

Estudi comparatiu de sistemes operatius basats en GNU/Linux per a dispositius mòbils

Treball final de carrera
Chin Man Yeung Tai
Primavera 2011

Tutor: Miquel Angel Senar Rosell



ÍNDEX DE CONTINGUTS

1. Introducció	4
1.1. Objectius.....	6
1.2. Motivació.....	7
1.3. Planificació inicial	8
2. Conceptes previs.....	9
2.1. Software lliure	9
2.2. Software Open Source.....	10
2.3. Tipus de llicències Open Source	11
2.4. GNU/Linux	12
2.5. Dispositiu mòbil.....	13
2.6. Benchmark.....	14
3. Estat de l'art dels sistemes operatius mòbils.....	14
3.1. Sistemes tancats.....	17
3.1.1. Bada.....	17
3.1.2. Blackberry	17
3.1.3. iOS	18
3.1.4. Symbian OS	18
3.1.5. webOS	19
3.1.6. Windows Phone	19
3.2. Sistemes lliures.....	20
3.2.5. Access Linux Platform	20
3.2.1. Android.....	20
3.2.2. Limo.....	21
3.2.3. Maemo	21
3.2.4. Meego	22
3.2.5. Moblin	22
3.2.6. Ophone.....	22
4. Distribucions seleccionats	23
4.1. Criteri de selecció	23
5. Anàlisis	23
6. Proves	36
6.1. Descripció	37
6.2. Resultats	38
7. Comparativa	42

8. Conclusions	43
8.1. Treball assolit	43
8.2. Treballs futurs.....	43
9. Referències.....	44
10. Annex	45

LLISTAT DE TAULES

1. Forecast: Mobile Communications Device Open OS Sales to End Users by OS	16
2. Taula de resultats de Ubuntu	37
3. Taula de resultats de Android.....	38
4. Taula de resultats de Meego	39
5. Taula de resultats de OPhone	40
6. Taula de resultats de webOS	41
7. Taula resum dels sistemes comparats	42

LLISTAT DE FIGURES

1. Il·lustració de les mascotes oficials de GNU i Linux	4
2. Planificació inicial	8
3. Diagrama de Gantt.....	8
4. Logotipus de la Free Software Foundation	11
5. Logotipus del Open Source	11
6. Smartphone.....	14
7. Mobile Operating System Market Share	16
8. Top 8 Mobile OS from Dec 08 to May 11	16
9. Logotipus Bada	18
10. Logotipus Blackberry.....	18
11. Logotipus iOS.....	19
12. Logotipus Symbian OS.....	19
13. Logotipus webOS	20
14. Logotipus Windows Phone	20
15. Logotipus Access Linux PLatform	21
16. Logotipus Android.....	21
17. Logotipus LiMo Foundation	22
18. Logotipus Maemo	22

19. Logotipus MeeGo.....	23
20. Logotipus Moblin	23
21. Logotipus OPhone	23
22. Captura de pantalla emulador Android	25
23. Diagrama de l'arquitectura del sistema Android	27
24. Captura de pantalla emulador MeeGo	28
25. Diagrama de l'arquitectura del sistema MeeGo.....	30
26. Captura de pantalla emulador OPhone	31
27. Diagrama de l'arquitectura del sistema OPhone.....	33
28. Captura de pantalla emulador webOS	34
29. Diagrama de l'arquitectura del sistema webOS	36
30. Gràfica de les mitjanes dels resultats de Ubuntu	39
31. Gràfica de les mitjanes dels resultats de Android	40
32. Gràfica de les mitjanes dels resultats de MeeGo	41
33. Gràfica de les mitjanes dels resultats de OPhone	42
34. Gràfica de les mitjanes dels resultats de webOS.....	43
35. Diagrama comparativa entre els sistemes	44

1. INTRODUCCIÓ

El sistema operatiu és el programari essencial destinat a la gestió dels recursos disponibles del maquinari per tal de concebre un sistema informàtic que administri i proveeixi serveis a les aplicacions. En els principis dels anys 80, Richard Stallman va iniciar el projecte anomenat GNU que té per finalitat la creació d'un sistema operatiu que segueix el model del sistema UNIX però sota el paradigma del programari lliure. Aquest projecte no va concloure degut a la manca d'un nucli, fins que Linus Torvalds va donar l'impuls definitiu a l'any 1991 amb la implementació del kernel Linux compatible amb el conjunt d'eines de GNU ja desenvolupat.



Fig 1. Il·lustració de les mascotes oficials de GNU i Linux

Fins ara, els sistemes informàtics han gaudit de innumerables millores tecnològiques, obligant als seus components a una renovació i una actualització quasi constant. Segons la llei de Moore, i altres proposicions d'experts en ciències computacionals com en Niklaus Wirth, el maquinari avança de forma desproporcionada respecte al programari. Les innovacions en hardware són constants mentre que el software a més de tornar-se cada cop més complex i difícil de mantenir, ha d'adaptar-se amb la màxima celeritat a les noves característiques del maquinari. I aquest fet és totalment vàlid pels sistemes operatius.

Aquest projecte tracta concretament el cas dels sistemes operatius GNU/Linux desenvolupats per plataformes mòbils. En el camp dels dispositius mòbils, que poden ser aparells de tot tipus, destaca un per sobre de tots i és sense dubtes el telèfon mòbil. Principalment, degut a la demanda popular de la comunicació per veu sense fils i en qualsevol indret, el mercat de la telefonia mòbil va trobar el seu lloc en l'economia global, tant és així que a partir dels anys 1990 va començar un procés massiu de comercialització i de desenvolupament.

Per una banda, les grans companyes van iniciar la producció d'aquests aparells i van llençar contínuament al mercat models cada cop més sofisticats, englobant cada cop més funcionalitats. Allò que era en principi un terminal que possibilita la comunicació per veu, va adquirir nous components, propis d'altres dispositius electrònics com ara els reproductors de música o les càmeres digitals de foto. La connectivitat d'aquestes plataformes va augmentar de forma considerable amb la introducció progressiva de noves tecnologies com ara la connexió per infrarojos, Bluetooth, GPRS, UMTS, incrementant d'aquesta forma la interacció amb els ordinadors i la xarxa directament. En lo que respecta la usabilitat, la millora més important es va produir amb l'arribada de la pantalla tàctil que va substituir parcial o completament el tradicional teclat físic.

Per altra banda, el programari sempre s'ha adaptat per donar suport a les innovacions de hardware. Al començament, la introducció sistemàtica de noves funcions i aplicacions va obligar als mòbils a incloure sistemes operatius relativament senzills que permetien un sistema de fitxers compatible amb els formats més populars dels sistemes operatius dominants. Paulatinament els serveis que s'oferien sobretot als ordinadors domèstics, s'extrapolaven cap als telèfons mòbils. A partir de l'any 2003, la revolució de les xarxes socials i altres eines de comunicació avançades via web, van conformar un gran incentiu per aquests aparells.

Així doncs, de la continua evolució del maquinari (autonomia, miniaturització, LEDs, pantalla tàctil, GPS...) i programari (aplicacions, navegació web, interfície gràfica d'usuari...) d'aquests telèfons i per factors tecnològics i socials com la Web 2.0 neixen els denominats "Smartphones" o telèfons intel·ligents que es podria considerar una de les màximes expressions de les tecnologies actuals aplicades al consum. Per tal d'enllaçar aquest conjunt de noves tecnologies, és imprescindible el suport de sistemes operatius específicament orientats a aquests dispositius per poder aprofitar els recursos i les funcionalitats que poden arribar a oferir.

1.1. OBJECTIUS

L'objectiu principal d'aquest projecte és aconseguir una comparativa rellevant dels sistemes operatius basats en GNU/Linux per a dispositius mòbils, a partir de les dades extretes d'una sèrie de proves que es realitzarà en cada distribució. Aquesta comparativa es centrarà en els aspectes tècnics de cada sistema, es a dir els mòduls i els components de la seva arquitectura, la integració del kernel de Linux, i les repercussions que tenen a l'hora de mesurar el rendiment i la eficiència.

Les proves s'intentaran aplicar per mesurar aspectes com:

- Ús de la memòria
- Sobrecarrega
- Nombre d'instruccions
- Temps d'escriptura de fitxer
- Temps de arrancada
- Temps d'aturada

A diferència de les distribucions dirigides pels ordinadors de sobretaula, aquests sistemes operatius que es volen analitzar són específics per a dispositius mòbils, suposant la dificultat de no disposar d'un suport genèric per executar les proves. S'emularan les imatges proveïts per la entitat responsable de cada sistema. Per això, és necessari establir un procediment per aclarir com es realitzaran les comprovacions i quines mesures s'adopten per mantenir la validesa d'aquestes.

El projecte també introduirà i explicarà els conceptes necessaris per entendre la selecció dels sistemes operatius. És necessari comprendre l'ideal del software lliure promogut per Richard Stallman, així com les diferents llicències compatibles amb la llicència pública general de GNU que busca garantir les llibertats tant dels programadors com dels usuaris. Igualment es farà una aproximació de la història i de la estructura del GNU/Linux per discernir del llegat d'aquesta plataforma en els nous sistemes operatius mòbils.

Adicionalment, es proporcionarà informació complementària sobre els sistemes disponibles actualment, per exemple, sota quina llicència es distribueixen, quines són les empreses o fundacions al càrrec del desenvolupament i del manteniment, o quines xifres d'ocupació de mercat representa cada distribució.

1.2. MOTIVACIÓ

El projecte està motivat pel conjunt de conceptes implicats entorn a la plataforma GNU/Linux. La idea sorgeix en base a la importància que suposa els nous dispositius mòbils en el món de les tecnologies de la informació i de la comunicació. No obstant, aquesta innovació no només afecta als mercats i al desenvolupament sinó també que probablement tindrà profundes conseqüències socials. Amb tota probabilitat, els dispositius mòbils com els “smartphones” constituïran les noves eines de comunicació entre persones, una eina que fusiona tots els mitjans i canals existents. Per això, penso que els sistemes operatius que els fan funcionar haurien d'estar construïts sobre un transfons el més ètic possible, subjecte a llicències que garanteixi les llibertats dels usuaris.

El desafiament que suposa GNU/Linux per les empreses que es basen en llicències tancades per lucrar-se és un altre font de motivació. Els dispositius mòbils són un producte relativament recent que encara està penetrant en el mercat però ja compta amb un gran suport de la comunitat del software lliure. La propagació i l'èxit que pot arribar a tenir en aquest segment pot constituir la prova de que el model de Stallman és viable.

1.3. PLANIFICACIÓ INICIAL

La planificació inicial de les tasques per assolir el projecte segueix la disposició següent:

	Nombre de tarea	Duració	Comienzo	Fin
1	Comparativa Sistemes Operatius	59 días	mar 29/03/11	vie 17/06/11
2	Recopilació d'informació	12 días	mar 29/03/11	mié 13/04/11
3	Recerca informació GNU/Linux	3 días	mar 29/03/11	jue 31/03/11
4	Recerca informació llicències	3 días	vie 01/04/11	mar 05/04/11
5	Recerca informació SO actuals	3 días	mié 06/04/11	vie 08/04/11
6	Selecció de SO	0 días	vie 08/04/11	vie 08/04/11
7	Recerca informació Benchmark	4 días	dom 10/04/11	mié 13/04/11
8	Selecció de Benchmarks	0 días	mié 13/04/11	mié 13/04/11
9	Preparació de l'entorn	16 días	jue 14/04/11	jue 05/05/11
10	Cerca i instal·lació d'eines	5 días	jue 14/04/11	mié 20/04/11
11	Estudi i aprenentatge de les eines	6 días	jue 21/04/11	jue 28/04/11
12	Proves d'emulació dels SO	5 días	vie 29/04/11	jue 05/05/11
13	Disseny i execució de proves	14 días	vie 06/05/11	mié 25/05/11
14	Disseny de les proves	4 días	vie 06/05/11	mié 11/05/11
15	Execució de les proves	10 días	jue 12/05/11	mié 25/05/11
16	Comparativa	3 días	lun 23/05/11	mié 25/05/11
17	Anàlisis i comparativa dels resultats	3 días	lun 23/05/11	mié 25/05/11
18	Documentació	25 días	lun 16/05/11	vie 17/06/11
19	Entregar memòria	0 días	lun 13/06/11	lun 13/06/11
20	Entregar presentació i video	0 días	vie 17/06/11	vie 17/06/11
21				

Fig.2 Planificació inicial

El diagrama de Gantt resultant seria:

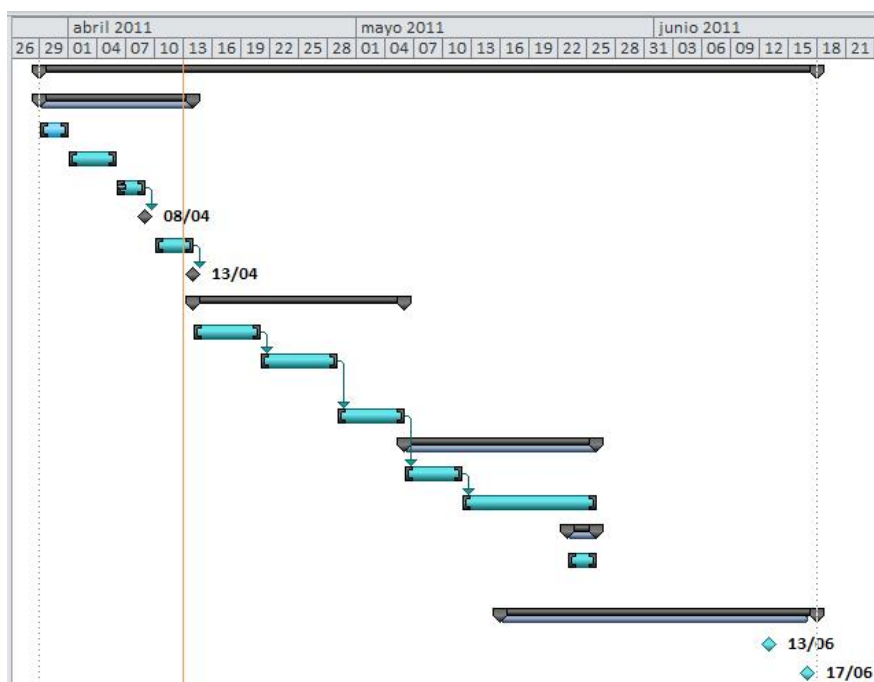


Fig 3. Diagrama de Gantt

2. CONCEPTES PREVIS

El coneixement dels conceptes previs relacionats amb l'àmbit d'aquest projecte és essencial pel bon enteniment del document. La principal noció a comprendre són els sistemes operatius GNU/Linux, per això s'explicaran els matisos implicats en el software lliure i la varietat de llicències que existeixen. Es detallaran les llicències actuals aplicats als sistemes operatius mòbils i les seves característiques. Finalment una descripció del concepte de benchmark aclarirà en que consisteixen les proves per provar l'eficiència dels sistemes subjectes a l'estudi.

2.1. SOFTWARE LLIURE

La legislació de cada país defineix el marc legal de les propietats intel·lectuals i els efectes subjectes a aquestes. Als Estats Units i a Europa, la llei aplica per defecte el copyright a les idees que una persona materialitza, qualsevol expressió, sigui text, imatge o so, està protegida pels drets d'autor. Llavors, la reproducció, la distribució i la creació de productes derivats, eixís com la explotació comercial sense autorització i consentiment del seu creador queda prohibit.

No va ser fins el 1980, casi dues dècades després de que la indústria del software s'esdevingués cada cop més competitiva, que la llicència copyright va començar a tenir vigència sobre el programari. En una època en que l'intercanvi de codi era una pràctica recurrent, el copyright provoca que el codi sigui propietat de la persona o empresa que l'hagi produït. Aquesta imposició va ser el factor que va incitar a Richard Stallman, un membre del departament d'Intel·ligència Artificial a la Massachusetts Institut of Technology, a iniciar el moviment del software lliure, Free Software Movement, en el 1983 i la creació de la Free Software Foundation uns anys més tard.

La seva intenció era defensar la possibilitat de continuar intercanviant codi sense perdre els drets "d'executar, copiar, distribuir, estudiar i millorar el programa" lliurement. Això va desembocar en la implementació del projecte GNU, un conjunt d'eines lliures compatibles amb UNIX i la confecció d'una nova llicència que protegeixi i garanteixi les llibertats dels programadors i dels usuaris. Aquestes llibertats fonamentals del programari lliure que promulga Stallman es defineixen de la manera següent:

- ***Llibertat 0: La llibertat per a executar el programa, per a qualsevol propòsit.***
- ***Llibertat 1: La llibertat d'estudiar com treballa el programa, i adaptar-lo a les necessitats pròpies.***
- ***Llibertat 2: La llibertat de redistribuir còpies per a poder ajudar als vostres veïns.***
- ***Llibertat 3: La llibertat per a millorar el programa, i alliberar les vostres millores al públic, per a què tota la comunitat pugui beneficiar-se.***

Libertats del programari lliure, font: gnu.org

El moviment del software lliure inspira altres variants com per exemple la del Open Source, que va sorgir a finals dels 90. La iniciativa Open Source insisteix en una nova definició del model de programari lliure i és important no confondre el “Free Software” i el “Open Source” encara que comparteixen molts punts en comú.

2.2. SOFTWARE OPEN SOURCE

A l'any 1998, Eric S. Raymond i Bruce Perens suggereixen que el nom de Free Software induïa al malentès en el terme anglès “Free” que pot significar tant “lliure” com a “gratuït” i socialment s'entenia com el tipus de programari que no suposa cap cost econòmic i que s'oposava a la comercialització. Per tant, van fundar la Iniciativa Open Source i van proposar el nom de Open Source (Codi Font Obert) enlloc de Free Software, la principal intenció d'aquesta és mantenir les llibertats associades al moviment de Stallman però també busca una conciliació del programari lliure amb la indústria del software. Per tant, es produeix una bifurcació de les filosofies de programari lliure.

La definició del Open Source es basa fonamentalment en la de Debian Free Software Guidelines (DFSG), les directives utilitzades pel Debian Project (del qual Perens era el líder del projecte) per poder classificar un programa com a lliure. Aquestes pautes afegeixen condicions que s'allunyen de la definició inicial del Free Software, finalment l'adaptació realitzada pel Open Source quedaria de la forma següent:

- 1. Lliure redistribució:**
el programari ha de poder ser regalat o venut lliurement
- 2. Codi font:**
ha d'estar inclòs o obtenir-se lliurement.
- 3. Permetre modificacions o treballs derivats:**
la redistribució de modificacions ha d'estar permesa.
- 4. Integritat del codi font de l'autor:**
les llicències poden requerir que les modificacions siguin redistribuïdes només com a pedaços deixant el codi original sense canvis.
- 5. No restringir el seu ús a cap persona o grup:**
ningú pot deixar-se fora.
- 6. No restringir el seu ús a cap activitat, línia de negoci o àrea d'iniciativa:**
els usuaris comercials no poden ser exclosos.
- 7. Distribució de la llicència:**
han d'aplicar-se els mateixos drets a tot el qui rebí el programa i la llicència ha de romandre intacta al distribuir o modificar el programari.
- 8. La llicència no ha de ser específica d'un producte:**
el programa no pot llicenciar-se solament com part d'una distribució major.
- 9. La llicència no ha de restringir altre programari:**
la llicència no pot obligar que algun altre programari que sigui distribuït amb el programari obert hagi també ser de codi obert.
- 10. La llicència ha de ser tecnològicament neutral:**
no ha de requerir-se l'acceptació de la llicència per mitjà d'un accés per clic de ratolí o d'altra forma específica del mitjà de suport del programari.

Definició del Open Source, font: <http://www.opensource.org/docs/osd>

2.3. TIPUS DE LLICÈNCIES OPEN SOURCE

Degut a les discrepàncies de les dues iniciatives, les llicències aprovades pel Free Software Foundation i la Open Source Initiative poden diferir. Encara que la majoria de llicències són aprovades per ambdós fundacions, existeixen algunes excepcions que són acceptades només unilateralment.



Fig 4. Logotipus de la Free Software Foundation



Fig 5. Logotipus de Open Source

Els desenvolupadors que desitgen alliberar el seu codi font en benefici de la comunitat informàtica poden escollir sota quines condicions ho fan. Una multitud de llicències recull la varietat de condicions i protegeix els drets dels informàtics. Les més importants són els que es llisten a continuació següents:

La llicència MIT

També coneguda com la llicència X confeccionada en 1988 a la MIT, és una de les primeres llicències Open Source, aprovada pel FSF i l'OSI, és també compatible amb GPL, GNU Public Licence. Senzillament concedeix el permís per tractar el codi sense restriccions sempre que s'indiqui el copyright en la capçalera del codi, però el titular del copyright, normalment el creador original, no assumirà cap responsabilitat en cas d'un ús inadequat del codi.

La llicència BSD

Juntament amb la llicència MIT, és de les primeres llicències Open Source, escrita a la Universitat de Califòrnia a Berkley, la primera versió obligava a incorporar crèdits referents a la universitat de Califòrnia i el laboratori Lawrence Berkley, degut a aquesta condició, la llicència no va ser del tot acceptada per l'OSI. En 1999 es va eliminar aquesta clàusula i la versió modificada de la llicència BSD és aprovada per FSF i OSI, és compatible amb GPL i funciona de manera similar a la llicència MIT.

La llicència Apache

La llicència Apache v.2, aprovada per la Apache Software Foundation en el 2004, és acceptada per FSF i OSI, i compatible amb GPL v.3 només en un sentit, el codi sota llicència Apache v.2 pot ser incorporat a codi sota llicència GPL v.3, però no en l'altre sentit, perquè segons ASF, obligaria a canviar la llicència a GPL v.3.

Aquesta llicència defineix detalladament en 9 clàusules les responsabilitats del creador, l'usuari, els contribuïdors i la relació amb el codi original. Permet protegir el codi derivat amb una altra llicència si compleix amb els requisits de Apache.

La llicència GPL

Concebuda en 1989 i publicada la versió 3 en 2007 per la Free Software Foundation, GNU General Public License és la llicència més important del software lliure i és aprovada també per la OSI. Aquesta llicència imposa que tots els treballs derivats s'han de distribuir sota la mateixa llicència GPL assegurant eixís els drets i llibertats de la persona que rep el codi. En la última versió de GPL, Stallman completa amb clàusules referent a les patents, les accions legals, el paper del contribuïdor i la compatibilitat amb altres llicències.

La llicència LGPL

Es tracta d'una altra llicència creada per la FSF en 1999, derivada del GPL però no tant restrictiva, GNU Less General Public License s'aplica sobretot a llibreries de subrutines per a que puguin ser utilitzades en programes no subjectes a la GPL, fins i tot propietaris. Aprovada per la OSI, és compatible amb GPL.

La llicència Mozilla Public License

La Mozilla Public License neix en 1998 quan Netscape va decidir alliberar el codi del navegador web Communicator, aquesta llicència combina les restriccions de la GPL i de la BSD, el codi sota la llicència MPL pot ser utilitzat en un programa protegit per un altre tipus de llicència, això provoca que no sigui compatible amb GPL, tanmateix és acceptada per FSF i OSI

2.4. GNU/LINUX

Els principis dels vuitanta van suposar sense dubte l'establiment dels fonaments de la filosofia del software lliure. Aquest fet es veu reflectit en dos actes molt rellevants a més de la fundació de la FSF. Per una banda, es va crear la llicència GPL, que a més d'inspirar altres llicències de programari lliure, va atorgar als desenvolupadors i usuaris els drets i la llibertat de manipular el codi dels programes sota GPL sense les restriccions habituals del copyright.

Per altra banda, s'iniciava el transcendental projecte GNU, també liderat per Richard Stallman, el qual consistia en un projecte destinat a la creació d'un sistema operatiu de lliure distribució compatible amb el sistema UNIX, un dels més importants des dels inicis dels setanta. L'acrònim GNU prové de "GNU is Not Unix", és un acrònim recursiu i destaca la importància de diferenciar-se de UNIX.

El sistema operatiu és un dels components essencials per poder fer servir un ordinador, per tant el moviment del software lliure també requeria lògicament d'un sistema operatiu lliure, una plataforma coherent amb la seva filosofia. Aquest sistema havia de proporcionar un conjunt de programes, -compiladors, editors de text, interfície gràfica, clients de correu, etc.- formant un entorn de desenvolupament lliure.

Òbviament, els elements que componen GNU estan subjectes a la llicència GPL i no s'utilitza mai codi privatiu per implementar el sistema.

El projecte GNU, des de 1985, compta amb el suport de la FSF en temes logístics i financers. Cinc anys més tard, en 1990, el sistema ja gairebé estava complet, disposava d'un conjunt d'eines, programes i llibreries per conformar un sistema operatiu funcional a l'excepció del gestor dels recursos del sistema, altrament dit, el kernel o nucli del sistema. Aquest encara estava en desenvolupament sota el nom de GNU Hurd i no estava lo suficientment avançat per gestionar el conjunt d'eines lliures disponibles.

Va ser llavors en 1992, que Linus Torvald, estudiant d'informàtica a Finlàndia, va publicar sota llicència GPL el seu nucli de sistema Linux. Es tracta d'un nucli monolític inspirat en el del sistema operatiu Minix i és compatible amb eines de GNU com el GCC (GNU C Compiler) o l'interpret de comandes BASH. Va substituir el GNU Hurd i finalment va completar el sistema GNU. A causa de la importància d'aquest componen, existeix una controvèrsia en la denominació de sistemes Linux, que haurien de dir-se GNU/Linux, donat que compta amb el conjunt d'eines GNU.

Al llarg dels anys, van sorgir centenars de distribucions de GNU/Linux, que són treballs derivats del projecte GNU. Cadascuna difereix en els components inclosos i poden ser dissenyades específicament per certs dispositius. Avui en dia es poden trobar distribucions amb suport comercial o mantingudes per la comunitat d'usuaris, algunes de les més conegudes són Debian, Slackware, Fedora, openSUSE i Ubuntu.

2.5. DISPOSITIUS MÒBILS



Fig 6. Smartphone

Avui en dia, la mobilitat de les persones i la seva productivitat estan intrínsecament relacionats. Per cobrir aquestes necessitats i també les de oci, es van concebre els dispositius mòbils de consum. Dos dels més populars que es poden trobar en el mercat tecnològic són els Netbooks i els telèfons mòbils. Amb l'arribada dels "smartphones" o telèfons intel·ligents en el món de la telefonia mòbil, s'ha introduït també la necessitat de gestionar els nous recursos mitjançant sistemes operatius.

A diferència dels Netbooks -que són merament ordinadors portàtils més petits i lleugers, la majoria compatible amb sistemes operatius de sobretaula donat que estan dissenyats amb una arquitectura x86-, els telèfons intel·ligents per tal de mantenir les característiques com la mida, el pes i l'ergonomia, i per maximitzar la relació entre el rendiment i l'autonomia, s'implementen amb una arquitectura totalment diferent anomenat ARM. Els microprocessadors ARM, Advanced RISC Machine, funcionen com indica el seu nom amb el conjunt d'instruccions RISC, Reduced Instruction Set Computing. Aquest model de disseny s'oposa al CISC, Complex Instruction Set Computing, i destaca sobretot per la seva senzillesa. Justament aquest factor permet el desenvolupament de dispositius més econòmics i de baix consum energètic.

Per consegüent, l'ús de l'arquitectura ARM comporta una incompatibilitat amb la implementació més comuna de sistemes operatius dissenyats per x86. Llavors era inevitable la implementació de programari adaptat a RISC. Diversos sistemes van ser concebuts per aquesta finalitat, un dels més importants, per exemple, va ser Symbian que comptava amb el suport de Nokia quan aquesta companyia dominava el mercat de terminals mòbils.

Des de fa més d'una dècada, els microprocessador ARM es troben present en més del 90% dels actuals telèfons mòbils, siguin o no "smartphones". Aquesta proliferació va fomentar el interès i la inversió de grans companyies, situant la plataforma ARM com a predilecta pels dispositius mòbils d'última generació. Incessablement es presenten millores en el rendiment i l'eficiència energètica degut a l'alta demanda.

2.6. BENCHMARK

El concepte de "benchmark", consisteix en l'avaluació del rendiment del software o del hardware amb l'ajuda de programes dissenyats a base d'algorismes exhaustius i simulacions de processament costoses pel sistema. Generalment els resultats es mesuren en funció del temps que es triga en resoldre les iteracions del programa.

Els mètodes de benchmark permeten obtenir informació sobre molts aspectes d'un sistema, velocitat d'escriptura, de lectura, eficiència algorísmica, ús de recursos, etc. Són útils per comparar les diferències en termes d'eficiència entre sistemes distints, malgrat tot, és necessari que les condicions siguin idèntiques per extreure dades rellevants. A tall d'exemple, si es vol comparar el maquinari, el programari instal·lat hauria de ser el mateix en la mesura de lo possible, i les proves han de ser obligatòriament les mateixes. En el cas de comparar el programari com ara el sistema operatiu, el testeig s'executarà en el mateix hardware per discriminar les diferències presents.

3. ESTAT DE L'ART DELS SISTEMES OPERATIUS MÒBILS

Actualment el nombre de sistemes operatius mòbils és molt gran i nous projectes, basats en codi Open Source o bé propietari, s'anuncien amb regularitat. El llançament de l'iPhone a l'any 2007 va suposar una revolució, la proposta d'un sistema operatiu amb una interfície gràfica extremadament amigable pel usuari va donar un gran impuls al sector dels telèfons intel·ligents.

Les empreses manufactures de terminals existents que implementen sistemes com Symbian o BlackBerry, es veuen obligats a competir amb aquesta innovadora forma d'integrar noves tecnologies com les pantalles tàctils, GPS, sensors de moviment, etc. La companyia Apple amb les consecutives versions de iPhone i el seu sistema iOS en només 3 anys ha aconseguit desbancar Nokia després de anys de domini en telefonia mòbil.

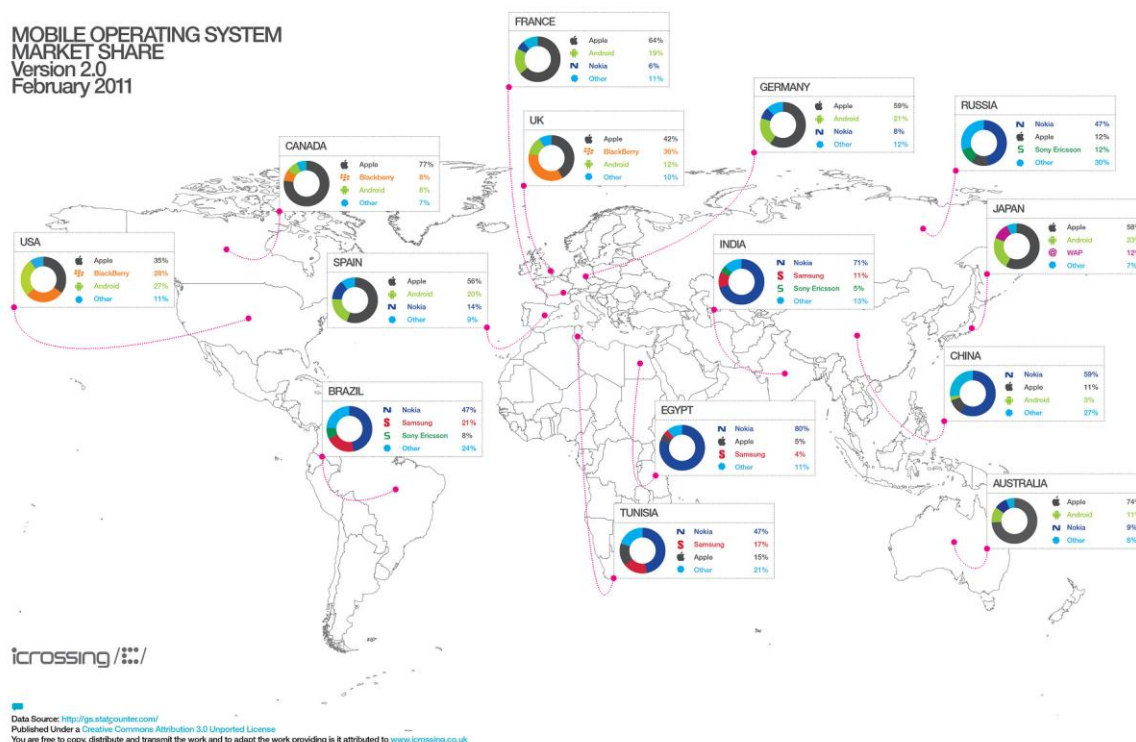


Fig 7. Mobile Operating System Market Share, font: icrossing.co.uk

En aquest mapa es pot apreciar la repartició dels sistemes operatius segons la companyia en el mercat mundial. La infografia està elaborada a partir de les dades extretes de gs.statcounter.com. Es pot constatar l'important creixement de Apple en els països més desenvolupats (en gris fosc), i al mateix temps la presència minvant de Nokia, que continua dominant en països on la tecnologia penetra amb menys celeritat (en blau fosc). És important destacar el lloc que ocupa Android (en verd), amb un percentatge cada cop més elevat en clara competència amb el iOS d'Apple.

En el següent gràfic queda representat les tendències segons sistema operatiu mòbil durant el període de desembre 2008 fins l'actualitat:

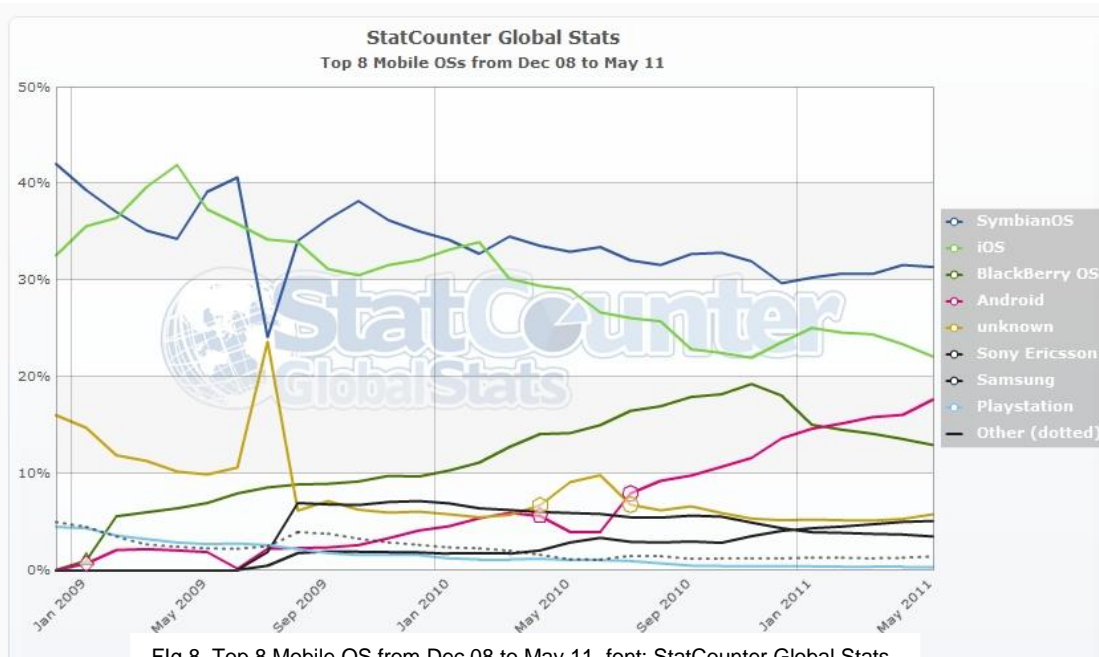


Fig 8. Top 8 Mobile OS from Dec 08 to May 11, font: StatCounter Global Stats

OS	2010	2011	2012	2015
Symbian	111,577	89,930	32,666	661
Market Share (%)	37.6	19.2	5.2	0.1
Android	67,225	179,873	310,088	539,318
Market Share (%)	22.7	38.5	49.2	48.8
Research In Motion	47,452	62,600	79,335	122,864
Market Share (%)	16	13.4	12.6	11.1
iOS	46,598	90,560	118,848	189,924
Market Share (%)	15.7	19.4	18.9	17.2
Microsoft	12,378	26,346	68,156	215,998
Market Share (%)	4.2	5.6	10.8	19.5
Other Operating Systems	11,417.40	18,392.30	21,383.70	36,133.90
Market Share (%)	3.8	3.9	3.4	3.3

Taula 1. Forecast: Mobile Communications Device Open OS Sales to End Users by OS (Thousands of Units)
Font: Gartner, abril 2011

El pronòstic mundial de sistemes operatius mòbils de Gartner per l'any 2015 calcula que Android serà el sistema operatiu més popular, mentre que Symbian de Nokia gairebé desapareixerà després del conveni amb Microsoft. Al mateix temps, el nombre de terminals amb iOS disminuirà i els telèfons de Microsoft amb el sistema Windows Phone superaran els de Apple.

Les dades de la taula són estimacions realitzades a partir de la informació disponible en abril 2011 i és molt difícil que la previsió encerti amb poc marge d'error les dades de l'any 2015 ja que les estratègies comercials de les companyies són desconegudes. Les fonts de Gartner i gs.countstat difereixen considerablement respecte a Android i Apple. No obstant, la taula aporta una visió general de la tendència dels sistemes operatius mòbils en competència actualment.

3.1. SISTEMES TANCATS

Els sistemes operatius que no estan subjectes a una llicència lliure segueixen una política de codi propietari que restringeix l'accés al codi font i no permet tampoc la seva lliure distribució o manipulació. Aquests sistemes permeten normalment als desenvolupadors implementar programes per distribuir-los comercialment, en canvi han de ser aprovats per la companyia que verifica que es compleixen les condicions de la seva pròpia llicència.

3.1.1. BADA

Nom del sistema operatiu	Bada
Desenvolupador	Samsung Electronics Co.,Ltd.
Any de publicació	2009
Família de S.O.	BSD
Versió actual	1.2.1
Dissenyat per arquitectura	ARM
Programat en	C++
Tipus de llicència	Propietari



Fig 9. Logotipus Bada

Després de comercialitzar els seus terminals amb Android o Windows Phone incorporats, Samsung finalment afegeix la seva línia de telèfons amb Bada, un sistema operatiu desenvolupat per la mateixa empresa. Aquest sistema operatiu s'ha proclamat com a obert, però realment es tracta d'un sistema tancat i segueix un model de desenvolupament similar al de iOS.

3.1.2 BLACKBERRY

Nom del sistema operatiu	BlackBerry OS
Desenvolupador	Research In Motion
Any de publicació	1999
Família de S.O.	BlackBerry
Versió actual	7.0
Dissenyat per arquitectura	ARM
Programat en	Java
Tipus de llicència	Propietari



Fig 10. Logotipus Blackberry

RIM és una de les primeres empreses en incorporar funcionalitats pròpies de telèfons intel·ligents en els seus dispositius. La línia Blackberry a competir amb els PDA i s'ha pogut mantenir operativa, l'arribada de les tecnologies 3G va permetre millorar els serveis que ja oferia amb la xarxa GSM. Fins fa poc, els terminals Blackberry no integraven pantalla tàctil i es caracteritzaven per un teclat físic QWERTY.

3.1.3 iOS

Nom del sistema operatiu	iOS
Desenvolupador	Apple Inc.
Any de publicació	2007
Família de S.O.	MAC OS X
Versió actual	4.3
Dissenyat per arquitectura	ARM
Programat en	C/C++, Objective C
Tipus de llicència	Propietari



Fig 11. Logotipus iOS

La empresa de Steve Jobs és popularment coneguda per la seva gama de reproductors de música iPod i els seus equips informàtics que inclouen un sistema operatiu propi, el MAC OS basat en UNIX. L'èxit del seu primer smartphone a l'any 2007 va ser rotund degut principalment a la integració de noves tecnologies i el detall al disseny tant en el seu software com en l'aparell en sí. Va popularitzar el concepte de les "Apps", es a dir aplicacions que es distribueixen en una mateixa plataforma oficial de Apple.

3.1.4 SYMBIAN

Nom del sistema operatiu	Symbian OS
Desenvolupador	Nokia, Symbian Foundation
Any de publicació	2001
Família de S.O.	Symbian
Versió actual	9.5
Dissenyat per arquitectura	ARM
Programat en	C++
Tipus de llicència	Propietari: Nokia Symbian Licence



Fig 12. Logotipus Symbian OS

El sistema operatiu Symbian va predominar en els terminals de telefonia mòbil juntament amb Nokia quan aquesta era la empresa més important del sector. Symbian oferia un sistema de gestió senzill bastant genèric per una gran varietat d'empreses fabricants de telèfons mòbils, Sony Ericsson, Nokia, Motorola entre les més grans. Amb l'aparició del iOS i Android, Symbian ha perdut tot el terreny comercial i a l'any la Symbian Foundation va anunciar que s'alliberaria el codi del sistema operatiu i es tornaria en un projecte Open Source, però al novembre 2010 la fundació va anunciar que Symbian OS deixaria de ser operatiu. Aquesta circumstància s'ha produït arrel de la decisió de Nokia de treballar conjuntament amb Microsoft.

3.1.5. webOS

Nom del sistema operatiu	webOS
Desenvolupador	Palm, HP
Any de publicació	2009
Família de S.O.	Linux
Versió actual	3.0
Dissenyat per arquitectura	ARM
Programat en	C
Tipus de llicència	Propietari amb components GNU GPL



Flg 13. Logotipus webOS

Palm Inc. era una coneguda empresa especialitzada en la manufactura de PDAs amb sistema operatiu PalmOS, a l'any 2009, va presentar un nou sistema operatiu anomenat webOS. Aquest integra tecnologies totalment enfocades als nous paradigmes de la web, com ara la computació en núvol o els recents estàndards de programació web HTML5, CSS3, Javascript, AJAX, amb funcionalitats natives del sistema com la gestió dels contactes, del calendari, d'arxius etc. Al febrer 2011, la empresa HP va adquirir Palm i ara és la propietària del sistema webOS.

3.1.6 WINDOWS MOBILE

Nom del sistema operatiu	Windows Phone
Desenvolupador	Microsoft Corporation
Any de publicació	1996
Família de S.O.	Windows Embedded Compact
Versió actual	7.0
Dissenyat per arquitectura	ARM, MIPS, x86
Programat en	.NET i altres
Tipus de llicència	Propietari: Microsoft EULA



Flg 14. Logotipus Windows Phone

La corporació Microsoft va estar present en els primers dispositius mòbils amb funcions avançades. Antigament sota el nom de Windows Mobile, el sistema operatiu de Microsoft per PDAs oferia un entorn similar al de Windows per ordinadors de sobretaula, amb un conjunt de programes com el navegador web Explorer i el set d'ofimàtica Microsoft Office adaptat per PDA.

Darrerament Microsoft va iniciar un apropament cap als projectes en Open Source, però inesperadament va decidir excloure tota aplicació sota llicència GNU GPL v.3, Affero GPL v.3 i LGPL v.3 de la plataforma oficial de distribució d'aplicacions "Windows Marketplace".

3.2. SISTEMES LLIURES

Els sistemes operatius lliures es distribueixen normalment sota llicències GNU GPL o al menys compatible, i concedeixen els drets a l'usuari de treballar el codi i de desenvolupar programes sense restriccions segons les llibertats del Software Lliure o la definició del Open Source.

3.2.1. ACCESS LINUX PLATFORM

Nom del sistema operatiu	Access Linux Platform
Desenvolupador	Access Co. Ltd.
Any de publicació	2007
Família de S.O.	Linux
Versió actual	2.4
Dissenyat per arquitectura	ARM
Programat en	C, C++
Tipus de llicència	GNU GPL v.2



Fig 15. Logotipus Access Linux Platform

L'Access Linux Platform és el sistema operatiu amb kernel Linux desenvolupat per la companyia japonesa Access. A la seva pàgina web es poden descarregar les eines de desenvolupament i es pot emular, però malauradament no es comercialitza cap dispositiu amb aquest sistema operatiu incorporat i la manca de notícies des de fa anys apunta a que el desenvolupament d'aquest projecte està abandonat o aturat.

3.2.2. ANDROID

Nom del sistema operatiu	Android
Desenvolupador	Open Handset Alliance, Google Inc.
Any de publicació	2008
Família de S.O.	Linux
Versió actual	3.1
Dissenyat per arquitectura	ARM, MIPS, x86
Programat en	C, C++, Java
Tipus de llicència	Apache v.2



Fig 16. Logotipus Android

Android és sense dubte el sistema lliure més important actualment entre els sistemes operatius dissenyats pels terminals intel·ligents. Inicialment desenvolupat per una petita empresa anomenada Android Inc., Google va acabar adquirint-la i des de llavors, el projecte Android no ha cessat de rebre suport de grans companyies que es sumen al consorci del Open Handset Alliance. Implementat a partir del kernel 2.6 de Linux, Android es va llançar al mercat com a sistema operatiu d'un dispositiu de la marca HTC.

3.2.3. LiMo

Nom del sistema operatiu	LiMo 4
Desenvolupador	LiMo Foundation
Any de publicació	-
Família de S.O.	Linux
Versió actual	-
Dissenyat per arquitectura	-
Programat en	-
Tipus de llicència	GNU GPL v.2, LGPL v.2.0 i v.2.1, Apache v.2.0



Fig 17. Logotipus LiMo Foundation

La fundació LiMo (Linux Mobile) és un consorci de empreses que, segons la informació disponible en línia, s'han reunit per "crear el primer sistema basat en Linux realment obert i hardware independent per a dispositius mòbils". Entre les empreses fundadores es troben NEC, Ntt Docomo, Panasonic, Samsung i Vodafone. La especificació de LiMo 4 està terminada, actualment es troba en fase d'implementació i es preveu la distribució del codi font a partir de juliol 2011.

3.2.4. MAEMO

Nom del sistema operatiu	Maemo
Desenvolupador	Nokia, Comunitat Maemo
Any de publicació	2005
Família de S.O.	Linux
Versió actual	5.0
Dissenyat per arquitectura	ARM
Programat en	C, C++
Tipus de llicència	GNU GPL



Fig 18. Logotipus Maemo

Maemo és una plataforma basada en Linux, desenvolupada principalment per Nokia i membres d'una extensa comunitat open source de més de 20000 inscrits. La comunitat Maemo col·labora estretament amb altres comunitats com la del Debian o amb Mozilla i van crear el sistema operatiu Maemo per a dispositius de Nokia en el 2005. El projecte va prosperar fins arribar a la cinquena versió del sistema i a l'any 2010 es va anunciar la fusió dels projectes Maemo i Moblin per donar lloc al sistema operatiu Meego.

3.2.5. MEEGO

Nom del sistema operatiu	Meego
Desenvolupador	Linux Foundation
Any de publicació	2010
Família de S.O.	Linux
Versió actual	1.2
Dissenyat per arquitectura	ARM, x86
Programat en	C++
Tipus de llicència	Multillicència: GPL, BSD...



Fig 19. Logotipus MeeGo

Meego és el resultat de la fusió en el 2010 dels projectes Maemo de Nokia i Moblin de Intel, està dissenyat per a dispositius mòbils d'arquitectura ARM i x86. A més de Nokia i Intel, participen empreses com Novell i AMD.

3.2.6. MOBLIN

Nom del sistema operatiu	Moblin
Desenvolupador	Intel / Linux Foundation
Any de publicació	2008
Família de S.O.	Linux
Versió actual	2.1
Dissenyat per arquitectura	x86
Programat en	C++
Tipus de llicència	GNU GPL

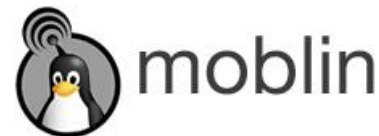


Fig 20. Logotipus Moblin

El disseny de Moblin està enfocat a la arquitectura x86, normalment per a dispositius com els netbooks. És projecte comú de Intel i la Linux Foundation que es combina en el 2010 amb Maemo per formar Meego.

3.2.7. OPHONE

Nom del sistema operatiu	OPhone (Open Mobile System Phone)
Desenvolupador	China Mobile
Any de publicació	2009
Família de S.O.	Linux
Versió actual	2.0
Dissenyat per arquitectura	ARM
Programat en	C, C++, Java
Tipus de llicència	Apache v.2, GNU GPL v.2



Fig 21. Logotipus OPhone

Dissenyat per China Mobile, OPhone utilitza les tecnologies de Android per crear un sistema modificat. Malgrat el poc temps d'existència, aquesta alternativa a l'Android ja es ven a China amb companyes com Lenovo, Motorola, Dell i LG.

4. DISTRIBUCIONS SELECCIONATS

Les distribucions de sistemes operatius GNU/Linux que s'analitzaran en aquest projecte són els següents:

- **Android**
- **Meego**
- **OPhone**
- **webOS**

4.1. CRITERIS DE SELECCIÓ

El principal criteri de selecció és que el sistema ha d'estar basat en GNU/Linux, els sistemes que compleixen aquesta condició són gairebé tots els sistemes lliures i webOS:

- Access Linux Platform
- Android
- LiMo
- Maemo
- Meego
- Moblin
- OPhone
- webOS

El segon criteri és que el sistema continuï actiu. Això descarta la distribució de Access Linux Platform, que des de l'any 2009 el projecte es va aturar completament, encara que no s'ha anunciat oficialment el seu tancament. Els sistemes Maemo i Moblin queden descartats també ja que Meego és el successor d'aquests dos sistemes. I finalment LiMo tampoc és candidat degut a que el sistema operatiu encara està en fase de desenvolupament i no es disposa de cap codi encara.

5. ANALISIS

Per començar és necessari poder emular cada sistema operatiu. La manera més senzilla és obtenir els SDK (Software Development Kits) que inclouen els emuladors i les imatges per simular el programari. El procés d'instal·lació es detalla en els annexos que expliquen breument on trobar les eines i les instruccions a seguir.

Un cop instal·lats, s'ha de poder entrar per intèrpret de comandes, ja que la informació accessible des de la interfície gràfica és quasi nul·la. Per conèixer els diferents sistemes operatius a grans trets, primer es mostraran les eines i comandes incloses per defecte. Aquestes es troben en la carpeta `system/bin/`, i es llistarà els binaris senzillament amb la comanda `ls`.

Després s'extraurà el resultat d'una execució de la comanda `top`, després d'iniciar el sistema per donar un cop d'ull als processos activats i les memòries que poden arribar a ocupar. S'esperarà un temps prudencial abans d'executar la comanda per tal de que el sistema s'estabilitzi i els processos d'inici s'aturin.

Per últim s'ha recopilat els diagrames oficials de l'arquitectura de cada sistema operatiu per poder observar i comparar les diferents implementacions dels elements que conformen el nucli del sistema i les capes superiors.

5.1. ANDROID



Fig 22. Captura de pantalla emulador Android

Per executar l'emulador amb la imatge de l'Android i entrar a la shell s'ha d'introduir la comanda següent

```
emulator -avd Android -shell
```

Eines incorporades:

am	ime	omx_tests	start
app_process	input	ping	stop
applypatch	InputChannel_test	pm	String8_test
applypatch_stati	InputDispatcher_test	pppd	surfaceflinger
c	InputPublisherAndConsumer_test	printenv	svc
audioloop	InputReader_test	ps	sync
bmgr	insmod	qemud	system_serve
bootanimation	installld	qemu-props	r
bugreport	ioctl	racoon	testid3
cat	ionice	radiooptions	toolbox
check_prereq	keystore	reboot	top
chmod	keystore_cli	record	umount
chown	kill	recovery	updater
cmp	linker	renice	uptime
dalvikvm	ln	rild	vdc
date	log	rm	vmstat
dd	logcat	rmdir	vold
debuggerd	logwrapper	rmmod	watchprops
dexopt	Looper_test	route	wipe
df	ls	rtp_test	
dhcpcd	lsmod	run-as	
dmesg	lsof	schedtest	
dumpstate	mediaserver	schedtop	
dumpsys	mkdir	sdcard	
dvz	monkey	sendevent	
flash_image	mount	service	
fsck_msdos	mtpd	servicemanag	
gdbjithelper	mv	er	
gdbserver	nandread	setconsole	
getevent	ndc	setprop	
getprop	netcfg	sh	
gzip	netd	showlease	
hd	netstat	skia_test	
id	newfs_msdos	sleep	
ifconfig	notify	smd	
iftop	ObbFile_test	stagefright	

Resultat de l'execució de la comanda top després de l'inici:

User 1%, System 2%, IOW 0%, IRQ 0%

User 6 + Nice 0 + Sys 9 + Idle 298 + IOW 0 + IRQ 0 + SIRQ 0 = 313

PID	CPU%	S	#THR	VSS	RSS	PCY	UID	Name
116	0%	S	9	85616K	24588K	fg	system	com.android.systemui
3	0%	S	1	0K	0K	fg	root	ksoftirqd/0
4	0%	S	1	0K	0K	fg	root	events/0
5	0%	S	1	0K	0K	fg	root	khelper
6	0%	S	1	0K	0K	fg	root	suspend
7	0%	S	1	0K	0K	fg	root	kblockd/0
8	0%	S	1	0K	0K	fg	root	cqueue
9	0%	S	1	0K	0K	fg	root	kseriod
10	0%	S	1	0K	0K	fg	root	kmmcd
11	0%	S	1	0K	0K	fg	root	pdflush
12	0%	S	1	0K	0K	fg	root	pdflush
13	0%	S	1	0K	0K	fg	root	kswapd0
14	0%	S	1	0K	0K	fg	root	aio/0
22	0%	S	1	0K	0K	fg	root	mtddblockd
23	0%	S	1	0K	0K	fg	root	kstriped
24	0%	S	1	0K	0K	fg	root	hid_compat
25	0%	S	1	0K	0K	fg	root	rpciod/0
26	0%	S	1	232K	136K	fg	root	/sbin/ueventd
27	0%	S	1	804K	276K	fg	system	/system/bin/servicemanager
28	0%	S	3	3864K	568K	fg	root	/system/bin/vold
29	0%	S	3	3836K	560K	fg	root	/system/bin/netd
30	0%	S	1	664K	264K	fg	root	/system/bin/debuggerd
31	0%	S	4	5396K	700K	fg	radio	/system/bin/rild
32	0%	S	1	74072K	27140K	fg	root	zygote
33	0%	S	5	16972K	3764K	fg	media	/system/bin/mediaserver
34	0%	S	1	812K	316K	fg	root	/system/bin/installd
35	0%	S	1	1744K	432K	fg	keystore	/system/bin/keystore
37	0%	S	1	824K	340K	fg	root	/system/bin/qemud
39	0%	S	1	732K	328K	fg	shell	/system/bin/sh
40	0%	S	2	1300K	140K	fg	root	/sbin/adbd
60	0%	S	44	135412K	36564K	fg	system	system_server
109	0%	S	8	86140K	22820K	fg	app_4	jp.co.omronsoft.openwnn
113	0%	S	19	99176K	24460K	fg	radio	com.android.phone
120	0%	S	10	96556K	29816K	fg	app_13	com.android.launcher
154	0%	S	8	86660K	21404K	bg	system	com.android.settings
1	0%	S	1	268K	180K	fg	root	/init
83	0%	S	9	84312K	21436K	bg	app_19	com.android.deskclock
202	0%	S	10	86352K	22104K	bg	app_0	android.process.media
215	0%	S	9	95600K	21724K	bg	app_15	com.android.mms
236	0%	S	9	85976K	22892K	bg	app_28	com.android.email
249	0%	S	8	84000K	21140K	bg	app_2	com.android.quicksearchbox
260	0%	S	8	83504K	20376K	bg	app_5	com.android.music
269	0%	S	8	82972K	19960K	bg	app_24	com.android.protips
175	0%	S	12	88612K	24000K	bg	app_6	android.process.acore
2	0%	S	1	0K	0K	fg	root	kthreadd

Arquitectura

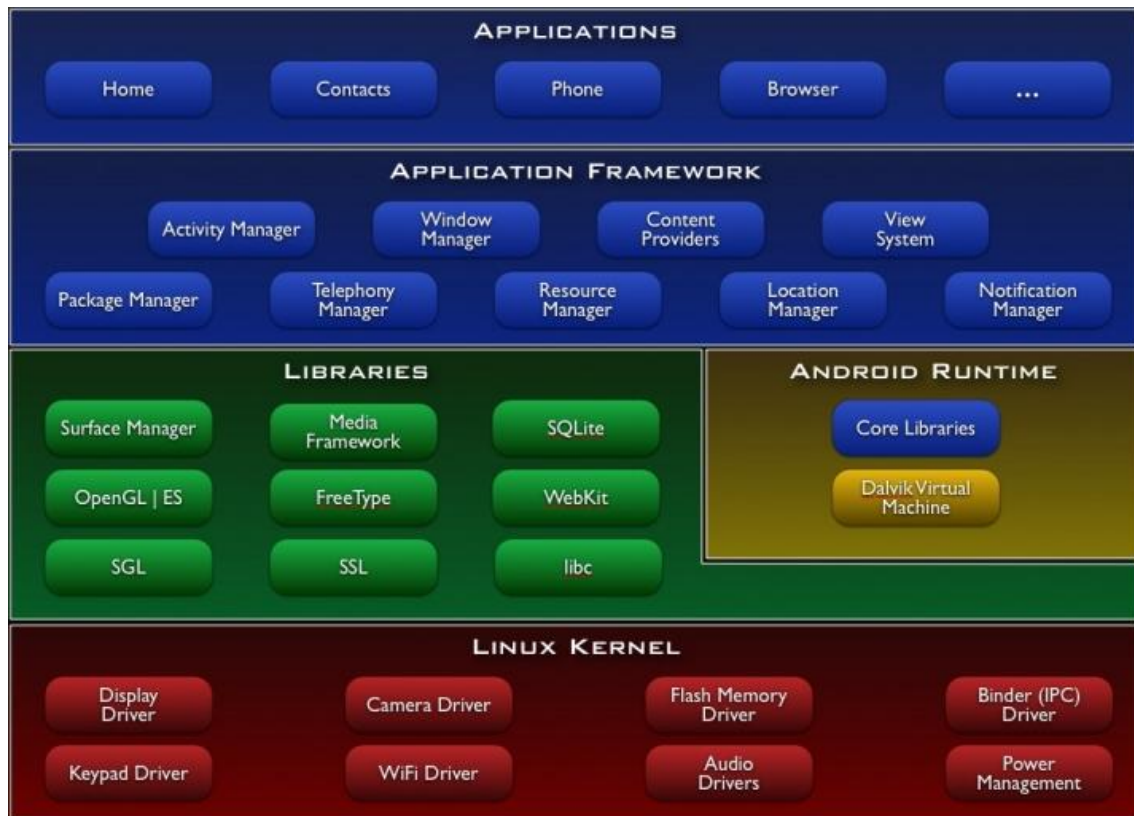


Fig 23. Diagrama de l'arquitectura del sistema Android

L'emulador que ve per defecte amb el SDK de Android és una adaptació del QEMU, es tracta d'un emulador sota llicències GNU GPL i LGPL capaç de simular diversos sistemes operatius en un ampli varietat de arquitectures.

Accedint mitjançant la shell, es pot apreciar que el conjunt de binaris per defecte és molt més reduït que el de una distribució de Linux més comú com Debian o Ubuntu. Un cop finalitzat l'inici del sistema, es detecta fins a 313 processos, 6 processos actius de l'usuari, 9 del sistema i la resta en mode "sleep".

El diagrama de la arquitectura de Android, mostra clarament el kernel monolític de Linux com a base de tota la estructura. Directament a sobre es troben les llibreries que es fa servir per comunicar amb l'entorn d'execució de l'Android i els frameworks de les aplicacions que donen interacció amb l'usuari.

5.2. MEEGO

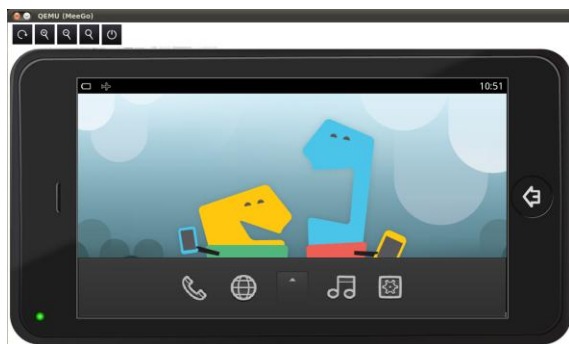


Fig 24. Captura de pantalla emulador MeeGo

Per executar l'emulador amb un "runtime" de MeeGo s'utilitza la comanda:

```
sudo mad remote -r <runtime name> poweron
```

Per accedir al sistema mitjançant interpret de comandes, s'ha d'efectuar una connexió ssh a través del port 6666:

```
ssh meego@127.0.0.1 -p 6666
```

Eines incorporades:

alsaunmute	gzip	tar
arch	hostname	taskset
awk	igawk	touch
basename	kbd_mode	tracepath
bash	link	tracepath6
cat	ln	true
chgrp	loadkeys	unlockmgr_server
chmod	login	umount
chown	ls	uname
cp	mkdir	unicode_start
cpio	mknod	unicode_stop
cut	mktemp	unlink
date	more	usleep
dbus-cleanup-sockets	mount	vi
dbus-daemon	mountpoint	view
dbus-monitor	mv	zcat
dbus-send	netstat	
dbus-uuidgen	nice	
dd	pgawk	
df	pgawk-3.1.7	
dmesg	ping	
dnsdomainname	ping6	
domainname	ps	
dumpkeys	pwd	
echo	rm	
egrep	rmdir	
env	rpm	
ex	rvi	
false	rview	
fgrep	sed	
find	setfont	
fusermount kill	sh	
gawk	sleep	
gawk-3.1.7	sort	
gettext	stty	
grep	su	
gunzip	sync	

Resultat de l'execució de la comanda top després de l'inici:

top - 15:32:58 up 4 min, 1 user, load average: 0.01, 0.07, 0.03
Tasks: 93 total, 1 running, 91 sleeping, 0 stopped, 1 zombie
Cpu(s): 0.0%us, 0.3%sy, 0.0%ni, 99.7%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 1031028k total, 312056k used, 718972k free, 7892k buffers
Swap: 0k total, 0k used, 0k free, 145632k cached

PID	USER	PR	NI	VIRTRES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	2100 480	424	S	0.0	0.0	0:01.34	init
2	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0 0	0	S	0.0	0.0	0:00.00	ksoftirqd/0
4	root	RT	0	0 0	0	S	0.0	0.0	0:00.00	migration/0
5	root	RT	0	0 0	0	S	0.0	0.0	0:00.00	watchdog/0
6	root	20	0	0 0	0	S	0.0	0.0	0:00.05	events/0
7	root	20	0	0 0	0	S	0.0	0.0	0:00.00	khelper
8	root	20	0	0 0	0	S	0.0	0.0	0:00.00	async/mgr
9	root	20	0	0 0	0	S	0.0	0.0	0:00.00	pm
10	root	20	0	0 0	0	S	0.0	0.0	0:00.00	sync_supers
11	root	20	0	0 0	0	S	0.0	0.0	0:00.00	bdi-default
12	root	20	0	0 0	0	S	0.0	0.0	0:00.03	kblockd/0
13	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kacpid
14	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kacpi_notify
15	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kacpi_hotplug
16	root	20	0	0 0	0	S	0.0	0.0	0:00.00	ata_aux
17	root	20	0	0 0	0	S	0.0	0.0	0:00.00	ata_sff/0
18	root	20	0	0 0	0	S	0.0	0.0	0:00.00	khubd
19	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kseriod
20	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kmmcd
21	root	20	0	0 0	0	S	0.0	0.0	0:00.00	cfg80211
22	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kondemand/0
23	root	20	0	0 0	0	S	0.0	0.0	0:00.00	kswapd0
24	root	20	0	0 0	0	S	0.0	0.0	0:00.00	aio/0
25	root	20	0	0 0	0	S	0.0	0.0	0:00.00	crypto/0
33	root	20	0	0 0	0	S	0.0	0.0	0:00.00	scsi_eh_0
34	root	20	0	0 0	0	S	0.0	0.0	0:00.00	scsi_eh_1

Arquitectura

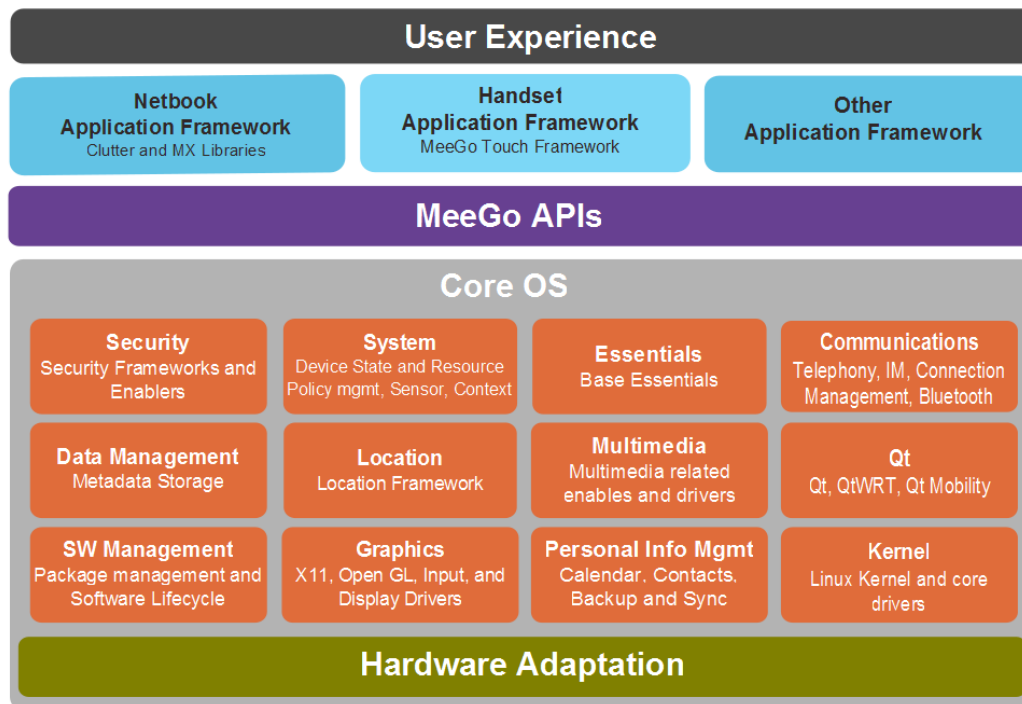


Fig 25. Diagrama de l'arquitectura del sistema MeeGo

El SDK de Meego inclou un emulador com el de l'Android, basat en QEMU però aquest és diferent i s'utilitza l'eina de desenvolupament creada per Maemo anomenat MADDE, "Maemo Application Development and Debugging Environment" per executar els entorns de QEMU.

Primer de tot, el llistat de binaris revela eines com per exemple el *vi*, *gzip*, que es distribueixen sota llicències diferents, BSD i GNU GPL respectivament, això explica per què Meego és multi llicència.

Després, l'execució de *top* mostra tan sols 93 processos, només 1 executant-se, que és el propi *top*, un zombie i 91 en sleep. El nombre de processos és molt inferior al que s'ha pogut comprovar en l'Android.

L'arquitectura d'aquest sistema també és molt diferent, les llibreries i frameworks que trobem normalment en nivells superiors a l'Android, en Meego es troben a nivell de core, molt més proper del nucli, formant el "Core OS". El pont cap als diferents conjunts d'aplicacions són els API's de Meego.

5.3. OPHONE



Per executar l'emulador amb la imatge de l'OPhone i accedir a la shell s'ha d'introduir la comanda següent

```
emulator -avd OPhone -shell
```

Flg 26. Captura de pantalla emulador OPhone

Eines incorporades:

agps_client	fsck_msdos	midletvm	sdutil
akmt	gdbserver	mkdir	sendevent
am	getevent	modem	service
amixer	getprop	modem_start.sh	servicemanager
aplay	gzip	modem_stop.sh	setconsole
app_process	hciattach	monkey	setprop
app_update	hciattach_dtr	mount	sh
applypatch	hciconfig	mountd	showlease
applypatch_static	hcitool	mtpd	sleep
at_pass_through	hd	mv	smd
audBPTtest	http_test	naga	spi-test
audhwtest	id	netcfg	start
avahi-autoipd	ifconfig	netstat	stop
bluetoothd	iftop	newfs_msdos	surfaceflinger
bmgr	iic-test	notify	svc
bootanimation	ime	omstcmd_demo	sync
borqs_wpa_cli	inno-test	ping	system_server
borqshooker	input	pm	tcmd
bugreport	insmod	pppd	test_cmmb
cat	installd	prdcfg_bootup	toolbox
cbmon	ioctl	PRDCFG_dump	top
check_prereq	iperf	printenv	tvtest
chmod	iptables	procedure	umount
chown	iwconfig	ps	updater
cmp	iwevent	qemud	user_auth_test
dalvikvm	iwlist	qemu-props	vmstat
date	iwpriv	racoon	vold
dbus-daemon	keystore	radiooptions	wapi_suppllicant
dd	keystore_cli	reboot	watchprops
debuggerd	kill	recovery	wipe
dexopt	lapi_unit	renice	wlanconfig
df	lapisrv	rilbox	wpa_cli
dhcpcd	linker	rild	wpa_suppllicant
dmesg	livetime	rm	
dumpcrash	ln	rmdir	
dumpstate	log	rmmod	
dumpsys	logagent	route	
dvz	logcat	savelog	
fattest	logwrapper	schedtest	
flash_image	ls	schedtop	
fota	lsmod	sdm	
fota_update	mediaserver	sdptool	

Resultat de l'execució de la comanda top després de l'inici:

User 24%, System 15%, IOW 0%, IRQ 0%

User 74 + Nice 6 + Sys 49 + Idle 197 + IOW 0 + IRQ 0 + SIRQ 0 = 326

PID	CPU%	S	#THR	VSS	RSS	PCY	UID	Name
63	18%	S	39	179536K	40912K	fg	system	system_server
154	6%	S	14	135144K	22524K	bg	app_23	com.db4o.servo.search
104	4%	S	15	192060K	29692K	fg	radio	com.android.phone
203	3%	R	1	928K	372K	fg	root	top
209	2%	S	13	135620K	22164K	bg	app_32	oms.dcd
6	0%	S	1	0K	0K	fg	root	suspend
7	0%	S	1	0K	0K	fg	root	kblockd/0
8	0%	S	1	0K	0K	fg	root	cqueue
9	0%	S	1	0K	0K	fg	root	kseriod
10	0%	S	1	0K	0K	fg	root	kmmcd
11	0%	S	1	0K	0K	fg	root	pdflush
12	0%	S	1	0K	0K	fg	root	pdflush
1	0%	S	1	560K	256K	fg	root	/init
14	0%	S	1	0K	0K	fg	root	aio/0
21	0%	S	1	0K	0K	fg	root	mtddblockd
22	0%	S	1	0K	0K	fg	root	hid_compat
23	0%	S	1	0K	0K	fg	root	rpciod/0
24	0%	S	1	764K	316K	fg	root	/system/bin/sh
25	0%	S	1	832K	248K	fg	system	/system/bin/servicemanager
26	0%	S	2	1876K	380K	fg	root	/system/bin/mountd
27	0%	S	1	696K	244K	fg	root	/system/bin/debuggerd
28	0%	S	12	13832K	912K	fg	radio	/system/bin/rild
29	0%	S	1	96956K	28320K	fg	root	zygote
30	0%	S	6	22268K	3884K	fg	media	/system/bin/mediaserver
32	0%	S	1	820K	280K	fg	root	/system/bin/installd
33	0%	S	1	1652K	392K	fg	keystore	/system/bin/keystore
34	0%	S	1	764K	316K	fg	root	/system/bin/sh
35	0%	S	1	852K	324K	fg	root	/system/bin/qemud
37	0%	S	2	1340K	164K	fg	root	/sbin/adbd
44	0%	S	1	816K	296K	fg	root	/system/bin/qemu-props
102	0%	S	7	128568K	20540K	fg	app_1	com.android.inputmethod.borqs
107	0%	S	14	170336K	40004K	fg	app_1	android.process.acore
111	0%	S	7	132880K	21380K	fg	app_18	oms.mms
113	0%	S	7	132448K	20028K	fg	system	android.process.omsservice
117	0%	S	7	152828K	35688K	fg	app_10	oms.home
175	0%	S	9	132932K	23056K	bg	system	oms.pim
221	0%	S	6	177804K	20616K	bg	system	com.android.settings
231	0%	S	7	126280K	20080K	bg	app_4	android.process.media
13	0%	S	1	0K	0K	fg	root	kswapd0
2	0%	S	1	0K	0K	fg	root	kthreadd
3	0%	S	1	0K	0K	fg	root	ksoftirqd/0
4	0%	S	1	0K	0K	fg	root	events/0
5	0%	S	1	0K	0K	fg	root	khelper

Arquitectura

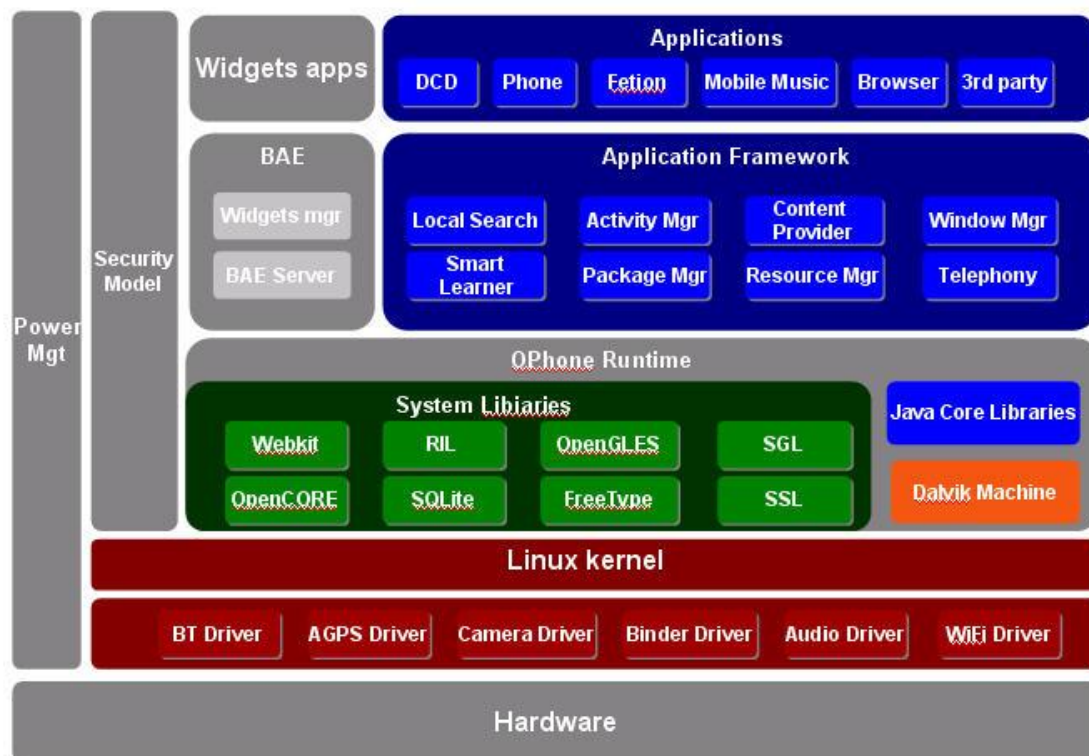


Fig 27. Diagrama de l'arquitectura OPhone

L'OPhone és un derivat del Android, les seves eines s'instal·len com a una extensió del SDK de Android. Per tant, l'emulador QEMU i l'entorn de desenvolupament són idèntics.

L'OPhone a primera vista incorpora molts més binaris que els altres sistemes. El conjunt d'eines incloses són utilitats típiques de GNU i altres BSD. L'execució de *top* mostra un sistema sobrecarregat amb una quantitat considerable de processos user i system corrent.

L'arquitectura del sistema és semblant a la del Android, a diferencia d'alguns afegits com el Power Management, el Security Model, els widgets i el BAE. L'entorn de OPhone engloba les llibreries i Java.

5.4. webOS

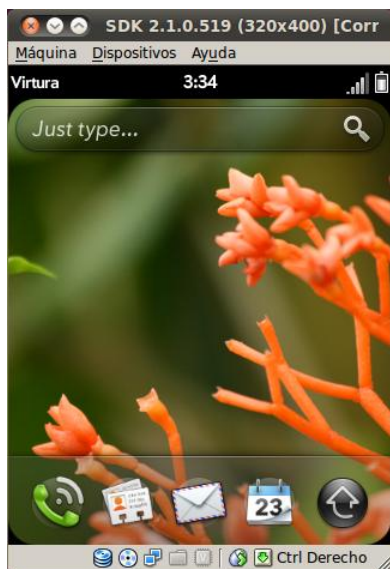


Fig 28. Captura de pantalla emulador webOS

Primer s'executa l'emulador de webOS:

```
palm-emulator
```

Després s'accedeix com amb el sistema Meego, amb una connexió ssh, però al port 5522:

```
ssh -p 5522 root@localhost
```

Eines incorporades:

addgroup	false	mountpoint	setarch
adduser	fgrep	mv	setserial
ash	grep	netstat	sh
busybox	gunzip	netstat.net-tools	sleep
cat	gzip	nice	stty
catv	hostname	node	su
chgrp	kill	node_spawner	sync
chmod	linux32	novacom	tar
chown	linux64	pidof	touch
cp	ln	pidof.sysvinit	true
date	login	ping	tzset
dd	ls	ps	umount
df	lsmod	pwd	uname
dmesg	mkdir	resizefat	usbmon
dnsdomainname	mknod	rm	usleep
dumpkmap	mktemp	rmdir	vi
echo	more	run-parts	zcat
egrep	mount	sed	

Resultat de l'execució de la comanda top després de l'inici:

```
top - 09:13:12 up 6 min, 1 user, load average: 0.01, 0.11, 0.07
Tasks: 61 total, 1 running, 60 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.0%us, 0.7%sy, 0.0%ni, 99.0%id, 0.0%wa, 0.0%hi, 0.3%si, 0.0%st
Mem: 515060k total, 216660k used, 298400k free, 21248k buffers
Swap: 0k total, 0k used, 0k free, 98072k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
4	root	15	-5	0	0	0	S	3.3	0.0	0:02.00	ksoftirqd/0
1	root	20	0	2088	1208	672	S	0.0	0.2	0:02.72	upstart
2	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	RT	-5	0	0	0	S	0.0	0.0	0:00.00	migration/0
5	root	15	-5	0	0	0	S	0.0	0.0	0:00.16	events/0
6	root	15	-5	0	0	0	S	0.0	0.0	0:00.04	khelper
64	root	15	-5	0	0	0	S	0.0	0.0	0:00.40	kblockd/0
66	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	kacpid
67	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	kacpi_notify
106	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	cqueue
110	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	ksuspend_usbd
116	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	khubd
119	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	kseriod
155	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pdflush
156	root	20	0	0	0	0	S	0.0	0.0	0:00.16	pdflush
157	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	kswapd0
158	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	aio/0

Arquitectura

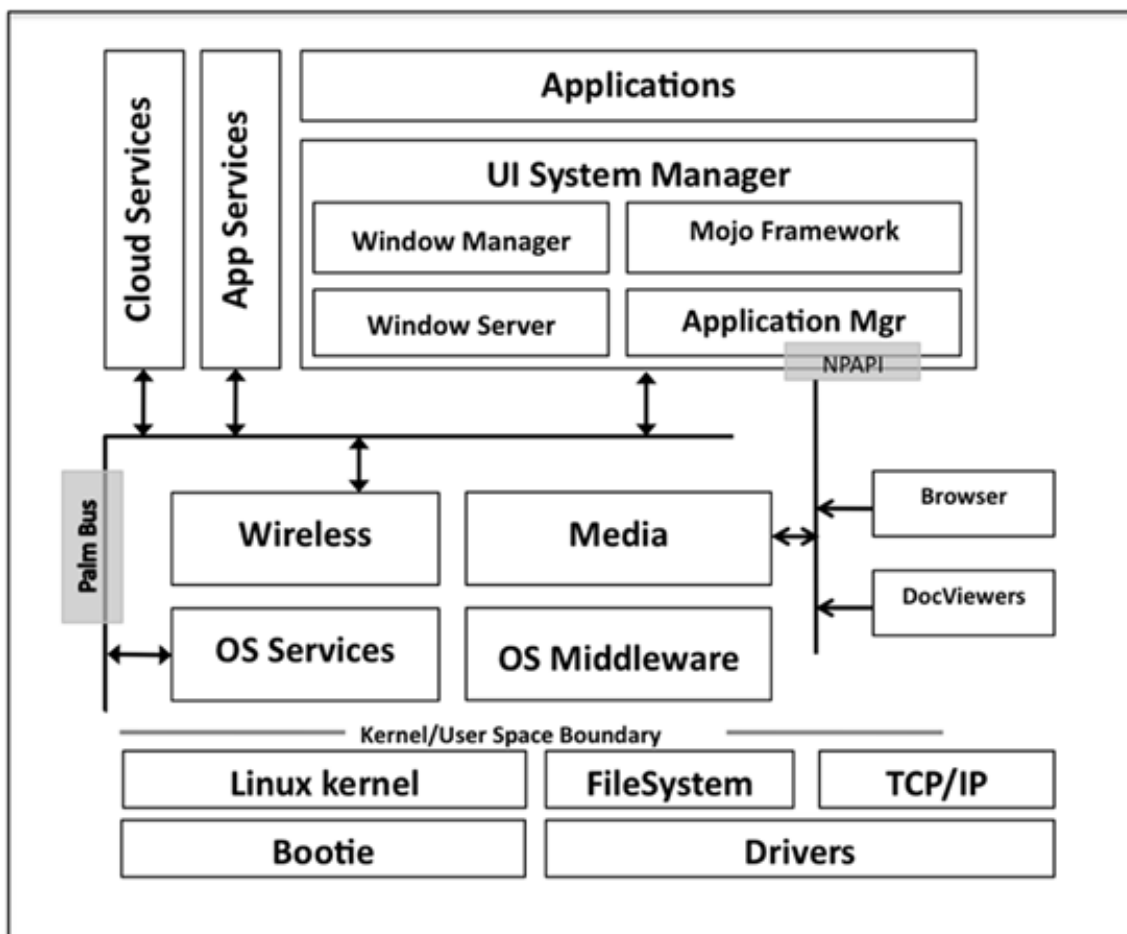


Fig 29. Diagrama de l'arquitectura del sistema webOS

El HP webOS només es pot simular en un emulador propi basat en VirtualBox. Es tracta d'una eina de virtualització implementada per Oracle i en 2007 va llençar la versió Open Source.

Com Meego, el sistema inclou un conjunt de binaris molt reduït i no només de GNU, sinó també de BSD. El top mostra un total de 61 processos amb només 1 executant-se que és el propi top.

El diagrama de l'arquitectura de webOS és realment interessant perquè mostra una estructura realment innovadora que no s'ha vist en altres sistemes operatius. La integració dels elements del kernel es comunica amb mòduls de diferents nivells a través d'un bus especial, el "Palm Bus". Aquest intercomunica els serveis del sistema operatiu directament amb els serveis en núvol, de les aplicacions i la connexió sense fils. Es pot notar doncs com el sistema concentra els recursos en donar serveis orientats a la xarxa.

6. PROVES

Abans de tot, es detallaran les característiques tècniques de l'entorn on s'executen les proves. El maquinari és un ordinador Samsung R510 amb microprocessador Intel Core 2 Duo P8400 a 2,26 GHz i el sistema operatiu és un Ubuntu Lucid Linx 10.04 de 32 bits.

En aquest entorn amb arquitectura x86, aconseguir emular els 4 sistemes amb les mateixes condicions és una tasca gairebé impossible, ja que no són compatibles totes les plataformes amb un mateix emulador com QEMU o VirtualBox. Aquest fet farà que els resultats estiguin clarament diferenciats segons el simulador, només Android i OPhone podrien ser vàlidament comparats. Tot i així, els resultats són significatius.

Existeixen moltes proves de Benchmark desenvolupades com a projectes en codi obert, es poden trobar per la xarxa paquets que contenen una col·lecció de tests per avaluar sistemes operatius Linux, com per exemple el HBenchmark, lozone, que poden retornar informació sobre l'ús de la memòria, la gestió de la sobrecarrega del sistema, temps d'escriptura de fitxer, temps de resolució d'un algoritme, etc.

Però malauradament els emuladors dificulten considerablement l'ús d'aquests suites de Benchmark, encara que es disposa del codi font, és necessari una adaptació per a cada sistema fent servir mètodes de compilació creuada (o "cross compiling" que permet crear codi executable per una plataforma diferent al sistema on es compila). Les eines de desenvolupament de software que proporciona cada sistema operatiu estan enfocades a implementació de les "Apps" i els procediments de "cross compiling" no estan estipulats oficialment.

Malgrat tot, s'ha optat per fer servir el suite de Benchmark v8 de Google. Donat que es tracta d'una adaptació de proves populars a JavaScript per executar-se mitjançant el navegador web, és compatible amb totes les plataformes. L'inconvenient en aquest cas és que queda condicionat per la eficiència del navegador i aquest suite està dissenyat principalment per testear el JavaScript.

Les limitacions dels emuladors impedeixen a més la instal·lació de programes a menys que es compili el codi font i sigui compatible amb el sistema operatiu, llavors és impossible disposar del mateix navegador en tots els sistemes. Com a referència complementària s'executa la prova en el sistema operatiu Ubuntu.

Llistat de navegadors web

Ubuntu:	Mozilla Firefox 4
Android:	navegador basat en Webkit
Meego:	Mozilla Fennec 4
OPhone:	navegador basat en Webkit
webOS:	navegador basat en Webkit

6.1. DESCRIPCIÓ

V8 Benchmark Suite

El paquet de proves online de Google V8 Benchmark Suite, actualment en versió 6, és una col·lecció de simulacions conegudes adaptades a JavaScript que es pot trobar a la pàgina:

<http://v8.googlecode.com/svn/data/benchmarks/v6/run.html>

Les proves que componen el Benchmark v8 són:

- **Richards**
Una simulació del kernel d'un sistema operatiu, es creen diverses classes de tasques i una sèrie de treballs associats amb cadascuna de les categories, la prova consisteix en mesurar l'eficiència en la gestió d'aquestes línees de treball. El codi va ser escrit en el llenguatge BCPL per Martin Richards en 1981.
- **DeltaBlue**
Algorisme que consisteix en resoldre equacions de restriccions va ser creat en per John Maloney i Mario Wolczko en Smalltalk.
- **Crypto**
Mesura la velocitat de xifratge i dexifratge RSA, va ser dissenyat per Tom Wu.
- **RayTrace**
Benchmark construït a base de l'algorisme concebut per generar imatges tridimensionals.
- **EarleyBoyer**
Creat a partir del llenguatge Scheme, avalua la manipulació d'estructures de dades i d'objectes.
- **RegExp**
Aquest Benchmark executa les 50 expressions regulars extretes de pàgines webs populars.
- **Splay**
Crea una estructura de dades recursiva com són els arbres binaris i comença a manipular les dades per comprovar la gestió de memòria.

El Benchmark retorna puntuacions després de cada passada, aquestes puntuacions quan més elevades són indiquen un bon rendiment. S'executarà per cada sistema i navegador 8 cops el suite de Benchmark i es calcularà una mitjana.

6.2. RESULTATS

Ubuntu Lucid Lynx, 10.04 32 bits

Prova	1 ^a	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	7 ^a	8 ^a	Mitjana
Richards	4835	4961	5031	4598	5020	4629	4678	5062	4851,75
DeltaBlue	3399	3292	3438	3598	3513	3548	3303	3452	3442,875
Crypto	3393	3420	3383	3415	3318	3369	3398	3398	3386,75
RayTrace	3094	3126	3088	3064	3124	3078	3115	3200	3111,125
EarleyBoyer	3405	3429	3404	3390	3311	3362	3324	3368	3374,125
RegExp	1120	1120	1118	1114	1127	1107	1105	1118	1116,125
Splay	4705	4696	4778	4724	4769	4769	4751	4761	4744,125
Global	3153	3161	3180	3151	3175	3142	3116	3196	3159,25

Taula 2. Taula de resultats de Ubuntu

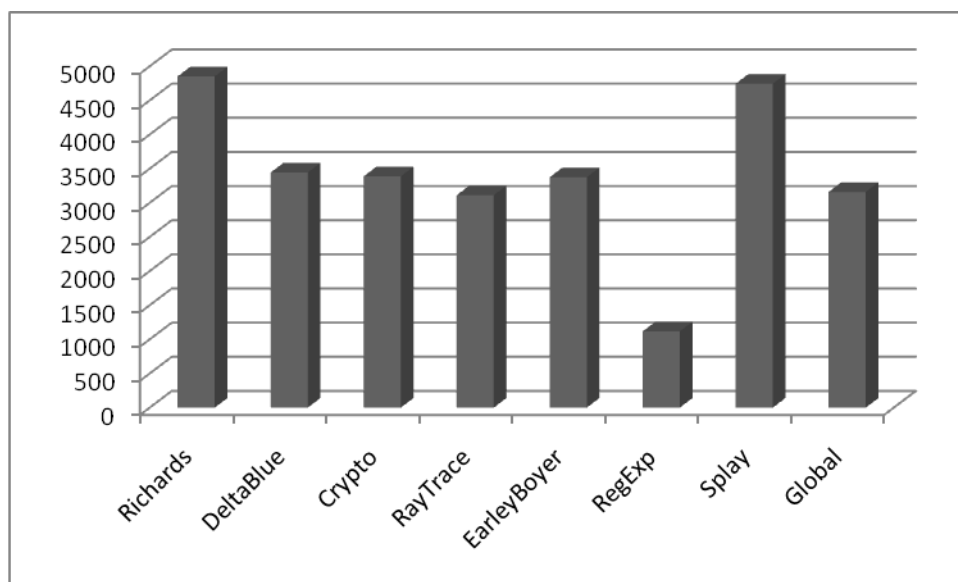


Fig 30. Gràfica de les mitjanes dels resultants de Ubuntu

Aquests són els resultats de les execucions en el sistema operatiu de Ubuntu amb un navegador Mozilla Firefox 4. Destaca el bon rendiment en la simulació de Richards i Splay, en canvi presenta resultats més pobres en la resolució d'expressions regulars.

Android

Prova	1 ^a	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	7 ^a	8 ^a	Mitjana
Richards	13,3	14	16,4	12	10,3	13,4	12	10,4	12,725
DeltaBlue	14	12,8	14,8	13,8	13,6	13,4	14,1	13,9	13,8
Crypto	4,01	4,14	4,07	4,08	3,86	4,14	4,02	4,04	4,045
RayTrace	31	30,4	29,7	31,2	30,1	30,9	31	31,4	30,7125
EarleyBoyer	30,5	30,6	32	30,8	31,4	30,5	31,5	31,1	31,05
RegExp	7,79	7,74	7,75	7,61	7,93	7,73	7,92	7,72	7,77375
Splay	13,3	13,4	80,1	13,4	81,1	13,3	13,6	76,4	32,6
Global	13,3	13,2	17,9	13,1	16,4	13,3	13,2	16,5	14,6125

Taula 3. Taula de resultats de Android

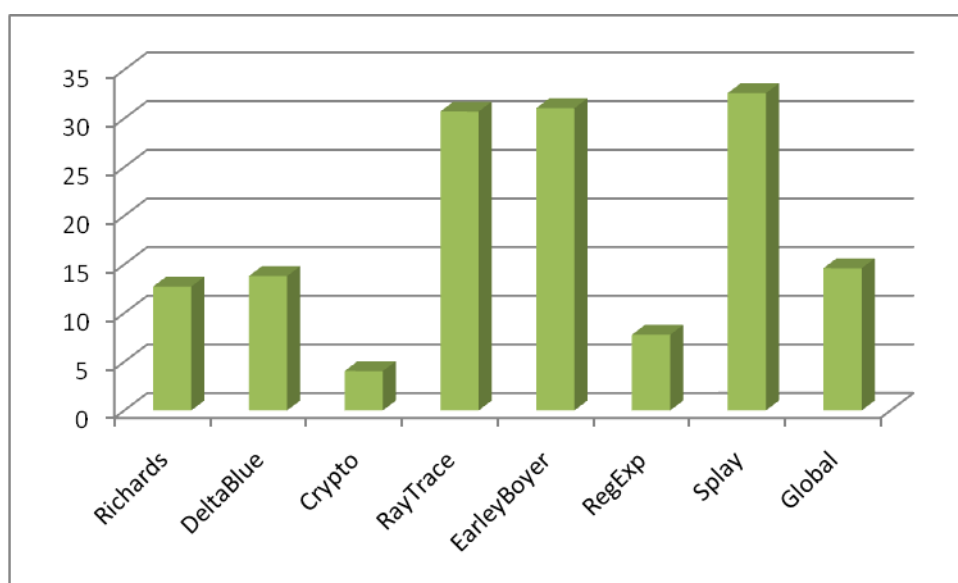


Fig 31. Gràfica de les mitjanes dels resultats de Android

Al igual que Ubuntu, la gestió de memòria és bona a l'hora d'executar Splay i el resultat de RegExp és més feble. Cal destacar que el rendiment amb les execucions de Richards, Deltablue i Crypto baixen significativament.

Meego

Prova	1 ^a	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	7 ^a	8 ^a	Mitjana
Richards	2466	2111	2520	2526	2132	2407	2495	2607	2408
DeltaBlue	988	722	1010	28,1	312	206	1028	1009	662,8875
Crypto	1385	364	307	1367	1305	1311	1344	1322	1088,125
RayTrace	319	355	342	339	364	346	321	327	339,125
EarleyBoyer	197	291	289	290	233	238	139	272	243,625
RegExp	261	97,8	94,2	98,2	255	129	171	92,6	149,85
Splay	616	625	654	572	587	633	380	602	583,625
Global	617	446	466	341	525	460	518	560	491,625

Taula 4. Taula de resultats de Meego

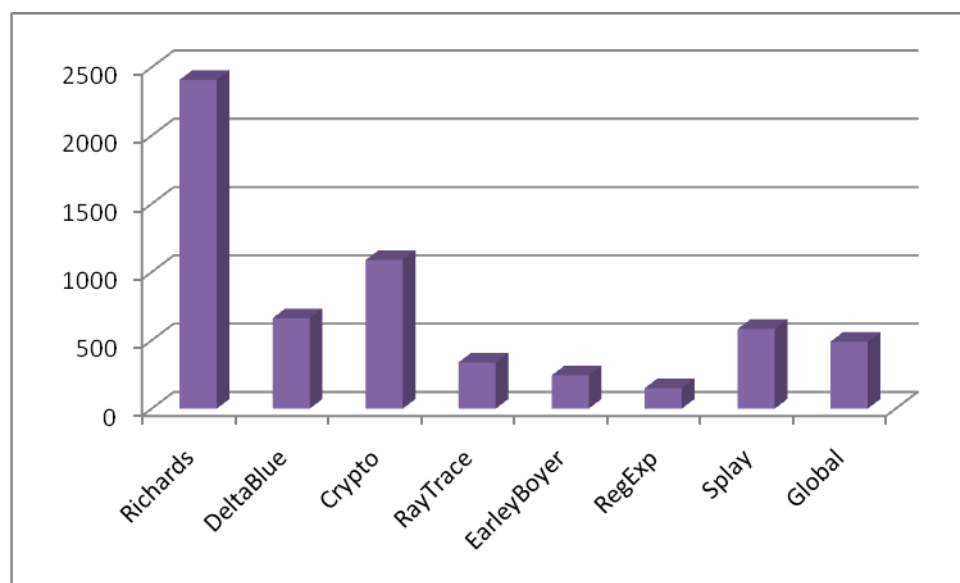


Fig 32. Gràfica de les mitjanes dels resultats de MeeGo

El sistema MeeGo amb el navegador per a dispositius mòbils de Mozilla Fennec 4, que utilitza el mateix motor que el Firefox 4, excedeix en el resultat de Richards i Crypto. La resta de simulacions obtenen

OPhone

Prova	1 ^a	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	7 ^a	8 ^a	Mitjana
Richards	5,51	2,25	2,85	5,61	2,02	5,12	3,3	5,29	3,99375
DeltaBlue	6,83	5,21	6,58	4,06	7,01	4	5,98	6,06	5,71625
Crypto	?	?	?	?	?	?	?	?	?
RayTrace	?	?	?	?	?	?	?	?	?
EarleyBoyer	?	?	?	?	?	?	?	?	?
RegExp	?	?	?	?	?	?	?	?	?
Splay	?	?	?	?	?	?	?	?	?
Global	?	?	?	?	?	?	?	?	?

Taula 5. Taula de resultats de OPhone

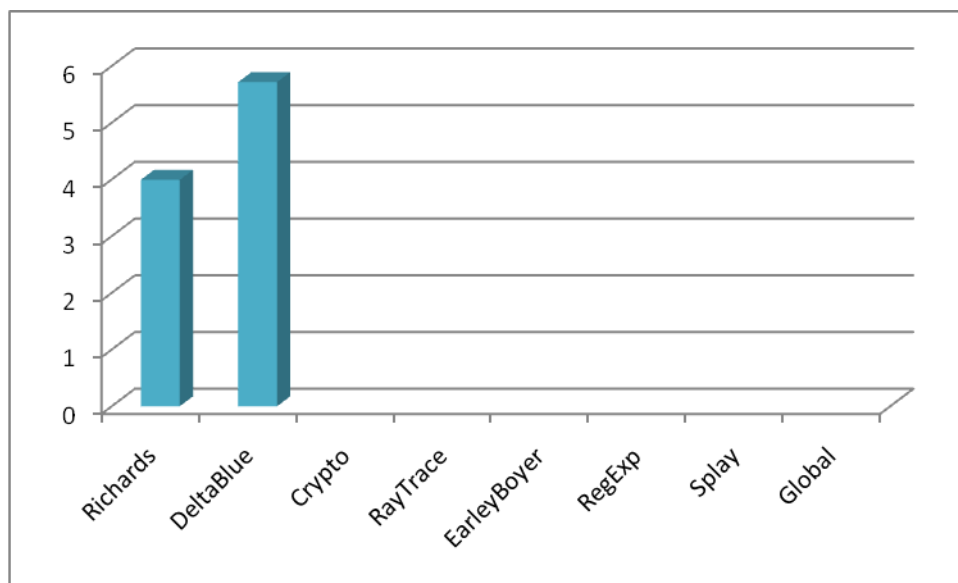


Fig 33. Gràfica de les mitjanes dels resultants de OPhone

El OPhone seguint els passos de Android, treu una puntuació lleugerament millor en la prova de DeltaBlue que la de Richards. No hi ha resultats de les altres proves perquè el navegador basat en WebKit de OPhone no ha sigut capaç de suportar les altres proves.

webOS

Prova	1 ^a	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	7 ^a	8 ^a	Mitjana
Richards	3789	3928	3938	3928	3900	4078	2602	3844	3750,875
DeltaBlue	2607	4714	4773	4745	4852	4529	2652	4821	4211,625
Crypto	3075	2880	3264	3274	3252	3269	3132	3272	3177,25
RayTrace	1189	6175	5759	6095	5870	4861	1419	5898	4658,25
EarleyBoyer	8735	9440	9267	9222	9563	8710	1893	9481	8288,875
RegExp	2010	2149	2122	2144	2118	1934	853	2138	1933,5
Splay	5802	7435	7128	6766	7161	6331	4403	7377	6550,375
Global	3231	4686	4685	4688	4723	4388	2151	3844	4049,5

Taula 6. Taula de resultats de webOS

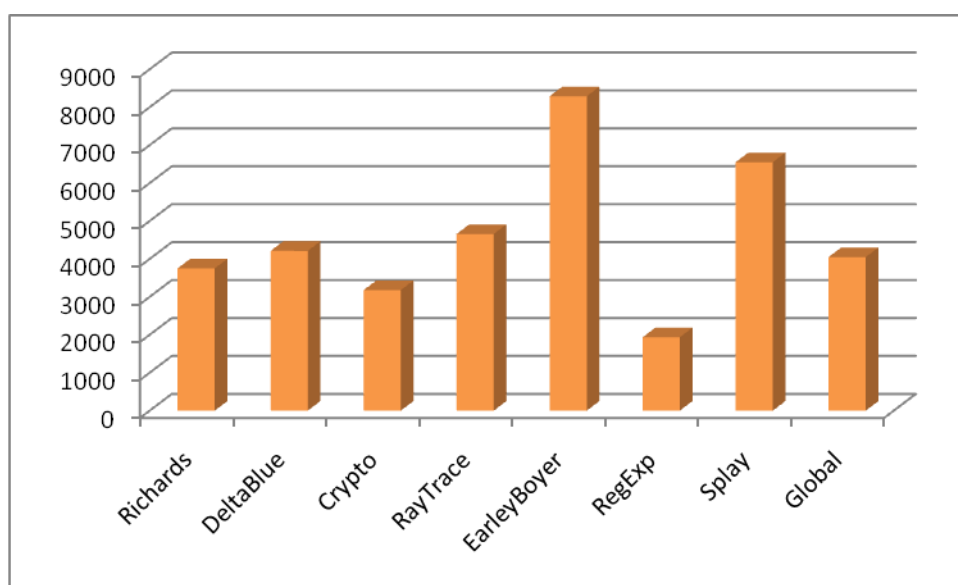


Fig 34. Gràfica de les mitjanes dels resultats de webOS

En webOS, els resultats de EarleyBoyer i SPlay són excel·lents i demostren que la gestió de estructures de dades i de la memòria són els punts forts d'aquest sistema amb navegador integrat. És important recalcar que, encara que webOS estigui emulat per VirtualBox, les seves puntuacions superen de llarg les del sistema operatiu de base, Ubuntu.

7. COMPARATIVA

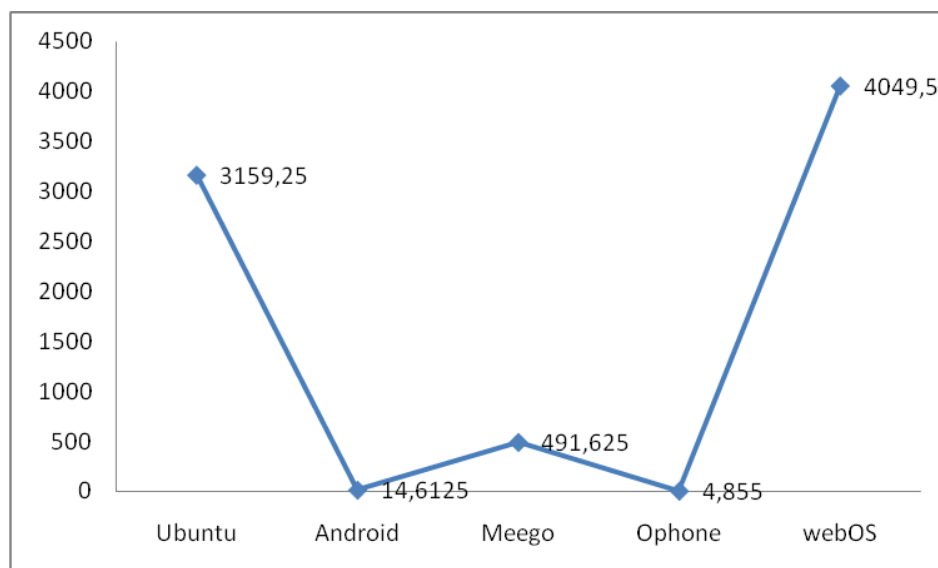


Fig 35. Gràfica comparativa entre els sistemes

Considerant que múltiples factors com els diferents emuladors i navegadors webs influeixen sense dubte en el rendiment d'aquestes puntuacions, es pot observar igualment aspectes interessants com el cas de l'OPhone, el sistema operatiu a partir de l'Android, no hagi pogut passar el joc de proves sencer, i de les simulacions que ha pogut passar, Richards i DeltaBlue, els resultats són inferiors als de Android.

MeeGo presenta valors relativament millors que els de Android i OPhone degut al navegador web Mozilla Fennec i l'emulador dissenyat inicialment per Maemo. La relació de les puntuacions denota un bon comportament en la simulació de kernel.

En canvi, webOS sorprèn amb unes xifres que superen fins i tot les de Ubuntu, aquestes dades posa en evidència dos fets: la virtualització eficient del Palm Emulator basat en VirtualBox i també l'arquitectura extremadament orientada a la web de webOS.

	Android	MeeGo	Ophone	webOS
Desenvolupador	Open Handset Alliance (Google)	Linux Foundation	Open Mobile System	HP
Versió	2.3.3	1.1.2	2.0	2.1
Família de S.O.	Linux	Linux	Linux	Linux
Arquitectura suportada	ARM, MIPS, x86	ARM, x86	ARM	ARM
Programat en	C, C++, Java	C++	C, C++, Java	C
Llicència	Apache v.2	Multi llicència open source, GPL, BSD	Apache v.2, GNU GPL v.2	Propietària amb components de codi obert

Taula 7. Taula resum dels sistemes comparats

8. CONCLUSIONS

Tot i que les proves de Benchmarking no són del tot consistents, els resultats mostren la gran variabilitat en el rendiment dels sistemes operatius i en els seus components com el navegador web. A més del gran impacte que poden tenir els mètodes de virtualització en el rendiment general d'un sistema.

Per les dades de mercat, les estadístiques més recents i les prediccions que s'ha recopilat, tot indica que l'actual tendència del sistema operatiu Android anirà en augment en el nombre de terminals mòbils. Al menys en el panorama occidental, donat que l'aparició del OPhone a China i la seva ràpida comercialització i desenvolupament pot invertir la situació favorable de l'Android en aquest país que es troba en tensió amb Google, la principal responsable del sistema.

Els altres sistemes com MeeGo i webOS tindran segurament un lloc important en el mercat, són projectes sòlids i competents, un en codi obert i l'altre tancat però amb gran compatibilitat amb open source. Aquestes plataformes són candidates a succeir els sistemes ja quasi extints Symbian i Palm.

Cal destacar que el sistema webOS, encara que estigués emulat, ha pogut donar els millors resultats en les proves de V8 superant fins i tot els del propi sistema operatiu Ubuntu. Això pot ser un clar indicador de la eficiència del HP webOS derivat d'un disseny considerablement innovador i enfocat a serveis en xarxa on un component com el navegador web té una ponderació més important.

8.1. TREBALL ASSOLIT

El principal treball assolit en aquest projecte és la introducció als conceptes de software lliure i codi obert, així com les nombroses llicències que estipulen les condicions d'ús del codi. S'ha assolit la presentació dels sistemes operatius basats en GNU/Linux actualment més importants i també s'ha analitzat les tendències d'aquests sistemes en els propers anys. A més s'ha efectuat la emulació seguit d'un petit estudi dels 4 sistemes operatius escollits.

L'objectiu de comparar les eficiències dels sistemes no està del tot complert, especialment perquè la emulació en una plataforma x86 de sistemes operatius dissenyats específicament per plataformes ARM dificulta exponencialment la maniobrabilitat. Les proves de Benchmark basats en V8 no són les proves més idònies per capturar dades concloents i tampoc abasta el ventall de informació extraïble d'un sistema. Per exemple un dels objectius de mesura era el temps d'arrencada, que no s'ha pogut assolir, al igual que la recopilació de informació sobre el rendiment de I/O no s'ha efectuat.

8.2. TREBALLS FUTURS

Els treballs futurs podrien englobar mètodes que millorin la comparativa, a més de realitzar-les sobre versions més recents. És primordial arribar a emular els diferents sistemes operatius en les mateixes condicions. Si no és emulat, aconseguir un maquinari ARM on instal·lar cada distribució.

Serà necessari igualment el desenvolupament o bé l'adaptació d'alguna manera dels programes de Benchmark per valorar el rendiment. Ja que els sistemes operatius mòbils no disposen de compiladors, s'hauria de fer servir compilació creuada o bé recompilar tot el sistema juntament amb els binaris. D'aquesta manera es podria integrar eines de mesura de rendiment com el Bootchart, per analitzar els processos durant l'inici del sistema o suites de Benchmark més importants com iOzone, HBench o Linpack.

El principal impediment per que es pugui realitzar aquest procediment és la disposició del codi font, no tots els sistemes l'ofereixen encara que estiguin basats en GNU/Linux, com per exemple el HP webOS.

8.3. VALORACIÓ PERSONAL

El resultat d'aquest projecte personalment considero que no és tan satisfactori com m'esperava, davant la impossibilitat de realitzar proves en condicions, la comparativa en rendiment no és prou consistent. Les hores invertides en investigar alguna solució viable per pal·liar la problemàtica de la diversificació d'emuladors i la integració de Benchmarks que satisfaci les condicions de tots els sistemes han sigut infructíferes.

No obstant aquest treball m'ha permès aprofundir en l'actualitat dels projectes de sistemes operatius lliures, que es presenten com a alternatives molt atractives i tenen davant un terreny molt fèrtil pel desenvolupament. Certament provar les 4 distribucions seleccionades i conèixer les noves arquitectures dels sistemes ha resultat de lo més interessant.

9. REFERÈNCIES

Llibres:

Mikko Välimäki, "The Rise of Open Source Licensing" (2005)

Andrew M. St. Laurent, "Understanding Open Source and Free Software Licensing" (2004)

Pàgines webs:

Android Developers, <http://developer.android.com/>

Bada Developers, <http://developer.bada.com>

GNU Operating System, <http://www.gnu.org/>

LiMo Foundation, <http://www.limofoundation.org>

LinuxDevices.com, <http://www.linuxfordevices.com>

Maemo, <http://maemo.org>

MeeGo, <http://meego.com>

Moblin, <http://moblin.org>

Open Handset Alliance, <http://www.openhandsetalliance.com>

OPhone SDN, <http://en.ophonesdn.com>

Palm Developers, <https://developer.palm.com>

Search Mobile Computing, <http://searchmobilecomputing.techtarget.com>

Stat Counter Global Stats, <http://gs.statcounter.com>

10. ANNEX

Instal·lació del SDK de Android

Primer de tot és necessari disposar del entorn de Java en el nostre sistema per tal de poder instal·lar i fer servir el SDK de Android, optar per una versió lliure com el openjdk és possible també i per exemple es pot obtenir amb la comanda següent:

```
$ sudo apt-get install openjdk-6-jdk
```

Acte seguit s'ha de descarregar l'arxiu tar amb el SDK de Android en l'adreça <http://developer.android.com/>

Descomprimir en la ruta desitjada i llavors, afegir-la al PATH del sistema editant l'arxiu /root/bashrc

Incloure la línia següent:

```
export PATH=$PATH:<ruta de la carpeta android>/tools/
```

Es reinicia la consola per recarregar el \$PATH, es pot comprovar la correcta introducció de la ruta de la carpeta android fent echo \$PATH. Si està present en la línia resultant, llavors es pot executar

```
$ android
```

S'obrirà el gestor de dispositius virtuals Android, AVD, on l'usuari disposa de les eines necessàries per crear, modificar i eliminar els dispositius que vulgui. A la pestanya "Installed packages" es pot procedir a una actualització per assegurar-se de tenir les últimes versions. També es pot fer mitjançant:

```
$ android update sdk
```

En el mateix gestor AVD es crea un dispositiu virtual per posteriorment emular-lo, en el cas d'aquest projecte s'utilitza la versió 2.3.3 de Android. Aquest pas es pot realitzar a través de la comanda:

```
$ android create avd -n <nom> -t <numero de target>
```

El nom és lliure però el numero de target s'obté amb:

```
$ android list target
```

Un cop creat ja es pot emular el dispositiu virtual:

```
$ emulator -avd <nom>
```

Instal·lació del SDK de MeeGo

Per instal·lar el SDK de MeeGo en una distribució Ubuntu, s'ha de crear l'arxiu `/etc/apt/sources.list.d/meego-sdk.list` amb la línia següent (substituint (distribution) pel nom de la distribució i (version) pel número de versió del sistema operatiu en ús :

```
deb http://repo.meego.com/MeeGo/sdk/host/repos/${distribution}/${version}/ /
```

Després s'afegeix les claus públiques al repositori

```
$ gpg --keyserver pgpkeys.mit.edu --recv 0BC7BEC479FC1F8A
```

```
$ gpg --export --armor 0BC7BEC479FC1F8A | sudo apt-key add -
```

S'actualitza aptitude amb:

```
$ sudo aptitude update
```

Es pot comprovar que el repositori de MeeGo s'ha afegit correctament:

```
$ apt-cache policy madde
```

En cas afirmatiu es procedeix a instal·lar el SDK:

```
$ sudo aptitude install meego-sdk
```

Per llistar els “runtimes” que es poden instal·lar es fa:

```
$ sudo mad-admin list runtimes
```

Aquest projecte utilitza el “runtime” `meego-handset-ia32-qemu-1.1.2-runtime`, per instal·larlo s'introdueix:

```
$ sudo mad-admin create -f -e meego-handset-ia32-qemu-1.1.2-runtime
```

Un cop instal·lat, l'emulador és executable amb la comanda:

```
$ sudo mad remote -r meego-handset-ia32-qemu-1.1.2-runtime poweron
```

Instal·lació del SDK de OPhone

El primer pas és afegir la ruta cap al directori del SDK de Android al arxiu `/root/.bashrc`:

```
export ANDROID_HOME="<ruta SDK Android>"
```

Després es descarrega via web el paquet SDK de OPhone a la pàgina <http://www.ophonesdn.com> i descomprimir el contingut en el directori “add-on” que es troba dins de la carpeta principal del SDK de Android

Lavors es crea el dispositiu virtual OPhone seguint els mateixos passos en el gestor AVD. Seleccionar China Mobile:OPhone 2.0 i crear el dispositiu virtual.

Ja es pot emular OPhone mitjançant:

```
$ emulator -avd OPhone (és recomanable fer servir el parametre -scale 0.5)
```

Instal·lació del SDK de webOS

Editar `/etc/apt/sources.list` i afegir:

```
deb http://download.virtualbox.org/virtualbox/debian lucid non-free
```

Després executar:

```
$ wget -q http://download.virtualbox.org/virtualbox/debian/oracle\_vbox.asc -O-  
| sudo apt-key add -
```

i llavors:

```
$ sudo apt-get update
```

Instal·lar VirtualBox:

```
$ sudo apt-get install virtualbox-3.2
```

Descarregar els paquets novacom i sdk des de https://developer.palm.com/content/resources/develop/sdk_pdk_download.html i instal·lar:

```
$ sudo dpkg -i palm-sdk_2.1.0-svn409992-pho519_i386.deb  
$ sudo dpkg -i palm-novacom_1.0.56_i386.deb
```

Ja es pot emular el webOS amb

```
$ palm-emulator
```