

บทที่ 2

แนวความคิดและผลงานวิจัยที่เกี่ยวข้อง

2.1 ลักษณะพิเศษของซอฟต์แวร์คอมพิวเตอร์

จากลักษณะพิเศษของสินค้าสาธารณะ (Public goods) 2 ประการ¹ ซึ่งสอดคล้องกับคุณสมบัติพิเศษของซอฟต์แวร์คอมพิวเตอร์ คือ

1. หลักของการยกเว้น (Principle of Exclusivity)

เนื่องจากซอฟต์แวร์คอมพิวเตอร์นั้นง่ายต่อการทำสำเนา (Copy) ขณะที่อุปกรณ์ที่ใช้เป็นตัวกลาง (Media) ในการทำสำเนา เช่น แผ่นบันทึกข้อมูล (Diskett), แถบบันทึกข้อมูล (Tape Backup) นั้นมีราคาค่อนข้างต่ำ ซอฟต์แวร์ต้นฉบับ (Master) เพียงหนึ่งชุดสามารถทำสำเนาได้โดยไม่มีสิ้นสุด เนื่องจากซอฟต์แวร์คอมพิวเตอร์เป็นสัญญาณแม่เหล็กที่ถูกบันทึกไว้ในแผ่นบันทึกข้อมูล การทำสำเนาก็สามารถทำได้เพียงบันทึกสัญญาณแม่เหล็กลงในแผ่นบันทึกข้อมูลเปล่าเท่านั้น ตัวกลางที่ทำสำเนาไปแล้วนั้นก็จะมีคุณสมบัติทุกประการ เหมือนกับซอฟต์แวร์ต้นฉบับ การผลิตหรือทำสำเนาของซอฟต์แวร์คอมพิวเตอร์จึงเท่ากับการทำให้เกิดต้นฉบับที่จะสามารถนำไปทำสำเนาได้อย่างไม่จำกัด การที่ผู้ใช้คอมพิวเตอร์รายหนึ่งนำซอฟต์แวร์คอมพิวเตอร์ดังกล่าวไปใช้ จากนั้นทำการสำเนาซอฟต์แวร์ดังกล่าวให้กับผู้ใช้รายอื่น ผู้ใช้รายแรกก็ไม่ได้เสียประโยชน์จากการใช้งานซอฟต์แวร์คอมพิวเตอร์ของผู้ใช้รายอื่น

ข้อในการพิจารณาดังกล่าวใช้ได้กับซอฟต์แวร์คอมพิวเตอร์ที่ไม่ได้ทำการป้องกันการทำสำเนา (Non-copy Protection) และซอฟต์แวร์ที่ไม่ต้องใช้กับอุปกรณ์ประกอบอื่น (Hardware Independent Software) ซึ่งซอฟต์แวร์เชิงธุรกิจจากต่างประเทศแทบทั้งหมดในปัจจุบัน มักไม่ทำการป้องกันการทำสำเนาเอาไว้

¹Edwin Mansfield, Microeconomics: Theory and Applications, 2nd ed. (New York: W. W. Norton & Company, Inc., 1975) p. 497-498.

2. ต้นทุนหน่วยเพิ่มของผู้ใช้รายต่อ ๆ ไปเป็นศูนย์ (Marginal cost for additional users is zero) จากคุณสมบัติที่สามารถทำสำเนาได้อย่างถูกต้อง และไม่มีจำกัดนี้ ทำให้ต้นทุนเพิ่มของผู้ใช้รายต่อ ๆ ไปต่ำมากจนอาจกล่าวได้ว่าเป็นศูนย์ ซึ่งหากพิจารณาเพียงประเด็นนี้ สินค้าประเภทซอฟต์แวร์คอมพิวเตอร์ก็น่าจะกล่าวได้ว่าเป็นสินค้าสาธารณะ (Public good) ได้ ซึ่งหากซอฟต์แวร์คอมพิวเตอร์ไม่ได้พัฒนาขึ้นเพื่อแสวงหาผลกำไร (Public domain) ก็อาจกล่าวได้ว่าการทำสำเนาซอฟต์แวร์ดังกล่าวทำให้ผลประโยชน์ส่วนเพิ่มของสังคม (Marginal social benefit) สูงขึ้น แต่ทั้งนี้ เนื่องจากการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์ไม่ได้ดำเนินการอยู่เฉพาะในหน่วยงานดังกล่าว โดยเฉพาะอย่างยิ่งธุรกิจผู้ผลิตซอฟต์แวร์คอมพิวเตอร์ เป็นแรงกระตุ้นให้เกิดการแข่งขัน ทำให้เกิดการเปลี่ยนแปลงในเทคโนโลยีให้เร็วไปอย่างรวดเร็ว บริษัทหรือหน่วยธุรกิจที่ทำการผลิตซอฟต์แวร์คอมพิวเตอร์จึงเป็นผู้เสียประโยชน์ ทั้งนี้เนื่องจากการทำสำเนาซอฟต์แวร์คอมพิวเตอร์เพิ่มขึ้นหนึ่งชุดก็เท่ากับบริษัทผู้ผลิตซอฟต์แวร์คอมพิวเตอร์ดังกล่าวสูญเสียรายได้จากการจำหน่ายซอฟต์แวร์ไปนั่นเอง

2.2 ทฤษฎีทางเศรษฐศาสตร์ที่เกี่ยวข้อง

รูปแบบตลาดและการแข่งขันในอุตสาหกรรมการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์เป็นลักษณะของตลาดที่มีการแข่งขันกึ่งผูกขาด (Monopolistic Competition) ทั้งนี้เนื่องจากลักษณะพิเศษ² ประการคือ

1. ผู้ผลิตหรือผู้ขายหลายราย (Large number of Firm or Seller) ถึงแม้ว่าการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์จะเป็นอุตสาหกรรมที่ต้องใช้เทคโนโลยีในการผลิตสูง แต่เทคโนโลยีดังกล่าวมักจะอยู่ในตัวแรงงานหรือบุคคลากรเป็นส่วนใหญ่

²Willis L Peterson, Principles of Economics : Micro. 5th ed. (Illinois: Richard D. Irwin, Inc., 1983) p. 199-202.

2. ความแตกต่างกันในลักษณะของสินค้าหรือผลิตภัณฑ์ (Product Differentiation) ซอฟต์แวร์คอมพิวเตอร์แต่ละตัวมีลักษณะที่แตกต่างกัน แม้ว่าจะเป็นซอฟต์แวร์คอมพิวเตอร์ประเภทเดียวกัน

3. กลุ่มของสินค้าที่มีลักษณะใกล้เคียงกันหรือมีลักษณะทดแทนกันได้ (Group Product or Substituted Product) ลักษณะดังกล่าวปรากฏอยู่ในอุตสาหกรรมการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์ ซึ่งหากจะทำการจำแนกตามประเภทของซอฟต์แวร์ (Software Classification) ซอฟต์แวร์ในแต่ละประเภทมีลักษณะของการทดแทนกันได้อย่างใกล้ชิด (Close Substituted) เช่น ผลิตภัณฑ์ในกลุ่มซอฟต์แวร์ประเภทตารางคำนวณอิเล็กทรอนิกส์ (Electronic Spreadsheet) ซึ่งมีวัตถุประสงค์เพื่อที่จะช่วยในการคำนวณซึ่งทำงานในลักษณะของแถว (row) และแนว (column) ถึงแม้ว่าผลิตภัณฑ์แต่ละชิ้นในกลุ่มซอฟต์แวร์ดังกล่าวจะมีลักษณะที่แตกต่างกันไป แต่เมื่อพิจารณาโดยใช้ลักษณะพื้นฐาน (Basic Feature) ของการใช้งานและประโยชน์การใช้งานของซอฟต์แวร์แต่ละตัวแล้ว ก็จะพบว่ามีความใกล้เคียงกันและสามารถทดแทนกันได้ได้อย่างใกล้ชิด

4. ค่าใช้จ่ายในด้านโฆษณา (Advertising Expense) ซึ่งจำเป็นอย่างยิ่งสำหรับสินค้าในตลาด Monopolistic Competition เพื่อแสดงความแตกต่างในตัวผลิตภัณฑ์ อันเป็นลักษณะของการโฆษณาประชาสัมพันธ์ทางด้านข่าวสาร (Information Advertising) ซึ่งจำเป็นอย่างยิ่งสำหรับผลิตภัณฑ์ที่ต้องใช้เทคโนโลยีในการผลิตสูงและต้องใช้ความรู้ของผู้ใช้พอสมควร ตลอดจนลักษณะการเปลี่ยนแปลงเชิงพลวัต (Dynamic) ค่อนข้างรวดเร็วดังเช่นอุตสาหกรรมการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์

จากลักษณะทั้งสี่ประการของอุตสาหกรรมการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์ หากพิจารณาลักษณะโดยรวมแล้ว ทุกประเด็นสอดคล้องกับลักษณะของตลาด Monopolistic Competition อย่างชัดเจน แต่หากพิจารณาถึงประเด็นของกลุ่มสินค้าที่มีความใกล้เคียงในการทดแทนกัน (Close Substituted Product Group) ในหลาย ๆ กลุ่มซอฟต์แวร์พบว่ามักมีผู้ขายหรือผู้ผลิตเพียงไม่กี่ราย และแต่ละรายก็มักจะมีส่วนแบ่งในตลาดของซอฟต์แวร์ประเภทนั้น ๆ ค่อนข้างมาก ซึ่งเป็นลักษณะของตลาดผู้ขายน้อยราย (Oligopoly) แฝงอยู่

2.3 อุปสงค์และอุปทานของซอฟต์แวร์คอมพิวเตอร์



อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ (Demand for Computer Software)

เนื่องจากอุตสาหกรรมการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์มีลักษณะกึ่งแข่งขันกึ่งผูกขาด (Monopolistic Competition) อุปสงค์ที่แต่ละหน่วยผลิตหรือหน่วยธุรกิจ ในอุตสาหกรรมดังกล่าวเผชิญจึงเป็นอุปสงค์ที่มีลักษณะทอดลงจากซ้ายไปขวา

ขณะที่คอมพิวเตอร์และซอฟต์แวร์คอมพิวเตอร์ เป็นสิ่งที่จำเป็นต้องใช้ควบคู่กันไป (Complement Goods) อุปสงค์ที่มีต่อคอมพิวเตอร์ย่อมส่งผลให้เกิดอุปสงค์ต่อเนื่อง (Derived Demand) ในซอฟต์แวร์คอมพิวเตอร์ด้วย ทั้งนี้เนื่องจากคอมพิวเตอร์เครื่องหนึ่ง ๆ สามารถใช้ซอฟต์แวร์ได้มากมายหลายประเภท ขึ้นอยู่กับวัตถุประสงค์ของการใช้งาน จึงสรุปได้ว่าอุปสงค์ที่เกิดขึ้นต่อเครื่องคอมพิวเตอร์ส่วนบุคคลหนึ่ง เครื่องส่งผลก่อให้เกิดอุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์จำนวนมาก

อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์สามารถแยกพิจารณาได้เป็น 2 ประเภท คือ

1. อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ของหน่วยธุรกิจ ซึ่งปัจจุบันหน่วยธุรกิจจำนวนมากได้นำคอมพิวเตอร์เข้ามาประยุกต์ใช้กับงานต่าง ๆ โดยเฉพาะอย่างยิ่ง หากธุรกิจดังกล่าวมีการนำระบบสารสนเทศโดยใช้ระบบคอมพิวเตอร์ (Computer-base Information System) หรือระบบการจัดการและการตัดสินใจทางธุรกิจโดยใช้คอมพิวเตอร์ (Computer-base Management and Decision System) หรือแม้แต่การนำอุปกรณ์คอมพิวเตอร์และเครื่องมือต่าง ๆ ที่เกี่ยวข้องเข้ามาใช้เป็นอุปกรณ์สำนักงานอัตโนมัติ (Office Automation - OA) ซึ่งโดยทั่วไปมีความต้องการใช้ซอฟต์แวร์ธุรกิจก็จะขึ้นอยู่กับลักษณะงานรวมทั้งรูปแบบของงานที่จะนำคอมพิวเตอร์เข้ามาประยุกต์ใช้ ซึ่ง ล้ำฝั่งแต่ซอฟต์แวร์ประยุกต์ (Application Software) ที่มีจำหน่ายในท้องตลาดในลักษณะซอฟต์แวร์สำเร็จรูป (Application Software Package, Off-the-shelf Software) อาจไม่สามารถตอบสนองกับลักษณะงานที่ปฏิบัติได้จริง จึงอาจจำเป็นต้องมีการปรับปรุงซอฟต์แวร์สำเร็จรูปดังกล่าว (Altered Off-the-shelf Software : Customized Package Software) ให้เหมาะสมกับระบบงานมากขึ้น ซึ่งหากซอฟต์แวร์ดังกล่าวยังไม่สามารถปรับใช้กับงานได้อย่างเหมาะสมเพียงพอ หน่วย

ธุรกิจอาจจำเป็นต้องจัดหาซอฟต์แวร์คอมพิวเตอร์โดยการพัฒนาซอฟต์แวร์ขึ้นมาใช้เอง รวมทั้งว่าจ้างบริษัทผลิตซอฟต์แวร์ (Software House) หรือโปรแกรมเมอร์ให้พัฒนาให้ (Custom-built/ tailored Software) ทางเลือกในการจัดหาซอฟต์แวร์ประเภทต่าง ๆ แสดงเปรียบเทียบได้ดังตาราง³

ตารางที่ 2.1 ตารางแสดงทางเลือกในการจัดหาซอฟต์แวร์ประเภทต่าง ๆ

	Package Software	Customized Software	Tailor-made Software
ระยะเวลาในการพัฒนา	เร็วที่สุด	ขึ้นอยู่กับเงื่อนไขในการเปลี่ยนแปลง	ช้าที่สุด
ความสามารถในการเข้ากันได้กับระบบงาน	บางที่อาจไม่	เป็นไปได้	แทบจะรับประกันได้
ต้นทุน	ถูกที่สุด	อยู่ระหว่างกลาง	แพงที่สุด

อย่างไรก็ตาม ซอฟต์แวร์ประยุกต์สำเร็จรูป (Application Software Package) ก็ยังมีความจำเป็นอย่างมากสำหรับงานธุรกิจโดยทั่วไป ซึ่งซอฟต์แวร์ธุรกิจ (Businessware) ดังกล่าวสามารถจำแนกได้เป็นประเภทดังนี้⁴

³David M. Kroenke and Kathleen A. Dolan, Business Computer Systems: An Introduction, 3rd ed. (Singapore: McGraw-Hill Book Co., 1989), p. 156.

⁴กองบรรณาธิการ, "ส่องโลกซอฟต์แวร์ทางธุรกิจ" ไมโครคอมพิวเตอร์ 79 (กุมภาพันธ์ 2535) : 200-205.

1. ซอฟต์แวร์ช่วยในการตัดสินใจและแก้ปัญหาทางธุรกิจ ประกอบไปด้วย

- 1.1 ซอฟต์แวร์การจัดการข้อมูล (Data Management)
- 1.2 ซอฟต์แวร์สร้างรูปแบบจำลองทางธุรกิจ (Modeling)
- 1.3 ซอฟต์แวร์การวิเคราะห์ทางสถิติ
(Statistical Analysis)
- 1.4 ซอฟต์แวร์งานวางแผน (Planning)
- 1.5 ซอฟต์แวร์เรียกค้นข้อมูล (Inquiry)
- 1.6 ซอฟต์แวร์กราฟิกทางธุรกิจ (Business Graphic)

2. ซอฟต์แวร์ที่ช่วยเพิ่มผลผลิตและประสิทธิภาพในการทำงานระดับบุคคล และซอฟต์แวร์จัดการสำนักงานอัตโนมัติ (Office Automation)

- 2.1 ซอฟต์แวร์ประมวลคำ (Word Processing)
- 2.2 ซอฟต์แวร์ตารางอิเล็กทรอนิกส์ (Electronic Spreadsheet)
- 2.3 ซอฟต์แวร์จัดการฐานข้อมูล (Database Management)
- 2.4 ซอฟต์แวร์กราฟิก (Graphic)
- 2.5 ซอฟต์แวร์ระบบสื่อสาร (Communication)
- 2.6 ซอฟต์แวร์งานพิมพ์ตั้งโต๊ะ (Desktop Publishing)
- 2.7 ซอฟต์แวร์ระบบบัญชี (Accounting)

3. ซอฟต์แวร์จัดการสารสนเทศที่ใช้ในองค์กร

อุปสงค์ต่อซอฟต์แวร์ธุรกิจส่วนใหญ่จะเป็นอุปสงค์ที่มีต่อซอฟต์แวร์คอมพิวเตอร์ที่สามารถแสดงผลเป็นภาษาไทยได้แทบทั้งสิ้น โดยเฉพาะอย่างยิ่ง ในกลุ่มซอฟต์แวร์ที่ช่วยเพิ่มผลผลิตและประสิทธิภาพในการทำงานระดับบุคคลและซอฟต์แวร์สำหรับการจัดการสำนักงานอัตโนมัติ (Office Automation) เช่น งานในระบบบัญชี งานประมวลคำ งานจัดการฐานข้อมูล เป็นต้น

จากที่กล่าวมาข้างต้น อาจสรุปได้ว่า อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ของหน่วยธุรกิจ ปัจจัยที่สำคัญที่สุดก็คือ ปริมาณงานของหน่วยธุรกิจ ซึ่งก็แน่นอนว่าการนำ

คอมพิวเตอร์มาใช้ นั้น คงไม่สามารถประยุกต์ใช้ได้กับงานทั้งหมดทุกประเภทของหน่วยธุรกิจ อย่างไรก็ตาม งานในสำนักงาน งานการบัญชี งานจัดการเอกสาร และงานที่มีลักษณะเดียวกัน ที่ต้องทำซ้ำ ๆ อยู่เป็นประจำ งานที่ต้องการความเที่ยงตรงนับไวหรืองานจัดการในสำนักงาน งานเหล่านี้มีแนวโน้มที่จะนำไมโครคอมพิวเตอร์เข้ามาใช้เพิ่มขึ้นเรื่อย ๆ ไม่ว่าจะเป็นการนำมาใช้ในลักษณะเครือข่าย (Area Network) หรือการนำมาใช้เป็นคอมพิวเตอร์ตั้งโต๊ะส่วนบุคคลธรรมดา (Stand-alone) อุปสงค์ที่เพิ่มขึ้นจากการนำคอมพิวเตอร์เข้ามาใช้งานก็ย่อมหมายถึงการอุปสงค์ต่อการใช้งานซอฟต์แวร์คอมพิวเตอร์ที่เพิ่มขึ้นด้วย ดังนั้นจึงอาจกล่าวโดยสรุปได้ว่า การเพิ่มขึ้นของกิจกรรมทางเศรษฐกิจย่อมส่งผลให้อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์เพิ่มมากขึ้นด้วย

2. อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ของสถาบันที่ไม่ใช่หน่วยธุรกิจ ซึ่งอาจแยกได้เป็น หน่วยงานในภาครัฐบาล สถาบันการศึกษา และบุคคลธรรมดาทั่วไป ทั้งนี้ หน่วยงานในภาครัฐบาลและสถาบันการศึกษา ก็เช่นเดียวกับหน่วยธุรกิจที่มีความจำเป็นต้องจัดการงานในสำนักงาน งานระดับบุคคล ระบบงานจัดการฐานข้อมูล และระบบงานเอกสารต่าง ๆ ทำให้เกิดความต้องการใช้คอมพิวเตอร์ในหน่วยงานเหล่านี้ อย่างไรก็ตาม การนำซอฟต์แวร์คอมพิวเตอร์มาใช้หน่วยงานเหล่านี้มักจะเป็นซอฟต์แวร์ที่มีมาให้พร้อมกับเครื่องมืออุปกรณ์คอมพิวเตอร์หรือ ซอฟต์แวร์สาธารณะ (Public Domain Software) หรือซอฟต์แวร์ที่หน่วยงานนั้น ๆ หรือหน่วยงานที่เกี่ยวข้องทำการพัฒนาขึ้นใช้ ทั้งนี้ เนื่องจากระบบการจัดซื้อ และขั้นตอนต่าง ๆ ที่ค่อนข้างยุ่งยากและซับซ้อนกว่าหน่วยงานธุรกิจ การปรับตัวในการใช้ซอฟต์แวร์คอมพิวเตอร์จึงเป็นไปได้อย่างล่าช้ากว่าหน่วยธุรกิจ ขณะที่ความต้องการใช้ซอฟต์แวร์คอมพิวเตอร์ส่วนบุคคลก็มีลักษณะที่ใกล้เคียงกัน การนำเอาซอฟต์แวร์คอมพิวเตอร์เข้ามาใช้สำหรับงานบุคคลยังคงมีอยู่ เช่น ซอฟต์แวร์สำหรับจัดการงานเอกสาร ซอฟต์แวร์จัดการตารางเวลาดนหมาย ซอฟต์แวร์ช่วยในการคำนวณและงานบัญชีส่วนบุคคล รวมทั้งซอฟต์แวร์ที่ให้ความบันเทิง เช่น ซอฟต์แวร์เกมส์ (Games Software) ต่าง ๆ ด้วย เป็นต้น

ประเด็นที่สำคัญ อันเป็นลักษณะเด่นพิเศษของอุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ก็คือ อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์มีลักษณะของการเปลี่ยนแปลงที่รวดเร็วมากจนอาจกล่าวได้ว่าอุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์เป็นอุปสงค์เชิงพลวัต (Dynamic Demand) ทั้งนี้ส่วนหนึ่งอาจเกี่ยวข้องกับลักษณะ โดยธรรมชาติของอุปสงค์ที่มีความเกี่ยวข้องกับรสนิยม (Taste) ของผู้ซื้อค่อนข้างมาก การที่มีซอฟต์แวร์ชนิดใหม่ที่มีคุณสมบัติพิเศษกว่าซอฟต์แวร์

ที่มีอยู่ในขณะนั้น หรือการเปลี่ยนแปลงในคุณสมบัติของซอฟต์แวร์ที่มีอยู่เดิม (Reversion) ส่งผลเป็นอย่างมากต่ออุปสงค์ของซอฟต์แวร์อื่น ๆ ช่วงชีวิต (Life Cycle) ของสินค้าซอฟต์แวร์คอมพิวเตอร์มักจะสั้นมาก เมื่อเปรียบเทียบกับสินค้าอุตสาหกรรมประเภทต่าง ๆ

อุปทานของซอฟต์แวร์คอมพิวเตอร์ (Supply of Computer Software)

ซอฟต์แวร์คอมพิวเตอร์สำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลที่สามารถจัดหามาเพื่อใช้งานในปัจจุบัน สามารถแบ่งได้เป็น 2 ประเภทใหญ่ ๆ คือ

1. ซอฟต์แวร์คอมพิวเตอร์ที่แสดงผลเป็นภาษาไทย ซึ่งแทบทั้งหมดผลิตและพัฒนาขึ้นโดยผู้ผลิตภายในประเทศ เพื่อสนองตอบต่อความต้องการใช้ซอฟต์แวร์ที่เป็นภาษาถิ่น (Local Language Software) ซึ่งมีทั้งซอฟต์แวร์ที่ผลิตเพื่อจำหน่ายในเชิงการค้าโดยบริษัทหรือหน่วยผลิตต่าง ๆ และซอฟต์แวร์ที่ทำการพัฒนาขึ้นเพื่อเผยแพร่แก่สาธารณชนโดยไม่คิดมูลค่า (Public Domain Software) ทั้งนี้ผู้ผลิตอาจเป็นได้ทั้งหน่วยงานของรัฐ สถาบันการศึกษา หรือแม้แต่หน่วยงานในภาคเอกชนเองด้วย ซอฟต์แวร์คอมพิวเตอร์ที่แสดงผลเป็นภาษาไทยนี้เป็นซอฟต์แวร์ประยุกต์ทั้งสิ้น (Application Software)

2. ซอฟต์แวร์คอมพิวเตอร์ที่แสดงผลเป็นภาษาต่างประเทศ ซึ่งมักได้แก่ซอฟต์แวร์ภาษาสากล เช่น ซอฟต์แวร์ภาษาอังกฤษ อุปทานของซอฟต์แวร์ดังกล่าวมีทั้งจากภายในประเทศและภายนอกประเทศ ซึ่งซอฟต์แวร์ประเภทนี้มักจะเป็นซอฟต์แวร์ที่ถูกนำเข้ามาจากต่างประเทศเป็นส่วนใหญ่ ทั้งในลักษณะของการนำเข้ามาจำหน่ายโดยได้รับอนุญาตจากเจ้าของซอฟต์แวร์นั้น ๆ หรือนำเข้าโดยผิดกฎหมายซึ่งเป็นการละเมิดสิทธิของเจ้าของและผู้ผลิต (Software Piracy) ซอฟต์แวร์เหล่านี้มีมากมายหลายประเภท ทั้งซอฟต์แวร์ประยุกต์ (Application Software) และซอฟต์แวร์จัดการระบบ (Operating System Software) ซึ่งเป็นที่น่าสังเกตว่า ในส่วนของซอฟต์แวร์ประเภท Operating System ซึ่งมีความจำเป็นต่อเครื่องคอมพิวเตอร์ส่วนบุคคลทุกเครื่องที่มีใช้อยู่ในปัจจุบันเป็นซอฟต์แวร์ที่ต้องนำเข้ามาจากต่างประเทศทั้งสิ้น

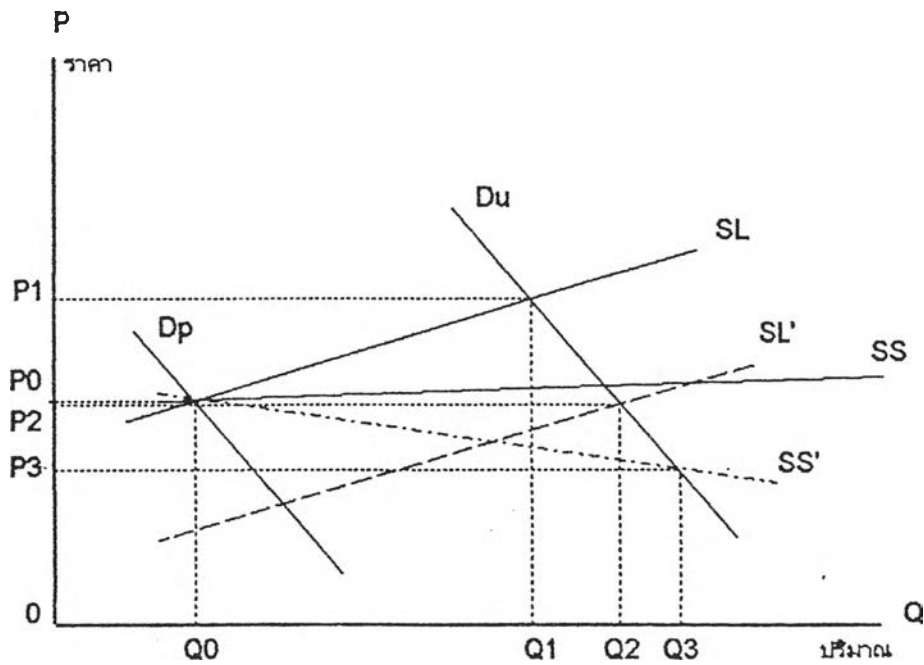
ขณะที่ซอฟต์แวร์คอมพิวเตอร์สามารถทำสำเนาได้อย่างง่ายดาย การนำเอาซอฟต์แวร์คอมพิวเตอร์มาใช้โดยไม่ได้รับอนุญาตจากบริษัทผู้ผลิต จึงเป็นเรื่องที่สามารถกระทำได้อย่างง่ายดาย ดังที่กำลังเป็นอยู่ในปัจจุบัน โดยเฉพาะกับซอฟต์แวร์คอมพิวเตอร์จากต่างประเทศ ซึ่งหากซื้อจากบริษัทผู้ผลิตหรือตัวแทนจำหน่ายจะมีราคาจำหน่ายค่อนข้างสูง การละเมิดสิทธิ์จึงมีอยู่อย่างค่อนข้างแพร่หลาย ปัญหานี้ทำให้ผู้ผลิต/พัฒนาซอฟต์แวร์คอมพิวเตอร์ภาษาไทยเมื่อจำหน่ายจำนวนมากหาทางแก้ไข โดยจัดทำอุปกรณ์พิเศษเพื่อใช้คู่กับซอฟต์แวร์ดังกล่าว ซึ่งหากไม่มีอุปกรณ์ดังกล่าวซอฟต์แวร์นั้น ๆ ก็จะไม่สามารถทำงานได้ ซอฟต์แวร์ที่จำเป็นต้องใช้ร่วมกับอุปกรณ์ประกอบนี้ (Hardware-dependent Software) เป็นหนทางหนึ่งที่ทำให้การละเมิดสิทธิ์บรรเทาลง แต่ก็เป็นการเพิ่มต้นทุนในการผลิตของซอฟต์แวร์ขึ้น ซึ่งก็แน่นอนว่าราคาจำหน่ายของซอฟต์แวร์ประเภทนี้ย่อมมีราคาสูงขึ้นด้วย และบางครั้งกลับดูเหมือนว่าซอฟต์แวร์ดังกล่าวเป็นสิ่งที่แถมมาเมื่อซื้ออุปกรณ์ดังกล่าวมาใช้ เช่น ซอฟต์แวร์ประมวลคำที่ต้องใช้ร่วมกับแผ่นวงจรแสดงผล (Display Card) เป็นต้น

อุปสงค์และอุปทานที่เกิดขึ้นจริง

เนื่องจากลักษณะพิเศษของสินค้าประเภทซอฟต์แวร์คอมพิวเตอร์ที่สามารถทำการสำเนาได้อย่างง่ายดาย ซึ่งไม่ว่าด้วยเหตุผลใดก็ตาม ทำให้มีผู้ใช้ซอฟต์แวร์คอมพิวเตอร์จำนวนมากทำการสำเนา โปรแกรมคอมพิวเตอร์มาใช้โดยไม่ยอมซื้อหาจากผู้ผลิต ทำให้เกิดลักษณะของอุปสงค์ที่มีต่อซอฟต์แวร์คอมพิวเตอร์ใน 2 ลักษณะคือ

1. อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ของผู้ซื้อ (Purchasers' Demand for Software) อุปสงค์ดังกล่าวเป็นอุปสงค์ที่ผู้ผลิตเผชิญอยู่ ซึ่งหากกล่าวโดยรวมอุปสงค์ดังกล่าวเมื่อพิจารณาถึงด้านอุปทานของซอฟต์แวร์ประกอบ เป็นตัวกำหนดระดับราคาตลาด (Market Price)
2. อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ของผู้ใช้ (Users' Demand for Software) อุปสงค์ประเภทนี้ครอบคลุมทั้งอุปสงค์ต่อซอฟต์แวร์ของผู้ซื้อ และอุปสงค์ต่อซอฟต์แวร์ที่ไม่ได้ทำการซื้อ อันได้แก่ ซอฟต์แวร์สาธารณะ (Public Domain Software) ซอฟต์แวร์ประเภท Shareware หรือแม้แต่ซอฟต์แวร์ที่ทำการสำเนาโดยไม่ถูกต้อง (Pirate Software)

ภาพที่ 2.1 อุปสงค์และอุปทานของซอฟต์แวร์คอมพิวเตอร์



- D_p = อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ของผู้ซื้อ
 D_u = อุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์ของผู้ใช้
 SS = อุปทานของซอฟต์แวร์คอมพิวเตอร์ระยะสั้น ซึ่งมีความยืดหยุ่นต่อราคา (Price Elasticity) สูง
 SS' = อุปทานของซอฟต์แวร์คอมพิวเตอร์ระยะสั้น เมื่อมีการเลื่อนขึ้น (Shift Up) ของอุปสงค์ของผู้ซื้อไปสู่อุปสงค์ของผู้ใช้
 SL = อุปทานของซอฟต์แวร์คอมพิวเตอร์ระยะยาว ที่มีลักษณะเช่นเดียวกับอุปทานของสินค้าทั่วไป
 SL' = อุปทานของซอฟต์แวร์คอมพิวเตอร์ระยะยาว เมื่อมีการเลื่อนขึ้น (Shift Up) ของอุปสงค์ของผู้ซื้อไปสู่อุปสงค์ของผู้ใช้

จากภาพจะพบว่า ในตลาดสินค้าประเภทซอฟต์แวร์คอมพิวเตอร์ โดยเฉพาะอย่างยิ่ง ซอฟต์แวร์ประเภทนอกจากลักษณะของอุปสงค์ที่มีต่อซอฟต์แวร์สามารถแบ่งแยกออกได้เป็น 2 ลักษณะแล้ว เนื่องจากลักษณะของการผลิตที่มีต้นทุนเพิ่ม (Marginal Cost) ต่ำมาก การเพิ่มกำลังการผลิตในซอฟต์แวร์ตัวหนึ่งสามารถทำได้ง่าย ในต้นทุนที่ต่ำ ทำให้อุปทานของการผลิต/พัฒนาในระยะสั้น (ซึ่งชนิดและจำนวนของซอฟต์แวร์หรือโครงการผลิต/พัฒนาซอฟต์แวร์คงที่ ขณะที่ปริมาณของซอฟต์แวร์เพิ่มขึ้น) มีความยืดหยุ่นต่อราคาสูง (Elasticity) และที่น่าพิจารณามากไปกว่านั้นก็คือ ลักษณะของความยืดหยุ่นดังกล่าวจะมีมากขึ้นเรื่อย ๆ จนกระทั่งเส้นอุปทานมีลักษณะทอดลงได้ (Downward) หรือความลาดชัน (Slope) ของเส้นอุปทานมีค่าเป็นลบได้ ทั้งนี้เนื่องจากลักษณะต้นทุนเพิ่มที่ลดลง การเพิ่มปริมาณการผลิตขึ้นเรื่อย ๆ ก็จะทำให้ต้นทุนเฉลี่ยของการผลิต (Average Production Cost) ลดต่ำลงเรื่อย ๆ การเสนอขายซอฟต์แวร์ในราคาต่ำลงมาก ๆ ขณะที่ราคาจำหน่ายดังกล่าวยังสูงกว่าต้นทุนเฉลี่ยของการผลิตก็เป็นเรื่องที่เป็นไปได้ การปรับตัวในระยะสั้นที่มีผลมาจากปริมาณสินค้าซอฟต์แวร์ที่จำหน่ายได้เพิ่มขึ้นนี้ ก็จะเป็นการหมุนเส้นอุปทานระยะสั้นตามเข็มนาฬิกา

ขณะที่อุปทานในระยะยาวของซอฟต์แวร์คอมพิวเตอร์ (ซึ่งชนิดและจำนวนของโครงการผลิต/พัฒนาซอฟต์แวร์เพิ่มขึ้น หรือการเพิ่มความหลากหลายในประเภท/ชนิดของซอฟต์แวร์ขึ้น) ก็จะเป็นเช่นเดียวกับอุปทานในสินค้าทั่วไป ทั้งนี้เนื่องจากการผลิต/พัฒนาซอฟต์แวร์จำเป็นต้องใช้เวลาในการคิดค้น ทดสอบ เป็นระยะเวลาหนึ่ง การเพิ่มขึ้นในปริมาณของประเภทซอฟต์แวร์จึงมีลักษณะทอดขึ้นจากซ้ายไปขวา (Upward Slope) ดังเช่นสินค้าทั่วไป อย่างไรก็ตามในระยะยาวแล้ว หากมีการเลื่อนขึ้นของอุปสงค์ต่อซอฟต์แวร์คอมพิวเตอร์แล้ว การยอมเสนอซอฟต์แวร์ในราคาที่ต่ำลงซึ่งเป็นลักษณะของการปรับตัวของผู้ผลิตในการนำเสนอซอฟต์แวร์ประเภทต่าง ๆ ย่อมที่จะเป็นไปได้ ทำให้เส้นอุปทานของซอฟต์แวร์ระยะยาวเลื่อนตัวลง (Shift Down) ได้ ดังแสดงในภาพ

แต่ทั้งนี้การปรับตัวของอุปทานผ่านราคา คงต้องพิจารณาถึงลักษณะความยืดหยุ่นต่อราคาของเส้นอุปสงค์ประกอบด้วย ซึ่งหากความยืดหยุ่นของเส้นอุปสงค์ต่อราคามีค่าน้อย การปรับตัวลงของราคาอาจเกิดขึ้นน้อยหรือไม่เกิดขึ้นเลยก็ได้

2.4 การพัฒนาซอฟต์แวร์คอมพิวเตอร์

ซอฟต์แวร์คอมพิวเตอร์ที่ใช้ร่วมกับคอมพิวเตอร์ส่วนบุคคลนั้นมีมากมายหลายประเภท การผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์ในแต่ละประเภทก็จะมีลักษณะพิเศษแตกต่างกันออกไป ตามแต่ลักษณะของซอฟต์แวร์นั้น ๆ เช่น การพัฒนาซอฟต์แวร์จัดการระบบ (Operating System) ก็จะมี ความแตกต่างไปจากซอฟต์แวร์ประเภทระบบจัดการฐานข้อมูล (Data-Base Management System - DBMS) แต่ลักษณะความแตกต่างกันดังกล่าวมักจะขึ้นอยู่กับเทคนิคในการออกแบบและเขียนโปรแกรม ความถนัดของผู้พัฒนา หรือโปรแกรมเมอร์ ตลอดจนรูปแบบของภาษาที่ใช้เขียนโปรแกรมดังกล่าว หรือแม้แต่ความต้องการของโปรแกรมเมอร์ที่ต้องการให้โปรแกรมดังกล่าวมีลักษณะเด่นประการใด เช่น ต้องการให้โปรแกรมมีการทำงานที่รวดเร็ว ต้องการให้การแสดงผลทางหน้าจอ แสดงภาพ (Monitor) สวยงาม ต้องการให้สามารถแสดงผลออกทางเครื่องพิมพ์ในลักษณะต่าง ๆ ได้ หรือต้องการให้โปรแกรมสามารถใช้งานพร้อมกันได้ทีละหลาย ๆ คน (MultiUser) ซึ่งความแตกต่างเหล่านี้มักเป็นข้อแตกต่างที่เกิดขึ้นจากกระบวนการเชิงเทคนิค (Technical Procedure) แทบทั้งสิ้น

อย่างไรก็ตาม การจำแนกขั้นตอนในการผลิตและพัฒนาโปรแกรมคอมพิวเตอร์ ไม่มีมาตรฐานการกำหนดที่แน่นอน ขั้นตอนการผลิตและพัฒนายังจะแตกต่างกันไปตามแต่ผู้ศึกษาและวิจัยแต่ละคน เช่น

ตามการจำแนกของ Wolveston⁵

1. ขั้นตอนนำเสนอโครงการ (Proposal)
2. ขั้นตอนศึกษาความต้องการ (Requirement Study)
3. ขั้นตอนกำหนดรูปแบบของผลิตภัณฑ์ (Product Specification)
4. ขั้นตอนการออกแบบรายละเอียด (Detailed Design)

⁵R. W. Wolveston, "The Cost of Developing Large-Scale Software," IEEE-TC C-23 (June 1974): p. 115-136.

5. ขั้นตอนการเข้ารหัส (Coding)
6. ขั้นตอนการทดสอบโมดูล (Module Test)
7. ขั้นตอนการทดสอบระบบ (System Test)
8. ขั้นตอนการติดตั้ง (Install)
9. ขั้นตอนการยอมรับ (Acception)

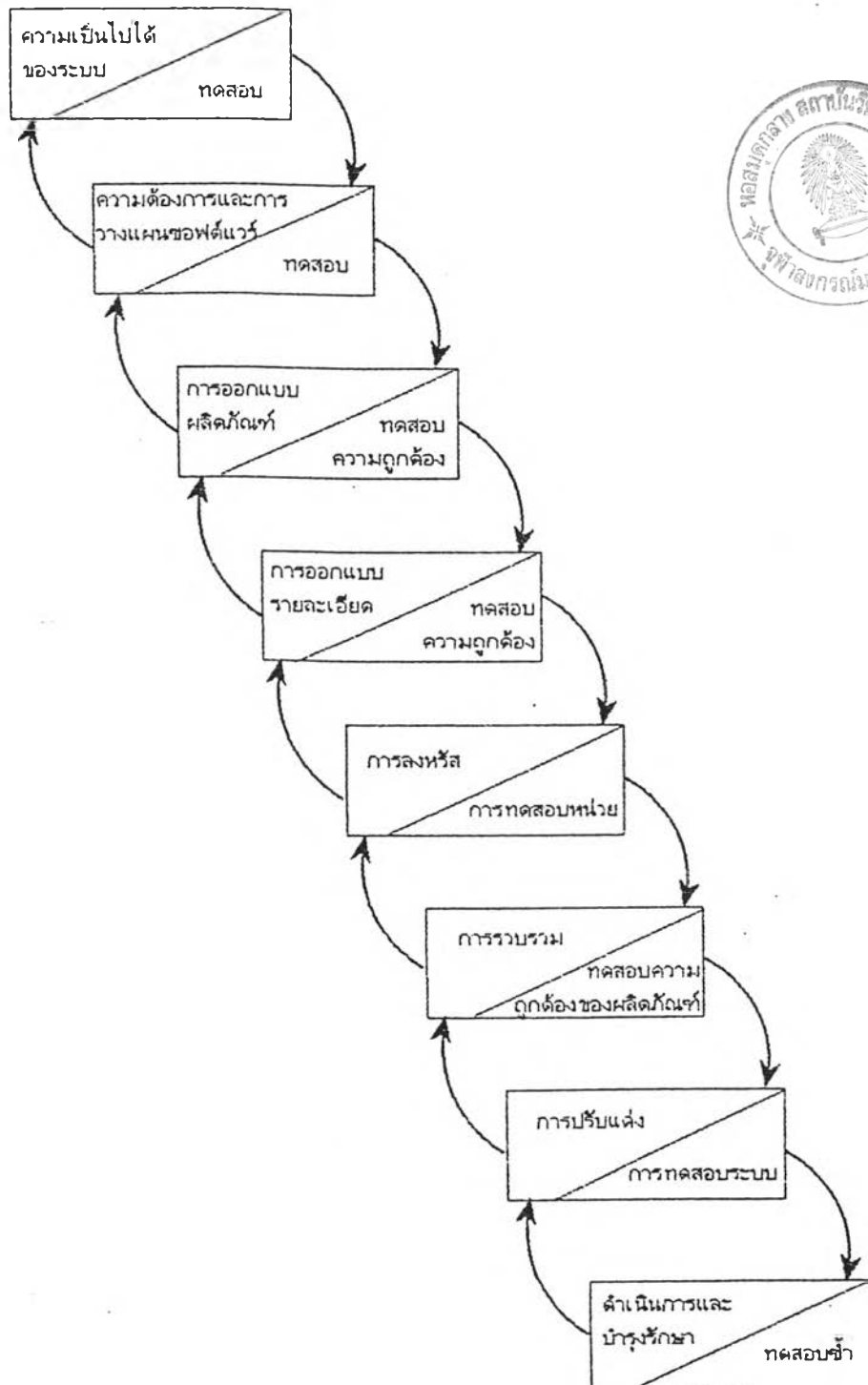
ตามการจำแนกของ Boehm⁶

1. ขั้นตอนการศึกษาความเป็นไปได้ของระบบ (Software Feasibility)
2. ขั้นตอนการวางแผนและศึกษาความต้องการของซอฟต์แวร์ (Software Plans and Requirement)
3. ขั้นตอนการออกแบบผลิตภัณฑ์ (Product Design)
4. ขั้นตอนการออกแบบรายละเอียด (Detail Design)
5. ขั้นตอนการเข้ารหัส (Code)
6. ขั้นตอนการรวม (Integration)
7. ขั้นตอนการเสริมแต่ง (Implementation)
8. ขั้นตอนการจัดการและบำรุงรักษา (Operation and Maintenance)

โดยทั้ง 8 ขั้นตอน มีความสัมพันธ์กันในลักษณะของแบบจำลองน้ำตก (Waterfall Model) ดังแสดง

⁶B. Boehm, Software Engineering Economics (Englewood Cliffs, New Jersey: Prentice-Hall, 1981)

ภาพที่ 2.2 แบบจำลองน้ำตก (Waterfall Model) ของการพัฒนาซอฟต์แวร์



แต่ทั้งนี้การผลิตและพัฒนาโปรแกรมคอมพิวเตอร์ไม่จำเป็นต้องดำเนินตามลำดับขั้นตอนต่าง ๆ ที่ระบุมาแล้วเสมอไป โดยเฉพาะการคาบเกี่ยวของเวลาในแต่ละขั้นตอน อาทิเช่น ขณะที่มีการนำเสนอโครงการ ก็อาจจะมีการศึกษาความต้องการและการกำหนดรูปแบบของผลิตภัณฑ์ไปในเวลาเดียวกัน พร้อมกันนั้นขอบเขตของงานในแต่ละขั้นตอนก็ยังไม่สามารถระบุได้อย่างแน่นอนและเป็นมาตรฐาน เช่น การกำหนดขั้นตอนวิธี (Algorithm) อาจอยู่ได้ทั้งขั้นตอนที่ 1. หรือขั้นตอนที่ 3. ก็ได้

จากขั้นตอนการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์ข้างต้น ผลผลิต (Output) รับในเบื้องต้นของการผลิต ก็ได้แก่ โปรแกรมรหัสต้นฉบับ (Source Code Program) ซึ่งมักจะเขียนขึ้นจากภาษาระดับสูง (High Level Languages) หรือภาษา Assembly ที่มีความใกล้เคียงกับภาษาเครื่อง (Machine Language) มากที่สุด จากนั้นจึงนำมาแปลด้วยตัวแปลภาษาให้เป็นโปรแกรมรหัสจุดหมาย (Object Code Program) หรือโปรแกรมรหัสเครื่อง (Machine Code Program) หรือโปรแกรมรหัสฐานสอง (Binary Code Program) ซึ่งเป็นชุดคำสั่งที่คอมพิวเตอร์สามารถเข้าใจได้ จากนั้นจึงนำไปทำการ Link เพื่อให้ได้โปรแกรมกระทำการ (Executive Program) ซึ่งเป็นโปรแกรมที่สามารถเรียกใช้โดยตรงได้ต่อไป

โปรแกรมคอมพิวเตอร์โปรแกรมหนึ่ง ๆ อาจไม่จำเป็นต้องเขียนขึ้นจากภาษาคอมพิวเตอร์ (Computer Language) เพียงภาษาเดียว เนื่องจากภาษาคอมพิวเตอร์แต่ละภาษาต่างก็มีจุดเด่นที่แตกต่างกัน ขึ้นอยู่กับวัตถุประสงค์ของการสร้างและนำภาษานั้น ๆ ไปใช้ ภาษาคอมพิวเตอร์อาจแบ่งได้ตามประเภทของตัวแปลภาษาได้ดังนี้

1. ตัวแปลภาษาระดับสูง (Compiler) ซึ่งเป็นตัวแปลภาษาระดับสูง (High Level Languages) ให้เป็นภาษาเครื่อง (Machine Language) โดยจะทำการแปลภาษาระดับสูงโดยจะทำการแปลทั้งโปรแกรม แล้วจึงทำตามชุดคำสั่งที่แปลภาษาดังกล่าวได้แก่ภาษา COBOL (Common Business Oriented Language), ภาษา Pascal, ภาษา C/C+/C++, ภาษา APL (A Programming Language) เป็นต้น

2. ตัวแปลภาษาระดับสูง (Interpreter) ซึ่งเป็นตัวแปลภาษาระดับสูง (High Level Languages) ให้เป็นภาษาเครื่อง (Machine Language) ข้อแตกต่างระหว่าง Compiler และ Interpreter ก็คือ Interpreter จะทำการแปลคำสั่ง

ภาษาระดับสูงที่ละบรรทัด แล้วทำตามคำสั่งนั้น จึงกลับไปแปลคำสั่งในบรรทัดต่อไป แล้วทำตามคำสั่งนั้นอีก ทำซ้ำอยู่เช่นนี้จนหมด ภาษาดังกล่าวได้แก่ ภาษา BASIC (Beginner's All-purpose Symbolic Instruction Code)

3. ตัวแปลภาษา Assembly (Assembler) ซึ่งเป็นตัวแปลภาษา Assembly ที่มีลักษณะอยู่ระหว่างภาษาระดับสูงและภาษาเครื่อง จุดเด่นของภาษานี้คือทำงานได้อย่างรวดเร็ว

จากลักษณะพิเศษที่สามารถจัดทำสำเนาได้อย่างง่ายดาย การจัดสรรทรัพยากรในการพัฒนาซอฟต์แวร์คอมพิวเตอร์สำหรับซอฟต์แวร์ประเภทที่ไม่ต้องใช้งานร่วมกับอุปกรณ์ประกอบ (Hardware-independent Software) แทบทั้งหมดจึงถูกทุ่มเทไปกับการพัฒนาซอฟต์แวร์ต้นฉบับ แต่หากซอฟต์แวร์ดังกล่าวเป็นซอฟต์แวร์ที่ต้องใช้ร่วมกับอุปกรณ์ประกอบ (Hardware-dependent Software) การผลิตซอฟต์แวร์ประเภทนี้ก็จะมีลักษณะเช่นเดียวกับการผลิตสินค้า 2 ชนิด คือ การผลิตตัวซอฟต์แวร์ และการผลิตอุปกรณ์ประกอบหรือฮาร์ดแวร์

การพัฒนาตัวซอฟต์แวร์คอมพิวเตอร์ก็เช่นเดียวกับการผลิตสินค้า โดยทั่วไปที่ปัจจัยการผลิตจะประกอบไปด้วย ปัจจัยทุนและปัจจัยแรงงาน ซึ่งปัจจัยทุนก็ได้แก่อุปกรณ์เครื่องคอมพิวเตอร์และส่วนประกอบต่าง ๆ รวมถึงอุปกรณ์เครื่องมือที่ช่วยในการพัฒนาโปรแกรม อาทิเช่น ซอฟต์แวร์ที่ใช้ในการพัฒนาซอฟต์แวร์เอง (Computer-aided Software Engineering, Computer-aided System Engineering) ซึ่งโดยส่วนใหญ่แล้วจะเป็นค่าใช้จ่ายคงที่ (Fixed Cost) ของผู้ผลิตซอฟต์แวร์ ขณะที่แรงงานที่ใช้ในการผลิตซอฟต์แวร์คอมพิวเตอร์แทบทั้งหมดเป็นแรงงานมีฝีมือ (Skill Labor) ทั้งนี้เนื่องจากการพัฒนาซอฟต์แวร์คอมพิวเตอร์มีความจำเป็นที่ต้องพึ่งพาความสามารถของบุคลากรเป็นอย่างมาก เช่น นักวิเคราะห์ระบบ (System Analyst), นักวิเคราะห์/นักเขียนโปรแกรม (Analyst/Programmer) และนักเขียนโปรแกรม (Programmer) เป็นผู้ถ่ายทอดเทคโนโลยีผ่านไปยังตัวโปรแกรม ซึ่งแตกต่างไปจากการผลิตสินค้าอุตสาหกรรมอีกหลาย ๆ ประเภท ที่เทคโนโลยีเป็นสิ่งที่ติดมากับสินค้าทุน การพัฒนาซอฟต์แวร์คอมพิวเตอร์ก็เป็นเช่นเดียวกับการผลิตระบบฐานความรู้โดยทั่วไป ที่ผู้ถ่ายทอดมีความสำคัญเป็นอย่างยิ่ง การวางแผนพัฒนาให้สอดคล้องกับความต้องการของผู้ใช้ การเขียนรหัสต้นฉบับ (Source Code) อย่างถูกต้องและใช้เทคโนโลยีที่ดีและเหมาะสม จะทำ

ให้ซอฟต์แวร์คอมพิวเตอร์ดังกล่าวสามารถตอบสนองความต้องการของผู้ใช้ ส่งผลต่อความสำเร็จในเชิงธุรกิจของผู้ผลิตและพัฒนา อันจะเป็นแรงจูงใจให้เกิดการพัฒนาอย่างต่อเนื่องต่อไป

ดังที่ได้กล่าวแล้วว่า ซอฟต์แวร์คอมพิวเตอร์เป็นสินค้าอุตสาหกรรมที่มีต้นทุนในการผลิตแตกต่างไปจากสินค้าประเภทอื่น ๆ อันเนื่องมาจากความสามารถในการลอกเลียนหรือทำสำเนา (Copy) ได้อย่างง่ายดาย ต้นทุนในด้านการวิจัยและพัฒนาซอฟต์แวร์ (Research and Development - R & D) สำหรับงานการผลิตซอฟต์แวร์ต้นฉบับจึงเป็นต้นทุนแทบทั้งหมดของงานด้านนี้ แต่ทั้งนี้โดยปกติแล้ว ผู้ผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์นั้น ไม่ได้มีเพียงต้นทุนในการผลิตเพียงอย่างเดียว ต้นทุนในการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์อาจแบ่งได้เป็นหัวข้อหลัก ๆ ดังนี้

1. ต้นทุนในด้านการวิจัยและพัฒนา (Research and Development Cost - R & D Cost) ซึ่งต้นทุนในประเภทนี้อาจแยกได้เป็นต้นทุนในขั้นตอนต่าง ๆ ในการผลิต (Stage of Production)
2. ต้นทุนในด้านการดำเนินการของบริษัท (Operation Cost)
3. ต้นทุนในการบำรุงรักษาและสนับสนุนหลังการขาย (Maintenance and Sale Support Cost)
4. ต้นทุนทางด้านการโฆษณา (Advertising Cost) ซึ่งเป็นสิ่งที่จำเป็นสำหรับธุรกิจหรือหน่วยผลิตที่อยู่ในตลาด Monopolistic Competition

นอกจากต้นทุนประเภทต่าง ๆ ข้างต้น ซอฟต์แวร์คอมพิวเตอร์หลายชนิดก็จำเป็นต้องใช้ร่วมกับอุปกรณ์ประกอบบางอย่าง ทั้งนี้ก็เพื่อเป็นการป้องกันการละเมิดสิทธิ์ของผู้ผลิตซอฟต์แวร์ การป้องกันการทำสำเนาดังกล่าวก็เท่ากับเป็นการเพิ่มต้นทุนในการผลิตให้กับผู้ผลิตซอฟต์แวร์นั้น ๆ ด้วย โดยเฉพาะอย่างยิ่งในประเทศไทย ต้นทุนในการป้องกันการทำสำเนา (Copy Protection Cost) จึงเป็นต้นทุนของกิจการซอฟต์แวร์คอมพิวเตอร์ที่สำคัญอีกประการหนึ่ง

ขณะที่ต้นทุนในส่วนของการพัฒนาและวิจัยเพื่อทำการผลิตซอฟต์แวร์ต้นฉบับเป็นต้นทุนหลักของการผลิต ต้นทุนในการผลิตซอฟต์แวร์หน่วยต่อ ๆ ไปจึงเป็นต้นทุนที่ต่ำมาก

เมื่อเปรียบเทียบกับต้นทุนหน่วยแรก (ซอฟต์แวร์ต้นฉบับ) ซึ่งเป็นลักษณะที่แตกต่างไปจากสินค้าอุตสาหกรรมประเภทอื่น ซึ่งเมื่อมีการพัฒนาผลิตภัณฑ์ต้นฉบับแล้ว ยังต้องมีการสร้างกระบวนการผลิตขึ้นเพื่อทำการผลิตในลักษณะการผลิตจำนวนมาก (Mass Product) ด้วยขณะที่กระบวนการผลิตในลักษณะ Mass Product ของซอฟต์แวร์คอมพิวเตอร์เป็นเพียงการจัดทำสำเนาซอฟต์แวร์และการจัดทำเอกสารประกอบการใช้งาน (User Manual) ขณะที่ต้นทุนในด้านการประกอบการของหน่วยธุรกิจ มีลักษณะค่อนข้างที่จะเป็นต้นทุนคงที่ (Fixed Cost) ทำให้เมื่อทำการเพิ่มการผลิตซอฟต์แวร์คอมพิวเตอร์มากขึ้น ก็จะเป็นการทำให้ต้นทุนเฉลี่ยของการผลิตยิ่งลดต่ำลง เป็นไปตามลักษณะของผลตอบแทนต่อขนาดเพิ่มขึ้น (Increasing Return to Scale) ขณะที่เมื่อมีการขยายการผลิตมากขึ้น ก็ทำให้เกิดต้นทุนในส่วนของการบำรุงรักษาและบริการหลังการขายเกิดขึ้น รวมถึงต้นทุนในด้านการโฆษณา จึงเป็นเรื่องที่ไม่อาจจะประกันได้ว่าเกิดการประหยัดอันเนื่องมาจากขนาดขึ้น (Economies of Scale) ในการผลิตซอฟต์แวร์คอมพิวเตอร์ แต่หากพิจารณาเฉพาะต้นทุนในการวิจัยและพัฒนาและการผลิตเพียงประการเดียวแล้ว (R & D and Production Cost) แล้ว การเพิ่มการผลิตซอฟต์แวร์คอมพิวเตอร์ขึ้น ย่อมทำให้เกิดการประหยัดเนื่องจากขนาดอย่างแน่นอน

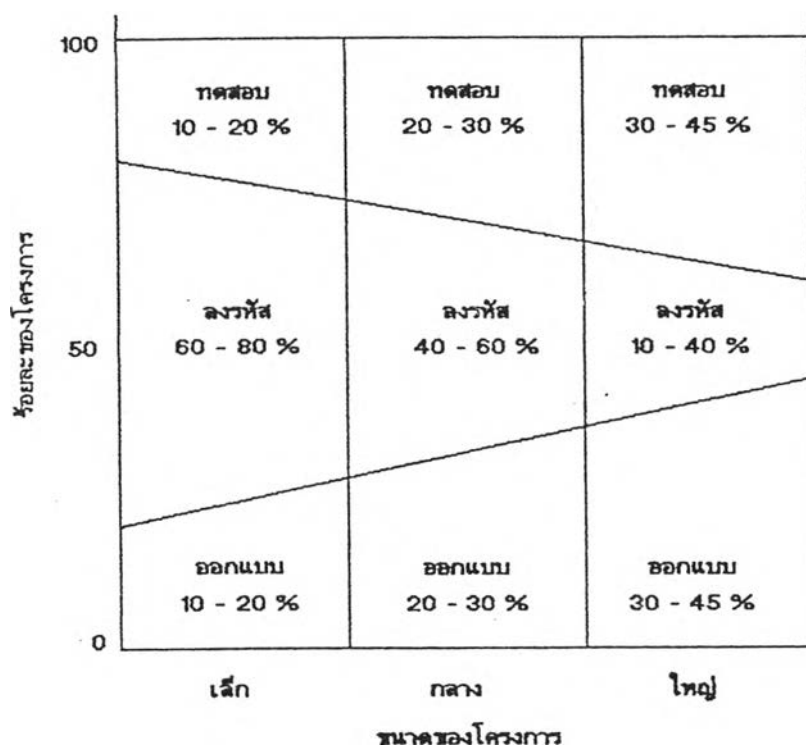
ซึ่งหากพิจารณาในด้านกระบวนการผลิตแล้ว พบว่าการผลิตและพัฒนาซอฟต์แวร์คอมพิวเตอร์เป็นอุตสาหกรรมที่มีลักษณะต้นทุนลดลง (Decreasing Cost Industry) ซึ่งเป็นลักษณะของอุตสาหกรรมที่ทำการผลิตจำนวนมาก (Mass Product) และเมื่อมีการผลิตจำนวนมากแล้ว ทำให้ต้นทุนในการผลิตลดต่ำลง ลักษณะดังกล่าวนี้ยังมีปรากฏอยู่ในอุตสาหกรรมการผลิตอุปกรณ์และเครื่องมือคอมพิวเตอร์⁷ ด้วย

⁷William A. McEachern, Microeconomics : A Contemporary Introduction (Cincinnati, Ohio : South-Western Publishing Co., 1988) p. 208-209.

2.5 ผลงานวิจัยที่เกี่ยวข้อง

Kustanowitz^๕ ทำการศึกษาความสัมพันธ์ระหว่างวัฏจักรชีวิตของระบบการพัฒนาระบบคอมพิวเตอร์ (System Life Cycle of Software Project) ซึ่งจะแปรผันไปตามขนาดของโครงการพัฒนา (Project Size) โดยที่โครงการขนาดเล็ก ขั้นตอนของการลงรหัสจะมีสัดส่วนที่มากกว่าขั้นตอนการออกแบบและการทดสอบ ขณะที่เมื่อขนาดของโครงการใหญ่มากขึ้น ขั้นตอนในการออกแบบและลงรหัสจะมีสัดส่วนที่มากขึ้นเป็นลำดับ

ภาพที่ 2.3 ความสัมพันธ์ระหว่างวัฏจักรชีวิตของระบบการพัฒนาระบบคอมพิวเตอร์กับขนาดของโครงการ



^๕A. L. Kustanowitz., "System Life Cycle Estimation : A New Approach to Estimating Resources for Application Program Development." COMPSAC, (1977): 165.

³Lientz และ Swanson ทำการศึกษาถึงสาเหตุในการพัฒนาเปลี่ยนแปลงรุ่น (Release) ของซอฟต์แวร์ ตลอดจนเปรียบเทียบความพยายามที่ใช้ในการบำรุงรักษาซอฟต์แวร์คอมพิวเตอร์ในเชิงร้อยละ (Percent of Software Maintenance Effort)

ตารางที่ 2.2 เหตุผลในการเปลี่ยนแปลงรุ่นของซอฟต์แวร์

สาเหตุของการเปลี่ยนแปลง (Reason for Changes)	ความพยายามในการบำรุงรักษา (%) (Software Maintenance Effort)
ความต้องการของผู้ใช้ (Enhancements for users)	41.8
การเปลี่ยนแปลงอันเนื่องมาจากการเปลี่ยนแปลงในข้อมูล (Changes arising from changes in input data)	17.4
การแก้ไขโปรแกรมโดยเร่งด่วน (Emergency program correction)	12.4
การแก้ไขโปรแกรมคำสั่งประจำ (Routine program debugging)	9.3
การเปลี่ยนแปลงอันเนื่องมาจากการเปลี่ยนแปลงอุปกรณ์ (Change arising from changes in hardware)	6.2
การปรับปรุงเพื่อเปลี่ยนแปลงเอกสารคู่มือ (Change to improve documentation)	5.5
ปรับปรุงรูปแบบการทำงาน (Improve performance)	4.0
อื่น ๆ (Miscellaneous)	3.4

³B. P. Lientz and E. B. Swanson, Software Maintenance Management : A Study of the Maintenance of Computer Application Software in 487 Data Processing Organizations, (Reading, Mass :Addison-Wesley, 1980), p. 185.

Mckeen¹⁰ ทำการศึกษาเปรียบเทียบร้อยละของการจัดสรรทรัพยากรในแต่ละกิจกรรมของการพัฒนาซอฟต์แวร์คอมพิวเตอร์ (Percentage Resource Allocation) โดยทำการรวบรวมผลงานของ Davis, Zelkowitz และ Shaw

ตารางที่ 2.3 การจัดสรรทรัพยากรในแต่ละกิจกรรมของการพัฒนาซอฟต์แวร์

วัฏจักรชีวิต (Life Cycle)	การจัดสรรทรัพยากร (Percentage Resource Allocation)		
	Davis	Zelkowitz	Shaw
ช่วง/กิจกรรม (Phase/Activity)			
การวิเคราะห์ ¹ (Analysis)	25	20 ²	25
การออกแบบ (Design)	20	15	10 ³
การลงรหัส (Coding)	25	45 ⁴	30
ทดสอบระบบ (System Test)	n/a ⁵	20	5
การปรับแต่ง (Implementation)	15	n/a ⁶	19

¹ การวิเคราะห์ครอบคลุมกิจกรรมในการพัฒนาทุกประเภทก่อนการออกแบบรายละเอียด

² ความพยายามที่ใช้ในการวิเคราะห์อาจต่ำเกินไป เนื่องจากข้อมูลดังกล่าวได้รับมาจากการพัฒนาระบบในสิ่งแวดล้อมทางการทหาร ดังนั้นกิจกรรมเมื่อเริ่มแรกซึ่งประกอบไปด้วยการวิเคราะห์ความเป็นไปได้และการศึกษาระบบเบื้องต้นไม่น่าจะนำมาคำนวณไว้ด้วย

³ กิจกรรมการกำหนดรายละเอียด (Detailed Design) ดังกล่าวประกอบไปด้วยการกำหนดลักษณะของระบบ (System Specifications) และความต้องการเชิงเทคนิค (Technical Requirements)

⁴ การลงรหัสและการทดสอบ Module ถูกรวมเข้าเป็นกิจกรรมเดียวกัน โดยปกติโปรแกรมเมอร์จะทำการทดสอบจุดหรือ Module ต่าง ๆ ของระบบเมื่อได้ลงรหัสเสร็จสิ้นแล้ว

⁵ กิจกรรมดังกล่าวถูกรวมอยู่ในขั้นตอนการเปลี่ยนแปลง (Conversion) โดย Davis

⁶ กิจกรรมดังกล่าวไม่มีรายงานไว้

¹⁰J. D. Mckeen, "Successful Development Strategies for Business Application System," MIS Quarterly Vol.7, No.3, (September 1983): p. 62.

Gotlieb¹¹ ทำการศึกษาค่าใช้จ่ายเปรียบเทียบในแต่ละขั้นตอนของโครงการผลิตซอฟต์แวร์ (Relative Expenditure of Effort in Stages of a Software Project) โดยทำการเปรียบเทียบร้อยละของความพยายาม (Percent of Expenditure of Effort) ที่ใช้ในแต่ละขั้นตอนของการผลิต ทั้งนี้โดยได้รวบรวมเปรียบเทียบกับงานศึกษาค้นคว้าของ Wolverton, Zelkowitz และ Boehm

ตารางที่ 2.4 ค่าใช้จ่ายเปรียบเทียบในแต่ละขั้นตอนการผลิตซอฟต์แวร์

	ค่าใช้จ่าย (%)									
	Wolverton			Boehm						
	1	2	3	4	5	6				
			Zelkowitz	Gottlieb	Small	Intermediate	Medium	Large	Gottlieb	
นำเสนอโครงการ	12		10	10	12	5	5	5	5	5
ศึกษาสิ่งที่ต้องการ		42		6						5
กำหนดลักษณะของผลิตภัณฑ์	18		10	10	13	15	15	15	15	10
การออกแบบรายละเอียด	16	33	15	15		25	24	23	22	15
การลงรหัส	20		20	20	19	40	38	36	34	25
การทดสอบ Module	20	13	25	25	19	15	18	21	24	20
การทดสอบระบบ					19					10
การติดตั้ง		14	12	20	20	12				5
การตรวจรับ										5

¹ Wolverton - 1974

² จากโครงการ Sage - 1974

³ จากโครงการ Gemini II - 1974

⁴ 1979

⁵ จากการศึกษาโครงการพัฒนาระบบฐานข้อมูลขนาดใหญ่ในอุตสาหกรรมโทรศัพท์ - 1979

⁶ ขนาดของข้อมูลต้นฉบับจาก Small, Intermediate, Medium และ Large อยู่ระหว่างประมาณ 2k, 8k, 32k และ 128k ตามลำดับ - 1981

¹¹C. C. Gotlieb, The Economics of Computers: Costs, Benefits, Policies, and Strategies (New Jersey: Prentice-Hall, Inc., 1985): 170.

Robert Schware¹² ทำการศึกษาอุตสาหกรรมการผลิตซอฟต์แวร์ของโลก และวิศวกรรมด้านซอฟต์แวร์ (The World Software Industry and Software Engineering) โดยเน้นหนักในแนวทางของโอกาสและเงื่อนไขสำหรับระบบเศรษฐกิจอุตสาหกรรมใหม่ (Opportunities and Constraints for Newly Industrialized Economies) แต่ทั้งนี้ได้กล่าวถึงลักษณะข้อมูลในอุตสาหกรรมนี้ที่มีลักษณะที่ไม่สามารถยึดหรือเชื่อถือได้มากนัก (Unreliable Data) การศึกษาแบ่งออกเป็นประเด็นต่าง ๆ ดังนี้

1. แนวโน้มหลัก (Major Trends) โดยแบ่งเป็นประเด็นได้ดังนี้
 - 1.1 การลดขนาดลง (Downsizing)
 - 1.2 การรวมกลุ่มกันของระบบซอฟต์แวร์ (System Software Integration)
 - 1.3 การรวมกลุ่มกันของบริษัท (Teaming by Companies)
 - 1.4 การต่อสู้กันระหว่างมาตรฐานของระบบปฏิบัติการ (The Battle for Operating System Standards)
 - 1.5 รากฐานของการพัฒนาแอปพลิเคชัน (Application Development Backlogs)
 - 1.6 ซอฟต์แวร์ชุดและผลิตภัณฑ์ต่างความต้องการของลูกค้า ใน ราคาของซอฟต์แวร์ชุด (Packaged Software and Customized Products at Package Prices)
 - 1.7 การรวมกันของระบบ (Systems integration)

2. การเปลี่ยนแปลงในธรรมชาติของการแข่งขันโดยรวม (The Changing Nature of Global Competition)

¹²Robert Schware, The World Software Industry and Software Engineering : Opportunities and Constraints for Newly Industrialized Economies, (Washington, D.C.: World Bank Technical Paper, 1989).

3. ขั้นตอนของการพัฒนาซอฟต์แวร์คอมพิวเตอร์และปัจจัยที่ต้องการ
(Stages of Software Development and Factor Requirements) โดยแบ่ง
ขั้นตอนในการพัฒนาซอฟต์แวร์ออกเป็น

- 3.1 การกำหนดความต้องการ (Requirement Specification)
- 3.2 การออกแบบ (Design)
- 3.3 การลงรหัสและการโปรแกรม (Coding/Programming)
- 3.4 การทดสอบ (Testing)
- 3.5 การบำรุงรักษา (Maintenance)



4. การจัดการการพัฒนาซอฟต์แวร์ (Software Development
Management) โดยแบ่งออกเป็น

- 4.1 มาตรฐานการประเมินซอฟต์แวร์ (Software Assessment
Management)
- 4.2 เมตริกซอฟต์แวร์ (Software Metrics)
- 4.3 การจัดการความเสี่ยงของซอฟต์แวร์ (Software Risk
Management)

5. ผลกระทบต่อเทคโนโลยีใหม่ (The Impact of New Technologies)
แบ่งออกเป็น

- 5.1 การช่วยเหลือในการพัฒนาซอฟต์แวร์แบบอัตโนมัติ
(Automated Software Development Aids)
- 5.2 ปัญญาประดิษฐ์และการพัฒนาซอฟต์แวร์ (Artificial
Intelligence and Software Development)
- 5.3 การศึกษาข้อมูลต้นทุนต่ำ (Low-cost Data
Communications)

6. การพัฒนาทรัพยากรมนุษย์ (Human Resource Development)

- 6.1 ความต้องการความชำนาญ (Skill Requirement)
- 6.2 การเลื่อนของการได้เปรียบโดยเปรียบเทียบในความต้องการแรงงาน (Shifting Comparative Advantage in Labor Requirements)
- 6.3 โครงการทางการศึกษาและฝึกอบรม (Education and Training Programs)

7. ทางเลือกสำหรับเศรษฐกิจอุตสาหกรรมใหม่ (Options for Newly Industrialized Economies - NIEs) โดยแบ่งออกเป็น

- 7.1 ซอฟต์แวร์สำหรับการส่งออก (Software for Export)
- 7.2 บริเวณการเติบโตของซอฟต์แวร์ (Software Growth Areas)
- 7.3 การดัดแปลงซอฟต์แวร์สำเร็จรูปที่มีอยู่แล้ว (Modify Existing Application Software)
- 7.4 การจัดหาและการปรับใช้ซอฟต์แวร์สาธารณะ (Acquire and Adapt Public Domain Software)
- 7.5 ใช้ระเบียบวิธีการพัฒนาระบบซอฟต์แวร์ (Use Software System Development Methodologies)
- 7.6 สร้างแนวร่วม (Build Alliances)
- 7.7 เพิ่มโครงการพัฒนากำลังคน (Expand Manpower Development Program)

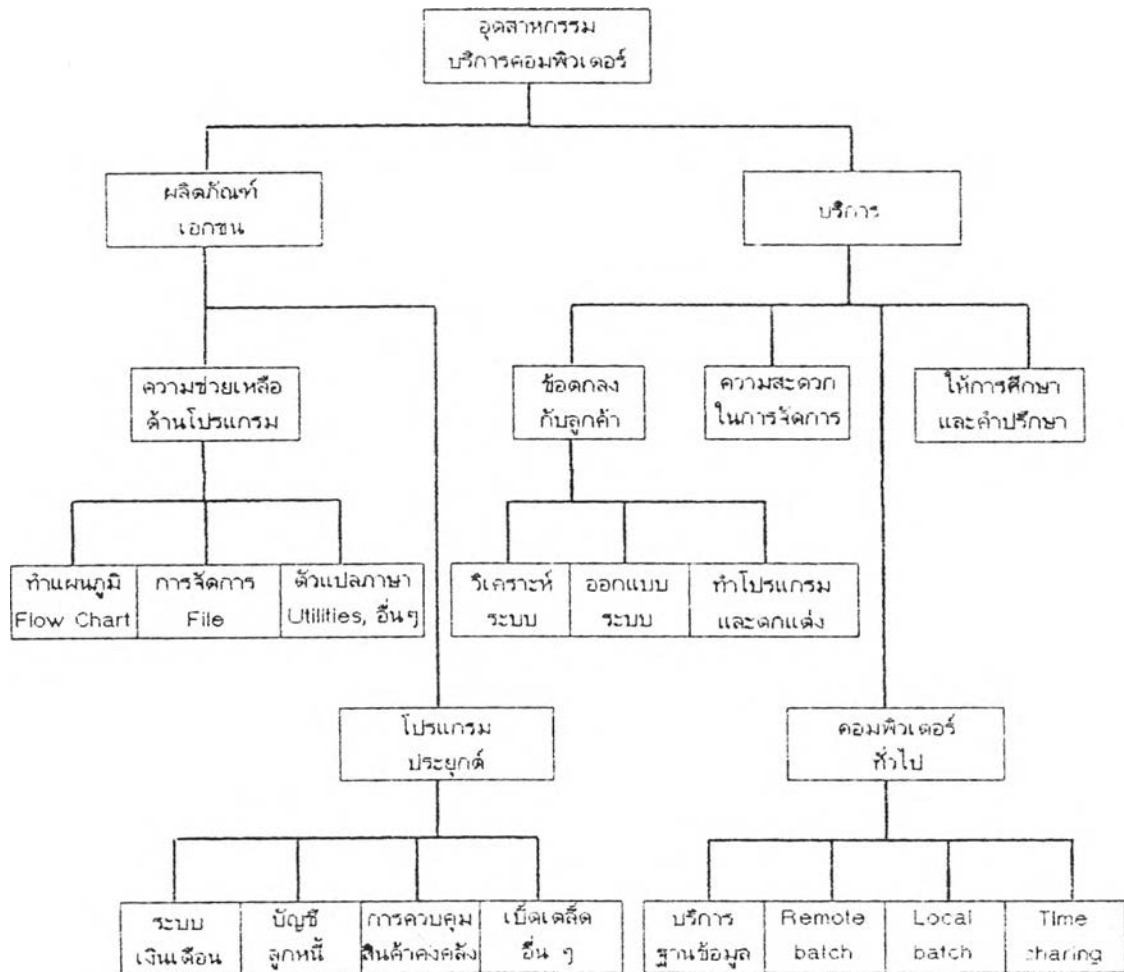
8. เศรษฐกิจอุตสาหกรรมใหม่กับอนาคตของวิศวกรรมซอฟต์แวร์ (NIEs and the Future of Software Engineering)

องค์การเพื่อการพัฒนาและร่วมมือทางเศรษฐกิจ (Organisation for Economic Co-operation and Development - OECD)¹³ ทำการศึกษาความเป็นสากลของซอฟต์แวร์และบริการคอมพิวเตอร์ (The Nationalisation of Software and Computer Services) โดยทำการศึกษาและประมาณลักษณะความเป็นสากล (International) ของอุตสาหกรรม และพยายามที่จะหาสาเหตุของปัญหาและอุปสรรคที่เกิดขึ้นหรืออาจเกิดขึ้นได้ (Actual or Potential Problems and Barriers) ในการค้าระหว่างประเทศ (รวมถึงการส่งออกโดยตรงและผลิตภัณฑ์ภายในประเทศที่ได้รับการสนับสนุนจากต่างประเทศ) ซึ่งการศึกษาดังกล่าวนี้เป็นการศึกษาทางด้านอุปทานของอุตสาหกรรมซอฟต์แวร์และการให้บริการทางด้านคอมพิวเตอร์ (Supply Side of the Software and Computer Services Industry)

การศึกษาดังกล่าวทำการสำรวจโดยใช้แบบสอบถาม (Questionair) โดยสอบถามไปยังบริษัทซึ่งอยู่ในสมาคมการค้าของประเทศต่าง ๆ เช่น สหรัฐอเมริกา อิตาลี โปรตุเกส คานาดา สหราชอาณาจักร เดนมาร์ค เบลเยียม ญี่ปุ่น นิวซีแลนด์ สเปน ฝรั่งเศส และเยอรมัน ซึ่งได้ผลสรุปต่าง ๆ ดังแสดงในตาราง

¹³Organization for Economic Co-operation and Development (OECD), The Internationalisation of Software and Computer Services, (France : Head of Publication Service, OECD, 1989).

ภาพที่ 2.4 อุตสาหกรรมบริการคอมพิวเตอร์



ตารางที่ 2.5 อุตสาหกรรมซอฟต์แวร์และบริการคอมพิวเตอร์ (รวมผู้ผลิตฮาร์ดแวร์)

รายได้มูลค่าเพิ่มทางการตลาดภายในประเทศ ปี 1985 หน่วย : ล้านดอลลาร์สหรัฐ

ประเทศ	Package Software	Custom Software	รวม ซอฟต์แวร์	บริการ ประมวลผล	รวม บริการ	รวม อุตสาหกรรม	สัดส่วน ซอฟต์แวร์ (%)
ฝรั่งเศส	859.90	1298.80	2158.70	1120.90	1239.90	3398.60	63.52
เยอรมัน	1157.00	707.00	1864.00	760.00	904.00	2768.00	67.34
อิตาลี	573.00	498.00	1071.00	460.10	521.10	1592.10	67.27
สหราชอาณาจักร	1093.70	737.20	1830.90	687.90	814.90	2645.80	69.20
ยุโรปตะวันตก (4 ประเทศ)	3683.60	3241.00	6924.60	3028.90	3479.90	10404.50	66.55
ออสเตรเลีย	113.00	75.00	188.00	89.90	104.90	292.90	64.19
เบลเยียม	173.20	127.10	300.30	214.20	236.20	536.50	55.97
เดนมาร์ก	115.00	94.00	209.00	267.00	285.00	494.00	42.31
ฟินแลนด์	87.00	99.00	186.00	173.90	190.90	376.90	49.35
เนเธอร์แลนด์	291.05	284.05	575.10	325.10	373.10	948.20	60.65
นอร์เวย์	102.00	86.00	188.00	233.80	249.80	437.80	42.94
สเปน	190.00	115.00	305.00	117.10	139.10	444.10	68.68
สวีเดน	195.00	149.00	344.00	301.90	329.90	673.90	51.05
สวิสเซอร์แลนด์	195.00	146.00	341.00	245.00	270.00	611.00	55.51
ยุโรปตะวันตก (13 ประเทศ)	5144.85	4416.15	9561.00	4996.80	5658.80	15219.80	55.51
ไอร์แลนด์	43.70	27.70	71.40	24.50	30.90	102.30	69.79
โปรตุเกส	15.10	9.90	25.00	13.20	17.20	42.20	59.24
ตุรกี	3.12	3.63	6.75	5.70	5.95	12.70	53.15
ยุโรปตะวันตก (16 ประเทศ)	5206.77	4457.38	9664.15	5040.20	5712.85	15377.00	62.85
กรีก	2.81	3.13	5.94	4.10	4.46	10.40	57.12
ยุโรปตะวันตก (17 ประเทศ)	5209.58	4460.51	9670.09	5044.30	5717.31	15387.40	62.84
ออสเตรเลีย	324.50	175.20	499.70	137.40	175.90	675.60	73.96
แคนาดา	388.00	208.70	596.70	454.10	572.70	1169.40	51.03
ญี่ปุ่น	270.10	2590.80	2860.90	3555.90	4206.10	7067.00	40.48
สหรัฐอเมริกา	12602.00	3944.00	16546.00	10479.00	14012.00	30558.00	54.15
สหรัฐอเมริกา	n.c.	n.c.	16265.00	13350.00	n.a.	n.a.	n.a.

ตารางที่ 2.6 อุตสาหกรรมซอฟต์แวร์ : สัดส่วนของผู้ผลิตอุปกรณ์ฮาร์ดแวร์

รายได้มูลค่าเพิ่มทางการตลาดภายในประเทศ ปี 1985 หน่วย ล้านดอลลาร์สหรัฐ

ประเทศ	Package Software		Custom Software/การเป็นที่ปรึกษา		รวมซอฟต์แวร์	
	ผู้ผลิตฮาร์ดแวร์	รวม	ผู้ผลิตฮาร์ดแวร์	รวม	ผู้ผลิตฮาร์ดแวร์	รวม
	%		%		%	
ฝรั่งเศส	51.6	859.9	7.5	1298.8	25.1	2158.7
เยอรมัน	54.0	1157.0	18.1	707.0	40.4	1864.0
อิตาลี	57.8	573.0	12.5	498.0	36.7	1071.0
สหราชอาณาจักร	55.5	1093.7	17.5	737.2	40.2	1830.9
ยุโรปตะวันตก						
(4 ประเทศ)	54.5	3683.6	12.9	3296.6	35.0	6924.6
ออสเตรเลีย	59.3	113.0	25.3	75.0	45.7	188.0
เบลเยียม	56.1	173.2	18.1	127.1	40.0	300.3
เดนมาร์ก	52.2	115.0	17.0	94.0	36.4	209.0
ฟินแลนด์	46.0	87.0	13.1	99.0	28.5	186.0
เนเธอร์แลนด์	45.4	291.1	12.0	284.1	28.9	575.1
นอร์เวย์	52.9	102.0	15.1	86.0	35.6	188.0
สเปน	59.5	190.0	26.1	115.0	46.9	305.0
สวีเดน	54.9	195.0	16.8	149.0	38.4	344.0
สวิสเซอร์แลนด์	52.3	185.0	17.1	146.0	37.2	341.0
ยุโรปตะวันตก						
(13 ประเทศ)	54.0	5144.9	13.9	4471.8	35.5	9561.0
ไอร์แลนด์	40.9	43.7	14.8	27.7	30.8	71.4
โปรตุเกส	60.0	15.1	20.0	9.9	44.2	25.0
ตุรกี	33.3	3.1	25.0	3.6	29.0	6.8
ยุโรปตะวันตก						
(16 ประเทศ)	53.9	5206.8	14.0	4513.0	35.5	9664.2
กรีซ	33.3	2.8		3.1	16.7	5.9
ยุโรปตะวันตก						
(17 ประเทศ)	53.9	5209.6	14.0	4516.1	35.5	9670.1

ตารางที่ 2.7 ความยุ่งยากที่อาจเกิดขึ้นได้ในกิจกรรมระหว่างประเทศ

(Possible Difficulties in International Activities)

หน่วย : เปอร์เซ็นต์ของการเห็นด้วย

ประเด็นทั่วไป	ประเทศกลุ่มสมาชิก OECD		ประเทศนอกกลุ่มสมาชิก	
	กิจการทั้งหมด	เฉพาะกิจการเพื่อส่งออก	กิจการทั้งหมด	เฉพาะกิจการเพื่อส่งออก
1. ข้อจำกัดในการเข้าถึงตลาดภายในประเทศ	6.67	5.88	30.36	36.84
2. ข้อจำกัดของสิทธิในการก่อตั้ง (การเปิดสำนักงานตัวแทน, สาขาและบริษัทสาขา)	16.44	16.33	34.62	38.24
3. ข้อจำกัดในการลงทุนในบริษัทท้องถิ่น	14.71	13.64	32.69	38.24
4. ข้อจำกัดในการถ่ายทอดเทคโนโลยี (โดยเฉพาะอย่างยิ่ง เกี่ยวกับข้อตกลงในการใช้สิทธิที่ถูกต้องตามกฎหมาย)	12.50	12.50	29.09	37.14
5. ข้อจำกัดของอัตราแลกเปลี่ยน (โดยเฉพาะอย่างยิ่ง การชำระค่าธรรมเนียมสิทธิ)	16.00	18.37	47.27	58.33
6. การขาดการคุ้มครองซอฟต์แวร์ตามกฎหมายของประเทศนั้น	26.47	27.70	37.25	32.26
7. ข้อขัดแย้งระหว่างประเทศที่เป็นเจ้าของซอฟต์แวร์และ ประเทศคู่ค้าเกี่ยวกับการคุ้มครองซอฟต์แวร์	19.12	18.18	30.61	30.00
8. ปัญหาในการให้บริการแก่ลูกค้าผู้ซื้อซอฟต์แวร์	27.78	31.91	25.00	24.24
9. ข้อจำกัดของการนำเข้าอุปกรณ์ฮาร์ดแวร์พร้อมกับ การจำหน่ายซอฟต์แวร์และ/หรือบริการภายในประเทศ	15.15	16.28	43.14	50.00
10. ข้อจำกัดในการเข้าถึงตลาดภาครัฐ	43.08	44.19	41.67	48.28
11. ความแตกต่างทางด้านภาษีและกิจการภายในประเทศ	20.90	20.93	29.17	33.33
12. ข้อจำกัดในการย้ายบุคคลากร เข้าไปภายในประเทศระยะสั้น	16.44	16.67	25.49	27.27
13. ข้อจำกัดในการย้ายบุคคลากร เข้าไปภายในประเทศระยะยาว	29.58	29.79	39.22	42.42
14. ข้อจำกัดอันเป็นผลมาจากกฎหมายคุ้มครองข้อมูล (Data Protection Laws)	18.18	17.78	15.22	12.90