บทที่ 8

VRML

8.1 ที่มาและวัตถุประสงค์ของภาษา VRML

VRML ย่อมาจาก Virtual Reality Modeling Language(Virtual Reality World) หรือเรียกว่า "เวอร์มอล" ซึ่งเป็นภาษาคอมพิวเตอร์ที่ใช้ในการสร้างรูปเสมือนจริงเป็นรูปภาพกราฟิก 3 มิติ ประกอบความสามารถในการโต้ตอบกับผู้ใช้ทันที(Real-time interactive)ที่ผ่านทางระบบเบราเซอร์ของระบบเวิลด์ไวด์เว็บ (World Wide Web) ของเครือข่ายอินเตอร์เน็ตเสมือนกับว่าผู้ใช้เข้าไปอยู่ในระบบ 3 มิตินั้นจริงๆ นอกจากความสามารถทางด้านการสร้างกราฟิก 3 มิติ แล้วยังสามารถนำเสนอด้วยสื่อมัลติมีเดียเพื่อเพิ่มความสมจริงมากยิ่งขึ้น เช่น ระบบเสียงที่เป็นลักษณะ 3 มิติ ภาพเคลื่อนไหว ซึ่งสามารถโต้ตอบการเปลี่ยนแปลงมุมมองของผู้ใช้ได้ในเวลาจริง (Real-time Interaction) โดยผ่านการรับรู้และเปลี่ยนแปลงมุมต่างๆ

ภาษา VRML อาศัยหลักการแสดงผลกราฟิกทั้ง 2 มิติ และ 3 มิติ โดยอาศัยวิธีการแบบ "โอเพ่นจีแอล(OpenGL)" ซึ่งเป็นวิธีการแสดงผลของบริษัท Silicon Graphic ที่ถูกนำเสนอในปี ค.ศ. 1992 ใช้สำหรับการแสดงค่าของวัตถุต่างๆ ไม่ว่าจะเป็นตำแหน่ง (Coordinate) รายละเอียดพื้นผิว (Texture) จุดเด่นของ โอเพ่นจีแอล คือไม่ขึ้นอยู่กับระบบปฏิบัติการ จึงทำให้ผู้พัฒนาโปรแกรมไม่จำเป็นต้องคำนึงถึงว่าจะใช้เครื่องชนิดใดและระบบปฏิบัติการใด ในการสร้างภาพนั้น โอเพ่นจีแอล จะทำการสร้างภาพวัตถุโดยสร้างรูปร่างพื้นฐานของวัตถุก่อน และเก็บใน เฟรมบัฟเฟอร์ (Frame Buffer) ภาพวัตถุพื้นฐานในระบบคอมพิวเตอร์ คือ จุด เส้นตรง ภาพหลายหลายเหลี่ยมและบิตแมป (Bitmap) ภาพของวัตถุที่สร้างขึ้นนั้นส่วนประกอบแต่ละส่วนจะเป็นอิสระต่อกัน ดังนั้นในการเปลี่ยนแปลงลักษณะของวัตถุหรือภาพจะไม่ต้องกระทำทั้งวัตถุ เพียงแต่กระทำเฉพาะส่วนที่ต้องการเท่านั้น ไม่จำเป็นต้องเปลี่ยนแปลงทั้งวัตถุ เมื่อภาพของวัตถุที่แยกเป็นรูปพื้นฐานถูกเก็บในเฟรมบัฟเฟอร์ แล้วการเปลี่ยนแปลงใดๆ ในเรื่องของคุณสมบัติของวัตถุจะถูกควบคุมโดยโอเพ่นจีแอล โดยตรงไม่ว่าจะเป็นการเปลี่ยนแปลงตำแหน่ง ขนาด สี หรือการกำหนดค่าความเข้มของแสงต่างๆ ส่วนการของการนำผลของค่าที่อยู่ในเฟรมบัฟเฟอร์ มาแสดงเป็นหน้าที่ของวินโดวส์ว่าจะแสดงค่าใดในจอภาพ

ภาษา VRML มี ลักษณะเด่นในการแสดงวัตถุทั้งคงที่และเคลื่อนไหวและยังสามารถทำงานร่วมกับมัลติมีเดียอื่นๆ เช่น เสียง (Voice), ภาพ (Image), ภาพยนตร์(Movies)โดยผ่านตัวประมวลผลคือ เบราเซอร์ (Browser) นอกจากนั้นยังสนับสนุนลักษณะ 3 มิติ แบบ แอพพริเคชันโปรแกรมมิ่งอินเตอร์เฟซ (Application Programming Interface, API) อีกด้วย ภาษา VRML ได้รับการรองรับจาก ISO (International Organization for Standardization) และ IEC (International Electronic Commission) ซึ่งเป็นหน่วยงานที่ทำหน้าที่ควบคุมมาตรฐานต่างๆ ของอินเตอร์เน็ต โดยอนุญาตมาตรฐาน ISO/IEC 14772

ภายใต้หัวข้อของ Information Technology-Computer graphics and Image Processing-vertebral Reality Modeling Language (VRML)

ภาษา VRML ทำงานภายใต้แบบอย่างพื้นฐานของ เว็บเบราเซอร์-เซิร์ฟเวอร์ (Web Browser-server)ทั่วไป โดยอาศัยรูปแบบยูอาแอล (URL) ซึ่งเป็นรูปแบบมาตรฐานสำหรับระบบเวิลด์ไวด์เว็บ โดยกำหนดให้ขึ้นต้นด้วย HTTP (Hypertext Transfer Protocol) โดยเซิร์ฟเวอร์ ภาษา VRML จึงถือว่าเป็นเครื่องมือในการสร้างโลกเสมือนจริงแบบใหม่ที่เรียกว่า “สังคมไร้พรมแดน”(Cyberspace) และ “ สังคมเสมือนจริง”(On-line Virtual Communications) ขึ้นมาจำลองสังคมมนุษย์ในโลกแห่งความจริงมาไว้ในโลกของคอมพิวเตอร์ 3 มิติ ทำหน้าที่แปลคำร้องของเบราเซอร์ ขณะดาวโหลด (download) เซิร์ฟเวอร์ทำการส่งเอกสารที่เป็น แทก (Tag) ของเอกสารหรือเรียกว่า มัลติเพอร์เพิส อินเตอร์เน็ต เมล เอกซ์เทนชัน (Multipurpose Internet Mail Extensions,MIME) ซึ่งภาษา VRML มีลักษณะเป็น cross-world/cross-vrml (x-world/x-vrml)โดยผู้ใช้สามารถดูด้วยเบราเซอร์ที่เรียกว่า วีอาเอ็มแอล เบราเซอร์ (VRML Browser) ได้โดยอาศัยไฟล์ข้อมูลในรูปแบบ VRML (*.wrl) ซึ่งเป็นรูปแบบกลาง สำหรับแลกเปลี่ยนข้อมูล 3 มิติ อาศัยการนำเสนอวัตถุ (Object) เป็นแบบเนสต์ (Nesting) โดยส่งข้อมูลรูปแบบทั้งหมดมาก่อนและตามด้วยระดับความละเอียดภายหลังอาศัยหลักการ แอลโอดี (LOD,Level Of Detail) เปลี่ยนแปลงไปมาโดยอัตโนมัติและทำการเรนเดอร์ (Render) เพื่อสร้างแบบจำลองกราฟิก 3 มิติ ที่วีอาเอ็มแอล เบราเซอร์ ส่วนเอกสารที่เป็นเสียงหรือวีดีโอจะถูกส่งมาตามลำดับ

ลักษณะเด่นของภาษา VRML อาจเรียกได้ว่าภาษา VRML มีลักษณะเป็นโฮมเพจ 3 มิติ (3D-Homepage) มีลักษณะภาษาคอมพิวเตอร์ชนิดง่ายๆ ไม่ขึ้นอยู่กับระบบคอมพิวเตอร์ใดๆ พร้อมทั้งตีพิมพ์เอกสารโฮมเพจที่เป็น 3 มิติ บนบริการเวิลด์ ไวด์ เว็บ (World Wide Web) ของเครือข่ายอินเตอร์เน็ต รวมทั้งการนำเสนอข้อมูลจะเกิดประโยชน์สูงสุดเมื่อสามารถเคลื่อนไหวเปลี่ยนมุมมองได้ ผู้เข้าชมสามารถมีส่วนร่วมในการโต้ตอบกับวัตถุที่อยู่ในฉากได้ ถือได้ว่าภาษา VRML เป็นเทคโนโลยีการนำเสนอข้อมูลหลายรูปแบบทั้งกราฟิก 2 มิติ และ 3 มิติ รวมทั้งข้อมูลมัลติมีเดีย เช่น ภาพนิ่ง ภาพเคลื่อนไหว และระบบเสียง 3 มิติ ทำให้เกิดระบบการติดต่อกับผู้ใช้ (User Interface) ในรูปแบบใหม่ขึ้นมาโดยระบบ 2 มิติ ที่ใช้อยู่ปัจจุบันกลายเป็นส่วนประกอบย่อยใน 3 มิติ

8.2 ข้อแตกต่างของภาษา VRML เวอร์ชัน 1.0 และ 2.0

ภาษา VRML ได้มีการพัฒนาจาก เวอร์ชัน 1.0 และมาเป็น 2.0 ในปัจจุบัน ซึ่งมีข้อแตกต่างอย่างชัดเจน 2 ส่วนคือ ในส่วนของการโต้ตอบกับผู้ใช้ (Interactive) กับส่วนของความสมจริง (Realistic) ซึ่งภาษา VRML เวอร์ชัน 1.0 ไม่มีคุณสมบัติดังกล่าวคือ ไม่มีการโต้ตอบและรูปทรงจะคงที่ไม่มีการเคลื่อนไหว นอกจากนั้นภาษา VRML เวอร์ชัน 2.0 สามารถทำงานร่วมกันกับภาษา Java และ Javascript รวมทั้งมีส่วนของเสียง 3 มิติ และภาพเคลื่อนไหวเข้ามาเพิ่มความสมจริงยิ่งขึ้น รวมถึงการเปลี่ยนแปลงชื่อโหนดดังตารางต่อไปนี้

ชื่อโหนด VRML เวอร์ชัน 1.0	ชื่อโหนด VRML เวอร์ชัน 2.0
AsciiText	Text
Coordinate3	Coordinate
Cube	Box
Info	WorldInfo
MatrixTransform	Transform
OrthogrphicCamera	Viewpoint
PerspectiveCamera	Viewpoint
Rotation	Transform
Scale	Transform
Separator	Transform
Texture2	Texture
Texture2Transform	TextureTransform
TextureCoordinate2	TextureCoordinate
Translation	Transform
WWWAnchor	Anchor
WWWInline	Inline

ตารางที่ 8.1 การเปรียบเทียบชื่อโหนดของเวอร์ชัน 1.0 กับเวอร์ชัน 2.0

8.3 ภาษา VRML และ ภาษา HTML

ภาษา VRML และ ภาษา HTML มีส่วนการสร้างขึ้นมาคล้ายกันคือการเขียนโปรแกรมขึ้นมาในลักษณะของเท็กซ์โหมด โดยใช้เท็กซ์อิดิเตอร์ เป็นเครื่องมือพื้นฐานแล้วจึงใช้เบราเซอร์เป็นตัวอินเตอร์พรีเตอร์ (Interpreter) ในทางกลับกันลักษณะที่แตกต่างกันชัดเจนคือ การประมวลผลออกมา ภาษา VRML เป็นเว็บ 3 มิติ ส่วนภาษา HTML เป็นเว็บ 2 มิติ จากคุณสมบัติข้างต้น ทำให้ภาษา VRML มีขนาดของไฟล์ใหญ่กว่าภาษา HTML

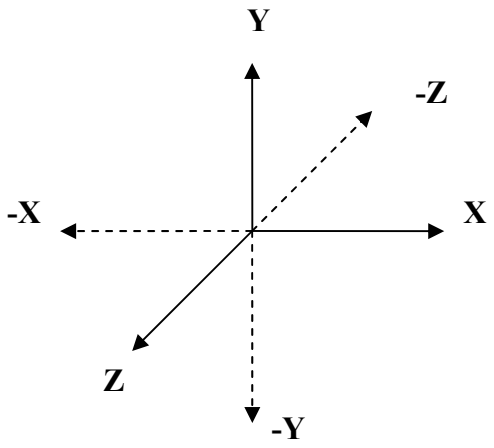
คุณสมบัติ	ภาษา HTML	ภาษา VRML
ลักษณะพื้นฐานโครงสร้างภาษา	Text mode	Text mode
ลักษณะการทำงาน	Web Browser-server	Web Browser-server
แกน	X,Y	X,Y,Z
การประมวลกราฟิกหรือเรนเดอร์	2 มิติ	3 มิติ
ขนาดไฟล์	เล็ก	ใหญ่
ชนิดของนามสกุล	.htm	.wrl

ตารางที่ 8.2 การเปรียบเทียบของภาษา VRML กับภาษา HTML

8.4 องค์ประกอบของภาษา VRML

8.4.1 ระบบแกน 3 มิติ

ภาษา VRML มีการทำงานภายใต้ระบบแกน 3 มิติ ซึ่งจะมีการใช้กฎมือขวาดังนี้



รูปที่ 8.1 ระบบแกน 3 มิติ

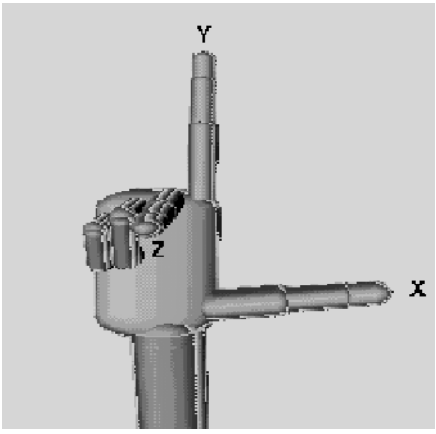
แกนประกอบไปด้วย 3 แกนคือแกน X,Y,Z โดยที่

แกน X มีทิศทางไปด้านขวาจากจุดกำเนิด (origin)

แกน Y มีทิศทางไปด้านบนจากจุดกำเนิด (origin)

แกน Z มีทิศทางตั้งฉากกับแกน X และแกน Y (มีทิศพุ่งออกมาทางนอกฉาก) ซึ่งภายในแต่ละแกนจะสามารถหมุนรอบแกนมันเอง

ในแต่ละแกนยังประกอบไปด้วยค่าบวกและค่าลบ ถ้ามีค่าเป็นบวกจะอยู่ด้านขวา (แกน X) และด้านบน (แกน Y) และด้านหน้า (แกน Z) เช่น พิกัด 1 0 1 แสดงว่าไปทางขวา 1 หน่วย และไปด้านหน้า 1 หน่วยในแกน Z ส่วนค่าลบจะมีทิศทางตรงข้ามกับค่าบวก รวมทั้งยังสามารถหมุนพร้อมกันได้โดยอาศัยการหมุนรอบจุดกำเนิด



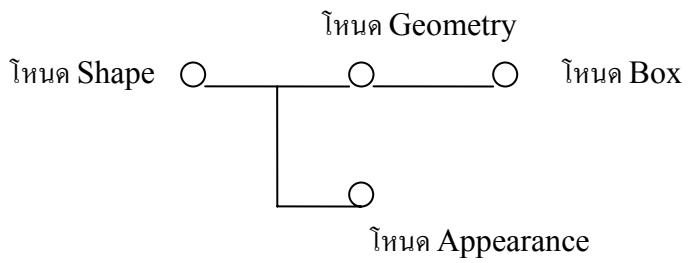
รูปที่ 8.2 การใช้กฎมือขวา

8.4.2 องค์ประกอบพื้นฐานในไฟล์ VRML

การสร้างวัตถุในโลกเสมือนจริงของภาษา VRML เกิดจากองค์ประกอบพื้นฐานคือ

- โหนด (node)
- ฟิลด์ (field)

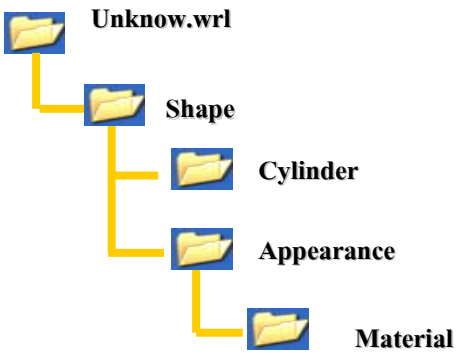
8.4.2.1 โหนด (node) คือหน่วยพื้นฐานในไฟล์ของภาษา VRML ทำหน้าที่เก็บค่าคุณสมบัติพื้นฐานต่างๆ ของฟิลด์ เช่น โหนด Shape, โหนด Timesensor เป็นต้น ซึ่งโหนดมีโครงสร้างอิสระ สามารถกำหนดโหนดเดียวภายในไฟล์ได้ แต่ฟิลด์ไม่สามารถกำหนดเองได้ ต้องอาศัยโหนดภายในไฟล์ สามารถกำหนดโหนดอื่นได้ (.ในแต่ละโหนดจะมีฟิลด์ที่ใช้เก็บคุณสมบัติต่างๆ) เช่น โหนด Shape จะประกอบไปด้วย โหนด Geometry และโหนด Appearance ดังรูปที่ 8.2



รูปที่ 8.3 ตัวอย่างโครงสร้างของโหนด

ภายในโหนด Geometry ยังสามารถแตกโหนดย่อยลงไปอีกได้ หรือเรียกว่า “โหนดลูก (children node)” เช่น โหนด Box และโหนด Appearance เป็นโหนดลูกของโหนด Shape เป็นต้น

8.4.2.2 ฟิลด์ (field) คือหน่วยพื้นฐานที่เล็กที่สุดของไฟล์ภาษา VRML โดยทำหน้าที่เก็บค่าคุณสมบัติของโหนดนั้นๆ เช่น โหนด Box ก็จะมีฟิลด์รอรับค่า ความกว้าง ความยาว ความสูง ภายในนั้นเอง จึงอาจสรุปได้ว่าหลายๆฟิลด์ประกอบกันเป็นโหนดและในทำนองเดียวกันหลายๆโหนดก็จะรวมกันเป็นวัตถุขึ้นมาชิ้นหนึ่ง



รูปที่ 8.4 ตัวอย่างวัตถุที่เกิดจากการนำโหนดหลายๆโหนดมาประกอบกัน

ฟิลด์ในภาษา VRML จะมีส่วนคล้ายกับตัวแปลของภาษาจาวา คือมี 2 ชนิดดังนี้

1. ชนิดซิงเกิล (single) เป็นตัวแปรชนิดซิงเกิล
2. ชนิดมัลติ (multi) เป็นตัวแปรอาร์เรย์ของตัวแปรชนิดซิงเกิล

จะเห็นได้ว่าโหนดเป็นเสมือนคลาสของภาษาจาวา ส่วนฟิลด์คล้ายตัวแปรซึ่งเก็บค่าคุณสมบัติต่างๆไว้

8.5 การสร้างไฟล์ VRML

การสร้างไฟล์ VRML นั้นหลักๆ จะมีอยู่ 2 แนวทางด้วยกันคือ

1. การเขียนโค้ดโดยใช้เท็กซ์อิดิเตอร์ ทั่วไป หรืออาจเป็น เท็กซ์อิดิเตอร์ สำหรับเขียนโค้ดภาษา VRML โดยเฉพาะ ซึ่งการสร้างด้วยวิธีนี้จะต้องเข้าใจโครงสร้างของภาษา และคุณสมบัติของโหนดและฟิลด์ต่างๆ ที่ใช้ในภาษาเป็นอย่างดี

2. การใช้ โปรแกรมสร้างโมเดล 3 มิติ เช่น 3D-Max หรือโปรแกรมอื่นๆที่สามารถสร้างโมเดล3 มิติได้ จากนั้นจึงทำการแปลงรูปแบบไฟล์ให้อยู่ในรูปแบบของไฟล์ VRML (*.wrl) โดยอาจเป็นเป็นการแปลงโดยใช้โปรแกรมที่สร้างโมเดลนั้น หรือใช้โปรแกรมที่ใช้สำหรับการแปลงรูปแบบของไฟล์ ข้อดีของการสร้างไฟล์VRML ด้วยวิธีนี้ก็คือทำได้ง่ายกว่าการเขียนโค้ดด้วยตัวเองโดยใช้เท็กซ์อิดิเตอร์และเหมาะสำหรับผู้ที่มีความรู้ทางด้านการใช้โปรแกรมสร้างโมเดล 3 มิติ แต่จะมีข้อเสียคือ ขนาดของไฟล์ที่ได้จะมีขนาดใหญ่มากเนื่องจากโครงสร้างของไฟล์นั้นโปรแกรมจะเป็นตัวกำหนด และโครงสร้างภาษาที่ออกมาจะทำความเข้าใจได้ยากเพราะมีความซับซ้อน

8.6 การจัดเตรียมองค์ประกอบเพื่อสนับสนุนการทำงานของภาษา VRML

การจัดเตรียมองค์ประกอบเพื่อสนับสนุนการทำงานของภาษา VRML นั้นจะต้องเตรียมทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์ เพื่อให้ VRML สามารถทำงานได้ ซึ่งประกอบด้วย

8.6.1 การเตรียมเบราเซอร์ของภาษา VRML

องค์ประกอบหนึ่งที่สำคัญของเบราเซอร์ซึ่งจะทำหน้าที่ในการเรนเดอร์หรือประมวลผลกราฟิก (rendering) แบบจำลองกราฟิก 3 มิติ ดังนั้นจึงจำเป็นต้องมีเบราเซอร์ที่สามารถสนับสนุนภาษา VRML หรือเรียกว่า VRML เบราเซอร์ ขึ้นเพื่อรองรับการทำงานของภาษา VRML ซึ่งซอฟต์แวร์เหล่านั้นได้แก่

- ระบบปฏิบัติการ Windows 95 ขึ้นไป
- เบราเซอร์ คือโปรแกรม Netscape Navigator version 3.01 ขึ้นไป หรือโปรแกรม Internet Explorer version 4.0 ขึ้นไป
- โปรแกรมเสริมปลักอิน(plug-in) ที่สามารถสนับสนุนภาษา VRML

8.6.2 การเตรียมทางด้านฮาร์ดแวร์

อุปกรณ์ฮาร์ดแวร์ที่สนับสนุนการประมวลผลของภาษา VRML ซึ่งควรคุณสมบัติอย่างต่ำดังต่อไปนี้

- ไมโครโปรเซสเซอร์ เพนเทียม 133 เมกะเฮิรตซ์ขึ้นไป
- หน่วยความจำหลัก (RAM) อย่างน้อย 32 เมกะไบต์

- การ์ดแสดงผลแบบ VGA หรือ SuperVGA
- แรมแสดงผล 2 เมกะไบต์ขึ้นไป จำนวนสี 256 สีขึ้นไป หรือชนิดที่สนับสนุนการทำงานลักษณะ 3 มิติ

8.7 สรุปลักษณะเด่นของภาษา VRML

ลักษณะเด่นๆ ของภาษา VRML อาจสรุปได้ดังนี้

- สร้างแบบจำลองกราฟิก 3 มิติ (3D Graphic model)
- สร้างการโต้ตอบกับผู้ใช้ทันที (Real-time Interactive)
- สร้าง แสง เสียงในระบบ 3 มิติ(Light and Sound 3D)
- สร้างภาพเคลื่อนไหว (Animation)
- มุมมองในการชมแบบจำลอง 3 มิติ ได้แก่ การเดิน (Walk) การหมุน (Rotate) การบิน (Fly)

8.8 รูปแบบไวยากรณ์โหนดต่างๆ (Syntex)

โหมดที่ใช้ในการรวมกลุ่มวัตถุ (Grouping node)

โหนด Transform

```
Transform {
        eventIn         MFNode          addChildren
        eventIn         MFNode          removeChildren
        exposedField    SFVec3f         center          0 0 0
        exposedField    MFNode          children        [ ]
        exposedField    SFRotation      rotation        0 0 1 0
        exposedField    SFVec3f         scale           1 1 1
        exposedField    SFRotation      scaleOrientation 0 0 1 0
        exposedField    SFVec3f         translation     0 0 0
        field           SFVec3f         bboxCenter      0 0 0
        field           SFVec3f         bboxSize        -1 -1 -1
        }
```

โหนดInline

```
Inline {
        exposedField    MFString        url             [ ]
        field           SFVec3f         bboxCenter      0 0 0
        field           SFVec3f         bboxSize        -1 -1 -1
        }
```

โหนด LOD

LOD {

exposedField	MFNode	leve	[ ]
field	SFVec3f	center	0 0 0
field	SFVec3f	range	[ ]

}

โหนด Switch

Switch {

exposedField	MFNode	level	[ ]
exposedField	SFInt32	whichChoice	-1

}

โหมดเกี่ยวกับการสร้างแสงชนิดต่างๆ ให้แก่วัตถุ

โหนด Directionalight

Directionalight {

exposedField	SFFloat	ambientIntensity	0
exposedField	SFColor	color	1 1 1
exposedField	SFVec3f	direction	0 0 -1
exposedField	SFFloat	intensity	1
exposedField	SFBool	on	TRUE

}

โหนด Positionlight

Positionlight {

exposedField	SFFloat	ambientIntensity	0
exposedField	SFVec3f	attenuation	1 0 0
exposedField	SFColor	color	1 1 1
exposedField	SFFloat	intensity	1
exposedField	SFBool	location	0 0 0
exposedField	SFBool	on	TRUE
exposedField	SFFloat	radius	1 0 0

}

โหนด SpotLight

SpotLight {

exposedField	SFFloat	ambientIntensity	0
exposedField	SFVec3f	attenuation	1 0 0
exposedField	SFFloat	beamWidth	1.570796
exposedField	SFColor	color	1 1 1
exposedField	SFFloat	cutoffangle	0.785398
exposedField	SFVec3f	direction	0 0 -1
exposedField	SFFloat	intensity	1
exposedField	SFVec3f	location	0 0 0
exposedField	SFBool	on	TRUE
exposedField	SFFloat	radius	1 0 0

}

โหมดเกี่ยวกับการสร้างรูปทรง

โหนด Shape

Shape {

exposedField	SFNode	appearance	NULL
exposedField	SFNode	geometry	NULL

}

โหนด Box

Box {

field	SFVec3fsize	2 2 2

}

โหนด Cone

Cone {

field	SFFloat	bottomRadius	1
field	SFFloat	height	2
field	SFBool	side	TRUE
field	SFBool	bottom	TRUE

}

โหนด ElevationGrid

ElevationGrid {

exposedField	SFNode	color	NULL
exposedField	SFNode	normal	NULL
exposedField	SFNode	textCoord	NULL
field	MFFloat	height	[ ]
field	SFBool	ccw	TRUE
field	SFBool	colorPerVertex	TRUE
field	SFFloat	creaseAngle	0
field	SFBool	normalPerVertex	TRUE
field	SFBool	solid	TRUE
field	SFint32	xDimension	0
field	SFFloat	xSpacing	0.0
field	SFint32	zDimension	0
field	SFFloat	zPacing	0.0

}

เหตุการณ์ (Event)

event	MFFloat	set_height

โหนด Extrusion

Extrusion {

field	SFBool	beginCap	TRUE
field	SFBool	CCW	TRUE
field	SFBool	convex	TRUE
field	SFFloat	creaseAngle	0
field	SFVec2f	crossSection	[11,1 -1,-1 -1,-1 1, 11]
field	SFBool	endCap endCap	TRUE
field	SFRotation	orientation	0010
field	SFVec2f	creaseAngle	0
field	SFBool	scale	1 1
field	SFBool	solid	TRUE
field	SFVec3f	spine	[0 0 0,0 1 0]

}

เหตุการณ์ (Event)

eventIn	MFVec2f	set_ceossSection
eventIn	MFRotation	set_orientation
eventIn	MFVec2f	set_scale
eventIn	MFVec3f	set_spine

โหนด IndexedFaceSet

IndexedFaceSet {

exposedField	SFNode	color	NULL
exposedField	SFNode	coord	NULL
exposedField	SFNode	convex	NULL
exposedField	SFNode	creaseAngle	NULL
field	SFBool	CCW	TRUE
field	SFInt32	colorIndex	[ ]
field	SFBool	colorPerVertex	TRUE
field	SFBool	convex	TRUE
field	SFInt32	coordIndex	[ ]
field	SFFloat	creaseAngle	0
field	SFInt32	normalIndex	[ ]
field	SFBool	normalPerVertex	TRUE
field	SFBool	solid	TRUE
field	SFInt32	texCoordIndex	[ ]

}

เหตุการณ์ (Event)

eventIn	MFInt32	set_colorIndex
eventIn	MFInt32	set_coordIndex
eventIn	MFInt32	set_normalIndex
eventIn	MFInt32	set_textCoordIndex.

โหนด IndexedLineSet

IndexedLineSet {

eventIn	MFInt32	set_colorIndex	
eventIn	MFInt32	set_coordIndex	
exposedField	SFNode	color	NULL

exposedField	SFNode	coord	NULL
field	MFInt32	colorIndex	[ ]
field	SFBool	colorPerVertex	TRUE
field	MFInt32	colorIndex	[ ]

}

โหนด Pointset

Pointset {

exposedField	SFNode	color	NULL
exposedField	SFNode	coord	NULL

}

โหนด Sphere

Sphere {

Field	SFFloat	radius	1

}

โหนด Text

Text {

exposedField	MFString	string	[ ]
exposedField	SFNode	coord	NULL
exposedField	MFFloat	length	[ ]
exposedField	SFFloat	maxExtent	0.0

}

โหนด Color

Color {

exposedField	MFColor	color	[ ]

}

โหนด Coordinate

Coordinate {

exposedField	MFVec3t	point	[ ]

}

โหนด Normal

Normal {

exposedField	MFVec3f	vector	[ ]

}

โหมดเกี่ยวกับการสร้างเสียง

โหนด Sound

Sound {

exposedField	SFVec3f	direction	0 0 1
exposedField	SFFloat	Intensity	1
exposedField	SFVec3f	location	0 0 0
exposedField	SFFloat	maxBack	1 0
exposedField	SFFloat	maxFront	1 0
exposedField	SFFloat	maxBack	1
exposedField	SFFloat	minFront	1
exposedField	SFFloat	priority	0
exposedField	SFNode	source	NULL
field	SFBool	spatialize	TRUE

}

โหนด Audioclip

Audioclip {

exposedField	SFString	description	“ “
exposedField	SFBool	loop	FALSE
exposedField	SFFloat	pitch	1.0
exposedField	SFTime	startTime	0
exposedField	SFTime	stopTime	0
exposedField	MFString	url	[ ]
exposedField	SFTime	duration_changed	
exposedField	SFBool	isActive	

}

โหมดที่ทำงานร่วมกับสคริปต์

โหนด Script

Script {

exposedField	MFString	url	[ ]
field	SFBool	directOutput	FALSE
field	SFBool	mustEvaluate	FALSE
field	fieldTypename	fieldname initialValue	

}

เหตุการณ์ (Event)

eventIn	eventTypeName	eventname
eventIn	eventTypeName	eventname

โหมดเกี่ยวกับการสร้างการโต้ตอบโดยอาศัยตัวตรวจจับ (Sensors)

โหนด CylinderSensor

CylinderSensor {

exposedField	SFFloat	diskAngle	0.262
exposedField	SFBool	enabled	TRUE
exposedField	SFFloat	maxAngle	-1
exposedField	SFFloat	minAngle	0
exposedField	SFRotation	offset	0 1 0 0
exposedField	SFBool	autoOffset	TRUE

}

เหตุการณ์ (Event)

eventOut	SFBool	isActive
eventOut	SFRotation	rotation_changed
eventOut	SFVec3f	trackPoint_changed

โหนด PlaneSensor

PlaneSensor {

exposedField	SFBool	enabled	TRUE
exposedField	SFVec2f	axPosition	-1 -1
exposedField	SFFloat	inPosition	0 0
exposedField	SFFloat	offset	0 0 0
exposedField	SFRotation	autoOffset	0 1 0 0

}

เหตุการณ์ (Event)

eventOut	SFBool	isActive
eventOut	SFVec3f	rackPoint_changed
eventOut	SFVec3f	ranslation_changed

โหนด ProximitySensor

ProximitySensor {

exposedField	SFVec3f	Center	0 0 0
exposedField	SFVec3f	size	0 0 0
exposedField	SFBool	enabled	TRUE

}

เหตุการณ์ (Event)

eventOut	SFBool	isActive
eventOut	SFVec3f	Position_changed
eventOut	SFTime	enterTime
eventOut	SFTime	exitTime

โหนด SphereSensor

SphereSensor {

exposedField	SFBool	enabled	TRUE
exposedField	SFRotation	offset	0 1 0 0
exposedField	SFBool	autoOffset	TRUE

}

เหตุการณ์ (Event)

eventOut	SFBool	isActive
eventOut	SFRotation	rotation_changed
eventOut	SFTime	trackPoint_ changed

โหนด TimeSensor

TimeSensor {

exposedField	SFTime	cycleInterval	1
exposedField	SFBool	offset	TRUE
exposedField	SFBool	loop	FALSE

exposedField	SFTime	startTime	0
exposedField	SFTime	stopTime	0

}

เหตุการณ์ (Event)

eventOut	SFTime	isActive
eventOut	SFFloat	fraction_changed
eventOut	SFTime	time

โหนด TouchSensor

TouchSensor {

exposedField	SFBool	enabled	TRUE
eventOut	SFVec3f	hitNormal_changed	
eventOut	SFVec3f	hitPoint_changed	

}

เหตุการณ์ (Event)

eventOut	SFVec2f	hitTextCoord_changed
eventOut	SFBool	isActive
eventOut	SFBool	isOver
eventOut	SFTime	touchTime

โหนด VisibilitySensor

VisibilitySensor {

exposedField	SFVec3f	center	0 0 0
eventOut	SFBool	enabled	TRUE
eventOut	SFVec3f	size	0 0 0

}

เหตุการณ์ (Event)

eventOut	SFTime	enterTime
eventOut	SFTime	exitTime
eventOut	SFBool	isActive

โหนดเกี่ยวกับการกำหนดคุณสมบัติพื้นผิวของวัตถุ (Apperance)

โหนด Appearance

Appearance {

exposedField	SFNode	material	NULL
exposedField	SFNode	texture	NULL
exposedField	SFNode	textureTransform	NULL

}

โหนด FontStyle

FontStyle {

field	SFString	family	“SERIF”
field	SFBool	horizontal	TRUE
field	SFString	justify	“BEGIN”
field	SFString	language	“ ”
field	SFBool	leftToRight	TRUE
field	SFFloat	size	1.0
field	SFFloat	spacing	1.0
field	SFStringstyle		“ ”
field	SFBool	toptoBottom	TRUE

}

โหนด ImageTexture

ImageTexture {

exposedField	MFString	url	[ ]
field	SFBool	repeatS	TRUE
field	SFBool	repeatT	TRUE

}

โหนด Material

Material {

exposedField	SFFloat	ambientIntensity	0.2
exposedField	SFColor	diffuseColor	0.8 0.8 0.8
exposedField	SFColor	emissiveColor	0 0 0
exposedField	SFFloat	shininess	0.2

exposedField	SFColor	specularColor	0 0 0
exposedField	SFFloat	transparency	0

}

โหนด Movietexture

Movietexture {

exposedField	SFBool	loop	FALSE
exposedField	SFFloat	speed	1
exposedField	SFTime	startTime	0
exposedField	SFTime	stopTime	0
exposedField	MFString	url	[ ]
field	SFBool	repeatS	TRUE
field	SFBool	repeatT	TRUE

}

เหตุการณ์ (Event)

eventOut	SFFloat	duration_changed
eventOut	SFBool	isActive

โหนด PixelTexture

PixelTexture {

exposedField	SFImage	image	0 0 0
field	SFBool	repeatS	TRUE
field	SFTime	repeatT	TRUE

}

โหนด TextureTransform

TextureTransform {

exposedField	SFVec2f	center	0 0 0
exposedField	SFFloat	rotation	0
exposedField	SFVec2f	scale	1 1
exposedField	SFVec2f	translation	0 0

}

โหนด TextureCoordinate

TextureCoordinate {

exposedField	MFVec2f	point	[ ]

}

โหนดที่ใช้ในการสร้างภาพเคลื่อนไหว (Interpolators)

โหนด ColorInterpolator

ColorInterpolator {

eventIn	SFFloat	set_fraction	
exposedField	MFFloat	key	[ ]
exposedField	MFColor	keyValue	[ ]

}

เหตุการณ์ (Event)

eventOut	SFColor	value_changed

โหนด CoordinateInterpolator

CoordinateInterpolator {

eventIn	SFFloat	set_fraction	
exposedField	MFFloat	key	[ ]
exposedField	MFVec3f	keyValue	[ ]

}

เหตุการณ์ (Event)

eventOut	SFColor	value_changed	[ ]

โหนด NormalInterpolator

NormalInterpolator {

exposedField	MFFloat	key	[ ]
exposedField	MFVec3f	keyValue	[ ]

}

เหตุการณ์ (Event)

eventIn	SFFloat	set_fraction	[ ]
eventOut	MFRotation	value_changed	[ ]

โหนด OrientationInterpolator

OrientationInterpolator {

exposedField	MFFloat	key	[ ]
exposedField	MFRotation	keyValue	[ ]

}

เหตุการณ์ (Event)

eventIn	SFFloat	set_fraction	[ ]
eventOut	SFRotation	value_changed	[ ]

โหนด PositionInterpolator

PositionInterpolator {

exposedField	MFFloat	key	[ ]
exposedField	MFVec3f	keyValue	[ ]

}

เหตุการณ์ (Event)

eventIn	SFFloat	set_fraction
eventOut	SFVec3f	value_changed

โหนด ScalarInterpolator

ScalarInterpolator {

exposedField	MFFloat	key	[ ]
exposedField	MFVec3f	keyValue	[ ]

}

เหตุการณ์ (Event)

eventIn	SFFloat	set_fraction
eventOut	SFFloat	value_changed

โหนดเกี่ยวกับการสร้างสิ่งแวดล้อม

โหนด Background

Background {

exposedField	MFFloat	groundAngle	[ ]
exposedField	MFColor	groundColor	[ ]
exposedField	MFString	backUrl	[ ]
exposedField	MFString	bottomUrl	[ ]

exposedField	MFString	frontUrl	[ ]
exposedField	MFString	leftUrl	[ ]
exposedField	MFString	rightUrl	[ ]
exposedField	MFString	topUrl	[ ]
exposedField	MFFloat	skyAngle	[ ]
exposedField	MFColor	skyColor	[ ]

}

เหตุการณ์ (Event)

eventIn	SFBool	set_bind
eventOut	SFBool	isBound

โหนด Fog

Fog {

exposedField	SFColor	color	1 1 1
exposedField	SFString	fogType	“LINEAR”
exposedField	SFFloat	visibilityRange	1000

}

เหตุการณ์ (Event)

eventIn	SFBool	set_bind
eventOut	SFBool	isBound

โหนด NavigationInfo

NavigationInfo {

exposedField	MFFloat	avatarSize	[0.25, 1.6,0.75]
exposedField	SFBool	headligh	TRUE
exposedField	SFFloat	speed	1.0
exposedField	MFString	Type	“WALK”
exposedField	SFFloat	visibilityLimit	0.0

}

เหตุการณ์ (Event)

eventIn	SFBool	set_bind
eventOut	SFBool	isBound

โหนด Viewpoint

Viewpoint {

exposedField	SFFloat	fieldOfView	0.785398
exposedField	SFBool	jump	TRUE
exposedField	SFRotation	orientation	0 0 1 0
exposedField	SFVec3f	position	0 0 0
field	SFString	description	“”

}

เหตุการณ์ (Event)

eventIn	SFBool	set_bind
eventOut	SFTime	bindTime_changed
eventOut	SFBool	isBound