

Дәріс 5 Жүйенің объект моделі.

5.1.Объекттер мен кластар

Объекті (object) нақты немесе абстрактылы мәннің көрінісі ретінде қызмет етеді – мысалы көршінің жүк машинасының, дербес компьютердің, химиялық үрдістің, банктік транзакцияның.

Объекті – бұл түсінік, абстракция немесе программалық қосымшаның контексіндегі анықтама, мағына және нақтылы шектеулермен берілген қандайда бір жүйедей әрбір объектінің өзіндік 3 қасиеті бар – қалып – күй, тәртібі және идентификациялық ,яғни біркелкілеу белгісі.

Объектінің қалып – күй (state)-бұл оның бар болу шарттарының біреуі. Объектінің қалып – күйі атрибут –қасиеттерінің жиынтығымен және басқа объектілермен байланыс арқылы анықталады, уақыт өте өзгеру мүмкіндігі бар. Мысалы ,университет курстары есебі жүйесі ұсынған нақты белсенді объект ‘процессор’ жүргізетін курстық ұсынысының көрсететін объект ‘жабық’ және ‘ашық’-екі қалып күйдің бірінде болуы мүмкін. Курсқа жиналған студенттердің саны 10-нан кіші болғанша объект ‘ашық’ қалып – күйінде болады. Тек 10-шы курсқа тіркелгеннен кейін ғана объект ‘жабық’ қалып – күйіне өтеді.

Мінез – құлық (behavior) сипаттамасы объекті өмірінің функциональды жағын қамтиды, басқа объектілерден келетін сұраныстарға қарым қатынасын анықтайды және операциялардың жиынтығы түрінде іске асырылады. Университеттік курстар есебі жүйесінде курсты жүргізу ұсынысы объектісінің мінез –құлқы екі негізгі функциямен анықталады –‘студентті тіркеу’ және ‘студенттің тіркеуін тоқтату’.

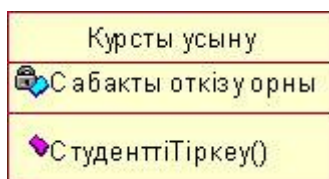
Идентификациялық (identify) немесе біркелкілік белгісі объектінің жеке қасиетін береді – соңғы қалып –күйі басқа объектілердің қалып –күйімен біркелкі болған жағдайда да.

Класс (Class) ортақ қасиеттері (атрибуттары), тәртібі (функциялары), семантикасы және басқа объектермен байланысы бар объектер тобын анықтайды. Класстың объектіні құруға арналған шаблон ретінде қарауға болады. Әрбір объект қандайда бір ғана кластың нұсқасы.

Дұрыс құрылған класс тек бір ғана абстракцияны бере алады. Мысалы, студент туралы мәлімет сақталған , сонымен қатар барлық оқу барлығында студент өткен курстар тізімі функциясы көрсетілген класты сәтті құрылған деп айта алмаймыз, өйткені ол екі әртүрлі операциялар тобын қамтиды.

Класстың атауы үшін пән облысына сәйкестендіріліп қабылданған терминдерді қолданған дұрыс. Класс аты ретінде жобаланатын түсінікті толығымен сипаттай алатын зат есіміңіз жекеше түрі қолданылады. Кейде қысқартылған атауларда қолданылып, класты құж аттандырғанда міндетті түрде мағынасын ашып көрсету керек. Егер аббревиатура бірдей мағыналы емес интерпретация жіберсе, онда сәйкесінше айтылудың толық түрін қолданамыз. UML–да класс зоналарға бөлінген тіктөртбұрыш түрінде кескінделеді. Жоғары зонада класс аты , ортасында оның құрылымы

(атрибуттар тізімі) оның астындағы сында тәртібі сипаттамаларын анықтайтын функциялар беріледі.



Классы қалай құрады

1. Browser терезесінің Logical View элементіне тышқан сілтеушіні қойып, контекстік менюді белсенді ету үшін оң батырмасын шертеміз.

2. менюді New--- Class элементтерін таңдау; Browser терезесінде кескінделген ағаш (бұтақ) жаңа класқа сәйкес келетін New- Class элементімен толықтырылады.

3. New-Class элементін таңдап және класстың қажет атауын енгізіп оны өзгертеміз.

Browser терезесінде кескінделген университеттік кластар есебі жүйесінің класстарының біреуі 5.1 суретте көрсетілген.



Сурет 5.1- Класстар есебі

Біз алдында қолдану варианттар диаграммасында байланыстар құруда стереотипті пайдалану туралы айтқанбыз. Жобаланған жүйенің элементтерінің жаңа түрін құруға мүмкіндік беретін нақты стереотиптер кластарға жауап береді. Кеңінен қолданылатын «стандарттың кластар стереотиптер қатарына келесілер жатады: entity(мән мазмұны) , boundary(шектеу) , control (бақылау) , utility(қолданбалы класс) , exception (ерекшелік).

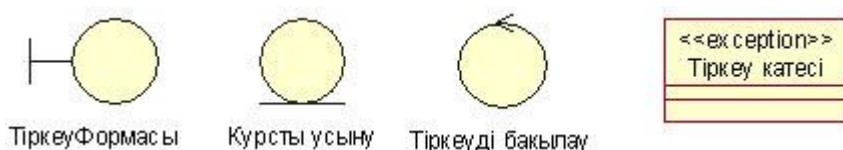
Диаграммаларда стереотиптер аттары бұрышты («ң») алынып кластар атынан кейін орналасады. Rational Rose жүйесі стереотиптер арнайы пиктограмманы бірлестіруге немесе оны сол немесе басқа түскен ерекшеленуге мүмкіндік береді. Entity , boundary және control стереотиптер мағынасы үшін Rational Unified Rose регламентінде қарастырылған пиктограмма мысалдары 4.4 суретте көрсетілген. Мұнда exception стереотипіне сәйкес келетін класс символының нұсқасы берілген.

Классы анықтайтын абстракцияны іздеу тәсілдерінің қатаң түрде жазылатын инструкциясы жоқ . Гради Буч (Grady Booch) бұл туралы былай айтқан : «Rational Unified Rose- те жобаланған жүйенің кластарын құру үшін заттардың класының стереотиптеріне, шегіне және басқарылуына мұқият қарау керек. Көрсетілген көптеген стереотиптер сараптаушы мен пән облысындағы дәрежесін, көрінісін және жүйемен басқарылуын анықтауға мүмкіндік береді.

Жобалау үрдісі өзіндік сипаттамалармен ерекшеленгендіктен өзінің өмірлік циклында жүйенің көптеген кластары өзгереді және толықтырылады. Үміткер класының бастапқы жиынтығы іс жүзіндегі ең соңғы нәтижеде жинақталған мүшелермен сәйкес келуі сирек құбылыс.

Мағыналық кластар

Мағыналық кластар (entity class) тұрақты сипаттамамен ерекшеленетін тәртіп және мәліметтер құрылымын жобалайды. Ұқсас типтің класы нақты әлемдегі заттың сапасын көрсетеді немесе жүйенің ішкі функцияларын орындауда пайдаланылады. Мағыналық кластар әдетте қоршаған ортаға тәуелді емес, яғни жүйенің жұмысына сыртқы құбылыстар мен жағдайлардың қалай әсер ететініне мән бермейді. Көптеген жағдайларда мұндай кластарды бірнеше әртүрлі қолданбаларды сәтті пайдалануға болады.



Сурет 5.2 –Класс және оның стереотиптері

Бірінші қадам қолданудың анықталған нұсқаларын қамтамасыз ететін құбылыстар ағымы талдауы нәтижесі негізінде жүйенің жауапкершілік сферасын шектеуден басталады. Мағыналық кластар, сол немесе басқа жауапкершілік сферасына қатысты функцияларды қолдау үшін жүйеге қажетті болады. Нақты бір функцияны сипаттау үшін зат есімнің қысқа ғана сөзі де жеткілікті. Алынған сипаттау тізімі пәндік облысқа қатысы жоқ фрагменттерді жою мақсатымен сүзілуге жіберіледі. Олар соңғысы немесе іске асырудың нақты ерекшеліктеріне арналған.

Мағыналық кластар, талапқа сай, жобалау базасында құрылады. Оларды нақты әлемнің затын және ұғымның абстракциясын беретіндіктен «мәндік деп жиі атайды.

Шекаралар кластары

Шекаралар класы (boundary class) қолданушылар мен жүйе жақтағылар (яғни активті субъектілер) үшін интерфейссті қамтамасыз ете отырып жүйемен және қоршаған ортамен қарым –қатынасы үрдісіне қызмет етеді. Мұндай кластар сыртқы әлеммен үздіксіз араласып тұратын жүйе бөлігін құрайды.

Шекаралар класын іздеу үшін әрбір «активті субъект / қолдану нұсқасы н жұп түрі талданады. Жобалау кезеңдерінде таңдалынған шекаралар кластар қарапайым жағдайда детальданудың төмен дәрежесімен ерекшеленеді. Мысалы, бұндай уақыттарда графикалық интерфейсінің терезесінің бөліктерін жасау (батырма, қосқыш, операциялар және т.б) ерекшеліктеріне тоқталмай-ақ толығымен функцияларын ғана құжаттандырып жобалауды мақсатқа айналдыру керек.

Қолданушының графикалық интерфейс элементтеріне қойылатын талаптар біркелкі емес. Алайда «достық интерфейс немесе майысқақ интерфейс» терминдері жиі қолданылғанымен де бәріне ұнау мүмкіндігі бар деп айта алмаймыз. Міне сондықтанда осындай ізденістерде коллективпен сараланған шешім және құрылған прототиптер өте жақсы нәтиже береді: тұтынушы болашақ жүйені ертерек «қолдану мүмкіндігіне ие болып, қаншалықты «достық» интерфейс екендігіне көз жеткізе алады. Кейін «достық» туралы табылған критерийлер шекаралар класының мінез- құлық сипаттамалары және құрылымы түрінде жинақталады. Жоспарлау үрдісі кезінде кластар оларды іске асыруға таңдап алынған механизмдер ерекшеліктері негізінде анықталады.

Жоғарыда айтылғандай, шекаралар класы құрылушы жүйенің басқа жүйелермен әсерлесу моделдеу үшін де қолданылады.

Басқару кластары

Бақылау кластары (control class)-бір немесе бірнеше қолдану нұсқаларын сәйкестендірмей жүйенің тәртібі сипаттамасы пайдаланғанда қолданылады және осы сұраулар тегінде жүйеде туындайтын және осы нұсқалар тегінде жүйеде туындайтын жағдайларды координаттайды. Басқару кластарын жүйесі қолдану нұсқау диаграммасын бейнелейтін абстракция ретінде қабылдауға болады. Әдетте басқару кластары нақты бір қолданбаға тәуелді болады.

Кластар диаграммасы

Моделде жаңа кластардың енгізілуінен ортақ суретті елестету күрделене түседі. Бұл жағдайда керекті пакеттер мен кластар жиыны ыңғайлы графикалық түрде көрсететін кластар диаграммасының (Class diagrams) көмегін пайдаланамыз. Модель кластарының негізгісі диаграммасы (main class diagram) көрсетеді. Әрбір пакеттің жеке меншік негізгі диаграммасы болады, онда пакетке тиісті ортақ қолданыстағы (Public) кластары беріледі. Қажет жағдайда қосымша кластар диаграммасы тұрғызылады, олар нақты бір қолданба нұсқаумен байланысқан. Төменде кейбір мысалдар келтірілген:

- жүйені іштен іске асыруға байланысты барлық класты көрсететін диаграмма .
- Бір немесе бірнеше кластардың тәртібі мен құрылымын сипаттайтын диаграмма.
- Кластың мұрагерлікке өту иерархиясын анықтайтын диаграмма.

Негізгі әдебиет: [1] – 131 б.

Қосымша әдебиет: [2] – 47б.

Бақылау сұрақтары:

1. Класс дегеніміз не?
2. Кластар диаграммасы не үшін қажет?
3. Басқару класстары?
4. Мәндік класс
5. Шекаралық класс?