คู่มือสำหรับโปรแกรมเมอร์
โปรแกรมถ่ายโอนโครงสร้างฐานข้อมูลเชิงสัมพันธ์
ไปยังโครงสร้างฐานข้อมูลเชิงวัตถุ

นายวุฒิธร วุฒิเจียรนัย

นายสมภพ นิธิไพจิตร์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

1. เครื่องมือที่ใช้ในการพัฒนาโปรเจค

1.1 Microsoft Visual Studio.NET

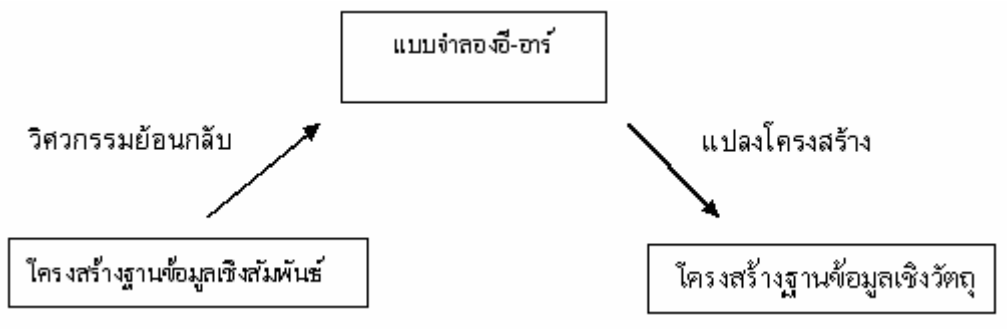
1.2 Microsoft Access 2000

1.3 Microsoft Windows XP

2. โครงโปรแกรมแบ่งออกเป็น 2 ส่วนใหญ่

2.1 ส่วนแปลงโครงสร้างฐานข้อมูล MS Access ไปยังแบบจำลองข้อมูลอี-อาร์

2.2 ส่วนแปลงแบบจำลองข้อมูล ER ไปยังโครงสร้างฐานข้อมูลเชิงวัตถุ



รูปที่ 1 โครงสร้างโปรแกรม

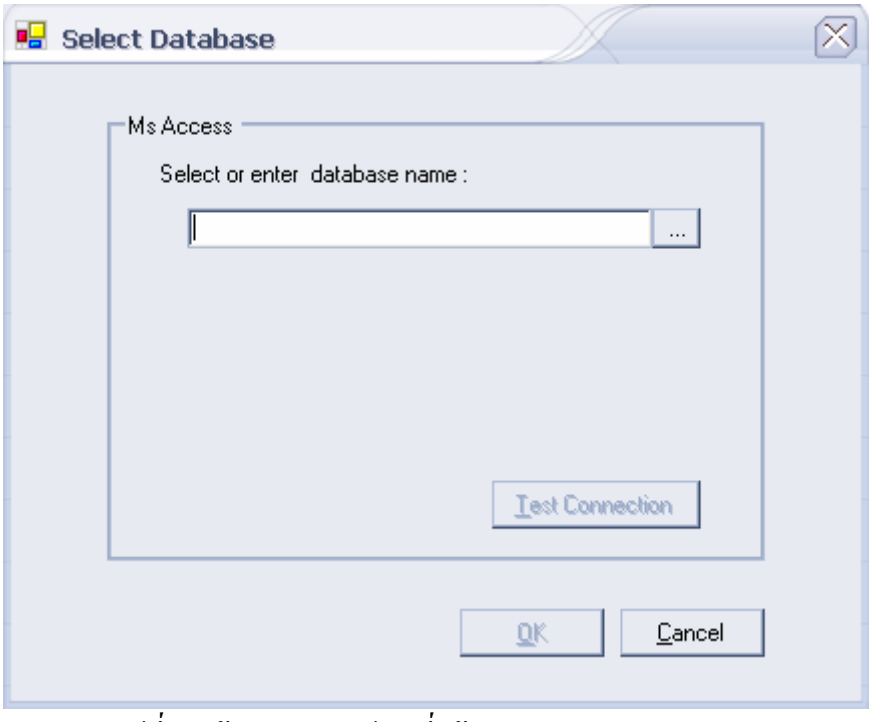
3. โปรแกรมส่วนแปลงโครงสร้างฐานข้อมูล MS Access ไปยังแบบจำลองข้อมูล อี-อาร์

ในส่วนนี้ประกอบด้วยคลาสหลัก 7 คลาส คือ

คลาส	หน้าที่การทำงาน
DatabaseControl	- จะเป็นตัวควบคุมการติดต่อกับฐานข้อมูล MS Access
SelectDatabase	- จะใช้ในการเลือกไฟล์ฐานข้อมูล MS Access
ShowRelationalSchema	- จะแสดงรายละเอียดโครงสร้างฐานข้อมูล MS Access
ERConverter	- จะเป็นคลาสที่ใช้ในการแปลงโครงสร้างฐานข้อมูล MS Access ไปยังแบบจำลองข้อมูลอี-อาร์
MetaER	- จะเป็นคลาสที่ใช้ในการเก็บรายละเอียดของอี-อาร์
ShowER	- จะทำหน้าที่แสดงรายละเอียดของอี-อาร์
SetDatabase	- จะส่งค่าไปยังฐานข้อมูลอี-อาร์เพื่อใช้ในการแปลงไปยังฐานข้อมูลเชิงวัตถุ

คลาส SelectDatabase

คลาสนี้จะเป็นหน้าจอเลือกไฟล์ติดต่อฐานข้อมูล MS Access โดยจะมีการเรียกใช้งาน DatabaseControl เพื่อติดต่อทดสอบการเชื่อมต่อฐานข้อมูล MS Access โดยทำการรับไฟล์จากผู้ใช้และทำการทดสอบการเชื่อมต่อกับไฟล์ฐานข้อมูล MS Access ว่าสามารถติดต่อได้หรือไม่



รูปที่ 1 หน้าจอแสดงออปเจคที่สร้างจากคลาส SelectDatabase

คลาส DatabaseControl

คลาสนี้จะเป็นตัวที่ทำการเชื่อมต่อกับฐานข้อมูล MS Access โดยจะเป็นตัวจัดการการเชื่อมต่อทั้งหมด กับฐานข้อมูล ถ้าหากคลาสใดจำเป็นที่จะต้องติดต่อกับฐานข้อมูล จะต้องทำการเรียกผ่านคลาสนี้เท่านั้น

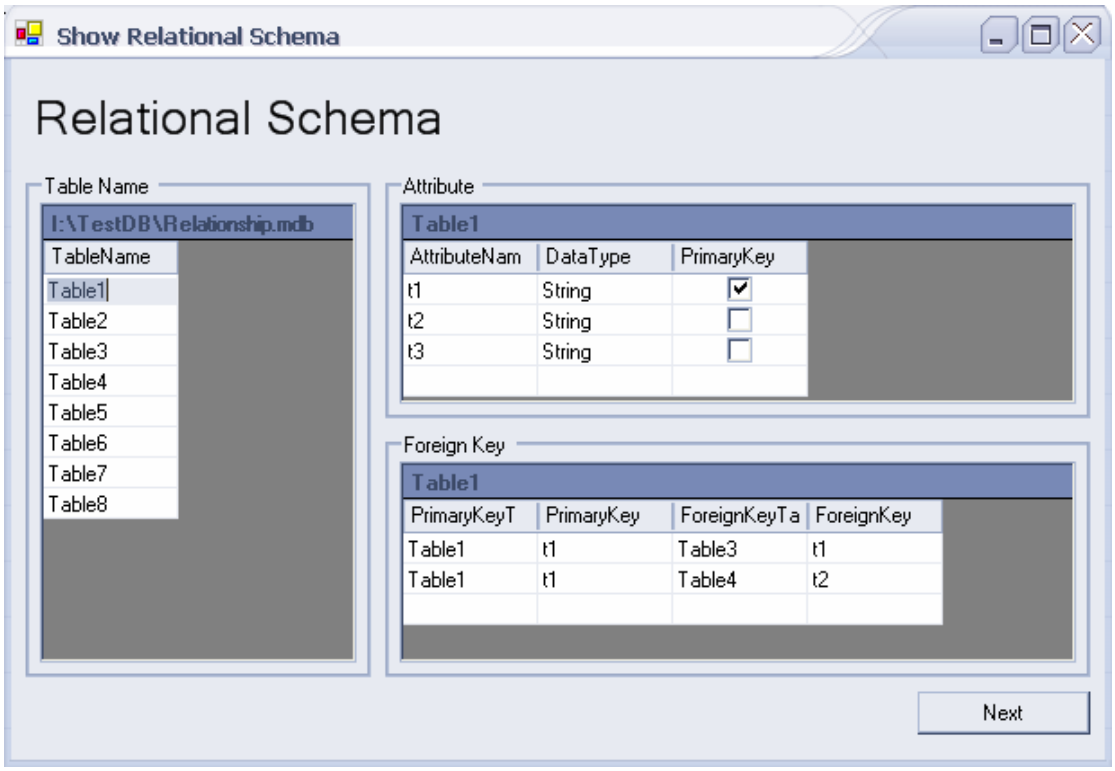
ในคลาสนี้จะประกอบไปด้วยฟังก์ชันการทำงานที่สำคัญได้แก่

ชื่อฟังก์ชัน	ลักษณะการทำงาน
GetSchema()	- จะใช้สำหรับดึงโครงสร้างทั้งหมดของฐานข้อมูล MS Access ออกมาโดยจะดึงออกมาในลักษณะ Dataset ซึ่งประกอบด้วยข้อมูลได้แก่ Primary key Foreign key
GetTableName()	- จะใช้ในการดึงชื่อตารางทั้งหมดจากโครงสร้างฐานข้อมูล MS Access ออกมาโดยจะดึงข้อมูลเป็นชื่อตารางแสดงออกมาในลักษณะ Datatable
TestConnection()	- จะใช้เพื่อทำการทดสอบการเชื่อมต่อกับฐานข้อมูลที่เลือกว่าสามารถติดต่อได้หรือไม่
GetPrimaryKey()	- จะดึงข้อมูล Primary key จากชื่อโครงสร้างฐานข้อมูล MS Access ออกมาโดยข้อมูลที่ได้ออกมาจะแสดงออกมาในลักษณะ Datatable
GetForeignKey()	- จะดึงข้อมูล Foreign key จากชื่อโครงสร้างฐานข้อมูล MS Access ออกมาโดยข้อมูลที่ได้ออกมาจะแสดงออกมาในลักษณะ Datatable
GetDetailTable()	- จะดึงข้อมูลของรายละเอียดของโครงสร้างฐานข้อมูล MS Access ว่ามีจำนวนแอททริบิวต์ จำนวน Primary key จำนวน Foreign key เท่าไรบ้าง

GetDatabaseName()	- จะดึงชื่อฐานข้อมูล MS Access ที่ทำการติดต่อด้วย
GetRowDistinct()	- จะแสดงจำนวนทั้งหมดของตารางที่ไม่ค่าไม่ซ้ำกัน
GetRow()	-จะแสดงจำนวนทั้งหมดของตาราง

คลาส ShowRelationalSchema

คลาสนี้จะเป็นการแสดงรายละเอียดของฐานข้อมูล MS Access ออกยังหน้าจอโดยจะดึงข้อมูลต่างๆ ภายในฐานข้อมูลแสดงออกยังหน้าจอ คือ ชื่อตารางทั้งหมด Primary key ที่อยู่ในตาราง และ foreign key ที่อยู่ในตาราง ในการทำงานจะสร้างออปเจคที่เกิดจากคลาส DatabaseControl และเรียกใช้งานเมธอดของคลาส เพื่อแสดงรายละเอียดของฐานข้อมูล



รูปที่ 2 หน้าจอแสดงออปเจคที่สร้างจากคลาส ShowRelationalSchema

คลาส ERConverter

คลาสนี้จะใช้ในการแปลงโครงสร้างฐานข้อมูล MS Access ไปยังแบบจำลองข้อมูลอี-อาร์ โดยจะมีขั้นตอนหลักๆ 5 ขั้นตอนได้แก่

- ตรวจสอบตารางที่เป็นเอนทิตีไทป์ทั้งหมด
- ตรวจสอบตารางที่เป็นเอนทิตีไทป์อ่อนแอทั้งหมด
- ตรวจสอบรีเลชั่นชิพอย่างง่าย
- ตรวจสอบรีเลชั่นชิพอย่างที่ซับซ้อน
- ตรวจสอบแอททริบิวต์ประเภทหลายค่า

คลาส MetaER

คลาสนี้จะทำการสร้าง Dataset เพื่อทำการเก็บรายละเอียดของแบบจำลองข้อมูลอี-อาร์ โดยภายใน Dataset จะประกอบด้วย 5 Datatable ได้แก่

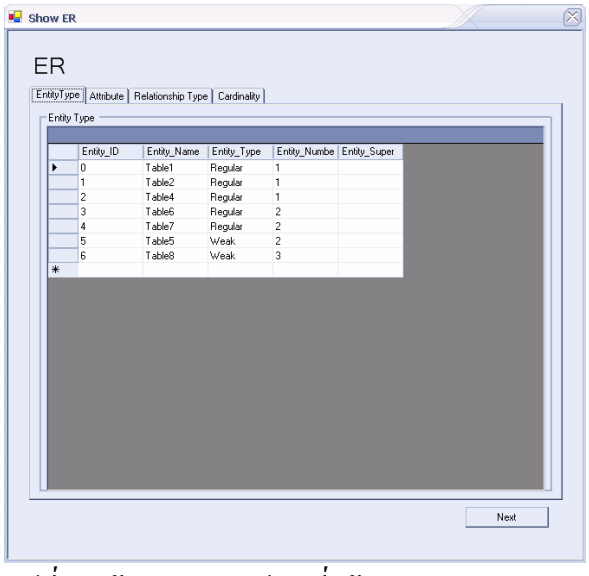
ชื่อ Datatable	รายละเอียดตาราง
Entity_Type	- เป็นตารางที่ใช้เก็บข้อมูลของเอนทิตีไทป์ทั้งหมดภายในแบบจำลองอี-อาร์
Relationship_Type	- เป็นตารางที่ใช้เก็บรีเลชั่นชิพไทป์ทั้งหมดภายในแบบจำลองอี-อาร์
Attribute	- เป็นตารางที่ใช้ในการเก็บแอททริบิวต์ทั้งหมดของแบบจำลองอี-อาร์
Entity_Cardinality	- ใช้ในการเก็บรายละเอียดของคาร์ดินอลลิตีของเอนทิตีไทป์
Relationship_Cardinality	- ใช้ในการเก็บรายละเอียดของคาร์ดินอลลิตีของรีเลชันชิพไทป์

ภายในคลาสจะประกอบด้วยฟังก์ชันที่สำคัญดังนี้

ชื่อฟังก์ชัน	หน้าที่การทำงาน
InsertEntityTable()	- เป็นฟังก์ชันที่ใช้ในการใส่ค่าเข้าไปในตารางเอนทิตี
InsertRelationshipTable()	- เป็นฟังก์ชันที่ใช้ในการใส่ค่าเข้าไปในตารางรีเลชั่นชิพ
InsertAttributeTable()	- เป็นฟังก์ชันที่ใช้ในการใส่ค่าเข้าไปในตารางแอททริบิวต์
InsertEntityCardinalityTable()	- เป็นฟังก์ชันที่ใช้ในการใส่ค่าเข้าไปในตารางคาร์ดินอลลิตีของเอนทิตีไทป์
InsertRelationshipCardinalityTable()	- เป็นฟังก์ชันที่ใช้ในการใส่ค่าเข้าไปในตารางคาร์ดินอลลิตีของรีเลชั่นชิพไทป์
FindEntity()	- เป็นฟังก์ชันในการหาเอนทิตี ID จากชื่อของเอนทิตี
FindRelationship()	- เป็นฟังก์ชันในการหารีเลชั่น ID จากชื่อของรีเลชั่น
FindAttribute()	- เป็นฟังก์ชันในการหาแอททริบิวต์ ID จากชื่อของแอททริบิวต์
InsertSuperClass()	- เป็นการใส่ข้อมูลแอททริบิวต์ SuperClass ของเอนทิตี
GetMetaER()	- ส่งค่า Dataset ที่เก็บรายละเอียดของมูลแบบจำลองข้อมูลอี-อาร์

คลาส ShowER

คลาสนี้ใช้ในการแสดงผลแบบจำลองอี-อาร์ ที่ได้จากคลาส ERConverter โดยรายละเอียดของแบบจำลองอี-อาร์ ที่อยู่ภายในคลาส MetaER โดยจะแสดงรายละเอียดของแบบอี-อาร์ ในตาราง Datagrid โดยสามารถแก้ไขรายละเอียดของข้อมูลที่อยู่ภายใน Datagrid ได้



รูปที่ 3 หน้าจอแสดงออปเจคที่สร้างจากคลาส ShowER

คลาส Setdatabase

คลาสนี้ใช้ในการส่งค่าแบบจำลองข้อมูลอี-อาร์ที่อยู่ใน ออปเจค MetaER ไปยังฐานข้อมูลแบบจำลองข้อมูลอี-อาร์ที่เหมาะสมกับโครงสร้างฐานข้อมูลเชิงวัตถุ ที่ทำการแยกแบบจำลองข้อมูลอี-อาร์ เพื่อที่จะสามารถนำแบบจำลองข้อมูลอี-อาร์สามารถประยุกต์ใช้กับงานได้หลากหลายมากขึ้น

ภายในคลาสประกอบด้วยฟังก์ชันหลักดังนี้

ชื่อฟังก์ชัน	หน้าที่การทำงาน
InsertEntityTable()	- ใส่ค่าเข้าไปในตาราง Entity
InsertRelationshipTable()	- ใส่ค่าเข้าไปในตาราง Relationship
InsertEntityAttributeTable()	- ใส่ค่าเข้าไปในตาราง Attribute
InsertRelationshipAttributeTable()	- ใส่ค่าเข้าไปในตาราง RelationshipAttribute
InsertStructureAttributeTable()	- ใส่ค่าเข้าไปในตาราง StructureAttribute
InsertCardinalityTable()	- ใส่ค่าเข้าไปในตาราง Cardinality
ConvertEntityType()	- ใช้ในการแปลงประเภทของเอนทิตีไทป์
FindEntityID()	- ใช้ในการหา Entity ID จากชื่อเอนทิตี
FindRelationshipID()	- ใช้ในการหา Relationship ID จากชื่อรีเลชั่นชิพ
FindStructName()	- ใช้ในการหา StructName จากชื่อแอททริบิวต์
ConvertDataType()	- ใช้ในการแปลงชนิดข้อมูลของแอททริบิวต์
ConvertAttributeType()	- ใช้ในการแปลงประเภทของแต่ละแอททริบิวต์
CheckArray()	- ใช้ในการตรวจสอบว่าแอททริบิวต์เป็นประเภทอาเรย์หรือไม่
ConvertCardinality()	- ใช้ในการแปลงประเภทของคาร์ดินอลลิตี
ChangeRelationshipName()	- ใช้ในการเปลี่ยนชื่อของรีเลชั่นชิพให้เหมาะสม

ChangeEntityName()	- ใช้ในการเปลี่ยนชื่อของเอนทิตีให้เหมาะสม
ChangeAttributeName()	- ใช้ในการเปลี่ยนชื่อของแอททริบิวต์ให้เหมาะสม
SetRelationalMetaER()	- ใช้ในการหาส่งค่าว่า ERใดที่จะทำการส่งข้อมูล

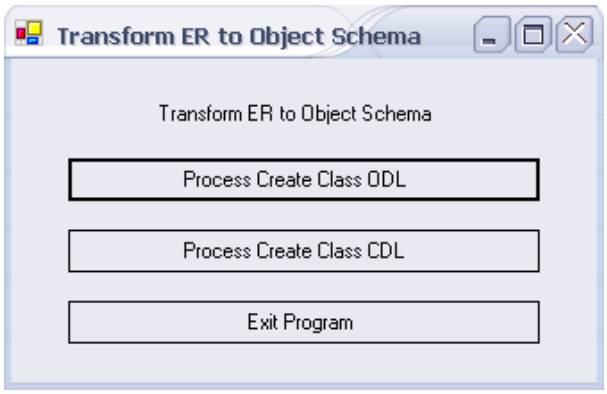
4. ส่วนแปลงแบบจำลองข้อมูล ER ไปยังโครงสร้างฐานข้อมูลเชิงวัตถุ

ประกอบไปด้วยการทำงาน 3 ส่วนคือ

1.ส่วนการแปลงจากแบบจำลองข้อมูลอี-อาร์ ไปเป็นโครงสร้างฐานข้อมูลเชิงวัตถุแบบ ODL

2.ส่วนการแปลงจากแบบจำลองข้อมูลอี-อาร์ ไปเป็นโครงสร้างฐานข้อมูลเชิงวัตถุแบบ CDL

3.ส่วนจบการทำงานของโปรแกรม



รูปที่ 4 หน้าจอแสดการเลือกรูปแบบไฟล์การแปลงโครงสร้างฐานข้อมูลเชิงวัตถุ

ในการแปลงแบบจำลองข้อมูล ER ไปยังโครงสร้างฐานข้อมูลเชิงวัตถุประกอบด้วยคลาสหลัก 6 คลาส คือ

คลาส	หน้าที่การทำงาน
tempClass	- เป็นคลาสที่ใช้เก็บรายละเอียดต่างๆ ของคลาสที่ได้มาจากการแปลงแบบจำลองข้อมูลอี-อาร์
CreateClass	- เป็นคลาสแม่ที่ประกอบด้วยตัวแปร และฟังก์ชั่นพื้นฐานของคลาส CreateClassODL และ CreateClassCDL
CreateClassODL	- เป็นคลาสที่ใช้ในการสร้างคลาสตามรูปแบบของไฟล์ ODL
CreateClassCDL	- เป็นคลาสที่ใช้ในการสร้างคลาสตามรูปแบบของไฟล์ CDL
makeODL	- เป็นคลาสที่ใช้ในการสร้างไฟล์ .ODL ที่ได้มาจากคลาส CreateClassODL
makeCDL	- เป็นคลาสที่ใช้ในการสร้างไฟล์ .CDL ที่ได้มาจากคลาส CreateClassCDL

คลาส tempClass

คลาสนี้จะเป็นคลาสที่ใช้เก็บรายละเอียดต่างๆ ของคลาส ประกอบตัวแปรต่างๆ ดังนี้คือ

ตัวแปร	ชนิดของข้อมูล	รายละเอียดของตัวแปร
className	String	- ใช้เก็บชื่อของคลาสที่มาจากการแปลงแบบจำลองข้อมูลอี-อาร์
superClass	String	- ใช้เก็บชื่อคลาสแม่ที่คลาสนี้มาการสืบทอดมา
Attribute	Array of String	- ใช้เก็บรายละเอียดต่างๆ ของแอททริบิวต์
Relationship	Array of String	- ใช้เก็บรายละเอียดต่างๆ ของความสัมพันธ์ระหว่างคลาส

และมีฟังก์ชั่นหลักๆ ภายในคลาสดังนี้คือ

ชื่อฟังก์ชัน	ลักษณะการทำงาน
setArrayToAttribute()	- ใช้กำหนดขนาดอาเรย์ให้กับตัวแปร attribute
setArrayToRelationship()	- ใช้กำหนดขนาดอาเรย์ให้กับตัวแปร relationship
getLastIndex()	- เป็นฟังก์ชั่นใช้หาค่า index ตัวสุดท้ายของตัวแปรอาเรย์ relationship

คลาส CreateClass

คลาสนี้เป็นเป็นคลาสแม่ที่ประกอบด้วยตัวแปร และฟังก์ชั่นพื้นฐานของคลาส CreateClassODL และ CreateClassCDL โดยประกอบด้วยตัวแปรหลักๆ ดังนี้คือ

ตัวแปร	ชนิดของข้อมูล	รายละเอียดของตัวแปร
projectName	String	- ใช้เก็บชื่อโปรเจ็คของคลาส
Total	Integer	- ใช้เก็บจำนวนคลาสทั้งหมดจากแบบจำลองข้อมูลอี-อาร์
Nary	Array of Integer	- ใช้เก็บ index ของเอ็นติตี้ไทพที่แปลงมาจากความสัมพันธ์แบบ N-ary relationship type
manyAttribute	Array of Integer	- ใช้เก็บ index ของแอททริบิวต์ที่มาจากความสัมพันธ์แบบ Many-to-many relationship
relationHasAttribute	Array of String	- ใช้เก็บ index ของคลาสที่แปลงมาจากความสัมพันธ์ที่มีแอททริบิวต์ประกอบอยู่ด้วย
Conn	OleDbConnection	- ใช้เชื่อมต่อฐานข้อมูลของ ERmete Schema

และมีฟังก์ชั่นหลักๆ ภายในคลาสดังนี้คือ

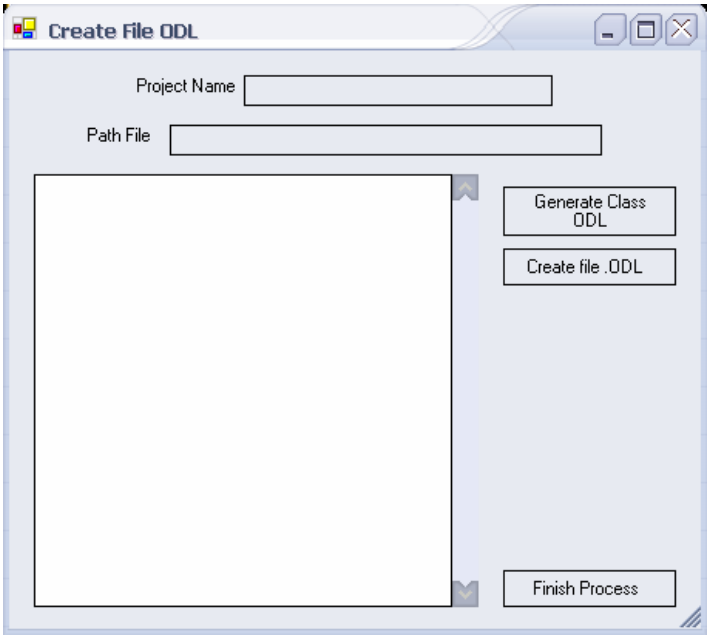
ชื่อฟังก์ชัน	ลักษณะการทำงาน
setConnection()	- จะใช้สำหรับเชื่อมต่อฐานข้อมูลของ ERmete Schema

change_first_char_to_upper()	- จะใช้เปลี่ยนตัวอักษรตัวแรกของชนิดของข้อมูลให้กลายเป็นตัวอักษรตัวใหญ่
find_rid_in_array_integer()	- จะใช้ในการตรวจสอบค่า index ว่ามีอยู่ในอาเรย์ของตัวแปรชนิดข้อมูลหรือไม่
Get_index_in_create_class()	- จะให้ค่า index ในอาเรย์ของ tempClass ที่มีชื่อตรงกับค่าอากิวเมนต์ (argument) ที่ส่งเข้าไป
get_num_total_in_relation_attribute()	- จะใช้หาจำนวนแอททริบิวต์ทั้งหมดจากความสัมพันธ์ที่มีแอททริบิวต์เป็นส่วนประกอบ หรือตัวแปร reAttribute
get_indexs_eid_in_relation_attribute()	- จะใช้หาค่า index ของความสัมพันธ์ที่มีแอททริบิวต์เป็นส่วนประกอบ หรือตัวแปร reattribute

คลาส CreateClassODL

เป็นคลาสที่ใช้ในการสร้างโครงสร้างฐานข้อมูลเชิงวัตถุตามรูปแบบของไฟล์ ODL โดยมีการสืบทอดคุณสมบัติต่างๆ มาจากคลาส CreateClass โดยภายในคลาส CreateClassODL มีตัวแปรอาเรย์ชนิดข้อมูล tempClass อยู่ใช้ในการสร้างจำนวนคลาสทั้งหมดที่ได้มาจากแบบจำลองข้อมูลอี-อาร์ และมีฟังก์ชั่นหลักๆ ภายในคลาสดังนี้คือ

ชื่อฟังก์ชัน	ลักษณะการทำงาน
GenClassODL()	- เป็นกระบวนการในการสร้างคลาสรูปแบบของไฟล์ ODL
totalClass()	- จะใช้หาจำนวนคลาสทั้งหมดของระบบ ที่ได้มาจากแบบจำลองข้อมูลอี-อาร์
listRIDrelationship()	- จะใช้กำหนดค่า index ของ relationship ที่มีแอททริบิวต์เป็นส่วนประกอบอยู่ลงในตัวแปรอาเรย์ reattribute ยกเว้น Many-to-many relationship
makeClassEntityType()	- จะใช้สร้างคลาสที่มาจากเอ็นติตี้ไทพในแบบจำลองข้อมูลอี-อาร์
makeClassNary()	- จะใช้สร้างคลาสที่มาจากความสัมพันธ์ระหว่างคลาสที่เป็นแบบ N-ary relationship ในแบบจำลองข้อมูลอี-อาร์
makeClassManyRelationship()	- จะใช้สร้างคลาสที่มาจากความสัมพันธ์ระหว่างคลาสที่เป็นแบบ Many-to-many relationship ที่แอททริบิวต์เป็นส่วนประกอบในแบบจำลองข้อมูลอี-อาร์
makeRelationshipForClassODL()	- ใช้กำหนดความสัมพันธ์ระหว่างคลาสให้กับคลาสที่ได้มาจากแบบจำลองข้อมูลอี-อาร์



รูปที่ 6 หน้าจอแสดคลาสที่สร้างจากคลาส CreateClassODL

คลาส CreateClassCDL

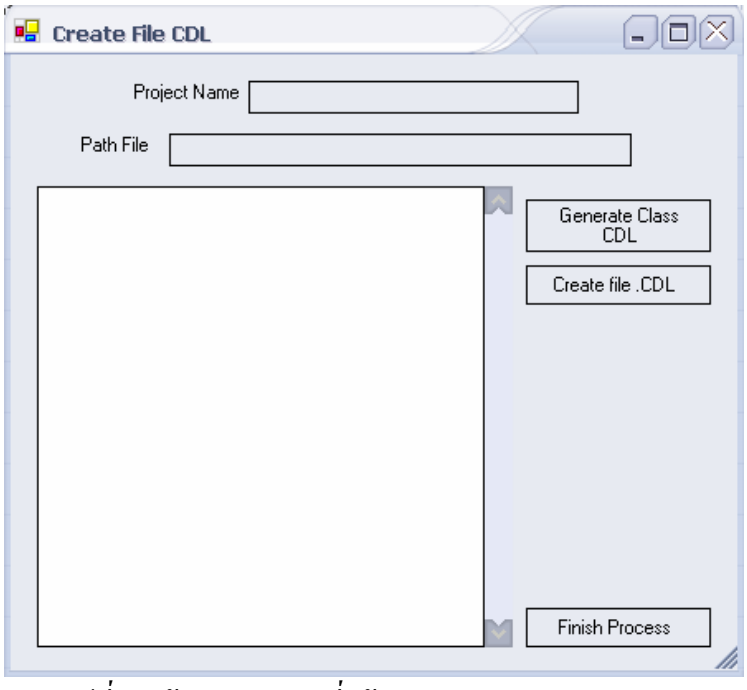
เป็นคลาสที่ใช้ในการสร้างโครงสร้างฐานข้อมูลเชิงวัตถุตามรูปแบบของไฟล์ CDL โดยมีการสืบทอดคุณสมบัติต่างๆ มาจากคลาส CreateClass โดยภายในคลาส CreateClassCDL ประกอบด้วยตัวแปรหลักๆ ดังนี้คือ

ตัวแปร	ชนิดของข้อมูล	รายละเอียดของตัวแปร
tmpClass	Array of tempClass	- ใช้เก็บรายละเอียดต่างๆ ของคลาส
structAttribute	Array of Integer	- เก็บค่า index ของคลาสที่เกิดมาจาก Composite Attribute

และมีฟังก์ชั่นหลักๆ ภายในคลาสดังนี้คือ

ชื่อฟังก์ชัน	ลักษณะการทำงาน
GenClassCDL()	- เป็นกระบวนการในการสร้างคลาสรูปแบบของไฟล์ CDL
totalClass()	- จะใช้หาจำนวนคลาสทั้งหมดของระบบ ที่ได้มาจากแบบจำลองข้อมูลอี-อาร์
listRIDrelationship()	- จะใช้กำหนดค่า index ของ relationship ที่มีแอททริบิวต์เป็นส่วนประกอบอยู่ลงในตัวแปรอาเรย์ reattribute ยกเว้น Many-to-many relationship
makeClassStructure()	- จะใช้สร้างคลาสที่มาจากแอททริบิวต์แบบ Composite Attribute
makeClassEntityType()	- จะใช้สร้างคลาสที่มาจากเอ็นติตี้ไทป์ในแบบจำลองข้อมูลอี-

	อาร์
makeClassNary()	- จะใช้สร้างคลาสที่มาจากความสัมพันธ์ระหว่างคลาสที่เป็นแบบ N-ary relationship ในแบบจำลองข้อมูลอี-อาร์
makeClassManyRelationship()	- จะใช้สร้างคลาสที่มาจากความสัมพันธ์ระหว่างคลาสที่เป็นแบบ Many-to-many relationship ที่แอททริบิวต์เป็นส่วนประกอบในแบบจำลองข้อมูลอี-อาร์
makeRelationshipForClassCDL()	- ใช้กำหนดความสัมพันธ์ระหว่างคลาสให้กับคลาสที่ได้มาจากแบบจำลองข้อมูลอี-อาร์



รูปที่ 7 หน้าจอแสดคลาสที่สร้างจากคลาส CreateClassCDL

คลาส makeODL

เป็นคลาสที่ใช้ในการสร้างไฟล์โครงสร้างฐานข้อมูลเชิงวัตถุตามรูปแบบของไฟล์ ODL โดยผลลัพธ์ของคลาสนี้ เราจะได้ไฟล์ .ODL ภายในคลาส makeODL ประกอบด้วยตัวแปรหลักๆ ดังนี้คือ

ตัวแปร	ชนิดของข้อมูล	รายละเอียดของตัวแปร
projectName	String	- ใช้เก็บชื่อโปรเจ็คของคลาส
tmpODL	Array of tempClass	- เก็บค่า index ของคลาสที่เกิดมาจาก Composite Attribute
Tout	TextBox	- ใช้เก็บรูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุตามรูปแบบของไฟล์ ODL

และมีฟังก์ชั่นหลักๆ ภายในคลาสดังนี้คือ

ชื่อฟังก์ชัน	ลักษณะการทำงาน
genFileODL()	- เป็นกระบวนการในการสร้างคลาสตามรูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุของไฟล์ ODL ใส่ลงในตัวแปร TextBox
createFileODL()	- จะนำข้อมูลที่อยู่ในตัวแปร TextBox มาสร้างเป็นคลาสตามรูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุของไฟล์ ODL

```
1  interface ATTRIBUTE {
2      attribute Integer ATTRIBUTEID;
3      attribute Integer DOCUMENTTYPEID;
4      attribute String ATTRIBUTENAME;
5      attribute String ATTRIBUTETYPE;
6      relationship Set<DOCUMENTTYPE> ATTRIBUTEDOCUMENTTYPEATTRIBUTEDOCUMENTTYPE
7          inverse DOCUMENTTYPE::DOCUMENTTYPEDOCUMENTTYPEATTRIBUTEATTRIBUTE;
8      relationship WORKATTRIBUTEing ATTRIBUTEWORKATTRIBUTE646WORKATTRIBUTEing
9          inverse WORKATTRIBUTEing::WORKATTRIBUTEingWORKATTRIBUTE646ATTRIBUTE;
10 }
11 interface CONDITION {
12     attribute Integer CONDITIONID;
13     attribute String STATEMENT;
14     attribute Integer NEXTSTATEID;
15     attribute Integer STATEID;
16     relationship Set<STATE> CONDITIONSTATECONDITIONSTATE
17         inverse STATE::STATESTATECONDITIONCONDITION;
18     relationship Set<STATE> CONDITIONSTATECONDITION2STATE
19         inverse STATE::STATESTATECONDITION2CONDITION;
20 }
21 interface DOCUMENTTYPE {
22     attribute Integer DOCUMENTTYPEID;
23     attribute String DOCUMENTTYPENAME;
24     attribute String DOCUMENTTYPEDESC;
25     relationship ATTRIBUTE DOCUMENTTYPEDOCUMENTTYPEATTRIBUTEATTRIBUTE
26         inverse ATTRIBUTE::ATTRIBUTEDOCUMENTTYPEATTRIBUTEDOCUMENTTYPE;
27     relationship WORKFLOW DOCUMENTTYPEDOCUMENTTYPEWORKFLOWWORKFLOW
28         inverse WORKFLOW::WORKFLOWDOCUMENTTYPEWORKFLOWDOCUMENTTYPE;
29 }
30 interface GROUP {
31     attribute Integer GROUPID;
32     attribute String GROUPNAME;
33     attribute String GROUPDESC;
34     relationship GROUPSTATEing GROUPGROUPSTATE649GROUPSTATEing
35         inverse GROUPSTATEing::GROUPSTATEingGROUPSTATE649GROUP;
36     relationship Set<USER> GROUPGROUPUSERUSER
37         inverse USER::USERGROUPUSERGROUP;
38     relationship Set<ROLE> GROUPROLEGROUPROLE
39         inverse ROLE::ROLEROLEGROUPGROUP;
40 }
```

รูปที่ 8 แสดรูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุที่สร้างจากคลาส makeODL

คลาส makeCDL

เป็นคลาสที่ใช้ในการสร้างไฟล์โครงสร้างฐานข้อมูลเชิงวัตถุตามรูปแบบของไฟล์ CDL โดยผลลัพธ์ของคลาสนี้ เราจะได้ไฟล์ .CDL ภายในคลาส makeCDL ประกอบด้วยตัวแปรหลักๆ ดังนี้คือ

ตัวแปร	ชนิดของข้อมูล	รายละเอียดของตัวแปร
projectName	String	- ใช้เก็บชื่อโปรเจ็คของคลาส
tmpCDL	Array of tempClass	- เก็บค่า index ของคลาสที่เกิดมาจาก Composite Attribute
Tout	TextBox	- ใช้เก็บรูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุตามรูปแบบของไฟล์ CDL

และมีฟังก์ชั่นหลักๆ ภายในคลาสดังนี้คือ

ชื่อฟังก์ชัน	ลักษณะการทำงาน
genFileCDL()	- เป็นกระบวนการในการสร้างคลาสตามรูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุของไฟล์ CDL ใส่ลงในตัวแปร TextBox
createFileCDL()	- จะนำข้อมูลที่อยู่ในตัวแปร TextBox มาสร้างเป็นคลาสตาม

	รูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุของไฟล์ CDL

```
1  class Relational.ATTRIBUTE
2  {
3          classtype = persistent;
4          super = %Persistent;
5          persistent;
6          attribute ATTRIBUTEID { type = %Integer; }
7          attribute DOCUMENTTYPEID { type = %Integer; }
8          attribute ATTRIBUTENAME { type = %String; }
9          attribute ATTRIBUTETYPE { type = %String; }
10         relationship ATTRIBUTEDOCUMENTTYPEATTRIBUTEDOCUMENTTYPE { type = Relational.DOCUMENTTYPE; cardinality = many; inverse = DOCUMENTTYPEDOCUMEN
11         relationship ATTRIBUTEWORKATTRIBUTE646WORKATTRIBUTEing { type = Relational.WORKATTRIBUTEing; cardinality = one; inverse = WORKATTRIBUTEingWORK
12 }
13 class Relational.CONDITION
14 {
15         classtype = persistent;
16         super = %Persistent;
17         persistent;
18         attribute CONDITIONID { type = %Integer; }
19         attribute STATEMENT { type = %String; }
20         attribute NEXTSTATEID { type = %Integer; }
21         attribute STATEID { type = %Integer; }
22         relationship CONDITIONSTATECONDITION1STATE { type = Relational.STATE; cardinality = many; inverse = STATESTATECONDITION1CONDITION; }
23         relationship CONDITIONSTATECONDITION2STATE { type = Relational.STATE; cardinality = many; inverse = STATESTATECONDITION2CONDITION; }
24 }
25 class Relational.DOCUMENTTYPE
26 {
27         classtype = persistent;
28         super = %Persistent;
29         persistent;
30         attribute DOCUMENTTYPEID { type = %Integer; }
31         attribute DOCUMENTTYPENAME { type = %String; }
32         attribute DOCUMENTTYPEDESC { type = %String; }
33         relationship DOCUMENTTYPEDOCUMENTTYPEATTRIBUTEATTRIBUTE { type = Relational.ATTRIBUTE; cardinality = one; inverse = ATTRIBUTEDOCUMENTTYPEATTR
34         relationship DOCUMENTTYPEDOCUMENTTYPEWORKFLOWWORKFLOW { type = Relational.WORKFLOW; cardinality = one; inverse = WORKFLOWDOCUMENTTYPEWO
35 }
36 class Relational.GROUP
37 {
38         classtype = persistent;
39         super = %Persistent;
40         persistent;
41         attribute GROUPID { type = %Integer; }
42         attribute GROUPNAME { type = %String; }
43         attribute GROUPDESC { type = %String; }
44         relationship GROUPGROUPSTATE648GROUPSTATEing { type = Relational.GROUPSTATEing; cardinality = one; inverse = GROUPSTATEingGROUPSTATE648GROUP; }
45         relationship GROUPGROUPUSER649GROUPUSERing { type = Relational.GROUPUSERing; cardinality = one; inverse = GROUPUSERingGROUPUSER649GROUP; }
46         relationship GROUPROLEGROUP649ROLEGROUPing { type = Relational.ROLEGROUPing; cardinality = one; inverse = ROLEGROUPingROLEGROUP649GROUP; }
47 }
```

รูปที่ 9 แสดรูปแบบโครงสร้างฐานข้อมูลเชิงวัตถุที่สร้างจากคลาส makeCDL