

Р. РАУД

## ОБРАБОТКА УПРАВЛЯЮЩИХ ПРЕДЛОЖЕНИЙ ЯЗЫКА УПРАВЛЕНИЯ ЗАДАНИЯМИ В ОС-32

Описывается метод предварительной обработки управляющих предложений (УП) языка управления заданиями (ЯУЗ) и приводится описание управляющей информации, определяющей содержание ЯУЗ в операционной системе ОС-32. Общая характеристика ОС-32 дана в [1].

### Введение

ОС-32 разработана для ЭВМ «Минск-32» стандартной конфигурации, к которой дополнительно подключена внешняя память с произвольным доступом (магнитные диски, барабаны) и выносные пульты для дистанционной обработки данных (терминалы).

Вычислительная система (ВС) под управлением ОС-32 работает в режиме мультипрограммной пакетной обработки с переменным числом заданий и дистанционной обработки одновременно. Основная единица работы — это задание. Задания независимы друг от друга. Они описываются для ВС управляющими предложениями, которые кодируются на ЯУЗ. Поток УП заданий заднего плана (пакетная обработка) вводится с устройства ввода перфокарт. С каждого терминала можно ввести УП для описания задания переднего плана (дистанционная обработка; см. [1]). Каждое задание разбивается на последовательно выполняемые шаги задания, каждый из которых определяется выполняемой программой и требуемыми для нее данными.

С одной стороны, ЯУЗ должен быть достаточно мощным для предоставления пользователю разнообразных возможностей описания своих заданий. С другой стороны, для удобства пользования ЯУЗом УП должны содержать только крайне необходимую информацию для описания задания. В связи с этим в ЯУЗ должна существовать возможность опускания параметров, которые получают обусловленные значения по умолчанию, и опускания конструкций, которые не имеют смысла при данном типе задания.

Содержание ЯУЗ (т. е. кодировка, смысл и наличие параметров, а особенно обусловленные значения параметров) может меняться в зависимости от требований при конкретном применении (напр., для заданий переднего и заднего планов, при введении дополнительных возможностей работы с файлами и т. д.), но внешняя форма — синтаксис — остается неизменной. Исходя из различных требований, очевидно, целесообразно разработать для каждого применения подходящий вариант ЯУЗ. Для перевода входной информации о задании, закодированной на разных ва-



риантах ЯУЗ, к стандартному виду служит в ОС-32 транслирующая программа (ТП, [1]).

Чтобы не писать для каждого варианта ЯУЗ новую программу, ТП разработана настраиваемой на конкретный ЯУЗ так наз. управляющей информацией (УИ). УИ содержит описание параметров и конструкции языка, допускаемый порядок их следования, обусловленные значения параметров и правила кодирования входной информации для конкретного варианта ЯУЗ. Блоки УИ хранятся в системной библиотеке. Настройка ТП производится автоматически.

Описания синтаксиса и структуры ЯУЗ, процесса обработки УП, а также структуры УИ приводятся ниже.

### Структура и синтаксис ЯУЗ

Входной информацией для ТП служат УП заданий, кодированные на ЯУЗ. Хотя содержание ЯУЗ различно при разных вариантах языка, его структура, а также синтаксис не меняются. Синтаксис ЯУЗ ОС-32 разработан исходя из JCL OS/360 [2] и ЯУЗ М-4000 [3] и приведен ниже в нотации Бэкуса, дополненной Айронсом.

<управляющее предложение> ::= // <предложение> <конечный символ>  
 <предложение> ::= <позиционный параметр> { <разделитель>  
 <позиционный параметр> } { <разделитель> <ключевой параметр> } |  
 <ключевой параметр> { <разделитель> <ключевой параметр> }  
 <позиционный параметр> ::= <параметр>  
 <ключевой параметр> ::= <ключевое слово> = <параметр>  
 <разделитель> ::= = □ |,  
 <ключевое слово> ::= <строка>  
 <строка> ::= <строчный символ> { <строчный символ> }  
 <параметр> ::= <строка> | <ключевое слово> | <предложение>  
 <символ> ::= <строчный символ> | <разделитель> | <конечный символ>

З а м е ч а н и я: 1. Символом может быть любой символ по ГОСТ 10859-64. Конечным символом может быть любой символ при условии, что он не входит в понятие <строчный символ> или <разделитель>. При перфорации на перфокарты подходит символ с кодом 1111110, что соответствует колонке без пробивок. Строчными символами могут быть любые символы, кроме конечного символа и разделителей.

2. Под определением <параметр> ::= <ключевое слово> подразумевается ключевое слово из фиксированного набора альтернативных ключевых слов.

3. Для некоторой УИ допускается сокращение записи. Опускаемые параметры заполняются автоматически заранее обусловленными наиболее вероятными их значениями. При опускании параметров общими являются следующие три правила:

а) разделители могут быть опущены только в том случае, если в данном предложении не последует других позиционных параметров;

б) ключевые параметры предложения можно писать в любой последовательности и при их опускании никаких специальных обозначений не требуется;

в) если опускается <параметр> ::= <предложение>, то скобки не нужны, причем их можно опустить и в том случае, если в предложении указывается только первый позиционный или один любой ключевой параметр. Скобки сохраняются, если этот единственный параметр сам является предложением и сохраняет свои скобки.

В ЯУЗ имеется четыре типа УП (по назначению). Наличие всех типов в конкретном варианте ЯУЗ необязательно.

1. УП описания задания отмечает начало УП данного задания и содержит общие данные о задании (напр., имя, учетную информацию, приоритет, предел времени, требуемое количество основной памяти и т. д.).



2. УП описания шага задания отмечает начало УП третьего типа данного шага задания и специфицирует программу, которая должна быть выполнена, параметры этой программы, предел времени и т. д.

3. УП описания данных содержит информацию о файле для данного шага задания (напр., имена файла и тома, длину записи и т. д.).

4. Разделительное УП служит для разделения информации во входном потоке.

Содержание и кодировка параметров всех четырех описанных типов могут быть различными для разных вариантов ЯУЗ.

Результатом работы ТП являются управляющие блоки (УБ) заданий, шагов заданий и файлов, которые помещаются в системный файл МОНИТ (с прямым доступом со связными записями) и каталогизируются в соответствующих каталогах. Информация в УБ размещается с учетом удобства ее дальнейшей обработки другими программами монитора (опущенные в УП параметры имеют обусловленные значения, числовые значения параметров преобразуются в требуемый вид и пакуются в соответствующих битах ячеек УБ и т. д.).

Типы УП приведены выше в порядке уменьшения уровня иерархии. Наивысший уровень иерархии в файле МОНИТ имеет каталог заданий. Всякий УБ каталогизируется в каталоге при УБ последнего УП, уровень иерархии которого выше настоящего. Например, адреса УБ заданий помещаются в каталог заданий МОНИТ, адреса УБ шагов заданий — в каталог при УБ соответствующего задания и т. д.

Для удобства использования ЯУЗ предусмотрены: а) возможность группировки УП в процедуры (описана в [1] и здесь не обсуждается) и б) возможность ссылки на параметры в предыдущих УП этого же задания.

Ссылка на параметры в предыдущих УП с помощью имен последних указывает системе, что значение этого параметра или группы параметров (предложение или несколько предложений в зависимости от места появления ссылки) то же, что и в УБ, на которое ссылка указывает. Какие именно значения параметров нужно перенести из других УП при ссылке на данном месте, определяется УИ используемого варианта ЯУЗ.

В общем случае ссылка имеет вид

\*.<имя 1>.<имя 2>

где: <имя 1> ::= <имя УП>

<имя 2> ::= <имя УП>

<имя УП> ::= <строка>

Какой конкретно параметр УП считается именем УП, определяется УИ. Имя 1 — это имя того УП, где хранится каталог УП, к которому ссылаются (имя 2). Если УП, которое ссылает, и УП, к которому ссылаются, принадлежат одному и тому же каталогу, то имя 1 опускается и ссылка принимает вид \*.<имя 2>.

### Управляющая информация

УИ определяет содержание конкретного варианта ЯУЗ, т. е. расположение параметров (позиционных, ключевых), ключевые слова, наличие предложений, требуемые преобразования значений параметров и допустимые символы в них, возможные места появления ссылок и параметры, которых это касается, а также место расположения значений параметров в соответствующем управляющем блоке (УБ) с точностью до бита.



Обработка УП производится в два этапа. На первом этапе выделяются значения параметров из последовательности входных символов УП, делаются требуемые по УИ преобразования и результаты помещаются в одну или несколько (указаны в УИ) ячеек макета. При этом исключаются ключевые слова и опущенные параметры получают значения по умолчанию. Также производится обработка процедур. Полученный макет служит входной информацией для второго этапа, где обрабатываются ссылки, формируются УБ (т. е. значения параметров помещаются в требуемые биты требуемых ячеек) и формируется файл МОНИТ (каталогизация УБ и запись в МОНИТ).

Согласно этапам обработки УИ имеет также две независимо используемые части, которые кодируются по-разному.

Первая часть УИ представляет собой последовательность описаний объектов и допустимых (в данном варианте ЯУЗ) конструкций ЯУЗ в порядке их допустимого появления в УП. В ней выделяются следующие четыре вида описаний объектов: 1) группы, 2) слова, 3) константы, 4) ключевых параметров, рассматриваемых в качестве элементов УИ.

Описание группы объединяет описания объектов некоторой логической группы ЯУЗ, где определено фиксированное количество позиционных параметров и единый набор ключевых параметров. Оно предназначено для группировки элементов УИ одного предложения ЯУЗ. После входа в описание группы последовательность символов УП обрабатывается по элементам УИ, объединенным в группу до появления разделительного символа или до конца группы (в этом случае УП ошибочно). Разделительным символом служит «)», если предложение начинается с символа «(». Если в предложении указывался только один параметр, то разделителями служат «.» или «□». Появление символа «)» означает также конец всех неоконченных групп до группы (incl.), которая начиналась с символа «(» (см. предыдущий раздел).

Описание слова содержит информацию для выделения, формального контроля и преобразования (если требуется) значения параметра УП.

Формальный контроль проверяет количество и допустимость символов при данном параметре. Конец параметра определяется появлением разделительного символа. Последовательность символов — значение параметра — может быть преобразована в восьмиричное число, десятичное число (двоично-десятичный код) или в последовательность битов.

Описание константы предназначено для задания значений параметров по умолчанию, а также для присваивания значений параметру, который определяется набором альтернативных ключевых слов (см. ниже).

При ключевых параметрах ключевое слово определяет смысл параметра, а значение определяется последовательностью символов, которые следуют за ключевым словом. Так как ключевые параметры предложения могут быть указаны в УП в любой последовательности и наличие ключевого параметра определяется наличием ключевого слова в УП, следует каждый раз брать под контроль все допустимые ключевые слова предложения. Для определения начала описаний набора ключевых слов предложения служит элемент УИ четвертого типа, за которым в УИ располагаются имена ключевых слов попеременно с описаниями слова, константы или групп.

При альтернативном наборе ключевых слов (см. предыдущий раздел) значение параметра определяется уже появлением одного слова из набора. Действия программы при нахождении ключевого слова из набора аналогичны описанному при ключевых параметрах. Разница состоит лишь в том, что следующие (за выделенным ключевым словом) символы УП должны интерпретироваться по описанию следующего объекта



(за описанием набора ключевых слов) в УИ, а не как значение ключевого параметра или как следующее ключевое слово из этого же набора. Для достижения этого описание набора альтернативных ключевых слов объединяется самостоятельным описанием группы. Выход из описания группы производится при появлении любого разделительного символа.

Значением параметра, которое определяется ключевым словом, может служить само ключевое слово в символах, однако в большинстве случаев целесообразнее кодировать это значение в нескольких битах УБ. Для этого ключевые слова располагаются в УИ попеременно с описаниями соответствующих констант или же с описаниями групп, так как иногда ключевое слово определяет также смысл параметров в УП. В определенном месте макета располагается служебная информация из УП для управления программой ТП. Например, тип УП, который в JCL-OS/360 [2] определяется вторым позиционным параметром. Тип УП используется на втором этапе обработки для управления каталогизацией и для контроля за следованием УП каждого задания, а также для определения точки входа во вторую часть УИ (соответствует типу УП).

В первой части УИ описания объектов всех типов УП объединяются в один связной массив, т. е. входная точка в УИ для всех типов УП одна и та же. Разветвление для разных типов УП естественно произвести по параметру, указывающему тип УП. В предложенном в [1] варианте ЯУЗ тип указывается вторым позиционным параметром УП. При объединении описаний всех типов в один связной массив на месте второго позиционного параметра получается набор альтернативных ключевых слов (ЗДН, ВЫП, ОД, см. [1]). Ключевому слову в УИ соответствует описание группы, которое объединяет описания допустимых в данном типе УП объектов.

На первом этапе обработки ссылки помещаются в специальное поле. В ячейку макета для этого параметра помещается адрес ссылки.

Вторая часть УИ второго этапа обработки определяет кодировку параметров в УБ. Перемещение информации из одной ячейки макета описывается одним элементом УИ. Каждый элемент упакован в одну ячейку памяти и содержит: а) адрес ячейки параметра в УБ; б) константу сдвига (учитывается, что числовые значения располагаются в последних битах ячейки макета); в) адрес логического выделителя (константы для логического выделения располагаются раздельно в УИ) и д) охват ссылки (если она допущена).

Элементы в УИ располагаются в последовательности соответствующих значений параметров в макете раздельно для каждого типа УБ. Входная точка в УИ настоящего типа УБ определяется по служебной ячейке макета, куда во время первого этапа обработки помещается адрес входной точки.

Если в очередной ячейке макета нет ссылки, то содержание этой ячейки сдвигается и логически складывается с ячейкой УБ, адрес которой специфицирован в данном элементе УИ. Если же в макете вместо значения параметра находится ссылка, то по указанным в ссылке именам УБ и адресу ячейки параметра (из элемента УИ) определяется место ссылаемого параметра, который затем выделяется специфицированной в элементе УИ константой логического выделения и логически складывается с соответствующей ячейкой УБ. С помощью одной ссылки можно сослаться на несколько параметров, значения которых в макете должны располагаться в смежных ячейках. По указателю



охвата ссылки в элементе УИ определяется количество следующих элементов УИ. Соответствующие им параметры следует выделять из УБ, на который ссылаются аналогично описанному выше.

### Заключение

ОС-32 — это универсальная операционная система, обеспечивающая выполнение заданий в мультипрограммном пакетном режиме и в режиме дистанционной обработки данных.

Все задания пользователя описываются для системы на языке управления заданиями (ЯУЗ), который предоставляет возможности для удобного описания всех типов заданий.

Форма (т. е. синтаксис языка, механизмы ссылок и процедур) ЯУЗ в ОС-32 является универсальной, а содержание определяется УИ. Программа монитора, обрабатывающая УП ЯУЗ, перед работой настраивается (с УИ) на конкретный вариант языка. Эта программа приводит УП любого варианта ЯУЗ к стандартному формату УБ, которые служат входной информацией для других программ монитора. УИ разрабатывается исходя из особенностей класса работ, для которого данный вариант языка предназначен.

Обработка УП производится в два этапа. На первом этапе выделяются значения параметров из последовательности входных символов УП, производятся требуемые по УИ преобразования и формальный контроль правильности значений. На втором этапе обрабатываются ссылки, формируются УБ и файл МОНИТ.

Соответственно этапам обработки УП УИ имеет также две части. Первая часть УИ содержит описание параметров во входной последовательности символов, значения параметров по умолчанию и описания групп, которые объединяют описания параметров предложения (по синтаксическому описанию) и других логических объединений параметров (напр., альтернативный набор ключевых слов). Первая часть УИ представляет собой один связной массив для всех типов УП. Разветвление по типам производится по параметру, указывающему тип УП. Предполагается, что обработка предшествующих параметров одинакова для всех типов УП.

Вторая часть УИ содержит описания упаковки значений параметров в УБ, а также описания действий программы при ссылках на значения того же параметра в предыдущих УП того же задания. Вторая часть имеет отдельную входную точку для всех типов УП, адрес которой определяется на первом этапе обработки по параметру типа УП.

Первая и вторая части УИ используются независимо. Первый этап обработки связан с устройством ввода перфокарт или терминалом, а второй — с дисковым томом, где находится файл УБ. Благодаря этому возможен режим параллельной работы, что повышает эффективность использования аппаратуры ЭВМ.

ЯУЗ — это связующее звено между ОС и пользователем, поэтому приспособляемость к конкретному применению является важнейшим его свойством. В ОС-32 это достигается возможностью разработки многих вариантов ЯУЗ, каждый из которых удобен для применения при описании определенного класса задач.

## ЛИТЕРАТУРА

1. Косе Г. Я., Куусик В. А., Куусик Л. Э., Рауд Р. К., Препринт № 7 Ин-та кибернетики АН ЭССР, Таллин, 1973.
2. Brown G. D., System/360. Job Control Language, New York—London—Sidney—Toronto, 1970.
3. АСВТ. Руководство по эксплуатации организующей и транслирующей систем, Кн. 2, Ч. 1, М., 1969.

Институт кибернетики  
Академии наук Эстонской ССР

Поступила в редакцию  
23/IV 1974

R. RAUD

### OPERATSIOONISÜSTEEMI OS-32 TÖÖDE JUHTIMISKEELE LAUSETE TÖÖTLEMINE

Artiklis kirjeldatakse arvutis «Minsk-32» kasutatava operatsioonisüsteemi tööde juhtimiskeele süntaksit ja transleeriva programmi häälestamiseks konkreetsele keelevariandile ettenähtud juhtinformatsiooni struktuuri.

R. RAUD

### JOB CONTROL LANGUAGE STATEMENTS TRANSLATION IN OPERATING SYSTEM OS-32

The syntax of Job Control Language for OS-32 (for computer "Minsk-32") is described. The idea of constructing various versions of languages for various applications is discussed. The structure of control information for adjusting the translating program into a certain version of language is also described.