

Execution of a SELECT statement

- SELECT col1 FROM Tab1 WHERE col2 = ...

How does the DBMS find the pieces of Data on disk?

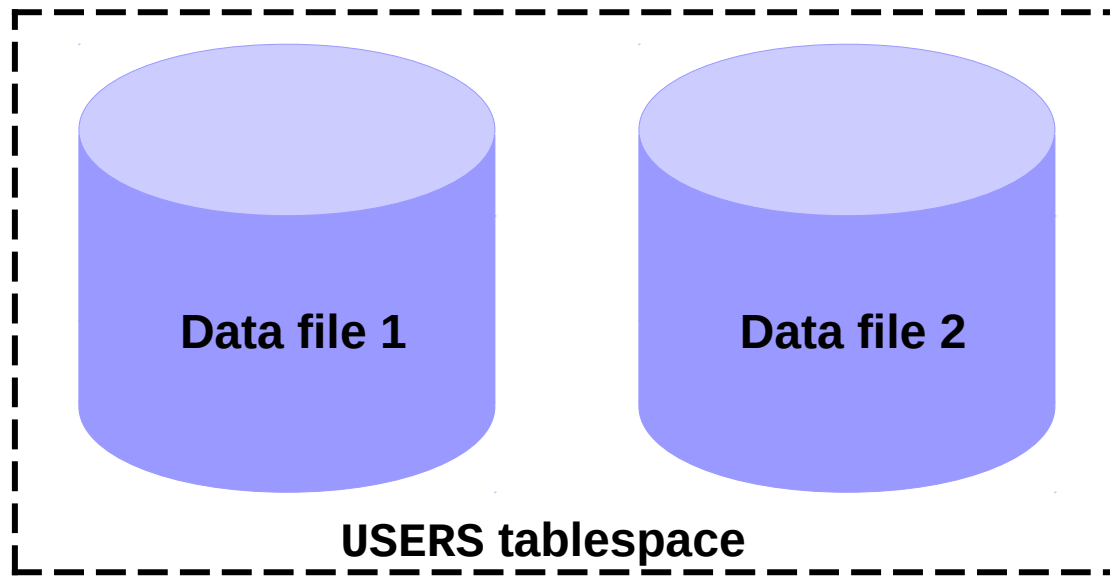
Data blocks

Records

Fields

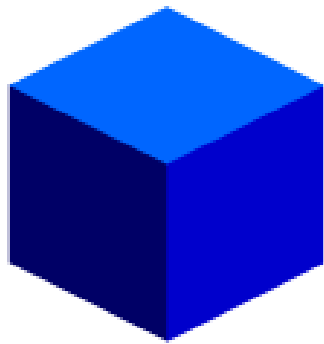
Tablespaces and Data Files

- Tablespaces consist of one or more data files.
- Data files belong to only one tablespace.



Segments, Extents, and Blocks

- Segments exist within a tablespace.
- Segments are made up of a collection of extents.
- Extents are a collection of data blocks.
- Data blocks are mapped to disk blocks.



Segment



Extents

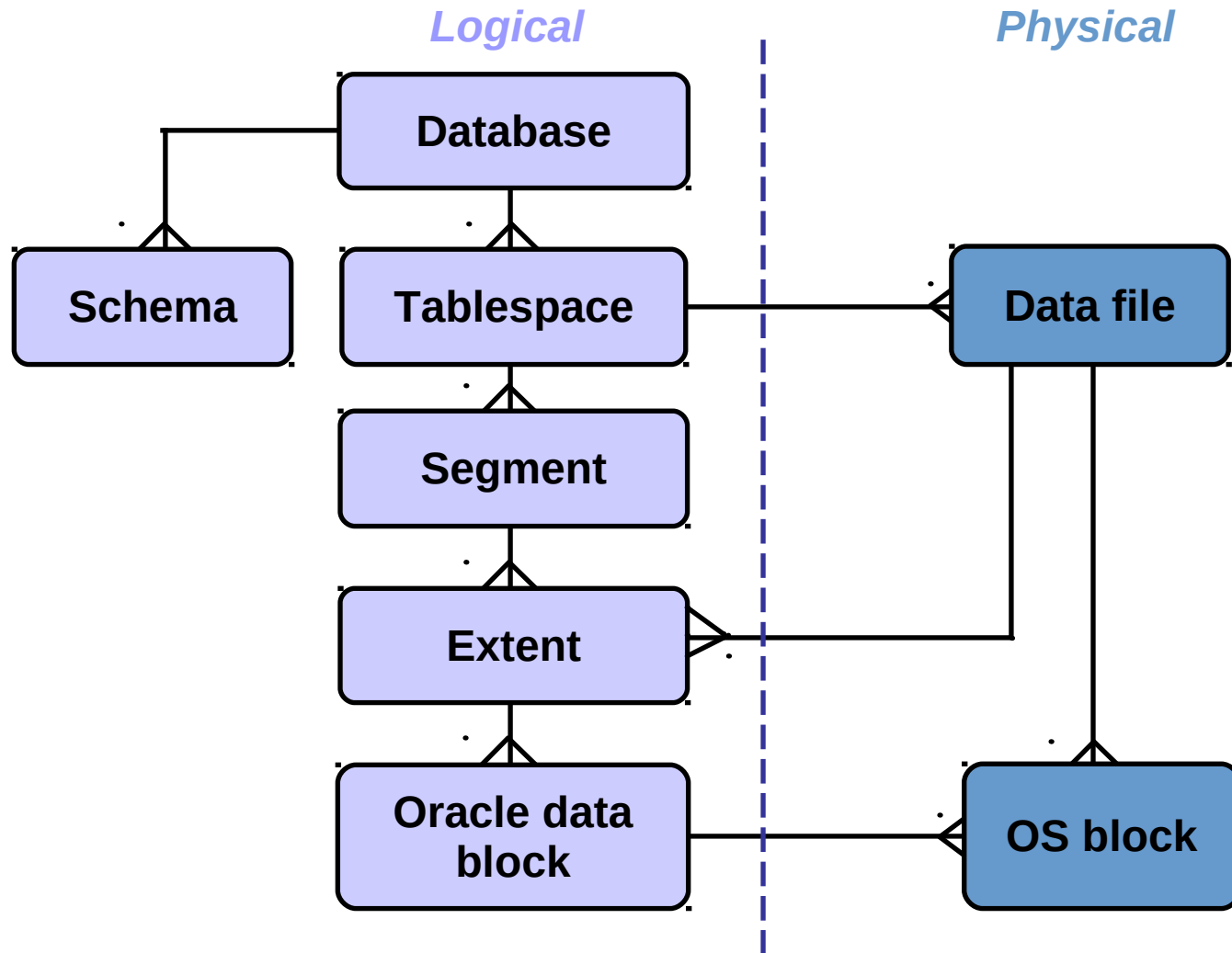


**Data
blocks**

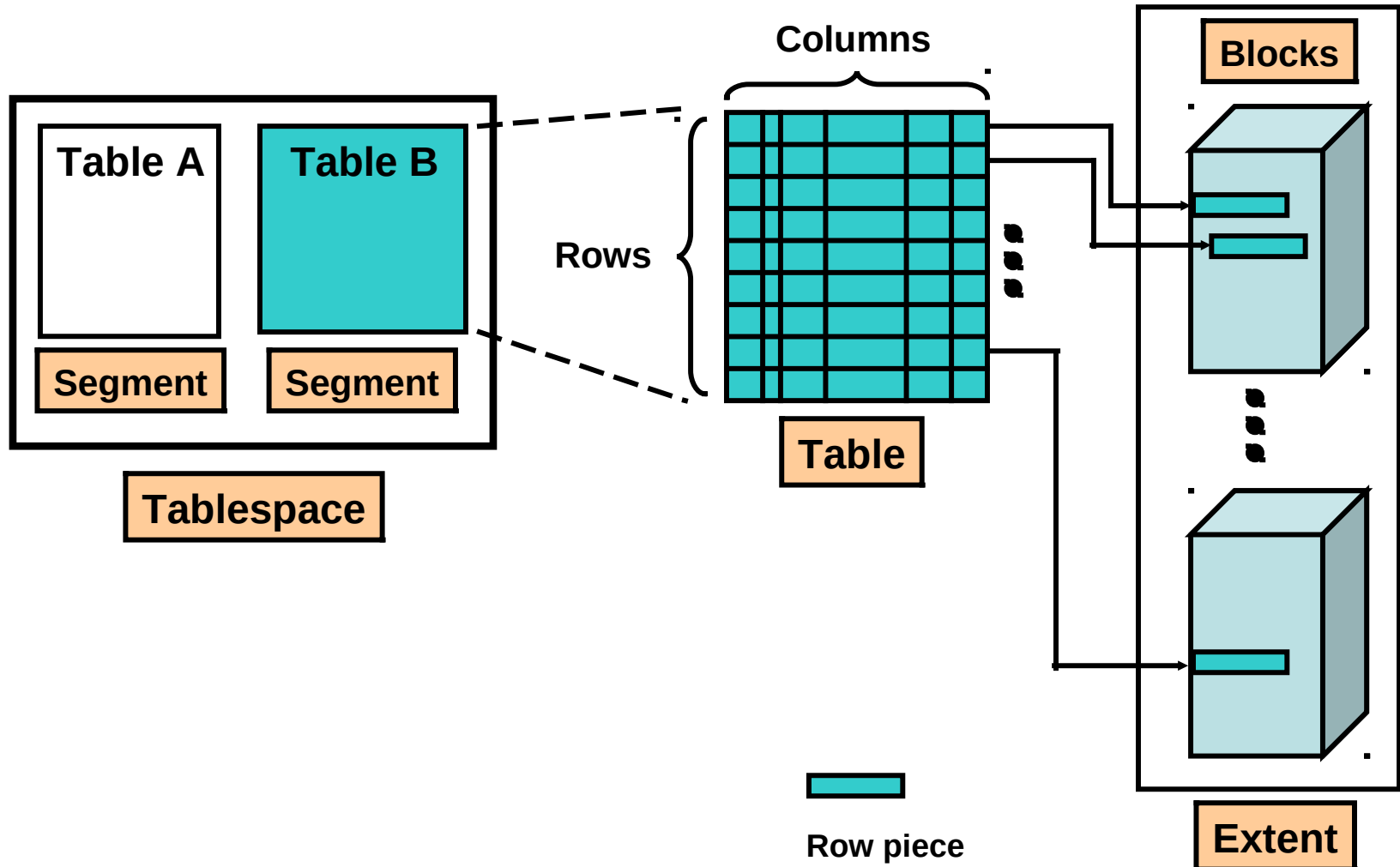


**Disk
blocks**

Logical and Physical Database Structures

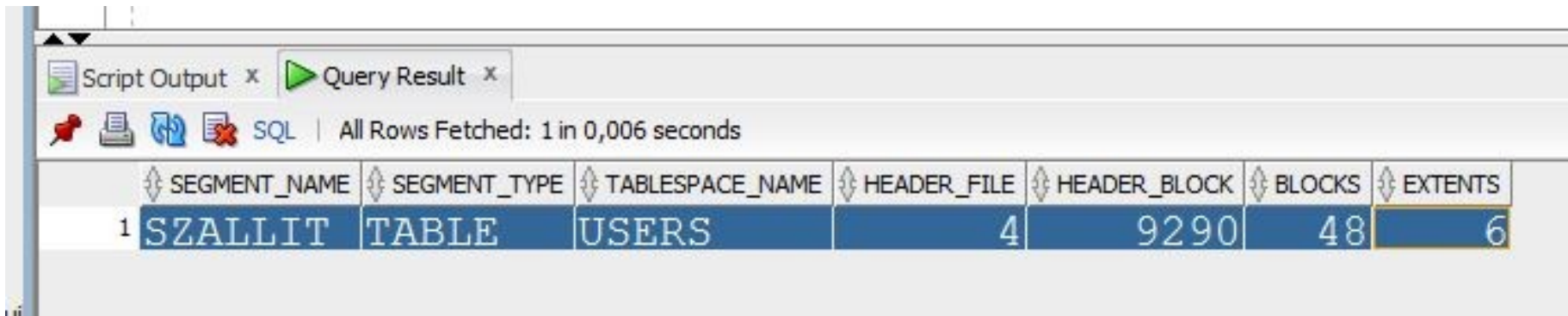


How Table Data Is Stored



Where Table Data is Stored?

```
SELECT segment_name, segment_type, tablespace_name,  
       header_file, header_block, blocks, extents  
FROM dba_segments where owner='NIKOVITS'  
AND segment_name='SZALLIT' AND  
segment_type='TABLE';
```

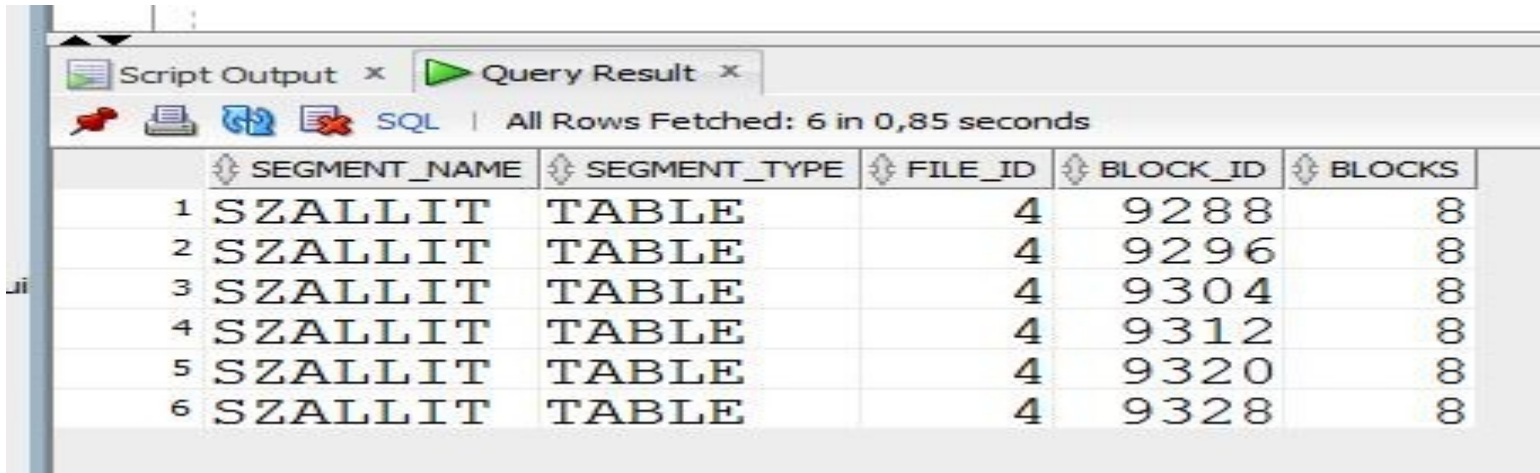


The screenshot shows a database query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying a table with 7 columns: SEGMENT_NAME, SEGMENT_TYPE, TABLESPACE_NAME, HEADER_FILE, HEADER_BLOCK, BLOCKS, and EXTENTS. The table contains one row of data for the segment named 'SZALLIT'.

	SEGMENT_NAME	SEGMENT_TYPE	TABLESPACE_NAME	HEADER_FILE	HEADER_BLOCK	BLOCKS	EXTENTS
1	SZALLIT	TABLE	USERS	4	9290	48	6

Where Table Data is Stored?

```
SELECT segment_name, segment_type,  
       file_id, block_id, blocks  
FROM dba_extents where owner='NIKOVITS'  
AND segment_name='SZALLIT' AND  
segment_type='TABLE';
```

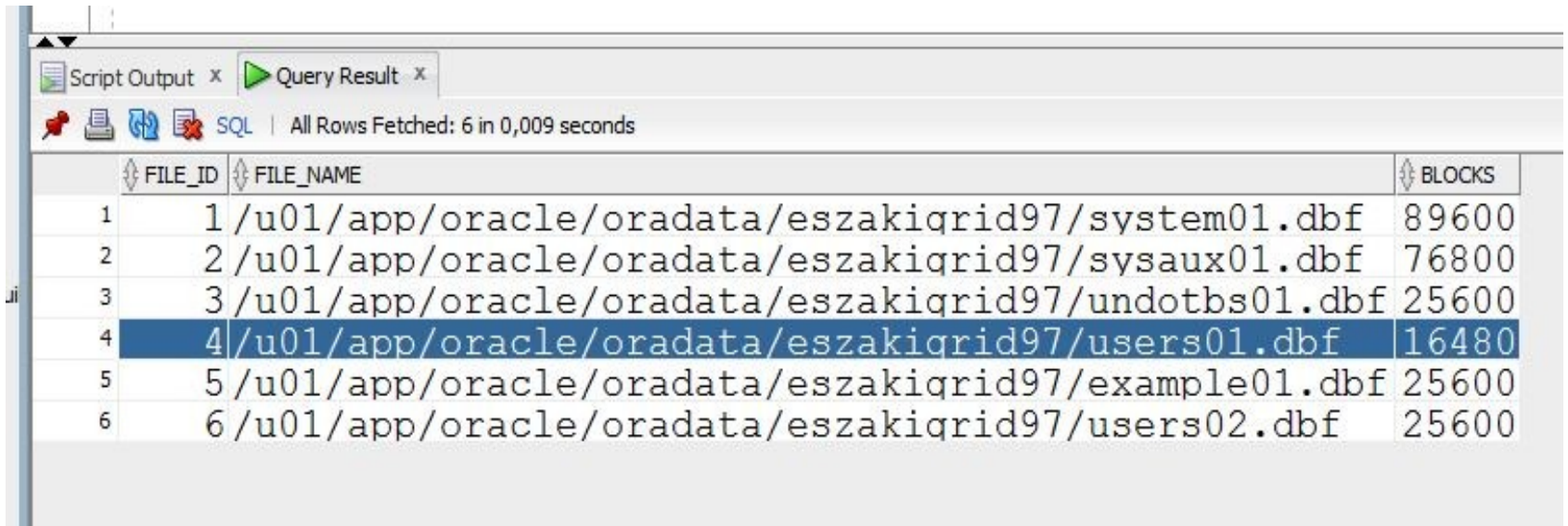


The screenshot shows a database query result window with the following data:

	SEGMENT_NAME	SEGMENT_TYPE	FILE_ID	BLOCK_ID	BLOCKS
1	SZALLIT	TABLE	4	9288	8
2	SZALLIT	TABLE	4	9296	8
3	SZALLIT	TABLE	4	9304	8
4	SZALLIT	TABLE	4	9312	8
5	SZALLIT	TABLE	4	9320	8
6	SZALLIT	TABLE	4	9328	8

Where Table Data is Stored?

```
SELECT file_id, file_name, blocks  
FROM dba_data_files;
```



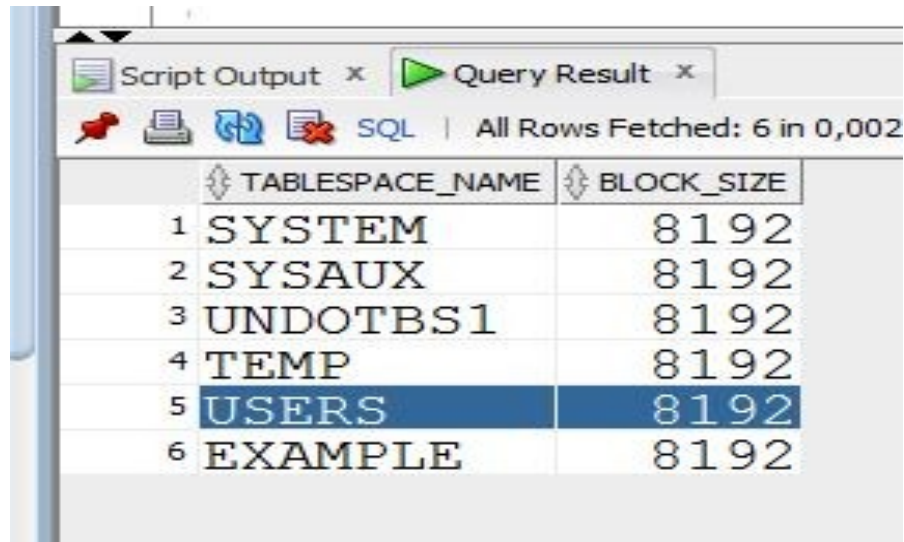
The screenshot shows a query result window in Oracle SQL Developer. The window has tabs for 'Script Output' and 'Query Result'. Below the tabs, there are icons for a pin, a printer, a refresh, and a SQL icon, followed by the text 'All Rows Fetched: 6 in 0,009 seconds'. The query result is displayed in a table with three columns: FILE_ID, FILE_NAME, and BLOCKS. The table contains six rows of data. The fourth row, which corresponds to the 'users01.dbf' file, is highlighted with a blue background.

FILE_ID	FILE_NAME	BLOCKS
1	/u01/app/oracle/oradata/eszakiqrid97/system01.dbf	89600
2	/u01/app/oracle/oradata/eszakiqrid97/sysaux01.dbf	76800
3	/u01/app/oracle/oradata/eszakiqrid97/undotbs01.dbf	25600
4	/u01/app/oracle/oradata/eszakiqrid97/users01.dbf	16480
5	/u01/app/oracle/oradata/eszakiqrid97/example01.dbf	25600
6	/u01/app/oracle/oradata/eszakiqrid97/users02.dbf	25600

Which part of the Datafile?

(What is the block size?)

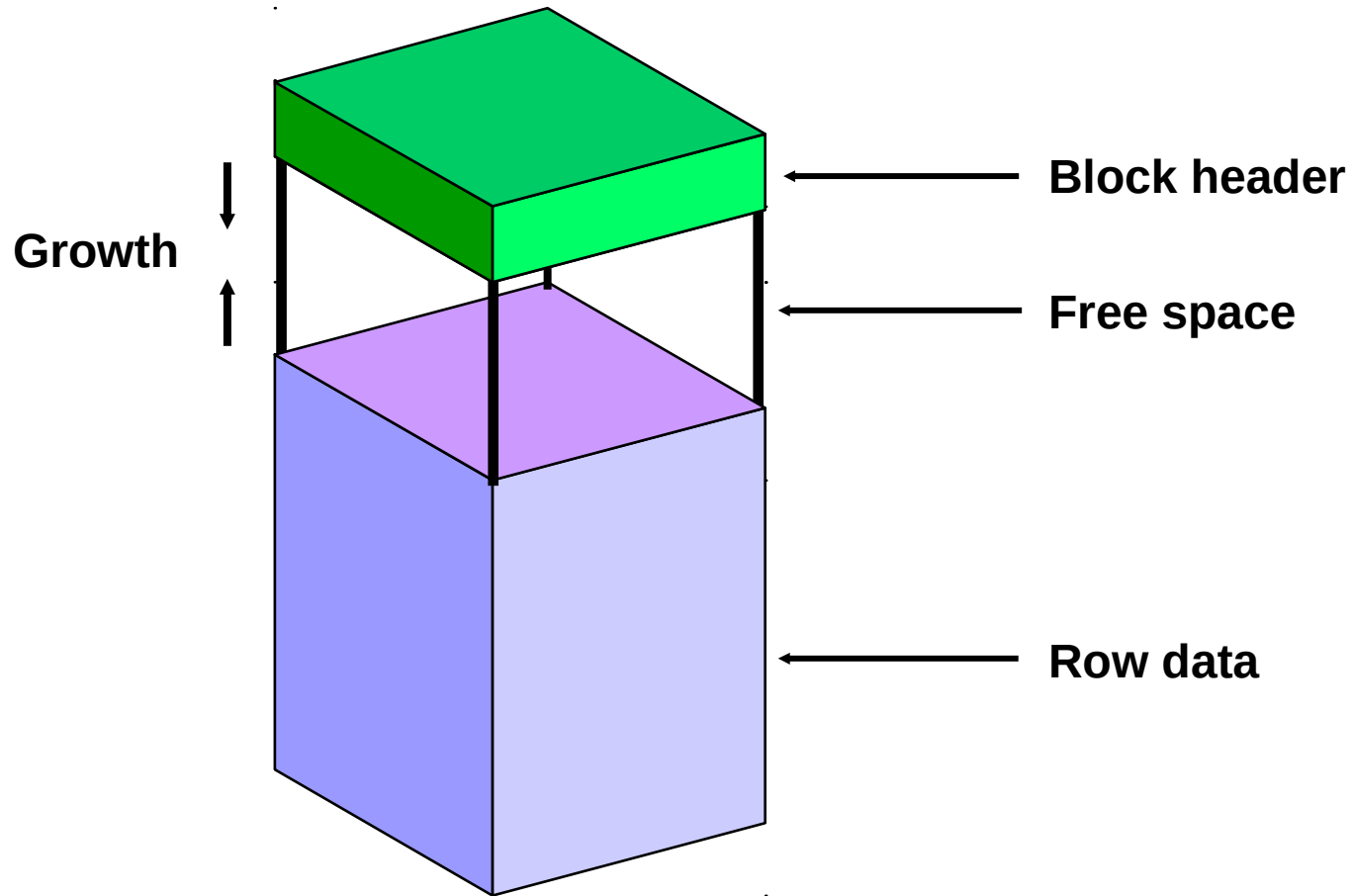
```
SELECT tablespace_name, block_size  
FROM dba_tablespaces;
```



The screenshot shows a SQL query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying a table with two columns: 'TABLESPACE_NAME' and 'BLOCK_SIZE'. The table contains six rows of data, with the fifth row, 'USERS', highlighted in blue. The status bar at the top indicates 'All Rows Fetched: 6 in 0,002'.

	TABLESPACE_NAME	BLOCK_SIZE
1	SYSTEM	8192
2	SYSAUX	8192
3	UNDOTBS1	8192
4	TEMP	8192
5	USERS	8192
6	EXAMPLE	8192

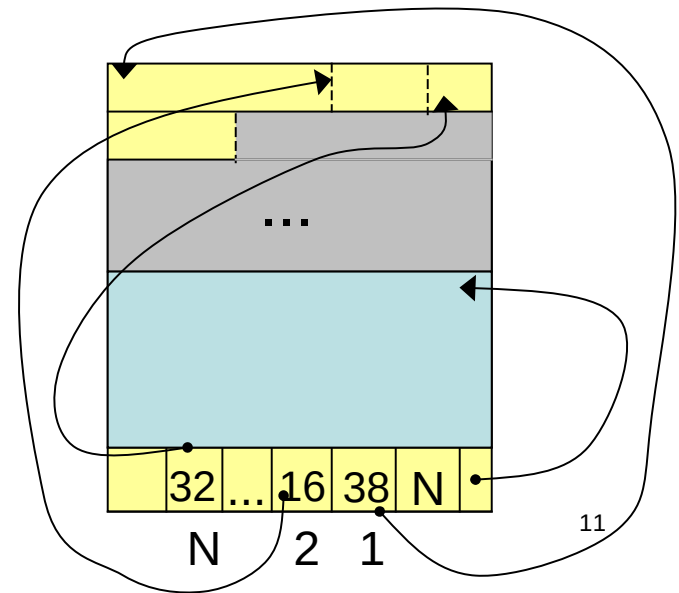
Anatomy of a Database Block



Variable-length records

(finding records within blocks)

- When do we have a file with variable-length records?
 - file contains records of multiple tables
 - create table t (field1 int, field2 varchar2(n))
- Problems:
 - Holes created upon deletion have variable size
 - Find large enough free space for new record
- Could use previous approaches: maximum record size
 - a lot of space wasted
- **Use slotted page structure**
 - Slot directory
 - Each slot storing offset, size of record
 - Record IDs: page number, slot number



Record Organization

(finding fields within records)

- Fixed-length record formats
 - Fields stored consecutively
- Variable-length record formats
 - Array of offsets
 - NULL values when start offset = end offset

