# 背景

在介绍性能压测概念与背景之前，首先解释下为什么要做性能压测。从09年的淘宝双十一大促导致多家合作银行后台系统接连宕机，到春运期间12306购票难，再到前不久聚美优品促销活动刚开始就遭秒杀。根据Amazon统计，每慢100毫秒，交易额下降1%。这些事件和统计数据为大家敲响了警钟，也客观说明了性能压测对于企业应用的重要性。

从具体的作用上讲，性能压测可以用于新系统上线支持、技术升级验证、业务峰值稳定性保障、站点容量规划以及性能瓶颈探测。

1、新系统上线支持

在新系统上线前，通过执行性能压测能够对系统的负载能力有较为清晰的认知，从而结合预估的潜在用户数量保障系统上线后的用户体验。

2、技术升级验证

在系统重构过程中，通过性能压测验证对比，可以有效验证新技术的高效性，指导系统重构。

3、业务峰值稳定性保障

在业务峰值到来前，通过充分的性能压测，确保大促活动等峰值业务稳定性，保障峰值业务不受损。

4、站点容量规划

通过性能压测实现对站点精细化的容量规划，指导分布式系统机器资源分配。

5、性能瓶颈探测

通过性能压测探测系统中的性能瓶颈点，进行针对性优化，从而提升系统性能。

综上所述，性能压测伴随着系统开发、重构、上线到优化的生命周期，因此有效的性能压测对系统的稳定性具有重要的指导意义，是系统生命周期中不可或缺的一部分。

# 概述

性能压测是通过自动化的测试工具模拟多种正常、峰值以及异常负载条件来对系统的各项性能指标进行测试。

从测试目的上性能压测又可以划分为负载测试、压力测试、并发测试、配置测试以及可靠性测试。

负载测试是测试当负载逐渐增加时，系统各项性能指标的变化情况。

压力测试是通过确定一个系统的瓶颈或者不能接受的性能点，来获得系统能提供的最大服务级别的测试。

并发测试通过模拟用户并发访问，测试多用户并发访问同一个软件、同一个模块或者数据记录时是否存在死锁等性能问题。

配置测试是通过对被测系统的软/硬件环境的调整，了解各种不同方法对软件系统的性能影响的程度，从而找到系统各项资源的最优分配原则。

可靠性测试是在给系统加载一定业务压力的情况下，使系统运行一段时间，以此检测系统是否稳定。

总的来说，性能压测是在对系统性能有一定程度了解的前提下，在确定的环境下针对压测需求进行的一种测试。

# 测试工具

## 选择

在选取合适的性能压测工具之前，我们需要先先了解执行一次完整的性能压测所需要的步骤：

1、确定性能压测目标：性能压测目标可能源于项目计划、业务方需求等

2、确定性能压测环境：为了尽可能发挥性能压测作用，性能压测环境应当尽可能同线上环境一致

3、确定性能压测通过标准：针对性能压测目标以及选取的性能压测环境，制定性能压测通过标准，对于不同于线上环境的性能压测环境，通过标准也应当适度放宽

4、设计性能压测：编排压测链路，构造性能压测数据，尽可能模拟真实的请求链路以及请求负载

5、执行性能压测：借助性能压测工具，按照设计执行性能压测

6、分析性能压测结果报告：分析解读性能压测结果报告，判定性能压测是否达到预期目标，若不满足，要基于性能压测结果报告分析原因

由上述步骤可知，一次成功的性能压测涉及到多个环节，从场景设计到施压再到分析，缺一不可。工欲善其事，必先利其器，而一款合适的性能工具意味着我们能够在尽可能短的时间内完成一次合理的性能压测，达到事半功倍的效果。

## 对比

在论述了性能压测必要性之后，如何选取性能压测工具成为一个重要的议题？本文选取了市场上主流性能压测工具：（ab）Apache Bench、LoadRunner、JMeter、阿里云PTS，并从多个方面出发分析了各个工具的优缺点，汇总后的优缺点如下表所示：



## 工具

### ab

ab是一款用来针对HTTP协议做性能压测的命令行工具，支持在本地环境发起测试请求，验证服务器的处理性能。它主要具有以下特点：

首先，作为一款开源工具，ab具有较好的扩展性，测试开发人员可以基于自身需求对其进行二次开发，同时它对HTTP协议支持度较好，比如支持设定HTTP请求头、支持Cookie以及HTTP的多种方法。

此外，使用ab时还可以通过指定性能压测产生的总请求数、并发数与压测时长控制性能压测，结合其能够输出性能压测过程中的TPS（每秒事务数）、RT（响应时延）等信息的特点，ab具有简单易上手的特点。

但ab也存在一些缺点，如无图形化界面支持，支持协议较为单一，只支持HTTP协议，缺少对HTTPS协议、WebSocket等协议的支持，对于较为复杂的性能压测场景，ab缺少链路编排、场景管理等支持，只能够对单一地址发起性能压测，此外，它的性能压测统计指标纬度较少，缺少性能压测过程中的数据统计，只能够在压测结束后获取相关的统计数据，无法实时获取系统负载等指标，难以应用于生产环境下的性能压测。

总的来说，ab作为一款命令行测试工具，适用于本地对支持HTTP协议的单一地址进行性能压测，但缺少相应的链路编排、场景管理、数据可视化等大规模性能压测基础功能，无法应用于生产环境。

### LoadRunner

LoadRunner，是一款发布于1993年11月的预测系统行为和性能的负载测试工具。通过以模拟上千万用户实施并发负载及实时性能监测的方式来确认和查找问题，LoadRunner作为一款历史悠久的商业性能压测工具，能够对整个企业架构进行测试。企业使用LoadRunner能最大限度地缩短测试时间，优化性能和加速应用系统的发布周期。LoadRunner可适用于各种体系架构的自动负载测试，能预测系统行为并评估系统性能。

LoadRunner从组件上可划分为四部分：

负载生成器：模拟用户对服务器发起请求

虚拟用户生成器：捕捉用户业务流，用于录制和生成脚本

控制器：用于提供场景设计与场景监控，能够实时监控脚本的运行情况

分析器：汇集来自各种负载生成器的日志并格式化报告，以便可视化运行结果数据和监控数据

从组件划分上可以看出 LoadRunner 对于性能压测拥有较为系统的支持，结合多个组件的功能特性，用户可以较为方便地设计复杂背景下的性能压测场景，例如结合场景设计设置虚拟用户数量、设置执行时间等，结合虚拟用户生成器实现复杂链路、场景的高效设计与编排。

此外，LoadRunner支持设置思考时间、集合点，还可以结合分析器实现压测报告统计数据、指标的可视化，助力测试人员理解性能压测结果。

但 LoadRunner 作为一款商业软件，价格较高，需要本地安装，安装过程较复杂，在实际设计执行压测时需要编写相应的脚本，对使用人员来说学习成本比较高，此外缺少监控告警等支持，性能压测过程中难以实时发现问题。

总的来说，LoadRunner 作为一款性能压测商业软件，功能较为齐全，使用者能够借助 LoadRunner 达到简单的性能压测场景编排、施压目标；但它也存在学习成本居高不下、扩展性差等缺点，此外支持的协议有限，不适合复杂的性能压测环境。

### Sysbench

### TPCC-MySQL

### JMeter

Apache JMeter是Apache组织开发的基于Java的压力测试工具。它可以用于测试静态和动态资源，例如静态文件、Java 小服务程序、CGI 脚本、Java 对象、数据库、FTP 服务器等等。另外，JMeter能够对应用程序做功能/回归测试，通过创建带有断言的脚本来验证你的程序返回了你期望的结果。为了最大限度的灵活性，JMeter允许使用正则表达式创建断言。同时JMeter支持对性能压测结果做图形分析。

JMeter 作为一款开源软件，扩展性强，具有强大的开源社区支持，社区内开发者活跃程度高，也正是在开源社区的积极发展下，JMeter 具有性能压测的诸多特性，如支持场景编排、断言设置，支持对多种资源施压，有图形化界面支持，支持脚本录制，使用人员能够较为简单的设计并发起性能压测，此外 JMeter 提供资源监控、性能压测报告生成等功能。

但在需要高负载施压的场景下，JMeter 需要部署分布式环境，部署成本比较高，在使用时，需要编写相应的脚本，而每个脚本文件只能保存一个测试用例，学习门槛居高不下的同时也不利于脚本的维护，此外它缺少监控告警等支持，在性能压测过程中使用人员难以借助 JMeter 实时发现问题。

### PTS

性能测试服务（Performance Testing Service，简称 PTS）是一个 SaaS 性能测试平台，提供场景 API 编排功能。结合阿里巴巴的自研平台和引擎，支持按需设定压测模式、压测量级、压测时间，快速发起压测，监控压测过程并生成报告等功能，同时也兼容开源工具 JMeter。

下面将从功能、性能、生态与监控四个方面展开介绍 PTS：

功能方面

PTS 提供了链路、场景编排压测报告导出的功能、，除了传统的并发模式（虚拟用户并发），PTS也支持 RPS 模式（Requests per Second），也即吞吐量模式，RPS 模式为 PTS 独有，具有能够更精准地衡量服务端系统能力等优点。为了降低发起性能压测的门槛，PTS 提供云端录制器，便于客户端的请求抓取，同时还可将抓取的请求一键导入到压测场景中；为了适配不同场景下的性能压测，PTS 支持创建服务等级协议 SLA（Service Level Agreement）规则，能够实现对业务压测场景更智能的控制和更全面合理的评价，同时，PTS 也提供了大量 SLA 模板供不同背景下的用户使用；此外，PTS 还支持定时压测，能够指定启动压测的日期、时间以及循环周期等，能够在任意时间段自由发起性能压测，释放人力。

性能方面

PTS 能够随机调度遍布全国各地的压测引擎，一分钟内快速启动性能压测，模拟真实环境下的用户请求；支持最高千万级的流量瞬时脉冲，多重机制确保压测流量及时停止；支持两种调速模式：自动递增和手动调整，压测流量调整秒级生效。

生态方面

PTS 支持添加阿里云生态内的云监控产品，如添加阿里云生态内的性能管理类产品ARMS，提供应用级别的监控，为性能压测提供问题定位的闭环能力；此外 PTS 云端集成 JMeter，用户只需在本地完成 JMeter 脚本调试，即可在 PTS 上快速发起压测。

监控方面

PTS 监控指标包括每个 API 的并发，RPS (Requests per Second)、响应时间、采样的日志等。同时从不同细分维度，统计了 API 请求的成功、失败情况和响应时间，能够帮助用户快速定位到系统的性能瓶颈。此外，PTS还能够结合阿里云生态内的云产品监控，如监控ECS、SLB及RDS等在内的各产品性能指标；为云上服务提供更为详尽的监控。

总的来说，阿里云 PTS 作为一款云服务，用户可以较低的学习成本快速借助 PTS 发器压测，对于阿里云的用户来说，PTS 能够紧密结合现有的阿里云服务，提供全方位的压测报告供用户快速定位性能瓶颈；对于 JMeter 用户，也能够以较低的成本迁移至 PTS，享受 PTS 的高阶功能。但 PTS 也存在一些问题，扩展性需要加强，例如需要支持更多网络协议。