

Nippu-ohjelmistotuotantoprojekti

Vaatimusdokumentti

Michael Forsström, Mikael Jokela, Ville V Nurmi, Ville A Nuutinen, Chen Zhao

Helsinki 1.7.2003

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Versiohistoria

Versio	Päiväys	Koostaja	Kommentti
0.1	8.6.2003	Mikael Jokela	Osa luvuista puuttuu vielä.
0.2	10.6.2003	Mikael Jokela	
0.3	13.6.2003	Mikael Jokela	Asiakkaan kommenttien perusteella korjattu versio
0.4	23.6.2003	Ville V Nurmi	Lopullista hyväksyntää varten valmisteltu versio, arkkitehtuuriin lisätty vastaanotto.
0.5	25.6.2003	Mikael Jokela	
0.9	27.6.2003	Mikael Jokela	FTR-kokouksessa korjattu versio
1.0	1.7.2003	Mikael Jokela	Jäädytetty ja asiakkaan hyväksymä versio, ei muutoksia 0.9:sta

Sisältö

1	Johdanto	1
2	Arkkitehtuuri ja toiminnallisuus	2
2.1	Yleinen arkkitehtuuri	2
2.2	Käyttöliittymä	3
2.3	Vastaanotto	3
2.4	Analysoija	3
2.5	Tulostushallinta	4
2.6	Tietoturvaominaisuudet	5
3	Ulkoiset liittymät	6
3.1	Käyttöliittymä	6
3.1.1	Peruskäyttäjän liittymä	6
3.1.2	Pääkäyttäjän liittymä	7
3.1.3	Visualisointiliittymä	8
3.2	Asetustiedostot	9
4	Rajoitteet	10
4.1	Ympäristö	10
4.2	Valmiit komponentit	10
4.3	Vähimmäistoiminta	10
5	Testaus	12
5.1	Katselmukset	12
5.2	Yksikkötestaus	12
5.3	Integrointitestaus	12
5.4	Järjestelmätestaus	13
Liitteet		
1	Käyttötapauksia	I
2	Sanasto	IV

1 Johdanto

Tässä vaatimusdokumentissa **esitellään Nippu-ohjelmistotuotantoprojektin tuotteena syntyvä Nippu-tulostuspalvelinohjelmisto**. Dokumentissa muun muassa määritellään ohjelmistoon kohdistuvat vaatimukset sekä ohjelmiston keskeinen toiminnallisuus ja rajoitukset. Nippu on osa Helsingin yliopiston tietojenkäsittelytieteen laitoksen ohjelmistotuotantoprojekti-kurssia.

Tämä dokumentti on tarkoitettu **projektin suunnitteluvaiheen pohjaksi**. Asiakas voi käyttää dokumenttia tuotettavan ohjelmiston kirjallisenä kuvauksena, jonka perusteella vaatimusten toteutuminen voidaan todentaa lopputuotteesta. Projektiin liittyvät hallinnolliset seikat on kirjattu projektisuunnitelmaan.

Nipun keskeisin tarkoitus on **saavuttaa tehokkuutta suorittamalla samankaltaiset palvelupyynnöt yhdessä**: jokaisen käyttäjän ei kannata yksitellen etsiä ja noutaa omia tulosteitaan, vaan Nipun avulla pyritään yhteiseen töiden noutoon tulostamalla fyysisesti toistensa läheltä tulleet työt järjestyksessä peräkkäin – yhdessä nipussa. Lisäksi pyritään kuormantasauksen avulla estämään yksittäisten tulostusjonojen tukkeutuminen ja lyhentämään käyttäjän odotusaikaa.

Nippua on tarkoitus käyttää Helsingin yliopistossa **koostavan ajoitusmenettelyn ja sen kannattavuuden tutkimisessa**. Ohjelmisto ei tule olemaan sellaisenaan valmis tuotantokäyttöön, mutta sitä voitaneen myöhemmin muokata tuotantokäyttöönkin sopivaksi. Nippu käyttää **Terttu-ajoittajaydintä**, joka on aikaisemmin tuotettu Helsingin yliopistossa ohjelmistotuotantoprojektina.

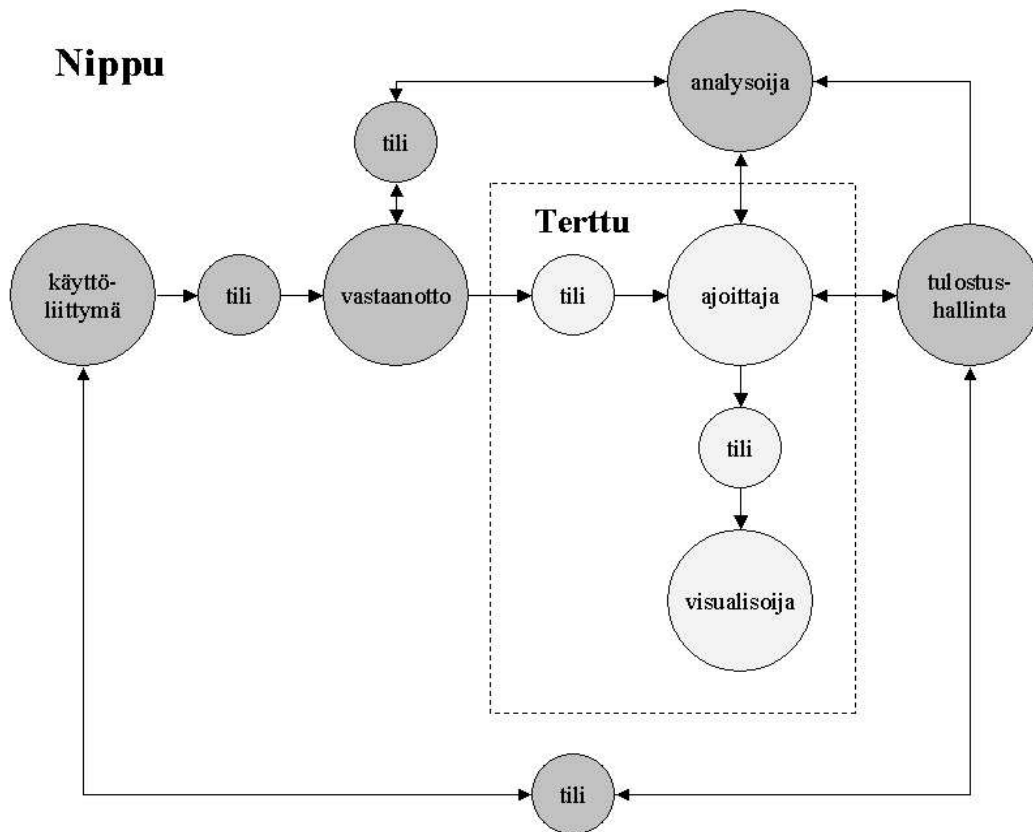
Liitteen 2 sanastossa selitetään tässä dokumentissa esiintyviä käsitteitä.

2 Arkkitehtuuri ja toiminnallisuus

Tässä luvussa esitellään Nipun yleinen arkkitehtuuri ja ohjelmistokomponentit. Kunkin Nippu-projektissa toteutettavan komponentin toiminnallisuus esitellään vielä erikseen.

2.1 Yleinen arkkitehtuuri

Nippu käyttää Tertun **ajoittajaa**, **visualisoijaa** ja **tietoliikennekomponenttia (tili)**. Näistä tärkein on ajoittaja, joka Nipun **analysoijan** kanssa muodostaa järjestelmän ytimen. Analysoijan lisäksi toteutetaan **käyttöliittymä**, tulostustöiden niputtamista tehostava **vastaanotto** sekä tulostinten toimintaa emuloiva **tulostushallinta**. Tertun **visualisoijan** avulla voidaan seurata ajoittajan toimintaa. Komponentit on merkitty kuvaan 1.



Kuva 1: Yleiskaavio Nipun arkkitehtuurista ohjelmistokomponenttitasolla. Nuolet kuvaavat tiedon kulkua järjestelmässä. (Pelkästään kuittaustyyppistä tietoa sisältävät tiedon kulkusuunnat on selvyiden vuoksi jätetty merkitsemättä.)

Tulostustyöt annetaan **käyttöliittymässä**, josta ne toimitetaan tietoliikennekomponentin kautta vastaanottoon. **Vastaanotossa** työt odottavat puskuroituina tietyn ajan muita sellaisia tulostustöitä, jotka voitaisiin mahdollisesti yhdistää samaksi nipuksi. Vastaanotto päättee **analysoijasta** saatujen tietojen perusteella, mitkä työt tulevat todennäköisesti yhdistymään. Seuraavaksi työt siirtyvät **ajoittajaan**, joka arvottaa niitä analysoijan avulla. Ajoittaja lähettää työt **tulostushallintaan** sopivassa järjestyksessä. Tulostushallinta siirtää saamansa tulostustyöt sopivaan tulostusjonoon ja lähettää ilmoituksen käyttöliittymälle, kun tietty työ on tulostettu. Tulostushallinta välittää analysoijalle tietoja tulostinten sijainneista.

2.2 Käyttöliittymä

Tulostustyöt annetaan järjestelmään käyttöliittymän kautta. Käyttöliittymä siirtää työt edelleen vastaanottoon ja näyttää tietoja muun muassa tulostusjonojen tilasta. Käyttöliittymän ominaisuuksista kerrotaan tarkemmin luvussa 3.1.

2.3 Vastaanotto

Vastaanoton tehtävä on **tehostaa samankaltaisten tulostustöiden yhdistelyä**. Se pitää kirjaa odottavista tulostustyönipuista. Uuden tulostustyön saapuessa vastaanotto tarkistaa suoraan analysoijalta, tulisiko tämä työ todennäköisesti yhdistymään vastaanotossa jo olevien, odottavien nippujen töihin. Jos näin näyttää käyvän, tulostustyö siirretään sopivaan odottavaan nippuun. Muussa tapauksessa se muodostaa uuden odottavan nippunsa. Niput odottavat puolittuvan summasarjan mukaisella tavalla (katso luku 3.1.2).

Tulostustyöt **lähetetään ajoittajalle**, kun odotusaika on kulunut. Jos ajoittaja ei ole käynnissä tai lähetys muusta syystä epäonnistuu, työt jäävät vastaanottoon, joka yrittää lähetystä myöhemmin uudelleen. Varsinaiset tulostustyöniput luodaan vasta ajoittajassa, joka ei välttämättä noudata vastaanoton muodostamaa tulostustöiden jakoa odottaviin nippuihin. Jos ajoittajaa ei saada yhdistämään saman käyttäjän tulostustöitä riittävän luotettavasti, vastaanotto voi myös yhdistää saman käyttäjän tulostustöitä analysoijan avulla.

2.4 Analysoija

Analysoijan tehtävä on luoda **tulostustyöstä kuvaus N-ulotteiseen koordinaatistoon**. N on vakio, joka määräytyy Nippu-projektin suunnitteluvaiheessa. Analysoijan tulee myös

arvioida töitä ja kertoa niiden kustannus yksittäin suoritettuna tai yhdistettyinä sekä tarvittaessa yhdistää kaksi tulostustyötä yhdeksi uudeksi työksi. Töiden yhdistelyssä on päätettävä niiden järkevistä tulostusjärjestyksestä. Töiden yhdisteestä pitää myös kyetä luoda kuvaus N-ulotteiseen koordinaatistoon.

N-ulotteisen avaruuden koordinaatistossa käytetään parametreina esimerkiksi käyttäjätunnusta, käyttäjän fyysistä sijaintia 3-ulotteisella koordinaatistolla ilmaistuna, tulostustyön sivumäärää, mahdollista pakotettua tulostimen valintaa sekä mahdollisesti joitakin näistä johdettuja parametreja. Analysoijan laskiessa kustannusta kahden työn yhdistämiselle otetaan huomioon samat parametrit asetustiedostossa määritellyillä painotuksilla. Haluttaessa tietty parametri voidaan jättää myös ottamatta huomioon. Parametrien tärkeysjärjestys on oletuksena seuraava:

1. Mahdollinen pakotettu tulostimen valinta.
2. Käyttäjätunnus. Työt pyritään yhdistämään aina, eli kustannus pienenee aina.
3. Käyttäjien fyysinen sijainti. Suositaan käyttäjiä, jotka sijaitsevat lähekkäin. Tässä huomioidaan myös tulostinten sijainti käyttäjiin nähden.
4. Töiden sivumäärä ja siitä laskettu tulostusnopeus parhaaksi katsotulla tulostimella.

2.5 Tulostushallinta

Tulostushallinta **käsittelee ajoittajalta saamiaan tulostustöitä**. Työt voivat olla yksittäisiä tulostustöitä tai analysoijan tekemiä tulostustöiden yhdistelmiä – nippuja. Tulostushallinta pitää kirjaa tulostimista ja niiden jonoista. Se siirtää tulostustyön tai tulostustyönipun sopivimpaan tulostusjonoon ja ilmoittaa käyttöliittymälle, kun tietyn työn tulostus on valmis. Tulostushallinta päättää parhaan jonon työn sivumäärän ja tulostinten tulostusnopeuksien sekä fyysisten sijaintien perusteella. Tulostushallinta ei enää muokkaa tulostusjonojen sisältöä vaan voi ainoastaan lisätä uusia töitä jonojen perään; ajoittaja ja analysoija ovat jo suorittaneet ajoituksen ja yhdistelyn.

Tulostinkohtaisissa jonoissa ei saa olla enempää kuin **yksi tulostustyönippu jonoa kohti**, sillä jonottaminen tapahtuu dynaamisesti ajoittajassa. Jos kaikilla tulostimilla on töitä jonoissa, ajoittajan käsittelyssä olevat tulostustyöt pyritään pitämään ajoittajassa, kunnes jokin tulostusjono tyhjenee. Näin ajoittajajydintä saadaan hyödynnettyä mahdollisimman tehokkaasti. Poikkeus tähän sääntöön ovat tulostustyöt, jotka on pakotettu jollekin tulosti-

melle. Tällaiset työt saadaan lisätä epätyhjään tulostusjonoon. Myös kaikki tietylle tulostimelle pakotetun tulostustyön kanssa samaan nippuun päätyneet muut tulostustyöt saadaan lisätä epätyhjään tulostusjonoon. Nippuun pyritään jättämään mahdollisuus säätää ajoituksen toimintaa tämäntyyppisissä tilanteissa.

On mahdollista, että **tietylle tulostimelle pakotetut tulostustyöt** valmistuvat ruuhkaisessa toimintaympäristössä keskimäärin muita töitä hitaammin. Etukäteen tehty tulostimen valintaan rajoittaa merkittävästi tulostushallinnan mahdollisuuksia sijoittaa työ vapaalle tulostimelle. Lisäksi tulostimen valinta etukäteen rajoittaa töiden ryhmittelyä ainakin silloin, kun saman tulostajan peräkkäiset työt pakotetaan eri tulostimille.

2.6 Tietoturvaominaisuudet

Mihinkään Nipun komponentteihin **ei sisälly sellaisia pääsynvalvonta- tai tietoturvaominaisuuksia**, joita vaadittaisiin todellisessa verkkoympäristössä. Koska tietoturvaominaisuudet on pääosin sivuutettu ohjelmiston suunnittelun alkuvaiheessa, tuotteen mahdollinen laajentaminen tuotantokäyttöön sopivaksi saattaa aiheuttaa ennalta arvaamattomia tietoturvaongelmia. Tämän takia tuotetta ei ole syytä edes testata tuotantokäytössä olevassa verkossa varmistamatta, että muut verkon käyttäjät eivät voi ottaa yhteyttä ohjelmiston tarjoamiin palveluihin.

Peruskäyttäjät voivat ottaa Nippuun yhteyden mistä tahansa verkkoasemasta ilman todennusta. Nippu sallii pääkäyttäjän yhteyden vain siitä verkkoasemasta, jossa ajoittaja on käynnissä. Nipun tietoliikenneyhteyksiin ei liity todennusta eikä salausta, mutta yhteyksien todennus ja salausta on mahdollista myöhemmin erikseen toteuttaa jollain ulkopuolisella tuotteella.

3 Ulkoiset liittymät

Tässä luvussa määritellään Nippu-projektissa toteutettavien ohjelmistokomponenttien ulkoiset liittymät. Ulkoisia liittymiä on kaksi: käyttöliittymä ja asetustiedostot.

3.1 Käyttöliittymä

Toteutettavan ohjelmiston tulostinten **toiminta esitetään visuaalisesti**. Visualisoinnin avulla tutkitaan, miten ohjelmisto toimii. Käyttöliittymässä esitetään tulostukseen lähetettävät työt tietoineen sekä eri tulostinten tilat. Oleellista on nähdä, mikä työ on tulostumassa milläkin tulostimella. Käyttöliittymä tehdään tarkoituksella tutkimuskäyttöön, joten se eroaa huomattavasti tavallisesta tulostuskäyttöliittymästä.

Käyttöliittymä toteutetaan yhtenä ohjelmakomponenttina, joka sisältää kaksi erilaista liittymää: peruskäyttäjän liittymä ja pääkäyttäjän liittymä. Kumpaankin sisältyy käyttäjän halutessa visualisointiliittymä. **Käyttötapauksia** esitetään liitteessä 1.

3.1.1 Peruskäyttäjän liittymä

Kun peruskäyttäjä käynnistää ohjelman, hän voi määrittellä komentorivillä **käyttäjätunnuksensa**. Käyttäjän **sijaintikoordinaatit** haetaan käyttäjätunnuksen perusteella erillisestä asetustiedostosta. Koordinaatit voivat olla esimerkiksi 3-ulotteisessa koordinaatistossa muodossa (x, y, z), jossa y on käytävän numero ja z on kerroksen numero. Siis esimerkiksi jos tulostaja olisi 5. kerroksessa käytävällä 3 kohdassa 2, hänen koordinaattinsa olisivat (2, 3, 5).

Käyttöliittymässä annetaan **tulostustyön sivumäärä ja tiedoston nimi**. Nämä ovat pakollisia syötettäviä parametreja. Halutessaan käyttäjä voi pudotusvalikosta pakottaa tulosteen sa tietylle tulostimelle. Peruskäyttäjän liittymässä näkyy käyttäjätunnus, jota ei voi kuitenkaan muuttaa. Liittymässä on myös painike työn lähettämistä varten.

3.1.2 Pääkäyttäjän liittymä

Pääkäyttäjän liittymä sisältää samanlaisen näkymän yksittäisten töiden lähettämistä varten kuin peruskäyttäjällä. Ainoa ero on, että pääkäyttäjä voi vapaasti **vaihtaa käyttäjä-tunnusta** pudotusvalikosta. Näin pääkäyttäjä voi lähettää yksittäisiä tulostustöitä millä tahansa käyttäjätunnuksella.

Peruskäyttäjän näkymän lisäksi pääkäyttäjällä on näkymä, jossa voi **lähettää satunnaisia tulostustöitä**. Pääkäyttäjä voi määrittää käyttöliittymässä, kuinka paljon testimateriaalia tuotetaan. Hän voi antaa esimerkiksi välin [10, 30], jossa tulostettavan työn koko sivumäärässä laskettuna on oltava. Lisäksi voidaan antaa lähetystiheysväli. Lähetystiheys on aika (sekunteina), jonka välein satunnaisia testidokumentteja lähetetään tulostettavaksi. Jos lähetystiheysväli olisi esimerkiksi [5, 20], niin ohjelma arpoisi jokaisen satunnaisen tulostustyön jälkeen ajan kyseiseltä väliltä ja lähettäisi seuraavan satunnaistulostustyön arvotun ajan jälkeen. Jos lähetystiheydestä annetaan vain välin ensimmäinen arvo (tai molemmat arvot annetaan samoina), niin ohjelma lähettää satunnaisia töitä kyseisen ajan välein vakiotiheydellä.

Pääkäyttäjän satunnaistyönäkymässä on pudotusvalikko, josta voi valita käyttäjätunnuksen. Silloin satunnaistyöt lähtevät koko ajan saman käyttäjän nimissä. Näin on mahdollista tutkia, kuinka saman käyttäjän useat työt sijoittuvat. Jos käyttäjätunnusvalikosta valitaan vaihtoehto "satunnainen", järjestelmä arpoo jokaiselle satunnaistyölle käyttäjän erikseen. Tällöin voidaan simuloida yleistä tulostuspalvelimen kuormitusta. Satunnaistyöt on mahdollista pakottaa tietyille tulostimelle yksittäisten tulostustöiden tapaan.

Käyttöliittymässä on painike, jolla satunnaistöiden lähettämisen voi kytkeä päälle ja vastaavasti pois päältä. Satunnaistöitä lähetetään vasta päälle kytkemisen jälkeen. Kun satunnaistöitä ei enää haluta tuottaa, on painikkeella kytkettävä satunnaistöiden tuottaminen pois päältä.

Pääkäyttäjän liittymä sisältää painikkeen, jolla **ajoittajan toiminnan voi pysäyttää** ja jälleen käynnistää. Jos painike on valittuna, järjestelmään voi edelleen normaalisti syöttää lisää tulostuspyyntöjä. Tällöin töitä ei kuitenkaan poistu järjestelmästä, jolloin järjestelmään saadaan asetettua haluttu kuorma. Järjestelmä saattaa itse automaattisesti pysäyttää ajoittajan esimerkiksi odottaessaan tulostusjonojen tyhjentymistä. Tällaisessa tilanteessa tämän

käyttöliittymän tarjoama ajoittajan pysäytyspainike ei ole käytettävissä.

Pääkäyttäjän liittymässä on kenttä, johon voi syöttää **maksimiviiveen palvelun saannille**. Oletetaan, että viiveeksi on asetettu 10. Tällöin järjestelmä odottaa $(1/2)*10$ sekuntia = 5 sekuntia työn saamisen jälkeen sellaisia tulostustöitä, joita voitaisiin yhdistellä ensiksi saadun kanssa. Jos tämän ajan sisällä uusia sopivia töitä ei tule, järjestelmä siirtää yksittäisen työn eteenpäin. Jos taas 5 sekunnin aikana tulee uusi tulostuspyyntö, joka näyttäisi mahdollisesti sopivan samaan nippuun ensiksi saadun työn kanssa, järjestelmä ei siirräkään ensimmäistä työtä eteenpäin heti vaan yhdistää nämä kaksi työtä. Tämän jälkeen odotetaan mahdollisia uusia tulostustöitä $(1/4)*10$ sekuntia = 2,5 sekuntia. Jos uusia sopivia tulostustöitä tulee taas, odotetaan seuraavalla kerralla enää $(1/8)*10$ sekuntia. Samaa nippuun mahdollisesti sopivia töitä siis odotetaan maksimiviiveajan kuluessa jatkuvasti vähemmän aikaa. Odotusaikojen muodostaman summan arvo on enintään asetettu maksimiviive. Yllä kuvatussa esimerkissä $(1/2)*10 + (1/4)*10 + (1/8)*10 + \dots$ sekuntia = 10 sekuntia. Yksi tulostaja tai niputuksen kannalta lähekkäinen tulostajaryhmä ei siis voi kaapata järjestelmää loputtomasti haltuunsa.

3.1.3 Visualisointiliittymä

Peruskäyttäjän ja pääkäyttäjän liittymät sisältävät saman tulostimia visualisoivan liittymän. Visualisointiliittymä **esittää järjestelmän tulostimet**, joita voi olla enintään kymmenen kappaletta. Lisäksi esitetään kaikki tulostinjonot, joista näkyvät jokaiseen työhön liittyvät käyttäjätunnus, työtunnus ja sivumäärä.

Käyttäjä voi **napsauttaa hiirellä jonossa näkyvää työtä**, jolloin näytetään tarkempia tietoja kyseisestä tulostustyöstä esimerkiksi ponnahdusikkunassa. Jos käyttäjä napsauttaa tulostinta, järjestelmä kertoo tarkemmin tulostimen jonosta. Toteutus voitaisiin tehdä esimerkiksi erillisenä ponnahdusikkunana, jossa näytetään kaikki kyseisellä tulostimella olevat työt tietoineen. Ikkunan näkymä voisi olla muuttumaton, jolloin käyttäjän olisi helpompaa tutkia jonon sisältöä. Ikkunassa voisi olla painike, josta saisi ikkunassa näkyvien tulostustöiden automaattisen päivityksen päälle.

3.2 Asetustiedostot

Järjestelmän **käyttäjiä ja tulostimia koskevat ympäristöasetukset** määritellään asetus-tiedostossa. Jokaisesta tulostimesta tiedostoon merkitään tulostimen tunnus ja nopeus sekä sen sijainti kolmiulotteisessa koordinaatistossa (x, y, z). Jokaisesta käyttäjästä tiedostoon kirjataan käyttäjätunnus ja vakiosijaintikoordinaatit (x, y, z). Lisäksi tiedostoon kirjataan, kuinka monta työtä järjestelmään voidaan enintään ottaa. Järjestelmä toteutetaan siten, että se toimii ainakin sadalla samanaikaisella tulostustyöllä. Samanaikaisten töiden enimmäismääräksi voi määritellä myös enemmän kuin sata, mutta tällöin järjestelmän suorituskyvyn mielekkyyttä ei enää taata.

Ympäristöasetuksia muutetaan käsin asetustiedostoa muokkaamalla. Käyttöliittymässä ei siis hallita tulostimia tai käyttäjien sijaintikoordinaatteja. Käyttöliittymän täytyy kuitenkin lukea nämä tiedot kyseisestä asetustiedostosta. Nipun asetustiedostoissa pyritään käyttämään samantapaista syntaksia kuin Tertun asetustiedostoissa.

4 Rajoitteet

Tässä luvussa esitellään Nippu-projektiin liittyvät rajoitteet.

4.1 Ympäristö

Nipun **pääasiallinen ohjelmointikieli on Java** (Java 2 versio 1.4.1). Ulkoiset rajapinnat ovat Javan mukaisia. Lisäksi käytetään mahdollisesti C-kieltä. Toteutusympäristö on Linux-käyttöjärjestelmä. Nämä vaatimukset täyttyvät Helsingin yliopiston tietojenkäsittelytieteen laitoksen tietokoneissa projektin ollessa käynnissä.

4.2 Valmiit komponentit

Nipun toteutuksessa käytetään **Tertun ajoittajaominaisuuksia ja ohjelmisto-arkkitehtuuria**. Nipussa voidaan tarvittaessa käyttää myös muita valmiita komponentteja. Terttuun liittyvät suunnittelu- ja toteutusratkaisut saattavat aiheuttaa rajoitteita Nipun toiminnallisuudelle. Ennalta tunnettuja Tertun mahdollisesti aiheuttamia ongelmia on kaksi: Ensinnäkin pieni joukko hyvin erilaisia ajoitettavia töitä ei tunnu koskaan tulevan suoritetuksi, ellei ajoittajan parametreja ole asetettu liioitellun salliviksi. Toiseksi Terttu vaikuttaa pitävän muutaman sekunnin viiveen ajoituspäätöstensä välissä.

4.3 Vähimmäistoiminta

Tulostajan **odotusaika tulosteiden valmistumiselle** ei saa olla keskimäärin suurempi kuin mitä se käytännössä olisi samassa toimintaympäristössä, jos tulostajan tulisi määrätä jokaiselle tulosteelleen pakotettu tulostin eikä keskitettyä ajoitusta olisi. Nipun toiminta saa kuitenkin aiheuttaa pienen viiveen verrattuna sellaiseen toimintaympäristöön, jossa ajoittajaa ei ole. Osa viiveen pituudesta määräytyy pääkäyttäjän asettamasta maksimiviiveestä palvelun saannille: jos tulostustöitä tulee harvakseltaan, järjestelmä joutuu usein turhaan odottamaan syntyviin nippuihin sopivia lisätöitä.

Nipun on ainakin joissakin selvissä tapauksissa kyettävä **ryhmittelemään tulostustöitä** siten, että yhtä tulostinta tarkasteltaessa kunkin käyttäjän työt tulostuvat peräkkäin. Siis muiden käyttäjien töitä ei saisi tulostua niiden välissä. Vastaavalla tavalla ryhmittelyä tulee

tapahtua myös tulostajien sijainnin suhteen: yhtä tulostinta tarkasteltaessa **lähekkäin sijaitsevien tulostajien antamat tulostustyöt** on ainakin joissakin selvissä tapauksissa suoritettava peräkkäin siten, että kaukana sijaitsevan tulostajan työ ei tulostu niiden välissä.

Nipun täytyy soveltua **toimintaympäristöihin**, joissa on enintään kymmenen tulostinta ja enintään sata samanaikaista tulostustyötä. Nipun **käyttöliittymän tulee olla selkeä** ja tehokas tällaisen toimintaympäristön hallitsemiseen ja siinä toimimiseen. Nipun tulee **prosessina toimia niin tehokkaasti**, että se ei itsessään aiheuta samanaikaisten tulostustöiden määrän mukaan kasautuvia tai muulla tavalla erityisen häiritseviä viiveitä tulostamiseen.

5 Testaus

Tässä luvussa esitellään ohjelmiston testauksessa ja laadunvarmistuksessa käytettäviä menetelmiä. Nippu-projektiin liittyvät keskeiset dokumentit tarkistetaan ennen niiden hyväksymistä. Toteutettavat komponentit testataan yksitellen yksikkötestauksena. Lisäksi komponenttien yhteistoiminnan muodostaman järjestelmän toiminnallisuus testataan kokonaisuutena. Lopuksi tullaan toteamaan, että Nippu täyttää sille asetetut vaatimukset.

5.1 Katselmukset

Keskeiset dokumentit, kuten vaatimus-, suunnittelu- ja testausdokumentti, **tarkistetaan suunnitteluvirheiden varalta** ennen niiden hyväksymistä. Näin pyritään havaitsemaan ongelmatilanteet mahdollisimman aikaisessa vaiheessa. Dokumenttien tarkistus suoritetaan katselmuksina. Tarkastettava dokumentti jaetaan katselmusta varten tietyille henkilöille tai mahdollisesti kaikille projektin jäsenille. Dokumentti täytyy luetuttaa ainakin yhdellä sellaisella henkilöllä, joka ei ole kirjoittanut sitä. Jos kaikki projektin jäsenet ovat kirjoittaneet dokumenttia, jaetaan katselmoinnin vastuualueet heille siten, että kukaan ei vastaa itse kirjoittamansa dokumentin osan tarkastamisesta. Nämä tietyt henkilöt lukevat dokumentin ja esittävät sen perusteella dokumentin keskeistä sisältöä koskevia huomioita ja korjaus ehdotuksia kokouksessa, jossa myös dokumentin vastuuhenkilö on läsnä.

5.2 Yksikkötestaus

Komponentteittain tehtävässä yksikkötestauksessa pyritään soveltamaan **mustalaatikkotestausta** sekä erityisen monimutkaisten komponenttien kohdalla lasilaatikkotestausta. Ensisijaisesti testataan **analysoijaa ja tulostushallintaa**. Toissijaisesti testataan ohjelmiston muita osia, esimerkiksi käyttöliittymää.

5.3 Integrointitestausta

Integrointitestausta kattaa komponenttien yhteistoiminnan. Tällä testauksella pyritään varmistamaan siitä, että **komponenttien välinen viestintä toimii virheettömästi**. Integrointitestausta perustana on **mustalaatikkotestausta**. Lisäksi tarkistetaan, että Tertun arkkitehtuuriin kuuluvat useat eri prosessit ovat loppukäyttäjän kannalta järkevästi hallittavissa

sekä pienimuotoisessa, simuloitussa koekäytössä että usean eri fyysisen verkkoaseman välillä tapahtuvassa käytössä.

5.4 Järjestelmätestaus

Asiakas Olli Lahti suorittaa **vaatimusten täyttymisen** varmistavan järjestelmätestauksen. Ensisijaiset perusteet projektin hyväksymiselle ovat lukkiutumatta toimiminen sadalla samanaikaisella tulostustyöllä sekä seuraavan skenaarion toteutuminen: Yksi käyttäjä lähettää viisi samanlaista, alle kymmenen sivun tulostustyötä kahden sekunnin sisällä. Samanaikaisesti muut käyttäjät lähettävät yhteensä kymmenen satunnaista tulostustyötä. Ensin mainitun yhden käyttäjän kaikkien viiden työn tulee tulostua peräkkäin samalla tulostimella.

Liite 1. Käyttötapauksia

Koska ohjelmistoa tullaan käyttämään vain tutkimukseen, tässä esiteltävät käyttötapaukset eivät edusta todellista tulostusta vaan erilaisia simuloituja tilanteita. Käyttötapauksiin on havainnollisuuden vuoksi lisätty muun muassa henkilöiden ja tulostinten nimiä. Näitä nimiä ei ole tuotteessa mitenkään kiinnitetty, vaan ne ovat vain esimerkkejä. Jokaisessa käyttötapauksessa oletetaan, että tarvittavat asetukset on etukäteen säädetty kyseiseen käyttötapaukseen sopivasti.

Toimistoympäristön asetus

Liisa haluaa alustaa testattavan toimintaympäristön tutkiakseen koostavaa ajoitusmenettelyä. Hän haluaa, että toimistossa on kuusi tulostinta. Liisa luo tekstieditorilla asetustiedoston, johon hän kirjaa kuusi tulostinta, niiden tunnukset, nopeudet ja sijaintikoordinaatit. Liisa tallentaa tiedoston. Ympäristö on käytettävissä ytimen käynnistyksen jälkeen.

Peruskäyttäjän tulostus (tapaus a)

Pekka haluaa simuloida tilannetta, jossa hän on juuri saanut valmiiksi tekstinsä huomista esitelmää varten ja haluaa tulostaa sen nyt. Pekka ottaa yhteyden järjestelmään peruskäyttäjänä. Pekka antaa tulostettavan työn nimeksi esitelma.ps. Lisäksi Pekka kirjoittaa työn koko -kenttään luvun 25. Tämän jälkeen hän painaa tulosta-painiketta, jolloin työn suoritus alkaa. Pekka tarkkailee näytöltä työnsä siirtymistä lopulliselle tulostimelle ja huomaa, että tulostin ps8 tulostaa hänen työtään.

Peruskäyttäjän tulostus (tapaus b)

Pekka haluaa simuloida tilannetta, jossa hän on juuri saanut valmiiksi tekstinsä huomista esitelmää varten ja haluaa tulostaa sen nyt pakotetusti tulostimelle ps4. Pekka ottaa yhteyden järjestelmään peruskäyttäjänä. Hän antaa tulostettavan työn nimeksi esitelma.ps. Lisäksi Pekka kirjoittaa työn koko -kenttään luvun 25. Tämän jälkeen hän valitsee tulostinvalikosta tulostimen ps4 ja painaa tulosta-painiketta, jolloin työn suoritus alkaa. Pekka tarkkailee näytöltä työnsä siirtymistä tulostimelle ps4.

Satunnaistulostustöiden lisääminen (tapaus a)

Liisa haluaa simuloida satunnaista tulostuspalvelimen käyttöä. Hän ottaa yhteyden järjestelmään pääkäyttäjänä. Hän asettaa järjestelmään työn koon alarajaksi 20 ja ylärajaksi 40. Tulostustiheydeksi Liisa antaa 5, sillä hän haluaa, että järjestelmä lähettää satunnaistulostustyön viiden sekunnin välein. Käyttäjätunnukseksi Liisa valitsee valikosta "satunnainen", sillä hän haluaa, että satunnaistyöt tulevat eri käyttäjiltä. Tiedot annettuaan hän painaa käynnistä-painiketta, jolloin järjestelmä aloittaa töiden lähettämisen. Hän huomaa, että ensimmäinen työ on kooltaan 28 sivua. Toinen työ ilmestyy viiden sekunnin kuluttua ja on kooltaan 32 sivua. Liisa tutkii jonkin aikaa tarkkaavaisesti jonojen muodostumista ja lopettaa sen jälkeen satunnaistulostustöiden tuottamisen napsauttamalla satunnaistöiden pysäytyspainiketta.

Satunnaistulostustöiden lisääminen (tapaus b)

Liisa haluaa simuloida tulostuspalvelimen käyttöä satunnaisilla omilla tulosteillaan. Hän ottaa järjestelmään yhteyden pääkäyttäjänä. Hän asettaa järjestelmään työn koon alarajaksi 20 ja ylärajaksi 40. Tulostustiheyden alarajaksi Liisa antaa 5 ja ylärajaksi 15, sillä hän haluaa, että järjestelmä lähettää satunnaistulostustyön 5–15 sekunnin välein. Käyttäjätunnukseksi Liisa valitsee valikosta oman käyttäjätunnuksensa. Tiedot annettuaan hän painaa käynnistä-painiketta, jolloin järjestelmä aloittaa töiden lähettämisen. Hän huomaa, että ensimmäinen työ on kooltaan 25 sivua. Toinen työ ilmestyy 12 sekunnin kuluttua ja on kooltaan 39 sivua. Liisa tutkii jonkin aikaa tarkkaavaisesti jonojen muodostumista ja lopettaa sen jälkeen satunnaistulostustöiden tuottamisen napsauttamalla satunnaistöiden pysäytyspainiketta.

Satunnaistulostustöiden lisääminen (tapaus c)

Liisa haluaa simuloida tulostuspalvelimen käyttöä suurella kuormituksella. Hän ottaa järjestelmään yhteyden pääkäyttäjänä. Hän asettaa järjestelmään työn koon alarajaksi 30 ja ylärajaksi 50. Tulostustiheyden alarajaksi Liisa antaa 5 ja ylärajaksi 15, sillä hän haluaa, että järjestelmä lähettää satunnaistulostustyön 5–15 sekunnin välein. Käyttäjätunnukseksi Liisa valitsee valikosta "satunnainen", sillä hän haluaa, että satunnaistyöt tulevat eri käyttäjiltä. Liisa syöttää lisäksi maksimiviiveeksi palvelun saannille 8. Tiedot annettuaan Liisa pysäyttää ajoittajan napsauttamalla ajoittajan pysäytyspainiketta, sillä hän haluaa, että töitä

ei poistu järjestelmästä sillä aikaa, kun hän syöttää satunnaistöitä. Tämän jälkeen Liisa aloittaa satunnaistöiden lähettämisen satunnaistöiden käynnistyspainikkeella. Liisa odottaa noin kolme minuuttia, kunnes järjestelmään on lähetetty runsaasti satunnaisia tulostustöitä ja järjestelmä on raskaasti kuormitettu. Lopulta hän napsauttaa ajoittajan käynnistyspainiketta, minkä jälkeen töitä alkaa siirtyä tulostimille. Liisa tutkii tarkkaavaisesti, miten maksimiviive palvelun saannille vaikuttaa tulostukseen.

Yhteiskäyttö (tapaus a)

Pekka ja Liisa haluavat tutkia tulostustöidensä lajittelua ja varsinkin niputtamista, kun he lähettelevät töitään tulostettavaksi. Pekka ja Liisa ovat ottaneet yhteyden järjestelmään peruskäyttäjinä. He ovat molemmat samassa huoneessa eri tietokoneilla. Tämän jälkeen molemmat lähettävät yhden sivun kokoisia töitä tulostettavaksi noin minuutin ajan. Pekka ja Liisa tutkivat, kuinka järjestelmä niputtaa heidän töitään yhteen, ja huomaavat, että heidän töistään muodostuu yksi yhtenäinen nippu, joka tulostuu tulostimelle ps3. Nipussa ensimmäisenä ovat Liisan työt ja sen jälkeen Pekan työt.

Yhteiskäyttö (tapaus b)

Pekka ja Liisa haluavat tutkia tulostustöidensä lajittelua ja varsinkin niputtamista, kun he lähettelevät töitään tulostettavaksi. Pekka ja Liisa ovat ottaneet yhteyden järjestelmään peruskäyttäjinä. He ovat molemmat samassa huoneessa eri tietokoneilla. Tämän jälkeen molemmat lähettävät erikokoisia töitä tulostettavaksi noin minuutin ajan. Pekka ja Liisa ovat sopineet, että he tulostavat töitä, joiden sivumäärät ovat noin 10–30 sivua. Pekka ja Liisa tutkivat, kuinka järjestelmä niputtaa heidän töitään yhteen, ja huomaavat, että Pekan työt tulostuvat yhdessä nipussa tulostimelle ps2 ja Liisan työt tulostimelle ps3, joka on noin kolmen metrin päässä tulostimesta ps2. Tämän jälkeen Pekka napsauttaa hiirellä tulostimen ps2 kuvaketta, jolloin hän voi tarkastella tulostimen jonoa tarkemmin. Liisa napsauttaa hiirellä erästä lähettämäänsä työtä, jolloin hän voi tarkastella työn ominaisuuksia.

Liite 2. Sanasto

C on yleiskäyttöinen, matalan tason ohjelmointikieli.

Java on yleiskäyttöinen, olioperustainen ohjelmointikieli.

Lasilaatikkotestaus on testausmenetelmä, jossa testitapausten valinta perustuu testattavan kohteen sisäisen rakenteen ja toteutuksen tuntemiseen. Testaus voidaan kohdistaa ohjelmointikoodin virheherkkiin rakenteisiin tai muihin ohjelmointitekniisiin epävarmuuksiin.

Linux on vapaasti saatava, unix-tyyppinen käyttöjärjestelmä.

Mustalaatikkotestaus on testausmenetelmä, jossa testitapausten valinta perustuu testauksen kohteen toiminnan sekä syöte- ja tulosterajapintojen tuntemiseen. Testitapaukset valitaan kohteen sisäistä rakennetta tai toteutusta tuntematta.

Nippu on sekä ohjelmistotuotantoprojektin että projektin tuotteena syntyvän tulostuspalvelinohjelmiston nimi. Nippu-nimi valittiin, koska simuloitava käyttäjä saa tuotteen avulla useat tulosteensa yhdessä nipussa. Nippu-projekti toteutetaan Helsingin yliopiston tietojenkäsittelytieteen laitoksella kesällä 2003.

Terttu on ajoittajaydin, joka sisältyy keskeisenä komponenttina Nippuun. Terttuun kuuluvat myös visualisoija ja tietoliikennekomponentti. Terttu on toteutettu Helsingin yliopiston Terttu-ohjelmistotuotantoprojektissa kesällä 2002.

Toimintaympäristö tarkoittaa sitä fyysistä tai simuloitua verkkoasemien muodostamaa verkkoa, jossa Nippu hallinnoi tulostusta.

Tulostaja tarkoittaa sellaista toimintaympäristössä toimivaa agenttia, joka synnyttää tulostuspyyntöjä.

Tuloste on tulostuskomennon suorittamisesta syntynyt lopputuote.

Tulostustyö on Nipun tai jonkin toimintaympäristön laitteen käsittelyssä oleva, tulostajan antama tulostuskomento.

Verkkoasema on yksittäinen tietoliikenneverkkoon liitetty laite, esimerkiksi tietokone tai tulostin.