชนิดคือ

ข.8.2.1. การกำหนดสี การนิยามคุณสมบัติชนิดสี ผู้เขียนโปรแกรมต้องกำหนดค่า สีแดง เขียว นำเงิน และอัลฟา (RGBA) ผ่านคุณสมบัติ IKRSDrawObj:: RenderColor

ข.8.2.2. การกำหนดชนิดและขนาดของเส้น เป็นการกำหนดชนิดของเส้น เช่น เส้นต่อเนื่อง เส้นประ โดยกำหนดเป็นรหัสของเลขฐานสิบหก และนำไปแปลงเป็นเลขฐานสองเพื่อนำไปสร้างเป็นรหัสสำหรับสร้างเส้น โดยตัวเลขศูนย์แทนช่องว่างและหนึ่งแทนเส้น ขนาดของเส้นจะถูกกำหนดเป็นจำนวนจริงที่มากกว่าศูนย์ การกำหนดค่าเหล่านี้จะกำหนดผ่านคุณสมบัติ IKRSDrawObj:: LineStyle

ข.8.2.3. การกำหนดขนาดจุด จะกำหนดเป็นจำนวนจริง ผ่านคุณสมบัติ IKRSDrawObj:: PointSize

ประวัติผู้เขียนวิทยานิพนธ์

นายกฤษณะ อุตมัง เกิดเมื่อวันที่ 22 กรกฎาคม พ.ศ.2520 ที่อำเภอเมือง นครราชสีมา จังหวัดนครราชสีมา สำเร็จการศึกษาระดับปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมเครื่องกล จากจุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2540 และได้เข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมเครื่องกล จุฬาลงกรณ์มหาวิทยาลัย ในปี พ.ศ.2541