# Chat未读消息拉取方案

**需求说明：**Chat使用Redis缓存每个用户会话的未读消息，具体的，每个用户会话的未消费的消息会保存在Redis上的一个LIST里，key为一个与会话ID有关的字符串。如何实现未读消息的拉取呢？

**实现思路：**

**使用乐观锁的实现**

一种实现是拉取的时候，lrange 0 -1取到所有值，然后delete这个key，由于lrange和delete之间没有原子性保证，所以要加一个锁，可以使用redis乐观锁。代码实现如下：

````

@Override

public List<DownlinkMessage> getUnreadMessages(String channel, String userId, String channelId) {

List<DownlinkMessage> unreadDms = cacheService.redisTemplate().

execute(new SessionCallback<List<DownlinkMessage>>() {

@Override

@SuppressWarnings("unchecked")

public <K, V> List<DownlinkMessage> execute(RedisOperations<K, V> ops) throws DataAccessException {

String tenantId = configurationService.getTenantSpaceIdById(channelId);

String unreadMsgsKey = InteractiveServiceUtils.generateUserSystemMessageKey(channelId, userId,

tenantId);

List<DownlinkMessage> dms = BaseCollectionUtils.newArrayList();

ops.watch((K) unreadMsgsKey);

List<String> msgStrs = cacheService.listRange(unreadMsgsKey, 0, NUMBER\_MINUS\_1);

msgStrs = Optional.ofNullable(msgStrs).orElse(Collections.emptyList());

for (String msgStr : msgStrs) {

DownlinkMessage dm = JSON.parseObject(msgStr, DownlinkMessage.class);

dms.add(dm);

}

ops.multi();

cacheService.deleteKey(unreadMsgsKey);

List<Object> resultUnreadMessages = ops.exec();

// 在试图清理未读消息的过程中未读消息列表被修改，则不清理，不返回未读消息。

if (CommonUtil.isEmpty(resultUnreadMessages)) {

dms = Collections.emptyList();

}

return dms;

}

});

return unreadDms;

}

````

这种实现有一个缺点是，如果消息LIST更新频繁，那么读取线程读取失败的几率会比较高（被乐观锁校验住），那么用户就会较长时间获取不到消息，体验会降低。

**无锁的实现**

有一种更好的方案，不需要乐观锁，方案如下：

想要拉取未读消息前，rename下这个key，变成${原始key}${UUID}，然后直接lrange 新key 0 -1，delete 新key，就好了。这种实现下由于消息PUSH还是往老key里push，与拉取消息时所读的key是不同的，所以不会存在冲突，不需要加锁。

**可提升点：**

1. 限制消息拉取的最大条数。