

# QT HYBRID -OHJELMAN SUORITUSKYKYRAPORTTI MAEMO- ALUSTALLA

Petri Lappalainen

Opinnäytetyö  
Kesäkuu 2010

Tietotekniikka  
Tekniikan ja liikenteen ala





Tekijä(t) LAPPALAINEN, Petri	Julkaisun laji Opinnäytetyö	Päivämäärä 07.06.2010
	Sivumäärä 33	Julkaisun kieli Suomi
	Luottamuksellisuus ( ) saakka	Verkojulkaisulupa myönnetty ( X )
Työn nimi QT HYBRID -OHJELMAN SUORITUSKYKYRAPORTTI MAEMO-ALUSTALLA		
Koulutusohjelma Tietotekniikka		
Työn ohjaaja(t) PELTOMÄKI, Juha		
Toimeksiantaja(t) Digia Finland Oyj VÄLIVERRONEN, Jukka		
Tiivistelmä <p>Tässä Digia Finland Oyj:lle tehdyssä työssä tarkoituksena oli raportoida kuinka Qt Hybrid -ohjelmat toimivat mobiililla Maemo-alustalla. Yhdistelmä Qt-kehitysympäristön komponentteja ja Internetissä käytettäviä tekniikoita voi olla hyvinkin raskas mobiililaitteella. Suorituskehon riittämättömyys antaa heti käyttäjälle huonon käyttökokemuksen. Käytetyn Maemo-alustan versio tässä työssä oli Maemo 5, joka tunnetaan myös nimellä Fremantle.</p> <p>Työssä kerrotaan Qt Hybrid -ohjelmasta ja kuinka sellainen luodaan. Mitä komponentteja käytetään ja mitkä ovat kehityksen vaiheet. Pääkomponentti Qt Hybrid -ohjelman luomiseen on Qt-kehitysympäristön tarjoaman Qt WebKit -selainmoottorin käyttäminen. Qt Webkit tekee mahdolliseksi käyttää ohjelmissa WWW-maailmasta löytyviä tekniikoita kuten JavaScriptiä ja CSS-tyylitiedostoja. Näitä WWW-tekniikoita hyödyntäen voidaan tehdä Qt Hybrid -ohjelmille mm. monipuolisia käyttöliittymiä. Koska kaikki komponentit toimivat samassa Qt-käyttöympäristössä, on niillä suora väylä kommunikointiin toistensa kanssa, mahdollistaen WWW-sisällön muokkaamisen ohjelmasta käsin. Qt tarjoaa myös yhteyden laitealustan palveluihin.</p> <p>Suorituskykymittausta varten kehitettiin Qt Hybrid -ohjelma, joka käyttää yleisiä tekniikoita WWW-maailmasta. Luotua ohjelmaa käyttäen suoritettiin sarja suorituskykytestejä Nokia N900-mobiilitietokoneella, joka toimii Maemo 5 -alustalla.</p> <p>Mittaustuloksista saatiin selville Qt Hybrid -ohjelman muistinkulutus ja prosessoriajankäyttö. Luotu Qt Hybrid -ohjelma toimi laitteessa ilman suurempia ongelmia. Muutamassa testissä esiintyi suorituskykyongelmia silloin, kun testattiin animaatiota JavaScriptin avulla. Yhteenvetona Qt Hybrid -ohjelmat ovat valmiita otettavaksi käyttöön Maemo-alustalla.</p>		
Avainsanat (asiasanat) Linux, Maemo, Qt, WebKit		
Muut tiedot		



Author(s) LAPPALAINEN, Petri	Type of publication Bachelor's Thesis	Date 07.06.2010
	Pages 33	Language Finnish
	Confidential ( ) Until	Permission for web publication ( X )
Title QT HYBRID APPLICATION'S PERFORMANCE STUDY ON MAEMO-PLATFORM		
Degree Programme Information Technology		
Tutor(s) PELTOMÄKI, Juha		
Assigned by Digia Finland Oyj VÄLIVERRONEN, Jukka		
Abstract <p>In the Bachelor's thesis assigned by Digia Finland Oyj the goal was to study how Qt Hybrid applications would work on a mobile platform like Maemo. Mixing native components from Qt-framework and technologies available from World Wide Web could be an issue to the performance of mobile devices and thus damaging the user experience. The Maemo platform version used in this study was Maemo 5, which also known as Fremantle.</p> <p>The thesis describes how Qt Hybrid applications are created and what the basic components and steps to make one are. The main component to make a Qt Hybrid application with Qt-framework is the usage of Qt WebKit layout engine. Using Qt WebKit allows users to create hybrid applications that are run as a native program on the target system. Qt WebKit allows the use of components from the WWW, like JavaScript and CSS, to make rich user interfaces. As the components are running on the same Qt-framework platform, a direct connection between them is possible. Qt also provides access to platform services.</p> <p>To study the performance, a Qt Hybrid application was created that utilizes the most commonly used technologies from the WWW. Series of performance tests were executed on Nokia N900 mobile device that uses Maemo 5 platform.</p> <p>The results from the performance tests were good in overall. The Qt Hybrid program was able to run on the device without any major problems. Some minor issues with constant drawing were noticed when executing it during complex animations with JavaScript. Also the memory consumption of the application was put to notice. In summary, Qt Hybrid applications are ready to be used in mobile platforms like Maemo.</p>		
Keywords Linux, Maemo, Qt, WebKit		
Miscellaneous		

# SISÄLTÖ

KÄSITTEET.....	3
1 TYÖN LÄHTÖKOHDAT.....	5
1.1 Toimeksiantaja.....	5
1.2 Tehtävän taustaa.....	5
2 QT HYBRID -OHJELMA.....	7
2.1 Qt Hybrid -ohjelman taustaa.....	7
2.2 Qt yleisesti.....	9
2.3 Qt WebKit ja web-komponentit.....	11
2.3.1 Yleistä Qt WebKitistä.....	11
2.3.2 JavaScript.....	12
2.3.3 Tyylitiedostot.....	13
2.4 Muita yhdistelmätekniikoita.....	14
3 YMPÄRISTÖ SUORITUSKYVYN MITTAUKSEEN.....	16
3.1 Maemo-ympäristö.....	16
3.2 N900 mobiilitietokone.....	17
3.3 Qt Hybrid -sovelluksen kehitys.....	18
3.4 WWW-sovellus.....	19
3.4.1 Yleistä WWW-sovelluksesta.....	19
3.4.2 Tekstin animointi.....	20
3.4.3 Tekstin ja kuvien animointi.....	21
3.4.4 Grafiikan piirto ja animaatio.....	22
3.4.5 Interaktiivinen kuvakaruselli.....	23
3.5 Qt-sovellus.....	23
4 SUORITUSKYKYMITTAUS.....	26
4.1 Lähtötilanne.....	26
4.2 Tulokset.....	27
5 POHDINTA.....	30
LÄHTEET.....	32

## KUVIOT

KUVIO 1. Qt WebKit -komponentin yhteydet.....	13
KUVIO 2. Näkymä Maemo 5 -työpöydälle Xephyrin kautta.....	17
KUVIO 3. Nokia N900 -mobiilitietokone.....	18
KUVIO 4. Testin 1 kuvakaappaus.....	20
KUVIO 5. Testin 2 kuvakaappaus.....	21
KUVIO 6. Testin 3 kuvakaappaus.....	22
KUVIO 7. Testin 4 kuvakaappaus. ....	23
KUVIO 8. Qt Hybrid -ohjelman projektitiedoston sisällöstä.....	24
KUVIO 9. Scratchbox-kohteiden konfigurointi.....	26

## TAULUKOT

TAULUKKO 1. Opera Mini -selaimen sivunlataukset.....	6
TAULUKKO 2. N900-laitteen muistinkäyttö testien aikana.....	27
TAULUKKO 3. N900-laitteen suoritinkäyttö testien aikana.....	28
TAULUKKO 4. Qt Hybrid -ohjelman tiedot suorituksen aikana.....	29
TAULUKKO 5. Kommentit Qt Hybrid -ohjelman muista huomioista.....	29

## KÄSITTEET

**CSS** - (Cascading Style Sheets) World Wide Web Consortium (W3C) ylläpitämä tyylikieli. Yksinkertainen tapa lisätä tyylejä (fontteja, värejä, rivityksiä) WWW-dokumenteille. Laajalti käytössä oleva graafisten WWW-selaimien tukema tekniikka.

**Eclipse** - Ohjelmistokehitysympäristö. Laaja tuki eri ohjelmointikielille laajennusten avulla.

**ESbox** - Eclipse-pohjainen Maemo-ohjelmistokehitysympäristö. Sisäänrakennettu tuki tarvittaviin komponentteihin, että ohjelmia voi kehittää helposti Maemo-ympäristössä.

**HTML** – (HyperText Markup Language) W3C:n ylläpitämä standardi, WWW-sivujen käyttämä kuvauskieli.

**JavaScript** - Oliopohjainen skriptikieli. Mahdollistaa dynaamisen sisällöntuotamisen WWW-palveluissa. Yleisesti tuettu kieli kaikissa WWW-selaimissa.

**jQuery** – Ilmainen ja avoimenlähdekoodin kevyt JavaScript kirjastokokoelma auttamaan asiakaspään JavaScript-ohjelmointia. Mahdollisuus myös laajentaa plugin-ominaisuuksien avulla.

**Maemo** – Debian Linuxiin pohjautuva mobiililaitteiden käyttöjärjestelmä. Suurilta osin avoimen lähdekoodin projekti. Pääkäyttökohde älypuhelimiin ja Internet-tablet tietokoneisiin. Nykyinen versio on Maemo 5 eli Fremantle.

**Qt** – Alustariippumaton ohjelmistojen ja graafisten käyttöliittymien kehitysympäristö. Tukee useita käyttöjärjestelmäympäristöjä. Sisältää useita luokkakirjastomoduuleita. Käyttää sisäisesti C++-ohjelmointikieltä, mutta tuki on myös saataville muutamalle muulle ohjelmointikielille (Java, Python, jne.).

**Qt Creator** – Ohjelmistokehitysympäristö Qt-ohjelmien tekemiseen. Julkaistu virallisesti Qt-versio 4.5 myötä.

**SDK** – (Software Development Kit) Yleinen työkalupaketti ohjelmistojen tekoon. Yleensä sisältää kaiken tarvittavan aloittaa ohjelmistojen tekemisen tiettyyn tarkoitukseen.

**Scratchbox** – Ristikääntötyökalupaketti. Täysin eristetty kääntöympäristö, joka mahdollistaa ohjelmien suorittamisen tiettyä laitekohdetta vasten. Kehitetty helpottamaan ohjelman kääntöä mobiileille Linux-alustoille, kuten Maemolle.

**WebKit** – Kehittynyt avoimen lähdekoodin WWW-selainmoottori. Käytössä useissa markkinoilla olevissa nettiselainohjelmissa. Qt sisältää WebKit-tuen oman QtWebKit-laajennuksen kautta.

**Web 2.0** – Nimitys tavasta jakaa tietoa ja toimia Internetissä käyttäjäläheisten palveluiden avulla.

**Widget** – Nimitys uudelleenkäytettävälle graafiselle käyttöliittymäkomponentille Qt-sovelluskehityksessä.

**Xephyr** - X-ikkunointijärjestelmä. Mahdollistaa mm. ikkunanäkymän Maeroon *Scratchboxin* kanssa.

**XML** – (Extensible Markup Language) W3C:n ylläpitämä standardi kuvauskieli käsitellä tietoa ihmisen luettavassa muodossa.

# 1 TYÖN LÄHTÖKOHDAT

## 1.1 Toimeksiantaja

Toimeksiantajana työssä toimi Digia Finland Oy:n Jyväskylän yksikkö. Digia on ohjelmistoalan yritys, jolla on toimintaa monella informaatioteknologian aloilla. Pääliiketoiminta-alueena toimivat yritysjärjestelmät ja mobiilisovellukset. Yritysjärjestelmien, mobiililaitteiden, käytettävyyssuunnittelun ja verkkosovellusten osaamisella asiakaslähtöinen Digia tarjoaa ratkaisuja monipuolisesti asiakkaiden tarpeisiin. (Digia 2010.)

Yrityksen palveluksessa työskentelee lähes 1500 työntekijää Suomessa, Ruotsissa, Venäjällä ja Kiinassa. Pörssiyrityksenä Digia on listattuna NASDAQ OMX Helsingissä tunnuksella DIG1V. (Digia 2010.)

## 1.2 Tehtävän taustaa

Useaa alustaa tukevan Qt-kehitysympäristön tuomat uudet tekniset mahdollisuudet olivat lähtökohta tätä opinnäytetyötä varten. Toimeksiantajalla ja itselläni oli kiinnostusta Qt:n mahdollistamiin erilaisiin tekniikoihin mobiileilla laitelustoilla. Yksi näistä tekniikoista on Qt Hybrid -ohjelmat. Qt:n alustariippumattomuus ja Qt WebKit mahdollistavat tämänlaisten ohjelmien kehittämisen. Yhdistelemällä WWW-maailman ja käyttöympäristössä toimivan sovelluksen parhaat puolet saadaan muodostettua hybrid-ohjelma. Tärkeimpänä lähtökohdana olikin käyttää Internetin viitoittamaa kehitystä, nimekkäänä käsite Web 2.0. Interaktiivisten ja käyttäjäläheisten WWW-palveluiden kehittyessä koko ajan voidaan hyviä ratkaisuja soveltaa kehitettäessä Qt Hybrid -ohjelmia haluttuun ympäristöön.



Internetin käyttö mobiilisti on ollut kasvavaa viimeiset 2 vuotta, kun katsoo Opera Softwaren ilmoittamia Opera Mini WWW-selaimen käyttölukuja heinäkuusta 2008 maaliskuulle 2010 väliseltä ajalta. Taulukossa 1 ilmoitetut sivunlataukset on saatu Opera Mini-selaimen käyttämisestä välityspalvelimista. (Opera Software 2010.)

TAULUKKO 1. Opera Mini -selaimen sivunlataukset.

Aika	Sivunlatauksia (Miljardeissa)
Huhtikuu 2008	n. 2,5
Lokakuu 2008	n. 5,1
Huhtikuu 2009	n. 8,7
Lokakuu 2009	n. 17,2
Maaliskuu 2010	n. 25,9

Kasvusta kertoo myös Quantcastin julkaisema Mobile Web Trends 2009 Report -julkaisu. Joulukuusta 2008 - joulukuun 2009 väliseltä ajalta mobiilin Internetin käyttö lisääntyi pohjois-Amerikassa 110 prosenttia. Maailmanlaajuisesti kasvua on tapahtunut kyseiseltä ajanjaksolta 148 prosenttia. Verrattaessa koko Internetin käyttöön mobiilin Internetin osuus koko maailmassa oli joulukuussa 2009 0.99 prosenttia. (Quantcast 2010.)

Käyttäjät ovat siis tottuneet käyttämään Internetiä mobiileilla laitteilla, joissa on omat käyttörajoituksensa. Yksi merkittävä tekijä, verrattaessa työpöytäympäristöön, on mobiililaitteen näytön koko ja sen käyttämä erottelukyky. On yleistä, että suosituista WWW-palveluista on olemassa myös oma versio mobiileille laitteille. Selainvaihtoehtotarjonta on myös laajentunut Internetin käytön kasvaessa.

Työn yhtenä tavoitteena oli kertoa hieman yleisesti Qt Hybrid -ohjelman käyttämisestä eri tekniikoista. Ymmärtämällä tausta saadaan käsitys, kuinka yleiskäyttöinen hybrid-ohjelma voi olla. Toinen tavoite oli suorituskyvyn tarkasteleminen, eli onko hybrid-ohjelmaa sulavaa käyttää mobiilissa Maemo 5 -alustassa (tunnetaan myös nimellä Fremantle) muistin kulutus ja muut tekijät huomioon ottaen. Qt-komponenttien yhdisteleminen voi olla raskasta mobiilissa Linux-ympäristössä. Markkinoilla oli tätä opinnäytetyötä kirjoittaessa vain yksi mobiili Maemo 5 -laite, Nokia N900 -mobiilitietokone, joten työssä oli otettu huomioon suorituskyky vain tässä laitteessa.

Suorituskyvyn mittausta varten työssä kehitettiin hybrid-sovellus Qt-kehitysympäristön avulla. Ohjelmassa käytetään moderneja yleisesti käytettyjä WWW-tekniikoiden komponentteja kuten JavaScriptiä ja tyyli tiedostoja. Kehitetyllä sovelluksella ajetaan suorituskykymittauksia Maemo 5:sta käytävällä N900-laitteella. Sovelluksessa yhdistelmä erilaisia Qt-tekniikoita voi olla mahdollisesti hyvinkin raskas mobiililla laitteella kuten N900. Laitteen suorituskykyrajoitteet voivat haitata käyttäjän käyttökokemusta. Lopputuloksena saadaan käsitys, ovatko tämän tapaiset sovellukset valmiita mobiileille laitteille, joissa resurssit ovat rajalliset verrattaessa moderniin tietokoneympäristöön.

## 2 QT HYBRID -OHJELMA

### 2.1 Qt Hybrid -ohjelman taustaa

Web 2.0:n myötä Internetin monipuoliset palvelut ovat tuoneet uudenlaisen ja kiinnostavan kokemuksen käyttäjille. WWW-sivujen määrä ja tieto Internetissä lisääntyy koko ajan. Käyttötapojen laajentuessa ja vaatimusten kasvaessa on selvää, että kehitys ja tarve WWW-tekniikoihin on olemassa. Tästä tilanteesta on hyvät lähtökohdat uudentyyppisten hybrid-ohjelmien kehittämiseen.

Hybrid-ohjelma on lyhyesti ilmaistuna yhdistelmä WWW-maailmaa ja käyttöympäristön sovellusta. Uutta tällaisessa ohjelmankehitystavassa on WWW-maailmasta löytyvien tekniikoiden käyttö, kun tehdään käyttöympäristössä yksilöllisesti toimivaa ns. natiivia sovellusta. Internetissä käytettävien tekniikoiden kehitys ajanut eteenpäin uutta tapaa, koska Web 2.0:n jälkeen interaktiiviset sivustot ovat tulleet hyvin suureksi osaksi Internet-palveluita. Hybrid-ajattelussa käytetäänkin hyödyksi jo hyväksi havaittuja tapoja ja tekniikoita WWW-palveluiden tekemisestä. Tällöin raja työpöytäsovelluksen ja WWW-sovelluksen kesken tulee olemaan hyvin ohut.

Mitä käyttöympäristö yhteyden mukaan tuominen tuo lisää WWW-sovelluksiin on, että ohjelman käyttämä tieto on aina saatavilla, koska sitä voidaan lukea ja kirjoittaa käyttöympäristössä olevasta muistista. Normaalilla WWW-sovelluksella on yleensä vaatimuksena yhteyden saaminen Internetiin, koska käytettävää tietoa WWW-sovelluksella on saatavilla vain rajoitetusti kerrallaan. Hybrid-ohjelmassa on mahdollista säilyttää tieto aina kohdejärjestelmässä, koska se toimii kuten muutkin järjestelmän sovellukset. Lisäetuina on myös mahdollisuus pääsyyn kohdejärjestelmän muiden palveluiden käyttöön, kuten esimerkiksi kiihdytettyyn grafiikkaan, satelliittipaikannuspalveluihin tai multimediasovelluksiin.

Yksi hybrid-ohjelmien tekoon mahdollistavista tekniikoista on Qt-kehitysympäristön käyttö. Qt:n valmiiden komponenttien tuki ja olemalla useaa kohdejärjestelmää tukeva hybrid-sovelluksella on heti mahdollisuus laajaan käyttöympäristöön. (Lal 2010.)

Seuraavana on listaus Qt-pohjaisten hybrid-ohjelmien molempien puoliskojen hyvistä ominaispiirteistä (Lal 2010.):

WWW-maailman hyvät puolet:

- HTML-, CSS-, JavaScript-standardit
- Laajasti tuettu
- Hyvät kirjastot ja ohjelmat kehitykseen

Qt-ympäristön hyvät puolet:

- Tuki usealle ympäristölle, uudelleenkäytettävä lähdekoodi
- Valmiit kirjastot palveluiden käyttöön
- Valmiit käyttöliittymäwidgetit
- Yhteys kohdejärjestelmään liitäntärajapintojen kautta

Nämä yhdistettynä saadaan ohjelma, jolla on seuraavanlaiset ominaisuudet:

- Nopea ottaa käyttöön
- Laaja tuki useille alustoille
- Pääsy käyttöympäristön ominaisuuksiin WWW:n kautta.

Qt-ympäristössä hybrid-ohjelma tarkoittaa Qt WebKit -komponentin ja muiden Qt-komponenttien yhdistelemistä. Qt WebKit, olemalla moderni WWW-moottori, mahdollistaa yhteyden käyttää WWW-maailmasta tuotuja tekniikoita.

Koska Qt:n komponentit toimivat samalla käyttöalustalla, on niillä suora yhteys toisiinsa. Yhteydellä saadaan väylä tiedon levitykseen WebKit:n sisältä käyttöalustaan ja myös toisinpäin. (Qt Development Frameworks 2010.)

## ***2.2 Qt yleisesti***

Qt on käyttöjärjestelmäriippumaton kehitysohjelma ohjelmistoille, jonka on kehityksen juuret alkavat vuodesta 1991. Tarkoituksena on ollut kehittää ohjelmien ja käyttöliittymien tekoon ympäristö, joka on alustariippumaton ja uudelleenkäytettävä ilman lähdekoodin uudelleenkirjoitusta. Nykyään Qt:n kehityksestä, markkinoinnista, myymisestä ja tuesta vastaa Nokia Oyj:n omistuksessa oleva Qt Development Frameworks, joka pitää pääkonttoriaan Norjan Oslossa. (Qt About us 2010.)

Qt:n pohjana on modulaarinen luokkakirjasto rakenne, jonka pääluokkana toimii Qt Core -moduuli. Moduuleita on tarjolla useita, versioiden kehittyessä vieläkin enemmän. Nimekkäimpänä ovat mm. GUI-moduuli (QtGui), joka sisältää käyttöliittymien tekoon liittyviä komponentteja, Networking-moduuli (QtNetwork) verkkoyhteyksien luomiseen ja multimedia-moduuli (Phonon) kuvien, äänien ja videoiden käsittelyyn. (Qt Modular Class Library 2010.)

Qt-ohjelmien kehityksen pääkielenä kehityksessä toimii C++, joka on suosittu olio-ohjelmointikieli. Mahdollisuus on myös kehittää muillakin kielillä erilaisten käännösten avulla, kuten esimerkiksi Javalla (Qt Jambi). Tätä kirjoittaessa Qt:n uusin vakaa versionumero on 4.6.2, joka on julkaistu joulukuussa 2009. Tässä virallisessa versiossa tuetut käyttöjärjestelmät ovat (Qt Platform 2010.)

- Windows
- Linux/X11
- Mac OS X
- Sulautetut Linux alustat
- Windows CE
- Symbian
- Maemo.

Muille alustoille tehtyjä avoimenlähdekoodiyhteisöjen käännöksiä on myös saatavilla. Ohjelmistokehitykseen on saatavilla erillinen SDK. Tämä on tarjolla kahdella eri lisenssiehdolla ja erillisillä SDK-paketeilla:

- Kaupallisella (Commercial)
- Avoimella (GNU Lesser General Public License, eli LGPL).

Qt SDK:n mukana tulee valmiiksi hyviä työkaluja, kuten version 4.5 myötä julkaistu sovelluskehitin Qt Creator. Sovellus tukee yleisiä kehitysympäristön toimintoja: projektin hallintaa, debuggausta ja kääntämistä. Lisänä on myös tuki Qt Designer -laajennukselle, jolla voi tehdä suoran näkymän avulla käyttöliittymiä. Ohjelmien kääntämiseen Windows-alustalla SDK:n mukana tulee MinGW-kääntäjä (Minimalist GNU for Windows), joka pohjautuu Unix-maailmasta tuttuun GCC-kääntäjään (GNU Compiler Collection). Qt:n laajan lisämoduulituen ansiosta Qt-sovellusten kehitystä ja kääntämistä voi tehdä myös muissakin sovelluskehittimissä. (Qt Platform 2010.)

Sovellusten suorittamiseen kohdeympäristössä on vaatimuksena Qt-kirjastojen löytyminen ja että ohjelma on käännetty kyseiseen kohdejärjestelmän tarpeilla. Esimerkiksi mobiilitietokone N900 käyttää ARM-prosessoriarkkitehtuuria, joten se täytyy ottaa huomioon käännöksen yhteydessä. Tarvittavat kirjastot voivat tulla jaetun sovelluksen mukana tai ne voidaan ladata erikseen toimimaan kaikkialla kohdejärjestelmässä.

## **2.3 Qt WebKit ja web-komponentit**

### **2.3.1 Yleistä Qt WebKitistä**

Qt WebKit pohjautuu suoraan sen esikuvaan WebKit-moottoriin. Moottori tukee useita nykypäivän WWW-tekniikoita ja on myös laajettavissa lisämoduulien avulla. Tuettuna Qt:ssa se on ollut alkaen versiosta 4.4, ja se päivitetään yleensä uuden version yhteydessä, jos vastaavasti alkuperäiseen WebKit-moottoriin on tullut muutoksia. Qt WebKitin nykyinen versio 4.6 toi mukanaan rajapintapäivityksiä ja tuen Symbian alustalle. (WebKit.org 2010.)

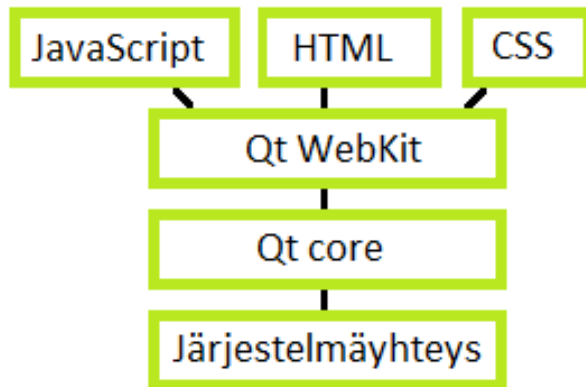
Qt WebKit on yksi pääkomponenteista kehitettäessä hybrid-ohjelmaa. Tämä luo yhteyden käyttäen WWW-maailman tarjoamia tekniikoita ohjelmissa tietenkin ottaen huomioon, mitä Qt WebKit tukee. Luvuissa 2.3.2 – 2.3.3 on kerrottu tarkemmin, mitkä tekniikat ovat erityisen käytettäviä. (WebKit.org 2010.)

### 2.3.2 JavaScript

JavaScript on vuonna 1995 esitelty, alun perin Netscape WWW-selaimeen tarkoitettu komentosarjakieli, jolla saadaan ohjelmallinen pääsy kohdejärjestelmän tietoihin. Se on nykyisin yleisesti tuettu kieli ja sitä ylläpitää Mozilla Foundation. JavaScriptiä on mahdollista käyttää asiakaspään- tai palvelinpään sovelluksissa. Yksi asiakaspään pääkäyttökohteista on muokata haettua tietoa reaaliaikaisesti, mikä mahdollistaa dynaamiset WWW-palvelut. Se tarkoittaa, että sivujen rakennetta voidaan muuttaa käyttötarkoitukseen sopivaksi ilman, että sivuja täytyy hakea uudestaan. Kun verrataan staattisiin WWW-sivuihin, mahdollistaa se parannuksia käyttöliittymän tekoon ja tiedon hakuun. Tästä hyödyistä johtuen moderneissa WWW-palveluissa JavaScriptin käyttö on melkein välttämätöntä. (JavaScript 2010.)

Tekniikkana JavaScript on ollut olemassa jo pitkään, joten WWW-sovellusten tekemiseen on tarjolla Internetissä monia valmiiksi tehtyjä JavaScript-kirjastoja. Useat tarjolla olevista kirjastoista ovat avoimen lähdekoodin projekteja, joten ennen käyttöönottoa tarvitsee tehdä vain lisenssiasioiden tarkistaminen. Esimerkkinä avoimesta projektista on suosittu asiakaspään jQuery-kirjasto, jota käytetään myös tämän työn suorituskykymittauksen kanssa.

Qt Hybrid -ohjelmassa Qt WebKitin käyttö mahdollistaa myös JavaScript-tiedon muokkaamisen kohdejärjestelmästä käsin. Koska kaikki komponentit toimivat samalla alustalla, on yhteys saatavissa myös WebKitin JavaScriptin käyttöön. Kommunikointi saadaan myös toimimaan kahteen suuntaan: JavaScriptistä Qt-ohjelmaan ja Qt-ohjelmasta JavaScriptiin (ks. kuvio 1).



KUVIO 1. Qt WebKit -komponentin yhteydet.

### 2.3.3 Tyylitiedostot

Tyylitiedostoja (CSS eli Cascading Style Sheets) käytetään dokumenttien ulkoasun/kuvauksen muokkaamiseen. CSS:n määritelmää ylläpitää World Wide Web Consortium (W3C). CSS-tyylitiedostot ovat yleisesti WWW-selaimien tukema tekniikka, mutta selainkohtaisia eroja, kuinka hyvin tyylitiedostot on tuettu. Pää tarkoitus on helpottaa WWW-sivujen komponenttien ulkoasun muokkausta, eli minkälaiselta eri osiot ja komponentit näyttävät käyttäjälle. Käyttämällä tyylitiedostoja on komponenttien muokkaus huomattavasti nopeampaa, kuin muokata jokaista erikseen saadakseen halutun näköisen lopputuloksen. (CSS introduction 2010.)

Yleisiä tyylitiedostojen käyttökohteita:

- kirjasimien ulkoasu
- tekstin väri, taustaväriydet
- tekstin/kuvien/elementtien sijoitus, reunat ja päällekkäisyydet



WWW-sivujen käytössä olevat tyylitiedostot voivat vaihdella sen mukaan, millainen käyttötarkoitus sivuilla sillä hetkellä on. Esimerkiksi työpöytäkäytössä olevissa WWW-selaimissa on usein tuettu WWW-sivujen tulostaminen. Sivun tyylitiedostossa voidaan määritellä, millainen tyyli on käytössä tulostusnäky-  
mässä. Toinen esimerkki on mobiililaitteiden käyttämät WWW-selaimet. Tyylitiedostoissa voidaan käyttää eri tyylimäärittelyä, kun käytössä on mobiililaitteen selain. Esimerkiksi kuvat ja tekstit voidaan sovittaa näkymään paremmin, kun käytössä kannettavan laitteen pienempi näyttö. (CSS introduction 2010.)

Qt:n käyttöliittymäkomponenteille on myös tuki tyylitiedostoille oman Qt Style Sheets -luokan avulla. Qt-tyylitiedostojen periaatteet, termit ja syntaksi on hyvin samanlainen kuin WWW-maailman CSS tiedostoissa. Tämä helpottaa Qt-käyttöliittymäkomponenttien ulkoasun muokkausta samalla tavalla kuin WWW-selainkäytössä. (Qt 4.6 Style Sheets 2010.)

Qt Hybrid -ohjelmassa onkin käytössä käyttöliittymän esittämiseen tyylitiedostot. Tyylitiedostoja voidaan muokata reaaliaikaisesti käytössä kuten JavaScriptia. Qt WebKit:n komponentille voidaan kertoa mitä tyylitiedostoa käytetään milläkin hetkellä.

## **2.4 Muita yhdistelmätekniikoita**

Qt Hybrid ei ole ainoa tekniikka joka yhdistää WWW-maailmaa ja laitekohtaista sovellusta. Nokian kehittämä Web Runtime -tekniikka (WRT) on yksi tällaisista tekniikoista. WRT-widgetit toimivat S60-alustan Symbian-käyttöjärjestelmällä varustetuissa laitteissa käyttöliittymänä Web 2.0 palveluille ja Internet sisällölle. Qt Hybrid -ohjelmien tapaan WRT-widgetit hyödyntävät WebKit-selaimmoottoria tiedon esittämiseen ja samalla tavoin käyttöliittymän tekoon WRT-widgeteissä käytetään HTML:ää, JavaScriptiä ja CSS-tyylitiedostoja. (Forum Nokia Web Runtime widgets 2010.)

WRT-widgeteissä JavaScriptillä saadaan pääsy järjestelmän sisäisiin palveluihin erillisen Nokia Platform Services 1.0 -rajapinnan kautta. Rajapintayhteys on ollut tuettuna valikoidusti S60-alustaisissa laitteissa versiosta 3.2:sta eteenpäin, mutta tuki myös muutama vanhempaan S60 3.1 -laitteeseen on saatavilla erillisen ohjelmistopäivityksen kautta. Rajapintayhteydellä on pääsy muun muassa järjestelmän kontakteihin (Contacts Service API), kalenteriin (Calendar Service API), GPS-paikkatietoihin (Location Service API) ja muihin asennettuihin ohjelmiin (AppManager Service API). Uudemmissa S60 5.0 -alustaisissa laitteissa, kuten esimerkiksi Nokia N97:ssä, on myös tuki Nokia Platform Services 2.0 -rajapinnalle, joka vielä lisää JavaScriptin mahdollistamia yhteyksiä järjestelmän palveluihin. WRT-widgeteistä saa myös lukunäkymän järjestelmän kotiruutuun, jos laitteen ohjelmistoversiosta sitä tukee. Valmiit WRT-widgetit jaetaan wgz-päätteisissä tiedostoissa. Tiedosto on ZIP-tiedostomuodossa pakattu hakemistorakenne, jonka sisältö täytyy olla WRT-määrittelyn mukainen. (Forum Nokia Web Runtime widgets 2010.)

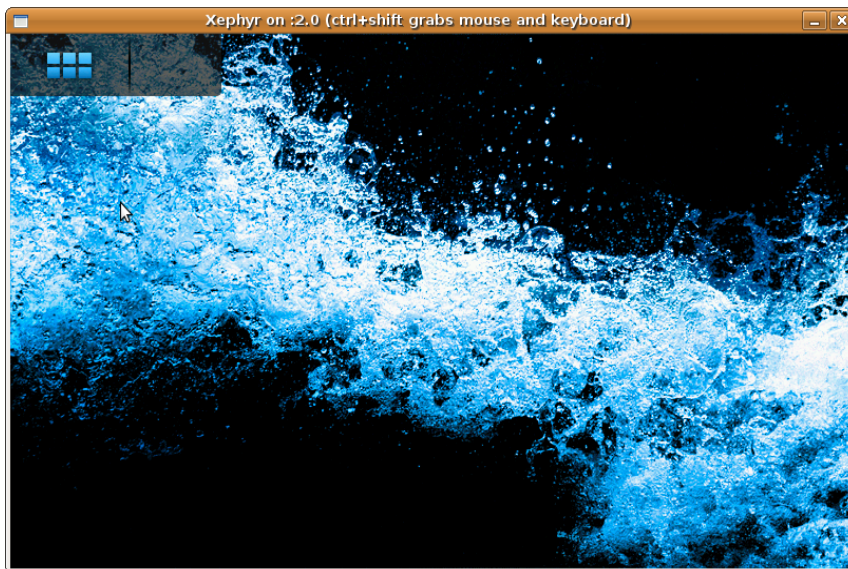
Apple iPhoneille on Applen kehittämä WWW-sovelluksen ja natiivi-sovelluksen yhdistelmätekniikka nimeltä iPhone web app. Tämä tekniikka käyttää laitteen Safari-selainta (pohjautuu myös WebKit-selainmoottoriin) WWW-sisällön esittämiseen. Peruseräilläään hyvin samanlainen kuin Qt Hybrid ja WRT-widgetit.

## 3 YMPÄRISTÖ SUORITUSKYVYN MITTAUKSEEN

### 3.1 Maemo-ympäristö

Maemo on Nokia Oyj:n kehittämä Linux-pohjainen ohjelmistoalusta mobiileille laitteille. Se on suurimmalta osin avointa lähdekoodia käyttävä alusta, pohjautuen Debian Linux-jakeluun. Ensimmäinen Maemo-versio julkaistiin vuoden 2005 marraskuussa nimellä OS2005. Uusin Maemo-versio on nimeltään Maemo 5. Käyttöliittymätyyli Maemo 5:ssä on nimeltään Hildon. Qt-versio 4.5 myötä tuli tuki Qt-käyttöliittymäkomponenteille, että ne näyttävät ulkoasuiltaan Maemo 5:ssä "Hildonimaisilta". (Maemo.org 2010.)

Maemo 5 ohjelmistokehityksen apuna käytetään Scratchbox-ristiinkääntötyökalua, jonka avulla voidaan tehdä kääntäminen Maemoa käyttävän laitteen (N900) ymmärtämään muotoon. Scratchbox on tehty helpottamaan ohjelmistokehitystä, koska N900-laite käyttää ARM-proessoriarkkitehtuuria, kun yleisesti kehitystyö tapahtuu PC-ympäristössä, joka toimii x86-arkkitehtuurilla. Scratchbox-asennuksen jälkeen tarvitaan Maemo 5 SDK:n asennus. Maemo 5 SDK tukee virallisesti Debian-pohjaisia x86-32 Linux-ympäristöjä, kuten Ubuntu. SDK:n asennus asentaa x86- ja armel-kääntötyökalut Scratchboxin sisään, että ohjelmia voi ajaa järjestelmässä. Koska Scratchbox toimii itsenäisenä järjestelmänä, pääsee siihen kirjautumaan sisään vain konsolin avulla. Graafisten ohjelmien suorittamiseen Scratchboxissa tarvitaan lisäksi saada käyttöliittymänäkymä(ks. kuvio 2). Tähän on apuna Xephyr X11-ikkunajärjestelmä. Qt-ohjelmien suorittamiseen Scratchboxiin on vielä asennettava erikseen Qt-kirjastot. (Scratchbox.org 2010; Maemo.org 2010.)



KUVIO 2. Näkymä Maemo 5 -työpöydälle Xephyrin kautta.

### 3.2 N900 mobiilitietokone

Opinnäytetyötä kirjoittaessa markkinoilla oli vain yksi Maemo 5 -käyttöjärjestelmällä varustettu laite: Nokia Oyj:n valmistama N900-multimediatietokone (ks. kuvio 3). Laite julkaistiin markkinoille vuoden 2009 loppupuoliskolla ja se oli ensimmäinen Maemo-laite jossa on mukana puhelinominaisuudet. Aikaisemmat Maemo-käyttöjärjestelmää käyttävät laitteet olivat pelkkiä Internet-päätelaitteita. Tiedonsyöttö laitteeseen tapahtuu joko kosketusnäytön tai sisäänrakennetun näppäimistön avulla. (Nokia Suomi. 2010.)

Listaus laitteen suorituskyvyn ominaisuuksista:

- 3,5” kosketusnäyttö
- Näytössä 800 X 480 pikselin erottelukyky
- Texas Instrumentsin OMAP3430 mikropiirisarja
- 600 MHz ARM Cortex-A8 prosessori
- PowerVR SGX 530 näytönohjain
- 430MHz C64x+ digitaalinen signaaliprosessori. (Nokia Suomi. 2010.)



KUVIO 3. Nokia N900 -mobiilitietokone.

Luodun Qt Hybrid -ohjelman ajamiseen laitteeseen tarvitsi asentaa erikseen  
4.6.2 Qt-kirjastot.

### **3.3 Qt Hybrid -sovelluksen kehitys**

Työssä oli tarkoituksena testata Qt Hybrid -ohjelman suorituskykyä ja tehdä sellainen ohjelma joka käyttää yleisiä WWW-maailman tekniikoita. Yksi pääpaino testaukselle oli JavaScriptin käyttäminen, koska se mahdollistaa käyttöliittymän interaktiiviset käytön. Muuten käyttöliittymä olisi vain kasa staattisia WWW-sivuja. Ohjelmassa käytetyt WWW-sivut ja lisäkirjastot ovat paikallisia tiedostoja, verkkoyhteyttä ei käytetä niiden hakemiseen, joskin se olisi myös mahdollista. Yhdessä testissä käytetään N900-laitteen verkkoyhteyttä kuvien lataamiseen Flickr-kuvapalvelusta.

Ohjelman kehitykseen työssä käytettiin valmista Maemo SDK Virtual Image -levy kuvaa, joka on valmis Ubuntu-käyttöjärjestelmään pohjautuva Linux-levy kuva. Kyseisessä käyttöjärjestelmässä on valmiiksi asennettuna tarvittavat ohjelmistot ja kirjastot Maemo ohjelmistokehitykseen.

Muita kehitystyön apuna hyödynnettyjä työkaluja olivat Qt SDK, Maemo 5 SDK, Scratchbox ja ESbox-ohjelmistokehitysympäristö. ESboxin käyttö helpottaa ohjelman suorittamista Scratchbox-ympäristössä, koska siinä on suoraan tuki Scratchboxin kanssa toimimiseen helpon käyttöliittymän avulla. WWW-sovelluksen teossa oli apuna käytetty Apatana Studio -ohjelmistoa.

## **3.4 WWW-sovellus**

### **3.4.1 Yleistä WWW-sovelluksesta**

Qt Hybrid -ohjelman WWW-sovellus käyttää seuraavia avoimia JavaScript-apukirjastoja

- The jQuery projektin jQuery versiot 1.3.3 ja 1.4.2 (Kaksoislisenssi MIT tai GPL)
- Simon Bairdin CoolClock versio 2.0 (BSD lisenssi)
- Jan Sorgallan jCarousel versio 0.2.4 (Kaksoislisenssi MIT tai GPL).

Monipuolisten JavaScript-kirjastojen käytöllä saadaan käsitys, kuinka tehokkaasti JavaScript toimii mobiilin alustan Qt Hybrid -ohjelmassa. Ohjelmassa testataan onko JavaScriptin käytössä suorituskykyrajoitteita, kun sitä käytetään käyttöliittymän suunnittelemiseen.

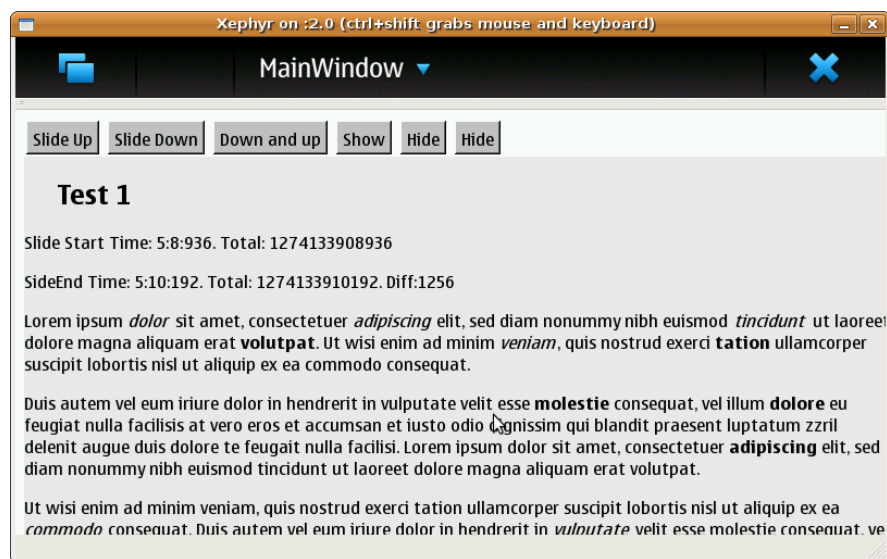
WWW-sovellus sisältää seuraat testit:

1. Animoitua tekstin muokkaamista jQuery JavaScript -kirjastolla
2. Animoitua tekstin ja kuvien muokkaamista jQuery-kirjastolla
3. Animointia ja kuvien piirtoa CoolClock-kirjastolla.
4. Interaktiivista kuvien selausta verkon läpi jCarousel-kirjastolla.

Luvuissa 3.4.2 – 3.4.5 käsitellään tarkemmin testien sisältö.

### 3.4.2 Tekstin animointi

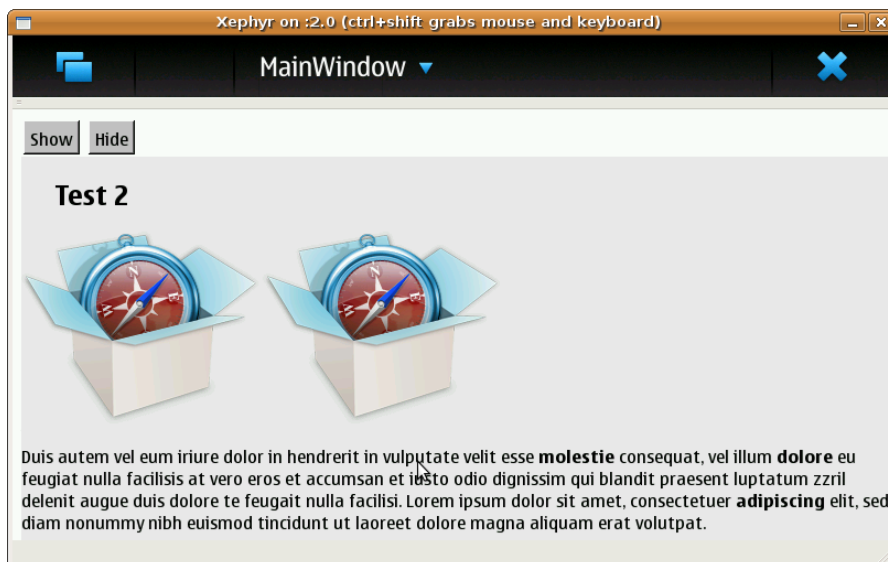
Ensimmäinen testi (ks. kuvio 4) sisältää jQuery-kirjaston animaatorajapinnan käyttöä Show-, Hide-, SlideUp- ja SlideDown-metodeilla. Metodeille on annettu parametriarvona *slow*, joka vastaa 600 millisekunnin kestoista animaatiota. Testissä kokeillaan, kuinka sulavaa animointi on kuudella tekstiä sisältävällä kappaleella. Jokainen kappale animoidaan erikseen. Merkkejä kappaleissa on yhteensä 2461.



KUVIO 4. Testin 1 kuvakaappaus.

### 3.4.3 Tekstin ja kuvien animointi

Toinen testi (ks. kuvio 5) on sisältönä samanlainen kun ensimmäinen, mutta siihen on lisätty samalla tekstin ja kuvien yhtäaikaiset häivytyanimaatiot jQuery-kirjaston Show- ja Hide-metodeilla. Parametriarvona metodeille on sama 600 millisekuntia animaatiota kohden kuten testissä 1. Samalla tavalla kuin ensimmäisessä testissä on tarkoituksena selvittää, kuinka kuvien lisäämien mahdollisesti muuttaa animaation sulavuutta. Jokainen kuva ja kappale animoidaan erikseen. Testi sisältää 4 kappaletta 35,6 kilotavun PNG-kuvatiedostoa tarkkuudella 215 x 174 -pikseliä. Tekstiosiossa on merkkejä yhteensä 2190 kappaletta.

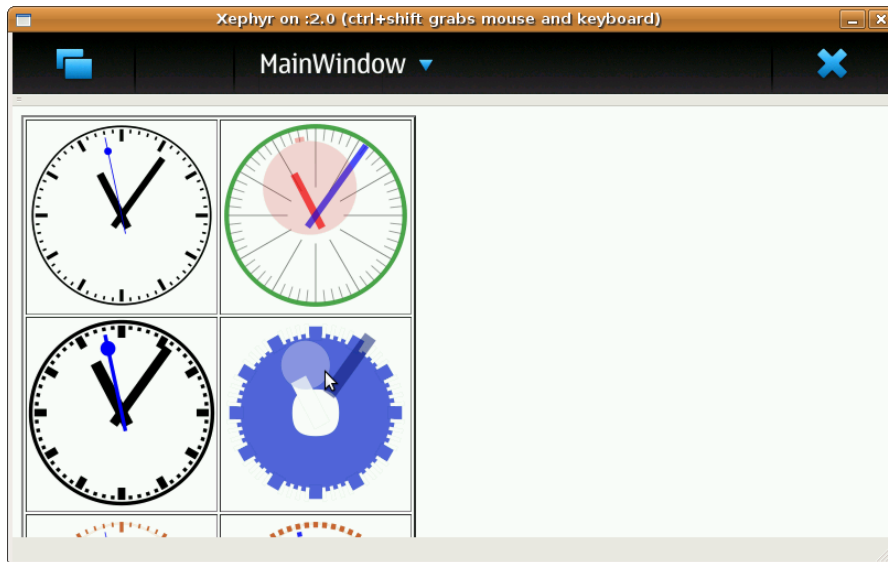


KUVIO 5. Testin 2 kuvakaappaus



### 3.4.4 Grafiikan piirto ja animaatio

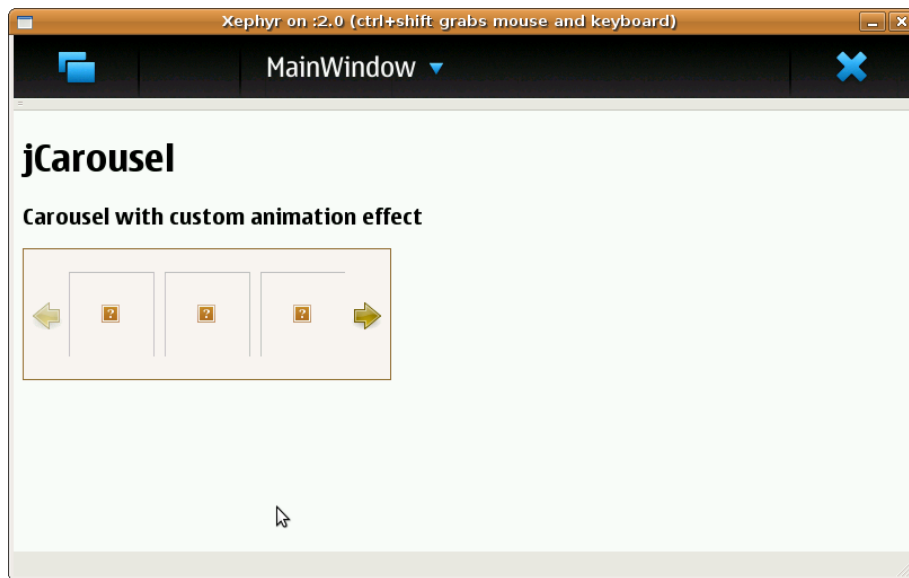
Kolmannessa (ks. kuvio 6) testataan piirtoa ja animointia CoolClock JavaScript -kirjastolla. Kirjasto näyttää reaaliaikaisesti kohdejärjestelmän kellonajan analogiseen kellon avulla. Testissä piirretään samanaikaisesti seitsemän erilaista ulkoasua käyttävää analogista kelloa. Testissä selvitetään kuinka sulavaa kellojen piirtäminen on reaaliajassa, esiintyykö korkeaa prosessorin ajankäyttöä. Jokainen kello piirretään CoolClockin perusasetuksilla, jolloin kellon säde on 85-pikseliä, mutta kellojen viisarit päivitetään sekunnin välein.



KUVIO 6. Testin 3 kuvakaappaus.

### 3.4.5 Interaktiivinen kuvakaruselli

Neljäs eli viimeinen testi (ks. kuvio 7) on interaktiivinen kuvakaruselli. Kuvakaruselli on tyypillinen WWW-sivuilla nähtävä kuvienselaustoiminto. Kirjasto käyttää tyylitiedostoja ja verkon läpi haettuja kuvia. Painamalla näkyvässä olevia nuolinäppäimiä siirtyvät kuvat sivusuunnassa ponnahtavan animaation avustuksella.



KUVIO 7. Testin 4 kuvakaappaus.

## 3.5 Qt-sovellus

Ensimmäinen askel Qt-sovelluksen tekemiseen on projektitiedoston luominen. Projektitiedoston luominen onnistuu Qt Creator -sovelluskehittimellä tai vaihtoehtona on käyttää erikseen qmake-ohjelmaa (Qt Creator viittaa myös tähän ohjelmaan). Luodussa sovelluksessa projektitiedoston pohjana käytettiin Qt Creator -sovelluskehittimen valmista mallipohjaa, joka sisältää tarvittavat viittauksen mallin lähdekoodeihin. Käytetty Qt-versio ohjelman suorittamiseen ja kääntämisessä oli 4.6.2. Kuviossa 8 nähdään käytetyn projektitiedoston sisältö.

```
1 # -----
2 # Project created by QtCreator 2010-05-09T19:23:40
3 # -----
4 QT += webkit \
5     xml \
6     xmlpatterns
7 TARGET = hybrid
8 TEMPLATE = app
9 SOURCES += main.cpp \
10     mainwindow.cpp
11 HEADERS += mainwindow.h
12 FORMS += mainwindow.ui
13 RESOURCES += html_hybrid/html.qrc
```

KUVIO 8. Qt Hybrid -ohjelman projektitiedoston sisällöstä

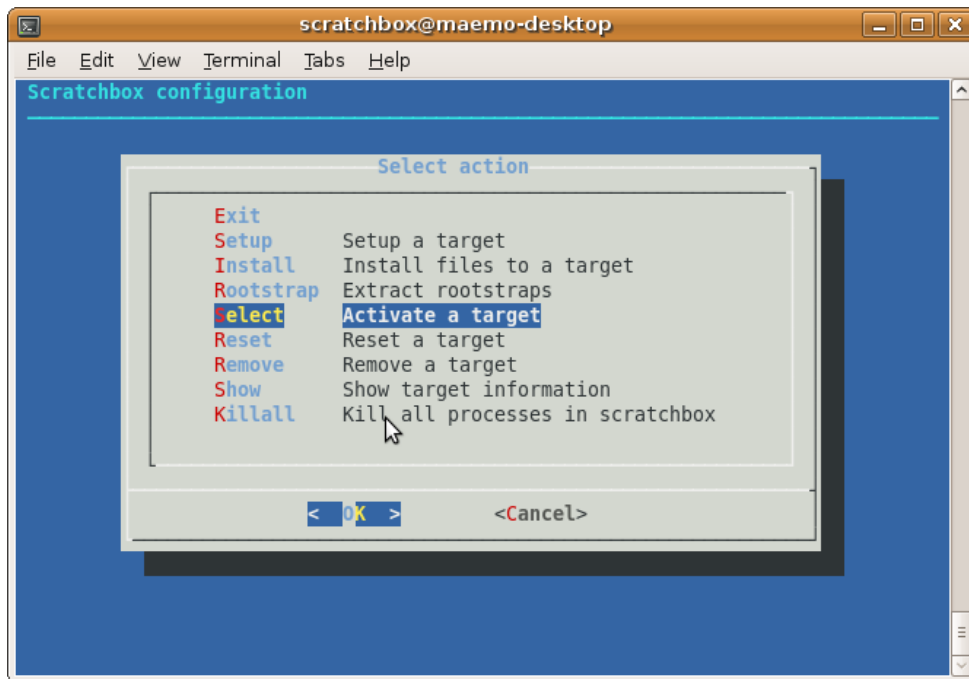
Projektitiedostossa on huomioitava, että siihen on lisätty QT-muuttujaan on asetettu "webkit"-viittaus. Viittauksen avulla kääntäjä osaa ottaa huomioon Qt WebKit-laajennuksen tarpeen. HEADERS- ja SOURCES-muuttujissa on määritelty sovelluksen käyttämät otsikko- ja lähdekoodi-tiedostot. TARGET-muuttuja kertoo käännettävän sovelluksen nimen. RESOURCES-muuttujassa määrillään Qt:n resurssijärjestelmätiedosto (Qt Resource System), joka sisältää kaikki WWW-sovelluksen tiedostot.

WWW-sovellus saadaan käyttöön, kun käytetään QwebView-luokkaa, joka on käyttöliittymäwidgetti Qt WebKitille. QwebView-luokalle annettiin viittaus HTML-tiedosto resurssiin, joka tässä sovelluksessa on aloitussivu WWW-sovellustesteille. Sovelluksessa QwebView-luokan sijoitus Qt:n käyttöliittymään tapahtuu erillisen UI-tiedoston avulla. Tiedosto on projektitiedoston FORMS-muuttujan alla oleva mainwindow.ui, sen luomisessa käytettiin Qt Designer-laajennusta. Tiedoston sisältää tiedon Qt:n UI-komponenttien ominaisuuksista, kuten sijainnista ja koosta, XML-muodossa.

Kun ohjelma toimii sovelluskehittimen käyttämässä ympäristössä, testataan seuraavaksi sen toimivuutta Maemo-ympäristössä. Tämä tehdään kääntämällä sovellus Scratchbox-ympäristössä N900-laitteen ymmärtämään muotoon. Ensimmäisenä täytyy kirjautua Scratchboxiin login-komennolla. Kirjautumisen jälkeen saadaan konsoliyhteys Scratchbox-ympäristöön, jonka jälkeen ohjelmia voi ajaa ns. virtuaalisen Maemon sisällä. Perusasetuksilla Scratchbox on x86-tilassa, jolloin ohjelmia voidaan ajaa virtuaalisesti. Käyttöliittymän näkymän saamiseksi on asennettava Xephyr X11 -ikkunajärjestelmä Scratchboxia käyttävään järjestelmään. Kun Xephyr on käynnissä ja yhdistettynä Scratchboxiin saadaan näkymä Maemon Hildon työpöydälle. Suorittamalla projektihakemistossa qmake-komento luodaan Makefile-tiedosto kääntämistä varten. Antamalla make-komento ohjelma käännetään. Sovelluksen voi nyt suorittaa komentorivin kautta, jolloin se aukeaa Xephyr-näkymässä.

Kääntäminen N900 laitteelle tehdään muuttamalla Scratchboxin asetuksia *sb-menu* -komennolla (ks. kuvio 9). Tämä komento avaa näkymän Scratchboxin kohde-asetuksiin, jonka kautta voidaan vaihtaa käytössä oleva kohdealusta Armel-muotoon. Syöttämällä qmake- ja make -komennot ohjelma käännetään nyt ARM-prosessiarkkitehtuurin ymmärtämään muotoon. Ohjelma on nyt valmis kopioitavaksi N900-laitteelle. Valmiit ohjelmat voidaan myös paketoita Maemon käyttämäksi Debian-paketiksi (.deb), jolloin ne asennetaan laitteeseen Maemo-ohjelmistohallintatyökalua (App Manager) käyttäen. Paketoinnilla myös määritellään mitä kirjastoriippuvuuksia ohjelmistolla on, kuten tässä luodussa ohjelmistolla on riippuvuutena Qt-kirjastot.

Ohjelmaa siirrettäessä USB-yhteydellä laitteeseen on otettava huomioon, että oletuksena käyttöjärjestelmälle näkyvä hakemistoyhteys sijaitsee laitteessa tiedostojärjestelmässä MyDocs-hakemiston alla. Kyseisessä hakemistossa ja sen alihakemistoissa ei ole ohjelmilla suoritusoikeutta. Luotu ohjelma täytyy vielä siirtää muualle laitteessa missä käyttäjällä on saatavilla suoritusoikeudet.



KUVIO 9. Scratchbox-kohteiden konfigurointi

## 4 SUORITUSKYKYMITTAUS

### 4.1 Lähtötilanne

Testissä käytettävän Nokia N900-laitteen tiedot:

- Ohjelmistoversio 1.2009.42-11
- Qt-versio 4.6.2
- Verkkoyhteys Elisan 3G-yhteydellä

Apuna mittauksessa käytetään Linux-maailmasta tuttua top-ohjelmaa, joka on myös Maemossa mukana. Ohjelmalla saadaan reaaliaikainen näkymä mitä sovelluksia on järjestelmässä ajossa. Ohjelma näyttää myös kunkin ohjelman prosessoriajankäytön ja muistinkulutuksen.

Mittaavalle top-ohjelmalle annettu komento:

```
top -b -n20 >top1.txt
```

*-b* eli *batch* määrittelyllä ohjelmalla voidaan kirjoittaa saatu tieto suoraan tiedostoon, kuten tässä tapauksessa *top1.txt* tiedostoon. Jokaisen testin jälkeen testattava ohjelma käynnistetään uudelleen ja top-komento syötetään uudelleen uudella tiedostoparametrilla. (Top manual. 2010)

Ohjelman sulavuuden mittaaminen tapahtuu silmämääräisesti. Suoritus tapahtuu root-käyttäjaoikeuksilla. Käyttöliittymän ohjaukseen käytetään N900-laitteen kosketusnäyttöä.

## 4.2 Tulokset

Ilmoitetut tulokset on otettu taulukoihin silloin, kun top-ohjelman ilmoittama prosessorin käyttö on ollut suurimmillaan. Taulukossa 2 nähdään N900-laitteessa olevan koko muistin käyttökohteet.

TAULUKKO 2. N900-laitteen muistinkäyttö testien aikana

Testi	used	free	shrd	buff	cached
1	216908K	28640K	0K	1936K	79632K
2	217628K	27920K	0K	2092K	79800K
3	218348K	27200K	0K	2368K	79640K
4	218708K	26840K	0K	2572K	79420K

Taulukon 2 sarakkeiden tunnisteet:

- **used**: Käytetty muisti
- **free**: Vapaa muisti
- **shrd**: Jaettu muisti
- **buff**: Puskuroitu muisti
- **cached**: Välimuisti.

Taulukossa 3 on N900-laitteen ilmoittama suorittimen käyttöasteet. Huomioitavaa on että neljännessä testissä systeemin suoritinkäyttö nousee muita testejä isommaksi, tämä johtuu testin 4 käyttämästä 3G-verkkoyhteydestä.

TAULUKKO 3. N900-laitteen suoritinkäyttö testien aikana

Testi	usr	sys	nice	idle
1	61.1%	7.3%	0.0%	31.4%
2	57.8%	5.5%	0.0%	35.6%
3	63.4%	4.2%	0.0%	31.8%
4	59.3%	26.2%	0.0%	13.5%

Taulukon 3 sarakkeiden tunnisteet:

- **usr**: Käyttäjän suoritinkäyttö.
- **sys**: Järjestelmän suoritinkäyttö.
- **nice**: Suorittimen nice-arvo.
- **idle**: Käyttämätön suoritinaika.

Luodun Qt Hybrid -ohjelman käyttämät resurssit ovat taulukossa 4. Muistinkäyttö on aika suurta Qt Hybrid -ohjelmassa, vertailuna N900-laitteen mukana tuleva WWW-selainohjelma käyttää muistia 14 prosenttia Youtube-videopalvelun mobiili-etusivulla.

TAULUKKO 4. Qt Hybrid -ohjelman tiedot suorituksen aikana

Testi	PID	USER	STAT	RSS	%MEM	%CPU
1	2199	root	R	25044	10.1	57.4
2	2202	root	R	25088	10.1	51.9
3	2210	root	R	25732	10.4	60.9
4	2213	root	R	26672	10.8	37.5

Taulukon 4 sarakkeiden tunnisteet:

- **PID**: Prosessin tunniste.
- **USER**: Prosessin suorittajan tunnus.
- **STAT**: Prosessin tila. R=Running, eli suorituksessa.
- **RSS**: Fyysisien muistin käyttö kilotavuissa.
- **%MEM**: Muistinkulutus prosenteissa.
- **%CPU**: Suoritinkäyttö prosenteissa.

Resurssimittauksessa esiintyneet muut huomiot on kirjattuna taulukossa 5. Huomiokohdat oli havaittu silmämääräisesti ja ”huomioitavaa”-sarakkeeseen on kerrottu miten ne vaikuttivat käyttökokemukseen.

TAULUKKO 5. Kommentit Qt Hybrid -ohjelman muista huomioista

Testi	Huomioitavaa
1	Testissä animaatio oli hieman nykivää, ei kuitenkaan häiritsevää, mutta helposti huomattavissa. Animaation kesto oli väliltä 1918 - 2160 millisekuntia.
2	Animaatio oli hieman pätkivää, samalla tavoin kuin ensimmäisessä testissä. Kuvien mukaan tuominen ei lisännyt animaation kestoja, eikä muuta verrattaessa ensimmäiseen testiin.
3	Testissä jokainen kello piirretään uudestaan sekunnin välein. Animaatiossa esiintyi pätkimistä, kellot eivät aina päivittyneet jokaisella sekunnilla.
4	Animaatio oli sulavaa. Kuvien selauksessa ei mitään havaintoja minkäänlaisesta hidastelusta.



Lopputuloksena suorituskykymittauksessa ei esiintynyt suorituskykyrajoitetta mikä kokonaan estäisi Qt Hybrid -ohjelmien käyttämiseen Maemo-alustalla. N900-laitteessa on tarpeeksi suorituskykyä Qt Hybrid -ohjelmien ajamiseen. Suorituskykymittauksessa kuitenkin esiintyi paikoitellen hidastelua etenkin JavaScript-animaatioiden käytössä (ks. Taulukko 5). Ohjelman JavaScript-to-teutukset testeissä 1 – 3 häiritsivät näkyvästi, ettei täysin sulavaa käyttökokemusta saatu. Testissä 4 samanlaisia tai muita ongelmia ei esiintynyt, vaikka kyseessä oli ainoa 3G-verkkoyhteyttä käyttävä testi.

## 5 POHDINTA

Tarkoituksena työssä oli selvittää onko Qt Hybrid -ohjelmalla suorituskykyesteitä toimiakseen Maemo 5 -käyttöjärjestelmässä. Testeissä saaduissa tuloksissa ei mitään tähän viittaavaa tullut esille. Pientä huomioitavaa esiintyi ohjelman muistinkulutuksessa ja animaatioiden piirtämisessä. Sulavaan JavaScript-animaatioiden piirtäminen ei kaikissa testeissä ollut. Syy tähän on luultavimmin N900-laitteen suorituskyvyn rajoitteet. Tulevien Qt-versioiden ja uudempien laitteiden myötä nämä pienet ongelmat luultavimmin korjaantuvat.

Qt-kehitysympäristön kanssa toimiminen Maemo-alustalla oli kokemusta kasvattava. Työn yhtenä isommista haasteista oli saada kehitetty Qt Hybrid -ohjelma toimimaan oikeassa Maemo 5 -käyttöjärjestelmää käyttävässä laitteessa. Asiaa ei helpottanut se, että laitteen ohjelmistopäivityspaikoissa Qt:n versiota 4.6.2 ei ollut vielä virallisesti vielä saatavilla. Tämän aiheutti lisää työtä, että luotu ohjelma voitiin suorittaa laitteella. Qt-ohjelmien kehitykseen on jo tullut parannuksia, sillä opinnäytetyön teon aikana julkaistiin ensimmäinen beta-versio uudesta Nokia Qt SDK:sta. Maemon versio 6:n tulevaisuus on kuitenkin epäselvä, sillä Maemo näyttäisi yhdistyvän Nokian ja Intelin yhteistyönä tulevaan MeeGo-projektiin. Maemo voi tietysti jäädä elämään pidemmäksikin aikaa avoimien lähdekoodiyhteisöjen voimin, mutta uudet Nokialta tulevat mobiilitietokoneet ovat luultavimmin pelkästään MeeGo-alustaisia.

Työn aikana kehitetty hybrid-ohjelma vastaa tekniikoiltaan mitä WWW-maailma tarjoaa tänä päivänä, eli runsasta JavaScriptin käyttöä. Jatkokehitystä ajatellen ohjelmia voisi kehittää enemmän oikean testiohjelman suuntaan, joka pystyisi itse tai kutsumalla toista ohjelmaa tarkastelemaan hybrid-ohjelman resurssien käyttöä ja raportoimaan siitä.

Internetin WWW-palveluiden käyttämät tekniikat viitoittavat tietä mihinkä hybrid-ohjelmat mahdollisesti tulevaisuudessa menevät. Mobiilimaailmaan hybrid-ohjelmat löytävät ainakin tiensä S60-alustalla toimivien WRT-widgettien muodossa. Se mitä Qt Hybrid tuo lisäksi WRT-widgetteihin verrattuna on järjestelmäriippumattomuuden, jonka uskoisi levittävän hybrid-ohjelmia laajempaan käyttöön. Yleistymisessä on otettava huomioon mobiiliohjelmistokauppojen suhtautuminen Qt Hybrid -sovelluksiin. HTML 5 -version esityksessä on suunnitteilla monia uusia parannuksia, jotka voivat tehdä hybrid-ohjelmista mielenkiintoisempia käyttäjille ja ohjelmistokehittäjille.

## LÄHTEET

CSS introduction. CSS-tyylitiedostojen tietosivusto. Viitattu 4.5.2010.  
[http://www.w3schools.com/css/css\\_intro.asp](http://www.w3schools.com/css/css_intro.asp)

Digia Finland Oy, tietoa Digia yrityksestä, Viitattu: 1.5.2010.  
<http://www.digia.com/C2256FEF0043E9C1/0/405000098?opendocument&lang=fi>

JavaScript - Mozilla Developer Center. Mozilla foundation ylläpitämä JavaScript sivusto. Viitattu 4.5.2010.  
<https://developer.mozilla.org/en/JavaScript>

Lal, R. 2010. Hybrid Application Development for Maemo N900 Device using Qt Webkit. Rajesh Lalin esitelmä hybrid-ohjelmista. Viitattu 1.5.2010.  
<http://www.slideshare.net/rajeshlal/hybrid-application-development-for-maemo-n900-device>

Maemo.org Wiki. 2010. Maemo tietosivusto. Viitattu 6.5.2010.  
<http://wiki.maemo.org/>

Nokia Suomi. 2010. N900-laitetiedot. Viitattu 7.5.2010.  
<http://www.nokia.fi/tuotteet/kaikki-puhelimet/nokia-n900>

Qt About us, Qt:n historia, Viitattu 1.5.2010.  
<http://qt.nokia.com/about/>

Qt Development Frameworks. 2010. Qt Features for Hybrid Web/Native Application Development PDF dokumentti. Viitattu: 1.5.2010.  
<http://qt.nokia.com/forms/whitepapers/reg-whitepaper-hybrid>

Qt Modular Class Library, Qt-luokkakirjastot, Viitattu 1.5.2010.  
<http://qt.nokia.com/products/library/modular-class-library>

Qt Platform, Tuetut käyttöjärjestelmälustat, Viitattu 1.5.2010.  
<http://qt.nokia.com/products/platform/platforms>

Qt Style Sheets. Qt 4.6 tyylitiedostojen tietosivusto. Viitattu 5.5.2010.  
<http://doc.trolltech.com/4.6/stylesheet.html>

Qt WebKit. WebKit:n Qt-käännöksen WWW-sivut. Viitattu 5.5.2010.  
<http://trac.webkit.org/wiki/QtWebKit>

Quantcast 2009 Mobile Web Trends Report. Mobiilin Webin kasvu vuonna 2009. Viitattu 10.5.2010.  
<http://ak.quantcast.com/docs/display/download/attachments/3080958/QCMob2009r11.pdf>

Scratchbox kotisivut. 2010. Tietoa Scratchbox-ristiinkääntäjästä. Viitattu 7.5.2010. <http://www.scratchbox.org>

State of the Mobile Web, March 2010. 2010. Opera Softwaren ilmoittamat Opera Mini-selaimen käyttöluvut. Viitattu 14.5.2010. <http://www.opera.com/smw/2010/03/>

Top manual. 2010. Top-ohjelman manuaali. Viitattu 10.5.2010. [http://linux.about.com/od/commands/l/blcmdl1\\_top.htm](http://linux.about.com/od/commands/l/blcmdl1_top.htm)

Forum Nokia Web Runtime widgets. 2010. WRT tietosivusto. Viitattu 20.5.2010. [http://www.forum.nokia.com/Technology\\_Topics/Web\\_Technologies/Web\\_Runtime/](http://www.forum.nokia.com/Technology_Topics/Web_Technologies/Web_Runtime/)