**GUAVA CACHE**

**UC:**

1. Basic usage
   1. Basic config
   2. Load single
   3. Listeners
   4. Eccezioni (null)
2. SearchKey diverse da StoringKey
3. LoadBatch
4. Eviction policy
   1. Maximum Size (LRU ) Last recently used criteria
   2. MaximuWeight (to detect cache max size)
   3. Time to idle (last access)
   4. Time to live (time from write)
5. WeakKeys
6. Stats
7. Auto refresh
8. Thundering Herd

Restituisce

CacheStats stats = cache.stats();

Aggiorna ad ogni chiamata I contatori.

Da LoadingCache docs:

1. Meccanismo Load:

carica da loader impl se non trova nella cache.  
Fino a che il valore nuovo non completa il load, il vecchio viene mostrato

/\*\*

\* Returns the value associated with {@code key} in this cache, first loading that value if necessary.   
No observable state associated with this cache is modified until loading completes.

loader

\* <p>Caches loaded by a {@link CacheLoader} will call {@link CacheLoader#load} to load new values into the cache. Newly loaded values are added to the cache using

{@code Cache.asMap().putIfAbsent} after loading has completed; if another value was associated with {@code key} while the new value was loading then a removal notification will be sent for the new value.

2) Come evita thundering herd:

\* <p>If another call to {@link #get} or {@link #getUnchecked} is currently loading the value for {@code key}, simply waits for that thread to finish and returns its loaded value. Note that multiple threads can concurrently load values for distinct keys.

\*

\* <p>If the cache loader associated with this cache is known not to throw checked

\* exceptions, then prefer {@link #getUnchecked} over this method.

\*

\* **@throws** ExecutionException if a checked exception was thrown while loading the value. ({@code

\* ExecutionException} is thrown <a

\* href="http://code.google.com/p/guava-libraries/wiki/CachesExplained#Interruption">even if

\* computation was interrupted by an {@code InterruptedException}</a>.)

\* **@throws** UncheckedExecutionException if an unchecked exception was thrown while loading the

\* value

\* **@throws** ExecutionError if an error was thrown while loading the value

\*/

V get(K key) **throws** ExecutionException;