### Memcached

Memcached是免费、开源、高性能、分布式内存对象缓存系统，它通过缓存对象来减少对数据库的访问。

Memcached之间不互相通信。

使用内存存储。重启会导致数据消失，使用LRU(Least Recently Used)算法来移除缓存。

Memcached并没有分布式功能，因为各个memcached不互相通信，怎么分布式取决于客户端。

安装Memcached，使用 memcached.exe –d install 命令安装memcached服务，然后启动服务，并将服务状态设为自动开机启动。

Memcached的key不能用控制字符或者包含空白，最长250字符。唯一标识一个数据。

Memcached的超时时间不要设置超过30天。

### Memcached命令

get:获取数据

set:存储数据

add:如果服务器上没有该key，存储数据

replace:如果服务器上有该key,存储数据

append:在key所表示的数据后加上新数据

prepend:在key表示的数据前加上新数据

cas:在多个客户端情况下，存储值如果该客户端上次获取该值后，该值没被其它客户端改变。

delete:删除某个key对应数据

Increment/Decrement:key必须存在，增1或者减1

touch:更新一个现存key的超时时间

flush\_all:失效所有的key

### BeitMemcached

Hash和服务器选择：

使用BeitMemcached客户端，该客户端使用Ketama算法，该算法根据key将数据映射到同一个服务器。该映射与机器无关。如果配置中服务器动态增或删除了，该算法仍然能够较完好的完成映射。如果一个server移除了，该算法能够将key原来映射到它上的，映射到其它服务器。如果一个服务器添加了，新服务器能够平滑的接管原来映射到其他服务器的key。

Socket Pool:

每个客户端为每个配置的服务器维护了一个socket连接池。并且使用该方法支持多线程使用各种MemCached命令。

序列化和编码

为了减小存储对象的大小，Beit使用下列方式进行编码：

Bool,byte,short,ushort,int,uint,long,ulong,float,double序列化为native byte.

DateTime序列化为long类型的Ticks.

String使用utf8

Byte[]直接使用，不做修改

其他类型使用binaryFormatter序列化。BinaryFormatter必须有Serializable。而XmlSerializer不需要。

当然有一个类型标志，来标识何种类型。

压缩

通常数据大于CompressionThreshold=128kb时，beit会自动压缩保存，获取时解压。

Failover故障切换

Beit没有提供failover，如果一个服务器关了，beit会标记它，然后周期性重试。所有定向到该机器上的调用立即失败。除非从配置文件中移除或者重启该服务器，否则定向到该服务器上的key仍然定向到该服务器。

日志和异常

总体上，beit尽量不抛出异常。并提供了LogAdapter适配日志。

使用Beit memcached

方式1：将ClientLibrary中的类复制到你的项目

方式2：将其编译为类库项目引用到你的工程