

Softari

NOKIAN OHJELMISTOLEHTI 3/86



Ohjelmistomarkkinoiden painopiste mikroissa

Mikrotietokoneet alkoivat muuttaa atk-maailmaa toisenlaiseksi 1970-luvun loppupuolella. Varsinainen vallankumous käynnistyi siitä, kun huomattiin, että mikro on todella jotain aivan uutta, ei ainoastaan aikaisempien tietokoneiden pienennetty ja halvennettu olomuoto.

Ensimmäiset mikrojen ohjelmistot olivat vielä pientietokonesovellusten alaspäin siirrettyjä versioita. Hyvin pian ilmaantui ohjelmistoista kuitenkin uusi sukupolvi, joka osasi hyödyntää henkilökohtaisten tietokoneiden uusia mahdollisuuksia: omassa käytössä olevaa tehokasta suoritinta, edullista grafiikkaa, sekä suorittimen ja näytön välistä suurta kaistaleveyttä, joka poisti sovellusten toteuttamista siihen saakka vaivanneen tietoliikenteen pulonkaulan.

Hyvin pian mikrot ja niiden ohjelmistot erkanivat kokonaan omaksi kulttuurikseen, joka tänä päivänä enää varsin vähän muistuttaa sitä pientietokoneen käyttöympäristöä, josta lähdettiin liikkeelle.

Mikrojen ohjelmistokehitys on johtanut myös joihinkin vääristymiin. Henkilökohtainen, omaehtoinen käyttömuoto on korostunut usein liiaksi ja peittänyt alleen tietokoneen perinteiset käyttötavat, monen käyttäjän järjestelmät, yhteiset tietokannat ja suurten aineistojen massakäsittelyn eräajoina. Myyjien tavaksi on tullut suosittaa mikroja ja uusia ohjelmistoja patenttilääkkeenä kaikkiin vaivoihin.

Mikrotietokoneiden ja niille sovitettujen ohjelmistojen hyväksikäytön voi kuitenkin katsoa olevan vasta aivan alussa. Seuraavassa on joukko näkemyksiä lähivuosien todennäköisistä kehityssuunnista.

Mikrojen uuden käyttötavan, Xerox Star/Apple Macintosh -tyyppisen käyttöliittymän ikkunoineen, kuvineen ja hiirineen nähdään tulevan nopeasti yleiseen käyttöön myös teollisuusstandardin mukaisissa mikroissa. Seurausena on oleellisesti helpompi ja loogisempi käyttötapa ja ennen kaikkea eri valmistajien ohjelmistojen keskenään samanlainen ulkoinen käyttäytyminen.

Integroidut työkaluohjelmistot väistynevät vähitellen ja tilalle tulee samoja standardeja noudattavien erillisten ohjelmien joukko.

Graafinen käyttöliittymä sitoo eri tuotteet keskenään yhteiskäyttöön pystyväksi kokonaisuudeksi.

Uusia tietokoneen käyttöalueita työntyy voimakkaasti esiin ensin mikroilla ja sitten myös suuremmilla koneilla. Esimerkkejä ovat yleiskäyttöinen kuvankäsittely business-graafikasta helppokäyttöiseen CAD:hen ja nopeasti tulossa oleva Desktop Publishing.

Lähiverkot yleistyvät nopeasti ja synnyttävät laajaa kysyntää hyville, verkossa tehokkaasti toimiville monen käyttäjän ohjelmistoille. Henkilökohtaisessa käytössä suosiota saavuttaneet tuotteet menettänevät jalansijaa alusta alkaen verkkopohjaisiksi suunnitelluille ohjelmistoille.

Hierarkkiset tietojärjestelmät nousevat vähitellen tärkeään asemaan ja synnyttävät sekä valmisohjelmia että työvälineitä, jotka soveltuvat yhteiskäyttöisiin mikro-keskuskone ja mikro-mini-keskuskone -ympäristöihin.

Standardimikrojen laitekilpailun tapaan myös ohjelmistojen alueelle ilmestyy menestystuotteiden näköiskopioita, jotka pystyvät riittävällä tasolla samaan palveluun kuin esikuvansa mutta murto-osalla niiden hinnasta. Ennakkoluulottomat yritykset voivat näiden tuotteiden avulla ohjata huomattavan osan ohjelmistokustannuksistaan muuhun käyttöön.

Räätälöityjen ohjelmistojen merkitys tulee kasvamaan, kun sovelluskehityksen välineet juuri mikroilla tehostuvat ja tietojärjestelmät yrityksissä muuttuvat yhä tärkeämmiksi kilpailutekijöiksi. Keskenään kilpailevat yritykset toteavat mahdolltomaksi hyödyntää samaa valmisratkaisua liiketoimintaan vaikuttavissa tietojenkäsittelytehtävissä.

Kotimainen ohjelmistovalmistus nousee aallonpohjastaan, kun yhä useammin nähdään amerikkalaisten standardiratkaisujen huono sopivuus suomalaiseen ympäristöön.

Yhteenvetona voi vielä todeta, että ohjelmistomarkkinoiden kehityksen painopiste näyttää pitkään pysyvän mikrojen alueella. Näköpiirissä olevat uudet tehomikrot, moni-ajokäyttöjärjestelmät, verkot, salkkukoneet ja monet muut kehitysnäkymät pitävät henkilökohtaiset tietokoneet jatkuvasti alan suurimman mielenkiinnon kohteena.

Softari

Nokian ohjelmistolehti

Päätoimittaja:

Hannu Markus
puh. (90) 567 2982

Toimitussihteeri:

Outi Kuusela
puh. (90) 567 2017

Piirroksat:

Taina Sonninen

Osoitemuutokset:

Jaana Lamberg
puh. (90) 567 2980

Toimituksen osoite:

Nokia Informaatiojärjestelmät
Softari-lehti
Kutomotie 16-18
PL 780
00101 HELSINKI

Painosmäärä: 11000

Kirjapaino: Vientipaino Oy

Julkaisija:

Nokia Informaatiojärjestelmät

ISSN 0782-8209

Kansikuva:

Kansikuvassamme 5-vuotias Anne Kaarina Släen pelaa omenan pudotus-peliä omalla salkkumikrollaan, jonka hän on itse ideoinut ja rakentanut Nokian saapaslaatikosta. Kotimaisen tietokoneollisuuden tulevaisuus lienee turvattu.

NOKIA

Tässä numerossa mm.

- MikroMikko 3
- MS-DOS 3.1
- Windows
- Project
- Tiedonhallinta-ohjelmien vertailu



EUREKA

Lue myös Nokian Eureka-projekteista s. 28-29

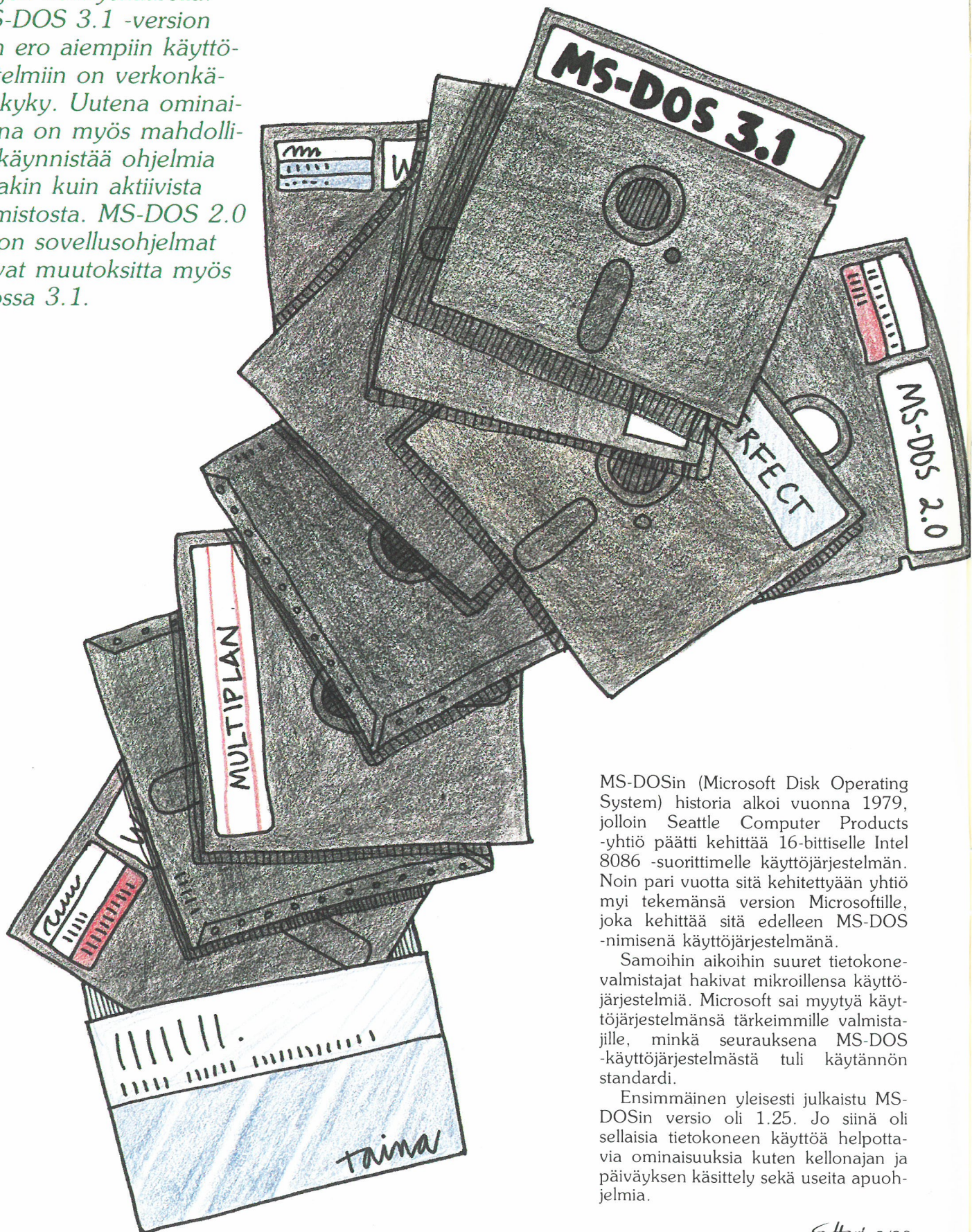


Viereisen sivun pääkirjoituksen on kirjoittanut dipl.ins. Tuomas Kotovirta. Hän toimii nykyisin konsulttina Tietojenkäsittelyneuvonta Oy:ssä. Kotovirralla on kokemusta tietokonealalla aina 60-luvun alusta alkaen. Tuolloin hän työskenteli Suomen IBM:llä. Vuosina 1967-85 Kotovirta toimi Valtion Tietokonekeskuksessa, viimeksi tutkimusosaston päällikkönä.

Mikrotietokoneiden käyttöjärjestelmien uusi tulokas on MS-DOS 3.1. Se on jälleen yksi virstanpylväs mikrojen kehityskaarella.

MS-DOS 3.1 -version suurin ero aiempiin käyttöjärjestelmiin on verkonkäyttelykyky. Uutena ominaisuutena on myös mahdollisuus käynnistää ohjelmia muistakin kuin aktiivista hakemistosta. MS-DOS 2.0 -version sovellusohjelmat toimivat muutoksitta myös versiossa 3.1.

MS-DOS 3.1 hallitsee verkotkin



MS-DOSin (Microsoft Disk Operating System) historia alkoi vuonna 1979, jolloin Seattle Computer Products -yhtiö päätti kehittää 16-bittiselle Intel 8086 -suorittimelle käyttöjärjestelmän. Noin pari vuotta sitä kehitettyään yhtiö myi tekemänsä version Microsoftille, joka kehittää sitä edelleen MS-DOS -nimisenä käyttöjärjestelmänä.

Samoihin aikoihin suuret tietokonevalmistajat hakivat mikroillensa käyttöjärjestelmiä. Microsoft sai myytyä käyttöjärjestelmänsä tärkeimmille valmistajille, minkä seurauksena MS-DOS -käyttöjärjestelmästä tuli käytännön standardi.

Ensimmäinen yleisesti julkaistu MS-DOSin versio oli 1.25. Jo siinä oli sellaisia tietokoneen käyttöä helpottavia ominaisuuksia kuten kellonajan ja päiväyksen käsittely sekä useita apuohjelmia.

Seuraava MS-DOS -versio 2.0 tuli markkinoille vuoden 1983 alussa. Se toi mukanaan paljon uusia ominaisuuksia kuten esimerkiksi hierarkkisen hakemistorakenteen sekä syötön ja tulostuksen ohjauksen tiedostoon tai toisen ohjelman syötteeksi. Yleispiirteensä oli aiempiin versioihin verrattuna suuri joustavuus, mikä näkyi mm. erilaisten laiteohjainten lisäämisen helppoutena.

Uutena piirteenä verkonkäsitteilykyky

Uuden MS-DOS 3.1 -version suurin ero aikaisempaan versioon 2.0 on verkko-ominaisuudet. Se nimittäin mahdollistaa yhteisten resurssien käytön paikallisverkon välityksellä.

Verkko-ominaisuudet näkyvät uusiina komentotulkin komentoina, joilla käyttäjät voivat esimerkiksi jakaa yhteisiä laitteita.

MS-DOS 3.1:ssä on ohjelmiojien käytettävissä joukko uusia systeemi-kutsuja. Niillä voidaan verkossa esimerkiksi lukita ja vapauttaa tiedostoja tai tiedoston osia. Systeemi-kutsuilla voidaan myös ohjata verkon laiteviittauksia.

Uusia systeemi-kutsuja käyttäen MS-DOS 3.1 -järjestelmään tehdyt sovellusohjelmat toimivat myös MS-NET -verkossa. Vaikka MS-DOS 3.1 on näin toiminnallisesti yhteensopiva MS-NET -verkkokäyttöjärjestelmän kanssa, se on kuitenkin yhden käyttäjän järjestelmä. MS-NET (Microsoft Networks) on itse asiassa MS-DOSin laajennus.

MS-DOS 3.1 -versiossa pystyy aiemmista poiketen käynnistämään ohjelmia polkujen takaa. Järjestelmässä on mahdollista tutkia ohjelmasta käsin, mistä hakemistosta ohjelma käynnistettiin. Taustatulostusohjelmissa voidaan käyttää polkuja tiedostonimisessä.

Uudessa MS-DOS 3.1 -versiossa loogisten levyasemien suurinta sallittua määrää on pienennetty 26:een. Aiempi 2.0 -järjestelmä kykenee käsittelemään peräti 63 levyasemaa.

MS-DOS 3.1 perinyt edeltäjiensä ominaisuuksia

MS-DOS 3.1:n kuten aiempienkin versioiden yksi tärkeimmistä piirteistä on laitetoimittajien ja ohjelmistotalojen laaja tuki. Se merkitsee ohjelmistovalikoiman runsautta.

MS-DOS 3.1 - kuten edeltäjänsäkin - on modulaarinen käyttöjärjestelmä. Niinpä järjestelmää on helppo laajentaa tai tarvittaessa karsia käyttämättömiä osia pois.

Järjestelmään on helppo lisätä tai poistaa laiteohjaimia. MS-DOS -järjestelmässä on Config.sys -niminen tiedosto, joka sisältää tarvittavien laiteohjaimien nimet. Kun MS-DOSia ladataan muistiin, se hakee laiteohjaimien nimet config.sys -tiedostosta ja sijoittaa ohjaimet koneen muistiin. Ominaisuuden ansiosta MS-DOS -käyttöjärjestelmää sanotaan konfiguroitavaksi.

MS-DOS -järjestelmässä prosessi voi käynnistää aliprosesseja, joiden suorituksen päätyttyä ohjaus palautuu takaisin käynnistäneeseen prosessiin. MS-DOSissa on mahdollista myös taustatoiminta, esimerkiksi tulostus kirjoittimelle. Näistä ominaisuuksista huolimatta MS-DOS 3.1 on luokiteltava yhden käyttäjän ja tehtävän käyttöjärjestelmäksi, koska järjestelmän palvelurutiinit eivät ole vapaakäytisiä. MS-DOSin päälle rakennetun Windowsin avulla on kuitenkin mahdollista käyttää samanaikaisesti useampia ohjelmia.

MS-DOS 3.1 käsittelee sekä tiedostoja että oheislaitteita tasavertaisesti. Käyttöjärjestelmän tuntemat tavalliset oheislaitteet käyttäytyvät komentotulkin kannalta tiedostojen tapaan. Esimerkiksi näytöltä voidaan kopioida teksti kirjoittimelle aivan samaan tapaan kuin tavallinen tiedosto levyltä toiselle.

MS-DOS mahdollistaa erilaisten kansallisten merkintätapojen huomioimisen. Järjestelmälle voidaan nimittäin kertoa, minkälainen on kyseisen maan merkintätapa kellonajalle, päiväykselle ja rahayksikölle.

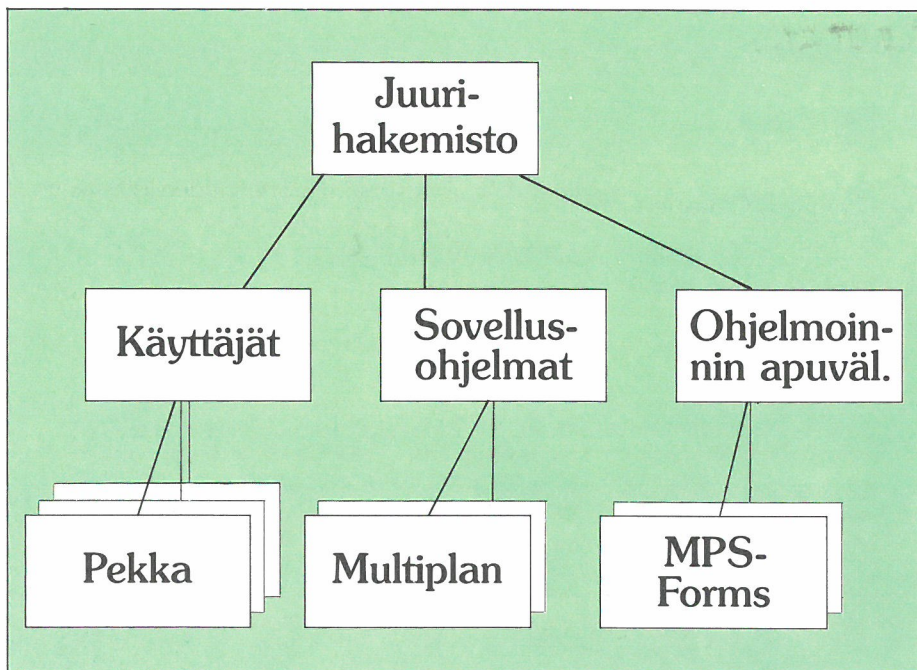
Käyttöjärjestelmän mukana toimitetaan useita apuohjelmia, joilla helpotetaan tietokoneella työskentelyä. Useimmat laitevalmistajat ovat näiden ohjelmien lisäksi valmistaneet omia apuohjelmiaan. Näin eri valmistajien MS-DOS -versioiden käyttö saattaa poiketa toisistaan.

Ohjelmayhteensopivuutta ylöspäin

MS-DOS 3.1 tuo sovellusohjelmiojien käyttöön uusia käyttöjärjestelmä-funktioita ja siten tietenkin entistä parempia palveluita.

Kaikista eroavaisuuksistaan huolimatta MS-DOS 2.0 -järjestelmään tehdyt sovellukset toimivat myös MS-DOS 3.1:ssä, sillä aiempien versioiden funktiot on säilytetty uudessa versiossa. Ylöspäin yhteensopivuuden periaate lienee jatkossakin kantava idea.

Uudelle käyttöjärjestelmäversiolle tehdyt ohjelmat eivät välttämättä toimi taas aiemmissa. Näin on, jos ohjelmassa on käytetty uusia funktiokutsuja. MS-DOS 3.1 -ympäristössä ohjelmioja saattaakin huomata, että on valittavissa esimerkiksi kaksi samaa toimintaa tekevää funktiokutsua. Vaihtoehtoista kannattaa yleensä valita uudempi kutsu, koska se edustanee kehittyneempää ratkaisua. Samalla tietenkin rajoitetaan ohjelman toimimista nykyistä aiemmissa järjestelmäversioissa.



MS-DOS -järjestelmässä on hierarkkinen hakemistorakenne. Jos usea henkilö käyttää samaa tietokonetta, kannattaa hierarkkisuutta käyttää hyväksi sekaannusten välttämiseksi. Esimerkiksi jokaisella käyttäjällä voisi olla oma hakemistonsa.

MS-DOS 3.1 ja PC-DOS 3.1 lähes samanlaisia

MS-DOSin lähisukulainen PC-DOS on käyttäjän näkökulmasta samanlainen järjestelmä. Käyttäjälle näkyviä eroja on lähinnä laitevalmistajien tekemissä, järjestelmän mukana tulevissa varusohjelmissa. Ei ole mikään ihme, että eroja on vain vähän, koska sekä MS-DOS että PC-DOS ovat lähtöisin samalta valmistajalta, Microsoftilta.

Käyttöjärjestelmien sisäisissä rakenteissa on joitakin eroja. Ohjelmien yhteensopivuudessa ei pitäisi esiintyä järjestelmien välillä ongelmia, jos ohjelmat käyttävät vain käyttöjärjestelmäkutsuja. Jotkut sovellusohjelmat kuitenkin ohittavat käyttöjärjestelmän ja käyttävät hyväkseen laitteen jotain eri-

koispiirrettä. Tällöin ohjelman siirrettävyys järjestelmästä toiseen hankaloituu.

Kirjallisuutta

Tässä lyhyessä artikkelissa on mahdollonta esitellä MS-DOS 3.1 -järjestelmää kovin laajasti. Asiasta kiinnostuneet löytävät hyödyllistä lisätietoa mm. seuraavista teoksista:

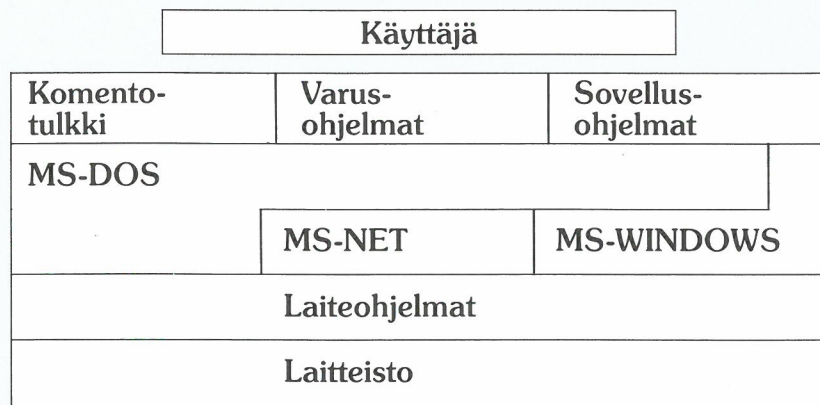
Jokinen Timo ja Linturi Risto: MS-DOS ja PC-DOS, Käyttäjän opas. Espoo 1986.

Norton Peter: MS-DOS and PC-DOS. London 1984. □

Mitä käyttöjärjestelmä on ?

Tietokone itsessään on vain erilaisista osista koottu rakennelma — kasa 'rautaa'. Henki tähän laitteeseen puhalletaan ohjelmien avulla. Sovellusohjelma ei kuitenkaan pääsääntöisesti käytä suoraan hyväkseen tietokoneen osasia, vaan se tarvitsee avukseen toisen ohjelman — käyttöjärjestelmän — kyetäkseen keskustelemaan laitteen kanssa.

Sovellusohjelman ja itse laitteen väliin on rakennettu sovittimeksi nimenomaan käyttöjärjestelmä, jonka avulla tarjotaan sovellusohjelmalle palveluja, kuten esimerkiksi levyltä luku tai tiedoston lähetys kirjoitinkanavalle. Näin vältetään myös kätevästi ohjelmoimasta näitä peruspalveluja erikseen jokaiseen tietokoneeseen ajettavaan sovellusohjelmaan.



Kuvassa on periaatteellinen esitys MS-DOS -käyttöjärjestelmän sijoittumisesta tietokoneympäristöön. Alin taso on tulkittavissa laitteistoksi: näppäinistö, näyttölaite ja ohjausyksikkö. Seuraavassa kerroksessa on laiteohjelmat, jotka toimivat välittäjinä MS-DOSin ja laitteen välissä. Laiteohjelmat on suunniteltu juuri käytettävää laitetta ajatellen, joten niitä ei voida siirtää sellaisenaan muihin tietokoneisiin.

MS-DOS -käyttöjärjestelmä toimii sovelluksen ja laiteohjelmien välissä. Se on itsessään suunniteltu toimivaksi missä hyvänsä tietokoneessa, koska se kommunikoi laitteiston kanssa laiteohjelmien avulla.

Sovellusohjelmat - ainakin periaatteessa - käyttävät vain MS-DOSin, komentotulkin tai varusohjelmien palveluita. Käyttäjä taas normaalisti käyttää sovellusohjelmia komentotulkin palveluita tai varusohjelmia.

Käyttöjärjestelmä on siis yksi tulkitsija monitasoisessa käyttäjän ja tietokoneen välisessä keskustelussa.

Olli Nurminen

MikroMikko 3:n M Yhteensopivu

MikroMikko 3 on yhteensopiva toisaalta kakkosen ja toisaalta teollisuusstandardin kanssa.

MikroMikko 2:n laaja kotimainen ohjelmisto, joka käyttää vain MS-DOSin palveluja, toimii sellaisenaan myös kolmosessa. Niin ikään markkinoilta saatavat PC/MS-DOS -valmisohjelmat käyvät kolmosseen.

MikroMikko 3 (Nokia ASC) julkistettiin kesäkuun alussa täydentämään MikroMikko-perhettä. Laitteen käyttöjärjestelmänä on Microsoftin MS-DOS 3.1.

Laite on monilta ominaisuuksiltaan — esimerkiksi grafiikaltaan — kehittyneempi kuin muut teollisuusstandardin mukaiset mikrot. Tämän sekä yhteensopivuuksien vuoksi Nokia on joutunut lisäämään MS-DOSiin useita varusohjelmia ja lisäpiirteitä, kuten valinta- eli pehmonäppäinten ja toimintatilojen määrittelyt.

Yhteensopivuus toimintatiloilla

Yhteensopivuus MikroMikko 2:n ja teollisuusstandardin mukaisten laitteiden kanssa on saavutettu toteuttamalla MikroMikko 3:n käyttöjärjestelmään kolme toimintatilaa: perus-, Nokia PC- ja jäljittelytila (emulointitila).

S-DOS:

vuutta eri mikrojen kanssa



MikroMikko 3

Perustilassa ajetaan PC/MS-DOS -valmisohjelmia, kun taas ns. Nokia PC -tilassa toimivat useat MikroMikko 2:n sovellukset. Jäljittelytilassa ajetaan niitä ohjelmia, jotka eivät pysty hyödyntämään kummankaan muun toimintatilan ominaisuuksia.

Yhteensopivuuteen on varauduttu myös useilla näytönkäsittelymahdollisuuksilla. MikroMikko 3:n DOS osaa teollisuusstandardin mukaisen sekä MikroMikko 2:n grafiikan lisäksi Hercules -mustavalkografiikan. Laitteeseen on saatavissa myös EGA-yhteensopiva (Enhanced Graphics Adapter) värinäyttö ja kortti.

MikroMikko 3:n suorittimena on kansainväliseksi teollisuusstandardiksi muodostunut 16-bittinen Intel 80286-mikrosuoritin. Sen kellotaajuus on kahdeksan megahertsia. Suuren kellotaajuuden ansiosta MikroMikko 3:n nopeus on huippuluokkaa.

Laitteessa on peruskokoonpanossaan keskusmuistia joko 512 tai 1024 kilotavua, mutta se on laajennettavissa muistinlaajennuskorteilla aina 16 megatavuun.

Koneen kahdeksasta laajennuskortista yksi on normaalisti varattu näytönohjaimelle ja toinen levy/levykeohjaimelle.

Massamuistiksi laitteeseen on saatavissa yksi tai kaksi 1,2 megatavun levykeasemaa sekä valinnaisena 20 tai 40 megatavun umpilevy (Winchester). Levykeasemat pystyvät lukemaan 360 ja 720 kilotavun levykkeitä sekä myös kirjoittamaan 720 kilotavun levykkeille.

Varmuuskopiointia varten voidaan ohjausyksikköön sijoittaa nauha-asema. Nauha-aseman kapasiteetti on 45 megatavua. Niinpä umpilevyn koko sisältö voidaan kerralla varmistaa nauhalle.

MikroMikko 3:n näyttö on ominaisuuksiltaan MikroMikko 2:n näytön veroinen. Se on 15-tuumainen ja siinä on tummat merkit vaalealla pohjalla. Korkean virkistystaajuuden (71 Hz) ansiosta näyttö on välkky-mätön. Sen erotuskyky voi olla jopa 720 x 400 kuvapistettä.

MikroMikko 3 on saatavissa myös 14 tuuman värinäytöllä varustettuna.





Kolmosen yhteensopivuutta vahvistavat vielä erilaiset kirjasinlajit (fontit). MikroMikko 3:ssa on kakkosen sekä teollisuusstandardin mukaiset kirjasinlajit. Kirjasinlajeja voidaan määritellä myös itse. Kirjasimet vaihtuvat automaattisesti toimintatilan mukana, mutta ne voi vaihtaa myös Fontload-ohjelmalla.

Ohjelmat kakkoselta kolmoselle

MikroMikko 2:lle kehitetyt ohjelmat voidaan siirtää kolmoselle. Aivan sellaisenaan ne eivät kuitenkaan välttämättä toimi MikroMikko 3:ssa, koska laitteistojen välillä on kaikesta huolimatta eroja jonkin verran. Ohjelmat, jotka käyttävät vain MS-DOSin palveluja osoittamatta suoraan laitetta, siirtyvät vaikeuksitta.

Useimmat Nokian myymät MikroMikko 2 -ohjelmat, kuten Multiplan tai WordPerfect, eivät toimi sellaisenaan ilman eri sovitusta MikroMikko 3:ssa. Syynä sovitustarpeeseen on esimerkiksi ohjelman suorat laiteosoitukset. Niitä on tyypillisesti mm. tietoliikenneohjelmissa. Samoin suoria laiteosoituksia on käytetty ohjelmissa niiden nopeuttamiseksi.

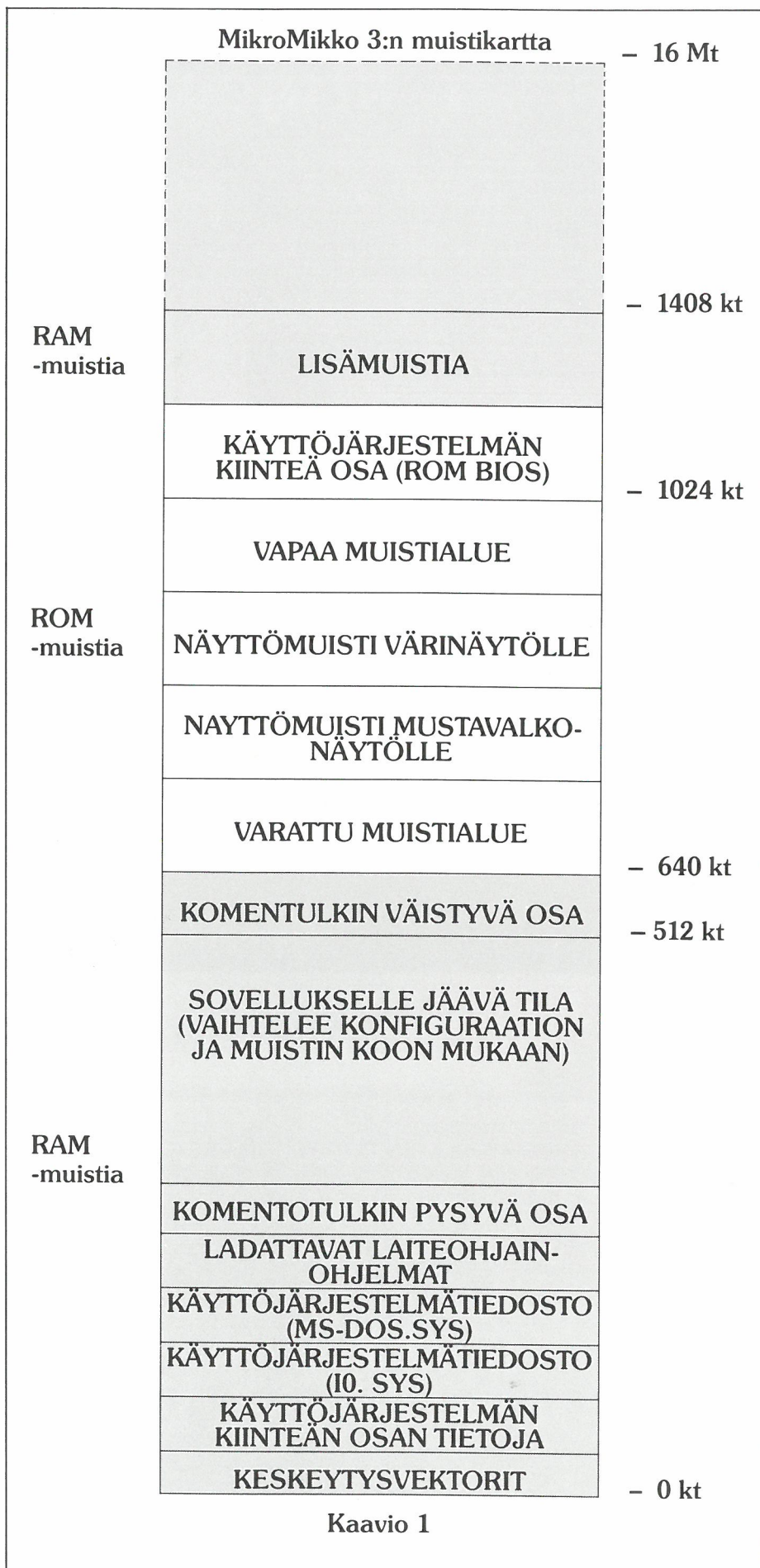
Nokian varusohjelmat helpottavat käyttöä

MikroMikko 3:n DOSin mukana toimitetaan myös Nokian kehittämiä varusohjelmia. Tällaisia ovat mm. Conf-, Hardconf-, Fontload-, Keydef-, Keyload-, Softdef-, Softload, Linconf-, Linload-, Modedef-, Modeset- ja Xmode-ohjelmat. Ne mahdollistavat mikron käytön erilaisissa käyttöympäristöissä.

Conf-ohjelman avulla voidaan tutkia ja päivittää käyttöjärjestelmän konfiguraatiota (Config.sys-tiedostoa). Sillä määritellään esimerkiksi ladattavat laiteohjainohjelmat (drivers), maakoodi ja tiedostopuskurien (buffers) lukumäärä.

Linconf-ohjelmalla voidaan muuttaa RS232-ohjaimen linjaliikenneparametreja. Tämä on tarpeen esimerkiksi käytettäessä erilaisia kirjoittimia.

Nokia PC -tilan Hardcopy-ohjaimen parametrit on asetettavissa Hardconf-ohjelmalla. Muiden toimintatilojen Hardcopy-ohjainten parametreja ei voi muuttaa.



Kaavio 1

Keydef-ohjelmalla MikroMikko 3:n näppäimistöködit voidaan määrittellä ja tallettaa käyttäjän tiedostoon myöhemmin ladattavaksi. Toinen mahdollisuus on määrittellä ja aktivoida näppäimistömääritykset heti käytettäväksi, jolloin ne häviävät virtaa katkaistaessa.

Erillisessä tiedostossa olevat näppäinmäärittelyt voidaan ladata Keyload-ohjelmalla. Sovellusohjelmissa näppäinmäärittelyt on ladattavissa tiedostosta myös Keyread-aliohjelmalla. Tämä on mahdollista C-, Pascal-, Fortran-, Cobol-, Assembler- ja Basic-kielissä ohjelmista.

Valintanäppäimet myös kolmosessa

MikroMikko 3:ssa ei ole tingitty MikroMikko 2:n käyttäjäystävällisyydestä. Näin ollen kakkosesta tutut valinta- eli pehmonäppäimet löytyvät myös kolmosesta. Valintanäppäinten uudelleenmäärittely ja käyttö ovat mahdollisia vain Nokia PC -tilassa. Varsinaisissa teollisuusstandardin mukaisissa ohjelmistoissa ei tunneta valintanäppäimiä.

Valintanäppäinten määrittelyjä voi luoda ja päivittää Softdef-ohjelmalla. Sovellusohjelma voi käsitellä 20 erilaista valintanäppäinjoukkoa, joista jokainen sisältää kaikkien kahdeksan valintanäppäimen määrittelyt. Kullekin näppäinjoukon näppäimelle voidaan määrittellä jokin toinen valintanäppäinjoukko, joka aktivoidaan käyttöön ko. näppäintä painettaessa. Valintanäppäinjoukkojen (20 kpl) määrittelyt talletetaan käyttäjän nimeämään tiedostoon. Sieltä ne voidaan ottaa käyttöön Softload-ohjelmalla tai sovellusohjelmaan mukaan liitettävän Softread-aliohjelman avulla. Aliohjelma on liitettävissä C-, Pascal-, Fortran-, Cobol-, Assembler- ja Basic-kielisiin ohjelmiin.

Fontloadin avulla voidaan tiedostosta ladata erilaisia kirjasinlajeja (fontteja). Näin saadaan esimerkiksi kreikkalainen merkkistö käyttöön. Merkkitiedostot tehdään Microsoft Windowsin Fontedit-ohjelmalla.

MikroMikko 3:n kiinteät toimintatilat, perus- ja Nokia PC -tila, ovat myös muuteltavissa. Uusia toimintatiloja luodaan ja muutellaan Modedef-apuohjelman avulla. Määritykset voidaan tallettaa tiedostoon, josta ne otettavissa käyttöön Modeset-ohjelmalla.

Yksittäistä aktiivisen toimintatilan ominaisuutta muutetaan Xmode-ohjelmalla. Tällaisia ovat mm. kelloaajuus ja merkkivalikoima.

MikroMikko 3:n MS-DOSissa on mukana useita standardista MS-DOSis-

takin tai PC-DOSista löytyviä varusohjelmia. Osaan näistä Nokia on tehnyt pieniä parannuksia, joissa on huomioitu nimenomaan MikroMikko 3:n laitteisto. Näistä voisi mainita Fdisk- ja Setup-ohjelmat. Fdiskillä voi jakaa umpilevyn useiksi loogisiksi levyiksi (partitioiksi), joita kutakin käytetään itsenäisinä levyinä. Setup-ohjelmalla talletetaan muistiin esimerkiksi kellonaika, päiväys ja järjestelmän lisäominaisuuksia. Näin ei esimerkiksi tarvitse jokaisen käynnistyksen yhteydessä antaa kellon aikaa tai päivämäärää, koska MikroMikko 3:n kello pysyy ajassa.

DOS tuntee levyt ja kirjoittimet

MikroMikko 3:n MS-DOS 3.1 osaa käsitellä useita eri levy- ja levykekokoja: 20 ja 40 megatavun umpilevyjä, 1,2 megatavun, 720 ja rajoitetusti myös 360 kilotavun levykkeitä.

MikroMikko 3:n DOS mahdollistaa useiden eri kirjoittimien käytön ohjelmasta riippuen. Käytettävissä ovat esimerkiksi NOKIA Laserjet -laserkirjoitin, NOKIA SP 320 -laatuksikirjoitin ja NOKIA DX 2100/2200 -matriisikirjoitin. Kirjoittimeksi kannattaa aina valita sellainen, jota nimenomaan Nokia tukee kolmoseen.

Miten MS-DOS sijoittuu muistiin

Kaaviossa 1 on esitetty yhden megatavun keskusmuistilla (RAM) varustetun MikroMikko 3:n muistikartta. Siitä voidaan nähdä, että MS-DOS kaikkine rutiineineen ja liityntöineen asettuu MikroMikko 3:n muistin alimpiin osoitteisiin. Siellä ovat mm. keskeytysvektorit, itse MS-DOS ilman laitevalmistajan lisäyksiä, ladattavat laiteohjaimet jne.

Sovellusohjelmille jäävä tila vaihtelee käyttöjärjestelmän konfiguraation viemästä tilasta ja muistin koosta. Ohjelmamuistin yläreunassa on komentotulkin väistytävä osa. Se on kaavion esimerkissä 640 kilotavun alapuolella. Jos MikroMikko 3:ssa on 512 kilotavun keskusmuisti, komentotulkin väistytävä osa asettuu tietenkin juuri tämän rajan alapuolella.

Luku/kirjoitusmuistia (RAM) on MS-DOSin ja sovellusohjelmien suoraan käyttöön maksimissaan 640 kilotavua. MS-DOS sinänsä voi osoittaa yhtä megatavua. Teollisuusstandardi on kuitenkin varannut 640 kilotavun ja

1 megatavun välisen alueen mm. luku- ja kirjoitusmuistille (ROM) ja näytönohjaimille.

Ylintä osoiteavaruutta (yli megatavun osaa) voidaan hyödyntää esimerkiksi määrittelemällä sinne virtuaalisia RAM-levykeitä (Vdisk-ohjelma). Näitä levykeitä voidaan käyttää sovellusohjelmista tai komentotulkista.

Jotkin sovellusohjelmat, kuten Lotus 1-2-3, osaavat käyttää ylintä luku/kirjoitusmuistin osaa sellaisenaan RAM-levykkeen tapaan, jolloin esimerkiksi taulukkolaskennassa voidaan käsitellä suurempia taulukoita kuin ilman ylemmän muistinosaa.

Kansallisuus huomioitu maakoodilla

Kansallisuustuki MS-DOSissa tarkoittaa, että rahayksiköt, kellonajat, päivämäärät ja isot/pienet kirjaimet ilmoitetaan kansallisessa muodossa.

Nokian MS-DOS tukee seuraavia kansallisuuksia: Alankomaat, Australia, Belgia, Espanja, Iso-Britania, Israel, Italia, Norja, Ranska, Ruotsi, Saksa, Suomi, Sveitsi, Tanska ja Yhdysvallat. Kolmonen olettaa kansallisuudeksi Yhdysvallat, ellei maakoodia ole määritetty Config.sys-tiedostossa.

Useita ohjelmia samanaikaisesti Windowsilla

Vaikka MS-DOS 3.1 on yhden käyttäjän järjestelmä, voidaan Windowsin avulla käyttää samanaikaisesti useampia ohjelmia. Kutakin omassa ikkunassaan. Ohjelmasta toiseen siirrytään joustavasti osoitinhiiren avulla tai muuttamalla näppäinpainalluksella. Samaten tietojen siirto ohjelmasta toiseen käy helposti.

MS-DOSin tiedostokäsittelyssä on huomioitu jo monen käyttäjän ympäristö. Niinpä versiolle 3.1 voidaan tehdä ohjelmia, jotka toimivat jatkossa myös verkkoympäristössä.

Runsaasti valmisohjelmistoja

Ohjelmointikieliksi saa Nokialta Alsysin Ada-kääntäjän ja Microsoftin Basic-, Pascal-, C-, Cobol- ja Fortran-kääntäjät sekä Basic-tulkin.

Lisäksi Nokia on kehittänyt näppäriä työvälineitä ohjelmistokehitystyöhön. Jo MikroMikko 2:sta tutut MPS-Index ja MPS-Forms ovat käytettävissä myös MikroMikko 3:ssa.

MikroMikko 2:een on saatavissa samoja suomenkielisiä ohjelmia eri toimialojen tarpeisiin. Useat näistä kotimaisten ohjelmistotalojen tuotteista on jo siirretty myös MikroMikko 3:lle.

Nokia tarjoaa kattavan valikoiman erilaisia maailmalla tunnettuja valmisohjelmia, jotka on sovitettu MikroMikko 3:een. Sovittamisessa on pyritty hyödyntämään laitteiston erikoispiirteitä kuten valintanäppäimiä. Siten valmisohjelmien käyttö MikroMikolla on helpompaa ja tehokkaampaa kuin monilla muilla mikroilla. Lisäksi Nokian markkinoimien oheislaitteiden kuten kirjoittimien toiminta näiden ohjelmien yhteydessä on taattu. Niin ikään Nokia tarjoaa sovittamiensa ohjelmien käyttäjille asiakastukea (MikkoInfo-puhelin-

neuvontaa) sekä käyttökoulutusta NI-TEC-koulutuskeskuksessa.

Nokian sovittamia valmisohjelmia löytyy tekstinkäsittelyyn (Edix + Wordix, WordPerfect), taulukkolaskentaan (Multiplan, Multigraph, MS-Chart, Lotus 1-2-3, Symphony), tiedonhallintaan (Kmies, MS-Rbase, PC-Focus, PC-Info, Oracle) ja projektinhallintaan (MS-Project). Ohjelmistovalikoima laajenee ja kehittyy koko ajan yhteistyössä maailman johtavien ohjelmistotoimittajien kanssa.

Ohjelmat toimivat tehokkaasti

Koska MikroMikko 3 noudattaa kansainvälisiä teollisuusstandardeja, siinä toimivat markkinoilta saatavissa olevat PC/MS-DOS -valmisohjelmat. Ja lisäksi huomattavasti tehokkaammin kuin monissa muissa mikroissa. Tämä on todettu eri testeissä (esim. Mikro 4/86 ja Computer Persönlich 18/86).

On kuitenkin korostettava, että Nokia takaa vain myymiensä ohjelmien toimivuuden huolimatta siitä, että kolmonen on teollisuusstandardin mukai-

nen. Jotkut sovellusohjelmat käyttävät laitteiden erikoispiirteitä hyväkseen poiketen näin standardista. Ostettaessa ohjelmaa muualta kuin Nokialta, on ehdottomasti varmistuttava siitä, että ohjelma toimii nimenomaan MikroMikko 3:n MS-DOSissa.

Kommunikoi muiden koneiden kanssa

Tietoliikenneohjelmien ja tarvittavien I/O-laajennuskorttien avulla MikroMikko 3 saadaan kommunikoidaan muiden koneiden kanssa. Sen tietoliikenneohjelmistoon kuuluvat mm. ASCOM (TTY) ja VT 220 -emulaattorit sekä Honeywell-työasemaohjelma WS 7300. Lisäksi Nokia tukee jatkossa BSC 3270 ja SNA/SDLC 3270 -ohjelmia.

Kolmonen on vielä tämän vuoden lopulla liitettävissä yhdessä MPS 10:ien, MikroMikko 2:ien ja PääteMikkojen kanssa myös Nokian lähiverkkoon. □

Vihjeitä ohjelmien siirtämiseksi MikroMikko 2:sta 3:een

Ohjelmien siirrosta pitäisi useimmissa tapauksissa selvitä alla olevien vihjeiden mukaan. Kaikki MikroMikko 2 -ohjelmat vaativat ainakin kohdan 1 huomioimista. Muita vihjeitä on sovellettava tarpeen mukaan.

1. MikroMikko 3 on asetettava Nokia PC -tilaan Modeset-komennolla ennen sovelluksen käynnistämistä. Komennon voi kirjoittaa esimerkiksi ohjelmaa käynnistävään komentotiedostoon.
2. Keyload-, Softload- ja Linload-ohjelmat on oltava kolmosen omia. Sovelluslevykeille on tarvittaessa kopioitava uudet.
3. Kakkosen Dispload-ohjelman tilalla on käytettävä kolmosen Fontloadia.
4. Jos MikroMikko 2 -sovellus käyttää Keyread.obj- tai Softread.obj-aliohjelmia, ne on vaihdettava kolmosen vastaaviin Keyread.lib- tai Softread.lib-moduleihin linkittämällä nämä sovelluksen objektikoodin kanssa.
5. Kakkosen Dispread-aliohjelmaa ei ole kolmosessa. Dispreadin sijaan on käytettävä kolmosen Fontloadia.
6. Kaikki suorat laiteosoitukset on muutettava lähdekieliseen ohjelmaan.

Windows uudistaa MikroMikon ilmeen



MikroMikko 3:n merkittävimpiä ohjelmistouutuuksia on Microsoft Windows. Se muuttaa olennaisesti MikroMikon käyttäjäliitännän. Näytön ikkunoissa voi käyttää useampia sovelluksia samanaikaisesti. Ja ohjelmien käyttö on yhdenmukaista.

Windows uudistaa myös sovelluskehityksen. Sen grafiikkaliitännä, kirjoitintuki ja sovellusten välinen tiedonsiirto helpottavat sovellusten kehittämistä ja parantavat niiden siirrettävyyttä.

Windows korvaa perinteisen käyttöjärjestelmän komentotason. Kun aiemmin käyttäjä kirjoitti käskyn ja siihen liittyvät parametrit näppäimistöltä, hän voi nyt Windowsia käyttäessään osoittaa toiminnon nimeä tai sitä kuvaavaa symbolia. Tarkentavat lisätiedot voi antaa tilanteen mukaan joko näppäimistöltä tai valitsemalla hiiren avulla tarjolla olevista vaihtoehdoista.

Windowsia ja sen alaisuudessa toimivia sovelluksia voidaan käyttää näppäimistöltä, mutta Windowsin toiminnot ja graafiset sovellukset ovat tehokkaimmillaan hiirtä käytettäessä.

Ikkunat auki ja kiinni

Sovellukset toimivat näytössä erillisissä ikkunoissa, jotka ovat rakenteeltaan yhtenäisiä. Näyttö voidaan jakaa yhteen tai useampaan ikkunaan käyttäjän ja sovellusten tarpeiden mukaan. Ikkunan 'puitteissa' näkyvät palkit osoittavat, mikä alue käsiteltävänä olevasta teksti- tai grafiikkatiedosta on esillä ikkunassa. Ikkunan 'yläkarmissa' on otsikkorivi ja sen alapuolella vaakasuuntainen valikko. Se sisältää luettelon sovelluksen päätoiminnoista. Kun valikon alkiota osoitetaan, sen alapuolelle ilmestyy 'rullaverho' -valikko, joka sisältää osatoiminnot tai toiminnon vaihtoehdot parametreit.

Sovelluksen ikkuna voidaan sulkea siten, että sovellusta kuvaava ikoni jää näytön alareunaan ikonialueelle. Tällöin sovellus jää aktiiviseksi ja sen ikkuna voidaan avata uudelleen siirtämällä ikoni takaisin näytölle.

Nykyiset mikrojen sovellukset on tyypillisesti toteutettu siten, että ne käyttävät koko näyttöruudun aluetta. Myös näiden käynnistäminen Windowsista on mahdollista. Kun sovelluksesta poistutaan, Windows palaa näyttöön.



MS-DOSin tiedostoja käsitellään ja komentoja annetaan Windowsin visuaalisen komentotulkin avulla. MS-DOSin komennot käynnistetään joko kirjoittamalla komento tai osoittamalla komentoa näytöllä olevasta tiedostoluettelosta.

Useita sovelluksia yhtäikää

Windows mahdollistaa usean sovelluksen vuorottaisen käytön. Se sallii usean sovelluksen lataamisen samanaikaisesti keskusmuistiin muistitilan sallimissa rajoissa. Näin sovellukset ovat käytettävissä välittömästi ilman uudelleenlatausta siirryttäessä ikkunasta toiseen.

Ohjelmistokanta uusiutuu suhteellisen hitaasti ja erityisesti Windowsiin perustuvien uusien ohjelmistojen kehittämiseen on varattava aikaa. Niinpä Windowsin kehittämisen keskeisiä tavoitteita on ollut, että sen kanssa voidaan käyttää myös olemassa olevia MS-DOS -sovelluksia.

Vanhat MS-DOS -sovellukset käytettyvät Windowsissa eri tavoin riippuen niiden tavasta käyttää laitteistoa.

Laiteriippuvuus kuvataan ohjelmisto-kohtaisissa PIF-tiedostoissa, joiden perusteella Windows jakaa työmuistin, näytön ja muut laiteresurssit sovellusta käynnistettäessä.

Sovellukset voidaan jakaa seuraaviin toiminnaltaan erilaisiin ryhmiin:

- Windows-sovellukset, jotka on kehitetty käyttäen Windowsin palveluja. Tällaisia sovelluksia ovat mm. Microsoftin Write, Paint sekä In-a-Vision.
- MS-DOS -sovellukset, jotka voidaan sovittaa toimimaan Windowsin ikkunassa, mutta eivät käytä Windowsin valikkoja ja muita toimintoja. Mm. Microsoftin Rbase ja Project voidaan asentaa toimimaan ikkunassa.
- MS-DOS -sovellukset, jotka voidaan käynnistää Windowsista siten, että ne varaavat koko näytön. Tällaisia ohjelmistoja ovat mm. Lotus 1-2-3 ja Symphony. Ohjelmiston käyttö voidaan yleensä keskeyttää ja käydä Windowsin kautta toisessa sovelluksessa.

MikroMikko 3:n korkealuokkainen näyttö soveltuu mainiosti Windowsin käyttöön. Windows on sovitettu MikroMikon näytönohjaimeen siten, että sen suurta erotuskykyä käytetään täysin hyväksi.

- MS-DOS -sovellukset, joiden suorituksen ajaksi Windows poistuu keskusmuistista. Windows latautuu uudelleen levyltä sovelluksesta palatessa.

Sovellusten välillä tietoa siirretään ns. Clipboard-aulun kautta. Osa käsiteltävästä tiedosta, on se sitten tekstiä, grafiikkaa tai muuta informaatiota, voidaan tallettaa tauluun ja palauttaa sieltä vaikkapa toiseen sovellukseen. Tiedonsiirto on sovellusohjelmasta riippumaton toiminto. Parhaiten siirto onnistuu varsinaisten Windows-sovellusten välillä.

Laiteriippumaton ohjelmistoympäristö

Sovellusohjelmien kehittäjän kannalta Windows on laajennus MS-DOS -käyttöjärjestelmään. MS-DOSista puuttuu graafinen näytönkäsittely. Siksi sovellusohjelmissa on otettava huomioon erilaisten näytönohjainten erot. Koska Windows sovitetaan käytettäviin näytönohjaimiin, sovellusohjelmat voivat käyttää hyväkseen Windowsin laitteista riippumatonta näytönkäsittelyä.

Kirjoitintuki on MS-DOS -sovelluksissa näytönkäsittelyn tavoin tulostuslaittekohtainen. Varsinkin tekstinkäsittely- ja grafiikkaohjelmissa erilaisten kirjoittimien ja piirtureiden soveltaminen ohjelmaan on merkittävä osa ohjelmiston kehittämistyöstä. Windowsissa sovellusohjelmat käyttävät yhteistä kirjoitintukea, jolloin sovellusohjelmiin ei sisälly tulostuslaittekohtaisia osia.

Windowsin kirjoitintuki huolehtii myös taustatulostuksesta.

Windowsin laiteriippumaton ohjelmistoympäristö parantaa sovellusten siirrettävyyttä laitteistosta toiseen. Näyttö- ja tulostuslaitteiden ohjauksen toteuttaminen siirtyy ohjelmistovalmistajilta laitevalmistajille. Windowsissa on valmiina tärkeimpien näytönohjainten ja tulostuslaitteiden ohjaukset.

Merkittävä Windowsin etu on yhtenäinen käyttäjäliitäntä. Windowsin palvelujen hyväksikäyttö johtaa sovelluskehityksessä samankaltaisiin toteutuksiin. Sovellusten käyttö on siten samanlaista ohjelmistotoimittajasta riippumatta. Niinpä uuden sovelluksen käyttöönottokynnys on myös entistä matalampi.

Windowsin asema ohjelmistotuotteenä on vahva, sillä sen valmistaja Microsoft on kehittänyt myös MS-DOS -käyttöjärjestelmän ja MS-Net -verkkokäyttöjärjestelmän. Tämä takaa Windowsin jatkuvan kehittymisen laitteistojen ja käyttöjärjestelmän mukana.

Työvälineitä Windows-sovellusten kehittämiseen

Sovellusohjelmien kehitys tapahtuu Windows Toolkit -ohjelmistopakettien avulla. Sitä käyttäen sovelluskehittäjä määrittelee valikot, komentoikkunat, ikonit ja muun sovelluksen ja käyttäjän välisen kommunikoinnin.

Windowsin ohjelmistoliitäntä on erittäin laaja verrattuna esimerkiksi MS-DOSin ohjelmistoliitäntään. Siksi Windows-sovellusten kehitys on vaativaa ja edellyttää perusteellista Windowsin arkkitehtuurin ja Windows Toolkitin hallintaa.

Sovelluskehittäjän kannattaa käyttää Windowsia mm. seuraavissa tilanteissa:

- Sovelluksessa tarvitaan kehittynyt graafinen käyttäjäliitäntä.
- Sovelluksen on tuettava laajaa ja uudistuvaa tulostuslaitteivalikoimaa.
- Sovellus muodostuu itsenäisistä kokonaisuuksista, joiden käyttö on vuorottaista ja joiden välille on toteutettava tiedonsiirto.
- Sovellus sisältää vuorovaikutteista grafiikan luontia tai tekstin muokkauksia.

Sovellusvalikoima laajenee

Windowsin mukana toimitetaan jo useita henkilökohtaista työskentelyä helpottavia ohjelmia, joista tärkeimpiä ovat:

- Write, tekstinkäsittely
- Paint, piirto-ohjelma
- Calendar, kalenteriohjelma
- Cardfile, kortisto-ohjelma
- Notepad, muistiinpano-ohjelma
- Calculator, laskin
- Clock, kello
- Terminal, pääte-emulaattori

Windows-sovellusten valikoima on laajenemassa nopeasti. Grafiikkaan perustuvat CAD- ja muut piirto-ohjelmistot ovat tyypillinen sovellusalue. Windows soveltuu myös esimerkiksi tietoliikenneohjelmistoihin. Tällöin pääteyhteys on käytettävissä samanaikaisesti työaseman paikallistoimintojen kanssa.

Useat johtavat ohjelmistovalmistajat, kuten Lotus ja Ashton-Tate, ovat jo ilmoittaneet kehittävänsä uusia ohjelmistotuotteita Windowsiin.

Nokia seuraa kiinteästi ohjelmistotarjonnan kehittymistä Windows-ympäristöön ja hankkii keskeiset ohjelmistotuotteet valikoimaansa. Windowsin tarvitsemat ohjaustiedostot sisältyvät jo nyt mahdollisimman valmiina Nokian toimittamiin MS-DOS -ohjelmistoihin. □

Projektin suunnitteluun ja ohjaukseen tarkoitettuja ATK-järjestelmiä on jo jonkin aikaa ollut saatavana isoille tietokoneille. Kalliin hintansa ja hankaluutensa takia niihin on kuitenkin siirrytty vasta, kun projektin koko on noussut markoissa mitattuna kymmeneen miljooniin.

Mikrotietokoneisiin tehdyt ohjelmistot laskevat kynnystä huomattavasti. Hyvän mikrossa toimivan projektiohjelman saa tänä päivänä jo noin viidellä tuhannella markalla. Mikroihin tehdyt ohjelmat ovat lisäksi helpokäyttöisempiä kuin kalliimmat esikuvansa.

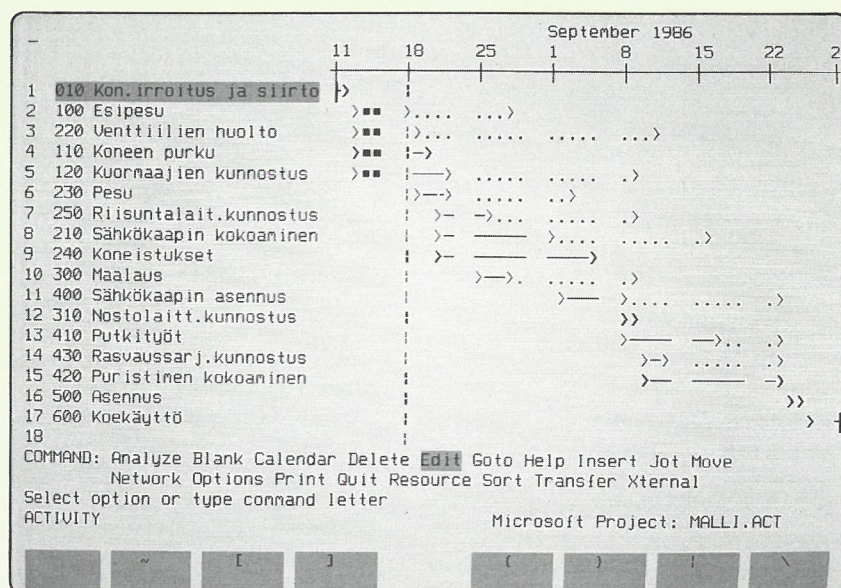
Mihin projektipäällikkö tietokonetta sitten tarvitsee?

On muistettava, että tietokoneohjelma on vain apuväline; se ei osaa laatia projektisuunnitelmaa itse. Yleinen sääntö onkin: Mikä ei toimi käsin, ei toimi myöskään koneella. Hyvä projektinhallintaohjelma helpottaa kuitenkin ratkaisevasti toimintaverkon piirtämistä, resurssien hallintaa ja kustannusarvion laatimista.

Yksinkertaisimmat projektiohjelmat ovat puhtaita aikatauluohjelmia. Niillä voidaan laatia toimintaverkko, josta tehtävien kestot ja riippuvuudet käyvät ilmi. Kehittyneemmissä ohjelmissa on mukana resurssien ja kustannusten hallinta sekä yksilöllisten kalenterien käyttö. Täysi hyöty ohjelmasta saadaan, jos se palvelee käyttäjäänsä koko projektin ajan, eli jos sitä voidaan käyttää myös projektin seurantaan ja ohjaukseen.

Mikro projektipäällikön työkaluna

Microsoft Project toimii 768 kilotavun keskusmuistilla varustetussa MikroMikko 2:ssa sekä kaikissa MikroMikko 3 -malleissa.



Kuva 1. Tehtävä- eli aktiviteettinäyttö

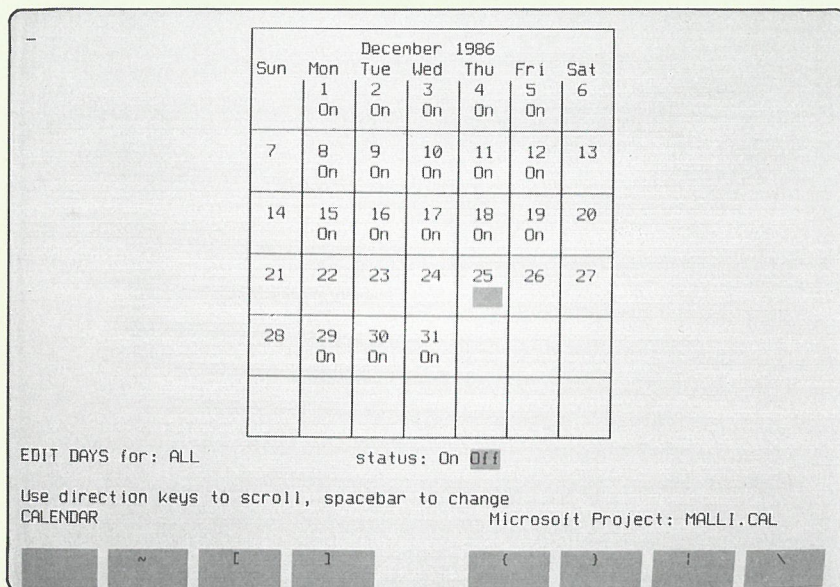
MS-Project yksi parhaista

Microsoftin tekemä Project on mikromarkkinoiden parhaita projektinhallintaohjelmia. Se on hinnaltaan edullinen mutta silti monipuolinen projektipäällikön työkalu. MS-Projectin versiossa 2.0 on useita hyviä ominaisuuksia, joita monesta paljon kalliimmasta ohjelmasta saa turhaan hakea.

MS-Project jakautuu toiminnallisesti kolmeen osaan: tehtävät, kalenterit ja

resurssit. Kun ohjelma käynnistetään, ruudulle ilmestyy tehtävänäyttö (kuva 1). Siinä projektin tehtävät näkyvät aika-akselille piirrettyinä janoina.

MS-Projectin käyttö on helppo oppia. Kaikki komennot näkyvät selväkielisinä näytön alalaidassa. Valinta tehdään painamalla komennon alkukirjainta tai siirtämällä kohdistin halutun komennon päälle ja painamalla kuitausnäppäintä. Esimerkiksi Multiplanin käyttäjä oppii MS-Projectin hetkessä.



Kuva 2. Kalenterinäyttö

Resource	Capacity	Unit Cost	Per	Days To Complete	Cost To Complete
1 Koneasentaja	3,0	53,00	Hour	55,0	23338,55
2 Apunies	2,0	35,00	Hour	33,7	9443,00
3 Putkiasentaja	2,0	50,00	Hour	17,6	7072,50
4 Trukki	1,0	1000,00	Day	0,0	0,00
5 Sähköasentaja	2,0	55,00	Hour	18,0	7920,00
6 Koneenhoitaja	1,0	37,00	Hour	2,0	592,00
7					
8					
9					
10					
11					
12					
13					
14					
15					

Cost to complete: 48366,05 Total cost of project: 61328,00

COMMAND: Activity Blank Calendar Delete Edit Goto Help Insert
Move Options Print Quit Sort Transfer View

Resource capacity exceeded by 2

Microsoft Project: MALLI.RES

Kuva 3. Resurssinäyttö

Sisäänrakennettu aputoiminto opastaa tarvittaessa käyttäjää ohjelman kaikissa vaiheissa.

Yksilölliset kalenterit

MS-Projectia käytettäessä on yleensä parasta aloittaa kalenterin määrittämisellä. Vain näppäimen painalluksella siirrytään tehtävänäytöstä kalenterinäyttöön (kuva 2).

Tehtävien kestot voidaan mitata jo-

ko työaikana tai jatkuvana aikana. Työajaksi lasketaan normaalisti viisi kahdeksantuntista työpäivää viikossa. Poikkeamat ilmoitetaan merkitsemällä ylimääräiset vapaa- tai työpäivät kalenteriin. Päiväkohtainen työaika voidaan merkitä jopa viiden minuutin tarkkuudella.

Eri resursseille voidaan haluttaessa laatia omat kalenterit. Tästä on hyötyä varsinkin lomakauteen, esim. kesään, ajoittuvia hankkeita suunniteltaessa.

Tehtävistä kriittinen polku

Kun kalenterien määrittäminen on valmis, siirrytään takaisin tehtävä- eli aktiviteetinäyttöön (kuva 1). Ohjelmalle syötetään tehtävien nimet, kestot sekä mahdolliset kiinteät aloitus- ja lopetuspäivät. Samalla ilmoitetaan tehtävien riippuvuussuhteet, eli mitkä ovat tietyn tehtävän edeltäjiä. Näistä tiedosta MS-Project piirtää heti janakaavion aikakselille.

Kaaviosta nähdään ns. kriittinen polku. Kriittiset tehtävät ovat sellaisia, joiden viivästyminen aiheuttaa koko projektin valmistumispäivän siirtymisen.

Tehtävänäytön aikaskaalaa voidaan muuttaa yhdellä näppäimen painalluksella. Tarkkuus voi vaihdella minuuteista puoleen vuoteen.

Tiedot projektin tehtävistä syötetään halutussa järjestyksessä. Jälkeenpäin järjestystä voidaan muuttaa joko käsin tai MS-Projectin monipuolisia lajittelumahdollisuuksia hyväksi käyttäen.

Jokaiselle tehtävälle määrätään myös sen tarvitsemat henkilö-, kone-, aine- ym. resurssit. Resurssit voidaan valita valmiista resurssiluettelosta. Jos kyseessä on uusi projekti, ei resurssiluetteloa vielä ole. Tällöin resurssit on nimettävä itse ja ohjelma muodostaa resurssiluettelon automaattisesti.

Resurssien käyttö optimaaliseksi

Resurssinäytössä (kuva 3) voidaan resurssiluettelon tietoja muokata yksityiskohtaisesti. Siinä ilmoitetaan käytettävissä olevien voimavarojen määrät.

Ohjelmalle voidaan syöttää resurssin käytöstä aiheutuvat yksikkökustannukset, jolloin MS-Project laskee projektin kokonaiskustannukset.

Resurssinäytön View-komento näyttää pylväskuviona voimavarojen kuormituksen ja varsinkin ne ajanjaksot, joina kapasiteetti ylittyy. Tehtävien ajoitusta muuttamalla projektipääallikkö voi tasata resurssien käyttöä. Ohjelma ei tee automaattista resurssitasasta.

Monipuoliset raportit

Aikataulujen ja resurssien laskenta ei yksinään riitä. Ohjelman on pystyttävä tulostamaan aikataulut, kustannuslaskelmat, tehtäväluettelot jne. myös paperille.

MS-Projectilla voidaan tulostaa 16 erilaista raporttia. Jokaista raporttityyp-
piä voidaan lisäksi muokata ennen tu-
lostusta. Esimerkiksi otsikoita voidaan
vapaasti muuttaa ja raporttiin voidaan
valita tulostettavaksi vain tietty osa tie-
doista. Graafiikkakirjoitinta käytettäes-
sä aikakaavio ja looginen toimintaverk-
ko saadaan ketjulomakkeelle sivusuun-
nassa tulostettuna lakanana.

Raportit voidaan tulostaa paitsi näy-
tölle tai kirjoittimelle myös tiedostoon,
josta ne ovat esimerkiksi tekstinkäsitte-
lyohjelmalla luettavissa ja muokattavis-
sa.

MS-Project tarjoaa myös yhteenso-
pivuuden muihin ohjelmiin. Kassavirta-
laskelmat ja resurssikustannukset voi-
daan siirtää suoraan taulukkolaskenta-
ohjelman (Multiplan, Lotus 1-2-3) käsi-
teltäviksi.

Projektin seurannan apuväline

MS-Project on sekä projektin suunnit-
telun että sen seurannan apuväline.

Ensimmäinen hyväksytty suunnitel-
ma voidaan tallettaa erikseen ns. en-
nustetiedostoon. Projektin edetessä tie-
toja ylläpidetään aktiivisessa projektitie-
dostossa. Tärkeintä on merkitä aika-
taulujen ja kustannusten poikkeamat.

Talletettu ennuste voidaan ottaa
näyttöön aktiivisen projektin rinnalle.
Viivästymiset ja niiden vaikutukset ai-
katauluun ovat tällöin selvästi nähtävis-
sä. Erojen havaitsemiseen on myös
valittavissa omat raportit.

Suuret projektit hallintaan

Eräs MS-Projectin parhaita ominai-
suuksia on sen kyky käsitellä suuria
projekteja ja useita toisistaan riippuvia
osaprojekteja.

Suuret projektit on kokonaisuuden
hallitsemiseksi syytä jakaa osaprojek-
teiksi. Pääprojektissa nämä osaprojektit
voidaan ottaa mukaan tehtävinä.
Osaprojektin alku- ja loppuajankohdat
sekä resurssien käyttö välittyy pääpro-
jektille.

Usein projektissa työskentelevät
henkilöt ovat samanaikaisesti monen-
kin eri hankkeen kimpussa. Tällaisen
tilanteen hallitseminen esimerkiksi tuo-
tekehitysosastolla on perinteisesti vai-
keata. MS-Projectilla voidaan erilliset
projektit koota yhteen kattoprojektiin.
Sen avulla voidaan seurata osaston
resurssien kuormitusta ja aikatauluja. □

Tiedonhallintaohjelmat testissä:

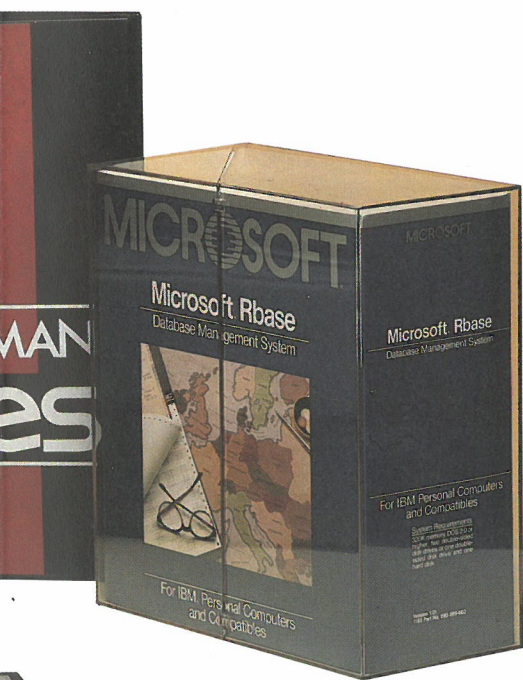
Eri ohjelmat eri tar-



*Mikroihin löytyy markkinoilta runsaasti erimerkkisiä tie-
donhallintaohjelmistoja, jotka soveltuvat erilaisiin käyttö-
tarkoituksiin. Mikä mihinkin? Me otimme selvää.*

*Rbase sopii parhaiten sovelluskehitykseen. Samoin
PC-INFO ja PC/FOCUS. dBASE II ja K-mies ovat
molemmat vanhentuneita, mutta niiden uusimmat ver-
siot dBASE III PLUS ja K-mies/2 ovat tehokkaita
loppukäyttäjien välineitä. dBASE III PLUS on oikea
valinta, jos tärkeimpänä kriteerinä on nopea tietueen eli
kortin haku.*

rkoituksiin



Nokia Informaatiojärjestelmissä käynnistettiin viime keväänä projekti, jonka tehtävänä oli suosituimpien mikroissa käytettävien tiedonhallintajärjestelmien vertailu. Vertailtavat tuotteet olivat: dBASE II, dBASE III PLUS, K-mies, PC/FOCUS, PC-INFO ja Rbase. Työryhmän ensisijaisena tavoitteena oli tuoda esiin ohjelmistojen eroavuudet.

Testit tehtiin niillä kehitysvälineillä, jotka ohjelmiston ostaja saa toimituspaketissa. Mitään ulkoapäin saatuja kääntäjiä tai muita apuvälineitä ei käytetty.

Oikea ohjelma oikeaan käyttöön

Oikean ohjelmiston valinta vaikuttaa tiedonhallinnan onnistumiseen. Se ei kuitenkaan takaa onnistumista. Suurin vaikutus on sillä, miten hyvin tietokanta on suunniteltu. Hyväkään tiedonhallintaohjelmisto ei pysty korjaamaan tietokannan suunnittelussa tehtyjä virheitä. Suunnittelun onnistumisen tärkeys korostuu isoilla tietokannoilla ja laajoilla sovelluksilla.

Vertailluista ohjelmistoista Rbase, PC-INFO ja PC/FOCUS ovat selvästi sovelluskehittäjän välineitä. PC/FOCUS on parhaimillaan suuria tietokantoja luotaessa. Nopeimmin sovelluksen tekee Rbasella. Rbaseen perehtyminen sen tarjoamilla apuvälineillä käy tiedonhallintaa tuntemattomaltakin ATK-alan ihmiseltä nopeasti. Unohtaa ei myöskään sovi Rbasen tarjoamia liitännöitä korkean tason ohjelmointikieliin. Ne ovat sovelluskehityksessä suuri etu.

dBASE II ja K-mies on tarkoitettu loppukäyttäjän välineiksi, mutta niiden käyttö on hankalaa. Loppukäyttäjä voi molemmilla tehdä omia sovelluksiaan, mutta niitä pystyy käyttämään myös sellaisinaan. Kovin isojen tietokantojen rakentamiseen eivät K-mies ja dBASE II kuitenkaan käy. Lisäksi molemmissa on ongelmana, että loppukäyttäjä joutuu kirjoittamaan komentonsa ohjelman komentokielellä.

dBASE II:n ja K-miehen uusissa versioissa, dBASE III PLUSissa ja K-mies/2:ssa on toimintoja nopeutettu ja ne soveltuvat paremmin loppukäyttäjille. Kummassakin on käyttäjäystävällisyyttä lisätty kehittämällä käyttäjän avuksi valikot komentojen antamiseen. Tällöin varsinaisen komentokielen käyttö ei ole enää välttämätöntä.

Suuren kortiston rakentamiseen sopii parhaiten dBASE III PLUS. Siinä haku avaintekstien mukaan käy nopeasti. Suoritus kestää vain muutamia sekunteja, vaikka tietokanta koostuisi tuhansista tietueista.

Relaatiomalli yleisin

Vertailluista ohjelmistoista PC/FOCUS perustuu hierarkkiseen malliin, muut relaatiomalliin.

Vertailluvien ohjelmistojen keskusmuisti- ja umpilevyvaatimukset on esitetty taulukossa 1. Tietokannan kokoa ei rajoita muu kuin käytettävissä oleva levytila. Lisäksi rajoittavia tekijöitä tie-

tokannan rakentamisessa voivat olla merkkien sekä kenttien määrät tietuetta kohden (Taulukko 2).

Kaikista tutkittavista tiedonhallintajärjestelmistä löytyvät seuraavat tietotyypit: kokonaisluku, liukuluku, teksti. Ja 'päiväys' puuttuu vain dBASE II:sta. Kaikissa muissa paitsi dBASEissa ja PC-INFOssa on myös tietotyyppi 'raha'.

'Looginen'-tyyppi on ainoastaan dBASEissa. dBASE III PLUSissa on vielä 'muistio'. 'Muistio'-tyyppiin voi tietokannan rakentaja kirjoittaa tietuekohtaisesti 0 - 4 kilotavua vapaamuotoista tekstiä.

Komentokieli yleisin käyttäjälitettä

Ohjelmistot ovat englanninkielisiä lukuunottamatta K-miestä, josta on saatavissa suomenkielinen versio. K-miehen lisäksi PC-INFO:n käyttöoppaat on suomennettu.

dBASE II:n ja K-miehen toiminnot käynnistetään kirjoittamalla komennot.

dBASE III PLUSissa on valikko-ohjattu käyttäjälitettä, jolloin käyttäjä voi valita valikosta haluamansa toiminnon. Halutessaan käyttäjä voi antaa myös yksittäisiä komentoja komentotilassa. K-miehen uudessa versiossa K-mies/2:ssa on myös valikko-ohjattu käyttäjälitettä.

Rbasessa, PC/FOCUSissa ja PC-INFOssa loppukäyttäjän käyttäjälitettä riippuu sovelluksen tekijän ratkaisuista. Kaikissa niissä on kuitenkin välineet valikkojen tekemiseen.

Rbasen mukana toimitettavalla Express-ohjelmalla voi tietokantajärjestelmän rakentaja nopeasti luoda tietokannan ja sovelluksen rungon. Muut toiminnot, kuten tiedon syöttö ja kyselyt, on toteutettava Rbasen komentokielellä. Rbase-sovellus tulee tehdä siten, ettei varsinaisen loppukäyttäjän tarvitse käyttää Rbasen kehitysympäristöä.

Sovelluskehitys nopeaa Rbasella

Rbase, PC-INFO ja PC/FOCUS on suunniteltu nimenomaan sovelluskehittäjän välineiksi.

PC/FOCUS vaatii hierarkkisen tiedostorakenteensa takia eniten suunnittelua tietokannan rakentamisessa. Samasta syystä tietokannan rakenteen muuttaminen PC/FOCUSella on vaikeaa.

Rbaseen on saatavilla korkean tason ohjelmointikielten liitännä, jota käyttäen Pascal-, FORTRAN- ja C-kielisistä ohjelmista voidaan käsitellä Rbase-tietokantoja. Vastaavasti K-miehessä on BASIC-ohjelmointikielen ja dBASE III PLUSassa C-kielen liitännä. Muilla vertailuilla ohjelmistoilla on sovelluskehitys tehtävä yksinomaan järjestelmän oman komentokielen avulla.

Rbasen Express-ohjelmalla sovellusten kehitys käy nopeasti. Expressillä rakennetaan sovellukseen peräkkäiset valikot sekä määritetään kunkin valinnan toiminnot. 'Apu'-toimintoa ei tarvitse sovellukseen kehittää erikseen, vaan sen voi rakentaa Expressin avulla samanaikaisesti sovelluksen kanssa.

Lukuisia lisäohjelmia

PC/FOCUSessa on runsaasti lisäpiirteitä. Grafiikalla voidaan esittää kyselyjen tuloksia havainnollisessa muodossa. PC/FOCUS sisältää useita tilastanalyysissä tarvittavia funktioita. Käytettävissä ovat regressio-, varianssi-, diskriminantti- ja faktorianalyysi sekä aikasarjat ja korrelaatiot.

K-mies-pakettiin sisältyy taulukkolaskenta. Lisäksi siihen on erillisenä saatavana mm. grafiikka- ja tekstinkäsittelyohjelma.

PC-INFOon on saatavissa Infograf-graafiikkaohjelma. Sitä voi käyttää tietokannan graafiseen esittämiseen.

Rbasen korkean tason ohjelmointikielten liitännäpaketti myydään erillisenä tuotteena. Muita lisäohjelmia ei Rbaseen ole saatavissa.

dBASE II:een sekä III PLUSaan löytyy ulkopuolisten tekemät kääntäjät. dBASEn valmistaja Ashton-Tate ei kuitenkaan anna niille mitään tukea. Sen sijaan se myy ja tukee C-kielen liitännäpakettia dBASE III PLUSaan.

Taulukko 1. Ohjelmiston keskusmuisti- ja umpilevyvaatimukset

Ohjelmisto	Keskusmuisti (min)	Umpilevy (min)
RBASE	768 ktavua	5 Mtavua
Kmies	256 ktavua	Ei *)
dBASE II	256 ktavua	Ei *)
dBASE III PLUS	256 ktavua	Ei *)
PC-INFO	370 ktavua	5 Mtavua
PC-FOCUS	768 ktavua	5 Mtavua

*) Ohjelma toimii myös kahden levykeaseman koneessa. Tällöin tietokannan koko ei voi kasvaa kovin suureksi.

Taulukko 2. Merkkien ja kenttien enimmäismäärät tietuetta kohden.

Ohjelmisto	Merkkejä/Tietue	Kenttiä/Tietue
RBASE	1 530	400
Kmies	65 534	255
dBASE II	1 000	32
dBASE III PLUS	2 000	128
PC-INFO	4 064	4 064
PC-FOCUS	4 096	250

Taulukko 3. Testitulokset sadalla tietueella (MikroMikko 2).

	K-mies	dBASE II	PC-INFO	Rbase	PC/FOCUS
Päivitys	2	7	10	3	32
Avainhaku	12	3	14	8	17
Haku	12	18	14	9	18
Poisto	27	26	33	10	13
Lajittelu	41	155	66	8	*)
Avainlajittelu	44	146	65	8	*)
Muutos	39	27	114	28	*)
Ascii-luku	26	10	39	19	39
yht.	203	392	355	93	

*) Ei ollut toteutettavissa järkevällä työmäärällä.

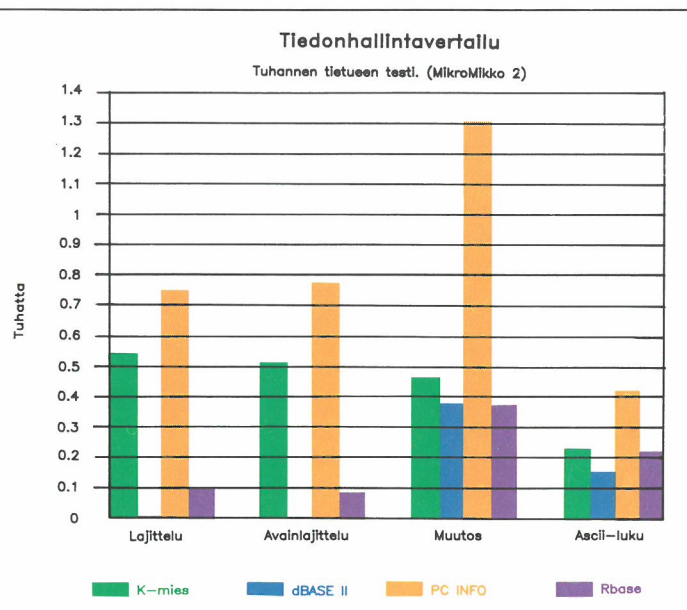
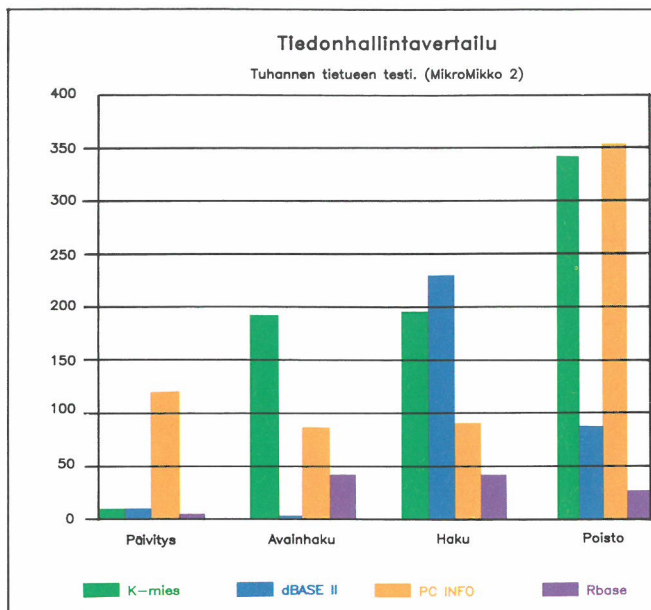
Taulukko 4. Testitulokset tuhannella tietueella (MikroMikko 2).

	K-mies	dBASE II	PC-INFO	Rbase
Päivitys	11	11	121	6
Avainhaku	192	3	88	42
Haku	197	232	92	42
Poisto	346	86	353	26
Lajittelu	542	*)	760	96
Avainlajittelu	522	*)	792	95
Muutos	467	382	1 308	389
Ascii-luku	222	148	430	211
yht.	2 499		3 944	907

*) Keskeytettiin kahden tunnin ajon jälkeen.

Taulukko 5. MikroMikko 3:lla tehtyt testit.

	Sadalla tietueella		Tuhannella tietueella	
	Rbase	dBASE III PLUS	Rbase	dBASE III PLUS
Päivitys	1	1	2	3
Avainhaku	5	1	30	2
Haku	5	9	29	16
Poisto	2	6	13	43
Lajittelu	7	23	84	224
Avainlajittelu	7	25	83	250
Muutos	14	34	182	314
Ascii-luku	7	5	88	35
yht.	48	104	511	887



Suorituskykytestit MikroMikko 2:lla

Vertailtavien ohjelmien suorituskykyä testattiin MikroMikko 2:lla, jossa oli 768 kilotavun keskusmuisti ja 20 megatavun umpilevy. Lisätestejä tehtiin MikroMikko 3:n kanssa. Tuotteista ainoastaan dBASE III PLUS on sellainen, joka ei toimi MikroMikko 2:ssa. Sitä verrattiin Rbasen kanssa MikroMikko 3:ssa.

Testejä varten rakennettiin jokaisella ohjelmistolla sama tietokanta. Samat tiedot luettiin ulkopuolisista tiedostoista jokaiseen tietokantaan.

Testejä suoritettiin sadan ja tuhannen tietueen tiedostoilla. Tietueet olivat tuhannen merkin pituisia.

Testejä oli kahdeksan:

Päivitys: Tietueiden kenttää päivitettiin ehdon mukaan.

Avainhaku: Tietuetta haettiin avainkentän mukaan.

Haku: Tietuetta haettiin muun kuin avainkentän mukaan.

Poisto: Tietueita poistettiin tietokannasta.

Lajittelu: Tiedosto lajiteltiin muun kuin avainkentän mukaan.

Avainlajittelu: Tiedosto lajiteltiin avainkentän mukaan.

Muutos: Tiedoston rakennetta muutettiin, kun tietueet oli jo luettu.

Ascii-luku: Siirrettiin testiaineisto ASCII-tiedostosta tietokantaan.

Rbase

Rbase on amerikkalaisesta R:Base 5000:sta kehitetty kansainvälinen versio, joka julkistettiin syksyllä 1985. Rbase toimii sekä PC-DOS että MS-DOS -käyttöjärjestelmillä.

Rbasen on valmistanut amerikkalainen Microrim Inc. Ohjelmisto kuuluu Microsoftin tuotevalikoimaan, sillä se on ostanut yksinoikeudet ohjelmistoon.

dBASE II

dBASE II on ensimmäisiä tietokantaohjelmistoja mikroille. Ja se on ollut yksi myydyimpiä mikrojen tiedonhallintaohjelmistoja.

Ensimmäinen dBASE II -versio kehitettiin 8080, 8085 ja Z-80 -pohjaisille mikroille ja se toimii käyttöjärjestelmissä CPM/M Z.X, CDOS tai CROMIX. Toinen versio ohjelmistosta on tehty 8088 ja 8086-pohjaisille mikroille ja sen käyttöjärjestelmävaihtoehdot ovat CP/M-86 ja MS-DOS.

dBASE II -ohjelmiston valmistaja on amerikkalainen Ashton-Tate.

dBASE III PLUS

dBASE III on dBASE II -ohjelmasta 16-bittisille mikroille kehitetty versio. dBASE III:sta on edelleen kehitetty tehokkaampi dBASE III PLUS.

dBASE III PLUSan valmistaja on Ashton-Tate.

PC/FOCUS

FOCUS on alunperin tehty suuriin koneisiin, sovitettu sitten minikoneisiin ja lopulta mikroihin. FOCUSen keskuskoneversio julkistettiin 1976 ja mikroversio PC/FOCUS 1983. Mikroissa käyttöjärjestelmänä on MS-DOS tai PC-DOS.

PC/FOCUSen valmistaja on amerikkalainen Information Builders Inc.

K-mies

K-mies, alkuperäiseltä nimeltään Knowledgeman, toimii vain 16-bittisissä mikroissa CP/M-86, PC-DOS tai MS-DOS -käyttöjärjestelmillä. Ohjelmiston tehokkuutta voidaan parantaa käytettävissä olevan muistikon mukaan.

K-miehen on valmistanut amerikkalainen Micro Data Base Systems Inc. Ohjelmiston suomentamisen on puolestaan tehnyt Tietotehdas.

PC-INFO

INFO on alunperin DPS 6 ja Vax-minikoneille kehitetty sovelluskehittäjä. Vuoden 1985 alusta INFOsta on ollut saatavana myös PC-versio, PC-INFO. Se on yhteensopiva INFO:n kanssa eli INFOlla tehdyt tiedostot ja ohjelmat ovat suoraan siirrettävissä mineiltä mikroon ja päinvastoin.

INFO:n valmistaja on amerikkalainen Henco Software Inc.

Testattavat tuotteet olivat toimintavarmoja. Ainoastaan dBASE II suoritti yhden käskyn epätäydellisesti. Luettassa tietueita ASCII-tiedostosta dBASE II-tiedostoon jäi sadan tietueen testissä neljä ja tuhannen tietueen testissä yksi lukematta. Ohjelma ei antanut tästä edes virheilmoitusta.

Testit ja tietokanta oli suunniteltu enemmän relaatiotietokantojen testaamiseen. Siksi hierarkisella PC/FOCUSella ei kaikkia testejä pystytty suorittamaan ja se oli mukana ainoastaan sadan tietueen testeissä. Isoille tietokannoille tarkoitettu PC/FOCUS ei siten päässyt oikeuksiinsa näissä testeissä. Se on maahantuojan mukaan suhteellisesti huomattavasti nopeampi isoilla tietokannoilla.

Mitkä olivat testien tulokset?

Testeissä havaittiin, että jokaisella ohjelmalla oli omat vahvat ja heikot puolensa. Ainoastaan Rbase oli sellainen, joka oli tasaisen vahva jokaisessa testissä. Se oli jokaisessa testissä joko nopein tai toiseksi nopein. Ylivoimainen se oli lajitteluissa sekä tietueiden poistamisessa tietokannasta.

Tietueiden päivitys kävi K-miehellä nopeasti. Muuten K-miehen suoritusajaka sadan tietueen tiedoston käsittelyssä oli muihin verrattuna keskimääräinen. Sitä vastoin tuhannen tietueen tiedostoilla suoritusajat olivat pitkiä muissa kuin päivityksessä.

dBASE II oli ehdottomasti nopein avainkentän mukaan tehdyssä tietuehaussa. Myös muutosten tekeminen tietokantaan kävi sillä nopeasti. Sen sijaan sovelluksissa usein tarvittavat lajittelut, joko avain- tai muun kentän mukaan, olivat dBASE II:lla todella hitaat. Tuhannen tietueen tiedostossa eivät dBASE II:lla onnistuneet lajittelut ollenkaan. Testit keskeytettiin kahden tunnin ajan jälkeen.

PC-INFO osoittautui testeissä varsin hitaaksi. Ainoastaan haut tiedostoista sujuivat siedettävän nopeasti. Varsinkin muutosten tekeminen tietokantaan on PC-INFOLLA muita selvästi hitaampaa.

dBASE III PLUSassa on avainkentän mukaan tapahtuva haku todella nopeaa. Sen sijaan lajittelun verkkaisuus osoittautui suurimmaksi ongelmaksi aivan kuten dBASE II:ssa. □

Tietoa tietorakenteista

Tiedonhallintajärjestelmät voidaan jakaa tietorakenteidensa perusteella relaatio-, hierarkia- ja verkkomalleihin.

Relaatiomallinen tietokanta koostuu relaatioista. Relaatio on joukko tietueita omassa tiedostossaan. Tietue on se kokonaisuus, jota voidaan verrata manuaalisen kortiston yhteen korttiin. Kortin eri tiedot vastaavat relaation attribuutti- eli ominaisuuskenttiä. Se kenttä tai ne kentät, jotka erottavat tietueen muista tietueista, kutsutaan avainkentiksi. Eri relaatioiden tietoja voidaan yhdistellä yhteisten kenttien avulla.

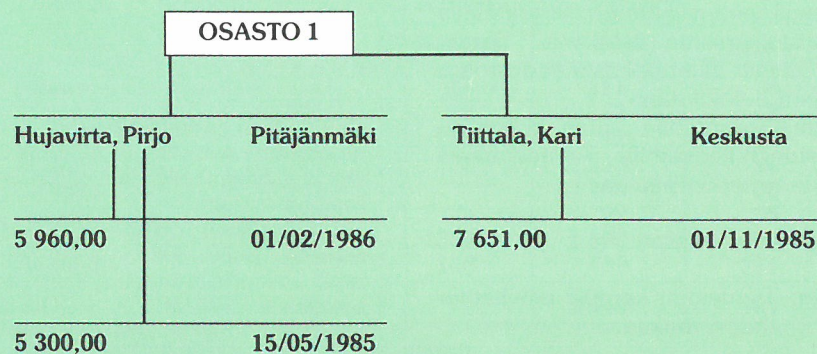
Esimerkiksi yrityksen henkilöstötiedot voitaisiin esittää kahdella relaatiolla:

HENKILÖ-relaatio		
henk.num.	nimi	osoite
29	Hujavirta, Pirjo	Pitäjänmäki
61	Tiittala, Kari	Keskusta
481	Ketomaa, Päivi	Tapiola

PALKKA-relaatio		
henk.num.	palkka	muutettu
29	5 960,00	01/02/1986
61	7 651,00	01/11/1985
481	9 322,00	10/05/1986

Henkilö-relaation kentät ovat henk.num., nimi ja osoite. Relaatio koostuu kolmesta tietueesta. Sekä henkilö- että palkka-relaatioiden avain on henk.num. Relaatioiden tietoja voidaan yhdistää, koska niillä on sama kenttä: henk.num.

Hierarkkisessa mallissa tietueet on tallennettu erilleen toisista. Tietueiden välille muodostetaan yhteyksiä siten, että muodostuu puumainen rakenne. Se edellyttää, että jokaiseen tietueeseen tulee ainoastaan yksi yhteys, paitsi juuri- eli lähtötason tietueisiin. Tietueesta voi lähteä useampia yhteyksiä. Tietueilla on eri tasoilla eri rakenteet.



Esimerkissä hierarkkisuus on yksinkertaista. Siinä on esitetty toiselle henkilölle myös hänen aiemmat palkkatietonsa. Kenttä henk.num. ei enää ole tarpeellinen, koska tietueiden välinen yhteys esitetään tietokannan rakenteessa.

Verkkomalli on muuten samanlainen kuin hierarkkinen, mutta siinä tietueiden väliset yhteydet voivat mennä ristiin. Yhteen tietueeseen voi tulla useita ja siitä voi lähteä useita yhteyksiä. Tiedoston rakenne ei pysy enää selkeän puumaisena, vaan syntyy yhteyksien verkko. Verkkomallia käytetään enemmän isojen tietokoneiden tietokantojen suunnittelussa.

Aiemmissa Softareissa olemme tarkastelleet Adan historiaa ja sen nykyistä käyttölaajuutta (Softari 1/86), sekä Adan ominaisuuksia ja vaikutuksia ohjelmistotyön laatuun (Softari 2/86).

ADA

ohjelmointiprojektin avuksi

Adaa, siihen liittyviä menetelmiä ja sen ohjelmakehitysympäristöä voidaan hyödyntää koko ohjelmointiprojektin ajan. Mutta miten? Ja mitkä ovat Adan, tai laajemmin Ada software engineering -ajattelun, vaikutukset projektin eri vaiheisiin?

Ohjelmointiprojektin vaiheet ovat määrittely, suunnittelu, ohjelmointi, testaus, käyttö ja ylläpito, jotka seuraavat toinen toistaan. Virheitä havaittaessa on aina tietysti palattava takaisin siihen vaiheeseen, missä virhe tehtiin ja korjattava se.

Ada määrittelyvaiheessa

Määrittelyvaiheen päätavoite on järjestelmän toiminnallisten ominaisuuksien määrittäminen ohjelmiston toimittajan ja asiakkaan yhteistyönä. Ohjelmiston määrittely tulisi tehdä siten, että ei kiinnitetä toteutustekniikoita, vaan määrittellään asiakkaan tarpeet ja dokumentoidaan ne.

Ohjelmointikieli, myös Ada, on ensisijaisesti ongelman ratkaisun liittyvä väline. Niinpä Adastakaan ei ole suurta

apua vielä määrittelyvaiheessa.

Koska Ada on laajat mahdollisuudet tarjoava yleiskäyttöinen ohjelmointikieli ja mahdollistaa siten erityyppisten ohjelmistojen toteuttamisen, määrittelyvaihe ei vielä sulje pois Adan käyttöä ohjelmiston toteutukseen. "Adalla voidaan toteuttaa kaikenlaisia mielihaluja, jos kustannukset ja aikataulut vain sen sallivat", toteaa suunnittelija Mikko Terho Softplanista.

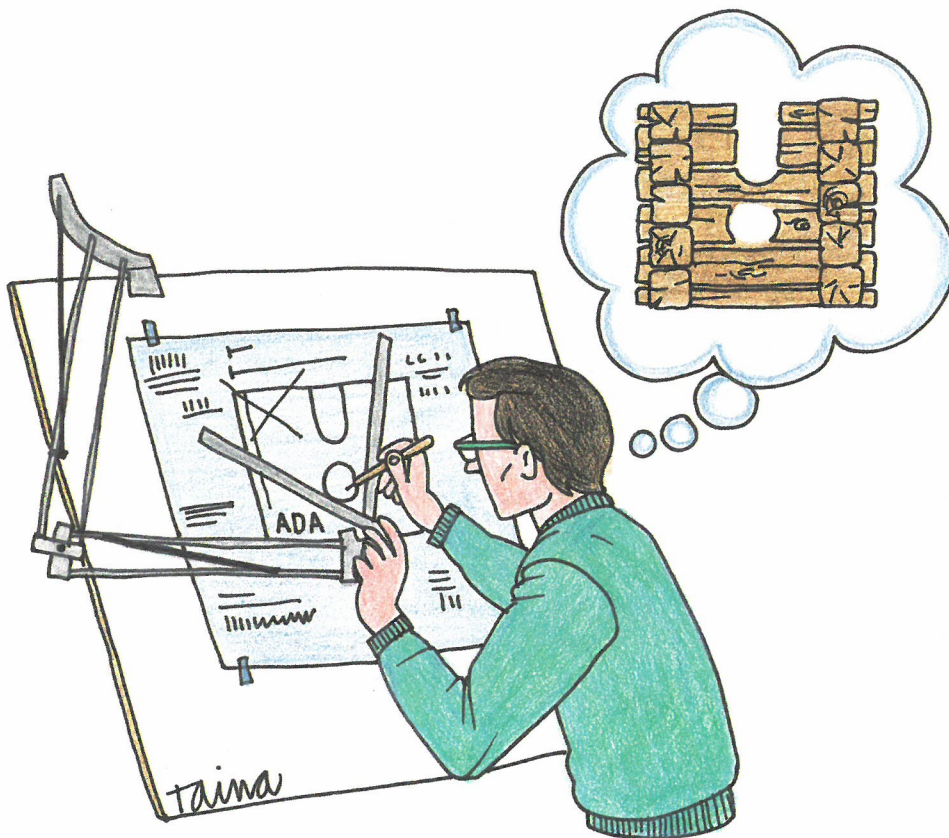
"Eräissä Ada-projekteissa on käytetty olemassa olevia määrittelymenetelmiä ja havaittu niiden sopivuus Adaympäristöön. Softech Yhdysvalloissa on käyttänyt Ada-ohjelmointiympäristönsä määrittelyssä SADT-menetelmää. JSD-menetelmää on kokeiltu Englannin hallituksen projekteissa. Softplan on puolestaan käyttänyt seinätekniikkaa useissa Ada-projekteissaan", kertoo suunnittelupäällikkö Pekka Lahtinen Softplanista.

Ada astuu kuvaan suunnittelussa

Suunnitteluvaiheessa laaditaan tekninen ratkaisu, joka täyttää määrittelyssä asetetut vaatimukset. Suunnittelussa voidaan sellaisenaan käyttää hyväksi Ada-kielen ominaisuuksia sekä ohjelmistoarkkitehtuurin että yksityiskohtaisen suunnittelun yhteydessä.

Ohjelmiston arkkitehtuurin suunnittelu lähtee toiminnallisesta määrittelystä, jossa toiminnot ja tietovarannot mallinnetaan olioiksi. Olioiden välinen tiedonsiirto ja niiden tekemät toimenpiteet muodostavat ohjelmiston.





Ohjelmointi nopeaa Adalla

Ohjelmointivaiheessa laaditaan suunniteltuihin Ada-pakkauksiin yksityiskohtainen koodi, joka toteuttaa pakkauksen määrittelyn. Tai mikä parempaa, valitaan aiemmin tehtyjen ohjelma-komponenttien joukosta valmis määrittelyjen mukainen pakkaus.

Kun pakkausten määrittelyosat on lyöty luokkaan jo suunnitteluvaiheessa, voi vastaavat pakkaurungot ohjelmoida rinnakkaisina itsenäisinä tehtävinä. Näin ohjelmointivaiheessa voidaan käyttää suurtakin projektiryhmää ja siten nopeuttaa projektin etenemistä.

"Ohjelmoinnissa on käytettävissä perinteiset ohjelmointimenetelmät. Esimerkiksi JSP ja jäsentävä strukturoitu ohjelmointi sopivat hyvin yhteen Adan kanssa", Lahtinen kertoo.

Ada-ohjelman kirjoittaminen ja kääntäminen on samanlaista kuin minä tahansa muun lausekielen. Mutta Ada-kielen määrittelyn mukaiset tyyppi- ja muut tarkistukset käännöskirjastosta takaavat, että virheetön käännöstulos sopii komponenttina rakennettavaan järjestelmään. Tosin käännöspuhtaan ohjelman saaminen voi olla tarkistusten takia työläämpää kuin muilla kielillä. Muilla kielillä Ada-kääntäjän huomioimat virheet saattavat tulla esiin vasta ajoaikana.

Ada-ohjelmakirjastosta voidaan pelkästään pääohjelman nimellä yhdistellä kehitettävä ohjelmisto. Ja järjestelmä huomauttaa, jos ohjelmiston komponentit eivät ole yhteensopivia, ts. samoja määrittelyjä käyttäviä.

Pakkauksen määrittelyjä muutettaessa järjestelmä merkitsee kaikki pakkaukseen viittaavat yksiköt uudelleen-käännettäviksi ja uudelleen-käännösten jälkeen ohjelmisto on taas yhteensopiva. Uudelleen-käännökset siis takaavat yhteensopivuuden, mutta ne kuormittavat myös kehityslaitetta. Uudelleen-käännökset ovat se hinta, joka Ada-ohjelmointiympäristön tarkistuksista joutuu maksamaan.

"Projektin kannalta ikävää on oman ohjelmiston ulkopuolelta tuleva uudelleen-käännösvaade. Kun esimerkiksi järjestelmän varusohjelmisto tai tiedon-

"Arkkitehtuurisuunnittelun menetelmä Ada-ympäristössä on OOD (Object Oriented Design). Menetelmällä löydetään ohjelmiston objektit (tietorakenteet, tiedostot, prosessit jne.) ja niiden toiminnalliset ominaisuudet. Objektien toteutustekniikkaan ei vielä tässä vaiheessa puututa. Objektit nähdään kokonaisuuksina, joita voidaan käsitellä vain annetuilla operaatioilla, toiminnoilla. Adan pakkaus rakenne tukee OOD-menetelmää, sillä objektit kuvataan pakkausmäärittelyinä", Lahtinen valaisee.

Yksityiskohtaisessa suunnittelussa ratkaistaan objektien toteutustapa eli objekteja vastaavien pakkausten runkojen sisältö. Ratkaisu kuvataan Adalla käyttäen sitä suunnittelukielen tavoin. Kaikkia toteutuksen yksityiskohtia ei vielä ratkaista. "Pakkausrunkojen yksityiskohtainen suunnittelu voi alkaa runkojen arkkitehtuurin suunnittelulla, johon käytetään jälleen OOD-menetelmää. Muita käyttökelpoisia menetelmiä ovat vaikkapa JSP ja VDM", Lahtinen lisää.

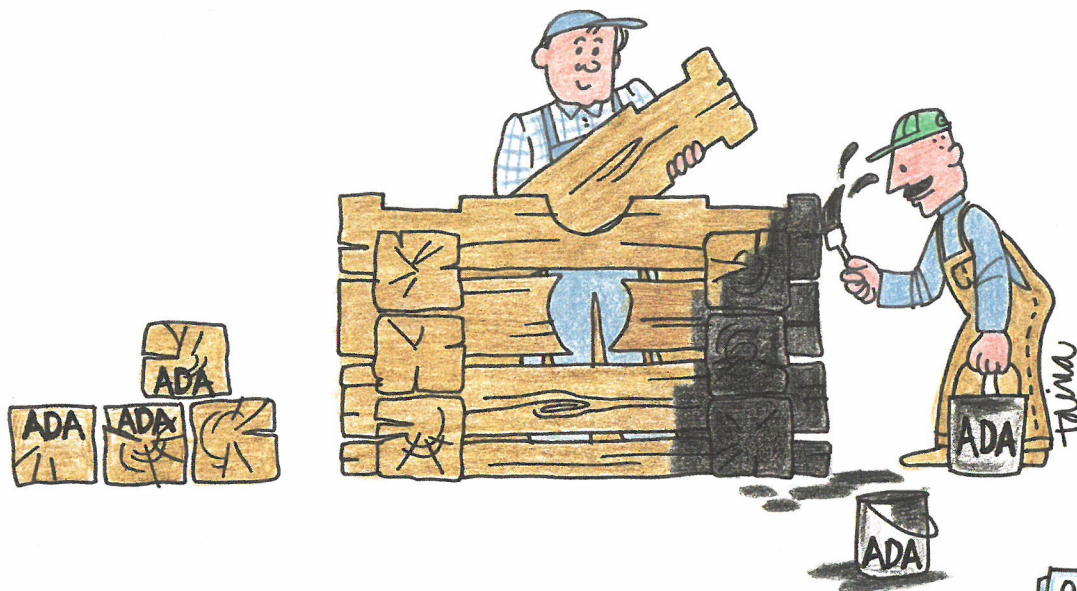
Adaa tai senkaltaista kieltä käyttämällä saadaan ohjelmiston liittymät ja jako hyvin suunniteltua. "Algoritmien ja käsittelyjen kuvaaminen yksityisen toiminnon osalta tällaisella suunnittelu-

kielellä johtaa helposti jo ohjelmointiin, jolloin ns. pseudokoodin ja varsinaisen ohjelmakoodin välillä ei ole suurta eroa", Lahtinen korostaa. Pseudokoodin sijaan Terho 'palastelisi' ohjelmiston niin pieniin ja yksinkertaisiin osiin, että toteutus voidaan jättää selkeästi ohjelmointivaiheeseen.

Ohjelmiston osan, pakkauksen, toteutus jätetään suunnitteluvaiheessa määrittelemättä ja keskitytään ohjelma-yksikön määrittelyyn. Eli suunnitteluvaiheessa ei Adalla kannata esittää ohjelmiston algoritmeja, vaan käyttää Adaa ohjelmiston jakamiseen riittävän pieniin yksiköihin, jotka voidaan toteuttaa sitten ohjelmointivaiheessa.

Kun suunnittelun tulokset kuvataan Adalla, voidaan jo tämän vaiheen apuvälineinä käyttää monia Adan työkaluja. Pakkausmäärittelyjen ja -runkojen muodollinen oikeellisuus voidaan tarkistaa Ada-kääntäjällä. Samoin pakkausten välisiä viittauksia voidaan tutkia ristiviittailuksista.

Adan korkean abstraktiotason rakenteet, ennen muuta pakkaus rakenne, sopivat erinomaisesti suunnitteluratkaisujen kuvaamiseen. Niinpä monissa projekteissa Adaa onkin käytetty suunnittelukielenä vaikka lopullinen toteutuskieli ei olekaan Ada.



hallintajärjestelmä muuttuu, tulee siitä riippuvat osat omasta sovelluksesta kääntää uudelleen”, Terho huomauttaa.

”Käännösruletin vastapainoksi saa Ada-ohjelmakirjastolta seuraavat edut: helpon ohjelmayksiköiden kokoamisen, automaattisen varmistuksen yhteensopivuudesta sekä näppärän tavan säilyttää ohjelmistoa ja ohjelmayksiköitä. Ada-ohjelmakirjaston hyviä puolia jää eniten kaipaamaan ohjelmoitaessa normaalissa PC-ympäristössä”, Terho tiivistää.

Ada-työkalut helpottavat testausta

Testausvaiheen tarkoituksena on varmistaa, että ohjelmisto täyttää sille asetetut vaatimukset. Koska ohjelmia ei pystytä todistamaan oikeiksi, pyritään testausvaiheessa yksinkertaisesti löytämään ohjelmistosta virheet, jotka sitten korjataan.

Jo Ada-kääntäjä tarkistaessaan, että ohjelma täyttää Ada-kielen semantiikan, löytää monia sellaisia virheitä, jotka perinteisissä ohjelmointijärjestelmissä ovat paljastuneet vasta testausvaiheessa.

Adan selkeä moduulirakenne, ohjelmiston pakkausjako, mahdollistaa kunkin moduulin testaamisen itsenäisesti jo ennen sen yhdistämistä koko ohjelmistoon. Tyypillisesti pakkauksia lähdeään testamaan alhaalta ylöspäin.

Ada-ympäristössä testataan ensisijaisesti pakkausten rajapintoja. Näin testausapuvälineiden laadinta voidaan aloittaa, kun rajapinta on määritelty. Rajapinnan toteutusta ei vielä tarvitse olla olemassa testausta aloitettaessa.

”Softplanissa pakkausrajapinnan testaus on automatisoitu ns. testipetigeneraattorilla. Generaattori analysoi pakkauksen määrittelyn ja tuottaa siitä pakkauksen testausohjelmiston lähdekoodin.”

”Testausohjelmiston käännöksen jälkeen voidaan sille syöttää suunnittelijoiden laatimat testitapaukset komento-tiedostoista. Niinpä virheiden etsintä pakkauksista on mekaanista ja nopeaa. Voimavarat voidaan keskittää havaittujen virheiden paikallistamiseen ja korjaamiseen. Virheiden paikallistamiseen on käytettävissä mm. jäljitysohjelma, jolla paikallistaminen on helppoa”, Terho kertoo.

Kun ohjelmiston rakennuspalikat (pakkaukset) on huolellisesti testattu, ei järjestelmätestauksessa enää löydy moniakaan virheitä; pakkausten välisten liittymien oikeellisuus kun sekin on tarkistettu jo käännösaikana.

Ylläpito hallinnassa

Ada-ohjelmisto otetaan käyttöön samoin kuin muutkin ohjelmistot. Ada-kielen tarkistukset ja nykyaikainen ohjelmankehitysympäristö ovat kylläkin vähentäneet ohjelmistojen virhemäärää.

Tulevaisuudessa valmiita ohjelma-komponentteja käyttämällä voidaan ohjelmisto saada suunnittelupöydältä käyttöön nopeammin kuin nykyään.

Kun projektin päättynyt ja ohjelmisto toimitettu, ohjelmakirjastot ja lähde-kieliset ohjelmat talletetaan yleensä normaalina varmuuskopiona. Sieltä ne voidaan palauttaa muutoksia ja ylläpitoa varten.

Kun muutoksia tehtäessä käytetään samoja kirjastoja kuin ohjelmistoa kehitettäessä, tulevat mukaan taas kaikki yhteensopivuuden tarkistukset ja siten muutokset pysyvät hallinnassa. ”Käytännön työssä on kuitenkin jäänyt kaipaamaan Ada-ohjelmakirjastoa, joka tukisi uuden rinnakkaisen version kehittämistä osittain samoista ohjelmayksiköistä”, Terho painottaa.

Ada on oikea kieli suuriin pitkäaikaisiin projekteihin sovellusalueesta riippumatta. Ada-kielisten ohjelmakomponenttien kehittäminen omiin tarpeisiin ja omaan käyttöön on mutkatonta. Ulkopuolelta ostettujen komponenttien käyttö ei sen sijaan ole aina mutkatonta, koska kirjastoratkaisut eri laitteissa ovat toisistaan poikkeavia. Näin ollen uudelleenkäytettävät ohjelmayksiköt tulisi siirtää eri Ada-toteutusten välillä lähdekielisessä muodossa. Tällöin käytönottoon kuluu jälleen käännösaikaa.

”Valitsisin projektini työkaluksi Adan. En siksi, että kielen konstruktiot olisivat edellä muita kieliä, vaan kielen ohjelmointiympäristöineen antaman kokonaisvaltaisen software engineering -ajattelun tuen takia”, Terho toteaa lopuksi. □

Ada-ohjelmakomponentti- kirjasto



Osa 4: Joukot ja tietorakenteinen ohjelmointi

Valmiiden ohjelmakomponenttien käyttö mahdollistaa uuden ajattelutavan ohjelmoinnissa, ns. tietorakenteisen ohjelmoinnin (Data Structured Programming). Tässä osassa esitellään joukkoja ja tietorakenteisen ohjelmoinnin periaatteita Ada-ympäristössä.

Joukko on Pascalista tuttu tyyppi (SET). Joukko muodostuu alkioista. Kukin alkio joko kuuluu joukkoon tai on sen ulkopuolella.

Joukon alkioiden järjestyksellä ei ole merkitystä päinvastoin kuin listoissa, jonoissa ja pinoissa. Alkio voidaan lisätä joukkoon, poistaa joukosta ja ohjelmassa voidaan myös tutkia alkion kuulumista joukkoon.

Joukko-käsite Adassa

Adassa ei ole ennalta määriteltyä joukkotyyppiä, vaan se on määriteltävä erikseen. Kun joukkoa määritellään,

on otettava huomioon, että joukon alkioiden maksimilukumäärä on ennalta määrätty /1/.

Joukot voidaan toteuttaa usealla eri tavalla. Jos joukon alkioiden maksimilukumäärä on pieni, käytännössä alle 256 alkioita, niin tehokkain toteutus on bittivektori. Tässä esitystavassa kutakin alkioita edustaa yksi bitti, joka ilmaisee onko alkio joukossa vai ei. Täten 256 alkioita käsittävän joukon esittämiseen keskusmuistissa tarvitaan vain 32 tavua pitkä muuttuja. Joukkomuuttuja on siten kiinteänmittainen riippumatta alkioden kulloisestakin lukumäärästä. Tämä helpottaa ja nopeuttaa käsittelyä.

Jos joukon alkioiden maksimilukumäärä on suuri, niin bittivektoriesitys ei

ole käytännöllinen. Tällöin joukon toteutukseen voidaan käyttää tässä artikkelisarjassa aiemmin esitettyä lineaarista listaa. Joukkoon kohdistuvat operaatiot ovat kummassakin tapauksessa samat.

Joukkotyyppi pitää toteuttaa tietokoneessa riittävästi ja täydellisesti. Tällä tarkoitetaan sitä, että ohjelmoijalla on käytössään riittävästi operaatioita joukon käsittelemiseksi. Täydellisellä tarkoitetaan sitä, että joukkotyypin kaikki ominaisuudet on otettu huomioon.

Jos usea rinnakkainen ohjelma käsittelee samaa joukkoa, tulee huolehtia, että ohjelma on suorittanut joukkooperaationsa loppuun ennenkuin toinen ohjelma käyttää tätä joukkoa. Joukkotyypin toteuttavassa pakkauksessa pitää huomioida myös virhetilanteet, kuten alkion poistaminen joukosta, jossa sitä ei ole. Virhetilanteiden käsittelyssä käytetään Adan poikkeusmekanismia. Tyhjää joukkoa käsitellään samoin kuin muitakin joukkoja.

Ohjelmoija voi määritellä joukon alkion tyypin. Joukon käsittelyoperaatiot ovat samoja tyypistä riippumatta. Riittää, että kirjastossa on yksi joukkopakkaus kutakin toteutustapaa kohden. Ohjelmassa tarvittavat joukkooperaatiot muodostetaan näistä geneerisistä pakkauksista ilmoittamalla vain alkion tyyppi.

Joukon operaatiot

Joukon käsittelyyn tarvitaan useita operaatioita. Alkion lisäämisen ja poistamisen lisäksi tarvitaan operaatiot joukon koon tutkimiseen, yksittäisen alkion läsnäolon testaamiseen, uuden joukon luontiin, joukon kopiointiin ja kahden joukon vertailuun. Niin ikään tarvitaan operaatiot testaamaan joukon kuulumista osajoukkona toiseen joukkoon, kahden joukon yhdistämiseen, erotukseen ja leikkaukseen.

Proseduuri INSERT (S, E) lisää joukkoon S alkion E. Jos alkion E tyyppi ei ole joukon S alkion tyyppiä, aiheutetaan ELEMENT_ERROR -poikeus.

Proseduuri DELETE (S, E) poistaa joukosta S alkion E. Yritettäessä poistaa alkio joukosta, jossa sitä ei ole, aiheutuu UNDERFLOW -poikeus. Siksi tarvitaankin funktio IS_MEMBER (S, E). Se palauttaa arvon TRUE, jos joukossa S on alkio E, muuten arvon FALSE.

Proseduurilla EMPTY (S) luodaan uusi joukko S ja asetetaan sen alkuarvoksi tyhjä joukko.

Joukon S kokoa voidaan kysyä funktiolla NUMBER_OF_ELEMENTS (S), joka palauttaa arvonaan joukossa

kutsuhetkellä olleiden alkioiden lukumäärän.

Kahta joukkoa voidaan verrata keskenään funktiolla EQUAL (S1, S2). Se palauttaa arvon TRUE, jos joukoissa S1 ja S2 on täsmälleen samat alkiot.

Joukko voidaan monistaa kopioimalla se toiseen joukkoon proseduurilla COPY (S1, S2), joka sijoittaa joukon S2 arvoksi joukon S1 sisällön.

Joukon kuulumista osajoukkona toiseen joukkoon testataan funktiolla INCLUSION (S1, S2). Se palauttaa arvon TRUE, jos joukon S1 kaikki alkiot sisältyvät joukkoon S2.

Joukot voidaan yhdistää yhdeksi joukoksi proseduurilla UNION (S1, S2, S3), joka muodostaa joukon S3 joukkojen S1 ja S2 alkioista.

Joukkoerotus muodostetaan proseduurilla DIFFERENCE (S1, S2, S3). Se muodostaa joukon S3 joukon S1 niistä alkioista, jotka eivät kuulu joukkoon S2.

Joukkojen leikkaus muodostetaan proseduurilla INTERSECTION (S1, S2, S3), joka muodostaa joukon S3 joukkojen S1 ja S2 yhteisistä alkioista.

Kurinalaisuutta tietorakenteisella ohjelmoinnilla

Ada-komponentteja hyödynnetään parhaiten ns. tietorakenteisessa ohjelmoinnissa.

Rakenteinen ohjelmointi (Structured Programming) on vanhastaan tuttu käsite. Sillä tarkoitetaan ohjelman tekemistä kolmesta eri perusrakenteesta: peräkkäisyys, valinta ja toisto. Paljon parjattua GOTO-käskyä ei käytetä.

Tietorakenteinen ohjelmointi tuo lisää kurinalaisuutta ohjelmointiin. Siinä rajoitetaan tiedon nimeämistä ja viittausta. Ohjelma koostuu samoista rakenteista kuin rakenteisessa ohjelmoinnissa, mutta lisäksi kielletään taulukoiden ja osoittimien käyttö /2/. Näiden tilalla käytetään joukkoja, jonoja ja pinoja. Ohjelmointi nämä rajoitukset huomioiden on mahdollista ja siitä on vielä odottamatonta hyötyäkin. Ohjelmissa, joissa käytetään joukkoja, jonoja ja pinoja, on vähemmän muuttujia ja tietoviittauksia kuin ohjelmissa, joissa käytetään taulukoita ja osoittimia.

Tärkeintä on kuitenkin se, että ohjelman rakenne on erilainen. Tiedon abstrahointi (abstraktiotasojen muodostaminen) on korkeammalla tasolla, mikä merkitsee sitä, että ohjelma vastaa paremmin ratkaistavan ongelman käsitteitä. Tuloksena on, että ohjelma on todennäköisemmin oikea.

Joukkoihin, jonoihin ja pinoihin ei liity tietoa alkion osoitteesta. Taulukoon sen sijaan liittyy alkion indeksi tai indeksit moniulotteisen taulukon tapauksessa. Ohjelmoijat ovat tottuneet käsittelemään osoitteita aivan samoin kuin ennenvanhaan GOTO-käskyä. Tämän tapainen ohjelmointi on siksi vaikeampaa kuin perinteinen ohjelmointi.

Sekä jonolla että pinolla on oma historiansa tiedon abstrahoinnina erityisissä tilanteissa. Pinoa on käytetty kääntäjissä ja jonoa käyttöjärjestelmissä. Kuitenkin molemmat ovat yleispäteviä työkaluja tiedon abstrahoinnissa.

Vaatii uutta ajattelutapaa

Samoin kuin rakenteinen ohjelmointi vaatii ohjelman perusteellisempaa suunnittelua, tietorakenteinen ohjelmointi vaatii uudenlaista ajattelutapaa. Etuna on kuitenkin ohjelman ymmärtämisen helpottuminen. Myös ohjelmien oikeaksi todistaminen on käytännössä oleellisesti helpompaa kuin muilla tyyliellä tehdyissä ohjelmissa. Oikeaksi todistaminen ei kuitenkaan vielä ole niin helppoa, että sitä käytännössä tehtäisiin.

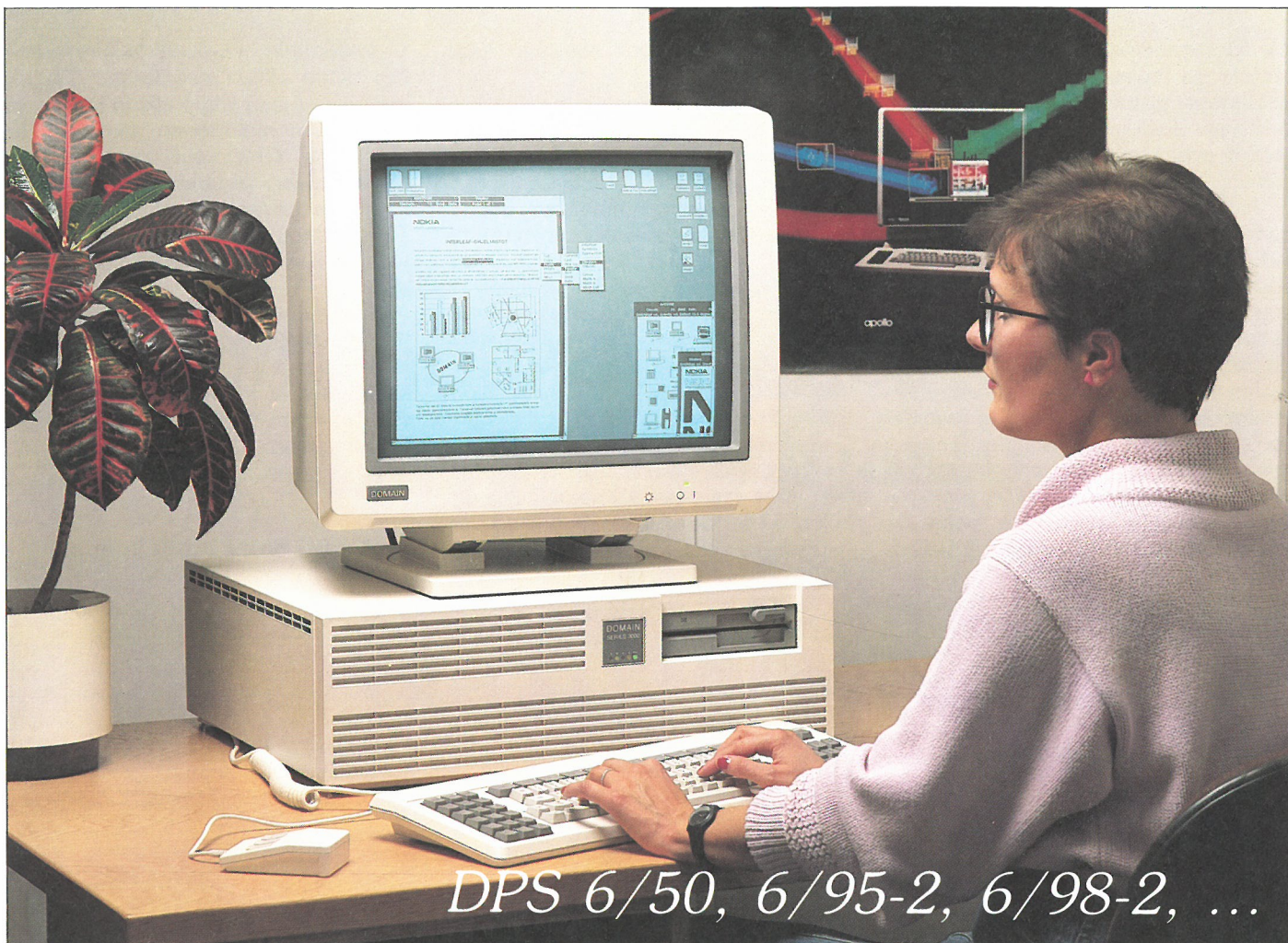
Tietorakenteinen ohjelmointi vaikuttaa ohjelman suunnittelijan ajattelutapaan. Ongelman ratkaisu tulee kuvattua ongelmanläheisesti ilman turhia apumuuttujia, kuten taulukkojen indeksit. Tällä tavalla tehdyt ohjelmat ovat lyhyempiä ja yksinkertaisempia kuin taulukoita ja osoittimia käyttävät ohjelmat. Ne ovat myös tehokkaampia. Taulukoita ja osoittimia toki tarvitaan joukkojen, jonojen ja pinojen toteutuksessa, mutta ohjelmoijalle ne ovat näkymättömiä.

Tietorakenteinen ohjelmointi merkitsee samanlaista muutosta ohjelman suunnittelijan ajattelutavassa kuin rakenteinen ohjelmointi aikoinaan. Nykyisissä ohjelmointikielissä ei ole suoraan käytettävissä jonoja ja pinoja, mutta Adalla tehdyt ohjelmakomponentit ovat käyttökelpoinen ratkaisu.

Viitteet

/1/ Swan, T.: Set Implementation, PC Technical Journal, February 1986.

/2/ Mills, H.D., Linger, R.C.: Data Structured Programming — Program Design without Arrays and Pointers, IEEE Transactions on Software Engineering, Vol. SE-12, No 2, February 1986.



DPS 6/50, 6/95-2, 6/98-2, ...

Uusia malleja kutossarjaan

Nokia on täydentänyt DPS 6 -laitesarjaa DPS 6/50, 6/95-2 ja 6/98-2 -koneilla. Samassa yhteydessä on kutossarjaan julkistettu myös oheislaitteita.

Niinpä laitteilla voidaan ajaa hyvinkin suuria sovelluksia sekä useita sovelluksia samanaikaisesti.

Kaikki uudet mallit ovat täysin yhteensopivia DPS 6 -sarjan muiden koneiden kanssa.

DPS 6 -sarjan koneet muodostavat yhtenäisen laitesarjan, joka mahdollistaa järjestelmän laajentamisen joustavasti käyttäjän tarpeiden mukaan. Sarjan alapäästä edustavat 16 bitin supermikrot ja yläpäästä yli sadan päätteen 32 bitin superminit.

DPS 6 -koneita valmistetaan USA:ssa, Englannissa, Ranskassa ja Italiassa. Kutosia on asennettu Suomessa noin 550 ja maailmalla kaikkiaan noin 60 000.

Kaikissa DPS 6 -koneissa on sama käyttöjärjestelmä GCOS6 MOD400, joten ohjelmistot ovat yhteensopivia kaikissa DPS 6:ssa. Sovellusohjelmat tarvitsee tehdä vain kerran ja minkä jälkeen niitä voi käyttää kaikissa DPS 6 -koneissa ilman uudelleenkirjoitusta.

Uudet kutoset

DPS 6/50 täydentää hyvin nykyistä DPS 6 -laitesarjaa. Se on jatkoa matalakabinettisille koneille, DPS 6/20:lle, DPS 6/22:lle ja DPS 6/40:lle.

Kutossarjaan on nyt julkistettu myös kaksisuorittimisia (-prosessorisia) koneita, DPS 6/95-2 ja DPS 6/98-2. Kahden suorittimen ansiosta koneet ovat noin 1,7 kertaa tehokkaampia kuin vastaavat yksisuorittimiset. Toinen suoritin lisää myös toimintavarmuutta.

Mallissa 6/98-2 on 6/95-2:sta poiketen levypuskurimuisti, jolla voidaan puskuroida levyhakuja ja siten lyhentää vasteaikoja sovelluksesta riippuen. Muuten koneet ovat samanlaisia. Keskusmuistia koneissa on 16 megatavua.

DPS 6/50 tehokas mini

DPS 6/50:ssä on sama suoritin kuin DPS 6/75:ssä ja siten se on erittäin tehokas minikone. Maksimikokoonpanossaan se on 48 käyttäjän laite ja keskusmuistia siihen saa kaksi megatavua.

Laite sisältää peruskokoonpanossaan 5,25 tuuman levykeaseman. Levykkeiden formatoitu kapasiteetti on 650 kilotavua.

DPS 6/50:een voidaan liittää kuusi levy-yksikköä ja käyttämällä uusia kiinteitä 413 megatavun FSD413-levyaseimia laitteen levykapasiteetti nousee aina 2400 megatavuun. Varmuuskopiointia varten koneeseen saa myös nauha-aseman.

Pienen kokonsa ja suuren tehonsa ansiosta DPS 6/50 soveltuu hyvin toimistoympäristöön, esimerkiksi TOIMI-toimistojärjestelmän laitteeksi.

Toimintavarmoja oheislaitteita

Uusien DPS 6 -mallien ohella on julkaistu myös kiinteä levyasema FSD413, nauha-asema MTU9635, puskuroitu levyasemaohjain DIRAM 32 sekä RS-232 -sovitin DCM 9643.

Pienikokoisen FSD413 winchester-levyn formatoitu kapasiteetti on peräti 413 megatavua. Lisäksi levyn hinta/kapasiteetti -suhde on alhainen. Koska levyt ja luku/kirjoituspäät on asennettu ulkoilmalta suojattuun koteloon, levyaseman toiminta on myös luotettavaa. Uuden puskuroidun ohjaimen ansiosta levyaseman siirtonopeus on jopa 1,8 megatavua sekunnissa.

Nauha-aseman MTU9635 nauhoitustiheys voi olla koodauksesta riippuen joko 6250 bpi (GCR-koodi) tai 1600 bpi (PE-koodi). Niinpä nauhan talletuskapasiteetti voi vaihdella 41 megatavusta 140 megatavuun.

Nauha-asemassa on monia perinteisesti mekaanisia toimintoja korvattu puolijohdetekniikalla. Asemassa ei ole sellaisia yleensä paljon huoltoa vaativia mekaanisia osia kuten vetorulla. Myöskään alipainekammioita ei tarvita, vaan nauhakuja on lyhyt. Näin aseman toimintavarmuus on hyvä.

Nauha-aseman suorituskyky on myös korkea valittavan nauhanopeuden ja 128 kilotavun muistipuskurin ansiosta.

Levyn I/O nopeammaksi

Levyasemien hitauden vuoksi keskuksikkö joutuu yleensä odottamaan levylle tulostuksen ja sieltä luvun ajan. Käyttämällä puskuroitua DIRAM 32 -levyasemaohjainta voidaan levyn I/O-toimintoja nopeuttaa. Tällöin moniajo-ympäristössä vasteajat lyhenevät ja työasemien määrää voidaan lisätä.

Peruskokoonpanossaan DIRAM 32:ssa on liitäntä yhdelle DPS 6:lle ja neljälle levyasemalle. Laajennuksien avulla ohjaimen voi kytkeä jopa 16 levyasemaa ja sen voi liittää neljään DPS 6 -koneeseen. Ohjaimen muistipuskurin koko on 2-16 megatavua.

Uudella DCM 9643 -sovitinella voidaan kasvattaa DPS 6 -koneiden RS-232 -liitäntöjen määrää. Sovittimen toiminta on valittavissa kytkimellä synkroniseksi tai asynkroniseksi. □



DPS 8 -julkistuksia

DPS 8 -laitesarjaan on elokuun lopulla julkaistu kaksi uutta laitesarjaa DPS 8/480 ja 8/780, sekä mm. MAGNA 8 ja PARS 8 -ohjelmistotuotteet.

MAGNA 8 on ns. neljännen sukupolven sovelluskehitin, joka generoi Cobol 68 ja Cobol 74-kielisiä ohjelmia. Näin kehitin mahdollistaa tietokantasovellusten rakentamisen ilman perinteistä ohjelmointia. Se merkitsee ohjelmistotyön tuottavuuden lisäystä ja ylläpidon helpottumista.

Helppokäyttöisyytensä ansiosta MAGNA 8 soveltuu loppukäyttäjienkin työvälineeksi esimerkiksi yksinkertaisten raporttien ja kyselyjen tekoon sekä henkilökohtaisten tiedostojen ylläpitoon.

MAGNA 8 pystyy käsittelemään useita erityyppisiä tiedostoja sekä DM4-tietokantoja. Niinpä sitä on mahdollista käyttää myös jo olemassa olevien sovellusohjelmistojen yhteydessä.

MAGNA 8 -kehittäjä käytetään vuorovaikutteisesti vastaamalla sen esittämiin kysymyksiin yksinkertaisin englanninkielisin komennoin. Kehittämistä löytyy myös HELP-toiminto, jolla käyttäjä saa näytölle kulloisessakin tilanteessa tarvittavia ohjeita.

Muista MAGNA 8:n ominaisuuksista mainittakoon oma integroitu tietohakemisto, mahdollisuus yhdistellä

tiedostoja relaatioperiaattein sekä liitännät useisiin mikrojen ohjelmistoihin.

PARS 8 -ohjelmalla DPS 8:n käyttäjä voi jatkuvasti seurata GCOS 8 SR2500 -järjestelmän käyttöä. Ohjelma kerää tietoja järjestelmän resurssien käytöstä ja esittää ne sekä yksityiskohtaisina raportteina tietystä tilanteesta että pitemmän aikavälin suuntausta kuvaavina tilastoina ja kaavioina.

Uusissa DPS 8 -laitteistoissa on parannettu ja tehostettu I/O-toimintoja, jolloin suoritintehon lähes optimaalinen käyttö on mahdollista. Niinpä laitteistot sopivat eritoten tapahtumakäsitteilyyn.

Uusittu laitesarja DPS 8/481, 482T, 483T ja 484T korvaa aiemmat DPS 8/47-DPS 8/62 -mallit. Niin ikään aiemmat DPS 8/70 -mallit korvataan laitesarjalla DPS 8/781, 782T, 783T ja 784T. Siirtyminen ja laajentaminen uusiin malleihin on tehty mahdollisimman helpoksi teknisesti ja taloudellisesti. Esimerkiksi keskuksikköt ja lisämuistit ovat halventuneet oleellisesti.

Uusia julkistuksia ovat myös MSS0680 -levyjärjestelmä ja I/O -sovitin (IOP), jonka avulla IBM:n isot levyasemat voidaan liittää DPS 8 -järjestelmään.



Kansainvälinen yhteistyö yritysten välillä on lujittumassa mm. Eureka-projektien myötä. Myös Nokia on mukana kahdessa tietotekniikan kannalta merkittävässä Eureka-hankkeessa, ja kolmaskin on jo näköpiirissä.

Haastattelimme Nokian Tutkimuskeskuksen johtajaa Viljo Hentistä, joka on Nokian yhdyshenkilönä käynnistänyt Nokian Eureka-projektit.

Tärkein projekteista on EAST (Eureka Advanced Software Technology). Hankkeessa on tavoitteena kehittää ohjelmointityökaluja, joilla ohjelmoinnin tuottavuutta voidaan nostaa merkittävästi.

Ohjelmakehitystyökalut on tarkoitus rakentaa useampaan laitteistoon sopiviksi. Lähtökohtana ovat tehokkaat kehittyneet työasemat, jotka on verkotettu keskenään. Laiterympäristöjä, joihin työkalut istutetaan, ovat esimerkiksi MikroVax 2, Apollo-tiöasemakoneet ja PC/AT yhteensopivat mikrot.

Nokia vah

Työkalut valmiiden perusosien ympärille

Siirrettävyyden vuoksi ohjelmointityökalut aiotaan rakentaa Unix-käyttöjärjestelmään, Unix System V:een. Sen päälle on rakennettu eräänlainen käyttöjärjestelmän laajennus EMERAUDE, joka on kehitetty Ranskassa EC:n ESPRIT-projektin pohjalta. Näiden valmiiden perusosien ympärille on EAST-projektin tavoitteena kehittää työkaluja, jotka käyttävät hyväksi sekä Unixia että EMERAUDEn kehittynyttä käyttäjäliitäntää ja valmiita perustyökaluja, kuten esimerkiksi objektien hallintajärjestelmää.

Projektin kustannusarvio on noin puoli miljardia markkaa. Työmäärältään se vastaa noin 600 miestyövuotta. Tiettyjä projektin osia on tarkoitus ostaa valmiina myös ulkopuolisilta yrityksiltä.

EAST-hanke merkitsee Nokialle ensisijaisesti kilpailukykyisen kaupallisen tuotteen synnyttämistä kansainvälisille markkinoille. Nokian toinen tavoite hankkeessa on kehittää ohjelmistosaamista ja -tuottavuutta yhdessä eturivin ohjelmistotalojen kanssa. Projektissa hankitaan myös kokemusta eurooppalaisesta yhteistyöstä ja luodaan tärkeitä kontakteja muihin yrityksiin.

Yhteistyönä seitsemän muun yrityksen kanssa

Projektissa on päävastuu vastaperustetulla ranskalaisella yrityksellä SFGL, jonka omistaa Honeywell Bull yhdessä kuuden ohjelmistotalon kanssa. Nokian ja SFGL:n lisäksi mukana ovat Tanskasta CRI, Italiasta DATAMAT, INTECS, SESA/Italia ja Selenia, sekä Sveitsistä CIR.

vasti mukana Eurekassa

EASTin johto on Pariisissa toimival-
la johtoryhmällä, joka vastaa projektin
arkkitehtuurin suunnittelusta ja koor-
dinoinnista. Eri maissa olevat työryh-
mät kehittävät ikään kuin alihankinta-
na tiettyjä projektin osia. EASTissa No-
kialta tulee työskentelemään Pariisissa
nelisen ja Suomessa enimmillään 10-
15 henkilöä. Ensimmäiset nokialaiset
siirtyvät Pariisiin jo syyskuussa.

Mukana myös rakentamassa tietoverkkoa

Toinen Nokian Eureka-projekteista on
Cosine, johon Nokia on mennyt mu-
kaan jälkeensä opetusministeriön
ehdotuksesta. Hankkeen tarkoituksena
on kehittää Euroopan yliopistojen
ja tutkimuslaitosten välinen tietoverk-
ko mahdollisimman pitkälti jo olemas-
sa olevista laite- ja ohjelmakomponen-
teista.

Projektin luonteen vuoksi Cosines-
sa on mukana useita maita. Kustan-
nusarvioltaan se on kuitenkin huomattavasti
vaativampi kuin EAST.

Kolmas hanke, johon liittymisestä
neuvotellaan, on nimeltään ES2 (Euro-
pean Silicon Structures). Siinä tavoit-
teena on kehittää integroitujen piirien
suunnittelu- ja valmistusmenetelmä,
jossa kuviot piisirun pinnalle synnytetään
suoraan elektronisuihkulla. Mas-
keja ei siis tarvita lainkaan.

Lisätietoja Nokian Eureka-projek-
teista antavat Tutkimuskeskuksen
johtaja Viljo Hentinen, puh.
(90) 567 2710, sekä osastopäällikkö
Matti Kilpi, puh. (90) 567 2783.

Mistä Eurekassa on kyse?

Eureka on yksi yritysten välinen yhteistyömuoto. Eureka tarkoituksena on jakaa informaatiota ja saada yrityksiä keskinäiseen yhteistyöhön, jonka tuloksena olisi kaupallisia tuotteita. Siksi Eureka-projektit ovat ensisijaisesti juuri yritysten eivätkä esimerkiksi tutkimuslaitosten välisiä hankkeita.

Jotta hanke tulisi hyväksytyksi ns. Eureka-projektiksi, sen tulee täyttää tietyt muodolliset kriteerit. Projektissa tulee mm. luoda uutta teknologiaa. Hankkeen tulee myös nostaa siinä mukana olevien yritysten kilpailukykyä. Lisäksi Eureka-projektilta edellytetään, että rajojen yli tapahtuva yhteistyö parantaa oleellisesti hankkeen onnistumisedellytyksiä.

Projektit hyväksytetään Euran ministerikokouksissa, joita on toistaiseksi pidetty kolme: aloituskokous Pariisissa heinäkuussa 1985, toinen Hannoverissa viime syksynä ja kolmas Lontoossa kesäkuun viimeisenä tänä vuonna. Seuraava kokous järjestetään Tukholmassa tämän vuoden lopulla.

Ministerikokousten välissä on tyypillisesti parikolme ns. valmistelemaa kokousta, joissa varsinaisesti käydään läpi ehdolla olevia projekteja ja todetaan, täytävätkö ne asetetut kriteerit.

Eureka-projektiksi nimeämisestä on projektille hyötyä lähinnä siinä, että sen on helpompi saada kansallista rahallista tuotekehitystukea.

Ministeri- ja valmistelevien kokousten ohkeen on Lontoon kokouksessa päätetty perustaa Euralle pysyvä sihteeristö Brysseliin.

Euran piiriin on hyväksytty toistaiseksi 72 projektia, joista 62 hyväksyttiin viimeisessä ministerikokouksessa. Eurekassa on tällä hetkellä 19 Länsi-Euroopan maata. Viimeisenä mukaan on otettu Islanti.

Suurin osa Eureka-maista on jo osallisena eri projekteissa. Ranska on ollut aktiivisin Eurekassa. Se on osallistujana 40 projektissa. Englanti on mukana 29:ssä ja Länsi-Saksa 15:ssä hankkeessa. Suomi on suhteellisesti ottaen keskimääräistä aktiivisemmin mukana. Suomalaisyhtiöitä on tällä erää kahdeksassa projektissa. Nokian lisäksi Eureka-projekteissa ovat mm. Kone, Vaisala ja Valmet.

Suomalainen sovelluskehitin maailmalle

Nokia on myynyt kehittämänsä Nopsa-sovelluskehittimen Englannin Honeywellille. Se jatkokehittää tuotetta ja markkinoi sitä GAIN-nimellä ympäri maailmaa.

Nopsa on ohjelmakehitin DPS 6 -koneiden TPS 6 -tapahtumakäsittelysovelluksien rakentamiseen. Se on atk-ammattilaisen monipuolinen työkalu vaativienkin sovellusten tekoon ja ylläpitoon.

Jo 80-luvun alusta on Nopsaa kehitelty Nokia Informaatiojärjestelmissä useista tiedonhallintaprojekteista saatujen kokemusten pohjalta. Nopsa perustuu toisaalta ns. Primas-työkalupakkiin ja toisaalta omaan tietohakemistoonsa, joiden avulla eri sovellusten näytöt ja tietokantamäärittelyt saadaan standardoitua ja mahdollisimman yhdenmukaisiksi. Ensimmäinen versio Nopsasta julkistettiin vuonna 1983.

Nopsa maailmalle

Viime vuoden keväällä Englannin Honeywell kiinnostui Nopsasta. Tutustuttuaan Nopsaan englantilaiset halusivat siihen muutamia uusia piirteitä. Ja pian he käynnistivätkin Nopsa-kokeiluprojekteja. Samalla aloitettiin neuvottelut Nopsan myynnistä ja sopimukseen päästiin vielä samana keväänä.

Nyt Englannin Honeywell on julkaissut kehittimen ja markkinoi sitä tuotenimellä GAIN (Generation of Applications for TPS 6). Englantilaiset ovat jo sopineet mm. amerikkalaisten kans-

sa kehittimen jälleenmyyntioikeuksista. Niinpä kehitin on kovaa vauhtia pääsemässä maailman laajuiseen jakeluun.

Myös GAINin jatkokehitykseen on Honeywell satsaamassa voimakkaasti. Tällä hetkellä tuotetta kehittää Englannissa kymmenkunta henkilöä. Kehitystyön periaatteena on yhteensopivuus ylöspäin. Näin ollen kaikki nykyiset Nopsat ovat jatkossakin päivitettävissä.

Raporttikehitin uutuutena

GAINiin sisältyy Nopsasta tuttujen tietokannan kuvaus-, näytönmaalaus- ja protoiluohjelmien sekä malliohjelmakehittimen lisäksi raporttikehitin. Muuten GAIN on sama kuin Nopsa. Erikieliset versiot on kehittämissä toteutettu parametritiedostolla.

GAIN-sovelluksen perustana on sovelluskuvasto eli aktiivinen tietohakemisto, johon jo sovelluksen suunnitteluvaiheessa määritellään sovelluskohdaisia tietoja. Tällaisia tietoja ovat kentät ja niiden ominaisuudet, näytöt, tietue- ja tiedostokuvaukset.

GAINin protoiluohjelmalla voidaan heti näytönmaalauksen jälkeen testata kyseistä näyttöä. Näin protoiluohjelma helpottaa testausta oleellisesti.

Kehittimellä Cobol-koodia

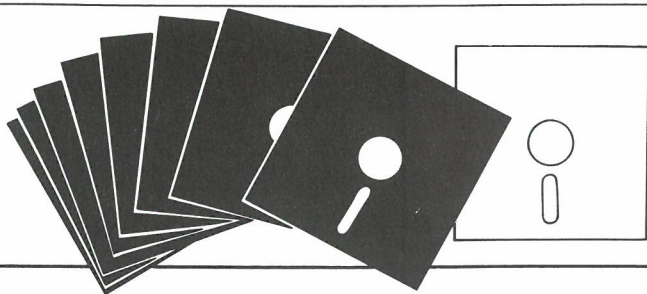
Sovelluskuvaston tiedoista generoidaan GAIN-kehittimellä halutut Cobol-kieliset sovellusohjelmat.

Kehittimen tuottamissa ohjelmissä käytetään standardoituja ratkaisuja, minkä johdosta ohjelmat muistuttavat toisiaan ja ovat helposti ylläpidettäviä. Standardeilla ratkaisulla on pyritty sekä tehokkaaseen koneen resurssien käyttöön että helppokäyttöisyyteen.

GAIN-sovelluksissa ovat valmiina mm. numeerisuus-, päivämäärä- ja henkilötunnustarkistukset sekä kenttäkohtaiset muokkaukset. Ohjelman Cobol-koodiin on helppo tehdä myös tarvittavia lisäyksiä ja muutoksia.

GAINin sovelluskuvasto on sinänsä ajan tasalla oleva kuvaus sovelluksesta. Kuvastoa voi tarkastella päätteeltä. Samaten kuvastosta voidaan tulostaa listaukset. Kuvastosta nähdään, missä näytöissä ja tietuekuvauksissa mitään kenttää käytetään. Lisäksi saadaan mm. täydelliset ristiviitteet kenttien ja tiedostojen käytöstä eri ohjelmissa. Kun sovellus toteutetaan GAINilla, se saadaan dokumentoitua automaattisesti.

Lisätietoja: Risto Taskula,
puh. (90) 187 4070



MikroMikko TopTen

Tämäkertainen MikroMikkojen ohjelmistolista perustuu kuluvaan vuoden touko-, kesä- ja heinäkuun myyntiluvuista laskettuihin suhteellisiin indekseihin.

Myydyn ohjelma tarkasteluajaksolla on ollut selvästi Multiplan, joka on näin palannut kymmenikköön. Suosituimmaksi tekstinkäsittelyohjelmaksi on jälleen noussut WordStar. Sen sijaan Edix sekä Wordix ovat pudonneet listan ulkopuolelle.

Mielenkiintoista on nähdä, miten MikroMikko 3:n tulo markkinoille alkaa näkyä listalla. Jo nyt MikroMikko 3:n Windows on yltänyt neljänneksi.

MikroMikko TopTen

1. (-)	Multiplan	112
2. (5)	WordStar	52
3. (6)	MS-Basic -tulkki	25
4. (-)	Windows	23
5. (3)	Multigraph	22
6. (1)	WordPerfect	21
7. (4)	MPS-Index	15
8. (2)	K-mies	14
9. (-)	NET 7800	13
10. (7)	Lotus 1-2-3	12

Suluissa sijoitus edellisessä TopTenissä (Softari 2/86).

MikroMikko 2 HP 3000 -laitteiston päätteeksi

MikroMikko 2:n tietoliikenneohjelmien valikoimaa on laajennettu uudella pääte-emulaattorilla, jonka avulla MikroMikko voi toimia HP 3000 -laitteiston pääteenä.

Emulaattorin on kehittänyt lappeenrantalainen Kaakontieto, joka on myynyt ohjelman yksinmyyntioikeudet Nokialle.

Emulaattorilla MikroMikko toimii HP 2622A -pääteen tavoin. Pääteyhteyden lisäksi ohjelma mahdollistaa oheiskirjoittimen käytön sekä tiedostojen siirron MikroMikon ja HP 3000 -laitteiston välillä.

Ohjelmaan on toteutettu kaikki ne HP 2622A -pääteen ominaisuudet, jotka laitteiden eroista huolimatta ovat olleet mielekkäitä. Emulaattorista puuttuvat vain seuraavat ominaisuudet, joista useimmat eivät ole tarpeen suomalaisissa käyttöympäristöissä:

- Language-määrittäminen
- Frame Rate -valinta
- RETURN-määrittäminen
- näytön sammutus ja sytytys ESC W -komennolla
- Start of Text -osoitin MODIFY-tiloissa
- Alternate Character Set on viivaalkiomerkit (Set B)

Sen sijaan lisäominaisuuksina HP-pääteeseen ovat:

- näyttöattribuutteihin on lisätty kentän suojaustoiminto
- näytön alarivi on varattu tilariviksi

Uudet ohjelmaversiot toimivat kaikissa MS-DOS -käyttöjärjestelmäversioissa. Ne toimivat myös lähiverkon (NetNet 2.4 tai uudempi) MikroMikko 2:ssa, joka on kytketty verkkoon IOE-kortin kautta. Ohjelmat päättelevät automaattisesti, onko kyseessä yksittäismikro vai verkko.

ASCOMin ja ASCOM-VDU202:n uusien versio on 2.29 CP 2.6 sekä Videotexin 5.6.

Microsoft Rbase MikroMikkoille

Microsoftin Rbase-tiedonhallintaohjelmisto on nyt saatavissa MikroMikko 2:lle ja 3:lle. Toimiakseen ohjelmisto vaatii umpilevyn.

Rbase on sovelluskehitykseen tarkoitettu, relaatiomalliin perustuva tiedonhallintaohjelmisto.

Rbasea voi käyttää komentotilassa antamalla SQL-tyyppisiä komentoja. Ohjelmiston mukana toimitetaan myös seuraavia apuvälineitä tietokannan ja sovellusten rakentamiseen:

- Express — valikko-ohjattu tietokannan ja sovellusohjelman määrittely-ohjelma
- Rcompile — kääntäjä, joka kääntää komentokieliset ohjelmat binäärimuotoon. Näin kääntämällä voidaan suojata ohjelmat ja nopeuttaa niiden suoritusta.
- Gateway — muunnosohjelma eri formaateilla talletettujen tiedostojen, kuten esimerkiksi Lotus-, dBASE-, Multiplan- ja ASCII-tiedostojen, muuttamiseksi Rbase-esitysmuotoon.

Lisäksi Rbaseen sisältyy välineet syöttölomakkeiden ja raporttien määrittelyyn.

Unto MikroMikko 3:lle

Pk-yritysten tarpeisiin suunnitellut Unto-taloushallinnon ohjelmat on sovitettu MikroMikko 3:lle.

Unto on Unicin ja Efectorin yhdessä alunperin MikroMikko 2:lle kehittämä ohjelmisto. Unton on todettu MikroMikko 3:ssa noin 50 % nopeammaksi kuin MikroMikko 2:ssa. Nopeus näkyy selvimmän paljon tiedostokäsittelyä vaativissa toiminnoissa, kuten kirjanpidon päivityksessä.

Uusia versioita MikroMikko 2:n tietoliikenne ohjelmista

ASCOM, ASCOM-VDU202 sekä Videotex-ohjelmista on julkistettu MikroMikko 2:lle uudet versiot.

Nokia
Softari-lehti
PL 780
00003 HELSINKI

Vastauslähetys
HKI 10 Lupa 2509

Nokia
maksaa
postimaksun

Softari 3/86

Palvelukortti

☐ Haluan esitteitä seuraavista
Nokian palveluista ja tuotteista.

☐ Pyydän, että minuun otetaan
yhteyttä.

puhelin: _____

Olen kiinnostunut _____

☐ Korjaan yllämainitut osoitetiedot.

☐ Poistan yllämainitut osoitetiedot.

☐ Lisään allamainitut osoitetiedot.

☐ Softarin ☐ NETin rekisteristä/rekisteriin.

yritys _____

osasto _____

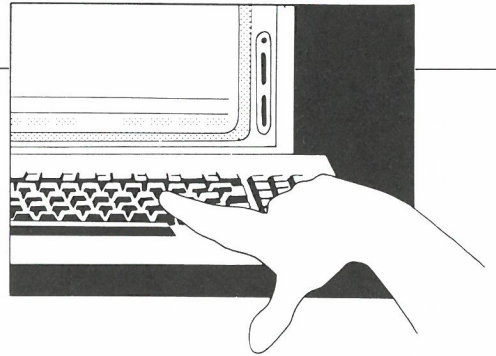
lähiosoite _____

postinro _____ postitoimip. _____

nimi _____

JOUKKOJULKAISU

Ilo irti Kt:stä



Tähän aikaan vuodesta ei kannata matkustaa kovin kauas Suomesta, onhan taas Kt-messujen tuuri. Ja niistä hän ei viitsi jäädä pois, on siellä niin lustia tällaiselle tavalliselle selväjärkiselle spesialistille.

Tosin onhan ulkomaillakin joitain messuja, joissa olisi jopa jotain näkemistäkin. Mutta kun kerta meikäläisen SICOB-matka-anomus tuli taas bumerangina takaisin — on se niin väärin — niin katsotaan nyt sitten se Kt. Ja sitä paitsi kertaakaan ei ole tarvinnut nauramatta pois lähteä.

Heittäydynpä kiinnostuneeksi — mutta kokemattomaksi — ostajaksi, jolle sitten ne maailman hienoisimmat — mutta halvimmat — systeemit yritetään myydä. Toiset yrittävät epätoivoisesti myydä ja meikäläinen vain kurillaan hauskuuttaa harmaata syksyään ja antaa lipeväkielisten myyntitykkien ylittää itsensä. Semmoista se on, kun kaikilla on maailman parhaat värit. Tulepa mieleeni vanha sananparsa, jossa tyhmää jymäytetään tai jotain sinnepäin.

Katsotaanpas. Tuttua superlatiivilaulua laulellaan osastolla kuin osastolla ja vanhan vatsakalvot eivät taaskaan varmaan kestä kuin yhden vierailukäynnin.

Aha, mikäs tässä on? Näyttelysasto suoraan ulkoavaruudesta. Tässähän taitaa olla sitä kuuluisaa eksotiikkaa ja mitkä mimmit! Ufox Computer Oy. Jaaha, eipä olla ennen tavattu, mutta kylläpä ovat löytäneetkin soman yhdistelmän mikrohousuisista esittelijöistä ja mikroista. Joo'o. Di boäng.

No mikäs se siinä on? Pikkuinen softitalo ainokaisensa kanssa. Onkohan sekään vielä testattu, että toimii ja — nii'i. Neljä miestä ja pari ellia esittelemässä ja kaikki —

hyvin menee. Näkis vaan näidenkin naamat, kun ne viikon päästä polla punasena setvii, että mitä nää kaikki laskut on ja kuka maksaa. Pienistä puroista ne virratkin... Mahtaa olla kannattavaa puuhaa, voi viitsi.

Kertakaikkiaan mukavat markkinat. Näiden tarpeellisuutta, vaan joskus tulee miettineeksi ikään kuin vastuuntuntoisen tietotekniikkafirman kannalta asiaa pohtiessa. Joskus olen miettinyt, että olisikohan se toimiva markkinointitempaus, jos joku nimekkäistä älyläisi jäädä pois. Mene ja tiedä. Hauskaahan olla pittää.

Entäs nämä isot talot. Yhtenäisellä puvustuksella on yritetty peittää koko esillä oleva sillisalaatti yhtenäiseksi. Onpa sööttiä, mutta sääliksi käy sitä johtajaa ja sekreä, jotka täältä yrittävät ekan systeeminsä ostaa. Kopiokoneet ja mainfreimit, tekoälyt ja puhelinjärjestelmät samassa sumpussa, otapa siitä oma veikeä tietojärjestelmäsi. Sil' tapaa. Mutta...

Tuollahan on ekan työn- ja opinantajani ständi. Siellä on käytävä, sinne ne kerääntyy kaikki vanhat frendit kuka mistäkin — suuren suomalaisen siipien suojaan. On se kumma, kun ne sillä junttihuumorillaan saa pidettyä jenginsä kasassa ja hengen päällä. Täytyypä ängetä takahuoneeseen. Siellä tapahtuu aina. Sitäpaitsi, joku tuore juppitudumyyminties varmaan erehtyy taas tarjomaan koko ehtoon. Ja vaimokin se saa taas aiheen syventää kiintymystään.

Softaaja