

K-238

ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

ДУБНА



10/5-77

1797/2-77

P11 - 10440

А.А.Карлов, Т.Ф.Смолякова

ОРГАНИЗАЦИЯ ДИАЛОГА
С МНОГОУРОВНЕВОЙ СТРУКТУРОЙ
В СИСТЕМЕ "ЧЕЛОВЕК-ЭВМ"
НА ОСНОВЕ УНИФИЦИРОВАННОГО МЕХАНИЗМА
ОБРАБОТКИ СООБЩЕНИЙ

1977

P11 - 10440

А.А.Карлов, Т.Ф.Смолякова

ОРГАНИЗАЦИЯ ДИАЛОГА
С МНОГОУРОВНЕВОЙ СТРУКТУРОЙ
В СИСТЕМЕ "ЧЕЛОВЕК-ЭВМ"
НА ОСНОВЕ УНИФИЦИРОВАННОГО МЕХАНИЗМА
ОБРАБОТКИ СООБЩЕНИЙ

*Направлено на 2-ю Венгерскую конференцию
по вычислительной науке /Будапешт, 1977/*



Карлов А.А., Смолякова Т.Ф.

P11 - 10440

Организация диалога с многоуровневой структурой в системе "Человек - ЭВМ" на основе унифицированного механизма обработки сообщений

Рассматривается унифицированный механизм для обработки сообщений с удаленной дисплейной станции ЭВМ БЭСМ-6, а также возможности реализации на его основе диалога человек - ЭВМ, имеющего многоуровневую структуру.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна 1977

Karlov A.A., Smolyakova T.F.

P11 - 10440

The Organization of a Message-Oriented Multilevel Dialog for Interactive Systems

General-purpose set of programs to handle messages from remote interactive display station and the possibilities available for users to organize a dialog with multilevel structure are considered.

Preprint of the Joint Institute for Nuclear Research. Dubna 1977

1. НАЗНАЧЕНИЕ И ВОЗМОЖНОСТИ

Одной из задач, возникающих при создании математического обеспечения любой диалоговой системы, является задача обработки сообщений, поступающих с терминала, а также организация выполнения тех или иных действий в ответ на каждое сообщение.

В Объединенном институте ядерных исследований на базе малой ЭВМ М-6000 /16К 16-разрядных слов/ создана удаленная дисплейная станция /УДС/^{1/}, обеспечивающая пользователям в режиме диалога доступ через линии связи к БЭСМ-6 центрального вычислительного комплекса Института^{2/}. Дисплейная станция, кроме оборудования для работы с перфоленкой, телетайпа, накопителя на магнитной ленте и символьного дисплея "Видеотон-340", имеет в своем составе графический дисплей СИГДА с клавиатурой и световым карандашом^{3/}. Основным режим работы УДС - это так называемый режим "черного ящика", когда на дисплейную станцию возлагается фиксированный набор функциональных возможностей, доступных программе БЭСМ-6 на уровне обращений к соответствующим подпрограммам^{4/}. Основное средство диалога со стороны человека - клавиатура графического дисплея; световой карандаш используется значительно реже.

В дополнение к библиотеке дисплейных подпрограмм, предназначенной для формирования, представления и обработки сложной графической информации, а также для управления УДС, на БЭСМ-6 ОИЯИ создан стандартный

/унифицированный/ механизм обработки сообщений, реализованный в виде набора подпрограмм, доступных пользователю на языке ФОРТРАН. Его назначение - предоставить программисту гибкие и естественные в использовании программные средства для организации диалога с УДС.

Преимуществами предложенного механизма являются:

1. Возможность организации диалога с многоуровневой структурой.
2. Наличие системных средств для изменения структуры диалога в процессе самого диалога.
3. Отсутствие необходимости в программировании для сообщений: запросов на ввод, операций обмена, синтаксического и, частично, семантического контроля.
4. Наличие аппарата стандартных диагностических сообщений, сопровождающих процесс диалога.
5. Наличие системных средств для выполнения стандартных процедур по управлению прохождением задач на центральной ЭВМ /перезапуск задачи, прекращение счета, хранение "истории" диалога с возможностью возврата на предыдущие его этапы и т.д./.

При описании структуры диалога пользователь в соответствии с требованиями конкретной задачи выбирает число уровней в структуре, число состояний на каждом уровне, для каждого состояния указывает связи между сообщениями /которые он выбирает по собственному усмотрению/ и исполнительными подпрограммами, а также описывает типы параметров этих подпрограмм.

В данном случае состояние диалога в некоторый момент времени рассматривается, в ограниченном смысле, как набор допустимых в этот момент сообщений, а под числом уровней понимается максимально возможное число последовательно вложенных друг в друга циклов ожидания сообщений.

Структура диалога может быть задана программистом статически или динамически. При статическом задании она описывается в начале работы /например, с помощью отдельной подпрограммы/. При динамическом задании новый набор допустимых сообщений и соответствие между ними и исполнительными подпрограммами задается в той подпрограмме, в которой происходит из-

менение состояния, благодаря чему, в частности, появляется возможность изменять структуру диалога в зависимости от промежуточных результатов работы, и требования к памяти, необходимой для описания структуры, уменьшаются.

Преимуществами статического задания является наглядность, удобство внесения изменений и дополнений, так как описание структуры может быть локализовано в одной или нескольких подпрограммах, специально предназначенных для этой цели. Естественно, полное описание структуры несколько увеличивает требования к объему памяти.

Возможность динамического описания структуры, кроме того, что она дает упомянутые выше преимущества, является еще и принципиально необходимой, так как позволяет подключать к программе пользователя специализированные диалоговые подсистемы обработки и анализа данных со своей собственной структурой диалога. Например, подсистема анализа функций двух переменных по отношению к программе пользователя может иметь свою собственную многоуровневую структуру диалога, которая по отношению к структуре диалога основной программы пользователя будет задаваться динамически при вхождении в подсистему.

2. СООБЩЕНИЯ

При работе с унифицированным механизмом в качестве сообщения рассматривается последовательность символов, состоящая из приказа и его параметров, если таковые имеются. Приказы делятся на системные и приказы пользователя. Системные приказы предназначены для выполнения стандартных, не зависящих от решаемой задачи, действий, и отличаются наличием специального символа (/) в первой позиции.

В качестве параметров сообщений могут быть заданы идентификаторы, числа и текстовые константы /как они определены в описании языка ФОРТРАН/.

Примеры системных сообщений

- /A, NAME - загрузка и выполнение подпрограммы /без параметров/с именем NAME;
- /S - перезапуск задачи пользователя;
- /R - возврат на предыдущий уровень в программе пользователя;
- /I - возврат на начальный уровень в программе пользователя;
- /P - задание контрольной распечатки при обмене информацией между БЭСМ-6 и УДС;
- /F - прекращение задачи пользователя и т.п.

Примеры возможных сообщений пользователя:

- DATA,3 - выдать на экран УДС для редактирования третью группу начальных параметров;
- GO - перейти на счет по основной программе;
- A,5,1,4HF8.4 - высветить на экране значение переменной A(5,1) в формате F8.4.

3. ТАБЛИЦА СВЯЗИ ПОДПРОГРАММ

По приказам пользователя с помощью рассматриваемого механизма осуществляется вызов указанных подпрограмм.

Набор допустимых в данный момент сообщений /приказов/ определяет текущее состояние диалога. Соответствие между отдельным сообщением и требуемой подпрограммой пользователь устанавливает через специальную таблицу связи подпрограмм /ТСП/ при описании структуры диалога. Таким образом, каждое состояние характеризуется своей ТСП.

Каждому приказу, допустимому в данном состоянии, в ТСП соответствует состоящий из трех слов элемент, первое слово которого содержит наименование /имя/ приказа, второй - имя исполнительной подпрограммы, а третье - описатель типа параметров исполнительной

подпрограммы. Последний необходим для обеспечения контроля за правильностью числа и типа присутствующих в сообщении параметров с тем, чтобы избежать в процессе диалога фатальных ситуаций, которые могли бы возникнуть при попытке некорректного /в смысле параметров/ обращения к исполнительной подпрограмме. Как и первые два слова элемента, описатель представляет собой шестисимвольную текстовую константу, в которой каждый символ слева направо описывает тип параметра соответственно с первого по /максимум/ шестой. Так, например, символ I используется для обозначения целого числа, символ R - для вещественного числа, символ V - для обозначения переменной, символ H - для текстовой константы и т.п. Нулевое значение описателя указывает на отсутствие параметров.

Примеры описателей типа параметров:

- IR - два параметра, первый из которых - целое число, второй - вещественное;
- VIR - три параметра, первый из которых - переменная /идентификатор/, второй - целое, третий - вещественное число;
- HIRRR - пять параметров, первый из которых - текстовая константа, второй - целое число, а третий, четвертый и пятый - вещественные числа.

Таким образом, если, например, в некотором состоянии по приказу XC пользователь должен организовать вызов подпрограммы XCROSS, для которой в качестве параметров в сообщении должны задаваться два целых числа, то в соответствующей этому состоянию ТСП должен находиться элемент из трех слов:

XC , XCROSS, H .

Считается, что отдельный элемент ТСП имеет имя, совпадающее с приказом.

Пользователь сам определяет размер ТСП для каждого состояния в зависимости от числа выбранных им

для данного состояния приказов. При статическом задании структуры диалога он должен использовать под таблицы связи специально выделенные им для этой цели массивы. В случае динамического задания структуры, кроме специальных массивов, возможно также использование тех рабочих массивов основной программы, которые свободны на текущий момент.

4. КАТАЛОГ ТАБЛИЦ СВЯЗИ

Как уже указывалось выше, при статическом задании структуры диалога пользователь описывает структуру в начале работы. Это задание состоит в формировании нескольких ТСП в зависимости от числа состояний в структуре и в установлении соответствия между каждой ТСП и приказом, по которому она должна стать текущей. Фактически в данном случае каждой ТСП присваивается имя, совпадающее с приказом, а на механизм обработки сообщений возлагается процедура объявления по заданному приказу текущей ТСП из числа сформированных ранее.

Таким образом, при формировании "поименованной" ТСП пользователь, кроме указания массива, выделяемого для данной ТСП, сообщает имя приказа, по которому она должна стать текущей. Для учета "поименованных" ТСП формируется каталог, в котором каждой ТСП соответствует элемент из трех слов, содержащий информацию о ее адресе и длине, а также о соответствующем ей приказе.

5. НАЗНАЧЕНИЕ ПОДПРОГРАММ

Для описания структуры диалога и организации взаимодействия на уровне сообщений унифицированный механизм предоставляет пользователю набор подпрограмм, основными из которых являются следующие:

BSLTB(IBUF, LB) - в качестве текущей ТСП задается массив **IBUF** длиной **LB** с предварительной очисткой

/обнулением/ массива; заполнение массива **IBUF** может быть выполнено с помощью операторов присваивания /например, в виде **IBUF(1)=2HXC**, **IBUF(2)=6HXCROSS**, **IBUF(3)=2HII**,... и т.д./ или с помощью подпрограммы **BSAMES**.

BSLT(ITAB,LT) - в качестве текущей ТСП задается массив **ITAB** длиной **LT** с сохранением содержимого массива; этой подпрограммой удобно пользоваться, когда содержимое массива уже определено, например, через оператор **DATA**.

BSAMES(P,SP,IT) - из заданных параметров формируется элемент из трех слов и добавляется в текущую ТСП на свободное место; все три параметра - текстовые константы, первая из которых представляет собой имя приказа, вторая - имя исполнительной подпрограммы, а третья - описатель типа параметров.

BSDMES(P) - из текущей ТСП удаляется элемент, соответствующий заданному приказу (**P**).

BSCAT(ICAT,LC) - массив **ICAT** длиной **LC** объявляется в качестве каталога ТСП при статическом описании структуры диалога.

BSLTC(ITAB,LT,N) - массив **ITAB** длиной **LT** регистрируется в каталоге как ТСП с именем **N**.

BSWAIT - обеспечивает выдачу запроса на прием сообщения, цикл ожидания сообщения на текущем уровне, анализ принятого сообщения на корректность и передачу управления на исполнительную подпрограмму в соответствии с текущей ТСП; выход из данной подпрограммы выполняется по специальному системному приказу "Возврат на предыдущий уровень" (**/R**), при этом автоматически восстанавливается ТСП, которая соответствовала состоянию на предыдущем уровне и информация о которой сохранялась в качестве "истории диалога" в специальном системном стеке ТСП.

6. ПРИМЕР ИСПОЛЬЗОВАНИЯ

Пусть перед программистом стоит задача, в которой необходимо подобрать набор начальных данных с тем, чтобы получить результат вычислений, отвечающий заданным критериям. В качестве нулевого приближения могут использоваться, например, данные, подготовленные на перфокартах, в дальнейшем в процессе диалога требуется их редактирование на основании анализа результатов вычислений. Вычисления проводятся по некоторому алгоритму, который может быть реализован в виде отдельной подпрограммы /или группы подпрограмм/. Анализ результатов на соответствие заданным критериям может проводиться с использованием всех возможностей дисплейной библиотеки по представлению и анализу графической информации.

Предположим, что пользователь выбрал логику работы, изображенную в виде блок-схемы на рис. 1. Здесь структура диалога имеет три уровня.

После ввода начальных данных следует цикл ожидания сообщения /уровень 1/. Допустим, что в этот момент в качестве допустимых пользователь выбрал два сообщения:

T,k - переход на подпрограмму представления на экране и редактирования k-той группы начальных данных /п/п EDIT(K) /;

G - переход на счет по основной программе /п/п GO /.

После выполнения вычислений по основной программе на первом уровне к числу допустимых должно быть добавлено сообщение:

A,i - переход на подпрограмму анализа по i-тому критерию /п/п ANAL (I) /.

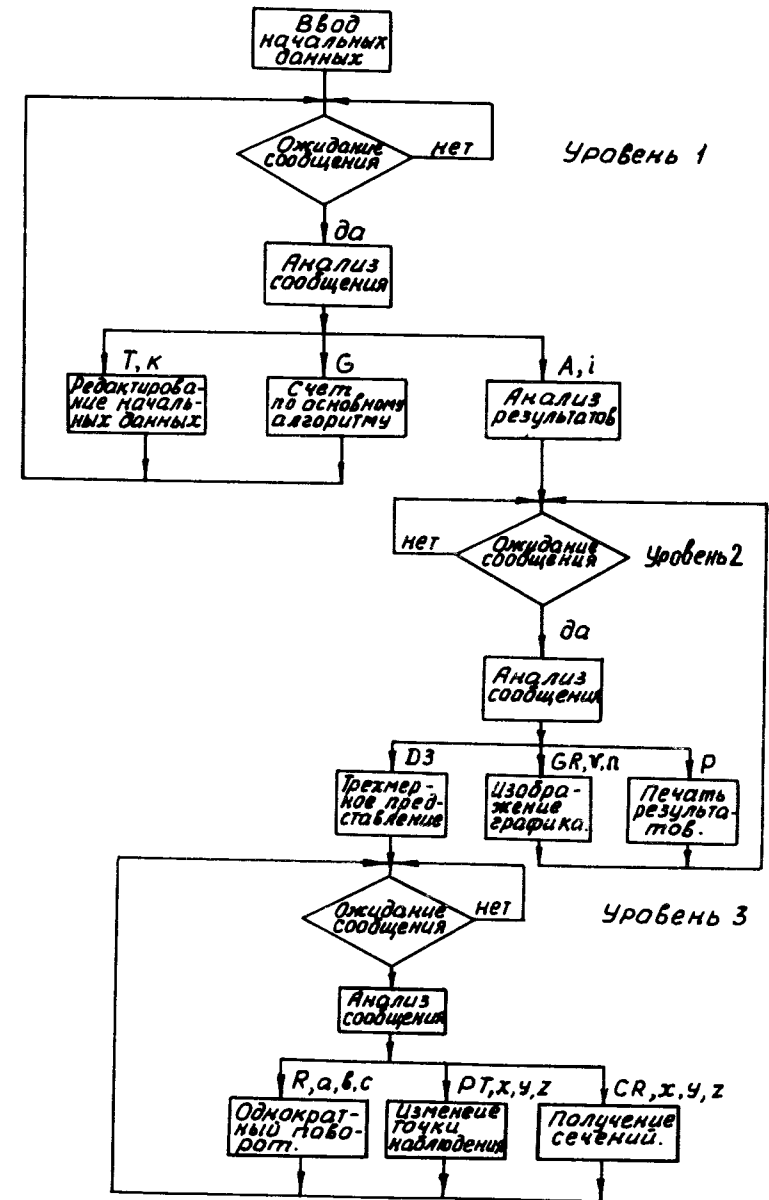


Рис. 1

Для организации диалога на этом уровне пользователь может выполнить, например, следующие действия в своей программе /подпрограмме/:

```
SUBROUTINE INIT
:
DIMENSION TAB (9)
:
CALL BSLTB (TAB,9)
CALL BSAMES (1HT, 4HEDIT, 1HI)
CALL BSAMES (1HG, 2HGO, ØB)
CALL BSWAIT
:
END
```

В подпрограмме GO после выполнения первоначального вычисления следует предусмотреть вызов:

```
CALL BSAMESS (1HA, 4HANAL, 1HI) ,
```

по которому в текущую ТСП будет добавлен элемент связи с подпрограммой анализа.

В свою очередь, в подпрограмме анализа результатов (ANAL) пользователь считает необходимым организовать новый цикл ожидания /уровень 2/, из которого затем он будет осуществлять вызов исполнительных подпрограмм по следующим сообщениям:

D3 - трехмерное представление результатов /п/п DISP3 /;
GR,v,n - построение графика функции /п/п GRAPH(V,N) /;
P - печать промежуточных результатов /п/п PRT/.

Для этой цели в подпрограмме ANAL следует описать новую ТСП и перейти на цикл ожидания /п/п BSWAIT /, например:

```
SUBROUTINE ANAL (I)
:
DIMENSION T(9)
DATA ( (T(J),J=1,9)=2HD3, 5HDISP3, ØB, 2HGR,
* 5HGRAPH, 2HVI, 1HP, 3HPRT, ØB)
:
CALL BSLT (T,9)
CALL BSWAIT
:
END
```

Аналогично может быть организован диалог на уровне 3.

Приведенные выше примеры программирования соответствуют динамическому заданию структуры диалога.

При статическом описании для рассматриваемого примера в подпрограмме INIT могут быть выполнены все действия по заданию структуры, и в тех исполнительных подпрограммах, где осуществляется переход на новый уровень, остается лишь обращение к подпрограмме цикла (BSWAIT). Например:

```
SUBROUTINE INIT
:
DIMENSION T1(9), T2(9), T3(9)
DATA (T1(I), I=1,9)=1HT,4HEDIT, 1HI,
* 1HG, 2HGO, ØB, 1HA, 4HANAL, 1HI)
DATA ( (T2(I), I=1,9)=2HD3, 5HDISP3, ØB, 2HGR,
* 5HGRAPH, 2HVI, 1HP, 3HPRT, ØB)
DATA ( (T3(I), I=1,9) =.....
DIMENSION CAT(9)
:
CALL BSCAT (CAT,9)
CALL BSLTC(T1,9,1HG)
CALL BSLTC(T2,9,1HA)
CALL BSLTC (T3,9,2HD3)
```


CALL BSLT (T1,6)

CALL BSWAIT

⋮

END

SUBROUTINE ANAL (I)

⋮

CALL BSWAIT

⋮

END

7. ЗАКЛЮЧЕНИЕ

Предлагаемый способ описания структуры диалога для программ, написанных на языке ФОРТРАН, и его реализация с помощью унифицированного механизма обработки сообщений требуют минимальных затрат на программирование, обладая вместе с тем значительной гибкостью в организации сложных структур диалога. Высокая эффективность предложенных средств проверена при решении на УДС ряда задач из области теоретической и экспериментальной физики.

ЛИТЕРАТУРА

1. Кавченко А.В. и др. ОИЯИ, Р10-9325, Дубна, 1975.
2. Говорун Н.Н. и др. Вычислительный комплекс ОИЯИ и перспективы его развития. "Автоматика и вычислительная техника", Рига, 1974, 6, с.62-68.
3. Резанов В.В., Саввов В.В., Маргулис Д.С. Агрегатный ряд видеотерминальных установок для ЭВМ третьего поколения. "Механизация и автоматизация управления", 1971, 4, с.50-53.
4. Заикин Н.С. и др. ОИЯИ, 10-9326, Дубна, 1975.

Рукопись поступила в издательский отдел
14 февраля 1977 года.