链滴

GuavaCache 简介

作者：god520

原文链接：https://ld246.com/article/1509891643770

来源网站：链滴

<h2 id="使用介绍">使用介绍</h2>
多线程高并发场景中往往是离不开 cache 的，需要根据不同的应用场景来选择不同的 cache，如分布式缓存如 redis、memcached，还有本地（进程）缓存如 ehcache、GuavaCache。

简单
强大
轻量级
不需要配置文件
能覆盖绝大多数使用 cache 的场景需求！

<h2 id="适用性">适用性</h2>
计算或检索一个值的代价很高，并且对同样的输入需要不止一次获取值的时候，就应当考虑使用存.
Guava Cache 与 ConcurrentMap 很相似，但也不完全一样。最基本的区别是 ConcurrentMap 会一直保存所有添加的元素，直到显式地移除。相对地，Guava Cache 为了限制内存占用，通常都定为自动回收元素。在某些场景下，尽管 LoadingCache 不回收元素，它也是很有用的，因为它会自加载缓存。
通常来说，Guava Cache 适用于：

你愿意消耗一些内存空间来提升速度。
你预料到某些键会被查询一次以上。
缓存中存放的数据总量不会超出内存容量。（Guava Cache 是单个应用运行时的本地缓存。它把数据存放到文件或外部服务器。如果这不符合你的需求，请尝试 Memcached 这类工具）

如果你的场景符合上述的每一条，Guava Cache 就适合你。
<h2 id="示例">示例</h2>
<pre><code class="highlight-chroma"> LoadingCache graphs = CacheBuilder.newBuilder()
        .maximumSize(1000)
        .expireAfterWrite(10, TimeUnit.MINUTES)
        .removalListener(MY_LISTENER)
        .build(
            new CacheLoader() {
                public Graph load(Key key) throws AnyException {
                    return createExpensiveGraph(key);
                }
            });

</code></pre>
<h2 id="Maven依赖">Maven 依赖</h2>
<blockquote>
<pre><code class="highlight-chroma"></code></pre>
</blockquote>

```
com.google.guava
guava
19.0
```

原文链接：GuavaCache 简介