VSAM - 26-09-2023

PS → PHYSICAL SEQUENTIAL/FLAT FILE
STPOS,LENGTH
PDS → MEMEBRS/PS

VIRTUAL STORAGE ACCESS METHODs -
IDCAMS(Integrated Data Catalogue Access Method Services)

STORE - ORGANIZED → BETTER FETCH
CATALOGUED → STORE IN THE MAIN DIRECTORY
META DATA.

VSAM → CLUSTERS.

RECFM/LRECL → RECORD SIZE
BLKSIZE → CONTROL INTERVAL(CI) → COLLECTION OF RECORDS
SPACE → TRACK/CYLINDER(PQ,SQ)

CLUSTER ALLOCATION PARAMETERS

DSN = HLQ.Q2.Q3
VOLUME/DEVICE/SERIAL
SPACE UNIT → TRACKS(PQ,SQ) SQ extended?
Controlintervalsize/cisize(512) multiples 512... 8kb
+ 2kb...32kb
RECORDSIZE(AVG,MAX) (80,80)
INDEXED/NONINDEXED/NUMBERED/LINEAR

Access methods:

1. Sequential → default → from the first record till EOF.
   (DOESN'T NEED RBA,KEY,RRN)
2. Random → chooses any one record with its RBA/key/rrn given.
   Fromaddress(80) toaddress(80) → esds
   Fromaddress(50) → terminated
3. Dynamic → Combination of random and sequential
   Starting at a random rba and then reading consecutive records sequentially till EOF or till another RBA/RRN/KEY.
   Fromaddress(400) toaddress(752)
   Fromaddress(400) → till eof

TYPES OF VSAM CLUSTERS.

1. ENTRY SEQUENCED DATA SET(ESDS).
   (give the RBA and take the record)
   The records are stored in the sequence in which they are entered.
   Every record is organized by giving it unique RBA.
   RBA - relative byte address. It is the sum of the sizes of the previous records in the control interval.

   a. Define cluster
   //JOBCARD
   //STEP1 EXEC PGM=IDCAMS

```
//sysprint dd sysout=*
//SYSOUT DD SYSOUT=*
//SYSIN DD *
        DEFINE CLUSTER ( -
        NAME(OZAGS1.ALWYN.REVAT.VSAM.ESDS) -
        TRACKS(3,2) -
        RECORDSIZE(80,80) -
        CONTROLINTERVALSIZE(512) -
        NONINDEXED -
        )
/*
```

b. Repro - populates dataset with data from another dataset

```
//SYSIN DD *
   REPRO
INDATASET(OZAGS1.ALWYN.REVAT.VSAM.DATA) -
OUTDATASET(OZAGS1.ALWYN.REVAT.VSAM.ESDS)
/*
OTHER OPTION.
   REPRO INFILE (DDNAME) -
        OUTFILE(DDNAME)
```

Note: Esds accepts any number of REPROs from same input dataset or from different input dataset.

Esds accepts duplicate records.

Skip(n) skips n number of records from the input.

Count(n)

c. Print the data by giving its rba

Fromaddress(rba) toaddress(rba)

2. Relative Record Data Set(RRDS).

(give the RRN and take the record)

Records are stored in empty slots. Every slot is given with a sequence number called RRN.

- Numbered.

a. Define

```
//SYSIN DD *
   DEFINE CLUSTER ( -
   NAME(OZAGS1.ALWYN.REVAT.VSAM.RRDS) -
   TRACKS(3,2) -
   RECORDSIZE(80,80) -
   CONTROLINTERVALSIZE(512) -
   NUMBERED -
   )
/*
```

b. Repro

```
   //SYSIN DD *
         REPRO INDATASET(INPUT DATASET NAME) -
         OUTDATSET(OZAGS1.ALWYN.REVAT.VSAM.RRDS)
   /*
```

Note: RRDS accepts REPRO ONLY ONCE.

It accepts duplicate records.

Skip(n) count(m)

c. Print

```
   //sysin dd *
         Indataset(rrds) -
```

Char -

Fromnumber(x) tonumber(x)

/*

Sequential

Random

dynamic

3. KEY SEQUNCED DATA SET.(KSDS)

(give the key take the record.)

Note: the data field must be known even before defining the cluster.

Consider and define any unique part of the record as the key.

The repro is allowed only if the key column of the input dataset contains sorted records.

9999 tommy Chennai 90000 9988987

.index → key of the record and the rba of the record

.data → data

a. Define

```
//SYSIN DD *
      DEFINE CLUSTER ( -
      NAME(OZAGS1.ALWYN.REVAT.VSAM.KSDS) -
      TRACKS(3,2) -
      RECORDSIZE(80,80) -
      CONTROLINTERVALSIZE(512) -
      FREESPACE(10,10) -
      KEY(4,0) -
      INDEXED -
      )
/*
KEYS(STARTING POS, OFFSET)
OFFSET → STARTING POS - 1
FREESPACE(CI%,CA%)
```

b. Repro.

```
        //sysin dd *
           Repro infile() -
                Outfile()
        /*
```

Note: Repro is allowed any number of times,
Provided,
No duplicate keys
No key out of sequence
(The lowest key in the input dataset must be higher than the highest key in the ksds cluster)

c. Print

```
//sysin dd *
   Print indataset(ozags1.alwyn.revat.vsam.ksds) -
   CHAR -
FROMKEY(1001) TOKEY(2034)
```

While inserting data into KSDS cluster.

- The system sorts it based on the key.
- While doing so, if the CI is full, CONTROL INTERVAL SPLIT happens.
- CI split breaks a CI into 2 halves and moves the records later half into the new CI.
- The new record will be stored accordingly.
- A control interval split might trigger CONTROL AREA split.

Ksds 2 index

- Sequence set – is the low level index for every control area. It contains the highest key of every CI in that CA.
- Index set – it is the high level index for every sequence set. It contains the highest key of every sequence set.

Control interval

R1 =80	R2=80	R3=90	Unused space	Cidf	rdf

Rdf→ record description field

The number of records of similar size

80/2

90

Cidf → CI description field

Rba of the free space

Length of the free space

4. Linear dataset (LDS)

Define

```
//SYSIN DD *
    DEFINE CLUSTER ( -
    NAME(OZAGS1.ALWYN.REVAT.VSAM.LDS) -
    TRACKS(3,2) -
    RECORDSIZE(80,80) -
    CONTROLINTERVALSIZE(512) -
    LINEAR -
    )
/*
```

IDCAMS FUNCTIONS:

1. DEFINE - GDG,CLUSTER,AIX,PATH
2. REPRO - COPY DATA FROM A DATASET TO ANOTHER DATASET
3. PRINT
4. DELETE
5. ALTER
6. LISTCAT
7. VERIFY → Ensures that the dataset is ready and available to be accessed.

8. EXPORT → cluster can be compressed and written in ps(gdg). Not human readable.
9. IMPORT → the exported data can be imported into a cluster to process.
10. BLDINDEX
11. PATH ENTRY