



# 钱塘（JTang）中间件平台软件 性能测试和稳定性测试报告

英特尔公司版权所有

二〇〇六年八月

# 目录

<b>1. 项目概要</b>	<b>11</b>
<b>2. 项目描述</b>	<b>12</b>
<b>3. 产品与系统测试配置概述</b>	<b>14</b>
3.1. 产品概述	14
3.2. 系统测试配置	16
<b>4. 性能测试</b>	<b>19</b>
4.1. JTang EJB 性能测试	19
4.1.1. Stateful Session Bean	19
4.1.2. Stateless Session Bean	21
4.2 三层架构 Web 应用	23
4.3 JTang Flow 性能测试	25
4.3.1 直接调用	25
4.3.2 通过 Web 方式	26
4.4 JTang Form 性能测试	27
4.5 JTang PMI 性能测试	29
4.6 JTang MQ 性能测试	30
4.6.1 P-P 模式	30
4.6.2 P-S 模式	36
<b>5 稳定性测试</b>	<b>43</b>
5.1 JTang EJB 稳定性测试	43
5.1.1 Socket 方式	43

5.1.2	RMI 方式 .....	44
5.2	JTang Flow 稳定性测试 .....	45
5.3	JTang MQ 稳定性测试 .....	46
6	性能分析 .....	48
6.1	JTang EJB 性能分析 .....	48
6.1.1	应用性能趋势曲线 .....	48
6.1.2	性能对比 .....	52
6.1.3	系统性能曲线 .....	57
6.2	三层架构 Web 应用 .....	59
6.3	JTang Flow 性能分析 .....	60
6.4	JTang Form 性能分析 .....	63
6.5	JTang PMI 性能分析 .....	64
6.6	JTang MQ 性能分析 .....	65
6.6.1	P-P 模式 .....	65
6.6.2	P-S 模式 .....	69
7	总结与建议 .....	75

## 目录：表格

表 3-1 测试环境硬件列表 .....	17
表 3-2 测试环境软件列表 .....	18
表 4-1 本地调用 Stateful Session Bean 性能数据 .....	19
表 4-2 远程调用 Stateful Session Bean 使用 Socket 协议性能数据 .....	20
表 4-3 远程调用 Stateful Session Bean 使用 RMI 协议性能数据 .....	20
表 4-4 远程调用 Stateful Session Bean 使用 HTTP 协议性能数据 .....	21
表 4-5 远程调用 Stateful Session Bean 使用 SSL 协议性能数据 .....	21
表 4-6 本地调用 Stateless Session Bean 性能数据 .....	22
表 4-7 远程调用 Stateless Session Bean 使用 Socket 协议性能数据 .....	22
表 4-8 远程调用 Stateless Session Bean 使用 RMI 协议性能数据 .....	22
表 4-9 远程调用 Stateless Session Bean 使用 HTTP 协议性能数据 .....	23
表 4-10 远程调用 Stateless Session Bean 使用 SSL 协议性能数据 .....	23
表 4-11 三层架构 Web 应用性能数据 .....	24
表 4-12 三层架构 Web 应用性能数据 .....	24
表 4-13 直接调用方式下 JTang Flow 性能数据 .....	25
表 4-14 直接调用方式下 JTang Flow 性能数据 .....	26
表 4-15 通过 Web 方式 JTang Flow 性能数据 .....	26
表 4-16 通过 Web 方式 JTang Flow 性能数据 .....	27
表 4-17 JTang Form 性能数据 .....	28

表 4-18 JTang Form 性能数据 .....	28
表 4-19 十万视图数据情况下 JTang Form 性能数据 .....	28
表 4-20 十万视图数据情况下 JTang Form 性能数据 .....	29
表 4-21 JTang PMI 性能数据 .....	29
表 4-22 JTang PMI 性能数据 .....	30
表 4-23 点对点模式下消息处理方式 .....	30
表 4-24 消息为 1K 大小 nP_nT 消息处理性能数据 .....	31
表 4-25 消息为 10K 大小 nP_nT 消息处理性能数据 .....	31
表 4-26 消息为 100K 大小 nP_nT 消息处理性能数据 .....	32
表 4-27 消息为 1K 大小 P_nT 消息处理性能数据 .....	32
表 4-28 消息为 1K 大小 P_nT(启动磁盘写缓存)消息处理性能数据 .....	32
表 4-29 消息为 10K 大小 P_nT(启动磁盘写缓存)消息处理性能数据 .....	33
表 4-30 消息为 100K 大小 P_nT(启动磁盘写缓存)消息处理性能数据 .....	33
表 4-31 消息为 1K 大小 nP_T 消息处理性能数据 .....	34
表 4-32 消息为 10K 大小 nP_T 消息处理性能数据 .....	34
表 4-33 消息为 100K 大小 nP_T 消息处理性能数据 .....	34
表 4-34 消息为 1K 大小 P_T 消息处理性能数据 .....	35
表 4-35 消息为 1K 大小 P_T 消息处理(启动磁盘写缓存)性能数据 .....	35
表 4-36 消息为 10K 大小 P_T 消息处理(启动磁盘写缓存)性能数据 .....	36
表 4-37 消息为 100K 大小 P_T 消息处理(启动磁盘写缓存)性能数据 .....	36
表 4-38 发布—订阅模式下消息处理方式 .....	36
表 4-39 消息为 1K 大小 nP_nD_nT 消息处理性能数据 .....	37

表 4-40 消息为 10K 大小 nP_nD_nT 消息处理性能数据.....	37
表 4-41 消息为 100K 大小 nP_nD_nT 消息处理性能数据.....	38
表 4-42 消息为 1K 大小 P_nD_nT 消息处理性能数据 .....	38
表 4-43 消息为 1K 大小 nP_D_nT 消息处理性能数据 .....	38
表 4-44 消息为 1K 大小 P_D_nT 消息处理性能数据 .....	39
表 4-45 消息为 10K 大小 P_D_nT 消息处理性能数据.....	39
表 4-46 消息为 100K 大小 P_D_nT 消息处理性能数据.....	39
表 4-47 消息为 1K 大小 nP_nD_T 消息处理性能数据 .....	40
表 4-48 消息为 10K 大小 nP_nD_T 消息处理性能数据.....	40
表 4-49 消息为 100K 大小 nP_nD_T 消息处理性能数据.....	40
表 4-50 消息为 1K 大小 P_nD_T 消息处理性能数据 .....	41
表 4-51 消息为 1K 大小 nP_nD_T 消息处理性能数据 .....	41
表 4-52 消息为 1K 大小 P_D_T 消息处理性能数据 .....	42
表 4-53 消息为 10K 大小 P_D_T 消息处理性能数据 .....	42
表 4-54 消息为 100K 大小 P_D_T 消息处理性能数据.....	42
表 5-1 Socket 方式 JTang EJB 稳定性测试数据.....	43
表 5-2 RMI 方式 JTang EJB 稳定性测试数据 .....	44
表 5-3 JTang Flow 稳定性测试数据 .....	45
表 5-4 消息体大小为 1K，JTang MQ 稳定性测试数据 .....	46

## 目录：图片

图 3-1 JTang 中间件平台整体架构 .....	14
图 3-2 测试环境系统拓扑结构 .....	16
图 6-1 Socket 方式下，Stateful Session Bean 压力性能趋势曲线 .....	48
图 6-2 RMI 方式下，Stateful Session Bean 压力性能趋势曲线 .....	49
图 6-3 HTTP 方式下，Stateful Session Bean 压力性能趋势曲线 .....	49
图 6-4 SSL 方式下，Stateful Session Bean 压力性能趋势曲线 .....	50
图 6-5 Socket 方式下，Stateless Session Bean 压力性能趋势曲线 .....	50
图 6-6 RMI 方式下，Stateless Session Bean 压力性能趋势曲线 .....	51
图 6-7 HTTP 方式下，Stateless Session Bean 压力性能趋势曲线 .....	51
图 6-8 SSL 方式下，Stateless Session Bean 压力性能趋势曲线 .....	52
图 6-9 Stateful Session Bean 在不同协议访问方式下的响应时间性能对比 .....	53
图 6-10 Stateful Session Bean 在不同协议访问方式下的 TPS 性能对比 .....	53
图 6-11 Stateless Session Bean 在不同协议访问方式下的响应时间性能对比 .....	54
图 6-12 Stateless Session Bean 在不同协议访问方式下的 TPS 性能对比 .....	54
图 6-13 Socket 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比 .....	55
图 6-14 Socket 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比 .....	55
图 6-15 HTTP 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比 .....	56
图 6-16 HTTP 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比 .....	56

图 6-17 Socket 方式下，Stateful Session Bean 系统性能表现曲线 .....	57
图 6-18 Socket 方式下，Stateless Session Bean 系统性能表现曲线 .....	58
图 6-19 RMI 方式下，Stateful Session Bean 系统性能表现曲线 .....	58
图 6-20 RMI 方式下，Stateless Session Bean 系统性能表现曲线 .....	59
图 6-21 三层架构的简单 Web 应用压力性能趋势曲线 .....	60
图 6-22 JTang Flow 在直接访问方式下压力性能趋势曲线 .....	61
图 6-23 JTang Flow 在 WEB 访问方式下压力性能趋势曲线 .....	61
图 6-24 JTang Flow 在直接访问方式下 CPU 利用率曲线 .....	62
图 6-25 JTang Flow 在 WEB 访问方式下 CPU 利用率曲线 .....	62
图 6-26 JTang Form 性能趋势曲线 .....	63
图 6-27 JTang PMI 性能趋势曲线 .....	64
图 6-28 JTang MQ 在点对点模式下不同消息处理方式性能对比 .....	65
图 6-29 点对点模式下 P_nT 处理方式，磁盘优化和非磁盘优化性能对比 .....	66
图 6-30 点对点模式下 P_T 处理方式，磁盘优化和非磁盘优化性能对比 .....	66
图 6-31 点对点模式下 nP_nT 处理方式下不同消息体大小性能对比 .....	67
图 6-32 点对点模式下 P_nT 处理方式下不同消息体大小性能对比 .....	67
图 6-33 点对点模式下 nP_T 处理方式下不同消息体大小性能对比 .....	68
图 6-34 点对点模式下 P_T 处理方式下不同消息体大小性能对比 .....	68
图 6-35 不同消息处理方式下的 CPU 利用率对比 .....	69
图 6-36 发布—订阅模式下不同消息处理方式下性能对比 .....	70
图 6-37 发布—订阅模式下不同消息处理方式下性能对比 .....	70
图 6-38 发布—订阅模式下 nP_nD_nT 处理方式下不同消息体大小性能对比 .....	71



图 6-39 发布-订阅模式下 P_D_nT 处理方式下不同消息体大小性能对比 .....	71
图 6-40 发布-订阅模式下 nP_nD_T 处理方式下不同消息体大小性能对比 .....	72
图 6-41 发布-订阅模式下 P_D_T 处理方式下不同消息体大小性能对比 .....	72
图 6-42 发布-订阅模式下不同消息处理方式下 CPU 利用率对比 .....	73
图 6-43 发布-订阅模式下不同消息处理方式下 CPU 利用率对比 .....	73
图 6-44 四路基于 Xeon 3.33GHz 的服务器(具有 Hyper-Threading)和两路基于双核 Xeon 2.66GHz 的服务器在 P-S 模式下，使用同样的消息处理方式下的性能对比 .....	74



## 1. 项目概要

受杭州国家软件产业基地有限公司（下称“国软公司”）的委托，英特尔解决方案中心（ISS）对具有自主知识产权的钱塘（JTang）中间件平台软件的性能和稳定性进行了全面测试。

钱塘(JTang)中间件平台软件是具有自主知识产权的基础中间件平台，其包括了一个符合 J2EE1.4 标准的应用服务器 JTang Server（提供 EJB、JMS、JMX、JACC、JNDI、Web Service 等服务）、面向上层应用系统快速开发的增值服务套件 JTang Service（提供 Workflow、Form、Report、SSO、PMI、File Server 等服务）、Eclipse、JBuilder 等开发工具的集成系统 JTang IDE、以及配置管理工具 JTang Console。

为了获取钱塘中间件平台软件在 Intel 服务器架构上的全面性能数据，以对钱塘中间件平台软件的性能和稳定性进行系统层的验证和分析，并获取今后软件升级和优化的第一手资料，英特尔解决方案中心的技术顾问同该平台软件的研发工程师们一起在英特尔解决方案中心，利用英特尔技术顾问在 Intel 的 IA（Intel Architecture）架构上的成熟经验和性能测试方法论,对钱塘中间件平台软件进行了系统层的性能测试和稳定性测试,并由此产生了《钱塘（JTang）中间件平台软件性能测试和稳定性测试报告》。

## 2. 项目描述

在性能测试部分，将对钱塘中间件平台软件的五个重要功能组件进行测试：

- JTang EJB 性能测试
- JTang MQ 性能测试
- JTang Flow 性能测试
- JTang Form 性能测试
- JTang PMI 性能测试

性能测试过程中所要采集的系统性能数据指标, 将主要包括：

- 并发用户数
- 服务器吞吐量 TPS 或 Messages/s
- 端到端响应时间
- 应用服务器和数据库服务器 CPU 利用率
- 应用服务器和数据库服务器内存利用率
- 应用服务器和数据库服务器磁盘 I/O 利用率
- 应用服务器和数据库服务器网络传输率
- JVM Heap 使用状况

主要的性能测试场景将包括：

1. 测试以不同并发用户同时进行创建 EJB 实例、执行实例方法、释放 EJB 实例的过程，JTang EJB 所表现的性能。测试场景分为对 Stateful Session Bean 测试和对 Stateless Session Bean 测试两大类，并分别测试本地调用、Socket、RMI、HTTP、SSL 五种调用 EJB 的方式。
2. 测试以不同并发用户同时进行工作流程实例和活动实例的创建过程，JTang Flow 所表现的性能。并分别进行直接调用和 Web 调用方式的测试。

3. 测试以不同并发用户同时进行 JTang Form 视图实例创建、视图数据生成、视图数据查询等一个完整过程的性能。
4. 测试以不同并发用户同时请求 JTang PMI 进行用户访问资源权限的检查和校验的过程的性能。
5. 由发送机以不同的发送线程持续发送消息给 JTang MQ 消息中间件，接收机以不同的接收线程持续从消息中间件接收自己的消息。分别测试点对点模式下的四种消息处理组合（nP\_nT, P\_nT, nP\_T, P\_T），和分别测试发布订阅模式下的八种消息处理组合（nP\_nD\_nT, P\_nD\_nT, nP\_D\_nT, P\_D\_nT, nP\_nD\_T, P\_nD\_T, nP\_D\_T, P\_D\_T）。

在稳定性测试部分，将对钱塘中间件平台软件的下面三个重要功能组件进行测试：

- JTang EJB 稳定性测试
- JTang MQ 稳定性测试
- JTang Flow 稳定性测试

在稳定性测试过程中，将采用性能测试的加压的脚本，以较大的工作负荷，持续 24 小时以上向钱塘中间件平台软件系统加压，观察钱塘中间件平台软件系统是否稳定运行，数据采集指标包括：

- 应用服务器内存运行的增量和 JVM Heap 使用状况，以判断是否存在内存泄露
- 服务器 CPU 利用率的走向，以判断是否有异常时间段
- 服务器磁盘 I/O 利用率的走向，以判断是否有异常时间段
- 服务器网络传输率的走向，以判断是否有异常时间段
- 服务器吞吐量 TPS 或 Messages/s

### 3. 产品与系统测试配置概述

#### 3.1. 产品概述

钱塘(JTang)中间件平台软件是在应用软件平台化的背景和趋势下，面向金融、证券行业、企业信息化、电子政务、移动通信等应用软件领域，按照 CMMI II 管理要求开发的系列化基础中间件平台，包括一个 J2EE 应用服务器 JTang Server，一套面向上层应用系统的增值服务 JTang Service，以及开发工具集成工具 JTang IDE、配置管理工具 JTang Console。JTang Middleware 旨在使用户缩短应用软件开发周期、增加软件客户化能力、提高软件的可靠性、降低软件开发人员的要求、节省软件开发费用。平台体系结构如下图：

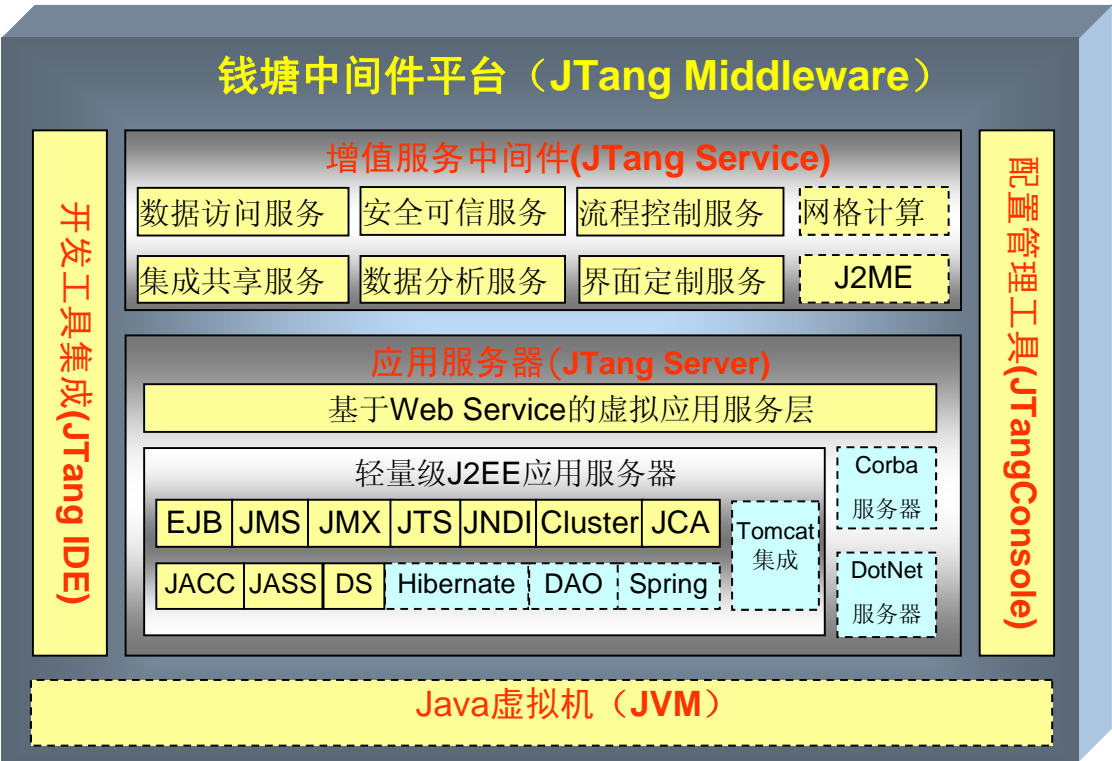


图 3-1 JTang 中间件平台整体架构

钱塘中间件平台具有以下的主要功能：

### 钱塘应用服务器 JTang Server

钱塘应用服务器（JTang Server）是一个符合 J2EE1.4 标准的轻量级应用服务器，包括 EJB 容器（会话 Bean 和消息驱动 Bean）、JMS 服务器、JMX、JNDI、Data Source、JACC、JAAS、Web Service、Tomcat 集成、Hibernate 集成、DAO 集成、Spring 集成等服务，其中 JTang EJB 容器是 JTang 应用服务器的企业级组件中间件，是实现应用程序业务逻辑的主要组件；钱塘消息服务器（JTang MQ）符合 JMS1.1 规范，提供 P2P 及 P/S 两种模式的可靠消息传递；Data Source 服务，符合 JDBC2.0 扩展标准，通过对物理数据源的封装，将实际物理数据源的配置与应用程序代码分离开，提高数据源操作的效率；系统提供 JACC 声明式权限管理及基于 JAAS 的认证和访问控制系统安全服务。

### 钱塘增值服务 JTang Service

钱塘增值服务中间件（JTang Service）面向应用软件开发，提供六类上层增值服务（安全可信、数据访问、数据分析、流程控制、集成共享、数据展示），目前的 2.0 版本，具体提供 workflow 服务 JTang Flow，报表服务 JTang Report，表单服务 JTang Form，权限管理框架 JTang PMI、分布式文件存储服务 JTang FileServer、单点登录服务 JTang SSO，其中钱塘 workflow 管理系统符合 WfMC（Workflow Management Coalition, workflow 管理联盟）标准，提供流程设计、流程执行和流程监控等功能；钱塘报表，提供基于 B/S 系统的报表服务，可按照用户要求定制各类报表；钱塘表单，是一个支持开放式标准的电子表单生成、部署与信息集成的工具；JTang PMI，可为各类应用系统提供用户认证、权限管理、访问控制等服务；钱塘单点登录服务，实现了一个支持多模式的 SSO（Single Sign On, 单点登录）系统，可实现对 Windows、Web 两类应用的自动登录；文件服务，为电子政务和电子商务海量文档数据提供分布式文件存储服务。

### 钱塘开发工具集成工具 JTang IDE

钱塘开发工具集成工具（JTang IDE）是一个将 JTang 应用服务器集成到 JBuilder 和 Eclipse 两个最流行的 Java 集成开发环境中的工具。通过这一工具，用户可以在 JBuilder 和 Eclipse 环境中启动、关闭 JTang 服务器，同时该工具提供 EJB，Web 描述符自动生成、动态部署、打包等辅助性的功能，大大提高基于 JTang 应用服务器的开发效率。

### 钱塘配置管理工具 JTang Console

钱塘应用服务器控制台（JTang Console），是一个基于 Web 浏览器的运行、配置、监视、维护钱塘应用服务器的管理平台，提供易用的企业应用部署及远程管理工具，用户可以以 Web 方式快速方便地管理 JTang 服务器。利用 JTang 控制台所提供的丰富功能，用户可以监视 JTang 服务器的运行情况、配置服务器的运行参数、在服务器中部署各种外部应用，同时提供多域配置管理服务。

## 3.2. 系统测试配置

为了进行 JTang 中间件平台软件的性能测试和稳定性测试，我们将搭建下面的网络和服务硬件环境：

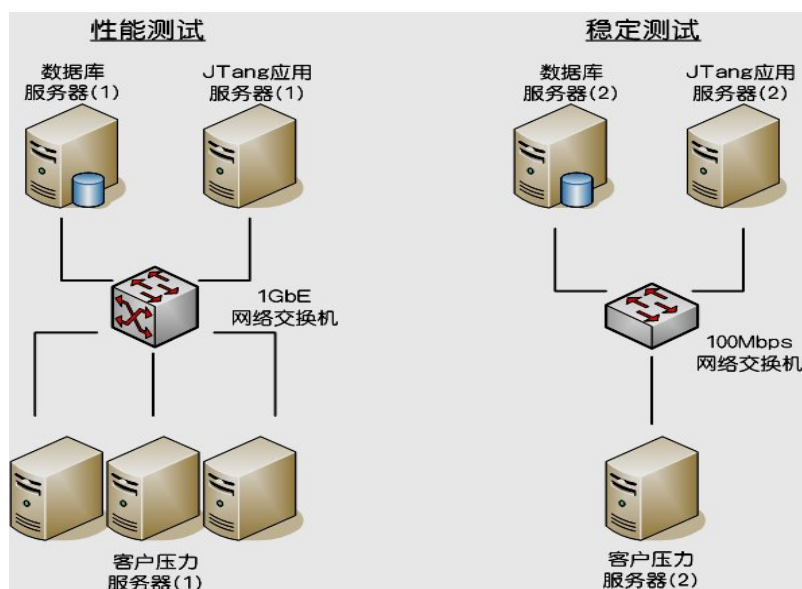


图 3-2 测试环境系统拓扑结构



为充分体现 JTang 中间件产品的三层架构的应用模式, 我们将 JTang 中间件应用服务器和数据库服务器分别置于不同的物理服务器之上, 以便清晰的鉴别在某性能测试场景下, 系统性能增长的瓶颈是会出现在应用服务器还是数据库服务器。我们知道, 基于双核英特尔 Xeon 处理器 5100 系列的服务器平台目前已迅速成为主流计算平台, 我们将使用一台基于双核 Intel Xeon 5160 处理器 (Woodcrest) 的双路服务器做性能测试环境中的 JTang 中间件应用服务器。

为了使性能测试和稳定性测试进行, 我们将使用下列硬件:

表 3-1 测试环境硬件列表

名称	配置	数量	用途
网络交换机	Intel NetStructure 100Mbps 交换机	1	组建百兆网络环境
网络交换机	3Com 4900 1Gbps 交换机	1	组建千兆网络环境
基于英特尔Xeon双核处理器的双路服务器	2 路 Xeon 双 核 处 理 器 Woodcrest, 主频2.66GHz, 3G 内存, 144GB内置SATA硬盘。	1	性能测试应用服务器 (1)
基于英特尔Xeon处理器的四路服务器	4 路 Xeon 处 理 器 , 主 频 3.33GHz, 4GB 内 存 , 72GB 内 置 SCSI 硬盘。	1	性能测试数据库服务器 (1)
基于英特尔Xeon处理器的双路服务器	2 路 Xeon 处 理 器 , 主 频 3.6GHz, 4GB 内 存 , 72GB 内 置 SCSI 硬盘。	1	稳定性测试应用服务器 (2)
基于英特尔Xeon处理器的四路服务器	4 路 Xeon 处 理 器 , 主 频 3.33GHz, 4GB内存, 72GB内置 SCSI 硬盘	1	稳定性测试数据库服务器 (2)

基于英特尔Xeon处理器的四路服务器	4 路 Xeon 处 理 器，主 频 2.8GHz，4GB内存,72GB内置 SCSI 硬盘。	1	稳定性测试压力机（2）
基于英特尔Xeon处理器的双路服务器	2 路 Xeon 处 理 器，主 频 3.2GHz，2GB内存,36GB内置 SCSI 硬盘	3	性能测试压力机（1）

JTang 中间件平台软件的性能测试和稳定性测试环境将由下列软件组成：

表 3-2 测试环境软件列表

名称
JTang 中间件平台软件 V2.0
Sun Microsystems JDK 1.5.0_07
Microsoft Windows Server 2003 Enterprise Edition (Service Pack 1)
Oracle 9i DB Enterprise Edition (Release 2)
Microsoft SQL Server 2000(Service Pack 4)
JTang 性能测试应用案例
JMeter 2.03

## 4. 性能测试

### 4.1. JTang EJB 性能测试

<压力过程描述>: 使用 JTang EJB 性能测试案例应用分别以不同的并发用户量，持续向 JTang 中间件平台服务器施加压力，每个场景将持续加压大约 10 分钟，系统性能数据采集从启动 JTang Sever 应用 30 秒后进行，以 10 秒为间隔采用系统性能数据，采样大约 9 分钟。

#### 4.1.1. Stateful Session Bean

<压力内容描述>: 在压力机上并发起多个虚拟用户，每个虚拟用户对应一个 Java 客户端线程。每个线程进行一个 Session Bean 的创建请求，并调用两个简单的 Method，然后进行释放 Session Bean 请求。在 Stateful Session Bean 的情况下，JTang 中间件服务器上的 Container 管理器，每接收到创建一个 Session Bean 的请求，都将在 Container 中新建一个 Session Bean 的实例，并负责管理 Session Bean 实例的整个生命周期的状态。

##### 4.1.1.1. 本地调用 Stateful Session Bean

通过 JSP 启动压力应用，并调用同一 Sun Microsystems Java JVM 下的应用服务器上的 Stateful Session Bean。

表 4-1 本地调用 Stateful Session Bean 性能数据

Users	TPS	RSP (ms)	HEAP (MB Used/MB Allocated)	APP- CPU% (Max/Min /Avg)	APP- Avail Mem(GB)	APP- DISK %	APP-NIC (MBytes/s)
200	11401.6	17	23.3/256	95/24/81	2.60	0.2	0
500	8472.2	58	31.9/256	95/85/90	2.58	0.4	0

- 系统数据 CPU%, Mem(GB), NIC(MBytes/s)采用 Windows 操作系统的系统监测工具 Perfmon 进行数据采集。

- TPS-每秒交易量 和 RSP(Response Time)-交易平均响应时间，由 JTang EJB 性能测试案例应用计算得出。一个交易的定义为，从创建一个 Session Bean，到使用 Session Bean 的方法，直至 Session Bean 完成任务被释放的一个完整过程。
- Heap 的状态，将取自 JVM 运行过程中所记录的 GC（Garbage Collection，垃圾回收）运行日志，并记录 Heap 最终的大小。

#### 4.1.1.2. 远程调用使用 Socket 协议

表 4-2 远程调用 Stateful Session Bean 使用 Socket 协议性能数据

Users	TPS	RSP (ms)	HEAP (MB Used /MB Allocated)	APP-CPU%	APP- Avail Mem (GB)	APP-DISK%	APP-NIC (MBytes/s)
100	1990.7	49	244.5/261	36	2.38	0.03	3.193
200	3573.1	55	247.3/264.3	66	2.38	0.03	5.671
500	3389.3	145.7	159.6/408.5	83	2.26	0.04	5.313
1000	3282.2	302	432.7/463.3	83	2.16	0.04	5.223

#### 4.1.1.3. 远程调用使用 RMI 协议

表 4-3 远程调用 Stateful Session Bean 使用 RMI 协议性能数据

Users	TPS	RSP (ms)	HEAP (MB Used /MB Allocated)	APP-CPU%	APP-Avail Mem (GB)	APP-DISK%	APP-NIC (MBytes/s)
100	177.7	562	32.8/256	23	2.58	0.03	10.6
200	342.5	585	78.8/256	47	2.47	0.03	20.4
500	341.9	1338.7	183.8/256	76	2.37	0.03	22.3
1000	313.4	3174.3	197.7/256	74	2.35	0.03	18.2

#### 4.1.1.4. 远程调用使用 HTTP 协议

表 4-4 远程调用 Stateful Session Bean 使用 HTTP 协议性能数据

Users	TPS	RSP (ms)	HEAP (MB Used /MB Allocated)	APP-CPU%	APP-Avail Mem (GB)	APP-DISK%	APP-NIC (MBytes/s)
100	200.7	497	17.1/256	34	2.59	0.03	1.50
200	201.8	989.5	21.6/256	35	2.59	0.03	1.49
500	191.3	2610	60.9/256	35	2.44	0.03	1.39

#### 4.1.1.5. 远程调用使用 SSL 协议

表 4-5 远程调用 Stateful Session Bean 使用 SSL 协议性能数据

Users	TPS	RSP (ms)	HEAP (MB Used/MB Allocated)	APP-CPU%	APP-Avail Mem (GB)	APP-DISK%	APP-NIC (MBytes/s)
100	1520.1	65	132.7/260.5	43.	2.38	0.04	2.98
200	2762.0	72	97.4/260.9	74	2.38	0.04	5.38
500	2386.7	291.7	381.0/445.5	66	2.35	0.04	4.51

#### 4.1.2. Stateless Session Bean

<压力内容描述>：在压力机上并发起多个虚拟用户，每个虚拟用户对应一个 Java 客户端线程。每个线程进行一个 Stateless Session Bean 的创建请求，并调用两个简单的 Method，然后进行释放 Session Bean 请求。在 Stateless Session Bean 的情况下，JTang 中间件服务器上的 Container 管理器，每接收到创建一个 Stateless Session Bean 的请求，将先在 Container 中的实例池寻找是否有可获取的已经创建好的 Stateless Session Bean 实例，若有，则直接使用它。JTang EJB 中间件服务器 Container 中的 Stateless Session Bean 实例池大小缺省数值为 100。

#### 4.1.2.1 本地调用 Stateless Session Bean

表 4-6 本地调用 Stateless Session Bean 性能数据

Users	TPS	RSP (ms)	HEAP (MB Used /MB Allocated)	APP-CPU% (Max/Min/Avg)	APP-Avail Mem (GB)	APP-DISK %	APP-NIC (MBytes/s)
200	17817.8	11	23.6/256	92/77/85	2.60	0.2	0
500	14122.0	34	34.7/256	91/76/83	2.58	0.2	0

#### 4.1.2.2 远程调用使用 Socket 协议

表 4-7 远程调用 Stateless Session Bean 使用 Socket 协议性能数据

Users	TPS	RSP(ms)	HEAP(MB Used /MB Allocated)	APP-CPU%	APP-Avail Mem (GB)	APP-DISK %	APP-NIC (MBytes/s)
100	2166.9	45	161.4/260.5	30	2.41	3.9	2.95
200	4206.2	47	246.8/263.8	49	2.40	0.05	5.71
500	5005.3	98.7	250.4/268.4	82	2.39	0.08	6.80
1000	4410.8	227.3	246.9/264.8	83	2.37	0.07	6.46

#### 4.1.2.3 远程调用使用 RMI 协议

表 4-8 远程调用 Stateless Session Bean 使用 RMI 协议性能数据

Users	TPS	RSP (ms)	HEAP (MB Used /MB Allocated)	APP-CPU%	APP-Avail Mem (GB)	APP-DISK%	APP-NIC (MBytes/s)
100	167.2	597	22.4/256	22	2.60	0.04	9.97
200	374.3	533	84.0/256	47	2.52	0.5	21.4
500	306.2	1625.3	229.1/256	81	2.38	0.06	22.7

#### 4.1.2.4 使用 HTTP 协议

表 4-9 远程调用 Stateless Session Bean 使用 HTTP 协议性能数据

Users	TPS	RSP (ms)	HEAP (MB Used/MB Allocated)	APP-CPU%	APP-Avail Mem (GB)	APP-DISK%	APP-NIC (MBytes/s)
100	201.1	496	17.5/256	34.4	2608	0.11	1.48
200	203.3	982.5	19.4/256	34.5	2613	0.07	1.46
500	196.6	2536.6	22/256	35.2	2409	0.4	1.39

#### 4.1.2.5 使用 SSL 协议

表 4-10 远程调用 Stateless Session Bean 使用 SSL 协议性能数据

Users	TPS	RSP (ms)	HEAP (MB Used/MB Allocated)	APP-CPU%	APP-Avail Mem (GB)	APP-DISK%	APP-NIC (MBytes/s)
100	1597.6	62	213.0/260.5	37	2.40	0.05	2.76
200	3206.0	62	244.8/261.8	63	2.39	0.05	5.54
500	2570.2	1494	266.0/295.9	53	2.39	0.06	4.49

## 4.2 三层架构 Web 应用

<压力内容描述>：将模仿一个简单的三层架构 Web 应用架构，前端客户端以 Web 浏览器的方式，通过 JSP 调用 JTang 中间件服务器上的 Session Bean，通过 Session Bean 访问操作后台数据库，维护项目和雇员的关系。数据库分为 3 张表：project 表记录项目的基本信息，employee 表记录雇员的基本信息，project\_employee 表维护项目和雇员的关系，可以是多对多的关系。客户端请求将随机调用四个对数据库的操作方法：

- 1) 雇员创建
- 2) 雇员删除(要连带删除关系表记录)
- 3) 项目创建

4) 项目删除(要连带删除关系表记录)

表 4-11 三层架构 Web 应用性能数据

Users	APP-CPU% (MAX/MIN/AVG)	APP-Avail Mem (GB)	APP-DISK %	APP-NIC (MBytes/s)	DB-CPU% (MAX/MIN/AVG)	DB-Avail Mem (GB)	DB-DISK %	DB-NIC (MBytes/s)
50	25/15/19	2.03	0.03	2.76	22/8/16	2.88	12	2.34
100	25/14/19	2.03	0.03	2.68	21/11/15	2.87	83	2.29
200	24/13/19	2.02	0.06	2.72	22/4/15	2.87	74	2.23
500	25/14/21	2.02	0.02	2.68	22/4/16	2.87	40	2.28

表 4-12 三层架构 Web 应用性能数据

Users	TPS	Response Time (ms)	HEAP (MB Used/MB Allocated )
50	437	112	172/309
100	419	231	178/309
200	417	458	202/381
500	428	1089	218/381



## 4.3 JTang Flow 性能测试

### 4.3.1 直接调用

<压力内容描述>：将测试创建流程实例和活动实例的过程，及其 workflow 执行引擎与数据库交互定义信息和状态信息的性能。在压力机上并发多个虚拟用户，每个虚拟用户对应一个 Java 客户端线程。每个线程完成一个下面的工作流程：

- 1) 获取流程定义对象 (Query DB)
- 2) 创建流程实例对象 (Insert DB)
- 3) 启动流程实例 (Update DB)
- 4) 创建流程实例的第一步活动实例 (Insert DB)
- 5) 创建活动实例相应的任务并分发给相应的任务执行者 (Insert and Update DB)

直接调用的方式，将采用由前端压力应用程序直接通过 RMI 的方式调用 JTang Flow 中间件应用服务器上的 workflow EJB，并使每个虚拟用户线程轮循运行 10 次。我们定义执行完一个完整的测试工作流程为一个完整的交易。JTang 中间件应用服务器 JVM 的 HEAP 启动参数为：-Xms256M -Xmx512M。

表 4-13 直接调用方式下 JTang Flow 性能数据

Users	APP-CPU% (Max/Min/Avg)	APP-Avail Mem (GB)	APP-DISK %	APP-NIC (MBytes/s)	DB-CPU %	DB-Avail Mem (GB)	DB-DISK %	DB-NIC (MBytes/s)
100	24/ 0 /8	2.3	0.07	4.0	16/0/9	3.0	113	0.47
200	47/0/18	2.15	0.08	7.46	26/0/13	3.0	116	0.50
500	55/0/22	2.07	0.06	7.78	56/0/25	3.0	110	0.52
1000	72/3/17	2.04	0.09	7.39	82/0/44	3.0	94.4	0.45

表 4-14 直接调用方式下 JTang Flow 性能数据

Users	TPS	Response Time (s)	Heap (MB Used/MB Allocated)
100	41	1.5	98/115
200	46	2.4	71//87
500	48	5.2	179/196
1000	41	15.3	188/204

### 4.3.2 通过 Web 方式

<压力内容描述>：每个虚拟用户完成的工作流程与“直接调用”的方式相同，只是前端压力应用通过 Web JSP 的方式调用 JTang Flow 中间件应用服务器上的工作流 EJB，该过程将引入 Tomcat 的参与。

JTang 中间件应用服务器 JVM 的 HEAP 启动参数为：-Xms256M -Xmx512M

表 4-15 通过 Web 方式 JTang Flow 性能数据

Users	APP-CPU % (Max/Min/Avg)	APP-Avail Mem (GB)	APP-DISK %	APP-NIC (MBytes/s)	DB-CPU %	DB-Avail Mem (GB)	DB-DISK %	DB-NIC (MBytes/s)
100	23/0/7	2.35	0.11	0.31	18/0/8	3.0	99	0.43
200	49/0/12	2.29	0.07	0.64	30/0/13	3.0	129.7	0.62
500	70/0 /15	2.20	0.03	0.80	63/0/30	3.0	138	0.75

表 4-16 通过 Web 方式 JTang Flow 性能数据

Users	TPS	Response Time (s)	Heap (MB Used)
100	42.8	1.95	48.2
200	58.9	3.22	86.8
500	66	7.18	77.9

## 4.4 JTang Form 性能测试

<压力内容描述>：将测试创建表单视图的执行性能，表单模板将使用已预先创建好的测试模板“浙江省科技计划项目合同申报”，该模板包括了“项目基本情况”、“预计经济效益”、“预计其他成果”等类型的标签，共 27 个标签。在进行性能测试时，在压力机上并发起多个虚拟用户，每个虚拟用户对应一个 Java 客户端线程。每个线程完成一个下面的执行过程：

- 1) 通过表单模板和表单视图名称，创建表单视图实例
- 2) 从数据库提取相关数据和计算视图数据
- 3) 更新或插入视图数据到数据库
- 4) 创建察看表单视图实例，读取视图数据

我们将使每个虚拟用户线程轮循运行数次。JTang 中间件应用服务器 JVM 的 HEAP 启动参数为：-Xms256M -Xmx512M。我们定义执行完上述一个完整的表单视图的操作流程为一个完整的交易。在无视图历史记录数据的情况下，进行的性能测试的数据记录如下：

表 4-17 JTang Form 性能数据

Users	APP-CPU% (Max /Min/Avg)	APP-Avail Mem (GB)	APP-DISK %	APP-NIC (MBytes/s)	DB-CPU%	DB-Avail Mem (GB)	DB-DISK %	DB-NIC (MBytes/s)
10	57/0/40	2.24	431	1.19	39/0/25	2.96	33	1.22
100	85/0/45	2.18	862	0.98	75/0/26	2.96	23.9	1.01
200	95/0/35	2.14	944	0.74	95/0/35	2.95	22.2	1.10

表 4-18 JTang Form 性能数据

Users	TPS	Response (s)	Heap (MB Used)
10	9.9	0.98	39.5
100	8.6	5.75	64.2
200	8.0	16.36	121.7

在有 10 万条视图数据历史记录的情况下,所进行的性能测试的数据记录如下:

表 4-19 十万视图数据情况下 JTang Form 性能数据

Users	APP-CPU% (Max/Min/Avg)	APP-Avail Mem (GB)	APP-DISK %	APP-NIC ( Mbytes/s) (Max/Min/Avg)	DB-CPU %	DB-Avail Mem (GB)	DB-DISK%	DB-NIC (MBytes/s) (Max/Min/Avg)
10	62/4/47	2.22	448	2.37/0/1.27	38/0/22	2.97	35.08	2.36/0/1.32
100	89/12/47	2.23	903	4.36/0/0.95	12/3/1	2.97	32.95	7.31/0/0.96
200	89/6/41	2.40	1093.1	11.77/0/0.72	16/0/4	2.87	20.36	18.07/0/0.82
500	100/25/49	2.32	1099.6	42.26/0/0.66	54/0/5	2.97	11.56	37.17/0/0.64

表 4-20 十万视图数据情况下 JTang Form 性能数据

Users	TPS	Response Time (s)
10	10.9	0.58
100	8.8	5.57
200	6.4	11.42
500	5.9	29.25

## 4.5 JTang PMI 性能测试

JTang PMI 的性能测试将使用数据库 Oracle 9i R2，初始数据为 100 万认证授权数据，JTang 中间件应用服务器 JVM 的 HEAP 启动参数为：-Xms256M -Xmx512M

<压力内容描述>：并发多个客户端线程，每个客户端线程模仿一个虚拟用户，每个虚拟用户将通过 JTang PMI 进行用户访问资源权限的检查校验：

- 1) 用户角色信息和资源相关的权限信息提取
- 2) 用户角色信息和资源相关的权限信息的关联比较。

我们将测试使用数据库服务器存储过程的方式对用户权限进行检查校验。

表 4-21 JTang PMI 性能数据

Users	APP-CPU% (Max/Min/Avg)	APP-Avail Mem (GB)	APP-DISK %	APP-NIC (MBytes/s)	DB-CPU% (Max/Min/Avg)	DB-Avail Mem (GB)	DB-DISK %	DB-NIC (MBytes/s)
20	6.1/0.2/1.2	2.6	0.065	0.03	100/13/46	2.2	7.2	0.11
50	0.8/0.1/0.3	2.5	0.02	0.03	100/13/51	2.2	7.4	0.12
100	0.7/0.2/0.5	2.5	0.019	0.03	100/13/89	2.3	4.7	0.11
200	0.7/0.1/0.3	2.5	0.017	0.03	100/13/96	2.2	0.5	0.11
500	1.5/0.1/0.5	2.5	0.46	0.03	100/13/95	2.2	2.4	0.11

表 4-22 JTang PMI 性能数据

Users	TPS	Response Time (s)	Heap (MB Used/MB Allocated)
20	64.9	0.3	15.9/256
50	63.9	0.8	16.4/256
100	64.8	1.5	17.8/256
200	65.2	2.9	17.1/256
500	64.9	7.1	17.2/256

## 4.6 JTang MQ 性能测试

<压力过程描述> 使用两台压力客户机，一台作为消息发送机，一台作为消息接收机。由发送机以不同的发送线程持续发送消息到 JTang MQ 消息中间件的相应的消息队列，接收机以相应数目的接收线程持续从消息中间件相应的消息队列中接收自己的消息。

### 4.6.1 P-P 模式

在点对点的方式下，测试四种消息处理组合机制：

表 4-23 点对点模式下消息处理方式

消息处理组合机制	描述
nP_nT	非持久化-非事物化
P_nT	持久化-非事物化
nP_T	非持久化-事物化
P_T	持久化-事物化

事务定义为以 5 个消息为一个完整事务。持久化操作的数据库采用 Derby，位于 JTang MQ 消息中间件的同一台物理服务器上。JTang MQ 消息中间件 Heap 启动初始参数为：-Xms1260M -Xmx1260M

#### 4.6.1.1 nP\_nT 消息处理

下面记录消息体为 1K 大小的性能数据:

表 4-24 消息为 1K 大小 nP\_nT 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	19/ 14/16	2.6	0.1	4.9	1848	1837	21.9
10/10/10/1	49/40/44	2.6	0.05	11.2	4399	4385	25.5
100/100/100/20	74/69/72	2.6	0.05	14.2	5548	5504	60.7
500/500/50/20	60/50/56	2.6	0.04	12.9	5060	4980	114.7

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

CPU%、Avail Mem、DISK%、NIC (MBytes/s) 为JTang消息中间件服务器系统性能值

S-Msgs/s: 为每秒所有发送线程所发送的消息数量

R-Msgs/s: 为每秒所有接收线程所接收的消息数量

下面记录消息体为 10K 大小的性能数据:

表 4-25 消息为 10K 大小 nP\_nT 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	21/17/19	2.6	0.04	13.1	620.81	614.38	50.8
500/500/50/20	80/74/78	2.59	0.04	56.1	2651	2639	23.7

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

下面记录消息体为 100K 大小的性能数据:

表 4-26 消息为 100K 大小 nP\_nT 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	14/10/12	2.6	0.03	10.5	51.4	50.3	58.1
500/500/50/20	80/75/79	2.47	0.04	70.2	341.6	336.2	1124.5

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

#### 4.6.1.2 P\_nT 消息处理

下面记录消息体为 1K 大小的性能数据:

表 4-27 消息为 1K 大小 P\_nT 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	16/14/15	2.6	24.7	1.7	645.92	643.42	14.1
10/10/10/1	22/15/17	1.9	30.8	2.0	785.34	407.35	143.1
100/100/100/20	17/10/14	2.4	103.8	1.2	501.7	479.89	108.5
500/500/50/20	23/19/21	2.6	33	2.1	819.83	815.85	31.2

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

我们使用磁盘性能优化的高级特性(启动磁盘写缓存), 再次对上面的场景进行测试, 消息体大小仍然为 1K。

表 4-28 消息为 1K 大小 P\_nT(启动磁盘写缓存)消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	23/18/20	2.6	32.6	2.1	841.6	841.3	16.9
10/10/10/1	32/20/27	1.8	48.8	2.8	1167.0	827.2	486.3
100/100/100/20	30/22/26	2.4	25.6	2.3	890.6	861.6	164.8



500/500/50/20	29/18/27	2.2	53.3	2.6	1019.8	892.3	261.1
---------------	----------	-----	------	-----	--------	-------	-------

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

下面记录消息体为 10K 大小并启用磁盘性能优化的高级特性的性能测试数据。

表 4-29 消息为 10K 大小 P\_nT(启动磁盘写缓存)消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	24/20/22	2.56	36.0	9.3	434.2	434.1	28.7
500/500/50/20	40/34/37	2.36	53.4	13.2	618.5	597.3	254.8

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

下面记录消息体为 100K 大小并启用磁盘性能优化的高级特性的性能测试数据。

表 4-30 消息为 100K 大小 P\_nT(启动磁盘写缓存)消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	15/11/13	2.56	40.9	7.9	38.1	38.1	26.5
500/500/50/20	41/33/37	2.4	64.1	20.7	99.8	99.7	75.9

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

### 4.6.1.3 nP\_T 消息处理

下面记录消息体为 1K 大小的性能数据:

表 4-31 消息为 1K 大小 nP\_T 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	26/17/23	2.6	0.7	8.5	3375.81	3361.63	26.9
10/10/10/1	41/9/29	1.9	0.04	7.7	3131.48	3110.66	638.5
100/100/100/20	68/0/40	1.9	0.03	12.2	5441.29	5151.76	573.4
500/500/50/20	67/22/50	1.9	0.04	15.2	6400.63	6222.98	462.3

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

下面记录消息体为 10K 大小的性能数据:

表 4-32 消息为 10K 大小 nP\_T 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	16/10/13	2.57	0.06	9.8	460.0	459.5	12.5
500/500/50/20	75/67/71	2.27	0.1	55.2	2609.8	2591.2	136.2

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

下面记录消息体为 100K 大小的性能数据:

表 4-33 消息为 100K 大小 nP\_T 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	14/9/11	2.57	0.04	9.8	47.4	47.4	16.6
500/500/50/20	75/26/68	1.4	0.3	55.6	268.8	273.1	247.7

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

#### 4.6.1.4 P\_T 消息处理

下面记录消息体为 1K 大小的性能数据:

表 4-34 消息为 1K 大小 P\_T 消息处理性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	17/13/15.5	2.6	28.5	1.6	643.45	639.67	41.9
10/10/10/1	18/12/14	1.9	25.5	1.6	679.78	250.67	579.5
100/100/100/20	17/8/10	2.3	61.2	1.0	394.88	181.72	380.2
500/500/50/20	19/8/11	2.35	56.2	1.0	400.58	207.21	410.1

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

我们使用磁盘性能优化的高级特性(启动磁盘写缓存), 再次对上面的场景进行测试, 消息体大小仍然为 1K。

表 4-35 消息为 1K 大小 P\_T 消息处理(启动磁盘写缓存)性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	23/20/22	2.6	38.1	2.2	888	887.7	27.8
10/10/10/1	27/16/22	1.76	40.1	2.3	941.3	543.3	568.4
100/100/100/20	23/15/17	2.15	93	1.6	665.2	315.8	380.2
500/500/50/20	20/15/17	2.1	83.3	1.5	614.0	396.3	403.6

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

下面记录消息体为 10K 大小并启用磁盘性能优化的高级特性的性能测试数据。

表 4-36 消息为 10K 大小 P\_T 消息处理(启动磁盘写缓存)性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	14/10/11	2.6	43.3	4.5	213.2	213.1	24.8
500/500/50/20	30/24/27	2.3	60.8	9.3	439.3	410.9	432.9

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

下面记录消息体为 100K 大小并启用磁盘性能优化的高级特性的性能数据。

表 4-37 消息为 100K 大小 P\_T 消息处理(启动磁盘写缓存)性能数据

S/R/C/Q	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	12/10/11	2.6	17.6	7.0	33.4	33.4	52.3
500/500/50/20	31/24/28	2.0	73.2	14.5	69.8	69.4	439.5

S/R/C/Q: 表示发送者数/接收者数/联接数/消息队列数

## 4.6.2 P-S 模式

在发布-订阅的方式下，测试八种消息处理组合机制：

表 4-38 发布-订阅模式下消息处理方式

消息处理组合机制	描述
nP_nD_nT	消息非持久化-用户联接非可持续化-非事物化
P_nD_nT	消息持久化-用户联接非可持续化-非事物化
nP_D_nT	消息非持久化-用户联接可持续化-非事物化
P_D_nT	消息持久化-用户联接可持续化-非事物化
nP_nD_T	消息非持久化--用户联接非可持续化-事物化
P_nD_T	消息持久化--用户联接非可持续化-事物化
nP_D_T	消息非持久化-用户联接可持续化-事物化
P_D_T	消息持久化-用户联接可持续化-事物化

事务定义为以 5 个消息为一个完整事务。持久化操作的数据库采用 Derby，位于 JTang MQ 消息中间件的同一台物理服务器上。JTang MQ 消息中间件 Heap 启动初始参数为：-Xms1260M -Xmx1260M

#### 4.6.2.1 nP\_nD\_nT 消息处理

下面记录消息体为 1K 大小的性能测试数据:

表 4-39 消息为 1K 大小 nP\_nD\_nT 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	18/14/16	2.6	0.03	4.7	1834.6	1825.3	8.8
10/10/10/10	69/64/67	1.9	0.06	18.0	7082.5	7078.9	9.8
50/50/50/50	65/58/62	2.56	0.1	14.4	5697.8	5663.8	9.1

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

下面记录消息体为 10K 大小的性能测试数据:

表 4-40 消息为 10K 大小 nP\_nD\_nT 消息处理性能数据

P/S /C/T	CPU% (Max /Min/Avg)	Avail Mem (GB)	DISK%	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	20/16/18	2.6	0.06	12.8	604.2	603.8	8.5
50/50/50/50	94/0/56	1.9	0.06	38.0	1877.1	1877.0	675.8

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

下面记录消息体为 100K 大小的性能测试数据:

表 4-41 消息为 100K 大小 nP\_nD\_nT 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	12/7/10	2.57	0.06	9.2	44.3	44.2	13.5
50/50/50/50	56/6/23	1.8	0.05	18.6	93.69	93.3	728.5

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

#### 4.6.2.2 P\_nD\_nT 消息处理

下面记录消息体为 1K 大小的性能测试数据:

表 4-42 消息为 1K 大小 P\_nD\_nT 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	17/14/15	2.6	0.04	4.7	1846.7	1839.0	8.7
10/10/10/10	70/64/67	2.55	0.06	18.1	7095.0	7091.7	9.7
50/50/50/50	65/61/63	2.6	0.1	14.5	5680.0	5656.0	9.2

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

#### 4.6.2.3 nP\_D\_nT 消息处理

下面记录消息体为 1K 大小的性能测试数据:

表 4-43 消息为 1K 大小 nP\_D\_nT 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	18/14/16	2.6	0.05	4.7	1842.6	1833.57	9.3
10/10/10/10	70/65/68	2.6	0.05	18.1	7098.6	7095.4	8.8
50/50/50/50	69/64/67	2.6	0.1	14.8	5804.9	5774.6	9.6

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

#### 4.6.2.4 P\_D\_nT 消息处理

下面记录消息体为 1K 大小的性能测试数据:

表 4-44 消息为 1K 大小 P\_D\_nT 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	22/17/20	2.57	32.8	2.0	788.8	787.7	19.0
10/10/10/10	28/23/27	2.3	48.5	2.5	961.7	862.6	176.0
50/50/50/50	28/24/26	2.47	48.5	2.4	924.0	920.1	195.0

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

下面记录消息体为 10K 大小的性能测试数据:

表 4-45 消息为 10K 大小 P\_D\_nT 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	22/19/21	2.6	36.5	9.2	430.2	429.9	11.4
50/50/50/50	30/25/28	2.3	65.7	9.3	437.6	437.6	150.6

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

下面记录消息体为 100K 大小的性能测试数据:

表 4-46 消息为 100K 大小 P\_D\_nT 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	15/11/13	2.56	44.5	8.1	38.8	38.8	42.4
50/50/50/50	31/21/26	2.25	78.0	13.9	67.77	67.7	282.0

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

#### 4.6.2.5 nP\_nD\_T 消息处理

下面记录消息体为 1K 大小的性能测试数据:

表 4-47 消息为 1K 大小 nP\_nD\_T 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	28/20/24	2.0	0.07	9.4	3851.9	3780.5	535.6
10/10/10/10	62/34/51	2.0	0.06	20.4	8396.5	8123.9	403.3
50/50/50/50	86/2/51	1.9	0.5	18.6	8375.0	8184.7	634.0

P/S/C/T: 表示发布者数/订阅者数/连接数/主题数

下面记录消息体为 10K 大小的性能测试数据:

表 4-48 消息为 10K 大小 nP\_nD\_T 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	1611/13	2.6	0.06	10.7	509.5	509.0	13.1
50/50/50/50	60/2/23	1.7	0.06	15.9	860.0	851.1	836.3

P/S/C/T: 表示发布者数/订阅者数/连接数/主题数

下面记录消息体为 100K 大小的性能测试数据:

表 4-49 消息为 100K 大小 nP\_nD\_T 消息处理性能数据

P/S/C/T	CPU% (Max /Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	13/10/11	2.57	0.05	10.7	51.9	51.9	17.6
50/50/50/50	66/36/47	1.8	0.05	37.0	186.3	184.4	715.0

P/S/C/T: 表示发布者数/订阅者数/连接数/主题数



#### 4.6.2.6 P\_nD\_T 消息处理

下面记录消息体为1K大小的性能测试数据:

表 4-50 消息为 1K 大小 P\_nD\_T 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	31/22/25	2.0	0.5	10.0	4075.5	3974.4	303.6
10/10/10/10	75/21/56	1.9	0.07	20.8	8557.0	8514.9	611.0
50/50/50/50	90/8/63	1.9	0.07	22.4	9944.4	9819.4	353.7

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

#### 4.6.2.7 nP\_D\_T 消息处理

下面记录消息体为 1K 大小的性能测试数据:

表 4-51 消息为 1K 大小 nP\_nD\_T 消息处理性能数据

P/S/C/T	CPU% (Max /Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	29/22/25	2.0	0.06	10.1	4107.1	4058.9	178.0
10/10/10/10	73/28/59	2.0	0.05	22.9	9644.4	9590.2	237.2
50/50/50/50	83/1/39	1.9	0.05	14.5	6265.4	6255.8	606.3

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

#### 4.6.2.8 P\_D\_T 消息处理

下面记录消息体为 1K 大小的性能测试数据:

表 4-52 消息为 1K 大小 P\_D\_T 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	23/21/22	2.57	40.5	2.2	892.1	886.5	27.5
10/10/10/10	23/14/18	2.25	91.1	1.5	704	600.8	327.4
50/50/50/50	22/19/21	2.4	50.8	1.7	701.6	639.1	202.8

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

下面记录消息体为 10K 大小的性能测试数据:

表 4-53 消息为 10K 大小 P\_D\_T 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	13/11/12	2.56	51.2	5.6	262.0	262.0	15.4
50/50/50/50	33/24/27	2.3	57.6	9.4	446.5	433.1	172.3

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

下面记录消息体为 100K 大小的性能测试数据:

表 4-54 消息为 100K 大小 P\_D\_T 消息处理性能数据

P/S/C/T	CPU% (Max/Min/Avg)	Avail Mem (GB)	DISK %	NIC (MBytes/s)	S- Msgs/s	R- Msgs/s	Heap (MB Used)
1/1/1/1	16/11/14	2.56	24.8	8.7	41.6	41.6	48.5
50/50/50/50	32/24/28	2.0	74.6	14.9	72.0	71.8	555.8

P/S/C/T: 表示发布者数/订阅者数/联接数/主题数

## 5 稳定性测试

### 5.1 JTang EJB 稳定性测试

#### 5.1.1 Socket 方式

JTang EJB 稳定性测试的网络和服务器环境使用专为稳定性测试所搭建的测试环境(见表 5-1 Socket 方式 JTang EJB 稳定性测试数据)。在稳定性测试过程中使用 Stateful Session Bean 性能测试中所使用的性能测试案例，以 1000 并发虚拟用户持续不间断的以 Socket 联接方式向 JTang 中间件服务器施加压力，经过 24 个小时，系统运行稳定，无异常发生。性能数据记录如下：

表 5-1 Socket 方式 JTang EJB 稳定性测试数据

运行时间 (s)	86,402
交易量	135,413,128
TPS	1,567
交易平均响应时间 (ms)	637
应用服务器 CPU%	48
应用服务器 Avail Mem (GB)	2.80
应用服务器 NIC (KB/s)	2,561
NIC Packets Outbound Errors	0
NIC Packets Received Errors	0
应用服务器 DISK%	1
JTang 中间件 JVM HEAP 启动参数	-Xms256m -Xmx1024m
运行 24 小时后已分配 HEAP 大小(MB)	383
运行 24 小时后，GC 后所使用 HEAP 大小 (MB)	252

### 5.1.2 RMI 方式

JTang EJB 稳定性测试的网络和服务器环境使用专为稳定性测试所搭建的测试环境(见表 3-2)。JTang EJB 稳定性测试的应用执行流程，将 Stateful Session Bean 性能测试和 Stateless Session Bean 性能测试中的操作流程融合成一个流程，即：

- 1) 创建一个 Stateful Session Bean 的实例
- 2) 调用 Stateful Session Bean 实例的两个 Methods
- 3) 释放删除 Stateful Session Bean 实例
- 4) 创建一个 Stateless Session Bean 的实例
- 5) 调用 Stateless Session Bean 实例的两个 Methods
- 6) 释放 Stateless Session Bean 实例

我们将上面一个完整流程定义成一个交易，并以 500 并发用户，以 RMI 的方式持续不间断的向 JTang 中间件服务器加压。起初 JTang 中间件服务器的 JVM 使用 Sun Microsystems JVM 1.4.2\_10 时，系统在持续压力下运行 10 个小时后，会出现网络访问故障，异常退出。后将 JVM 升级为 Sun Microsystems JVM 1.5.0\_07，系统在持续 500 并发用户的压力下，经过 24 小时不间断运行，未出现故障。性能数据记录如下：

表 5-2 RMI 方式 JTang EJB 稳定性测试数据

运行时间 (s)	86,403
交易量	12,753,468
TPS	148
交易平均响应时间 (ms)	6,774
应用服务器 CPU%	38
应用服务器 Avail Mem (GB)	3.01
应用服务器 NIC (KB/s)	3,110
NIC Packets Outbound Errors	0

NIC Packets Received Errors	0
应用服务器 DISK %	1
JTang 中间件 JVM HEAP 启动参数	-Xms256m -Xmx1024m
运行 24 小时后已分配 HEAP 大小(MB)	392
运行 24 小时后，GC 后所使用 HEAP 大小 (MB)	35

## 5.2 JTang Flow 稳定性测试

在 JTang Flow 的稳定性测试过程中，使用 Web 调用方式，使用同性能测试一样的网络环境和服务器环境(见表 5-3 JTang Flow 稳定性测试数据)。以 50 并发用户，持续无停顿时间的向应用服务器不间断施加工作压力负荷 16 个小时以上，系统仍可正常运行，无异常表现。系统在运行 16 个小时的性能数据记录如下：

表 5-3 JTang Flow 稳定性测试数据

运行时间(s)	57,960
交易量	254,326
TPS	4.4
交易响应时间 (s)	11.2
应用服务器 CPU%	2
应用服务器 Avail Mem (GB)	2.2
应用服务器 NIC (KB/s)	40
应用服务器 DISK%	0.5
DB 服务器 CPU%	98
DB 服务器 Avail Mem (GB)	2.8
DB 服务器 NIC (MB/s)	20
DB 服务器 DISK%	9.6
SQL Server Cache Hit Ratio %	82.5

SQL Server Dead Locks/s	0
SQL Server Lock Requests/s	2,233,626
SQL Server Lock Waits/s	150
SQL Server Lock Average Wait Time (ms)	50
SQL Server Lock Timeouts/s	12
SQL Server Latch Waits/s	180
SQL Server Latch Average Wait Time (ms)	40

从上面的数据可以看出，数据库服务器已处于满负荷状态，成为系统性能的瓶颈。

### 5.3 JTang MQ 稳定性测试

为了测试 JTang MQ 消息中间件服务器在长时间大压力下运行的稳定性，我们在多种压力场景下长时间运行 JTang MQ 消息中间件系统，测试过程中使用 1K 大小的消息体。

表 5-4 消息体大小为 1K，JTang MQ 稳定性测试数据

测试场景	APP-CPU (Max/Min /Avg)	APP- Avail Mem (GB)	APP DISK %	APP-NIC (MBytes/s)	Sending Msgs/s	Receiving Msgs/s	Heap Util. (MB)	测试环境
<b>P-P 下P_nT 方式</b> S/R/C/Q:50/ 50/50/20 运行64小时	35/17/32	2.3	52.1	1.4	545.7	544.9	444.6	使用 100M 网络下的 二路 Xeon 3.6GHz 服 务器
<b>P-P 下nP_T 方式</b> S/R/C/Q:50/ 50/50/20 运行64小时	44/0/28	2.76	0.3	6.0	2464.8	2464.6	113.8	使用 100M 网络下的 四路 Xeon 3.33GHz 服务器
<b>P-S下 P_D_nT方式</b> P/S/C/T:50/5 0/50/50 运行40小时	31/24/28	2.6	45.4	2.8	1091.2	1091.2	36.2	使用 1G网 络下的二 路Xeon 双 核 服务器
<b>P-S下 nP_nD_T方 式</b>	91/13/66	2.2	0.3	14.9	5872.8	5871.2	677.8	使用 1G网 络下的四 路 Xeon

P/S/C/T:50/50/50/50 运行64小时								3.33GHz 服务器
P-P 下 nP_nT 方式 S/R/C/Q:50/50/50/20 运行24小时	98/94/97	3.27	0.7	13.1	5148.4	5148.4	15.7	使用100M 网络下的 二路 Xeon 3.6GHz 服 务器
P-P 下P_T 方式 S/R/C/Q:50/50/50/20 运行24小时	26/13/18	2.3	79.3	0.3	112.8	109.0	742.3	使用1G网 络下的四 路 Xeon 3.33GHz 服务器
P-P 下P_T 方式 S/R/C/Q:50/50/50/20 运行24小时	8/3/5	2.7	80.4	0.3	119.0	117.2	370.1	使用100M 网络下的 四路 Xeon 3.33GHz 服务器
P-S下 nP_nD_nT 方式 P/S/C/T:50/50/50/50 运行24小时	86/12/48	2.1	0.9	7.3	2814.8	2810.2	12.9	使用1G网 络下的四 路 Xeon 3.33GHz 服务器
P-S下P_D_T 方式 P/S/C/T:50/50/50/50 运行24小时	25/11/19	2.67	75.1	0.8	301.9	299.5	464.6	使用100M 网络下的 二路 Xeon 3.6GHz 服 务器
P-P 下nP_nT 方式 S/R/C/Q:50/50/50/20 运行15小时	20/5/16	1.6	88.7	1.5	574.1	569.8	584.8	使用1G网 络下的二 路 Xeon 双 核服务器
P-P 下nP_T 方式 S/R/C/Q:50/50/50/20 运行15小时	80/13/53	2.1	0.9	9.6	3849.0	3844.6	362.4	使用1G网 络下的四 路 Xeon 3.33GHz 服务器

在上面的稳定性测试场景中，JTang MQ 消息中间件服务器均稳定运行，无异常表现，消息接收速度和消息发送速度在整个测试运行区间表现均匀。

## 6 性能分析

### 6.1 JTang EJB 性能分析

#### 6.1.1 应用性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateful Session Bean 在 Socket 方式下，在测试过程中所表现的压力性能趋势曲线：

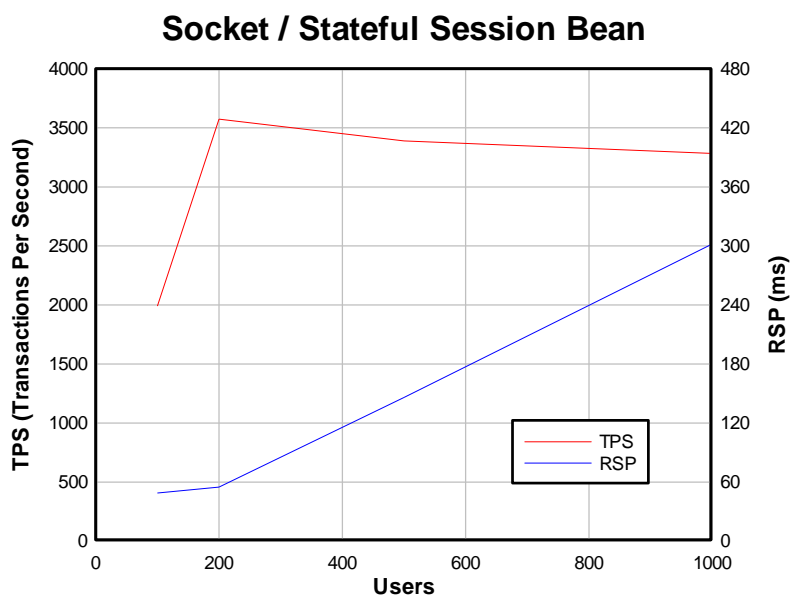


图 6-1 Socket 方式下，Stateful Session Bean 压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateful Session Bean 在 RMI 方式下，在测试过程中所表现的压力性能趋势曲线：



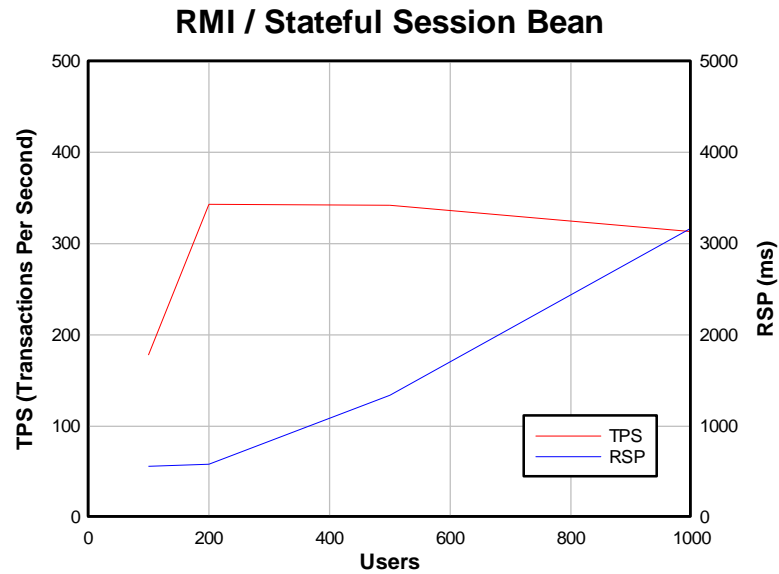


图 6-2 RMI 方式下，Stateful Session Bean 压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateful Session Bean 在 HTTP 方式下，在测试过程中所表现的压力性能趋势曲线：

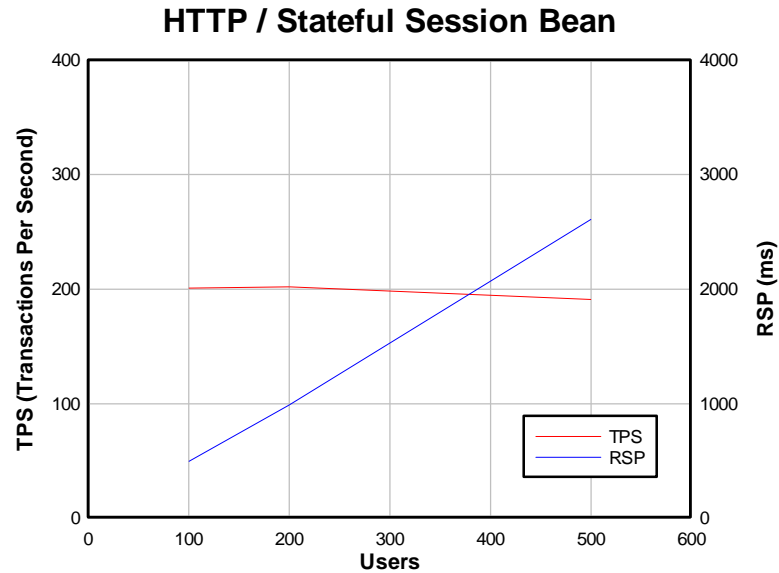


图 6-3 HTTP 方式下，Stateful Session Bean 压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateful Session Bean 在 SSL 方式下，在测试过程中所表现的压力性能趋势曲线：

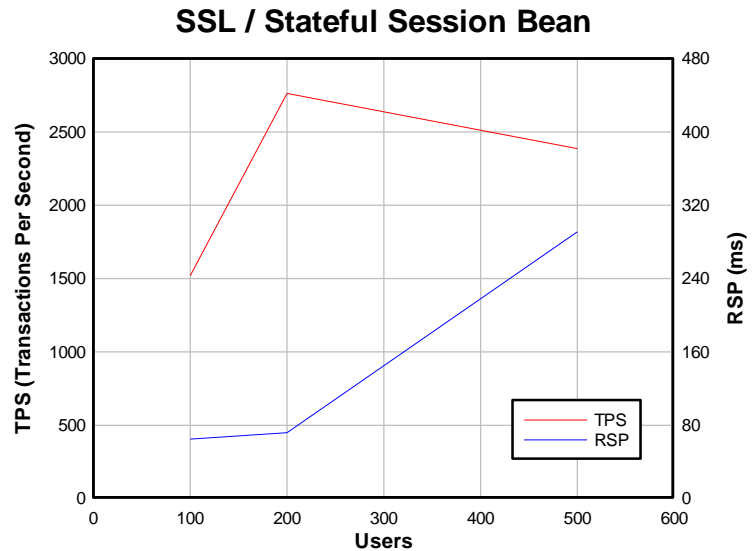


图 6-4 SSL 方式下，Stateful Session Bean 压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateless Session Bean 在 Socket 方式下，在测试过程中所表现的压力性能趋势曲线：

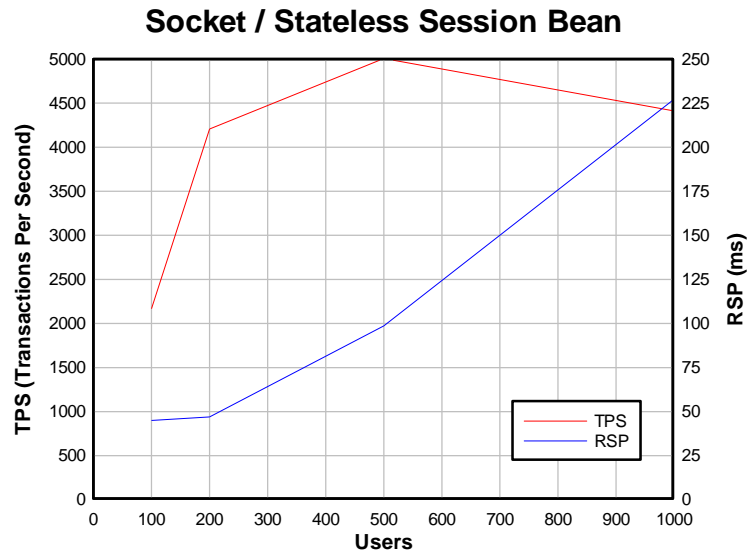


图 6-5 Socket 方式下，Stateless Session Bean 压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateless Session Bean 在 RMI 方式下，在测试过程中所表现的压力性能趋势曲线：

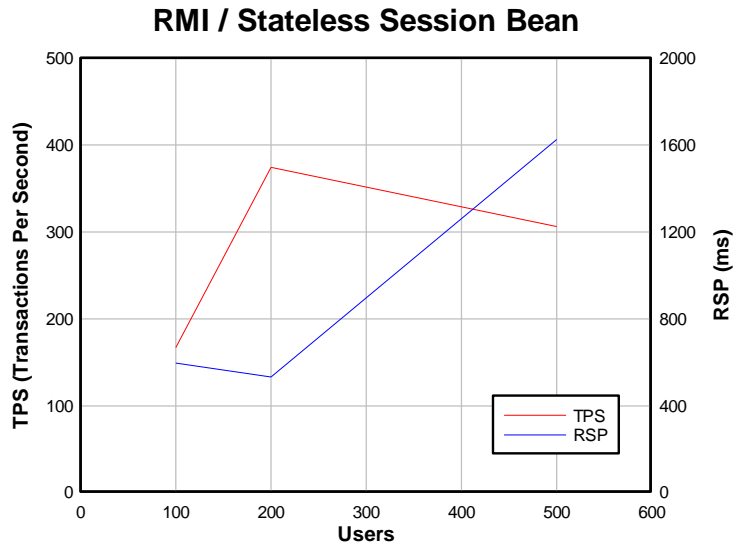


图 6-6 RMI 方式下，Stateless Session Bean 压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateless Session Bean 在 HTTP 方式下，在测试过程中所表现的压力性能趋势曲线：

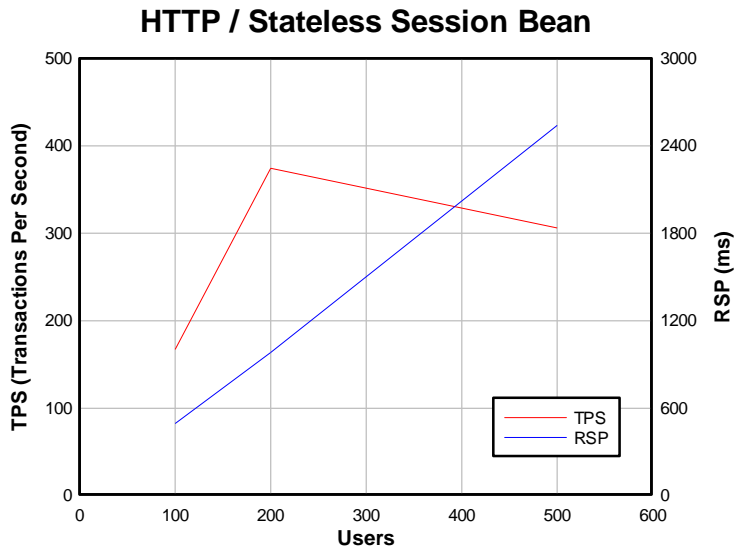


图 6-7 HTTP 方式下，Stateless Session Bean 压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang 中间件 Stateless Session Bean 在 SSL 方式下，在测试过程中所表现的压力性能趋势曲线：

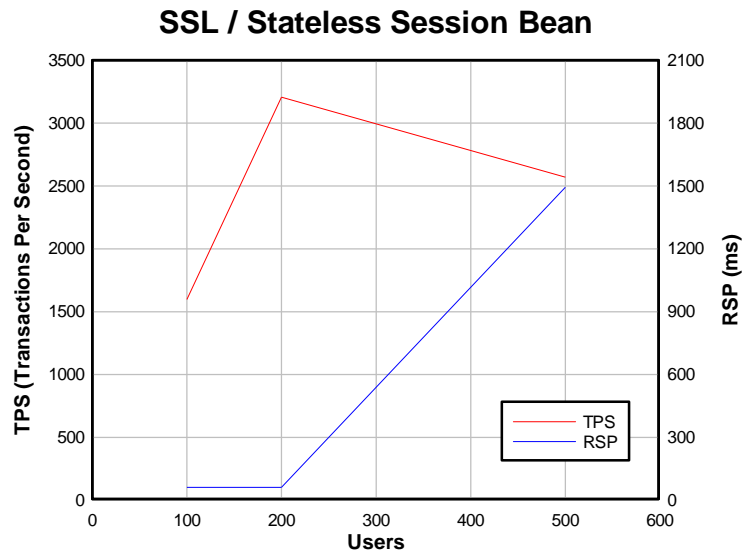


图 6-8 SSL 方式下，Stateless Session Bean 压力性能趋势曲线

从上面的压力性能趋势曲线我们可以看出，在各种协议的访问方式下，JTang EJB 中间件的压力性能趋势均符合通常应用的压力性能趋势，即随着并发用户的增加，或者说工作负荷压力的增加，服务器的服务吞吐量（表现为 TPS）会达到一个峰值，系统的响应时间会随着工作负荷压力的增加而增加。

6.1.2 性能对比

下图，展示了在 200 并发用户和 500 并发用户的情况下，JTang EJB 中间件 Stateful Session Bean 在不同协议的访问方式下的性能(响应时间和 TPS)对比：

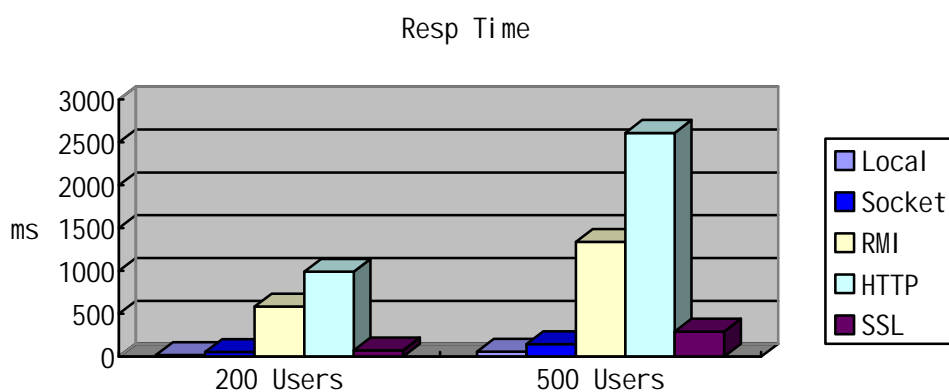


图 6-9 Stateful Session Bean 在不同协议访问方式下的响应时间性能对比

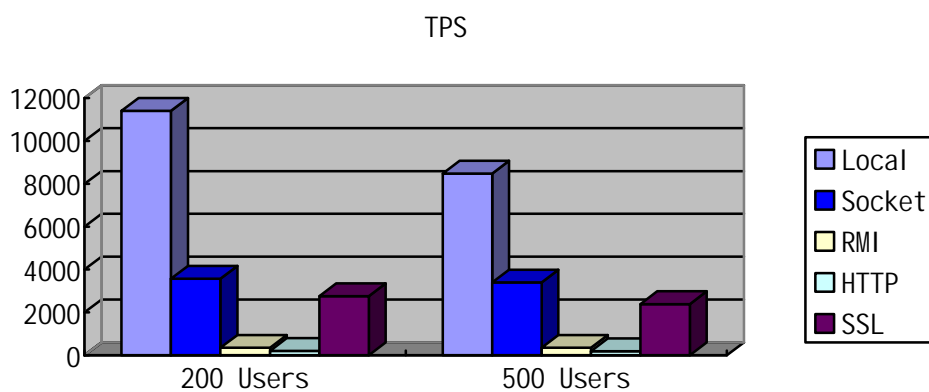


图 6-10 Stateful Session Bean 在不同协议访问方式下的 TPS 性能对比

从上图可以看出应用系统在本地调用方式下表现的性能最好，在 Socket 方式下次之，在 HTTP 方式下最差。

对于 JTang EJB 中间件 Stateless Session Bean 在不同协议的访问方式下的性能对比也类同。下图，展示了在 200 并发用户和 500 并发用户情况下，JTang EJB 中间件 Stateless Session Bean 在不同协议的访问方式下的性能对比图：

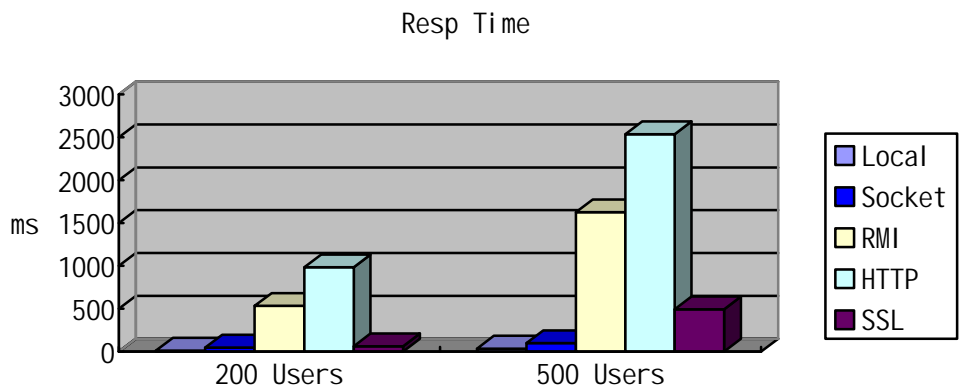


图 6-11 Stateless Session Bean 在不同协议访问方式下的响应时间性能对比

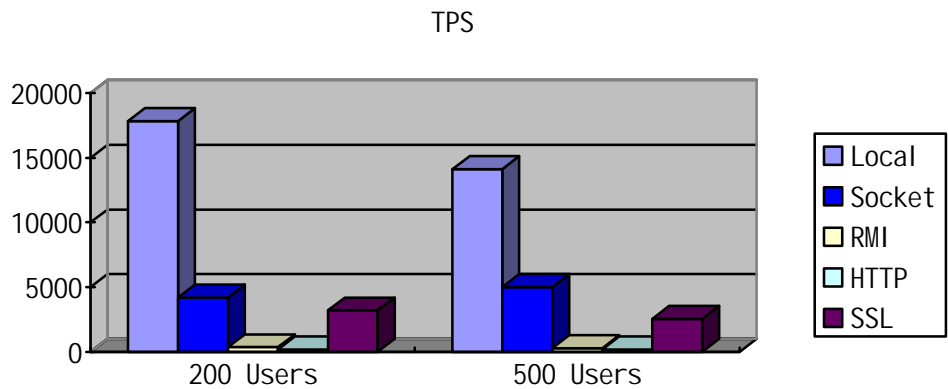


图 6-12 Stateless Session Bean 在不同协议访问方式下的 TPS 性能对比

下图给出了在 Socket 联接方式下 JTang EJB 中间件在测试过程中，Stateful Session Bean 与 Stateless Session Bean 所表现的应用性能对比：

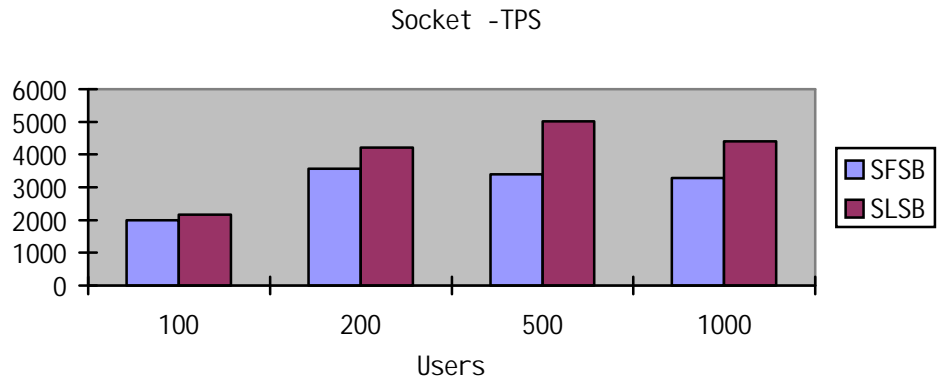


图 6-13 Socket 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比

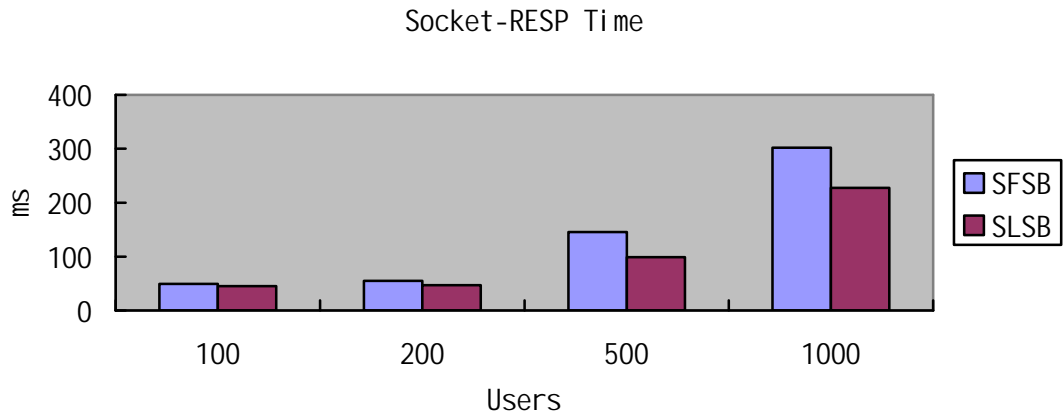


图 6-14 Socket 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比

从上图看，JTang EJB 中间件在 Socket 方式测试过程中，Stateless Session Bean 表现出的性能大致优于 Stateful Session Bean 百分之二十左右。

下图给出了 HTTP 联接方式下 JTang EJB 中间件在测试过程中，Stateful Session Bean 与 Stateless Session Bean 所表现的应用性能对比：

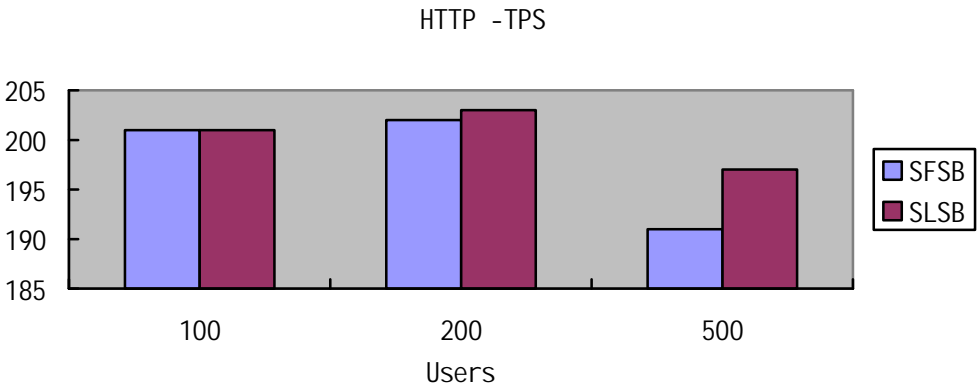


图 6-15 HTTP 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比

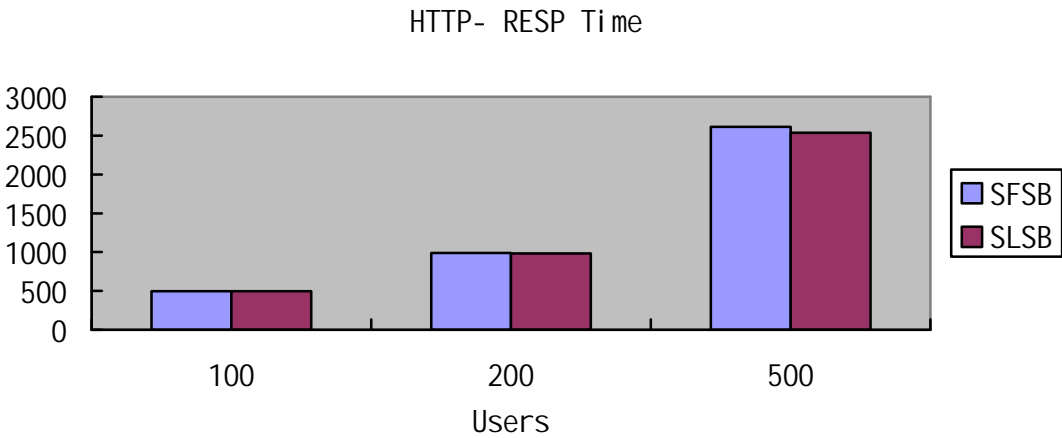


图 6-16 HTTP 方式下，Stateful Session Bean 和 Stateless Session Bean 的性能对比

从上图看，JTang EJB 中间件在 HTTP 联接方式的测试过程中，Stateless Session Bean 表现出的性能与 Stateful Session Bean 相比，并无大的差别，看来应用系统在其他方面的处理，比如 HTTP 请求解析和 HTTP 页面生成等处理，占据了系统处理的大部分时间。同



样，对于 JTang EJB 中间件在 RMI 联接方式下，Stateless Session Bean 与 Stateful Session Bean 在测试过程中也表现出了类似的性能对比。

### 6.1.3 系统性能曲线

纵观整个 JTang EJB 的测试结果数据，系统对磁盘 I/O 的操作很少，因此磁盘 I/O 在所有的 JTang EJB 测试场景中，不会成为系统性能增长的瓶颈。而 CPU 和网络在高负荷的工作压力下，都表现了较高的利用率。下图展示了 JTang EJB 中间件在 Socket 方式下，Stateful Session Bean 系统性能表现曲线：

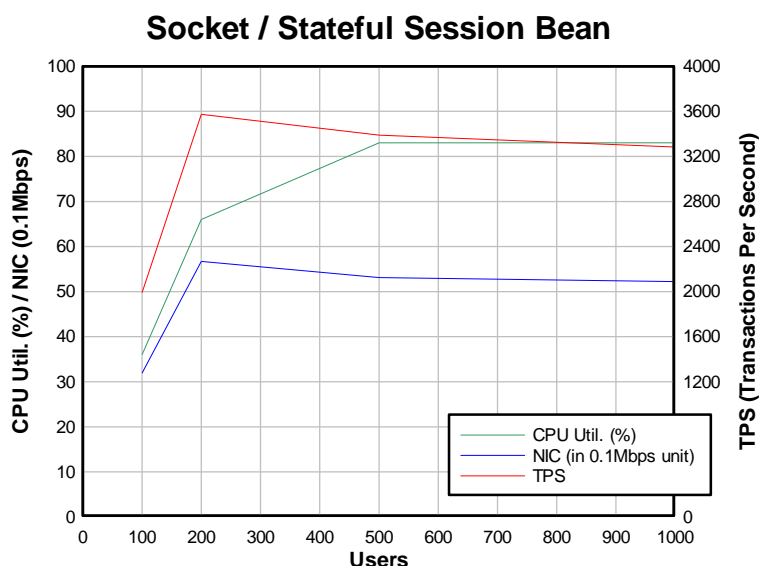


图 6-17 Socket 方式下，Stateful Session Bean 系统性能表现曲线

从上图可以看出，网络性能指标在不同的压力下所表现的系统性能曲线的走向趋势与应用服务器服务吞吐量—TPS 性能曲线的走向趋势和峰值拐点一致，另外考虑到性能测试过程中的网络环境为千兆网，因此在上图的系统性能表现曲线上，网络的吞吐量在 200 并发用户处达到峰值点后不再增长，不是网络处理达到饱和极限所致，而是由于应用服务器的服务吞吐量达到饱和所致。同样，在测试过程中，类似的系统性能表现趋势也体现在 Stateless Session Bean 上，如下图所示：

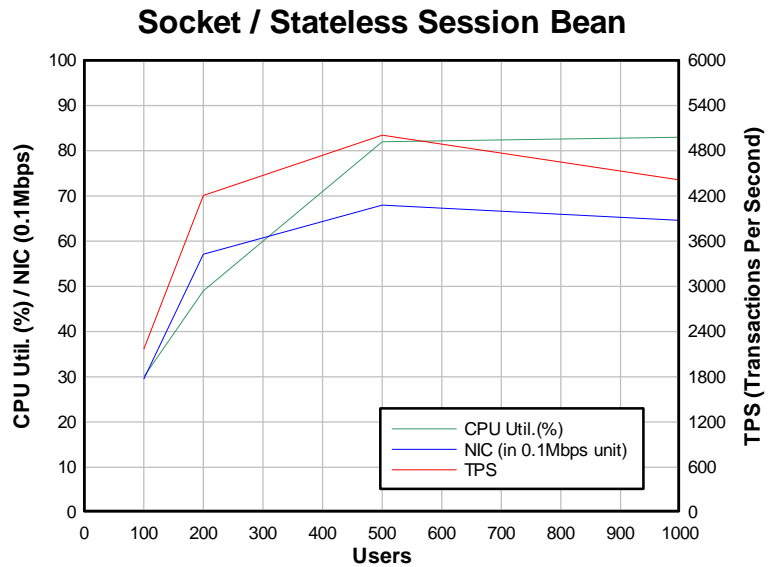


图 6-18 Socket 方式下，Stateless Session Bean 系统性能表现曲线

下图展示了 JTang EJB 中间件在 RMI 联接方式下，Stateful Session Bean 系统性能表现曲线：

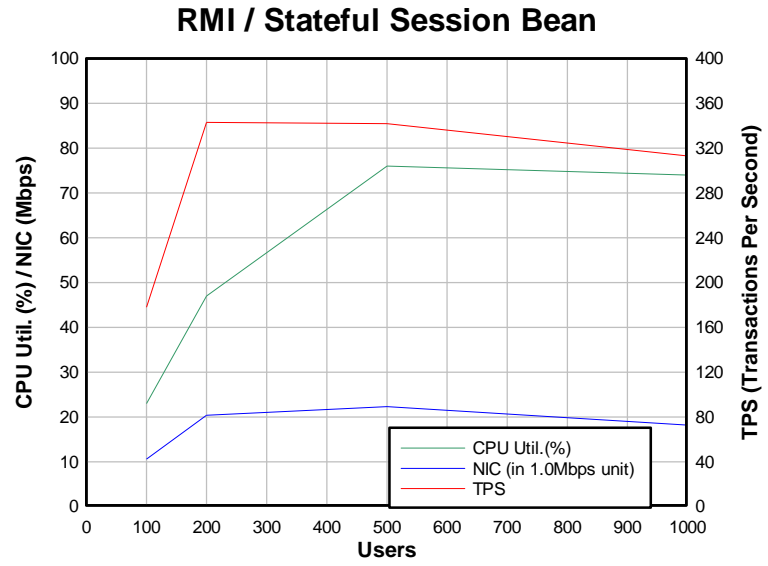


图 6-19 RMI 方式下，Stateful Session Bean 系统性能表现曲线

下图展示了 JTang EJB 中间件在 RMI 联接方式下，Stateless Session Bean 系统性能表现曲线：

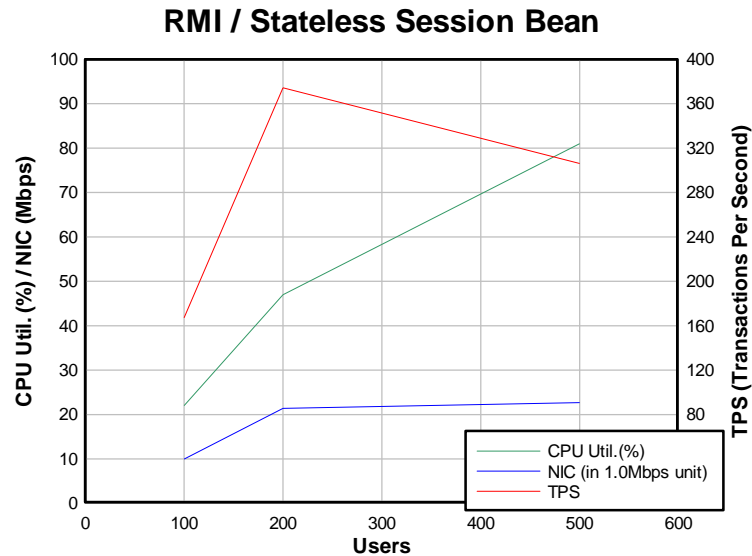


图 6-20 RMI 方式下，Stateless Session Bean 系统性能表现曲线

## 6.2 三层架构 Web 应用

下图给出了在不同的并发用户的压力下，所测试的三层架构的简单 Web 应用，在测试过程中所表现的压力性能趋势曲线：

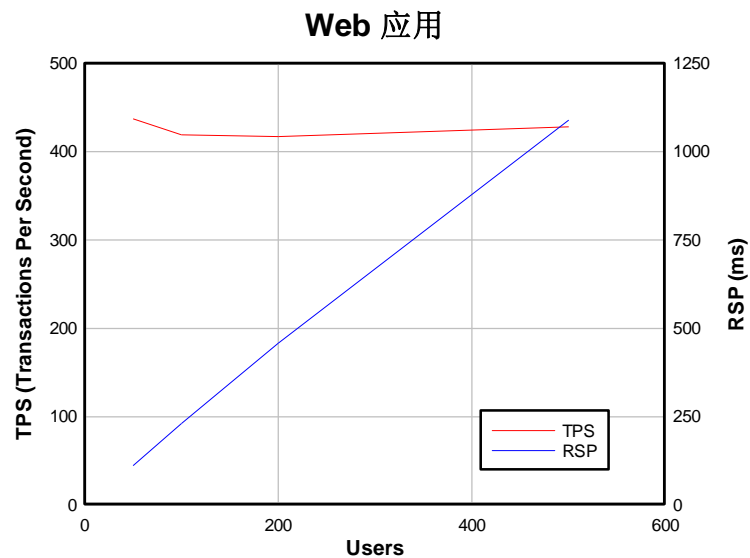


图 6-21 三层架构的简单 Web 应用压力性能趋势曲线

从性能测试数据来看，系统资源利用较平稳，增加应用服务器系统的工作负载压力，系统的响应时间相应增长，应用系统服务的并发执行度已达到最大。

### 6.3 JTang Flow 性能分析

下图给出了在不同的并发用户的压力下，JTang Flow 中间件在直接访问的方式下，在性能测试过程中所表现的压力性能趋势曲线：

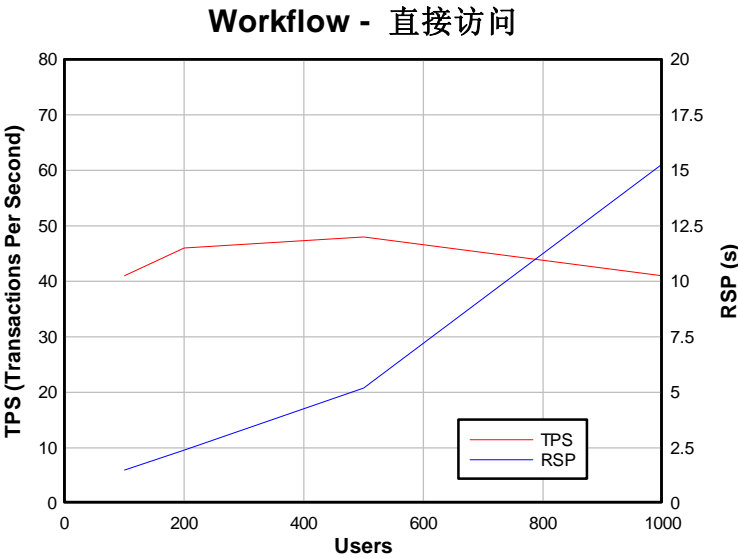


图 6-22 JTang Flow 在直接访问方式下压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang Flow 中间件在 Web 访问的方式下，在性能测试过程中所表现的服务器系统 CPU% 增长趋势曲线：

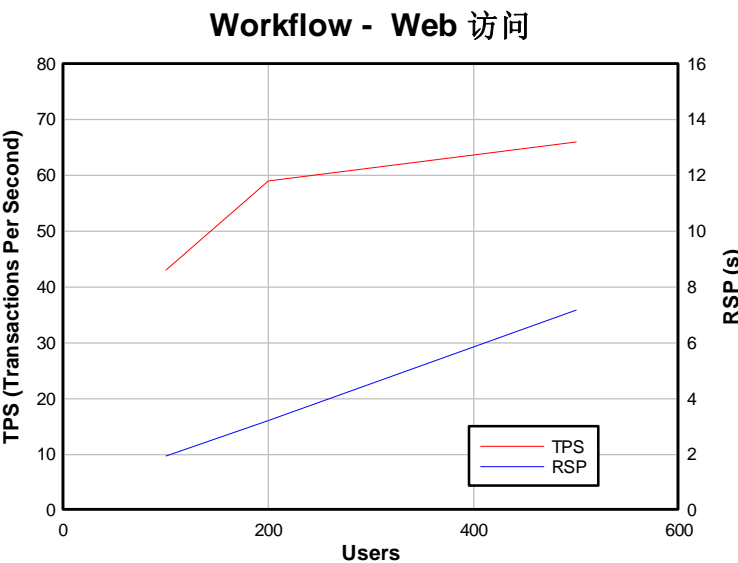


图 6-23 JTang Flow 在 WEB 访问方式下压力性能趋势曲线

下图给出了在不同的并发用户的压力下，JTang Flow 中间件在直接访问方式下和 Web 访问方式下，在性能测试过程中所表现的服务器系统 CPU% 增长趋势曲线：

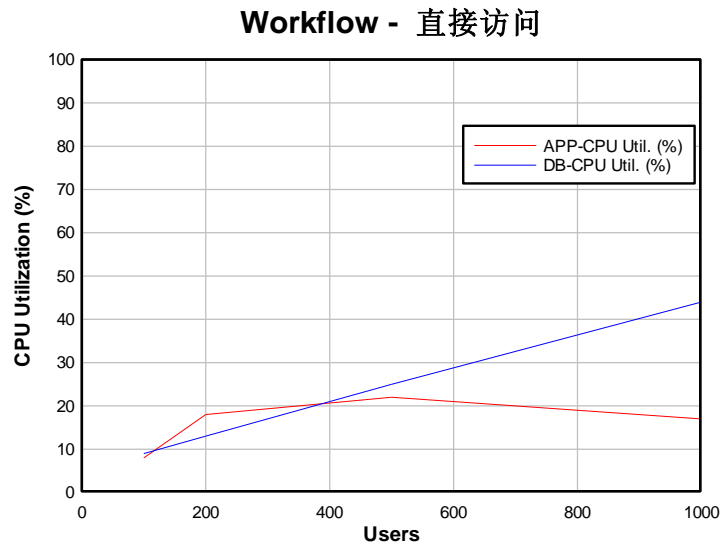


图 6-24 JTang Flow 在直接访问方式下 CPU 利用率曲线

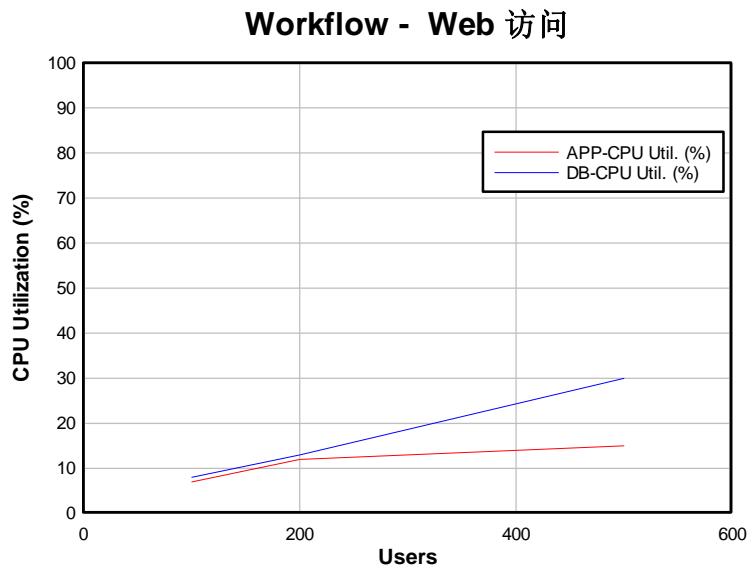


图 6-25 JTang Flow 在 WEB 访问方式下 CPU 利用率曲线

从性能测试数据来看，无论是直接访问方式还是 Web 访问方式，系统内存、网络都处于较低的利用率。而数据库服务器的磁盘操作都较为繁忙，数据库服务器系统 CPU 利用率随系统工作负荷的增加呈线性的上升，数据库服务器的数据处理能力将成为 JTang Flow 应用系统性能提升的关键。

## 6.4 JTang Form 性能分析

下图给出了在不同的并发用户的压力下，JTang Form 中间件在有 10 万条视图历史记录数据的情况下，在性能测试过程中所表现的压力性能趋势曲线：

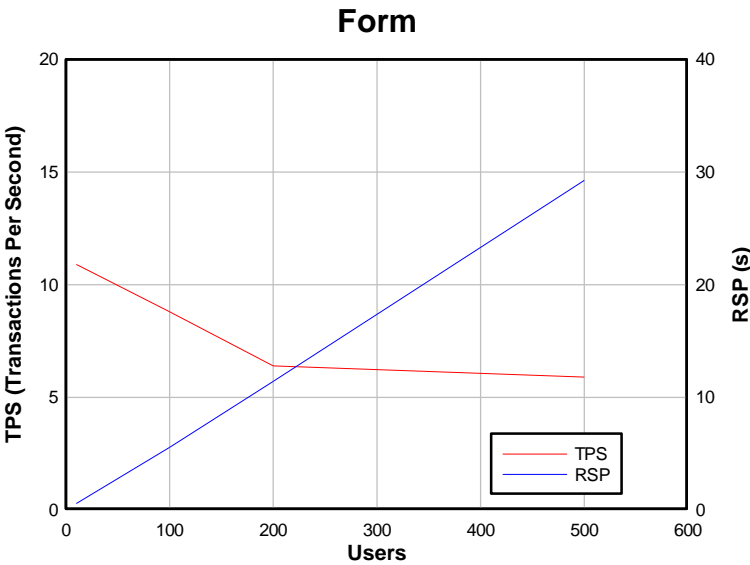


图 6-26 JTang Form 性能趋势曲线

从性能测试数据来看，应用服务器的磁盘操作非常繁忙，在所有的 JTang Form 性能测试场景，磁盘 I/O 的服务率达到过 100%。应用服务器的磁盘 I/O 将成为 JTang Form 应用系统性能提升的关键。

### 6.5 JTang PMI 性能分析

下图给出了在不同的并发用户的压力下，JTang PMI 中间件在性能测试过程中所表现的  
压力性能趋势曲线：

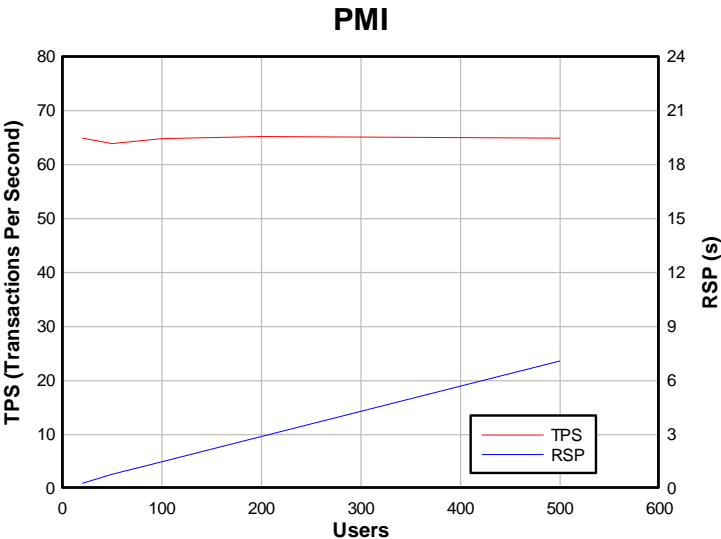


图 6-27 JTang PMI 性能趋势曲线

从系统性能测试数据来看，应用服务器各项系统资源利用率都极低，系统压力主要在数据库服务器。JTang PMI 的运算主体是数据库服务器上的存储过程，JTang PMI 模块性能主要取决于数据库服务器对 PMI 存储过程的执行效率。



## 6.6 JTang MQ 性能分析

### 6.6.1 P-P 模式

下图展示了 JTang MQ 消息中间件在一定的发送者数和接收者数的测试场景下，所表现的不同消息处理方式的性能对比。

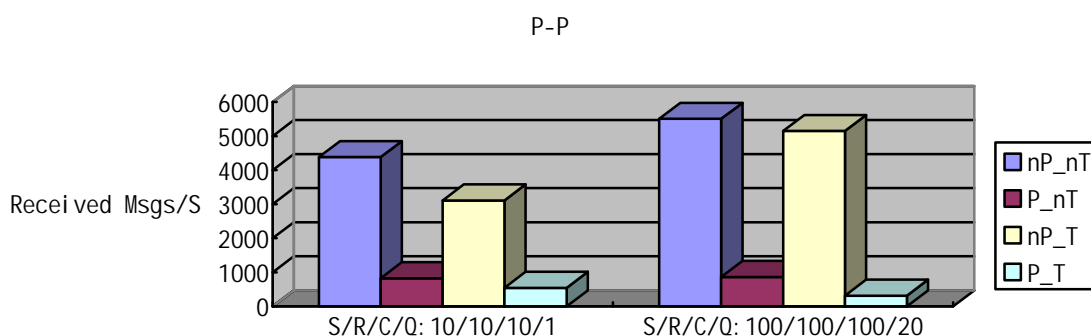


图 6-28 JTang MQ 在点对点模式下不同消息处理方式性能对比

由上图可以看出，消息的持久化处理，由于要进行数据库数据的写操作，会使消息处理的速度降低 5 倍以上。相对而言，消息的事务化处理，使消息处理的速度的降低的幅度要小得多。

消息的持久化处理，由于要进行大量的数据库数据的写操作，因此磁盘的性能至关重要。下图展示了 JTang MQ 消息中间件在不使用磁盘性能优化的高级特性和使用磁盘性能优化的高级特性(启动磁盘写缓存)，在 P\_nT 消息处理方式下的测试场景所表现的消息处理性能对比。

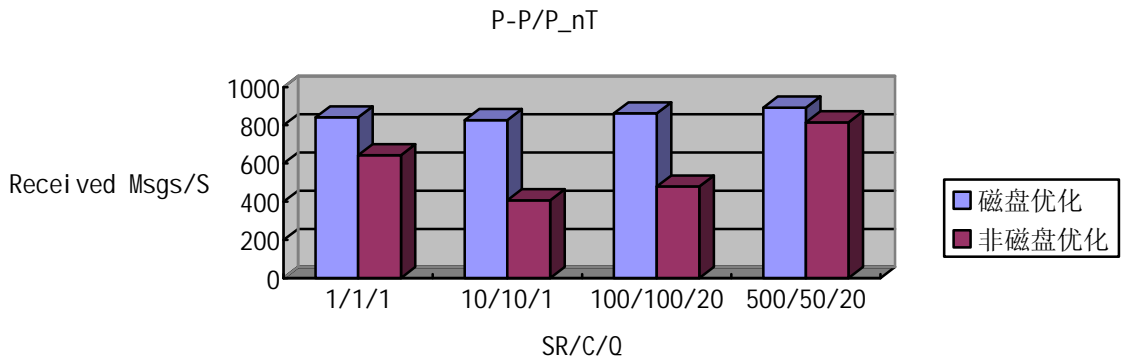


图 6-29 点对点模式下 P\_nT 处理方式，磁盘优化和非磁盘优化性能对比

下图展示了 JTang MQ 消息中间件在不使用磁盘性能优化的高级特性和使用磁盘性能优化的高级特性，P\_T 消息处理方式下的测试场景所表现的消息处理性能对比。

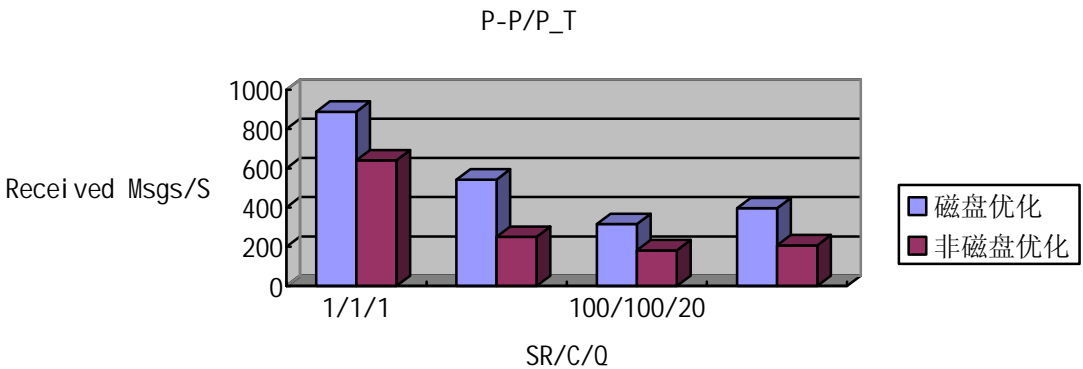


图 6-30 点对点模式下 P\_T 处理方式，磁盘优化和非磁盘优化性能对比

下图展示了 JTang MQ 消息中间件在 nP\_nT 消息处理方式下的测试场景，使用不同大小的消息体所表现的消息处理性能对比。

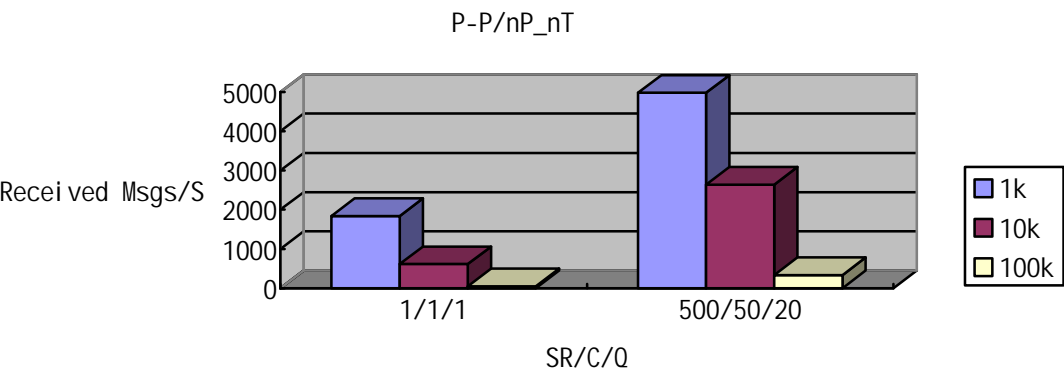


图 6-31 点对点模式下 nP\_nT 处理方式下不同消息体大小性能对比

下图展示了 JTang MQ 消息中间件在 P\_nT 消息处理方式下的测试场景，使用不同大小的消息体所表现的消息处理性能对比。

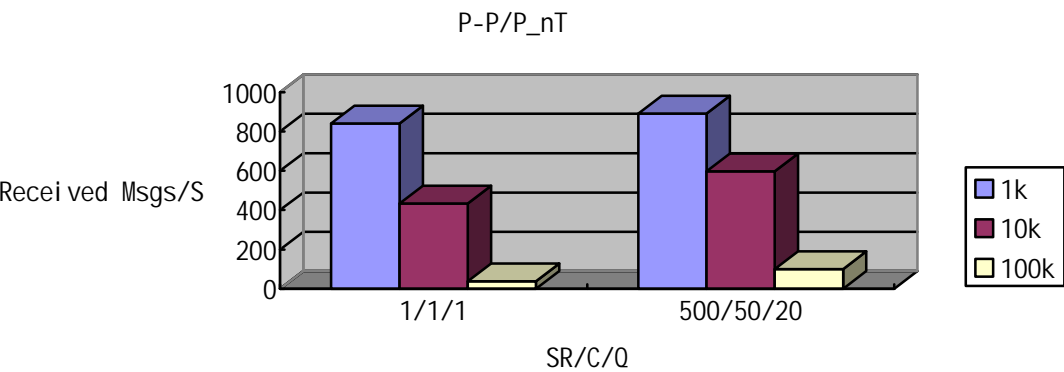


图 6-32 点对点模式下 P\_nT 处理方式下不同消息体大小性能对比

下图展示了 JTang MQ 消息中间件在 nP\_T 消息处理方式下的测试场景，使用不同大小的消息体所表现的消息处理性能对比。

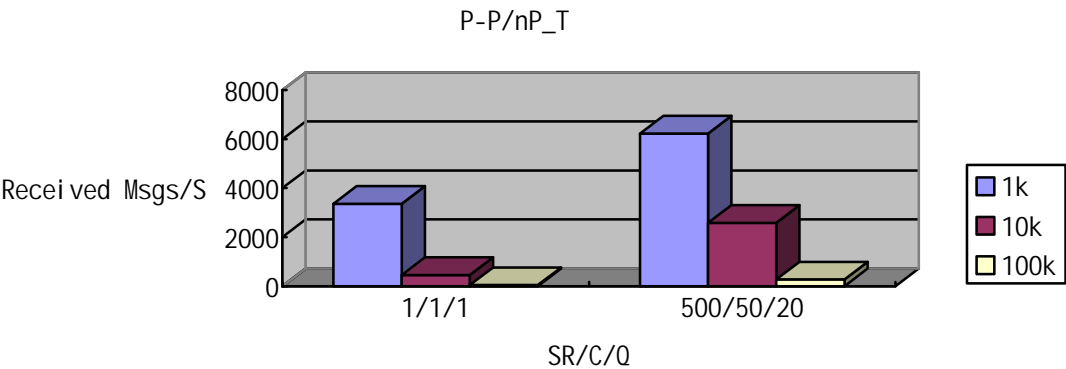


图 6-33 点对点模式下 nP\_T 处理方式下不同消息体大小性能对比

下图展示了 JTang MQ 消息中间件在 P\_T 消息处理方式下的测试场景，使用不同大小的消息体所表现的消息处理性能对比。

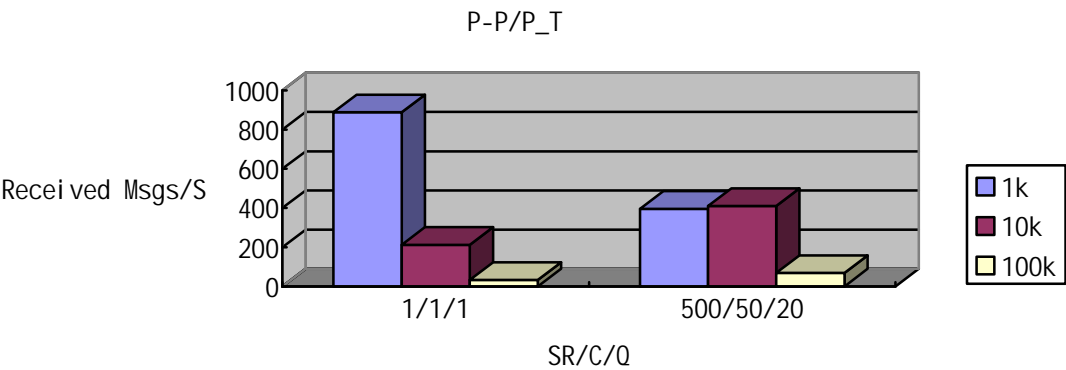


图 6-34 点对点模式下 P\_T 处理方式下不同消息体大小性能对比

下图展示了在较多发送和接收联接的情况下，JTang MQ 消息中间件在不同消息处理方式下的 CPU 利用率对比(消息体为 1k 大小)：

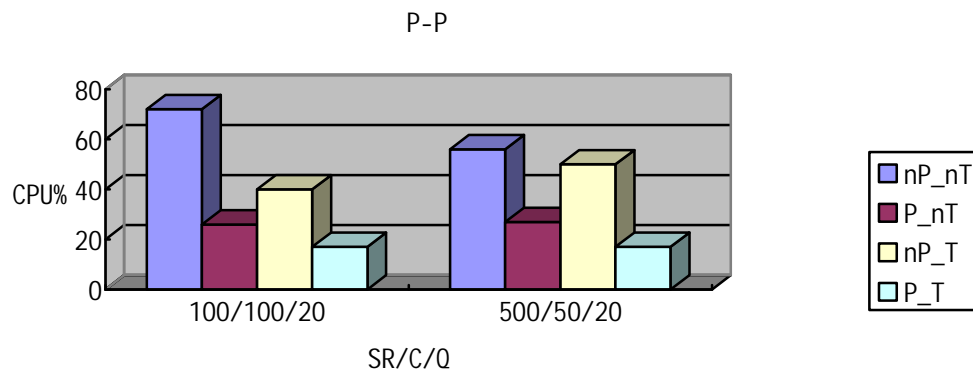


图 6-35 不同消息处理方式下的 CPU 利用率对比

从上图可以看出，消息的持久化处理由于需要写数据库，会加长应用系统的顺序执行，减小并发度，导致系统的 CPU 利用率的降低。

### 6.6.2 P-S 模式

下面两个图分别展示了 JTang MQ 消息中间件在 P/S/C/T 为 10/10/10/10 和 50/50/50/50 下，使用不同的消息处理方式，在性能测试中所表现的性能对比。

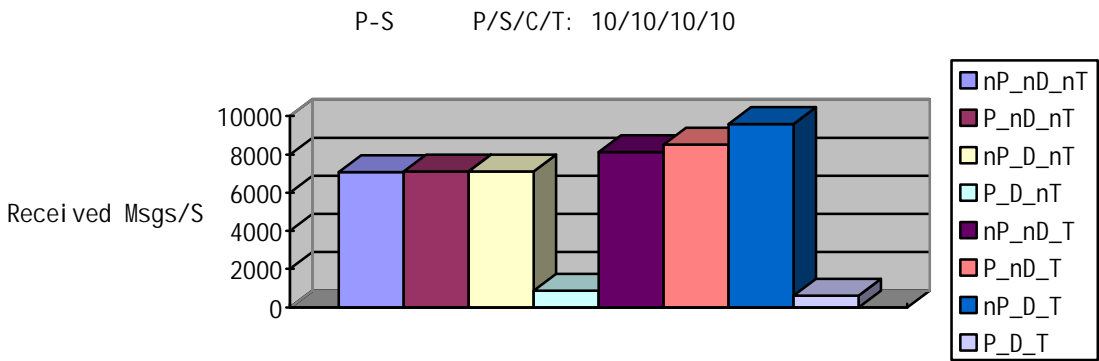


图 6-36 发布-订阅模式下不同消息处理方式下性能对比

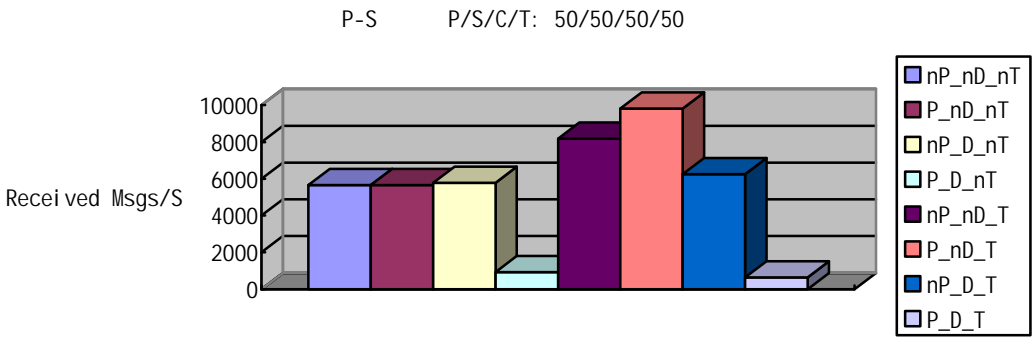


图 6-37 发布-订阅模式下不同消息处理方式下性能对比

由上两图可以看出，在消息的持久化处理且用户联接的可持续化处理的组合方式下，由于会进行数据库数据的写操作，从而使消息处理的速度大幅的降低。

下图展示了 JTang MQ 消息中间件在 nP\_nD\_nT 方式下的测试场景使用不同大小的消息体所表现的消息处理性能对比。

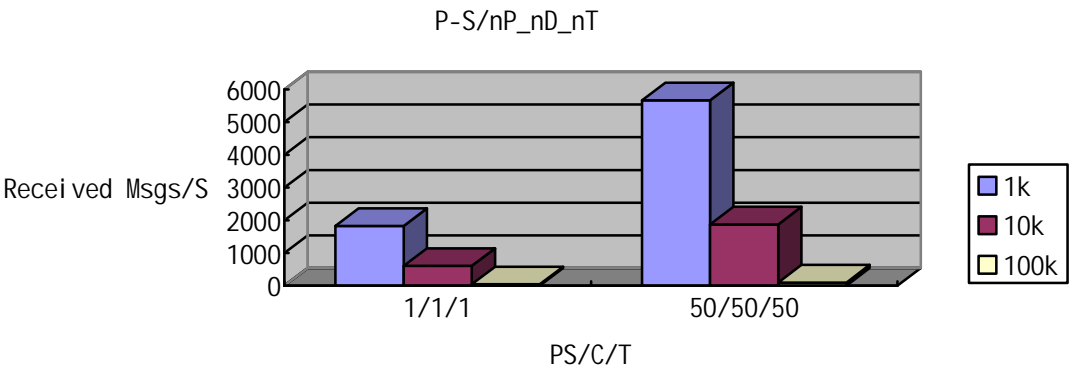


图 6-38 发布-订阅模式下 nP\_nD\_nT 处理方式下不同消息体大小性能对比

下图展示了 JTang MQ 消息中间件在 P\_D\_nT 方式下的测试场景使用不同大小的消息体所表现的消息处理性能对比。

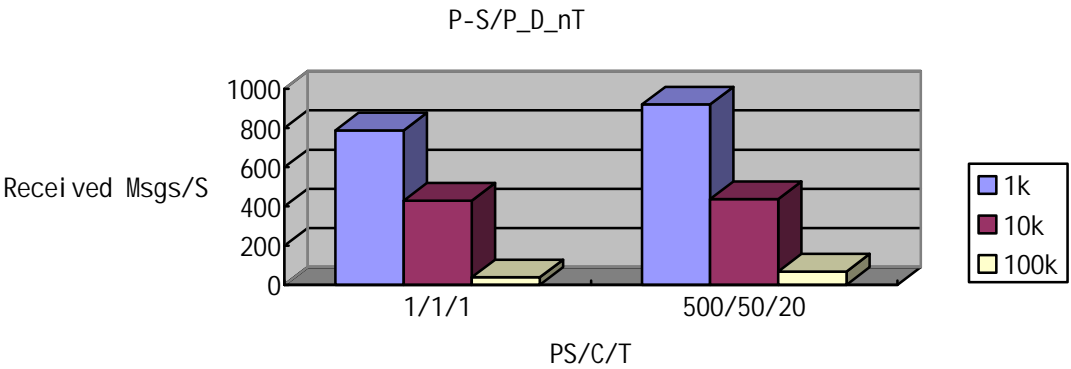


图 6-39 发布-订阅模式下 P\_D\_nT 处理方式下不同消息体大小性能对比

下图展示了 JTang MQ 消息中间件在 nP\_nD\_T 方式下的测试场景使用不同大小的消息体所表现的消息处理性能对比。

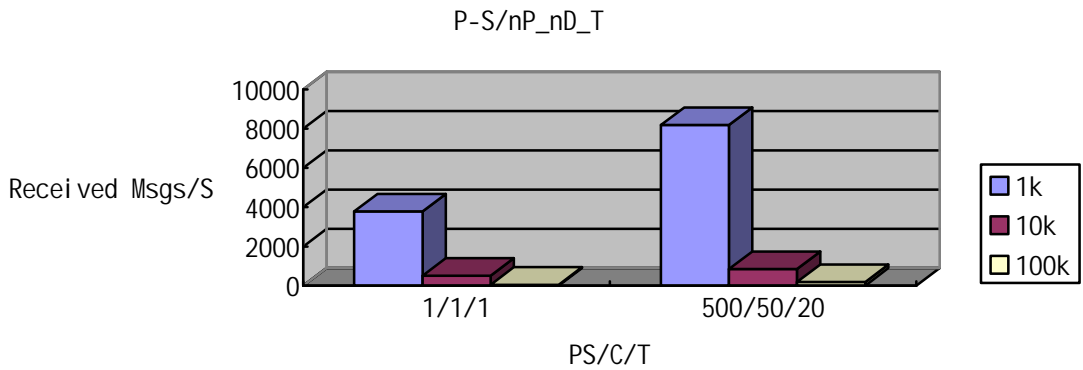


图 6-40 发布-订阅模式下 nP\_nD\_T 处理方式下不同消息体大小性能对比

下图展示了 JTang MQ 消息中间件在 P\_D\_T 方式下的测试场景使用不同大小的消息体所表现的消息处理性能对比。

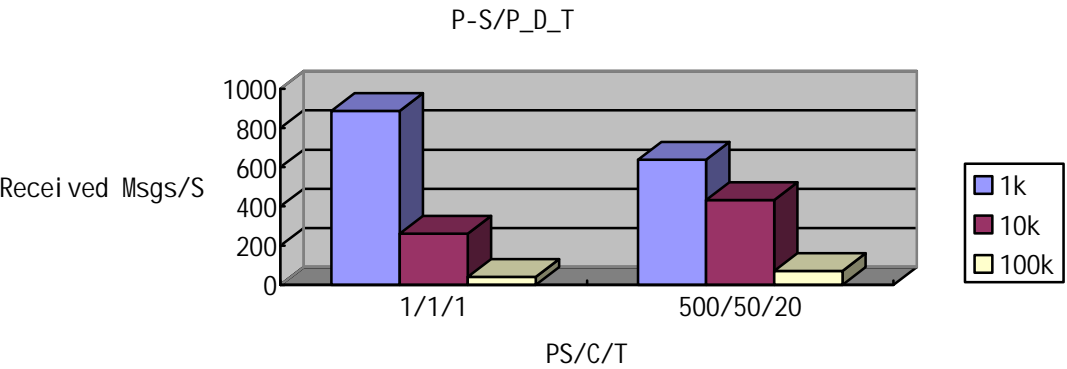


图 6-41 发布-订阅模式下 P\_D\_T 处理方式下不同消息体大小性能对比

下图展示了在较多发布者和订阅者联接的情况下，JTang MQ 消息中间件在不同消息处理方式下的 CPU 利用率对比(消息体为 1K 大小)：



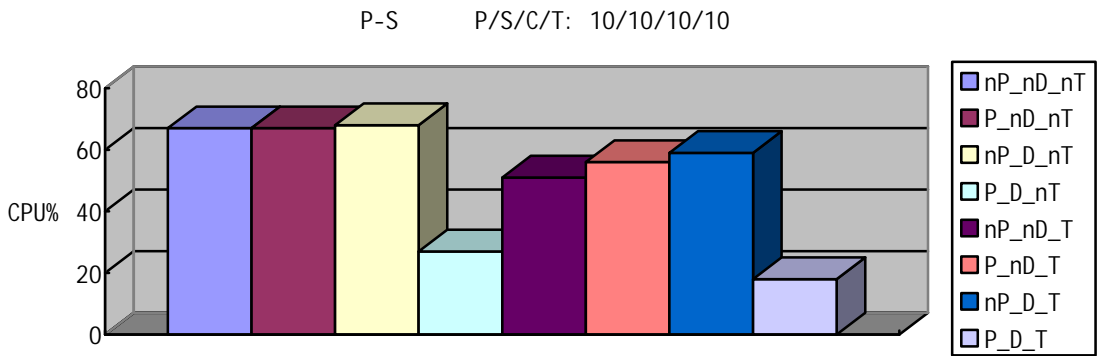


图 6-42 发布-订阅模式下不同消息处理方式下 CPU 利用率对比

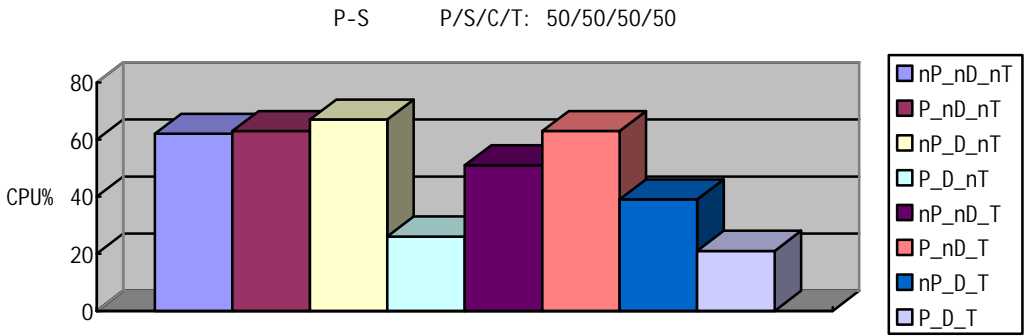


图 6-43 发布-订阅模式下不同消息处理方式下 CPU 利用率对比

从上图可以看出，在消息的持久化和用户联接的可持续化的组合方式下，由于会导致进行数据库数据的写操作，会加长应用系统的顺序执行，减小并发度，导致系统的 CPU 利用率的降低。

下图，对在同样的千兆网络环境下，使用 4 路基于 Xeon 3.33GHz 处理器的服务器(具有 Hyper-Threading) 和 2 路基于 Xeon 51602.66GHz 双核处理器的服务器在 P-S 模式下，使用同样的消息处理方式下的性能对比：

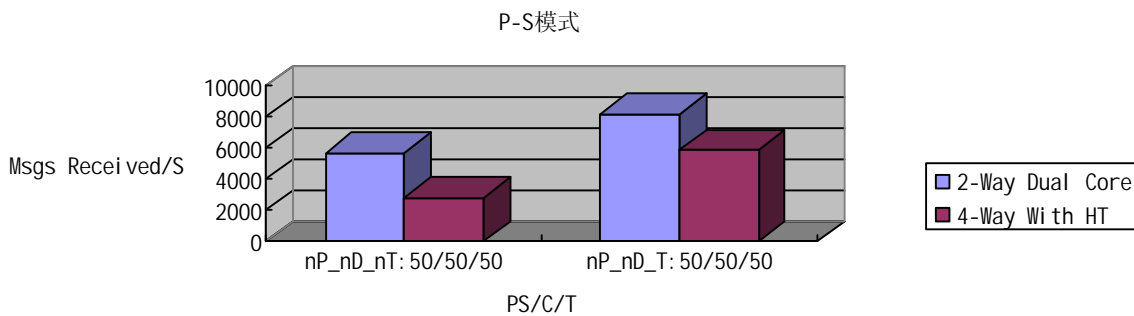


图 6-44 四路基于 Xeon 3.33GHz 的服务器(具有 Hyper-Threading)和两路基于双核 Xeon 2.66GHz 的服务器在 P-S 模式下，使用同样的消息处理方式下的性能对比

从上图可以看出，在我们所设定的测试场景，2 路基于 Xeon 5160 2.66GHz 双核处理器的服务器表现出比 4 路基于具有超线程的单核 Xeon 3.33GHz 处理器的服务器更强的性能(百分之三十以上)。

## 7 总结与建议

通过对钱塘（JTang）中间件平台软件的主要模块的性能测试和稳定性测试，我们可以获取下列经验知识总结和建议：

- I 钱塘（JTang）中间件平台软件作为轻量级应用服务器，在我们的测试场景中，所有模块的测试均测试到了 500 并发用户（某些模块测试到了 1000 并发用户），并在如此高的并发用户下，系统的性能仍然保持可用性，系统响应时间处于可以接受的范围。
- I 许多模块的网络吞吐量，如 JTang MQ、JTang EJB，在大并发用户数的某些测试场景中，都超过了 100Mb/s，因此在实际的应用部署中，网络环境带宽是要考虑的一个重要因素之一。
- I 在我们所设定的测试场景，基于 2 路 Xeon 5160 2.66GHz 双核处理器的服务器表现出比 4 路基于具有超线程的 Xeon 3.33GHz 单核处理器的服务器更强的性能。因此基于 Xeon 5160 双核处理器的服务器可以为钱塘中间件平台软件的客户的实施项目带来更高的性价比。
- I 在 JTang Form 的性能压力测试中，应用服务器的磁盘利用率处于满负荷状态，应用服务器的磁盘操作较为繁忙。
- I 在 JTang Flow 的性能压力测试中，数据库服务器的磁盘利用率处于满负荷状态。JTang Flow 的性能提高的关键在数据库服务器上。
- I 在 JTang PMI 的性能压力测试中，认证授权信息存放于数据库，数据库存储过程成为认证授权数据处理的执行主体。数据库存储过程的并发执行效率决定了 JTang PMI 的并发性能。
- I 在 JTang EJB 的稳定性测试中，在大并发用户压力负荷下的长时间运行，高版本的 Sun Microsystems JVM 1.5.0\_07 的系统可靠性更好。

- I 在 JTang MQ 的性能测试中，需要将消息体保存到数据库的消息处理模式下，如点对点模式下的持久化消息处理方式，发布订阅模式下的消息持久化且用户联接可持续化的组合方式下，磁盘数据库写操作的性能对整个应用系统的性能影响很大，启动磁盘写缓存的高级特性，往往可以带来应用系统百分之几十的性能的提升。

本文件所提供之信息与 INTEL®产品相关。除非 INTEL 就此类产品的销售合同条款另有规定，INTEL 不承担任何责任，且 INTEL 对与销售和/或使用 INTEL 产品相关的问题不做任何明示或暗示的担保，包括是否符合产品特定目的，商销性或存在任何侵犯专利、版权、或其他知识产权的侵权行为。

Intel 可不给予任何通知随时变更技术参数、产品描述和计划。

Intel 公司可能就当前所提供之材料拥有相关专利或专利申请权、商标权、版权、或其他知识产权。英特尔提供本文件和其他资料的行为并不表明英特尔以明示或暗示方式，或以禁止反言或其他任何方式许可该等专利、商标、版权、或其他知识产权的使用。

Intel 产品并不意在应用于医疗、救生、维持生命、关键控制或安全系统或核设施应用等领域。

本文件和相关资料项下的数据均为模拟数据，仅做参考之用。该等数据经模拟运行架构模拟器获得。任何系统硬件或软件设计或配置的差异均可能影响实际性能。

性能测试和评估使用了特定的计算机系统和/或部件，反映了此种测试下 INTEL 产品的大致性能表现。任何系统硬件或软件设计或配置的差异均可能影响实际性能。买方应该同时咨询其它信息渠道以评估所要购买的系统或部件之性能。关于更多有关性能测试和 INTEL 产品的信息，请参考 [www.intel.com/procs/perf/limits.htm](http://www.intel.com/procs/perf/limits.htm) 或拨打 (U.S.) 1-800-628-8686 or 1-916-356-3104.

© 2006, Intel 公司版权所有