```
──────────────── MODULE cacherequirements ────────────────
EXTENDS Naturals

CONSTANTS
    KEYS  The full set of keys in the database

VARIABLES
    database,   database[key] = DataVersion
    cache   cache[key] = CacheValue

 The maximum number of versions a key can have in this model
MaxVersions ≜ 4

 Data versions are scoped to an individual key
DataVersion ≜ Nat

 Represents the absence of a value in a cache
CacheMiss ≜ [type : {"miss"}]

 Represents the presence of a value in a cache, as well as the value
CacheHit ≜ [type : {"hit"}, version : DataVersion]


DatabaseAndCacheConsistent ≜
    ∀ k ∈ KEYS :
         If the key is in cache
        ∨ ∧ cache[k] ∈ CacheHit
             It should be the same version as the database
          ∧ cache[k].version = database[k]
         A cache miss is also okay. A cache won't hold everything
        ∨ cache[k] ∈ CacheMiss

 This means that at some point, the database and cache are consistent.
 It is important to note that this is not eventual consistency.
 This only says it needs to be eventually consistent once.
EventuallyDatabaseAndCacheConsistent ≜ ◇DatabaseAndCacheConsistent

 The cache must be always eventually consistent.
AlwaysEventuallyDatabaseAndCacheConsistent ≜
    □EventuallyDatabaseAndCacheConsistent

 Used as a state constraint to prevent unbounded testing
 with infinite versions.
DatabaseRecordsDoNotExceedMaxVersion ≜
    ∀ k ∈ KEYS :
        database[k] < MaxVersions

──────────────────────────────────────────────────────────
```

\ * Modification History
\ * Last modified Sun *Nov* 12 01:50:39 *CST* 2023 by *wengjialin*
\ * Created *Mon Nov* 06 00:57:56 *CST* 2023 by *wengjialin*