X$BH Fixed Table Buffer Cache Diagram  
  
 Column     Type Description  
 ~~~~~~     ~~~~~ ~~~~~~~~~~~  
 ADDR        RAW(4) Hex address of the Buffer Header.  
 INDX        NUMBER Buffer Header number  
 BUF#        NUMBER  
 HLADDR      RAW(4) Hash Chain Latch Address  
 See . ADDR  
  
 LRU\_FLAG    NUMBER     8.1+ LRU flag  
            KCBBHLDF   0x01     8.1  LRU Dump Flag used in debug print routine  
            KCBBHLMT   0x02     8.1  moved to tail of lru (for extended stats)  
            KCBBHLAL   0x04     8.1  on auxiliary list  
            KCBBHLHB   0x08     8.1  hot buffer - not in cold portion of lru  
  
 FLAG        NUMBER  
            KCBBHFBD   0x00001       buffer dirty  
            KCBBHFAM   0x00002  7.3  about to modify; try not to start io  
            KCBBHFAM   0x00002  8.0  about to modify; try not to start io  
            KCBBHNAC   0x00002  8.1  notify dbwr after change  
            KCBBHFMS   0x00004       modification started, no new writes  
            KCBBHFBL   0x00008       block logged  
            KCBBHFTD   0x00010       temporary data - no redo for changes  
            KCBBHFBW   0x00020       being written; can't modify  
            KCBBHFWW   0x00040       waiting for write to finish  
            KCBBHFCK   0x00080  7.3  checkpoint asap  
                       0x00080  8.0  not used  
            KCBBHFMW   0x00080  8.1  multiple waiters when gc lock acquired  
            KCBBHFRR   0x00100       recovery reading, do not reuse, being read  
            KCBBHFUL   0x00200       unlink from lock element - make non-current  
            KCBBHFDG   0x00400       write block & stop using for lock down grade  
            KCBBHFCW   0x00800       write block for cross instance call  
            KCBBHFCR   0x01000       reading from disk into KCBBHCR buffer  
            KCBBHFGC   0x02000       has been gotten in current mode  
            KCBBHFST   0x04000       stale - unused CR buf made from current  
                       0x08000  7.3  Not used.  
            KCBBHFDP   0x08000  8.0  deferred ping  
            KCBBHFDP   0x08000  8.1  deferred ping  
            KCBBHFDA   0x10000       Direct Access to buffer contents  
            KCBBHFHD   0x20000       Hash chain Dump used in debug print routine  
            KCBBHFIR   0x40000       Ignore Redo for instance recovery  
            KCBBHFSQ   0x80000       sequential scan only flag  
            KCBBHFNW  0x100000  7.3  Set to indicate a buffer that is NEW  
                      0x100000  8.0  Not used  
            KCBBHFBP  0x100000  8.1  Indicates that buffer was prefetched  
            KCBBHFRW  0x200000  7.3  re-write if being written (sort)  
                      0x200000  8.0  Not used  
            KCBBHFFW  0x200000  8.1  Buffer has been written once  
            KCBBHFFB  0x400000       buffer is "logically" flushed  
            KCBBHFRS  0x800000       ReSilvered already - do not redirty  
            KCBBHFKW 0x1000000  7.3  ckpt writing flag to avoid rescan \*/  
                     0x1000000  8.0  Not used  
            KCBBHDRC 0x1000000  8.1  buffer is nocache  
                     0x2000000  7.3  Not used  
            KCBBHFRG 0x2000000  8.0  Redo Generated since block read  
            KCBBHFRG 0x2000000  8.1  Redo Generated since block read  
            KCBBHFWS 0x10000000 8.0  Skipped write for checkpoint.  
            KCBBHFDB 0x20000000 8.1  buffer is directly from a foreign DB  
            KCBBHFAW 0x40000000 8.0  Flush after writing  
            KCBBHFAW 0x40000000 8.1  Flush after writing  
  
 TS#         NUMBER 8.X Tablespace number  
 DBARFIL     NUMBER 8.X Relative file number of block  
 DBAFIL      NUMBER 7.3 File number of block  
DBABLK      NUMBER Block number of block  
 CLASS       NUMBER See Note 33434.1  
               1,'data block',  
               2,'sort block',  
               3,'save undo block',  
               4,'segment header',  
               5,'save undo header',  
               6,'free list',  
               7,'extent map',  
               8,'1st level bmb',  
               9,'2nd level bmb',  
               10,'3rd level bmb',  
               11,'bitmap block',  
               12,'bitmap index block',  
               13,'file header block',  
               14,'unused',  
               15,'system undo header',  
               16,'system undo block',  
               17,'undo header',  
               18,'undo block'                -- since 10g  
  
 STATE       NUMBER  
            KCBBHFREE         0       buffer free  
            KCBBHEXLCUR       1       buffer current (and if DFS locked X)  
            KCBBHSHRCUR       2       buffer current (and if DFS locked S)  
            KCBBHCR           3       buffer consistant read  
            KCBBHREADING      4       Being read  
            KCBBHMRECOVERY    5       media recovery (current & special)  
            KCBBHIRECOVERY    6       Instance recovery (somewhat special)  
  
 MODE\_HELD   NUMBER    Mode buffer held in (MODE pre 7.3)  
    0=KCBMNULL, KCBMSHARE, KCBMEXCL  
  
 CHANGES     NUMBER  
 CSTATE      NUMBER  
 X\_TO\_NULL   NUMBER Count of PINGS out (OPS)  
 DIRTY\_QUEUE NUMBER You wont normally see buffers on the LRUW  
 LE\_ADDR     RAW(4) Lock Element address (OPS)  
 SET\_DS      RAW(4) Buffer cache set this buffer is under  
     OBJ         NUMBER       Data object number  
 TCH     NUMBER 8.1 Touch Count  
 TIM     NUMBER 8.1 Touch Time  
 BA          RAW(4)  
 CR\_SCN\_BAS  NUMBER       Consistent Read SCN base  
 CR\_SCN\_WRP  NUMBER       Consistent Read SCN wrap  
 CR\_XID\_USN  NUMBER CR XID Undo segment no  
 CR\_XID\_SLT  NUMBER CR XID slot  
 CR\_XID\_SQN  NUMBER CR XID Sequence  
 CR\_UBA\_FIL  NUMBER CR UBA file  
 CR\_UBA\_BLK  NUMBER CR UBA Block  
 CR\_UBA\_SEQ  NUMBER CR UBA sequence  
 CR\_UBA\_REC  NUMBER CR UBA record  
 CR\_SFL      NUMBER  
 LRBA\_SEQ    NUMBER } Lowest RBA needed to recover block in cache  
 LRBA\_BNO    NUMBER }  
 LRBA\_BOF    NUMBER }  
  
 HRBA\_SEQ    NUMBER } Redo RBA to be flushed BEFORE this block  
 HRBA\_BNO    NUMBER } can be written out  
 HRBA\_BOF    NUMBER       }  
  
 RRBA\_SEQ    NUMBER } Block recovery RBA  
 RRBA\_BNO    NUMBER }  
 RRBA\_BOF    NUMBER }  
 NXT\_HASH    NUMBER Next buffer on this hash chain  
 PRV\_HASH    NUMBER Previous buffer on this hash chain  
 NXT\_LRU     NUMBER Next buffer on the LRU  
 PRV\_LRU     NUMBER Previous buffer on the LRU  
 US\_NXT      RAW(4)  
 US\_PRV      RAW(4)  
 WA\_NXT      RAW(4)  
 WA\_PRV      RAW(4)  
 ACC         RAW(4)  
 MOD         RAW(4)