



СООБЩЕНИЯ  
ОБЪЕДИНЕННОГО  
ИНСТИТУТА  
ЯДЕРНЫХ  
ИССЛЕДОВАНИЙ  
ДУБНА

4612/2-81

7/9-81

P10-81-361



Т.А.Стриж

ВХОДНОЙ ЯЗЫК  
СИСТЕМЫ ГЕНЕРАЦИИ ПРОГРАММ  
ОБРАБОТКИ ФИЛЬМОВОЙ ИНФОРМАЦИИ

1981

Разработанная в ЛВТА ОИЯИ система генерации /1,2/ обеспечивает широкому кругу пользователей возможность самостоятельно осуществлять сборку, трансляцию, создание библиотеки требующейся программы обработки фильмовой информации /3/ из имеющихся модулей и организовывать счет по созданной библиотеке.

Одним из важных элементов этой системы является входной язык, описанию которого посвящено данное сообщение.

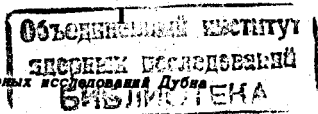
## I. Особенности системы генерации программ обработки фильмовой информации

Система генерации служит для автоматизации процесса получения различных версий программ обработки. При ее создании учитывались как специфика организации программ обработки фильмовой информации /3/, так и особенности их использования /4/. В связи с этим была разработана оригинальная методика генерации программ из текстовых фрагментов, которая, в отличие от описанных в литературе /5,6/, базируется на широком использовании уже имеющегося математического обеспечения, организованного в виде банков текстовых фрагментов /7/.

Генерация программ в созданной системе ведется в два этапа /1/. На первом по запросам пользователя формируется задание программисту системы уратснх /7/, а на втором происходит сборка, трансляция, организация библиотеки программы и, если нужно, передача на счет по библиотеке.

Язык запросов /2/ предназначен для описания нужной программы. Он состоит из отдельных лаконичных предложений, в которые включены хорошо знакомые пользователям термины. Эти предложения заносятся в словарь системы.

Отсутствие в языке запросов специальных символов, ограничителей и т.п., в отличие от языков, применяемых в системах обработки данных /8/, сделало его более удобным для широкого круга пользователей. Дальнейшее облегчение работы пользователей достигается за счет совмещения в рамках одного задания ЭВМ описания программы и выполняемых



с ней операций ( сборка, трансляция, организация библиотеки и т.п. )

В общем случае в процессе генерации программ можно выделить следующие этапы:

- описание требующейся программы;
- организацию доступа к банкам текстовых фрагментов;
- перевод описания программы на язык директив программы-сборщика УРАТСНУ /7/;
- сборку текста требуемой программы;
- трансляцию собранной программы;
- создание библиотеки собранной программы;
- ввод данных;
- счет по собранной программе;
- запись результатов счета.

Кроме этого необходимо подчеркнуть еще одну важную особенность разработанной системы генерации, а именно то, что она предназначена для стандартизации работ с любыми пакетами \* включенными в комплекс программ обработки фильмовой информации /2,9,10/.

Реализация описанных выше особенностей системы генерации привела к созданию входного языка, который имеет

- изобразительные средства, описывающие конкретную задачу в знакомых пользователю терминах;
- директивы, определяющие различные этапы генерации программ;
- средства организации операций ввода/вывода;
- доступ к информационно-обучающей службе системы.

При его разработке был использован язык ССЛ (Cyber Control Language) /11/, входящий в математическое обеспечение ЭВМ CDC-6500. На рис. I проиллюстрированы возможности, предоставляемые входным языком пользователям. Слева приведено "обычное" задание ЭВМ CDC-6500 на генерацию программы /2/, справа - это же задание с использованием входного языка системы.

```
JOB CARD
ACCOUNT CARD
REDUCE.
ATTACH, TAPE9, DATA, ID=MINE, MR=1.
ATTACH, PAM, GENERFILES, ID=MINE, MR=1.
ATTACH, ULIB6, ID=LCTIVA, MR=1, PW=R.
LIBRARY, ULIB6.
YPATCHY, , , D, AWAY.
REWIND, TAPE7, JOBFIL.
YTØBIN, JOBFIL, CATALOG, , g-+g, AWAY.
RETURN, AWAY, PAM.
ATTACH, ASK, READ, ID=MINE, MR=1.
ASK.
YSEARCH, CATALOG, CRADLE, TAPE10, DUM.
REWIND, CRADLE.
RETURN, TAPE7, JOBFIL, ASK, TAPE10.
ATTACH, HYCDE, HYCDEPAM, ID=MINE, MR=1.
ATTACH, GEOM, GEOMPAM, ID=MINE, MR=1.
ATTACH, CONTROL, CONTROLUSERPAM, ID=LCTIVA, MR=1.
YPATCHY, , , CRADLE.
RETURN, CRADLE, HYCDE, GEOM, CONTROL.
FTN, I=OMAIN, L=0.
FTN, I=ASM, B=BINLIB, L=0.
RETURN, OMAIN, ASM.
EDITLIB, L=AW.
RETURN, AW, BINLIB.
ATTACH, GENSECLIB, ID=MINE, MR=1, PW=R.
ATTACH, HYDRALIB, ID=MINE, MR=1, PW=R.
LDSET (PRESET=ZERO, LIB=USLIB/HYDRAL/GENSECL)
LOAD, LGO.
MAIN, TITLE.
7/8/9 EØR
+EXE.
+ASM, N=21, F=TAPE5.
+ASM, N=31, F=JOBFIL.
+USE, DICTCONTROL.
+USE, CONTROLCATALOG, T=EXE, DIV.
+PAM.
+QUIT.
7/8/9 EØR.
CHAMBER LUDMILA
CHECK EVENTS
INPUT HPD
EXP35 LUDMILA
7/8/9 EØR
LIBRARY (USLIB, NEW)
ADD ( * , BINLIB)
FINISH.
ENDRUN.
```

```
JOB CARD
ACCOUNT CARD
REDUCE.
ATTACH, TAPE9, DATA, ID=MINE, MR=1.
ATTACH, P, NEWPROCLIB, ID=MINE, MR=1.
LIBRARY, P.
WANT, CHAMBER LUDMILA, CHECK
EVENTS, INPUTHPD, EXP35LUDMILA.
GO, CONTROL, O.
```

Рис. I

\* Под пакетом программ здесь понимается набор программ, обеспечивающих, соответственно, обработку данных с жидководородных камер ("Людмила" и ВПК-100) /13/, контроль результатов обмера, создание лент суммарных результатов, ведение каталога обрабатываемых событий и т.п.

## 2. Входной язык системы генерации

Входной язык системы рассчитан на пользователя непрограммиста, от которого требуется умение сформулировать задачу в терминах имеющегося словаря. Основным элементом языка является директива. Она состоит из имени и списка параметров. Число символов в имени директивы не может превышать 7, а число параметров в списке - 50.

Ниже приводится описание синтаксиса входного языка с использованием металингвистических формул<sup>12/</sup>.

```
<задание> ::= <текст задания> <конец задания>
<текст задания> ::= <элемент задания> | <текст задания> <элемент задания>
<элемент задания> ::= <директива>
<конец задания> ::= GEND
<директива> ::= <имя директивы> <признак конца директивы> | <имя директивы> << список параметров >>
<имя директивы> ::= GO/GENPACK/RUNPACK/GOGEN/RUN/COLLECT/PFTN/LIB/EX/WANT/UNIT/CHANGE/ARRAY/INQUIRY/ALL/EXAMPL/AFILE/DIST
<признак конца директивы> ::= .
<список параметров> ::= <параметр> | <список параметров> , <параметр>
<параметр> ::= <пусто> | <значение параметра>
<пусто> ::=
<значение параметра> ::= < имя пакета > | < идентификатор > | < целое число > | < цифра > | < строка символов > | < тип устройства >
< имя пакета > ::= LUDMILA/CONTROL/VEK100/RISK/MIS/EDIT/ONLSUMX/STATUS
< идентификатор > ::= < буква > | < буква > ^ < цифра > | < буква > < цифра > ^ < цифра >
< буква > ::= A/B/C/D.../Z
< цифра > ::= 0/1/2/3/4/5/6/7/8/9
< целое число > ::= < цифра > | < цифра > ^ < цифра >
< строка символов > ::= $ < буква > < символ > ^ < цифра >
< символ > ::= - / . / + / ( ) /
< тип устройства > ::= DVSN/PFN/DSN/DID
```

Принцип умолчания при написании директив позволяет опустить любой параметр директивы. Если опускается параметр из середины или начала списка, то его место должно быть выделено запятыми.

## 3. Основные директивы языка

Все директивы входного языка делятся на описывающие, ввода/вывода, и иницирующие. В составленном для генерации программы задании необходимо описать нужную программу, а затем указать основные этапы генерации ( см. раздел I ). Последние могут осуществляться как с передачей управления пользователю после каждого этапа, так и после выполнения нескольких или всех этапов генерации.

Каждый из этапов иницируется определенной директивой языка, которые можно условно разделить на две группы.

Первая группа - директивы, выполняющие роль "справочника" и иницирующие работу информационно-обучающего раздела системы как в пакетном, так и диалоговом режимах. К этой группе относятся директивы INQUIRY и ALL .

INQUIRY (< имя пакета >) служит для получения информации о пакете программ, название которого указано в качестве параметра, в пакетном режиме.

ALL - позволяет включить информационно-обучающий режим системы генерации при диалоге. Дальнейшая работа в этом случае осуществляется по "подсказке" пользователю выбора пути решения задачи.

Второй группой иницирующих директив являются директивы, обеспечивающие комплексный подход к решению проблемы генерации. К этой группе относятся директивы GO, GENPACK, RUNPACK, EX, LIB, PFTN, COLLECT и т.п.

Описывающие директивы служат для ввода определенных конструкций языка запроса конкретного пакета программ<sup>17, 9/</sup>. К ним относятся директивы WANT, ARRAY, UNIT, CHANGE .

Директивы ввода/вывода обеспечивают подключения нужных файлов или магнитных лент с данными и запись результатов счета. К ним относятся директивы RDATA, REND, RESULT . В таблице приведены все основные директивы входного языка, их параметры и назначение.

Необходимо отметить, что использование описывающих директив и директив ввода/вывода не является обязательным при работе в системе генерации. Первые можно заменить вводом предложений языка запросов ( см. рис I ) в виде отдельного рекорда в задаче пользователя<sup>12/</sup>. Вторые - обычными управляющими операторами операционной системы ЭВМ.

Параметры некоторых директив имеют строго определенные значения. Так, для < способа задания директив языка запросов > возможны два значения: 0 - в случае использования директив входного языка, 1 - для задания предложений языка запросов в виде отдельного рекорда в задании. По умолчанию предполагается последний способ ( I ). Для < способа задания титлов > отведено четыре значения:

- 1 - титлы в библиотеке текстовых фрагментов
- 0 - титлы в задании пользователя в виде отдельного рекорда с директивами
- 2 - титлы считываются со специального файла с логическим именем TITLE , подключаемого пользователем до этапа счета
- 3 - часть титлов считывается с файла TITLESF , часть берется из библиотеки текстовых фрагментов

По умолчанию предполагается использование значения 1 .

В качестве идентификаторов перманентных файлов, имен дисков, шифров и т.п. используются любые последовательности символов, опознаваемые ОС ЭВМ CDC-6500 . По умолчанию, если не указано имя диска, подразумевается использование системного диска. Для остальных идентификаторов принцип умолчания приводит к фатальной ошибке.

Приведем несколько примеров написания директив.

GO, STATUS, 0 . иницируется весь процесс генерации для пакета STATUS по запросам, записанным обычным способом.

## СПИСОК ДИРЕКТИВ

Имя директивы	Параметры директивы	Назначение директивы
1	2	3
GO	⟨имя пакета⟩, ⟨способ задания директив языка запросов⟩	инициирует выполнение всех этапов генерации от первого до последнего без промежуточной передачи управления пользователю
GENPACK	⟨имя пакета⟩, ⟨способ задания директив языка запросов⟩	инициирует ввод описывающих директив в пакетном режиме
RUNPACK	⟨имя пакета⟩, ⟨способ задания титлов⟩	инициирует подключение файлов, сборку, трансляцию, создание библиотеки и организацию счета в пакетном режиме
GOGEN	⟨имя пакета⟩, ⟨имя перманентного файла для записи задания редактору⟩, ⟨идентификатор перманентного файла⟩	инициирует ввод описывающих директив в интерактивном режиме при ресурсах пользователя меньше 30 тыс. слов
RUN	⟨имя пакета⟩, ⟨шифр пользователя⟩, ⟨время счета⟩, ⟨шифр лаборатории⟩, ⟨имя перманентного файла с заданием редактору⟩, ⟨идентификатор перманентного файла⟩, ⟨место хранения данных для счета⟩, ⟨имя перманентного файла с данными⟩, ⟨идентификатор этого перманентного файла⟩, ⟨имя диска с данными⟩, ⟨номер диска⟩, ⟨имя перманентного файла с титлами⟩, ⟨его идентификатор⟩	инициирует подключение файлов, сборку, трансляцию, организацию библиотек и счет. Отправляет задачу пользователю на выполнение в пакетном режиме.
COLLECT	⟨имя пакета⟩	инициирует сборку программы редактором UPLATSHU.
PFTN	⟨имя пакета⟩	инициирует трансляцию собранной программы

Имя директивы	Параметры директивы	Назначение директивы
LIV	⟨имя пакета⟩, ⟨имя перманентного файла, на который записывается готовая библиотека⟩, ⟨идентификатор файла⟩, ⟨номер диска⟩, ⟨имя диска⟩	инициирует создание временной или постоянной библиотеки программы
EX	⟨имя пакета⟩, ⟨способ задания титлов⟩	инициирует счет по программе
WANT	⟨предложение языка запросов 1⟩, ⟨предложение языка запросов 2...⟩	описывает требуемую программу в терминах языка запросов
CHANGE	⟨название заменяемой константы 1⟩, ⟨ее старое значение⟩, ⟨новое значение⟩, ⟨название константы 2⟩, ⟨ее старое значение⟩, ⟨новое значение⟩...	описывает названия заменяемых констант с указанием старых и новых значений
ARRAY	⟨идентификатор массива 1⟩, ⟨длина массива 1⟩, ⟨идентификатор массива 2⟩, ⟨длина массива 2⟩...	описывает идентификаторы массивов переменной длины и их длину
UNIT	⟨тип устройства 1⟩, ⟨его номер или идентификатор⟩, ⟨тип устройства 2⟩, ⟨его номер или идентификатор⟩...	описывает необходимые внешние устройства, их типы и имена
INQUIRY	⟨имя пакета⟩	инициирует выдачу на печать информационного обеспечения пакета
ALL		инициирует включение диалогового режима в системе генерации
EXAMPL	⟨имя пакета⟩	инициирует выдачу на печать или экран терминала примера составления задания на генерацию программы в пакетном режиме
APFILE	⟨имя пакета⟩, ⟨тип подключаемых библиотек⟩	инициирует подключение библиотек как системы генерации, так и текстовых фрагментов
DICT	⟨имя пакета⟩	инициирует выдачу на печать словаря для данного пакета
RDATA RESULT REND	⟨имя ленты⟩, ⟨тип устройства⟩, ⟨тип ленты⟩, ⟨номер диска⟩, ⟨или ленты⟩, ⟨имя диска⟩, ⟨имя перманентного файла⟩, ⟨номер цикла⟩	обеспечивает ввод/вывод данных с магнитных лент или дисков

GOGEN(CONTROL, CRADLE, LCTIVA)

инициируется этап генерации задания редактору для сборки программы, которое записывается на перманентном файле с именем CRADLE и ID=LCTIVA

LIV(CONTROLLIV, MINE, 120, VANJA)

инициируется создание библиотеки программы, записываемой на диск номер I20 с именем VANJA на файл CONTROLLIV с ID=MINE

WANT(INPUT HPD, CHAMBER LUDMILA, CHECK EVENTS, DOUBLET TEST RESULTS)

описывает задание для сборки программы, позволяющей проверить события с камерн LUDMILA, обмеренные на HPD, и получить результаты проверки дублетов.

CHANGE(NMIN, 4, 6) позволяет изменить в генерируемой программе значение константы с мнемоническим именем NMIN со старого значения "4" на новое "6".

CHANGE(TAMAX, \$0.45\$, \$0.65\$)

позволяет изменить значение константы TAMAX с 0.45 на 0.65.

ARRAY(MASTER, 9)

определяет размерность массива с идентификатором MASTER равной 9 в программах пакета STATUS.

UNIT(DVSN, 10, DSN, IVANOV, PFN, NEW, YID, MINE)

подключает для работы в рамках генерируемого пакета программ STATUS диск номер 10 с именем IVANOV и файл NEW с ID=MINE на этом диске.

Диагностика ошибок при обращении к системе генерации на входном языке выдается пользователю в протоколе задачи при работе в пакетном режиме или на экран дисплея в интерактивном режиме. Ошибки в имени директивы опознаются на уровне языка SCL и в протокол выдается диагностика

SCL-137 PROCEDURE NOT FOUND

Ошибки в параметрах описывающих директив анализируются в процессе их обработки и диагностика о них выдается после окончания обработки всех директив этого типа. В протоколе в этом случае присутствует диагностика

ERROR IN REQUEST

При работе в пакетном режиме любая из ошибок ведет к прекращению дальнейшей обработки задания. В случае диалогового режима работы имеется возможность исправить директиву и продолжать работу.

Обращение к системе генерации на ЭВМ CDC-6500 имеет вид

ATTACH, P, NEWPROCLIV, ID=LCTIVA, MR=1  
LIBRARY, P.

В заключение автор считает своим приятным долгом поблагодарить

чл.корр. АН СССР Н.Н.Говоруна за поддержку работы, В.Г.Иванова за постоянный интерес, поддержку и большую помощь в работе, С.Г.Бадалян, А.Дирнера, И.И.Шелонцева, О.В.Благодарову за полезные обсуждения и ценные замечания. Особенно признателен автор А.А.Корнейчуку, сделавшему ряд конструктивных предложений.

#### Литература

1. Говорун Н.Н. и др. ОИЯИ, P10-II612, Дубна, 1978.
2. Бадалян С.Г. и др. ОИЯИ, P10-II911, Дубна, 1978.
3. Буздавина Н.А. и др. ОИЯИ, IO-II447, Дубна, 1978.
4. Абдурахимов А.У. и др. ОИЯИ, P10-80-697, Дубна, 1980.
5. Говорун Н.Н. и др. В кн.: Программирование и математические методы решения физических задач, ОИЯИ, IO-II-II264, Дубна, 1978.
6. Степаненко В.А. ОИЯИ, P10-II792, Дубна, 1978; ОИЯИ, P10-II793, Дубна, 1978. Воеводин В.В. и др. В кн.: Вычислительные методы и программирование, Вып. 22, М., Изд-во Моск. ун-та, 1974.
7. Klein H. and Zoll J., PATSCHY-4, CERN, Program Library, Geneva, 1977.
8. Степаненко В.А. В кн.: Программирование и математические методы решения физических задач, ОИЯИ, IO-II-II264, Дубна, 1978. Баула В.Г. В кн.: Вычислительные методы и программирование, Вып. 29, М., Изд-во Моск. ун-та, 1979. Говорун Н.Н. и др. ОИЯИ, IO-II052, Дубна, 1977. Нефедьева Л.С. и др. ОИЯИ, P10-80-861, Дубна, 1980.
9. Вишняков В.В. и др. ОИЯИ, P10-I2094, Дубна, 1979.
10. Бадалян С.Г. и др. ОИЯИ, IO-80-664, Дубна, 1980.
11. NOS/BE-1 Reference Manual, Pub.No60493800, Control Data Corporation, St. Paul, Minnesota, USA.
12. Хигман Б. Сравнительное изучение языков программирования, "МИР", М., 1974.
13. Belonogov A.V. et al. Nuclear Instrum. and Methods, 1963, 20, p. 114; Богуславский Н.В. и др. ОИЯИ, I3-4466, Дубна, 1969.

Рукопись поступила в издательский отдел  
29 мая 1981 года.