Скриптов език за обработка и визуализация на данни от учебни сесии

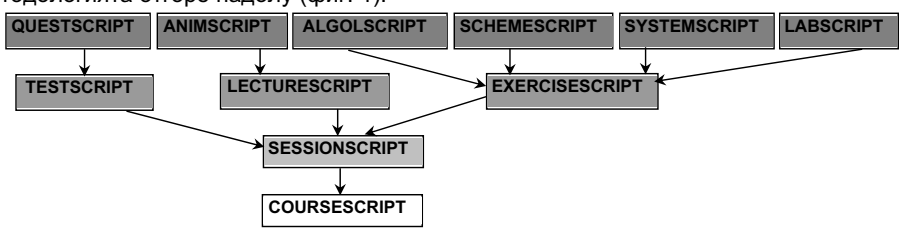
Октай Кър, Ирина Желязкова

A Script language for Processing and Visualization of Learning Sesions Data: *The present work continues a sequence of papers, concerning a teacher's tool for processing and visualization of learning session data. Consequently in three paragraphs the tree with the keywords, syntax and example programs are presented and commented.*

Key words: *Script Language, Processing and Visualization, Learning Sessions Data.*

ВЪВЕДЕНИЕ

За описание на знанията в интегрирана среда за индивидуализирано планирано обучение (ИСИПО) се разработва скриптов език **COURSESCRIPT,** независим от обучаващия курс (ОК) [5]. По същество този език представлява йерархична структура от няколко подезици за описание на предметни и педагогически знания като при специфицирането, реализирането и тестването му се прилага методологията отгоре надолу (фиг. 1).



Фиг. 1. Йерархията на езика COURSESCRIPT

Всеки език на горното ниво служи за описание на предметните знания на автора (А) и обучавания (О). Те се извличат в относително самостоятелна задачно-ориентирана среда (ЗОС) чрез онтологично конструиране на решението на определен познавателен тип задачи. Досега разработените подезици на това ниво са предназначени за описание на: различни видове тестови въпроси (**QUESTSCRIPT**) [7], лекционни понятия **ANIMSCRIPT** [8], блок-схеми на алгоритми (**ALGOLSCRIPT**) [9], различни видове схеми (**SCHEMESCRIPT**) [10] и сложни динамични системи (**SYSTEMSCRIPT**) [6]. Планира се специфицирането и на подезик **LABSCRIPT** за описание на лабораторни обекти за измерване, контрол и управление**.** При това ако първите два подезика са независими, останалите са зависими от типа на задачите за упражнения.

Всеки от езиците на средно ниво (**TESTSCRIPT**, **EXERCISESCRIPT**, **LECTURESCRIPT**) описва педагогическите знания на П, необходими за планиране, наблюдение и изпълнение на съответната учебна сесия, т.е. лекция, тест, упражнение.

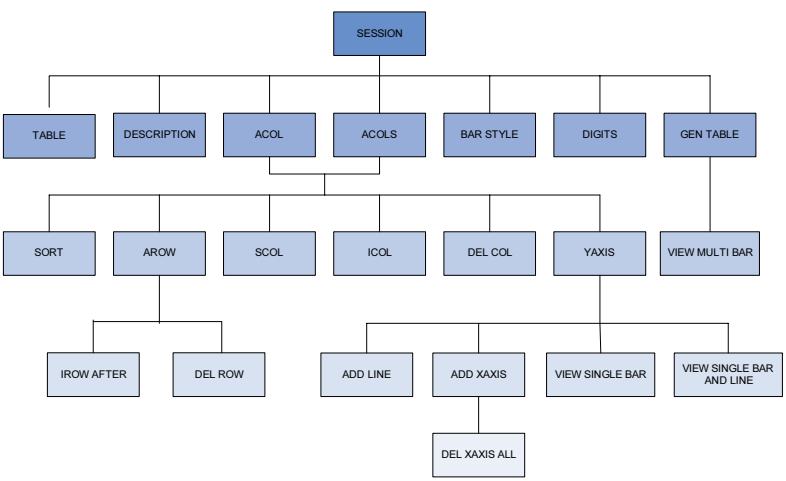
Изброените по-горе подезици могат да бъдат класифицирани като езици **от много високо ниво**, понеже ключовите им думи са близки до тези от общоприетия език за съответния ОК. Суфиксът **SCRIPT** в наименованието на всеки означава, че съответните знания се съхраняват в обикновени текстови файлове, след което се интерпретират от съответен езиков процесор. От гледна точка на програмирането, подезикът на ниво задача представлява множество от подпрограми, а подезикът на ниво сесия – програма, която ги извиква. Подезиците на горното ниво са **вътрешни** като основно се използват за автоматично изчисляване на следните унифицирани параметри на задачите, в частност въпросите (обем на знанията, степен на

системна подсказка, ниво на трудност, очаквано време за изпълнение), както и коефициента на близост на знанията на обучавания (О) до тези на втора (А). Подезиците на следващото по-ниско ниво се използват за автоматично изчисляване на аналогичните параметри (допълнително и скалата за оценка) на тестовете, лекциите и упражненията. Всички изброени подезици са **визуални**, понеже изискват от потребителите да манипулират само с графичните компоненти на потребителския интерфейс (менюта, бутони, икони, изображения и др.) и изцяло генерират скрипта. Въпреки това А/П може да използва съответния подезик, за да увеличи гъвкавостта и продуктивността на преподавателската си дейност, докато О не е необходимо да ги познава изобщо.

Обект на разглеждане в настоящата работа е подезик за обработка и визуализиране на данни от учебни сесии, наречен **SESSIONSCRIPT**. В йерархията на езика **COURSESCRIPT** той се разполага на ниво по-ниско от сессийното (фиг.1). Тук в три отделни параграфа се представят и коментират дървото с ключовите думи на езика, синтаксиса на програмите на този език, както и три примерни програми за изясняване на семантиката на езиковите конструкции. Заключението съдържа основните резултати и бъдещи планове на колектива.

ДЪРВОТО С КЛЮЧОВИТЕ ДУМИ

Дървото с ключовите думи на езика **SESIONSCRIPT** е дадено на фиг. 2. От него се вижда, че тази първа версия на езика има само 21 съчетания от думи. Всеки връх отговаря на команда. Изключение правят два върха (**DIGITS** и **BARSTYLE**), които представляват локални директиви, т.е. всяка следваща директива от даден вид отменя действието на предишната. Всички ключови думи лесно могат да се запомнят, тъй като представляват съкращения от английски език и напомнят за предназначението на съответните команди и директиви.



Фиг. 2. Дървото с ключовите думи на езика

СТРУКТУРА НА ПРОГРАМА

За описание на общата структура на програмите на този език използваме метаезика на Бекус-Наур (фиг. 3). Ключовите думи на езика **SESSIONSCRIPT** са отбелязани с главни и удебелени букви, а специалните символи имат следния смисъл: **::=** – дефинира синтактична конструкция; **_** – свързва думи в наименование на конструкция; **|** – разделя две алтернативни конструкции (функция ИЛИ); **{ }** – заграждат повтарящи се конструкции; **[ ]** – заграждат незадължителни конструкции; **< >** - заграждат име на конструкция, която все още не е дефинирана.

```
<SESSIONSCRIPT_програма> ::= SESSION
TABLE "<низ>"
DESCRIPTION "<низ>"
{<описание_на_колона>}
{<описание_на_ред>}
{<описание_на_диаграма>}
{<описание_на_израз>}
GEN TABLE ("<низ>",<низ>{[,<низ>]})
DIGITS <цяло>

<описание_на_колона> ::=
ACOL "<низ>" [=<низ>]
ACOLS ("<низ>{[,"<низ>"]})
ICOL "<низ>" AFTER "<низ>"
DEL COL [<цяло>] | "<низ>" | ALL
SCOL "<низ>","<низ>"
SORT "<низ>"
"<низ>"=<низ> | <израз> | <вградена_функция>

< описание_на_ред> ::=
AROW ("<низ> | <реално>" {[,"<низ> | <реално>"]})
IROW AFTER [<цяло>]  [("<низ> | <реално>" [, "<низ> | <реално>" {[, "<низ> | <реално>"])]})
DEL ROW [<цяло>] | ALL

< описание_на_диаграма> ::=
YAXIS "<низ>"
ADD XAXIS "<низ> | <реално>"
ADD LINE <низ>
BARSTYLE <цяло>
DEL  XAXIS ALL
VIEW SINGLE BAR | SINGLE BAR AND LINE | MULTI BAR
< описание_на_израз > ::= <низ>=<израз>
<низ> ::= <буква>{[<буква>|<цифра>]}
<цяло> ::= <цифра>{[<цифра>]}
<реално> ::=<цяло> | <цяло>.<цяло>{[<цяло>]}
<вградена_функция> ::=
SUM | AVG | MAX | MIN (<низ> {[ , | .. <низ>]})
```

Фиг. 3. Описание на езика SESSIONSCRIPT с метаезика на Бекус-Наур

Операндите на <израз> могат да бъдат числови константи (цели и реални) или имена на предварително описани параметри. В съответствие с приоритета операциите, допустими в <израз>, са: +, -, *,/. Смяната на приоритета е възможна с използване на ( ).

ПРИМЕРНИ ПРОГРАМИ

За демонстриране на възможностите на първата версия на езика **SESSIONSCRIPT** тук са представени три програми. Програмата на фиг. 4 е предназначена да създаде таблица. В първия ред е зададено физическото име на файла **TABLE1**, а във втория – надписът на таблицата, която в случая съдържа резултати на студенти по дисциплината **PROGRAMIRANE1**. Първоначално да добавени 6 колони, като първата е с наименование **STUDENT**, а останалите са с наименования от **Q1** до **Q5**. Добавени са и два реда, всеки един съдържащ факултетния номер на студента и неговите резултати във вид на цели числа. След

това са добавени четири колони с наименованията на допустимите вградени функции съответно за намиране на: сума (**SUMA**), средно-аритметична стойност (**AVGG**), максимална (**MAXI**) и минимална стойност (**MINI**). Броят на значещите цифри (**1**) след десетичната точка при изчисленията е зададен с директивата **DIGITS**. Следващите 4 реда са зададени формулите за изчисляване на вградените функции с два различни начина за задаване на аргументите: с използването на пълен списък (например, **Q1,Q2,Q3,Q4,Q5**) и с използването на диапазон (например, **Q1…Q5**). Редовете 16 и 17 демонстрират оператори за присвояване, а редовете 18 и 19 вмъкване на празна колона с наименование **T0** след колона **Q5** и запълване на клетките й с изчисляваната стойност на **X**. Следващият фрагмент от 20 до 24 представлява изчертаването на координатната система като по оста Y се разполагат стойностите на функцията **SUMA**, а по оста **X** – факултетните номера на двама студенти. Следва изчертаване на хоризонтална линия с предварително изчислена стойност **L**. Ред 25 означава сортиране на колона **Q1**, а ред 26 размяна на колоните **Q1** и **Q2**. Последните три реда изтриват първия ред, колона **Q4**, а след това и всички останали колони.

```
 1  TABLE "TABLE1"
 2   DESCRIPTION "PROGRAMIRANE"
 3   ACOL “STUDENT”
 4   ACOLS (“Q1","Q2","Q3","Q4","Q5")
 5  AROW ("063526","4","5","9","9","9”)
 6  AROW ("063527","9","5","4","7","9")
 7  ACOL "SUMA"
 8  ACOL "AVGG"
 9  ACOL “MAXI”
10  ACOL "MINI"
11  DIGITS 1
12  "SUMA" = SUM(Q1..Q3,Q4,Q5)
13  "AVGG" = AVG(Q1..Q5)
14  "MAXI" = MAX(Q1..Q5)
15 "MINI" = MIN(Q1..Q5)
16  PMAX = 45
17  X = SUMA * 6 / PMAX
18  ICOL "T0" AFTER "Q5"
19  "T0" = X
20  YAXIS “SUMA”
21  ADD XAXIS "063526"
22  ADD XAXIS "063527"
23  L = AVG(SUMA)
24  ADD LINE L
25  SORT “Q1”
26  SCOL “Q1”,”Q2”
27  DEL ROW [1]
28  DEL COL “Q4”
29  DEL COL ALL
```

Фиг. 4. Създаване на таблица

Програмата на фиг. 5 е предназначена да генерира таблица. Първият ред задава с директивата **DIGITS** точността на следващите изчисления **4** знака след десетичната точка. Редовете от 2 до 6 представляват оператори за присвояване, от дясната страна на които фигурират различни варианти на аритметични изрази от фиксирана стойност през променлива и операция до израз с използване на скоби. По същество последният ред генерира таблица с колони **STUDENT** от тип символен низ и три реда, наименовани като M0, M1 и M2 от реален тип.

```
1  DIGITS 4
2  PMAX = 158
3  TMAX = 120
4  M0 = SUMA / PMAX
5  M1 = ( 2 * SUMA – PMAX )  / PMAX
6  M2 = M0 * TMAX / T0
7  GEN TABLE ("STUDENT",M0,M1,M2)
```

Фиг. 5. Генериране на таблица

Програмата на фиг. 5 демонстрира ограничените засега възможности на езика за изчертаване на различни видове диаграми. С директивата **BARSTYLE** на първия ред се задава един от следните стилове на баровете, които са различно оцветени: 0 – паралелепипед без преливащ цвят; 1 – права триъгълна пирамида; 2 – обратна триъгълна пирамида; 3 – цилиндър, 4 – вертикална елипса, 5 – стрелка нагоре, 6 – паралелепипед с преливащ цвят. Следващите два реда съответстват на изчертаване на единични бар диаграми съответно без и с хоризонтална линия, а последният – на мулти бар диаграма.

```
1 BARSTYLE 4
2 VIEW SINGLE BAR
3 VIEW SINGLE BAR AND LINE
4 VIEW MULTI BAR
```

Фиг. 6. Изчертаване на бар-диаграми

ЗАКЛЮЧЕНИЕ

Специфициран е скриптов език от много високо ниво за таблична обработка и визуализация на данни от учебни сесии (тест, лекция, упражнение). Представлява подезик от език за описание на педагогически знания за различни обучаващи курове. Подобно на други вече реализирани подезици има малко на брой ключови думи, прост и лесен синтаксис, но за разлика от тях не е вътрешен.

Реализацията му в инструментално средство може да подобри гъвкавостта и продуктивността на обработката от преподавателя без да изисква от него професионални умения по програмиране. При наличието на готови програми те могат да бъдат лесно модифицирани на ниво текстов файл със стандартен текстов редактор.

Предвижда се постепенно разширяване на езика с нови изчислителни функции и нови генерирани диаграми в съответствие с предварително разработената и проверена с реални данни методика с използването на EXCEL.

ЛИТЕРАТУРА

[1] Zheliazkova I. I., A Domain-Independent Language for Courseware Knowledge Description, *Proceedings of the Second National Conference on E-Leaning in the Higher Education*, 14th –16th July, Kiten, Bulgaria, 2006, pp. 44-48.

[2] Georgiev G. T., Zheliazkova I. I., Language for Representation Knowledge for Training in Dynamic systems, *National Journal " Automatics and Informatics ",* No 3, 1999, pp. 52-55 (in Bulgarian).

[3] Zheliazkova I. I., Andreeva M. H., An Intelligent Multimedia Environment for Knowledge Testing, *E- learning and the Knowledge Society*, Gent & Brussels, Belgium, 6-8 September 2004, pp. 3.13.1-3.13.24.

[4] Zheliazkova I. I., Stefanova S. P., Programming Language for Knowledge Representation of an Animation based on Didactic Petri Nets, *National Journal " Automatics and Informatics"*, No. 5-6, 2000, pp. 21-25.

[5] Zheliazkova I. I., Atanasova G. E., A Visual Language for Algorithm Knowledge**,** *Proceedings of the International Conference on Computer Systems and Technologies (e-Learning),* Rousse, Bulgaria, 17-18 June, 2004, pp. IV.24-1- IV.24-6.

[5] Zheliazkova I. I., Georgiev G. T., Valkova P. L., A Visual Language for Structural Knowledge Description, *Newsletter of the Union of Scientists – Rousse, Section of Mathematics, Informatics, and Physics*, Vol. 5, 2004, pp. 111-120.

За контакти:

Доц. д-р Ирина Желязкова, Катедра "Компютърни системи и технологии", Русенски университет "Ангел Кънчев", тел.: 082-888 711, e-mail: Irina@ecs.ru.acad.bg

Докладът е рецензиран.