Switch logical logs, change physical logs

Logical Logs -> It can contain the change commands(It's like a redo logs in Oracle)
Physical Logs -> Before Images (Like Undo)

Physical Restore ->If a failure causes the database server to go offline, you must restore all the database server data. This type of restore is a full-system restore.
Logical Restore If you have performed logical log backups, use this procedure to restore the logical log files.

=> If we want to change anything in physical & logical logs structure, we make sure that we need to take the level 0 backup for restoration purpose after changing the architecture of informix

=> Suppose we can delete anything in T1 table, the delete can took place in Logical logs & physical log can't be touched
But once you hit commit, the data can be deleted in physical logs and we can't roll back the data in informix.

To Create the Temp DB Space:
==========

onspaces -c -d tempdbs -k 6 -t -p /opt/informix/rdisk2/drinformix3 -o 2400000 -s 2400000

-o => Offset should be multiple of pagesize

Once we create the db space, we need to add that in onconfig file ->
We can also cross verify whether it's created or not by using -> ***onstat -d*** (Offset must be T for temp DB space)

```
informix@adetrmdb2:/opt/informix> onstat -d

IBM Informix Dynamic Server Version 14.10.FC8WE -- On-Line -- Up 15 days 13:19:38 -- 946176 Kbytes
2024-02-16 14:07:14

Dbspaces
address          number   flags      fchunk   nchunks  pgsize   flags    owner    name
10df41028        1        0x1        1        2        2048     N  BA    informix rootdbs
10e8d0028        2        0x2001     2        1        2048     N TBA    informix tempdbspace
10e8d0268        3        0x1        3        3        6144     N  BA    informix datadbs1
10e8d04a8        4        0x1        5        1        2048     N  BA    informix logdbs1
 4 active, 2047 maximum

Chunks
address          chunk/dbs     offset     size       free       bpages     flags pathname
10df41268        1      1      0          500000     236138               PO-B-- /opt/informix/rdsk/rinformix21
10e8d2028        2      2      500000     250000     249947               PO-B-- /opt/informix/rdsk/rinformix21
10e8d3028        3      3      0          833333     4                    PO-B-- /opt/informix/rdsk/rinformix22
10e8d4028        4      3      0          833333     132435               PO-B-- /opt/informix/rdsk/rinformix23
10e8d5028        5      4      0          2500000    1999947              PO-B-- /opt/informix/rdsk/rinformix24
10e8d6028        6      3      0          833333     833330               PO-B-- /opt/informix/rdsk/rinformix25
10e8d1028        7      1      0          2499999    2491076              PO-B-- /opt/informix/rdsk/rinformix27
 7 active, 32766 maximum

NOTE: The values in the "size" and "free" columns for DBspace chunks are
      displayed in terms of "pgsize" of the DBspace to which they belong.
```

The offset value should be change depending on the operating system

T**o change the same in ONCONFIG file:**
---
onmode -wf DBSPACETEMP=tempdbspace,tempdbs

To check whether updated in config or not -> ***onstat -c | grep DBSPACE***

If we want to check if our created db space is working or not:
---
1. First we need to reset the values -> ***onstat -z***
2. To check weather all values has been reset or not -> ***onstat -p & onstat -D***

The temp db space is used for sorting and temp tables
dbaccess -> To go to the db

To create the temp table:

create temp tab1 (col1 char(10), col2 integer) with no log

No log -> To avoid the data to be inserted or deleted in transactional logs

To insert the data in tab1:

insert into tab1
select t.tabname, t.tabid
from systables as t, syscolumns c, syscolumns c1 ==> This won't be created in all existing temp db spaces

select t.tabname, t.tabid
from systables as t, syscolumns c, syscolumns c1
into temp tab1 with no log ==> This can create tab1 table in both temp db spaces

If we want to check whether dbspace is AUTO Extendable or not:

onstat -c | grep AUTO

onstat -c | grep EXTEND

Synonym:
=======
If we want to access data from table with synonym, we will get the same data as main table

create table kalu (
f1 int,
f2 char(42));

insert into values kalu (1,"test");
insert into values kalu (2,"test2");

create synonym sales for kalu;

select * from kalu;
select * from sales; -> Both can retrive the same data

If we want to see the owner of table:

select *
from systables
where tabname = "kalu";

It is also possible to create table kalu with the different owner but the table name is unique.
The lock level "P" indicates that PAGE -> We can not create the table with lock level PAGE
We need to change the lock level mode to row:

alter table kalu lock mode (row);

If we want to change the Page level lock mode to Row level, we need to check the onconfig file

onstat -c | grep DEF

To check which object is located at which DB space:

oncheck -pe|more

To check the physical & logical logs we can use -> ***onstat -l***

phybegin	physize	phypos	phyused	%used
1:263	250000	164992	354	0.14

address	number	flags	uniqid	begin	size	used	%used
10e0b8890	6	U-B-----	29493	7:53	5000	5000	100.00

begin -> **7 indicates the chunks**
Dbspaces

address	number	flags	fchunk	nchunks	pgsize	flags	owner	name
10df41028	1	0x1	1	2	2048	N BA	informix	rootdbs
10e8c8aa8	2	0x2001	2	1	2048	N TBA	informix	tempdbs
10e8c8ce8	3	0x1	3	2	6144	N BA	informix	datadbs1
10e8c9028	4	0x1	5	2	6144	N BA	informix	datadbs2
10e8c9268	5	0x1	7	2	2048	N BA	informix	logdbs
10e8c94a8	6	0x2001	9	1	2048	N TBA	informix	tempdbspace

If we wanted to check the long transactions & rollback time:

onstat -x

```
informix@adetrmdb2:/opt/informix> onstat -x

IBM Informix Dynamic Server Version 14.10.FC8WE -- On-Line -- Up 15 days 13:36:29 -- 946176 Kbytes
2024-02-16 14:24:04

Transactions
                                                                                    est.
address          flags userthread       locks  begin_logpos      current logpos    isol    rb_time  retrys coord
10e09c028        A---- 10e053028        0      -                 -                 COMMIT  -        0
10e09c3a8        A---- 10e053928        0      -                 -                 COMMIT  -        0
10e09c728        A---- 10e054228        0      -                 -                 COMMIT  -        0
10e09caa8        A---- 10e054b28        0      -                 -                 COMMIT  -        0
10e09ce28        A---- 10e055428        0      -                 -                 COMMIT  -        0
10e09d1a8        A---- 10e055d28        0      -                 -                 COMMIT  -        0
10e09d528        A---- 10e056628        0      -                 -                 COMMIT  -        0
```

onstat -u | grep 10e09c028 -> We will get exact SID of transaction address

onstat -g ses 10 -> we will get complete details of unix process ID

onmode -l -> To move/switch the logical logs from current to next

o***nmode -c*** -> To move the checkpoint from previous to current logical log