

**ЎЗБЕКИСТОН АЛОҚА ВА АХБОРОТЛАШТИРИШ
АГЕНТЛИГИ**

**ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ
УНИВЕРСИТЕТИ
ФАРҒОНА ФИЛИАЛИ**

“Ахборот технологиялари” факультети

“Ахборот технологиялари” кафедраси

**“Операцион тизимлар ва офис иловалари”
ФАНИДАН**

МАЪРУЗАЛАР МАТНИ



Фарғона – 2012

ЎЗБЕКИСТОН АЛОҚА ВА АХБОРОТЛАШТИРИШ АГЕНТЛИГИ

**ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ
ФАРҒОНА ФИЛИАЛИ**

“ Ахборот технологиялари” факультети

“ Ахборот технологиялари ” кафедраси

**“ОПЕРАЦИОН ТИЗИМЛАР ВА ОФИС ИЛОВАЛАРИ”
Ф А Н И Д А Н**

МАЪРУЗАЛАР МАТНИ

ТАТУ Фарғона филиали
Услубий Кенгашида
тасдиқланган.
Баённома № 1
29 август 2013 й.

© ТАТУ Фарғона филиали - 2013

Ушбу маърузалар матни «5521900-Информатика ва ахборот технологиялари (тармоқлар бўйича)» бакалавр йўналиши талабалари учун яратилган бўлиб, унинг мақсади “Операцион тизимлар ва офис иловалари” фанидан назарий ва амалий билимлар беришга мўлжалланган. Олий ва ўрта махсус таълим вазирлигининг 2008 йил 23 август 263-сонли буруқ билан тасдиқланган намунавий дастур асосида тузилган.

Маърузалар матнини тайёрлаш жараёнида Тошкент ахборот технологиялари университети “АТДТ” кафедраси ўқитувчиси М.З.Бабамухамедованинг ушбу фан бўйича тайёрлаган маърузалар матнидан ҳам фойдаланилган.

Маърузалар матни “Ахборот технологиялари” кафедрасининг услубий семинарида кўриб чиқилган (Баённома № 1, 27 август 2013 йил).

Маърузалар матни ТАТУ Фарғона филиали Услубий Кенгашида кўриб чиқилган (Баённома № 1, 29 август 2013 йил).

Тузувчилар:

ТАТУ Фарғона филиали
“Ахборот технологиялари” кафедраси
катта ўқитувчи Ш.А. Умаров
ТАТУ “АТДТ” кафедраси
М.З.Бабамухамедова

Такризлар:

ТАТУ Фарғона филиали “Ахборот технологиялари” кафедраси доценти,
техника фанлари номзоди
Ф.Ю.Полвонов
ФарДУ “Амалий математика ва информатика” кафедраси доценти,
физика-математика фанлари номзоди
А.Мирзакулов

Кириш

Маълумки, ахборот технологияларининг инсон фаолияти ҳамма жабҳаларга кириб бориш жараёни борган сари ривожланиб бормоқда. Бу турли-туман ҳисоблаш техникасидан фойдаланувчилар сони ҳам кўпайиб бормоқдаки, бунда икки қарама-қарши тенденцияни ривожланиши кузатилмоқда. Бир томондан, ахборот технологиялари борган сари мураккаблашмоқда ва уларни қўллаш учун, ва уларни кейинги ривожланиши учун жуда чуқур билимлар талаб қилинади. Бошқа томонда, фойдаланувчиларнинг компьютерлар билан мулоқати соддалашмоқда. Компьютерлар ва ахборот тизимлари борган сари “дўстона” бўлиб бормоқда, ва ҳатто улар информатика ва ҳисоблаш техникаси соҳасида мутахассис бўлмаган одамлар учун ҳам тушунарли бўлиб бормоқда. Бу нарса, энг аввало фойдаланувчилар ва уларнинг дастурлари, ҳисоблаш техникаси билан махсус (тизимли) дастурий таъминот – операцион тизим орқали мулоқат қилганликлари учунгина юзага келди.

ОТ, фойдаланувчилар учун ҳам, бажарилувчи иловалар учун ҳам интерфейсни ташкил этади. Фойдаланувчилар ва кўпгина хизматчи дастурлар, ДТдан ҳамма дастурларда тез-тез учрайдиган амалларни бажаришни сўрайди. Бундай амалларга, биринчи навбатда, киритиш-чиқариш, бирор-бир дастурини ишга тушириш ва тўхтатиш, қўшимча хотира блокини олиш ёки уни бўшатиш ва бошқа кўпгина амаллар киради. Бундай амалларни ҳар сафар дастурлаш ва иккилик код сифатида дастур ичига бевосита киритиш мақсадга мувофиқ эмас балки, уларни биргаликда йиғиб дастурдан “сўров” орқали бажаришга бериш қулайдир. Бу ўз навбатида ОТ нинг муҳим функцияларидан биридир. Амалий дастурлар, ва кўпгина тизимли ишлов берувчи дастурлар, (м-н, дастурлаш тизимлари ёки маълумотларни бошқариш тизимлари) компьютер аппаратураси билан бевосита боғлана олмайди ва улар билан ОТ га мурожаат этиш орқали боғланадилар. Фойдаланувчилар учун ҳам, ОТ командасини киритиш билан ёки тизим таклиф қиладиган мумкин бўлган ҳаракатлар орқали ўз дастурлари ва компьютер билан мулоқат қиладилар. Бундай ўзаро мулоқат фақат ОТ орқалигина амалга оширилади. Бундай муҳим функцияни бажаришдан ташқари, ОТ ҳисоблаш ресурсларини самарали тақсимлаш ва ҳисоблашни ишончли ташкил этишга жавоб беради.

ОТ асосларини ва уларнинг ишлаш принципларини билиш, компьютердан самарали фойдаланишга олиб келади. ОТ ларни чуқур ўрганиш, авваламбор бу билимларни, дастурий таъминот яратишда ишлатишга имкон беради.

Албатта, бизнинг мамлакатимизда ҳозирги вақтда мураккаб ахборот тизимларини ишлаб чиқиш, замонавий ОТ ларда ишлашга мўлжалланган дастурлар, мажмуаси ва иловалар яратишлари жадал сураътда олиб борилмоқда. Шунинг учун ҳам ОТ ларни ва уларни ишлаш принциплари ва ҳисоблашларни ташкил этишни билиш зарурдир. Ҳозирги вақтда, компьютерда ишлаш камлик қилади, балки ҳисоблашларни ташкил этишни тушуниш шартдир.

1-Маъруза. Операцион тизим таърифи. ОТ ривожланиш босқичлари.

ОТ функциялари ОТ таърифи.

Ҳисоблаш тизимини мавқеини асосан унинг ОТ и белгилайди. Шунга қарамасдан, ҳисоблаш тизимидан фаол фойдаланувчилар, кўпинча, унга таъриф беришда анчайин қийналадилар. Бу нарса, қисман, ОТ бир-бири билан унга боғлиқ бўлмаган иккита функцияни бажариши билан боғлиқдир: бу фойдаланувчига, дастурчига кенгайтирилган, виртуал машина имкониятини яратиш билан қулайлик яратиш ва иккинчи компьютернинг ресурсларини рационал бошқариш билан ундан самарали фойдаланишни оширишдир.

ОТ кенгайтирилган машина сифатида.

Кўпгина компьютерлардан машина тиллари даражасида фойдаланиш анча мураккабдир, айниқса бу киритиш-чиқариш масалаларига тегишлидир. М-н, юмшоқ дискдан маълумотлар блокинни ўқишни ташкил этиш учун дастурчига 16 турли командалардан фойдаланишига тўғри келади, уларнинг ҳар бири 13 та параметрни аниқлашни талаб қилади, яъни масалан: дискдан блок тартиб рақами, йўлдаги сектор тартиб рақами ва ҳ.к.лар. диск билан бажариладиган амал тугалланиши билан, контроллер, таҳлил қилиниши керак бўлган хатолик мавжудлигини ва типларини кўрсатувчи 23 та қийматни қайтаради. Киритиш ва чиқариш масалаларини дастурлашни реал ҳақиқий муаммоларига чуқур эътибор бермаган ҳолда ҳам, дастурчилар орасида бу амалларни дастурлаш билан шуғулланишни хоҳловчилар тапилиши даргумондир. Диск билан ишлашда дастурчи-фойдаланувчига, дискни ҳар бири ўз номига эга бўлган файллар тўпламидан иборат деб тасаввур қилиш кифоядир.

Файл билан ишлаш, уни очиш, ўқиш ва ёзиш амалларини бажариш ва файлни ёпишдан иборатдир. М-н, бунда, частотали модуляцияни мукаммалаштириш ёки ўқийдиган механизм “головка” лари ҳолати, жойини ўзгариш каби саволлар фойдаланувчини безовта қилиши керак эмас. Дастурчидан қурилмалар (аппаратура) моҳиятини ҳаммасини яшириб, унга кўрсатилган файлларни қулай ва содда ўқиш, ёки ёзиш, кўришни имконини берадиган дастур- бу албатта ОТ дир. Худди шу каби, ОТ дастурчиларни диск жамламаси аппаратурасидан ажратиб, унга оддий файл интерфейсини тақдим этади, ва бу ҳолда ОТ узилишларни қайта ишлаш, таймерни ва оператив хотирани бошқариш ва талай шу каби қуйи даражадаги муаммолар билан боғлиқ ёқимсиз амалларни ўз зиммасига олади. Ҳар бир ҳолда, фойдаланувчи, реал аппаратура билан иш кўриш ўрнига мулоқот учун қулай ва соддадир. Бу нуқтаи назардан, ОТ фойдаланувчига маълум кенгайтирилган ёки виртуал машинани тақдим этадики, уни дастурлаш ҳам осон ва у билан ишлаш соддадир, албатта бу реал машина тақшил этадиган аппаратура билан бевосита ишлаш қулай ва енгилдир.

ОТ ресурсларни бошқарувчи сифатида.

ОТ, авваламбор фойдаланувчига қулай интерфейс яратувчидир деган гоё албатта, масалани юқоридан пастга қараб назар солишга мос келади.

Бошқа нуқтаи назар, яъни пастдан юқорига қараб назар ташлаш, бу ОТ га мураккаб тизимнинг ҳамма қисмларини бошқарувчи механизм назар солишдир.

Замонавий ҳисоблаш тизимлари, процессорлар, хотира, таймерлар, дисклар, жамғармалар, тармоқ коммуникацияқурилмалари, принтерлар ва бошқа қурилмалардан иборатдир. Иккинчи ёндашишга мос равишда ОТ нинг функцияси, процессорлар, яъни ресурсларни рақобатдош жараёнлар орасида тақсимлашдан иборатдир. ОТ ҳисоблаш машина ресурсларини жамисини шундай бошқариш керакки, уни ишлаши максимал самарадорликни таъминлаши зарурдир. Самарадорлик кўрсаткичи, м-н, тизим ўтказувчанлик қобилияти ёки реактивлиги бўлиши мумкин.

Ресурсларни бошқариш, масала ресурси типига боғлиқ бўлмаган иккита умумий масалани ечишни ўз ичига олади:

- ресурсни режалаштириш –яъни берилган ресурсни кимга, қачон ва тақсимлашдан иборатдир;
- ресурс ҳолатини кузатиш – ресурсни банд ёки бўшлиғи, бўлинадиган ресурслар ҳақида эса ресурснинг қанча қисми эса тақсимланмаганлиги ҳақидаги оператив маълумотни олиб туришдан иборатдир.

Ресурсларни бошқаришни умумий масаласини ечишда, турли ОТ лар турли алгоритмлардан иборатдир, бу эса ўз навбатида ОТ ларни умумий қиёфаси, унумдорлик характеристикалари, қўлланилиш соҳалари ва ҳатто фойдаланувчи интерфейсини юқори даражада ОТ вақтни бўлиш тизими, пакетли ишлов бериш тизими ёки реал вақт тизимига мутаносиблигини белгилайди.

Тизимли дастури таъминот (ДТ) энг қуйи дастурий таъминотдир. Бундай дастурий таъминотга қуйидагилар киради: Операцион тизим-ОТ, файлларни бошқарув тизимлари, ОТ билан фойдаланувчи мулоқоти учун интерфейс қобиклари, дастурлаш тизимлари, утилиталар.

Операцион тизим-бу тизимли бошқарувчи дастурларнинг зарурий маълумот массивлари билан тартибга солинган кетма-кетлигидир. У фойдаланувчи дастурларининг бажарилиши ва режелаштириш, ҳисоблаш тизимларининг барча ресурсларини (дастурлар, маълумотлар, аппаратура ва бошқа тақсимланадиган ва бошқариладиган объектларини), фойдаланувчига улардан самарали фойдаланиш имконини берадиган ва маълум маънода ҳисоблаш машинаси терминларида тузилган масалаларни ечишга мўлжалланган.

ОТ махсус дастур ва микродастурлардан иборат бўлиб, улар аппаратурадан фойдаланиш имконини таъминлайди. Амалий дастурий таъминот албатта ОТ бошқаруви остида ишлайди.

ОТлар асосий функциялари:

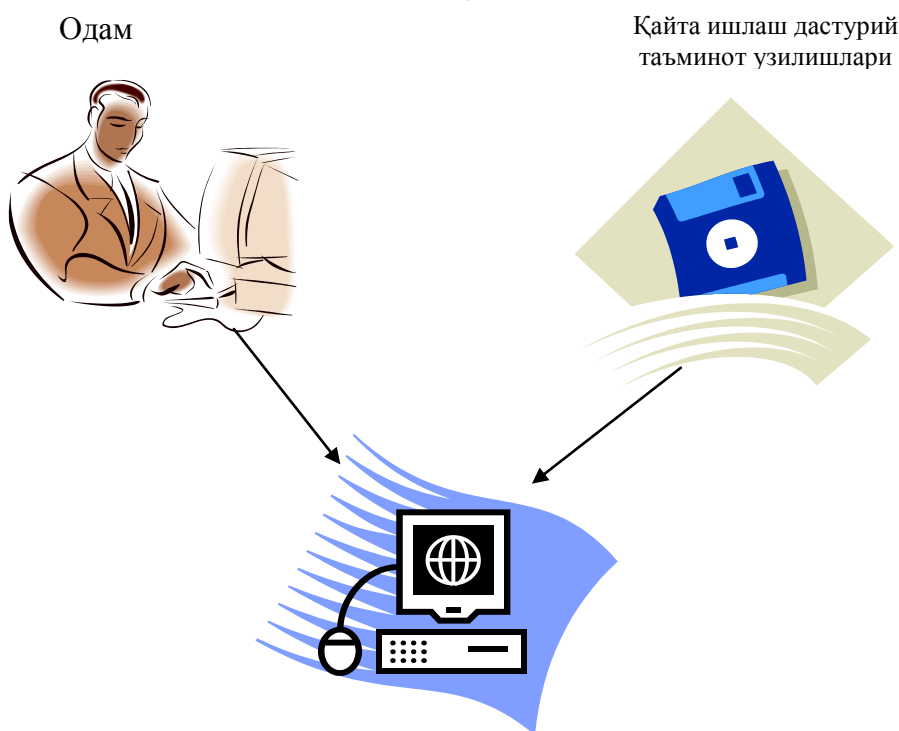
- фойдаланувчидан (ёки тизим операторидан) маълум тилда тузилган команда ёки топшириқларни қабул қилиш ва уларга ишлов бериш.

Топшириқлар операторлар, матн кўрсатмалари (директивалар) ёки манипулятор (м-н сичқонча ёрдамида) бажариладиган кўрсатмалар ёрдамида берилади. Бу командалар, авваламбор, дастурларни ишга тушириш (тўхтатиш, тўхтатиб туриш) билан боғлиқдир, файллар устидаги амаллар (жорий каталогда файллар рўйхатини олиш, у ёки бу файлни яратиш, номини ўзгартириш, нусхасини олиш, жойини ўзгартириш ва х.к.) билан боғлиқдир, умуман олганда бошқа командалар ҳам мавжуддир;

- ижро қилиниши керак бўлган дастурларни оператив хотирага юклаш.
- хотирани бошқариш, аксари барча замонавий тизимларда эса виртуал хотирани ташкил этиш;
- барча датсур ва маълумотларни идентификация қилиш;
- дастурларни ишга тушириш (унга бошқарувни узатиш, натижада процессор дастурни бошқаради);
- бажарилаяпган иловалардан келаяпган турли сўровномаларни қабул қилиш ва бажариш. ОТ жуда кўп сонли тизимли функцияларни (сервисларни) бажара олиши мумкин, улар бажарилаяпган иловалардан сўралиши мумкин. Бу сервисларга мурожаатлар маълум қоидаларга мос равишда амалга оширилиши мумкин, бу эса ўз навбатида бу ОТнинг амалий дастурлаш интерфейсини аниқлайди (Application Program Interface, API);
- барча киритиш-чиқариш амалларига хизмат қилади;
- файлларни боқшариш тизимлари (ФБТ) ишини ва/ёки маълумотлар базасини бошқариш тизимлари (МББТ) ишини таъминлаш, бу эса ўз навбатида бутун дастурий таъминот самарасини кескин равишда оширади;
- мультидастурлаш режими, таъминлаш, яъни битта ёки бир нечта дастурларни битта процессорда параллел бажарилишни ташкил этиш-бу эса уларни бир вақтда бажарилиши тасаввурини ҳосил қилади;
- берилган хизмат қилиш дисциплиналари ва стратегияларига асосан масалаларни режалаштириш ва диспечерлаштириш;
- бажарилаяпган дастурлар орасида маълумотлар ва маълумотлар алмашиш механизминини ташкил этиш;
- тармоқ ОТ лари учун, боғланган компьютерлар орасидаги мулоқотни таъминлаш функциясидир;
- битта дастурни бошқа дастур таъсиридан ҳимоя қилиш, маълумотларни сақланишини таъминлаш, операцион тизимни ўзини компьютерда бажарилаяпган иловалардан ҳимоялаш;
- фойдаланувчиларни аутентификация ва муаллифлаштириш(кўпгина диаллогли ОТ учун). Аутентификация – фойдаланувчи номи ва паролени қайд ёзувидаги қийматга мослигини текшириш. Агар фойдаланувчи кириш номи (login) ва унинг пароли мос келса, демак у ўша фойдаланувчидир. Авторлаштириш (муаллифлаштириш) дегани, аутентификациядан ўтган фойдаланувчига маълум ҳуқуқ ва имтиёзлар

берилиб, у компьютерда нима қила олиши мумкин ёки нима қила олмаслигини аниқлайди;

- реал вақт режимида жавоб бериш вақти қатъий чегаралирини кондиради;
 - фойдаланувчилар ўз датурларини ишлаб чиқишда фойдаланадиган дастурлаш тизими ишини таъминлаш;
 - тизимни қисман ишдан чиқиши ҳолатида хизмат кўрсатиш;
- ОТ, компьютер аппарат таъминотини фойдаланувчилар амалий дастурларидан ажратади. Фойдаланувчи ҳам, унинг дастури ҳам компьютер билан ОТ интерфейс орқали ўзаро алоқада бўлади. Буларни қуйидаги расмда кўрсатиш мумкин.



Расм. Фойдаланувчи ва унинг дастурлари компьютер билан ОТ орқали ўзаро алоқаси.

ОТ ривожланиш босқичлари. Биринчи давр (1945-1955 йиллар).

Ҳаммага маълумки, компьютер инглиз математиги Чарльз Бэббид томонидан 18-аср охирида кашф этилди. Унинг “аналитик машина”си ҳақиқатда ишлай олмади, чунки у вақтдаги технологиялар ҳисоблаш техникаси учун зарур бўлган аниқ механика деталларини тайёрлаш бўйича зарур талабларни қондирадиган технологиялар мавжуд бўлмаган. Яна энг асосий нарса, у вақтда компьютер операцион тизимга эга бўлмаган.

Рақамли ҳисоблаш машиналарини яратишда, иккинчи жаҳон урушидан кейин маълум прогресс-ривожланиш юз берди. 40 йиллар ўрталарида 1-чи лампали машиналар яратилди. У вақтда айтиб бериш мутахассислар ҳисоблаш машиналарини ҳам лойиҳалашда, ҳам эксплуатация қилишда ва дастурлашда ҳам шу гуруҳ мутахассислари иштирок этганлар. Бу жараён

кўпроқ, компьютердан инструмент-ускуна сифатида турли амалий соҳалар масалаларини ечишда фойдаланиш эмас, балки ҳисоблаш техникаси соҳасидаги илмий-тадқиқот ишига яқинроқ эди.

Дастурлаш фақат машина тилида амалга оширилар эди. ОТ тўғрисида гап ҳам йўқ эди, чунки ҳисоблаш жараёни ташкил қилиш масалалари, ҳар дастурчи томонидан бошқарув пульти орқали “қўлда” ечилар эди.

Пульт олдида фақат битта фойдаланувчи ўтириш мумкин эди. Дастур машина хотирасига энг яхши ҳолатда перфокарта колодасидан киритилар эди, одатда эса ўтказиш панели (панель переключателей) ёрдамида юкланар эди.

Ҳисоблаш тизими бир вақтнинг ўзида фақат битта операцияни (киритиш-чиқариш ёки ҳисоблашлар) бажарар эди. Дастурни созлаш бошқариш панелидан хотира ва машина регистри ҳолатини ўрганиш ёрдамида олиб борилар эди. Бу давр охирида биринчи тизимли дастурий таъминот юзага келди; 1951-1952 йиллар символли тиллар (Fortran ва бошқ) дан биринчи компиляторлар версиялари юзага келди, 1954й эса IBM-701 учун Ассемблер ишлаб чиқилди.

Вақтнинг энг кўп қисми дастурни ишга туширишга кетиб қолди, дастурларнинг ўзи эса қатъий равишда кетма-кет ишлов бериш режими деб аталди.

Хулоса қилиб айтганда, биринчи давр, ҳисоблаш тизимларининг юқори нарҳи, уларнинг сони камлиги ва фойдаланишнинг паст самарали билан белгиланди.

Иккинчи давр (1955-1965 йиллар).

50 йил ўрталарига келиб, ҳаммага маълумки янги техник база-ярим ўтказгич элементларни юзага келиши билан, ҳисоблаш техникаси ривожланишида янги давр бошланди. Иккинчи авлод компьютерлари ишончлироқ бўлиб қолди, чунки улар амалий муҳим масалаларни бажариш даражасида узлуксиз равишда узоқ ишлай оладиган имкониятга эга бўлдилар. Айнан шу даврда ҳисоблаш техникаси билан ишлайдиган мутахассислар-дастурчилар, операторлар, эксплуатациячилар ва ҳисоблаш машинасини ишлаб чиқарувчиларга ажралдилар.

Шу йилларда биринчи алогритмик тиллар юзага келди ва натижада биринчи тизимли дастурлар-компиляторлар ҳам яратилди. Процессор вақти қиймати (нарҳи) ошди, бу эса дастурлар орасидаги вақтни қисқартиришни талаб қилди.

Биринчи пакетли ишлов бериш тизимлари юзага келди, бу тизимларда дастурларни ишга тушириш кетма-кетлигини автоматлаштирилди ва шу билан бирга процессор юкланиш коэффиценти ошди. Пакетли ишлов бериш тизимларини замонавий ОТ ларининг биринчи вариантлари дейиш мумкин, чунки улар ҳисоблаш тизимини бошқаришга мўлжалланган биринчи тизимли дастурлар эди.

Пакетли ишлов бериш тизимларини амалга оширишда, топшириқларни бошқариш формаллашган тили ишлаб чиқилди, унинг ёрдамида дастурчи

тизимга ва операторга ҳисоблаш машинасида қайси ишни бажармоқчи эканлиги ҳақида маълумот беради. Бир нечта топшириқлар мажмуаси, қоида бўйича перфокарталар “колода”си кўринишида бўлиб топшириқлар пакети номини олди.

Учинчи давр (1965-1980 йиллар).

Ҳисоблаш машиналари ривожланишида кейинги муҳим даври шу йилларга тўғри келади. Бу вақтда, техник базада қуйидаги ўзгаришлар юз берди: алоҳида ярим ўтказгич элементлардан (транзистор типидagi) интеграл микросхемаларга ўтилди, бу эса янги учинчи авлодга, янги имкониятлар яратди. Бу даврнинг ўзига хос хусусиятларидан бири, интеграл микросхемаларда яратилган биринчи дастурий-мутаносиб машиналардир, яъни IBM/360 машиналари яериясидир. 60- йиллар бошида яратилган бу машиналар оиласи иккинчи авлод машиналаридан баҳо/унумдорлик кўрсаткичи бўйича олдинга анчагина ўтиб кетди. Тезда, дастурий-мутаносиб машиналар ғоясини умум тан олинди.

Дастурий мутаносиблик ОТ ларни ҳам мутаносиблигини талаб қилди. Бундай операцион тизимлар ҳам катта ЭҲМ да ҳам, кичик ҳисоблаш тизимларида ҳам, турли периферик қурилмаларнинг кам сони ва кўп сони билан ҳам, тижорат соҳасида ҳам, илмий-тадқиқот соҳаларида ҳам ишлай олиши керак.

Шундай ҳамма қарама-қарши талабларни қондирадиган асосда қуриладиган операцион тизимлар жуда мураккаб “монстр”лар бўлиб чиқди. Улар кўп миллионли ассемблер қаторларидан иборат булиб, минглаб дастурчилар томонидан ёзлиган бўлиб, минглаб хатоларни ўз ичига олади, улар минглаб тузатишларга олиб келади. Операцион тизимнинг ҳар бир янги версиясида бирор хатолар тузатилиб, янги юзага келди. Кўпгина муаммолар ва жуда катта ўлчамга қарамасдан OS/360 ва унга ўхшаш 3-чи авлод операцион тизимлари ҳақиқатдан ҳам истеъмолчиларнинг кўпгина талабларини қондирдилар. Бу авлоднинг энг катта эришган ютуқларидан бир мультидастурлашни амалга оширишдир.

Мультидастурлаш – бу ҳисоблаш жараёнининг ташкил қилиш усули бўлиб, битта процессорда навбат билан бир нечта дастур бажарилади.

Битта дастур киритиш-чиқаришни амалга оширгунча кенг дастурларни олдинги кетма-кет бажарилишдаги каби (бир дастурли режим), процессор тўхтаб турмайди, балки бошқа дастурни бажаради (кўп дастурли режим). Бунда ҳар бир дастур оператив хотирадаги бўлим деб аталувчи ўз қисмига юклайди.

Бошқа янгилик – спулинг (spooling) деб аталади. Спулинг у вақтда ҳисоблаш жараёнини ташкил этиш усулларида бири бўлиб, унга мос равишда топшириқ перфокартадан дискга ҳисоблаш марказида пайдо бўлиш тартибида ёзилади, кейин эса навбатдаги топшириқ тугалланиши билан, янги топшириқ дискдан бўшаган бўлимга юкланади.

Пакетли ишлов беришни мультидастурлашли амалга ошириш билан бирга, ОТларнинг янги типи –вақтни ажратиш тизимлари юзага келди. Ажратилган вақт тизимларида қўлланиладиган мультидастурлаш варианты,

хар бир фойдаланувчи учун ҳисоблаш машинасидан ягона фойдаланиш тасаввурини ҳосил қилишга имкон беради.

Мультидастурлашни юзага келиши х.т. тузилишига чуқур ўзгартиришлар киритишни талаб қилади. Бунда асосий ролни аппарат томонидан қулланиш катта рол ўйнайди, унинг асосий хусусиятлари қуйида келтирилган:

- Химоя механизмини амалга ошириш. Дастурлар мустақил равишда ресурсларни тақсिलाш имконига эга бўлиши керак эмас, бу имтиёзли ва имтиёзсиз командаларни келиб чиқди. Имтиёзли командалар ОТ томонидан бажарилади.
- Узилишлар мавжудилиги. Ташқи узилишлар ОТ ни асинхрон ходиса, м-н кириш-чиқиш операцияси тугалланганлиги ҳақида огохлантиради. Ички узилиш, ОТ аралашуви зарур бўлганда юз беради, м-н химояни бузишга ҳаракат ёки нолга бўлиш.
- Архитектурада параллелизмни ривожлантириш. Хотирага бевосита мурожаат ва кириш-чиқиш каналини ташкил этиш, марказий процессорни қийин операцияларни бажаришдан халос этади.

Албатта, мультидастурлашни ташкил этишда ОТ роли жуда муҳимдир.

У қуйидаги операциялар учун жавоб беради:

- Тизимли чақириқлар ёрдамида ОТ ва амалий дастурлар орасида интерфейсни ташкил этиш
- Хотирадаги топшириқлардан навбат ташкил этиш ва топшириқ учун процессор ажратиш учун процессордан фойдаланишни режалаштириш
- Бир топшириқдан иккинчисига ўтиш, ҳисоблашларни тўғри ташкил этиш учун контекстни сақлаш
- Хотира чегараланган ресурс бўлгани учун, хотирани бошқариш стратегияси зарур, яъни хотирадан маълумотларни олиш, жойлаштириш ва алмаштириш жараёнларини тартибга солиш талаб қилинади.
- Маълумотларни ташқи жамламаларда файл кўринишида сақлашни ва маълум файлларни фақат аниқ фойдаланувчилар фақат аниқ фойдаланувчилар фойдалана олишни ташкил этиш.
- Дастурларга санкцияли маълумот алмашиш талаб этилгани учун, уларни коммуникация воситалари билан таъминлаш зарур.
- Маълумотларни тўғри тақсимлаш учун, зиддиятли ҳолатларни ечишга тўғри келади, бу кўпинча турли ресурслар билан ишлашда рўй беради, шунинг учун ҳаракатларни дастурлар билан синхронлаштириш.

Вақт ажратиш тизимларида фойдаланувчи, дастурни интерактив режимда созлаш имконига эга бўлди, бунда у маълумотли дискга перфокарта орқали эмас, бевосита клавиатурадан киритиш мумкин бўлди. On-line файлларни юзага келиши ривожланган файл тизимларини ишлаб чиқиш заруриятини келтириб чиқарди.

4-чи давр (1980дан – ҳозирги вақтгача).

Операцион тизимлар ривожланишидаги кейинги давр катта интеграл схемаларни (БИС) юзага келиши билан боғлиқ бўлган даврдир. Бу йилларда интеграция даражаси кескин ўсиши ва микросхемалар арзонлашиши юз берди. Компьютердан алоҳида фойдаланувчилар фойдаланиши имкони юзага келди, ва шахсий компьютерлар даври бошланди.

Архитектура жиҳатидан, шахсий компьютерлар, миникомпьютерлар типлари синфларидан ҳеч нарсаси билан фарқ қилмас эдилар, фақат уларнинг баҳоларида фарқ бўлди. Агар миникомпьютер корхона ва университет бўлимига шахсий ҳисоблаш марказига эга бўлишига имкон берган бўлса, шахсий компьютер эса бундай имкониятни алоҳида инсон учун яратди.

Компьютерлар ҳисоблаш техникаси соҳасида мутахассис бўлмаганлар ҳам кенг кўламда фойдалана бошладилар, бу эса ўз навбатида “дўст” дастурий таъминотни яратишни талаб этди, бу дастурчиларни алоҳида ўрнидан қўзгатди.

Операцион тизимлар бозорида иккита тизим устунлик қила бошладилар: MS-DOS ва UNIX ОТ лари. Бир фойдаланувчили MS-DOS ОТ лари Intel 8088 асосида қурилган микропроцессорлар, ва кейин 80286, 80386 ва 80486 асосида қурилган компьютерларда фойдаланилди.

Мультидастурли, кўпфойдаланувчили UNIX ОТ и интел-бўлмаган компьютерлар муҳитида устунлик қила бошлади, айниқса юқори унумдорликка эга бўлган RISC-процессорлар учун.

80-йиллар ўрталарида, тармоқ ёки тақсимланган ОТ лар бошқарувчи остида ишлайдиган шахсий компьютерлар кескин тарзда ривожлана бошлади.

Тармоқ ОТ ларида, фойдаланувчи тармоқда бошқа компьютерлар мавжудлиги ҳақида билишлари ва бошқа компьютерга унинг ресурсларидан, асосан файлларидан фойдаланиш учун бошқа компьютерга мантиқан киришлари керак.

Тармоқдаги ҳар бир машина, компьютернинг автоном операцион тизимидан тармоқда ишлашга имкон берадиган қўшимча воситаларга эга бўлган, локал операцион тизимини бажаради.

Тармоқ ОТи, бир процессорли компьютер ОТидан асосли фарқ қилмайди. Уларнинг таркибида, албатта, тармоқ интерфейсини қўлловчи (тармоқ адаптери драйвери) ва шу билан бирга тармоқдаги бошқа компьютерларга масофадаги кириш воситалари ва масофадаги файлларга мурожаат воситалари мавжуддир, аммо бу қўшимчалар операцион тизимни структурасини тубдан ўзгартирмайди.

Ҳисоблаш тизимларини ривожланиши босқичларини кўриб чиқиб, биз ривожланиш жараёнида мумтоз (классик) ОТ лар бажарган 6 та асосий функцияларни ажратишимиз мумкин:

- Топшириқларни режалаштириш ва процессордан фойдаланиш;
- Дастурларни коммуникация ва синхронизация воситалари билан таъминлаш;
- Хотирани бошқариш;

- Файл тизимини бошқариш;
- Киритиш-чиқаришни бошқариш;
- Хавфсизликни таъминлаш.

Ҳар бир келтирилган функциялар одатда ОТ таркибидаги компоненталаридан бири сифатида амалга оширилган. Улар бошиданок, ОТ таркибий қисми сифатида яратилган эмас, улар ривожланиш жараёнида юзага келди. Инсон яратган ҳисоблаш тизими ривожланиши (эволюцияси) шу йўлдан кетди, аммо ҳеч ким бу йўл ривожланишнинг ягона мумкин бўлган йўли деб исбот қилолмайди.

ОТ лар, шу айти вақтда уларнинг мавжудлиги-ҳисоблаш тизимидан оқилона фойдаланишдир, шунинг учун ҳам улар мавжуд.

2-маъруза. Ҳисоблаш тизимлари таркибий қисмлари.

1. Ҳисоблаш тизими (компьютер ва тармоқлар) таркиби.

Ҳисоблаш тизими таркиби конфигурация деб аталади. Одатда ҳисоблаш техникасининг аппарат ва дастурий воситалари алоҳида олиб ўрганилади. Шунинг учун ҳам мос равишда ҳисоблаш тизимлари аппарат конфигурацияси ва дастурий конфигурацияси алоҳида олиб ўрганилади. Бундай бўлиниш ахборот технологиялари учун муҳим аҳамиятга эгадир, чунки кўп ҳолларда алоҳида олинган масала ечимини ҳам аппарат, ҳам дастур воситалари ёрдамида таъминлаш мумкин.

1.1. Аппарат таъминоти.

Ҳисоблаш тизимларининг аппарат таъминоти таркибига, аппарат конфигурацияни ташкил этувчи қурилма ва асбоблар киради. Замонавий компьютер ва ҳисоблаш мажмуа (комплекс)лари блок-модулли конструкция (тузилиш)дан иборат. Маълум ишларни бажаришга зарур бўлган аппарат конфигурацияни тайёр блок ва қисмлардан йиғиб олиш мумкин.

Қурилмаларнинг, марказий процессорга (Central Processing UNIT, CPU) нисбатан жойлашишига қараб ташқи ва ички қурилмаларга ажратамиз.

Ташқи қурилмалар, қоида бўйича, маълумотларни киритиш ва чиқариш қурилмалари-дир, уларни одатда периферик қурилмалар ҳам деб аталади. Бундан ташқари маълумотларни узоқ сақлашга мўлжалланган қурилмалар ҳам ташқи қурилмаларга киради.

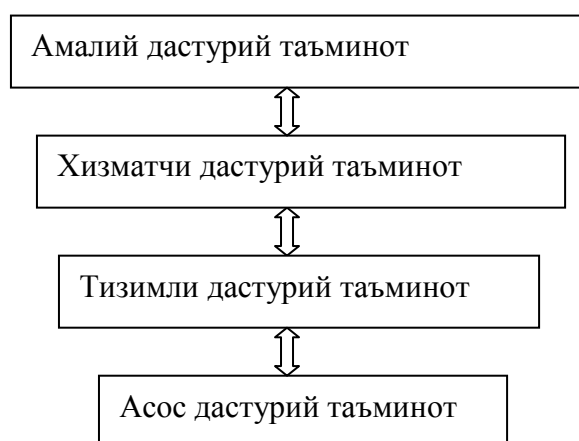
Алоҳида блок ва қисмлар орасидаги келишувчанлик, биргаликда ишлашдаги мосланувчанлик, аппаратли интерфейс деб аталувчи ўтиш аппарат-мантикий қурилмалари ёрдамида бажарилади. Ҳисоблаш техникасидаги аппарат интерфейсига белгиланган стандартлар протоколлар дейилади. Шундай қилиб, протокол – бу қурилма яратувчилари томонидан, бу қурилманинг бошқа қурилмалар билан муваффақиятли ва келишилган ҳолда биргаликда ишлаши учун, ишлаб чиқиладиган техник шартлар мажмуасидир.

1.2. Дастурий таъминот

Дастур – буйруқларнинг тартибланган кетма-кетлигидир. Компьютер учун тузилган ҳар дастур вазифаси – аппарат воситаларни бошқаришдир. Биринчи қарашда дастурнинг қурилмалар билан ҳеч қандай боғлиқлиги йўқдек кўринади, яъни масалан, дастур киритиш қурилмаларидан маълумот киритишни ва чиқариш қурилмаларига ҳам маълумот чиқаришни талаб қилмаса ҳам, барибир унинг иши компьютернинг аппарат қурилмаларини бошқаришга асосланган.

Компьютерда, дастурий ва аппарат таъминот, доимо узилмас алоқада ва узлуксиз боғланишда ишлайди. Биз бу икки категорияни алоҳида кўриб чиқаётганимизга қарамасдан, улар орасида диалектик алоқа мавжудлиги ва уларни алоҳида кўриб чиқиш шартли эканлигини эсдан чиқармаслик керак.

Компьютерлар ва ҳисоблаш тизимларининг дастурий таъминоти тузилишини дастурий конфигурация деб ҳам аталади. Дастурлар орасида худди компьютернинг физик қисмлари орасидаги каби ўзаро алоқа мавжуд. Аксарият кўпгина дастурлар, қуйроқ даражадаги бошқа дастурларга таяниб ишлайди. Бундай боғланиш дастурлараро интерфейс дейилади. Бундай интерфейс (мулоқот) нинг мавжудлиги техник шартлар ва ўзаро алоқа қоидаларига асосланган бўлса ҳам, амалда у дастурий таъминотни ўзаро алоқада бўлган бир нечта сатҳлар (даража)ларга тақсимлаш билан таъминланади. Дастур таъминоти сатҳлари пирамида тузилишига эгадир. Ҳар бир кейинги сатҳ олдинги сатҳлар дастурий таъминотига таянади. Бундай ажратиш, ҳисоблаш тизимининг дастурларни ўрнатишдан бошлаб, то амалда эксплуатация қилиш ва техник хизмат кўрсатишгача бўлган иш фаолиятининг ҳамма босқичлари учун қулайдир. Шунга алоҳида этибор бериш керакки, ҳар бир юқоридаги сатҳ бутун тизимнинг функционаллигини оширади. Масалан, асос дастурий таъминоти сатҳига эга бўлган ҳисоблаш тизими кўп функцияларни бажара олмайди, аммо у тизимли дастурий таъминотни ўрнатишга имкон беради, яъни шароит яратади.



Расм 1.

Асос дастурий таъминоти. Дастурий таъминотнинг энг қуйи сатҳи-асос дастурий таъминотидан иборатдир. Бу таъминот асос аппарат воситалари билан алоқага жавоб беради. Қоида бўйича, асос дастурий воситалари бевосита асос қурилмалари таркибига киради ва доимий хотира деб аталадиган махсус микросхемаларда сақланади. Дастур ва маълумотлар доимий хотира (ДХ) микросхемаларига уларни ишлаб чиқиш вақтида ёзилади ва уларни ишлаш жараёнида ўзгартириш мумкин эмас.

Амалда, ишлатиш вақтида асос дастурий воситаларини ўзгартириш зарурияти келиб чиқса, ДХ микросхемаси ўрнига қайта дастурлаш имконига эга бўлган доимий хотира қурилмасидан фойдаланилади. Бу ҳолда ДХ мазмунини ҳисоблаш тизими таркибида бевосита ўзгартириш мумкин (бундай технология флеш технология деб аталади), ёки ҳисоблаш тизимидан ташқарида, махсус программатор деб аталадиган қурилмаларда бажарилади.

Тизимли дастурий таъминот. Бу сатҳдаги дастур, компьютер тизимининг бошқа дастурлари ва бевосита аппарат таъминоти билан ўзаро боғланишни таъминлайди, яъни бу дастурлар даллоллик вазифасини ўтайди.

Бутун ҳисоблаш тизимининг эксплуатация (ишлатиш) кўрсаткичлари иш сатҳининг дастурий таъминотига боғлиқдир.

Масалан, ҳисоблаш тизимига янги қурилма улаш вақтида бошқа дастурларни шу қурилма билан боғланишини таъминлаш учун, тизимли даражада дастур ўрнатилиши керак. Аниқ қурилмалар билан ўзаро боғланишга жавоб берувчи дастурлар қурилма драйверлари дейилади ва улар тизимли сатҳ дастурий таъминоти таркибига киради.

Тизимли сатҳ дастурларининг бошқа синфи фойдаланувчи билан боғланишга жавоб беради. Айнан шу дастурлар ёрдамида фойдаланувчи, ҳисоблаш тизимига маълумотларни киритиш, уни бошқариш ва натижаларни ўзига қулай кўринишда олиш имконига эга бўлади. Бундай дастурий воситалар, фойдаланувчи интерфейсини таъминлаш воситалари деб аталади. Компьютерда ишлаш қулайлиги ва иш жойи унумдолиги бу воситалар билан бевосита боғлиқдир. Тизимли сатҳ дастурий таъминоти мажмуаси компьютер операцион тизими ядросини ташкил этади. Операцион тизимнинг тўлиқ тушунчасини биз кейинги бобларда кўриб ўтамиз, бу ерда эса биз фақат, агар компьютер тизимли сатҳ дастурий таъминоти билан жиҳозланган бўлса, у ҳолда компьютер янада юқори даражадаги дастурни ўрнатишга ва энг асосийси дастурий воситаларнинг қурилмалар билан ўзаро алоқасига тайёрлигини билдиради. Яъни операцион тизим ядроси мавжудлиги – инсонни ҳисоблаш тизимида амалий ишларни бажариш имкониятининг зарурий шартидир.

Хизматчи дастурий таъминот. Бу дастурлар ҳам асос дастурий таъминот билан, ҳам тизимли дастурий таъминот дастурлари билан боғланган. Хизматчи дастурларнинг асосий вазифаси (уларни утилиталар деб ҳам аталади) компьютер тизимини текшириш, сошлаш ва тузатишдан иборатдир. Кўп ҳолларда улар, тизимли дастурларнинг функциясини кенгайтиришга ва яхшилашга мўлжаллангандир. Баъзида, бу дастурлар, бошиданок ОТ таркибига киритилган бўлиши мумкин, баъзида эса улар ОТ функциясини кенгайтиришга хизмат қилади.

Амалий дастурий таъминот. Бу сатҳ дастурий таъминоти, мазкур иш жойида аниқ масалаларни ечишга ёрдам берадиган амалий дастурлар мажмуасини ташкил этади. Бу масалалар қамраб олган соҳалар жуда кўп бўлиб, улар ишлаб чиқариш, илмий-техник, ижод, ўқитиш ва дам олишга мўлжалланган масалаларини ўз ичига олади. Бу дастурлар кўпфункционаллиги сабаби, инсон фаолияти ҳар хил соҳалари учун амалий дастурлар ва иловалар мавжудлигидир.

Демак амалий дастурий таъминот ва тизимли дастурий таъминот ўртасида ўзаро бевосита алоқа бор экан (биринчиси иккинчисига таянади), ҳисоблаш тизими универсаллиги, амалий дастурий таъминот оммавийлиги ва компьютер функционал имкониятлари кенг қўламлиги фойдаланилаётган операцион тизим типи, унинг ядроси қандай тизимли воситаларни ўз ичига

олганлиги ва у уч томонлама ўзаро боғланиш, яъни инсон – дастур – қурилма боғланишни қай тарзда таъминлашига бевосита боғлиқдир.

1.3. Амалий дастурий воситаларни синфларга ажратиш.

Матн редакторлари (тахрирлагичлар). Амалий дастурларнинг бу синфи дастурларнинг асосий функцияси матнли маълумотларни киритиш ва тахрирлашдан иборатдир. Қўшимча функциялари эса киритиш ва тахрирлаш жараёнини автоматлаштиришдир. Маълумотларни киритиш, чиқариш ва сақлаш учун, матн редакторлари тизимли дастурий таъминотни чақиради ва ундан фойдаланади. Аммо бу ҳолат ихтиёрий амалий дастурлар учун ҳам хосдир.

Компьютер тизими билан ўзаро мулоқатда бўлиш кўникмаларини ҳосил қилишда ва амалий дастурий таъминот билан танишишда, ишни одатда матн редакторларидан бошлашади.

Матн процессорлари. Матн процессорларининг редакторлардан фарқи шундаки, улар матнни киргизиб, тахрирлабгина қолмай, балки уни форматлайди ҳам. Мос равишда матн процессорлари асосий воситаларига (графика, жадвал) натижавий ҳужжатни ташкил этувчилари – матн, графика, жадвал ва бошқа объектлар ўзаро алоқаларини таъминлаш воситалари киради, қўшимчаларига эса-форматлаштириш жараёнини автоматлаштириш воситалари киради.

Ҳужжатлар билан ишлашнинг замонавий услуби (стиль) иккита альтернатив ёндошишни – қоғоздаги ҳужжатлар ва электрон ҳужжатлар (қоғозсиз технология) билан ишлашни кўзда тутади.

Шунинг учун ҳам, матн процессорлари воситалари билан ҳужжатларни форматлаш тўғрисида гапирилганда, иккита ҳар хил йўналишлар – босмадан чиқаришга мўлжалланган ҳужжатларни форматлаш ва экранда акс эттиришга мўлжалланган электрон ҳужжатларни форматлаш кўзда тутилади. Бу йўналишлар усул ва методлари бир-биридан тубдан фарқ қилади. Мос равишда, матн процессорлари ҳам бир-биридан фарқ қилади, аммо уларнинг кўплари ўзида бу икки йўналишни бирлаштиради.

График редакторлар. Бу синфга хос дастурлар график тасвирларни қайта ишлаш ва (ёки) яратишга мўлжалланган. Бу синфда қуйидаги категориялар мавжуд: растрли редакторлар, векторли редакторлар ва уч ўлчамли графика билан ишловчи дастурий воситалар (3D-редакторлар).

Растрли редакторлар, график объект, растрни ташкил этувчи нуктлар комбинацияси кўринишида берилган бўлса, бу тасвирларда ранглар ва ёрқинлик асосий ролни ўйнайди. Бундай ёндошиш, график тасвир ҳар хил ёрқинликда бўлса ва объект элементлари ранги тўғрисидаги маълумот унинг формаси тўғрисидаги маълумотдан аҳамиятли бўлган ҳолларда самаралидир. Бундай хусусиятлар кўпроқ фотография ва полиграфия тасвирларига хосдир. Растр редакторлари тасвирларга ишлов беришда, фотоэффект ва бадиий композицияларни яратишда кенг қўлланилади.

Векторли редакторлар, растрлилардан тасвирлар тўғрисидаги маълумотларни тасвирлаш усули билан фарқ қилади. Векторли тасвирнинг элементар объекти нукта эмас, балки чизикдир. Бундай ёндашиш чизма графика ишлари учун хосдир. Бу ҳолда тасвирнинг алоҳида нукталари ранги эмас, балки чизиклар формаси кўпроқ аҳамиятга эгадир. Векторли редакторларда ҳар бир чизик 3-чи даражали математик чизик сифатида кўрилади, ва шунга мос равишда у нукталар комбинацияси кўринишида эмас, балки математик формула сифатида тасвирланади (компьютерда бу формуланинг сонли коэффициентлари сақланади). Бундай тасвир, растрлига қараганда анчагина ихчам бўлиб, маълумотлар кам жойни эгаллайди. Аммо ҳар қандай объектни куриш, нукталарни экранда оддий тасвирлаш билан эмас, балки узлуксиз равишда эгри чизик параметрларини экран ва босма тасвир координатларида қайта ҳисоблаш билан олиб борилади. Албатта, векторли графикада ишлаш, қуввати катта ҳисоблаш тизимларини талаб қилади.

Векторли редакторлар тасвирлар яратиш учун қулай, аммо амалда тайёр расмларга ишлов беришда ишлатилмайди. Улар кўпроқ реклама бизнесида ишлатилади ва уларни полеграфик нашрлар муқовасини безашда фойдаланилади. Демак, улар бадиий иш чизма ишига яқин бўлган ҳамма ҳолларда ишлатилади.

Уч ўлчамли редакторлар. Бу редакторлардан уч ўлчамли композицияларни яратишда фойдаланилади. Улар икки хил ўзига хос хусусиятга эга. Биринчидан, тасвирланаяпган объектни уч ўлчамлилигини кўрсатиш учун объект сирти хоссалари билан ёруғлик манбаси ўртасидаги ўзаро таъсирни мос равишда бошқариш; иккинчидан, уч ўлчамли анимацияни яратиш имконини беради. Шунинг учун ҳам уч ўлчамли графика редакторларини 3D-аниматорлар деб аталади.

Маълумотлар базасини бошқариш тизимлари. Маълумотлар базаси деб, жадвал кўринишида ташкил этилган катта ҳажмдаги маълумотлар базаларига айтилади. Маълумотлар базаларини бошқариш тизимларининг асосий функциялари қуйидагилардир:

- Маълумотлар базасининг бўш (тўлдирилмаган) тузилиши (структура)ни яратиш;
- Маълумотлар базасини тўлдириш ёки бошқа МБнинг жадвалидан жўнатиш (импорт) воситалари билан таъминлаш;
- Маълумотларга мурожаат имконияти, ва шу билан бирга қидирув ва фильтрация воситалари билан таъминлаш.

МБ ининг кўп тизимлари, қўшимча равишда маълумотларга ишлов бериш ва уларни оддий таҳлил қилиш имконига эга. Натижада, МБ ининг мавжуд жадваллари асосида янгиларини яратиш мумкин. Тармоқ технологияларининг жадвал суратда ривожланиши, маълумотлар базаларини бошқариш тизимларига, умумжаҳон компьютер тармоқлари серверларида жойлашган тақсимланган ва масофадаги ресурслар билан ишлаш имконияти талаби қўйилади.

Электрон жадваллар. Электрон жадваллар ҳар хил типдаги маълумотларни сақлаш ва уларга ишлов беришнинг турли комплекс воситаларига эга. Маълум даражада электрон жадваллар, маълумотлар базаларини бошқариш тизимларига ўхшаш, аммо уларда асосий урғу маълумотларни сақлаш ва уларга мурожаатни таъминлашга эмас, балки маълумотларни уларнинг мазмунига мос равишда ўзгартиришга бурилади.

МБ лари асосан маълумотларнинг ҳар хил типлари (сонли ва матнли маълумотлардан тортиб то мультимедиали маълумотларгача) билан ишлайди, электрон жадваллар эса кўпроқ сонли маълумотлар билан ишлайди. Аммо шу билан бирга, электрон жадваллар сонли типдаги маълумотлар билан ишлашнинг бир қанча усулларини тақдим этади.

Лойиҳалашнинг автоматлаштирилган тизимлари (CAD-тизимлар). Бу тизимлар лойиҳа ва конструкторлик ишларини автоматлаштиришга мўлжалланган. Улар машинасозлик, асбобсозлик ва архитектурада қўлланилади. Чизма-график ишлардан ташқари бу тизимлар оддий ҳисоблашларни (м-н, деталлар чидамлилигини) бажаради ва кенг қўламдаги маълумотлар базаларидан тайёр конструктив элементларни танлайди.

CAD-тизимларнинг яна муҳим хусусияти шундан иборатки, у лойиҳалашнинг ҳамма босқичларида, лойиҳани техник шартлар, қоида ва меёрлар билан автоматик тарзда таъминлайди, бу эса архитектор ва конструкторларни ижодий характерга эга бўлмаган ишлардан озод қилади. Масалан, машинасозлик CAD-тизимлари умумий чизма асосида, автоматик тарзда алоҳида деталларнинг ишчи чизмасини ва керакли техник хужжатларни тайёрлайди.

Кичик нашриёт тизимлари. Бу синф дастурларининг вазифаси полиграфия нашрларини териш жараёнини автоматлаштиришга мўлжалланган. Бу синф дастурлари, матн редакторлари ва автоматлашган лойиҳалаш тизимлари орасидаги ўринни эгаллайди.

Эксперт тизимлари. Бу тизимлар, билимлар базаларидаги маълумотларни таҳлил қилиш ва улар асосида, фойдаланувчи сўровномаси бўйича тавсиялар беришга мўлжалланган.

Бундай тизимлар, ечим қабул қилиш учун кенг қўламда махсус билимлар талаб қилинган ҳолларда қўлланилади. Бундай тизимлар қўлланиладиган асосий соҳалар ҳуқуқшунослик, медицина, фармакология ва бошқалардир. Медицина эксперт тизимлари, касаллик белгилари бўйича ташхис (диагноз) қўйиш, дори-дармон тайинлаш ва даволаш курсини режасини аниқлашга ёрдам беради. Ҳуқуқшуносликда эса, ҳодиса белгилари бўйича, айбловчи ва ғимоя қилувчи томонлари учун чоралар белгилаш тартиби ва ҳукм қабул қилишда ёрдам беради.

Эксперт тизимларининг ўзига хос хусусияти уларнинг ўзини сифатини ошириш ва ривожлантириш хусусиятидир. Бошланғич маълумотлар, билимлар базасида фактлар кўринишида сақланади, эксперт мутахассислар томонидан улар орасида муносабатларнинг маълум тизими ўрнатилади. Ва шундан сўнг, эксперт тизими у ёки бу саволлар бўйича маслаҳат ва тавсиялар беради.

HTML (Web) редакторлар. Бу ўзида, матн ва график редакторлари хоссаларини бирлаштирувчи редакторлар синфидир. Улар Web-хужжатларни тайёрлашга мўлжаллангандир. Web хужжатлар деб, уларни тайёрлашда, интернетда маълумотларни узатиш ва қабул қилиш билан боғлиқ бўлган қатор хусусиятлар ҳисобга олинган электрон хужжатларга айтилади.

Назарий жihatдан Web-хужжатларни яратиш учун оддий матн редактори ва процессорларидан, ва шу билан бирга векторли график редакторларнинг баъзиларидан ҳам фойдаланиш мумкин. Аммо Web-редакторлар Web-дизайнерларнинг иш унумдорлигини оширишнинг қатор хусусиятларига эга. Шунинг учун ҳам, бу синф дастурларидан электрон хужжатлар ва мультимедиа нашрларини тайёрлашда фойдаланиш мумкин.

Браузерлар-(Web ни кўриш воситалари). Бу категорияга HTML форматида (бу формат хужжатлари Web-хужжат сифатида ишлатилади) яратилган ва электрон хужжатларни кўриш учун мўлжалланган дастур воситалари киради. Замонавий браузерлар ёрдамида текст ва графикани кўрибгина қолмасдан, балки мусиқа, овоз, интернетдаги радио эшиттиришларни эшитиш, видеоконференцияларни кўриш, электрон почта хизматидан фойдаланиш, телеконференциялар тизимида ишлаш ва бошқа кўпгина ишларни бажариш мумкин.

Иш юритишнинг интеграллашган тизимлари. Бу дастурлар, бошлиқ иш жойини автоматлаштириш воситаларидан иборатдир. Бундай тизимнинг асосий функцияларига оддий хужжатларни яратиш, тузатиш ва форматлаш, электрон почта, факсимил ва телефон алоқа функцияларини марказлаштириш, корхона хужжат алмашинувини кузатиш (мониторинг), корхона бўлимлари фаолиятини координациялаш, раҳбарият ва хўжалик фаолиятини оптималлаштириш, ва сўровномалар бўйича маълумотлар бериш киради.

Бухгалтерия тизимлари. Бу махсуслаштирилган тизимлар бўлиб, улар ўз ичига матн ва жадвал редакторлари, электрон жадваллар ва маълумотлар базаларининг функцияларини олади. У корхона бирламчи бухгалтерия хужжатларини тайёрлаш ва ҳисобга олишни автоматлаштириш, бухгалтерлик ҳисоботларни олиб боришга мўлжалланган. Бундан ташқари бу тизим, корхонанинг, солиқ ва статистик ҳисоб ташкилотларига берадиган формада тайёрланадиган мунтазам ҳисоботларини олиб бориш учун ишлатилади. Бу ҳисоботлар корхоналарнинг ишлаб чиқариш, хўжалик ва молиявий фаолиятини акс эттиради. Албатта бу ҳисоботларни юқорида келтирилган бошқа тизимлар ёрдамида бажариш мумкин. Аммо бухгалтерия тизими, ҳар хил воситалар муҳитларини битта тизимда мужассамлаганлиги билан қулайдир.

Молиявий аналитик тизимлар. Бу синф дастурлари, банк ва биржа каби ташкилотларда фойдаланилади. Улар молия, товар ва хом ашё бозорларидаги ҳолатни назорат қилиш ва олдиндан кўра олиш, рўй бераётган ҳодисаларни таҳлил қилиш, ахборот ва ҳисоботлар тайёрлаш учун ишлатилади.

Геомаълумот тизимлар. Аэрокосмос ва топографик усулларда олинган маълумот асосида картография ва геодезия ишларини автоматлаштириш учун мўлжалланган тизимлардир.

Видеомонтаж тизимлари. Бу тизимлар видеоматериалларга рақамли ишлов бериш, уларни монтаж қилиш, видеоэффектлар яратиш, дефектларни олиб ташлаш, товуш, титр ва субтитр қўшиш учун мўлжалланган.

Амалий дастурий воситаларининг алоҳида категориялари, ўқитиш, малакани ошириш, маълумот ва кўнгил очар (развлекательные) тизим ва дастурлардан иборатдир. Бу дастурларнинг ўзига хос хусусиятлари шундан иборатки, уларда мультимедиа ташкил этувчиларига бўлган талаб юқоридир. Яъни мусиқа композициялари, график анимация ва видеоматериаллардан фойдаланиш кўзда тутилади.

1.4. Хизматчи дастурий воситаларни синфларга ажратиш

Файл диспетчерлари (файл менеджерлари). Бу синф дастурлари ёрдамида файл структурасига хизмат қилиш билан боғлиқ бўлган кўпгина амаллар бажарилади, яъни: нусха олиш, жойини ўзгартириш, файл номини ўзгартириш, каталог (папка) яратиш, файлларни қидириш ва файл структураида навигация. Бу мақсадларга мўлжалланган дастур воситалари одатда тизимли сатх дастурлари таркибига киради ва ОТ билан биргаликда ўрнатилади. Аммо, компьютер билан ишлаш қулайлигини ошириш учун кўпгина фойдаланувчилар, қўшимча хизматчи дастурларни ўрнатадилар.

Маълумотларни зичлаштириш воситалари (архиваторлар). Улар архивлар яратиш учун мўлжалланган. Маълумотларни архивлаштириш, файл ва каталогларнинг катта гуруҳларини битта архив файлига жамлаш ҳисобига сақлашни осонлаштиради. Бу ҳолда архив файллари маълумотларни юқори даражада зичлаштириб ёзиш ҳисобига, маълумотларни сақлаш қурилмалари самарадорлигини ошириш имконини беради. Архиваторлар кўпинча қимматли маълумотлардан резерв нусха олиш учун ҳам фойдаланилади.

Кўриш ва акс эттириш воситалари. Одатда маълумотлар файллари билан ишлаш учун уларни ўз муҳитига, яъни улар ўзи ишлаб чиқилган амалий муҳитга юклаш керак. Бу эса, ҳужжатларни кўриб чиқиш ва уларга ўзгартириш киритиш имконини беради. Аммо ҳужжатларни ўзгартирмасдан фақат кўриб чиқиш зарурияти бўлган ҳолларда, ҳар хил типдаги ҳужжатларни кўришга имкон берадиган оддий ва универсал воситалардан фойдаланиш қулайдир.

Диагностика воситалари. Аппарат ва дастурий таъминот диагностика жараёнини автоматлаштириш учун мўлжалланган. Улар керакли текширишларни ўтказиб, йиғилган маълумотларни қулай ва яққол кўринишда беради. Бу маълумотлардан, бузилишларни тузатиш учунгина ҳам эмас, балки компьютер тизими ишини оптималлаштириш учун ҳам ишлатилади.

Назорат (мониторинг) воситалари. Назорат воситаларини баъзан мониторлар дейилади. Улар, компьютер тизимида рўй берадиган жараёнларни кузатиш имконини беради. Бунда икки хил ҳолат бўлиши мумкин: реал вақт режимида кузатиш ёки натижаларни махсус протокол файлига ёзиш билан назорат қилиш. Биринчи ҳолат одатда ҳисоблаш тизимини оптималлаштириш йўлини қидиришда ва унинг самарадорлигини оширишда фойдаланилади. Иккинчи ҳолат одатда, мониторинг автоматик ҳолда ва (ёки) масофадан бажарилса фойдаланилади.

Ўрнатиш мониторлари. Бу категория дастурлари дастурий таъминотни ўрнатишни назорат қилиш учун мўлжалланган. У ёки бу дастурни ўрнатиш ёки олиб ташлашда, бошқа дастурларни ишловчанлиги бузилиши мумкин. Ўрнатиш мониторлари эса, атроф дастурий муҳитини ўзгариши ва ҳолатини кузатадилар, дастурлар орасидаги янги алоқалар пайдо бўлишини ҳисобга оладилар ва олдинги ўрнатилган дастурларни олиб ташлашда йўқолган алоқаларни қайта тиклайдилар.

Дастурларни ўрнатиш ва олиб ташлашни бошқарадиган оддий воситалар одатда операцион тизимнинг таркибига киради ва улар тизимли дастурий таъминоти сатҳида жойлашган бўлади, аммо улар етарли эмас. Шунинг учун ҳам, юқори даражадаги ишончлилик талаб қилинадиган ҳисоблаш тизимларида қўшимча шундай хизматчи дастурлардан фойдаланилади.

Коммуникация воситалари (коммуникация дастурлари). Электрон алоқа ва компьютер тармоқлари пайдо бўлиши билан бу синф дастурлари жуда катта аҳамиятга эга бўлиб қолди. Улар узок масофадаги компьютерлар билан алоқа ўрнатади, электрон почта хабарларини узатишга, телеконференциялар ишига ёрдам беради, факсмил хабарларни узатишни таъминлайди ва компьютер тармоқларидаги кўп амалларни бажаради.

Компьютер хавфсизлигини таъминлаш воситалари. Бу кенг қўламдаги категорияга маълумотларни бузилишлардан химоя воситалари, ва шу билан бирга маълумотларга ҳуқуқсиз мурожаат, уларни кўриш ва ўзгартиришдан химоя воситалари киради. Бу дастурларга, масалан, активирис дастурий таъминоти киради.

3-маъруза. Операцион тизим структураси

ОТ ни синфларга ажратиш. ОТ тузилиши. Ядро, команда процессори, киритиш-чиқариш тизими, файл тизими. ОТ қуриш принциплари. Виртуал машина тушунчаси.

ОТ лар бажарадиган вазифасига, масалаларга ишлов бериш режимига, тизим билан боғланиш усулига, қуриш усулига ва х.к.лар бўйича синфларга ажратилади. Вазифасига қараб синфларга ажратишда ОТ нинг турлари кўплиги яққол ташланади. Чунки ҳисоблаш техникаси турлари қанча бўлса, ОТ турлари ҳам шунчадир: мейнфрейм ОТ, сервер ОТ, кўппроцессорли, ШК учун, реал вақт ОТ, мобил ва х.к. ОТ ларига бўлинади.

Мейнфреймлар, ШК лардан киритиш-чиқариш имкониятлари билан фарқ қилади, улар терабайт ҳажмларда маълумотлар ишлов бериш имконини беради. Мейнфреймлар ОТ лари киритиш-чиқариш амаллари кўп бўлган, бир вақтда бажариладиган топшириқлар тўпламига ишлов беришга йўналтирилган. Қоида бўйича, уч хил хизмат қилиш таклиқ қилинади: пакетли ишлов бериш, транзакцияли ишлов бериш (гуруҳли амаллар) ва вақтни бўлиш тизимлари. Пакетли ишлов беришда, масалаларга фойдаланувчисиз ишлов берилади. М-н, ҳар хил ҳисоботлар тузиш пакетли ишлов бериш режимида бажарилади. Транзакцияли ишлов беришда, кўп сонли кичик талабномалар (запросы) бажарилади, м-н, билетларни олдиндан буюртма орқали банд қилиш, кредит карточкалари билан бажариладиган амаллар ва х.к.лар. Талабномалар катта эмас, аммо тизим бир вақтда секундига юзлаб ва минглаб операцияларни бажаради. Вақтни бўлиш режимида, тизим кўп сонли масофадаги фойдаланувчиларга бир вақтнинг ўзида битта машинада масалаларни бажариш имконини беради. Бунга кўп фойдаланувчили МБ ни мисол қилиб олиш мумкин. OS/390 мейнфрейми ОТ мисол бўлади.

Серверлар бир вақтнинг ўзида кўп сонли фойдаланувчиларга хизмат қилади ва уларга ўзаро дастурий ва аппарат ресурсларни бўлиб олиш имконини беради. Серверлар босмадан чиқариш қурилмалари, интернет ва файллар билан ишлаш имконини яратади. Серверда Web-саҳифалар сақланади ва талабларга ишлов беради, UNIX, Windows 2000, Linux сервер ОТ ларига мисол бўла олади.

Бир нечта процессорлар бирлаштирилган тизимларда ишлаш учун, махсус ОТ лар талаб қилинади. Кўп процессорли ОТ лар махсус алоқа имкониятига эга бўлган сервер ОТ ларидан иборатдир.

ШК ОТ асосий вазифаси – фойдаланувчига қулай интерфейс яратишдир. Бу ОТ лар матнлар билан, электрон жадваллар билан, интернетга мурожатда ва х.к.лар учун фойдаланилади. Бунга мисол Windows, Linux ва х.к. ОТлардир.

Реал вақт ОТ ларидан, ҳисоблаш тизими бошқарадиган жараёнлар қатъий вақт чегараларини қониқтириши керак бўлган ҳолларда фойдаланилади. Агар ҳодисалар кўрсатилган қатъий вақт диапазонида рўй

бериши керак бўлса – бу қатъий PBT дир. Агар вақти-вақти билан амаллар бажарилиши муддатини ўтказиб юбориш мумкин бўлса, м-н, рақамли аудио ва мультимедиа тизимлари, бу мослашувчан PBT дир.

Мисол қилиб VxWorks ва QNX ОТ ларини келтириш мумкин.

Чўнтак, мобил компьютерлар ва кўпгина майший ва бошқа қўрилмаларни бошқариш учун (телевизор, мобил телефон ва х.к.лар) ўз ОТ лари ишлатилади. Улар PBT лари характеристикаларига эга бўлиши мумкин, фақат кичик ўлчам, кичик хотира ва чегараланган қувватга эгадир. Мисол учун Palm OS ва Windows CE.

Монолит ОТ ларда тизим ҳамма қисмлари ўзаро мустаҳкам боғланган. Шунинг учун ҳам унинг у ёки бу қисмини ўзгартириш ва олиб ташлаш От нинг бутун архитектурасини мукамал билишни талаб қилади ва бошқа модулларни ўзгартириш заруриятини келтириб чиқаради. Бу ҳолларда, микроядро функциялари ягона адрес маконида бажарилгани учунқатор муаммолар келиб чиқиши мумкин. Бу эса ўз навбатида низолар келиб чиқиши хавфини ва янги драйверларни ишга тушириш муаммоларини келтириб чиқаради. Бундай тизимлар бўлакларга бўлинмаган, яъни сруктурага эга эмас. ОТ процедуралар фиғиндисидан иборат бўлиб, уларнинг ҳар бири зарурият туғилганда ихтиёрий бошқа процедурани чақириши мумкин. Бундай тизимни тузиш учун ҳамма алоҳида процедуралар компеляция қилиниб, компановщик ёрдамида ягона объект файлига бирлаштирилади. Монолит тизимлар, узилишлар механизмини қувватлаши мумкин. Бу ҳолда ОТ ни қисман структуралаштириш зарур: юқори сатҳда бош дастур жойлашган бўлиб, талаб қилинган хизматчи процедурани чақиради. Ундан қуйида эса тизимли чақириқларни бажарувчи хизматчи процедуралар жойлашади. Улардан ҳам қуйида эса, тизимли процедураларга хизмат қилувчи утилиталар жойлашгандир.

Кўп сатҳли тизимлар, сатҳлар иерархияси кўринишида ташкил этилгандир. Бундай илк тизимга ТНЕ тизими мисол бўла олади, у 1968 йилда Дейкстри томонидан тузилган. У 6 та сатҳдан иборат эди. 0-чи сатҳ – процессорни тақсимлаш ва кўпмасалалик бўлса, 1-чи сатҳда хотирани бошқариш, 2-чи оператор-жараён алоқаси, 3-чи киритиш-чиқаришни бошқариш, 4-сатҳ – фойдаланувчи дастурлари, 5-сатҳ оператор. 0-чи сатҳ, узилиш ёки таймер кетиши ҳолларида жараёнларни биридан бошқасига ўтиб, процессор вақтини тақсимлаш билан машғул бўлган. Бу сатҳдан юқори сатҳларда тизим кетма-кет жараёнлардан иборат бўлиб, уларнинг ҳар бирини, битта процессорда бир нечта жараён ишга туширилганидан кўрқмасдан дастурлаш мумкин бўлган. Яъни 0-чи сатҳ процессор хотирани бошқарган. 2-чи сатҳ оператор консоли ва жараёнлар боғланишини бошқарган. Бу сатҳдан юқоридаги жараёнлар ўз шахсий консолларига эгадирлар. 3-сатҳ киритиш-чиқариш қурилмалари ва маълумотларни буферлашни бошқарган. 3-чи сатҳдан юқоридаги ихтиёрий жараёнлар, киритиш-чиқаришнинг аниқ қурилмалари биланэмас, балки фойдаланувчи учун қулай бўлган КЧ қурилмаларининг абстракт характеристикалари билан

ишлаган. Кўп сатҳли тизимларнинг концепцияларнинг кейинги умумлашуви MULTICS тизимларида амалга оширилган.

Виртуал машиналар иккита принципти мукамаллаштириш асосида ривожланган:

1. вақтни бўлиш тизимлари кўп масалаликни таъминлайди, 2. Бевосита қурилмалар билан ишлашдан фарқли равишда, қулай интерфейсга эга бўлган кенгайтирилган машина. Бу кўринишда илк ОТ га VM/370 ни мисол қилса бўлади. Виртуал машина монитори қурилмалар билан ишлайди ва юқори сатҳларга бир нечта виртуал машинани бериб, кўпмасалаликни таъминлайди. Бошқа ОТ лардан фарқли равишда, бу виртуал машиналар кенгайтирилган эмас, балки ядро ва фойдаланувчи киритиш-чиқариш, узилиш ва х.к.лар режимидан иборат аппаратуранинг аниқ нусхасидан иборатдир. Натижада, бундай виртуал машиналар ҳар бирида ихтиёрий ОТ ишга туширилиши мумкин. Дастур тизимли чақирикни бажарганда, у VM/370 да эмас, балки виртуал машинадаги ОТ ни узади. Виртуал машина ҳолатида кўпмасалалик ядро даражасида амалга оширилади, у фойдаланувчи ОТ дан ажратилган. Ҳозирги вақтда ВМ дан бошқача ҳолатда фойдаланилади, м-н, бир нечта операцион тизими муҳитини ташкил этиш учун. Бунга мисол бўлиб, VDM – машина (Virtual Doc Machine) ни келтириш мумкин, бу тизим ҳимояланган тизим бўлиб, MS-DOS нинг тўлиқ муҳитини ва унинг иловаларини бажарилиши учун консол тақдим этади. Бир вақтнинг ўзида, амалда, VDM сессияларнинг ихтиёрий сони бажарилиши мумкин. ВМ тушунчаси Java – аплетларни тузишда ҳам фойдаланилади. Java компилятор JVM учун код тузади. Бу код ихтиёрий платформада, JVM интерпретатор мавжуд платформада бажарилиши мумкин.

ВМ концепциясининг ривожланиши фойдаланувчини, кўп ресурсли реал компьютернинг абслют нусхаси билан таъминлайдиган тизим юзага келишига олиб келди. Ядро режимининг қуйи сатҳида, ВМ учун ресурсларни тақсимловчи ва улардан фойдаланишни ҳимояловчи, экзоядро деб аталувчи дастур ишлайди. Ҳар бир ВМ, фойдаланувчи сатҳида, ўз шахсий ОТ билан ишлайди, фақат фарқ шундаки-ресурслар мажмуасини тақдим этиш чегараланганидир. Схема устунлиги, ВМ адресларини диск реал адресларига ўзгартириш жадвали талаб қилинмаслигидан иборатдир, чунки ҳар бир ВМ учун ўз адрес блоклари ажратилади.

Замонавий ОТ ларда, кодларни юқори сатҳларга ўтказиш, ядро режимида минимал зарурий функцияларни “микроядро” деб аталувчи қисмда қолдириш тенденцияси кўрсатилмоқда.

Микроядро қуйидаги хизматларни (сервисларни) таъминлайди:

- виртуал хотирани бошқариш
- топширик ва оқимлар
- жараёнлараро коммуникациялар
- киритиш-чиқариш ва узилишларни бошқариш
- хост ва процессор хизматлари.

ОТ учун хос бўлган бошқа функциялар, хабарни ўзаро узатиш орқали мулоқат қилувчи, модулли қўшимча-жараёнлар сифатида лойиҳаланиши

мумкин. Амални бажариш учун талабнома олиб, фойдаланувчи жараёни (клиент), хизмат қилувчи жараёнига (серверга) талабнома жўнатади, у ўз навбатида ишлов бериб, жавоб қайтаради.

ОТ ни бўлакларга бўлиш оқибатида, бўлакларнинг ҳар бири тизимнинг бир элементини бошқаради, ва ҳар бир бўлак кичик ва бошқарилувчан бўлиб қолади. Ҳамма серверлар фойдаланувчи режимида жараён каби ишлагани учун, улар қурилмаларга бевосита мурожаат қила олмайдилар, шунинг учун тизим бузилишларга қатъий бўлиб қолади. ОТ нинг баъзи функциялари, м-н, киритиш-чиқариш қурилмалари регистрга командаларни юклашни фойдаланувчи маконидаги дастурлардан бажариш амалда мумкин эмас. Ечимлардан бири шундан иборатки, сервернинг критик жараёнлари (м-н, қурилма драйверлари), ядро режимидан ишга туширилади, аммо бошқа жараёнлар билан ахборотларни узатиш анаънавий схемаси бўйича боғланади. Клиент-сервер моделининг устунлиги яна шундан иборатки, у тақсимланган тизмларга қулай мослашади. Ҳақиқатда, ҳар бир бўлак мустақил бўлгандан кейин, уларнинг ихтиёрийси масофадаги машинада осон бажарилиши мумкин. Шунда клиент нуктаи назаридан ҳам шу жараён боради: талабнома жўнатилади ва жавоб қайтарилади.

Тизимли бошқариш ва ишлов бериш дастурлари мажмуаси (комплекс) сифатида ОТ, ҳисоблашларни самарали ва ишончли бажарилишини таъминлаши керак бўлган ўзаро боғланган дастур модуллари ва маълумотлар структурасининг жуда мураккаб “конгломератини” ташкил этади.

Операцион тизимнинг кўпгина потенциал имкониятлари, унинг техник ва истёъмол параметрлари-буларнинг ҳаммаси асосан ОТ архитектураси билан-унинг тузилиши (структураси) ва уни қуриш принциплари билан аниқланади.

ОТ ни қуриш асосий принциплари

Частота принципи. Дастур алгоритмларида, ишлов бериладиган массивларда амал ва катталикларни фойдаланиш частотасига қараб ажратишга асосланган. Кўп марта ишлатиладиган амал ва маълумотларга тезроқ мурожаат қилишни таъминлаш учун, уларни оператив хотирага жойлаштирилади. Бундай мурожаатнинг асосий воситаси, кўп сатҳли режалаштиришни ташкил этишдир. Узоқ муддатли режалаштиришга тизим фаолиятининг камёб ва узун амаллари ажратилса, қисқа муддатли режалаштиришга эса кўп ишлатиладиган ва қисқа амаллар ажратилади. Тизим дастурлаш бажарилишини инициализация қилади ёки узади, динамик тарзда талаб қилинадиган ресурсларни беради ва қайтиб олади, энг биринчи навбатда бу ресурслар – хотира ва процессордир.

Модуллилик принципи. Модул-бу тизимнинг тугалланган элементи бўлиб, у модуллараро интерфейсга мос равишда бажарилгандир. Модул таърифи бўйича, уни ихтиёрий бошқасига, мос интерфейсс мавжуд бўлганда алмаштириш имконини назарда тутати. Кўпинча, ОТни қуришда имтиёзга эга бўлган, қайта кирадиган ва ринтерабель модуллар катта аҳамиятга эгадир. Имтиёзга эга бўлган модуллар.... имтиёзли режимда амалга ошади, бу

режимда узилишлар тизими ўчирилади, ва ҳеч қандай ташқи ҳодиса ҳисоблашлар кетма-кетлигини буза олмайди. рентерабл модуллар бажарилишни (ижрони) кўп марта, такроран узилишини ва бошқа масалалардан қайта ишга туширишни назарда тутати. Бунинг учун, оралик ҳисоблашларни сақлаш ва узилган нуқтадан уларга қайтиш таъминланади. Қайта кирадиган модуллар кўр марта параллел фойдаланишни назарда тутати, аммо узилишни назарда тутмайди. Улар имтиёзли блоклардан ташкил топган бўлиб, уларга қайта мурожаат, бу блокларнинг бирортасининг тугалланганидан кейин мумкин бўлади. Модуллилик принципи, тизимнинг технологик ва эксплуатация хоссаларини акс эттиради. Фойдаланишнинг максимал самарадорлиги, агар бу принцип ОТ га ҳам, амалий дастурларга ҳам аппаратурага ҳам хос бўлса.

Функционал танланиш принципи. Бу принцип, ҳисоблашлар унумдорлигини ошириш мақсадида, доимий равишда оператив хотирада бўлиши керак бўлган модулларни ажратишни назарда тутати. ОТ нинг бу қисми ядро дейилади.

Бир томонда оператив хотирада қанча модуллар кўп бўлса, амаллар бажарилиш тезлиги шунча юқори бўлади. Бошқа томондан, ядро банд қиладиган хотира хажми жуда катта бўлиши мумкин эмас, чунки акс ҳолда амалий масалаларга ишлов бериш самараси паст бўлади. Ядро ўз таркибига узилишларни бошқариш модуллари, мультимасалаликни таъминловчи жараёнлар орасида бошқарувни узтиш модуллари, хотирани тақсимлаш модули ва х.к.ларни олади.

ОТ ни генерация қилиш принципи. Бу принцип, ечиладиган масала ва ҳисоблаш тизимининг конфигурациясидан келиб чиққан ҳолда, ОТ ни созлашга имкон берадиган ОТ ядроси архитектурасини ташкил этиш принципини белгилайди. Бу процедура жуда кам ҳолларда, ОТ ни узок вақт давомида эксплуатация қилиш олдидан бажарилади.

Генерация жараёни махсус генератор-дастури ва мос кириш тили ёрдамида амалга оширилади. Генерация натижасида ОТ нинг, тизимли модул ва катталиклардан иборат тўлиқ версияси вужудга келади. Модуллилик принципи генерацияни аҳамиятли даражада соддалаштиради. Бу принцип айниқса Linux ОТ ларида яққол кўзга ташланади, унда нафақат ОТ ядроси генерация қилинади, юкланадиган транзит модуллари таркибини кўрсатади. Бошқа ОТ ларда конфигурация қилиш инсталляция жараёнида бажарилади.

Функционал ортиқчалилик принципи. Бу принцип айнан бир амални, ҳар хил воситалар билан бажариш имкониятини ҳисобга олади. ОТ таркибига ресурсларни бошқарувчи бир неча хил мониторлар ва файлларни бошқарувчи бир неча хил мониторлар ва файлларни бошқарувчи бир нечта тизимлар ва х.к.лар киради. Бу эса шў навбатида, ОТ ни ҳисоблаш тизимини аниқ конфигурациясига тез ва етарли даражада мослашишга, аниқ синф масалаларини ечишда техник воситаларни самарали юклашни максимал таъминлашга ва шунда максимал унумдорликка эришишга олиб келади.

Стандарт ҳолатлар принципи (по умолчанию). Тизим билан ишлашда, ҳам генерация босқичида ҳам, тизимлар билан боғланишни ташкил

этишни енгиллаштириш учун қўлланилади. Принцип тизимидаги фойдаланувчи дастурини характерловчи ва уларнинг бажарилиш вақтини олдиндан аниқловчи, қурилма конфигурацияси, модуллар ва жараёнлар структурасини тавсифларини тизимда сақлашга асослангандир. Бу маълумотни фойдаланувчи тизими, маълумот берилмаган бўлса, ёки атаёдан аниқлаштирилмаган бўлса, фойдаланади. Умуман, бу принципни қўллаш, фойдаланувчи тизим билан ишляпган вақтда, у ўрнатадиган параметрларни қисқартириш имконини беради.

Жойини ўзгартириш принципи. Бу принцип модулларнинг бажарилиши, уларнинг хотирада жойлашган ўрнига боғлиқмаслигини кўзда тутди. Модул матнини, уни хотирада жойлашувига мос равишда созлаш махсус механизмлар, ёки унинг бажарилиши давомида амалга оширилади. Созлаш, командаларнинг адрес қисмида фойдаланадиган ҳақиқий адресларни аниқлашдан иборат бўлиб, айти ОТлар учун қабул қилинган оператив хотирани тақсимлаш алгоритми ва қўлланиладиган адреслаш усули билан аниқланади. У фойдаланувчи дастурларига ҳам тақсимланади.

Виртуаллаштириш принципи. Бу тизим ягона марказлашган схемадан фойдаланиб, тизим структурасини, жараёнларни режалаштирувчилар (планировшиклар) ва ресурс (мониторлари) тақсимловчилари маълум мажмуаси кўринишида тасвирлашга имкон беради. Виртуаллик концепцияси, виртуал машина тушунчасида аксланади. Ихтиёрий ОТ, ҳақиқатда, фойдаланувчидан, реал аппарат ва бошқа ресурсларни яшириб, уларни маълум абстракциялар билан алмаштиради. Натижада, фойдаланувчилар виртуал машинани, уларнинг дастурларини қабул қилувчи ва уларни бажариб, натижа берувчи етарли даражадаги абстракт қурилма сифатида фойдаланадилар ва тасаввур қиладилар. Фойдаланувчини, умуман ҳисоблаш тизими реал конфигурацияси ва унинг компоненталаридан самарали фойдаланиш физиктирмайди. Бир нечта параллел жараёнлар учун, бир вақтнинг ўзида реал тизимда мавжуд бўлмаган нарсадан бир вақтда фойдаланиш тасаввури ҳосил қилинади.

ВМ, реал архитектурани ҳам акс эттириши мумкин, аммо бу ҳолда архитектура элементлари кўпинча система билан ишлашни соддалаштирувчи, мукаммалаштирувчи янги параметрлар билан чиқадилар. Фойдаланувчи нуқтаи-назарида, идеал машина қуйидагиларга эга бўлиши керак:

- ишлаши мантиқи жихатидан бир хил тарздаги, чегараланмаган ҳажмга эга бўлган виртуал хотира;
- параллел равишда бир-бирига таъсир қиладиган ва ишлай оладиган виртуал процессорларнинг ихтиёрий миқдори;
- виртуал машина хотирасига кетма-кет ва параллел, синхрон ва асинхрон мурожаат этишга кодир бўлган виртуал ташқи қурилмаларнинг ихтиёрий миқдори (сони) маълумотлар ҳажми чегараланмаганда идеал машинага яқинлаштирилган, ОТ томонидан амалга ошириладиган виртуал машина қанчалик катта бўлса, яъни архитектурали мантиқий характеристикаси реалдан қанчалик фарқ

килса, демак виртуалликнинг шунчалик юқори даражасига эришилган бўлади. ОТ бир-бири ичига жойлаштирилган ВМ иерархияси сифатида курилади. Дастурларнинг қуйи сатҳи машинанинг аппарат воситаларидир.

Кейинги сатҳ эса дастурий бўлиб, қуйи сатҳ билан биргаликда, машина янги хоссаларга эга бўлишига ёрдам беради. Ҳар бир янги сатҳ маълумотларга ишлов бериш функция имкониятларини кенгайтириш имконини бериб, қуйи сатҳларга мурожаатни осонлаштиради.

ВМ ларни иерархик тартибга солиш устунликларга эга бўлиш, яъни лойиха доимийлиги, дастур тизимлари ишончлилиги, ишлаб чиқиш муддатлари қисқариши, қатор муаммоларга эга. Уларнинг асосийлари: виртуаллаштириш сатҳлари сонини ва хоссаларини аниқлаш, ОТ нинг ҳар бир сатҳига зарурий қисмларни киритиш қоидаларини аниқлаш. Абстрактлаштириш (виртуализация) алоҳида сатҳлари хоссалари:

1. Ҳар бир сатҳда, юқори сатҳлар мавжудлиги ва хоссалари тўғрисида ҳеч нарса маълум эмас.
2. Ҳар бир сатҳда, бошқа сатҳлар ички тузилиши тўғрисида ҳеч нарса маълум эмас. Улар орасидаги боғланиш олдиндан белгиланган қатъий қоидалар орқали олиб борилади.
3. Ҳар бир сатҳ бир нечта модулдан иборат, уларнинг баъзилари ички ҳисобланади ва уларга бошқа сатҳлар мурожаат қилиши мумкин. Қолган модуллар номи юқори сатҳларга маълум ва шу сатҳлар билан боғлана олади.
4. Ҳар бир сатҳ маълум ресурсларга эга, у ўз ресурслари абстракцияларини (виртуал ресурсларни) бошқа сатҳлардан яшириши ёки таклиф қилиши мумкин.
5. Ҳар бир сатҳ, тизимда маълумотларнинг маълум абстракциясини таъминлайди.
6. Ҳар бир сатҳда, бошқа сатҳга нисбатан қилинаётган таклиф минимал бўлиши шарт.
7. Сатҳлар орасидаги боғланиш аниқ аргументлар, бир сатҳдан иккинчисига узатиладиган аргументлар билан чегараланган бўлиши керак.
8. Глобал маълумотлардан бир нечта сатҳлар фойдаланиши мумкин эмас.
9. Ҳар бир сатҳ бошқа сатҳлар билан мустахкамроқ ва кучсиз боғланиши керак.
10. Абстракция сатҳи орқали бажариладиган ҳар қандай функция ягона киришга эга бўлиши керак.

Дастурий таъминотни ташқи қурилмаларга боғлиқ эмаслиги (муустақиллиги) принципи. Бу принцип, дастурнинг аниқ қурилмалар билан боғланиши, дастурларни трансляция даражасида эмас, балки ундан фойдаланишни режалаштириш давридалигидан иборатдир. Дастурларнинг янги қурилмалар билан ишлаши вақтида, қайта компиляция қилиниши талаб қилинмайди. Бу принцип кўпгина ОТларда амалга оширилади.

Мутаносиблик принципи (совместимость). Бу принцип, бир ОТ учун яратилган дастур таъминотининг (ДТ) бошқа ОТ ва шу ОТ нинг олдинги версияларида ҳам бажарилиш имкониятини белгилайди. Мутаносиблик ижро файллари ва дастур берилган матни даражасида бўлиши мумкин. Биринчи ҳолатда тайёр дастурни бошқа ОТда ишга тушириш мумкин. Бунинг учун микропроцессор командаси, тизимли ва кутубхона чақириқлари даражасидаги мутаносиблик талаб қилинади. Қоида бўйича, машина кодини қайта кодлаш имконини берадиган ва уларни бошқа процессорлар терминларидаги эквивалент командалар кетма-кетлигига алмаштирадиган махсус ишлаб чиқиладиган эмуляторлардан фойдаланилади. Бошланғич матн даражасидаги мутаносиблик, мос транслятор мавжудлигини, тизимли ва кутубхона чақириқлари даражасидаги мутаносибликни талаб қилади.

Очиқлик ва қўшимча имкониятлар қўшиш принципи. Очиқлилик таҳлил учун нафақат тизимли мутахассисларга балки фойдаланувчиларга ҳам имконият борлигини кўзда тутди. Қўшимча имкониятлар қўшиш, ОТ таркибига янги модуллар қўшиш ва мавжудларини ўзгартириш (модификация) имконини беради.

ОТ ни микроядро структурасидан фойдаланиб, клиент-сервер технологиясига асосан қуриш, қўшимча имкониятлар қўшиш кенг имконият яратади. Бу ҳолда ОТ имтиёзли бошқарувчи дастурлар ва имтиёзсиз сервер-хизматлар мажмуаси тарзида қурилади. Асосий қисм ўзгартирилмасдан қолиб, серверлар осон ўзгартирилади, алмаштирилади ва қўшимча қўшилади.

Мобиллилик принципи (кўчириб ўтказиш). Бу принцип ОТни бир платформадан, бошқа типдаги платформага кўчириш имконини назарда тутди. Кўчириб ўтказиладиган ОТ ни ишлаб чиқишда қуйидаги қоидаларга риоя қилинади: ОТ нинг деярли катта қисми, фойдаланишга мўлжалланган ҳамма платформаларда трансляторлари мавжуд бўлган тилда ёзилади. Бу юқори даражадаги, қоида бўйича С тилидир. Ассемблердаги дастур умумий ҳолда, кўчириб бўлмайдиган дастурдир. Кейин, аппарат ресурслари билан бевосита муносабатда бўлган код фрагментлари олиб ташланади ёки камайтирилади. Аппаратга боғлиқ код, бир нечта яхши локаллаштирилган модулларда ажратилган ҳолда бўлади.

Хавфсизлик принципи. Бир фойдаланувчи ресурсларини бошқа фойдаланувчидан ҳимояни, ва ҳамма тизимни ресурсларни фақат битта фойдаланувчи эгаллаб олишидан ҳимояни кўзда тутди., бундан ташқари бу принцип ўз ичига, ҳуқуқсиз мурожаатдан ҳимояни ҳам олади. NCSC (National Computer Security Center), 198й чиқарилган “оранживая книга” га асосан, тизимлар 7 та категорияга: D, C1, C2, B1, B2, B3, A1 га бўлинади, бу ерда А максимал ҳимояланган тизимдир. Аксарият кўпгина замонавий ОТ лар C2 синфга мансубдир. Бу синф қуйидагиларни таъминлайди.

- фойдаланувчини ягона ном ва парол билан тизимга киришга имкон берадиган, махфий кириш воситалари.
- ресурс эгасига, унинг ресурсидан фойдаланишга кимнинг ҳуқуқи бор-йўқлигини аниқлайдиган мурожаатни танлаб назорат қилиш;

- ҳисобга олиш ва кузатиш (аудит) воситалари, улар тизимли ресурсларга мурожаат ва тизим хавфсизлиги билан боғлиқ бўлган ходисаларни аниқлаш ва топишга имконни таъминлайди.

А синфи тизимни, маълум хавфсизлик кўрсаткичларига формал, математик жихатдан мослигини исботлашини талаб қилади. А синфида, хавфсизликни бошқариш механизми, процессор вақтининг 90% ини банд қилади. ОТ да ҳимояни таъминлашни бир нечта йўналиши амалга оширилади. Улардан бири, процессор ишини икки контекстда олиб бориш, яъни вақтнинг ҳар бир онда процессор ОТ таркибидаги дастурни ёки От таркибига кирмайдиган амалий ёки хизматчи дастурни бажариш мумкин. Ҳар қандай бўлинадиган ресурсларга фойдаланувчи ва хизматчи дастурлар томонидан бевосита мурожаатни таъқиқлаш учун, машина кодлари таркибига, ресурсларни тақсимловчи ва фойдаланишни бошқарувчи махсус имтиёзли командалар киритилади. Бу командаларни фақат ОТ га бажариш рухсат этилади. Уларнинг бажарилиш назорати аппарат қисми томонидан бажарилади. Бундай командани бажаришга ҳаракат қилинган ҳолда узилиш рўй беради, ва процессор имтиёзли режимга ўтказилади. Ҳимоя принципини амалга ошириш учун, оператив хотирадаги дастур матни ва маълумотларни ҳимоя қилиш механизмидан фойдаланилади. Бунда энг кўп тарқалган усул-контекст ҳимояланишидир. Дастурлар ва фойдаланувчилар учун хотиранинг маълум қисми ажратилади, ва бу чегарадан чиқилса ҳимоя бўйича узилиш рўй беради. Назорат механизми, аппарат тарзда, регистрлар чегараланганлиги ва хотира калитлари асосида амалга оширилади. Файлларда маълумотларни сақлашнинг ҳар хил ҳимоя усуллари қўлланилади. Энг оддий ҳимоя усули-паролли усулдир.

4-маъруза. Операцион тизим архитектураси ва куриш принциплари. Модуллик, махсус режим, виртуаллаштириш, мобиллик, мутаносиблик, очиклик, ҳисоблашларни ҳимоялаш в х.к. принциплари. Микроядорли операцион тизимлар. Микроядроли операцион тизим. Операцион тизимларга куйилган талаблар.

Биринчи дастурлар бевосита машина кодларида яратилган. Бунинг учун, микропроцессор архитектурасини ва унинг асосидаги тизимни мукамал билиш зарур. Ҳисоблаш техникасининг ривожланиши борасида, кўп учрайдиган амалларни (операцияларни) ажратиб, улар учун дастурий модуллар яратилиб, кейинчалик улардан дастурий таъминотда фойдалана бошладилар. Шундай қилиб, 50-чи йилларда, биринчи дастурлаш тизимларини яратишда киритиш чиқариш амаллари учун, кейинчалик математик амал ва функцияларни ҳисоблаш учун модуллар яратилди. Кейинчалик, ривожланиш қори даражадаги трансляторларни яратилишига олиб келди, яъни операторлар ўрнига зарурий функцияларни чақириқларни кўйиш имконияти келиб чиқди. Кутубхоналар сони ошиб борди. Натижада, амалий дастурий таъминот ишлаб чиқувчиларидан тизим архитектурасини мукамал билиш талаб қилинмай қолди. Улар, дастурий тизимга мос чақириқлар билан мурожаат ва улардан керакли сервис ва функцияларни олиш имконига эга бўлдилар. Бу дастурий тизим ОТ дир.

Замонавий ОТ асосий ташкил этувчилари – бу ядро, киритиш-чиқариш тизими, команда процессори, файл тизими. Ядро масалалар ва ресурсларни бошқариш, синхронлаштириш ва ўзаро боғланиши бўйича асос функцияларни таъминлайди. Команда процессори, командаларни қабул қилиш ва уларга ишлов бериш, фойдаланувчи талаби бўйича мос хизматларни чақиришни таъминлайди.

Киритиш ва чиқариш тизими, ташқи қурилмалар билан маълумотларни киритиш ва чиқариш масаласини таъминлайди.

ОТ кутубхоналарида бу функцияларнинг мавжудлиги, ҳар бир уларни дастурлаш тизими воситалари билан ҳар бир дастурга қўшмаслик имконини беради. Дастурлаш тизимлари фақат, киритиш-чиқариш тизими кодларига мурожаатни генерация қилади ва катталикларни тайёрлайди. КЧ тизими, киритиш-чиқариш қурилмалари турлари кўп бўлгани учун энг мураккаб ҳисобланади. Бунда, нафақат самарали бошқариш, балки амалий дастурчиларга қурилмалардан абстрактлаштиришга имкон берадиган қулай ва самарали виртуал интерфейсни таъминлайди. Бошқа томондан, параллел бажариладиган талай масалаларни киритиш-чиқариш қурилмаларига мурожаатни таъминлаш талаб қилинади. Баъзи КЧ дастурчиларидан баъзилари қурилмалардан мустақилдир, уларни КЧ кўпгина қурилмаларига, қўллаш мумкин.

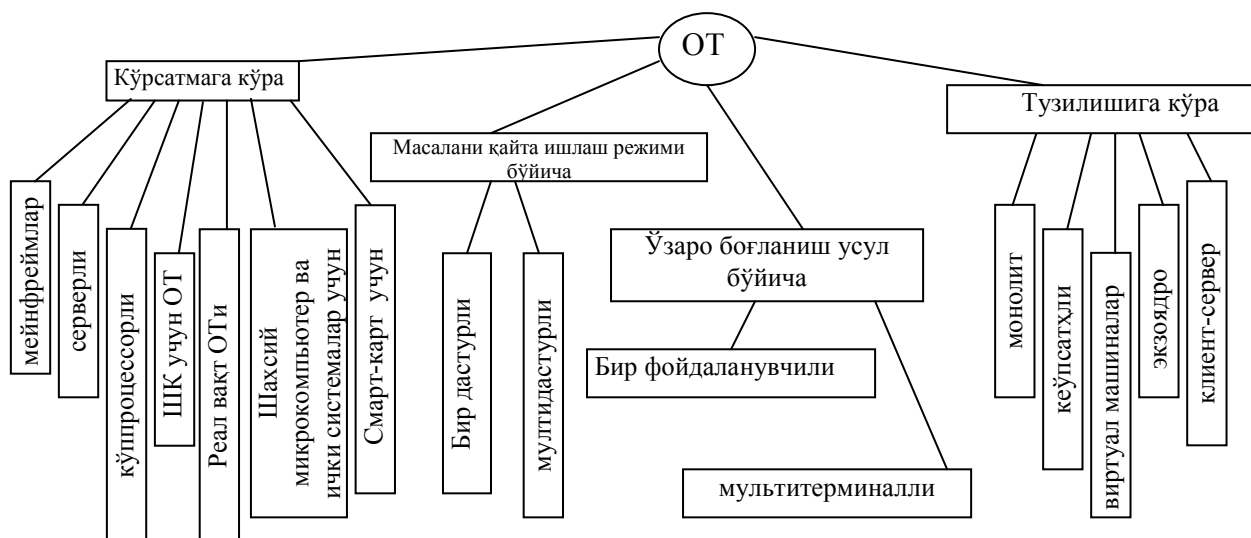
Файл – бир хил тузилишга эга бўлган ёзувлар мажмуаси кўринишида ташкил этилган маълумотлар тўпламидир.

Файл тизими, фойдаланувчига маълумотлар тузилишининг мантикий даражаси ва амаллар билан иш кўриш имконини беради. Файл тизими,

маълумотларни дискда ёки бошқа маълумот жамламасида ташкил қилиш усулини белгилайди.

Барча замонавий ОТ лар ўз файл бошқарув тизимига эга. Улар замонавий ОТ ларнинг аксарият кўпчилигида асосий ҳисобланади. ФБТ и диск соҳасини марказлашган ҳолда тақсимлаш ва маълумотларни бошқариш муаммосини ечади.

ФБТ, фойдаланувчилардан киритиш-чиқариш фурилмалари ва диск хусусиятларини яширган ҳолда, файл ва каталоглар билан ишлаш учун кенг хизмат (сервис) имкониятларини беради. UNIX да дискларда файл тизими ва асос файл тизими алоҳида-алоҳида мавжуддир ва улар бир-бирига боғлиқ эмас. UNIX да йўлга диск номи ёки унинг номерини кўшиш имконини бермайди, чунки бу ОТ учун қурилмаларга қатъий боғланишни келтириб чиқаради. “Mount” тизими чақириғи юмшоқ диск тизимини асос файл тизимидаги кўрсатилган жойга кўшиш (монтировка) имконини беради. UNIX да яна бир тушунча- махсус файллар тушунчасидир.



Расм ОТ ни синфларга ажратиш.

ОТ тизимли бошқарувчи ва ишлов берувчи дастур мажмуасидан иборат бўлиб, булар дастур модуллари ва маълумотлар тузилмаларининг ўзаро боғланганмураккаб тузилмасидир, улар ҳисоблашларнинг ишончли ва самарали бажарилишини таъминлаши зарурдир. ОТ нинг кўпгина потенциал имкониятлари, унинг техник ва истеъмол параметрлари – буларнинг ҳаммаси, тизим архитектураси ОТ нинг структураси, қурилиш принциплари билан белгиланади.

Шу вақтгача биз ОТ ларга ташқи тарафдан назар солдик, яъни ОТ қандай функцияларни бажаради. Энди эса, ОТ ларни ички қисмини ва уларни қуришда қандай ёндашишлар мавжудлигини кўриб чиқамиз.

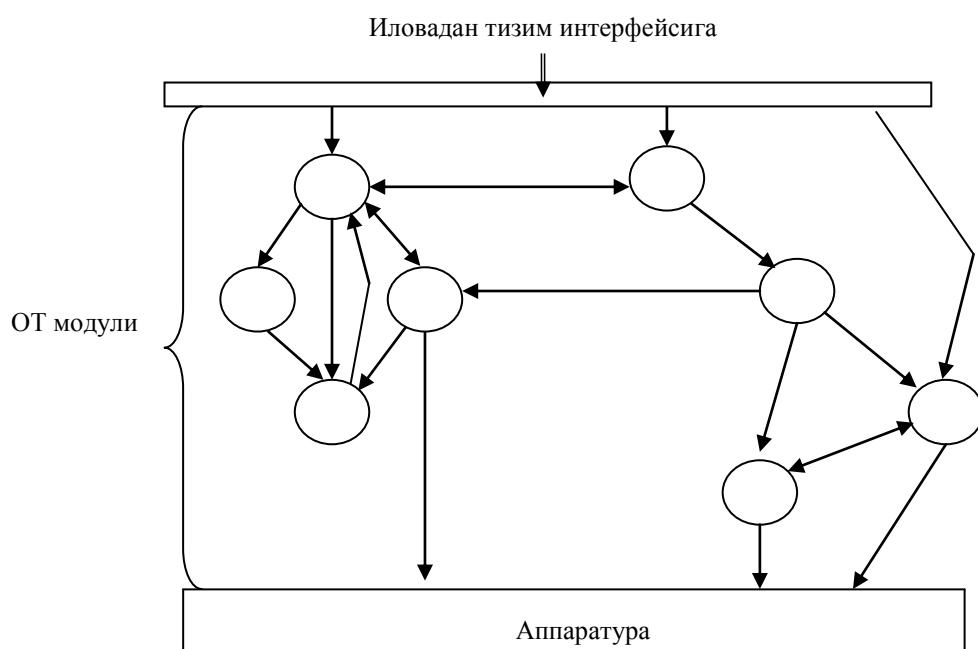
Замонавий ОТ ларга қўйиладиган талабларни қондиришда унинг структурали қурилиши (тузилиши) катта аҳамиятга эгадир, ОТ лар ўз ривожланиш босқичларида, монолит тизимлардан то яхши

структуралаштирилган, ривожланиш хусусиятига эга бўлган, кенгайтириш ва янги платформаларга ўтказувчанлик хусусиятига эга бўлган тизимларгача бўлган йўлни босиб ўтдилар.

Монолит тизимлар (монолит ядро).

Моҳиятига аҳамият бериладиган бўлса, ОТ-оддий дастурдир, шунинг учун ҳам, уни деярли барча дастурлар каби ташкил этиш, яъни процедура ва функциялардан иборат кўринишда яратиш тўғрироқ бўлади. Бу ҳолда ОТ компоненталари мустақил модуллардан иборат бўлмай, балки битта катта дастур ташкил этувчилардан иборат бўлади. ОТ нинг бундай тузилиши монолит ядро (monolithic kernel) дейилади.

Умумий ҳолда, монолит тизим “структураси” унинг структураси йўқлигидир. ОТ, ҳар бири зарур вақтда бир-бирини чақириши мумкин бўлган процедуралар тўпламидан иборатдир. Бу ҳолда, тизимнинг ҳар бир процедураси яхши йўлга қўйилган интерфейсга эгадир. Бундай монолит тизимни кўриш учун, ҳамма алоҳида процедураларни компиляция қилиб, кейин уларни ягона объект файлига бирлаштирилади (компоновкачи ёрдамида). Бундай тизимларга, UNIX нинг илк версиялари ва Nowell Net Ware мисол бўла олиши мумкин. Ҳар бир процедура бир-бири билан боғлана олади. Модулли структурадан фарqli равишда, модулли структурада ҳар бир процедурани махсус кириш нуқталари орқали чақириш мумкин.



5.2-расм. Монолит ОТ ни оддий структуралаштириш.

Хатто бундай монолит тизимлар ҳар барибир озгина структуралаштирилган бўлади. ОТ қўллайдиган тизимли чақириқларга мурожаатда параметрлар қатъий белгиланган. М-н, регистрлар ёки стек каби жойларга жойлаштирилади, кейин эса, ядро чақириғи ёки супервизор чақириғи каби махсус узилиш командалари бажарилади. Бу команда

машинани фойдаланувчи режимдан ядро режимига (супервизор режими ҳам дейиш мумкин) ўтказди ва бошқарувни ОТ га узатади. Кейин эса ОТ, қайси тизимли чақириқ бажарилиши кераклигини аниқлаш учун, чақириқ параметрларини текширади. Бундай сўнг, ОТ процедураларга кўрсаткични ўз ичига олган жадвални белгилайди ва мос процедурани чақиради.

Бундай ташкил этилган ОТ қуйидаги структурага эга бўлади:

1. Бош дастур – у талаб қилинадиган сервис процедураларини чақиради.
2. Тизимли чақириқларни амалга оширадиган хизматчи процедуралар тўплами.
3. Хизматчи процедураларга хизмати қиладиган утилиталар тўплами.

Бу моделда, ҳар бир тизимли чақириқ учун биттадан хизматчи процедура мавжуддир. Утилиталар, бир нечта сервис процедураларга керак бўлган функцияларни бажаради. процедураларни учта қатламга бўлиш қуйидаги расмда кўрсатилган.

Кўп сатҳли тизимлар.

Олдинги ёндашишни умумлаштирилгани, ОТ ни сатҳлар (қатламлар) иерархияси сифатида ташкил этишдир. ОТ функциялари қатламларни ташкил этади, м-н, файл тизими, жараёнларни ва қурилмаларни бошқариш ва х.к. Ҳар бир қатлам, фақат ўзининг бевосита қўшниси, юқорисидаги ёки пастдаги қатламлар билан алоқа қила олади. Амалий дастурлар ёки ОТ нинг модуллари сўровни, шу қатламлар бўйича юқори ва пастга узатиш мумкин.

Шу тахлитда қурилган биринчи тизим, Дейкстра ва унинг талабалари томонидан 1968 йилда қурилган бўлиб, ТНЕ тизими деб аталади.

Тизим 6 та қатламдан иборат эди, 0-чи қатлам, процессор вақтини тақсимлаш билан машғул бўлади, у жараёнларни узилиши бўйича ёки вақт ўтиши билан бошқа жараёнларга ўтказди.

1-чи қатлам хотирани бошқарган, у оператив хотира ва магнит баробани соҳасини, оператив хотирада жой етмаган жараёнлар қисмларига (саҳифалар) тақсимлаган, яъни 1-чи қатлам виртуал хотира функциясини бажарган.

2-чи қатлам оператор консоли ва жараён ўртасидаги алоқани бошқарган. Бу қатлам ёрдамида, ҳар бир қатлам ўзининг шахсий оператор консолига эга бўлади.

3-чи қатлам ёрдамида ҳар бир жараён, аниқ қурилмалар билан ишлаш ўрнига (уларнинг турли хусусиятларини ҳисобга олган ҳолда) фойдаланувчига қулай характеристикаларга эга бўлган киритиш-чиқариш абстракт қурилмаларга мурожаат этган.

4-чи қатламда, кириш-чиқиш қурилмаларини бошқариш, хотира ва консол, жараёнлар билан ҳеч қандай алоқаси бўлмаган фойдаланувчи дастурлари ишлаган.

ТНЕ тизимида, кўп қатламли схема ишлаб чиқиш (целим разработки) мақсадларига хизмат қилган, чунки кеун тизимнинг ҳамма қисмлари умумий объект модулига компановка қилинган.

Кўп қатламлилиқнинг кенг умумлаштирилиши MULTICS ОТ ларида амалга оширилди. MULTICS тизимида ҳар бир қатлам (халқа деб аталувчи) юқорида жойлашганидан кўра имтиёзлироқдир.

Юқори қатламдаги процедура, қуйидаги процедурани чақирмоқчи бўлса, у мос тизимли чақириқни бажариши лозим, яъни чақириқ бажарилиши олдида параметрлари синчиклаб текшириладиган TRAP (узилиш) командаси бажарилши лозимдир.

MULTICS да ОТ, ҳар бир фойдаланувчи жараёнининг адрес макони қисмидан иборат бўлса ҳам, аппаратура, маълумотлар химоясини хотира сегментлари даражасида химоя қилади. Масалан, маълум сегментларга фақат ёзиш учун, бошқаларига фақат ўқиш ёки бажариш учун рухсат беради.

MULTICS да ёндошишнинг устунлиги шундан иборатки, у фойдаланувчи тизими структурасига ҳам кенгайтирилиши мумкин. М-н, профессор- талабалар дастурларини баҳолаш ва тестлаш учун дастур ёзиши мумкин, ва бу дастурни n -чи қатламда ишга тушириши мумкин, бу вақтда талаба дастурлари $n+1$ чи қатламда ишлайди, чунки улар ўз баҳоларини ўзгартира олмайди.

Кўп қатламли ёндашиш, UNIX ОТ ининг турли вариантларини амалга оширишда ишлатилган.

Амалда, бундай структурали ёндашиш одатда ёхши ишлайди, бугунги кунда у кўпинча монолит тарзда қабул қилинади.

Кўп қатламли структурага эга бўлган тизимларда, битта қатламни оиб ташлаб, бошқаси билан алмаштириш, қатламлар орасидаги интерфейснинг ўзига хослиги учун, мураккабдир. Янги функцияларни қўшиш ва мавжудларини ўзгартириш ОТ ни мукамал билиш кўп вақтни талаб қилади. ОТ лар узоқ яшайдилар ва кенгайиш ва ривожланиш имкониятига эга бўлишлари кераклиги учун монолит ёндашиш назардан қолди ва унинг ўрнига клиент-сервер модели ва у билан қатъий боғланган микроядро концепцияси келди.

Клиент-сервер ва микроядро модели

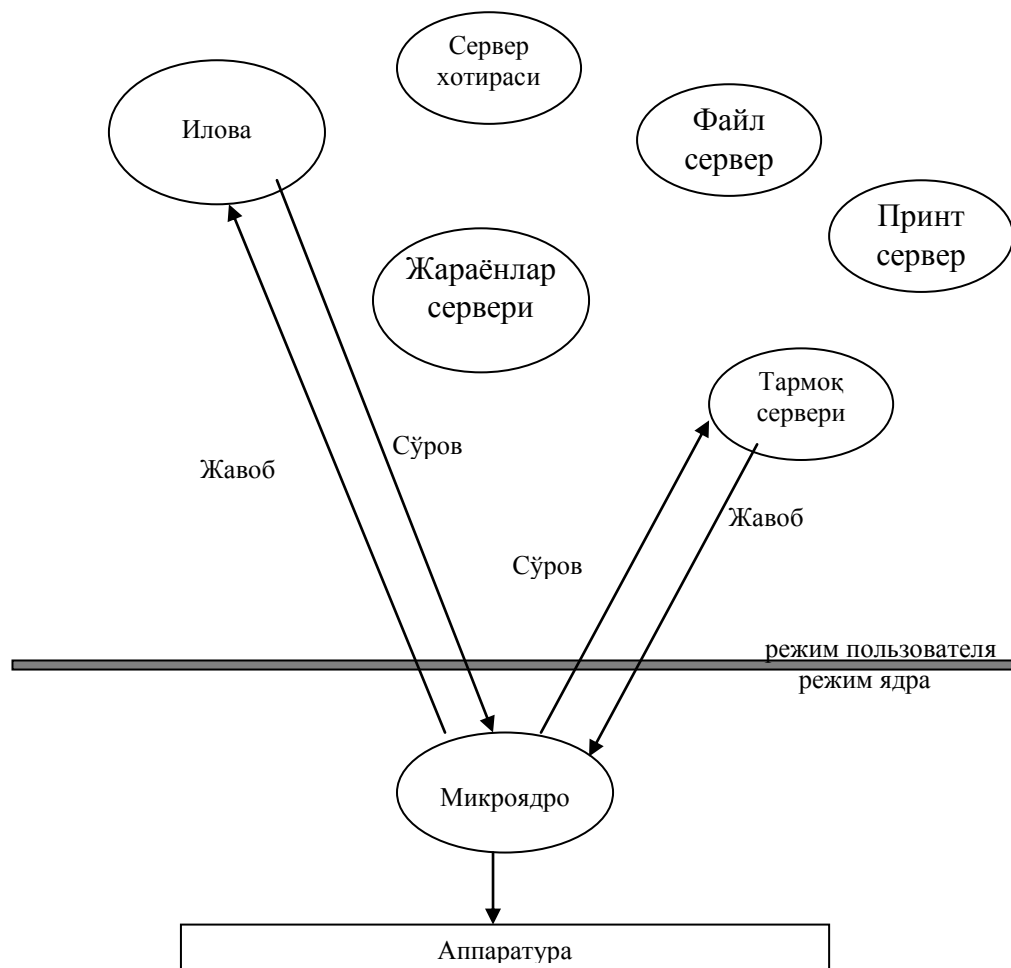
Клиент-сервер модели – ОТ ни структуралаштиришга яна бир ёндашишдир. Кенг маънода клиент-сервер модели, бирор-бир сервис (хизмат) истеъмолчиси- мижоз (клиент) – дастур компонентаси, ва бу сервисни таъминловчи – сервер-дастур компонентаси мавжудлигини тақозо (назарда тутати) этади.

Клиент ва сервер ўртасидаги ўзаро алоқа шундай стандартлаштириладигани, сервер ҳар хил усуллар билан амалга оширилган ва, балки ишлаб чиқувчи корхоналари турли бўлган клиентларга хизмат қилиши мумкин. Бунда асосий шарт, улар сўровларни серверга тушунарли усулда беришлари керак.

Одатда айирбошлаш (обмен) сабабчиси (инициатор) клиент ҳисобланади, у кутиш ҳолатидаги серверга хизмат кўрсатишни чўраб

сўровнома юборади. Битта дастур компоненти баъзи хизматга нисбатан сервер бўлиши мумкин.

Бу модел муваффақиятли тарзда нафақат ОТ ларини куришда, балки дастурий таъминотнинг барча кўринишларида (уровнях) қўлланилади, фақат баъзи ҳолларда тор-махсус маънога эгадир.



5.2-расм. ОТ клиент сервер структураси.

ОТ ни структуралаштириш ғоясига нисбатан олганда ғоя шундан иборатки, бунда уни бир нечта жараёнларга-серверларга бўлинади, уларнинг ҳар бири алоҳида сервис функциялар мажмуасини бажаради – м-н, хотирани бошқариш, жараёнларни яратиш ёки режалаштириш. Ҳар бир сервер фойдаланувчи режимида бажарилади. ОТ ни бошқа компонентаси ёки амалий дастур, сервисни сўраб серверга хабар жўнатади. ОТ ядроси (микроядро) имтиёзли режимида ишлаган ҳолда, керакли серверга хабар еткази, сервер операцияни бажаради, ундан сўнг ядро, бошқа хабар билан клиентга натижани жўнатади.

Бунда микроядродан фойдаланиш, ОТ функцияларини вертикал тақсимланишини горизонтал тақсимланишига алмаштириди.

Микроядродан юқорида жойлашган компоненталар, микроядро орқали юбориладиган хабарлардан фойдаланса ҳам бир-бири билан бевосита алоқа қиладилар. Микроядро йўналтирувчи ролини ўйнайди. У хабарни текширади,

серверлар ва клиентларга жўнатади ва аппаратурага мурожаат имконини беради.

Бу назарий модел клиент-сервер тизимини мукамал тавсифидир, унда ядро фақат хабар узатиш воситаларидан иборатдир.

Ҳақиқатда эса, ОТ структурасида амалга оширилган клиент-сервер модели турли вариантлари, бир-биридан ядро режимида бажарадиган ишлари хажми билан тубдан фарқ қилиш мумкин.

Микроядро ОТ асосида ётадиган, энг муҳим функцияларни амалга оширади. Бу нисбатан муҳим бўлмаган тизим хизмат ва иловалар учун базисдир.

Масала айнан шундан иборатки, тизимли функциялардан қайси бирини аҳамиятсиз деб ҳисоблаб мос равишда, ядрога қўшмасликдир, бу масала кўп вақтдан бери микроядро ғояси тарафдорлари орасида кўпгина баҳсларга сабаб бўлмоқда. Умумий ҳолда, ОТ нинг анъанавий тарзда ажралмас ҳисобланадиган қисмлари-файл тизимлари, ойналарни бошқариш ва хавфсизликни таъминлаш – ядро билан ва ўзаро алоқа қиладиган периферик модуллар бўлиб қолмоқда.

Бажариладиган ишларни (функцияларни) микроядро ва уни ўраб турган модуллар орасида тақсимлаш асосий принципи қуйидагичадир: микроядро таркибига фақат супервизор режимида ва имтиёзли маконда ижро этиладиган функциялар киритилади. Булар машинага боғлиқ функциялар (бир нечта процессорни қўллаш функцияси ҳам), жараёнларни бошқариш баъзи функциялари, узилишларга ишлов бериш, хабар жўнатишни қўллаш, киритиш-чиқариш қурилмаларини бошқариш баъзи функциялари (қурилмаларни регистрга командаларни юклаш билан боғлиқ бўлган). От нинг бу функцияларини фойдаланувчи маконида ишляпган дастурларнинг бажариши қийин, баъзи вақтда эса мумкин эмас.

Бу муаммони ечишнинг икки йўли бор. Биринчи йўл, бир нечта, процессор иш режимида сезгир серверларни ядро маконига жойлаштиришдир, бу эса ўз навбатида уларга аппаратурага тўлиқ мурожаатни таъмилайди, ва шу билан бирга бошқа жараёнлар билан оддий усул билан алоқани таъминлайди. Бундай ёндашиш, м-н, Windows NT ни ишлаб чиқишда фойдаланилган: микроядродан ташқари, имтиёзли режимида, executive бошқарувчи дастур деб аталувчи қисми ишлайди. У ўз ичига виртуал хотирани, объектларни, киритиш-чиқариш ва файл тизимларини (тармоқ драйвеларини) жараёнлар алоқасини ва қисман хавфсизлик тизимини бошқарувчи компоненталарни олади.

Бошқа йўл, ядрога, фақат сервернинг, ечимини амалга оширадиган механизмидан иборат қисмини қолдиришдан иборат, ечимни қабул қилишга жавоб берадиган қисмини эса фойдаланувчи соҳасига жойлаштирилади.

Қурилмалар драйверлари ядро ташқарисида ҳам, ядро ичида ҳам жойлашган бўлиши мумкин. Қурилмалар драйверини ядродан ажратиш, От динамик конфигурациясини мумкинлигини имконини беради. Динамик конфигурациядан ташқари, қурилмалар драйверларини фойдаланувчи режими жараёнлари сифатида қабул қилишнинг бошқа сабаблари ҳам

мавжуд. МББТ, м-н, ўз драйверига эгадир. Бу ёндошиш, тизимни кўчириб ўтказиш имконини оширади, чунки кўп ҳолларда қурилмалар драйверлари аппарат қисмдан абстракцияланади.

Ҳозирги вақтда айнан-клиент сервер модели ва микроядро концепциясидан фойдаланиб қурилган ОТ лар, энг юқори даражада, замонавий ОТ ларга қўйиладиган талабларга жавоб беради.

Кўчириб ўтказишнинг юқори даражаси, бутун машинага боғлиқ код, микроядро изоляция (ажратилади) қилинади, шунинг учун ҳам тизимни янги порцессорга кўчириш кам ўзгаришларни талаба қилади ва уларнинг ҳаммаси мантиқан биргаликда гуруҳлаштирилган.

Микроядро технологияси, кўпгина турли ОТ лар учун ёзилган дастурлар мутаносиблигини таъминлайдиган амалий муҳитлар асосидир. Қуйида жойлашган ОТ дан, амалий дастурлар интерфейсини абстракциялаштириб, микроядро амалий дастурларга сарфланган маблағлар бир неча йил давомида ҳатто ОТ ва процессорлар алмашса ҳам бекорга сарф бўлмайди.

Кенгайтиришлик хоссаси ҳам, замонавий ОТ ларга қўйилган муҳим талаблардан биридир. ОТ, DOS каби кичик, ёки UNIX каби каттами, барибир маълум вақтдан сўнг, унинг конструкциясига бошидан киритилмаган хоссаларга эга бўлиши учун, уни кенгайтириш зарур бўлиб қолади. Монолит ОТ ларнинг ўсиб бораётган мураккаблиги ОТ га ўзгартириш киритишни (унинг кейинчалик иши ишончли бажариладиган ҳолда) қийинлаштиради, баъзи ҳолларда эса имкон ҳам бермайди.

Микроядронинг қатъий белгиланган интерфейсининг чегараланган мажмуаси ОТ ривожланишининг мунтазам ўсишига йўл очади.

Одатда ОТ фақат ядро режимида бажарилади, амалий дастурлар эса, улар ядрога тизимли функцияларни бажариш учун ядрога мурожаат қилган ҳоллардан бошқа ҳолларда фойдаланувчи режимида бажарилади. Микроядро қурилган тизимлар, оддий тизимдан фарқли равишда, ўзининг сервер тизимларининг оддий амалий дастурлар каби, фойдаланувчи режимида бажаради. Бундай тузилиш (структура) микроядро яхлитлигига таъсир кўрсатмасдан серверларни ўзгартириш ва қўшиш имконини беради.

Клиент-сервер моделидан фойдаланиш ишончлиликини оширади. Ҳар бир сервер ўз хотира соҳасида алоҳида жараён сифатида бажарилади, ва шундай қилиб бошқа жараёнлардан ҳимояланган. Агар алоҳида сервер бузилса, у тўхтамасдан ва ОТ нинг бошқа қисмларини бузмасдан қайта ишга тушириш мумкин.

Бу модел, тақсимланган ҳисоблашлар учун жуда мосдир, чунки алоҳида серверлар мультипроцессорли компьютерда ёки ҳатто ҳар хил компьютерларда ишлаши мумкин.

Объектга мўлжалланган ёндашиш.

Микроядро технологияси, мунтазам равишда ривожланадиган модулли тизимларга асос бўлган бўлса ҳам, у тўлиқ равишда тизимни кенгайтириш имконини таъминлай олмади.

Ҳозирги вақтда, бу мақсадга энг тўғри келадиган объектга мўлжалланган ёндашиш, бу ҳолда дастурни ҳар бир компонентаси функция жиҳатдан бир-биридан (бошқаларидан) ажратилгандир.

Бу ёндашишнинг асосий тушунчаси “объект”дир. Объект-дастур ва маълумотларнинг бирлиги бўлиб, у бошқа объектлар билан ахборот узатиш ва қабул қилиш орқали боғланади. Объект аниқ буюм, амалий дастур ёки ҳужжатни ёки баъзи абстракциялар-жараён, ҳодисларни акс эттириши мумкин.

Объект дастурлари функциялари шу объект маълумотлари устида бажариладиган амаллар рўйхатини белгилайди. Объект-мижоз, объект-сервер функциясини бажарилишини сўраб бошқа объектга хабар юбориши мумкин.

Объектлар моҳиятни тавсифлаш мумкин. Умумий объектлардан аниқроқ объектларни келтириб чиқарувчи, мерос бериш механизми хоссалари мавжуддир.

Объект маълумотлари ички тузилиши кузатишдан яшириндир. Объект маълумотларини ихтиёрий тарзда ўзгартириб бўлмайди. Объектдан маълумот олиш ёки маълумотларни объектга жоёлаштириш учун, мос объект функцияларини чақириш керак бўлади. Бу ҳол объектни, ундан фойдаланадиган коддан ажратади. Дастурчи бошқа объект функцияларига мурожаат қилиши, ёки бошқа объект улар тузилишини билмай туриб хоссаларини мерос қилиш йўли билан янги объект куриши мумкин. Бу хосса инкапсуляция дейилади.

Шундай қилиб, объект ташқи дунё учун, яхши ишлаб чиқилган интерфейсли “қора қути” кўринишида намоён бўлади. Бу хусусият, уларда мавжуд иловаларни объект кўринишида, уларда ҳеч нарсаларни ўзгартирмасдан, жойлаштиришга имкон беради.

Объектга мўлжалланган ёндашишдан фойдаланиш айниқса тизимли дастурчилар учун қулайдир, чунки тизимли даражадаги объектлардан фойдаланиб, тизимни яхлитлигини бузмасдан туриб, уларни ўзларига мослаштириб ОТ га чуқур кириб бориши мумкин.

Аммо бу ёндашиш тақсимланган ҳисоблаш муҳитларини амалга оширишда келажакда муҳим натижа беради.

Объектга мўлжалланган ёндашиш ўзига яраша муаммоларга эга, аммо шунга қарамасдан, бу ёндашиш дастурий таъмотни конструкциялашда энг муҳим аҳамиятга эга йўналишдир.

**5-мәруза. Операцион тизимларда жараён ва ресурс тушунчаси.
Ресурсларнинг умумий синфларга ажратилиши. Ресурслар тузилиши,
ташкили элементлари, амалга ошириш шакллари ва х.к. лар буйича
синфларга ажратилиши.**

Умумий ҳолда ҳар истеъмол қилинадиган (унинг), истеъмолчилар учун маълум қийматга эга бўлган объект ресурсдир. Ресурслар захирадаги ажратиладиган бирлигига қараб, чекланган ва чекланмаган бўлади. Ресурс чекланганлиги истеъмолчилар орасида ҳаётий заруриятларга олиб келади. Заруриятларни тартибга солиш учун, ресурслар истеъмолчилар орасида уларни юқори даражада қониқтирадиган маълум қоидалар бўйича тақсимланиши зарур. Ресурсларнинг умумий классификациясини қуйидагича тасаввур қилиш мумкин:

- мавжудлиги ҳақиқийлигига қараб: физик ва виртуал
- хоссаларнинг кенгайтириш имконига қараб: мосланувчан (эластик) ва қатъий
- фаоллик даражасига қараб: фаол ва пассив
- мавжудлик вақтига қараб: асосий ва иккинчи даражали
- тузилиши (структура) жиҳатидан: оддий ва мураккаб
- типланиши бўйича: ишлаб чиқиладиган ва истеъмол қилинадиган
- амалга ошириш формаси бўйича: юмшоқ ва қаттиқ
- функционаллик ортиқчалигига қараб: қиммат ва арзон
- фойдаланиш характериға қараб: параллел фойдаланилувчи ва кетма-кет фойдаланилувчи.

Физик ресурслар деб, реал мавжуд бўлган ва уни фойдаланувчилар орасида тақсимланганда унга хос бўлган физик хусусиятларга эга бўлади (сақлаб қолади).

Виртуал ресурс-физик ресурснинг қандайдир бошқача моделидир. Виртуал ресурс, фойдаланувчи тасаввур қилган ҳолда мавжуд эмас. Модел сифатида виртуал ресурс дастурий-аппарат формада амалга оширилади. Бу маънода виртуал ресурс мавжуддир. Аммо виртуал ресурс фойдаланувчига у билан ишлаш вақтида, нафақат физик ресурс, яъни моделлаштириш объектига хос хоссаларнигина эмас, балки унга хос бўлмаган хоссаларни ҳам намоён қилади.

Хоссаларни кенгайтириш имконияти белгиси ресурсни куриш имкониятини характерлайди. Виртуаллаштиришга, яъни ўз хоссаларини кенгайтириш имконини берадиган ресурс-мосланувчан ёки эластик дейилади. Қатъий ресурс, бу ўз ички хоссалари бўйича виртуаллаштиришга йўл қўймайдиган ресурсга айтилади.

Фаол (актив) ресурсдан фойдаланишда, у бошқа ресурс ёки жараёнларни (ёки ўзига нисбатан ҳам) ўзгартирувчи амалларни қўллайди ёки ўзгартиришга олиб келади. Пассив ресурс бундай хусусиятга эга эмас. Бундай объект устида мумкин бўлган амаллар бажарилади, бунда унинг ҳолати ўзгаради, яъни ички ёки ташқи характеристикалари. Марказий

процессор – фаол ресурс, талаб бўйича ажратиладиган хотира пассив ресурсдир.

“Мавжудлик (существование) вақти” белгисига қараб фарқлаш, улардан фойдаланадиган жараёнларга нисбатан олинган динамикасига қараб олиб борилади. Агар ресурс тизимда жараён туғилишигача мавжуд бўлса ва жараён мавжудлиги интервалида мурожаат қилиш мумкин бўлса, у ҳолда бу ресурс айни жараён учун доимийдир. Вақтинчалик ресурс кўрилаяпган жараён мавжудлик вақтида тизимдадинамик ҳолда пайдо бўлиши ва йўқ қилиниши шу жараён орқали ёки бошқа жараёнлар-тизимли ёки фойдаланувчи жараёнлари орқали олиб борилиши мумкин. Демак, ресурслар жараёнларнинг тизимли ўзаро боғланиш маълум қоидалари бўйича ажратилади. Шунинг учун ҳам баъзи жараёнларга нисбатан доимий бўлган ресурс бошқасига нисбатан вақтинчалик ёки акси бўлиши мумкин.

Ресурсларнинг муҳимлиги даражасига нисбатан синфларга ажратиш зарурияти, иккита сабаб билан белгилангандир: зарур ишловчанлик билан таъминланиши ва жараёнларни бошқариш ва ресурсларни тақсимлашдаги мосланувчанлигини ошириш. Бунинг учун асосий ва иккинчи даражали ресурслар ажратилади. Ресурс аниқ жараёнга нисбатан асосий бўлади, шу ҳолдаки, агар унинг тақсимланишисиз жараён ривожлана олмаса. Бундай ресурсларга энг аввало марказий процессор ва оператив хотира киради. Агар ресурслар тақсимланмаганда жараённи альтернатив ривожланишига йўл қўйса, бундай ресурслар иккинчи даражали ресурслар дейилади. (м-н, МД.МЛ).

Ресурсларни қиммат ва арзон ресурсларга бўлиниши, уларни тақсимлашдаги функционал ортиқчалиги принципини амалга оширишдан келиб чиқади. Фойдаланувчи олидида танлаш масаласи туради – керак ресурсни тез олиб бу хизмат учун қиммат нарх бериш, ёки талаб қилинган ресурс тақсимланишини кутиб, ундан фойдаланилганда арзон нарх тўлаш. Тизимда альтернатив ресурслар мавжуд бўлса, улардан фойдаланишнинг ҳар хил нархлари киритилади.

Тузилиши жиҳатидан олинган белги, ресурсда бирор структура (тузилиш) борлигини кўрсатади. Агар ресурс ташкил этувчи элементлардан иборат бўлмаса ва тақсимланганда бир бутун-яхлит ҳолатда олинса у оддий дейилади. Мураккаб ресурс маълум структура билан белгиланади. У ўз таркибида бир хил характеристикага эга бўлган (фойдаланувчи нуктаи-назаридан) қатор элементлардан ташкил топгандир. Фойдаланувчи-жараёнларга, уларнинг ресурсга талаблари бўйича, мураккаб ресурснинг қайси элементлари ажратилишининг фарқи йўқдир. Оддий ва мураккаб ресурслар ҳолатлари сони билан ҳам фарқланади. Оддий ресурс, агар бирор-бир жараёнга фойданишга берилса-банд, ёки бўш бўлиши мумкин. Мураккаб ресурс, агар унинг элементларидан бирортаси ҳам фойдаланиш учун тақсимланмаган бўлса “бўш” ҳолатида бўлади. Агар унинг ҳамма элементлари фойдаланиш учун берилган бўлса, у “банд” ҳолатида, агар ресурслар элементлари қисми тақсимланган бўлса, у ҳолда ресурс “қисман банд” бўлади.

Ресурсларни у ёки бу асосга нисбатан ресурсларни тақсимлаш механизмини кўраяпганда тақсимланадиган ресурслар характери ҳисобга олиш муҳим аҳамиятга эгадир. Бу белгига қараб ҳам ресурс моҳияти ҳисобга олинади ва ресурсдан фойдаланилгандан сўнг унинг қайта тикланиши имконияти ҳисобга олинади. Тикланиш имкониятига қараб, ресурслар ишлаб чиқилувчи ва истеъмол қилинувчиларга бўлинади.

Ҳар бир ресурсга нисбатан фойдаланувчи-жараён учта типдаги ҳаракатни бажаради деб фараз қилинади: талаб, фойдаланиш ва бўшатиш. Агар тизим томонидан ресурсни тақсимлашда бажариладиган амаллар кўп такрорланувчи “талаб-фойдаланиш-бўшатиш” кетма-кетликда бажариши мумкин бўлса, у ҳолда бундай ресурс ишлаб чиқариладиган ресурс дейилади. У қайтгандан сўнг, бошқа жараён томонидан ишлатилиши мумкин бўлади. Шунинг учун ҳам, ресурсдан ҳар гал фойдаланилгандан сўнг, унинг ўзгариш кўринишини ҳисобга олинмаса, ресурс ҳаёт вақтини, у ўз функционал хоссаларини йўқотмагунча чексиз катта ёки етарли даражада катта деб ҳисоблаш мумкин. Маълум категориядаги ресурсларга нисбатан амалларни қуйидаги тартибда бажарилиши тўғри бўлади: бўшатиш-талаб-фойдаланиш, ундан кейин эса истеъмол қилинадиган деб аталувчи ресурс истеъмол сферасидан олиб ташланади (м-н, - ишлаб чиқувчи – истеъмолчи муносабати). Истеъмол қилинадиган ресурсни ҳаёт муддати, яъни бўшатиш ва фойдаланиш амаллари бажарилиши орасидаги вақт билан белгиланади ва у чеклидир. Ишлаб чиқувчи жараёни ва истеъмолчи жараёнига нисбатан истеъмол қилинувчи ресурслар ўзларини вақтинчалик каби тутадилар.

Ресурс табиати ва (ёки) фойдаланиладиган ресурсни тақсимлаш қоидаси, бир нечта жараён ўртасида тақсимланадиган ресурсдан параллел ёки кетма-кет фойдаланиш схемаси билан белгилангандир. Кетма-кет схема, кетма-кет фойдаланиладиган деб аталувчи ресурсга нисбатан, вақт бўйича қуйидаги қатъий амаллар занжири “талаб-ижро-бўшатиш” бажарилишини назарда тутди. Параллел жараёнлар учун бундай амаллар занжири критик соҳадан иборатдир ва улар олдиндан белгиланган бир-бирини рад этиш қоидасига асосан бажарилиши керак. Шунинг учун ҳам, кетма-кет фойдаланиладиган ва бир нечта параллел жараёнлар орасида тақсимланадиган ресурслар – критик ресурс деб аталади. Параллел схема, бир вақтнинг ўзида параллел фойдаланувчи битта ресурсдан фойдаланишни назарда тутди, шунинг учун ҳам бирдан ортиқ жараёндан параллел фойдаланувчи деб аталади. Бундай фойдаланиш, ҳар бир жараён мантикий ривожланишига ҳеч қандай хатоликка йўл қўймаслиги керак.

Амалга ошириш формал бўйича юмшоқ ва қаттиқ ресурсларга бўлинади. Қаттиқ ресурслар деганда, машина аппарат компонентаси инсон ресурси тушунилади, қолган ҳамма ресурслар юмшоқ ресурслардир. Қаттиқ ва юмшоқ ресурслар орасидаги фарқ мураккабликдан, ва нархидан ташқари уларнинг бузилиш ва рад этиш ҳолатларига қатъийлиги ва ишлаш қобилятининг тикланишидир. Юмшоқ ресурслар синфида икки типни ажратамиз, дастурли ва ахборот ресурсларига.

Агар юмшоқ ресурс нусха олишга йўл қўйса, ва ресурс-оригиналдан ва ресурс-копиядан фойдаланиш бир хил бўлса, у ҳолда бундай ресурс дастурли юмшоқ дастурдир. Акс ҳолда уларни ахборот ресурс типига киритиш мумкиндир, булар дастурлар, файллар, массивлар. Юмшоқ ахборот ресурслари умуман нусха олишга йўл қўймайдлар, агар йўл қўйса, у вақт функциясидир. Бу ҳар хил турдаги истеъмол ресурслардир: хабар, узилиш сигналлари, ОТ га, ҳар хил хизматларга бўлган талаб, синхронлаштириш сигналларидир. Бундай хабар ва сигналлар маълумотли аҳамиятга эгадир (аммо фақат чекли вақт интервали асосида). М-н, хотира ячейкасига, вақти-вақти билан баъзи хабарларни ёзиб турилса, у ҳолда аниқ хабарни бошқа янги хабар келгунча ячейкага ёзиш вақтида, ундан нусха олиш мумкиндир. Кейинги нусха олиш, танланган хабардан фойдаланишдан бошқа натижа олишга олиб келади.

ОТ ларда ресурс тушунчаси, одатда, қайта такрор фойдаланиладиган ресурсларга нисбатан, нисбатан қатъий ва етишмайдиган объектларга нисбатан ишлатилади, уларга талаб берилиши, фойдаланилиши ва озод бўлиши мумкиндир. Ресурслар тақсимланадиган бўлади, у ҳолда бир нечта жараёнлар улардан бир вақта фойдаланилади ёки параллел, ёки тақсимланмайдиган бўлади (у ҳолда ресурс фақат битта жараён томонидан ишлатилади).

Илк ОТларни ишлаб чиқиш вақтида процессор вақти, хотира, киртиш-чиқариш каналлари ва периферик қурилмалар ресурс ҳисобланган. Кейинчалик ресурс тушунчаси универсал ва умуйроқ бўлиб қолди. Уларга ҳар хил типдаги дастурий ва ахборот ресурслари (тизим нуктаи-назаридан улар объектҳисобланадилар) киради, уларни тақсимлаш мумкин ва уларга мурожаатни бошқариш мумкин.

Асосий ресурслардан бири процессордир. Бунда процессор фақат кўппроцессорли тизимлардагина ресурс сифатида чиқади, бир процессорли тизимларда эса процессор вақти ресурс ҳисобланади. Уни тақсимлаш параллел схема бўйича олиб борилади.

Кейинги ресурс кўриниши – хотирадир. У ҳам бир вақтнинг ўзида тақсимланиши (хотирада бир вақтда бир нечта жараён бор бўлса) ва параллел (хотира жараёнларга навбат билан берилади) тақсимланиши мумкин. Оператив хотирани жараёнлар орасида самарали тақсимлаш энг долзарб масаладир. Умумий ҳолда шахсан хотира ва унга мурожаат ҳар хил ресурслардир. Уларнинг ҳар бири бир-биридан боғлиқ бўлмаган ҳолда берилиши мумкин аммо хотира билан тўлиқ ишлаш учун иккалови зарурдир. М-н, ташқи хотира бир вақтда тақсимланиши, унга мурожаат эса навбат билан тақсимланиши мумкин.

Ташқи қурилма яна битта ресурс кўринишидир. Бевосита мурожаат механизми мавжуд бўлса, улар бир вақтда тақсимланадилар. Агар қурилма фақат кетма-кет мурожаатга эга бўлса, у тақсимланадиган ресурс ҳисобланмайди, м-н, принтер, магнит лентадаги жамлама.

Дастурий модулар ҳам ресурслардан бири бўлиб ҳисобланади. Бир марта фойдаланиладиган ресурслар фақат бир марта тўғри бажарилиши мумкин.

Иш вақтида улар ё ўз кодларини ёки берилган катталикларни бузишлари мумкин. Бундай модулар бўлинмайдиган ресурслардир. Қайта фойдаланадиган модулар имтиёзли, имтиёзли бўлмаган, реентерабел ва қайта кирадиган бўлиши мумкин.

Катталиклар (маълумотлар) ахборот ресурслар сифатида қаралиши мумкин. Булар ё оператив хотирадаги ўзгарувчилар ёки файллар бўлиши мумкин. Маълумотлардан фақат ўқиш учун фойдаланилса, уларни осон ажратиш мумкин. Жараёнларга бу кўринишдаги ресурсларни ўзгартиришга рухсат берилса, уларни бўлиш муаммоси анча мураккаблашади.

Яна шундай ресурслар борки, уларни эгаси жараёнлардан бемалол олиши мумкин, м-н, хотира. Баъзиларини эса эгасидан, ҳисоблаш натижасини йўқ қилмасдан олиб бўлмайди. М-н, компакт-дискка ёзишни тўхтатиб бўлмайди.

Назорат саволлари:

1. Ресурслар таърифи ва асосий ресурсларни айтиб беринг.
2. Ресурслар қандай белгилар бўйича синфларга ажратилади.
3. Структурага эга ресурсларни айтиб ўтинг.
4. Замонавий ва биринчи ОТлардаги қандай объектлар ресурслар деб ҳисобланган?
5. Ресурсларни виртуаллаштириш деганда нимани тушунасиш.
6. Ресурсларнинг қандай турлари мавжуд.

7-Маъруза. ОТларда жараён тушунчаси.

Режа:

1. Жараён тушунчаси.
2. Жараён ҳолати
3. Жараёнлар устидаги амаллар
4. Жараёнларни режалаштириш
5. Режалаштириш кўрсаткичлари ва алгоритмлари.

Операцион тизимларнинг ишини ўрганишда асосий тушунчалардан бири бўлиб, устида тизим аниқ амаллар бажарадиган асосий динамик объектлар сифатида олинадиган жараёнлар ҳисобланади.

Жараён тушунчаси.

Олдинги маърузуларда, ОТ тушунчасини қараб ўтганимизда, биз кўпинча “дастур” ва “топширик” сўзларини кўп ишлатдик. Масалан, ҳисоблаш тизими битта ёки бир нечта дастурни бажаради, ОТ топширикни режалаштиради, дастурлар маълумотлар алмашади ва хоказо. Биз бу сўзларни умумий ҳолда ишлатдик ва сиз ҳар бир аниқ ҳолда нимани тушунаётганингизни тахминан тасаввур қилар эдик. Аммо бир хил сўзлар ҳар хил ҳолатда масалан, ҳисоблаш тизими ишлов бермайдиган статик ҳолатдаги объектларни ҳам (масалан, дискдаги файллар) ва ижро жараёнидаги бўлган динамик ҳолатдаги объектларни ҳам билдирар эди. Бу ҳолат ОТ лар умумий хоссалари тўғрисида гапирганимизда, яъни унинг ички қурилмалари ва ўзини тутишига эътибор берилмаган ҳолатда мумкин эди. Аммо энди замонавий компьютер тизимлари ишини деталлашган ҳолда ўрганганимизда биз амалларни(терминологияни) аниқлаштиришимизга тўғри келади.

“Дастур” ва “топширик” терминаллари статик, фаол бўлмаган(неактивний) объектларни тавсифлаш учун ишлатилади. Дастур эса, бажарилиш жараёнида динамик, фаол объектга айланади. Унинг ишлаши борасида компьютер турли командаларга ишло беради ва ўзгарувчилар қийматларини ўзгартиради. Дастур бажарилиши учун, ОТ маълум сон оператив хотира ажратиши, унга киритиш чиқариш қурилмаларни ва файлларни боғлаши, яъни бутун ҳисоблаш тизими ресурслари ҳисобидан маълум қисмини резервлаб қўйиши керак. Уларнинг сони ва конфигурацияси вақт ўтиши билан ўзгариши мумкин. Бундай компьютер тизимлари ичидаги фаол объектларни тавсифлаш учун “дастур” ва “топширик” терминлари ўрнмга янги “жараён” терминини ишлатамиз.

Жараён ҳолати.

Кўпинча дабиётларда соддалаштириш учун, жараённи бажарилиш вақтидаги дастурни характерлайдиган абстрапена сифатида олиш тавсия этилади.

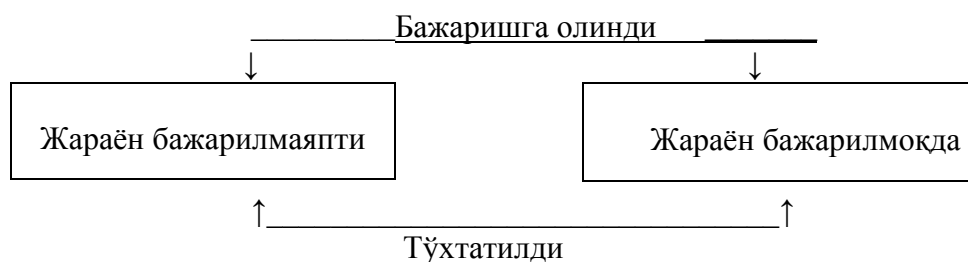
Жараён ОТ бошқаруви остида ҳисобланади. Бундай қабул қилишда ҳисоблаш тизимларида бажариладиган ҳамма нарс(фақат фойдаланувчи

дастурларигина эмас, балки ОТнинг ҳам маълум исимлари ҳам) жараёнлар тўплами сифатида ташкил қилингандир.

Бир процессорли компьютер тизимида вақтнинг ҳар бир момнотида фақат битта жараён бажарилиши мумкин. Мультидастурли ҳисоблаш тизимларида бир нечта жараённи псевдапараллел қайта ишлаш процессорни бир жараёндан иккинчисига ўтказиш ёрдамида амалга оширилади. Бир жараён бажарилгунча, қолганлари ўз навбати кутади.

Кўриниб турибдики ҳар бир жараён минимум икки ҳолатда бўлиши мумкин:

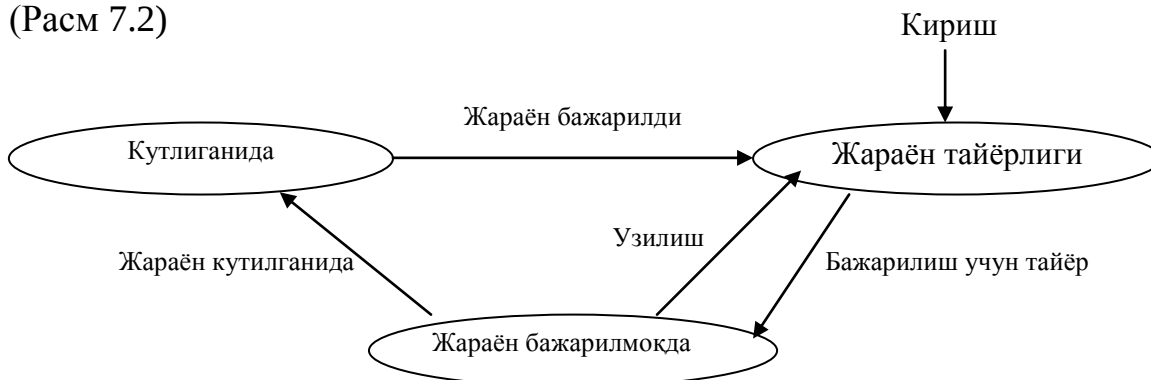
Жараён бажарилмоқда ва жараён бажарилмаяпти. Бундай моделдаги жараёнлар ҳолати диаграммаси қуйидаги расмда кўрсатилган.



Баъжарилаяпган ҳолатдаги жараён, маълум вақтдан сўнг ОТ томонидан тугалланиши, ёки тўхтатилиши ва яна баъжарилмаяпган ҳолатга ўтказилиши мумкин. Жараён тўхтатилиши иккита сабаб билан рўй бериши мумкин: уни ишини даво эттириши учун бирор бир ходиса талаб этилса, (масалан, кириш –чиқиш операциясини тугаллаш) ёки ОТ томонидан иш жараён учун ажратилган вақт тугаганда рўй беради. Шундан сўнг ОТ белгиланган алгоритм бўйича баъжарилмаяпган ҳолатидаги жараёнлардан бирини танлайди ва бу жараённи баъжарилаяпган ҳолатга ўтказди. Тизимда пайдо бўлаяпган янги жараён, бошқа жараён баъжарилмаяпган ҳолатга ўтказилади.

Бу жуда қулай моделдир. У баъжарилишга танланган жараён у тўхтатилишига сабаб бўлган ходисани кутиши ва амалда баъжаришга тайёр бўлмаслиги мумкин. Бундай ҳолатдан кутилиш учун жараён баъжарилмаяпган ҳолатни янги иккинчи ҳолатга бўламиз: тайёрлик ва кутиш ҳолатлари.

(Расм 7.2)



Расм 7.2. Жараён ҳолатининг янада тўлиқ диаграммаси

Тизимда пайдо бўлаяпган ҳар қандай янги жараён тайёрлик ҳолатига тушади. Операцион тизим режалаштиришнинг бирор бир алгоритмидан фойдаланиб, тайёр жараёнлардан бирини танлаб, уни бажарилиш ҳолатига ўтказди.

Бажарилиш ҳолатида жараён дастурий кодини бевосита бажарилиши рўй беради. Жараённинг бу ҳолатидан унча сабаб бўйича чиқиш мумкин:

- ОТ бу жараённинг фаолиятини тўхтатади;
- У ўз фаолиятини маълум ходиса рўй бермагунча давом эттира олмайди ва ОТ уни “кутиш” ҳолатига ўтказди;
- Хисоблаш тизимида узилиш рўй бериши билан(масалан, бажарилишга ажратилган вақт тугаши билан таймердан узилиш) уни тайёрлик ҳолатига ўтказилади.

Кутиш вақтлари тайёргарлик ҳолатига жараён, кутилаяпган ходиса рўй бериши билан ўтади ва у яна бажарилиш учун танланиши мумкин. Кейинчалик режалаштириш алгоритми ҳақида сўз борса, бизнинг моделда яна бир операция қўйилади: бу жараён приоритетини ўзгаришидир.

Жараённи яратиш ва тугаллаш операциялари бир мартталик операциялардир, чунки ортиқ қўлланилмайди баъзи тизимли жараёнлар, хисоблаш тизими иши вақтида ҳеч қачон тугалланмайди.

Жараён ҳолатини ўзгариши билан боғлиқ бўлган, ҳо у ишга тушириш ёки блокировка бўлсин, қоида бўйича кўп мартталик хисобланади.

Жараён контексти ва Process Control Block (жараён дискриптори)

ОТ, жараён устидаги амалларни бажара олиши учун, ҳар бир жараён ОТда маълум маълумотлар структураси сифатида тасвирланиши лозим бу структура(тузилма) шу жараёнга хос маълумотларни ўз ичига олади. Бу маълумотлар қуйидагилар:

- Жараён ҳолати
- Жараён дастурли(счётчиги) ҳисоблагичи, ёки бошқача қилиб айтганда, жараён учун кейинги бажариладиган команда адреси.
- Процессор регистри таркиби.
- Хотирани бошқариш ва процессордан фойдаланишни режалаштириш учун зарур маълумотлар(жараён приоритети, адрес макони, ўлчами ва жойлашган ўрни ва хоказолар.)
- Хисоб(қайд) маълумотлари жараён идентификация номери, қайси фойдаланувчи унинг ишини инициализация қилди, жараённинг процессордан фойдаланиш умумий вақти ва хоказолар
- Киритиш –чиқариш қурилмалари билан боғлиқ маълумотлар(масалан, жараёнга қандай қурилмалар боғланган, очик файллар жадвали ва хоказолар).

Албатта бу маълумотлар тизимини ва таркиби ҳар бир ОТга боғлиқдир. Кўпгина ОТларда жараённи характерловчи маълумот битта эмас, балки бир нечта маълумотлар структураида сақланади. Бу структуралар ҳар хил номланиши, юқорида келтирилган маълумотларни бир қисмини ёки қўшимча

маълумотларни ҳам ўз ичига олиши мумкин. Уни жараён дискриптори, PCB(Process Control Block) ёки жараённи бошқариш блоки деб номлаш мумкин.

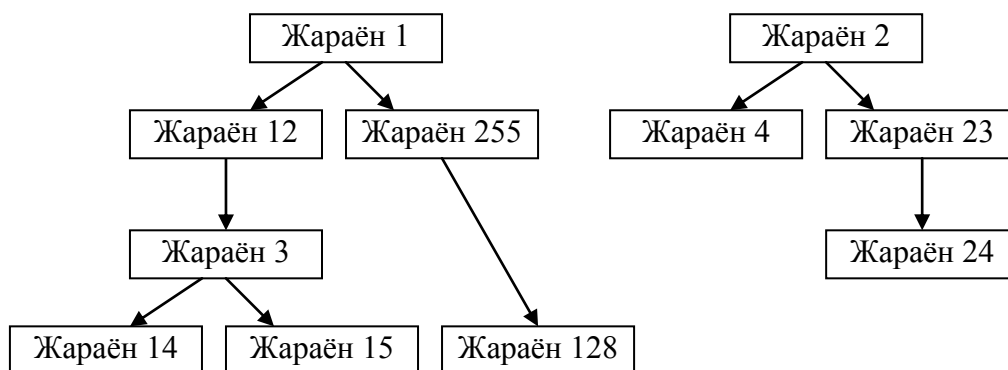
Бир марталик амаллар (операциялар)

Жараённинг компьютердаги мураккаб ҳаёт йўли уни туғилишидан бошланади. Жараёнлар концепциясини қўлловчи ихтиёрий ОТ, жараён яратиш воситасига эга бўлиши керак.

Энг оддий тизимларда(масалан, фақат битта аниқ илова иши учун лойихалаштирилган тизимларда) ҳамма жараёнлар тизим сатрида туғилиши мумкин. Мураккаброқ операцион тизимлар, жараёнларни зарурият бўйича динвмик холда яратадилар.

Операцион тизим стартидан сўнг, янги жараён туғилиши сабабчиси бўлиб махсус тизимли чақирик бажарган фойдаланувчи жараёни ёки операцион тизим бўлиши мумкин, яъни натижада яна жараён бўлиши мумкин.

Янги жараён туғилишига сабаб бўлган жараён ота жараён(parent process) дейилади, қайтадан янги яратилган жараён–фарзанд жараён дейилади(child process). Фарзанд жараён ўз навбатида яна янги фарзанд жараённи яратиши мумкин ва тизим ичида жараённинг генеологик дарахтини тўпламини ҳосил қиладилар, яъни генеологик ўрмон ҳосил бўлади.



Расм.7.3 соддалаштирилган жараёнларнинг ўрмони. Стрелка ота – фарзанд муносабатини кўрсатади.

Жараён туғилишида тизим янги PCB тузади бунда жараён туғилиши ҳолати билан уни тўлдирга бошлайди, янги жараён ўзининг ягона идентификация тартиб рақамини олади. Бирор бир жараён тугаши билан бўшаган идентификация номери бошқа жараён учун ишлатилиши мумкин.

Одатда ўзининг функцияларини бажариш учун жараён –фарзанд маълум ресурсларни талаб қилади: хотира, файллар, киритиш –чиқариш қурилмалари ва хоказолар. Уларни ажратишнинг икки хил ҳолати бор. Янги жараён ўзига ота жараён ресурсларини олиши мумкин. Бунда у ресурсларни ота –жараён ёки бошқа фарзанд –жараёнлар билан бўлишади ёки ресурсларин бевосита операцион тизимдан олиши мумкин. Ажратилган ресурслар ҳақидаги маълумот PCBга киритилади.

Кўп марталик операциялар.

Бир марталик операциялар жараёнлар ОТ бошқаруви остидаги сонини ўзгаришига олиб келади ва ҳар доим маълум ресурсларни ажратилиши ва бўшаши билан боғлиқдир. Кўп марталик операциялар операцион тизимдаги жараёнлар сонини ўзгаришига олиб келмайди ва ресурсларни ажратиш ва бўшаши билан боғлиқ бўлиши шарт эмас.

Жараёнлар устида кўп марталик операцияларни бажариш учун ОТ қандай ишларни бажаришини кўриб чиқамиз.

Жараённи ишга тушириш ОТ, тайёр турган жараёнлар орасидан бирини бажаришга танлайди. Танланган жараён учун уни бажарилиши учун оператив хотирада зарур маълумот билан таъминлайди. Кейин жараён ҳолати бажарилиш ҳолатига ўзгартирилади ва бошқарув жараён командалари хисоблагичи(счётчик)га кўрсатиладиган командасига узатилади. Хамма зарур маълумотлар жараён РСВдан олинади.

Жараённи тўхтатиш. Бажарилиш ҳолатидаги жараён иши, бирор бир узилиш натижасида тўхтади. Процессор автоматик тарзда командалар счётчигини сақлайди ва бошқарувни бу узилишга ишлов берувчи махсус адресга узатади. ОТ жараённи тайёрлик ҳолатига ўтказди ва узилишни катта ишлашга, яъни узилишга олиб келинган ҳолат учун маълум операцияларни бажаради.

Жараённи блокировка қилиш. Жараён ўз ишини хисоблаш тизимида бирор бир ходиса рўй бермагунча давом эттира олмайди. Шунинг учун жараён маълум тизимли чақириқ билан ОТга мурожаат қилади. ОТ тизимли чақириқни қайта ишлайди(киритиш –чиқариш операцияларини инициализация қилади, бирор қурилмани бўшабини ёки ходиса рўй беришини кутаётган жараёнлар навбатига қўшади ва хоказолар.), жараённи бажарилиш ҳолатидан кутиш ҳолатига ўтказди.

Жараённи блоклардан чиқариш(разблокирование). Тизимда бирор ходиса рўй бергандан сўнг, ОТ айнан қандай ходиса рўй берганлигини аниқлаши зарур. Кейин ОТ, қайси жараён шу ходисани кутиш ҳолатида эканлигини аниқлайди ва шундай жараён бўлса уни тайёрлик ҳолатига ўтказди.(бунда ОТ ходиса рўй бериши билан боғлиқ бўлган амалларни бажаради.)

Процессорни бир жараёндан иккинчисига тўғри(коррект) ўтказиш учун бажарилаётган жараён контекстини сақлаши ва процессор ўтказадиган жараён контекстини тиклаш зарур. Бундай жараёнлар ишланганлигини сақлаш/тиклаш процедураси контекстни ўтказиш дейилади.

Жараён тушунчаси, ОТ бошқаруви остидаги бажариладиган командалар тўплами, улар билан боғлиқ ресурслар ва унинг бажарилиши жорий моменти билан характерланади. Ихтиёрий вақтда жараён тўлиқ равишда ўзининг контексти, яъни регистирли, тизимли ва фойдаланувчи қисмларидан ташкил топган контексти билан тасвирланади. ОТларда жараён аниқ маълумотлар структураси –РСВ билан тасвирланади. РСВ –регистирли ва тизимли контекстларни акс эттиради. Жараёнлар бешта асосий ҳолатларда бўлиши мумкин: туғилиш, тайёрлик, бажарилиш, кутиш, бажарилишни тугаллаш.

Бир ҳолатдан иккинчисига жараён ОТ ёрдамида, улар устида бирор бир амал бажарилиши натижасида ўтказилади. ОТ жараёнлар устида қуйидаги операцияларни бажариши мумкин: жараён яратиш, жараён тугаллаш, жараённи тўхтатиб туриш, жараённи блокировка қилиш, жараённи блоклашдан чиқариш. Жараён приоритетини ўзгартириш.

Жараёнларни режалаштириш.

Хар гал, чегараланган ресурслар ва уларнинг бир нечта истеъмолчилари билан иш қўрилганда, масалан, мисол учун меҳнат жамоасида маош фондини тақсимлаш дейлик, биз мавжуд ресурсларни истеъмолчилар ўртасида тақсимлаш билан шуғулланишимизга тўғри келади, ёки бошқача айтганда ресурслардан фойдаланишни режалаштиришимизга тўғри келади. Бундай режалаштириш аниқ қўйилган мақсадларга (яъни, масалан, ресурсларни тақсимлаш ҳисобича биз нимага эга бўлмоқчимиз) ва бу мақсадларга мос ва истеъмолчи параметрларига таянадиган алгоритмларга эга бўлиши керак.

Режалаштириш даражалари(уровни).

Юқорида биз, ҳисоблаш тизимидаги икки хил режалаштириш: топшириқларни ва процессордан фойдаланишни режалаштириш ҳақида сўз юритган эдик.

Топшириқларни режалаштириш жараёнларни узоқ муддатга режалаштириш сифатида фойдаланилади. У, тизимда, унинг мультидастурлаш даражасини, яъни бир вақтнинг ўзида мавжуд бўлган жараёнлар сонини аниқлаб, жараёнларни юзага келишига жавоб беради. Агар тизимнинг мультидастурлаш даражаси доимий бўлиб турса, компьютердаги жараёнлар ўртача сони ўзгармайди, у ҳолда янги жараёнлар фақат олдин юкланганлари тугаллангандан кейин пайдо бўлади. Шунинг учун ҳам узоқ муддатга режалаштириш кам ишлатилади чунки, янги жараёнлар пайдо бўлиши орасида ўнлаб минутлар ўтиши мумкин.

Процессордан фойдаланишни режалаштириш, жараёнларни қисқа муддатга режалаштириш сифатида фойдаланилади. У, масалан, бажариладиган жараённинг киритиш –чиқариш қурилмалари ёки вақтнинг маълум интервали тугаллангандан сўнг амалга оширилади. Масалан у, ҳам қисқа муддатли режалаштириш 100 миллисекундда бир марттадан кам амалга оширилмайди.

Баъзи ҳисоблаш тизимларида, унумдорликни ошириш учун, қисман бажарилаяпган жараённи оператив хотирадан дискка вақтинча жўнатиш ва кейинроқ эса уни бажарилишини давом эттириш учун орқага қайтариш мумкин. Бундай процеду *paswapping*, яъни таржимада “ўтказиш(перекачка)”ни билдирса ҳам, таржимасиз “свонинг” термини ишлатилади. Жараёнлардан қайсисини ва қачон дискка ва орқага қайта ўтказишни, одатда, жараёнларни режалаштиришнинг қўшимча даражаси – ўртача муддатли режалаштириш ёрдамида амалга оширилади

Режалаштириш кўрсаткичи ва алгоритмларга талаблар.

Жараёнларни режалаштириш ҳар бир даражаси учун, ҳар турли жуда кўп алгоритмларни таклиф қилиш мумкин. Қайси алгоритмни танлаш, ҳисоблаш тизими ечадиган масалалар ва биз режалаштиришдан фойдаланиб эришмоқчи бўлган мақсадларимизга боғлиқдир. Бу мақсадлар қуйидагилардир.

- **Хаққонийлик** –компютер тизимида, ҳар бир жараён ва топшириқ учун процессордан фойдаланиш вақтининг маълум қисми ажратилишига кафолат бериш. Яъни, бир фойдаланувчи жараёни ҳар доим процессор вақтини банд қилиши ва бошқа фойдаланувчи жараёни бажарилмай туришига йўл қўймаслик.

- **Самарадорлик** –процессор иш вақтининг ҳамма 100%ни банд қилишга ҳаракат қилиш. Бунда у, бажаришга тайёр жараёнларни кутиб туриши керак эмас. Реал ҳисоблаш тизимларида процессор юкланиши 40та 90%гача ўзгариб туради.

- **Тўлиқ фойдаланиш вақтининг қисқариши**(turn aroid time) –жараёни старт ёки топшириқни юклашга навбат қўйиши ва уни тугаллаши орасидаги минимал вақтни таъминлаш.

- **Кутиш вақтини қисқартириш()** –жараёнларнинг тайёр ҳолати ва юклашга навбатни бериш вақтини қисқартириш.

- **Жавоб бериш вақтини қисқартириш** –жараённинг интерактив тизимларда фойдаланувчи сўровига жавоб бериш учун керак вақтини минималлаштириш.

Режалаштиришнинг қўйилган мақсадларига боғлиқ бўлмаган ҳолда, алгоритмлар қуйидаги хоссаларга эга бўлиши керак.

- **Аниқ бўлиши** керак, масалан, битта топшириқ ҳар доим бир хил вақда бажарилиши зарур.

- **Минимал харажатлар билан боғлиқ бўлиши** керак. Масалан, процессорнинг ҳар бир юз миллисекундига, жараён ўзининг бажарилишига қайси процессорни олиши мумкинлигини аниқлаш учун 200 миллисекунд керак бўлса, бундай алгоритмни қўллаш мақсадга мувофиқ эмас.

- **Ҳисоблаш тизими ресурсларини бир хил тақсимлаш** зарур, бунда кам фойдаланиладиган ресурсларни банд қиладиган жараёнларга имтиёз бериш керак.

- **Масштаблаштириш хоссасига эга бўлиш**, яъни юклама ошганда ишловчанлик қобилятини йўқотмаслик.

Юқорида келтирилган мақсад ва хоссалар бир –бирига қарама –қаршидир. Алгоритмни бир критерист(кўрсаткич) нуқтаси назаридан яхшиласак, иккинчиси нуқтаи –назаридан ҳолат ёмон томонга ўзгаради.

Режалаштириш параметрлари

Қўйилган мақсадларни амалга ошириш учун, яхши алгоритмлар, тизимдаги жараёнларнинг қандайдир характеристикаларига, юкламага

навбатдаги топшириқларга ҳисоблаш тизими ҳолатига, бошқача қилиб айтганда режалаштириш параметрларига таяниши зарур.

Ҳамма режалаштириш параметрларини иккита катта гуруҳларга бўлиши мумкин: статик параметрлар ва динамик параметрлар. Статик параметрлар ҳисоблаш тизими иш вақтида ўзгармайди, динамиклари эса тескариси, доимо ўзгаришда бўлади.

Тизимнинг статик параметрларига унинг ресурсларининг чегаравий қийматларини (оператив хотира ҳажми, свопинг учун дискдаги хотира максимал сони, уланган киритиш –чиқариш қурилмаларининг сони ва хоказолар). Тизимнинг динамик параметрлари айна вақтдаги бўш ресурслар сонини тавсифлайди.

Жараён статик параметрларига, қоида бўйича юклаш вақтига хос характеристикалар киради.

- Жараён қайси фойдаланувчи томонидан ишга туширилган ва қайси фойдаланувчи топшириқни шакллантирган.

- Қўйилган масала бажарилиш приоритети қандай, яъни масала қай даражада муҳим

- Фойдаланувчи томонидан масалани ечиш учун қанча процессор вақти сўралган.

- Процессор ва киритиш –чиқариш амалини бажариш вақти нисбати қандай

- Топшириқ учун, ҳисоблаш тизимининг қайси ресурслари(оператив хотира, киритиш –чиқариш қурилмалари, махсус кутубхоналар, тизимли дастурлар ва хоказолар) ва қанча миқдорда керак.

Узоқ муддатга режалаштириш алгоритмлари ўз ишларида ҳисоблаш тизимининг динамик ва статик параметрларидан ва жараёнларнинг параметрларидан(жараёнлар динамик параметрлари топшириқни юклаш этапида ҳали номаълум бўлади).

Ўртача муддатли ва қисқа муддатли режалаштириш алгоритмлари, кўшимча равишда жараёнларнинг динамик характеристикаларидан фойдаланадилар. Ўртача муддатли режалаштиришда бундай характеристика сифатида қуйидаги маълумотлардан фойдаланилади:

- Жараёни дискка ёки оператив хотирага юкланган моментдан қанча вақт ўтди;

- Жараён қанча оператив хотира эгаллайди;

- Жараёнга қанча процессор вақти ажратилди;

Режалаштириш жараёни ОТнинг “режалаштирувчи” деб аталадиган қисми орқали бажарилади. Режалаштирувчи, бажаришга, тайёр ҳолатдаги жараён ичидан янги жараёни қуйидаги тўртта ҳолларда танлаш ҳақида ечим қабул қилади:

1. Жараён бажарилиш ҳолатидан, бажарилиш тугалланди ҳолатига ўтишда

2. Жараён бажарилиш ҳолатидан кутиш ҳолатига ўтишда

3. Жараён бажарилиш холатидан, тайёрлик холатига ўтишида

4. жараён кутиш холатидан, тайёрлик холатига ўтишида.

Режалаштиришнинг турли –туман алгоритмлари мавжуддир, улар ҳар турли масалалар учун самарали ва турли мақсадларга эришишга мўлжаллангандир.

Масалан, 1.First –come, First –Served(FCFS) –биринчи келди, биринчи хизмат кўрсатилди.

2. Round Robin(RR) –болалар корусели. Бу FCFS ни модемфикация қилинган кўринишидир.

3. Shortest – Job – First (SJF) –биринчининг энг қисқа вақти ва хоказолар.ъ

Хисоблаш тизими N та фойдаланувчи интерактив режимда ишляпган бўлса, ҳар бир фойдаланувчи ўзида процессор вақтининг $\sim 1/N$ қисмига эга деб хисобланиши кафолатлайдиган режалаштириш алгоритмини қўллаш мумкин.

Хисоблаш тизимининг энг чегараланган ресурсларидан бири процессор вақтидир. Уларни кўп сонли жараёнлар орасида тақсимлаш учун тизимга жараёнларни режалаштириш процедурасини қўллашга тўғри келади. Режалаштиришнинг хисоблаш тизими холатига таъсирининг давомийлиги даражасига қараб, жараёнларни қисқа муддатли, ўртача муддатли ва узок муддатли режалаштиришларга бўлинади. Режалаштириш аниқ алгоритмлари қўйилган мақсадлардан, ечилиаяпган масала синфларига боғлиқ бўлиб, жараёнларнинг статик ва динамик параметрларига ва компьютер тизимларига таянади. Режалаштиришнинг сиқиб чиқарадиган ва сиқиб чиқармайдиган режимлари ажратилади.

Сиқиб чиқарилмайдиган режалаштириш режимида, бажариладиган жараён бошқа жараёнга процессорни фақат хохиши билан бериши мумкин, сиқиб чиқарадиган режимда эса, бажарилаяпган ўзига боғлиқ бўлмаган холда чиқарилади.

Энг оддий сиқиб чиқармайдиган режалаштириш алгоритми –FCFSдир, у қисқа жараёнларни сезиларли даражада ушлаб қолиши мумкин(тайёрлик холатига вақтида ўтмаган жараёнларни).

Вақтни ажратиш тизимларида кенг тарқалган алгоритм бу сиқиб чиқарадиган алгоритм –RRдир.

Сиқиб чиқарадиган алгоритмлар ичида жараёнларнинг ўртача кутиш вақти жихатидан оптимал алгоритм -SJF алгоритмидир.

6-маъруза. Жараён ҳолатлари ва ҳолат диаграммаси. Узилишлар. Узилишлар механизми ва роли. Жараёнларни бошқариш: диспетчерлаш ва контекстлар. Жараён дискриптори.

7-маъруза. Хотира ва унинг акси, виртуал адрес макони. Хотирани бошқаришнинг умумий принциплари. Хотирани статик ва динамик булимлар буйича таксимлаш.

Компьютер тизими бош асосий масаласи-дастурни бошқаришдир. Дастурлар ва уларнинг мурожаат қиладиган маълумотлари, бажарилиш жараёнида оператив хотирада (хеч бўлмаса қисман) жойлашган бўлиши шарт. Операцион тизимга, хотирани, фойдаланувчи жараёнлари ва ОТ компоненталари орасида таксимлашга тўғри келади. Операцион тизимнинг бу фаолияти хотирани бошқариш дейилади. Шундай қилиб, хотира (storage memory) синчиклаб бошқаришни талаб этадиган ресурсдир. Яқин кунларгача хотира энг қиммат ресурс хисобланган.

Операцион тизимнинг, хотирани бошқарадиган қисми, хотира менежери дейилади.

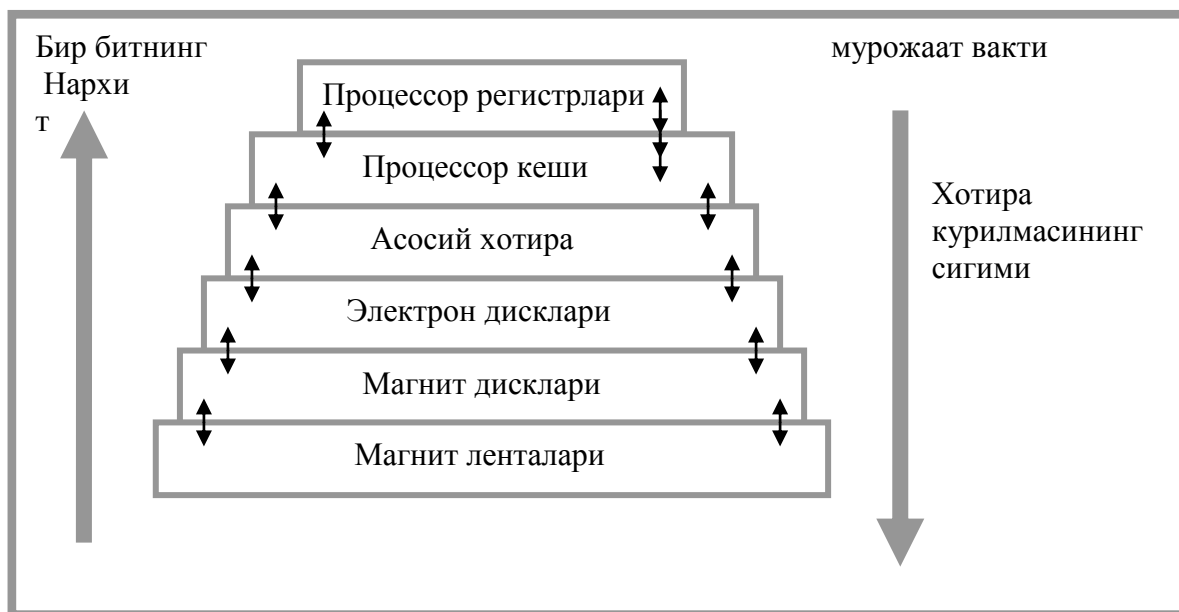
Компьютер хотирасини физик тузилиши (ташkil этилиши)

Компьютернинг хотира қурилмаси икки хил турга: асосий (бош хотира, тезкор хотира, физик хотира) ва иккиламчи (ички хотира) хотирага бўлинади.

Асосий хотира бир байтли тартибланган ячейка массивига эга бўлиб, ҳар бир ячейка ўзининг адресига (номери) эга. Процессор буйруқларни асосий хотирадан олади, қайта ишлайди ва бажаради. Буйруқларни бажаришда асосий хотиранинг бир нечта ячейкаларига мурожаат қилишга тўғри келади. Одатда асосий хотира яримўтказгичли технология асосида тайёрланади шунинг учун хотирадаги маълумотлар электр манбасидан узилгандан сўнг ўчиб кетади.

Иккиламчи хотира (бу асосан дисклардир) бу чизиқли бирлик адресга эга бўлган жой ва уларни кетма-кет жойлашган байтлар ташkil қиладди. Иккиламчи хотиранинг тезкор хотирадан фарқи шундаки, у алоҳида энергияга, катта ҳажмга, ва самарали фойдаланиш имкониятига эга.

8.1 расмдаги кўрсатилган схемага яна бир нечта оралиқ сатхларни кўшиш мумкин. Ҳар хил кўринишдаги хотиралар иерархияга, мурожаат вақти камайиб бориши, нархини ошириши ва сигими ошириши тарзида бирлашиши мумкин.



8.1 расм. Хотира иерархияси.

Кўпбосқичли схемалар куйидагича ишлатилади. Маълумотлар одатда хотиранинг юқори сатхларидан кидирилади, агар у ердан топилмаса, маълумотлар катта номерли сатхларда ҳам сакланади. Шунинг учун, у кейинги сатхдан кидира бошлайди. Агар керакли маълумотни топса, уни юқорироқ сатхга ўтказди.

Маҳаллийлик (локаллик)

Маълум бўлишича, бу усулда бошқаришни ташкил этиш хотира сатхларига киришни ва алоқа частотасини камайтиради.

Бу ерда муҳим ролни, чегараланган вақт давомида, хотира адресларининг кичик булагини билан ишлаш хоссаси уйнайди. Бу эмпирик жихатдан кузатиладиган хосса локаллик принципи еки мурожаатларни локаллаштириш дейилади.

Процессор КЭШини, қурилмаларнинг бир қисми ҳисобланади, шунинг учун ОТнинг хотира менежери, асосан маълумотларни компютернинг асосий ва ички хотира қисмига тақсимлаш билан шуғулланади. Баъзи схемаларда тезкор ва ички хотира ўртасидаги оқимни дастурчи бошқаради. Аммо бу боғланиш дастурчи вақтини йўқотади, шу сабабли бу ишни ОТга юклашга ҳаракат қилинади.

Физик хотирада маълумотларни реал жойлашишини курсатувчи асосий хотирадаги адреслар- физик адреслар деб аталади. Дастур ишлайдиган физик адреслар тўплами, физик адреслар майдони деб аталади.

Мантикий (логик) хотира.

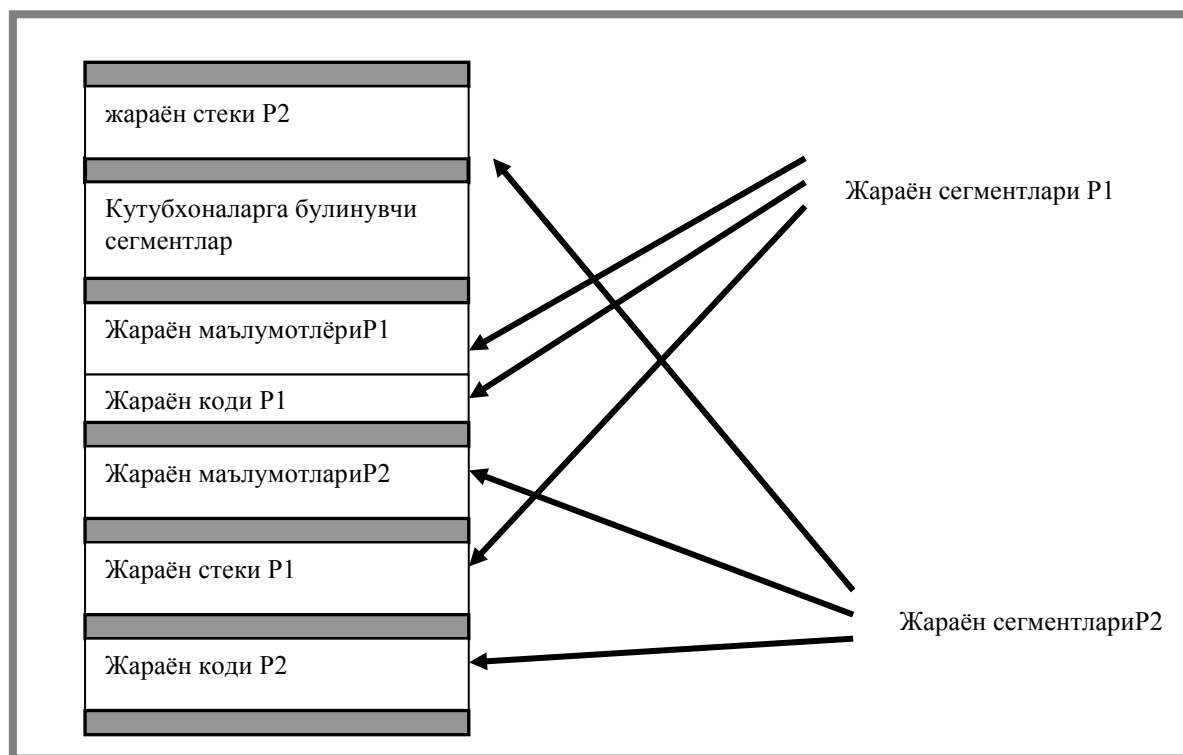
Хотирани, ячейкалар чизикли туплами қурилишида аппарат ташкил этиш, дастурчининг дастур ва маълумотлар сакланиши қурилиши ҳақидаги тасавури билан мос келмайди. Қўпгина дастурлар бир-бирига боғлиқ бўлмаган ҳолда яратилган модуллардан ташкил топган.

Баъзан жараён таркибига кирувчи ҳамма модуллар хотирада кетма-кет жойлашади ва чизикли адреслар майдонини ташкил қилади. Бироқ кўпинча модуллар хотиранинг турли жойларида жойлаштирилади ва турлича фойдаланилади.

Хотирани бошариш схемасида, фойдаланувчининг бундай тасавурига мос келадиган маълумот ва дастурларни сақлаш, сегментация дейилади. Сегмент-хотиранинг аниқ кўрсатилаган қисми бўлиб, унинг ички қисмида чизикли адресларни қўллаб қувватлайди. Сегмент процедура, массив, стек ёки скаляр микдорлардан ташкил топган булади, лекин одатда аралаш типдаги маълумотлардан иборат бўлмайди.

Бошида сегментлар, дастур коди фрагментларини (матн редактори, тригонометрик кутубхона в ах.к.) жараёнлар билан умумлаштириш заруриятидан келиб чиққан бўлиши керак, чунки уларсиз ҳар бир жараён узининг адрес маконида маълумотларнинг яна бир нусхасини сақлашига тугри келар эди. Хотиранинг, тизим бир нечта жараённинг маълумотларини акс эттирадиган алоҳида қисмлари бўлиб улар сегментлар деб ном олди.

Хотира шундай қилиб, чизикли қурилишдан икки улчамли қурилишга келди. Адрес икки компонентдан иборат бўлиб, улар: сегмент номер ва сегмент ичидаги жойлашган урнидир. Кейинчалик, жараённинг турли компоненталарини (дастур коди, маълумотлар, стек в ах.к.) турли сегментларда жойлаштириш қулай бўлиб қолди. Яна шу нарса аниқ бўлиб қолдики, аниқ сегмент ишини, унга сегментда сақланадиган маълумотлар устида бажарилиши рухсат берилган операциялар, масалан, мурожаат ҳукуки ва операциялар типини каби атрибутлар қийматини бериб, назорат қилиш мумкин бўлиб қолди.



8.2расм. Жараён сегментларининг компьютер хотирасида жойлашиши.

Баъзи ,жараенни адрес маконини тасвирлайдиган сегментлар 8.2 – расмда курсатилган.

Аксарият замонавий операцион тизимлар хотирани сегмент бошқарув хусусиятига эга. Отларнинг баъзи архитектураларида (масалан Интел) сегментлаш қурулмалар томонидан кулланади.

Жараён мурожаат киладиган адреслар,оператив хотирада мавжуд бўлган реал адреслардан шу тахлитда фарк килади. Ҳар бир аниқ ҳолатда дастур фойдаланадиган адрес, ҳар хил усуллар ердамида тасвирланиши мумкин. Масалан, адрес, берилган матнда одатда символли бўлади. Компилятор бу символли адрес ва ўзгарадиган адресларни боғлайди (масалан, n байт модул бошидан). Дастур генерациялаган бундай адрес одатда мантикий адрес(виртуал хотирали тизимларда у кўпинча виртуал хотира) деб номланади. Барча мантикий адреслар туплами мантикий(виртуал) адреслар майдони деб аталади.

Адреслар боғланиши

Демак, мантикий ва физик адреслар майдонлари, ташкил этилиши ва ўлчами бўйича бир бирига мос эмас. Мантикий адреслар майдонинг максимал ўлчами одатда процессорнинг разряди билан аниқланади (масалан 2^{32}), ва замонавий тизимларда физик адреслар майдонининг ҳажмидан кўзга кўринарли даражада юкори булади. Шундай экан, процессор ва операцион тизим асосий хотирада жойлашган дастурни дастур кодига, реал физик адресга таянган ҳолда юлни акс эттириш керак. Бундай кўринишда адресларни тасвирлаш адресларни трансляцияси ёки адресларни боғлаш деб номланади.

Мантикий адреснинг физик адрес билан боғланиши дастур операторининг бажарилишигача ёки бажарилиш вақтида амалга оширилиши шарт. Бундай ҳолда, курсатмаларни ва маълумотларни хотирага боғлаш куйидаги қадамлар буйича амалга оширилади.

- Компиляция босқичи..
- Юклаш босқичи
- Бажарилиш босқичи.

Хотиранинг бошқарув тизими функцияси.

Хотирадан самарали фойдаланишни таъминлаш учун операцион тизим куйидаги функцияларни бажариши лозим:

Физик хотирани аниқ бир соҳасида жараен адреслар тўпламини акс эттириш;

Қарама-қарши жараёнлар ўртасида хотирани тақсимлаш;

Жараенлар адреслар майдонида рухсатни бошқариш;

Оператив хотирада жой қолмаганда, ташқи хотирага жараёнларни (қисман ёки тулик) юклаш;

Бўш ва банд хотирани ҳисобга олиш.

Хотира бошқарувининг энг оддий схемалари.

Дастлабки операцион тизимларда хотирани бошқаришнинг энг оддий методлари қўлланилган. Бошида фойдаланувчининг ҳар бир жараёни асосий хотирага кўчирилиши керак бўлган, хотира узлуксиз майдонини банд қилган, тизим эса қушимча фойдаланувчи жараенларга бир вақтнинг узида асосий хотирада жойлашиб турганича хизмат курсатади. Кейин “оддий свопинг” (аввалги тизимлардагидек, ҳар бир жараенни асосий хотирага жойлаштиради, аммо баъзи курсаткичларга асосан баъзи жараенлар обзорини асосий хотирадан ташқи хотирага тулик чиқаради ва уни бошқа жараен образи билан алмаштиради) пайдо бўлди. Бу турдаги схема нафақат тарихий қийматга эга. Ҳозирги вақтда улар укув-машқ ва илмий – тажриба модели операцион тизимларида, шунингдек операцион тизим «тикилган» ҳолда ўрнатилган компьютерларда(embedded) қўлланилади.

Қатъий белгиланган(фиксирланган) булимли схемалар

Тезкор хотирани бошқаришнинг энг оддий йўли уни олдиндан(генерация босқичида ёки тизим юкланиши вақтида) бир қанча қатъий белгиланган улчамдаги булимларга булишдан иборатдир.

Келиб тушаётган жараёнлар у ёки бу бўлимга жойлаштирилади. Шу сабабли физик адреслар майдонининг шартли бўлиниши юзага келади.

Жараённинг мантикий ва физик адреслари боғланиши уни аниқ бир бўлимга юклаш вақтида ёки баъзан компиляция вақтида юзага келади.

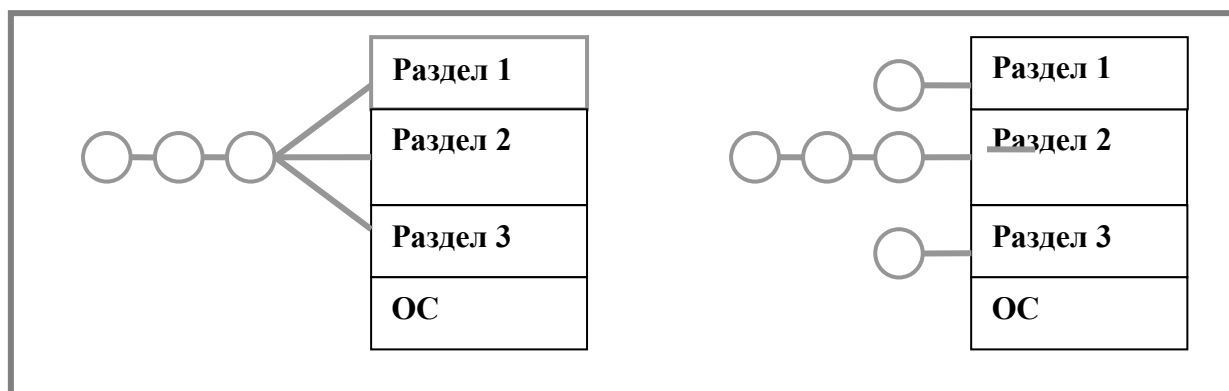
Ҳар бир бўлим ўзининг жараёнлар навбатиغا эга, ёки ҳамма бўлимлар учун жараёнлар глобал навбати мавжуд бўлиши мумкин.

Бу схема ИБМ ОС/360 (МФТ), ДЕС РСХ-11 ва шунга яқин бошқа системаларда қўлланилган.

Ҳотирани бошқариш тизими жараёни хажмини баҳолайди, унга мос келувчи бўлимни танлайди, жараёни бу бўлимга юклайди ва адресларни созлайди.

8.4 расмда фиксирланган бўлимли схемалар кўрсатилган: (а) ғнавбати умумий бўлган жараёнлар, (б) - алоҳида навбатли жараёнлар

Бу схеманинг камчилиги кўриниб турибдики, бир вақтда бажариладиган жараёнлар сони бўлимлар сони билан чекланган. Бошқа муҳим камчилигишундан иборатки, тақлиф қилинаётган схема, ички фрагментлашдан, яъни жараёнга ажратилган, аммо ишлатилмаган хотира қисмини йўқотиш билан қаттиқ зарарланади. Фрагментация, жараён ўзига ажратилган бўлимни тулиқ банд қилмаслиги ёки баъзи бўлимлар, бажариладиган фойдаланувчи дастурлари учун кичик бўлганлигидан келиб чиқади.



8.4-расм

Оверлейли(қопланган)тузилиш.

Жараён мантиқий адреслар майдони ҳажми ,унга ажратилган булим ҳажмидан катта (ёки энг катта ҳажмдан ҳам катта) бўлган ҳолатларда, баъзан оверлей номли ёки қопланадиган тузилишли ташкил этадиган техникадан фойдаланилади.

Асосий ғоя – фақат айна вақтда керак бўлган дастур кўрсатмаларини хотирада сақлаб туришдир.

Оверлай тузилиш дастурининг кодининг дискда аниқ хотира кўринишида бўлади ва оверлай керакли вақтда уни драйвер орқали ўқиб ишлатади. Оверлей структуранинг тавсифини ёзиш учун одатда махсус содда (overlay description language) тилдан фойдаланилади. Дастурда

ишлатиладиган ҳамма файллар дастурнинг ички чақириқларига дарахт кўринишдаги файл ёрдамида тўлдирилади.

Шуни назарда тутиш керакки, оверлейли структурани ташкил этиш кўп жихатдан локаллик хоссасига боғлиқдир, бу эса ўз навбатида хотирада айти вақтда фақат керак маълумотларни сақлаш имконини беради.

Динамик тақсимланиш. Алмаштириш(свопинг).

Пакетли тизимлар билан ишлашда фиксирланган бўлимлар билан ишлаб, бошқа ҳеч қандай мураккаб нарсалардан фойдаланмаслик ҳам мумкин. Вақтни тақсимлаш тизимлари билан ишлаш вақтида, хотира ҳамма фойдаланувчилар жараёнларини ўзида ушлаб тура олмайдиган ҳолат рўй бериши мумкин. Бу ҳолда свопингдан фойдаланишга тўғри келади. Свопинг-бу жараёнларни асосий хотирадан диска ва орқага тўлиқ ўтказишдир. Жараёнларни дискка қисман юклаш саҳифали ташкил этилган тизимда амалга оширилади.

Юкланган жараёнлар ҳудди ўша адреслар майдонига ёки бошқа жойга қайтарилиши мумкин. Бу чеклаш боғланиш методи хусусиятидан келиб чиқади. Боғланиш схемаси учун, бажариш босқичида жараёнларни хотиранинг бошқа жойига кўчириш мумкин.

Свопинг хотирани бошқаришга бевосита алоқаси йўқ, у кўпроқ жараёнларни режалаштириш билан боғлиқдир.

8-маъруза. Белгиланган, узгартириладиган чегаралар буйича тақсимлаш. Сегментли, саҳифали, сегмент-саҳифали ташкил этиш. Виртуал хотирани ташкил этиш усуллари.

Ўзгарувчан бўлимли схемалар.

Қоида бўйича свопинг тизими фиксирланган бўлимларга асосланаши мумкин. Аммо динамик тақсимлаш ёки ўзгарувчи бўлимли схемалар самарали ҳисобланади. Чунки улар ҳамма жараёнлар тўлиқ равишда хотирада жойлашганда, яъни свопинг бўлмаган ҳолларда қўлланилади.

Бу ҳолда, бошида хотира бутунлай бўш ва олдиндан бўлимларга бўлинган эмас. Янгидан келаяпган масалага қатъий равишда керакли хотиранинг ўзи ажратилади. (ундан кўп эмас)

Жараён чиқарилган дан сўнг, хотира вақтинча бўшатилади. Бир қанча вақт ўтгандан сўнг хотира турли ўлчамдаги ўзгарувчили сонли бўлимлардан иборат бўлиб қолади. Ёнма-ён бўлган бўш жойлар бирлаштирилиши мумкин.

Саҳифали хотира

Юқорида тавсифланган схемаларда хотирадан самарали фойдаланилмайди, шунинг

Учун ҳам хотирани тақсимлашнинг замонавий схемаларида жараёни оператив хотирада узлуксиз блок сифатида жойлаштириш кўзда тутилмаган.

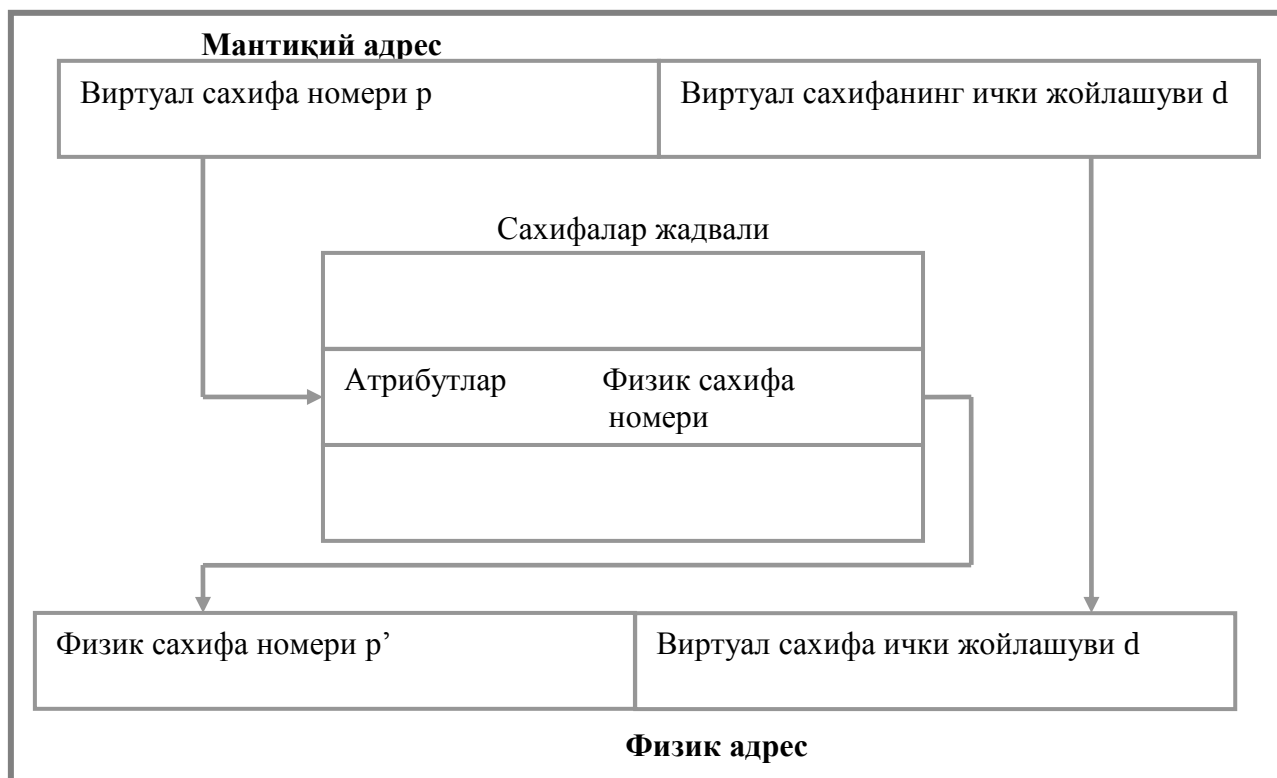
Хотирани саҳифали ташкил этишда энг оддий ва энг кенг тарқалган усул (ёки paging), хотиранинг ҳам мантикий адресли майдони, ҳам физик майдонини бир хил ўлчамдаги саҳифа ва блоklar тўплами кўринишида ташкил этишдир. Бунда мантикий саҳифалар (page) юзага келади ва уларга мос физик хотира бирликлари-физик саҳифалар ёки саҳифа кадрлари деб аталади (page frames). Саҳифалар (ва саҳифа кадрлари) одатда 2 сонинг даражасидан иборат бўлган фиксирланган қатъий узунликка эгадир ва улар бир – бири билан кесишмайди. Хар бир кадр маълумотларнинг бир саҳифасини ўз ичига олади. Хотирани бундай ташкил этишда ташқи фрагментация ўрин бўлмайди ва ички фрагментациядан келиб чиқадиган йўқотиш фақат охириги саҳифалардан келиб чиқадиган йўқотиш билан чегараланади..

Саҳифали тизимда мантикий адрес – тартибланган жуфтлик (p,d) дан иборат, бу ерда p виртуал хотира саҳифаси тартиб рақами, d-эса бу саҳифа доирасидаги элемент ўрнини билдиради. Адрес маконини саҳифаларга бўлиш ҳисоблаш тизими томонидан, дастурчи аралашмаган ҳолда амалга оширилади. Шунинг учун ҳам, адрес, операцион тизим нуқтаи наридангина икки ўлчамлидир, дастурчи нуқтаи-назаридан эса жараён адрес макони чизиқли ҳисобланади.

Юқорида келтирилган схема, жараёнларни тўлиқ жойлаштириш учун кадрларнинг узлуксиз соҳаси етарли бўлмаган ҳолларда ҳам, жараёни юклаш имконини беради. Аммо, бу схемада адресни трансляциялаш учун битта асос регистри етарли эмас. Мантикий адресларни физик адресларда акс

эйтириш,мантикий саҳифаларни физик саҳифаларда акс эйтиришга келтирилади ва оператив ҳётирада сақланадиган саҳифалар жадвалидан иборат бўлади.Баъзида ,саҳифалар жадвали –жадвал кўринишидаги чизиқли –бўлакли функция ҳам дейилади.

Мантикий манзилнинг интерпретацияси 8.7 расмда кўрсатилган. Бунда бажариладиган жараён $v=(p,d)$ мантикий адресга мурожаат қилади,ва акс ттириш механизми саҳифа тартиб рақами p ни саҳифалар жадвалидан қидиради , бу саҳифа p^* саҳифа кадрида жойлашганлигини аниқлайди ва реал адрес p^* ни d га айлантиради.



8.7 расм. Хотирани саҳифали ташкил этишда мантикий ва физик адресларнинг боғланиши.

Саҳифалар жадвали (page table) процессорнинг махсус регистиди манзиллаштирилади ва кадрлар номерини мантикий адрес бўйича аниқлашга ёрдам беради. Бу асосий масаладан ташқари саҳифалар жадвали қаторида ёзилган атрибутлар ёрдамидаи аниқ саҳифага мурожаат назорати ва уни химоясини ташкил этиш мумкин.

Сегментли ва сегмент – саҳифали ҳотира.

Ҳотираларни бошқаришни Яна иккита схемаси мавжуд: сегментли ва сегмент – саҳифали. Сегментлар саҳифалардан фарқли равишда ўзгарувчи ўлчамга эга бўлишади. Хотирани сегментли ташкил этишда виртуал адрес ,дастурчи учун ҳам ,операцион тизим учун ҳам икки ўлчамли бўлади ва икки майдондан: сегмент тартиб рақами ва сегмент ичидаги жой рақамидан иборат бўлади.

. Айтиб ўтиш лозимки, тасвирлаш қулай бўлиши учун ОТ ёрдамида чизиқли адрес икки ўлчамлиликка келтирилган саҳифали ташкил этишдан фарқли равишда, бу ерда адрес икки ўлчамлилиги фойдаланувчини жаранни байтларнинг чизиқли массиви кўринишида эмас, балки ўзгарувчи узунликдаги сегментлар тўплами кўришида тасавур қилиш натижасидир. (маълумот, код, стек,...).

Қуйи босқичдаги дастурлаш тилларида дастур тузувчи дастурчилар сегментли структура ҳақида билишлари лозим. Бунда сегмент регистраларини ўзгартиришга олиб келади. Мантиқий манзиллар майдони – сегментлар йиғиндиси. Ҳар бир сегмент исмга, ўлчамга ва бошқа параметрларга эга. Дастурчи битта манзил берадиган саҳифаларнинг номерига бўлинадиган саҳифали схемалардан фарқли равишда сегментли структурада манзил иккита ўлчамдан иборат бўлади.: сегмент номи ва жой ўзгариши.

Ҳар бир сегмент – 0 дан бошланадиган манзилларнинг тўғри чизиқли кетма-кетлигидир. Сегментнинг максимал размери процессорнинг разряди билан аниқланди. (бунда 32 разрядда манзиллашда 2^{32} байт ёки 4 Гб). Сегмент манзили динамик равишда ўзгариши мумкин (масалан стек сегменти). Сегментлар жадвалининг элементида сегмент бошланиши физик манзилидан ташқари одатда сегмент узунлиги ҳам берилади.

**9-маъруза. Операцион тизимларда кириш ва чиқишни бошқариш.
Операцион тизимларда кириш ва чиқишни ташкил этиш асосий
концепциялари, кириш ва чиқишни бошқариш режимлари.**

10-маъруза. Файл тизимлари. Файл тизими функциялари ва маълумотлар иерархияси. FAT файл тизимлари. Файлларни жойлаштириш жадвали. Бошқа файл тизимлари. Улар имкониятлари. Файл ва каталогларга мурожаат ва чегаралари.

Файлларни бошқариш тизимлари, кўпгина замонавий ОТ ларнинг асосини ташкил этади. М-н, UNIX ОТ и, файл тизимсиз ишламайди, унда файл тизими асосий тушунчалардан биридир. Ҳамма замонавий ОТ лар файллардан ва улар билан ишлаш учун мос дастур таъминтидан фойдаланади. Гап шундаки, биринчидан, файл тизими орқали маълумотлар бўйича кўпгина ишлов берувчи дастурлар боғланади. Иккинчидан, бу тизим орқали диск маконини марказлаштирилган ҳолда тақсимлаш ва маълумотларни бошқариш муаммолари ечилади. Ва ниҳоят, фойдаланувчилар ўз маълумотларига мурожаат қилишнинг осон усулларига эга бўладилар, бу маълумотларни, таўқи хотира қурилмаларида жойлаштирадилар.

Турли ОТ лар ва ташқи хотиранинг турли қурилмалари учун яратилган кўпгина файл тизимлари мавжуддир. Уларда, мос равишда маълумотларни жойлама (носитель) да жойлаштиришнинг ҳар хил принципларидан фойдаланилади. Биз, FAT, FAT 32 ва NTFS файл тизимлари билан танишамиз. Айниқса, ҳозирги кунда энг кўп тарқалган файл тизими билан – NTFS билан танишиш муҳим аҳамиятга эгадир.

Файл тизими функциялари ва маълумотлар иерархияси.

Файл деганда, одатда номланган, бир хил тузилишга эга бўлган ёзувлардан ташкил топган маълумотлар тўплами тушунилади. Бу маълумотларни бошқариш учун, мос равишда файл тизимлари яратилади. Файл тизими, маълумотлар мантикий структурасинининг ва уларга ишлов бери жараёнида бажариладиган амаллар билан иш олиб бориш имконини беради. Айнан файл тизими, маълумотларни дискларда ёки бирор-бир бошқа жамламада ташкил этиш усулини аниқлайди. Файл тизимининг қабул қилинган спецификациялари бўйича, файллар билан ишлашни амалга оширувчи махсус тизимли дастурий таъминотни, кўпинча файлларни бошқарув тизими дейилади. Айнан, файлларни бошқариш тизими, файл маълумотларини яратиш, йўқотиш ташкил этиш, ўқиш, ёзиш, модификация қилиш ва жойини ўзгартириш вазифалар билан бирга файлларга мурожаатни ва файллар томонидан фойдаланиладиган ресурсларни бошқаришга жавоб беради.

Файлларни бошқариш тизимининг “ФБТ” асосий вазифаси, бизга керакли бўлган ёзувнинг аниқ физик адресини кўрсатиб, қуйи даражада мурожаат ўрнига, файл кўринишида ташкил этилган маълумотларга мурожаатнинг қулай усулини беришидир, яъни файл номи ва ундаги ёзувни номини кўрсатиб, мантиқан мурожаат қилишга имкон беришидир.

Файлларни бошқариш тизими ёрдамида, фойдаланувчиларга қуйидаги имкониятлар яратилади:

- фойдаланувчиларнинг мулоқат функцияларини (унинг маълумотлари билан) амалга оширувчи ва файлларни бошқариш тизимидан фаол фойдаланувчи, махсус бошқарувчи функциялар ёрдамида ёки ўз дастурларидан, номли маълумотлар тўпламини (файлларни) яратиш, олиб ташлаш ва қайта номлаш (ва бошқа операциялар);
- дисксиз периферик қурилмалар билан файл каби ишлаш;
- файллар орасида, қурилмалар ўртасида (ва тескари) маълумотлар алмашилиш;
- файлларни бошқариш тизими дастурий модулларига мурожаат усули билан файллар билан ишлаш (API нинг бир қисми файллар билан ишлашга мўлжалланган);
- файлларни ҳуқуқсиз мурожаатдан ҳимоя қилиш.

Қоида бўйича ҳамма замонавий ОТ лар ўз файлларини бошқариш тизимларига эгадир. Баъзи ОТ лар эса, бир нечта файл тизимлари билан ишлаш имконига эгадирлар (бир нечтаси ичидан биттаси ёки бир нечтаси билан бир вақтда). Бу ҳолларда, монтировка қилинадиган файл тизимлари тўғрисида сўз боради (монтировка қилинадиган файлларни бошқариш тизимларини қўшимча сифатида ўрнатиш мумкин) ва бу борада улар мусақилдир.

Шу нарса маълумки, файлларни бошқарув тизими, ОТ нинг асосий компонентаси бўлган ҳолда, ундан мустақил эмас, чунки API нинг мос чақириқларидан фаол фойдаланади. Шу билан бирга, ФБТ нинг ўзи API ни янги чақириқлар билан бойитади.

Демак, шуни таъкидлаш мумкинки, ихтиёрий ФБТ ўз-ўзига мавжуд эмас, балки улар аниқ ОТ муҳитида ишлаш учун яратилади. Мисол учун ҳаммага топиш файл тизими FAT ни келтириш мумкин (File Allocation Table – файлларни жойлаштириш жадвали). Бу тизим, файлларни бошқариш тизими сифатида кўп ҳолларда ишлатилади.

Бу файл тизимининг турли ОТ ларда ишлайдиган, турли версиялари мавжуддир: FAT, FAT 12, FAT 16, super FAT ва х.к.лар.

Бу ФБТ нинг MS DOS ОТ дан то Windows NT билан ишлайдиган версиялари мавжуд. Бошқача қилиб айтганда, маълум файллар тизимига мос ташкил этилган файллар билан ишлашда, ҳар бир ОТ учун ўз файлларни бошқариш тизими ишлаб чиқиши шарт. Ва бу файл тизими, қайси ОТ учун мўлжалланган бўлса, ўша ОТ да ишлайди, аммо бошқа файлларни бошқарув тизими ёрдамида яратилган файлларга мурожаат имконини таъминлайди (бошқа ОТ ФБТ бу ОТ ФБТ принциплари асосида ишлайдиган бўлиши керак).

Ш.к., файл тизими – бу маълумотларнинг қабул қилнган сертификациялари бўйича ташкил этилган тўпламлари мажмуаси бўлиб, улар файлларга мурожаат учун зарур бўлган адресли маълумотларни олиш усулини аниқлайди.

Демак, файл тизими термини, авваламбор файлларга ташкил этилган маълумотларга мурожаат принципларини аниқлайди. Худи шу терминни у ёки бу жамламада жойлашган аниқ файлларга нисбатан ҳам ишлаштиш мумкин. Файлларни бошқариш термини эса, файл тизимининг аниқ реализациясига (амалга ошишига) нисбатан ишлатиш мумкин, яъни бу тизим – аниқ ОТ ларда файллар билан ишлашни таъминлайдиган дастур модуллари тўпламидир.

Инсон ишлайдиган маълумотлар, одатда структуралаштирилгандир. Бу аввало, маълумотларни самарали сақлашни ташкил этишга имкон беради, кидирувни осонлаштиради, номлашда қўшимча имкониятлар яратади. Худди шунингдек, файллар билан ишлашда ҳам иложи борица структуралаштириш механизмини киритиш керак. Ҳаммадан осони иерархик муносабатларни ташкил этишдир. Бунинг учун, каталог тушунчасини, киритиш directory етарлидир. Каталог, файл кўринишида ташкил этилган маълумотлар тўғрисидаги маълумотни ўз ичига олади.

Бошқача айтганда, каталогда файллар дискриптори сақланади. Агар файллар, блок қурилмасида ташкил этилган бўлса, айнан каталог ёрдамида ФБТ, шу маълумотлар жойлашган блок адресларини топади. Каталог, махсус тизимли маълумот структурасига эмас, (асосий, илдиз каталог дейилади), файл ўзи ҳам бўлиши мумкин. Бундай файл-каталог, махсус тизимли қийматга эга бўлиб, уни бошқа оддий файллар орасида ажратиш зарурдир. Файл Каталог кўпинча “подкаталог” (subdirectory) деб аталади. Агар файл каталог, бошқа файллар ҳақида маълумотни ўзида сақласа, улар орасида ҳам ўз навбатида файл-каталоглар бўлса, биз ҳеч қандай чегараланмаган иерархияга эга бўламиз.

Ундан ташқари, бундай файл объектларни файл каталог киритиш, файл тизимини нафақат структуралаш, илдиз каталогдаги элементлар сонини чегараланмаган муаммосини ҳал қилади. Файл каталогида элементлар сонига чегара йўқ, шунинг учун катта ўлчамдаги каталогларни яратиш мумкин.

FAT файл тизими.

FAT файл тизими, ўз номига қуйидаги маълумотларни ўз ичига олган оддий жадвал орқали эга бўлган:

- файл ёки унинг фрагментлари учун ажратилган, мантиқий дискнинг бевосита адресланувчи қисмлари (участкалари);
- диск макони бўш соҳалари;
- дискнинг дефектли соҳалари (бу соҳалар дефект жойларига эга бўлиб, маълумотларни ўқиш ва ёзишни хатосиз бажаришга кафолат бермайди).

FAT файл тизимида, ихтиёрий мантиқий диск икки соҳага: тизимли соҳа ва маълумотлар соҳасига бўлинади.

Мантикий диск тизимли соҳаси, форматлаш вақтида инициаллаштирилади, кейинчалик эса, файл структураси билан ишлаганда янгиланади. Мантикий диск, маълумотлар соҳаси оддий файл ва файл каталогларни ўз ичига олади; бу объектлар иерархияни ташкил этади, бу объектлар илдиз каталогга бўйсунди.

Каталог элементи файл объектини тавсифлайди, у оддий файл ёки файл- каталог бўлиши мумкин. Маълумотлар соҳасига, тизимли соҳадан фарқли равишда, ОТ нинг фойдаланувчи интерфейси орқали мурожаат қилинади. Тизимли соҳа қуйидаги ташкил этувчилардан иборатдир (мантикий адрес соҳасида кетма-кет жойлашган):

- юкланиш ёзуви (Boot Record, BR);
- резервланган секторлар (Reserved. Sectors, Res.Sec);
- файлларни жойлаштириш жадвали (FAT);
- илдиз каталог (Root Directory, R Dir).

Файлларни жойлаштириш жадвали

Файлларни жойлаштириш жадвали, жуда муҳим маълумотлар структурасидир. Айтиш мумкинки, у, маълумотлар соҳасининг ҳолати ва унинг у ёки бу файл объектига мутаносиблиги тавсифланадиган маълумотлар соҳаси адрес картасидан иборат.

Маълумотлар соҳаси, кластерларга бўлинади. Кластер бу мантикий диск адрес маконида бир ёки бир нечта айқаш секторлардан иборатдир (аникроғи-фақат маълумот соҳасида). Кластер, файлга ажратиладиган, хотира дискининг адресланадиган минимал бирлигидир. Кластерлар, мантикий дискнинг маълумотлар соҳасидаги адресланадиган бирлиги сонини камайитириш учун киритилгандир.

V FAT ва FAT 32 файл тизимлари

FAT бошланғич файл тизими муҳим характеристикалардан бири файл номлари 8.3 форматидан фойдаланишдир. FAT стандарт тизимига (FAT 16 кўз тутилмоқда) яна 2 та, кенг тарқалган MS ОТ – Windows 95 ва Windows NT учун кўринишлари: V FAT (FAT виртуал тизими) ва FAT 32 тизимидир. Ҳозирги вақтда FAT 32 – Windows Millennium Edition, Windows 2000 ва Windows XP томонидан қўлланилади, бу FAT 32, тизимининг Windows NT ва Linux учун ҳам ишлаб чиқилган версиялари мавжуддир.

FAT ва V FAT файл тизимлари асосий камчилиги, мантикий дискнинг ката ўлчамларидаги кластерлашдаги йўқотишлар ва мантикий диск ўлчамига бўлган чегараланилардир. Шунинг учун ҳам MS Win 95 OEM Service Release 2 учун FAT 32 келди. У тўлиқ мустақил 32 разрядли файл тизимидир ва у олинги версияларга нисбатан кўп мукаммаликларга эгадир. Энг асосийси, FAT 32 диск соҳасини самарали сарфлайди.

HPFS файл тизими.

HPFS файл тизими (High File System – юқори унумдорликка эга бўлган файл тизимидир) биринчи марта OS/2 ва Law Manager OT ларида пайдо бўлди. Бу файл тизими, IBM ва MS компания мутахассислари томонидан MVS, VM/EMS файл тизимлари ва виртуал мкрожаат усусли тажрибаси асосида ишлаб чиқилди. HPFS кўпмасалалик режими файл тизими сифатида яратила бошлади ва катта ўлчамли дисклардаги файллар билан ишлашда юқори унумдорликни таъминлаш учун мўлжалланган эди.

HPFS асоси қилиб олинган файлларни дискда жойлаштириш принципи, файл тизимини унумдорлигини, ва унинг ишончлилиги ва бузилишларга катъийлилигини оширади.

HPFS файл тизими, FAT билан таққослаганда қуйидаги устунликларга эгадир:

- юқори унумдорлик;
- ишончлилик;
- файл ва каталогларга мурожаатни мосланувчи ҳолда бошқариш имконини берадиган кенгайтирилган атрибутларни қўллаш;
- диск маконидан самарали фойдаланиш.

Бу устунликлар HPFS структурасидан келиб чиқади.

NTFS файл тизими

NTFS (New Technology File System –янги технология файл тизими) файл тизими номида янги сўзи мавжуддир. Ҳақиқатда, NTFS файл тизими, таниқли FAT 16 (ва ҳатто FAT 32)га нисбатан сезиларли мукамалликлар ва ўзгаришларни ўз ичига олган. Фойдаланувчи нуктаи-назаридан қараганда, файллар ҳар доимдагидек (олдиндагидек), Windows муҳитида ишлаш вақтида кўпинча “папка” деб аталувчи каталогларда сақланади. Аммо унда талай янги хусусият ва имкониятлар пайдо бўлди.

NTFS файл тизими асосий имкониятлари.

NTFS ни лойиҳалашда алоҳида диққатни ишончлиликка, каталог ва файлларга мурожаатни чегаралаш механизмига, кенгайтирилган функционалликка , катта ҳажмдаги дискларни қўллашга ва х.к.ларга қаратилди. Бу тизим OS/2 V.3 доирасида ишлаб чиқилад бошлади, шунинг учун ҳам у HPFS файл тизими кўпгина қизиқарли хусусиятларини олган.

Ишончлилик. Юқори унумдорликка эга бўлган ва биргаликда фойдаланиш тизимлари, юқори ишончлиликка эга бўлиши керак. Бу эса NTFS тизимининг энг муҳим элементида иборатдир. NTFS тизими, ўз-ўзини тиклашнинг маълум воситаларига эга. Бу воситалар тизим яхлитлиги (бутунлиги) ни текширадиган, яъни транзакция журналинин олиб борадиган турли механизмларни ўз ичига олади.

NTFS, қайта юкланмасдан тинимсиз ишлашни таъминлаши муҳим бўлган сервер файл тизими сифатида ишлаб чиқилгани учун, унда худди HPFS даги каби, дефектли секторларни аварияли алмаштириш механизми ишончлиликни ошириш учун киритилган. Бошқача айтганда, маълумотларни ўқишда тизим рад этса, у ҳолда файл тизими бу маълумотларни ўқишга ва бу

мақсад учун диск маконини махсус резервлаб қайта кўчиради, дефект жойини-секторни белгилаб, унга бошқа мурожаат қилмайди.

Файл ва каталогларга мурожаат чегаралари. NTFS файл тизими Windows NT OT и хавфсизлик объект моделини қўллайди ва ҳамма том, каталог ва файлларга мустақил объект сифатида қарайди. NTFS тизими хавфсизликни файл ва каталоглар даражасида таъминлайди. Бу дегани, том, каталог ва файлларга мурожаат ҳуқуқи, фойдаланувчи ҳисоб ёзуви ва у мансуб бўлган гуруҳга боғлиқдир. Фойдаланувчи ҳар гал файл тизими объектига мурожаат қилганида унинг ҳуқуқи объектнинг мурожаатни бошқариш рўйхатидан текширилади (ACL). Агар фойдаланувчи зарур ҳуқуқга эга бўлса, унинг сўровномаси қондирилади, акс ҳолда сўровнома рад этилади.

Кенгайтирилган функционаллик. NTFS тизими мумкин бўлган кенгайтириш ҳисобга олиниб лойиҳалаштирилган. Унда кўпгина қўшимча имкониятлар акс этирилган- юқори даражада бузилишларга қатъийлик, бошқа файл тизимлари эмуляцияси, кучли хавфсизлик модели, маълумотлар оқимига параллел ишлов бериш ва файл атрибутларини яратиш (фойдаланувчи белгилайдиган).

Windows 2000/XP тизимларида, NTFS файл тизимидан фойдаланишда квотлаштиришни киритиш мумкин, бу фойдаланувчилар ўз файлларини диск маконининг уларга ажратилган квотаси чегарасида сақлаши мумкин.

NTFS тизими катта дисклар билан ишлашни ҳисобга олиб яратилган.

NTFS рухсатлари-бу фойдаланувчиларнинг объектларга мурожаатини чегаралаш учун берилган файл ва каталогларнинг махсус кенгайтирилган тўплами. Улар фақат NTFS файл тизми ўрнатилган томларда мавжуддир. Рухсатлар мосланувчан ҳимояни таъминлайди, чунки уларни ҳам каталог, ҳам алоҳида файлларга қўллаш мумкин; улар локал фойдаланувчиларга (ҳимояланган папка ва файллар бўлган компьютерларда ишлайдиган) ҳам, тармоқ орқали ресурсларга уланадиган фойдаланувчиларга ҳам мансубдир. Рухсат ва ҳуқуқларни чалкаштириб бўлмайди. Булар умуман ҳар хил тушунчалардир.

NTFS рухсатлари, биринчи навбатда, ресурсларни шу ресурс жойлашган компьютерда ишляпган фойдаланувчидан ҳимоя учун ишлатилади. Аммо улардан, масофадаги умумий папкага тармоқ орқали мурожаат қилувчи фойдаланувчиларга ҳам ишлатиш мумкин.

NTFS рухсатлар ҳимояни юқори даражада танлашни таъминлайди: ҳар бир папкадаги файлга ўз рухсатини ўрнатиш мумкин. Ҳар бир фойдаланувчининг рухсатлари, маълум амаллар бўйича фарқланади. Демак, ҳар бир файл объекти ўзининг мурожаатини бошқариш рўйхатига эга.

Ҳар бир файл объекти мурожаат маскасига эга (access mask-маска даступа). Мурожаат маскаси ўз ичига стандарт (standard), махсус (specific) ва кариндошлик (generic)-мурожаат ҳуқуқларига эга:

- стандарт мурожаат ҳуқуқи-ҳамма ҳимоя қилинган объектлар учун умумий амалларни аниқлайди;

- махсус мурожаат ҳуқуқи, файлни объектларга мос бўлган асосий ҳуқуқларни кўрсатади;
- қариндошлик мурожаат ҳуқуқлари тизими томонидан фойдаланилади, ular стандарт ва махсус ҳуқуқларкомбинациясини аниқлайди.

NTFS рухсатлари Windows NT 4.0 ва Windows 2000/XP операцион тизимлари учун рухсатлар ҳар хил келитирилган.

NTFS нинг Windows NT 4.0 даги рухсатлари. NTFS нинг файл ва каталогларга мурожаати Windows 4.0 даги рухсатлари индивидуал, стандарт ва махсус бўлади.

Индивидуал рухсатлар. Фойдаланувчига у ёки бу типдаги мурожаатни бериш имконияти индивидуал рухсатлар дейилади. Windows NT 4.0 да бундай рухсатлар 6 та: ўқиш (read), ёзиш (write), бажариш (execute), олиб ташлаш (delete), рухсатларни ўзгартириш (Change Permissions) ва эгасини ўзгартириш (Take Ownership).

Стандарт рухсатлар. Ҳар гал индивидуал рухсатларнинг комбинациясидан фойдаланмаслик учун, NTFS нинг стандарт рухсатлари киритилган.

- List - кўриш
- Add - қўшиш
- Add & Read – ўқиш ва ёзиш
- Change -ўзгартириш
- Full Control – тўлиқ мурожаат.

Махсус рухсатлар. Бу индивидуал рухсатларнинг стандарт рухсатлар билан мос келмайдиган комбинациялари: R,W,X,D,P ва O.

NTFS рухсатларини қўллаш. NTFS рухсатлари фойдаланувчилар ва гуруҳлар ҳисоб ёзувларига, худди умумий тармоқ ресурсларига мурожаат рухсатлари каби берилади. Фойдаланувчи рухсатни бевосита ёки рухсати бўлган битта ё бир нечта гуруҳ аъзоси бўлиб олиши мумкин.

Windows 2000 ва Windows XP ОТ ларида индивидуал, стандарт ва махсус рухсатлар ўрнига, фойдаланувчи интерфейсида 13 та рухсатлар мавжуд.

11-мәъруза. Тармоқ операцион тизимлари. Тармоқ операцион тизим таркиби: локал ресурсларни бошқариш воситалари, уз шахсий ресурсларини ва хизматларини умум фойдаланишга бериш-сервер кисми, масофадаги ресурс ва хизматларга мурожаат воситалари- операцион тизим клиент кисми ва коммуникация кисми.

Тармоқ операцион тизимининг структураси

Тармоқ операцион тизими ҳар қандай ҳисоблаш тармоғининг асосини ташкил этади. Тармоқдаги ҳар бир компьютер маълум бир даражада автоном ҳисобланади, шунинг учун тармоқ операцион тизими остида кенг маънода маълумотлар алмашиш мақсадида ўзаро алоқа қилувчи ва ресурсларни битта қоида – протоколлар асосида тақсимлаб берувчи алоҳида компьютерлар операцион тизимларининг йиғиндиси тушунилади. Тор маънода Тармоқ операцион тизими – бу алоҳида компьютернинг унга тармоқда ишлашни таъминлаб берувчи операцион тизимидир.



10.1-Расм. Тармоқ операцион тизимининг структураси.

Алоҳида машинанинг тармоқ операцион тизимида бир қанча қисмларни ажратиш мумкин (10.1-расм.):

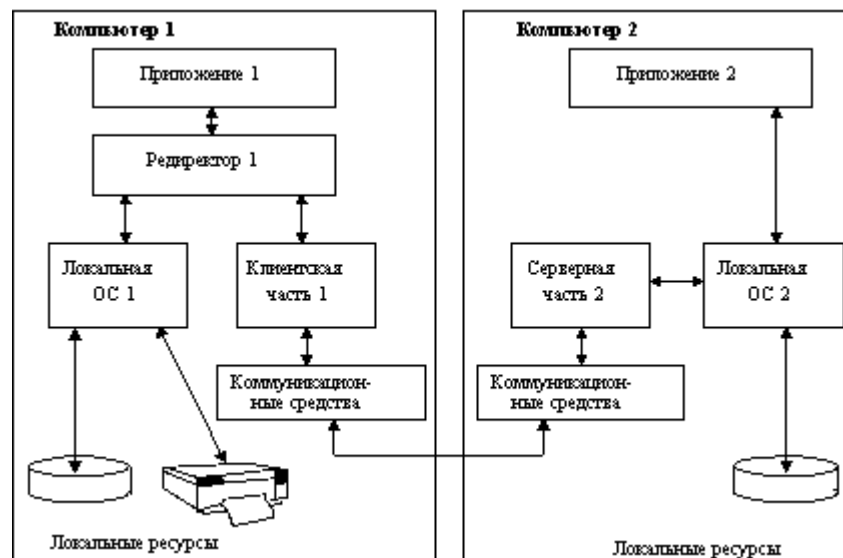
- Компьютернинг локал ресурсларини бошқариш воситалари: жараёнлар ўртасидаги оператив хотирани тақсимлаш функциялари, жараёнларни режалаштириш ва диспетчеризациялаш, мультипроцессор машиналарида процессорларни бошқариш, периферия қурилмаларини бошқариш ва локал операцион тизими ресурсларини бошқаришнинг бошқа функциялари.
- Шахсий ресурслар ва хизматларни умумий фойдаланишга тақдим этиш воситалари - операцион тизимнинг сервер қисми (сервер). Бу воситалар файллар ва ёзувларни биргаликда ишлашида керак бўладиган ҳимоясини, тармоқ ресурслари исмларини маълумотномасини олиб боришни, шахсий файл тизими ва маълумотлар базасига масофадан

рухсат бериш сўровларини қайта ишлашни, масофадаги фойдаланувчиларни ўзининг периферия қурилмаларига сўровлари навбатини бошқаришни таъминлайди.

- Масофадаги ресурс ва хизматларга бўлган рухсат сўровлари воситаси ва уларни ишлатиш – ОТ нинг клиент қисмидир(редактор). Бу қисм аниқланишни ва иловалардан ва фойдаланувчилардан сўровлар тармоғидаги узоқлаштирилган ресурсларга қайта йўналтиришни бажаради, шу билан бирга сўров локал формадаги иловалардан тушади, лекин тармоққа сервер талабларига тўғри келадиган бошқа формада узатилади. Клиент қисм яна сервер жавобларини қабул қилиш ва уларни локал форматга ўтказишни ҳам амалга оширади, шунинг учун иловада локал ва узоқлашган сўровларни бажаришни фарқи йўқ.
- ОТ нинг коммуникацион воситалари ёрдамида тармоқда маълумот алмашилиши рўй беради. Бу қисм маълумотларни адресацияси ва буферизациясини, тармоқда маълумот узатиш маршрутини танлаш, узатиш ишончлилиги кабиларни таъминлайди, яъни маълумотларни ўтказиб қўйиш воситаси ҳисобланади.

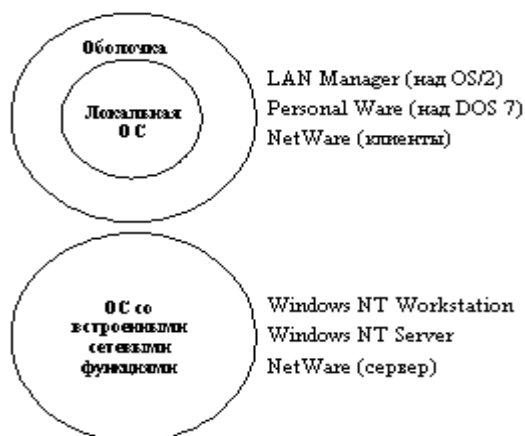
Бирор конкрет компьютерга қўйилган функцияга қараб унинг операцион тизимида ё клиент қисми ё сервер қисми йўқ бўлиши мумкин.

10.2 – расмда тармоқ компонентларини ўзаро алоқаси кўрсатилган. Бу ерда компьютер 1 “соф” клиент вазифасини, компьютер 2 эса “соф” сервер вазифасини бажаради, мос равишда биринчи машинада сервер қисми иккинчи машинада клиент қисми йўқдир. Расмда клиент қисмининг редиректор компонентаси алоҳида кўрсатилган. Худди шу редиректор иловалардан келаётган ҳамма сўровларни ушлаб олади ва уларни таҳлил қилади. Агар берилган компьютер ресурсига сўров жўнатилаган бўлса, унда у локал ОТ нинг таалуқли осттизимида қайта адресланади, агарда бу сўров узоқлаштирилган ресурсга бўлса, унда у тармоққа қайта йўналтирилади. Бунда клиент қисм локал формадан тармоқ форматига ўтувчи сўров яратади ва уни кўрсатилган серверга маълумотлар етказилишига жавоб берувчи транспорт осттизимида беради. Компьютер 2 ОТнинг сервер қисми сўровни қабул қилади, қайта кўради ва уни ўзининг локал ОТ сини бажариш учун бериб юборади. Натижа олинган кейин, сервер транспорт осттизимида мурожаат қилади ва сўров берган клиентга жавобни юборади. Клиент қисм натижани керакли форматга ўтказади ва уни сўров жўнатган дастурга қайта адреслайди.



10.2 – расм. компьютерларнинг ўзаро алоқасидаги операцион тизим компонентларининг ўзаро алоқаси.

Амалиётда тармоқ операцион тизимларини куришда бир нечта ёндашишлар



юзага келди (10.3 – расм).

10.3 – расм. Тармоқ ОТ ларни қуриш вариантлари

Биринчи тармоқ ОТ лари ҳозирги локал ОТларнинг йиғиндисидан ва унинг ўрнатилган тармоқ қобиғидан иборат бўлган. Шу билан бирга локал ОТ га асосий тармоқ функцияларини бажарувчи тармоқ қобиғи учун керак бўладиган минимум тармоқ функциялари ўрнатилган. Бундай ёндашишга мисол бўлиб ҳар бир компьютерда MS DOS операцион тизимининг ишлатилишини олиш мумкин(учинчи версиясидан бошлаб унда файлларга қўшма рухсатни таъминлайдиган файлларини ва ёзувларни қулфлаш каби ўрнатилган функциялари пайдо бўлди). Локал ОТ устидан тармоқ қобиғи кўринишида тармоқ ОТларини қуриш принциплари замонавий ОТларда ишлатилади, буларга, масалан, LANtastic ёки Personal Ware киритиш мумкин.

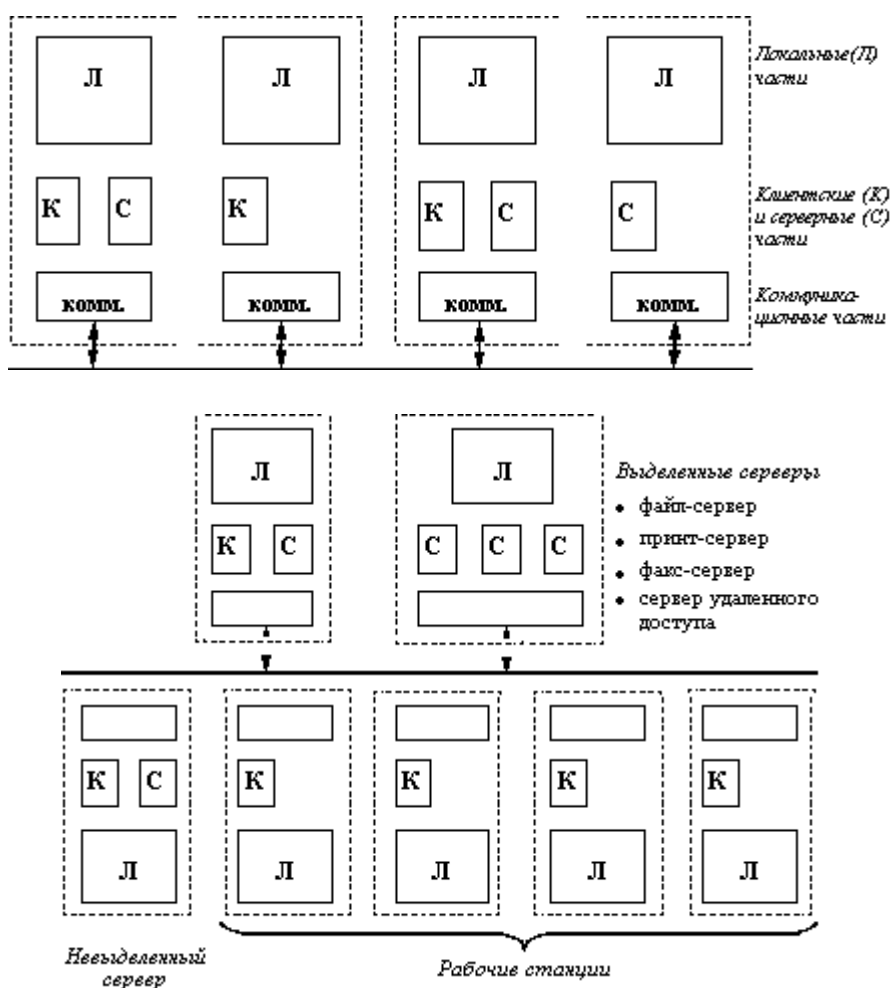
Бироқ бошданок тармоқда ишлаш учун мўлжалланган операцион тизимларни ишлаб чиқиш йўли анча эффектли кўринади. Бундай ОТ ларнинг тармоқ функциялари уларнинг асосий тизим модуллари ичига чуқур ўрнатилган бўлади, бу эса уларнинг логик мукамаллиги, осон ишлатилиши ва модификацияси, шунингдек юқори унумдорлигини таъминлайди. Бундай ОТ га мисол қилиб Microsoft фирмасининг Windows NT тизими ҳисобланади, бу тизим ўрнатилган тармоқ воситалари ҳисобига маълумотларни ишлаб чиқарилиши ва ҳавфсизлигини шу фирмани ўзидан чиққан (IBM фирмаси билан бирга) LAN Manager ОТ таққослаганда анча юқори кўрсаткичларини таъминлайди.

Бир рангли тармоқ операцион тизими ва ажратилган серверли операцион тизим

Тармоқ компьютерлари ўртасида функцияларни қандай бўлинганлигига қараб, тармоқ ОТ ва шу билан тармоқларнинг ўзи ҳам иккита синфга бўлинади: бир рангли ва икки рангли (10.4 – расм). Охиргисини кўпроқ ажратилган серверли тармоқлар деб аташади.

а)

(б)



10.4 – расм. (а) – Бир рангли тармоқ, (б) – Икки рангли тармоқ.

Агар компьютер ўзининг ресурсларини тармоқдаги бошқа фойдаланувчиларга тақдим қилса, унда у сервер ролини ўйнайди. Бунда бошқа машина ресурсларига мурожаат қилувчи компьютер клиент ҳисобланади. Юқорида айтилганидек тармоқда ишлайдиган компьютер ё клиент функциясини, ё сервер функциясини, ёки иккаласининг ҳам функциясини бажариши мумкин.

Агар бирор бир сервер функцияларининг бажарилиши компьютернинг асосий вазифаси бўлса(масалан, файлларни тармоқдаги фойдаланувчиларга умумий фойдаланишга тақдим этиш, факсдан умумий фойдаланшни ташкил қилиш ёки тармоқдаги фойдаланувчиларни ўзининг дастурларидан фойдаланишга рухсат бериш), унда бу компьютер ажратилган сервер деб аталади. Сервернинг қайси ресурси ажратувчи эканлигига қараб, у файл – сервер, факс – сервер, принт – сервер, илова сервер ва ҳ.к деб аталади.

Шубҳасиз, ажратилган серверларда шунақа ОТ ни ўрнатиш мақсадга мувофиқ, улар у ёки бу сервер функцияларини оптимал бажаришга йўналтирилган бўлиши керак. Шунинг учун ажратилган серверли тармоқларда таркибига сервер қисмлари имкониятлари билан фарқ қилувчи бир нечта вариантли ОТлари қирувчи тармоқ операцион тизимлари ишлатилади. Масалан, Novell NetWare тармоқ ОТи файл – сервер сифатида ишлайдиган серверли вариантга эга, яна ҳар хил локал ОТ лар билан ишлайдиган ишчи станциялар учун қобиғинга эга. ОТ ларнинг бошқа мисоли сифатида, тармоқни ажратилган серверли қуришга мўлжалланган ОТ Windows NT ҳисобланади. NetWare дан фарқли ўлароқ Windows NT нинг иккала варианты , Windows NT Server(ажратилган сервер учун) ва Windows NT Workstation(ишчи станциялар учун) ҳам клиент ҳам сервер функцияларини қўллаб қувватлайди. Лекин Windows NT нинг серверли варианты тармоқдаги бошқа фойдаланувчиларга компютеридаги ресурсларини кўрсатиш учун кўпроқ имкониятларга эга, чунки у анча кенг функцияларни бажариши мумкин, катта миқдордаги клиентларнинг бир вақтда уланишини қўллайди, анча кенг ҳимоя воситаларига ҳам эгадир.

Ажратилган серверни унинг асосий вазифаси билан боғлиқ бўлмаган оддий масаларни ечиш учун ишлатиладиган компьютер сифатида ишлатиш мумкин эмас, бу унинг сервер сифатида ишлаш унумдорлигини камайтиради. Шуларни ҳисобга олган ҳолда Novell NetWare ОТ ининг сервер қисмида оддий дастурларни бажариш имкониятлари умуман кўзда тутилмаган, яъни клиент қисми йўқ, ишчи станцияларда эса сервер компонентлари мавжуд эмас. Бироқ бошқа тармоқ ОТларда ажратилган сервер клиент қисмининг ишлаши бемалол мумкин. Масалан, Windows NT Server бошқаруви остида локал фойдаланувчининг тармоқдаги бошқа компьютернинг ресурсларига сўров тушиши билан ОТ клиент қисмининг функцияларини бажарадиган оддий дастурлари юкланиши мумкин. Бунда Windows NT Workstation ОТи ўрнатилган ишчи станциялар ажратилмаган сервер функцияларини бажариши мумкин.

Энг муҳими шуни тушуниш керакки, ажратилган серверли тармоқларда ҳамма компьютерлар умумий ҳолда ҳам сервер ҳам клиент ролини бажара олса ҳам, бу тармоқ функционал симметрик эмас, аппарат ва дастурий жихатдан унда икки турли компьютерлардир – бири, юқори даражада сервер функцияларини бажаришга қаратилган ва сервер ОТлари остида ишлайдиган, иккинчиси эса асосан клиент функцияларини бажаришга қаратилган ва шунга мос ОТлар остида ишлайдиганлар. Функционал симметрик эмаслик, қоида бўйича, аппарат симметрик эмасликни келтириб чиқаради, бунда ажратилган серверлар учун катта оператив ва ташқи хотирали анча кучли компьютерлар ишлатилади.

Бир рангли тармоқларда ҳамма компьютерлар бир – бирларининг ресурсларига бўлган рухсат ҳуқуқлари тенгдир. Ҳар бир фойдаланувчи ўз хоҳишига кўра компютеридаги бирон бир ресурсни умумий фойдаланишга қўйиши мумкин, кейин бошқа фойдаланувчилар ундан фойдаланиши мумкин. Бундай тармоқларда ҳамма компьютерларда тенг потенциалли имкониятлар яратадиган бир хил ОТлар ўрнатилади. Бир рангли тармоқлар LANtastic, Personal Ware, Windows for Workgroup, Windows NT Workstation каби ОТлар базасида қурилган бўлиши мумкин.

Бир рангли тармоқларда шунингдек функционал симметрик эмаслик келиб чиқиши мумкин: биринчи ҳолда бир фойдаланувчилар бошқалар билан ўз ресурсларини бўлишишни хохламайди, бунда у клиент ролинни бажаради, иккинчи ҳолда администратор фақатгина умумий фойдаланишда бўлган ресурсларни ташкил қиладиган функцияларни боғлаб қўяди ва бу билан серверлик вазифасини бажаради, учинчи ҳолда локал фойдаланувчи ўзининг ресурсларидан бошқалар фойдаланишига қарши бўлмаса ва ўзи бошқа компьютерга мурожаатини истисно қилмаса, унда бунга ўрнатиладиган ОТ ҳам сервер ҳам клиент қисмига эга бўлиши керак. Ажратилган серверли тармоқларга қараганда, бир рангли тармоқларда функционал йўналтиришга(клиентга ёки серверга) боғлиқ ҳолдаги ОТлар йўқ. Ҳамма вариациялар бир хил вариантли ОТлар конфигурацияси воситалари билан амалга оширилади.

Бир рангли тармоқларни ташкил қилиш ва ишлатиш қулай, бироқ улар асосан унча катта бўлмаган фойдаланувчилар гуруҳларини бирлаштиришда қўлланилади, бу фойдаланувчилар сақланадиган маълумотлар ҳажмига, унинг рухсат этилмаган киришдан ва тезлигига киришдан сақлашга катта шартлар қўймайди. Бу характеристикаларга оширилган шартлар қўйилганда энг моси икки рангли тармоқлар ҳисобланади, бунда сервер ўзининг ресурслари билан фойдаланувчиларга хизмат кўрсатиш масаласини яхшироқ ҳал қилади, чунки унинг аппаратураси ва тармоқ операцион тизими шу мақсад учун махсус ишланган.

Ишчи гуруҳлар учун ОТ ва корхона масштабидаги тармоқлар учун ОТ

Тармоқ операцион тизимлари ҳар хил хусусиятларга нимага мўлжалланганлигига қараб эга бўлиши мумкин:

- *Бўлим тармоғи* – умумий масаларни ечадиган унча катта бўлмаган ишчилар гуруҳи томонидан ишлатилади. Бўлим тармоғининг асосий мақсади илова, маълумот, лазер принтерлари ва модемлар каби локал ресурсларни тақсимлаш ҳисобланади. Бўлим тармоқлари одатда қисм тармоқларга бўлинмайди.
- *Кампус тармоқлари* – бир нечта бўлим тармоқларини алоҳида бинода ёки корхонанинг бирор ҳудуди ичида бирлаштиради. Бу тармоқлар ҳали ҳам локал тармоқлар ҳисобланади, шундай бўлсада улар бир неча километр квадратни қоплаши мумкин. Бундай тармоқнинг сервиси бўлим тармоғининг ўзаро алоқасини, корхона маълумотлар базасига рухсатини, факс – серверга, юқори тезликли модем ва принтерларга рухсатини ўзида мужассам этади.
- *Корхона тармоқлари(корпоратив тармоқлар)* – алоҳида корхонанинг ҳамма ҳудудларининг бор компьютерларини бирлаштиради. Улар шаҳар, вилоят ва ҳаттоки бутун қитъани ҳам қоплаши мумкин. Бундай тармоқларда фойдаланувчиларга бошқа ишчи группалар, бошқа бўлимлар, остбўлимлар ва корпорация штаб – квартираларидаги маълумотлар ва иловаларга рухсат берилади.

Бўлим масштабидаги тармоқларда ишлатиладиган операцион тизимларнинг асосий вазифаси бўлиб илова, маълумотлар, лазер принтерлари ва ҳаттоки паст тезликли модемлар каби ресурсларни тақсимлашни йўлга қўйиш ҳисобланади. Одатда бўлим тармоқлари бир ёки иккита файл сервери ва 30 тадан кўп бўлмаган фойдаланувчиларга эга бўлади. Бўлим даражасидаги вазифаларни бошқариш нисбатан осон. Администратор вазифасига янги фойдаланувчиларни қўшиш, оддий қайтаришларни йўқотиш, янги тугунларни ва дастурий таъминотни янги версияларини ўрнатиш кабилар киради. Бўлим тармоқларининг операцион тизимлари ўзларига ўхшаб яхши ишланган ва ҳар хилдир. Бундай тармоқ одатда битта ёки максимум иккита ОТ ишлатиши мумкин. Кўпроқ бу тармоқ ажратилган серверли NetWare 3.x ёки Windows NT, ёки бўлмаса бир рангли тармоқ, масалан, Windows for Workgroups тармоғи бўлиши мумкин.

Бўлим тармоқларининг фойдаланувчилари ва администраторлари кўп ўтмай корхонасидаги бошқа бўлимлар маълумотларига рухсат олиш билан ўз ишининг эффективлигини ошириши мумкинлигини тушунишади. Агар савдо – сотиқ билан шуғулланувчи ходим конкрет маҳсулотнинг характеристикаларига рухсатни олса ва уларни ўзининг презентациясига қўшса, унда бу маълумот анча янги бўлади ва харидорларга катта таъсир ўтказди. Агар маркетинг бўлими муҳандимлар бўлими томонидан эндигина ишланаётган маҳсулот характеристикасига рухсатга эга бўлса, унда у ишлаб чиқариш тугаши билан маркетинг материалларини тезда тайёрлаши мумкин. Демак, тармоқ эволюциясининг кейинги қадами бўлиб бўлимларнинг бир нечта локал тармоқларни бинолар ёки бинолар гуруҳларига бирлаштириш ҳисобланади.

Бундай тармоқлар кампус тармоқлари дейилади. Кампус тармоқлари бир неча километргача чўзилиши мумкин, лекин бунда глобал туташтириш шарт эмас.

Кампус тармоғида ишлаётган операцион тизим, бир бўлим ходимлари учун бошқа бўлим файл ва ресурсларига рухсатини таъминлаб бериши керак. Кампус тармоғи ОТлари томонидан кўрсатилаётган хизматлар файл ва принтерларни оддий ажратиш билан чегараланиб қолмайди, балки факс – сервер ва юқори тезликли модем серверлари кабиларга бўлган рухсатни таъминлаб беради. Берилган синф операцион тизими томонидан берилаётган муҳим сервиси бўлиб корпоратив маълумотлар базасига рухсат ҳисобланади. Ҳудди шу кампус тармоқлари даражасида интеграция муаммолари бошланади. Умумий ҳолда, бўлимлар ўзлари учун компьютер турлари, тармоқ қурилмалари ва тармоқ операцион тизимларини танлаб олишган. Масалан, муҳандислик бўлими UNIX операцион тизимини ва Ethernet тармоқ ускуналарини ишлатиши мумкин, савдо – сотиқ бўлими DOS/Novell операцион муҳитини ва Token Ring ускуналарини ишлатиши мумкин. Кўп ҳолларда кампус тармоғи ҳар хил компьютер системаларини бирлаштиради, бу пайтда бўлим тармоқлари бир турдаги компьютерларни ишлатади.

Корпоратив тармоқ корхонанинг ҳамма остбўлимларини туташтиради, умумий ҳолда улар бир – биридан сезиларли масофада жойлашган бўлади. Корпоратив тармоқлар локал тармоқлар ёки алоҳида компьютерларни туташтириш учун глобал алоқа(WAN links)ни ишлатади.

Корпоратив тармоқ фойдаланувчиларига бўлим ва кампус тармоқларида бор хизмат ва иловалар талаб қилинади, қўшимчасига яна бир қанча қўшимча иловалар ва хизматлар, масалан, мейнфрейм ва мини компьютерлар иловалари ва глобал алоқаларга рухсат талаб қилинади. Локал тармоқ ёки ишчи гуруҳ учун ОТ ишлаб чиқарилаётганда, унинг асосий мажбурияти бўлиб локал уланган фойдаланувчилар ўртасида файллар ва бошқа тармоқ ресурсларини тақсимлаш ҳисобланади. Бундай муносабат корхона даражаси учун қўлланилмайди. Файл ва принтерларни тақсимлаш билан боғлиқ базали сервислар қаторида, корпорациялар учун ишлаб чиқариладиган тармоқ ОТлари анча кенг сервислар тўпламини қўллаши керак. Бундай тўпламга почта хизмати, коллектив иш воситалари, узоқлашган фойдаланувчиларни қўллаш, факс – сервис, овоз хабарларини қайта ишлаш, видеоконференциялар ташкил қилиш ва ҳ.к. лар қиради.

Бундан ташқари, корпоратив тармоқлар учун кичик масштабдаги тармоқларнинг масалаларини анъанавий йўл билан ечишдаги мавжуд усуллар ва ёндашишлар яроқсиз бўлиб чиқди. Биринчи планга шундай муаммолар ва масалалар чиқдики, улар ишчи гуруҳ тармоқларида, бўлим ва ҳаттоки кампус тармоқларида ё иккинчи даражали қийматга эга бўлган ёки умуман активлигини кўрсатмаган. Масалан, кичикроқ тармоқ учун энг оддий масала фойдаланувчилар ҳақидаги маълумотларни олиб бориш корхона масштабидаги тармоқлар учун катта муаммога айланди. Глобал алоқаларни ишлатиш эса корпоратив ОТлардан паст тезликли чизикларда яхши

ишлайдиган протоколлар ёрдами ва бир хил анъанавий ишлатиладиган протоколлардан воз кечиш талаб қилади.

Корпоратив ОТларни белгиларига қуйидаги хусусиятларни киритиш мумкин.
Иловалар ёрдами. Корпоратив тармоқларда мураккаб иловалар бажарилади, улар бажарилиши учун катта ҳисоблаш қувватини талаб қилади. Бундай иловалар бир нечта қисмларга бўлинади, масалан, битта компьютерда маълумотлар базасига бўлган сўровларни бажарилиши билан боғлиқ иловаларнинг бир қисми бажарилади, бошқасида файл – сервисга бўлган сўровлар, клиент машиналарида эса илова маълумотларини қайта ишлаш мантиғини амалга оширувчи ва фойдаланувчи билан интерфейсни амалга оширувчи қисми бажарилади. Корпорация дастурий тизимлари учун умумий ҳисоблаш қисми клиентларнинг ишчи станциялари учун жуда катта ҳажмли ва кўтара олмайдиган бўлиши мумкин, шунинг учун агарда уларнинг мураккаб ва ҳисоблаш муносабатидаги қисмларини махсус шунга мўлжалланган компьютер – *илова серверига* кўчирилса иловалар эффектлироқ бажарилади. Иловалар сервери кучли аппарат платформасида жойлашиши керак. Илова сервери ОТлари ҳисоблашларни юқори ишлаб чиқаришини таъминлаши керак, демак кўп йўлли қайта ишлашни, сиқиб чиқарувчи кўп масалаликни, мультипроцесслашни, виртуал хотирани ва энг кўп танилган амалий муҳитлар(UNIX, Windows, MS-DOS, OS/2) ни таъминлаши керак. Бундай муносабатда NetWare тармоқ ОТни корпоратив маҳсулотларга киритиш қийин, чунки унда иловалар серверига қўйиладиган деярли ҳамма талаблар мавжуд эмас. Шу билан бирга Windows NTдаги универсал иловаларни яхши қўллаши унга корпоратив маҳсулотлар дунёсида ўрин эгаллашига курашишга ёрдам беради.

Маълумотнома хизмати. Корпоратив ОТ ҳамма фойдаланувчилар ва ресурслар ҳақидаги маълумотларни шундай сақлаш қобилятига эга бўлиши керакки, унда битта марказий нуқтадан уни бошқаришни таъминлаши керак. Катта ташкилотларга ўхшаб корпоратив тармоқ ўзи ҳақидаги иложи борица бутун маълумотларни(фойдаланувчилар, сервер, ишчи станциялар ҳақидаги маълумотлардан тортиб, то кабел тизими маълумотларигача) марказий сақланишига муҳтождир. Бу маълумотларни маълумотлар базаси кўринишида сақлашни табиий равишда ташкил қилиш. Бу базадан маълумотлар кўп тармоқ тизимлари иловалари томонидан талаб қилиниши мумкин, биринчи навбатда бошқариш ва администаторлаш тизимлари томонидан. Бундан ташқари, бундай база электрон почтани ташкил қилиш, бирга ишлаш тизимлари, ҳавфсизлик хизмати, тармоқнинг дастурий ва аппарат таъминоти инвентаризацияси хизмати ҳамда деярли ҳамма йирик бизнес – иловалар учун фойдалидир.

Идеал ҳолатда маълумотномали информация ягона маълумотлар базаси кўринишида амалга оширилиши керак. Масалан, Windows NT да камида бешта ҳар хил турли маълумотномали маълумотлар базаси мавжуд. Домен(NT Domain Directory Service)нинг асосий маълумотномаси фойдаланувчилар ҳақида уларни тармоққа логик киришни ташкил қилишда ишлатадиган маълумотларни сақлайди. Шу фойдаланувчилар ҳақидаги

маълумотлар Microsoft Mail электрон почтаси томонидан ишлатиладиган бошқа маълумотнома таркибида ҳам бўлади. Яна учта маълумотлар базаси паст даражали адреслар рухсатини қўллайди: WINS – Netbios – исмларининг IP – адресларга мослигини ўрнатади, DNS маълумотномаси – домен исмлари сервери – NT тармоқларини интернетга уланишида фойда келтираркан, ва ниҳоят, DHCP протоколи маълумотномаси тармоқ компьютерларига IP – адресларни автоматик тарзда тайинлашида ишлатилади. Идеалга яқин маълумотларномалар хизмати Banyan(Streetwork III маҳсулоти) фирмаси ва Novell (NetWare Directory Services) фирмаси томонидан келтирилганларда жойлашган, улар ҳамма тармоқ иловалари учун ягона маълумотномани таклиф қилади. Тармоқ операцион тизимлари учун ягона маълумотномалар хизмати борлиги корпоративликнинг энг муҳим белгиларидан биридир.

Ҳавфсизлик. Корпоратив тармоқ ОТ лари учун асосий муҳимлигини маълумотларни ҳавфсизлиги масалалари ташкил қилади. Бир тарафдан, йирик масшабли тармоқларда объектив равишда санкцияланмаган рухсат(кириш) учун кўп имкониятлар мавжуд, сабабларига қуйидагилар киради: маълумотларни демарказлаштириш ва “қонуний” нуқталарни катта тақсимланганлиги, фойдаланувчилар сонининг кўплиги, шунингдек, тармоққа уланишда санкцияланмаган нуқталар сонининг кўплиги кабилардир. Бошқа тарафдан, корпоратив бизнес – иловалар корпорациянинг бутунлигича самарали ишлаши учун муҳим қийматга эга бўлган маълумотлар билан ишлайди. Шундай маълумотларни ҳимояси учун ҳар хил аппарат воситалари қаторида корпоратив тармоқларда ҳимоя воситаларининг операцион тизими томонидан тақдим этилган ҳамма спектрлари(сайланган ёки мандат рухсат ҳуқуқлари, фойдаланувчилар аутентификациясининг мураккаб процедуралари, дастурий шифрация) ишлатилади.

12-март. Тармоқ операцион тизимлари синфлари. Тармоқ операцион тизимларини куриш вариантлари. Бир хил мавкели ва ажратилган серверли тармоқ операцион тизимлари, ишчи гуруҳ (бу\лим), корхона (компус) масштабидаги ва корпоратив тармоқ операцион тизимлари. Корпоратив тармоқ операцион тизимлари, корпоративлик хусусиятлари.

“Қайси тармоқ ОТ лари яхшироқ” деган савол ҳар доим, компьютерга тегишли газета ва журналларда жуда кўп муҳокама қилинади.

Кимдир Windows NT – энг идеал иловалар сервери дейди, м-н, Билл Гейтс “Windows NT – UNIX” нинг, UNIX га қараганда энг яхши варианты деган. (уни албатта тушунса бўлади).

Ҳар бир ўз фикрини билдираяпган муаллифлар, ўқувчиларнинг ҳаммасида нарҳи қиммат қурилмалар бир деган ишонч билан гапирадилар.

Биз эса амалий тажрибалардан келиб чиққан ҳолда, асосий тармоқ ОТ ларини яъни Novell Netware 4.1, Microsoft Windows NT Server 4.0 ва UNIX ОТ ларни, уларни админстрлаш ва реал кўп платформали гетероген платформада ишлатиш хусусиятларига асосланиб, уларнинг асосий сиёсий характеристикасини берамиз. Албатта, бу ОТ лардан бошқа ОТ лар ҳам кўп, аммо биз ўзимизда, СНГ давлатлари ва Россияда кўпроқ фойдаланилган шу ОТ ларни мисол қилиб олдик. Албатта, бу ОТ ларнинг ҳам турли версиялари мавжуд, баъзи ҳолларда уларни маълум иловалар билан ишлатиш мумкин ва х.к.лардир, аммо биз уларнинг умумий характеристикаси бўйича таққослаймиз. Ва энг умумий кўрсаткичлар бўйича уларни кўриб чиқамиз.

ОТ лар асос характеристикалари.

Бу уччала ОТ лар кўп масалалидир. Аммо Netware да кўп масалалик-кооперативдир, шу билан бир қаторда, жараёнлар серверда тизим ядро режимида бажарилади (яъни жараёнлар хотираси химояси йўқдир).

Бу хусусият эса ўз навбатида бу тизимни, иловалар сервери сифатида қўллашнихавфли қилиб қўяди, чунки ихтиёрий нотўғри ёзилган илова серверни “осиб” қўйиши мумкин. Шу билан бирга айнан шу хусусият, ОТ ни юқори даражада унумдорликка эришишига олиб келади, чунки бу хосса жараёндан жараёнга ўтишда, фойдаланувчи режимида ядро режимига ўтишда ва тескарисида ҳам харажатларни камайтиради ва тез ўтишга имкон беради, натижада тизим катта тезликда ишлайди. Бир хил шароитда, Netware Windows NT ёки UNIX га нисбатан юқори унумдорликка эга.

Уччала ОТ ҳам симметрик, кўпроцессорли ишлов беришни қувватлайди (SMP) аммо бу хусусият Netware учун муҳим эмас, шунинг учун ҳам буни фақат сертификацияланган серверларнинг тор доирасида ишлатилади. Албатта кўпроцессорли ишлов беришни Windows NT ёки UNIX учун қўллаш қулай, яъни шуни таъкидлаш мумкинки, UNIX нинг кўп версиялари, ўнлаб процессорли серверларда аъло даражада ишлайди, уларнинг масштабластирилганли Windows NT га нисбатан юқори.

UNIX асосидаги кластер тизимлари ҳам ўзларини яхши кўрсатганлар, аммо бу учун UNIX нинг ҳамма версиялари ҳам ярай бермайди. Windows NT

учун кластер тизими тугалланган эмас. Netware тўлиқ аппарат қатъийликни таъминлайди. (SFT III), Windows NT да бундай ўхшаш имконият йўқ, (бундай хусусият индивидуал бўлиб, аппарат таъминотга боғлиқ бўлса ҳам)

Бузилишларга қатъийлик учун намуна албатта – мейнфреймлардир.

ОТ хавфсизлиги

Netware 4.1 C2 синфи хавфсизлик талабларига жавоб беради (тармоқ конфигурацияларига қўйиладиган (“красная книга”)), Windows NT 3.5 – эса ишчи станциялар учун C2 синфи (“оранжевая книга”).

Microsoft, Windows NT ни ҳам “оранжевая книга” ҳам “красная книга” бўйича C2 синфига мослигини сертификация қилинган.

Ҳамма замонавий UNIX тизимларда, ё асосда, ёки қўшимча модул асосида C2 хавфсизлик синфига мос келади (ишчи синфлар учун) баъзи ва ҳолларда юқори синфига ҳам мос келади. UNIX нинг баъзи версиялари, C2 нинг C2 синфига ҳам мос келади.

Windows NT да C2 синфи талабларини қўллаш ядро даажасида амалга оширилган. Бу эса, баъзи экспертларга, Windows NT UNIX га нисбатан юқорироқ хавфсизлик даражасини таъминлайди деб ҳисоблашига асос бермоқда. Аммо бу ундай эмас. Агар тизим сертификация қилинган бўлса, бошқа сўз бўлиши мумкин эмас.

Кўп фойдаланувчилик режим.

UNIX нинг жиддий устунлиги, Windows NT ва Netware да бўлмаган кўп фойдаланувчиликни қўллашдир.

Битта UNIX – машинага, хатто ШК асосида бўлса ҳам, ўнлаб алфавит рақамли терминалларни улаш мумкин. Бу нима учун керак? Масалан, транспорт агентлигини олайлик (поезд ёки автобуслар учун битта сотувчи, ёки аҳолидан жамғарма қабул қилувчи банк). Microsoft нуқтаи-назаридан ҳамма шундай жройларни Windows ли ШК лар билан жихозлаш керак., ва уларга МББТ ўрнатиб, (албатта MS SQL Server), клиент-сервер режимида ишлайдиган тармоқ ўрнатиш керак. Аммо бундай иш жойларига график интерфейс тез тармоқ каналлари ва юқори унумдорликка эга бўлган ШК керак эмас. Албатта бу жуда чиройли, аммо бу рационал эмас, бу тежамли эмас. Бундай жойларда, UNIX машина қўйиб, секин осинхрон каналлари орқали қиммат бўлмаган алфавит-рақамли терминаллар қўйиш арзон тушади. Бунда, клиент сервер архитектурали мураккаб тармоқ керак эмас.

Агар терминаллар жуда кўп талаб этилса, у ҳолда мейнфрейм сотиб олиш қулай (анчайин арзон нархда). Яхши мейнфрейм юзлаб, хатто минглаб терминалларни тортиши мумкин.

Тармоқ график интерфейси билан иш бошқачадир. Очiq тизимлар оламида график интерфейсни тармоқда қўлловчи учун XWin.System стандарт ҳисобланади. Бу тизимда, топшириқ бажарилувчи тизим X-клиент деб аталади. Маълумот киритиладиган қурилма (ШК ёки X.терминал) X-сервер дейилади. Бунда X-клиент ва X-сервер битта компьютерда, ёки ҳар хил, баъзан мутаносиб бўлмаган компьютерларда амалга оширилади. XWin

System тизимлари аниқ ОТ дан мустақил яратилган бўлса ҳам уни ҳар ОТ ларда амалга ошириш катта фарқ қилади. Бу тизим, асосан UNIX тизимларига мўлжалланган, ва кўпгина кенгайтмаларига эга, бу эса уни имкониятини оширади.

NetWare учун XWin System қўлланиши йўқдир (оддий X.Console дан ташқари).

Тармоқ ресурсларини мантиқий ташкил этиш.

NetWare 4.1 да тармоқ инфраструктурасини ташкил этишнинг энг яхши воситаси – NetWare (NDS) каталоглар хизматидир. У тармоқ ресурслари иерархиясини глобал даражада ташкил этишга имкон беради. Бунинг учун фойдаланувчига ихтиёрий ресурсга мурожаат қилиш учун тармоқда бир марта ҳисобдан ўтса етарлидир. NDS нинг камчилиги бу хизмат учун амалий дастурлар етарли эмаслигидир.

NDS нинг бошқа платформа учун амалга ошиши тўғрисида гап кетса, Nowell фирмаси NDS га лицензияни, ОТ лар ишлаб чиқувчиларнинг ҳаммасига сотишини мўлжаллаган.

Windows NT Server да тармоқ доменлар асосида яратилади, бу эса ўз навбатида корпоратив мижозлар учун, каталоглар хизматида нисбатан қулай эмас.

UNIX учун NIS тармоқ ахборот тизимидан (SUN фирмаси томонидан ишлаб чиқилган) фойдаланиш одатий ҳолдир. Бу хизмат, доменлар асосида тармоқ ресурсларини ташкил этишга имкон беради, аммо Windows NT дан фарқли равишда доменлар ўртасида ишончли муносабатларни ўрнатишга имкон бермайди.

NIS ни бошқариш мураккаброқ масала: кўп матнли файлларни “қўлда” таҳрирлашга тўғри келади.

Ҳозирги вақтда UNIX баъзи муаллифлари ўз каталоглар хизматини, одатда X.500 стандарт асосида ишлаб чиқаяптилар. Аммо, улар учун Microsoft ёки Novell фирмаси каталоглар хизматидан фойдаланиш мақсадга мувофиқдир.

Очиғини айтганда, корпоратив гетероген тармоқ қуришнинг энг нозик жойи, бу тармоқ ресурслари тарқоқлиги ва ягона хизматни (каталог ёки домен) ШК лардан то мейнфрейм, суперкомпьютерларга қўллаш имкони йўқлигидир.

Файл ва босмадан чиқариш хизматлари

Тест натижалари бўйича энг унумдор файл тизимида албатта NetWare 4.1 эгадир. Унга жуда яқин Windows NT келади, аммо барибир мижозлар кўп сонли бўлганда ва серверни юқори унумли тармоқ каналига улашда NetWare архитектура хусусиятлари маълум устунликка эга.

Тармоқ файл тизими – бу UNIX нинг энг нозик жойидир. Очик тизимлар дунёсида ва UNIX учун стандарт бўлиб NFS ҳисобланади. У, Sun фирмаси томонидан ўн беш йиллар олдин яратилган. Ўшандан бошла, UNIX

учун, кўплаб мукамалроқ тармоқ файл тизимлари таклиф этилди, аммо улар кенг тарқалиб кетмади.

NFS паст унумдорликка эга бўлиб, ҳуқуқсиз мурожаатни бошқариш воситалари чегараланган ва етарли даражада химояланмаган. Шу билан бир қаторда UNIX нинг кўп тизимлари максимал файл ўлчовига 2 Гбайт чегарага эга. (замонавий ОТ лар учун бу жуда кам).

Босмага чиқариш хизмати бу уччала ОТ ларда ўз функционал имкониятлари бўйича етарли даражада яқиндир, ва уларнинг қайси бир яхшироқ эканлигини баҳолаш қийиндир. Фақат шуниси аниқки, Windows NT да босмага чиқаришни бошқариш енгилроқ, UNIX да эса мураккабдир.

Аппарат платформаларни қўллаш.

NetWare 4.1, фақат Intel x86 процессорли компьютерларда ўрнатилиши мумкин. Бир вақтда Novell фирмаси бу ОТ ни Power PC га ўтказмоқчи бўлди, аммо бу фикрдан воз кечишга тўғри келди.

UNIX ОТ ни ҳар қандай платформаларга ўрнатиш мумкиндек туюлади. Аммо бундай эмас, бир платформадан иккинчисига ўтказишда, тизим матнларини жиддий ўзгартиришга тўғри келади. UNIX нинг ҳар бир версияси одатда бир аппарат платформага мўлжаллангандир.

Windows NT Intel x86, DEC Alpha, MIPS ва Power PC платформаларда мавжуддир. UNIX нинг версияларидан бири бўлмиш Solaris SPARC, x86 ва Power PC процессорларида ҳам амалга оширилган. ОТ ларни бошқа платформаларида ишлаши учун мослаштирилган (“адаптированный”) дастурлар зарур.

Коммуникациялар сервери

Уччала системанинг ҳаммаси, масофадаги сервер сифатида ишлатилиши мумкин, аммо бунинг учун оддий компьютер эмас, балки м-н махсуслаштирилган Access Builder ва х.к. серверлардан фойдаланган афзалроқдир.

Маълумотларга гуруҳли ишлов бериш, электрон почта, факс-серверлари платформаси сифатида уччала ОТ ишлатилиши мумкин, аммо UNIX учун бундай дастурлар анча қимматдир ва уларни бошқариш анча мураккабдир.

Web сервер уччала ОТ ларда мавжуддир. Аммо web-сервер сифатида кўпинча UNIX дан фойдаланилади, айниқса Solaris дан фойдаланилади. Аммо Россия ва СНГ давлатларида ҳозирча асосий ўринни Windows NT Server 4.0 (ўзининг Internet Information Server и билан) чунки бу илова тизим билан бирга текинга берилади, бу эса ўз навбатида кучли фактдир.

Аммо барибир ҳозиргача Internet UNIX –серверлар маконидир. Улар, протоколлар ва TCP/IP иловаларнинг максимал сонини қўллайди. Шунинг учун ҳам, агар Internet дан жиддий фойдаланмоқчи бўлсангиз, UNIX дан фойдаланиш мақсадга мувофиқдир.

Локал тармоқларни Internet га улашда муҳим ва асосий ролни - брандмауэрлар ўйнайди. Амалда уларнинг деярли барчаси UNIX нинг у ёки бу версияси асосида амалган оширилган.

Маълумотлар базаси (МБ) сервери. Сиқиб чиқарадиган кўпмасалалилик ва хотира химояси бўлмагани учун NetWare МБ серверига мос платформа эмас. Windows NT ва UNIX- машиналар ўзларини МБ сервери сифатида яхши тавсия этганлар, аммо катта масштаблаштириш хусусияти, кластер технологияга эга эканлиги учун UNIX кучли МБ серверига кўпроқ мосдир.

Амалий дастурлар сервери

Негадир иловалар сервери тўғрисида гап кетса, МБ сервери тушунилади. Тўғри МБ сервери муҳим, аммо бу фақат мумкин бўлган иловалардан биридир. Нима учун иловалар сервери билан МБ серверини чалкаштириш керак.

Биз ОТ ни иловалар сервери дастурий таъминоти учун дастурий платформа сифатида фойдаланишни кўриб ўтамиз, м-н, ҳисоблашлар сервери сифатида.

Кўп фойдаланувчили интерфейс хусусияти қўшиб ишлаб чиқилмагани учун Windows NT бу хизмат жуда яхши тўғри келмайди. NetWare тўғрисида бу ҳақда гапирмаса ҳам бўлади.

Дейлик, жуда қиммат ва катта ресурс талаб қиладиган дастурлар (яъни жиддий, лойиҳалашни автоматлаштирувчи тизимлар FAT учун, ёки ядро реакторида динамик жараёнларни тасвирлаш ва ҳисоблаш дастури), кучли, “Nunber Crashing” (сонларни майдаловчи) категориясидаги ҳисоблаш техникаси керак бўлади. Албатта, унга UNIX ни ўрнатишади, чунки шу ҳолдагина уни кучли ҳисоблаш хусусиятидан бир вақтда, бир нечта фойдаланувчилар фойдалан олишлари керак. Бунинг учун, одатда турли типдаги терминаллар, ишчи станциялар ёки ШК даги терминал эмуляторларидан фойдаланамиз.

Албатта Windows NT ва XWin System ни қўйиб кўриш мумкин. Балки бу арзонроқ ҳам тушади. Фақат NT га зарур дастур топилиши ҳам муаммодир (Alpha процессори учун). Кучли график пакетлар билан ишлаш бу ҳолда қийиндир.

Аммо Windows NT да ўз устунликлари ҳам бор, м-н Network OLE ни қувватлашдир. UNIX учун бундай спецификациялар энди ишлаб чиқилмоқда, ва улар Windows асосидаги мижоз ўринлари билан қандай ишлаши маълум эмас.

Админстрлаш

Админстрлашнинг қулай ва соддалиги, администратор утилиталарининг хусусиятларига эмас, балки ОТ нинг ўзининг имкониятларига ҳам боғлиқдир.

NetWare 4.1 қулай утилиталарининг мажмуасига эга бўлса ҳам, Windows NT Server 4.0 –рақобатдан четдадир. Чунки у яхши ўйлаб чиқилган

интерфейс ва қўшимча, утилиталарнинг бой имкониятлари уни қулай бўлишига олиб келган.

Аммо бу ОТ ларда кўпгина муҳим утилиталар мавжуд эмас (м-н, UNIX даги shell каби команда процессори).

UNIX да аҳвол бутунлай бошқача. Бу ОТ утилиталарининг бой ва катта тўпламига эгаки, улар бир нечта бошқа ОТ ларга ҳам етар эди. Фақат команда процессорларининг ўзигина, минимум 3 тадир. Аммо UNIX дастурлари ҳар хил ташкилотлар ва одамлар томонидан ишлаб чиқилган учун, баъзида бир-бири билан фойдаланувчи интерфейси мос тушмайди. Уларнинг кўпчилиги, ҳозиргача команда қатори билан ишлайди. Бундан ҳам ёмони шундаки, битта утилита, UNIX нинг ҳар хил версияларида аргумент ва опцияларнинг турли мажмуаларига эгадир. Шунинг учун ҳам, Solaris администраторини AIX нинг администраторига ўтказиш учун, уни олдин қайта ўргатиш зарурдир.

Яна шуни айтиш керакки, UNIX нинг ҳолати ҳозир яхши томонга ўзгариб бормоқда. Ҳозир умумий стандартларга мос равишда, бу ОТ нинг кўп версиялари, график интерфейсли яхши ўйланган администратор утилиталари билан таъминланган.

Мижозларни (клиентларни) қўллаш

NetWare ва Windows NT ШК тармоқларига хизмат қилишга мўлжалланган бўлгани учун, улар асосий (клиент) мижоз ОТ лари:

MS-DOS, Win 3x, Windows 95, Windows NT Workstation, OS/2, Macintosh System 75. мижоз сифатида икки тизимнинг баъзи версиялари ҳам қўлланади, аммо ҳаммаси ҳам ҳар доим жойида эмас. Эталон (намуна) сифатида NetWare 4.1 билан “шаффоф” интеграцияга эга бўлган SCO UNIX Ware2.1 хизмат қилади.

UNIX асосида қурилган тармоқларда, шу вақтгача, мижоз ОТ ни қувватлаш жиддий масала эди (UNIX асосидаги мижозларни ҳисобга олмаганда). Аммо ҳозир деярли барча UNIX версиялар. DOS/Win мижозлари билан интеграция воситаларига эгадир. Ундан ташқари, Windows 95 ва Windows NT Workstation энди Stp, telnet, ping, Internet Explorer ва бошқа иловалар билан бирга чиқарилмоқда. Албатта, бу жуда кўп сонни ташкил этмайди, ва шу билан бирга –уларнинг баъзилари имкониятлари, тўғриси айтганда, жуда катта эмас.

Ҳозирги вақтда, ШК ОТ учун дастурлар бозорида, UNIX нинг ҳамма хизматлар спектрини амалга оширувчи, турли фирмалар дастурий маҳсулотлари (нафақат UNIX) мавжуддир.

Novell ҳам турли хизматлар учун иловалар пакетини тақдим этган. Бу пакет иловалари NetWare билан яхши интеграцияланади.

Нарҳи

Албатта биринчи марта қараганда, энг арзон тармоқ OT Windows NT га ўхшайди, ундан кейинги ўринда NetWare, ва уларнинг ичида энг қиммати UNIX бўлиб кўринади (прайс лист маълумотларига кўра ҳам).

Ахвол шунақа, аммо биз шуни таъкидлаб айтамикки харажатлар, сиз қандай хизмат ва иловаларга эга бўлмоқчисиз ва тизимни қандай юритмоқчи эканлигизга боғлиқдир. Тежамкорлик ҳар доим ҳар доим яхши натижага олиб келмайди.

Тармоқ OT ни, молиявий томонини эсдан чиқармасдан туриб, функционал имкониятларидан келиб чиққан ҳолда танлаш керак. Агар, сиз Windows NT асосида гетероген тармоқ қурмоқчи бўлсангиз, ҳар хил фирмалардан талай турли дастурларни сотиб олиш керак (баъзи вақтда улар бир-бири билан яхши келиша олмайди). Бу эса ўз навбатида қимматга тушади (Windows NT прайс-листида кўрсатилганга нисбатан).

Бу ҳолда, NetWare ҳолати яхшироқ, чунки турли таркибли тизимлар интеграцияси учун дастурларнинг тўлиқ спектрини ишлаб чиқади.

UNIX мижозлар ҳақида. Ишчи ўринларда Windows машина жиддий учтунликларини ҳеч ким тан олмайди

Афсуски, дунёда мукамаллик йўқдир. Чунки ҳеч қайси тармоқ OT и корпоратив гетероген тармоқни яратишда қўйиладиган талабларни барчасини қондира олмайди: уларнинг ҳар бири ўз устунлиги ва камчилигига эгадир.

Гетероген муҳитда, ШК лардан ташқари UNIX машиналар бор бўлган муҳитда, энг яхши ёндошиш – тармоқ OT ларидан биргаликда фойдаланишдан иборатдир (UNIX +Windows NT Server 4.0 ёки UNIX + NetWare 4.1). аммо бундай муҳитда NT алоҳида устунликка эга бўлмайди, чунки иловалар сервери ва NFS серверлар сифатида UNIX дан фойдаланиш мақсадга мувофиқдир. ШК ларга хизмат кўрсатишда эса, NetWare 4.1 нинг энг яхши хизмат каталогидан фойдаланиш афзал.

Аммо, UNIX машина йўқ жойда, Windows NT Server дан фойдаланиш анчайин устунлигига эга.

13-март. Замонавий операцион тизимларлар қисқача обзори. UNIX операцион тизимларлари оиласи. Асосий тушунчалар. Архитектура хусусиятлари ва умумий характеристикаси. Фойдаланувчилар уч даражали иерархияси. Фойдаланувчи интерфейси. Жараёнлар бажарилиши. Файл тизими. Windows NT тармок операцион тизимлар ва бошқа тармок операцион тизимларлари билан қиёсий характеристикаси.

Энди ОТ ларга тегишли асосий тушунчалар билан танишганимиздан, ҳисоблаш жараёнларини ташкил этиш аниқ механизмларини ўрганганимиздан сўнг, қисқача замонавий ОТ лар хусусиятларини кўриб чиқамиз.

Шуни таъкидлаш лозимки, энг замонавий бўлиб, ҳозирги вақтда MS компаниясининг Windows оиласига мансуб ОТ лар ҳисобланади. Бу Windows 95/98/ME, Windows NT/2000 ва Windows XP/2003 янги авлодидир.

Ҳозир эса, биз, MS маҳсулотига мансуб бўлмаган ОТ лар, UNIX га ўхшаш Linux ва Free BSD, QNX ва OS/2 ларни кўриб чиқамиз. Биз, UNIX оиласи ОТ ларининг архитектура хусусиятларини кўриб чиқамиз. QNX тизими эса жуда таниқли ва энг яхши, реал вақт тизимларига мансубдир. OS/2 ОТи эса, амалда ишлатилмаяпган бўлса ҳам, у энг биринчи, бир нечта операцион муҳитни қўлловчи тўлақонли ва ишончли мультидастурли ва мультимасалали ОТ дир.

UNIX ОТ лар оиласи.

UNIX, энг яхши амалга оширилган, мультидастурли ва кшпфойдаланувчили оддий ОТ дир. Ўз вақтида, у, дастурий таъминотни ишлаб чиқувчиинструментал тизим сифатида лойихалаштирилган эди. UNIX нинг ҳар хил хусусиятларга эга бўлган версиялари, унинг қийматини оширади. Биринчи версияси, жуда кичик оператив хотирага эга бўлган, компьютерларда фойдаланиш мумкин бўлган бўлса, унинг иккинчи версиясини ишлаб чиқишда, мутахассислар, ассемблер тизимидан воз кечиб, нафақат тизимли, балки амалий дастурларни ҳам ишлаб чиқиш мумкин бўлган, юқори даражали тилни (стили) ишлаб чиқдилар. Шунинг учун ҳам нафақат UNIX тизимли, балки унда бажарилувчи иловалар ҳам енгил кўчириб ўтказилувчи (мобил) хусусиятига эга бўлди. С тилидан ўтказувчи компилятор, ҳамма дастурларга, тизимдаги ресурслардан самарали фойдаланиш имконини берадиган кодни беради.

Архитектура умумий характеристикаси ва хусусиятлари.

Бу тизимни ишлаб чиқишда биринчи асосий мақсад соддаликка эришиш ва функцияларни минимал миқдори билан иш олиб бориш эди. Ҳамма реал муаммолар, фойдаланувчи дастурларга қолдирилди.

Иккинчи мақсад эса – умумийликдир бу дегани кўп ҳолларда, бир хил усул ва механизмлардан фойдаланишлик хусусияти, м-н:

- файлларга, киритиш-чиқариш қурилмалари ва жараёнлар аро маълумотлар буферига мурожаатлар, бир хил примитивлар ёрдамида бажарилади;
- номловчи, альтернатив ном берувчи ва ҳуқуқсиз (несанкционированный доступ) мурожаат механизмлари, ҳам маълумотлар файлларига, ҳам каталог ва қурилмаларга ҳам ишлатилади;
- бир хил механизмлар, ҳам дастурли, ҳам интегралланувчи узилишларга нисбатан ҳам ишлатилади.

Учинчи мақсад эса, мураккаб масалаларни ечишда, мавжуд кичик дастурлардан биргаликда фойдаланиб, яъни уларни янгидан ишлаб чиқмасдан ечиш имкониятини яратиш эди.

Ва ниҳоят 4-чи мақсад эса, нафақат процессор вақти, балки қолган бошқа ресурсларни ҳам тақсимловчи, самарали механизми-мультитерминал ОТ ни яратишдан иборат эди. Мультитерминал ОТларда, биринчи ўринда, ҳисоблаш жараёнларини бошқа ҳисоблаш жараёнлари аралашувидан ҳимоя масалалари туради.

UNIX ОТ и, файл тизимига боғлиқ бўлмаган кучли ва содда команда тилига эгадир. Бундай имкониятни амалга оширишнинг энг муҳим хусусияти, шундан иборатки, бир дастур натижаси, иккинчи дастур учун бошланғич маълумот бўлиши мумкин. Бу дегани, катта дастур композициялари, мавжуд кичик дастурлар ёрдамида яратилиши мумкинлигидир. Бунда янги дастур яратишга хожат йўқлигидир.

UNIX – тизими тизимли ва амалий дастурларига, матн редакторлари, команда тилининг дастурловчи интерпретаторлари, бир нечта оммавий дастурлаш тиллари компилятори (C, C++, ассемблер, PERL, FORTRAN ва ҳ.к.лар), компановкачилар (дастурлараро алоқа редакторлари), созловчилар (отладчики), кўпсонли тизимли ва фойдаланувчи дастурлари кутубхонаси, маълумотлар базасини юритиш ва ажратиш воситалари, кўп сонли администрловчи ва хизмат қилувчи дастурлар киради. Бу дастурларнинг анчайин қисми учун ҳужжатлар мавжуд бўлиб, дастур матнлари яхши изохлангандир. Бундан ташқари, ҳужжат ва тавчифлардан фойдаланувчи интерактив режимда фойдалана олиш мумкин.

Тўлиқ ҳимояга эга бўлган файл тизимидан фойдаланилади, қурилмалар мустақиллиги таъминланади.

UNIX тизими марказий қисми-ядродир (kernel). У кўп сонли модуллардан иборат бўлиб, архитектура жиҳатдан монолит ҳисобланади. Аммо, ядрога ҳар доим 3 та тизимни ажратиш мумкин: жараёнларни бошқариш; файлларни бошқариш; марказий қисм ва приферик қурилмалар ўртасида киритиш ва чиқариш амалларини бошқариш.

Жараёнларни бошқариш тизими жараёнларни диспетчерлаш, ва бажаришни, уларни синхронлаштиришни ва ҳар турли жараёнлар аро алоқани ташкил этади. Жараёнларни бошқариш асосий функцияси – бу оператив хотирани бошқариш ва (замонавий тизимлар учун) виртуал хотирани ташкил этишдир.

Файлларни бошқариш тизими, жараёнларни бошқариш тизими билан ҳам, драйверлар билан ҳам қатъий боғлангандир. Компьютер таркибидаги курилмаларга ва ечилаяпган масалаларга қараб, ядро қайта компиляция қилиниши мумкин. Ҳамма драйверлар ҳам ядро таркибида бўлмаслиги мумкин, бир қисми ядродан чақирилиши мумкин. Бундан ташқари, тизимли функцияларининг жуда кўпчилиги, ядрога кирмайдиган, аммо ядродан чақириладиган тизимли дастур модуллари ёрдамида бажарилади.

Ядро, бошқа тизимли модуллар билан бажариши керак бўлган функциялар қатъий равишда стандартлаштирилган.

Юқорида айтилганлар ҳисобига, UNIX нинг турли версиялари ва турли аппарат таъминоти ўртасидаги кодларни кўчириб ўтказувчанликка эришилади.

Асосий тушунчалар

UNIX ОТ ни асосий устунлиги шундан иборатки, бу тизим ҳам сонли тушунчаларга асосланади.

Виртуал машина. UNIX тизими кўп фойдаланувчилidir. Ҳар бир фойдаланувчига, у рўйхатдан ўтгандан сўнг, виртуал компьютер берилади, бу виртуал компьютер ҳамма зарурий ресурсларга эга: процессор (процессор вақти, доира ёки коруселли, дипетчерлаш ва динамик приоритетлардан фойдаланиб ажратилади), оператив хотира, курилма, файллар. Бундай виртуал компьютер жорий ҳолати “образ” деб аталади. Жараён-образли бажарилиши дейиш мумкин.

Жараён образи қуйидагилардан ташкил топади:

- образ хотираси
- процессор умумий регистрлари қиймати
- файл жорий каталоги
- бошқа маълумотлар

Фойдаланувчи

Юқорида айтиб ўтганимиздек, UNIX ОТ и кўп фойдаланувчили интерактив тизим сифатида назарда тутилган эди. Бошқача айтганда, UNIX мультитерминалли ишлаш учун мўлжалланган. Иш бошлаш учун, фойдаланувчи ўз кириш пароли ва номини (name, login) киритади, бу ҳол ҳисобга олинган фойдаланувчи учун ўринlidir.

Янги фойдаланувчини одатда администратор ҳисобга олади. Фойдаланувчи ўз ҳисоб номини ўзгартира олмайди, аммо ўз паролини ўрнатиши ёки ўзгартириш мумкин. Пароллар алоҳида файлда кодланган ҳолда сақланади.

UNIX ОТ ядроси ҳар бир фойдаланувчини унинг идентификатори (user Identifier, UID) орқали идентификация қилади. (тизимда ҳисобга олинadиган ягона бутун қиймат орқали). Бундан ташқари, ҳар бир фойдаланувчи маълум гуруҳга мансубдир, бу ҳол ҳам маълум бутун қиймат орқали идентификация қилинади. (Group Identifier, GID). UID ва GID қиймати ҳар бир ҳисобга олинган фойдаланувчи учун, тизим ҳисоб файлларида сақланади ва

фойдаланувчи тизимга кирганда ишга команда интерпретатори бажариладиган жараёнга ёзилади. Ва бу қиймат, шу фойдаланувчи номидан ишга тушириладиган ҳар бир жараёнга меърос бўлиб ўтади ва ядро томонидан файлларга мурожаат, дастурлар бажарилиши ҳуқуқини назорат қилиш учун фойдаланилади.

UNIX ОТ и файл тизими дарахт структурасига эга. Ҳар бир ҳисобга олинган фойдаланувчи учун, файл тизими маълум каталоги мос келади (бу “домашний” каталог дейилади). Тизимга киришда, фойдаланувчи ўз каталогидаги барча каталог ва файллардан чексиз фойдаланиш ҳуқуқига эга бўлади. Фойдаланувчи ўз каталогидаги каталог ва файлларни яратиши, олиб ташлаши ва ўзгартириши мумкин. Бошқа файлларга мурожаат ҳуқуқига қараб, чегараланган бўлади.

Супер фойдаланувчи.

Албатта, ҳисобша олинган фойдаланувчи сифатида иш тутадиган тизим администратори, бутун тизимни бошқариш учун оддий фойдаланувчига нисбатан кўпроқ ҳуқуқга эгадир. UNIX ОТ да бу ҳол администратор UID га ягона нол қийматини бериш билан ечилади, бу фойдаланувчи демак суперфойдаланувчидир (яъни администратор).

Бу фойдаланувчи тизим устидан тўлиқ назоратга эгадир. Суперфойдаланувчи учун ресурслардан фойдаланишга чегара йўқдир. Оддий фойдаланувчиларга файл ўлчами, бўлинадиган хотира максимал хажм ва х.к.ларга чегара қўйилган бўлади. Администратор бу чегараларни, бошқа фойдаланувчилар ўзгартириши мумкин. Администратор ҳам ўзи учун оддий ҳисоб ёзуви ташкил этади, аммо тизимни администрлаш учун “SU” командасидан фойдаланади (қайтиш командаси эса “exit”).

Фойдаланувчи интерфейси. Фойдаланувчининг UNIX тизими билан мулоқоти команда тилига асослангандир. Фойдаланувчи тизимга киргандан сўнг, унинг учун команда интерпретаторларидан бири ишга тушади. Одатда, тизимда бир-бирига ўхшаш, аммо ўз имкониятлари билан фарқ қиладиган тилли команда интерпретаторлари қўлланилади. Ихтиёрий команда интерпретатори UNIX ОТ и учун – қобик (“оболочка” – spell дир). Чунки ҳар қандай интерпретатор тизим ядроси ташқи муҳитидан иборатдир Linux тизимида команда интерпретатори – “bash” дир.

Чақирилган команда интерпретатори, фойдаланувчига команда қатори киритишга таклиф беради (ёки командалар қатори ва х.к.). Навбатдаги команда бажарилганд сўнг, терминал экранига натижа чиққандан сўнг, интерпретатор яна команда қаторини киритишга таклиф беради, шундай қилиб тизимдан чиқмагунча шу ҳол давом этади.

UNIX да фойдаланувчи команда тиллари етарли даражада соддадир ва шу билан бирга мураккаб дастурлар ёзиш учун етарли даражада кучлидир.

Ҳозирги вақтда кўпроқ график интерфейслардан фойдаланилгани учун, UNIX ОТ ларида ҳам кўпинча X-Window да ишлашмоқда. X-Window – бу фойдаланувчиларга ўз тизимлари ва масалалари билан график режимда мулоқот имконини беради.

Команда катори, команда номи (бажариладиган файл номи), унинг кетидан аргументлар рўйхатидан иборатдир. Ҳар қандай команда тили уч қисмдан иборатдир:

- хизматчи конструкциялар; улар матн катори билан амаллар бажариш ва оддий командалар асосида мураккаб командалар яратишга имкон беради.
- қўйилган командалар (встроенный) команда тили томнидан бевосита бажариладиган команда
- алоҳида бажариладиган файллар билан ифодаланадиган командалар.

Охирги кўринишдаги командалар ўз ичига стандарт команда (тизимли утилиталар) ва фойдаланувчи яратган командалардан иборатдир.

Жараёнлар. UNIX тизимида жараён – бу мумтоз маънода шахсий виртуал маконда бажариладиган жараёндир. Фойдаланувчи тизимга кирганда, команда интерпретатори дастури бажариладиган жараён автоматик тарзда яратилади. Агар команда интерпретаторига, бажарилаяпган файлга мос команда учраса, у янги жараён яратади ва унда “main” функциясидан бошлаб мос дастурни ишга туширади. Бу ишга туширилган дастур ўз навбатида жараён яратиши ва унда бошқа дастурни ишга туширади ва х.к. Янги жараёни ташкил этиш ва унда дастурни ишга тушириш учун иккита тизимли чақириш API – fork () ва exec (имя выполняемого файла – бажарилувчи файл номи). fork () – тизимли чақириш, ҳолати асосий жараён ҳолатига ўхшаш бўлган янги адрес маконини яратишга олиб келади. (яъни унда ўша дастур ва маълумотлар мавжуддир). Янги жараён учун, ҳамма маълумотлар сегментик нусхалари ташкил этилади.

UNIX ОТ и ишлаши. Жараёнларни бажарилиши.

Жараёнлар икки ҳолатдан биттасида бажарилиши мумкин: фойдаланувчи ва тизимли фойдаланувчи ҳолатида, жараён фойдаланувчи дастурини бажаради ва фойдаланувчи маълумотлар сегментидан фойдаланиши мумкин. Тизимли ҳолатда жараён ядро дастурини бажаради ва тизимли маълумотлар сегментидан фойдаланиши мумкин.

Фойдаланувчи жараёнига тизимли функцияни бажариш талаб этилса, у тизимли чақириқ қилади. Амалда, тизим адросини дастур сифатида чақириғи рўй беради. Шу вақтдан бошлаб, тизимли чақириқдан бошлаб, жараён тизимли жараён ҳисобланади. Шундай қилиб, фойдаланувчи ва тизимли жараёнлар, бир жараённинг икки фазосидир, аммо улар бир-бири билан ҳеч қачон кесишмайдилар. Ҳар бир фазо ўз стекидан фойдаланади.

Масала стеки, аргумент, локал ўзгарувчилар ва масала режимида бажариладиган функциялар маълумотларни ўз ичига олади. Диспетчер жараёни фойдаланувчи фазасига эга эмас.

UNIX тизимларида вақт ажратилиши ташкил этилади, яъни ҳар бир жараёнга вақт кванти ажратилади, ёки квант тугаши билан у тўхтади ва вақт янги кванти берилиши билан, у ўзининг бажарилишини давом эттиради.

Диспетчерлаш механизми, ҳамма жараёнлар орасида процессор вақтини тақсимлайди. Фойдаланувчи жараёнларига приоритет, у олган процессор вақтига қараб берилади. Ҳамма тизимли жараёнлар, фойдаланувчи

жараёнларига нисбатан юқори приоритетга эгадир ва шунинг учун ҳам биринчи навбатда уларга хизмат қилинади.

Киритиш ва чиқариш тизими. UNIX да киритиш ва чиқариш функциялари асосан 5 та тизимли чақириш билан амалга оширилади: open, close, read, write ва seek.

Файл тизими. UNIX тизимида файл ихтиёрий мурожаатли символлар мажмуасидан иборатдир. Файлга маълумотлар, фойдаланувчи томонидан жойлаштирилиши мумкин, ва у бошқа структурага эга бўлмайди.

Файл тизими структураси.

Жараёнлар орасидаги ўзаро алоқа. UNIX ОТ и клиент-сервер технологиясига тўлиқ жавоб беради. Бу универсал модел ихтиёрий мураккабликдаги, шу билан бир қаторда тармоқ тизимларини куриш учун асос бўлиб хизмат қилади. Клиент-сервер модели принципида ишлайдиган дастур тизимларини куриш учун UNIX да қуйидаги механизмлар мавжуд:

- сигналлар
- семафорлар
- дастурли каналлар
- хабарлар (сообщении) навбати
- хотирани бўлинадиган сегментлари
- масофадаги процедураларни чақириш

Сигналлар.

Виртуал компьютерда жараён бажарилишини кўриб чиқсак, (фойдаланувчига бериладиган) у ҳолда бундай тизимда, стандарт талабларга жавоб берадиган узилишлар тизими мавжуд бўлиши керак:

- фавқулотда ҳолатларга ишлов бериш
- ички ва ташқи узилишиларга ишлов бериш воситалари
- узилишлар тизимини бошқариш воситалари

Бу ҳамма талабларга UNIX да сигналлар механизми жавоб беради, у нафақат сигналларни қабул қилиб ишлов беришга имкон беради, балки уларни юзага келтириб бошқа жараёнларга (машиналарга) жўнатишга имкон беради. Сигналлар синхрон ва асинхрон бўлиши мумкин.

Сигналлар, жараёнлар орасидаги ўзаро алоқани оддий формаси сифатида қараши мумкин. Улар бир жараёндан иккинчисига ёки ОТ ядросидан бирор жараёнга, маълум ҳодиса юзага келганлиги ҳақида хабар бериши учун фойдаланилади.

Дастур каналлари UNIX тизимида ўзаро алоқа ва жараёнларни синхронлаштирувчи муҳим воситадир.

Жараёнлар орасида маълумотлар алмашинуви учун, хабар навбати механизми фойдаланилади.

14-маъруза. Linux операцион тизимларлари оиласи. Асосий тушунчалар. Архитектура хусусиятлари ва умумий характеристикаси.

Linux бу замонавий UNIX га ўхшаш, POSIX стандартини қондирувчи шахсий компьютерлар ва ишчи станциялар учун яратилган ОТ дир. Linux бу эркин тарқатиладиган UNIX – тизими версиясидир. Бу тизимни Линус Торвалд ишлаб чиқган бўлиб, у кодларни очиқ қилиб яратиш шартини таклиф қилди. Ихтиёрий фойдаланувчи коддай фойдаланиши ва ўзгартириши мумкин, аммо бу ҳолда албатта у тизимнинг модулларига киритган кодини очиқ қолдириши шарт. Тизимнинг ҳамма компоненталари (хатто берилган матнлар ҳам) эркин нусха олиш ва чегараланмаган сонли фойдаланувчиларга ўрнатиш шarti билан, лицензияли тарқатилади.

Шундай қилиб, Linux тизими кўп сонли дастурчилар ва интернет орқали бир-бири билан мулоқот қилувчи UNIX тизими фидойилари ёрдамида яратилди.

Бошида Linux тизими, “қўлбола” UNIX га ўхшаш тизими сифатида i80 386 процессорли IBM PC типдаги машиналарга мўлжалланган эди. Аммо кейинчалик Linux – шу даражада оммавийлашиб кетдики, уларни шу даражада кўп команиялар қўллаб-қувватладики, ҳозирги вақтда бу операцион тизимнинг амалдаги версиялари деярли ҳамма типдаги процессор ва компьютерлар учун ишлаб чиқилди. Linux асосида суперкомпьютерлар ҳам яратилаёпти. Тизим кластерлаштириш, замонавий интерфейс ва технологияларни қўллайди.

Linux – кўп масалали, кўп фойдаланувчили тўла қонли операцион тизимдир (худди UNIX бошқа версиялари каби). Бу, бир вақтнинг ўзида, битта машинада, кўп фойдаланувчилар, параллел ҳолда, кўпгина дастурларни бажарган ҳолда ишлаши деган сўздир.

Linux тизими. UNIX учун қатор стандартлар билан берилган матнлар даражасида мутаносибдир (совместим). UNIX учун интернет орқали эркин тарқатиладиган датсурлар, Linux учун, амалда кам ўзгартиришларсиз компиляция қилиниши мумкин. Бундан ташқари, Linux учун ҳамма берилган матнлар, яъни ядро, қурилмалар драйверлари, кутубхоналар, фойдаланувчи дастурлари ва инструментал воситалар эркин тарқатилади.

Linux, маълумотларни сақлаш учун турли типдаги файл тизимларини қўллайди. EXT2FS каби файл тизими Linux учун махсус яратилган. М-н, Minix-1 ва Xinix каби файл тизимлари ҳам қўлланилади. Бундан ташқари, FAT асосидаги файлни бошқариш тизими амалга оширилган, бу эса бу файл тизими бўлимларидаги файлларга бевосита мурожаатга имкон беради. HPFS, NTFS ва FAT32 ларга мурожаат ва файлларни бошқариш тизими вариантлари яратилган.

Анъанавий UNIX тизимларидаги каби, Linux бизга маълум 3 та тизимни ўз ичига олган микроядрога эга.

Free BSD ОТ и. Linux ОТ дан ташқари эркин тарзда тарқатиладиган операцион тизимлар оиласига кирувчи Free BSD ни ҳам айтиш мумкин. Бу

ОТ лар орасидаги принципиал ва энг муҳим фарқ шундаки, келишувга кўра, Linux тизимига ҳар ким ўз ўзгартиришларини киритиши мумкин, аммо бу ҳолда у ўзини кодини очик ҳолда колдириши керак. Аммо ҳамма компаниялар бунга рози эмас. Кўпчилик, берилган матнлар ва тайёр ечимлардан фойдаланишни хоҳлайдилар, аммо ўз дастурий таъминот сирларини очиклари келтирмайди. Шунинг учун ҳам, бу ОТ учун дистрибутивлар ишлаб чиқувчи компаниялар мавжуд. Ҳар бир компания ўз ОТ дан ташқари унга ўз инсталляторини, утилиталарни, шу билан бирга дастурлар пакетини, конфигураторларни ва ниҳоят амалий дастурлар пакетининг катта тўпламини қўшади. Бунда у, ўз тизимига ўз ўзгаришларини бошқалар билан келишмасдан киритиши мумкин.

Linux га қарама-қарши равишда, Free BSD ОТ и ўз координаторига эгадир, бу калифорния Беркли университетидир. Хоҳлаган одам бу ОТ код матнларини ўрганиши ва унга ўз ўзгартиришларини киритишни таклиф этиши мумкин, аммо бу ўзгаришлар киритиладиган сўз эмас, ҳатто ўзгаришлар фойдали бўлса ҳам. Бунга фақат координатор ҳукуқи бор. Шундай қилиб, Free BSD – бу UNIX га ўхшаш ОТ, у ҳам очик кодли, унинг ядроси микроядро принципида қурилган.