

Fedora Linux Core 4

นายฉัตรชัย ดำดี (46320388)
นายทรงยศ คชนิล (46320511)
นายอรรณพ สุวัฒนพิเศษ (46321105)
นายอุทิศ ศักดิ์สิทธิ์ (46321139)

คณะวิทยาศาสตร์ เอกวิทยาการคอมพิวเตอร์ ชั้นปีที่ 3 ปีการศึกษา พ.ศ. 2548
มหาวิทยาลัยนเรศวร

สารบัญ

1 ประวัติความเป็นมาของ Linux	3
1.1 ประวัติของ Unix	3
1.2 ประวัติของ Linux	4
1.3 ประวัติ Fedora Linux	6
2 การประยุกต์ใช้งาน OS	8
3 ความต้องการของ Hardware	10
3.1 CPU	10
3.2 Hard Drive	10
3.3 Memory	11
3.4 อื่น ๆ	11
4 ระบบ RAID	12
4.1 ประวัติของ RAID	12
4.2 ประโยชน์ของ RAID	12
4.3 จะเรียก Inexpensive หรือ independent ?	12
4.4 ชนิดของ RAID ที่สนับสนุน	13
5 ระบบ File System	14
5.1 ext2 หรือ Second extended file system	14
5.2 ext3 หรือ Third extended file system	14
6 ระบบ Memory Management	16
ก เอกสารอ้างอิง	18

สารบัญรูป

1.1	แผนภาพตระกูลของ Unix และ Unix Compatible (รูปจาก http://en.wikipedia.org/wiki/Unix)	4
1.2	Linus Torvalds, ผู้ให้กำเนิด Linux kernel (รูปจาก http://en.wikipedia.org/wiki/Linux)	4
1.3	Richard Matthew Stallman, ผู้ก่อตั้งโครงการ GNU (รูปจาก http://en.wikipedia.org/wiki/Linux)	5
1.4	Andrew S. Tanenbaum, Professor of Computer Science, Vrije University	5
1.5	Red Hat Enterprise Linux 4 (รูปจาก http://en.wikipedia.org/wiki/Red_Hat_Enterprise_Linux)	6
1.6	Fedora Core 4 ทำงานบน GNOME desktop environment (รูปจาก http://en.wikipedia.org/wiki/GNOME)	7
2.1	GNOME desktop environment (รูปจาก http://en.wikipedia.org/wiki/GNOME)	8
2.2	KDE desktop environment	8
5.1	Ext File System Layout	15
6.1	Logical address in Linux	16

บทที่ 1

ประวัติความเป็นมาของ Linux

ก่อนที่จะเรารู้ประวัติของ Fedora Linux Core 4 เราควรจะได้รู้ประวัติของ Unix และ Linux โดยทั่วไปเสียก่อน

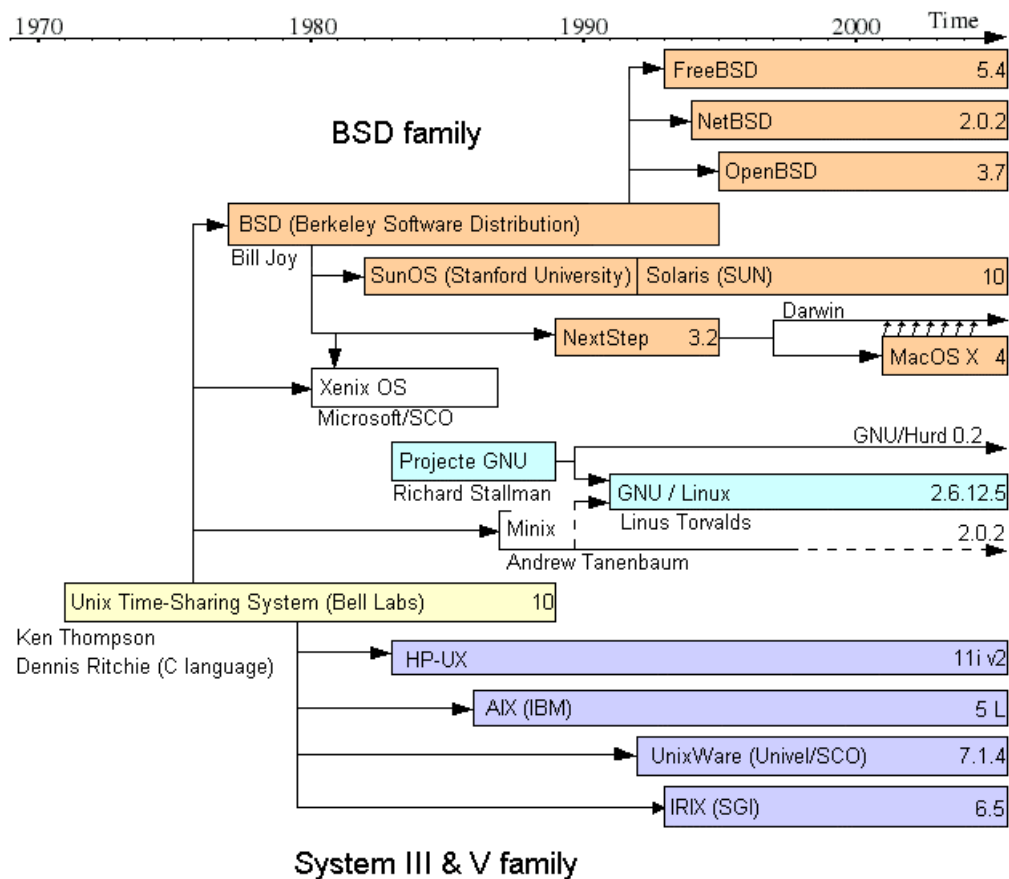
1.1 ประวัติของ Unix

ระบบปฏิบัติการ Unix มีต้นกำเนิดจากห้องปฏิบัติการวิจัย Bell Labs (ใน AT&T) ประเทศสหรัฐอเมริกา โดย Ken Thompson และ Dennis Ritchie ปี 1969 โดยการร่วมมือของ สถาบัน MIT (Massachusetts Institute of Technology), ห้องปฏิบัติการวิจัย AT&T Bell Labs และบริษัท GE (General Electric) เพื่อพัฒนาโครงการ Multics ในปี 1960 เพื่อพัฒนาระบบปฏิบัติการสำหรับ Mainframe Computer รุ่น GE 635 โดยให้มีความสามารถในการโต้ตอบ (Interactive) มีระบบอำนวยความสะดวกต่อการใช้แฟ้มข้อมูลร่วมกัน แต่เกิดปัญหาหลายประการ จนกระทั่งห้องปฏิบัติการวิจัย Bell Labs ได้ลาออกจากโครงการไป แต่โครงการก็ยังดำเนินการต่อโดย Ken Thompson และ Dennis Ritchie ซึ่งทำงานกับ Bell Labs พร้อม ๆ กันไปด้วย ต่อมา Ken & Dennis ได้ร่วมกันพัฒนาระบบปฏิบัติการใหม่ เพื่อทำงานบนเครื่อง PDP-7 และใช้ชื่อว่าระบบปฏิบัติการ Unix เพื่อให้ออกเสียงใกล้เคียงกับระบบ Multics ดังนั้นต้นกำเนิดของ Unix ก็คือ Multics นั่นเอง ไม่ว่าจะเป็นชื่อ ระบบแฟ้มข้อมูลที่ใช้ แนวคิดของตัวแปรคำสั่ง (Shell) หลังจากนั้นทั้งสองได้พัฒนามาเป็น Version 2 เพื่อทำงานบนเครื่องรุ่น PDP-11/20 โดยใช้ภาษา Assembly และได้พัฒนาปรับปรุงด้วยภาษา C (ภาษา C ก็พัฒนาที่ห้องวิจัย Bell Labs เช่นกัน เพื่อทำงานบนระบบ Unix) และเผยแพร่ไปสู่มหาวิทยาลัยต่าง ๆ ด้วย Version 6 ในปี ค.ศ. 1976

ในปี ค.ศ. 1978 Version 7 ก็พัฒนาออกมาและเป็นต้นแบบของระบบ Unix รุ่นใหม่ ๆ หลังจากนั้น AT&T ซึ่งเป็นองค์กรแม่ของ Bell Labs ได้เป็นผู้รับผิดชอบ และควบคุมการออกตัวระบบปฏิบัติการ Unix ดังนั้นจึงกลายเป็นผลิตภัณฑ์แทนที่จะเป็นเครื่องมือวิจัย AT&T ได้พัฒนา Unix ออกมาใช้งานภายนอก ภายใต้ชื่อ System III ในปี 1982 และปี 1983 ก็ออก System V และพัฒนามาเรื่อย ๆ จนได้รับความนิยมในปัจจุบัน หลังจากนั้นก็มีผู้พัฒนา Unix เพิ่มขึ้นมา เช่น University of California at Berkley ได้พัฒนา BSD Unix (Berkley Software Distribution) ต่อมาหน่วยงานกระทรวงกลาโหมของสหรัฐฯ (Defense Advanced Research Projects Agency - DARPA) ได้ให้ทุนกับ Berkley ในการพัฒนา Unix และเกิด BSD Version 4 เพื่อสนับสนุนเครือข่ายของ DARPA ที่ใช้โปรโตคอลในการสื่อสาร คือ TCP/IP Version ล่าสุดของ Berkley คือ BSD Version 4.4 ที่ออกมาในปี ค.ศ. 1993 โดยมีความสามารถสนับสนุน Protocol X.25 หลังจากนั้น Berkley ก็หยุดการพัฒนา Unix นอกจาก Berkley ยังมีผู้พัฒนารายอื่น เช่น บริษัทซันไมโครซิสเต็ม ก็ได้พัฒนา Sun OS และ Solaris บริษัท DEC ได้พัฒนา Ultrix และเปลี่ยนชื่อเป็น OSF/1 บริษัทไมโครซอฟต์พัฒนา XENIX บริษัทไอบีเอ็มพัฒนา AIX แต่ไม่ว่าจะเป็นค่ายใดก็ตาม ต่างก็ยึดแนวทางของ BSD หรือไม่ก็ System V ทั้งนี้ ปัจจุบัน Unix เป็นเครื่องหมายการค้าจดทะเบียน (Registered Trademark) ของหน่วยงานที่ชื่อ The Open Group ซึ่งจะทำหน้าที่กำหนด และรับรองมาตรฐานของระบบปฏิบัติการ Unix

ระบบปฏิบัติการ Unix มี 2 ลักษณะ คือ

1. ระบบปฏิบัติการที่ได้มาตรฐาน Unix เป็นระบบปฏิบัติการที่ใช้มาตรฐานของ The Open Group ในการพัฒนาขึ้นมา เช่น Digital Unix, SCO Unix, IBM's Open Edition เป็นต้น (รูปที่ 1.1)
2. ระบบปฏิบัติการคล้าย Unix (Unix Compatible, Unix like systems หรือ Unix Clone) เป็นระบบที่มีลักษณะคล้ายระบบ Unix แต่ไม่ได้จดทะเบียนรับรองเป็นทางการ เช่น Sun Solaris, IBM AIX, Linux เป็นต้น (รูปที่ 1.1)



รูปที่ 1.1: แผนภาพตระกูลของ Unix และ Unix Compatible (รูปจาก <http://en.wikipedia.org/wiki/Unix>)

1.2 ประวัติของ Linux

ลินุกซ์ ถือกำเนิดขึ้นในฟินแลนด์ ปี ค.ศ. 1980 โดย Linus Trovalds (รูปที่ 1.2) นักศึกษาภาควิชาวิทยาการคอมพิวเตอร์ (Computer Science) ในมหาวิทยาลัย เฮลซิงกิ

ปี ค.ศ. 1983 Richard Matthew Stallman (รูปที่ 1.3) ได้ก่อตั้งโครงการ GNU ขึ้น ซึ่งถือว่าโครงการนี้เป็นส่วนหนึ่งของระบบ Linux โดยจุดมุ่งหมายโครงการ GNU คือ ต้องการพัฒนาระบบปฏิบัติการ Unix ให้ทำงานได้อย่างมีประสิทธิภาพ สมบูรณ์มากขึ้น การเริ่มต้นการพัฒนามีตั้งแต่ปี ค.ศ. 1990 หลังจากที่โครงการนี้ถูกละทิ้งและถูกคัดค้านนานมาก จนกระทั่ง Thomas Bushnell ได้ริเริ่มพัฒนาปรับปรุง The BSD 44-Lite Kernel ขึ้นมา

ในระหว่างนั้นในปี ค.ศ. 1991 โครงการนี้ได้กลายเป็นงานอดิเรก หลังจากสำเร็จการศึกษามหาวิทยาลัยแล้ว Trovalds ถือเป็นตัวแบบในการพัฒนา Minix ในรุ่น 0.01 นี้ถือว่ามีเครื่องมือที่เพียงพอสำหรับระบบ POSIX ที่ใช้เรียก Linux ที่รันบน GNU Bash Shell และมีการพัฒนาอย่างต่อเนื่องและอย่างรวดเร็ว



รูปที่ 1.2: Linus Torvalds, ผู้ให้กำเนิด Linux kernel (รูปจาก <http://en.wikipedia.org/wiki/Linux>)



รูปที่ 1.3: Richard Matthew Stallman, ผู้ก่อตั้งโครงการ GNU (รูปจาก <http://en.wikipedia.org/wiki/Linux>)



รูปที่ 1.4: Andrew S. Tanenbaum, Professor of Computer Science, Vrije University

ปัจจุบัน Trovalds ยังคงมุ่งมั่นพัฒนาระบบต่อไป เพื่อให้สามารถรันบน X วินโดว์ ได้ และต้องแมสคอสยังคงให้นักแพนกวินอยู่เป็นสัญลักษณ์แทนระบบ Linux ข้อความที่ ลินุส ทอร์วัลดส์ ได้เผยแพร่ไว้ในกลุ่มข่าว comp.os.minix ซึ่งถือเป็นจุดเริ่มต้นการพัฒนาไปสู่ระบบปฏิบัติการ Linux

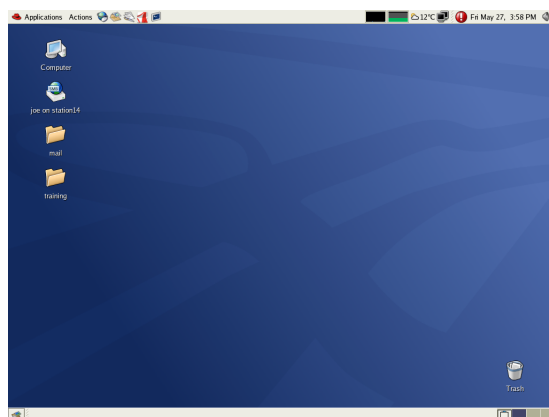
ลินุส เห็นว่าระบบมินิกซ์ (Minix) ที่เป็นระบบยูนิกซ์บนพีซีในขณะนั้น ซึ่งทำการพัฒนาโดย ศ.แอนดรูว์ ทาเนนบาวม (Andrew S. Tannenbaum) (รูปที่ 1.4) ยังมีความสามารถไม่เพียงพอแก่ความต้องการ จึงได้เริ่มต้นทำการพัฒนาระบบยูนิกซ์ของตนเองขึ้นมา โดยจุดประสงค์อีกประการ คือต้องการทำความเข้าใจในวิธาระบบปฏิบัติการคอมพิวเตอร์ด้วยเมื่อเขาเริ่มพัฒนาลินุกซ์ ไปช่วงหนึ่งแล้ว เขาก็ได้ทำการชักชวนให้นักพัฒนาโปรแกรมอื่น ๆ มาช่วยทำการพัฒนาลินุกซ์ ซึ่งความร่วมมือส่วนใหญ่ก็จะเป็นความร่วมมือผ่านทางอินเทอร์เน็ต

ลินุส จะเป็นคนรวบรวมโปรแกรมที่ผู้พัฒนาต่าง ๆ ได้ร่วมกันทำการพัฒนาขึ้นมาและแจกจ่ายให้ทดลองใช้เพื่อทดสอบหาข้อบกพร่อง ที่น่าสนใจก็คืองานต่าง ๆ เหล่านี้ผู้คนทั้งหมดต่างก็ทำงานโดยไม่คิดค่าตอบแทน และทำงานผ่านอินเทอร์เน็ตทั้งหมด

ปัจจุบันเวอร์ชันล่าสุดของระบบลินุกซ์ที่ได้ประกาศออกมาคือเวอร์ชัน 2.6.14 (01/11/05) ข้อสังเกตในเรื่องเลขรหัสเวอร์ชันนี้ก็คือ ถ้ารหัสเวอร์ชันหลังทศนิยมตัวแรกเป็นเลขคู่เช่น 1.0.x, 1.2.x เวอร์ชันเหล่านี้จะถือว่าเป็นเวอร์ชันที่เสถียรแล้วและมีความมั่นคงในระดับหนึ่ง แต่ถ้าเป็นเลขคี่เช่น 1.1.x, 1.3.x จะถือว่าเป็นเวอร์ชันทดสอบ ซึ่งในเวอร์ชันเหล่านี้จะมีการเพิ่มเติมความสามารถใหม่ ๆ ลงไป และยังต้องทำการทดสอบหาข้อผิดพลาดต่าง ๆ อยู่

สรุปโดยรวมคือ GNU/Linux ได้ชื่อว่าเป็นระบบปฏิบัติการซึ่งพัฒนาในรูปแบบของ Open Source ที่ได้รับความนิยมมากที่สุด คำว่า Open Source นั้นเราหมายถึงการพัฒนาที่เปิดเผยแพร่รหัสต้นฉบับของโปรแกรม (ซึ่งถือว่าเป็นสิ่งมีค่าที่โปรแกรมเมอร์หวงแหนที่สุด) ให้ผู้อื่นสามารถศึกษา หรือพัฒนาต่อยอดและแก้ไขโปรแกรมได้ การที่ Linux เป็น Open Source นี้เอง ทำให้เกิดการพัฒนาค่อยๆเกิดขึ้นมากมาย ทำให้เกิดแนวทางการพัฒนา Linux ออกไปเป็นหลาย ๆ รูปแบบ (หรืออาจเรียกได้ว่าหลายค่าย) ซึ่งเรามักนิยมเรียกว่า Linux Distribution ซึ่งบางค่ายอาจถูกพัฒนาขึ้นโดยนักศึกษาหรืออาสาสมัครที่มีใจรัก แต่ก็ยังมี Linux อีกหลายค่ายที่พัฒนาขึ้นเพื่อใช้ในระบบงานสำคัญองค์กรธุรกิจขนาดใหญ่ ที่ต้องการความมั่นใจและเชื่อถือได้ ในปัจจุบัน Linux ได้รับการยอมรับและการสนับสนุนจากบริษัทขนาดใหญ่ เช่น IBM, Novell และ Sun Microsystems ด้วย

1.3 ประวัติ Fedora Linux



รูปที่ 1.5: Red Hat Enterprise Linux 4 (รูปจาก http://en.wikipedia.org/wiki/Red_Hat_Enterprise_Linux)

Linux Distribution ที่ได้ชื่อว่ามีชื่อเสียงมากที่สุดทั้งในประเทศไทยและทั่วโลก ก็คงจะไม่พ้น Red Hat Linux (รูปที่ 1.5) ซึ่งพัฒนาโดยบริษัท Red Hat Inc. ถ้าจะถามว่าเพราะอะไร ก็คงจะตอบได้ว่า Distribution ของ Red Hat นั้น มีการทำงานที่เชื่อถือได้ในความเสถียร (คือระบบไม่ล่ม หรือเกิด Downtime เองง่าย ๆ) สามารถใช้งานได้ดีทั้งการทำให้เป็น Server ระดับองค์กร หรือจะใช้งานในลักษณะเป็น Desktop ส่วนตัวที่บ้าน มีหน้าจอการติดตั้งที่ผู้ใช้ทั่ว ๆ ไปสามารถทำความเข้าใจได้ง่าย มีเครื่องมือการทำงานแบบ Graphic Interface เหมือนใน Windows และที่สำคัญที่สุดก็คือมีให้เลือกได้ทั้งในรูปแบบที่ให้ Download ได้ฟรีสำหรับผู้ใช้ทั่วไปที่ไม่ต้องการ Support หรือถ้าผู้ซื้อเป็นองค์กรต้องการรูปแบบที่เป็น Package พร้อมการบริการหลังการขายก็มีให้เลือกเช่นกัน

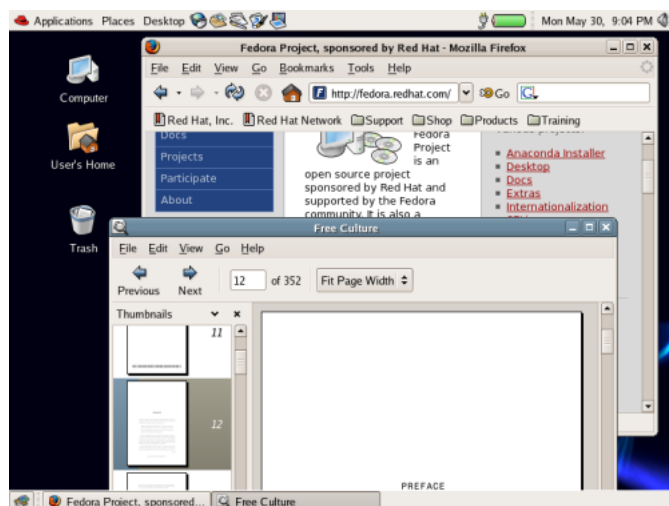
ในช่วงปลายปี 2003 ที่ผ่านมา ผู้ที่ติดตามความเคลื่อนไหวในวงการ Open Source คงได้ทราบข่าวที่ว่า Red Hat จะไม่พัฒนาผลิตภัณฑ์ที่ชื่อ Red Hat Linux อีกต่อไปแล้ว ซึ่งสร้างความประหลาดใจ และวิตกกังวลกันไปทั่วว่าจะเกิดอะไรขึ้นกับ Linux เมื่อ Linux Distribution อันดับ 1 ของโลกหยุดการพัฒนาแล้ว แต่ท่ามกลางความไม่แน่ใจนั้นเอง ก็มีการพูดถึงชื่อของ Linux Distribution ที่เกิดขึ้นมาใหม่อีก 2 ชื่อ ซึ่งชื่อแรกนั้น เป็นชื่อที่ทุกคนรู้จักดี แต่มาในรูปแบบใหม่คือ Red Hat Enterprise Linux แต่อีกชื่อหนึ่งคือ Fedora Linux เป็นการพัฒนาในรูปแบบที่เรียกว่าเป็น Community Project คือพัฒนาโดยกลุ่มผู้ใช้ Linux ร่วมกันผ่านระบบอินเทอร์เน็ต และมี Red Hat เป็นผู้ที่สนับสนุนอยู่เบื้องหลัง นี่เป็นสิ่งที่แตกต่างจากการพัฒนา Red Hat Linux ในแบบเดิม ๆ ที่แนวทางการพัฒนาจะมาจากทางบริษัท Red Hat Inc. เอง ดังนั้น Fedora Linux จะมีการพัฒนา Package ต่าง ๆ เป็นเวอร์ชันใหม่ได้รวดเร็วกว่า ถูกใจผู้ใช้มากกว่าเดิม และที่สำคัญ ตัว Fedora Linux ก็ยังถูกใช้เป็นฐานในการพัฒนา Red Hat Enterprise Linux ด้วย

สำหรับ Red Hat Enterprise Linux นั้นถูกพัฒนาต่อเนื่องมาจาก Red Hat Advance Server นั้นเอง ซึ่งตอนนี้เป็นเวอร์ชัน 3 แล้ว โดยจะถูกแบ่งออกเป็น 3 รุ่นคือ

1. Advance Server (AS) สำหรับใช้งานเป็น Server ระบบงานขนาดใหญ่ เช่น ระดับ Data center
2. Entry Server (ES) สำหรับ Server ระดับทั่ว ๆ ไปในองค์กร เช่น Mail Server, File/Print Server
3. Workstation (WS) สำหรับเครื่อง Desktop และ Workstation ในองค์กร
4. Desktop - สำหรับเครื่องที่ใช้งานทั่วไปที่ใช้งานเพียงคนเดียว

จะเห็นได้ว่า Red Hat Enterprise Linux จะเน้นตลาดลูกค้าที่เป็นทางด้านองค์กรอย่างชัดเจน หรือถ้าจะสรุปให้เข้าใจง่าย ๆ ก็คือ Fedora เน้นกลุ่มผู้ใช้ทั่วไป และเป็นเหมือน Beta tester ให้กับ Red Hat Enterprise Linux ซึ่งเน้นในระดับองค์กรนั่นเอง

Fedora Core (รูปที่ 1.6) เป็น RPM-based (Red Hat Packages Management - Base) ของ Linux distribution หนึ่งในตลาดของ Linux โดยการพัฒนาของกลุ่มผู้ใช้ และนักพัฒนาอิสระ โดยได้รับการสนับสนุนจาก Red Hat



รูปที่ 1.6: Fedora Core 4 ทำงานบน GNOME desktop environment (รูปจาก <http://en.wikipedia.org/wiki/GNOME>)

ประวัติ Fedora Linux แต่ละ Core มีดังนี้

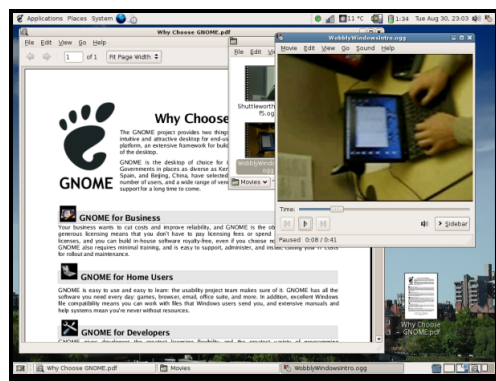
- **Fedora Core 1 (FC1)**, ชื่อรหัสพัฒนา Cambridge, ชื่อรหัสพัฒนาอย่างเป็นทางการ Yarrow) โดยพัฒนาสำเร็จและแจกจ่ายในวันที่ 6 พฤศจิกายน 2003 และถ่ายโอนให้ Fedora Legacy ในวันที่ 20 พฤศจิกายน 2004 โดยระบบหลัก ๆ ทำงานบนพื้นฐานของ Red Hat Linux 9 ที่มี Automated Updates ซึ่งสามารถทำงานบน Notebook ที่สนับสนุน ACPI และ cpufreq โดยทำงานได้เร็วขึ้นในช่วงเริ่มทำงาน และสนับสนุน AMD64 ในเดือน มีนาคม 2004
- **Fedora Core 2 (FC2)**, ชื่อรหัสพัฒนา Tettnang) โดยพัฒนาสำเร็จและแจกจ่ายในวันที่ 18 พฤษภาคม 2004 และถ่ายโอนให้กับ Fedora Legacy ในวันที่ 11 เมษายน 2005 โดยมีมาพร้อมกับ Linux Kernel 2.6, GNOME 2.6, KDE 3.2.1, และ SELinux โดยในรุ่นนี้ก็ได้เปลี่ยนแปลงระบบ XFree86 ไปใช้ X.Org Server แทนด้วย แต่ในรุ่นนี้มีปัญหาต่าง ๆ มากมาย ทั้งในการติดตั้ง และการใช้งาน โดยเฉพาะเรื่อง dual-booting ที่ทำงานร่วมกับ Windows XP
- **Fedora Core 3 (FC3)**, ชื่อรหัสพัฒนา Heidelberg), เป็นระบบที่ยังไม่มีความเสถียรภาพมากนัก โดยพัฒนาในส่วนของ i386 และ AMD64 สำเร็จและแจกจ่ายในวันที่ 8 พฤศจิกายน 2004 โดยมีมาพร้อมกับ GNOME 2.8 และ KDE 3.3.0, X.Org Server 6.8.1, Xen virtualizer, และ Linux kernel รุ่น 2.6.9
- **Fedora Core 4 (FC4)**, ชื่อรหัสพัฒนา Stentz) เป็นระบบที่มีความเสถียรภาพที่สุดในปัจจุบัน (25 ธันวาคม 2005) โดยพัฒนาในส่วนของ i386, AMD64 และ PowerPC สำเร็จและแจกจ่ายในวันที่ 13 มิถุนายน 2005 มาพร้อมกับ GNOME 2.10 และ KDE 3.4, GCC 4.0, gcj-compiled รุ่นสำหรับ Eclipse IDE และ Linux kernel รุ่น 2.6.11.
- **Fedora Core 5 Test 1** เป็นรุ่นทดสอบ ซึ่งยังไม่เสถียรภาพ และยังไม่ปล่อยให้ใช้งานทั่วไป (เหมาะสำหรับนักพัฒนา หรือผู้ชำนาญการเท่านั้น) แต่จากข้อมูลในเว็บไซต์คาดว่าพัฒนาสำเร็จและแจกจ่ายได้ในวันที่ 27 กุมภาพันธ์ 2006 โดยปฏิทินการพัฒนาของ Fedora Core 5 มีดังนี้
 - 21 พฤศจิกายน 2005 - Fedora Core 5 Test 1
 - 23 ธันวาคม 2005 - Fedora Core 5 Test 2
 - 23 มกราคม 2006 - Fedora Core 5 Test 3
 - 27 กุมภาพันธ์ 2006 - Fedora Core 5 Final (พัฒนาสำเร็จและแจกจ่าย)

บทที่ 2

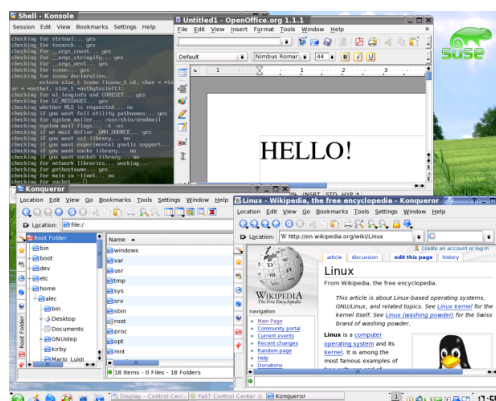
การประยุกต์ใช้งาน OS

Fedora มี Packages มากกว่า 5,000 packages โดยอาศัยการทำงานผ่าน GUI installation ตัวอย่าง Packages ต่างๆ ได้แก่

- ระบบ File Manager ใช้ระบบของ Nautilus (คล้าย ๆ Windows Explorer) และ Default Window Manager เป็น Metacity
- ระบบ Desktop Environment จะใช้ GNOME เป็นมาตรฐาน (รูปที่ 2.1)



รูปที่ 2.1: GNOME desktop environment (รูปจาก <http://en.wikipedia.org/wiki/GNOME>)



รูปที่ 2.2: KDE desktop environment (รูปจาก <http://en.wikipedia.org/wiki/KDE>)

- ระบบ Themes จะเป็น ClearLooks Theme จาก GNOME (รูปที่ 2.1) และ Bluecurve Theme จาก KDE (รูปที่ 2.2) เป็นต้น
- ระบบซอฟต์แวร์ทำงานด้านสำนักงาน ได้ติดตั้ง OpenOffice.org , K office , GNOME Office แล้วแต่จะเลือกติดตั้ง หรือติดตั้งทั้งหมดก็ได้เช่นกัน แต่ที่นิยมใช้จะเป็น OpenOffice.org ซึ่งในปัจจุบันอยู่ใน Version 2.0 ที่มี

ความสามารถใกล้เคียง Microsoft Office System มากที่สุด โดยมีซอฟต์แวร์ตัวอื่น ๆ ที่มีความสามารถเทียบเท่ากับในระบบ Microsoft Windows ที่ทำให้เราสามารถทำงานได้อย่างราบรื่นอย่างมาก โดยดูได้จากตารางเปรียบเทียบในหัวข้อถัดไป

- **ระบบ Server ต่าง ๆ (Web Server, Mail Server, DHCP Server, DNS Server, Internet Sharing, Proxy Server, File sharing Server ฯลฯ)** โดยสามารถทำงานได้ดีกับเครื่องแม่ข่าย ที่มี Lan Card มากกว่า 1 ใบ ท่านสามารถตั้งให้ Lan Card ใบแรกติดต่อกับ Internet และ Lan Card อีกใบต่อกับระบบ Network ภายใน เพื่อให้ Fedora Linux ทำหน้าที่แจก IP Address (DHCP Server), Share Internet, เป็น Proxy Server และ DNS Cache Server เพื่อให้การใช้งาน Internet รวดเร็วยิ่งขึ้น, เป็น Firewall เพื่อรักษา ความปลอดภัย, รวมถึงทำโปรแกรมที่ติดต่อกับ Internet ผ่าน Port ที่แปลกปลอมใด ๆ สำหรับ เครื่องแม่ข่าย ภายในที่มี Lan Card เพียงใบเดียว ท่านสามารถปรับแต่งให้ Fedora Linux แจก IP Address (DHCP Server) และเป็น Proxy Server ได้อย่างง่ายดาย * เครื่องแม่ข่าย ที่มี Lan Card ใบเดียว ไม่สามารถใช้เป็น Firewall พร้อม ๆ กับการ Share Internet ได้ Apache Web Server ที่มีคนใช้มากที่สุดในโลก ที่ทำให้เราสามารถทำระบบ Website ขององค์กรได้อย่างง่ายดาย นอกจากนี้ ยังสามารถเพิ่ม Web page ส่วนตัวสำหรับผู้ใช้งานแต่ละคนของระบบได้ด้วย ผู้ใช้แต่ละท่านสามารถทำ Web page และนำขึ้นมาไว้ที่เครื่องแม่ข่ายได้หลายวิธี เช่น ผ่านระบบ Network Place ของ Windows หรือผ่านssh ของ Unix/Linux หรือผ่าน FTP เป็นต้น Samba Server ทำให้ท่านสามารถนำ File จากเครื่อง Clientที่เป็น Windows มาแลกเปลี่ยนกับ Fedora Linux ได้ง่าย และผู้ใช้แต่ละท่านสามารถนำ File เอกสารมาเก็บ (หรือ backup) และแลกเปลี่ยนกันได้ง่าย Admin สามารถจัดการ Folder ต่าง ๆ ที่ Share ที่เครื่อง Fedora Linux ได้โดยง่ายโดยผ่าน Web Admin Tools นอกจากนี้ Fedora Linux ยังสามารถแลกเปลี่ยน File กับ Operating System (OS) อื่น ๆ ที่ไม่ใช่ Windows ได้หลากหลายช่องการทำงาน ให้เป็น Intranet Mail Server ที่รองรับการทำงานภายในองค์กร และ Internet Mail Server ซึ่งทำหน้าที่เป็น Mail Server ใน Internet อย่างแท้จริง ด้วยการปรับแต่งง่าย ๆ ผ่าน Webmin Tools Mail โดเมนที่ Fedora Linux นี้มาพร้อมกับระบบป้องกันการ Spam และ ไวรัสอยู่ในตัว นอกจากนี้ Fedora Linux ยังรองรับระบบ Visual Host อีกด้วย
- **ระบบ Backup and Restore** ซึ่งท่านสามารถสั่งให้ระบบทำการ Backup ข้อมูลสำคัญต่าง ๆ ลง CD/DVD ซึ่งท่านสามารถนำข้อมูลมา Restore ได้ภายหลัง ทำให้ท่านสามารถย้ายเครื่องแม่ข่าย หรือ Upgrade เครื่อง ได้โดยง่าย
- **งานศึกษาและวิจัย** ประเทศใด ๆ ในโลก ล้วนนิยมใช้ Unix operating system/Linux ในการศึกษาและวิจัยต่าง ๆ สภาพแวดล้อมเหมาะกับการพัฒนา เขียนโปรแกรมประสิทธิภาพสูง และทำงานได้จริง
- **Programming language C/C++, Java, Perl, Python, Ada, Tcl/tk, อื่น ๆ**
- **ระบบ Embedded systems และ Portable device** มีการใช้ Linux เพื่อควบคุมการทำงานของอุปกรณ์ต่าง ๆ มากมาย เช่น พู่กันยนต์, Palm เป็นต้น
- **Graphics Rendering, Realtime, Image Processing and Simulation** Linux ใช้สำหรับงานประเภทที่เวลาตอบสนอง มีความสำคัญสูง, คำนวณ และ ประมวลผล เช่น

Daimler Chrysler ใช้ Linux ในการวิเคราะห์ทางด้าน Image Processing อนาคตจะหยุดเอง ถ้าพบว่ามีความเสี่ยงทางด้านหน้า

บริษัท Digital Domain ซึ่งรับหน้าที่ทำระบบ CGI (Computer Generated Imagery) ให้กับภาพยนตร์ เรื่อง Titanic ใช้ Linux ในการ Render ตัวเรือ Titanic ขนาดยักษ์ผ่านระบบ Graphics Rendering บน Linux

บริษัท Square ใช้ Render Farm ประมาณ 900+ CPU เป็น Linux Cluster และใช้ Software ชื่อว่า Pixar RenderMan for Linux สำหรับ render ภาพ
- อื่น ๆ อีกมากมาย

บทที่ 3

ความต้องการของ Hardware

3.1 CPU

CPU 32-bit x86 systems

- ขั้นต่ำคือ Pentium โดย Fedora Core 4 นี้จะ optimized ตัวรหัสสำหรับ CPU Pentium 4 แต่ก็ยังคงสนับสนุนทุกๆ CPU อื่นๆ ด้วย (เช่น Pentium, Pentium Pro, Pentium II, Pentium III รวมไปถึง AMD และ VIA)
- โดยสัญญาณนาฬิกาของ CPU ที่ทำงานได้ดีใน text-mode คือ 200 MHz ใน Pentium หรือสูงกว่า
- โดยสัญญาณนาฬิกาของ CPU ที่ทำงานได้ดีใน graphical-mode คือ 400 MHz ใน Pentium II หรือสูงกว่า

CPU 64-bit x86_64 systems

- CPU AMD64 processors (โดยใช้ได้ทั้ง Athlon64 และ Opteron)
- Intel processors ที่มี Intel Extended Memory 64 Technology หรือที่เรียกย่อๆ ว่า Intel EM64T

CPU PowerPC (PPC)

- PowerPC G3 / POWER4
- Fedora Core 4 สนับสนุนในรุ่น New World ของ Apple Power Macintosh ที่เริ่มส่งสินค้าใน 1999 โดยที่สามารถทำงานได้บน 64-bit G5 processor และ POWER processors ใน IBM eServer pSeries แต่ในปัจจุบัน 32-bit IBM RS/6000 นั้นยังไม่สนับสนุน
 - โดยสัญญาณนาฬิกาของ CPU ที่ทำงานได้ดีใน text-mode คือ 233 MHz G3 หรือสูงกว่า
 - โดยสัญญาณนาฬิกาของ CPU ที่ทำงานได้ดีใน graphical-mode คือ 400 MHz G3 หรือสูงกว่า

3.2 Hard Drive

CPU 32-bit x86 systems)

- Custom Installation (ต่ำสุด): 620MB
- Server: 1.1GB
- Personal Desktop: 2.3GB
- Workstation: 3.0GB
- Custom Installation (ลงระบบทุก ๆ อย่าง): 6.9GB

CPU 64-bit x86_64 systems

- Custom Installation (ต่ำสุด): 900MB
- Server: 1.5GB
- Personal Desktop: 2.7GB
- Workstation: 3.4GB
- Custom Installation: (ลงระบบทุก ๆ อย่าง) 7.5GB

CPU PowerPC (PPC)

- เหมือน CPU 64-bit x86_64 system

3.3 Memory

CPU 32-bit x86 systems

- ขั้นต่ำสำหรับ text-mode: 64MB
- ขั้นต่ำสำหรับ graphical-mode: 192MB (แนะนำสำหรับ for graphical-mode: 256MB)

CPU 64-bit x86_64 systems

- ขั้นต่ำสำหรับ text-mode: 128MB
- ขั้นต่ำสำหรับ graphical-mode: 256MB (แนะนำสำหรับ for graphical-mode: 512MB)

CPU PowerPC (PPC)

- เหมือน CPU 64-bit x86_64 systems

3.4 อื่น ๆ

- Keyboard*, Mouse*
- Floppy Drive*, Optical Drive*, Monitor*

* จะมีหรือไม่มีก็ได้ขึ้นอยู่กับการใช้งานในแต่ละงาน และแต่ละช่วงเวลา เช่นในตอนติดตั้งอาจจำเป็นต้องใช้ *Optical Drive* แต่พอทำการติดตั้งเสร็จแล้วก็ไม่ต้องใช้ *Optical Drive* อีกต่อไปก็นำเอา *Optical Drive* ออกไปได้เลย

บทที่ 4

ระบบ RAID

RAID ย่อมาจาก Redundant Array of Independent Disks หรือ Redundant Array of Inexpensive Disks เป็นระบบในการจัดเก็บแบบ Hard Drive หลายๆ ตัว สำหรับในการแลกเปลี่ยน หรือสำรองข้อมูลบนสื่อเก็บข้อมูล โดยทั่วไปแล้ว RAID จะใช้ Hard Drive หลายตัวแล้วรวมมันเป็น Hard Drive เสมือนเพียง 1 ตัว (หรือเรียกอีกอย่างว่า single logical unit)

4.1 ประวัติของ RAID

- ในปี 1978 Norman Ken Ouchi นักวิจัยของ IBM ได้รับสิทธิบัตร U.S. Patent 4,092,732 ในหัวข้อ "System for recovering data stored in failed memory unit" หรือระบบสำหรับการเรียกคืนข้อมูลจากการที่หน่วยความจำล้มเหลว
- ในปี 1988 RAID level 1 - 5 ได้คิดค้นขึ้นโดย David A. Patterson, Garth A. Gibson และ Randy H. Katz ในผลงานวิจัยชื่อ "A Case for Redundant Arrays of Inexpensive Disks (RAID)" โดยตีพิมพ์ในวารสารงานวิจัย SIGMOD Conference 1988 หน้า 109 - 116

4.2 ประโยชน์ของ RAID

- ทำให้ อัตราการโอนถ่ายข้อมูล (Data Transfer Rates) สูงกว่าอัตราการโอนถ่ายข้อมูลฮาร์ดดิสต์เพียงตัวเดียว
- ทำให้การเพิ่มความจุของระบบจัดเก็บข้อมูลนั้นไม่จำกัดในลักษณะ single logical unit ในขณะที่การเพิ่มฮาร์ดดิสต์ในแบบปกติ แม้จะเป็นการเพิ่มความจุ แต่ก็เป็นการเพิ่มจำนวนไดรฟ์ไปในตัว (C,D,E เป็นต้น) เพราะ ไขว่ของเครื่องออกแบบไว้เช่นนั้น ทำให้การจัดเก็บอาจ จะต้องแยกกันเก็บและ ทำให้ลักษณะการจัดกระจายซึ่งยุ่งยากในการบริหารจัดการ
- สร้างความปลอดภัยให้ข้อมูลด้วยระบบ Data redundancy/ Fault tolerance คุณจึงอุ่นใจได้ว่าข้อมูลจะไม่มีวันสูญหาย ถ้าหากมีฮาร์ดดิสต์ ตัวใดตัวหนึ่งในอะเรย์ เสียอย่างกะทันหัน นอกจากนั้นแล้วคุณก็ยังสามารถแก้ไขฮาร์ดดิสต์ตัวที่เสียโดย การถอดเปลี่ยนแทนที่ด้วยฮาร์ดดิสต์ตัวใหม่โดยที่ไม่ต้อง Down ระบบซึ่งต่างจากการต่อฮาร์ดดิสต์ในแบบทั่วไปหาก ฮาร์ดดิสต์เสีย ขึ้นมาก็เท่ากับข้อมูลเกิดสูญหายในทันที

4.3 จะเรียก Inexpensive หรือ independent ?

โดยความหมายในตัว I ใน RAID แล้วตรงกับ ความหมาย independent มากกว่า inexpensive โดยจากการที่ RAID นั้นสามารถใช้งานได้หลากหลายรูปแบบ และไม่ยึดติดกับระบบใดระบบหนึ่งเช่น SCSI หรือ IDE/ATA หรือแม้แต่ Flash memory ก็มีความคิดที่จำนำมาทำ RAID ด้วยเช่นกัน ทำให้น่าจะใช้คำว่า Independent ที่แปลว่าอิสระ มากกว่า Inexpensive ที่แปลว่าราคาถูก หรือประหยัด เพราะในความเป็นจริง ก็มีราคาในการจัดการระบบ RAID ที่แพงกว่าธรรมดาอยู่แล้ว

4.4 ชนิดของ RAID ที่สนับสนุน

รูปแบบลักษณะของ RAID แบ่งออกเป็น Level ซึ่งแตกต่างกันออกไป แต่ก็มีแบ่งประเภทของ Level อีกด้วยเป็น 3 ประเภท คือ Standard RAID levels, Nested RAID Levels, Proprietary RAID levels ซึ่งในรายงานนี้เราจะขอพูดในส่วนของ ระบบ Standard RAID levels เท่านั้น เพราะใช้กันมากที่สุด

Standard RAID levels มี Level ดังนี้

- **RAID Level 0** หรือเรียกอีกอย่างว่า Striped Disk Array without Fault Tolerance เป็นการใช้เทคนิค performance-oriented data mapping หรืออีกอย่างหนึ่งคือ การทำงานกระจายข้อมูลที่เรียกว่า Data Block และนำ Data Block กระจายไปเก็บไว้ในฮาร์ดดิสก์ที่ทำงานร่วมกันใน RAID ในแต่ละตัว ทำให้การทำงานในการอ่าน และเขียนข้อมูลรวดเร็วมากขึ้น, การทำงานไม่ต้องมีการคำนวณ parity เพื่อตรวจสอบการทำงานที่ผิดพลาด, การออกแบบเป็นไปอย่างเรียบง่าย และติดตั้งง่าย แต่ RAID 0 ยังไม่ถือว่าเป็น RAID ที่แท้จริง เพราะไม่มีส่วนที่รับประกันความปลอดภัยของข้อมูล (data redundancy) ถ้า Hard Drive ตัวใดตัวหนึ่งเสียหาย ข้อมูลทั้งหมดก็เสียหายไปด้วย จำแนก Hard Drive ขั้นต่ำคือ 2 Drive
- **RAID Level 1** หรือเรียกอีกอย่างว่า Disk Mirroring เป็นการใช้เทคนิค provides redundancy by writing identical หรืออีกนัยหนึ่งคือ การสร้างสำเนาข้อมูลโดยตรง แล้วนำไปเก็บไว้ในฮาร์ดดิสก์ที่ทำงานร่วมกันใน RAID ในแต่ละตัว ทำให้การทำงานในการอ่านข้อมูลรวดเร็วมากขึ้น แต่ไม่เท่ากับ RAID Level 0 และการทำงานไม่ต้องมีการคำนวณ parity เพื่อตรวจสอบการทำงานที่ผิดพลาด, การออกแบบเป็นไปอย่างเรียบง่าย และติดตั้งง่าย และที่ดีกว่า RAID Level 0 คือการฟื้นฟูสภาพจากการที่ Drive ล้มเหลวนั้นง่ายมาก เนื่องจากข้อมูลจะยังคงเก็บอยู่ใน Drive อีกตัวหนึ่งเสมอ จำแนก Hard Drive ขั้นต่ำคือ 2 Drive
- **RAID Level 4** หรือเรียกอีกอย่างว่า Independent Data Disks with Shared Parity Disk เป็นการใช้เทคนิค striping และ parity bit เข้ามาจัดการ จึงต้องใช้จำนวน Hard Drive ขั้นต่ำคือ 3 Drive โดย 2 Drive แรกติดตั้งแบบ RAID Level 0 และ อีก Drive จะเก็บค่า parity bit เอาไว้ซึ่งจะอธิบายความในอีกนัยหนึ่งว่า ก่อนจะบันทึกข้อมูลใหม่ลงไป ข้อมูลเก่าที่ strip ไว้จะต้องถูกอ่านขึ้นมาก่อน รวมถึง strip ของ parity ที่เป็น bit จากนั้นจึงบันทึกข้อมูลใหม่ลงไปพร้อมกับข้อมูล parity ที่ได้รับการคำนวณใหม่ ดังนั้นการบันทึกแต่ละ strip จะเกิดการอ่าน 2 ครั้ง และบันทึกข้อมูล 2 ครั้ง ส่วนการอ่านข้อมูลจะมีลักษณะการทำงานเหมือนกับ RAID Level 0 เพียงแต่จะมีการตรวจสอบความถูกต้องของข้อมูลจากค่าของ parity bit เสมอ ทำให้เกิดปัญหาจุดคอขวดของการทำงานขึ้นมา เพราะข้อมูลต่างๆ จะประดังเข้ามาใน Drive ที่เก็บ parity ตลอดเวลา
- **RAID Level 5** หรือเรียกอีกอย่างว่า Independent Data Disk with Distributed Parity Blocks เป็นการใช้เทคนิค striping และ parity bit เหมือนกับ RAID Level 4 เพียงแต่จะแตกต่างตรงที่แทนที่จะมีการใช้ 2 Drive แรกติดตั้งแบบ RAID Level 0 และ อีก Drive จะเก็บค่า parity bit เปลี่ยนมา ทุกๆ Drive จะมีทั้งข้อมูล strip และ parity bit โดยจะกระจายไปทุกๆ Drive ตัวอย่างเช่น ใช้ RAID Level 5 กับจำนวน Drive 3 ตัว โดยข้อมูลจะ stripping และส่งให้กับ Drive หมายเลข 1 และ 2 ส่วนหมายเลข 3 จะเก็บ parity bit และข้อมูลตัวต่อมาเข้ามา ข้อมูลก็จะ stripping และส่งให้กับ Drive หมายเลข 2 และ 3 ส่วนหมายเลข 1 จะเก็บ parity bit ไว้ ทำวนเวียนต่อไปแบบนี้เรื่อยๆ
- **Linear RAID** เป็นการรวมกลุ่มของการสร้าง Drive เสมือนขนาดใหญ่ โดยที่ linear RAID เป็นการทำ allocated sequentially drive (disk) space หรือการทำให้ drive ทำการเก็บข้อมูลต่อไปได้เรื่อยๆ ไม่มีที่สิ้นสุด โดยที่ เมื่อ drive ตัวแรกเก็บข้อมูลจนเต็มแล้ว ระบบจะทำการเก็บข้อมูลมาที่ drive ตัวที่สองที่ได้เตรียมไว้ และเมื่อ drive ตัวที่สองเต็มก็ทำการเก็บข้อมูลต่อมายัง drive ตัวที่สาม แต่ถ้าสมาชิกในกลุ่มของระบบ linear RAID ข้อมูลในส่วน ของ Drive ที่เสียไป ก็จะใช้งานไม่ได้

บทที่ 5

ระบบ File System

ระบบ File System (FS) ของ Fedora Linux Core 4 นั้นโดยมาตรฐานแล้วคือ Extended file system หรือ ext เป็น file system ตัวแรก ที่สร้างขึ้นมาเพื่อใช้สำหรับระบบปฏิบัติการ Linux โดยได้ทำการออกแบบเพิ่มเติมจากความสามารถที่จำกัดจาก Minix File system และจาก xiafs โดยมีอยู่ 2 รุ่น คือ ext2 และ ext3 แต่ก็ยังมี File System อื่นๆหลายตัวที่ Fedora Linux สามารถทำงานได้ เช่น NILFS, ReiserFS และ Reiser4

5.1 ext2 หรือ Second extended file system

ext2 เป็น File System ที่ทำงานบน Linux kernel ทุกตัว ซึ่งออกแบบโดย Remy Card ในปี 1993 ซึ่งทำงานได้เร็วและมีประสิทธิภาพมาก แต่ยังไม่สนับสนุน การทำงานแบบ Journaling File System * และระบบ Journaling File System นี้มีใน ext3

ext2 เป็น file system พื้นฐานใน Red Hat Linux, Fedora Linux Core และ Debian Linux

* Journaling File System (FS) คือระบบไฟล์อย่างหนึ่งที่คอยบันทึก metadata อยู่ตลอดเวลาเพื่อป้องกันปัญหาข้อมูลผิดพลาดและข้อมูลพัง ซึ่ง Journaling FS นี้ก็มีอยู่หลายชนิดด้วยกัน เช่น NTFS ของ Windows, XFS ของ IRIX, EXT3, Reiser4, ReiserFS ของ Linux, และ HFS+ ของ Mac OS X เป็นต้น การใช้ระบบ Journaling FS มีข้อดีก็คือ จะช่วยป้องกันปัญหา เวลาที่เกิด system failure เช่นเครื่อง crash หรือว่าต้อง force reset (โดยการกดปุ่ม power .. ไม่ผ่าน shutdown process) ซึ่งก็ไม่ได้ช่วยให้ปัญหาหมดไป 100% แต่ว่าก็ดีกว่าไม่มี ส่วนประสิทธิภาพ ก็ถือว่าแย่กว่า Non-Journaling นิดหน่อยเท่านั้น (อย่า นิดหน่อยเท่านั้น .. ถ้าไม่ใช่ benchmark tool เพื่อทดสอบ ไม่เห็นความแตกต่างแต่อย่างใด), รวบรวม รวบรวม

5.2 ext3 หรือ Third extended file system

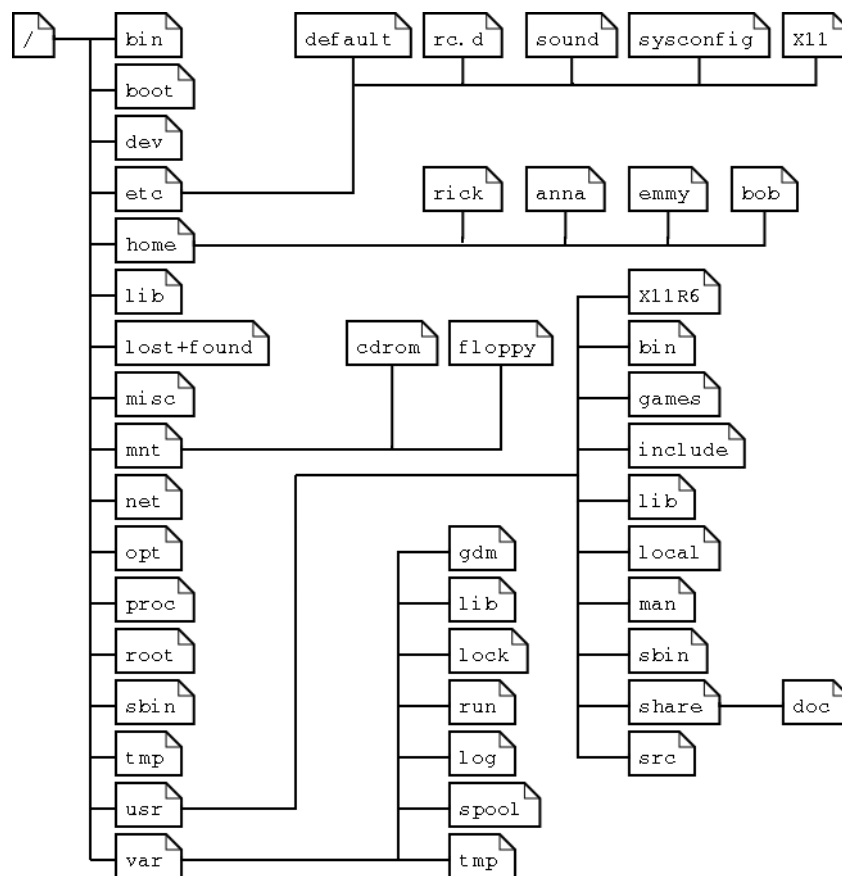
มีคุณสมบัติพื้นฐานต่าง ๆ แบบเดียวกับ ext2 แต่ใน ext3 นั้นมีการเพิ่มความสามารถขึ้นมาอีก 3 อย่างคือ

1. Journaling File System
2. H-tree (hashed tree) directory indexes
3. Online filesystem resizing

โดย ext3 ยังเป็น file system พื้นฐานใน Red Hat Linux, Fedora Linux Core, Ubuntu Linux และ Debian Linux

โดยสรุปได้จากข้อมูลด้านล่างนี้ทั้ง ext2 และ ext3

- ผู้พัฒนา : Stephen Tweedie (ออกแบบ ext3 และ เขียนระบบ), Remy Card (ออกแบบ ext2, ร่วมพัฒนา ext3 และ เขียนระบบ), Theodore Ts'o (ระบบเครื่องมือ และ แก้ไขปรับปรุง), Andreas Gruenbacher (xattrs และ ACLs), Andreas Dilger (ออกแบบระบบ online resizing), และที่ไม่ได้กล่าวอีกมาก
- ชื่อเต็ม : Second extended file system (ext2) และ Third extended file system (ext3)
- เริ่มนำมาใช้ : เดือนเมษายน 1992 (Linux kernel 0.96c) สำหรับ ext2 และ เดือนพฤศจิกายน 2001 (ใน Linux kernel 2.4.15) สำหรับ ext3
- โครงสร้าง Directory : inode table (ext2) และ H-tree (ext3)
- File allocation : bitmap (free space) และ table (metadata)
- Bad blocks : Table
- ขนาดไฟล์ที่ใหญ่ที่สุดที่รองรับ : 2TB
- จำนวนไฟล์สูงสุดที่รองรับ : เกือบจะไม่จำกัด
- ขนาดของปริมาณพื้นที่สำหรับเก็บข้อมูลสูงสุด : 32TB
- ตั้งชื่อที่มีความยาวสูงสุด : 255 ตัวอักษร
- ระบบสิทธิเข้าถึงไฟล์ : Unix permissions, ACLs และ arbitrary security attributes (Linux kernel 2.6 ขึ้นไป)



รูปที่ 5.1: Ext File System Layout

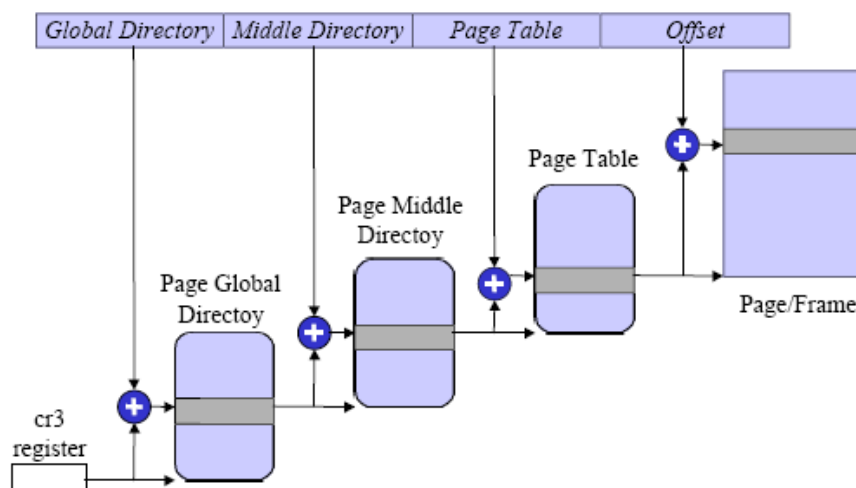
บทที่ 6

ระบบ Memory Management

ส่วนของ kernel ของลินุกซ์ค่อนข้างซับซ้อน แต่ในจะแนะนำต่อไปนี่ คือ ส่วนของ memory management ซึ่งจะเป็นในส่วนที่เราต้องทำความเข้าใจ

View of a logical address in Linux

(For x86 processors, *Middle Directory* is 0 bits)



รูปที่ 6.1: Logical address in Linux

- ใช้ระบบ Paging
- ระบบการทำงานร่วมกันของ หน่วยความจำทางกายภาพ (Physical memory) และหน่วยความจำทางตรรกะ (Logical memory) ผ่าน page table
- การจัดการเฟรม (frame management)
- กระบวนการอ้างอิงตำแหน่ง (Process address space) โดยใช้
 - Memory descriptors
 - Memory regions
 - Page faults
- ระบบที่ใช้ร่วมกับ Intel x86 ใช้ segmentation *
- Linux จะใช้การทำงานแบบ segmentation
- Linux ใช้ paging ด้วยการ

- แบ่ง page ขนาด 4KB สำหรับ x86 CPU และ 8KB สำหรับ Alpha CPU
- ใช้ page table 3 ระดับ โดยใช้ 64-bit addresses space
- ในระบบ x86 processors
 - * ใช้ page table 2 ระดับ โดยใช้ 32 bit addresses space
 - * ใช้ Segment + Offset จะได้ 4 GB Linear address (หรือ 32 bits) โดยแบ่งเป็น user space ที่ 3 GB และ kernel space ที่ 1GB
 - * paging ต้องสนับสนุนโดย hardware
 - * จัดการโดย TLB (Translation Lookaside Buffer) **

* Segmentation (การแบ่งหน้าจะแบ่งให้มีขนาดเท่ากัน) แต่การแบ่งเป็นเซกเมนต์จะแบ่งโปรแกรมออกเป็น ส่วน ๆ ไม่เท่ากัน และมีการใช้ตำแหน่งทางตรรกะ อ้างอิงตำแหน่งจริงเช่นกัน และมีการใช้ข้อมูล 2 ส่วนคือ เลขที่เซกเมนต์ และระยะเริ่มต้นของเซกเมนต์(Offset) สำหรับผลของการแบ่งเซกเมนต์ทำให้เกิดชิ้นส่วนไม่เท่ากัน (Dynamic partitioning) ซึ่งลดปัญหาการสูญเสียพื้นที่ (Internal fragmentation)

** TLB : Translation lookaside buffer (บัฟเฟอร์ค้นหาที่อยู่) ทุกครั้งที่เรียกใช้หน่วยความจำเสมือน ย่อมเรียกใช้หน่วยความจำหลัก 2 ครั้ง คือ การอ่านตารางหน้า และอ่านข้อมูลจริงจากหน่วยความจำหลัก ดังนั้นการทำงานแบบนี้จึงใช้เวลาถึง 2 เท่า จึงมีการใช้ cach memory ซึ่งทำหน้าที่เก็บตารางหน้าที่เคยถูกเรียกใช้ หรือเรียกว่า บัฟเฟอร์ค้นหาที่อยู่ (TLB : Translation Lookaside Buffer) ถ้าพบใน TLB จะเรียก TLB hit ถ้าไม่พบเรียก TLB miss และเข้าไปอ่านตารางหน้า เมื่ออ่านแล้วก็นำมาเพิ่มใน TLB สำหรับโอกาสที่จะถูกเรียกในอนาคต

ภาคผนวก ก

เอกสารอ้างอิง

- ระบบปฏิบัติการยูนิกซ์ (NECTEC)
<http://www.nectec.or.th/courseware/program/Unix/0001.html>
- Unix (วิกิพีเดีย สารานุกรมเสรี)
<http://en.wikipedia.org/wiki/Unix>
- แนะนำลินุกซ์เบื้องต้น และประวัติความเป็นมา (Kaiwal Development Team)
http://xn--12cu4b7b4dza6l.th/pub/linux.sis/3.1/docs/LTP/02_2intro.html
- ลินุกซ์ (วิกิพีเดีย สารานุกรมเสรี)
<http://th.wikipedia.org/wiki/ลินุกซ์>
- Unix กับ Linux ต่างกันอย่างไร? (วีร์ สัตยมาศ)
<http://veer.exteen.com/20050804/Unix-linux>
- Fedora Core (วิกิพีเดีย สารานุกรมเสรี)
http://en.wikipedia.org/wiki/Fedora_Core
- Fedora Project, sponsored by Red Hat (Red Hat, Inc.)
<http://fedora.redhat.com/>
- Red Hat Enterprise Linux (วิกิพีเดีย สารานุกรมเสรี)
http://en.wikipedia.org/wiki/Red_Hat_Enterprise_Linux
- Journaling filesystem (วิกิพีเดีย สารานุกรมเสรี)
http://en.wikipedia.org/wiki/Journaling_filesystem
- General overview of the Linux file system (Machtelt Garrels)
http://www.faqs.org/docs/linux_intro/sect_03_01.html
- The Official Red Hat Linux Reference Guide RAID (Red Hat, Inc.)
<http://www.redhat.com/docs/manuals/linux/RHL-6.2-Manual/ref-guide/ch-raid.html>
- Comparison of file systems (วิกิพีเดีย สารานุกรมเสรี)
http://en.wikipedia.org/wiki/Comparison_of_file_systems
- Linux Memory Management (Craig M. Grube)
<http://www.cs.purdue.edu/homes/li/cs690Z/Outline/vmm.pdf>
- Lecture Overview Linux Memory Management (Dr. Richard Hall, Torsten Fink)
<http://www.inf.fu-berlin.de/lehre/SS01/OS/Lectures/Lecture14.pdf>
- แนะนำเพิ่มเติมเนื้อหา Journaling filesystem, ประวัติ และการจัดรูปแบบ (ithilien_rp (รวิทัต ภู่อล่ำ))
<http://www.cs.su.ac.th/>