



ubTools

ubSQL 4.2.2



Introduction

- What is ubSQL ?
- Unload Architecture
- Commands
- Parameters



What is ubSQL ?

- Command line application.
- Unloads Oracle data to text files.
- Runs DML,DDL statements.



Supported Objects

- Any objects that work within a SELECT statement.



Supported Data Types

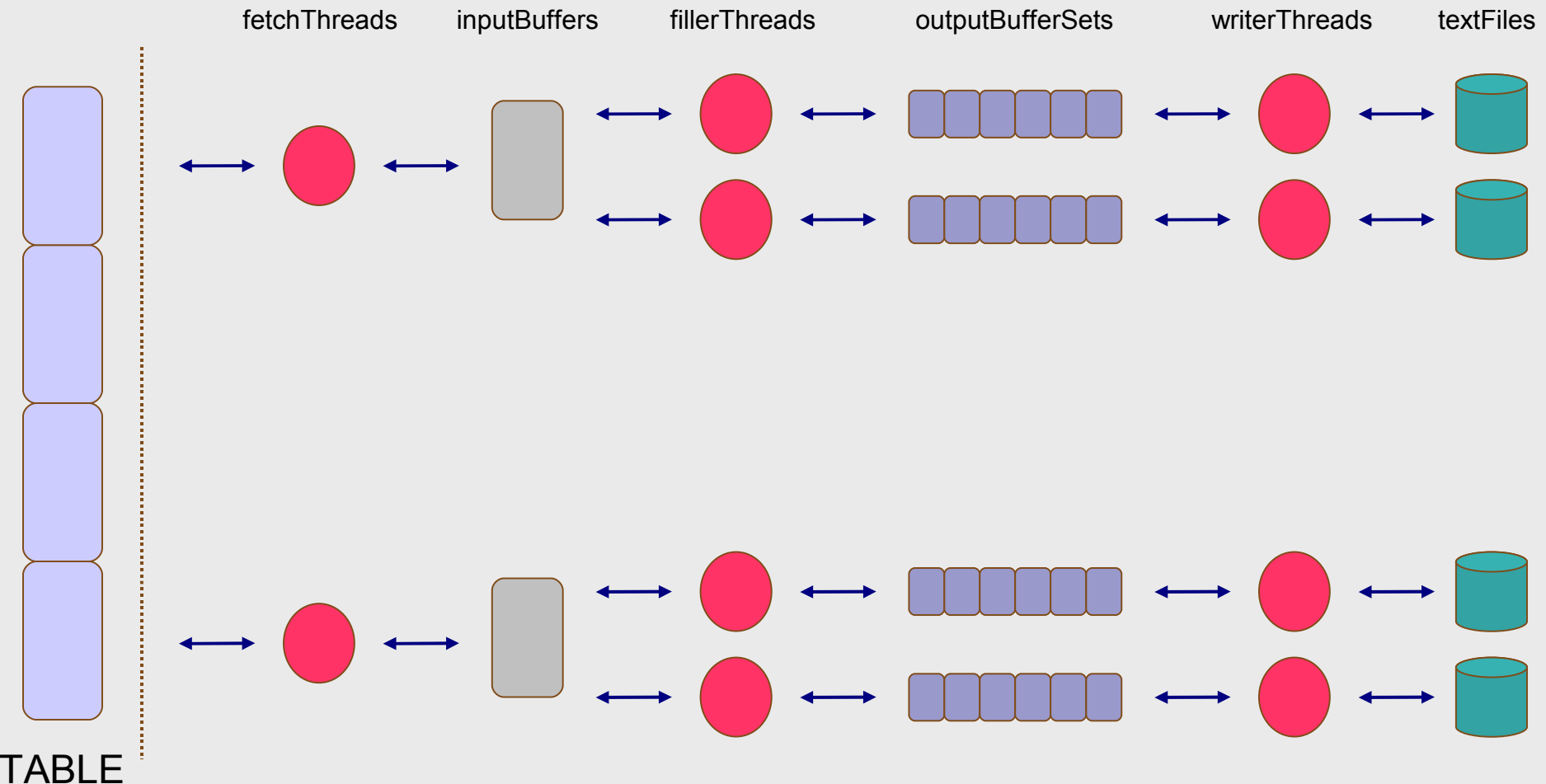
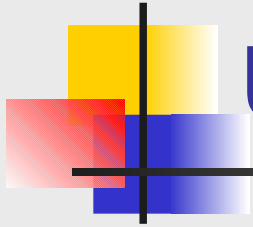
- All Numeric data types
- CHAR, VARCHAR2
- DATE
- RAW
- TIMESTAMP, TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH LOCAL TIME ZONE
- INTERVAL YEAR TO MONTH, INTERVAL DAY TO SECOND



Easy to port

- Used POSIX compatible system calls.
- Easy to port to different platforms.

Unload Architecture





Commands

- `connect <username>/<password>@<tnsAlias>;`
- `show parameters;`
- `set <paramName>=<paramValues>;`
- `unload <any syntax used in SELECT statement>;`
- `create sql_loader control_file ...;`
- `exit;`
- Any Oracle DML,DDL statements.



CONNECT Command

```
oracle@linux:~/ubSQL/src> ./ubSQL
ubSQL: Release 4.2.2 - Production - _POSIX_VERSION: 200112
Copyright (c) 2005, ubTools, Adana/Turkey. All rights reserved.
Not connected to an instance.

ubSQL> connect dunal/dunal@dunal;
Connected.

ubSQL>
```

SHOW Command

```
ubSQL> show parameters;
```

NAME	MODIFIABLE	VALUE
FETCH_THREADS	TRUE	1
TABLE_SPLIT_LEVEL	TRUE	AUTO
NLS_EXPANSION_FACTOR	TRUE	1
EXIT_ON_ERROR	FALSE	FALSE
SESSION_PARAMETERS	TRUE	nls_date_format=RRDDSSSSS
ARRAY_FETCH_ROW_COUNT	TRUE	250
COLUMN_DELIMITER	TRUE	
LINE_DELIMITER	TRUE	x'0A'
FEEDBACK_ROWS	TRUE	1000000
LOG_FILE_NAME	FALSE	ubSQL.log
LOG_FILE_LEVEL	FALSE	3
OUTPUT_BUFFER_SIZE	TRUE	65536
OUTPUT_BUFFERS_PER_FILE	TRUE	100
OUTPUT_FILE_NAMES	TRUE	../output/4.txt
LICENSE_KEY	FALSE	EVALUATION

```
ubSQL>
```



SET Command

```
ubSQL> SET OUTPUT_FILE_NAMES=a.txt,b.txt,c.txt;
```

```
Parameter altered.
```

```
ubSQL> set SESSION_PARAMETERS=nls_date_format=RRDDSSSSS  
      timed_statistics=true;
```

```
Parameter altered.
```

```
ubSQL> set LINE_DELIMITER=X'0A';
```

```
Parameter altered.
```

```
ubSQL>
```



UNLOAD Command

```
ubSQL> unload s4.f1,s4.f2 from system.sil4 s4,system.sil8 s8 where  
s4.f1=s8.f1;
```

```
1 rows unloaded by ThreadSet 0.
```

```
ThreadSet:0, FetchThread, non-idle elapsed time...: 0.00
```

```
ThreadSet:0, FillerThread:0, non-idle elapsed time...: 0.00
```

```
ThreadSet:0, WriterThread:0, non-idle elapsed time...: 0.00
```

```
Service time...: 0.04
```

```
Wait time.....: 0.16
```

```
Elapsed time...: 0.16
```

```
ubSQL>
```



CREATE SQL_LOADER Command

```
ubSQL> create sql_loader control_file=../0/test.ctl parameters="
options(direct=true)
load data
infile '../0/4.txt'
into table dunal.test
append"
for select * from sil_date;

CREATE completed.

ubSQL>
```



Oracle DML/DDL Commands

```
ubSQL> insert into system.sil8 values(1);
```

```
1 rows inserted.
```

```
ubSQL> drop table system.sil8;
```

```
DROP completed.
```

```
ubSQL>
```



EXIT Command

```
ubSQL> exit;
```

```
oracle@linux:~/ubSQL>
```

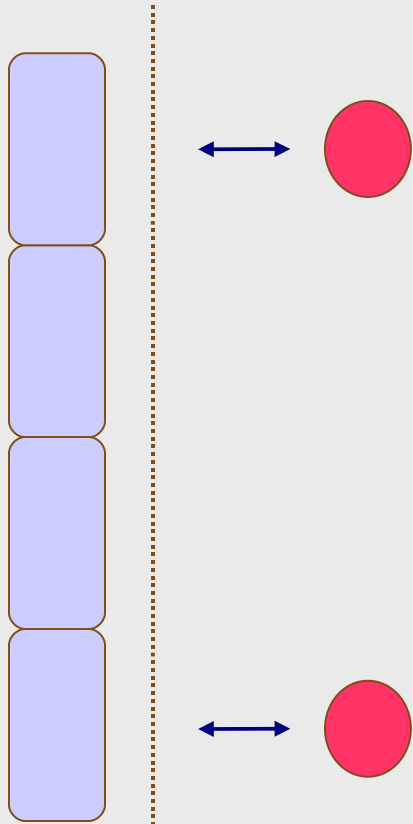


Parameter: FETCH_THREADS

- Definition: Number of Oracle unload sessions.
- Implementation: Each session reads its range and unloads to its output files.

Parameter: FETCH_THREADS

fetchThreads



FETCH_THREADS = ORACLE UNLOAD SESSIONS

TABLE



Parameter: TABLE_SPLIT_LEVEL

- Definition: Splits table if FETCH_THREADS parameter is greater than 1.
- Implementation: It has 4 different options:

TABLE: Split operation is done in table level.

PARTITION: Split operation is tried to be done in partition level.

SUBPARTITION: Split operation is tried to be done in subpartition level.

AUTO: If statement includes *partition* word then split operation is done in partition level. Else if statement includes *subpartition* word then split operation is done in subpartition level. Otherwise, split operation is done in table level.



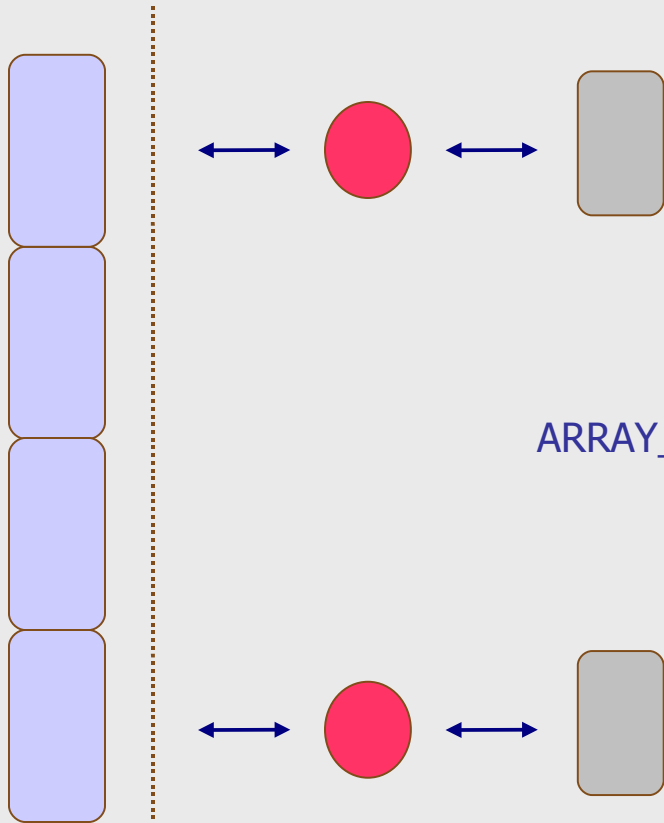
Parameter:

ARRAY_FETCH_ROW_COUNT

- Definition: Number of rows per FETCH.
- Implementation: Number of ARRAY_FETCH_ROW_COUNT returned per FETCH.

Parameter: ARRAY_FETCH_ROW_COUNT

fetchThreads inputBuffers



`ARRAY_FETCH_ROW_COUNT` = Rows per fetch into each inputBuffer.

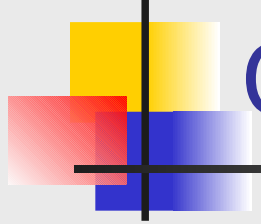


Parameter:

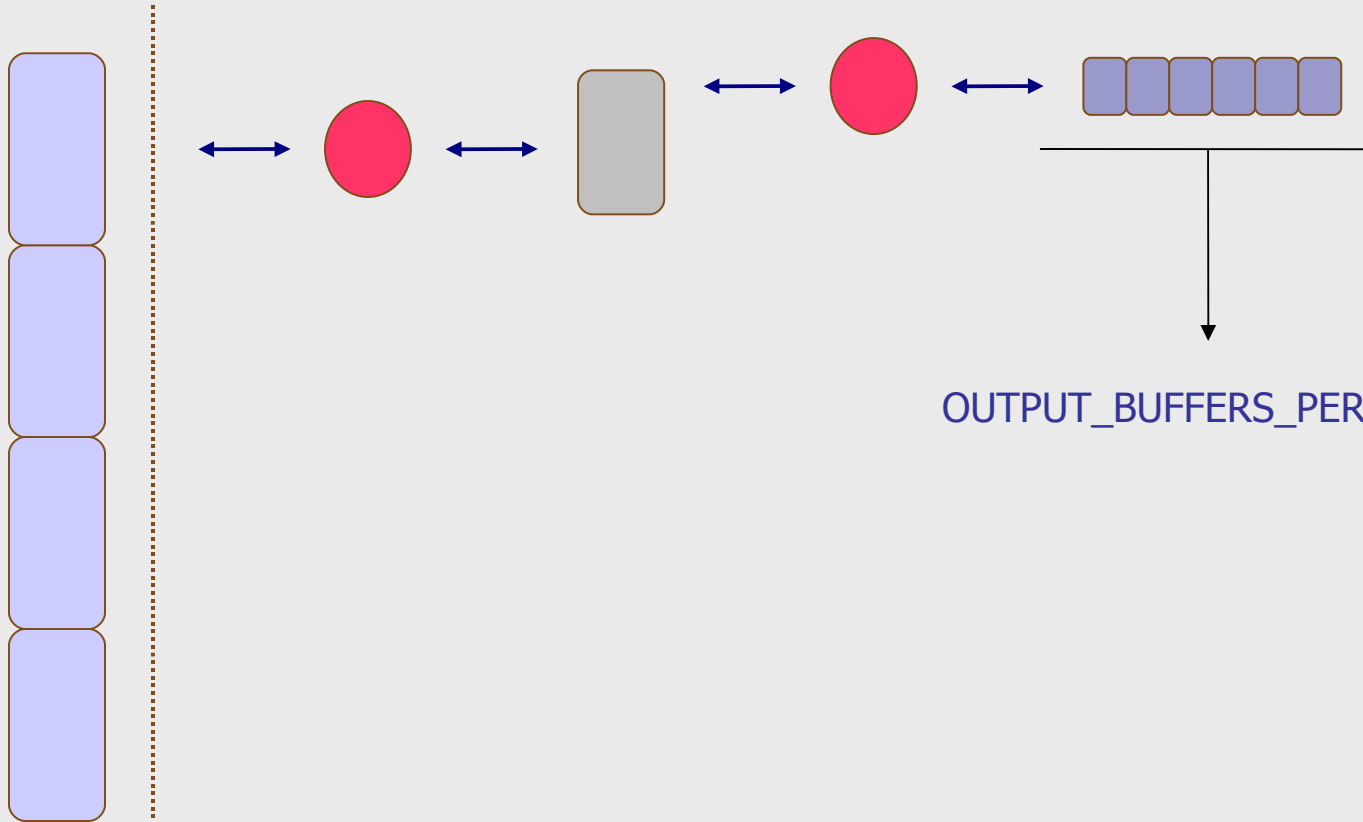
OUTPUT_BUFFERS_PER_FILE

- Definition: Number of output buffers per output file.
- Implementation: If one output buffer is filled, another output buffer is used.

Parameter: OUTPUT_BUFFERS_PER_FILE



fetchThreads inputBuffers fillerThreads outputBufferSets



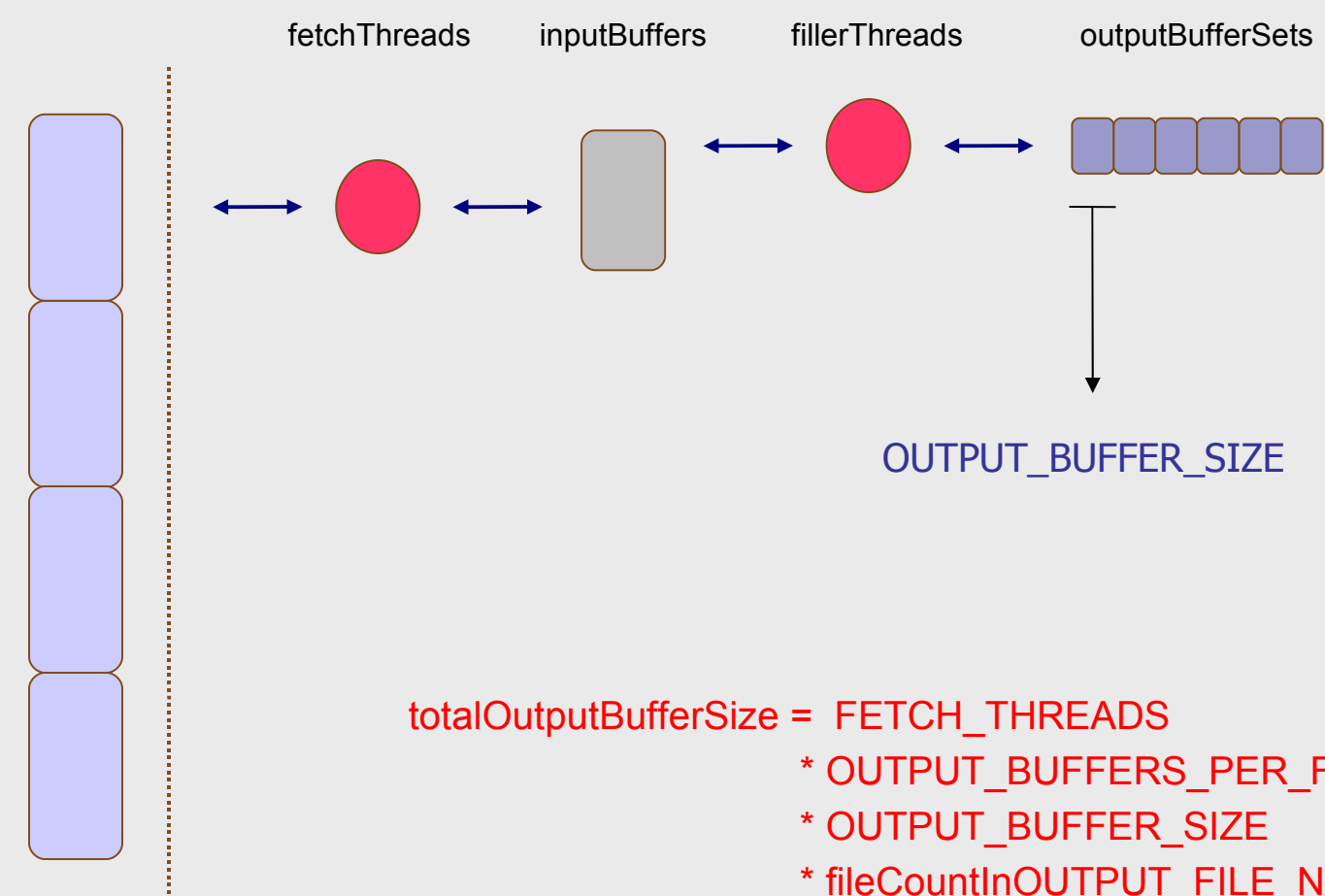
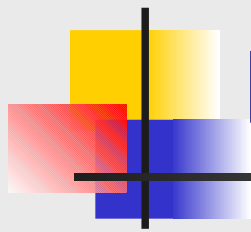
OUTPUT_BUFFERS_PER_FILE



Parameter: OUTPUT_BUFFER_SIZE

- Definition: Number of bytes of each output buffer.
- Implementation: If output buffer is not enough to hold new data, output buffer is written to output file.

Parameter: OUTPUT_BUFFER_SIZE



$$\begin{aligned}
 \text{totalOutputBufferSize} &= \text{FETCH_THREADS} \\
 &\quad * \text{OUTPUT_BUFFERS_PER_FILE} \\
 &\quad * \text{OUTPUT_BUFFER_SIZE} \\
 &\quad * \text{fileCountInOUTPUT_FILE_NAMES}
 \end{aligned}$$

TABLE

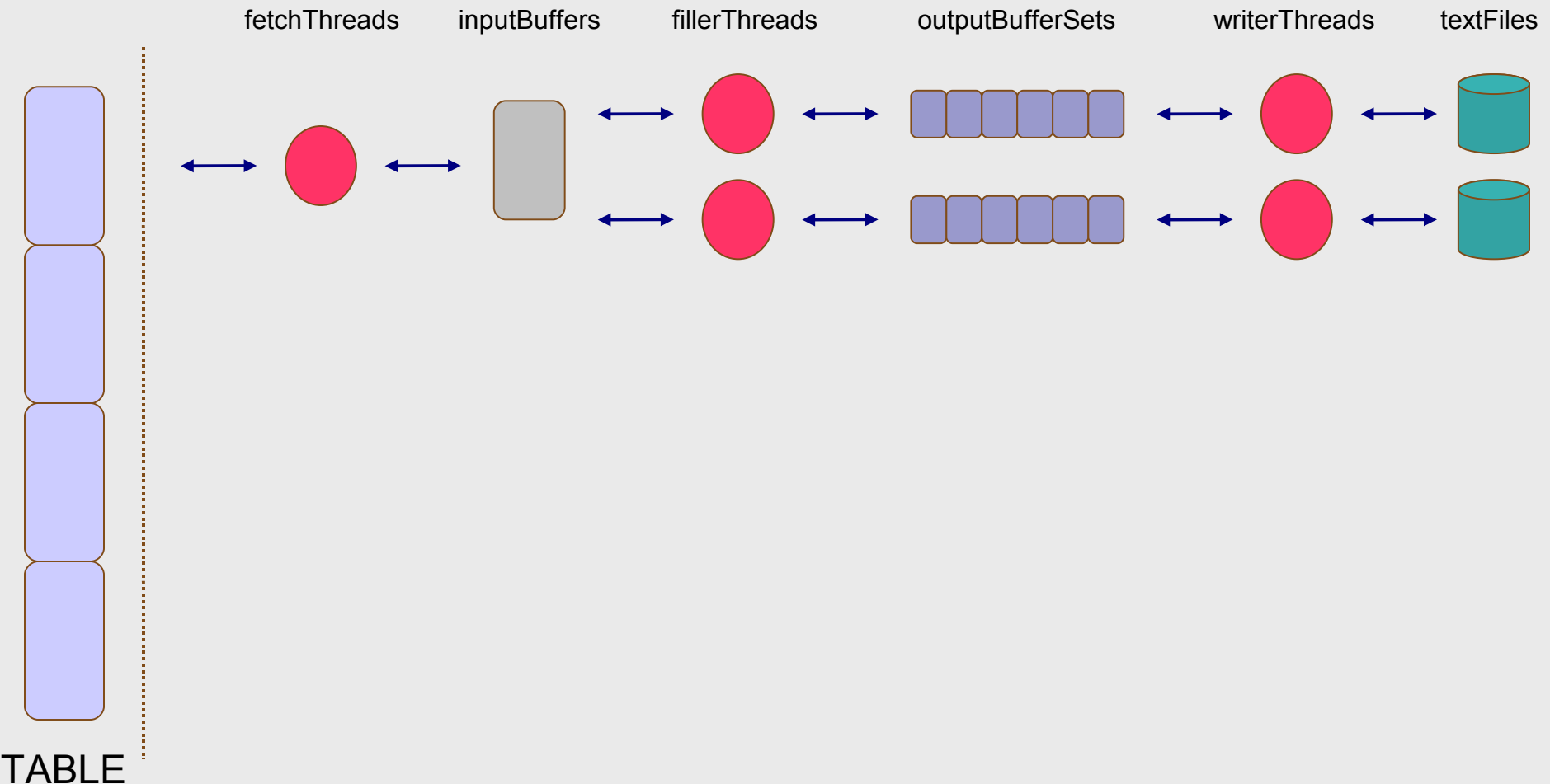
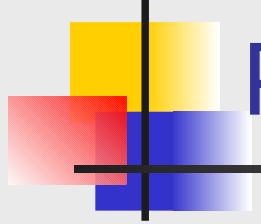


Parameter: OUTPUT_FILE_NAMES

- Definition: Output file names.
- Implementation: Each file has its own buffers grouped by OUTPUT_BUFFERS_PER_FILE that is called Buffer Set.

File names are appended by session thread ID if FETCH_THREADS is greater than 1.

Parameter: OUTPUT_FILE_NAMES





Parameter: NLS_EXPANSION_FACTOR

- Definition: NLS expansion factor of character types.
- Implementation: Byte size of character types in database is multiplied by NLS_EXPANSION_FACTOR to allocate large enough buffers in client, in case of buffer expansion during multibyte NLS conversion between client and database .



Parameter: EXIT_ON_ERROR

- Definition: Exit behavior whenever an error occurred.
- Implementation: If EXIT_ON_ERROR=TRUE, whenever an error occurs, application exits. If EXIT_ON_ERROR=FALSE, application exits only an internal error occurs.

If application exits successfully, it returns 0; otherwise it returns a non zero value.



Parameter: SESSION_PARAMETERS

- Definition: Session parameters of Oracle sessions.
- Implementation: ALTER SESSION SET command is invoked internally with the value of SESSION_PARAMETERS.

If ALTER SESSION SET is invoked by the command line, it's set for the current command line session only. It's NOT propagated to unload sessions since unload sessions are created during unload, and released after unload completed.



Parameter: COLUMN_DELIMITER

- Definition: Column delimiter.
- Implementation: It can be a single character or a string. HEXADECIMAL values are not accepted.



Parameter: LINE_DELIMITER

- Definition: Line delimiter.
- Implementation: It must be a single character. HEXADECIMAL values are accepted if it is enclosed in x" or X", such as X'0A'.



Parameter: FEEDBACK_ROWS

- Definition: Number of rows to give feedback about unload operation.
- Implementation: After Each FEEDBACK_ROWS unloaded, a message to screen is printed to give feedback.



Parameter: LICENSE_KEY

- Definition: License key.
- Implementation: It has 3 different options:

EVALUATION: Makes evaluation.

Limits FETCH_THREADS parameter up to 2, and limits number of unloaded rows per FETCH_THREADS up to 10,000,000. That means total unloaded row count limit is 20,000,000.

All other features are available.

NULL: Generates a pre-license key. ubTools provides a production license key by pre-license key.

<licenseKEY>: Production license key provided by ubTools.



Online Demo



ubTools

ubSQL 4.2.2