

Kesäkoodi 2006–Scribus Projektisuunnitelma

Riku Leino
riku@scribus.info

Scribus

Sisällys

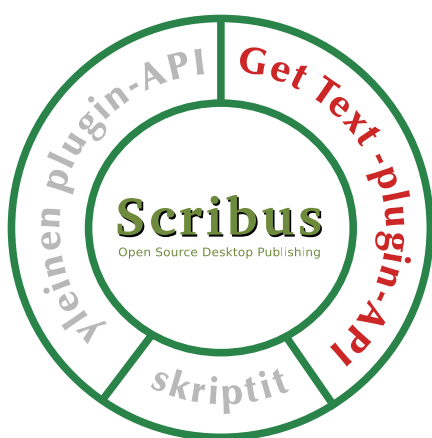
Johdanto	1
Toteutettava kokonaisuus	1
Rajapinta tekstin tuonnista vastaaville liitännäisille	1
Tekstitiedostot	3
OpenDocument Text -dokumentit	3
Teksti- ja tyylisuotimet	3
Rich Text Format -dokumentit	3
Dokumentaatio	4
Prosessin kulku	4
Kokoarvio	5
Aikataulu	5

Johdanto

Scribus on avoimen lähdekoodin taitto- ja julkaisuohjelma. Scribusin vahvuus on sen tuottamien PDF-dokumenttien korkea laatu. Scribus on suunnattu Linux-työpöytäympäristöön, mutta siitä on juuri julkaistu natiivit versiot Mac OS X:lle ja Windowsille.

Tässä dokumentissa kuvataan ohjelmakokonaisuus Scribusiin tehtäväksi Kesäkoodi 2006 -projektin puitteissa.

Toteutettava kokonaisuus



Scribusin kanssa työskennellyt tietää sen pahimmaksi puutteeksi sujuvan tekstin esittämisen, kun käsitellään suuria määriä tekstiä sisältäviä asiakirjoja. Ongelmaan on tulossa ratkaisu seuraavien kahden–neljän kehityshaaran julkaisun myötä. Tämän lisäksi uusi tekstinkäsittelystä vastaava komponentti lisää Scribusiin tuen monimutkaisille merkistöille, joissa merkit voivat limittyä päällekkäin muodostaakseen kirjainmerkin. Tällaista ratkaisua vaativia kieliä ovat muun muassa

arabia ja useat Intiassa puhuttavat kielet. Kolmas tärkeä ominaisuus, jonka uusi tekstikomponentti ratkaisee on Scribusista vielä tällä hetkellä puuttuvat merkkityylit.

Muutokset tekstinkäsittelystä vastaavaan komponenttiin ovat siinä määrin merkittäviä, että ne vaativat myös tekstin tuonnin komponenttien uudelleenkirjoitusta. Ilman sujuvaa tekstin tuontia ei ohjelmalla juurikaan ole merkitystä. Scribusin uuden tekstinkäsittelystä vastaavan komponentin ja Kesäkoodi 2006 -projektin aikataulut osuvat niin hyvin kohdalleen, että katsoin seuraavissa kappaleissa kuvattujen tekstin tuontiin liittyvien kokonaisuuksien toteuttamisen parhaaksi mahdolliseksi tavaksi edistää Scribusista Kesäkoodi 2006 -projektin puitteissa.

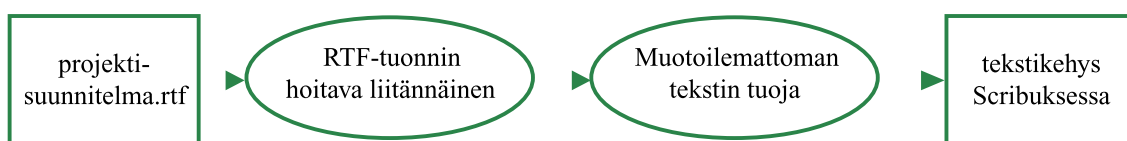
Rajapinta tekstin tuonnista vastaaville liittämisille

Muotoillun tekstin tuonti Scribusiin voidaan käsittää yhdeksi laajaksi kokonaisuudeksi, jossa teksti tuodaan Scribusiin sen tuntemattomasta tiedostomuodosta ja muunnetaan sen ymmärtämään muotoon. Tätä prosessia tukemaan toteutan rajapinnan, jonka avulla teksti ja siihen liittyvä muotoilu voidaan mahdollisimman yksinkertaisesti tuoda tekstikehykseen Scribusissa. Tekstin tuonnin hoitavan liittämisestä tehtävänä on muuntaa teksti tallennusmuodostaan rajapinnan kuvaamaan Scribusin ymmärtämään

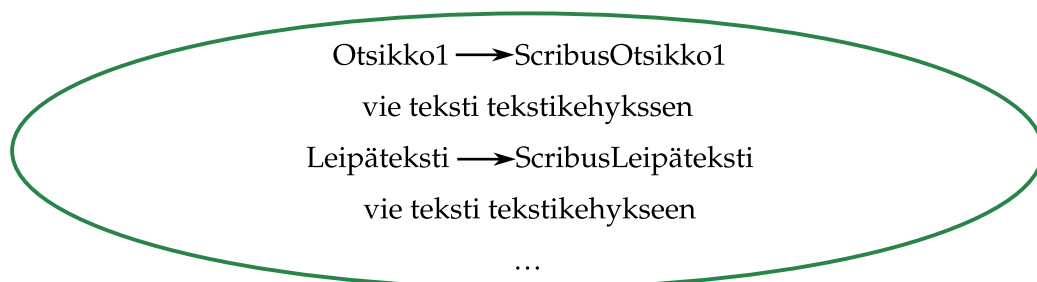
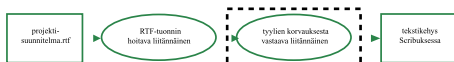
muotoon. Rajapinta toimii yhteytenä tekstin tuonnin hoitavien liitännäisten ja Scribuksen alemman tason tekstinkäsittelystä vastaavan komponentin välillä.

Tekstin tuonnista vastaavia liitännäisiä on kahden tyyppisiä. Tärkein on varmasti tuoja, joka muuntaa Scribukselle tuntemattoman tiedostomuodon sisältämän tekstin Scribuksen ymmärtämään muotoon. Toinen tuojatyyppi on tekstin ja sen muotoilun muokkauksessa toimivat suotimet. Näiden avulla voidaan jo tekstiä tuotaessa etsiä ja korvata tekstistä löytyvät typografiset virheet kuten kahden peräkkäisen välilyönnin korvaaminen yhdellä, ascii-heittomerkin korvaaminen oikealla kokolainausmerkillä sekä useita vastaavia yleisiä virheitä. Tärkein muotoiluun perustuva suodin lienee kaiken muotoilun tekstistä poistava suodin.

Julkaisun ulkonäöstä vastaava henkilö ei useinkaan halua noudattaa tekstin kirjoittajan muotoilua vaan toteuttaa omaan ja julkaisunsa tyyliin sopivan muotoilun taitto-ohjelmassa. Jotta suotimien käyttö olisi mahdollisimman tehokasta toteutan tekstin tuonnin rajapinnan niin, että sillä kirjoitettuja liitännäisiä voidaan putkittaa (liittää yhden ulostulo toisen sisäänmenoon). Tällä periaattella toimiva ketju, joka poistaa tuotavasta tekstistä kaikki muotoilut, on kuvattuna alla.



Vastaavalla tavalla voidaan ottaa käyttöön alkuperäisessä tekstin muotoilussa käytetyt tyylit. Graafisen suunnittelijan ei tarvitse antaa valtaa kirjoittajan tekemille muotoiluille vaan hän voi valjastaa tekstissä käytetyt tyylit omaan käyttöönsä putkittamalla muotoillun tekstin tuonnista vastaavan liitännäisen tyylien korvauksen hoitavaan liitännäiseen. Näin taitosta vastaava henkilö voi korvata alkuperäisessä tekstiasiakirjassa käytetyt tyylit Scribuksessa tekemillään julkaisun ulkoasua vastaavilla tyyleillä. Ketju tiedostosta tekstikehykseen on edellisen kaltainen. Toisena liitännäisenä tällöin on vain tyylien korvaamiseen erikoistunut liitännäinen.



Putkittaminen ei tuo etuja ainoastaan Scribuksella työsekenteleille graafisen alan ammattilaisille vaan myös liitännäisiä kirjoittaville ohjelmoijille. Liitännäistä kirjoitettaessa ei tarvitse huolehtia muusta kuin valitun tekstiformaatin mahdollisimman tarkasta tuonnista. Liitännäisen kirjoittajan ei esimerkiksi enää tarvitse varautua pelkän tekstin tuontiin tiedostomuodosta vaan voi luottaa, että hänen kirjoittamansa liitännäinen voidaan putkittaa muotoilun poistavaan liitännäiseen, jos käyttäjä haluaa tuoda tiedostosta ainoastaan tekstin. Parhaimmillaan hyvin toteutettu rajapinta liitännäisille houkuttelee uusia, Scribusta ennestään tuntemattomia ohjelmoijia, sen pariin.

Tekstitiedostot

Yksinkertaisin ja varmastikin käytetyin liitännäinen on tavallisen tekstitiedoston tuonnin hoitava liitännäinen. Tämä tulee olemaan ensimmäinen uusi liitännäinen uuden tekstin tuonnista vastaavan rajapinnan toteutuksen jälkeen. Tärkein huomioitava asia tekstitiedostoa tuotaessa on sen merkkikoodaus. Tekstitiedoston tuonnista vastaava liitännäinen toimii putkitettaessa minkä tahansa muun liitännäisen kanssa kaiken muotoilun poistavana suodinliitännäisenä.

OpenDocument Text -dokumentit

OpenDocument format on uusi OASIS industry consortiumin kehittämä standardi toimisto-ohjelmien tiedostojen tallennuksen ja välityksen tarpeisiin. Merkittävimmät avoimen lähdekoodin toimisto-ohjelmistot ovat ottaneet tai ottamassa standardia käyttöön oletustallennusmuotonaan. Suosio avointen ohjelmien parissa on myös syy miksi ODF on korkealla tekstin tuonnista vastaavien liitännäisten toteuttamisvaiheessa. Scribuksen nykyinen ODF-tuoja perustuu OpenOffice.org 1.0 -tiedostomuodon tuojaan eikä sitä ole tästä juuri kehitetty. Tavoitteenani on tehdä ODF-liitännäinen, joka vastaa standardia täysin Scribuksen sallimissa puitteissa.

Teksti- ja tyyliuotimet

Teksti ja tyyliuotimilla voidaan tekstiä tuotaessa suorittaa etsi ja korvaa -tyyppisiä toimintoja. Liitännäinen toimii ainoastaan putkitettuna toisen liitännäisen kanssa. Tärkeimmät liitännäisen tarjoamat toiminnot ovat etsi ja korvaa tekstile, etsi ja korvaa tyyliä sekä tyylin käyttö tekstitiedoston markatuille kohdille (Käytä kappaletyylillä *Otsikko1* kaikille kappaleille, jotka alkavat merkkijonolla "\01".)

Rich Text Format -dokumentit

Scribuksen ensimmäinen natiivi Windows-versio, julkaistiin tammikuun loppupuolella. Uuden käyttöjärjestelmän myötä tulee Scribuksen käyttäjiksi runsaasti henkilöitä, jotka käyttävät tekstin kirjoittamiseen Microsoft Wor-

dia. Rich Text Format (RTF) on Wordin tuntema tiedostomuoto. Toteuttamalla liitännäinen RTF-tekstin tuontiin saadaan myös Wordilla tuotettu muotoiltu teksti tuotua Scribukseen hävittämättä muotoiluja.

RTF tarjoaa myös toisen merkittävän edun. Se on suosittu tallennusmuoto muotoiltua tekstiä käsittelevissä ohjelmissa ja usein, jos ohjelma ei tue RTF-muotoa, saattaa sille löytyä komentorivipohjainen muuntotyökalu tiedostomuodon muuttamiseksi RTF-muotoon. Tällöin putkitusta voitaisiin käyttää siten, että tehdään liitännäinen tuntemattomalle tiedostomuodolle, joka käyttää apunaan komentorivipohjaista jo olemassa olevaa työkalua ja putkitetaan tämä RTF-tekstin tuojaan. Tällaista lähestymistapaa voi harkita tiedostomuodoille, joiden ei katsota olevan tarpeeksi tärkeitä, jotta rajallisia kehitysresursseja kannattaisi uhrata tiedostomuodon tuonnista vastaavan liitännäisen tekemiseen.

Rich Text Format on minulle tuntematon tiedostomuoto. Tästä johtuen se toteutetaan projektin viimeisenä liitännäisenä, kun kaikki muu on jo saatu valmiiksi. Tämä takaa projektin kaikkien muiden osakokonaisuuksien toteuttamisen aikataulussa ja jos aikataulusta tapahtuu lipsumista näkyy se ainoastaan tämän liitännäisen valmistumisessa. Liitännäinen toteutetaan joka tapauksessa mahtui sen toteutus kokonaisuudessaan kolmen kuukauden aikarajaan tai ei.

Dokumentaatio

Tässä esittelemälläni ohjelmakokonaisuudella on kaksi käyttäjäryhmää – ohjelman loppukäyttäjät sekä liitännäisiä kirjoittavat ohjelmoijat. Kumpiakin varten on kirjoitettava ohjeet miten tekstin tuonti toimii ja miten liitännäisiä kirjoitetaan. Ohjelmoijia varten toteutettava dokumentaatio voidaan pääosin kirjoittaa jo koodatessa, jolloin koodin joukkoon kirjoitettava API-dokumentaatio tehdään selkeäksi huomioiden tarpeet, joita Scribusta ennestään tuntemattomalla ohjelmoijalla on.

Prosessin kulku

Suunnitelmassa täysin uusia asioita minulle ovat ainoastaan RTF-tiedostomuoto sekä putkittamiseen liittyvät yksityiskohdat. Tästä johtuen merkittävä panos prosessimalleja valittaessa on aikaisemmalla kokemuksella vastaavia järjestelmiä toteutettaessa.

Laajin ja monimutkaisin vaihe kehityksessä on itse liitännäisrajapinnan toteuttaminen siten, että se tukee putkitusta ja muita edellisissä kappaleissa kuvattuja ominaisuuksia. Liitännäisten toteuttaminen samaan aikaan rajapinnan kanssa tukee rajapinnan kehitystä siten, että iteroiva kehitys rajapinnalle on helposti toteutettavissa. Rajapinnan ensimmäisen version valmistuttua toteutetaan ensimmäinen liitännäinen. Liitännäisen kehitys-

työn aikana huomioidaan rajapinnan puutteet ja palataan tarvittaessa takaisin rajapinnan toteutukseen. Tämä sykli toistuu kaikkien projektissa toteutettavien liitännäisten kanssa, jolloin rajapinta ja siihen kuuluva muu toteutus on iteroitu useaan kertaan kehitystyön aikana.

Järjestelmän testaus on suoraviivainen prosessi. Sitä varten on hankittava edustava kokoelma asiakirjoja, joita voidaan tuoda Scribukseen ja verrata näin syntynyttä tekstiä muotoiluineen samaan asiakirjaan sen ollessa avoimena sen tuottaneessa ohjelmassa. On tärkeää, että tämä toistetaan jokaisen rajapintaan kohdistuneen iteraatiokierroksen jälkeen, jotta mahdollisimman varhain havaitaan virheet, jotka ovat voineet syntyä, kun rajapinnan toteutusta on korjattu perustuen ainoastaan yhden liitännäisen kehityksen aikana havaittuihin ongelmiin.

Toteutuskielenä on C++ käyttäen Qt-ohjelmistokehystä. Ohjelman kääntämisessä ja virheiden etsimisessä on käytössä tavanomaiset Linuxille löytyvät työkalut kuten GCC ja GDB. ODF-tiedostomuodon käsittelyssä käytetään lisäksi zlib- ja libxml2-ohjelmakirjastoja.

Kokoarvio

Tuotettavan ohjelmakokonaisuuden kokoarvio perustuu aikaisempiin toteutuksiin samantyyppisistä ratkaisuista ja on ainoastaan suuntaa-antava.

rajapinta liitännäisille	2000
tekstitiedostot	200
OpenDocument Text -dokumentit	1500
teksti- ja tyyllisuotimet	1500
Rich Text Format -dokumentit	1500
yhteensä	6700 riviä

Aikataulu

Alla olevassa kuvassa on projektiin käytettävä kolmen kuukauden aika arvioituina projektin osakokonaisuuksiin kuluvin aikoina.

