

UNIMOD – АВТОМАТТЫ-БАҒЫТТАЛҒАН ПАКЕТІН МОДЕЛДЕУ ЕСЕПТЕРІНДЕ ҚОЛДАНУ

Сауханова Мағрипа Серғазықызы, магистрант,
Л. Н. Гумилев атындағы Еуразия ұлттық университеті, Астана қаласы
saukhanova@mail.ru

UNIMOD – АВТОМАТТЫ-БАҒЫТТАЛҒАН ПАКЕТІН МОДЕЛДЕУ ЕСЕПТЕРІНДЕ ҚОЛДАНУ

Автоматты программалау — программа жұмысын сипаттау үшін ақырлы автоматтарды қолданатын программалау стилі. Автоматты программалауда ақырлы автомат программаны ерекшелеу, жобалау, жүзеге асыру, жөндеу, құжаттау, сүйемелдеу барысында қолданылады.

Автоматты бағытталған программаларды құруды және орындауды UniMod пакетінің көмегімен жүзеге асыруға болады. Автоматтың оқиғалар және басқару объектілерімен арасындағы байланыс схемаларын көрсету үшін UML тілінің кластар диаграммасының нотациясы, ал өту графтары үшін UML тілінің күй диаграммаларының нотациясы қолданылады.

UniMod пакеті байланыс схемасы мен өту графына сәйкестендірілген UML кластары мен күйлер диаграммаларын құруға, өңдеуге мүмкіндік береді. UniMod пакеті - *Eclipse* платформасына қондарылатын модуль (*plug-in*).

UML тілінің стандартында диаграмма синтаксисі мен семантикасы шектеулер жиынтығымен анықталған. Дұрыс құрылған диаграмма осы шектеулер жиынтығын қанағаттандыруы керек.

Автоматты программалауды насихаттаушы А.А. Шалыто осы шектеулер жиынтығын одан әрі кеңейтті, олар:

- диаграммадағы барлық күйлерге қол жеткізу мүмкіндігі;
- кез-келген күйден шығатын өту доғаларындағы шарттар толық болып, өзара қайшылықта болмауы.

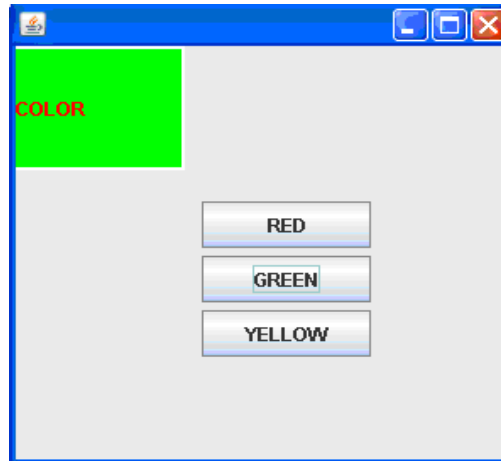
UniMod пакеті

- мәтіндік программалау тілінде жазылатын код көлемін азайтуға;
- байланыс схемаларын және өту графтарын құруға, оларды жобалау құжаттарына қосуға;
- программаның орындалу тәртібін формалды және көрнекі түрде сипаттауға, оларды өзгерту кезінде көп жағдайда тек өту графына ғана өзгерістер енгізуге;
- программа логикасының жеке шоғырлануына байланысты жобаны қолданысқа енгізуді жеңілдетуге мүмкіндік береді.

Өзіне ұқсас басқа өнімдермен салыстырғанда UniMod пакетінің келесі артықшылықтары бар:

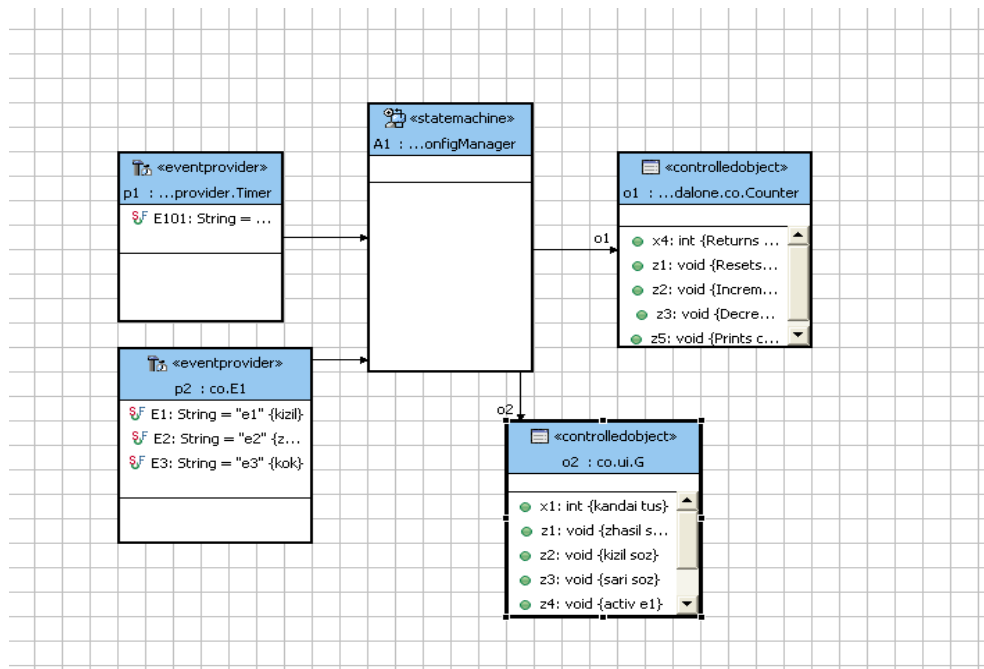
- моделде өзара байланысқан автоматтар жүйесін қолдану арқылы күрделі есептің әрекеттер тәртібін кіші есептерге декомпозициялауға болады;
- кірістірілген күйлермен қатар кірістірілген автоматтар қолданылады, олардың тереңдігі шектеусіз;
- шақырулар арқылы автоматтар өзара әрекеттесе алады;
- программаны бірден толығымен құруға мүмкіндік береді;
- жоба ашық жоба болып табылады.

UniMod пакетінің көмегімен формада COLOR сөзін белгілі бір уақыт аралықтарында әртүрлі түстермен жазу және осы сөз орналасқан раманың түсін басқару пернелері арқылы өзгерту мысалын қарастырайық. Төмендегі суретте программаның орындалу барысындағы кескіні бейнеленген.



1 сурет. Программаның орындалу барысындағы форманың бейнесі

Бұл жобаны құру үшін алдымен автоматтың байланыс схемасын құрамыз.



2 сурет. Автоматтың байланыс схемасы

Мұнда *p1* және *p2* кластары – оқиғалар бастауы, *o1* және *o2* кластары – басқару объектілері.

P1 бастауы – UniMod ортасында құрылған бірдей уақыттар арасында оқиғаларды орындаушы таймер. *P2* бастауы пернені басу оқиғасын жеткізеді. Келесі кестелерде оқиғалар бастауы берілген.

P1 класының оқиғалары

Оқиға	Сипаттамасы
e101	Секунд өтті

p2 класының оқиғалары

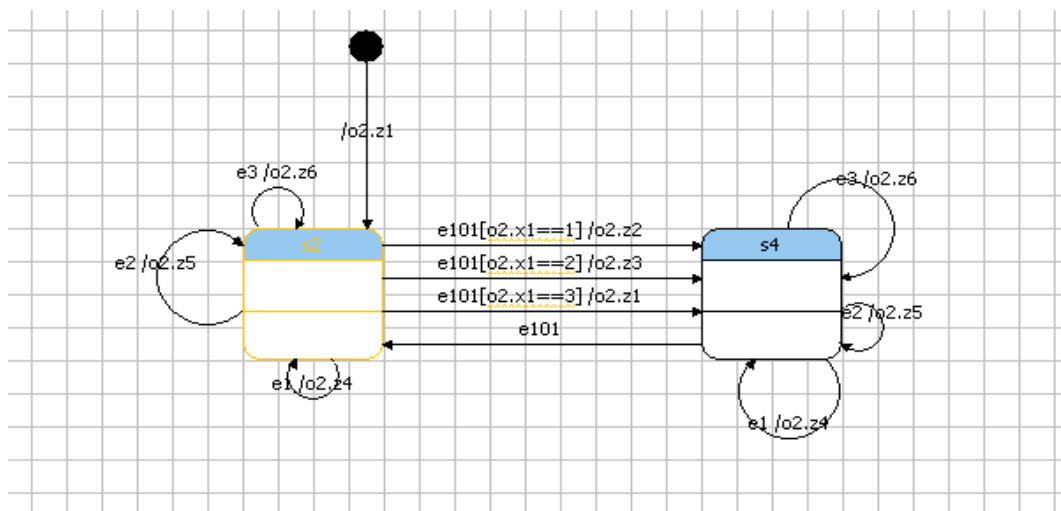
Оқиға	Сипаттама
-------	-----------

e1	Пайдаланушы раманы қызыл түске ауыстырды
e2	Пайдаланушы раманы жасыл түске ауыстырды
e3	Пайдаланушы раманы сары түске ауыстырды

A1 автоматына $p1$ және $p2$ бастауларынан оқиғалар жеткізіледі. Автомат $o2$ басқару объектісінің келесі әдістерін шақыра алады.
 $O1$ басқару объектісінің кіріс және шығыс әдістері

Әдіс	Сипаттама
x1	Ағымдағы түс қандай?
Z1	Сөзді жасыл түске ауыстыру
z2	Сөзді қызыл түске ауыстыру
z3	Сөзді сары түске ауыстыру
z4	Жақтауды жасыл түске бояу
z5	Жақтауды қызыл түске бояу
z6	Жақтауды сары түске бояу

Келесі қадамда автоматтың жұмыс істеу логикасына қатысты өту графы сызылады.



3 сурет. Автоматтың өту графы.

Автоматтың $s2$ және $s4$ екі күйі бар. Бастапқы күйден $s2$ күйіне өту кезінде сөз жасыл түсті болады ($o2$ басқару объектісінің $z1$ әдісі шақырылады). Әрбір секунд сайын ($e101$ оқиғасы) $s2$ күйінен $s4$ күйіне және керісінше ауыстырулар жүзеге асырылады. $s4$ күйіне өту кезінде $x1$ кіріс әрекетінің мәні тексеріледі, оған сәйкес түс ауыстырылады. Ал басқару пернелерін басу арқылы сөздің жақтауларының түсі пернеге сәйкес түстерге ауыстырылады.

Өту графын құру кезінде доғалардағы шарттар толық болып, өзара қайшылықта болмауы керек. Бұл шектеу орындалмаған жағдайда сәйкес доға қызыл бояумен ерекшеленеді. Оны жөндеу үшін контексті менюден Quick Fix жолы таңдалып, ұсынылған шарттар тізімінің ішінен қажеттісі алынады. Автоматты орындауға жіберу үшін контексті менюден Launch A1 жолы таңдалады.

Экранда шығарылатын форманы құратын ConPV классы және оның Vm интерфейсі жазылады. Басқару пернелеріне сәйкес туындайтын оқиғалар көзін байланыстыру үшін Java тіліндегі ActionListener класының actionPerformed(ActionEvent e) әдісі қолданылады. Оның анықтамасы төмендегідей:

```
public void actionPerformed(ActionEvent e){
    if ( e.getSource()==b1)
```

```

        fireEvent(co.E1.E1);
    if ( e.getSource()==b2)
        fireEvent(co.E1.E2);
    if ( e.getSource()==b3)
        fireEvent(co.E1.E3);
    }

```

UniMod пакеті программаның ішкі логикасына сәйкес кодын автоматты түрде өту графына сәйкестендіріп өзі құрастырды. Программисттің жауапкершілігінде автоматтың байланыс схемасын, өту графын және басқару объектілерінің кодын құру ғана қалады. о2 басқару объектісінің әдістері G класының әдістері түрінде жазылады:

```

package co.ui;
import com.evelopers.unimod.runtime.ControlledObject;
import com.evelopers.unimod.runtime.context.StateMachineContext;
import co.ui.ConPV;
import com.evelopers.unimod.adapter.standalone.util.InputPopup;
public class G implements ControlledObject {
int y=0;
    /**
     * @unimod.action.descr zhasil soz
     */
    public void z1(StateMachineContext context) {
        ConPV.getvm().L();
        y=1;
    }
    /**
     * @unimod.action.descr kizil soz
     */
    public void z2(StateMachineContext context) {
        ConPV.getvm().L1();
        y=2;
    }
    /**
     * @unimod.action.descr sari soz
     */
    public void z3(StateMachineContext context) {
        ConPV.getvm().L2();
        y=3;
    }
    /**
     * @unimod.action.descr activ e1
     */
    public void z4(StateMachineContext context) {
        ConPV.getvm().L3();
    }
    /**
     * @unimod.action.descr activ e2
     */
    public void z5(StateMachineContext context) {
        ConPV.getvm().L4();
    }
    /**

```

```

    * @unimod.action.descr activ e3
    */
public void z6(StateMachineContext context) {
    ConPV.getvm().L5();
}
/**
 * @unimod.action.descr kandai tus
 */
public int x1(StateMachineContext context) {
    switch (y){
        case 1: return 1;
        case 2: return 2;
        case 3: return 3;
    }
    return 0;
}
}

```

Көріп отырғанымыздай программаның негізгі басқару логикасы UniMod пакетінің көмегімен автоматты түрде құрылады да, тек программистке басқару объектілеріне және экранда бейнеленетін формаға ғана қатысты қарапайым программалар кодын жазу жүктеледі.

Қолданылған әдебиеттер:

1. Шалыто А.А. SWITCH- технология. Автоматизация и программирование задач логического управления. СПб.: Наука, 1998
2. Шалыто А.А. Автоматное проектирование программ. Алгоритмизация и программирование задач логического управления. Журнал «Известия Академии наук. Теория и системы управления». №6. Ноябрь-Декабрь 2000. с. 63-81
3. Поликарпов Н. И., Шалыто А.А. Автоматное программирование. Учебно-методическое пособие. СПб., 2007

Түйіндеме

Статьяда қарастырылған автоматты программалау автоматтарды қолдануда олардың өту графтарының берілуімен ерекшелінеді.

UniMod инструменталды құралы автоматты-бағытталған программаларды құруды және оларды орындауды қамтамасыз етеді.

UniMod пакетін қолданып автоматты және оның өту графын кадамдап құру мысалы көрсетілген.

Резюме

Особенность рассматриваемого в статье автоматного программирования заключается в том, что при его использовании автоматы задаются графами переходов.

Инструментальное средство UniMod обеспечивает разработку и выполнение автоматически - ориентированных программ.

Предложен пример по шагового построения автомата и графа перехода с использованием пакета UniMod.