ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ
ДУБНА

E10 - 8855

D.D.Arnaudov, N.N.Govorun

INFORMATION RETRIEVAL SYSTEM OF JINR

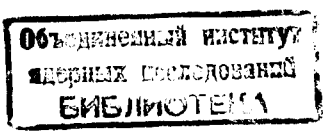
1975

E10 - 8855

D.D.Arnaudov, N.N.Govorun

INFORMATION RETRIEVAL SYSTEM OF JINR

Submitted to the Fifth Cronfield Conference on Mechanised Information Storage and Retrieval Systems



Арнаудов Д.Д., Говорун Н.Н.

E10 - 8855

Информационно-поисковая система ОИЯИ

Рассматриваются принципы построения и структура основных информационных массивов ИПС ОИЯИ. Система реализована на алгоритмическом языке КОБОЛ на ЭВМ СДС-6200.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Препринт Объединенного института ядерных исследований
Дубна 1975

Arnaudov D.D., Govorun N.N.

E10 - 8855

Information Retrivol System of JINR

The main principles and the structure of the basic files of the system are discussed. The Information Retrieval System if realized on the base of the algorithmical language COBOL.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Preprint of the Joint Institute for Nuclear Research
Dubna 1975

The solution of the file organization problem is connected with many difficulties which are due to the contradictions between the memory space and the response time. That is why a great deal of attention must be paid to the main principles according to which an efficient system can be built. The principles for building the Information Retrieval System (ISR) of JINR are described in [1]. In accordance with these principles we shall describe the organization of the basic files of the system (see Fig. la).

There are four main files in the system:

1. MD file. This is the file where the abtsracts of the documents are gathered. It does not take place in searching, that is why we shall not describe it further.
2. OMPOD file. This is the basic file in the system where the index records of the documents are gathered. It has a multilist structure.
3. MZD file. This is the file of the list heads.
4. OMD file. This is the file of descriptors.

The file OMPOD

The index records of each document are described by a set of descriptors. These descriptors, grouped together, form the node of a multilist. The number of a given node (the index of a node in the file) is not the part of the record. It coincides with the number (the index) of the first element of the node where the first descriptor is situated. The description of a node is shown in Fig. lb.

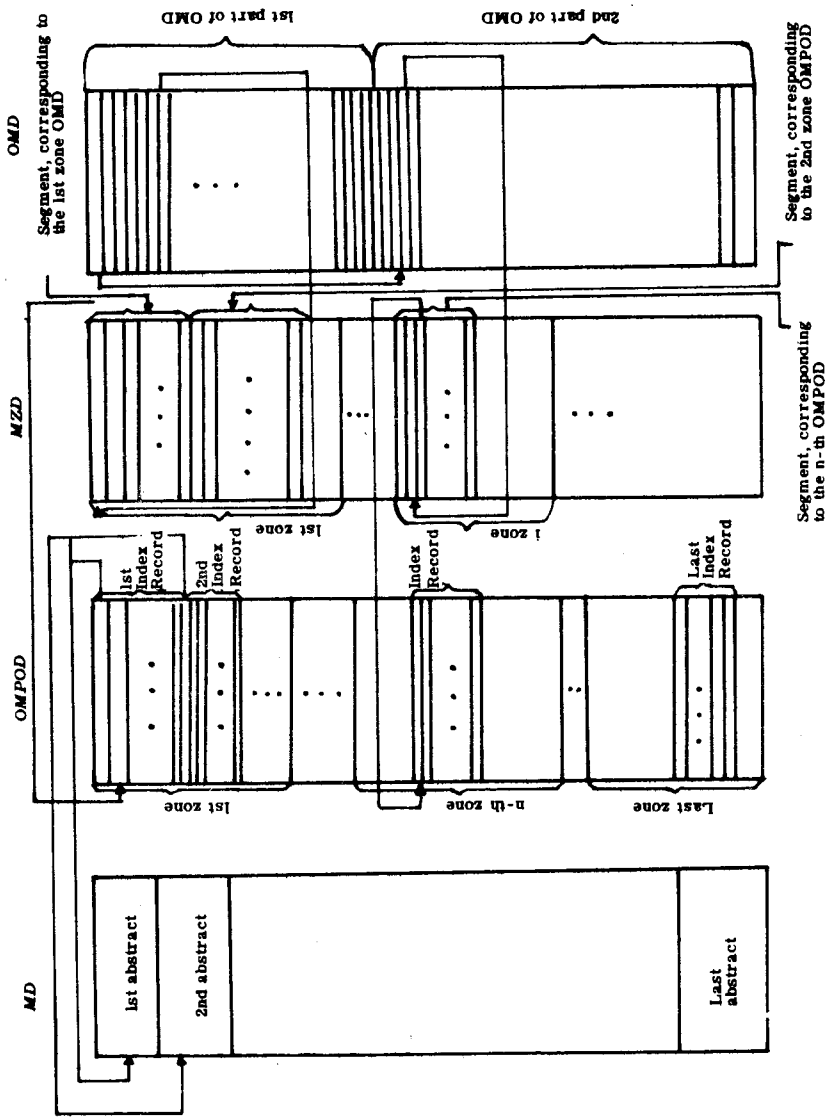


Fig. 1

1	a_{i1}	b_{i1}	c_{i1}	d_{i1}
0	a_{i2}	b_{i2}	c_{i2}	d_{i2}
.				
.				
.				
0	a_{ik-1}	b_{ik-1}	c_{ik-1}	d_{ik-1}
2	a_{ik}	b_{ik}	c_{ik}	d_{ik}

Fig. 1b

The ciphers in the first raw can be 1, 0, 2. A cipher "1" indicates the beginning of a node, "0" is the middle of a node, "2" is the last element of a node.

The figure a_i is the code of a concrete descriptor, which is used in the index record of a document. At the same time a_i is the number (the index) of an element of the file on the disk, where the first part of the file OMD is situated.

A code b_i is a link address, i.e., this is the index of a node element in the zone, where the same descriptor can be met again.

A code c_i marks a reservation field for an additional information.

A code d_j is a link address giving the correspondence between the index record of a document and its abstract, which is situated in the MD file. In this case the code shows the disk number and the number of an element on the disk from which the abstract of a document begins.

So we have a unique format for each index record which gives the opportunity of forming a record with a variable length. At the same time, by means of the link addresses, all the index records of a given descriptor are chained forming the list of a descriptor. We can enter the beginning of a list by means of a list head which is situated in the MZD file (see further the description of a MZD file).

It should be noted, that the disk memory of the OMPOD file is divided into logical partitions called "zones". The list nodes are positioned in these partitions. It means that each list is localized within a zone. This technique is often called cellular multilist. It is clear then, that one descriptor may have lists in different zones. The structure of the OMPOD file is shown in *Fig. 2a.*

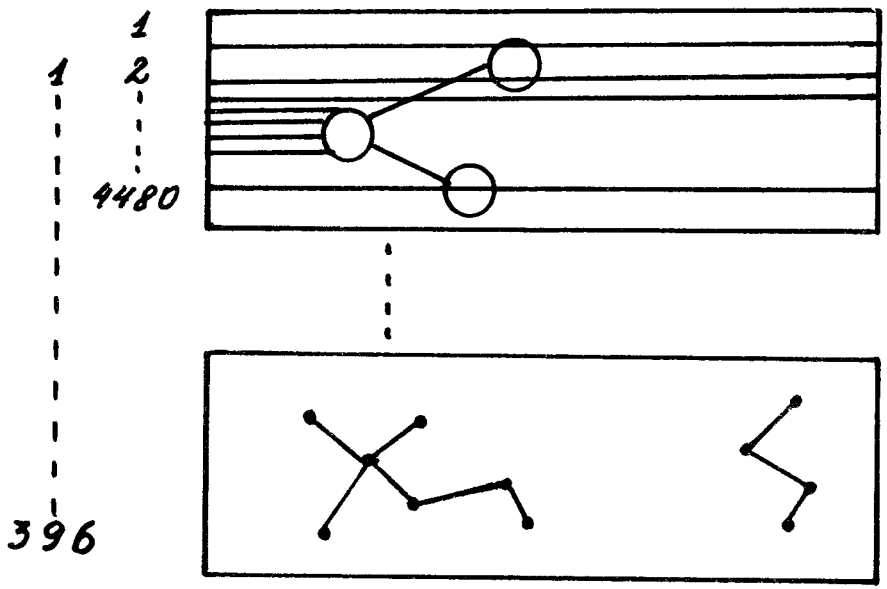


Fig. 2a.

Each element of the node has a format which is shown in Fig. 2b.

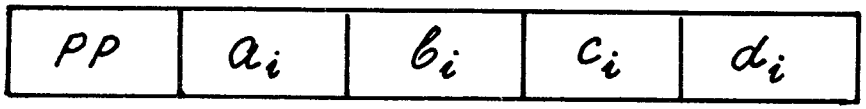


Fig. 2b

PP -one symbol; a_i -five symbols; b_i - four symbols; c_i - two symbols; d_i - eight symbols.

Each element comprises two words (one CDC word has 60 bits) memory. The size of each zone is half of a cylinder (the characteristics of a disk pack are described in /4/). This size gives the opportunity of using the "natural" segmentation of a disk. It is of great importance for the retrieval process, because when searching is done in a particular zone, then a given list search is effected without any further head motion (once a head has been positioned to a given cylinder).

According to an appropriate format 396 zones can be situated on the disk medium (CDC disk 841). If on an average, ten descriptors are used for each index record, then in one zone 448 index records can be located. It means that 177408 index records can be written on the disk pack.

The OMPOD file is formed chronologically. In the process of forming, if in the node of a document in a particular zone a given descriptor is met for the first time, then the list of this descriptor must be formed. In this case, the head of this list is updated in the file MZD.

The MZD file is the file of list heads. Each element of this file is characterized with six different codes: a, c_i, N_{ij}, d_j, e_j, c'_j, where

a is a field for an additional information (a reservation field);

c_i -is a link address pointing to the next list head of the same descriptor which is located in the same zone of the MZD file;

N_{ij} -is the number of the index records in the L_j -list of this descriptor;

d_j, e_j - are the numbers of a disk and that of a zone, where the L_j -list is located;

c'_j -is the address of the element of the first list node for the given descriptor in the e_j zone.

As we have said, one or more list heads belonging to one and the same descriptor, may be met in a particular zone of the MZD file. In order to shorten the total number of the elements in the OMD file, and to improve the retrieval strategy /3/, all of the heads of a particular descriptor in the e_j zone are chained in a list. All such lists are localized in the MZD zone.

In this way, each line in the MZD file (see Fig. la) is the head of a given descriptor. At the same time it is connected with the first line, "representing" this descriptor in the OMPOD zone which corresponds to an appropriate segment in the MZD file. Each element of the MZD file consists of 20 symbols which are distributed as follows:

a – four symbols;
c_i – four symbols;
N_{ij} – four symbols;
d^j – one symbol;
e^j_j – three symbols;
c_j – four symbols.

The OMD file

This file plays the part of a descriptor directory. It also has a list structure. The elements, which consist of the addresses corresponding to each first element of the head lists in the MZD file (see Fig. 3), form a chain list structure. The codes of the descriptors are not shown evidently in this file. They coincide with the indexes of the elements in the first part of the file. The OMD file is divided logically in two parts. Each line in the first part corresponds to a particular descriptor and it is connected with:

a) a line in the MZD file which is the first line corresponding to a particular descriptor in a given zone of MZD;

b) a line in the second part of OMD, if a particular descriptor has list heads located in different zones of the MZD file.

In the second part of OMD each line corresponds to a particular descriptor which has either a line in the first part of OMD, or a line in the second part of OMD which is located upper than the given line. Each line in the second part is connected with:

a) a line in the zone of MZD which corresponds to this descriptor:

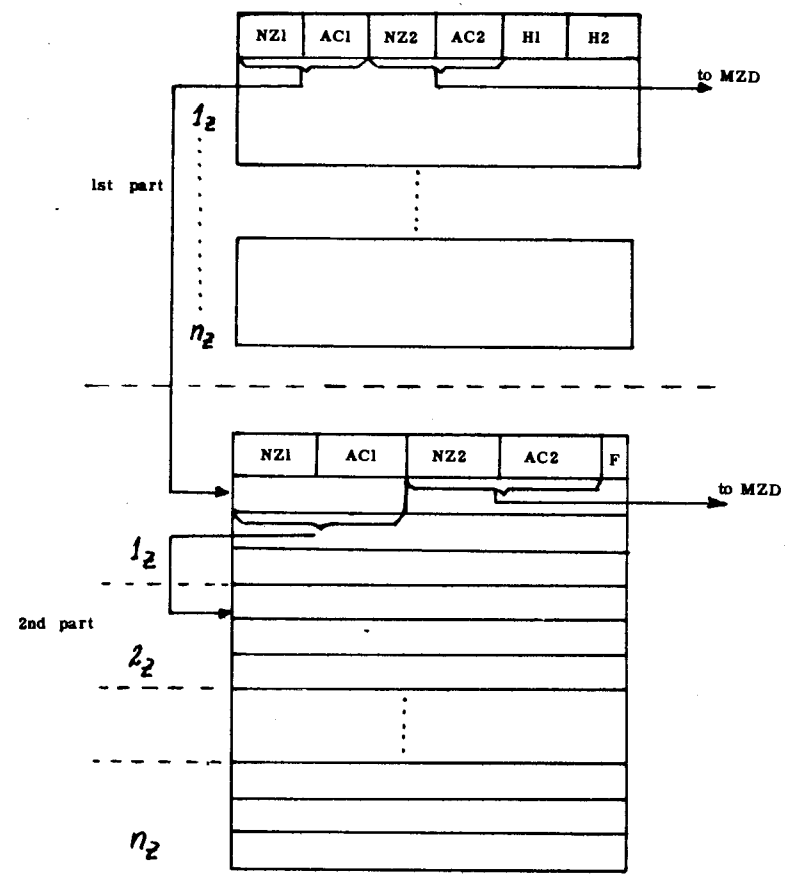


Fig. 3

b) a line in the second part of OMD, if the particular descriptor has a line in the zone of MZD which is connected neither with a line in the first part of OMD, nor with a line situated upper in the second part of OMD.

Each element in OMD consists of the following information (see Fig. 4).

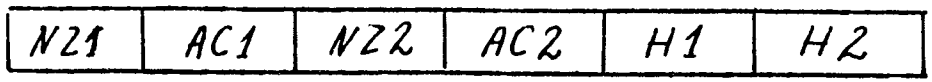


Fig. 4

The figures under the fields represent the number of the symbols.

In this case NZl, ACl represent the address of the next element in the list of OMD, where NZl is the number of the zone, ACl is the number of the element in the zone. The address of the list head in the MZD file is represented by NZ2, AC2. The field Hl determines the number of the elements in the chain list of the NZl zone; H2 is the number of the elements in the chain list of the HZ2 zone.

One should note, that OMD is also logically partitioned in zones. But the lists are not localized in the zone. Here the logical partition is connected with the efficiency of the search strategy and the minimizing of the moves of a magnetic head during the retrieval of the queries /3/.

As we see, the structure of our files represents the modification of a cellular multilist structure.

A very important notion is the concept of a "zone". By means of this logical partition we can solve two problems:

1) The notion "zone" is the part of a searching procedure for matching the descriptors of a query and index records in OMPOD /3/.

2) The zone is "tool" for overlapping the searches in the lists corresponding to a particular zone. At the same time the number of the moves of the magnetic head is minimized.

In the searching strategy /3/ it is shown, that in order to minimize the response time, the access time during the searching procedure must be minimized.

As is known, the average access time may be calculated according to an expression /1/.

$$t_s = t_p + t_0 + t_{Tp} \qquad (1)$$

t_p is the seek time of a particular cylinder;
t_0 – the rotation time of the disk pack;
t_{Tp} – the transmission time.

For the disk pack of the CDC-6200, $t_p = 75$ ms; $t_0 = 25$ ms. The time for transmission of n symbols can be calculated, using the following expression:

$$t_{Tp} = \frac{n}{Q} \times t_0 ,$$

where Q is the number of the symbols in the track. In our case

$$t_{Tp} = \frac{n}{9016} \times t_0 ; \quad n = 4508 ; \quad t_{Tp} = 1/2\, t_0 \text{ ms}.$$

When the information of the whole zone is accessed and read in the operative memory, the access time (t_b) can be calculated according to the expression (2)

$$t_b = t_p + 20t_0 + 10t_0 = t_p + 30t_0 , \qquad (2)$$

In our case

$$t_b = 75 + 500 + 250 \qquad t_b = 825 \text{ ms}.$$

It means that when the whole zone is read, the time of 825 ms is needed. But in this case there is no latency time.

If a separater element in the zone is read (for OMD and MZD it consists of two words, and for OMPOD this is an index record - twenty words), then the average access time is

$$t_{bl} = t_p + 1/2\, t_0 + t_{Tp}. \qquad (3)$$

Table 1

Number of queries	Number of descriptors in a query	Time of the CP /sec /	Astronomical time / sec /
I	6	I7	28
3	I8	2I	36
5	27	24	44
I4	9I	29	6I
50	323	47	95

In this case t_{Tp} is small enough and we do not take it in mind.

$$t_{bl} = t_p + 1/2\, t_0 \approx 87{,}5 \text{ ms}. \qquad (4)$$

If we have in mind (2) and (4), in the process of searching it must be determined when the whole zone must be read, and when there are separate elements of the zone (it can be done, of course, if there is an appropriate software). This decision, in the simplest case, can be taken (for instance, for the OMPOD) according to the number of the elements which must be read in the different lists, corresponding to a particular zone. This number must be compared with K, and if it is greater, then it is better to read a whole zone.

$$t_b = k\, t_{bl} \;;\; k = \frac{t_b}{t_{bl}} \approx 10.$$

Hence, if more than ten accesses are needed in a particular zone, then it is better to read the whole zone in the operative memory and there can be retrieved all of the corresponding lists.

According to /2/ and /3/ the information retrieval system of JINR is built. In table 1, the response time is given in a package regime, when 1, 3, 5, 14, 50 queries are retrieved.

In table 2, the memory on the disk pack (in zones) is given for situating the basic files (the numbers of documents are 100, 1000, 4000). Each document is represented in the form of an index record in OMPOD which is indexed by ten descriptors on an average.

In table 3, the time is given, which is required for the creating of these files.

Table 2

Number of documents / Memory zones	100	1000	4000
OMD /sec:part/	0	0	0,36
MZD	0,07	0,58	1,9
OMPOD	0,15	2,04	6,52

Table 3

Number of documents	100	1000	4000
Astr. time / sec /	150	1500	6000
CP time / sec /	15	1000	3000

References

1. *N.N.Govorun, D.D.Arnaudov. JINR Preprint, P10-8785, Dubna, 1975 (In Russian).*
2. *D.D.Arnaudov. JINR Preprint, P10-8621, Dubna, 1975 (In Russian).*
3. *D.D.Arnaudov. JINR Preprint, 10-7949, Dubna, 1974 (In Russian).*
4. *D.D.Arnaudov. JINR Preprint, P10-7586, Dubna, 1973. (In Russian).*

Received by Publishing Department on May 7, 1975.