Дәріс 14.

Интернет қосымшаларын құру технологиясы.

Серверге программалық құралдары.

«Клиент-сервер» технологиясындағы программалау.

ISAPI, CGI спецификациясы

Серверді программалау тілдері.

Сценарилар көмегімен серверде сақталған мәліметтер базасына және басқа да ресурстарға файлға кіруге, рұқсат алуға болады, және де сервердің орталықтандырылған ресурстарына, электрондық почта, факс қызметі сияқты.

Серверде орындалатын сценарилердің тағы бір артықшылығы болып, қарсылық тудырмайтын және басқарылатын ортаның функциялануы сіздің кодыңыз көп нұсқалы браузер емес, ал бір нұсқалы серверде орындалады.

Бірақ сервер жағындағы сценариларды қолдану кезінде 3 негізгі кедергі бар:

-Сервердегі скрипті енгізу web мастерден жүйелік администратордан арнайы рұқсат алуды талап етеді;

-Серверде орындалатын сценарилармен өзара қарым қатынастар болу үшін қолданушы тышқанмен ссылкіде немесе беттегі кнопканы шерту керек, содан кейін сценариді орындағанын және жауап қайтаруын күтеді. Динамикалық HTML мен қарым – қатынасты қоданумен тез орындалады;

-Сценариді тестілеу үшін серверге өз арнайы www сервері болуы керек, типі өндірістік нұсқа түрінде болуы жөн;

Кең спектрлі тапсырманы шешу үшін қазіргі уақытта ең негізгі шарт болып сервер жағындағы пограммалау болып табылады. Ол рұқсат етеді:

-Пішін көмегімен қолданушымен енгізілген сервердегі мәліметтерді алу және өңдеу;

-Платформаға, браузер клиентке тәуелсіз динамикалық web документтер құру;

-Мәліметтер базасының серверінің, яғни серверде орналасқан мәліметтерге динамикалық доступпен қамтамасыз ету (Мұндай доступ тәсілінде серверде сақталған мәліметтер өзгерген кезде HTML документі автоматты түрде өзгереді);

-Типтік тапсырмаларды орындауға арналған сервердің компоненттерді қолдану (Мысалғы, мынадай жарнамалық баннерлердің циклдік ауысуы және т.б.)

-Қолданушының аутентификациясын жүзеге асыру;

-Клиенттің браузері жөнінде мәліметтер алу;

-Клиенттің жағынан кілттерді құрастыру және оқу;

«Клиент-сервер» технологиясы

Көптеген Word wide web қосымшалары web серверімен басқарылатын ішкі программаларды қолдануға негізделген. Берілген программаларды қолдану динамикалық жаңару мәліметтерімен web қосымшасын құруға мүмкіндік береді. Мәліметтер базасында сақталған немесе шешілетін тапсырмалардың бизнес ережелеріне бағынышты генерилданатын. Web серверімен және шақырылатын программалар арасындағы байланыс үшін CGI кеңінен қолданылады, windows программасы үшін сондай-ақ Unix ортасындағы функционалданатын қосымшалардың нақтылығы бар. CGI, HTTP, Web сервер типтерінің ақпараттық серверімен ішкі қолданбалы программасының интерфейстің стандарты болып табылады. Әдетте, www серверінен гипертексті документтер статикалық мәліметтерден тұрады.

CGI-дің көмегімен CGI программаларын құруға болады, олар **шлюз** деп аталып қолданбалы жүйелер, мәліметтер базасын басқару жүйесі, электрондық кесте жұмыстық графика тағы да басқалардың қарым – қатынасы экранға қолданушының динамикалық ақпаратын шығара алады.

Шлюз программасы www серверінің нақты масштаб уақытында іске асырылады. www сервер қолданушының шлюз тапсырмасын қамтамасыз етеді, ал ол өз кезегінде қолданбалы жүйелік ортаны қолдана отырып, қолданушының экранына өңделген сұраныс нәтижесін қайтарады. Программа-шлюз C/C++, Fortran, Perl, TCL, Unix Schell, Visual Basic, Apple Script. тілдерінде кодталған болуы мүмкін. Ол әдетте, директорияда cgi-bin www сервер атымен орындалатын модуль сияқты болып жазылады. Бүкіл интернет және www-дің кейбір бөліктері «клиент-сервер» технологиясы бойынша жұмыс істейді, яғни барлық программалық қамтамасыздандыру клиенттік және серверлік бөліктерге бөлінеді. Сондай – ақ олардың арасында функцияналдық міндеттер ажыратылған.

Клиент жұмыстың принципінсервердің ішкі алгоритмдік іске асырылуын білмейді, білуі тиіс де емес, ал сервер клиенттің жұмыстарына араласпайды. Бұл бөліктердің ара қатынасы үшін арнайы протокол шығарылған (бұл жағдайда HTTP протоколы), және де сервермен клиент арасындағы қарым – қатынасты осы протокол ішінде іске асырылады. Браузерге бәрібір серверде қандай программалық қамтамасыздандыру тұрғаны, онда қандай операциялық жүйенің болуы, серверде сұраныс документтері қай жерде жатқаны (жалпы жатыр ма, олар арнайы программалармен лезде генералдануы мүмкін). Сервер сіздің браузеріңіздің жұмысына араласпайды, клиент берілген ақпаратпен не істейді, қалай көрсетеді, дискі де сақтайды ма, әлде сақтамайды ма серверге бәрібір. Әдетте, серверге сұраныстан деп түсіндіріледі (бұл сервердегі мәліметтердің унифицилданған тапсырыстың пішіні).

Сервермен байланысты орнату кезінде алдымен www IP адресіндегі www.siemens.com секілді символдық аттардың трансляциясы болады, ал

кейін TCP/IP - IP мәліметтерімен қосылуы іске асырылады. Серверге HTTP мәліметтеріне сұранысы жіберілген кезде клиент жауап келгенін күтеді. Клиенттен сұраныс болғанша HTTP сервері сұранысты күтіп, «ұйықтап жатады». Клиент байланыс орнатқан кезде сервер «оянады» және бірілген сұранысты қабылдап, оларды өңдеуге кіріседі. Сервердің сұранысымен не істейтіні-сервердің өзіне ғана белгілі. Барлық монупуляциялардың жалғыз нәтижесі-ол клиенттің күтіп отырған жауабы болып табылады. Сервер жауап қайтарғаннан кейін ол байланысты үзіп қайтадан «ұйқыға түседі» және қателік шыққан кезде HTTP транзакциясы әр аптада аяқталуы мүмкін. Егер документтер табылмаған жағдайда немесе сізде оларға құқығыңыз болмаса (ресурсқа кіру шектеуші болуы мүмкін) немесе қателік коды беріледі. Ал егер барлығы дұрыс болса, документте сақталған ақпарат, типі жөніндегі мәліметтерді қоса отырып, жауап ретінде қайтарылып отырылады.

ISAPI

CGI секілді ISAPI спецификациясы web сервермен және қосымша программалармен арасындағы қарым қатынас ережелерін анықтайды. Серверде орындалатын программаны құру үшін альтернативтік тілтабысу неге керек екенін түсіну үшін, CGI стандартының артықшылықтары мен кемшіліктерін атап өтейік. CGI-дің сөзсіз артықшылығы болып оның универсалдылығы болып табылады. CGI әр тілде жазылуы және компьютерде әр түрлі архитектурада орындалуы мүмкін.

Егер сценариді жазған кезде барлық ережелерді ескерген болсаңыз, ол кезде сіздің құрған программаңыз кез-келген web серверімен дұрыс қарым қатынаста болатынына сенімді болуыңызға болады. CGF-дің қарапайымдылығы бұл стандарттың кең таралуына мүмкіндік береді. Бірақ бұл жетістіктермен қатар CGI сценариясында кейбір кемшіліктер де кездеседі.Олардың ең бастысы болып ресурстардың тиімсіз қолданылуы болып табылады. Сценариді шақыруды ұсынатын клиенттің әр сұранысы компьютер-серверде бөлек процесс туындатады. Процестің құрамындағы жұмысты орындауға қарағанда тәуелсіз процесті орындау көбірекресурсты талап етеді.

Ресурстың өндірістігін жоғарлату және шығынды азайтуға ұмтылу корпоративтік шығуына әкеліп соқты. Ең атақтысы NSAPI және ISAPI болып табылады. NSAPI программасының CGI сценарисынан басты айырмашылығы болып, ISAPI программасының орындалмайтынын файлды емес динамикалық кітапхана (DDL) болып табылады. Осының арқасында программаны жеке процесс ретінде емес, ал web сервердің бір тұтас ағыны ретінде программаны іске қосу мүмкіндігі туды. Тәуелсіз поцесс жұмысын орындауға өарағанда тұтас ағын жұмысын орындауға аз ресурс жұмсалады. Тұтас ағын оның процесін туындатқан мекенжайлық кеңістікті қолданады және жеке процеске қарағанда тезірек жұмыс істейді. Процестің аяқталу кезінде жадыдан шығады, ал келесі сұраныс түскен кезде дискіге қайта жүктеледі. Бұл CGI-дің тиімді жұмыс істеуіне кері әсерін тигізеді.

Сұраныстың қарқынды түсуі кезінде дискіні әрдайым ауыстыру қажеттілігі серверде қосымша жүк пайда болады.

CGI-сценариасына қарағанда ISAPI программасы жұмысын аяқтағаннан кейін жадыда резиденттік болып қалады және келесі сұраныстарды өңдеуде қолданылады. Бірақта ISAPI программасының артықшылығы кейде кемшіліктерге айналып кетеді. Егер CGI-сценариасының универсалдылығы үшін жеткіліксіз өндірістілікпен төлеу керек болса, ол кезде жоғары өндірістілік үшін ISAPI программасының негізін салушылар универсалдылығымен төлейді. ISAPI программасының негізін қалаушылардың тілдер арсеналы CGI-сценараисының негізін қалаушыларға қарағанда аз болып келеді. ISAPI программасында неғұрлым көп жазғысы келетін адамға C++-тан басқа амал жоқ. ISAPI программасының екінші кемшілігі біріншісіне қараңанда әлде қайда күрделірек. Берілген типтегі программа серверде туындалған тұтас ағын ретінде орындалады, ол сервердің мекен жайлық кеңістігін қоланады. Соған байланысты ISAPI программасының ұателігі авариалық аяқталуды ғана емес, сонымен қатар серверді істен шығаруы мүмкін. ISAPI спецификациясын қолданатын программалар екі категорияларға бөлінеді:

- Кеңейтілу. Функционалды түрде CGI-сценариасынан олардың айырмашылығы аз. CGI-сценариасы сияқты кеңейтілу web серверінде қосымша мүмкіндіктерді іске асырады, бірақ мүлде өзгеше құрылады. ISAPI кеңейтілуіне осы бөлім арналған;

- Фильтірлер. Бұл программаның өзгеше класы. Кеңейтілу web сервердің мүмкіндіктерін көбейтетін болса, фильтірлер оның жұмыс тәртібін қзгертеді. ISAPI кеңейтілуі оның жұмысы кезінде web серверімен байланысқан динамикалық кітапхананы (DDL) танытады.

ISAPI кеңейтілуі орындау кезінде негізі болып және берілген типтегі программаның CGI-сценариасынан айырмашылығын ажыратудың оңай тәсілі – кеңейтілудің шақыру процедурасын қарастыру және де жоғарыда көрсетілген функциялар мен параметрлердің міндеті боып табылады. Клиенттің сұранысы нақты URL программасын шақырған кезде ғана болатын кеңейтілуге қарағанда, ISAPI фильтірлері клиенттің серверге әр сұраныс жіберу кезінде басқара алады. web серверді іске қосқанда программалық-фильтір жадыда жүктеледі және сервердің жұмыс уақыты аяқталғанға дейін резидентті боып қалады.

Сценарилар көмегімен серверде сақталған мәліметтер базасына және басқа да ресурстарға файлға кіруге, рұқсат алуға болады, және де сервердің орталықтандырылған ресурстарына, электрондық почта, факс қызметі сияқты.

Серверде орындалатын сценарилердің тағы бір артықшылығы болып, қарсылық тудырмайтын және басқарылатын ортаның функциялануы сіздің кодыңыз көп нұсқалы браузер емес, ал бір нұсқалы серверде орындалады.

Бірақ сервер жағындағы сценариларды қолдану кезінде 3 негізгі кедергі бар:

-Сервердегі скрипті енгізу web мастерден жүйелік администратордан арнайы рұқсат алуды талап етеді;

-Серверде орындалатын сценарилармен өзара қарым қатынастар болу үшін

қолданушы тышқанмен ссылкіде немесе беттегі кнопканы шерту керек, содан кейін сценариді орындағанын және жауап қайтаруын күтеді. Динамикалық HTML мен қарым – қатынасты қоданумен тез орындалады;

-Сценариді тестілеу үшін серверге өз арнайы www сервері болуы керек, типі өндірістік нұсқа түрінде болуы жөн;

Кең спектрлі тапсырманы шешу үшін қазіргі уақытта ең негізгі шарт болып сервер жағындағы пограммалау болып табылады. Ол рұқсат етеді:

-Пішін көмегімен қолданушымен енгізілген сервердегі мәліметтерді алу және өңдеу;

-Платформаға, браузер клиентке тәуелсіз динамикалық web документтер құру;

-Мәліметтер базасының серверінің, яғни серверде орналасқан мәліметтерге динамикалық доступпен қамтамасыз ету (Мұндай доступ тәсілінде серверде сақталған мәліметтер өзгерген кезде HTML документі автоматты түрде өзгереді);

-Типтік тапсырмаларды орындауға арналған сервердің компоненттерді қолдану (Мысалғы, мынадай жарнамалық баннерлердің циклдік ауысуы және т.б.)

-Қолданушының аутентификациясын жүзеге асыру;

-Клиенттің браузері жөнінде мәліметтер алу;

-Клиенттің жағынан кілттерді құрастыру және оқу;

«Клиент-сервер» технологиясы

Көптеген Word wide web қосымшалары web серверімен басқарылатын ішкі программаларды қолдануға негізделген. Берілген программаларды қолдану динамикалық жаңару мәліметтерімен web қосымшасын құруға мүмкіндік береді. Мәліметтер базасында сақталған немесе шешілетін тапсырмалардың бизнес ережелеріне бағынышты генерилданатын. Web серверімен және шақырылатын программалар арасындағы байланыс үшін CGI кеңінен қолданылады, windows программасы үшін сондай-ақ Unix ортасындағы функционалданатын қосымшалардың нақтылығы бар. CGI, HTTP, Web сервер типтерінің ақпараттық серверімен ішкі қолданбалы программасының интерфейстің стандарты болып табылады. Әдетте, www серверінен гипертексті документтер статикалық мәліметтерден тұрады.

CGI-дің көмегімен CGI программаларын құруға болады, олар **шлюз** деп аталып қолданбалы жүйелер, мәліметтер базасын басқару жүйесі, электрондық кесте жұмыстық графика тағы да басқалардың қарым – қатынасы экранға қолданушының динамикалық ақпаратын шығара алады.

Шлюз программасы www серверінің нақты масштаб уақытында іске асырылады. www сервер қолданушының шлюз тапсырмасын қамтамасыз етеді, ал ол өз кезегінде қолданбалы жүйелік ортаны қолдана отырып, қолданушының экранына өңделген сұраныс нәтижесін қайтарады. Программа-шлюз C/C++, Fortran, Perl, TCL, Unix Schell, Visual Basic, Apple Script. тілдерінде кодталған болуы мүмкін. Ол әдетте, директорияда cgi-bin www сервер атымен орындалатын модуль сияқты болып жазылады. Бүкіл интернет және www-дің кейбір бөліктері «клиент-сервер» технологиясы бойынша жұмыс істейді, яғни барлық программалық қамтамасыздандыру клиенттік және серверлік бөліктерге бөлінеді. Сондай – ақ олардың арасында функцияналдық міндеттер ажыратылған.

Клиент жұмыстың принципінсервердің ішкі алгоритмдік іске асырылуын білмейді, білуі тиіс де емес, ал сервер клиенттің жұмыстарына араласпайды. Бұл бөліктердің ара қатынасы үшін арнайы протокол шығарылған (бұл жағдайда HTTP протоколы), және де сервермен клиент арасындағы қарым – қатынасты осы протокол ішінде іске асырылады. Браузерге бәрібір серверде қандай программалық қамтамасыздандыру тұрғаны, онда қандай операциялық жүйенің болуы, серверде сұраныс документтері қай жерде жатқаны (жалпы жатыр ма, олар арнайы программалармен лезде генералдануы мүмкін). Сервер сіздің

браузеріңіздің жұмысына араласпайды, клиент берілген ақпаратпен не істейді, қалай көрсетеді, дискі де сақтайды ма, әлде сақтамайды ма серверге бәрібір. Әдетте, серверге сұраныстан деп түсіндіріледі (бұл сервердегі мәліметтердің унифицилданған тапсырыстың пішіні).

Сервермен байланысты орнату кезінде алдымен www IP адресіндегі www.siemens.com секілді символдық аттардың трансляциясы болады, ал кейін TCP/IP - IP мәліметтерімен қосылуы іске асырылады. Серверге HTTP мәліметтеріне сұранысы жіберілген кезде клиент жауап келгенін күтеді. Клиенттен сұраныс болғанша HTTP сервері сұранысты күтіп, «ұйықтап жатады». Клиент байланыс орнатқан кезде сервер
«оянады» және бірілген сұранысты қабылдап, оларды өңдеуге кіріседі. Сервердің сұранысымен не істейтіні-сервердің өзіне ғана белгілі. Барлық монупуляциялардың жалғыз нәтижесі-ол клиенттің күтіп отырған жауабы болып табылады. Сервер жауап қайтарғаннан кейін ол байланысты үзіп қайтадан «ұйқыға түседі» және қателік шыққан кезде HTTP транзакциясы әр аптада аяқталуы мүмкін. Егер документтер табылмаған жағдайда немесе сізде оларға құқығыңыз болмаса (ресурсқа кіру шектеуші болуы мүмкін) немесе қателік коды беріледі. Ал егер барлығы дұрыс болса, документте сақталған ақпарат, типі жөніндегі мәліметтерді қоса отырып, жауап ретінде қайтарылып отырылады.

ISAPI

CGI секілді ISAPI спецификациясы web сервермен және қосымша программалармен арасындағы қарым қатынас ережелерін анықтайды. Серверде орындалатын программаны құру үшін альтернативтік тілтабысу неге керек екенін түсіну үшін, CGI стандартының артықшылықтары мен кемшіліктерін атап өтейік. CGI-дің сөзсіз артықшылығы болып оның универсалдылығы болып табылады. CGI әр тілде жазылуы және компьютерде әр түрлі архитектурада орындалуы мүмкін.

Егер сценариді жазған кезде барлық ережелерді ескерген болсаңыз, ол кезде сіздің құрған программаңыз кез-келген web серверімен дұрыс қарым қатынаста болатынына сенімді болуыңызға болады. CGF-дің қарапайымдылығы бұл стандарттың кең таралуына мүмкіндік береді. Бірақ бұл жетістіктермен қатар CGI сценариясында кейбір кемшіліктер де кездеседі.Олардың ең бастысы болып ресурстардың тиімсіз қолданылуы болып табылады. Сценариді шақыруды ұсынатын клиенттің әр сұранысы компьютер-серверде бөлек процесс туындатады. Процестің құрамындағы жұмысты орындауға қарағанда тәуелсіз процесті орындау көбірек ресурсты талап етеді.

Ресурстың өндірістігін жоғарлату және шығынды азайтуға ұмтылу корпоративтік шығуына әкеліп соқты. Ең атақтысы NSAPI және ISAPI болып табылады. NSAPI программасының CGI сценарисынан басты айырмашылығы болып, ISAPI программасының орындалмайтынын файлды емес динамикалық кітапхана (DDL) болып табылады. Осының арқасында программаны жеке процесс ретінде емес, ал web сервердің бір тұтас ағыны ретінде программаны іске қосу мүмкіндігі туды. Тәуелсіз поцесс жұмысын орындауға өарағанда тұтас ағын жұмысын орындауға аз ресурс жұмсалады. Тұтас ағын оның процесін туындатқан мекенжайлық кеңістікті қолданады және жеке процеске қарағанда тезірек жұмыс істейді. Процестің аяқталу кезінде жадыдан шығады, ал келесі сұраныс түскен кезде дискіге қайта жүктеледі. Бұл CGI-дің тиімді жұмыс істеуіне кері әсерін тигізеді. Сұраныстың қарқынды түсуі кезінде дискіні әрдайым ауыстыру қажеттілігі серверде қосымша жүк пайда болады.

CGI-сценариасына қарағанда ISAPI программасы жұмысын аяқтағаннан кейін жадыда резиденттік болып қалады және келесі сұраныстарды өңдеуде қолданылады. Бірақта ISAPI программасының артықшылығы кейде кемшіліктерге айналып кетеді. Егер CGI-сценариасының универсалдылығы үшін жеткіліксіз өндірістілікпен төлеу керек болса, ол кезде жоғары өндірістілік үшін ISAPI программасының негізін салушылар универсалдылығымен төлейді. ISAPI программасының негізін қалаушылардың тілдер арсеналы CGI-сценараисының негізін қалаушыларға қарағанда аз болып келеді. ISAPI программасында неғұрлым көп жазғысы келетін адамға C++-тан басқа амал жоқ. ISAPI программасының екінші кемшілігі біріншісіне қараңанда әлде қайда күрделірек. Берілген типтегі программа серверде туындалған тұтас ағын ретінде орындалады, ол сервердің мекен жайлық кеңістігін қоланады. Соған байланысты ISAPI программасының ұателігі авариалық аяқталуды ғана емес, сонымен қатар серверді істен шығаруы мүмкін. ISAPI спецификациясын қолданатын программалар екі категорияларға бөлінеді:

- Кеңейтілу. Функционалды түрде CGI-сценариасынан олардың айырмашылығы аз. CGI-сценариасы сияқты кеңейтілу web серверінде қосымша мүмкіндіктерді іске асырады, бірақ мүлде өзгеше құрылады. ISAPI кеңейтілуіне осы бөлім арналған;
- Фильтірлер. Бұл программаның өзгеше класы. Кеңейтілу web сервердің мүмкіндіктерін көбейтетін болса, фильтірлер оның жұмыс тәртібін қзгертеді. ISAPI кеңейтілуі оның жұмысы кезінде web серверімен байланысқан динамикалық кітапхананы (DDL) танытады.

ISAPI кеңейтілуі орындау кезінде негізі болып және берілген типтегі программаның CGI-сценариасынан айырмашылығын ажыратудың оңай тәсілі – кеңейтілудің шақыру процедурасын қарастыру және де жоғарыда көрсетілген функциялар мен параметрлердің міндеті боып табылады. Клиенттің сұранысы нақты URL программасын шақырған кезде ғана болатын кеңейтілуге қарағанда, ISAPI фильтірлері клиенттің серверге әр сұраныс жіберу кезінде басқара алады. web серверді іске қосқанда программалық-фильтір жадыда жүктеледі және сервердің жұмыс уақыты аяқталғанға дейін резидентті боып қалады.

Бақылау сұрақтары:

1. Сервер жағында орындалатын интернет қосымшаларын құрастыру кезіндегі ерекшеліктері қандай?
2. CGI технологиясының ерекшеліктері қандай?
3. ISAPI технологиясының ерекшеліктері қандай?
4. Сервер жағында орындалатын технологияның кемшіліктері мен артықшылықтары неде?