

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 ภาษาจาวา

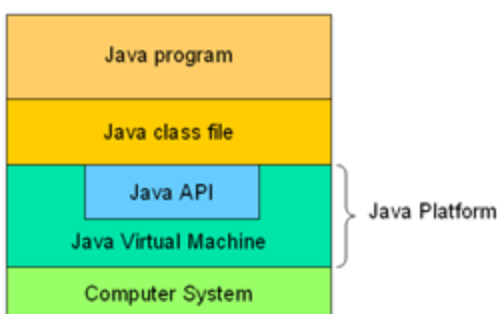
เป็นภาษาโปรแกรมเชิงวัตถุ (Object-Oriented) ที่ถูกพัฒนาขึ้นโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ จากซันไมโครซิสเต็มส์ ภาษาจาวานั้นได้ถูกพัฒนาขึ้นเพื่อจุดประสงค์ในการ ใช้แทนภาษาซีพลัสพลัส (C++) โดยมีรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ทีฟซี (Objective-C) แม้ว่าชื่อจาวาจะมีความคล้ายคลึงกับ จาวา สคริปต์ (Javascript) แต่ทั้งสองภาษานี้ไม่ได้มีความเกี่ยวข้องกันแต่อย่างใด ปัจจุบันมาตรฐานของภาษาจาวาดูแลโดย Java Community Process ซึ่งเป็นกระบวนการอย่างเป็นทางการ ที่อนุญาตให้ผู้ที่สนใจเข้าร่วมกำหนดความสามารถในจาวา แพลตฟอร์ม (Platform) ได้

2.1.1 ประวัติความเป็นมาของภาษาจาวา

ภาษาจาวาได้รับการพัฒนาขึ้นโดยที่ถูกพัฒนาขึ้นโดย ดอกเตอร์ เจมส์ กอสลิง และวิศวกรคนอื่นๆ จากบริษัท ซันไมโครซิสเต็มส์ ในปี พ.ศ. 2534 แต่เดิมภาษา Java มีชื่อว่า ภาษาโอ๊ค ซึ่งเป็นชื่อที่ได้มาจากต้นไม้ที่บ้านของทีมงานทำงานอยู่ ภายหลังจากที่มีการพัฒนาต่อมาเรื่อยๆ ก็ได้มีการเปลี่ยนชื่อเป็น ภาษาจาวา ตามชื่อกาแฟของทีมงานที่ใช้ดื่ม โดยการพัฒนาภาษานี้เป็นส่วนหนึ่งของ โครงการกรีน (the Green Project) และมีเป้าหมายคือ สร้างผลิตภัณฑ์อุปกรณ์อิเล็กทรอนิกส์ที่มีค่าใช้จ่ายต่ำ ใช้งานง่าย มีข้อผิดพลาดน้อยที่สุด และสามารถปรับใช้กับอุปกรณ์ใดๆ ก็ได้ ทำให้ภาษาจาวามีข้อดีเหนือกว่าภาษาอื่นๆ โดยภาษาจาวาจะสามารถนำไปใช้กับอุปกรณ์ต่างๆ โดยไม่ต้องคอมไพล์โปรแกรมใหม่ส่งผลให้ไม่จำกัดอยู่กับเครื่องหรือระบบปฏิบัติการเพียงแค่ระบบปฏิบัติการเดียว ถึงแม้ว่าภาษาจาวาในช่วงแรกจะจำกัดอยู่กับเฉพาะ WWW (World Wide Web) และ อินเทอร์เน็ต แต่ในปัจจุบันก็ได้มีการนำไปประยุกต์ใช้งานในด้านซอฟต์แวร์ต่างๆมากมาย ซึ่งมีตั้งแต่ขนาดเล็กไปจนถึงขนาดใหญ่ นอกจากนี้จาวายังนำไปใช้เป็นภาษาสำหรับอุปกรณ์ต่างๆ เช่น อุปกรณ์ ขนาดเล็ก หรือ โทรศัพท์มือถือ รวมทั้งยังได้รับความนิยมนำไปใช้กับอุปกรณ์สำหรับเข้าสู่อินเทอร์เน็ตโดยไม่ใช้คอมพิวเตอร์อีกด้วย

2.1.2 สถาปัตยกรรมของภาษาจาวา

สถาปัตยกรรมของจาวา มี 4 ส่วนด้วยกัน ดังรูปที่ 1



รูปที่ 1 สถาปัตยกรรมของภาษาจาวา

(อ้างอิงโดย <http://www.iyathai.com/>)

Java Program Language คือ โปรแกรมที่เขียนด้วยภาษา java (.java) ในรูปของ text ที่สามารถอ่านได้ เรียกว่า ซอร์สโค้ด (Source code) , Java Class File คือ ซอร์สโค้ด (Source code) ที่ถูกแปลง (compile) เป็น (.class) หรือ ไบต์โค้ด (byte code) ที่อยู่ในรูปของคำสั่งที่ Java Virtual Machine เข้าใจ , Java API คือ กลุ่มของ ready-made software components โดยจะรวมอยู่ใน ไลบรารีของคลาสและ อินเทอร์เน็ต ซึ่งสามารถนำมาใช้ได้โดยไม่ต้องเขียนเอง Java Virtual Machine (JVM) คือ ส่วนที่จะไปติดต่อสั่งงานโดยตรงต่อคอมพิวเตอร์ ภายในประกอบด้วย , Class loader ทำหน้าที่โหลด Class file จากโปรแกรมและจาก Java API , Execution engine ทำการแปล (Interpret) ไบต์โค้ด ซึ่งมีหลายรูปแบบ เช่น just in time

2.1.3 คุณสมบัติของภาษาจาวา

ภาษาจาวา เป็นภาษาที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุแบบสมบูรณ์ เหมาะสำหรับ การพัฒนาระบบที่มีความซับซ้อนยุ่งยาก นอกจากนี้ การพัฒนาโปรแกรมแบบวัตถุจะช่วยให้เราสามารถใช้อำนาจหรือชื่อต่างๆ ที่มีอยู่ในระบบงานนั้นมาใช้ในการออกแบบโปรแกรมได้ ทำให้เข้าใจได้ง่ายขึ้น ซึ่งการใช้โปรแกรมที่เขียนขึ้นโดยใช้ภาษาจาวา จะมีความสามารถในการทำงานในระบบปฏิบัติการที่ต่างกันได้ โดยไม่จำเป็นต้องดัดแปลงแก้ไขโปรแกรม ภาษาจาวามีการตรวจสอบข้อผิดพลาดในหลายๆ ขั้นตอน ไม่ว่าจะอยู่ในช่วง compile time หรือ runtime ทำให้เกิดความผิดพลาดได้ยากกว่า และช่วยให้ debug โปรแกรมได้ง่าย ภาษาจาวาเป็นภาษาที่ง่ายต่อการเรียนรู้ศึกษา โดยถูกออกแบบมาเพื่อมีความปลอดภัยสูงตั้งแต่แรก ทำให้โปรแกรมที่เขียนขึ้นด้วยจาวามีความปลอดภัยมากกว่าโปรแกรมที่เขียนขึ้นด้วยภาษาอื่น มี IDE, application server, และ library ต่างๆ มากมายสำหรับจาวาที่เรา

สามารถใช้งานได้ โดยไม่ต้องเสียค่าใช้จ่าย ทำให้เราสามารถลดค่าใช้จ่ายที่สูญเสียไปกับการซื้อ tool และ s/w ต่างๆ

2.1.4 จุดมุ่งหมายในการใช้งานภาษาจาวา

2.1.4.1 เขียนโปรแกรมในเชิงวัตถุ

2.1.4.2 โปรแกรมที่เขียนไม่ขึ้นกับ Platform ใดๆ

2.1.4.3 โปรแกรมสามารถเรียกใช้งานจากระยะไกลได้อย่างปลอดภัย

2.1.4.4 เหมาะสมกับการนำไปใช้ในระบบเครือข่าย เพราะมีไลบรารีที่เหมาะสม

2.2 ระบบปฏิบัติการแอนดรอยด์

2.2.1 ความรู้เบื้องต้นเกี่ยวกับระบบปฏิบัติการแอนดรอยด์

แอนดรอยด์ (Android) เป็นระบบปฏิบัติการที่มีพื้นฐานอยู่บนลินุกซ์ (Linux) ถูกออกแบบมาเพื่อประยุกต์ใช้กับอุปกรณ์ประเภทจอสัมผัส ซึ่งในปัจจุบันนี้ได้มีการใช้งานกับอุปกรณ์หลากหลายชนิด เช่น สมาร์ทโฟน สาเหตุดังกล่าวเนื่องจากว่าเป็นระบบปฏิบัติการแบบเปิด (Open Source) ซึ่งไม่ว่าใครก็สามารถนำระบบปฏิบัติการแอนดรอยด์ไปพัฒนาต่อได้ และระบบนี้มีบริษัท กูเกิล (Google Inc.) เป็นผู้ริเริ่มพัฒนาขึ้น จึงเป็นสาเหตุที่ทำให้ได้รับความนิยมเป็นอย่างสูง ทั้งนี้การพัฒนาโปรแกรมเพื่อใช้งานก็มี Android SDK (Software Development Kit) ซึ่งการพัฒนาโปรแกรม การที่จะใช้งาน Android SDK นั้นจะมีการใช้โครงสร้างพื้นฐานมาจากภาษาจาวา ดังนั้นผู้พัฒนานั้นจะต้องมีความรู้ของภาษาจาวาเพื่อใช้ในการเขียนโปรแกรมของระบบ อีกทั้งยังมีตลาดให้เผยแพร่โปรแกรมผ่านทาง Android Market หรือที่รู้จักกันในชื่อว่า Play Store

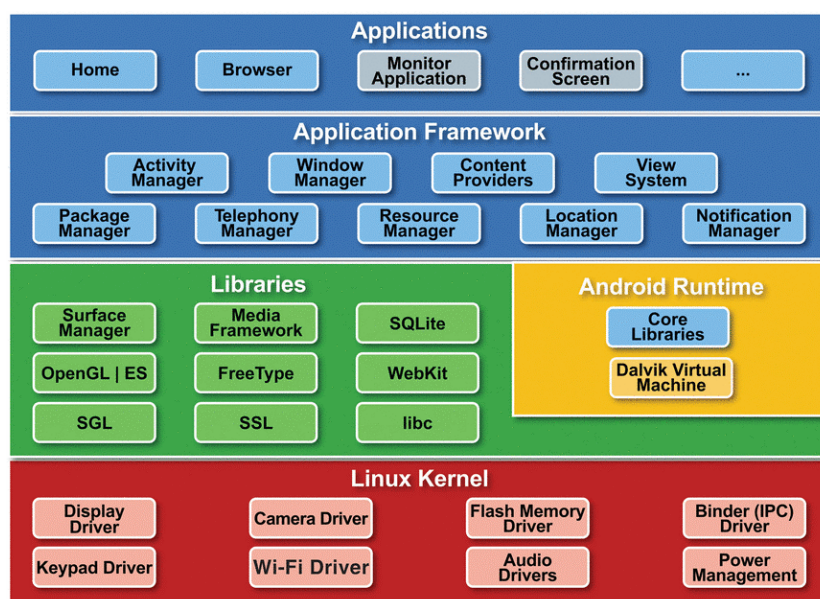
2.2.2 ประวัติความเป็นมา

ในปี พ.ศ. 2546 นายแอนดี้ รูบิน (Andy Rubin) ได้คิดค้นระบบปฏิบัติการแอนดรอยด์ขึ้นซึ่งในขณะนั้นระบบนี้ถูกพัฒนาโดยบริษัทแอนดรอยด์ (Android Inc.) จากนั้นในปี พ.ศ. 2548 บริษัทกูเกิลก็ได้เข้ามาซื้อกิจการทำให้บริษัทแอนดรอยด์ได้กลายมาเป็นบริษัทลูกของบริษัท กูเกิล และนายแอนดี้ รูบิน ยังคงเป็นผู้ดำเนินงานอยู่ในทีมพัฒนาระบบปฏิบัติการแอนดรอยด์ โดยที่ระบบปฏิบัติการนี้มีการพัฒนามาจากการนำเอา แกนกลางของระบบปฏิบัติการลินุกซ์ (Linux Kernel) ซึ่งเป็นระบบปฏิบัติการที่ออกแบบมาเพื่อทำงานเป็นเซิร์ฟเวอร์มาพัฒนาต่อเพื่อให้กลายเป็น ระบบปฏิบัติการบนอุปกรณ์พกพา (Mobile Operating System) โดยปัจจุบันเวอร์ชันต่างๆของ

ระบบปฏิบัติการแอนดรอยด์จะมีชื่อรหัสเป็นขนมหวาน และเรียงตามตัวอักษรภาษาอังกฤษ ในเดือนพฤศจิกายนปี พ.ศ. 2550 บริษัทกูเกิ้ลได้ก่อตั้งสมาคม OHA (Open Handset Alliance) เพื่อเป็นหน่วยงานกลางในการกำหนดมาตรฐานของอุปกรณ์พกพาและระบบปฏิบัติการแอนดรอยด์ อีกทั้งยังเป็นที่ยู่อักครั้งแรกต่อสาธารณชนอีกด้วย และในเดือนกันยายนปี พ.ศ. 2551 ได้ใช้งานระบบปฏิบัติการแอนดรอยด์รุ่น 1.0 ครั้งแรก

2.2.3 โครงสร้างของแอนดรอยด์

นักพัฒนาแอนดรอยด์มีสิ่งจำเป็นที่จะต้องรู้คือ โครงสร้างของระบบปฏิบัติการแอนดรอยด์ โดยโครงสร้างดังกล่าวจะถูกแบ่งออกเป็นส่วนๆ ดังรูปที่ 2



รูปที่ 2 สถาปัตยกรรมของแอนดรอยด์

(อ้างอิงโดย <http://satworks.blogspot.com>)

ส่วนแรกคือส่วน Linux kernel เป็น Hardware Abstraction Layer ซึ่งเป็นตัวกลางระหว่างฮาร์ดแวร์กับส่วนของซอฟต์แวร์ที่อยู่ถัดขึ้นไป ทำหน้าที่บริหารจัดการทรัพยากรต่างๆของเครื่อง ส่วนต่อมาคือส่วนของ Libraries เป็นส่วนที่เขียนด้วย C หรือ C++ ทั้งหมด โดยเป็นส่วนที่เก็บรวบรวม Libraries ต่างๆไว้ใช้ในโค้ด อีกส่วนหนึ่งก็คือส่วนของ Android Runtime ซึ่งประกอบไปด้วย Core Library และ Dalvik Virtual Machine โดย Dalvik Virtual Machine จะมีหน้าที่ในการรันโค้ดแอนดรอยด์ต่างๆ ส่วน Core Library จะเป็น Java Library ซึ่งจะเหมือนกับใน Java Standard Edition (Java SE) แต่บางไลบรารีใน Java SE จะไม่มีในแอนดรอยด์ ส่วนถัดมาคือส่วน

Application Framework เป็นส่วนของเฟรมเวิร์ก (Framework) ที่ไว้ใช้ในการพัฒนาแอปพลิเคชันต่างๆ ส่วนสุดท้ายคือส่วนของ Application เป็นส่วนบนสุดของสถาปัตยกรรมแอนดรอยด์กล่าวคือ ส่วนนี้คือส่วนที่เป็นแอปพลิเคชันต่างๆไม่ว่าจะเป็นแอปพลิเคชันที่ติดมากับเครื่อง หรือจะเป็นแอปพลิเคชันที่สร้างขึ้นเอง

2.2.4 จุดเด่นของแอนดรอยด์

ระบบปฏิบัติการแอนดรอยด์เป็นระบบปฏิบัติการซึ่ง มีนักพัฒนาโปรแกรมและกลุ่มผู้ใช้งานที่ให้ความสำคัญกับระบบปฏิบัติการแอนดรอยด์เพิ่มมากขึ้น เนื่องมาจากระบบปฏิบัติการนี้มีการเจาะตลาดของการค้าได้กับกลุ่มบุคคลในหลายๆระดับ อีกทั้งผู้ที่ทำผลิตภัณฑ์มีการคิดค้นอุปกรณ์ หรือ โปรแกรมใหม่ๆอยู่อย่างต่อเนื่อง และได้มีการนำเอาระบบปฏิบัติการแอนดรอยด์ไปใช้ในสินค้าโดยเฉพาะอย่างยิ่ง กลุ่มสินค้าที่เป็นมือถือนรุ่นใหม่และอุปกรณ์จอสัมผัส หรือจะเป็นอุปกรณ์อื่นๆ เช่น นาฬิกา เป็นต้น โดยประสิทธิภาพของอุปกรณ์แต่ละเครื่องจะแตกต่างกันออกไป เช่น ขนาดหน้าจอ ระบบโทรศัพท์ ความเร็วของหน่วยประมวลผล ปริมาณหน่วยความจำ แม้กระทั่งอุปกรณ์ตรวจจับต่างๆ หากมองในด้านของการพัฒนาโปรแกรม ทางบริษัท กูเกิ้ล ได้มีการพัฒนาโครงสร้างของแอปพลิเคชันไว้สำหรับนักพัฒนาใช้งานได้อย่างสะดวก ทั้งยังมีโปรแกรม Android Studio ให้นักพัฒนาได้เลือกใช้ ทำให้ไม่เกิดปัญหาเมื่อนำชุดโปรแกรมที่พัฒนาขึ้นมาไปใช้กับอุปกรณ์ ที่มีคุณลักษณะที่แตกต่างกัน

2.3 แอนดรอยสตูดิโอ

เป็น IDE Tools เวอร์ชันใหม่ล่าสุดที่ทาง Google ที่ได้ปล่อยออกมา โดยมีจุดประสงค์คือการนำเอาไปใช้ในการพัฒนาโปรแกรม Android โดยเฉพาะ โดยได้รับการพัฒนาจากแนวคิดพื้นฐาน IntelliJ IDEA ซึ่งมีการทำงานคล้าย ๆ กับการทำงานของ Eclipse และ Android ADT Plugin ส่วนจุดประสงค์ของ Android Studio นั้นคือต้องการพัฒนาเครื่องมือ IDE ที่สามารถพัฒนา App บน Android ให้มีประสิทธิภาพเพิ่มมากขึ้น ทั้งด้านการออกแบบ GUI ที่ช่วยให้สามารถจำลองตัว App ในมุมมองที่ไม่เหมือนกันบน Smart Phone แต่ละรุ่น และสามารถแสดงผลบางอย่างได้ทันท่วงทีโดยไม่ต้องทำการรัน App บน อีมูเลเตอร์ (Emulator) ดังรูปที่ 3



รูปที่ 3 โปรแกรมแอนดรอยด์สตูดิโอ

(อ้างอิงโดย <https://lh3.googleusercontent.com>)

2.3.1 จุดเด่นของแอนดรอยด์สตูดิโอ

- 2.3.1.1 สามารถจำลองหน้า UI (User Interface) ได้หลายหน้า
- 2.3.1.2 มี Memory Monitoring เอาไว้ตรวจสอบการใช้งานพื้นที่ความจำของแอปพลิเคชัน
- 2.3.1.3 มีโมดูล (Module) ที่หลากหลาย
- 2.3.1.4 มีความฉลาดมากกว่า อีคลิปส์ (Eclipse)
- 2.3.1.5 ใช้งานง่าย และเป็นที่นิยมใช้ในปัจจุบัน

2.4 ภาษา PHP

PHP ย่อมาจาก Personal Home Page Tools คือภาษาที่ใช้ในการเขียนโปรแกรมทางฝั่งเซิร์ฟเวอร์ โดยภาษา PHP เป็นภาษาจำพวก scripting language ซึ่งเป็นภาษาที่จะเก็บคำสั่งต่างๆไว้ในไฟล์ script และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างภาษาสคริปต์ก็อย่างเช่น Perl ,JavaScript เป็นต้น PHP ต่างจากภาษาสคริปต์อื่นๆตรงที่ได้รับการออกแบบพัฒนามา เพื่อใช้งานในการสร้าง เอกสาร HTML ต่างๆ โดยสามารถสอดแทรกโค้ดหรือแก้ไขโค้ดได้โดยอัตโนมัติ และด้วยความที่ PHP เป็นภาษา server-side ทุกๆครั้งที่เซิร์ฟเวอร์ให้บริการ จะประมวลผลไฟล์ PHP ก่อนแล้วจึงค่อยส่งผลลัพธ์กลับไปให้เว็บเบราว์เซอร์ PHP ถือว่าเป็นเครื่องมือที่สำคัญชนิดหนึ่ง ที่เอาไว้สำหรับสร้าง Dynamic Web pages ให้มีประสิทธิภาพและมีลูกเล่นมากขึ้น

2.4.1 ประวัติของภาษา PHP

PHP ถูกคิดค้นขึ้นในปี ค.ศ.1994 โดย Rasmus Lerdorf ต่อมาได้รับความนิยมเป็นจำนวนมาก จึงได้ออกเป็นแพ็คเกจ "Personal Home Page" ซึ่งเป็นที่มาของ PHP ในปี ค.ศ.1995 ได้มีการพัฒนาตัวแปลภาษา PHP ขึ้นมาใหม่ โดยใช้ชื่อว่า PHP/FI เวอร์ชัน 2 ซึ่งได้เพิ่มฟังก์ชันในการรับข้อมูลที่ส่งมาจากฟอร์มของ HTML (จึงมีชื่อว่า FI หรือ Form Interpreter) นอกจากนี้ยังเพิ่มความสามารถในการติดต่อกับฐานข้อมูลอีกด้วย จึงทำให้ผู้คนเริ่มหันมาสนใจ PHP กันมากขึ้น และในปี 1997 มีผู้ร่วมพัฒนา PHP เพิ่มอีก 2 คน ก็คือ Zeev Suraski และ Andi Gutmans โดยได้แก้ไขจุดบกพร่องต่างๆ และเพิ่มเติมเครื่องมือของ PHP ให้มากขึ้น

2.4.2 จุดเด่นของภาษา PHP

เป็นภาษาที่มีลักษณะเป็นแบบ Open source ทำให้ไม่ต้องเสียค่าใช้จ่ายในการใช้งานในภาษา PHP และเป็นโปรแกรมทางฝั่งเซิร์ฟเวอร์ทำให้มีขีดความสามารถที่สูง และ PHP ยังเป็นภาษาที่ทำงานได้ในหลายๆ ระบบปฏิบัติการไม่ว่าจะเป็น Windows, Mac OS หรือ Unix และ PHP ยังสามารถทำงานกับ Web server ได้หลายชนิดไม่ว่าจะเป็น Apache, OmniHttpd, Internet Information Service(IIS) เป็นต้น และนอกจากนี้ภาษา PHP ยังเป็นภาษาที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ ทำให้โปรแกรมมีความยืดหยุ่น และสามารถทำความเข้าใจได้ง่าย

2.5 MySQL

MySQL คือ ระบบจัดการฐานข้อมูล ที่มีหน้าที่ในการเก็บข้อมูลต่างๆที่โปรแกรมเมอร์ได้สร้างขึ้น โดย MySQL จะใช้ภาษา SQL ในการจัดการฐานข้อมูลโดยเฉพาะ โดยการจัดการฐานข้อมูลนั้นมีหลายแบบไม่ว่าจะเป็นการสร้างฐานข้อมูล การเพิ่มข้อมูล การลบข้อมูล และการแก้ไขข้อมูล และ MySQL ยังมีหน้าที่ในการเป็นตัวกลางระหว่างผู้ใช้กับฐานข้อมูล ซึ่ง MySQL เป็นระบบจัดการฐานข้อมูล (Database management system) สำหรับฐานข้อมูลเชิงสัมพันธ์ โดยสามารถติดต่อ MySQL ผ่านโปรแกรมภาษาต่างๆได้ไม่ว่าจะเป็น ภาษา PHP, Java, C++ เป็นต้น

2.5.1 ประวัติ MySQL

MySQL เป็นระบบจัดการฐานข้อมูล (Relational Database Management System) ที่ใช้ภาษา SQL แม้ว่า MySQL จะเป็นซอฟต์แวร์โอเพนซอร์ส (Open Source) แต่มีความแตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ที่ประเทศสวีเดน โดยจัดการ MySQL ทั้งในแบบที่ใช้ฟรี และแบบที่ใช้ใน

เชิงธุรกิจ และ MySQL ได้ถูกคิดค้นขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius. ปัจจุบันบริษัทซันไมโครซิสเต็มส์ได้เข้าซื้อกิจการของ MySQL AB เรียบร้อยแล้ว ฉะนั้นผลิตภัณฑ์ภายใต้ MySQL AB ทั้งหมดจะตกเป็นของบริษัท ซันไมโครซิสเต็มส์

2.5.2 ข้อดีของ MySQL

2.5.2.1 สามารถนำซอร์สโค้ดมาพัฒนาต่อยอดได้

2.5.2.2 ใช้ได้กับทุกระบบทุกแพลตฟอร์ม

2.5.2.3 ใช้ทรัพยากรเครื่องน้อยแต่คุณภาพสูง

2.5.2.4 เซฟเวอร์ (Server) ส่วนใหญ่รองรับ MySQL

2.5.2.5 เนื่องจากเป็นที่นิยมจึงหาข้อมูลการใช้งานได้

2.6 ไบรารี Tesseract

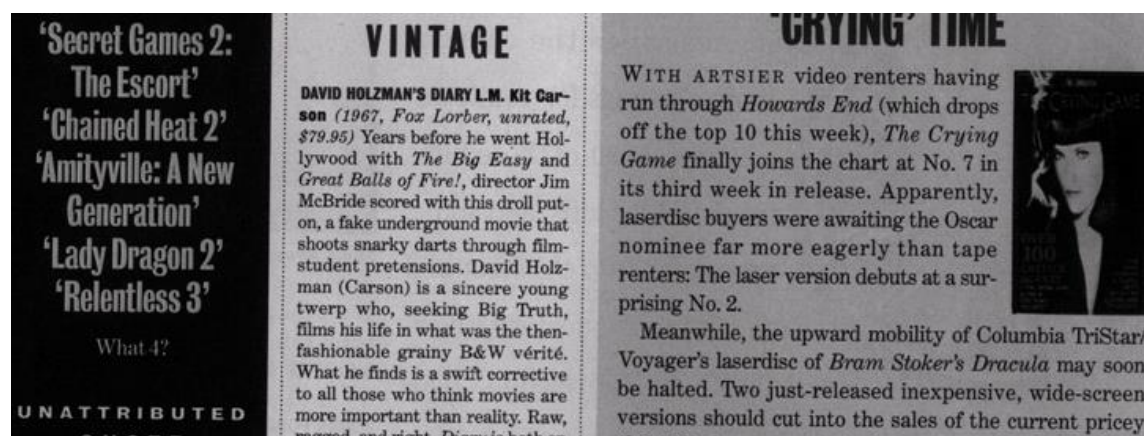
2.6.1 ประวัติ Tesseract

Tesseract เป็น open-source OCR engine ที่ถูกพัฒนาโดย บริษัท HP ระหว่างปี 1984-1994 Tesseract ได้รับชื่อเสียงเป็นอย่างมากในงาน UNLV Annual Test of OCR Accuracy ซึ่งเป็นงานประกวดแข่งขันความแม่นยำของการทำ OCR ในปี 1995 เนื่องจากมีความแม่นยำที่สูง

Tesseract เริ่มจากการเป็น งานวิจัยในระดับปริญญาเอกของ HP Labs หลังจากนั้นก็ถูกผลักดันเนื่องจากทาง HP เห็นความเป็นไปได้ในการนำมาใช้กับเครื่อง scanner แต่โปรเจกต์ไม่ประสบความสำเร็จเท่าที่ควรจึงไม่ได้รับการพัฒนาต่อ เนื่องจากอัลกอริทึมซึ่งคิดค้นขึ้นมาอ่านค่าได้จากสิ่งพิมพ์ที่มีคุณภาพดีเลิศเท่านั้น แต่ผลจากการผลักดันทำให้ tesseract มีความแม่นยำสูงกว่าอุปกรณ์อื่นๆ บนตลาดมาก แต่กลับไม่สามารถนำไปทำเป็นผลิตภัณฑ์ได้ หลังจากนั้นก็มีความพยายามที่จะพัฒนาเพื่อสร้างสินค้าลงตลาดอีกหลายครั้งแต่ก็ไม่สำเร็จในปี 2005 ทางบริษัท HP จึงได้ปล่อย tesseract ออกมาเป็น open source

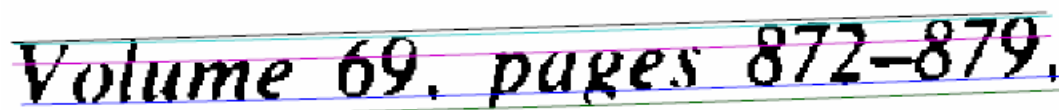
2.6.2 หลักการทำงาน

2.6.2.1 Adaptive Thresholding การเปลี่ยน input image ให้เป็น binary image แบบที่สีขาวดำ แยกกันอย่างชัดเจน ดังรูปที่ 4



รูปที่ 4 ภาพตัวอย่างการทำ Adaptive Thresholding

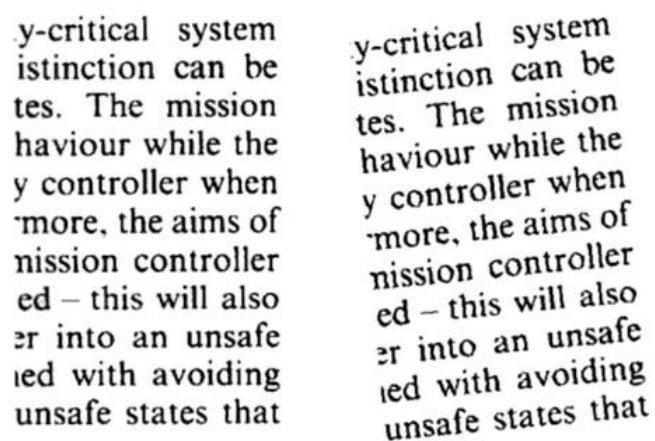
2.6.2.2 หาเส้นบรรทัด (baseline) ด้วย สมการ quadratic splines จากทฤษฎี skew และ curl หลังจากนั้นหา ascender line และ descender line จากการกระจัดจากเส้น baseline ดังรูปที่ 5



รูปที่ 5 Ascender Descender และ Base Lines

จากรูปจะพบว่ามี baseline 2 ชุด โดยเส้นดำ 2 เส้นเกิดจาก Connected component analysis หลังจาก ทำกระบวนการ filter blob ก็พบว่า เส้นดำไม่ใช่ baseline ที่แท้จริง โดยเส้นสีเขียว ม่วง และน้ำเงิน จะได้หลังจากการทำ Sort blobsจาก นั้นจึงทำ initial rows จะได้ baseline ที่แท้จริงคือเส้นสีน้ำเงินและเส้นสีเขียว ส่วนเส้นสีม่วงคือ ascender line ที่แสดงถึงตัวอักษรบรรทัดเดียวกันแต่ font มีขนาดเล็กกว่า

2.6.2.2.1 SKEW คือการหมุนรูปที่เอียงให้กลับมาอยู่ในองศาปกติ ดังรูปที่ 6



รูปที่ 6 รูปที่เอียงจากการถ่าย (ขวา) และ รูปต้นแบบ (ซ้าย)

โดยอัลกอริทึมในการเลือกองศาในการหมุนรูป ดังที่แสดงในรูปที่ 7

```

for theta = -angle to +angle by resolution do {
  for r = 0 to NROWS do
    for c = 0 to NCOLS do
      if (image[r][c] == BLACK) {
        rotate(r,c,theta,&new_row);
        ++proj_prof[new_row];
      }

      angle_measure[theta] = criterion_function(proj_prof[]);
    }

  skew_angle = choose_skew(angle_measure[]);
}

```

รูปที่ 7 อัลกอริทึมในการหาองศาเพื่อหมุนรูป

2.6.2.2.2 ทำการหา baseline

2.6.2.2.2.1 Connected component analysis เป็นการอ่าน pixel รูปทีละจุด แล้วหาจุดที่มีค่าความเข้มสีที่ใกล้เคียงกันแล้วลากเส้นต่อเรียงกันเป็นแถว มีวิธีการแบ่งแยกดังนี้

ให้ \mathcal{N}_s เป็น neighborhood system และ ให้ $c(s)$ เป็น set of neighbors ที่เชื่อมกับจุด s .

For all s and r , the set $c(s)$ ต้องมีคุณสมบัติดังนี้

– $c(s) \subset \mathcal{N}_s$

– $r \in c(s) \Leftrightarrow s \in c(r)$

จะได้เซตของจุดที่มีความเข้มสีใกล้เคียงกันมาเรียกว่า blob

2.6.2.2.2.2 Filter blobs คือกรอง blob ที่มีความต่างด้าน property ของ pixel ออกไป เช่น มีความสว่างที่แตกต่างกัน สีที่ต่างกัน เป็นต้น

2.6.2.2.2.3 Sort blobs คือการเรียง blob ที่มีค่าความเข้มสีใกล้เคียงกัน โดยมีสมมุติฐานว่า blob ที่มีค่าความเข้มสีใกล้เคียงกันน่าจะเป็นจุดในภาพต้นแบบที่อยู่ใกล้กัน เมื่อผ่านกระบวนการ sort blob จะได้รูปที่มีจุดที่ค่า blob ใกล้กันเป็นแถวๆ

2.6.2.2.2.4 ทำ initial rows เพื่อนำค่าที่เรียงเสร็จแล้วมาหาแถวแรกของบรรทัด จะทำให้ได้ภาพที่ใกล้เคียงกับภาพต้นแบบมากที่สุด ดังรูปที่ 8

Set the running average y shift to zero.

For each blob in sorted order:

Find the existing row which has most vertical overlap
with the blob

If there is no overlapping row

Then

Make a new row and put the blob in it.

Record the top and bottom coordinates of the blob
as the top and bottom of the row.

Else

Add the blob to the row.

Expand the top and bottom limits of the row with
the top and bottom of the blob, clipping the row
height to a limit.

Update the running average y shift with the
bottom of the blob.

Endif

Endfor

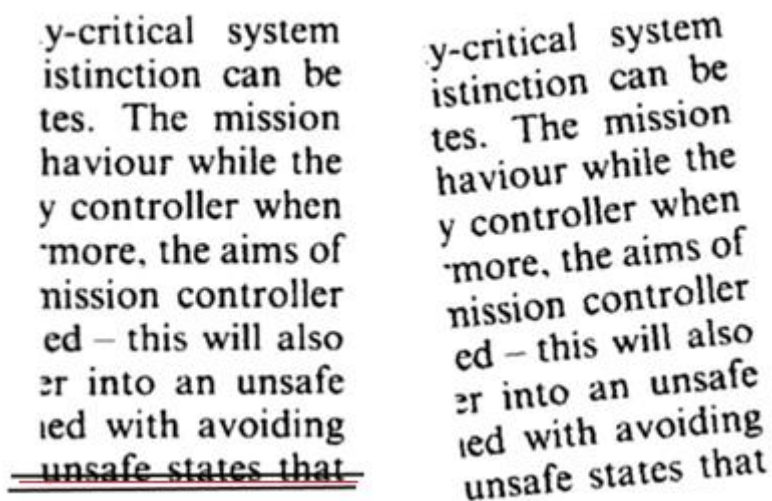
รูปที่ 8 อัลกอริทึมการทำ initial row

จากขั้นตอนที่ 2.6.2.2.1 ถึงขั้นตอนที่ 2.6.2.2.4 จะทำให้รูปที่ได้ ไม่เกิดการโค้งงอ ดังรูปที่ 9



รูปที่ 9 ภาพที่เกิดการโค้งงอ (ทางด้านซ้าย) ภาพที่ไม่เกิดการโค้งงอ (ทางด้านขวา)

2.6.2.2.5 หาเส้นแบ่งบรรทัดล่างของทุกบรรทัดโดยใส่ baseline จากการหาแนวของ blob ที่มีสีตัวอักษรใกล้เคียงกัน จากรูปที่ 10 เส้น baseline ด้านบนวิ่งผ่านเส้นสีดำ ส่วนด้านล่างผ่านเส้นสีขาวทั้งหมดซึ่งนำมาใช้คำนวณหาเส้น baseline ที่แท้จริง โดยเส้น baseline ที่แท้จริง (เส้นสีแดง) จะอยู่ตรงกลางของเส้นทั้งสอง ดังรูปที่ 10



รูปที่ 10 รูปต้นแบบที่มีการจัดวางที่ถูกต้อง (ซ้าย) รูปที่หมุน 5 องศา (ขวา)

2.6.2.3 CURL คือหาช่องว่างของแต่ละตัวอักษรและสับแยก ดังรูปที่ 11

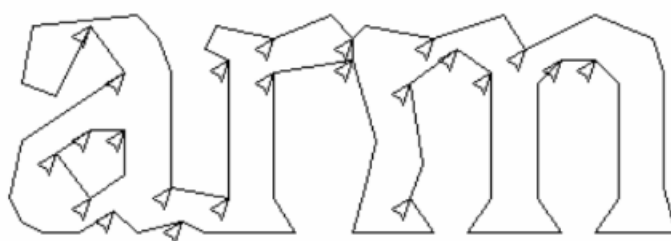


รูปที่ 11 การตัดแบ่งคำ

2.6.3 วิธีการตรวจจับคำศัพท์

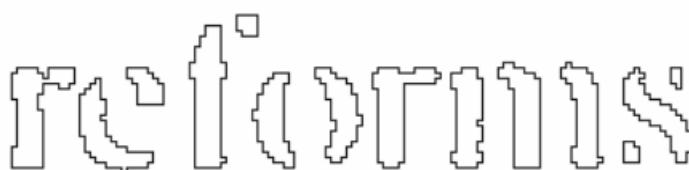
2.6.3.1 การตัดตัวอักษรในคำที่ติดกันโดยใช้วิธีการตรวจจับขอบของตัวอักษรแต่ละตัว

ดังรูปที่ 12



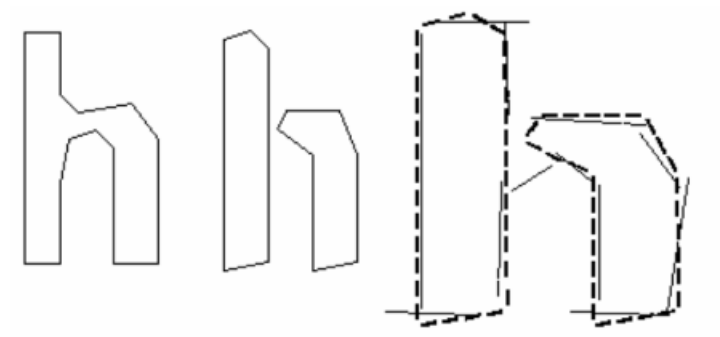
รูปที่ 12 ภาพการตัดอักษร

2.6.3.2 นำตัวอักษรที่ตัดได้ซึ่งมีความเสียหายบางส่วนมาประกอบ ดังรูปที่ 13



รูปที่ 13 การประกอบตัวอักษร

โดยมีหลักการคือตัดแยกตัวอักษรใน prototype (ต้นแบบที่ถูกต้อง) ที่มีความคล้ายคลึงกับ input ที่ได้รับมาแล้ว
จึงนำตัวอักษร prototype ที่คัดแยกแล้วมาเทียบกับ training data แล้วหาอักษรที่ใกล้เคียงที่สุดดังรูปที่ 14



รูปที่ 14 ตัวอักษรต้นแบบที่ถูกตัด (ซ้าย) ตัวอักษรที่โดนตัดเป็นตัว (กลาง) การนำตัวอักษรที่โดนตัดมาเทียบกับ
ตัวอักษรต้นแบบ (ขวา)