Double Buffering
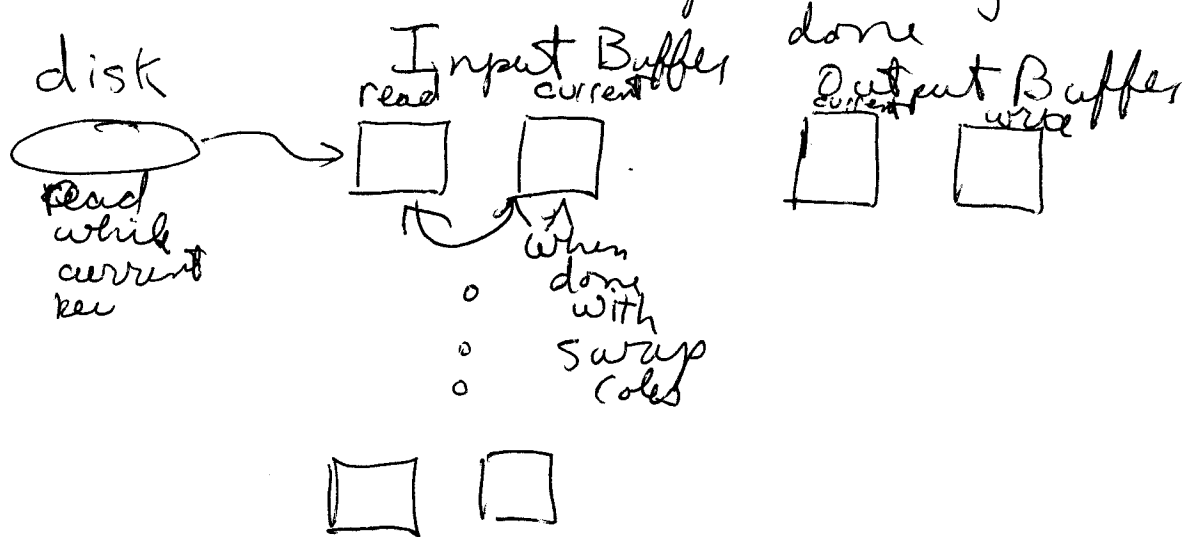
Idea: want to avoid having to suspend computations while an I/O request is being filled

Want to keep CPU busy while I/O's being done

disk Input Buffer Output Buffer

read current current write

o
o   When done
o   with swap codes

End External Sorting

Start Query Evaluation & System Catalog

<u>System Catalog</u> — the data base the DBMS uses to maintain information about itself. i.e., what tables, indexes, users the DBMS is managing.

Other names for system catalog are the <u>data dictionary</u>, <u>catalog tables</u>, or just <u>catalog</u>.   ← Oracle

What kind of information is stored?

For each table:

    Its table name, the file name (or some identifier) and the file structure (heap file)

    The attribute name and type of each of its attributes

    The index name of each index on the table

    The integrity constraints on the table

For each index:

    The index name & structure (ex B^+-index)
    The search key attributes

For each view:

    Its view name & definition

Info about users — won't talk too much about

Some statistics about tables & indexes also stored? (only occasionally updated)

cardinality: number of tuples in a rel'n.

Size: number of pages in a table

Index Cardinality: # distinct keys in an index

Index Size: # of leaf pages used to store index

Index height: number of non leaf level

Index Range: high & low key

Oracle data dictionary has three catagories of views on its data dictionary: all, user, & DBA.   all - all object user has privelege on. user - owned by user
DBA - for administrator

EX) Can query a ~~~~ catalog like any other view!

```
SELECT *
FROM   ALL_CATALOG
WHERE  OWNER = "POLLETT";
```

↑ all objects I am allowed to see owned by Pollett

Information about columns in a table can be ~~~~ discovered using

```
USER_TAB_COLUMNS
```

About tables...

```
USER_TABLES
```

About indexes. –

```
USER_INDEXES
```

About views. –

```
USER_VIEWS
```

To force recalculation of statistics of a table can use

```
ANALYZE TABLE MY_TABLE
COMPUTE STATISTICS;
```


$i_7 * 128 + a_6 * 64 + a_5 * 32 + + 9_0$

128

0