หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

จากเนื้อหาในบทนำที่ผ่านมาได้แสดงถึงโครงร่างของตัวงานวิจัยเองทั้งหมด ในบทที่ 2 นี้จะนำเสนอแนวคิดและทฤษฎี ตลอดจนเอกสารและงานวิจัยที่เกี่ยวข้อง เพื่อเป็นข้อมูลการเปรียบเทียบกับงานวิจัยนี้ต่อไป

2.1 แนวคิดและทฤษฎี

2.1.1 การใช้ส่วนประกอบซอฟต์แวร์ในงานพัฒนาโปรแกรม

รูปแบบการพัฒนาซอฟต์แวร์โดยใช้ส่วนประกอบซอฟต์แวร์ทำให้เกิดผลประโยชน์ [6] [10] ดังนี้

1. ทำให้เกิดการพัฒนาที่เป็นอิสระจากกัน ก่อให้เกิดความเชี่ยวชาญแต่ละด้านในกลุ่มนักพัฒนา รวมถึงมีการแข่งขันกันระหว่างผู้ผลิตในแต่ละด้าน

2. การพัฒนาซอฟต์แวร์บนเฟรมเวิร์คที่มีมาตรฐานจะทำให้สามารถนำส่วนประกอบซอฟต์แวร์ที่มีอยู่แล้วมาใช้ได้ง่าย ทำให้ลดเวลาและค่าใช้จ่ายในการพัฒนา

3. การพัฒนางานบนพื้นฐานของส่วนประกอบซอฟต์แวร์ช่วยลดงานในขั้นตอนการออกแบบโครงสร้าง และสถาปัตยกรรม

4. สามารถกำหนดทิศทางการพัฒนาลักษณะประจำและข้อมูลที่สนใจ โดยกำหนดไว้ในเฟรมเวิร์คของงานเฉพาะด้าน

5. ทำให้เกิดมาตรฐาน และความถูกต้องตรงกันในการทำงานระหว่างองค์กรที่มีการพัฒนางานโดยใช้ส่วนประกอบซอฟต์แวร์

6. ทำให้การบำรุงรักษา หรือปรับปรุงแก้ไขสามารถกระทำเป็นส่วนๆได้ ช่วยลดขั้นตอนและความซับซ้อนของงาน

7. กระจายความเสี่ยงของธุรกิจอุตสาหกรรมซอฟต์แวร์ใหญ่ไปในแต่ละกลุ่มของผู้ผลิตส่วนประกอบซอฟต์แวร์

บนอินเทอร์เน็ตซึ่งเป็นเครือข่ายข้อมูลสาธารณะขนาดใหญ่ ที่มีทั้งอุปสงค์และอุปทานของส่วนประกอบซอฟต์แวร์จำนวนมาก แต่การค้นหาส่วนประกอบซอฟต์แวร์นั้นทำได้ยาก มีความพยายามในการสร้างระบบเก็บส่วนประกอบซอฟต์แวร์ที่นำมาใช้ร่วมกัน หรือระบบซึ่งเก็บข้อมูลของลักษณะประจำและข้อมูลของส่วนประกอบซอฟต์แวร์ [1] [8] เพื่อช่วยสนับสนุนให้เกิดความสะดวกในการค้นหาและนำไปใช้

2.1.2 การค้นหาผู้จำหน่ายส่วนประกอบซอฟต์แวร์

การใช้โปรแกรมค้นหาทั่วไปในการค้นหาจะได้รับผลลัพธ์ซึ่งไม่เกี่ยวข้องกับส่วนประกอบซอฟต์แวร์รวมเข้ามาด้วย เพื่อให้งานวิจัยสามารถดำเนินต่อไปได้โดยมีประสิทธิภาพ จึงได้รวบรวมรายชื่อผู้จำหน่ายส่วนประกอบซอฟต์แวร์จากเว็บเพจของตัวแทนจำหน่าย [6] แล้วคัดเลือกขึ้นมาจำนวนหนึ่งโดยพิจารณาจำนวนครั้งที่พบในกลุ่มของเว็บเพจของตัวแทนจำหน่าย

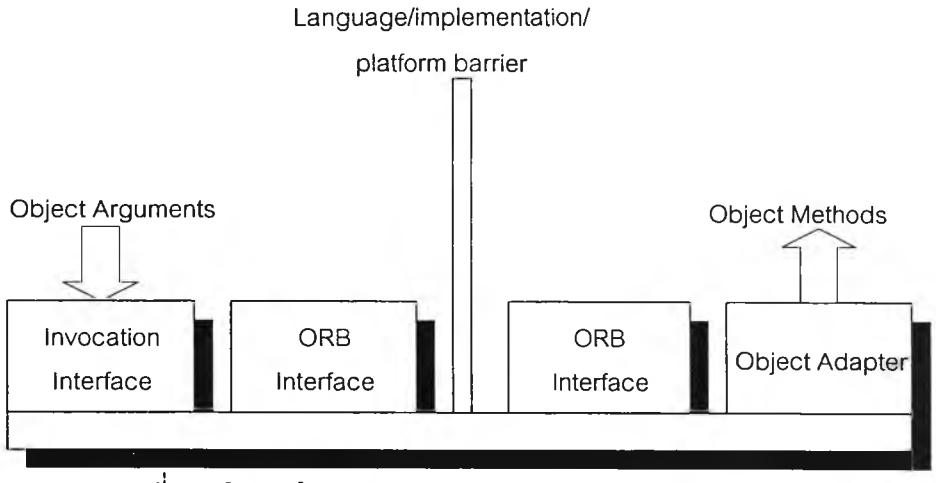
เมื่อได้รายชื่อผู้จำหน่ายส่วนประกอบซอฟต์แวร์ซึ่งกำหนดให้เป็นกลุ่มประชากรที่สนใจ จึงเข้าไปตรวจสอบและรวบรวมข้อมูลจากเว็บเพจของผู้จำหน่ายเพื่อนำมาวิเคราะห์และนำเสนอในงานวิจัยต่อไป

2.2.2 เทคโนโลยีของส่วนประกอบซอฟต์แวร์ [2]

ในส่วนนี้ผู้เขียนขอใช้ภาษาอังกฤษในศัพท์บางคำเพื่อให้เกิดความเข้าใจตรงกันที่สะดวกขึ้น แต่บางส่วนจะใช้ภาษาไทยและระบุภาษาอังกฤษไว้ในวงเล็บ นอกจากนี้ยังได้รวบรวมคำศัพท์ส่วนหนึ่งตามราชบัณฑิตยสถานไว้ในภาคผนวก ง

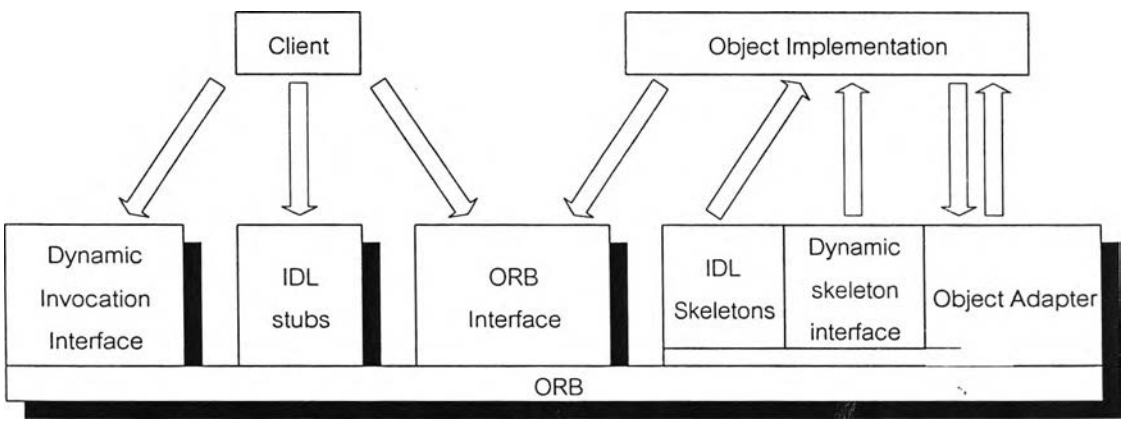
เนื้อหาในส่วนนี้จะขอกล่าวถึงเทคโนโลยีหลักของส่วนประกอบซอฟต์แวร์ 3 สาย อันได้แก่ CORBA ซึ่งเป็นเทคโนโลยีมาตรฐานส่วนประกอบซอฟต์แวร์ถือกำเนิดเริ่มต้นมาสำหรับระดับองค์กร (Enterprise) ActiveX หรือ COM เป็นเทคโนโลยีมาตรฐานส่วนประกอบซอฟต์แวร์ซึ่งมีบทบาทบนเครื่องเดสก์ท็อปที่มีแพลตฟอร์มของไมโครซอฟท์ และสุดท้าย Java หรือ JavaBean ของซันซึ่งเริ่มต้นจากการพัฒนาบนอินเทอร์เน็ตและเว็บ และได้จุดเด่นจากภาษา Java ซึ่งสามารถทำงานได้บนทุกแพลตฟอร์ม

2.2.2.1 CORBA ถือกำเนิดจาก OMG (Object Management Group) ซึ่งก่อตั้งเมื่อ 1989 โดยเป็นองค์กรที่ไม่มุ่งเน้นผลกำไร (Non-Profit Organization) ความพยายามเริ่มแรกที่ OMG ต้องการคือ ทำอย่างไรที่จะนำส่วนประกอบซอฟต์แวร์ซึ่งมีลักษณะเชิงวัตถุ (Object) สามารถทำงานร่วมกันโดยกระจายอยู่บนเครื่องต่างๆซึ่งเป็นอิสระแก่กันทั้งในด้านแพลตฟอร์มและภาษาที่ใช้ในการโปรแกรม



รูปที่ 2.1 โครงสร้างอย่างง่ายของ ORB-based system

CORBA เริ่มต้นนั้นทำงานโดย 3 ส่วนประกอบหลัก ได้แก่ ส่วนของ Invocation Interfaces, ORB (Object Request Broker) และ ส่วนของ Object Adapter โดย Invocation Interfaces ของออบเจ็กต์หนึ่งจะร้องขอให้เชื่อมโยงการติดต่อกับอีกออบเจ็กต์หนึ่งซึ่งมีลักษณะเป็น Late Binding ผ่านทาง ORB และ Object Adapter โดยมี Arguments จาก Invocaiton Interface ไปเรียก Method ของอีกออบเจ็กต์ตามรูปที่ 2.1 การทำงานส่วนนี้จะสำเร็จได้ขึ้นกับว่าส่วนต่อประสานในทุกออบเจ็กต์ต้องได้รับการนิยาม (Interface Definition) ไว้ในภาษากลาง (Common Language) สำหรับให้ทุกภาษาโปรแกรมสามารถเชื่อม (Bindings) มายังภาษากลางนี้ได้ ภาษากลางนี้เรียกว่า OMG IDL (OMG Interface Definition Language) ซึ่งมีคอมไพเลอร์สำหรับแปลภาษาโปรแกรมเก็บไว้ใน Interface Repository



รูปที่ 2.2 Common Object Request Broker Architecture (CORBA)

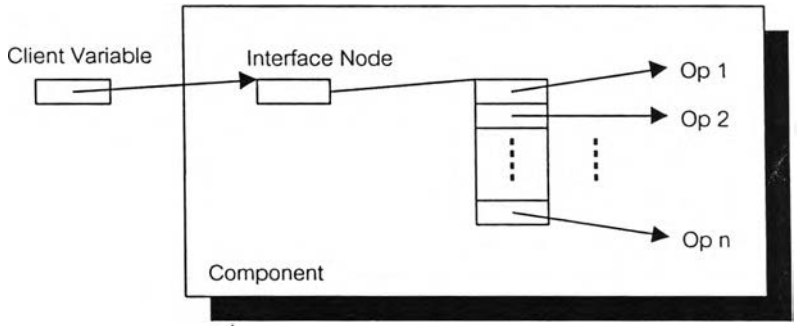
รูปที่ 2.2 ข้างต้นแสดง CORBA ซึ่งมีลักษณะคล้ายกับรูปที่ 2.1 โดยมีส่วน Stubs ซึ่งเปรียบเสมือน Local Object สำหรับ Client และส่วน Skeletons คือ Server Side Object

จากนั้นในปี 1997 OMG ได้ขยายแนวคิดออกไปเป็น OMA (Object Management Architecture) จาก CORBA 2.0 ซึ่งได้กำหนดคุณสมบัติของ OMG IDL, Language Bindings, Invocation Interfaces, Object Adapters, Interface and Implementation Repositories และ Object Servers ไว้ โดย OMA ได้เพิ่ม CORBAservices, CORBAfacilities และ Application object specification ในส่วน CORBAservices หรือ Object Services นั้น ได้ถูกกำหนดไว้ 16 Services สามารถแบ่งเป็นประเภทได้ 2 ประเภท คือ Service ที่สนับสนุนการคอมพิวเตอร์แบบกระจาย (Distributed Computing) ในองค์กร เช่น Object Trader Service ซึ่งอนุญาตให้ออบเจ็กต์ประกาศ Service โดยการ Register ไว้ แล้ว Client สามารถค้นหา Service ที่ต้องการโดย Text Search และประเภทของ Services อีกประเภทหนึ่งคือ Service ที่สนับสนุน Fined-Grain Objects เช่น การตรวจสอบ (Tracking) เวอร์ชัน และการจัดการเรื่องความเข้ากันได้ (Compatibility) ของออบเจ็กต์ในระบบ

ตัวอย่างของผู้ที่จัดสร้างระบบงานตามมาตรฐาน CORBA เช่น Orbix ของ IONA และ Visibroker ของ Visigenic

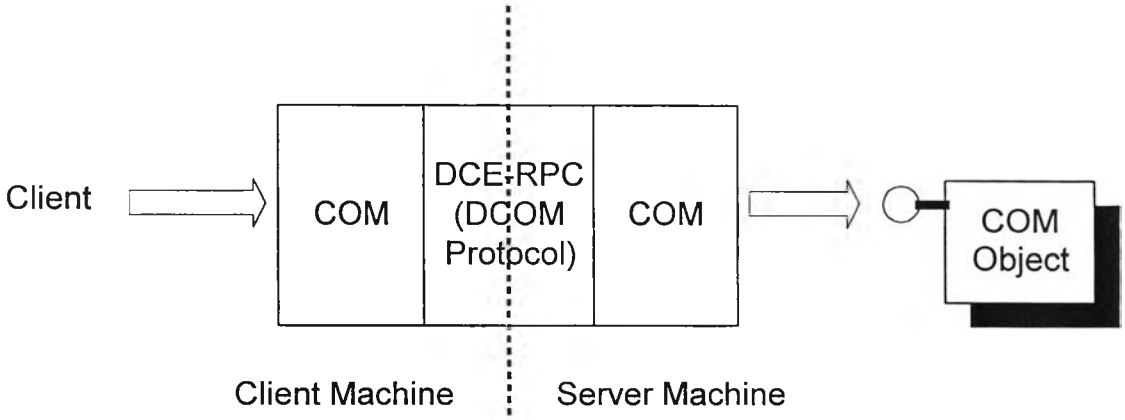
2.2.2.2 ActiveX และ COM ถือกำเนิดจากไมโครซอฟต์ โดยเป็นการสืบสานความสำเร็จจาก VBX และ OLE โดยที่ ActiveX ก็คือชื่อเรียกใหม่ของ OLE ซึ่งผู้อ่านอาจคุ้นเคยจาก Compound Document ซึ่งนิยมใช้ในไมโครซอฟต์ออฟฟิศ ในขณะที่ COM ถูกไมโครซอฟต์พยายามพัฒนาให้เป็นมาตรฐานของสถาปัตยกรรมของส่วนประกอบซอฟต์แวร์เพื่อรองรับส่วนประกอบต่างๆ ซึ่งรวมถึง ActiveX

COM (Component Object Model) เป็นมาตรฐานทางไบนารี (Binary standard) ซึ่งกำหนดในส่วนการต่อประสาน (Interface) ของส่วนประกอบซอฟต์แวร์ โดยใช้ตัวชี้ (Pointer) ชี้ไปที่ Interface และชี้ไปยังตารางที่เก็บตัวแปรฟังก์ชันตาม Method ที่ระบุไว้ของแต่ละภาษาโปรแกรม



รูปที่ 2.3 COM Interface ในระดับไบนารี

จากส่วน Interface Pointer ในรูปที่ 2.3 นั้นจะทำหน้าที่ส่งผ่านพารามิเตอร์ไปยัง Interface Operations (Op ในรูป) ส่วน COM Interface ของออบเจ็กต์นี้เองทำให้ออบเจ็กต์ทำงานต่างๆผ่าน Operations เหล่านี้เชื่อมโยงไปยัง Client ได้ แต่ละ Interface ที่ Client เรียกใช้จะมี Interface Identifiers (IID) ซึ่งมีขนาด 128 บิต และรับประกันว่า Unique ซึ่งสำหรับอ้างอิงในการสร้างโปรแกรม และเป็น Polymorphism



รูปที่ 2.4 ลักษณะของ DCOM

DCOM เป็นมาตรฐานที่ไมโครซอฟต์สร้างซึ่งขยายการใช้งาน COM Object ดังที่กล่าวข้างต้นผ่านข้ามเครื่องต่างๆ (Distributed Computing) ส่วน DCOM Protocol เป็นมาตรฐานซึ่งอาศัยการเชื่อมโยงโดยโปรโตคอลทางเครือข่ายคอมพิวเตอร์เช่น IP เป็นต้น เช่นเดียวกับ CORBA ในส่วน DCOM เองก็มี Services ที่นำเสนอ เช่น Security Service

2.2.2.3 Javabean หรือ Java Components ถือกำเนิดในปี 1996 โดย JavaSoft คำว่า bean หมายถึง กลุ่มของคลาสและทรัพยากร (Class and Resources) ลักษณะสำคัญของ Bean คือ

1. Events ซึ่งใช้ในการติดต่อกับ Bean อื่นๆ โดยการประกาศ Events หรือรอฟัง Events ที่ Bean อื่นๆ ประกาศ

2. Properties คือลักษณะคุณสมบัติของ Bean ซึ่งสามารถถูกเปลี่ยนแปลงโดย Methods ได้ และจากการเปลี่ยนแปลงนั้นสามารถนำไปกำหนดสถานะของ Events ได้

3. Introspection เป็นขบวนการในการค้นหา Properties, Events และ Methods ใน Bean หนึ่งๆซึ่งได้นิยามไว้ ขบวนการนี้อาศัยความสามารถของ Reflection ซึ่งเป็นคุณลักษณะของภาษาโปรแกรม Java ในการตรวจสอบข้อมูลของคลาส

4. Customization เป็นการ Customize Bean Instance โดยผ่านทางการเปลี่ยนแปลง Properties ของ Bean

5. Persistence ช่วยให้สามารถเก็บบันทึก (Save) สถานะการเปลี่ยนแปลง Properties ต่างๆ ไว้ใช้เมื่อต้องการได้

จากที่กล่าวถึง JavaBean นั้นเป็นส่วนประกอบซอฟต์แวร์ในด้าน Client สำหรับในด้าน Server และการเชื่อมสำหรับใช้งานในองค์กรนั้น EJB (Enterprise JavaBeans) ได้ถือกำเนิดขึ้นเพื่อรองรับงานดังกล่าว โดย Client สามารถติดต่อกับ EJB โดยใช้ RMI (Remote Method Invocation) ซึ่งจะสร้าง Proxy Object เป็นตัวกลางในการติดต่อระหว่าง Local และ Remote Object

จากเทคโนโลยีของส่วนประกอบซอฟต์แวร์ที่ได้กล่าวมาข้างต้น มีสิ่งที่เหมือนและแตกต่างกัน ซึ่งอาจทำให้ผู้อ่านมองเห็นภาพได้ชัดเจนขึ้น จะเห็นได้ว่าทั้ง 3 แนวทางนั้นอ้างอิงหลักการของ Late Binding Mechanism, Encapsulation และ Dynamic Polymorphism ซึ่งอาจกล่าวได้เป็นลักษณะเชิงวัตถุ ทางด้านส่วนที่ต่างกัน CORBA นั้นมีมาตรฐานที่รัดกุมในส่วนของมาตรฐานในการรองรับทุกภาษาโปรแกรม และมีนิยาม Service ต่างๆที่ครบถ้วนที่สุด แต่การใช้งานจริงและเครื่องมือที่ใช้ในการพัฒนายังมีน้อยกว่า ในขณะที่ Java นั้นมีข้อได้เปรียบที่สามารถนำไปใช้ในทุกแพลตฟอร์ม ภาษาโปรแกรมที่ใช้จึงควรเป็น Java จึงจะได้ประสิทธิภาพที่สูงกว่า ในด้าน Services นั้นยังมีน้อยกว่า เมื่อเทียบกับ CORBA และ ActiveX ด้านเครื่องมือที่ใช้ในการพัฒนานั้นกำลังมีมากขึ้นเรื่อยๆ จนเกือบจะทัดเทียมกับ ActiveX ทางด้าน ActiveX นั้นมีบทบาทมากเฉพาะในส่วนแพลตฟอร์มของไมโครซอฟต์ แต่มีโมเมนตัมจากเครื่องมือพัฒนา และฐานการตลาดของ ส่วนประกอบซอฟต์แวร์ยุคแรกๆ ที่ยังช่วยเสริมแรงอยู่

2.2 เอกสารและงานวิจัยที่เกี่ยวข้อง

2.2.1 An Internet-based Information System for Cooperative Software Reuse [1] หรือ REGINA Software Bibliothek (RSB-Project) เริ่มขึ้นในปี 1995 เกิดจากความร่วมมือกันระหว่างมหาวิทยาลัยอาร์คเคน และกลุ่ม REGINA ซึ่งประกอบด้วยบริษัทผู้ผลิตส่วนประกอบซอฟต์แวร์จำนวน 30 บริษัท รวมกับหน่วยงานและสถาบันวิจัยในเขตอาร์คเคนจำนวน 10 หน่วยงาน

วัตถุประสงค์เริ่มแรกเพื่อสร้างคลังส่วนประกอบซอฟต์แวร์ส่วนกลางสำหรับสนับสนุนการใช้ส่วนประกอบซอฟต์แวร์ร่วมกัน แต่ไม่ประสบความสำเร็จเท่าที่ควร เนื่องจากซอฟต์แวร์แต่ละส่วนที่มีอยู่ของทั้ง 30 บริษัทซึ่งอยู่ในกลุ่มความร่วมมือ ไม่ได้ถูกพัฒนาขึ้นมาเพื่อ

การนำกลับมาใช้ซ้ำเป็นส่วนประกอบซอฟต์แวร์ ซึ่งกลุ่มความร่วมมือได้ประเมินงานการปรับเปลี่ยนซอฟต์แวร์ในส่วนนี้ไว้น้อยกว่าที่คาดไว้ อีกทั้งไม่มีบริษัทใดทราบแน่ชัดว่าส่วนประกอบซอฟต์แวร์ของตนจะพัฒนาปรับเปลี่ยนขึ้นมาแล้วจะมีงานโปรเจกต์ใดนำไปใช้ ทำให้กลุ่มความร่วมมือเปลี่ยนจุดสนใจไปอยู่ที่การสร้างระบบข้อมูลข่าวสารของส่วนประกอบซอฟต์แวร์บนอินเทอร์เน็ตแทน

ระบบข้อมูลที่กลุ่มความร่วมมือดังกล่าวสร้างขึ้นมีการออกแบบ จัดกลุ่มแบ่งประเภท และแยกลักษณะประจำและข้อมูลของส่วนประกอบซอฟต์แวร์ซึ่งสามารถตรวจสอบได้ผ่านเว็บเพจบนอินเทอร์เน็ต ลักษณะประจำและข้อมูลที่จัดแบ่งไว้ในฐานข้อมูลบนเว็บได้แก่ ประเภทของส่วนประกอบซอฟต์แวร์ โดเมนงานประยุกต์ ลักษณะของผู้จำหน่าย

2.2.2 AGORA: A Search Engine for Software Components [7] กำเนิดขึ้นมาโดยสถาบันวิศวกรรมซอฟต์แวร์ของมหาวิทยาลัยคาร์เนกี เมลอน เพื่อเป็นต้นแบบของ โปรแกรมค้นหา ในการทำดัชนีฐานข้อมูลของซอฟต์แวร์โดยอัตโนมัติ

ข้อมูลใน AGORA มีจำนวนน้อยเมื่อเทียบกับ RSB-Project แต่มีการจัดแบ่งประเภทที่แตกต่างกัน ลักษณะประจำและข้อมูลที่ได้มีการจัดแบ่งไว้ได้แก่ ประเภทของส่วนประกอบซอฟต์แวร์ โดเมนงานอุตสาหกรรม

อนึ่ง สถานะของโครงการ AGORA ในปัจจุบันได้หยุดการให้บริการทางเว็บเพจทำให้งานวิจัยต้องรวบรวมข้อมูลจากทางเอกสารเผยแพร่เท่านั้น

2.2.3 THE SOFTWARE COMPONENT MARKET ON THE INTERNET CURRENT STATUS AND CONDITIONS FOR GROWTH [9] งานวิจัยนี้มุ่งที่จะนำเสนอสภาพตลาดของส่วนประกอบซอฟต์แวร์โดยได้ใช้ โปรแกรมค้นหา ในการค้นหาผู้จำหน่ายทั้งสิ้น 38 ราย โดยมีทั้งกลุ่มตัวแทนจำหน่าย กลุ่มผู้ให้ข้อมูลส่วนประกอบซอฟต์แวร์ และผู้ผลิตส่วนประกอบซอฟต์แวร์ พร้อมทั้งได้รวบรวมข้อมูลลักษณะประจำและข้อมูลของส่วนประกอบซอฟต์แวร์และผู้จำหน่าย ได้แก่ ชนิดของส่วนประกอบซอฟต์แวร์ ประเภทของผู้จำหน่าย ลักษณะเอกสารของส่วนประกอบซอฟต์แวร์

อนึ่งงานวิจัยนี้ยังได้ให้ข้อเสนอแนะบทบาทและทิศทางของตลาดของส่วนประกอบซอฟต์แวร์ที่จะทำให้ตลาดมีการขยายตัวมากขึ้น และได้อ้างถึงการจัดอันดับข้อเสนอแนะดังกล่าวโดยใช้การสำรวจความคิดเห็นของผู้เชี่ยวชาญในสายงานส่วนประกอบซอฟต์แวร์โดยมีผู้

สนใจให้ความร่วมมือจากทั่วโลกทั้งสิ้นประมาณ 35 ราย โดยได้นำเสนอเงื่อนไขซึ่งได้จากผลการสำรวจเช่น

ผู้จำหน่ายจะต้องทำให้เกิดความสะดวกและง่ายในการประเมินส่วนประกอบซอฟต์แวร์โดยนำเสนอข้อมูลต่างๆ

ข้อกำหนดหรือลักษณะประจำและข้อมูลของส่วนประกอบซอฟต์แวร์จะต้องสอดคล้องตามมาตรฐาน

ตลาดของส่วนประกอบซอฟต์แวร์ต้องเน้นไปทางการนำไปใช้อีกครั้งแบบ black-box และมีมาตรการรักษาผลประโยชน์และลิขสิทธิ์ของผู้ผลิต

2.2.4 CBDiforum Report: Buying Software Components [6] รายงานฉบับนี้พยายามผลักดันให้มีการใช้ส่วนประกอบซอฟต์แวร์ นำเสนอสภาพตลาดและการประเมินนำส่วนประกอบซอฟต์แวร์มาใช้ โดยได้มีการจัดแบ่งข้อมูล ลักษณะประจำและข้อมูลและการประเมินออกเป็น 5 กลุ่มได้แก่

กลุ่มโดเมน ประกอบด้วย ลักษณะประจำและข้อมูลตามหน้าที่ ลักษณะประจำและข้อมูลอื่น การวิเคราะห์ช่องว่างหรือระยะห่างเทียบกับความต้องการ (Gap Analysis) ลักษณะสถาปัตยกรรม และลักษณะตามมาตรฐานของธุรกิจ

กลุ่มการทำให้เกิดผล ประกอบด้วย ขั้นตอนการติดตั้ง การรวมเบ็ดเสร็จ การทดสอบ คุณภาพและการรับรอง การปรับแต่งให้ตรงความต้องการ และความมั่นคง

กลุ่มการบำรุงรักษา ประกอบด้วย การบำรุงรักษาในส่วนโปรแกรมต่อประสานและการบำรุงรักษาในส่วนการปรับแต่งให้ตรงความต้องการ

กลุ่มข้อมูลเชิงพาณิชย์ ประกอบด้วย การเข้าถือสิทธิ์ ค่าใช้จ่าย ลิขสิทธิ์ การสนับสนุน และการชำระเงิน

กลุ่มข้อมูลทางเทคนิค ประกอบด้วย ระบบปฏิบัติการ แพลตฟอร์ม เทคโนโลยีของส่วนประกอบซอฟต์แวร์ ข้อกำหนดสำหรับการพัฒนา และข้อมูลการทดสอบประสิทธิภาพ

2.2.5 Identifying and Qualifying Reusable Software Components [4] งานวิจัยนี้ได้นำเสนอขั้นตอนในการคัดเลือกและประเมินส่วนประกอบซอฟต์แวร์ การประเมินดังกล่าว

ต้องพิจารณาตามลักษณะประจำและข้อมูลของส่วนประกอบซอฟต์แวร์ซึ่งประกอบด้วยประโยชน์ ใช้สอย ค่าใช้จ่าย และคุณภาพ

2.2.6 Software Licensing Models in the EDA Industry [3] รายงานฉบับนี้ได้กล่าวถึงโมเดลของลิขสิทธิ์ซอฟต์แวร์ซึ่งมีใช้อยู่ในอุตสาหกรรมอีดีเอ และการเลือกใช้โดยพิจารณาจากเงื่อนไขพร้อมทั้งยกตัวอย่างกรณีศึกษาประกอบ ตัวอย่างโมเดลลิขสิทธิ์ได้แก่ สิขสิทธิ์แบบตามสถานีเชื่อมโยง (Workstation) ลิขสิทธิ์แบบลอยตัว (Floating) ลิขสิทธิ์แบบเฉพาะแห่ง (Site) ลิขสิทธิ์แบบให้ทั้งองค์กร (Enterprise) ลิขสิทธิ์แบบเช่า (Rental) ลิขสิทธิ์แบบคิดต่อจำนวนการใช้ (Per Usage)

2.2.7 Component Software Beyond Object-Oriented Programming [2] หนังสือเล่มนี้กล่าวถึงพื้นฐานทั่วไปของเทคโนโลยีของส่วนประกอบซอฟต์แวร์ ลักษณะเชิงวัตถุ ปัจจัยที่ต้องคำนึงถึงเมื่อสร้างงานจากส่วนประกอบซอฟต์แวร์ โครงสร้างสถาปัตยกรรมในรูปแบบของ ActiveX Java และ CORBA พร้อมทั้งแนวทางในอนาคต รวมถึงแง่มุมด้านการตลาด

2.2.8 K-BACEE: A Knowledge-Based Automated Component Ensemble Evaluation Tool [8] งานวิจัยดังกล่าวนำเสนอรูปแบบการสร้างระบบเก็บและค้นหาข้อมูลสำหรับช่วยในการประเมินนำส่วนประกอบซอฟต์แวร์มาใช้โดยใช้ XML ในการทำระบุและค้นหารายละเอียดจาก Repository แล้วจัดกลุ่มโดยใช้ JRules ช่วยในการประเมิน

หลังจากที่ได้นำเสนอบทนำ และ เนื้อหาในบทที่ 2 เอกสารและงานวิจัยที่เกี่ยวข้องเรียบร้อยแล้ว ในบทที่ 3 จะได้กล่าวถึงเนื้อหาของขั้นตอนการทำวิจัยโดยละเอียดต่อไป