

Ц 8406

K-984

3120/2-76

СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

ДУБНА



9/vii-76

10 - 9764

Х.Кэниг, Ю.Лэтц, Л.С.Нефедьева, Г.Штиллер

ИСПОЛЬЗОВАНИЕ СИСТЕМЫ DEROT
ДЛЯ СОЗДАНИЯ ТРАНСЛЯТОРА С ЯЗЫКА СОС

1976

10 - 9764

Х.Кэниг, Ю.Лэтцч, Л.С.Нефедьева, Г.Штиллер

**ИСПОЛЬЗОВАНИЕ СИСТЕМЫ DEROT
ДЛЯ СОЗДАНИЯ ТРАНСЛЯТОРА С ЯЗЫКА СОС**

Быстрый рост разнообразного числа электронных вычислительных машин, применяемых для научных и инженерных расчетов, и еще более быстрый рост числа работников, использующих эти машины в своей научной и практической деятельности, привел к созданию и внедрению большого числа языков программирования (более 700 за последние 20 лет).

Очевидно, что такой широкий спектр языков необозрим. В связи с этим встает вопрос о их дальнейшем развитии. Решение вопроса может быть достигнуто путем систематизации такого рода языков и выработки общих принципов их построения. Возникает естественная необходимость в разработке методических основ для работы с языками программирования, в нахождении разумных связей между ними. Решение всех этих вопросов особо важное значение имеет для совершенствования технологии строения трансляторов, для применения языков программирования и для обучения этим языкам.

I. Взаимоотношение универсальных и специализированных

Я З Ы К О В

В настоящее время четко вырисовываются два направления в развитии языков программирования:

1. Интеграция – создание универсальных языков с широким аспектом применения.

2. Дифференциация – создание специализированных языков с проблемно-ориентированным применением.

Универсальные языки имеют большие возможности, но зато требуют больших затрат на их описание, реализацию и изучение. В случае применения этих языков в специальных областях, с одной стороны, используются не все возможности, а с другой – они часто не позволяют коротко и ясно выразить специфику проблемы. В этих случаях целесообразнее использовать специализированные языки /1-4/.

Специализированные языки учитывают терминологию специальности, ее ориентацию, привычный образ мыслей специалистов. Комплекс всего этого облегчает изучение языка, позволяет писать короткие и обозримые программы, что, в свою очередь, уменьшает вероятность ошибок.

Исходя из вышесказанного, может создаться впечатление, что оба направления противоречивы, но это не так. Оба направления являются только двумя различными тесно связанными между собой методическими аспектами развития /1,2/.

1. Обобщение и систематизация различных концепций языков программирования с целью их стандартизации.

2. Проблемно-ориентированная специализация общих концепций языков программирования.

В специализированные языки, помимо специализированных операторов, как правило, надо вводить ряд операторов общего назначения, например, циклы, операторы описания данных и т.д., т.е. набор операторов, который обычно хорошо представлен в универсальных языках. Стремление разработать оригинальный специализированный

язык повлекло за собой создание большого количества синтаксических и семантических определений для однотипных операторов. Поэтому целесообразно приблизить эти понятия к аналогичным в универсальных языках.

К разработке специализированных языков имеются разные подходы. Можно, например, разработать специализированный язык, который наряду со своими специализированными элементами использует некоторый набор элементов универсального языка /7/. Другой подход – "расширение" универсального языка новыми специализированными элементами при использовании средств универсального языка /2/. Такую возможность предоставляет оператор описания новых операций в языке АЛГОЛ-68 /8/.

Последний подход позволяет "стыковаться" языкам из одного "семейства", поддерживать коммуникацию между специализированными направлениями и упрощает реализацию трансляторов. Но, с другой стороны, очевидно, что качество (ясность, простота, структура) языка зависит от качества универсального языка, на котором основывается это "семейство" специализированных языков. Отсюда следует, что универсальные языки должны иметь хорошо развитую структуру /3,6/.

2. О реализации специализированных языков

Любой из подходов к разработке специализированного языка требует создания транслятора. Обилие специализированных языков и ЭВМ диктует создание большого количества трансляторов. Это и повлекло за собой идею разработки методов автоматизированного создания трансляторов. Примером такого подхода является система **DEPOT**, разработанная и созданная в секции "Математика" Технического университета города Дрездена (ГДР). Она эксплуатируется два года на ЭВМ БЭСМ-6, и в настоящее время создается ее вариант на ЭВМ типа ЕС /5,6/.

Система использует метаязык **DEPOT-MS** (см. приложение I), на котором описываются основные правила трансляции элементов специализированного языка в элементы языка высокого уровня, принятого на данной ЭВМ. Транслятор системы **DEPOT**, основываясь на синтаксическом описании специализированного языка, создает транслятор перевода с этого языка на язык высокого уровня (см. приложение 2). Этот транслятор хранится потом в личной библиотеке системы **DEPOT**.

Каждое задание, написанное на специализированном языке, вызывает этот транслятор, который после перевода задания вызывает транслятор с языка высокого уровня (см. приложение 3). Система **DEPOT**, таким образом, дает возможность сравнительно простого и быстрого способа создания трансляторов для специализированных языков без привлечения специалистов по созданию трансляторов.

Создание транслятора описываемой системой **DEPOT** для данного специализированного языка упрощается, если выполняются следующие условия:

1. На данной ЭВМ существует универсальный язык, достаточно мощный для описания значений вновь созданных проблемно-ориентированных элементов.

2. Специализированный язык использует подобные универсальному языку элементы.

3. Об языке общения системы СОС

Модульная система обработки спектров (СОС)^{/7/} разработана в ЛВТА ОИЯИ на ЭВМ БЭСМ-6. При разработке системы преследовались следующие цели:

1. Создание библиотеки программных модулей для обработки спектров.

2. Организация приема, накопления и обработки экспериментальных данных.

Система состоит из 4-х частей:

1. Библиотека программных модулей.

2. Набор программ для организации работы с файлами^{/9/} (понятие "файл" было введено для эффективного решения организации работы с массивами данных).

3. Набор программ для обмена данными между БЭСМ-6 и другими ЭВМ ОИЯИ с помощью магнитных лент.

4. Язык общения.

Язык общения СОС был введен, чтобы дать пользователю удобный аппарат общения с системой. С помощью языка осуществляется обращение к программным модулям и организуется работа с файлами.

Решение проблемы общения между системой и пользователем с помощью специального языка не является обязательным для такого рода систем. Другое решение, например, - это обращение к системе на языке высокого уровня (ФОРТРАН, АЛГОЛ и т.д.). Такой подход потребовал бы от пользователя писать в нашем случае много дополнительной информации для системы, что, естественно, влечет за собой ошибки.

Язык общения СОС имеет фортраноподобный вид. Операторы по своей структуре разделяются на два типа: операторы для организации обработки и операторы для организации работы с файлами.

Операторы организации обработки:

PROGRAM - оператор начала задания,

CALL - оператор обращения к программным модулям,

DO - оператор цикла,
COMMON - оператор описания общих блоков задания,
END - оператор окончания задания.

Эти операторы взяты почти без изменений из ФОРТРАНА, потому что их семантика в языке СОС немногим отличается от семантики в ФОРТРАНе.

Операторы второй группы являются специализированными операторами для работы с файлами. Они описаны новыми языковыми элементами. К этой группе принадлежат следующие операторы:

FILE - оператор описания файлов,
CREATE - оператор организации новых файлов,
DELETE - оператор исключения файлов,
RENAME - оператор переименования файлов,
WRITE - оператор обмена файлами,
ADD - оператор составления массивов данных.

4. О реализации языка общения СОС с помощью DEPOT

Для языка общения СОС в Дубне создается самостоятельный транслятор на язык ФОРТРАН.

Кроме того, была произведена в Техническом университете города Дрездена реализация языка СОС с помощью DEPOT (см. приложение 2 и 3). В качестве языка высокого уровня был выбран язык ФОРТРАН на ЭВМ БЭСМ-6 (примеры перевода СОС в ФОРТРАН=см. приложение 4 и 5).

Основная цель проделанной работы - это накопить опыт для работы с системой DEPOT и показать эффективность и целесообразность такой методики.

Реализация языка системы СОС с помощью системы DEPOT была произведена за 60 часов (для создания дубненского варианта транс-

лятора необходимо четыре месяца/человека). Сюда вошло время изучения языка СОС, описание его синтаксиса и семантики через язык DEPOT-MS, транслирование. Не вошло время на изучение самого языка DEPOT-MS и языка ФОРТРАН.

Время на ЭВМ для создания транслятора в системе DEPOT составило 3 минуты коммерческого времени.

Время транслирования задания через транслятор в системе DEPOT - 6960мс (см. приложение 4) и 5660мс (см. приложение 5).

Безусловно, это время больше времени транслирования задания через транслятор, который переводит специализированный язык в машинный код. Но при этом надо учитывать следующие аспекты:

1. Задания, составляемые пользователями для работы с системами, аналогичными СОС, как правило, короткие, и поэтому время трансляции и выполнения их несущественно.

2. Использование системы DEPOT позволяет быстро получать трансляторы для специализированных языков, что, в свою очередь, дает возможность проводить оперативно дальнейшие усовершенствования и расширения языка.

3. Наличие системы DEPOT на различных ЭВМ обеспечивает возможность быстрого и простого перехода с одной машины на другую.

С этих позиций системы типа DEPOT можно рассматривать как удобный вспомогательный аппарат для создания трансляторов с различных специализированных языков.

Заключение

Авторы в данной работе старались показать пути подхода к вопросам создания специализированных языков и их трансляторов. При этом они хотели подчеркнуть значение создания методических разработок в этой области.

ПРИЛОЖЕНИЕ 1

Пример описания трансляции специализированного языка и языка ФОРТРАН с помощью метаязыка DEPOТ-MS.

Оператор DELETE

В языке COC: <delete statement> ::=DELETE <list of files>
(<tape>)|DELETE <list of files>

На ФОРТРАНе: CALL CDELET(<list of files>, <number of files>, <tape>)
или CALL CDEL (<list of files>, <number>)

Пример: в языке COC : DELETE X,Y (TAPE1)

на ФОРТРАНе: CALL CDELET(X,Y,2,TAPE1)

```
'COMMENT' -----D E L E T E S T -----
DELETETEST=
  <DELETE>
  L:='SERIE'('UNITE' 'LIST' FILES 'WITH' IDENTIFIER,<,>)
  [<(> 'UNITE' 'LIST' TAPES 'WITH' P2:=-IDENTIFIER <)>] 'OPTION'
  'ON' 'ERROR' [' PRINT' '-E-05-' [ 'NOT'<;>] ** [0:100] ]
'GENERATE'
  < CALL >
  'CASE' 0+1 {CDEL(>,<CDELET(>)
  L <,> N [<,>P2] 'OPTION' <)>.
```

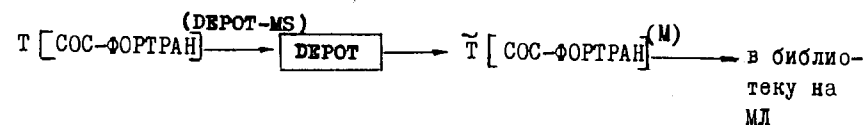
Описывается подпрограмма DELETETEST, которая вызывает транслятор СОС-ФОРТРАН, когда он встречает ключевое слово DELETE. Список файлов присваивается списковой переменной L. Кроме того, идентификаторы файлов записываются в общий список файлов. P2 получает идентификатор ленты, который тоже регистрируется в списке лент. Выражение [] 'OPTION' указывает, что идентификатор ленты может отсутствовать.

Оператор 'GENERATE' описывает текст, который генерируется. Сначала CALL, потом название подпрограммы в зависи-

мости от описания магнитной ленты, далее-список файлов L, число файлов и в конце-идентификатор ленты P2, если он есть. Между этими элементами генерируются нужные запятые и скобки.

ПРИЛОЖЕНИЕ 2

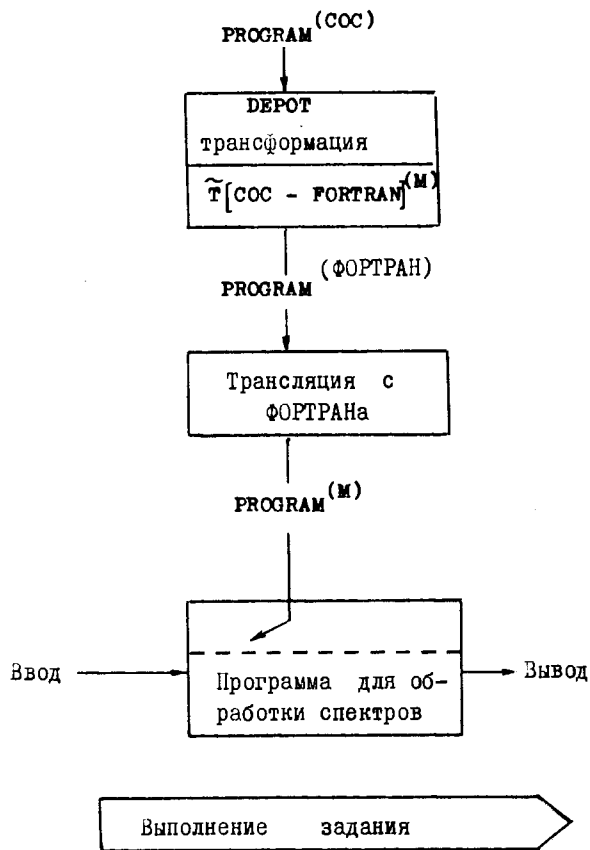
ГЕНЕРАЦИЯ ПРЕДВАРИТЕЛЬНОГО ТРАНСЛЯТОРА СОС-ФОРТРАН



Программа T [COC-FORTRAN] (DEPOТ-MS), написанная в метаязыке DEPOТ-MS, описывает основные правила трансляции элементов языка общения СОС в элементы языка ФОРТРАН. Система DEPOТ генерирует предварительный транслятор T [COC-FORTRAN] (M), который содержит следующие составные части:

- программа для лексического анализа;
- программа для синтаксического анализа;
- программа для генерации текста на языке ФОРТРАН.

Этот транслятор хранится в личной библиотеке на магнитной ленте.



ПРИЛОЖЕНИЕ 3. Схема трансляции COB-FOOTPAH-машинный код.

Трансформация

Трансляция

ПРИЛОЖЕНИЕ 4: пример трансформации COB-FOOTPAH

а) задание, написанное на языке общения COB.

```

!SPECIAL PROGRAM
!SPECIAL PROGRAM
TIME PASS1 START:          129660 MS
    1.PROGRAM TEST1;
    2.COMMON/COC2/COC2(1024);
    3.FILE FON,RESULT(INPUT,TAPE01);
    4.FILE PARA(INPUT,(15,18,I10));
    5.FILE SPECT(OUTPUT,TAPE01);
    6.FILE HELP(WORK);
    7.CREATE SPECTR(TIME,TAPE01)=SPECT1,SPECT2,SPECT3;
    8.WRITE SPECTR(1),SPECTR(11),SPECTR(21) INTO HELP;
    9.DO K=1,3;
    10. DO J=2,10;
    11. SUMME(SPECTR(K+J-1),HELP(K),HELP(K));
    12. OD;
    13.OD;
    14.SUMME(HELP(1),HELP(2),HELP(1));
    15.SUMME(HELP(1),HELP(3),HELP(1));
    16.CUTFON(HELP(1),10,1000,FON);
    17.FIT(HELP(1),PARA,HELP(2),RESULT(5));
    18.WRITE HELP(1),HELP(2) INTO SPECT;
    19.DELETE GAMMA,GAM,GAMPON(TAPE02);
    20.END

! END OF SPECIAL PROGRAM

TIME PASS1 END   :          131600 MS
TIME PASS2 START:          133120
TIME PASS2 END   :          135540
TIME PASS 3 START:          136700
TIME PASS 3 END   :          137120
  
```


б) задание после трансформации СОС-ФОРТРАН

```
PROGRAM TEST1
INTEGER COC12
COMMON/COC1/COC1(4096)/COC2/COC2(1024)
COMMON/COC12/COC12(2)
COMMON//FON,
1      RESULT,
1      SPECT,

1      SPECTR,
1      SPECT1,
1      SPECT2,
1      SPECT3,
1      GAMMA,
1      GAM,
1      GAMFON,
1      TAPE01,
1      TAPE02,
1      CCCCCC

COC12(1)=4096
COC12(2)=1024
FON=6HFON
RESULT=6HRESULT
SPECT=6HSPECT
SPECTR=6HSPECTR
SPECT1=6HSPECT1
SPECT2=6HSPECT2
SPECT3=6HSPECT3
GAMM=6HGAMMA
GAM=6HGAM
GAMFON=6HGAMFON
TAPE01=6HTAPE01
TAPE02=6HTAPE02
CALL CLOAD
CALL CAT
CALL CINPUT(FON,RESULT,2,TAPE01)
CALL CFCARI(PARA,
159H(I5,I8,I10)
2 )
```

```
CALL COUT(SPECT,1,TAPE01)
CALL CFWORK(HELP,1)
CALL CREATT($SPECT3,SPECT2,SPECT1,3,TAPE01,SPECTR)
CALL CFTOF(21,SPECTR,11,SPECTR,1,SPECTR,6,HELP)
DO 9002 K=1,3
DO 9001 J=2,10
CALL SUMME(SPECTR,K+J-1,HELP,K,HELP,K)
9001 CONTINUE
9002 CONTINUE
CALL SUMME(HELP,1,HELP,2,HELP,1)
CALL SUMME(HELP,1,HELP,3,HELP,1)
CALL CUTFON(HELP,1,10,1000,FON)
CALL FIT(HELP,1,PARA,HELP,2,RESULT,5)
CALL CFTOF(2,HELP,1,HELP,4,SPECT)
CALL CDELET(GAMMA,GAM,GAMFON,3,TAPE02)
CALL COCEND
END
```

ПРИЛОЖЕНИЕ 5: Пример трансформации СОС-ФОРТРАН.

а) задание, написанное на языке общения СОС.

!SPECIAL PROGRAM

!SPECIAL PROGRAM

TIME PASS1 START:

```
1.PROGRAM TEST2;
2.FILE GAM1,PEAK(UPDATE,TAPE01);
3.FILE GAMMA(WORK);
4.FILE GAM3(WORK,1024);
5.CREATE GAM2(INPUT,TAPE01)=ABC,AA0001,AA0002;
6.DO J=1,3;
7. CMULT(GAM1(J),0.0334,GAMMA(J));
8. SUMME(GAMMA(J),GAM2(J),GAMMA(J));
9. SELECT(GAMMA(J),50,100,GAM3);
10.OD;
11.ADD PEAK TO GAM3;
12.DELETE PEAK;
13.WRITE GAMMA,GAM3 INTO GAM1(1);
```

14.END

END OF SPECIAL PROGRAM
TIME PASS1 END : 140780 MS
TIME PASS2 START: 142000
TIME PASS2 END : 144000
TIME PASS 3 START: 145040
TIME PASS 3 END : 145360

6) задание после трансформации СОС - ФОРТРАН

```
PROGRAM TEST2
INTEGER COC12
COMMON/COC1/COC1(4096)/COC2/COC2(4096)
COMMON/COC12/COC12(2)
COMMON//GAM1,
1      PEAK,
1      GAM2,
1      ABC,
1      AA0001,
1      AA0002,
1      TAPE01,
1      CCCCC

COC12(1)=4096
COC12(2)=4096
GAM1=6HGAM1
PEAK=6HPEAK
GAM2=6HGAM2
ABC=6HABC
AA0001=6HAA0001
AA0002=6HAA0002
TAPE01=6HTAPE01
CALL CLOAD
CALL CAT
CALL CUPDAT(GAM1,PEAK,2,TAPE01)
CALL CFWORK(GAMMA,1)
CALL CBWORK(GAM3,1024,1)
CALL CREATI(AA0002,AA0001,ABC,3,TAPE01,GAM2)
DO 9001 J=1,3
```

```
CALL CMULT(GAM1,J,0.0334,GAMMA,J)
CALL SUMME(GAMMA,J,GAM2,J,GAMMA,J)
CALL SELECT(GAMMA,J,50,100,GAM3)
9001 CONTINUE
CALL COADD(PEAK,1,GAM3)
CALL CDEL(PEAK,1)
CALL CFTOP(GAM3,GAMMA,2,GAM1,1)
CALL COCEND
END
```

Замечание: OD - является признаком окончания цикла.

ЛИТЕРАТУРА

1. N.J.Lehmann. Probleme und Bedeutung von Programmiersprachen Elektronische Informationsverarbeitung und Kybernetik 11(1975), 4-6, S.200-206.
2. N.J.Lehmann, G.Stiller. Einige methodische Aspekte der Entwicklung und Nutzung höherer Programmiersprachen, Rechentechnik/Datenverarbeitung (im Druck).
3. N.Wirth. On design of programming languages, Proc. IFIP-Congress 1974, Stockholm, S.386-393.
4. H.Zemanek. Semiotics and Programming Languages, Com.ACM 9(1966), 3, S.139-143.
5. J.Bormann, J.Litzsch. Definition und Realisierung von Fachsprachen mit DEPOT. Dissertation, TU Dresden, 1974.

6. J. Lützech. Eine (realisierte) Metasprache zur Definition von Syntax und Semantik von Fachsprachen. Elektronische Informationsverarbeitung und Kybernetik 11(1975)4-6, S.360-370.
7. Л.С.Нефедьева, Х.Кэниг, Т.С.Рерих, А.И.Салтыков, В.Н.Ягафорова. Автоматизированная система обработки спектров (СОС) на машине БЭСМ-6. Труды всесоюзного совещания "Системы автоматизации научных исследований", Рига, 1975.
8. С.Н.Lindsey, S.G. van der Meulen. Informal introduction to Algol 68, North-Holland Publishing Company-Amsterdam, 1971.
9. Х.Кэниг, Л.С.Нефедьева. ОИЯИ, IO-8556, Дубна, 1975.

Рукопись поступила в издательский отдел
5 мая 1976 года.