TAMPEREEN TEKNILLINEN KORKEAKOULU
Sähkötekniikan osasto

Pekka Järveläinen

Supertietokoneen grafiikkaympäristö

Diplomityö

Aihe hyväksytty osastoneuvoston kokouksessa
21.8.1991
Tarkastaja: vs. apul. prof. Hannu-Matti Järvinen

Alkulause

Tämän diplomityön valvojana on Tampereen teknillisen korkeakoulun sähkötekniikan osastolta toiminut vs. apul. prof. Hannu-Matti Järvinen, jolle haluan esittää parhaat kiitokseni arvokkaista neuvoista ja myönteisestä suhtautumisesta.

Ystäviäni ja työtovereitani TLP:ssa haluan kiittää hyvästä avusta. Työn ohjaajaa Pirjo-Leena Forsströmia kiitän erittäin kiinnostavasta ja haastavasta aiheesta sekä kannustuksesta työn aikana.

Espoon Otaniemessä 7.1.1992

Pekka Järveläinen

Katajaharjuntie 13 A 5
00200 HELSINKI
puh. (90) 678 578

SISÄLLYSLUETTELO

Alkulause II
Tiivistelmä V
Kuvat VI
Taulukot VI
Käytetyt lyhenteet VII
1 Johdanto 1
2 Tietokonegrafiikka 2
2.1 Käsitteet 2
2.1.1 Värit 2
2.1.2 Kuvanmuodostus 3
2.1.3 Visualisointi 4
2.2 Standardit 4
2.2.1 GKS 4
2.2.2 CGM ja CGI 5
2.2.3 PHIGS 6
2.3 Tietokonegrafiikan kehitys 7
3 Ikkunointijärjestelmät 8
3.1 X11-ikkunointi 8
3.1.1 Hajautus 8
3.1.2 Laitteisto 9
3.1.3 Ikkunat ja niiden hallinta 10
3.1.4 Grafiikka 11
3.1.5 Ohjelmointi 11
3.2 NeWS 14
3.2.1 PostScript 14
3.2.2 Toimintatapa 15
3.2.3 Grafiikka 15
3.2.4 Ohjelmointi 15
3.3 Muut ikkunointijärjestelmät 15
4 Käyttöliittymät 17
4.1 Ominaisuudet 17
4.2 Graafiset käyttöliittymät 18
4.2.1 Motif 18
4.2.2 Open Look 19
4.3 Kommunikaatiomuodot 20
4.4 Johtopäätöksiä 21

5 Supertietokoneympäristö 22
5.1 Tieteellisen laskennan palvelu 22
5.1.1 Tiedotus 22
5.1.2 Cray X-MP EA/432 22
5.1.3 Edustakoneet ja hajautus 23
5.1.4 Tiedostopalvelin 23
5.1.5 Lähiverkko 23
5.1.6 FUNET 24
5.2 Grafiikka 25
5.2.1 Ikkunointijärjestelmät 25
5.2.2 Käyttöliittymät 25
5.2.3 Kuvaformaatit 26
5.2.4 Laitteisto 26
5.3 Visualisointiohjelmistot 27
5.3.1 Numeroaineiston muuttaminen kuvaksi 27
5.3.2 Julkaisukelpoiset kuvat 28
5.3.3 Animaatio ja realistiset kuvat 29
5.3.4 Visualisoinnin kehitys 30
6 Xftp 32
6.1 Ftp 32
6.2 Määrittely 32
6.3 Suunnittelu 32
6.3.1 Xdtm 32
6.3.2 Prosessien välinen kommunikointi 32
6.4 Käyttöliittymä 33
6.5 Toteutus 35
6.5.1 Ftptool 35
6.6 Kokemuksia 36
7 Esimerkki sovellusohjelmien käytön helpottamisesta X-ympäristössä 38
7.1 Ongelma 38
7.2 Ratkaisu 38
7.2.1 Taustavalikko 38
7.2.2 Näppäimistön ohjelmointi 40
7.3 Kokemuksia 41
8 Johtopäätökset 42
Lähdeluettelo 43
Hakemisto 46

Tiivistelmä

TAMPEREEN TEKNILLINEN KORKEAKOULU
Sähkötekniikan osasto
Ohjelmistotekniikka
JÄRVELÄINEN, PEKKA: Supertietokoneen grafiikkaympäristö
Diplomityö, 47 s., 10 liites.
Tarkastaja: vs. apul. prof. Hannu-Matti Järvinen
Rahoittaja: Tieteellisen laskennan palvelu
Tammikuu 1992

Tietokonegrafiikalla pystytään havainnollistamaan suuria tietomääriä, joita esiintyy erityisesti supertietokoneympäristössä. Tietokoneverkot mahdollistavat supertietokoneen etäiskäytön henkilökohtaiselta tietokoneelta. Hajautettuihin ikkunointijärjestelmiin X ja NeWS perustuvat graafiset käyttöliittymät toimivat myös verkon yli, jolloin sovellusohjelmien käyttö ja visualisointi supertietokoneympäristössä on yhtä helppoa kuin mikrotietokoneessa.

Työssä suunnitellaan ja toteutetaan graafinen käyttöliittymä ftp-ohjelmalle ja toteutetaan esimerkki sovellusten käynnistämisestä X-ikkunointiympäristön taustavalikosta. Työssä selvitetään tietokonegrafiikan peruskäsitteitä ja standardeja sekä käyttöliittymiä ja hajautettuja ikkunointijärjestelmiä. Selvityksessä käytetään esimerkkinä Tieteellisen laskennan palvelun supertietokoneympäristöä.

Esimerkki sovellusohjelmien käynnistämisestä X-ympäristössä osoittaa, että käyttöympäristöä voidaan huomattavasti kehittää käyttämällä X:n taustavalikoita ja näppäimistön hallintaa. TLP:ssä ei juurikaan pystytä vaikuttumaan käytettävien ohjelmien käyttöliittymiin, mutta ohjelmien käyttöympäristöön vaikuttamisen esimerkki osoittaa mahdolliseksi. TLP:n tavoitteena tulisikin olla tarjota hyvä käyttö- ja ohjelmankehitysympäristö.

Ohjelmistokehityksessä on pyrittävä löytämään standardirajapinnat ja hyödyntämään niitä. X-ikkunointijärjestelmä muodostaa yhden hyvin määritellyn ja toimivan standardirajapinnan, jota kannattaa hyödyntää.

Kuvat

Kuva 1 Heijastuminen valon aallonpituuden funktiona vähentävässä värijärjestelmässä. 2
Kuva 2 Tietokoneen grafiikkapalvelujen standardikerrokset 6
Kuva 3 Hierarkkinen malli tietokoneesta 8
Kuva 4 X-ohjelmoinnin tasot 13
Kuva 5 SGI:n ikkunointijärjestelmä 4Sight 16
Kuva 6 Käyttöliittymän osuus eri sovelluksissa 17
Kuva 7 Open Look File Managerin perusikkuna 19
Kuva 8 Funet 24
Kuva 9 SGI VGX bittitasot 27
Kuva 10 xftp-ohjelman rakenne 33
Kuva 11 xftp 34
Kuva 12 xftp ja kohoikkuna 34
Kuva 13 Ftptool 36
Kuva 14 Patran-valikon luonti ikkunanhallintaohjelman alustustiedostossa 39
Kuva 15 patran1.csh komentotiedosto 40
Kuva 16 xrdbpatran resurssitiedosto funktionäppäinten ohjelmoimiseksi X-ympäristössä 40

Taulukot

Taulukko 1 Hiiren näppäinten toiminnot Motif-käyttöliittymässä . . . 18
Taulukko 2 Käyttöliittymien vertailua 20

Käytetyt lyhenteet

2D	two dimensional, kaksiulotteinen
3D	three dimensional, kolmiulotteinen
ACM	Association for Computing Machinery, tietotekniikka-yritysten yhdistys
AES	Application Environment Specification, API
ANSI	American National Standards Institute, standardoimisjärjestö
API	Application Programming Interface, ohjelmointiliittymä
ASCII	American Standard Code for Information Interchange, merkistö
AT&T	American Telephone&Telegraph, yritys
CAD	Computer Aided Desing, tietokonetuettu suunnittelu
CGI	Computer Graphics Interface, standardi
CGM	Computer Graphics Metafile, standardi
CSC	Centre for Scientific Computing, TLP
CSS	Central Structure Store, PHIGS:n tietorakenne
DECNET	Digital Equipment Corporationin tietoliikennekäytäntö
DGL	Distributed Graphics Library, SGI:n grafiikkakirjasto
DIN	Deutsches Institut für Normung, standardoimisjärjestö
MS-, PC-DOS	Disk Operating System, PC:n käyttöjärjestelmä
DPS	Display PostScript, PostScriptin laajennus
DVI	Digital Video Interactive, vuorovaikutteinen video
DVSD	Disspla Virtual Storage Data file, tiedostoformaatti
FDDI	Fiber Distributed Data Interface, tietoliikennekäytäntö
FEM	Finite Element Method, elementtimenetelmä
ftp	file transfer protocol, tietoliikenneohjelma
FUNET	Finnish University and reseach NETwork, tietoliikenneverkko
GKS	Graphical Kernel System, ISO- ja ANSI-standardi
GUI	Graphical User Interface, graafinen käyttöliitymä
HCI	Human-Computer Interaction, käyttöliittymä
HDF	Hierarchical Data Format, tiedostoformaatti
HiPPI	HIgh Performance Parallel Interface, tietoliikennekäytäntö
HP	Hewlett-Packard, yritys
IBM	International Business Machines, yritys
ISO	International Standards Organization, standardoimisjärjestö
LAB	LAnguage Binding, ohjelmointiliittymä

MCAE	Mechanical Computer-Aided Engineering, tietokonetuettu tekniikka
MIPS	Millions of Instructions Per Second, tehokkuuden yksikkö
MIT	Massachusetts Institute of Tecnology, yliopisto
MPGS	Multi Purpose Graphics System, visualisointiohjelmisto
MRM	Motif Resource Manager, resurssihallitsija
mwm	motif window manager, ikkunanhallintaohjelma
NCSA	National Center for Supercomputing Applications, USA:n TLP
NeWS	Network/extensible Window System, ikkunointijärjestelmä
NQS	Network Queuing System, eräajojärjestelmä
OLIT	Open Look Intrinsics Toolkit
olwm	open look window manager, ikkunanhallintaohjelma
olvwm	open look virtual window manager, ikkunanhallintaohjelma
OSF	Open Software Foundation, laitevalmistajien liittouma
PC	Personal Computer, henkilökohtainen tietokone
PHIGS	Programmer's Hierarchical Interactive Graphics System, standardi
PHIGS PLUS	PHIGS Plus Lumiere Und Surfaces, standardi
PEX	PHIGS Extension to X, X:n 3D grafiikkalaajennus
PEXIM	PHIGS (subset) EXtended for Immediate Mode, yksi API PEXille
PM	Presentation Manager, OS/2:n ikkunointi
POSIX	Portable Operating System Interface, IEEE-standardi
RAID	Redundant Array of Inexpensive Disks, levyryhmäteknikka
RGB	Red, Green, Blue, värijärjestelmä
SGI	Silicon Graphics Inc., yritys
SIGGRAPH	Special Interest Group on Computer Graphics, yhdistys
SSD	Solid-state Storage Device, Crayn virtuaalilevy
TCP/IP	Transmission Control Protocol/Internet Protocol, tietoliikennekäytäntö
TLP	Tieteellisen laskennan palvelu
TNT	The NeWS Toolkit
twm	tab (Tom's) window manager, ikkunanhallintaohjelma
UI	Unix International, laitevalmistajien liittouma
UIDT	User Interface Development Tool, sovelluskehittimiä
UIL	User Interface Language, ohjelmointikieli
UNICOS	UNIx for Cray Operating System, käyttöjärjestelmä
VGA	Video Graphics Array, PC:n näyttöstandardi

VTKK	Valtion tietokonekeskus
VTT	Valtion teknillinen tutkimuskeskus
WINTERP	Widget INTERPreter, ohjelmisto
WYSIWYG	What You See Is What You Get, käyttöliittymämalli

1. Johdanto

Vielä runsaat kymmenen vuotta sitten tietokonelaitteet olivat kalliita ja ohjelmistuotannon painopiste oli suoritusajan ja muistinkäytön optimoinnissa. Laitteistojen nopeasti halvetessa on ohjelmistuotantoa tehostettu siirtymällä runsaammin resursseja kuluttaviin menetelmiin, esimerkiksi olio-ohjelmointiin ja sovelluskehitinten käyttöön. Tämä kehitys on tehnyt ohjelmista massatuotteita, ja nykyään käyttäjien oppimiskustannusten pienentäminen entistä parempien käyttöliittymien avulla on huomattava keino pienentää tietotekniikan kustannuksia.

Yhä useammilla käyttäjillä on mahdollisuus grafiikkaan. Graafiset sovellukset vaativat yleensä paljon laskentakapasiteettia [Kos90]. Lisääntyvän laskentatehon ansiosta voidaan grafiikkaa käyttää yhä monimutkaisemmilla sovellusalueilla.

Perinteisesti supertietokoneita on käytetty edustakoneen välityksellä. Käyttäjä on valmistellut ajonsa edustakoneella, lähettänyt sen suoritettavaksi supertietokoneeseen ja tarkastellut tuloksia taas edustakoneessa. Syynä superkoneiden eräkäyttöön on ollut niiden soveltumattomuus keskustelevaan työskentelyyn. Esimerkiksi Cray X-MP:ssä ei ole lainkaan erillistä oheisprosessoria, joka huolehtisi pääteliikenteestä, vaan jokainen päätteeltä näppäilty merkki keskeyttää työn yhdessä sen prosessoreista. Koska Crayssä ei ole virtuaalimuistia, niin yksi näppäimen painallus saattaa pahimmassa tapauksessa aiheuttaa kokonaisen työn siirtämisen keskusmuistista levymuistiin ja takaisin. Hajautetut ikkunointijärjestelmät voivat vähentää keskeytyksiä. [Hei88]

Tässä työssä supertietokoneen grafiikkaympäristö on jaettu kolmeen osaan:

1. tietokonegrafiikka ja visualisointi,
2. ikkunointijärjestelmät ja
3. käyttöliittymät.

Toinen luku kertoo tietokonegrafiikasta yleensä ja määrittelee käytettäviä käsitteitä. Grafiikka selostetaan perusteista lähtien, mutta tietotekniikan peruskäsitteet oletetaan tunnetuksi. Niihin tutustumiseen sopii [RH91]. **Kolmas luku** kuvailee TLP:ssa käytettävät hajautetut ikkunointijärjestelmät. Keskitetyt ikkunointijärjestelmät ja niiden käyttöliittymät, esimerkiksi Microsoftin Windows ja Presentation Manager ja Applen Macintosh, eivät kuulu tämän työn aiheeseen. **Neljäs luku** esittelee vallitsevat käyttöliittymästandardit Open Lookin ja Motifin. **Viides luku** kuvailee TLP:n, Tieteellisen laskennan palvelun supertietokoneympäristöä ja erityisesti sen visualisointiohjelmistoja. **Kuudennessa luvussa** suunitellaan ja toteutetaan graafinen käyttöliittymä ftp-ohjelmalle. **Seitsemännessä luvussa** toteutetaan esimerkki X-ikkunointijärjestelmän mahdollisuuksista helpottaa sovellusohjelmien käyttöä.

Tämän työn tavoitteena on tutkia hajautettujen ikkunointijärjestelmien ja niitä käyttävien graafisten käyttöliittymien mahdollisuuksia parannettaessa supertietokoneen grafiikka- ja käyttöympäristöä.

Supertietokoneympäristöä on tutkittu Timo Sinkkilän diplomityössä ja Kimmo Kosken lisensiaattityössä [Sin90, Kos90]. Pyry Tiaisen ja Matti Karin diplomityöt [Kar90, Tia91] käsittelevät käyttöliittymiä.

2. Tietokonegrafiikka

Tietokonegrafiikka on erilaisten ilmiöiden mallintamista ja kuvamista tietokoneen ja siihen liittyvien syöte- ja tulostinlaitteiden avulla [Fol90]. Kuvankäsittelyssä *(image prosessing)* muokataan tietokoneen muistiin tallennettua kuvaa; kuvalla ei ole enää yhteyttä siihen malliin, jota se esittää, tai sitten kuvasta yritetään selvittää, mitä se esittää. Yhteistermi tietokonegrafiikalle ja kuvankäsittelylle on *visual computing*.

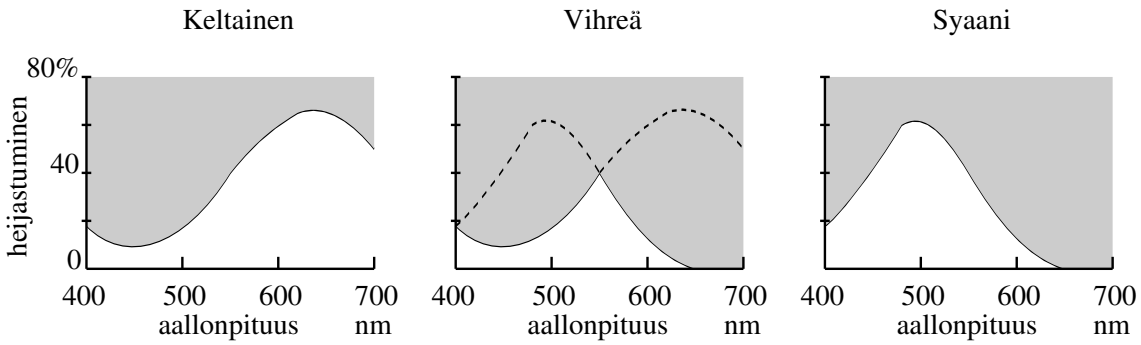
Vuorovaikutteisessa tietokonegrafiikassa voidaan syötelaitteiden avulla muuttaa tulostinlaitteella näkyvää kuvaa. Kuvan muuttaminen tapahtuu mallin tasolla eikä kuvaruudun bittitasolla. Yksi tietokonegrafiikan sovellusalueista onkin kommunikointi koneen kanssa eli käyttöliittymät. Muista sovellusalueista voidaan mainita tieteellinen visualisointi, animaatio, julkaisu, esitysgrafiikka, kartat, piirrosohjelmat, taide, prosessien valvonta, mallintaminen, suunnittelu *(CAD)*, visuaalinen ohjelmointi ja opetus.

2.1 Käsitteet

Pikseli (*pixel*; kuva-alkio, rasteripiste, kuvapiste) on kuvan pienin osa, jolla on oma väri tai intensiteetti. Rasterikuva *(pixmap)* on suorakulmainen pikselijoukko. Laitteistotasolla mustavalkoisessa rasterikuvassa eli bittikartassa yksi bitti, nolla tai yksi, vastaa yhtä pikseliä. Pistegrafiikassa (rasteri-, bittikartta-) käsitellään kuvan yksittäisiä pisteitä, pikseleitä. Vektorigrafiikassa kuva koostuu monimutkaisemmista osista, esimerkiksi viivoista ja ympyröistä.

2.1.1 Värit

Pinnan väri määräytyy sen ominaisuudesta heijastaa valon tiettyjä aallonpituuksia. Esimerkiksi keltaiset pinnat näkyvät keltaisina, koska ne heijastavat suurimman osan niihin osuvasta keltaisesta valosta, aallonpituuksista 570–580 nm. Keltaisen, vihreän ja syaanin (sinivihreä) värin heijastamat aallonpituudet ovat kuvassa 1. Kuvasta selviää, kuinka sekoittamalla syaania ja keltaista väriä syntyy vihreää.



Kuva 1. Heijastuminen valon aallonpituuden funktiona vähentävässä värijärjestelmässä.

Koska väriaineita sekoitettaessa heijastuva valo vähenee, kutsutaan tälläistä värien muodostumista vähentäväksi *(subtractive color mixture)*. [Dur87]

Valolähteen säteilemässä valossa eri aallonpituudet summautuvat eli värit muodostuvat additiivisesti. Näin toimivat esimerkiksi kuvaputkeen (CRT, cathode ray tube; katodisädeputki) perustuvat näyttölaitteet. RGB (Red, Green, Blue; punainen, vihreä, sininen) on summaava värijärjestelmä, jossa väri määritellään kolmella kunkin perusvärin intensiteettiä kuvaavalla arvolla [Hal89]. CMY (Cyan, Magenta, Yellow; syaani eli sinivihreä, magenta eli violetti, keltainen) on vastaava joissakin tulostuslaitteisa käytetty vähentävä värijärjestelmä.

Koulun kuvaamataidon tunneilla opetetaan vähentävä värinmuodostus, mutta harvat ihmiset tuntevat summaavan värinmuodostuksen, jossa esimerkiksi punaisesta ja vihreästä syntyy keltaista. Myös punaista ja vihreää heijastava sekä sinistä absorboiva pinta on keltainen. Värit voidaan määritellä myös sävyn, saturaation ja valoisuuden tai kirkkauden perusteella. HLS- (Hue, Lightness, Saturation) ja HSB- (Hue, Saturation, Brightness) värijärjestelmissä sävyt ovat ympyrässä valoisuusasteikon ympärillä liitteen 1 kuvan mukaisesti.

Bittitasot tarkoittavat värikuvan yhden pikselin värin esittämiseen käytettävissä olevien bittien lukumäärää. Esimerkiksi RGB-järjestelmän kutakin kolmea väriä kohti voi olla kahdeksan bittiä, jolloin pikselin värin määräytymiseen tarvitaan 24 bittiä. Tällöin kuvassa voi olla 16 777 216 väriä. Tavallisimmissa näyttölaitteissa on nykyisin pikseliä kohti kahdeksan tai neljä bittiä, jotka indeksoivat pikselin värin määräävää värikarttaa eli palettia. Sekoitus *(dithering)* tarkoittaa vierekkäisten pikseleiden värien sekoittumista niin että katsojan silmän verkkokalvolla yhteen sulautuvat pikselit muodostavat keskiarvosävyn.

2.1.2 Kuvanmuodostus

Kuvanmuodostuksessa *(rendering)* lasketaan mallista rasterikuva. Varjostuksessa pinnan kirkkaus lasketaan esimerkiksi valaistuksen ja pinnan heijastusominaisuuksien ja läpinäkyvyyden perusteella. Pintarakenne *(texture)* kuvaa pinnan ominaisuuksia. Leikkaaminen poistaa ikkunan tai kuva-alueen ulkopuolelle osuvan grafiikan.

Piiloviivat ja -pinnat ovat kuvan toisten osien taakse jääviä osia. Z-puskuriin tallennetaan kunkin kuvapisteen z-koordinaatti, syvyys. Kuvanmuodostuksessa pikselin arvo muuttuu vain, jos se on lähempänä katsojaa kuin Z-puskurissa oleva pikselin nykyinen etäisyys. Tämä algoritmi piiloviivojen ja -pintojen poistamiseen voidaan toteuttaa laitteistolla tai ohjelmistolla. Liitteessä 2 on esimerkkikuvasarja kuvanmuodostuksesta.

Vokselit ovat kappaleen tilavuusalkioita. Tilavuussävytyksessä *(volume rendering)* vokseleilla on väri- ja läpinäkyvyysarvot. Kuva tuotetetaan katselutasolle pikseli kerrallaan yhdistämällä kuhunkin pikseliin niiden vokseleiden tieto, jotka kyseisen pikselin ja silmän paikan välinen suora lävistää.

Tulostinlaitteen epälineaarisuus kumotaan gamma-korjauksessa, esimerkiksi monitorin fosforin epälineaarinen reaktio suhteessa elektronitykin ohjausjännitteeseen korjataan. Pehmennyksessä *(antialiasing)* vähennetään laskostumisilmiötä eli näytön erotuskyvylle liian suuret kuvan taajuuskomponentit suodatetaan pois, esimerkiksi tasataan vinon viivan portaisuutta. Samalla voidaan laskea *motion blur* eli liikkeen

epäselvyys ja *athmospheric effects* eli ilmakehässä tapahtuvan valon sironnan aiheuttama kaukaisten kohteiden hämärtyminen.

2.1.3 Visualisointi

Satelliitit, radioteleskoopit, magneettiresonanssi-, röntgentietokonetomografialaitteet sekä hiukkaskiihdyttimet ja -törmäyttimet ovat esimerkkejä laitteista, jotka tuottavat valtavia määriä mittaustuloksia ja muuta numeroaineistoa. Koska useisiin tieteellisen laskennan sovelluksiin liittyy runsaasti informaatiota, on tiedon esitystapa erittäin tärkeä. Näkeminen on tehokas tapa havainnoida ja niinpä visualisointi ja graafinen esittäminen ovat luonnollisia tekijöitä parannettaessa informaation ymmärrettävyyttä. [For90]

Visualisoinnissa symbolinen tieto muunnetaan geometriseksi. Näin voidaan nähdä näkymätön, esimerkiksi molekyylien liittyminen toisiinsa. Visualisoinnilla voidaan parantaa aineistojen mieltämistä, luoda uudenlaisia malleja, yhdistellä tietoa, tiivistää kokemuksia ja havaintoja sekä opettaa.

2.2 Standardit

Standardit helpottavat sovellusohjelmien siirrettävyyttä. Standardit kertovat laitevalmistajille, millaisia ominaisuuksia laitteilta odotetaan. Helpottamalla sovellusohjelmoijan työtä standardit mahdollistavat grafiikan laajemman käytön ohjelmissa.

2.2.1 GKS

GKS *(Graphical Kernel System)* on ANSI- ja ISO-standardi 2D-tietokonegrafiikan ohjelmointiin. Sen loi saksalaisen DIN:n asettama ryhmä ACM SIGGRAPH:n *(Association for Computing Machinery*: *Special Interest Group on Computer Graphics*) aloitteesta vuonna 1978. GKS:n kehittäminen on kuitenkin ollut jatkuvaa, ja vuonna 1985 hyväksyttiin GKS:n kaksiulotteinen ISO-standardi tasolla 2B. Tasorakenne on kaksiulotteinen, numerot kuvaavat tulostus- ja kirjaimet syöttömahdollisuuksia. Alempi taso on aina osajoukko ylemmästä tasosta.

USA:ssa käytetään laajennusta nimeltä CORE-79, jossa sekä 2D- että 3D-esitykset ovat mahdollisia. ISO:n vastaava laajennus GKS-3D on toiminnallisesti melko samanlainen kuin CORE-79, mutta ohjelmointiliittymä on erilainen. Esimerkiksi funktioiden nimet, parametrien järjestys ja virheilmoitusten numerot eroavat. Tällä hetkellä maailmalla on käytössä useita eri toimittajien GKS-pakkauksia. Ne erovat kuitenkin toisistaan sen verran, että niihin pohjautuvat ohjelmat vaativat pieniä muutoksia kirjastosta toiseen siirryttäessä.

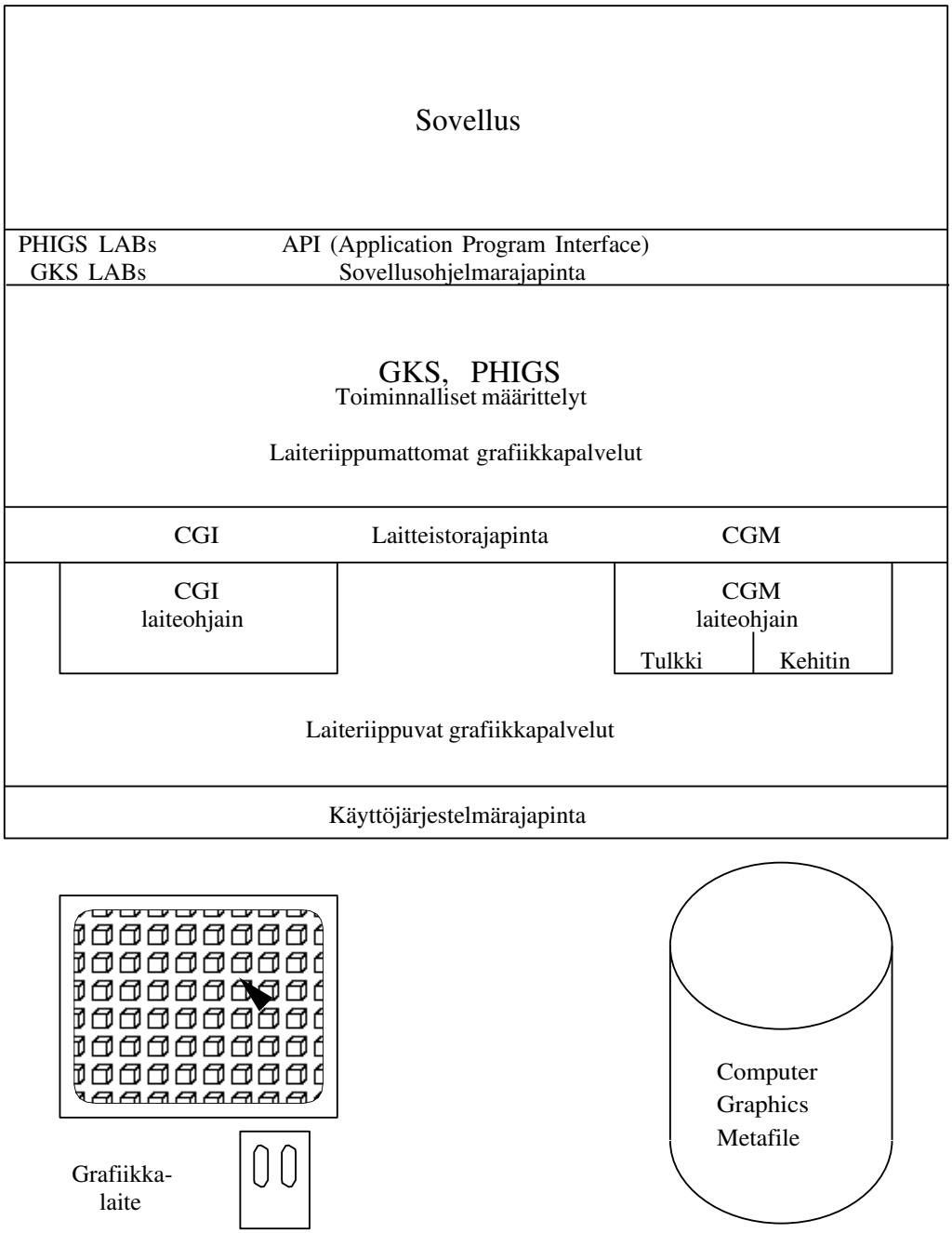
Standardissa määritellään parin sadan ohjelmointikielestä riippumattoman grafiikan perusalkion toiminta. Tulostuksen perusalkiot ovat *polyline, polymarker, text, fill area, cell array* ja *generalized drawing primitive*. Ne käyttävät maailmankoordinaatteja (käyttäjän koordinaatit), jotka kuvataan normalisoiduiksi laitteen koordinaateiksi (0:sta 1:teen) ja laitteen koordinaateiksi (laitteiston käyttämät, esim. -1023:sta +1023:een) kertomalla koordinaatit transformaatiomatriisilla. Tällöin kuvaa voidaan siirtää, kiertää ja sen kokoa muuttaa. Tulostuksen perusalkiot ovat yksinkertaisia. Esimerkiksi viivanpiirtofunktio saa parametreinaan vain pisteet, joiden kautta viiva kulkee. Ohjelman tila määrää viivan ominaisuudet kuten leveyden ja värin. Tämä

tila määrätään grafiikka-attribuutteilla, joita ovat kirjasinlaji, kirjainten välit, kulmat, väri, intensiteetti, viivatyyli (leveys, varjostus), välkytys, näkyvyys ja poiminta.

Valokynän, ristikon tai hiiren avulla kohdistinta näytössä liikuttamalla voidaan poimia tietty olio, segmentti. Segmentissä on koottu grafiikkakäskyjä joukoksi, johon voi kohdistaa toimintoja, esimerkiksi tuhoaminen tai värin vaihto. Ohjelmointiliittymä (LAB, LAnguage Binding) Fortraniin ja Pascaliin on standardoitu erikseen.

2.2.2 CGM ja CGI

CGM (Computer Graphics Metafile) ja CGI (Computer Graphics Interface) ovat laitteistoliittymästandardeja eli ne määrittelevät rajapinnan laiteriippumattoman ja laiteriippuvan kerroksen välille, kuva 2. Niiden toteutukset ovat laiteohjaimia. CGI on vuorovaikutteisia laitteita varten. CGM-tiedostoon tallennetaan kuvan muodostavat tulostusrutiinit attribuutteineen.



Kuva 2. Tietokoneen grafiikkapalvelujen standardikerrokset

2.2.3 PHIGS

PHIGS, Programmer's Hierarchical Interactive Graphics System, on ANSI- ja ISO-standardi. Pohjana on GKS, johon on lisätty 3D-piirtorutiinit ja rakenteellisuus. PHIGS PLUS:aan, PHIGS Plus Lumiere Und Surfaces, on edelleen lisätty esimerkiksi valot ja varjostukset.

PHIGS-kuvat tallennetaan rakenteisiin. Rakenne on lista tulostuskäskyjä, niihin vaikuttavia ominaisuuksia, geometrisia kuvauksia ja viittauksia toisiin rakenteisiin. Listassa on myös rakenteen nimi. Lisäksi PHIGS:ssa on komennot rakenteiden hallintaan: näyttämiseen ja poistamiseen kuvaruudulta, rakenteen editointiin ja tallentamiseen. PHIGS:n puutteena on huono laajennettavuus.

2.3 Tietokonegrafiikan kehitys

Tietotekniikan nopea edistys on mahdollistanut tietokonegrafiikan kehityksen viivagrafiikasta rasterigrafiikaksi ja mustavalkoisesta värilliseksi. Standardeistakin näkyy grafiikan nykyinen kehitys kaksiulotteisesta kolmiulotteiseksi. Tulevaisuuden kehityssuunta on yksittäisistä kuvista animaatioksi.

Multimedia on tekstin, kuvien, äänen, videon ja muun datan yhdistämistä. Hypermediassa muodostetaan lisäksi yhteyksiä joiden avulla käyttäjä voi hakea ja yhdistellä eri muotoista tietoa kuten hypertekstissä. Hyperteksti on moniulotteinen tekstintallennusrakenne, joka sisältää ohjelmoituja viittauksia muihin tekstikohtiin.

Intel Corporationin kehittämä vuorovaikutteinen video DVI (Digital Video Interactive) sisällytetään Presentation Manageriin vuoteen 1994 mennessä. Euroopan systeemitoimintojen johtajan Daniel Louppen mukaan Intel suunnittelee sijoittavansa DVI:n suoraan suoritinpiirille. [Tol90]

3. Ikkunointijärjestelmät

Käyttöjärjestelmä on ohjelmisto, joka toimii liitäntänä sovellusohjelmien ja laitteiston välillä. Ikkunointijärjestelmä toimii samoin liitäntänä kuvaruudun, hiiren ja näppäimistön sekä sovellusohjelman välillä. Käyttöjärjestelmien tapaan ikkunointijärjestelmä tarjoaa oheislaitteiden ohjaukseen ja muistinhallintaan liittyviä palveluja, huolehtii laiteresurssien jakamisesta ohjelmille, ohjelmien suojaamisesta muissa ohjelmissa olevilta virheellisyyksiltä jne. Varhaisimmat ikkunointijärjestelmät olivatkin osa käyttöjärjestelmää.

Uudemmat ikkunointijärjestelmät ovat kehittyneet erilleen käyttöjärjestelmistä muodostaen kuvan 3 mukaisesti oman kerroksensa käyttöjärjestelmän ja sovelluksen välille. Samalla ikkunointijärjestelmiin on tullut monia käyttöjärjestelmien ominaisuuksia kuten prosessien välinen kommunikointi. Nyt ikkunointijärjestelmät ovat laajenemassa uudet multimediaoheislaitteet kattaviksi mahdollistaen ääni- ja videosovellukset.

Kuva 3. Hierarkkinen malli tietokoneesta

3.1 X11-ikkunointi

X on hajautettujen ikkunointijärjestelmien de facto standardi. X:n määrittelevät dokumentit luetellaan viitteessä [MIT91b]. X mahdollistaa useat, rinnakkaiset sovellukset, korkealaatuisen tekstin, grafiikan ja kuvat. Sen etuja ovat toimivuus erilaisissa verkkoympäristöissä, käyttöjärjestelmä- ja laiteriippumattomuus, siirrettävyys ja saatavuus.

3.1.1 Hajautus

X on hajautettu ikkunointijärjestelmä eli sovellusohjelman ei tarvitse olla samassa koneessa kuin näyttö. X perustuu asiakas-palvelin-malliin (*client-server*). Piirtääkseen tai kirjoittaakseen kuvaruudulle asiakas, sovellusohjelma, luo ensiksi yhteyden palvelimeen. Yhdessä laitteistossa palvelee kaikkia X-sovelluksia yksi X-palvelin, jolle eri asiakkaat lähettävät pyyntöjä *(request)*. Yleensä pyyntöihin ei vaadita kuittausta eli kyseessä on sokea kommunikointi. Verkko on ohjelmoijan kannalta läpinäkyvä eli samaa sovellusta voidaan käyttää paikallisesti tai verkon yli.

Hajautus on toteutettu laiteriippumattomalla asynkronisella X-protokollalla, joka toimii eri verkkoprotokollien, esimerkiksi DECNET ja TCP, päällä sekä eri liityntärajapintojen kuten socket ja stream läpi. X huolehtii myös tiedon esitystavan automaattisesta muuntamisesta. Esimerkiksi kokonaislukujen tavujärjestys vaihtelee eri

arkkitehtuureissa. Kun yhteys asiakkaan ja palvelimen välillä syntyy, palvelin havaitsee asiakkaan tavujärjestyksen ja automaattisesti muuntaa kaiken kommunikaation tähän järjestykseen. Järjestelmän osakokonaisuuksien ei tarvitse käyttää samaa laitteistoarkkitehtuuria tai käyttöjärjestelmäympäristöä.

X:ään voidaan lisätä uusia ominaisuuksia. Laajennukset eivät häiritse olemassa olevien sovellusten toimintaa.

Tapahtumat

Palvelin lähettää asiakkaille tapahtumia *(event)*, jotka ovat vastauksia pyyntöihin, virheilmoituksia tai käyttäjän syötteitä. Tapahtumia syntyy esimerkiksi käyttäjän liikuttaessa hiirtä tai painaessa näppäintä. Esimerkiksi siirretään kohdetta näytöllä [Kar90]:

1. hiirtä liikutetaan ja ohjain lähettää palvelimelle tapahtuman,
2. palvelin herää ja lukee tapahtuman,
3. palvelin lähettää tapahtuman oikealle asiakkaalle ja nukahtaa,
4. asiakas herää, lukee tapahtuman, poistaa kohteen ja piirtää sen uuteen paikkaan lähettämällä jonon komentoja palvelimelle sekä nukahtaa ja
5. palvelin herää, lukee komennot, suorittaa ne ja nukahtaa.

Tapahtumia aiheuttavat myös muutokset kuvaruudulla tai toiset sovellukset. Esimerkiksi ikkunan paljastuessa toisen ikkunan alta syntyy *expose*-tapahtuma. X-sovellusten on osattava käsitellä tämä tapahtuma ja päivitettävä paljastunut ikkuna tai sen osa.

3.1.2 Laitteisto

Siirrettävyytensä ansiosta X on saatavissa lähes kaikille tietokoneille IBM-PC-yhteensopivista ja Apple Macintoshista uusimpiin grafiikkatyöasemiin ja Cray-superkoneisiin. Lähdekoodi on ilmainen ja se on saatavilla esimerkiksi FUNET:n koneelta **nic.funet.fi** [MIT91a] (katso 5.1.6 FUNET sivulla 24).

X:n laitteistoriippumattomuus mahdollistaa saman sovelluksen käytön eri syöttö- ja tulostuslaitteilla ilman käännöksiä tai muutoksia ohjelmakoodiin. Esimerkiksi samaa supertietokoneessa toimivaa sovellusta voi kaksi käyttäjää yhtäaikaa käyttää työaseman värilliseltä näytöltä ja mustavalkoiselta Macintoshilta. Tärkeimmät sovellusohjelmoijalle näkyvät laitteiston piirteet ovat värien tai harmaasävyjen lukumäärä ja kuvaruudun koko.

Laitteistovaatimuksista tärkein on bittikarttanäyttö. X:ssä termeillä näyttö ja kuvaruutu (*display* ja *screen*) on erikoismerkitys. Näyttö tarkoittaa kaikkia yhden palvelimen resursseja ja vähimmäiskokoonpanon muodostavat kuvaruutu, näppäimistö ja hiiri. Kokoonpanoon voi kuulua useita kuvaruutuja. Kuvaruutu ei välttämättä vaadi omaa kuvaputkea vaan se voi olla vaikkapa värinäytön mustavalkoinen kuvapuskuri. Palvelin voi olla tehokas Unix-insinöörityöasema, mikrotietokone tai erityinen X-pääte [Kar90]. Luonnollisin X-palvelin on työasema.

X-pääte

X-pääte eroaa työasemasta siinä, että X-päätteessä suoritetaan vain yhtä ohjelmaa eli X-palvelinta. Kaikki X-asiakasprosessit, yleensä myös ikkunanhallintaohjelma, täytyy ajaa jossakin tietokoneessa. Tämä vaatii suurta toimintavarmuutta ympäristöltä.

X-päätteen etuna työasemaan verrattuna voidaan pitää edullista hintaa ja vähäistä ylläpidon tarvetta [Lum90b]. Hankittaessa työasema saadaan lisää laskentatehoa. X-päätteelle on varattava ainakin 2 MIPS:ä tehoa isäntäkoneessa [Jär90]. Hintavertailussa osuus isäntäkoneen hinnasta ja X-päätteen tulevat ohjelmistopäivitykset on syytä lisätä X-pääteen hintaan. X-päätteiden ja työasemien verkkoliikennettä on tutkittu viitteessä [Iko90].

Tarvittavan muistin määrä vaihtelee eri valmistajien X-päätteillä. Muistin määrä joudutaan ylimitoittamaan tai runsaasti muistia käyttävistä sovelluksista on luovuttava, koska nykyisissä X-päätteissä ei ole virtuaalimuistia.

Mikrotietokone

Käytettäessä PC:tä X-työasemana ongelmia aiheuttavat normaalin PC:n hitaus, sen pieni näyttö ja näytön suppea värivalikoima. Nykyisen PC:n näyttöstandardin VGA:n kuvan koko on 640 x 480 pikseliä ja käytössä on 16 samanaikaista väriä. X-käyttöön soveltuu 386–suorittimella ja VGA:ta paremmalla näytöllä varustettu PC. Käyttöä voi hankaloittaa mikron yksi- tai kaksinäppäiminen hiiri; esimerkiksi Open Look- ja Motif-käyttöliittymissä oletetaan kolminäppäiminen hiiri.

3.1.3 Ikkunat ja niiden hallinta

Ikkuna on näytön osa, jota hallitaan erikseen. Ikkunat pystyvät peittämään toisensa kokonaan tai osittain, niitä voi siirrellä näytöllä ja niiden kokoa voi muuttaa. X:ssä pohjimmaisena taustalla on koko näytön kattava juuri-ikkuna (*root window, workspace*), jonka päälle muut ikkunat muodostavat hierarkkisen rakennelman.

Ikoni eli kuvake on näytöllä näkyvä kuvallinen symboli, jota vastaa joku toimenpide, ohjelma tai ikkuna. Ikkunat voi ikonisoida (sulkea, pienentää) ikoneiksi. Ikonit voi aktivoida eli suurentaa takaisin ikkunoiksi. Ikonin aktivointi voi käynnistää ohjelman. Ikkunointijärjestelmälle ikonit ovat pieniä ikkunoita.

Ikkunahallintaohjelma (*window manager*, ikkunahallitsija, ikkunapalvelin) on tavallinen asiakas, joka mahdollistaa käyttäjälle ikkunoiden hallinnan. Se siirtää ikkunoita ruudulla, muuttaa ikkunan muotoa ja kokoa sekä ikkunoiden järjestystä, valitsee kohdeikkunan ja muuttaa ikkunan ikoniksi ja päinvastoin. Ikkunanhallintaohjelmien alustustiedostoissa määritellään tavallisesti valikoita (*menu*), jotka avautuvat käyttäjän painaessa jotain hiiren nappulaa juuri-ikkunassa. Tässä työssä näitä valikoita kutsutaan taustavalikoiksi.

ICCCM (Inter-Client Communications Conventions Manual) määrittelee samaa palvelinta käyttävien X-sovellusten välisen yhteistyön. Esimerkiksi hiirellä ”maalaamalla” voi valita tekstiä toisesta ikkunasta ja kopioida sen toiseen ikkunaan ja sovellukseen. Erityisen tärkeää on sovelluksen yhteistyö ikkuna- ja resurssihallitsijan kanssa.

3.1.4 Grafiikka

X:n grafiikkaominaisuuksia ovat viivan, nelikulmion ja erilaisten ympyrämuotojen piirrot. Piirroksen ja taustan väriä, viivan leveyttä, katkonaisuutta, liittymistä toiseen viivaan ja päätepisteen muotoa voidaan säätää. Tämä suoritetaan muuttamalla ohjelman graafista tilaa *(graphics context)* kuten esimerkiksi GKS:ssä. Kuvioista voidaan piirtää vain ääriviivat tai ne voidaan täyttää. Origo on vasemmassa yläkulmassa. Tekstin tulostukseen X käyttää tehokkuussyistä bittikarttakirjasintyyppejä. X tukee sekä 8- että 16-bittisiä kirjaimistoja.

X-termi *drawable* tarkoittaa ikkunaa tai *pixmap*ia. *Pixmap* vastaa muuten ikkunaa, mutta sitä ei näytetä kuvaruudulla, vaan se on palvelimen muistissa. Rasterikuvaa tarkoittava yleiskäsite on siis rajattu tarkoittamaan ikkunointijärjestelmän tietorakennetta. Bitmap on *pixmap*, jossa on vain yksi bitti pikseliä kohden eli kaksivärinen kuva. *Ximage* on kuten *pixmap*, mutta se sijaitsee asiakkaan, eikä palvelimen, muistissa.

Yksi X:n grafiikan ominaisuuksista on värit. X tukee erilaisia väritoteutuksia mustavalkoisesta 32-bittiä/pikseli värikuviin. Sovellus, joka haluaa piirtää punaisen neliön, voi esimerkiksi kysyä palvelimelta väriä vastaavan värikartan indeksin tarvitsematta tietää mitään laitteen todellisista värien esitystavasta. Tosin sovellusohjelmoijan on varauduttava myös mustavalkonäyttöihin.

PEX

X11R5:n mukana on laajennus PEX, PHIGS Extension to X, joka tukee 3D-grafiikkaa ja erityisesti PHIGS:ia ja PHIGS PLUS:aa [Tho90]. Laajennuksen ohjelmointiliittymä on PHIGS:n mukainen, mutta myös muita ohjelmointiliitymiä on olemassa, esimerkiksi suppeampi ohjelmointiliittymä PEXIM, PHIGS (subset) EXtended for Immediate Mode. [XTC91]

PEX-SI toteutuksessa jokainen ikkuna on oma 3D-työasemansa. Toteutus tukee kuvat sisältävän tietorakenteen CSS:n, *Central Structure Store*n, sijoittamista joko palvelimeen tai asiakkaaseen. Samoin kuvanmuodostus voi tapahtua vaiheittain kummassakin. Näin on noudatettu X:n periaatetta tarjota palveluja ja jättää sovelluksen päätettäväksi, miten se niitä käyttää.

3.1.5 Ohjelmointi

X-ohjelmointiin on käytettävissä eritasoisia ohjelmointiliittymiä (*Application Programming Interface*, API). Sovellusohjelma voi käyttää X:ää usean rajapinnan läpi kuvan 4 mukaisesti. Käyttöliittymään perustuvan ohjelman rakenne on erilainen kuin perinteisen kysymys/vastaus-käyttöliittymään perustuvan ohjelman, joka kysyy käyttäjältä syötteet käsittelyn edellyttämässä järjestyksessä. X-ohjelmointi rakentuu vastakutsuperiaatteelle, jossa liitetään funktiot tapahtumiin. Vastaanotettuaan tapahtuman ikkunointijärjestelmä kutsuu sovelluksen funktiota.

Esimerkki

Esimerkkinä X-ikkunointijärjestelmän aiheuttamista ohjelmointiongelmista tässä pohditaan ikkunan paljastumisen vaatimaa kuvan piirtämistä. X tarjoaa neljä ratkaisua.

1. Palvelin voi säilyttää rasterikuvaa muistissaan, *backing store*ssa, ja huolehtia sovelluksesta riippumatta kuvan palautuksesta. Sovellus ei saa luottaa *backing store*n olemassaoloon, sillä sitä ei ole kaikissa palvelimissa (PC) tai käyttäjä voi estää sen. Jollei *backing store* ole käytössä, palvelin lähettää *expose*-tapahtuman sovellukselle, jonka on piirrettävä kuva uudestaan jollakin seuraavista tavoista.

2. Sovellus voi säilyttää rasterikuvaa palvelimen muistissa, *pixmap*ssa, josta se on nopeasti sovelluksen päivitettävissä ikkunaan ilman kuvan siirtoa verkon yli. Ongelmia tulee tässäkin tapauksessa muistiltaan rajoitettujen palvelinten (PC, X-pääte) kanssa.

3. *Ximage* on rasterikuva kuten *pixmap*, mutta se on sovelluksen muistissa. Sen siirtäminen palvelimeen kuormittaa verkkoa, sillä rasterikuvat ovat suuria.

4. Tavallisin vaihtoehto on piirtää kuva aina *expose*-tapahtuman jälkeen X:n piirtoalkiolla. Yksinkertaisten kuvien kohdalla tämä onnistuukin hyvin, mutta monimutkaiset kuvat kuormittavat verkkoa.

Ongelmana on, että ohjelma ei voi tietää, onko sen käyttöympäristössä kriittinen resurssi muisti vai verkko. Ohjelmien pitäisi tarjota käyttäjälle mahdollisuus valita säilytetäänkö kuvaa palvelimen muistissa *pixmap*ssa vai piirtääkö sovellus kuvan aina uudestaan verkon yli.

Xlib

Xlib on kokoelma aliohjelmia ja makroja, joilla voidaan kommunikoida X-protokollan läpi palvelimen kanssa. Xlib muuntaa sovelluksen pyynnöt verkkomuotoon ja lähettää ne palvelimelle. Samoin sovellus voi Xlib:lla ottaa verkosta vastaan tapahtumat. Vaikka ohjelmointiin käytettäisiin ylemmän tason työkaluja, täytyy grafiikkaan käyttää Xlib-aliohjelmia. Ohjelmoinnista tällä tasolla kertoo [Jon89].

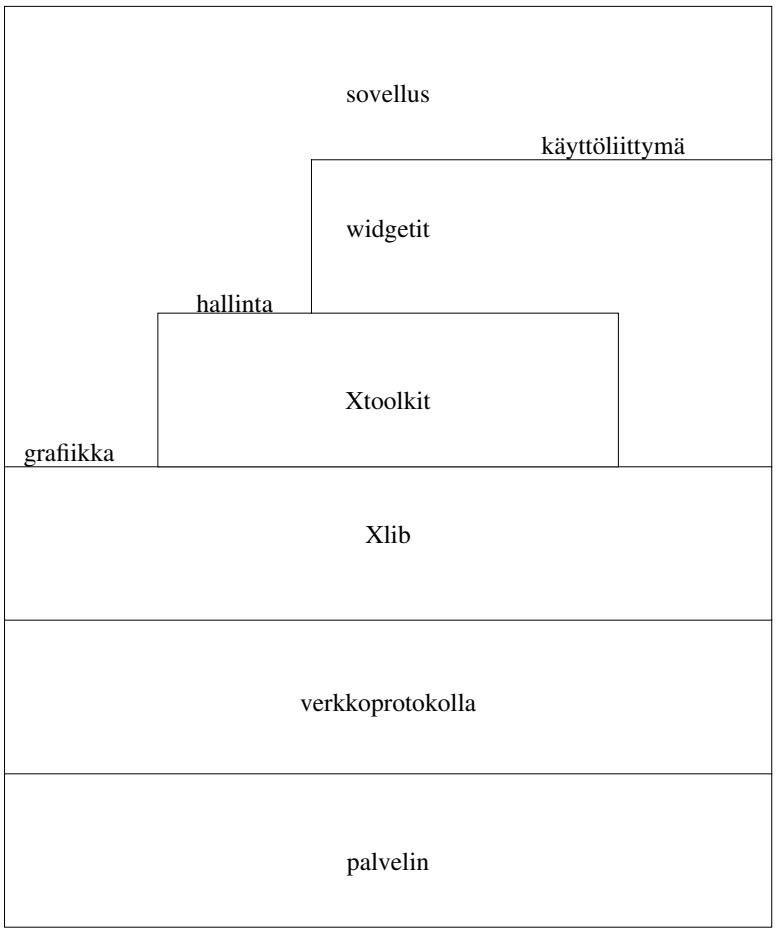
Widget

Widget on tietorakenne, joka yhdistää X-ikkunan ja siihen liittyvät toiminnot. Esimerkiksi vierityspalkki, painonappi, lomake ja tekstilaatikko ovat widgeteja. Widget on olio-ohjelmoinnin olio. Widgetit muodostavat luokkahierarkian, jossa uudet widgetit luodaan jo olemassa olevasta luokasta lisäämällä siihen joitakin uusia ominaisuuksia.

Xtoolkit

Xtoolkit on työkalupakki sovelluksen käyttöliittymän suunnitteluun X-ympäristössä. Se koostuu joukosta yleisiä käyttöliittymäfunktioita ja funktioita widgetien muuttamiseen ja luomiseen. Laite- ja käyttöliittymätoimittajat tarjoavat omia laajempia työkalupakkejaan.

X:n mukana tulee Athena-widget kirjasto. Motif- ja Open Look-käyttöliittymiin kuuluvat omat kirjastonsa. XView-kirjasto poikkeaa muista, sillä se ei pohjaudu Xtoolkitiin ja widgeteihin. Ohjelmoinnista kertoo [You90b]. Kirjasta on OSF/Motif Edition, jonka esimerkit ovat saatavilla tietokoneelle [You90a].



Kuva 4. X-ohjelmoinnin tasot

Graafiset kehittimet

Helppokäyttöisen ja nopeasti opittavan käyttöliittymän ohjelmoinnissa on paljon työtä. Siksi tarvitaan aliohjelmakirjastojen lisäksi käyttöliittymäkehittimiä. Kehittimillä voi poimia käytettävissä olevia widgettejä hiirellä, sijoitella ne ikkunaan ja parametroida. Ohjelmoija piirtää käyttöliittymää ja ohjelma tuottaa siitä automaattisesti valmista koodia. Esimerkkejä näistä *User Interface Development Tool*, UIDT-ohjelmista ovat Architect, UIMX, XBUILD, dirt, XRT/graph ja DevGuide. Erityisen mielenkiintoiselta vaikuttaa Open Interface-ohjelmisto, jolla tehtyjen käyttöliittymien luvataan toimivan ilman muutoksia Macintoshissa, Windowsissa, Motifissa, Open Lookissa ja Presentation Managerissa [Gra91].

Muut kielet kuin C

Tulkattavat ohjelmointikielet sopivat hyvin käyttöliittymäkehitykseen, koska sovelluksen ulkonäköä ja toimintaa voi muunnella ajonaikana ilman raskasta uudelleenkääntämistä. Tulkattavuus kuitenkin hidastaa ohjelman suoritusta, joten rajoittavaksi tekijäksi voi muodostua järjestelmän nopeus [Sav91]. Esimerkkejä muista ohjelmoitiliittymistä ovat [Fek91] ja [May91]

UIL, User Interface Language, on käyttöliittymien kuvailukieli. Sillä voi määritellä käyttöliittymän oliot (valikot, viestit, painonapit...) ja mitä funktioita kutsutaan

käyttäjän toimien tuloksena. UIL:aa käytettäessä määritellään käyttöliittymä omassa tiedostossaan (.uil), joka käännetään *User Interface Definition* eli UID-tiedostoksi uil-kääntäjällä. Sovellusohjelma käyttää MRM:n, *Motif Resource Manager*n, ajonaikaisia funktioita avatakseen UID-tiedoston, jossa on käyttöliittymän kuvaus. MRM rakentaa argumenttilistat ja kutsuu Motif-widgetit luovia funktioita. Koska käyttöliittymä ohjelmoidaan erikseen, voi sovelluksella olla useita käyttöliittymiä. Esimerkiksi käyttöliittymän kieli voi määräytyä ympäristömuuttujan perusteella ohjelmaa käynnistettäessä. [Ope90b]

3.2 NeWS

NeWS on hajautettu ja laitteistoriippumaton ikkunointijärjestelmä kuten X. Myös se perustuu tapahtumiin ja asiakas-palvelin malliin. Sen etuja ovat kehittynyt kuvan mallinnus, joustavuus, laajennettavuus ja täydellinen laiteriippumattomuus. Sun Microsystems kehitti 1984 alkaneessa projektissaan NeWS:n ikkunointijärjestelmäksi työasemilleen. NeWS on suunniteltu siirrettäväksi erilaisille laitteistoille ja käyttöjärjestelmille. NeWS on siirretty esimerkiksi OS/2:lle.

3.2.1 PostScript

PostScript on yksinkertainen, rakenteinen ohjelmointikieli, jossa on erityisen hyvä grafiikka. PostScript on tulkattava ja funktionaalinen. Sitä käytetään sivunkuvauskielenä tulostimissa, joissa se on de facto-standardi. Yleensä PostScript ohjelmia ei kirjoiteta käsin vaan niitä tuottavat sovellusohjelmat tulostuksekseen. [Gos89]

PostScriptin toiminta perustuu pinoarkkitehtuuriin. Tästä seuraa helposti vaikeasti ymmärrättäviä ohjelmia. Perustietotyypit ovat numero, merkkijono ja nimi. Rakenteisia tyyppejä ovat taulukko, jonka alkiot voivat olla mitä tyyppiä tahansa, ja hakemisto, joka yhdistää operaation nimen ja vastaavan toiminnon. [Sys90]

PostScriptin grafiikka perustuu ääriviiva/maalaus-malliin (*stencil/paint model*), jossa viivat tai alueet määritellään algoritmisin ääriviivoin *(path)*. Äärettömän ohuista viivoista ja käyristä koostuvien ääriviivojen rajaaman alueen voi maalata värillä, pintarakenteella tai kuvalla. Malli on laitteistoriippumaton. Se tarjoaa täydellisen 2D-grafiikan korkealla abstraktiotasolla. Grafiikkakäskyillä voidaan [Ado85b, Ado85a]:

1. muuttaa grafiikkatilaa, esimerkiksi asettaa viivan leveys;
2. määritellä kuvaus sisäisestä koordinaatistosta tulostuslaitteen koordinaatistoon, mikä mahdollistaa vapaamuotoiset skaalaukset, venytykset ja pyöritykset;
3. määrätä kynän sijainti ja liikutella kynää;
4. määritellä, muuttaa ja valita kirjasinlaji;
5. muodostaa rasterikuva maalauskäskyillä;
6. hallita kuvan siirto muistista tulostuslaitteelle.

PostScriptin uusi versio Postscript Level II aiheuttaa ongelmia, koska vanhat kirjoittimet eivät ymmärrä sitä. PostScriptin laajennus DaScript Plus sisältää PHIGS PLUS:n toiminnot eli 3D-grafiikan.

3.2.2 Toimintatapa

Asiakas kommunikoi palvelimen kanssa lähettäen sille PostScript-ohjelmia. Käyttäjä voi lähettää PostScript-tiedoston palvelijalle psh-komennolla tai keskustella palvelijan kanssa sen tulkatessa kirjoitetut rivit. PostScriptin etuja ovat verkossa lähetettävän tietomäärän pienuus ja parempi suorituskyky. Koska PostScriptissä voi määritellä funktioita, on järjestelmä laajennettavissa. Usein käytetyt viestijonot voidaan määritellä uusiksi lyhyiksi viesteiksi. Esimerkiksi ruudukon piirtämiseksi ei tarvitse lähettää suurta määrää viivoja, vaan voi lähettää pari silmukkaa sisältävän ohjelman. Verkkoliikenteen määrää voidaan näin huomattavasti pienentää.

NeWS-palvelinprosessi koostuu monista *lightweight*-prosesseista, joilla on yhteinen muisti- ja nimiavaruus, mutta eri pino. Nämä prosessit tulkkaavat asiakkaiden lähettämiä PostScript-ohjelmia, tulostavat kuvaruudulle ja vastaanottavat tapahtumia näppäimistöltä ja hiireltä. Asiakkaat voivat ohjelmoida palvelimen omien erityistarpeittensa mukaiseksi. Asiakas voi esimerkiksi ohjelmoida palvelimen käsittelemään hiiren liikkeet, jolloin verkkoliikenne vähenee.

3.2.3 Grafiikka

Koska NeWS-grafiikka perustuu PostScriptiin, sovellusten ei tarvitse ottaa huomioon kuvaruudun tarkkuutta, tai edes onko se värillinen vai mustavalkoinen. Esimerkki täydellisestä laitteiston ja sovelluksen eristämisestä on, että sovelluksen ei tarvitse edes tietää onko tulostuslaite näyttö vai kirjoitin. Kuitenkin NeWS-palvelin voi helposti hyödyntää laitteiston erikoismahdollisuuksia kuvan tuottamisessa. [Gos89]

3.2.4 Ohjelmointi

NeWS:a ohjelmoidaan PostScript-kielellä, jota on huomattavasti laajennettu. Ohjelmointi perustuu olioille. Käytettävissä on työkalupakkeja (Lite, TNT), jotka sisältävät valmiita luokkia valikkojen, painonappien, vieritinpalkkien ym. olioiden luomiseksi. Esimerkiksi komento */new LiteWindow send* luo ilmentymän LiteWindow-luokan oliosta lähettämällä oliolle sanoman **uusi**.

Canvas on alue ruudulla, jossa asiakassovellus voi esittää tietoa käyttäjälle. Canvas vastaa X:n ikkuna-käsitettä muodostaen samanlaisen hierarkkisen järjestelmän. Tapahtumat välittävät kaikenlaisia sanomia NeWS-prosessien välillä. Myös C-ohjelmat voivat kommunikoida palvelijan kanssa.

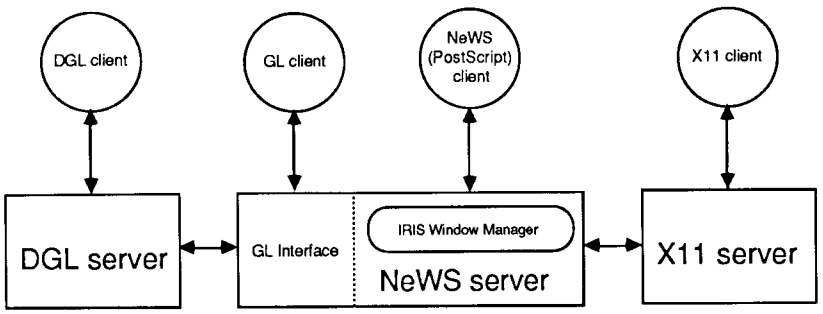
3.3 Muut ikkunointijärjestelmät

OpenWindows on Sun Microsystemsin ikkunointijärjestelmä, jossa on sekä X11- että NeWS-palvelin. Lisäksi samalla ruudulla voi olla Sunin oman ikkunointijärjestelmän SunViewn ikkunoita, mutta niiden toiminta ei ole taattua. OpenWindows toteuttaa Open Look-käyttöliittymän. OpenWindowsiin kuuluu Open Look-käyttöliittymän mukainen ikkunanhallintaohjelma olwm (Open Look Window Manager) ja työpöytäohjelmia (File Manager, Mail Tool, Print Tool, Tape Tool, Text Editor, Calculator, Calendar Manager...).

Display PostScriptissä, DPS:ssä, PostScriptiä on laajennettu sopivaksi vuorovaikutteiseen moniprosessointiympäristöön. DPS:ssä ei ole olio-ohjelmointilaajennusta tai mahdollisuutta ohjelmoida palvelinta käsittelemään tapahtumia kuten NeWS:ssä.

NeXT:n DPS:ä voi käyttää myös C-ohjelmissa ja laajennus kattaa myös kuvankäsittelyn.

SGI:n (Silicon Graphic Inc./Iris) työasemien ikkunointijärjestelmä on 4Sight. Siinä on kuvan 5 osoittamalla tavalla yhdistetty kolme palvelinta: DGL, NeWS ja X11. 4Sight ja Iris window manager pystyvät siis hallitsemaan samanaikaisesti useita GL-, DGL-, PostScript- ja X11–ikkunoita ja -asiakkaita. Graphics Library (GL) on 3D-grafiikkakirjasto. Se vaatii erikoislaitteiston, joka on tällä hetkellä olemassa Silicon Graphics- ja IBM-työasemille ja PC:lle. Hajautettu versio Distributed Graphics Library (DGL) tulee myös Craylle ja muille superkoneille. SGI siirtyy uudessa käyttö-

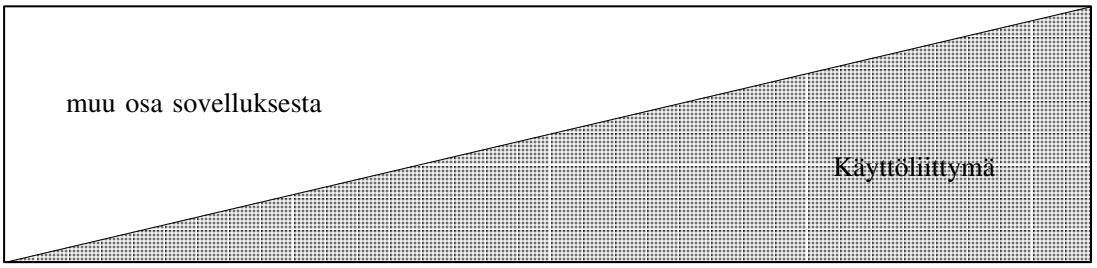


Kuva 5. SGI:n ikkunointijärjestelmä 4Sight

järjestelmäversiossaan IRIX 4.0 uuteen X11R4- ja Motif 1.1–pohjaiseen ikkunointijärjestelmään. DGL ja GL yhdistyvät. NeWS korvautuu Display PostScriptilla, joka on myös PostScriptin laajennuksiin perustuva ikkunointijärjestelmä. Valitettavasti laajennukset ovat erilaisia, eli NeWS ja Display PostScript eivät ole yhteensopivia.

4. Käyttöliittymät

Käyttäjän havaitsema osa ohjelmasta on käyttöliittymä *(HCI, Human-Computer Interaction, User Interface)*. Kuvan 6 mukaisesti käyttöliittymän osuus vaihtelee erityyppisissä ohjelmistoissa. Tieteellisen visualisoinnin ohjelmistoissa käyttöliittymän osuus on suuri.



EDI (Electronic Document Interchange)

Tieteellinen visualisointi

Kuva 6. Käyttöliittymän osuus eri sovelluksissa [KB91]

4.1 Ominaisuudet

Hyvä käyttöliittymä on tehokas: tehtäviin kuluu vähän aikaa. Uusi käyttäjä oppii sovelluksen käytön nopeasti ja sovelluksen käyttö palautuu hyvin mieleen pitkän tauon jälkeen tai epäsäännöllisessä käytössä. Käyttäjän virheiden vähäinen lukumäärä on erityisen tärkeä esimerkiksi lääketieteellisissä, ilmailullisissa tai sotilaallisissa sovelluksissa. Lisäksi käyttöliittymän pitäisi houkutella ohjelman ostajia ja viehättää käyttäjää. [Fol90]

Yhtenäinen käyttöliittymä mahdollistaa käyttäjän oppimien menetelmien ja toimintatapojen hyödyntämisen yli sovellus- ja laitteistorajojen. Siirtyessään uusiin sovelluksiin käyttäjän ei tarvitse opetella uusia hallintatapoja. Tämä vähentää muutosvastarintaa ja koulutustarvetta. Ohjelmistojen pitää olla johdonmukaisia sekä sisäisesti että verrattuna muihin ohjelmiin.

Lisäksi käyttöliittymän pitää olla joustava, eli käyttäjän on voitava muokata sovellusta mieltymystensä mukaan. Esimerkiksi sovelluksen käytön olisi oltava mahdollista sekä näppäimistöltä että hiirellä. Vasenkätisen tai muuten erilaiseen hiireen tottuneen pitäisi voida vaihtaa hiiren näppäinten järjestys. Toiminnot pitää voida perua tai käyttäjältä pyytää kuittaus ennen toimintoa [Ope90c].

4.2 Graafiset käyttöliittymät

Graafiset käyttöliittymät (*Graphical User Interface*, GUI) perustuvat suorakäsittelyyn *(direct manipulation)*. Oliot, niiden ominaisuudet ja suhteet kuvataan visuaalisena esityksenä, jota käyttäjä käsittelee. Visuaalinen esitys voi olla kuvina, ikoneina tai tekstinä. Hyvin suunnitellut ikonit voivat kuluttaa vähemmän tilaa näytöllä ja olla tunnistettavissa nopeammin kuin teksti. Lisäksi ne voivat olla ymmärrettävissä kansainvälisesti. Suora käsittely muistuttaa todellista maailmaa, jossa saadaan välitön näkyvä vastine käyttäjän toiminnolle.

Suorakäsittelyä toteutetaan eri tavoilla. ”Siirrä ja sijoita”-mallissa (*drag and drop*) käyttäjä poimii kohteen, siirtää sen toisen sovelluksen päälle ja pudottaa sen käsiteltäväksi. Esimerkki tästä on tiedostojen tuhoaminen siirtämällä ne roskakoriin. Toinen käytetty malli on ”leikkaa ja liimaa” (*cut and paste*), joka eroaa ”siirrä ja sijoita”-mallista kaksivaiheisuudellaan. Ensimmäisessä sovelluksessa data siirretään tai kopioidaan leikekirjaan *(clipboard)*, josta se kopioidaan toiseen sovellukseen.

WYSIWYG (What You See Is What You Get; saat mitä näet)-sovelluksissa kuvaruudulla näkyvä kuva, jota käyttäjä muokkaa, näyttää samalta kuin lopputulos. Koska näyttö ja lopullinen tulostinlaite kuten laserkirjoitin harvoin ovat yhtä tarkkoja, erityisesti värien vastaavuus ei ole täydellinen. Tavallisimpia WYSIWYG-sovelluksia ovat tekstinkäsittelyohjelmat.

4.2.1 Motif

OSF/Motif on Open Software Foundationin (OSF) X-ympäristössä toteutettu käyttöliittymä. Ryhmittymän perustivat IBM, DEC, HP, Apollo, Siemens ja Nixdorf vastustamaan AT&T:n pyrkimyksiä tehdä Unixistaan yleinen standardi.

Motif-käyttöliittymä koostuu ikkunanhallintaohjelmasta mwm *(Motif Window Manager)*, työkalupakista, UIL-kuvauskielestä, tyylioppaasta OSF/Motif Style Guide ja näihin perustuvista sovelluksista [Ope90c]. Tyyliopas määrittelee käyttöliittymän ulkonäön ja toiminnan. Koska OSF/Motifin tyyliopas perustuu HP:n ja Microsoftin yhteiseen projektiin, vastaa sen käyttäytyminen OS/2 Presentation Managerin käyttäytymistä niin ikkunoiden rakenteen, valikkojärjestelmien, hiiren toiminnan kuin yleisten käyttömekanismien osalta. [Kar90]

OSF käyttää ohjelmointiliittymästä (API) termiä Application Environment Specification (AES). Motifin ohjelmointiliittymän määrittelee [Ope90a]. Käyttäjän oppaan [Ope90d] mukaan hiiren näppäinten toiminnot ovat taulukon 1 mukaiset. Ohjelmiston

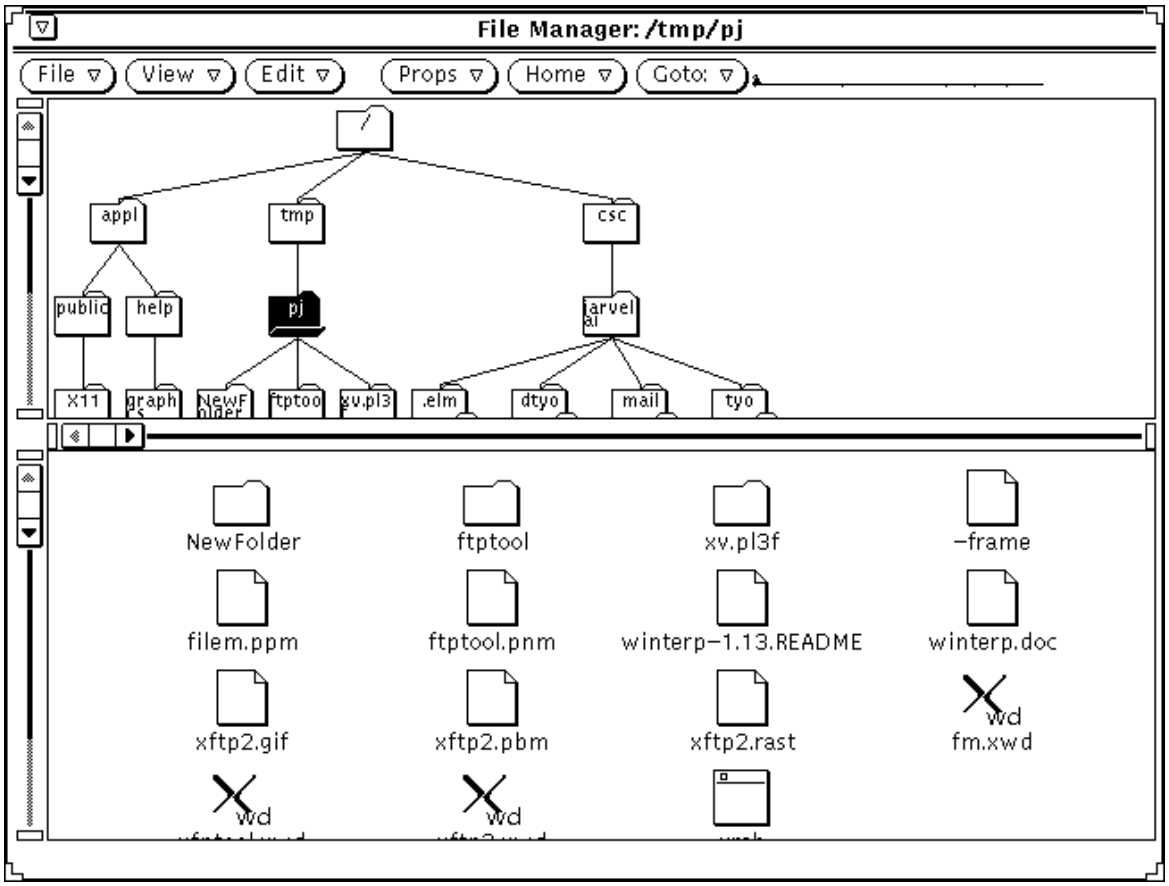
vasen	valinta	*select*
keskimmäinen	valikko	*menu*
oikea	sovellus	*custom*

Taulukko 1. Hiiren näppäinten toiminnot Motif-käyttöliittymässä

mukana nauhalla tuleva OSF/Motif Style Guide kuitenkin kertoo keskimmäisellä näppäimellä siirreltävän ja kopioitavan olioita *(drag)* [Ope90e]. Dokumentoinnin ja ohjelmistojen virheellisyyksien vuoksi lähdekoodi on tarpeellinen ainakin omien sovellusten tuottamisessa.

4.2.2 Open Look

Open Look on UI:n (Unix International) graafinen käyttöliittymä. UI on kansainvälinen tietokonevalmistajien (AT&T, Sun, Gould, Olivetti, Toshiba, Unisys, jne) yhteistyöelin. Open Lookin määrittelevät Open Look Graphical User Interface Trademark Guide ja [Sun90a, Sun90b]. Ikkunointiympäristön, työkalupakin sekä tiedosto- ja prosessinhallintaohjelmien toiminnallisten määrittelyjen olemassaolo mahdollistaa eri toteutusten tekemisen käyttöliittymästä. Tämä on merkittävä etu Motifiin verrattuna, jossa on standardoitu paremminkin ohjelmointiliittymä tietylle käyttöliittymälle kuin itse käyttöliittymä. Eri ikkunointijärjestelmille onkin useita Open Lookin mukaisia ohjelmointiliittymiä, esimerkiksi X:lle XView ja OLIT (Open Look Intrinsics Toolkit) ja NeWS:lle NDE ja TNT (The NeWS Toolkit).



Kuva 7. Open Look File Managerin perusikkuna

File Manager kuvassa 7 on OpenWindows toteutus Open Look tiedostonhallintaohjelmasta. File Managerissa käytetään sekä ”siirrä ja sijoita” että ”leikkaa ja liimaa” käyttöliittymämalleja. File Manager ja roskakori tuovat Macintoshin ja PC:n ikkunointikäyttöliittymien helppouden myös X-Unix-ympäristöön. Myös Motif-käyttöliittymän mukaisia File Manageria vastaavia työpöytäohjelmia on saatavilla. Aloittelijalle voidaan luoda sovelluskohtainen käyttöympäristö, jossa hänen ei tarvitse tietää mitään Unixista. Harvoin käytetyt tai vaikeat Unix komennot kuten **find** on kokeneenkin käyttäjän helpompi valita File Managerin valikoista.

Taustavalikon ”*properties...*”–valinnalla saa ikkunan, jolla voi muunnella ympäristöä sopivaksi. Sillä voi vaikuttaa väreihin, ikonien sijoitteluun, valikoihin, hiiren

asetuksiin, äänimerkkeihin ja vieritintangon sijaintiin. Myös jotkut laitetoimittajat ovat toteuttaneet samanlaisia ohjelmia Motif-ympäristöönsä. Samoja asioita voi X-ympäristössä yleisemminkin asettaa alustustiedostoja editoimalla.

Hiiren näppäinten toiminnot ovat:

1. valinta,
2. säätö, tarkentaa valintaa tai poistaa valinnat ja
3. valikko, näyttää valikon ja valitsee.

Erona on, että Motifissa valinnat tehdään myös valikoista vasemmanpuoleisella valinta-näppäimellä, kun taas Open Lookissa valikkoa käsitellään oikeanpuoleisella valikko-näppäimellä. Toiminnallisten erojen lisäksi Motif ja Open Look eroavat toisistaan myös termistöltään.

4.3 Kommunikaatiomuodot

Valikkoliittymissä ohjelma esittää käyttäjälle luettelon kussakin tilanteessa mahdollisista toimenpiteistä. Käyttäjä valitsee valikosta haluamansa vaihtoehdon hiirellä tai näppäimistöllä. Lomakeliittymässä tietokoneen näytöllä täytettävä lomake muistutaa paperista esikuvaansa. Valikkoja ja lomakkeita käytetään sekä graafisissa että tekstipohjaisissa käyttöliittymissä. Luonnolliset kielet ovat toistaiseksi olleet liian vaikeita tietokoneen ymmärtää. Siksi yleisempi onkin yksinkertainen ja täsmällinen komentokieli. Komentokieleen kuuluvat komentosanat ja niihin liittyvät lisämäärittelyt. Toistaiseksi komennot kirjoitetaan näppäimistöltä kunnes puheentunnistus yleistyy. Kysymys/vastaus-liittymissä ohjelma kysyy käyttäjältä syötteet käsittelyn edellyttämässä järjestyksessä. Taulukossa 2 [Fol90] s. 403 vertaillaan ihmisen ja tietokoneen välisiä kommunnikaatiomuotoja. WYSIWYG ei ole täydellinen käyttöliittymämalli, joten sen kohdalla on taulukossa aukkoja. Taulukosta puuttuvat kyselykieli- ja toimintanäppäinkäyttöliittymät.

Taulukko 2. Käyttöliittymien vertailua

	lyhyt oppimis-aika	käytön nopeus	pieni virhe-alttius	laajennet-tavuus	ei tarvita konekirjoitus taitoa
WYSIWYG	+		+	-	
suora-käsittely	+	+/-	+	-	+
valikot	+/-	+/-	+	+/-	+
lomake	+	+	+	+/-	-
komento-kieli	-	+	-	+	-
luonnollinen kieli	+	+/-	-	+	-
kysymys/ vastaus	+	-	+	+	-

4.4 Johtopäätöksiä

Graafiset käyttöliittymät ovat yleensä helpompia oppia ja yhtenäisempiä kuin tekstipohjaiset, mutta käyttö saattaa olla hitaampaa, jollei käyttöliittymä ole joustava ja tarjoa kokeneelle käyttäjälle mahdollisuutta hyödyntää kokemustaan. Hyvässä käyttöliittymässä onkin sopivasti yhdistelty eri tekniikkoja, esimerkiksi lomakkeita, makroja ja grafiikkaa. Tyyliohjeiden noudattaminen parantaa käyttöliittymien laatua.

Vaikka Open Look- ja Motif-ohjelmia voi ajaa samalla ruudulla, käyttäjä menettää tällöin käyttöliittymän yhtenäisyyden. Kumpaakin käyttöliittymää tukee niin suuri määrä rahaa ja arvovaltaa, ettei kummankaan nopea katoaminen ole odotettavissa [Sys91].

5. Supertietokoneympäristö

Aikakautensa tehokkaimpia tietokoneita nimitetään supertietokoneiksi [Lum90a]. Tunnetuin tehokkuuden mittari on vektoroituvaan liukulukulaskentaan keskittyvä Dongarran LINPACK-testi. Erityisesti suurissa tehtävissä tehokkuus edellyttää nopeaa muistia ja väylää.

Ohjelmistot ovat tärkeämpiä kuin laitteisto supertietokoneympäristössä kuten yleensäkin tietojenkäsittelyssä. Tätä todistaa suljettuun IBM:n arkkitehtuuriin perustuvien japanilaisten superkoneiden huono menestys verrattuna yhdysvaltalaisten Unix-koneisiin.

5.1 Tieteellisen laskennan palvelu

Tieteellisen laskennan palvelu (TLP) on tieteelliseen tietojenkäsittelyyn erikoistunut Valtion tietokonekeskuksen (VTKK) tulosyksikkö. Tieteellisen laskennan palvelu huolehtii valtiovarainministeriön omistaman Cray X-MP-supertietokonejärjestelmän ja opetusministeriön Convex-tietokoneen käyttöön, ylläpitoon ja tieteellisen laskennan asiakastukeen liittyvistä tehtävistä.

TLP:n toimintaa valvovat valtiovarainministeriön asettama supertietokoneen pääkäyttäjien edustajista koostuva supertietokonejohtoryhmä ja opetusministeriön asettama tieteellinen johtoryhmä. Supertietokonetoiminnan rahoittavat pääkäyttäjät sovittujen käyttöosuuksien suhteessa. Käyttöosuudet ovat vuonna 1992 seuraavat:

60% opetusministeriö ja korkeakoulut,
25% Ilmatieteen laitos ja
15% Valtion teknillinen tutkimuskeskus, VTT.

5.1.1 Tiedotus

TLP julkaisee kolmea lehteä: Korkeakoulujen atk-uutiset, /csc/news ja Supermenu. Korkeakoulujen atk-uutiset on opetusministeriön ja korkeakoulujen yhteinen lehti. /csc/news on englannin kielinen superlaskennan lehti, jossa on myös lyhyitä artikkeleita tieteellisen laskennan tutkimusprojekteista. Supermenu sisältää ohjeita, vihjeitä ja muuta teknistä tietoa käyttäjille. Lisäksi TLP järjestää seminaareja sekä koulutusta laitteistojen ja ohjelmistojen käytöstä.

Tietotekniikkaa käytetään myös tiedotuksessa; TLP:n tiedotukset julkaistaan valtakunnallisessa uutisryhmässä **csc.tiedotukset**. TLP:n X:n käyttäjille on **nic.funet.fi**-koneella paikallinen uutisryhmä **csc.x**. Pysyvät ohjeet käyttäjille on tallennettu /appl/help-hakemistoon **finsun.csc.fi**-koneella ja henkilökunnalle /csc/public/ohjeet-hakemistoon.

5.1.2 Cray X-MP EA/432

Cray X-MP EA/432 on Suomen ensimmäinen ja toistaiseksi ainoa supertietokone. Koneessa on neljä suoritinta ja 32 megasanaa eli 256 megatavua keskusmuistia. Saman verran on virtuaalilevyä SSD (Solid-state Storage Device).

Käyttöjärjestelmä on System V Unixiin perustuva ja Berkeley laajennuksilla täydennetty Unicos, joka täyttää ISO:n POSIX-stadardin [ISO90]. Erityispiirteitä ovat virtuaalimuistin puuttuminen ja eräajojärjestelmä NQS (Network Queuing System).

5.1.3 Edustakoneet ja hajautus

TLP:n superkoneympäristö on hajautettu. Laskentapalvelimena toimivan Crayn lisäksi ympäristöön kuuluu tiedostopalvelin ja graafisia palvelimia. Koneita pyritään käyttämään niille parhaiten sopiviin tehtäviin. Crayllä vuorovaikutteisen käytön sijasta suositaan erätoitä ja muun kuin laskennan tekemistä toisilla koneilla. Esimerkiksi emacs-editoria ei ole ollut Crayllä, vaan kirjoittaminen on tarkoitettu tehtäväksi edustakoneilla. Finsunissa on emacsissa laajennus, jolla Crayssä olevaa tiedostoa voi editoida kuten paikallistakin.

Lisenssiehtojen vuoksi Craytä voi käyttää vain jonkin edustakoneen kautta. Crayn edustakoneena ovat TLP:ssa Sun 4/390 (**finsun**) ja Convex 3840 (**convex**) ja joukko työasemia. Kaavio TLP:n laiteympäristöstä on liitteessä 3. Asiakkailla on omia edustakoneita, esimerkiksi VTT:lla on oma Sun 4/390 (**vttux1**).

5.1.4 Tiedostopalvelin

Ennen tiedostopalvelimen hankintaa supertietokoneympäristön pahimpana pullonkaulana on ollut sekä pysyvään tallennukseen tarkoitetun että ajoaikaisille tilapäis- ja työtiedostoille varatun tallennustilan puute. Superlaskentaympäristössä syntyy päivässä noin 2 - 3 gigatavua eli vuodessa 0,5 - 1 teratavua arkistoitavaa dataa. Supertietokoneen varmuuskopiotallenteiden ja arkistonauhakasettien määrä on tällä hetkellä 400 gigatavua. Tallenteiden ylläpito on ollut ongelmallista. Supertietokoneen ympäristön laitteistoissa on tähän saakka ollut käytettävissä levytilaa yhteensä noin 40 gigatavua.

Convex C220-tiedostopalvelinratkaisu perustuu UniTree-ohjelmistoon, joka huolehtii myös automaattisesti pitkään käyttämättä olleiden tiedostojen siirtämisestä nauhalle. Tallenteiden arkistoksi vuokrataan Valtion Tietokonekeskukselta StorageTek-kasettirobotin kapasiteettia. Convex C220-laitteistossa on kaksi prosessoria ja 128 megatavua keskusmuistia. Tiedostopalvelimen ensisijaiseksi tallennusvälineeksi tulee noin 30 gigatavua keskinopeaa levyä (3 MB/s). Levyryhmätekniikat *(Disk Striping, Redundant Array of Inexpensive Disks (RAID))*, joissa levyasema on koottu useista levyistä, nostavat todelliset siirtonopeudet tasolle 10-20 MB/s.

5.1.5 Lähiverkko

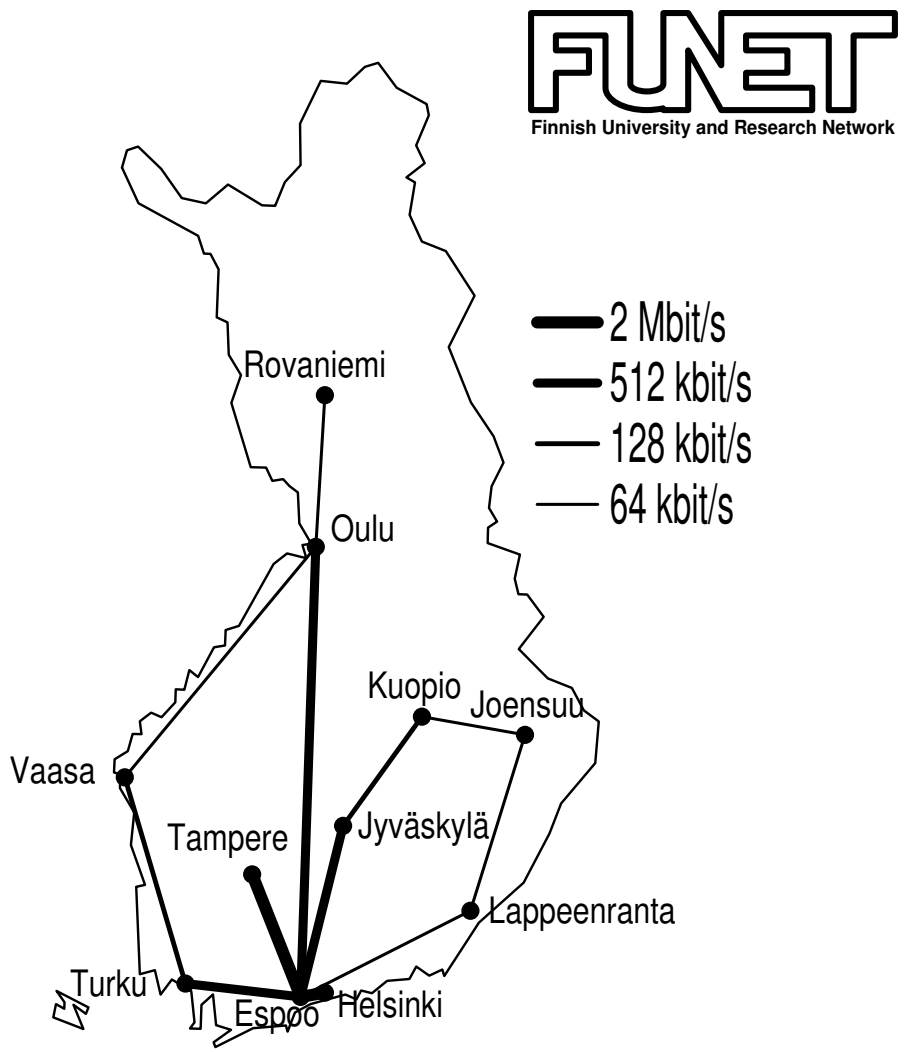
Supertietokoneympäristön hierarkkisen lähiverkon ytimenä on UltraNet. UltraNetiin on liitetty suurta tiedonsiirtonopeutta vaativat tietokoneet: Cray, Convex C3840, Convex C220 -tiedostopalvelin, Iris 4D/380VGX-visualisointilaitteisto ja Ciscon Ultra/FDDI/Ethernet-reititin. Reititin välittää liikennettä UltraNetin ja FDDI-verkon sekä Ethernet-verkon välillä. UltraNetin siirtonopeus maksimissaan on 1 Gbit/s; TLP:ssa käytettävät liitännät tukevat 100-125 Mbit/s-nopeutta. FDDI-verkon maksimisiirtonopeus on 100 Mbit/s ja Ethernet -verkon 10 Mbit/s.

Televisiokuvan tasoisen animaation siirtäminen vaatii 180 Mbit/s nopeuden tietoliikenneverkolta. Koska työaseman kuvaruutu on televisiokuvaa suurempi tarkkuu-

deltaan, vastaava nopeus on 780 Mbit/s [Sin90]. Vaikka kuvaa yleensä tiivistetään verkossa siirrettäessä, verkon on syytä olla mahdollisimman nopea.

5.1.6 FUNET

Finnish University and Research Network (FUNET) on sekä valtakunnallinen tietoliikenneverkko että kyseistä verkkoa ylläpitävä ja kehittävä TLP:n kustannusyksikkö. FUNET-verkko yhdistää TLP:n, Suomen korkeakoulut sekä useita tutkimuslaitoksia ja yritysten tutkimusosastoja maanlaajuiseksi tutkimusverkoksi. Kuvassa 8 on esitetty FUNET-runkoverkko lokakuussa 1991. FUNET on osa maailmanlaajuista internet-verkkoa. Tällä hetkellä tiedonsiirtonopeudet ovat välillä 64 Kbit/s ja 2 Mbit/s.



Kuva 8. Funet

Funetilla on oma tiedostopalvelin, jonka nimi on **nic.funet.fi**. **Nic** muistuttaa kirjastoa, sillä se on yleisesti käytettävissä verkossa. **Nic**issä on saatavilla julkisohjelmia ja muita tiedostoja. Esimerkki tiedostopalvelijan käytöstä löytyy liitteestä 4. Esimerkissä käytetyllä anonymous ftp:llä kaikkien internetissa olevien koneiden käyttäjät voivat siirtää tiedostoja antamalla käyttäjän nimeksi anonymous tai ftp.

Tämän diplomityön lähdeluettelossa on dokumentteja, jotka on julkaistu vain tiedostoina. Vuosiluvun jälkeen viitteessä on mainittu dokumentin tyyppi, koneen.nimi[koneen.verkko.osoite]:hakemistopolku/tiedosto tai hakemisto. Ennen vuosilukua on tekijän sähköpostiosoite. Liitteen 4 esimerkkiin viitattaisiin ”Uusi Testamentti. ASCII-dokumentti, nic.funet.fi[128.214.6.100]:-pub/doc/bible/finnish/ut.asc.Z-tiedosto”. Levytilan säästämiseksi lähes kaikki tiedostot on pakattu. Tiedostopalvelinten käyttöä vaikeuttaakin tiedostojen pakkaus, vaikka pakkausohjelmat ovatkin saatavilla tiedostopalvelimissa. Liitteen 4 esimerkissä käytetään uncompress-ohjelmaa tiedoston palauttamiseksi ASCII-muotoon.

Useita tiedostoja on tiedostopalvelimessa yhdistetty yhdeksi tiedostoksi tar-ohjelmalla, jota tarvitaan tiedostojen saamiseksi taas erillisiksi. Dokumentit ovat tavallisimmin ASCII- tai PostScript-tyyppisiä. ASCII-dokumentit ovat luettavissa ja tulostettavissa paperille sellaisenaan, lukuunottamatta mahdollisia vaikeuksia ä- ja ö-merkkien kanssa. Jollei käytettävissä ole PostScriptiä tulkitsevaa kirjoitinta, tiedostoja voi lukea ghostscriptillä tai muilla tulkkiohjelmilla.

5.2 Grafiikka

5.2.1 Ikkunointijärjestelmät

Kaikilla TLP:n tietokoneilla on käytettävissä X-ikkunointijärjestelmä. NeWS on käytettävissä joko erikseen tai toisen ikkunointijärjestelmän osana kaikilla muilla koneilla paitsi Convexeilla. **Finsun**ille on asennettu OpenWindows ikkunointijärjestelmän versio 2. Palvelimen käyttö vaatii Sun4-työaseman. SGI:n työasemilla käytetään 4Sight:a.

Monet TLP:ssä käytetyt sovellusohjelmat osaavat hyödyntää X:ää, esimerkiksi publisher, framemaker, maple, fidap, fluent, nekton, patran, apE, khoros, disspla, NCAR, matlab, idl ja mathematica, joka tukee myös NeWS:iä.

5.2.2 Käyttöliittymät

Tätä diplomityötä aloitettaessa TLP:ssa olivat käytössä ikkunointijärjestelmien mukana tulleet käyttöliittymät. Puuttuvat Motif ja Open Look hankittiin ja otettiin käyttöön osana tätä työtä. TLP:lle hankittiin Motifin lähdekoodi, koska sen tutkimus- ja oppilaitoshinta on edullinen verrattuna binäärien hintaan. Käyttöliittymäkirjastot ovat osoittautuneet tarpeellisiksi, sillä jotkut sovellusohjelmat käyttävät niitä. TLP:ssa on käytössä myös olvwm-ikkunanhallintaohjelma.

TLP:n oma ohjelmistuotanto on vähäistä. Asiakkaat kirjoittavat omat ohjelmansa tai käyttävät valmisohjelmia. TLP ei siis juurikaan pysty vaikuttamaan käytettävien ohjelmien käyttöliittymiin. Tavoitteena on tarjota hyvä käyttö- ja ohjelmankehitysympäristö. TLP:lle graafisten käyttöliittymien merkitys on, että asiakkailla on mahdollisuus käyttää tuttua käyttöliittymää. Kahden graafisen käyttöliittymästandardin haitat korostuvat TLP:n tapaisessa hajautetussa monitoimittajaympäristössä.

Suurin epäkohta TLP:ssä on epäyhtenäisyys. TLP:ssa on nyt viisi eri Unix-ympäristöä: Crayn Unicos, Convexien ConvexOS, Iristen IRIX sekä Sun4:n ja Sun3:n eri versiot SunOs:sta. Pyrkimyksissä yhtenäiseen hakemistorakenteeseen on onnistuttu vain käyttäjien kotihakemistojen osalta.

5.2.3 Kuvaformaatit

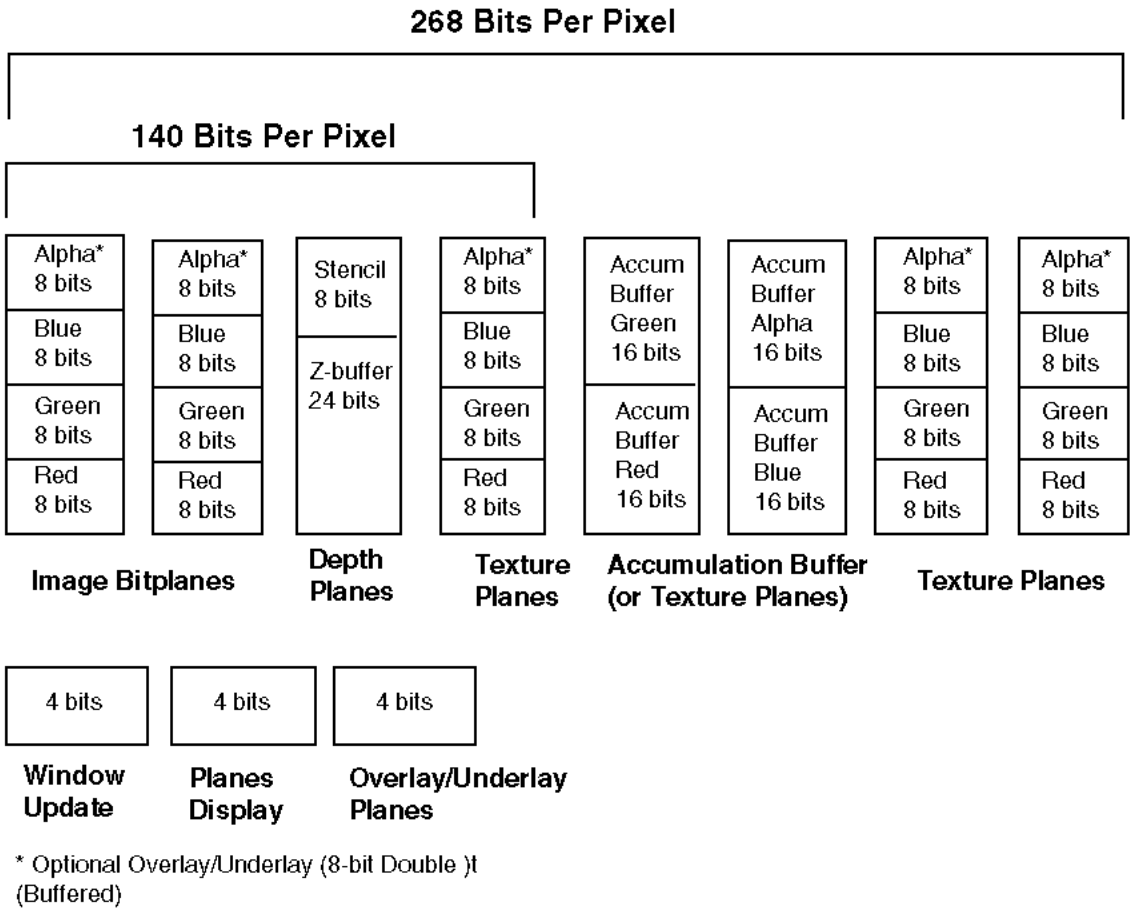
Erilaiset tiedostoformaatit aiheuttavat paljon turhaa työtä. Rakenteelliset tiedostot on standardoitu, mutta rasteritiedostoformaatteja on käytössä liian monta. Macintoshin kuvanlukija tuottaa tiff-kuvia. Publisher-julkaisuohjelma, jolla tämä diplomityökin on taitettu, pystyy lukemaan vain Sunin rasteritiedostoja ja PostScriptia ulkopuolisista formaateista. Xwd-ohjelmalla, jolla voi kaapata X-ikkunassa olevan kuvan tiedostoon, on omat tiedostoformaattinsa. Pienimpään tilaan kuvat pakkaa gif-formaatti. JPEG-formaatilla saavutetaan gif-formaatia parempi tiivistyssuhde värikkäillä ja mutkikkailla kuvilla, jos voidaan sallia informaation menetys tiivistyksessä eli tiivistetty kuva eroaa alkuperäisestä.

Kuvaformaattia valittaessa kannattaisi miettiä, onko kuvaformaatti tiivistetty, niin että se säästää levytilaa sekä kuvan kirjoittaminen levylle ja lukeminen levyltä käy nopeasti, vai tehdäänkö tiivistys tarvittaessa esimerkiksi Unix:n compress-ohjelmalla. TLP:ssä on osana tätä diplomityötä otettu käyttöön pbmplus- ja imtools-kuvaformaattien muunnosohjelmistot [Pos91, Imc91]. Pbmplus-kuvankäsittelypaketti koostuu suuresta joukosta ohjelmia, jotka muokkaavat kuvia eri formaateista pbmplussan formaatteihin ja pbmplussan formaateista toisiin formaatteihin. Kuvia voi myös käännellä, leikata ja niiden kokoa muuttaa. Imtools on hieman suppeampi ohjelmisto, joka suorittaa muunnokset suoraan.

5.2.4 Laitteisto

TLP:n visualisointilaboratoriossa on Silicon Graphics Iris 4D/380VGX grafiikkapalvelimeen **Azalea**an liitetty animaatiolaitteisto, joka koostuu Sonyn laservideolevystä, U-Matic- ja VHS-videonauhureista ja tarvittavista hallintalaitteista. Kaavio visualisointilaboratoriosta on liitteessä 5. Videoita on tuotettu noin kuusi tuntia. Lisäksi käytössä on viisi Personal Iris-työasemaa. Värikuvat paperille tai piirtoheitinkalvolle tulostetaan Tektronixin Phaser PostScript-tulostimella. Liitteiden 1, 2 ja 6 värikuvat on tulostettu Phaserillä. Muita tulostuslaitteita ovat Versatec-väripintapiirturi ja Benson-kynäpiirturi. [TLP90]

Grafiikkapalvelin **Azalea**ssa on 8 kappaletta 33 MHz Mipsin R3000 prosessoreja, muistia 192 megatavua, 2,2 gigatavua levyä ja VGX-grafiikkakortti. VGX-grafiikka toteuttaa laitteistolla monet yleensä ohjelmiston laskemat operaatiot: väritasa-arvoalueet, pehmennykset, *athmospheric effects*, *motion blur*, objektien leikkaukset mielivaltaisilla tasoilla ja useat valaistusmallit. **Azalea**ssa on 19” 60Hz (120Hz lomitettu stereo) 1280x1024 pikselin kuvaruutu. VGX-grafiikassa jokaista pikseliä varten on kuvan 9 mukaisesti 140–268 bittiä dataa, mikä yksistään vaatii 23 megatavua muistia. Kuva muodostetaan RGB-värijärjestelmällä ja kunkin värin intensiteettiä varten on 8 bittiä.



Kuva 9. SGI VGX bittitasot

5.3 Visualisointiohjelmistot

Käyttäjän näkökulmasta visualisointi voidaan jakaa kolmeen ryhmään:

1. numeroaineiston muuttaminen kuvaksi,
2. julkaisukelpoiset kuvat ja
3. animaatio ja realistiset kuvat.

5.3.1 Numeroaineiston muuttaminen kuvaksi

Tavallisin tehtävä visualisoinnisssa on numeroaineiston muuttaminen kuvaksi. Monet sovellusohjelmat tekevät sen itse tai data voidaan visualisoida taulukkolaskenta- tai muulla tutulla ohjelmalla. TLP:n X-ympäristössä tällaiseen visualisointiin on apE sekä NCSA:n Ximage ja XDataSlice.

HDF ja Ximage

HDF (Hierarchical Data Format) on NCSA:n tiedostoformaatti kuvien, lukujen ja tekstin siirtämiseen eri tietokoneiden välillä. Yksi tiedosto voi sisältää useita kuvia ja dataa.

HDF-luku- ja -kirjoituskutsuja käyttävillä ohjelmilla luodut tiedostot voi siirtää koneelta toiselle ilman mitään muunnoksia. HDF-tiedostoihin voi myös tallentaa vapaamuotoisen kuvauksen tiedoston sisällöstä, kenttien otsikot, kuvan koon ja muutkin tarpeelliset tiedot. HDF-tiedosto on siis itsedokumentoiva.

NCSA Ximage on datan ja värikuvien analysointityökalu, joka toimii X-ikkunointijärjestelmässä. Se näyttää tiedoston sisällön numeroina tai kuvana suorakulmaisessa tai napakordinaatistossa. Kuvan väritystä voi käsitellä ja kuvat voi esittää animaationa. XDataSlicella voi käsitellä vastaavasti 3D-dataa.

ApE

ApE (Animation Production Environment) on yleiskäyttöinen visualisointijärjestelmä. Se perustuu tietovuo-malliin. Graafisen käyttöliittymän avulla käyttäjä ohjelmoi visualisointi moduleista verkon, jonka läpi tieto virtaa. ApE muistuttaa paljon AVS:ää (Application Visualisation System).

ApE:n parhaita puolia ovat sen toiminta X-ympäristössä ja saatavuus monille koneille. Vaikka ohjelmisto on modulaarinen, sen laajentaminen lisäämällä uusia moduleja ei ole helppoa. Ongelmina ovat ohjelmiston vaatimat runsaat resussit.

MPGS

Multi Purpose Graphics System (MPGS) on Crayn ja Silicon Graphicsin 4D työaseman välillä toimiva grafiikkapakkaus. Se soveltuu erityisesti elementtimenetelmällä laskettujen tulosten havainnollistamiseen. Suuria tulostiedostoja ei tarvitse siirtää edustakoneelle.

5.3.2 Julkaisukelpoiset kuvat

Julkaisua varten kuvan akseleille merkitään suureen nimet tai tunnukset ja mittayksikön tunnukset. Kuva tulostetaan hyvälaatuisesti. TLP:ssä tämän tason ohjelmia ovat Disspla, UNiRAS ja NCAR.

UNiRAS

UNiRAS on European-Software-Contractors A/S:n kehittelemä rasterigrafiikkaan ja värien käyttöön erikoistunut graafinen paketti. Se mahdollistaa tulostuksen hyvin suurelle joukolle erilaisia laitteita, sen värikoordinaatit ovat laiteriippumattomia, ja se osaa käyttää hyväksi työaseman paikallisen kapasiteetin.

TLP:n ympäristöön on asennettu UNiRAS-paketista seuraavat osat [Vis90]:

1. RASPAK/C - 2- ja 3D-perusgrafiikka,
2. SEISPAK - seismologian sovellukset,
3. GEOPAK - geologian sovellukset,
4. GEOINT - interpolointia GEOPAK-kirjastoa varten,
5. KRIGPAK - interpolointia,
6. BIZPAK - esitysgrafiikka ja
7. GIMAGE - digitoidun kuvan tulostamiseen (satelliittikuvat yms.)

NCAR

NCAR on National Center for Atmospheric Researchissä kehitetty tieteellisiin esityksiin tarkoitettu grafiikkapakkaus. Se on käyttäjän ohjelmasta kutsuttavien Fortran77–aliohjelmien kirjasto, joka on erityisesti meteorologien suosiossa ympäri maailmaa.

NCAR:lla saa tulostettua esimerkiksi pintoja, korkeuskäyriä, ääriviivakuvia, vektorikenttiä, maailmankarttoja kymmenellä eri projektiolla, väri- ja harmaasävykuvia sekä tekstejä eri kirjasimilla.

NCAR koostuu useasta osasesta seuraavasti: Utilities-osa on käyttöliittymä pakkaukseen, ja pitää sisällään Fortran-ohjelmat grafiikan tuottamiseen. Utilities-paketin alapuolella ovat SPPS ja GKS. SPPS on GKS-pohjainen emulaattori vanhoja NCAR PLOT-kutsuja varten. GKS-osa puolestaan ohjaa tulostuksen Computer Graphics Metafile, CGM-tiedostoon. NCAR:sta löytyy pakkaus nimeltään Metafile Translator, joka muuntaa CGM-koodin halutulle tulostuslaitteelle sopivaksi. Värien määrittämiseen se käyttää RGB-arvoja.

Uusi versio 3.0 pitää sisällään muun hyvän muassa X11-liittymän. TLP:ssä NCAR:a on käytetty karttaprojektioiden tulostukseen ja tiedostojen muuntamiseen. [Vis90]

CA-Disspla

Disspla on monipuolinen kaksi- ja kolmiulotteista grafiikkaa tulostamaan kykenevä kirjasto-ohjelma, jossa on noin 400 käyttäjälle näkyvää rutiinia. Sen osia ovat:

-objektikirjastot Fortran77:lle ja Fortran66:lle,
-linkatut, suoritettavat ohjelmat Codebook-optiota varten,
-Disspla Virtual Storage Data file, DVSD, kuvatiedostorakenne ja
-Disspla Workstation Description Information files, WDI, laiteohjaimet.

Disspla koostuu käyttäjän kannalta useista eritasoisista grafiikkamahdollisuuksista. Ensinnäkin perustasosta, Basic System, jossa on perustyökalut kaksi- ja kolmiulotteiselle grafiikalle, sekä DISSPOP-tiedostojen siirtoon. Toisena on esitysgrafiikkataso, joka sisältää varjostettuja kirjasintyyppejä, piirakkakuvia ja muita esitysgrafiikan tarvikkeita. Kolmantena on tuottotaso, johon kuuluvat Codebook ja Dynamics. Neljäntenä on ns. Engineering-taso, joka on tarkoitettu vaativalle käyttäjälle.

Tavallisten Dissplan osien lisäksi TLP:ssa ovat Codebook ja Dynamics. Codebook on vuorovaikutteinen ohjelma, joka luo käyttäjän kuvauksesta Dissplaa hyväksikäyttävän Fortran-ohjelman. Dynamics on Dissplan vuorovaikutteisen grafiikan osa. Dynamics sisältää segmentointiin ja graafiseen syöttöön tarvittavat aliohjelmat.

TLP:n asiakkaat ovat käyttäneet Disspla-ohjelmistoa mm. saasteenleviämismallien esittämiseen, lintujen muuttoreittien piirtoon ja tieteellisten julkaisujen kuvien tekoon. [Vis90] Esimerkki Dissplalla tehdystä kuvasta on liittessä 6.

5.3.3 Animaatio ja realistiset kuvat

Jos visualisoitavana on tuttu esine, vaikkapa auto, on luonnolista pyrkiä luomaan autosta mahdollisimman realistinen kuva. Jos mallin yhtenä ulottuvuutena on aika, on samoin luonnollista esittää data animaatioina. TLP:ssa onkin animaatioita ja realistisia kuvia varten supergrafiikkatyöasema DGL-grafiikkakirjastoineen sekä Wavefront- ja Movie.ByU-ohjelmistot.

Wavefront

Wavefront-ohjelmistolla luodaan realistisia 3D-kuvia.

1. Model-geometriaeditorilla määritellään kuvassa esiintyvien olioiden muodot.
2. Preview-animaatiosekvenssien määrittäjä hallitsee olioiden koon, sijainnin ja liikkeen sekä valojen ja kameroiden paikat.
3. Image-kuvanluontiohjelma laskee kuvan pehmennettynä ja gamma-korjattuna tiedostoon RGB-arvoina halutulla tarkkuudella.

Lisäksi Wavefrontiin kuuluu monenlaisia apuohjelmia materiaalien ja tekstuurien määrittämiseen ja luomiseen, videoiden kontrollointiin sekä esimerkiksi pyörähdyssymmetristen geometrioiden luontiin. Ohjelmien käyttö käy joustavimmin Iris-työaseman konsolilta, mutta jos mallin geometria ja liikemäärittelyt ovat valmiina, voi Wavefrontia käyttää ASCII-päätteeltäkin. Se on parhaimmillaan animaatioiden teossa.

Movie.ByU

Movie.ByU on yleiskäyttöinen grafiikkaohjelmisto. Movie-ohjelmisto koostuu Fortran77-ohjelmista, joiden avulla esitetään ja käsitellään matemaattisia, topologisia ja arkkitehtonisia malleja, joiden geometria voidaan kuvata polygonaalielementeillä tai ääriviivamenetelmillä. Polygonaalielementit saadaan usein elementti- tai differenssianalyysistä.

Movie-ohjelmiston tärkein käyttökohde on elementti- tai differenssimenetelmällä saadun datan jälkikäsittely, visualisointi. Sitä voidaan myös käyttää datan esikäsittelyssä mallien suunnitteluun ja tarkistukseen. Kummassakin tapauksessa käyttäjällä tulee olla ohjelmia, joiden avulla data voidaan muuntaa Movie-formaatista analyysiohjelmiston vaatimaan muotoon tai päinvastoin. Movie-ohjelmistolla voidaan tehdä animaatioita ja elokuvia.

TLP:ssä Movie.ByU:ta on käytetty virtausmekaniikan tulosten esittämiseen ja rakenneanalyysivideoiden tekoon. [Vis90]

Funcs

Funcs on Juha Ruokolaisen TLP:ssa tekemä grafiikkaohjelmisto Iris-työasemille. Funcsilla voi kuvata 2D- ja 3D-hiladataa vektori- ja skalaarimuodossa. Animaatiota voi editoita kuva kerrallaan tai interpoloida muutamasta avainkuvasta. Ohjelmaa voi ohjata komentoja antamalla tai hiirellä valikoista. [Lin90]

5.3.4 Visualisoinnin kehitys

Visualisoinnin merkitys kasvaa. Lisääntyvän laskentakapasiteetin ansiosta suurten mallien laskeminen nopeutuu ja tulosten analysointi helpottuu. Tällöin kiinnostus visualisointia kohtaan kasvaa edelleen ja sovellusalueet lisääntyvät, mikä tuo mukanaan yhä uusia käyttäjiä. Näin kapasiteetin tarve kasvanee nopeammin kuin resurssit.

Lähes kaikki tutkijat saavat tai hankkivat työpöydälleen verkotetun oman tietokoneen, jolla on mahdollisuus esittää grafiikkaa. Valmisohjelmien käyttö lisääntyy. Graafisten käyttöliittymien yleistyminen helpottaa ohjelmien käyttöä. Visualisoinnista tulee vuorovaikutteista. Omalta tietokoneelta voi seurata visuaalisesti laskennan etenemistä

supertietokoneessa. TLP:n tärkein tehtävä on neuvoa ja tiedottaa visualisoinnin mahdollisuuksista. Lisäksi TLP tarjoaa erityispalveluja, kuten videoiden tuottaminen, jotka vaativat suhteellisen kallista laitteistoa.

6. Xftp

Tavoittena oli tutkia olemassa olevien ohjelmien käytön helpottamista lisäämällä niihin graafinen käyttöliittymä. Esimerkkinä toteutettiin xftp, jossa korvataan ftp-ohjelman komentokieleen perustuva käyttöliittymä graafisella käyttöliittymällä.

6.1 Ftp

FTP (File Transfer Protocol) [PR85] on tiedostonsiirtokäytäntö TCP/IP-ympäristössä. Yleensä puhutaan tiedostojen siirrosta vaikka tiedostot kopiodaan. Tästä tekstissä ftp (pienellä kirjoitettuna) tarkoittaa FTP:n toteutusta Unix-ympäristössä, ftp-ohjelmaa. Esimerkki ftp:n käytöstä on liitteessä 4.

6.2 Määrittely

Xftp:n toiminnoiksi määriteltiin ftp:n tärkeimmät toiminnot. Käyttöliittymä määriteltiin Macintoshin hyperftp:n tasoiseksi.

6.3 Suunnittelu

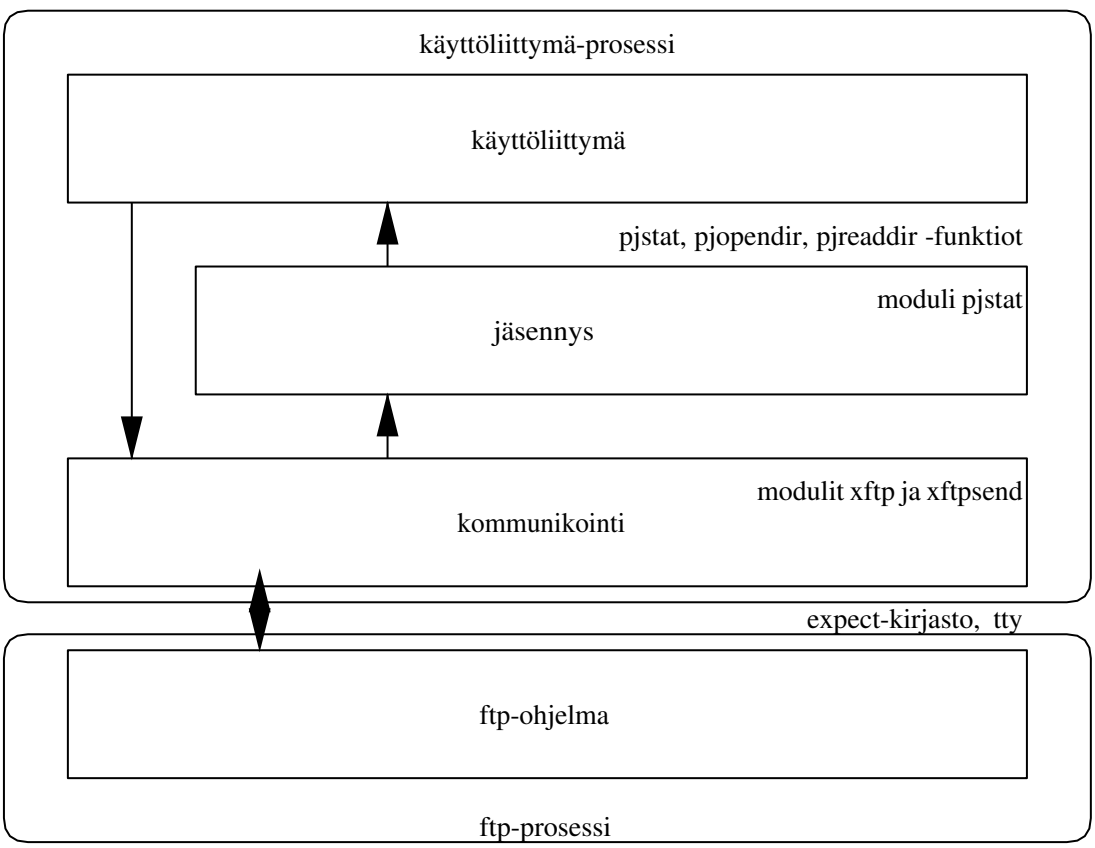
6.3.1 Xdtm

Xftp:n käyttöliittymä pohjautuu Xdtm-ohjelmaan [Gro91]. Xdtm (X Desktop Manager) on julkisohjelma. Se vastaa lähinnä OpenWindowsin ja muiden ikkunointijärjestelmien *file manager*-ohjelmia mahdollistaen tiedostojen kopioinnin, siirron, tuhoamisen ja esittämisen ikoneina. Tässä työssä sen toimintaa laajennettiin kattamaan tiedostojen siirto tietokoneiden välillä. Koska Xdtm:ssä käyttöliittymä oli lähes valmiina, ei käytetty mitään UIDT-ohjelmia.

Xdtm:n käyttöliittymä perustuu Athena-widgeteihin ja kahteen niistä luotuun omaan widgetiin: XedwList ja XedwForm. Vastaavan Athena-widgetin ominaisuuksien lisäksi XedwList-widgetissa voi valita useita listan jäseniä ja esittää listassa tiedoston nimen ja vastaavan ikonin. Widgetiä käytetään hakemistolistausten esittämiseen xftp:ssä. XedwForm-widgetillä järjestellään muita widgetejä, eikä se näy käyttäjälle.

6.3.2 Prosessien välinen kommunikointi

Ftp-ohjelmasta oli käytettävissä lähdekoodi. Sitä ei kuitenkaan käytetty hyväksi ohjelmankehityksessä, koska sen siirrettävyys ei ole täydellinen. Ftp-ohjelman muuntamisesta paremmin siirrettäväksi tai vastaavien toimintojen toteuttamisesta omassa ohjelmassa luovuttiin, koska ftp-ohjelman ajaminen käyttöliittymäohjelman alaisuudessa vaikutti helpommalta. Ohjelma koostuukin kahdesta prosessista: käyttöliittymä- ja ftp-prosessit. Prosessit kommunikoivat keskenään expect-kirjaston [Lib90] mukaisesti tty:n kautta, koska lähteen [Roc85] esimerkin mukainen kommunikointi kaksisuuntaisten putkien avulla ei onnistunut. Ftp-prosessi kuvittelee kirjoittavansa normaalisti päätteelle, joka käyttöliittymäprosessille näkyy tavallisena tiedostona. Vaikka käytetään kahta prosessia, ei mitään tapahdu rinnakkaisesti, vaan käyttöliittymäprosessi odottaa ftp:n komentojen suorituksen ajan.

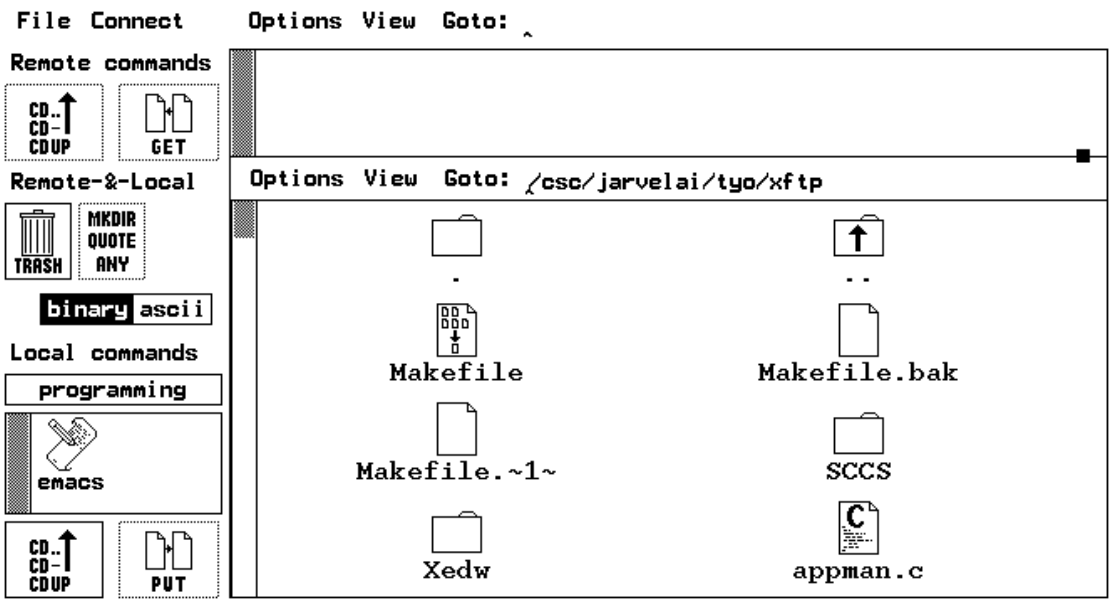


Kuva 10. xftp-ohjelman rakenne

Käyttöliittymä-prosessi jakaantuu kuvan 10 mukaisesti kommunikointi-, jäsennys- ja käyttöliittymäosuuksiin. Kommunikointiosuus lukee ftp:n tulosteet ja kirjoittaa ftp:lle komennot. Jäsennys sopisi hyvin toteuttavaksi Unix:n lex- ja yacc-ohjelmilla. Koska kokeilua varten kehitetty yksinkertainen jäsennin toimii tyydyttävästi, näin ei kuitenkaan tehty. Jäsennin- ja käyttöliittymäosuudet liittyvät toisiinsa Unix:n stat-, opendir- ja readdir-systeemikutsuja vastaavilla funktioilla. Näin käyttöliittymän hakemiston visualisoiva osuus on helposti vaihdettavissa xftp:n käyttämästä xdtm:n XedwList-widgetistä Motifin FileSelectionBox-widgetiin tms.

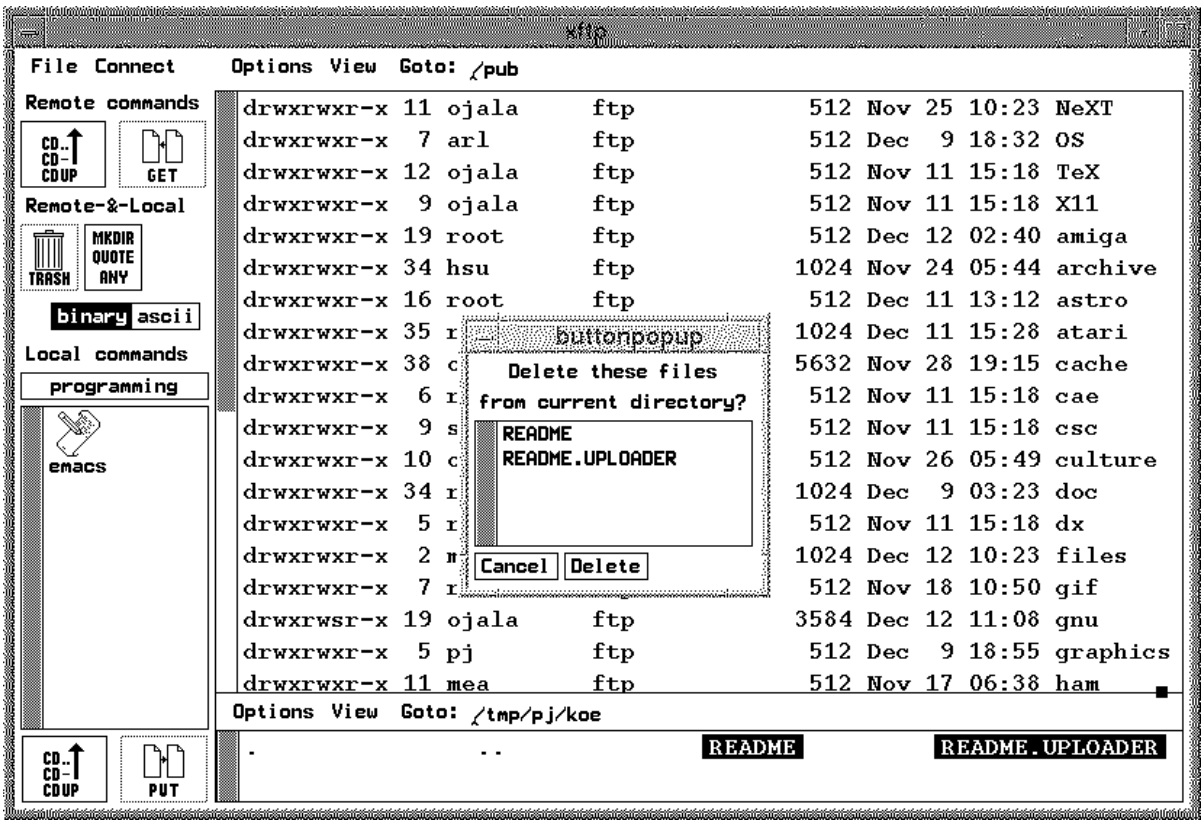
6.4 Käyttöliittymä

Koska toteutuksen pohjana käytettiin xdtm:ää, käyttöliittymässä on enemmän ominaisuuksia kuin hyperftp:ssä. Ohjelma olisi mahdollista laajentaa käsittäämään Macintoshin työpöytä-mallin mukainen käyttöliittymä, mutta tällä hetkellä ohjelma käyttää Motif-tyylin mukaista kohde-toiminto-mallia; ensiksi valitaan halutut tiedostot ja sitten niille suoritettava toimenpide. Käyttöliittymä on toteutettu Athena- ja Xedw-widgeteillä, joita hallitaan Xtoolkitin avulla.



Kuva 11. xftp

Kuvassa 11 näkyy xftp ajon alussa. Yhteys toiseen koneeseen aloitetaan yläreunan Connect-valikosta. Valikon toiminnot ovat lista käyttäjän .netrc tiedostossa olevista koneista ja Open...-toiminto, joka kyselee kohoikkunoissa tarpeelliset tiedot yhteyden ottamiseksi. Kuvassa 11 oikealla ovat päällekkäin hakemistoikkunat. Alemmassa ikkunassa näkyy ikoneina sen hakemiston sisältö, missä käyttäjä käynnisti ohjelman.



Kuva 12. xftp ja kohoikkuna

Hakemistoikkunoiden yläpuolella olevista valikoista voidaan valita, miten ja mitä tietoa hakemistoista esitetään ja siirtyä hakemistosta toiseen. Hakemistoikkunoiden kokojen suhdetta voidaan säätää ikkunoiden välissä olevan valikkopalkin oikeassa yläreunassa sijaitsevaa neliötä siirtämällä. Kuvassa 12 on avattu yhteys vieraaseen koneeseen, jonka hakemisto esitetään ylemmässä ikkunassa. Kuvat 11 ja 12 ovat eri mittakaavassa ja jälkimmäisessä on mwm-ikkunanhallintaohjelman kehyksiä ikkunoiden ympärillä.

Kuvien 11 ja 12 vasemmassa reunassa on toimintonappeja, jotka on ryhmitelty kolmeen ryhmään.

1. Ylimpänä ovat toiseen koneeseen liittyvät toiminnot.
2. Välissä ovat roskakori-, anybutton-toimintanapit ja ascii/binary-valintanappi. Roskakoriin voi heittää sekä toisessa että paikallisessa koneessa olevia tiedostoja. Roskakorin oikealla puolella on anybutton-toimintanappi. Anybutton avaa putkahduskyselyn, jossa kirjoitettu komento lähetetään ftp:lle. Näin nekin ftp:n komennot, joita xftp:ssä ei ole toteutettu graafisesti, ovat käytettävissä.
3. Alimpana olevissa paikallisissa komennoissa on toimintanappien lisäksi Xdtm:stä peräisin oleva mahdollisuus käynnistää ohjelma antaen sille tiedosto parametriksi.

Koska kuvassa 11 yhteyttä toiseen koneeseen ei ole vielä avattu, osa toimintanapeista ei ole käytettävissä, mikä näkyy niiden katkonaisesta ääriviivasta.

6.5 Toteutus

Xftp kehitettiin Convex 220 tietokoneessa. Koneen käyttöjärjestelmä ConvexOS9.1 on BSD4.2 pohjainen Unix. Työssä käytettiin X11R4 ikkunointijärjestelmää. Koska Xdtm oli toteutettu C-ohjelmointikielellä, myös xftp toteutettiin C-kielellä. Se onkin luonteva kieli Unix- ja X-ympäristössä.

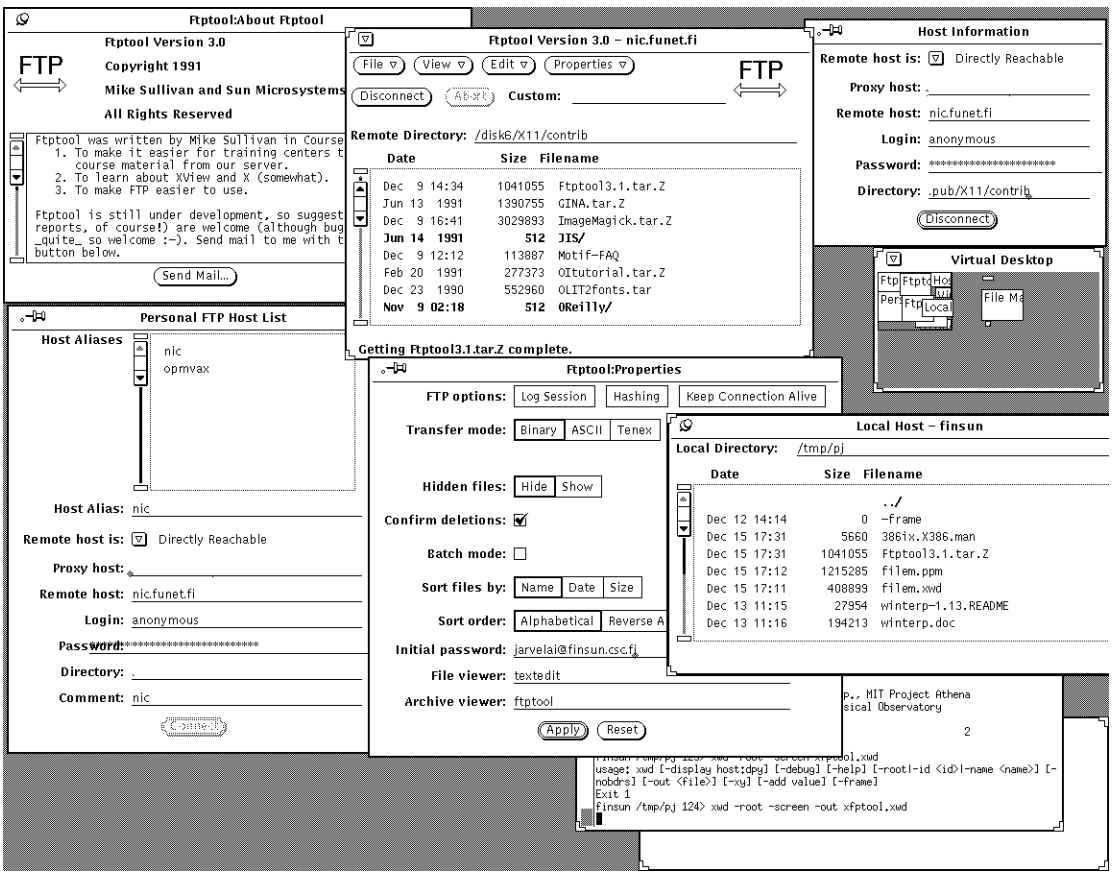
Ongelmia aihettivat C-kielen kaksi eri versiota: uusi ANSI-C [ISO90, KR88] ja vanha C [KR78]. Convexin C-kääntäjä on ANSI-C:n mukainen, mutta siinä on yhteensopivuus valitsin -pcc vanhoja C-ohjelmia varten. Convexin ReleaseNotesin mukaan on tätä lippua käytettävä X-ohjelmia käännettäessä. Koska xdtm on kirjoitettu ANSI-C:llä, olisi ohjelma ollut muunnettava K&R:n mukaiseksi C:ksi sen siirtämiseksi Convexille. Näin ei kuitenkaan tarvinnut tehdä, koska ohjelmat saattoi kääntää ilman -pcc lippua, jota tarvittiin vasta linkkausvaiheessa. Tämä estää kuitenkin ANSI-C:n mukaisten kirjastojen käytön. Onneksi xdtm:n mukana tuli mm. oma strstr-funktio, kun valmiit ANSI-C kirjastot eivät olleet käytettävissä.

Xftp:n graafinen käyttöliittymä toteuttaa kaikki ftp:n tärkeimmät käskyt: *open, user, cd, lcd, dir, binary, ascii, get |more, get, put, mget, mput, delete, pwd* ja *bye*. Näiden käskyjen toiminnoista puuttuu vain mahdollisuus tiedostojen uudelleennimeämiseen siirron aikana.

6.5.1 Ftptool

xftp:n kehitys keskeytettiin viimeistelyvaiheessa. Keskeyttäminen johtui julkaistusta paremmasta graafisesta käyttöliittymästä ftp-ohjelmalle, Ftptoolista [Sul91]. Ftptoolissa on toteutttu suurempi osa ftp:n komennoista kuin xftp:ssä. Lisäksi Ftptoolissa joitakin ylimääräisiä ominaisuuksia kuten hakemistojen pitäminen muistissa, mikä vähentää verkkoliikennettä. Ohjelmassa on mahdollisuus määritellä henkilökohtai-

nen lista isäntäkoneista. Tämä lista talletetaan selväkielisenä salasanoineen käyttäjän kotihakemistoon .ftptoolrc-tiedostoon, mikä ei ole turvallista. Kuvassa 13 on keskellä



Kuva 13. Ftptool

ylhäällä Ftptoolin perusikkuna ja sen ympärillä muita Ftptoolin ikkunoita. Virtual-Desktop ja oikeassa alareunassa olevat ikkunat, joissa ollaan tallentamassa kuvaruutua tiedostoon, eivät kuulu Ftptooliin. Käyttöliittymä on Open Lookin mukainen.

Toteutukseltaan Ftptool on hyvin samanlainen kuin xftp. Myös Ftptool käynnistää ftp-ohjelman ja keskustelee sen kanssa. Tärkein ero on ulkoasussa; Ftptoolin käyttöliittymä perustuu XView-kirjastoon. XView ei ole niin levinnyt kuin Athena-widgetit ja XView:n siirtäminen uudelle koneelle saattaa vaatia jonkin verran työtä, mikä hieman rajoittaa ohjelman siirrettävyyttä.

6.6 Kokemuksia

Aikaisemman kokemuksen ansiosta graafisen käyttöliittynän suunnittelu ja toteuttaminen oli kohtuullisen helppoa. Kelvollinen käyttöliittymä syntyi parin kuukauden työllä neljän kuukauden aikana. Täydellisemmän Ftptoolin tekemiseen on kulunut kolme kuukautta. Xftp:n koko on noin 10 000 riviä kommentoitua C-ohjelmaa. Xdtm koko on 5 000 riviä koodia. Suurin osa xftp:n käyttöliittymästä perustuu muunneltuun Xdtm:n koodiin. Jäsennyksen ja kommunikoinnin osuus on vain noin tuhat riviä. Silti siihen kului puolet ohjelmakehitykseen käytetystä ajasta, sillä käyttöliittymän kehitys oli selkeää Xdtm:n koodin ja tietorakenteiden monistamista.

Vaikka Xdtm on toteutettu erittäin siististi ANSI-C:llä, ei koodin uudelleenkäytettävyys ollut kovin hyvä. Ainoastaan olio-ohjelmointiin perustuviin Xedw-widgeteihin ei tarvinnut tehdä muutoksia.

Tehokkuus riippuu mitattavasta suureesta. Keskusyksikköaikaa xftp kuluttaa vähemmän kuin ftp:n tulostuksiin kuluu aikaa xterm-ohjelmalta. Kuvaruudun vieritys xterm-ohjelmassa on ilmeisesti raskaampi operaatio kuin vastaavan tiedostolistauksen jäsennys xftp:ssä. Muistin kulutus on X-ohjelmaksikin suuri: kaksi megatavua. Ftp:llä saatujen hakemistolistausten tallentaminen muistiin lisäisi muistinkulutusta, mutta mahdollistaisi vähäisemmän verkkoliikenteen, kun hakemisto, jossa on jo käyty, löytyisi ohjelman tietorakenteista.

Ftp-ohjelman käyttö osana xftp:tä oli virhe, sillä eri toteutukset ftp-ohjelmasta poikkeavat toisistaan. Esimerkiksi Convexin ftp:ssa pelkät tiedoston nimet tulostaa nlist-komento, finsunin ftp:ssä ls-komento. Tämän vuoksi olisi ollut järkevämpää luoda kokonaan uusi ftp-ohjelma standardin protokollan päälle kuin käyttää pohjana vanhaa ohjelmaa, jonka päälle käyttöliittymä rakennettiin. Jokainen Unix on hieman erilainen, toisissa ftp:llä saatu tiedostolista on aakkosjärjestyksessä, toisissa epäjärjestyksessä, toisissa . ja .. tiedostot ovat mukana, mutta eivät kaikissa, jne. Epäyhtenäisyys mutkisti ohjelman toteutusta.

Vaikka xftp lähes täyttää täyttää toiminnallisen määrittelyn ja käyttöliittymältään jopa reilusti ylittää vaatimukset, koekäyttäjät eivät ole siitä innostuneita. Käyttöliittymä muistuttaa niin paljon Macintosh-ympäristöä, että käyttöliittymän pitäisi toteuttaa se täydellisesti. Nyt pienet erot häiritsevät.

7. Esimerkki sovellusohjelmien käytön helpottamisesta X-ympäristössä

7.1 Ongelma

Abaqus on yleiskäyttöinen rakenneanalyysiohjelma, joka käyttää elementtimenetelmää eli FEM:iä. Ohjelmalla voidaan määrittää jännitys- ja muodonmuutostiloja, simuloida materiaalien käyttäytymistä ja ratkaista lämmönsiirtymistehtäviä. Ohjelmassa on monipuolinen materiaalikirjasto, johon kuuluu esimerkiksi raudoitetun betonin ja huokoisen aineen materiaalimallit. Lisäksi käyttäjä voi itse ohjelmoida omat materiaalimallinsa. Ohjelma on tehokas ja soveltuu erityisen hyvin epälineaarisiin tehtäviin. TLP:ssa Abaqusta on käytetty paperin kuituverkon analysointiin, maa-aineksen käyttäytymisen simulointiin, puun kuivumisen mallintamiseen ja autonkorin dynaamisten värähtelyiden tutkimiseen.

Patran on yleiskäyttöinen 3D Mechanical Computer-Aided Engineering (MCAE) ohjelmisto. Sitä käytetään FEM-laskennan esi- ja jälkikäsittelyyn eli mallinnukseen ja tulosten esittämiseen graafisesti, esimerkiksi yhdessä Abaquksen kanssa.

Abaqus on käytettävissä Crayllä ja Patran kaikilla muilla TLP:n koneilla paitsi Crayllä. Tällä järjestelyllä voidaan suurta graafista tehoa vaativat esi- ja jälkikäsittely ajaa tarkoituksenmukaisimmilla laitteilla eli työasemilla ja tyypillisesti eräajopohjaiset analyysit laskentapalvelimena toimivassa Crayssä. Ongelma on tämän hajautetun järjestelmän hallinta. Käyttäjän on muistettava monia komentoja tiedostojen siirtoja ja muuntamisia varten. Toiminnot ovat usein rutiininomaisia. Esimerkkikomentotiedosto Patranin ja Abaquksen käytöstä on liitteenä 7.

7.2 Ratkaisu

Tärkein arviointiperuste ratkaisulle on yleiskäyttöisyys. Ratkaisun on toimittava kaikkien TLP:n sovellusohjelmien kohdalla, eikä vain Patranin ja Abaquksen. Koska laitehankinnat vaikuttavat TLP-ympäristöön, ratkaisun piti olla joustava. Tarkoitus oli erityisesti tutkia, miten X-ikkunointijärjestelmässä toimivat graafiset käyttöliittymät voisivat helpottaa Patranin ja Abaquksen käyttöä. Siksi tekstipohjaisessa ympäristössä käytetyt komentotiedostot ja komentojen uudelleen nimeäminen eivät riittäneet ratkaisuksi.

Ennen työn aloittamista oletettiin ratkaisun vaativan erillisen ohjelman tekemistä. Kun ongelma oli tarkasti määritelty, havaittiin myös, että taustavalikot voisivat ratkaista ongelman. Erillisen ohjelman kirjoittamisesta ja taustavalikoihin perustuvasta ratkaisusta valittiin toteutettavaksi taustavalikoihin perustuva ratkaisu, koska se oli huomattavasti helpompi toteuttaa ja tarjosi riittävän toiminnallisuuden.

7.2.1 Taustavalikko

Ohjelmat alustetaan ja käynnistetään X:n taustavalikosta. Taustavalikko on ikkunanhallintaohjelman ominaisuus ja se määritellään ikkunanhallintaohjelman alustustiedostossa. Oletusarvoisesti mwm olettaa alustustiedoston olevan käyttäjän kotihakemistossa nimellä .mwmrc, twm nimellä .twmrc ja olwm nimellä .openwin-menu. Twm:n ja mwm:n funktioissa on eroja, mutta kuvan 14 listauksen ”patranmenu”:n

luonti toimii kummassakin. Olwm:lle syntaksia pitää hieman muuttaa. Kuvan 14

```
menu "patranmenu"
{
  "finsun alku"     !"xrsh finsun xterm -ls -e
       /appl/public/X11/taustamenu/esim/patran1.csh &"
  "cray era abaqus" !"xrsh finsun xterm -ls -e
       /appl/public/X11/taustamenu/esim/patran2.csh &"
  "cray abapat"     !"xterm -T Cray -name Cray -sb -sl 512
 -geom 80x24 -fn 8x13 -fb 8x135bold -e rlogin cray.csc.fi &"
  "finsun loppu"    !"xrsh finsun xterm -ls -e
       /appl/public/X11/taustamenu/esim/patran3.csh &"
}

menu "Normal-left"
{
  "Commands"          f.title
  "motif"             f.menu "me"
  "patran"            f.menu "patranmenu"
  "X Figure Editor"   !"xfig -display ${DISPLAY} &"
  "Manual"            !"xman &"
  "Xrn "              !"xrn &"
  "Dump Windows"      !"xwd|xpr -device ps|lpr -Ppost &"
  "Calculator"        f.menu "laskimet"
  "Monitor"           f.menu  "Monitor"
  "Games"             f.menu "games"
  "more..."           f.menu "xsoftaa"
}
```

Kuva 14. Patran-valikon luonti ikkunanhallintaohjelman alustustiedostossa

listauksessa ”Normal-left”-valikko putkahtaa esille käyttäjän painaessa hiiren vasemman puoleista nappia. Valikko ”patranmenu” on valikon ”Normal-left” alivalikko (*cascading, submenu*). Valikoiden ensimmäisenä rivillä lainausmerkeissä oleva teksti näkyy käyttäjälle valikossa. Kun käyttäjä on valinnut, suoritetaan huutomerkin jälkeinen komento. Seuraavan tason valikon nimen määrittää f.menu. Valikot ovat siis monitasoisia.

Shell on Unix:n komentotulkki. Xrsh-komento on Unix:in rsh eli *remote shell*-komento, joka lisäksi välittää ympäristömuuttujan DISPLAY. Tämä komento suoritetaan koneella, jossa ikkunanhallintaohjelmaa ajetaan. Tältä koneelta on oltava oikeus ajaa rsh finsunilla.

Valittaessa ”finsun alku”, ”cray era abaqus” tai ”finsun loppu” käynnistetään siis finsunilla xterm-ohjelma. Xterm on VT102- ja Tektronix 4014-pääte-emulaatoriohjelma. Sen valitsin ls käynnistää pääte-emulaattori-ikkunan shellin login-shellinä eli käyttäjän .login tiedosto luetaan. Valitsin –e ajaa ohjelman, tässä tapauksessa komentotiedoston patran*.csh.

```
#!/bin/csh
#patran1.csh
alias setup 'source /appl/setup/\!*'
setup patran
xrdb -merge /appl/public/X11/taustamenu/xrdbpatran
cd ${PATRANDIR}
cp /appl/public/X11/taustamenu/esim/palkki.* ${PATRANDIR}
cp /appl/public/X11/taustamenu/esim/ajo*.ses ${PATRANDIR}
patran -d bat -s ajo1.ses
patran -d xwn -s ajo2.ses
```

Kuva 15. patran1.csh komentotiedosto

Komentotiedostossa patran1.csh kuvassa 15 lisätään hakupolkuun Patranin sijainti setup-komennolla. Näppäimistö ohjelmoidaan ja esimerkkitiedostot kopioidaan patran-hakemistoon. Patranilla luodaan analysoitava malli. Myös muut patran*.csh tiedostot vastaavat liitteen 7 esimerkkikomentotiedostoa.

7.2.2 Näppäimistön ohjelmointi

```
!xrdbpatran
*Cray*VT100.translations: #override
<Key>F8:string("rcp s41m1.dis.* finsun:${PATRANDIR}")
        string(0x0d) \n\
<Key>F9:string("exit") string(0x0d) \n\
<Key>F6:string("cd ${PATRANDIR}") string(0x0d) \n\
<Key>F7:string("ls -la") string(0x0d)
```

Kuva 16. xrdbpatran resurssitiedosto
funktionäppäinten ohjelmoimiseksi X-ympäristössä

Xrdb-ohjelmaa käytetään resurssihallitsijan tietokannan asettamiseen ja lukemiseen. Cray-nimisen ikkunan VT100–näppäimistöltä ohjelmoidaan uudestaan neljä funktionäppäintä kuvan 16 xrdbpatran resurssitiedoston listauksessa. F6–näppäimen painalluksella pääsee patran-hakemistoon, F7:llä näkee hakemiston sisällön, F8 kopioi abapat muunnosohjelman tuloksena syntyneen tiedoston finsuniin ja F9 lopettaa pääteistunnon Crayhin, jolloin myös Cray-ikkuna häviää. Heksadesimaaliluku **d** on ASCII rivinvaihto eli funktionäppäimen lisäksi ei tarvitse painaa rivinvaihtoa.

7.3 Kokemuksia

Näppäimistöä voidaan säädellä ohjelma- ja ikkunakohtaisesti. Ikkunan nimi esimerkiksi xterm-ohjelmassa on käyttäjän määrättävissä. Jokaisen sovelluksen käyttöä varten on mahdollista luoda oma ikkuna, joten näppäimistö voidaan virittää sovelluskohtaisesti. Määrätty näppäin voitaisiin ohjelmoida lähettämään kullekin sovellukselle tarvittava komento *exit, bye, quit* tai *stop* sovelluksesta poistumiseksi.

Tässä on myös tarvetta jatkokehittelyyn. Ensimmäiseksi tarvittaisiin ohjelma, joka näyttäisi, mitä funktionäppäimiin on ohjelmoitu. Koska eri päätteissä on erilainen näppäimistö, on vaikea tietää, mitä funktionäppäimiä on käytettävissä. Tämän vuoksi tarvitaan ohjelma, jolla voi helposti ohjelmoida näppäimiä.

8. Johtopäätökset

Tietokonegrafiikalla pystytään havainnollistamaan suuria tietomääriä, joita esiintyy erityisesti supertietokoneympäristössä. Tietokoneverkot mahdollistavat supertietokoneen etäiskäytön henkilökohtaiselta tietokoneelta. Hajautettuihin ikkunointijärjestelmiin X ja NeWS perustuvat graafiset käyttöliittymät toimivat myös verkon yli, jolloin sovellusohjelmien käyttö ja visualisointi supertietokoneympäristössä on yhtä helppoa kuin mikrotietokoneessa.

TLP:ssa on pyritty noudattamaan standardeja, niin de facto kuin standardoimisorganisaatioiden virallistamiakin. Koska TLP:n tarkoituksena on nimensä mukaisesti palvelu, asiakkaita täytyy tukea vaikka he standardeista poikkeavatkin. Tämä korostaa tiedottamisen ja valistuksen merkitystä. Hyvät työkalut ovat jo olemassa, käyttäjille täytyisi vain kertoa niistä. Ohjelmien välinen yhteistoiminta on ongelma, jopa ”standardin mukaisten” ohjelmien välillä.

Esimerkki sovellusohjelmien käynnistämisestä X-ympäristössä osoittaa, että käyttöympäristöä voidaan huomattavasti kehittää käyttämällä X:n taustavalikoita ja näppäimistön hallintaa. TLP:ssä ei juurikaan pystytä vaikuttumaan käytettävien ohjelmien käyttöliittymiin, mutta ohjelmien käyttöympäristöön vaikuttamisen esimerkki osoittaa mahdolliseksi. TLP:n tavoitteena tulisikin olla tarjota hyvä käyttö- ja ohjelmankehitysympäristö.

Vaikka xftp lähes täyttää toiminnallisen määrittelyn ja käyttöliittymältään jopa reilusti ylittää vaatimukset, koekäyttäjät eivät ole siitä innostuneita. Käyttöliittymä muistuttaa niin paljon Macintosh-ympäristöä, että käyttöliittymän pitäisi toteuttaa se täydellisesti.

Unix-toteutusten erot vaikeuttivat xftp:n kehitystä. Ohjelmistokehityksessä onkin pyrittävä löytämään standardirajapinnat ja hyödyntämään niitä. Ympäristön yhdenmukaisuutta voidaan toivottavasti parantaa seuraamalla Unix:n standardoitumiskehitystä. X-ikkunointijärjestelmä muodostaa yhden hyvin määritellyn ja toimivan standardirajapinnan, jota kannattaa hyödyntää.

Lähdeluettelo

[Ado85a] Adobe Systems. PostScript language reference manual. Addison-Wesley, Reading, MA, 1985. ISBN 0-201-10169-6.

[Ado85b] Adobe Systems. PostScript language tutorial and cookbook. Addison-Wesley, Reading, MA, 1985. ISBN 0-201-10179-3.

[Dur87] Durrett, H. John, editor. Color and the Computer. 1250 Sixth Avenue, San Diego, California 92101, Academic Press, 1987. ISBN 0-12-225210-1.

[Fek91] Fekete, Jean-Daniel. WWL, a Widget Wrapper Library for C++. Laboratoire de Recherche en Informatique, Bât. 490, Faculté d'Orsay, 81405 Orsay Cedex, France, jdf@lri.lri.fr, 1991. ohjelmisto, nic.funet.fi-[128.214.6.100]:pub/X11/contrib/WWL-1.2.tar.Z-tiedosto.

[Fol90] Foley, James D., et al. Computer Graphics: Principles and Practice, second edition. Addison-Wesley, 1990. ISBN 0-201-12110-7.

[For90] Forsström, Pirjo-Leena. Ero kuva-analyysin ja -synteesin välillä on aikansa elänyt. *Ajantasa*, (3):24–25, 1990.

[Gos89] Gosling, James. The NeWS Book, An Introduction to the Network/extensible Window System. Sun technical reference library. Sringer-Verlag, 1989. ISBN 0-387-96915-2, 3-540-96915-2.

[Gra91] Graafisen liittymän voi siirtää. *Tietoviikko*, (41):8, 5.12.1991.

[Gro91] Groenendaal, Edward. xdtm. 1991. ohjelmisto, ugle.unit.no[129.241.1.97]:-pub/X11R4/contrib/xdtm-1.5.tar.Z-tiedosto.

[Hal89] Hall, Roy. Illumination and color in computer generated imagery. Springer-Verlag, New York, 1989. ISBN 0-387-96774-5, 3-540-96774-5.

[Hei88] Heinänen, Juha. Hajautetut ikkunointijärjestelmät. *Korkeakoulujen atk-uutiset*, (3-4):26–28, 1988.

[Iko90] Ikonen, Ari. Hajautettujen ikkunointijärjestelmien kuormittavuustutkimus. Diplomityö. TTKK, Sähkötekniikan osasto, 1990.

[Imc91] Image Tools. San Diego Supercomputer Center, P.O. Box 85607, San Diego, CA 92186-9784, Attn: Julie Shisler, info@sdsc.edu, 1991. ohjelmisto, nic.funet.fi[128.214.6.100]:pub/sci/graphics/programs/imtools-hakemisto.

[ISO90] ISO/IEC 9945 IEEE Std. 1003.1. Portable Operating System Interfase (POSIX) - Part1: System Application Program Interface (API) [C Language]. The Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, NY 10017, USA, 1990. ISBN 1-55937-061-0.

[Jär90] Järvinen, Hannu-Matti. Tekninen kommentti. *uutisryhmä tut.ttky*, 19 Nov 1990.

[Jon89] Jones, Oliver. Introduction to the X Window system. Prentice Hall, Englewood Cliffs, New Jersey 07632, 1989. ISBN 0-13-499997-5.

[Kar90] Kari, Matti. Graafinen käyttöliitymä motif-ympäristössä. Diplomityö. TTKK, Sähkötekniikan osasto, 1990.

[KB91] Kuutti, Kari ja Bannon, Liam J. Some confusions at the interface. In M.I.Nurninen, P.Järvinen, ja Weir, G., editors, *Precedings of Human Jobs and Computer Interface Conference*, pages 94–112, University of Tampere, Finland, 1991.

[Kos90] Koski, Kimmo. Suomalaisen superkoneympäristön kehitys. Lisensiaattityö. Tkk, Sähköteknillinen osasto, 1990.

[KR78] Kernighan, B.W. ja Ritchie, D.M. The C programming language. Prentice-Hall, Englewood Cliffs, N.J. 07632, 1978.

[KR88] Kernighan, B.W. ja Ritchie, D.M. The C programming language, second edition, ANSI C. Prentice-Hall, Englewood Cliffs, N.J. 07632, 1988. ISBN 0-13-110362-8.

[Lib90] Libes, Don. expect: Curing those uncontrollable fits of interaction. In *Proceedings of the Summer 1990 USENIX Conference*, Anaheim, California, June, 1990.

[Lin90] Lindberg, Klaus. Grafiikkaa ja animaatioita funcsilla. *Supermenu*, (20):50–55, 1990.

[Lum90a] Lumiala, Veli-Matti. Tieteellisen laskennan hajautetun tietojenkäsittelyympäristön hallinta. Diplomityö. Tkk, Sähköteknillinen osasto, 1990.

[Lum90b] Lumiala, Veli-Matti. X-pääte työvälineenä. *Korkeakoulujen atk-uutiset*, (3):27, 1990.

[May91] Mayer, Niels P. Winterp. Hewlett-Packard Laboratories, Human-Computer Interaction Department, P.O. Box 10490, Palo Alto, CA, mayer@hplabs.hp.com, 1991. ohjelmisto, nic.funet.fi[128.214.6.100]:-pub/X11/R5/contrib.untarred/contrib/Motif/clients/winterp-hakemisto.

[MIT91a] MIT X Consortium. X11R5. 1991. ohjelmisto, nic.funet.fi[128.214.6.100]:-pub/X11/R5-hakemisto.

[MIT91b] MIT X Consortium. XStandards. 1991. nroff-dokumentti, export.lcs.mit.edu-[18.24.0.12]:pub/R5untarred/mit/man/Standards.man-tiedosto.

[Ope90a] Open Software Foundation. Application Environment Specification (AES) User Environment Volume, revision a. Prentice Hall, Englewood Cliffs, New Jersey 07632, 1990. ISBN 0-13-640483-9.

[Ope90b] Open Software Foundation. OSF/Motif Programmere's Guide, revision 1.0. Prentice Hall, Englewood Cliffs, New Jersey 07632, 1990. ISBN 0-13-640525-8.

[Ope90c] Open Software Foundation. OSF/Motif Style Guide, revision 1.0. Prentice Hall, Englewood Cliffs, New Jersey 07632, 1990. ISBN 0-13-640491-X.

[Ope90d] Open Software Foundation. OSF/Motif Style Guide, revision 1.1. 1990.

[Ope90e] Open Software Foundation. OSF/Motif User's Guide, revision 1.0. Prentice Hall, Englewood Cliffs, New Jersey 07632, 1990. ISBN 0-13-640509-6.

[Pos91] Poskanzer, Jef. pbmplus. 1991. ohjelmisto, nic.funet.fi[128.214.6.100]:-pub/X11/contrib/pbmplus30oct91.tar.Z-tiedosto.

[PR85] Postel, J.B. ja Reynolds, J.K. File Transfer Protocol. 1985. ASCII-dokumentti, nic.funet.fi[128.214.6.100]:pub/netinfo/rfc/900-999/rfc0959.txt-tiedosto.

[RH91] Roponen, Seppo ja Harmo, Timo. Sinä ja tietokone. Gaudeamus, Helsinki, 1991. ISBN 951-662-522-3.

[Roc85] Rochkind, Marc J. Advanced UNIX programming. Prentice-Hall, Englewood Cliffs, N.J. 07632, 1985. ISBN 0-13-011800-1.

[Sav91] Savioja, Lauri. Viestien selaustyöaseman toteutus Motif-ympäristössä. Diplomityö. Tkk, Tietotekniikan osasto, 1991.

[Sin90] Sinkkilä, Timo. Supertietokoneen Cray X-MP liittäminen hajautettuun tietojenkäsittely-ympäristöön. Diplomityö. Tkk, Sähköteknillinen osasto, 1990.

[Sul91] Sullivan, Mike. ftptool. Mike.Sullivan@Sun.COM, 1991. ohjelmisto, nic.funet.fi[128.214.6.100]:pub/X11/contrib/Ftptool3.1.tar.Z-tiedosto.

[Sun90a] Sun Microsystems, Inc. Open Look Graphical User Interface Application Style Guidelines, third printing. Addison-Wesley, 1990. ISBN 0-201-52364-7.

[Sun90b] Sun Microsystems, Inc. Open Look Graphical User Interface Functional Specification, third printing. Addison-Wesley, 1990. ISBN 0-201-52365-5.

[Sys90] Systä, Kari. Suoritettavien spesifikaatioiden visualisointi- ja animointijärjestelmä. Lisensiaattityö. TTKK, Sähkötekniikan osasto, 1990.

[Sys91] Systä, Kari. Unix tampereen teknillisessa korkeakoulussa. *Korkeakoulujen atk-uutiset*, (2):11, 1991.

[Tho90] Thomas, S.W. X and PEX Programming. Eurographics 90 Tutorial Note 3, 1990.

[Tia91] Tiainen, Pyry. SAA/CUA-standardin mukaisen käyttöliittymän suunnittelu. Diplomityö. TTKK, Sähkötekniikan osasto, 1991.

[TLP90] Tieteellisen laskennan palvelun toimintakertomus vuodelta 1989. Valtion painatuskeskus, Helsinki, 1990.

[Tol90] Tolonen, Pekka. Intel kehittää multimediamikroa. *Tietotekniikka*, (8), 90.

[Vis90] Tieteellisen laskennan palvelun esite Visualisointi. 1990.

[XTC91] 5th Annual X Technical Conference, Jan 14-16 1991. Sponsored by MIT X Consortium, Boston Marriot Copley Place, 1991.

[You90a] Young, Douglas A. Kirjan 'The X Window System, Programming and Application with Xt, OSF/Motif Edition' esimerkit tietokoneluettavassa muodossa. 1990. ohjelmisto, nic.funet.fi[128.214.6.100]:-pub/X11/contrib/young.motif.tar.Z-tiedosto.

[You90b] Young, Douglas A. The X Window System, Programming and Application with Xt, OSF/Motif Edition. Prentice Hall, Englewood Cliffs, New Jersey 07632, 1990. ISBN 0-13-497074-8.

Hakemisto

A

antialiasing 3
ApE 28
asiakas-palvelin-malli 8

B

backing store 12
bittikartta 2

C

Central Structure Store 11
CGI (Computer Graphics Interface) . 5
CGM (Computer Graphics Metafile) 5
client 8
clipboard 18
CMY 3
Cray X-MP EA/432 22

D

direct manipulation 18
display 9
Display PostScript 15
Disspla 28
Distributed Graphics Library (DGL) 16
dithering 3
drawable 11

E

edustakone 23
event 9
expose 9

F

funcs 30
FUNET 24

G

GKS *(Graphical Kernel System)* . . 4
graphical user interface 18
graphics context 11
Graphics Library (GL) 16

H

HDF 27
HLS 3
HSB 3
hypermedia 7

I

ICCCM 10
ikkuna 10
ikkunahallintaohjelma 10
ikkunointijärjestelmä 8
ikoni 10
Inter-Client Communications Conventions Manual 10

K

komentokieli 20
kuvake 10
kuvanmuodostus 3
käyttöliittymä 17

L

leikekirja 18
leikkaa ja liimaa 18
lomake 20

M

Motif 18
movie.byu 29, 30
multi purpose graphics system . . 28
multimedia 7
mwm *(motif window manager)* . . 18

N

NCAR 28
NeWS 14

O

olwm (open look window manager) 15
Open Look 19
Open Software Foundation 18
OpenWindows 15

P

PEX 11
pexim 11
PHIGS 6
piiloviiva 3
pikseli 2
pixmap 11
PostScript 14
pyyntö 8

R

rasterikuva 2
rendering 3
RGB 3

S

screen 9
sekoitus 3
server 8
4sight 16
siirrä ja sijoita 18
suorakäsittely 18

T

tapahtumat 9
taustavalikko 10
texture 3
Tieteellisen laskennan palvelu . . 22
tietokonegrafiikka 2
TLP 22

U

UIDT 13
UIL, User Interface Language . . 13
UNiRAS 28
Unix International 19
user interface development tool . 13

V

valikko 20
visualisointi 4
vokseli 3
värit 2

W

wavefront 29
widget 12
window manager 10
WYSIWYG 18

X

X-pääte 10
X-palvelin 8
X11-ikkunointi 8
XDataSlice 28
ximage 11, 12, 28
Xlib 12
Xtoolkit 12

Z

Z-puskuri 3

Liite 4.

```
Script started on Sun Mar 24 16:51:14 1991
sun4 /csc/jarvelai/seka/hk 141> ftp nic.funet.fi
Connected to nic.funet.fi.
220 nic.funet.fi FTP server (Version 4.471 +mea@nic.funet.fi
Thu Feb 21 19:27:02 EET 1991) ready.
Name (nic.funet.fi:jarvelai): ftp
331 Guest login ok, send ident (your email address) as password.
Password:
230-Guest login ok, access restrictions apply.
230-There are 11 active foreign anonymous customer(s) presently out of max 16.
230-Local time is now Sun Mar 24 16:54:33 1991
230-There are constant overload problems on our line to USA...
230-  Be carefull with "cd ..".  See  /README for info about how
230-  symlinks might affect your view of, and movement in archive.
230-Some service glitches may occur while we re-arrange those GBytes..
230-We have special access features, see file /README
230 It was last updated Thu Feb 21 19:30:35 1991 - 30.9 days ago
ftp> cd pub/doc/bible/finnish
250 CWD command successful.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for .
-rw-rw-r--  1 pj        ftp         397080 Mar 11 07:57 ut.asc.Z
226 Transfer complete.
66 bytes received in 0.16 seconds (1.3 Kbytes/s)
ftp> bin
200 Type set to I.
ftp> get ut.asc.Z
200 PORT command successful.
150 Opening BINARY mode data connection for ut.asc.Z (396423 bytes).
226 Transfer complete.
local: ut.asc.Z remote: ut.asc.Z
396423 bytes received in 1.3 seconds (2.9e+02 Kbytes/s)
ftp> bye
221 Goodbye.
sun4 /csc/jarvelai/seka/hk 142> uncompress ut.asc.Z
sun4 /csc/jarvelai/seka/hk 143> t ut.asc
Mt.1.1. Jeesuksen Kristuksen, Daavidin pojan, Aabrahamin pojan, syntykirja.
Mt.1.2. Aabrahamille syntyi Iisak, Iisakille syntyi Jaakob,
Jaakobille syntyi Juuda ja tämän veljet;
Mt.1.3. Juudalle syntyi Faares ja Sera Taamarista, Faareelle
syntyi Esrom, Esromille syntyi Aram;
Mt.1.4. Aramille syntyi Aminadab, Aminadabille syntyi Nahasson,
Nahassonille syntyi Salmon;
```

Esimerkissä siirretään Uuden Testamentin sisältävän tiedosto ut.asc.Z tiedostopalvelijasta finsunille ftp-ohjelmalla.

Liite 7

```
#!/bin/csh
banner 'x.csh'
if ( "$1" == "" || "$2" == "" || "$3" == "" ) then
 echo "Usage: $0 X-DISPLAY_name Patran-device Username"
 exit 1
endif
#
alias setup 'source /appl/setup/\!*'
set naytto=$1
set patranlaite=$2
set kayttaja=$3
#
preprocess:
banner 'preproc'
#
#--- Otetaan Patran kayttoon ja asetetaan X-naytto ---
#
setenv DISPLAY ${naytto}:0:0
setup patran
#
#--- Ajetaan ensimmainen Patran-sessio ---
# (generoidaan uusi tietokanta)
#
cat > ajo1.ses <<!
GO
1
6
!
patran -d $patranlaite -s ajo1.ses
#
#--- Ajetaan toinen Patran-sessio ---
# (kaytetaan uutta tietokantaa)
#
cat > ajo2.ses <<!
GO
3
5
1
2
palkki.out
N
Y
PL
SET,LABN,OF
SET,LABE,OF
PL
STOP
```

```
!
patran -d $patranlaite -s ajo2.ses
#
#--- Ajetaan Patran => Abaqus -konversio ---
#
cat > ajo3.ses <<!

4
1

1
palkki.out
y
6
!
pataba < ajo3.ses
#
#--- Ajetaan jobgen , jotta saataisiin
#    ABAQUS-ajovirta Crayta varten ---
#
cat > ajo4.ses <<!
# user=${kayttaja}
# QSUB -r  palkki
# QSUB -q  csc
# QSUB -lT 90
# QSUB -lm 3000Kw
# QSUB -eo
# QSUB -s  /bin/sh
# QSUB
set +x
newacct -l;saldo;who;jstat
mkdir -p /tmp/${kayttaja}/cray;cd /tmp/${kayttaja}/cray
echo "*** Execution directory is: /tmp/${kayttaja}/cray ***"
fetch palkki.inp -t'palkki.inp'
echo "abaqus palkki"
abaqus palkki
dispose palkki.a06 -t'/tmp/${kayttaja}/finsun/palkki.a06'
dispose palkki.sta -t'/tmp/${kayttaja}/finsun/palkki.sta'
pwd;ls -al;who;jstat
!
banner 'cray-run'
echo "Enter your Cray"
cray submit ajo4.ses
sleep 5
cray status own
#
#--- Seuraava 'goto' on tassa siksi,
#    etta ei tehtaisi tulevaa rlogin crayhin !
#
```

```
goto postprocess
#
#--- Seuraavaksi mennaan Craylle ajamaan tuloksien
#    takaisinkonversio (abapat) -
#
rlogin cray
#
#   ************************************
#-- Nama ovat jo sitten Crayn komentoja !!
#   ************************************
#
cd /tmp/$kayttaja/finsun
ls -al palkki.fil
#
#-- Ajetaanpa Crayssa abapat-konversio interaktiivisesti
#
cat > ajo5.ses <<!

7
1
2

1
palkki.fil
y
9
!
abapat < ajo5.ses
rcp s41m1.dis.* finsun:/tmp/$kayttaja/finsun/
exit
#
#   **********************************
#-- Tahan loppuivatkin Cray-komennot !!
#   **********************************
#
postprocess:
banner 'postproc'
cat > ajo6.ses <<!
GO
3
4
2
3
4
1
palkki.dis
1
2
7
```

```
STOP
!
patran -d $patranlaite -s ajo6.ses
#
#-- Loppu !
#
banner 'the end'
```