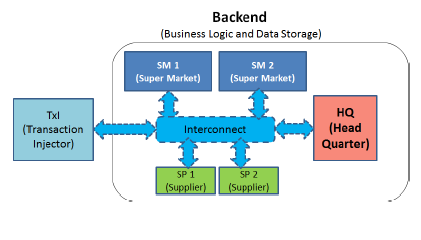
1. **SPECjbb2013组件：**

通过查阅SPECjbb2013的design document得知，SPECjbb2013测试集主要有三部分组成：Controller（控制器）、Transaction Injector（事务注入器）和Backend（后端）。

Controller指示测试集中其他组件的执行。在测试的开始，Controller等待其他组件发来的握手信息。一旦其他组件启动了，它们就会发送握手信息给Controller。Controller开始在controller.log和controller.out中记录运行的过程。Controller也在二进制log文件中记录具体的运行信息以在后面生成报告。与此同时，其它的组件也开始记录各自的log文件。

Transaction Injector（TxI）是请求的产生器和跟踪器。当收到握手信息后，Controller向每个TxI发送配置参数。Controller在所有TxI平均分配总的注入速率。TxI发送三种请求：probe requests, saturate requests, service requests。TxI为每种请求都分配了各自的线程池。TxI一旦接受到了要发布的请求，就把请求分配给映射到的Backend并记录它的响应时间。

Backend是运用到了商业逻辑的组件，它处理来自TxI的请求、存储数据和执行后台任务等。在每个Backend中存在着三个实体：Supermarkets, Suppliers, Headquarters。每个Backend存在两个Supermarkets和两个Suppliers以及一个Headquarters。Customer在SM中购买商品并把交易的记录发送给HQ。随着时间的流逝，SM中的商品会减少，这就会导致SM向SP中请求商品的补充。SP可以向SM中无穷无尽地补充商品。



通过测试在不同参数情况下，SPECjbb2013的运行结果，发现影响max-JOPS和critical-JOPS的主要参数是input.number\_customers和input.number\_products（在/config/ specjbb2013.props,默认值都是10000）。在一定范围了减少input.number\_customers，可以增加max-JOPS和critical-JOPS的值。在一定范围了增加input.number\_products，可以增加max-JOPS和critical-JOPS的值。但是由于机器的限制，导致input.number\_products高于一定值时SPECjbb2013无法运行。通过实验的实验结果，发现当customers和products都等于10000时有最大的max-JOPS和critical-JOPS。

为了控制请求注入的速度，可以修改controller.rtcurve.step（默认值为0.01），以此来达到模拟特殊场景的效果。

测试还发现运行一组Group大约占用机器2GB的内存。

以上结果是在四核i3-2120和8GB内存情况下SPECjbb2013运行得到的。

1. 测试脚本

程序测试脚本为安装目录下的test\_jops\_max.sh 脚本中各自安放了一个Controller（控制器）、Transaction Injector（事务注入器）和Backend（后端）。

更多测试方案：

