



Androidene

Sluttrapport

INF5261

Jon Lønne og Øyvind Marthinsen

7. mai 2008



Androidene

Innholdsfortegnelse

Introduksjon	2
Forord	2
Bakgrunn	2
Problemområde	2
Problemstilling	3
Metode	4
Verktøy	4
Teori	4
Introduksjon til Android	4
Synopsis	4
Hvem står bak	4
Lisensiering	5
Teknisk introduksjon til Android	5
Innledning	5
Arkitektur	5
Utviklingsverktøy	6
Applikasjonenes livssykel	6
Oppsummering	7
Case	8
Idé	8
Brukergrensesnitt	8
Litt teknisk	10
Funn	11
Bruk av utviklingsverktøy	11
Åpenhet	11
Inkompabilitet	12
Medieoppmerksomhet	13
Popularitet	13
Uttalelser fra konkurrenter	14
Android Developer Challenge	15
Diskusjon	16
Bruk av utviklingsverktøy	16
Åpenhet	16
Inkompabilitet	16
Medieoppmerksomhet	17
Konklusjon	18
Litteraturliste	19



Introduksjon

Forord

I vår prosjektoppgave ønsker vi å utforske Googles mobilsatsing Android. Android er en mobilplattform som har som mål å gjøre det enklere for utviklere å lage applikasjoner til mobile enheter. Vi ønsker å lære mer om Android gjennom å utvikle en prototype på plattformen: en handleliste for mobiltelefoner. Vi tror en handleliste-applikasjon vil være en nyttig applikasjon for mange familier, samtidig ligger det store utfordringer i å designe den.

Bakgrunn

Gruppen består av to masterstudenter i informatikk. Vi skal skrive masteroppgave sammen, innenfor mobile systemer. Vårt utgangspunkt for oppgaven er å se nærmere på mobile plattformer. Dette er en av hovedgrunnene til at vi i dette prosjektet har valgt å titte nærmere på Android.

Problemområde

Alle datamaskiner benytter seg av et operativsystem (OS), som har ansvar for og fordeler maskinens resurser til programmene som kjører på maskinen. Etterhvert som mobiltelefoner har utviklet seg til mer avanserte datamaskiner og har fått flere bruksområder, har vi fått et behov for kraftigere og mer avanserte operativsystemer.

I dag er det kun de enkleste mobiltelefonene som kjører på operativsystemer fra håndsettutvikleren selv; de fleste nye telefoner i dag benytter seg av operativsystemer utviklet av andre selskaper. I 2006 kom 67% av solgte smarttelefoner med operativsystem fra Symbian^[1], et britisk selskap hvor Nokia har nesten 50% eierandel, etterfulgt av Ericsson, Sony Ericsson og Panasonic, med fler^[2]. Microsoft, med sin mobile utgave av Windows, og RIM (som utvikler BlackBerry) hadde henholdsvis markedsandeler på 13% og 10% i 2007^[1]. Disse tre selskapene, som tilbyr proprietære operativsystemer til mobiltelefoner, hadde tilsammen 90% av markedet i 2007. Det finnes i dag flere selskaper som utvikler Linux-baserte mobilplattformer med åpen kildekode, deriblant Openmoko og norske Trolltech, men disse er i en tidlig fase og har ikke betydelige markedsandeler per i dag.

Googles inntog på den mobile arenaen har fått mye oppmerksomhet den siste tiden^[3]. I dette prosjektet ønsker vi å se nærmere på deres mobile plattform Android, og se hva som ligger bak all oppmerksomheten. Det eksisterer allerede liknende prosjekter, som Openmoko og norske Trolltechs Qtopia Phone Edition (Greenphone), som er åpen kildekode-baserte



Androidene

mobilplattformer. Må det en stor aktør som Google for å få nok trykk bak en åpen mobilplattform? Operatører har begynt å merke ulempene med så mange forskjellige operativsystemer, og har nå begynt å sette begrensninger på hvor mange OS'er de ønsker å tilby til sine kunder^[4].

Da det ikke er planlagt Android-drevne telefoner før tidligst høsten 2008^[5], har vi primært valgt å se på Android fra et utviklingsperspektiv.

Problemstilling

Vi skal se på Google Android-plattformen og finne ut hva den tilbyr, hvordan den er å utvikle på og om den er så åpen som Google skal ha det til.

I arbeid med dette prosjektet ønsker vi å:

- ▶ Gjøre en gjennomgang av artikler som omhandler Googles mobilsatsing.
- ▶ Utvikle en applikasjon for å få et inntrykk av hvordan Android er å jobbe med som plattform. Vi vil også sette denne opp mot noen mobilitetsbegrep hentet fra pensum.
- ▶ Sammenligne hvordan Android er i forhold til eksisterende mobilplattformer.
- ▶ Finne ut av hvor åpen plattformen er.
- ▶ Se på Android i media.



Metode

For å finne ut hva Google har å by på med Android, har vi tenkt til å gjøre et dykk i ressursene de har offentliggjort så langt. Vi har tenkt til å se på SDK'en, lese dokumentasjon og artikler som omhandler Googles inntog i den mobile verden. I tillegg har vi valgt å utvikle en prototypeapplikasjon. Gjennom arbeid med prototypen ønsker vi å oppnå et dypere innblikk av Androids tekniske muligheter og begrensninger.

Verktøy

Til utarbeidelse av denne oppgaven har vi benyttet oss av en wiki til å skrive rapporter, referater fra møter og notater fra funn i artikler, med mer. Med en wiki har vi mulighet til å kollaborere på tekst, og får samlet alt relevant innhold rundt prosjektet på et sted.

I arbeidet med prototypen har vi benyttet oss av et versjonskontrollsystem, kalt [Subversion](#). Bruk av dette verktøyet gjør det enklere å programmere sammen, da man kan jobbe på de samme filene samtidig.

Teori

Introduksjon til Android

Synopsis

Android er en mobilplattform utviklet av Google med støtte fra Open Handset Alliance. Den er basert på Linux, og vil tilgjengeliggjøres under en åpen kildekode-lisens. Android består av et operativsystem, mellomvare og tilbyr nøkkelapplikasjoner som telefonfunksjonalitet, kontaktliste og kart. Telefonens kjernefunksjonalitet er stilt på lik linje med applikasjoner utviklet av tredjeparter, og er med andre ord ment som å være en åpen plattform. Plattformen skal også inkludere alt håndsett-utviklere og operatører trenger for å kunne tilby en telefon på markedet.

Hvem står bak

Open Handset Alliance består av selskaper fra forskjellige markeder. Foreløpige medlemmer som produserer håndsett er HTC, LG, Motorola og Samsung, og maskinvareleverandører som Broadcom, Intel, nVidia og Qualcomm. På operatørsiden er det store navn som NTT DoCoMo, Sprint Nextel og T-Mobile med i alliansen. Siden plattformen skal være åpen, har



Androidene

alle muligheter for å utvikle applikasjoner på plattformen, men det er også flere software-selskaper med i alliansen, som eBay og naturligvis Google.^[6]

Lisensiering

Google har gitt uttrykk for at de ønsker å gi ut Android under Apache 2.0-lisensen^[7], som er en åpen kildekode-lisens vennlig for kommersielle formål. I praksis betyr det at tilbydere av håndsett kan utvikle unike applikasjoner og nyvinninger som de ikke trenger å gi tilbake til prosjektet, som igjen betyr at hver tilbyder kan by på sitt unike innhold - samtidig som man har full tilgang til tredjepartsapplikasjoner som er tilgjengelig for alle håndsett basert på denne plattformen. På denne måten kan håndsett-tilbydere og nettverksoperatører tilby skreddersydde løsninger til sine kunder, uten at dette går utover tilgangen på de åpne applikasjonene.^[8]

Teknisk introduksjon til Android

Innledning

Android-utviklingen er ledet av Google, med bistand fra Open Handset Alliance.

Utviklingsplattformen er for øyeblikket utgitt for å gi utviklere et tidlig innblikk i hva den har å by på. Når plattformen har en mer ferdig form, har Google gitt uttrykk for at de planlegger å gi ut de fleste komponentene under Apache 2.0-lisensen.

Når vi først leser om Android er det lett å tro at det bare er et operativsystem, men Android inneholder også mellomvare og nøkkelapplikasjoner. Det er gitt ut et sett med utviklingsverktøy for å hjelpe utviklingen av Android-applikasjoner. Alle Android-applikasjoner blir skrevet i programmeringsspråket Java. I tillegg er det lagt opp til bruk av XML, blant annet til konfigurasjon og for å forme layouten.

Vi har brukt Googles offisielle Android-sider som basis for våre undersøkelser av plattformen. Sidene er tilgjengelig på <http://code.google.com/android/>.

Arkitektur

Som tidligere nevnt består Android av et operativsystem, mellomvare og noen nøkkelapplikasjoner. Android sin arkitektur kan deles inn i fem deler: applikasjoner, applikasjonsrammeverk, bibliotek, Android kjøretid og Linux-kjernen.^[9]

Android kommer med en rekke essensielle applikasjoner for en moderne mobiltelefon: e-postklient, kalender, kontaktliste, kart og en nettleser. Tredjepartsapplikasjonene vil ligge under den samme delen av arkitekturen, og er altså likestilt med nøkkelapplikasjonene som følger med plattformen.



Androidene

Android-arkitekturen inneholder et rammeverk for applikasjoner. De sentrale applikasjonene fra Google, nevnt ovenfor, benytter seg også av dette rammeverket. Målet med dette laget i arkitekturen er å gjøre det enklere å lage applikasjoner. Rammeverket tilbyr alle applikasjoner å dele sin funksjonalitet med andre programmer. I tillegg til dette ligger det en rekke tjenester og systemer under alle applikasjoner. Et eksempel er hendelsehandleren som gir alle applikasjoner mulighet til å vise advarsler om hendelser i et statusfelt.

Android inkluderer også flere biblioteker, som er tilgjengelig gjennom applikasjonsrammeverket. Et par eksempler her er SQLite, en lettvekts databasemotor og Media-bibliotek som støtter avspilling flere populære lyd- og videoformater.

Android kjøretid inkluderer et kjernebibliotek som inneholder det meste av funksjonaliteten man har tilgjengelig i programmeringsspråket Java. Hver Android-applikasjon blir kjørt i sin egen prosess og som en egen instans i en Dalvik virtuell maskin, som er optimisert for minnebegrensningene Android-telefoner har.^[9] Det at Android har sin egen virtuelle maskin, gjør at applikasjoner skrevet for Android ikke er ikke kompatible med Suns virtuelle maskin og deres Java ME-plattform (mer om dette senere i oppgaven).

Android er avhengig av Linux-kjerne versjon 2.6 for kjernefunksjonalitet som sikkerhet, minnehåndtering, prosesshåndtering, nettverkslag og drivere.^[9]

Utviklingsverktøy

Android sitt sett med utviklingsverktøy inneholder flere verktøy som hjelper deg å utvikle mobile applikasjoner på Android-plattformen. De to viktigste er Android-emulatoren som lar deg teste applikasjonen på en virtuell mobil og den andre er en plugin til utviklingsverktøyet Eclipse. I tillegg til disse to finnes det en rekke programmer for å debugge, pakke og installere applikasjonen på emulatoren.

Applikasjonenes livssykel

En applikasjons livssykel er en sentral og viktig del av Android-plattformen^[10]. Et viktig element å få fram her er at hver applikasjon i Android kjøres i en egen prosess. Dog dukker det opp en uvanlig egenskap her: applikasjonen har ikke selv direkte kontroll over dens egen levetid. Dette blir bestemt av systemet på bakgrunn av hvilke deler av applikasjonen som kjøres, hvor viktig disse delene er for brukeren og hvor mye minne som er tilgjengelig i systemet.

Siden systemet kan drepe applikasjoner når det begynner å gå tomt for minne, er det svært viktig at utviklere vet hvordan de skal bruke applikasjons komponenter som virker inn på



prosessens levetid. For å avgjøre hvilke prosesser som skal drepes blir de lagt i et hierarki basert på hvor viktig prosessen er. Hierarkiet delt inn i fem forskjellige nivåer:

1. Forgrunnprosess – nødvendig for det brukeren gjør nå.
2. Synlig prosess – inneholder en aktivitet som er synlig for brukeren på skjermen, men ikke i forgrunnen.
3. Tjenesteprosess – inneholder en tjeneste. Tjenesteprosseser er vanligvis ikke synlige, men de gjør viktige ting for brukeren.
4. Bakgrunnprosess – inneholder en aktivitet som ikke er synlig for brukeren for øyeblikket.
5. En tom prosess – inneholder ingen aktive applikasjonskomponenter. Eneste grunnen for å ta vare på en slik prosess, er for å korte ned starttiden neste gang en komponent i applikasjon skal kjøres.

Oppsummering

Android-plattform er støttet opp av Open Handset Alliance, som består av flere store navn innen IT-bransjen. Google har gitt uttrykk for at plattformen skal utgis under en åpen kildekode-lisens. Android har et tradisjonelt oppsett som utviklere er kjent med fra før og vil ha mulighet til å sette seg raskt inn i. Java er et kjent språk for mange, og likheten med Suns Java ME-plattform (som brukes i ordinære Java-baserte mobilapplikasjoner) er mange. En utfordringen utviklere kommer til å støte på er hvordan livssykelten til en Android-applikasjon skal implementeres.

Case



Androidene

For å få en bedre innsikt i hva Android er og hva den har å tilby, satte vi oss ned og skisserte en prototype. Denne prototypen har vi brukt som et case i dette prosjektet.

Idé

Ideen vår går ut på å lage en kollaborativ handleliste, en applikasjon hele familien har tilgang på fra sine mobiltelefoner. Man kan opprette en ny handleliste, og deretter kan alle i familien legge til varer på denne listen. Personen som skal handle den dagen kan bruke handlelisten i butikken og "huke av" ting etterhvert som man legger det i den fysiske handlekurven.

Et annet bruksområde hvor handlelisteapplikasjon kan være et nyttig hjelpemiddel, er for eksempel i helsesektoren: hjelpepleiere handler inn for eldre og uføre personer. I artikkelen "Mobility in Collaboration" ser Paul Luff og Christian Heath på hvordan mobilitet kan brukes i kollaborativt arbeid og kaster et lys over en del utfordringer som oppstår^[1].

Hovedmålet med handlelisteapplikasjon er at den er kollaborativ og enkel å bruke. Den åpner for mange spennende muligheter og kan benytte mange av Androids funksjoner. For eksempel kan man legge til støtte for maler: En handlelistemalen går ut på at brukeren spesifiserer en hverdagsmal som inneholder brød, melk og andre hverdagslige produkter man vanligvis kjøper. Helgemalen kan på en annen side inneholde vin og lørdagsgodt.

Brukergrensesnitt

Etter å ha drodlet rundt handlelisteapplikasjonen, kom vi frem til at det vi aller mest ønsket var å fremstille handlelistene ved hjelp av ikoner for varene. Vi har prøvd å vise dette i den vedlagte utkastet av prototypen (figur #). Tanken er at man skal dra en og en vare ned i handlekurven nederst for å legge til ting i handlelisten. På samme måte ser vi for oss at applikasjonen kan fungere i butikken; varene tas ned i handlekurven etterhvert som de fysisk legges i kurven, og varen blir borte fra listen. På en enkel måte kan man da se hva man har





Androidene

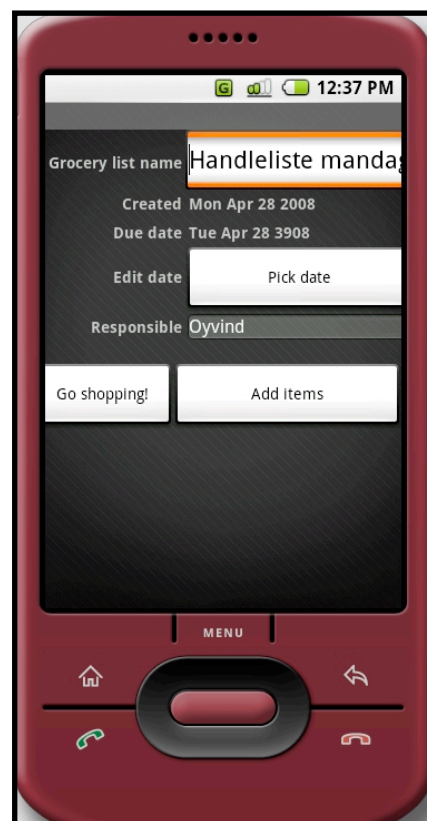
glemt å kjøpe. Vi har lyst til å utforske denne muligheten, da den benytter seg av Androids touch-funksjonalitet.

En av utfordringene med å bruke ikoner til hver vare, er at alle varer må da ha et bilde og være lagt inn i applikasjonen. En måte vi tenkte å løse dette problemet på, er å la brukeren legge inn flere varer ved hjelp av mobilens kamerafunksjonalitet. Dette kan gjøre ting vel avansert for vår prototype, men det er i hvertfall muligheter om vi senere skulle bygge ut funksjonaliteten videre ved et senere tidspunkt. At brukeren legger inn varer kun som tekst kan være en enklere løsning.

En annen mulighet er å bruke lister. Med dette tenker vi et skjermbilde der tingene du skal handle ligger i en liste. Man kan så sette et merke ved de tingene man putter i handlekurven. Dette er en enklere og mer tradisjonell form enn ikoner. Folk er allerede vant til å håndtere lister når de er ute å handler, derav navnet handleliste. Om vi velger å bruke lister i vår applikasjon, ville det sannsynligvis være enklere å lage applikasjon. Dog ville den lide av at den ble mindre visuell attraktiv og muligens ikke like intuitiv. Et annet argument for bruk av lister er at det også ville fungere på eldre mobiltelefoner.

Vi ser også for oss at det hadde vært praktisk for en person som bruker handlelisteapplikasjon å få vite hvilke produkter i handlelisten som er tilgjengelig i det området han/henne er i butikken. For eksempel om han/henne er i den delen av matbutikken hvor meieriprodukter er plassert ville melk og ost bli visst på håndsettet. For å realisere noe slikt måtte en mengde sensorer bli puttet rundt i butikken og de måtte fungere sømløst med den håndholdte enheten. En slik funksjonalitet ville være sterkt knyttet til blant annet mobilitetsbegrepet Ubiquitous Computing^[12].

Siden flere personer skal samarbeide om en liste ville det være en god idé at applikasjon gir beskjed om en person har tatt på seg ansvaret med å kjøpe inn varene. Ved en slik tilnærming vil handlelisteapplikasjonen komme inn på en del problematikk knyttet til Context-Aware Communication. I artikkelen "Context-Aware Communication" bruker Schilit, B., Hilbert D. og Trevor, J en bred definisjon av begrepet og de tar for seg Context-Aware systemer som søker å lette og megle menneske-til-menneske kommunikasjon. De spesifiserer også fem





Androidene

designmål for slike applikasjoner som ville være relevante for oss om vi bestemmer oss for å implementer slik funksjonalitet^[13].

Litt teknisk

En av de største utfordringene er å gjøre handlelisteapplikasjonen kollabrativ. Vi ser for oss at dette kan løses ved å bruke en web service som lagring. Selv ved bruk av web services er det nødvendig å ha en datastruktur i mobilapplikasjonen for mellomlagring (caching), og vi har derfor valgt å begynne med å lage en simpel ikke-kollabrativ prototype i Android. Hvis vi skulle videreutviklet prototypen, har vi lyst til å implementere en webbasert backend for lagring og utveksling av data mellom forskjellige håndsett.

Funn



Bruk av utviklingsverktøy

Vi sitter igjen med en positiv opplevelse av utviklingsmiljøet til Android. For det første var det veldig enkelt å sette opp, takket være gode guider på Androids nettsider^[14]. Tilleggsfunksjonen (pluginen) til Eclipse pakker alle utviklingsverktøyene inn i Eclipse, slik at utvikleren enkelt får tilgang til disse. Utvikleren har også tilgang til alle utviklingsverktøy utenfor Eclipse, men må da sette seg litt mer inn i hvordan de skal brukes.



En sentral del av utviklingsverktøyet er emulatoren (se figur #). Emulatoren gir deg mulighet til å teste applikasjonen din på en virtuell mobiltelefon. Siden den virtuelle telefonen er grafisk, vil du få et inntrykk av hvordan brukerens interaksjon med telefonen vil være. I figur # ser du hvordan en aktivitet i vår handlelisteapplikasjon ser ut. På grunn av begrenset tid til prosjektet, ble casen vår hovedsakelig brukt til å få bedre innsikt i Android-plattformen.

Med Android følger det også med et grafisk program for å feilsøke applikasjoner. Dette programmet er kalt Dalvik Debug Monitor Service (DDMS)^[15]. DDMS inneholder en rekke essensielle og nyttige verktøy, som oversikt over tråder, bruk av minne, logg av hendelser, dumping av skjermbilde, med mer. Kort oppsummert var det veldig greit å komme i gang med utvikling på Android.

Åpenhet

Etter at hurra-stemningen hadde lagt seg i etterkant av Googles annonsering av den åpen kildekode-baserte plattformen Android, har det vært flere som har uttrykt skepsis i forhold til at Android ikke er åpen nok^[16]. Som vi har nevnt tidligere, har Google gitt uttrykk for at de ønsker å gi ut Android-kildekoden under Apache-lisensen, noe som betyr at tilbydere av håndsett kan utvikle lukkede applikasjoner og forbedringer av plattformen - uten å måtte gi dette tilbake til prosjektet. Steven Mosher, VP for markedsføring i OpenMoko (en annen åpen,



Androidene

Linux-basert mobilplattform), har uttalt at "[Android is] using 'open source', but cynically neglecting its principles."^[16]

Mens noen ser på Androids åpenhet med skepsis, er det andre som ser lovende på Googles satsning på åpenhet. Jim Zemlin, sjef i Linux Foundation, påpeker forskjellene på andre, proprietære plattformer som Symbian og Microsoft tilbyr og mener forskjellen ligger i om kildekoden er åpen eller ikke^[16].

Inkompabilitet

Google har gjort noen valg som gjør Android-plattformen og applikasjoner bygget på den inkompatible med eksisterende Java-rammeverk for mobile enheter. For det første har Google valgt å bruke en annen virtuell maskin for eksekvering av Java bytekode, nemlig Dalvik. Google sier at Dalvik er tilpasset telefoner med begrenset minne, og at hver applikasjon har sin egen virtuelle maskin^[9], og derfor bedre egnet på telefoner enn tradisjonelle virtuelle maskiner. Men siden applikasjoner bygget for Android og applikasjoner bygget for ordinære Java-kapable mobiltelefoner benytter to inkompatible virtuelle maskiner, vil det ikke være mulig å benytte programmer utviklet på den ene plattform på den andre.

Rich Green, VP i Suns software-avdeling, sa i etterkant av lanseringen av Android at "Anything that creates a more diverse or fractured platform is not in (developers') best interests."^[17] Han uttrykte bekymring for at Google hadde gått sin egen vei, og ikke benyttet seg av standardbibliotekene som allerede eksisterer for mobile plattformer: "We're reaching out to Google and assuming they'll be reaching out to us to ensure these platforms and APIs will be compatible so deployment on a wide variety of platforms will be possible."^[17]

Suns mobilplattform, Java ME (tidligere J2ME), var i 2006 tilgjengelig på på 1,2 milliarder håndsett^[18].

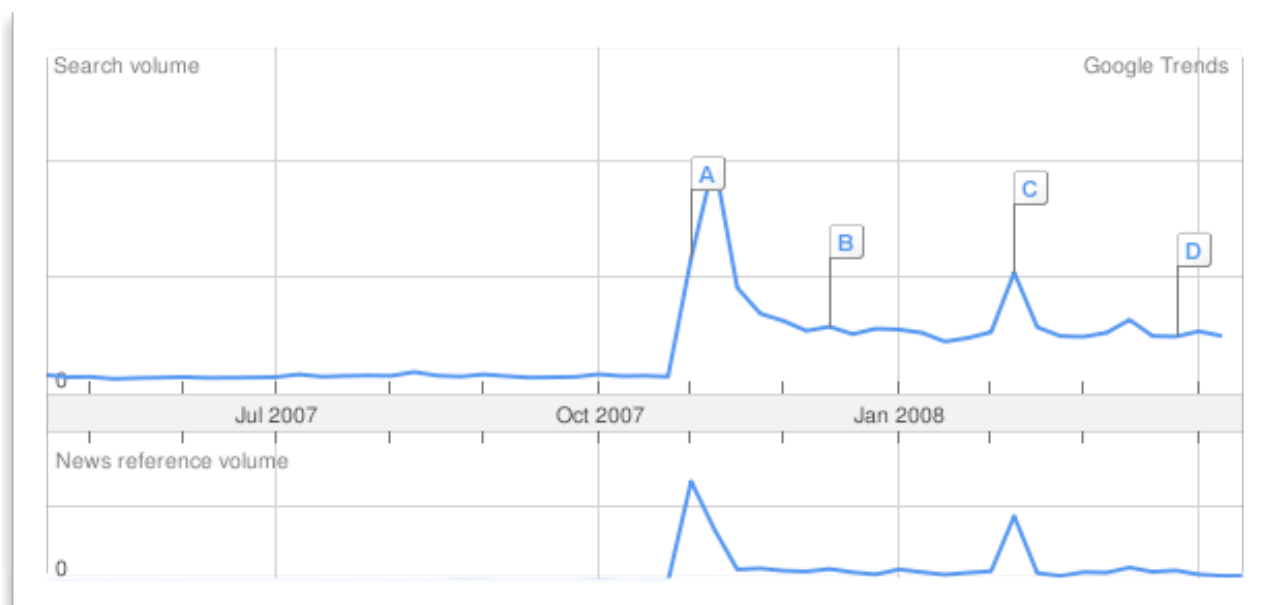


Androidene

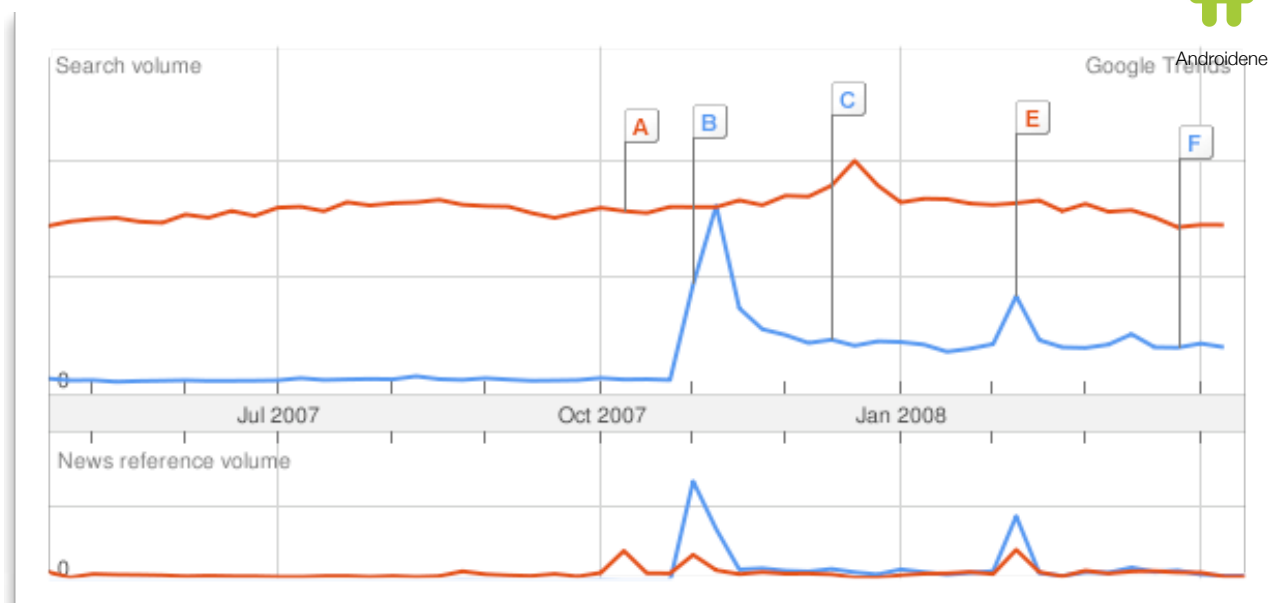
Medieoppmerksomhet

Popularitet

For å finne ut litt mer om Androids popularitet og hyppighet i media, valgte vi å bruke [Google Trends](#). Google Trends er et verktøy for å finne ut hvor ofte et emne er blitt søkt på i Google over et gitt tidsrom. Google Trends viser også hvor mye emnet frekventerer i [Google News](#).



I figuren over ser vi trenden på emnet "android": en stabil graf mot november 2007, før den stiger voldsomt (punkt A). Søk før november 2007 er nok stort sett rettet mot robotdefinisjonen av ordet "android" og ikke Google sin mobilplattform, så dette er viktig å ha i bakhodet når vi analyserer grafen. Grunnen til at grafen går opp i november er Open Handset Alliance sin lansering av Android^[19] den 5. november. Neste kurve på grafen (punkt C) kom under [The Mobile World Congress](#) i Barcelona 11. til 14. februar 2008. Under denne konferansen ble de første Android-baserte telefonene vist frem^[20]. Av grafen får vi også et inntrykk av hvilket nivå populariteten ligger på utenfor periodene med de store lanseringene.



I figuren over ser vi søketrenden på Android opp mot Symbian. Denne grafen gir oss et interessant bilde av Androids popularitet opp mot en av dens største konkurrenter. Vi ser her at søkefrekvensen på "symbian" helt klart overgår "android". I nyhetsbilde er situasjonen en annen, hvor det er omtrent like mange saker om begge plattformer. Vi drøfter dette funnet mer i detalj under diskusjon.

Uttalelser fra konkurrenter

Etter Open Handset Alliance sin lansering av Android, kom det en rekke pressemeldinger fra deres største konkurrenter. Symbians John Forsyth uttalte at "We have been going nine years and have probably seen a dozen new platforms come in and tell us we are under attack. We take it seriously but we are the ones with real phones, real phone platforms and a wealth of volume built up over years."^[21]

Direktøren for markedsføring i Microsoft kom med en lignende uttalelse: "It really sounds that they are getting a whole bunch of people together to build a phone and that's something we've been doing for five years, I don't understand the impact that they are going to have."^[21]

Talsperson i Apple, Natalie Kerris, kom med en mer nøytral uttalelse: "We have a great relationship with Google and this doesn't change anything. They are certainly an important partner for iPhone."^[21]



Androidene

Android Developer Challenge

For å lokke utviklere til Android-plattformen, annonserte Google før årsskifte en utviklerkonkurranse^[22]. Google har lovet å gi ut totalt 10 millioner dollar i premie, som vil bli fordelt på de femti beste applikasjonene som blir sendt inn.



Diskusjon

Bruk av utviklingsverktøy

I kjølevannet av lanseringen av Android, kom det noen kritiske artikler rettet mot plattformen. En gjenganger var at flere mente den ble sluppet for tidlig^[23]. Forbrukerteknologiarkitekten Adam MacBeth sa blant annet "Functionality is not there, is poorly documented or just doesn't work. It's clearly not ready for prime time."^[24] I arbeidet med vår case, var det få av de kritiserte punktene vi kjente oss igjen i. Vi tror at grunnen til dette er at det er kommet flere nye versjoner av SDK'en der de mest åpenbare feil og mangler har blitt rettet opp. Et punkt i Android-plattformen som ble spesielt kritisert, er mangelen på en bugtracker^[23]. Under vårt arbeid med Android oppdaget vi derimot at de nå har fått opp en [bugtracker](#). Etter vår erfaring med plattformen, har vi fått inntrykk av at Google tar tilbakemeldingene seriøst og de har rettet opp mye av det de fikk kritikk for i startfasen.

Åpenhet

Som tidligere nevnt, skriver Google i den nåværende avtalen til SDK'en at de ønsker å gi den ut under Apache-lisensen når den har en mer ferdig form. Men inntil videre, er det Google som har full kontroll over kildekoden til samtlige Android-komponenter. Det er også flere av komponentene som Google ikke har frigitt koden til, noe som har medført at deler av samfunnet rundt Android har følt seg ekskuldert; de mener at de ikke får være med på å forme plattformen. Google har sagt at visse deler av koden er for rotete og undergår store endringer, men at de planlegger å åpne opp så snart de føler den er komplett nok til at den kan kommenteres på.^[25] Det gjenstår å se om Android blir en så åpen plattform som Google vil ha det til, og det blir spennende å se i hvilken grad samfunnet rundt får mulighet til å være med å forme plattformen. Om Googles satsning ikke er åpen nok, er det liten tvil om at den er mer åpen enn hva konkurrerende plattformer som eksempelvis Microsoft og Symbian er per i dag^[16].

Inkompabilitet

Sun har gått ut mot Google og uttrykt bekymring om at Android kommer til å skape en splittelse av utviklere av Java-baserte mobilapplikasjoner. Java ME (tidligere J2ME) har i flere år vært en etablert plattform for utvikling av mobile applikasjoner, og i 2006 hadde åtte av ti håndsett som ble solgt støtte for Java ME^[18]. For utviklere betyr inkompabiliteten mellom Android og Java ME at applikasjonene må utvikles mot én plattform. Med det store antallet



Java ME-kompatible håndsett som allerede finnes, mot Androids null håndsett i salg, vil Java ME-applikasjoner ha flere potensielle sluttbrukere i flere år fremover.

Google har valgt å satse på egenutviklet teknologi, som skal være bedre egnet for mobile enheter enn eksisterende teknologi. Da det ikke har kommet Android-baserte håndsett enda, er det vanskelig å si hvor stor betydning inkompatibilitet med eksisterende applikasjoner og telefoner har å si. Det blir spennende å se hvilke aktører som blir med på Googles satsning, og om dette er starten på en ny standard for mobile plattformer.

Medieoppmerksomhet

I Google Trends-grafen fra funn-avsnittet, satte vi Android opp mot Symbian. Vi så at Symbian er et mer populært søkeord, men at det var mer jevnt i nyhetsbilde. En av grunnene til dette er at Android fortsatt er en nyhet, mens Symbian har eksistert i mange år. Vi tror allikevel at merkenavnet Google har mye å si. Google ble nylig kåret til det sterkeste merkevarenavnet av analysefirmaet Millward Brown^[26]. En uttalelse fra Symbians John Forsyth tyder på at de er av samme oppfatning: "If Google was not involved the industry would have just yawned and rolled over."^[21]

Det er liten tvil om at Google og Open Handset Alliance forsøker å skape publisitet rundt Android. Det godt eksempel på dette er Android Developer Challenge, som skaper presse og samtidig tiltrekker utviklere til plattformen. Lanseringen av Android i en såpass tidlig fase av prosjektet har også ført med seg litt ugunstig publisitet. Det er spesielt mangelen på mobiltelefoner som benytter seg av plattformen som er blitt nevnt som en svakhet, noe Microsofts karismatiske administrerende direktør Steve Ballmer passet på å påpeke: "Well of course their efforts are just some words on paper right now, it's hard to do a very clear comparison [with Windows Mobile]."^[27]

Som vi kan se av uttalelsene i funn-avsnittet, har de konkurrerende selskapene varierte synspunkter på Googles initiativ. En gjenganger er dog at de ikke ser på Android som en stor utfordrer, men dette er nok influert av at det er uttalelser til pressen det er snakk om. Hva som blir sagt bak lukkede dører kan være noe helt annet, og hva som blir sagt der har vi ikke noe grunnlag for å spekulere i.



Androidene

Konklusjon

Googles mobile satsning, Android, er en spennende plattform teknisk sett. Den ser også lovende ut i sin åpenhet, noe som kan vise seg å være et konkurransefortrinn foran de store, lukkede aktørene som er på markedet per i dag. Det finnes plattformer med lignende intensjoner som Android, i OpenMoko og norske Trolltechs GreenPhone, men vi tror det trengs et navn som Google for å få trykk bak en åpen kildekode-basert mobilplattform.

Vi sitter igjen med et inntrykk av at utviklingsverktøyene som er gjort tilgjengelig for Android virker modne. Det var enkelt å komme i gang med utviklingen av prototypen vår, og det var artig å utvikle på plattformen. Vi har også inntrykk av at Google tar tilbakemeldinger fra publikum seriøst, og de har rettet opp mye av det de ble kritisert for når utviklerverktoyene først ble gjort tilgjengelig. Det er ikke alle som er like fornøyd med Googles åpenhet rundt Android enda, men vi mener at denne plattformen er mye mer åpen enn hva eksempelvis Symbian og Microsoft sine plattformer er for øyeblikket. Det blir spennende å følge med på når Google gir ut hele kildekoden til plattformen, slik de har sagt de skal gjøre, og hvordan de tar til seg innspill fra samfunnet rundt fremover.

Android-plattformen er ikke kompatibel med eksisterende Java-rammeverk for mobiltelefoner, noe som var tilgjengelig på 1,2 milliarder håndsett i 2006. For utviklere og operatører betyr dette nok en mobilplattform å forholde seg til, og det blir spennende å se hvilke aktører de får med seg på sin satsning. Da det allerede eksisterer et titalls liknende Linux-baserte mobilplattformer, ønsker vi å avslutte med en uttalelse fra Symbians John Forsyth: "If Google was not involved the industry would have just yawned and rolled over".



Androidene

Litteraturliste

1. Canals (5. februar 2008). [Smart mobile device shipments hit 118 million in 2007, up 53% on 2006](#). Besøkt 6. mai 2008.
2. Symbian. [Symbian Fast Facts Q4 2007](#). Besøkt 6. mai 2008.
3. Zeichick, Alan (desember 2007) «Look Ma, no wires» – [netWorker](#), bind 4, nr. 11, s. 5-8.
4. Sylvers, Eric (12. mars 2007). [As Mobile Phones Grow More Complex, Carriers Insist on Fewer Operating Systems](#). Besøkt 6. mai 2008.
5. German, Kent (24. april 2008). [T-Mobile to carry Android phone by year's end](#). Besøkt 7. mai 2008.
6. Open Handset Alliance. [Open Handset Alliance Members](#). Besøkt 15. mars 2008.
7. Google (12. februar 2008). [Android Software Development Kit License Agreement](#). Besøkt 22. april 2008.
8. Paul, Ryan (6. november 2007). [Why Google chose the Apache Software License over GPLv2 for Android](#). Besøkt 22. april 2008.
9. Google. [What is Android?](#). Besøkt 22. april 2008.
10. Google. [Life Cycle of an Android Application](#). Besøkt 5. mars 2008.
11. Luff, Paul. Heath, Christian (1998) «Mobility in Collaboration» – CSCW/ACM.
12. Weiser; Gold; Brown (1999) «The origins of ubiquitous computing research at PARC in the late 1980s» – IBM SYSTEMS JOURNAL, bind 38, nr. 4.
13. Schilit, B., Hilbert D., Trevor, J. (2002) «Context-Aware Communication» – IEEE Wireless Communications.
14. Google. [Installing the SDK](#). Besøkt 23. april 2008.
15. Google. [Dalvik Debug Monitor Service](#). Besøkt 23. april 2008.
16. Adams, Paul (14. november 2007). [Some Open-Source Advocates Find Google's Android a Sinister Threat](#). Besøkt 24. april 2008.
17. Shankland, Stephen (14. november 2007). [Sun's worried that Google Android could fracture Java](#). Besøkt 24. april 2008.



Androidene

18. Hosanee, Matt (15. april 2008). [Developing Rich Mobile Apps using the JavaME Open Source Platform](#). Besøkt 28. april 2008.
19. Open Handset Alliance (5. november 2007). [Industry Leaders Announce Open Platform for Mobile Devices](#). Besøkt 24. april 2008.
20. Riley, Duncan (11. februar 2008). [GPhone Prototype Debuts At Mobile World Congress](#). Besøkt 28. april 2008.
21. Reuters (5. november 2007). [Rivals dismiss threat of Google mobile platform](#).
22. Google. [Android Developer Challenge](#). Besøkt 28. april 2008.
23. Paul, Ryan (19. desember 2007). [Developing apps for Google Android](#). Besøkt 29. april 2008.
24. Morrison, Scott (19. desember 2007). [Glitches Bug Google's Android Software](#). Besøkt 29. april 2008.
25. Paul, Ryan (15. april 2008). [Google talks up Android: open ecosystems lead to innovation](#). Besøkt 28. april 2008.
26. Millward Bown (21. april 2008). [Brandz ranking](#). Besøkt 28. april 2008.
27. Williams, Martyn (11. august 2007). [Google's Android just a press release, says Ballmer](#). Besøkt 28. april 2008.