



IBM Software Group

IBM性能测试解决方案

Rational. software

IBM 软件部 程燕宾



ON DEMAND BUSINESS™

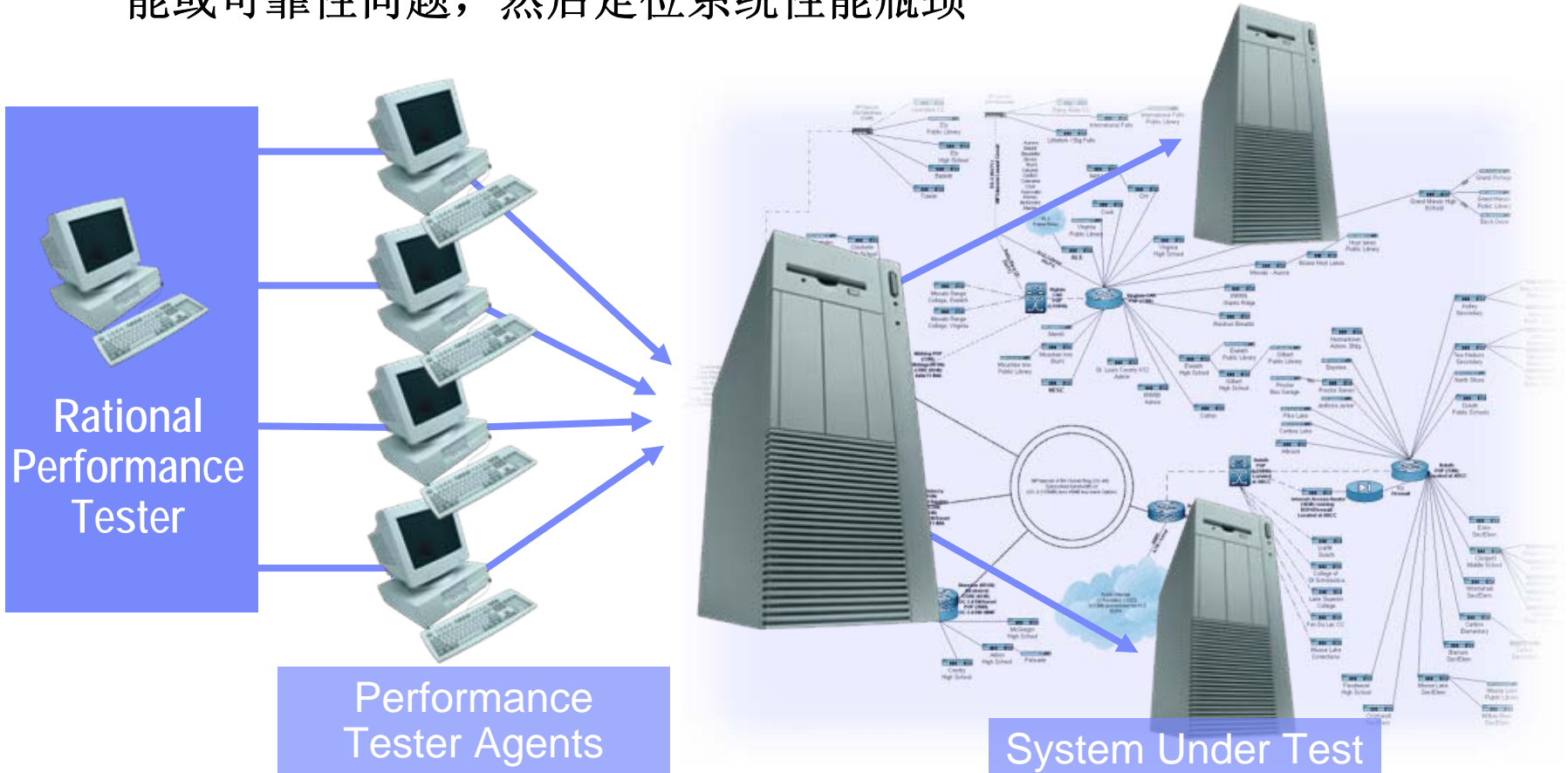
议程

- 性能测试综述
- 如何进行性能测试
- IBM性能测试解决方案
- 总结



什么是性能测试?

- 利用负载生成工具模拟实际用户访问系统, 从而发现应用或系统性能或可靠性问题, 然后定位系统性能瓶颈



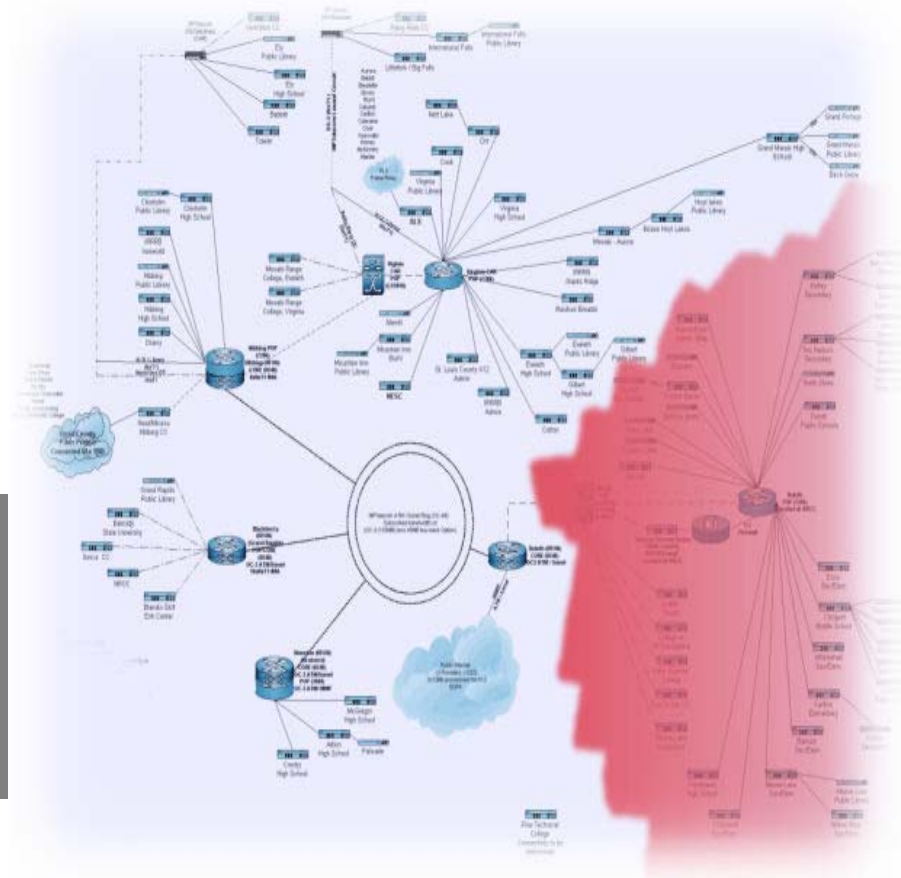
性能测试目标

- 确定应用的响应时间
- 确定系统将支持多少用户
- 确定最佳系统配置
- 找出系统在较重负载情况下的潜在问题



为什么做性能测试

- 通过收集应用性能数据，检验对业务的可能影响
- 从用户角度提高质量
 - ▶ 关键性能属性：应用相应时间和速度
- 尽早发现主要应用缺陷和架构问题，降低风险



每缺陷成本

1	10	100
设计	测试	产品

性能测试类型

- 负载测试
- 容量测试
- 压力测试
- 疲劳测试
- 配置测试
- 并发测试



议程

- 性能测试综述
- 如何进行性能测试
- IBM性能测试解决方案
- 总结

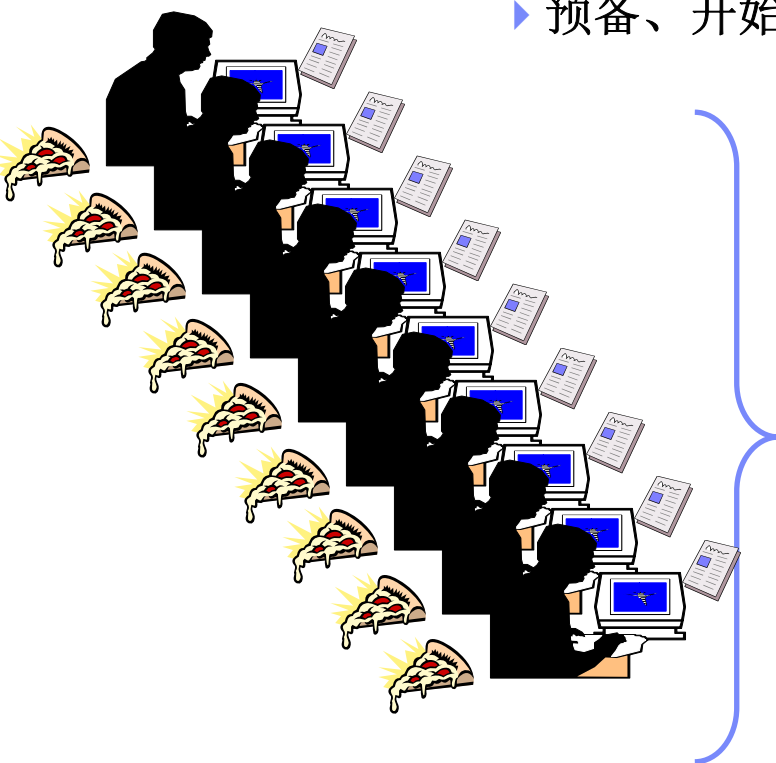


手工性能测试

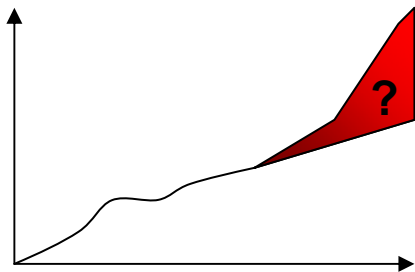
- 手工进行
 - ▶ 找**50**个人
 - ▶ 提供操作步骤
 - ▶ 预备、开始☺



被测系统



手工性能测试的问题



- 不可扩展
 - ▶ 如果支持**100**，**1000**，**10000**用户，怎么办？
 - ▶ 主观推断不可取



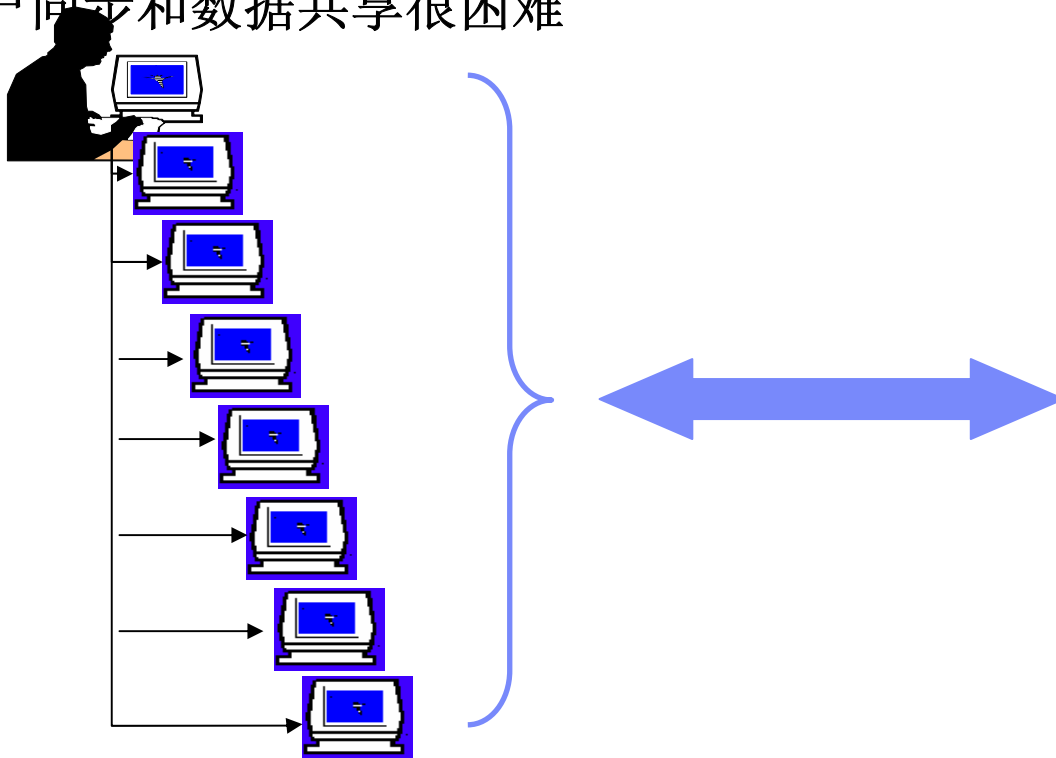
- 不精确
 - ▶ 不同人有不同的访问系统习惯：生手和熟手



- 不可再现
 - ▶ 不能期望每次都加相同的负载

基于硬件的测试

- 可扩展性仍是一个问题
 - ▶ 一个模拟用户需要一台物理计算机
- 用户同步和数据共享很困难

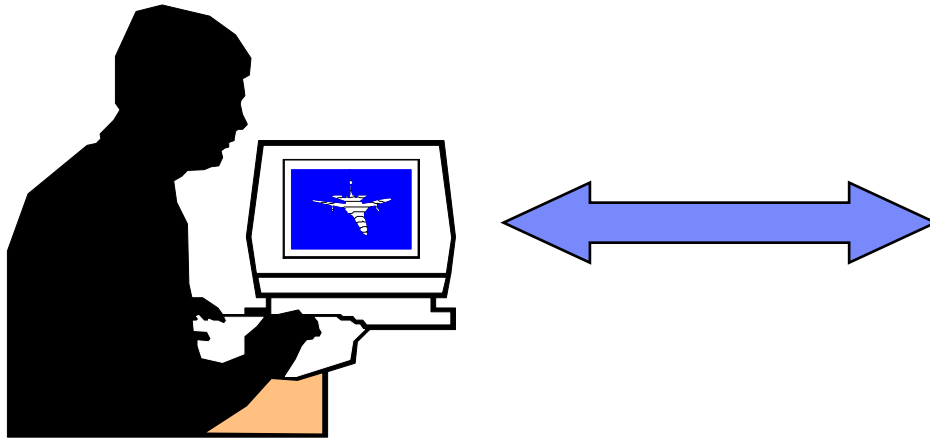


被测系统



基于软件的测试

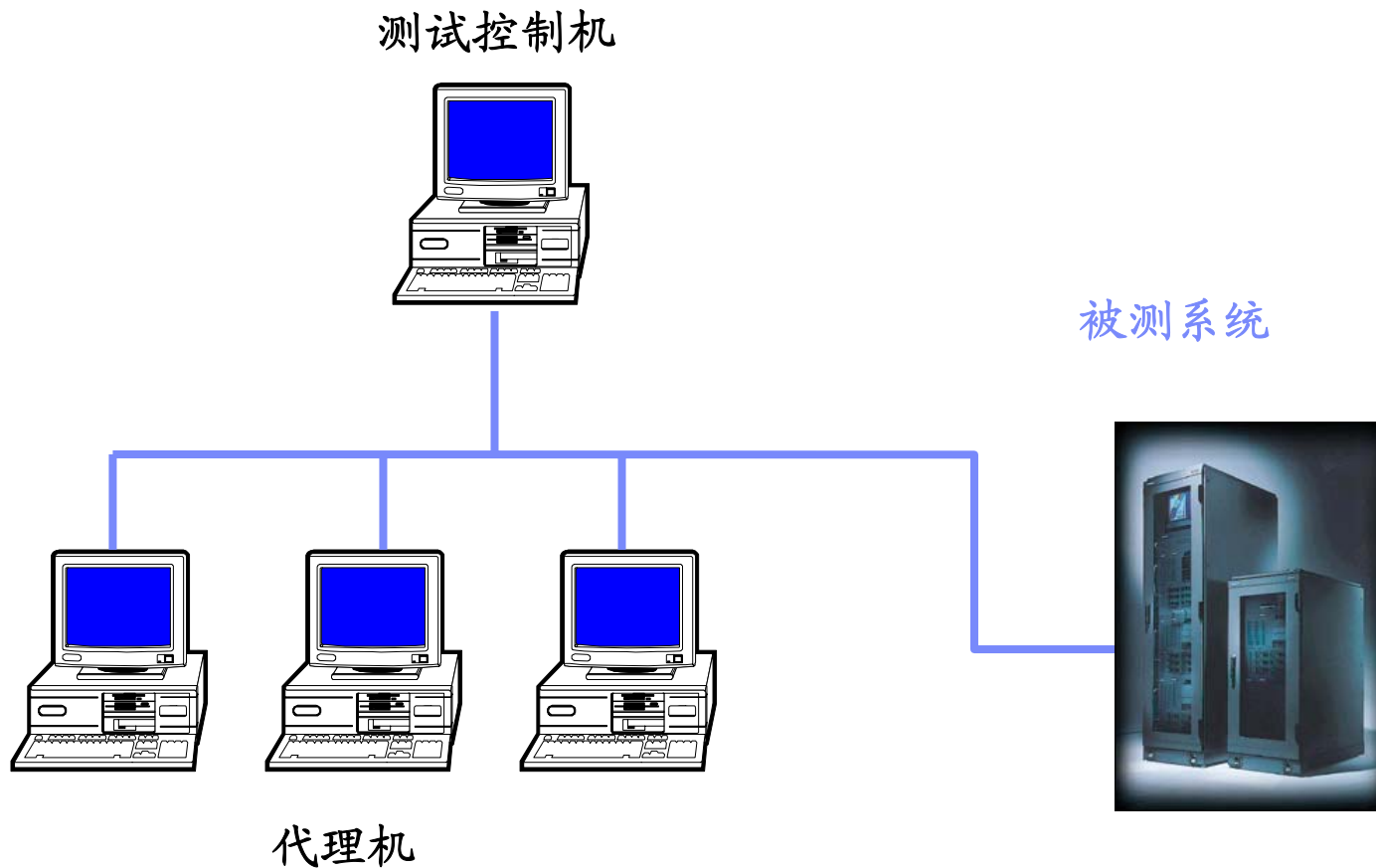
- 同步和数据共享不是问题
- 增加用户数受到测试机的性能限制



被测系统



软、硬结合进行性能测试



性能测试框架（1）

- 测试计划和设计
 - ▶ 场景分析
 - ▶ 性能测试计划
- 测试开发和确认
 - ▶ 记录事务
 - ▶ 修改测试
 - ▶ 设计场景
 - ▶ 确认并调试测试



性能测试框架（2）

- 执行测试并分析
 - ▶ 重置系统和数据状态
 - ▶ 运行监控工具
 - ▶ 运行测试
 - ▶ 分析结果
- 安装测试环境
 - ▶ 测试试验环境
 - ▶ 系统软件
 - ▶ 数据集
 - ▶ 测试和监控工具



议程

- 性能测试综述
- 如何进行性能测试
- IBM性能测试解决方案
- 总结



Rational 软件质量保证解决方案

业务

软件质量保证解决方案

测试管理

需求

- 捕获需求
- 描述用例

RequisitePro

测试

- 管理测试活动
- 测试和业务需求关联

变更

- 跟踪变更
- 变更统计分析

缺陷

- 缺陷跟踪
- 缺陷统计分析

ClearQuest

ClearCase

测试执行

开发测试

Test RealTime
Application
Developer
PurifyPlus

功能测试

自动

Functional Tester
Robot

手工

- 完成业务测试
- Manual Tester

性能测试

模拟用户负载

Performance
Tester
Robot

质量度量

测试进度报告

产品质量报告

团队绩效报告

开发

测试



用 IBM Rational Performance Tester 进行性能测试 初学者和专业人员的自动化测试工具



- **IBM Rational Performance Tester**
 - ▶ 基于WEB系统的测试
- **主要优点**
 - ▶ 保证工作生产力
 - 隐藏复杂度使工作完成
 - ▶ 先进的数据访问和处理
 - 数据变量及其同步
 - ▶ 低成本的性能测试工具
 - 低资源需要
 - 集成桌面开发工具

创建一个性能测试

创建一个性能测试只需要3个步骤



建立脚本

- 脚本创建时需要考虑的因素
 - ▶ 可视化测试编辑器,变量化输入数据,关联服务器响应

创建一个性能测试

创建一个性能测试只需要3个步骤



建立脚本

制定工作负载(Workload)

- 脚本创建时需要考虑的因素
 - ▶ 可视化测试编辑器,变量化输入数据,关联服务器响应
- 工作负载的考虑因素
 - ▶ 精确描述用户的真实工作负载



创建一个性能测试

创建一个性能测试只需要3个步骤



建立脚本

制定工作负载(Workload)

执行和测试

- 脚本创建时需要考虑的因素
 - ▶ 可视化测试编辑器,变量化输入数据,关联服务器响应
- 工作负载的考虑因素
 - ▶ 精确描述用户的真实工作负载
- 执行和分析的考虑因素
 - ▶ 验证响应&找到系统瓶颈



创建一个性能测试 建立脚本

- 捕获用户和服务器的交互来创建测试
- 创建的测试结果在树视图(**tree view**)中显示为一系列访问过的页面
- 创建脚本时的关键考虑点
 - ▶ 消除编程的需要
 - ▶ 不同的用户,不同的输入数据
 - ▶ 关联系统响应数据



没有代码的测试



- ▶ 测试在树视图(tree view)中显示为一系列访问过的页面以及他们包含的元素,如图等.
- ▶ 不需要编程就可以创建一个测试.
- ▶ 为了简化测试的创建和维护,底层细节缺省为隐藏,但是高级用户可以很容易就访问到

Performance Test - RoomReservation

Test Contents

This section shows the test contents

- RoomReservation
 - Whistler Blackcomb - Ski Resort Website - Whistler, Canada
 - Guest Services
 - Room Reservations
 - Specify Room Details
 - Blackcomb Lodge - Room Details
 - My Reservations
 - Reservation Confirmation Details

Buttons: Add, Insert, Remove, Up, Down



创建一个性能测试 建立脚本



- 自动创建参数化的测试脚本
 - ▶ 记录时,可变值被变量所代替
 - ▶ 回放时,变量数据可以从数据池加载
 - ▶ 回放数据可以从外步数据源导入
 - ▶ 不需要用户编码



创建一个性能测试 建立脚本



- 自动关联服务器响应值
 - ▶ 在录制时捕获的服务器响应被替换为变量
 - ▶ 回放时,变量被从服务器返回的数据所替代
 - ▶ 每个用户根据他们的活动收到唯一的数据
 - ▶ 不需要用户写任何代码



创建一个性能测试

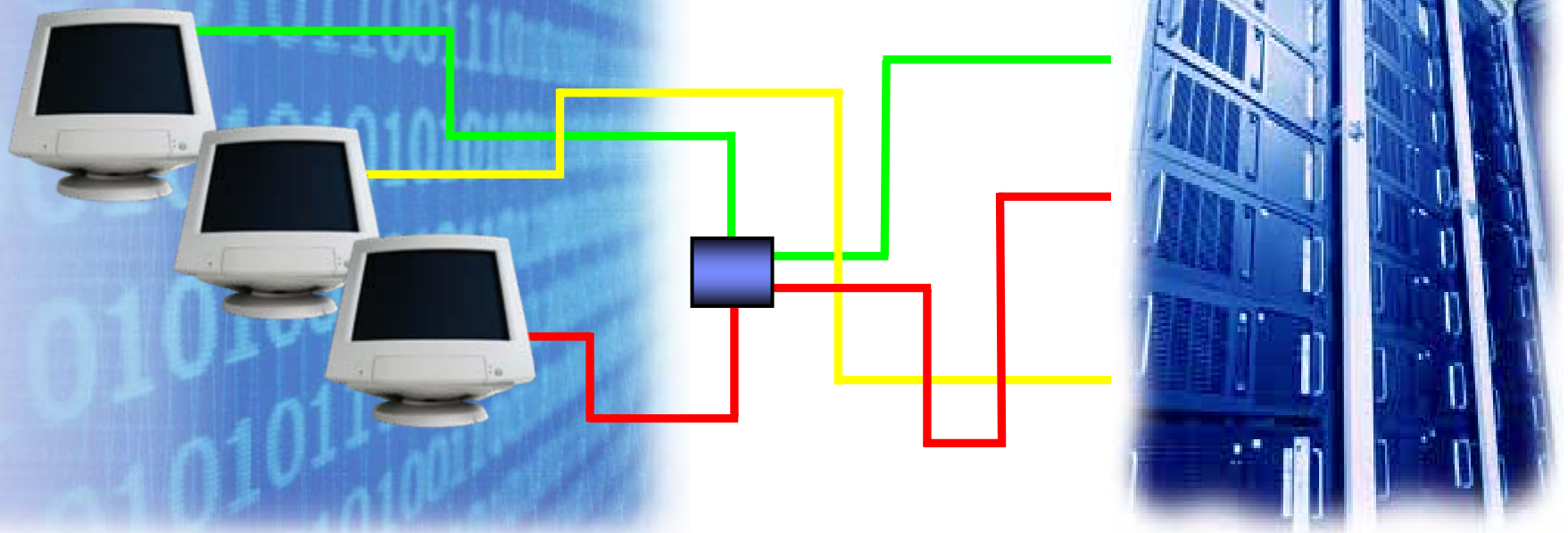
制定工作负载

- 测试被组装起来以执行
 - ▶ 进度表定义了脚本执行的顺序
 - ▶ 进度表定义了被用来生成负载的代理机器
 - Linux 和 Windows 代理
- 创建工作负载时的关键考虑:
 - ▶ 可以灵活的来精确建模复杂的工作负载
 - ▶ 在运行时可以动态的增加负载来找到临界点



创建一个性能测试

制定工作负载



- 强大并且灵活的工作负载的制定
 - ▶ 不需要代码的可视化工作负载制定编辑器
 - ▶ 协调时序和依赖
 - ▶ 精确的建模真实用户工作负载
 - ▶ 在运行时动态的增加负载

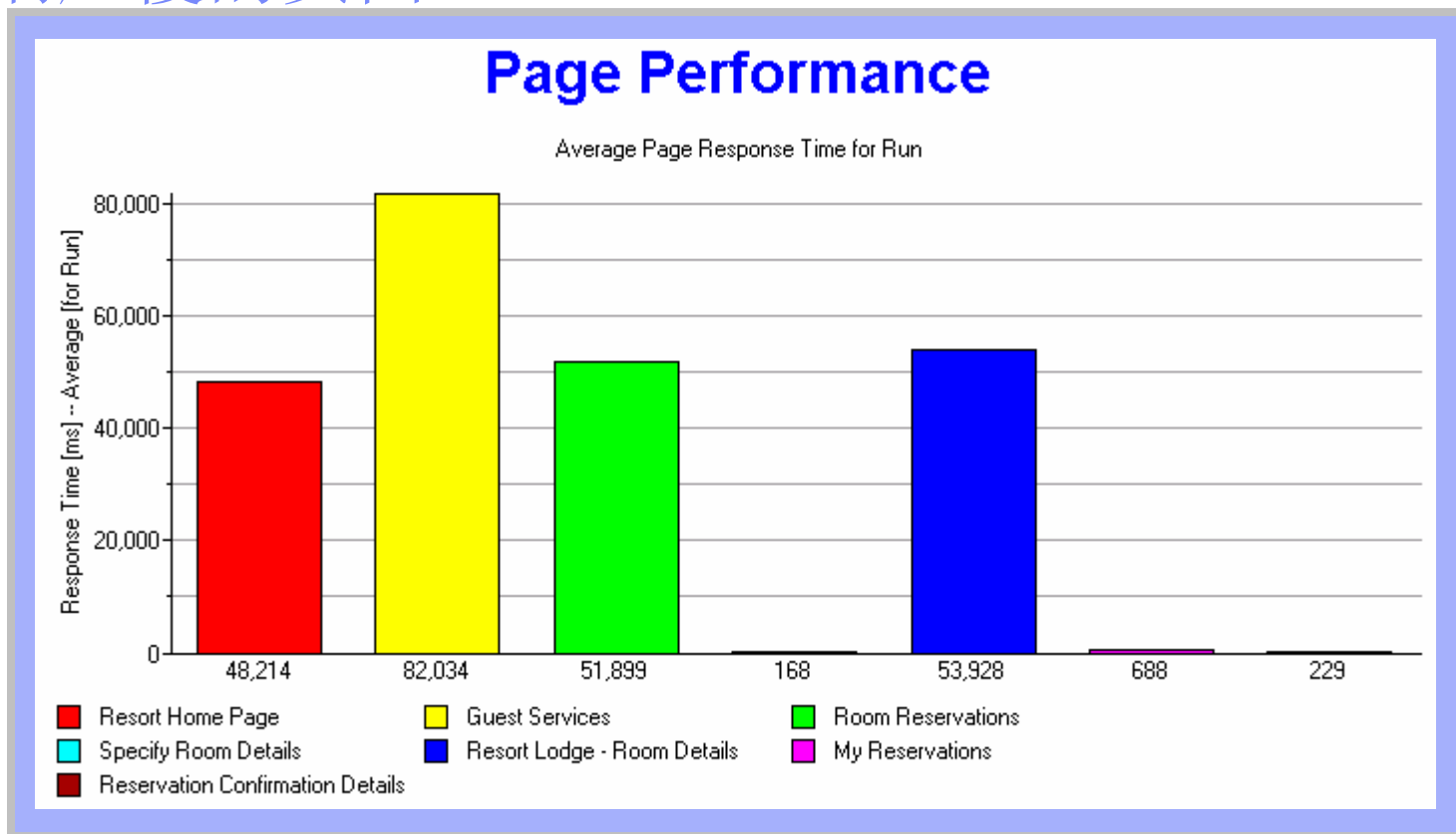
创建一个性能测试

执行和分析

- 中央控制台来执行分布式的测试
 - ▶ 主控制台可以协调和监控所有的活动
 - ▶ 可以用Windows和Linux的代理来生成负载
- 分析时的关键考虑
 - ▶ 找到响应速度慢的页面
 - ▶ 找到资源瓶颈

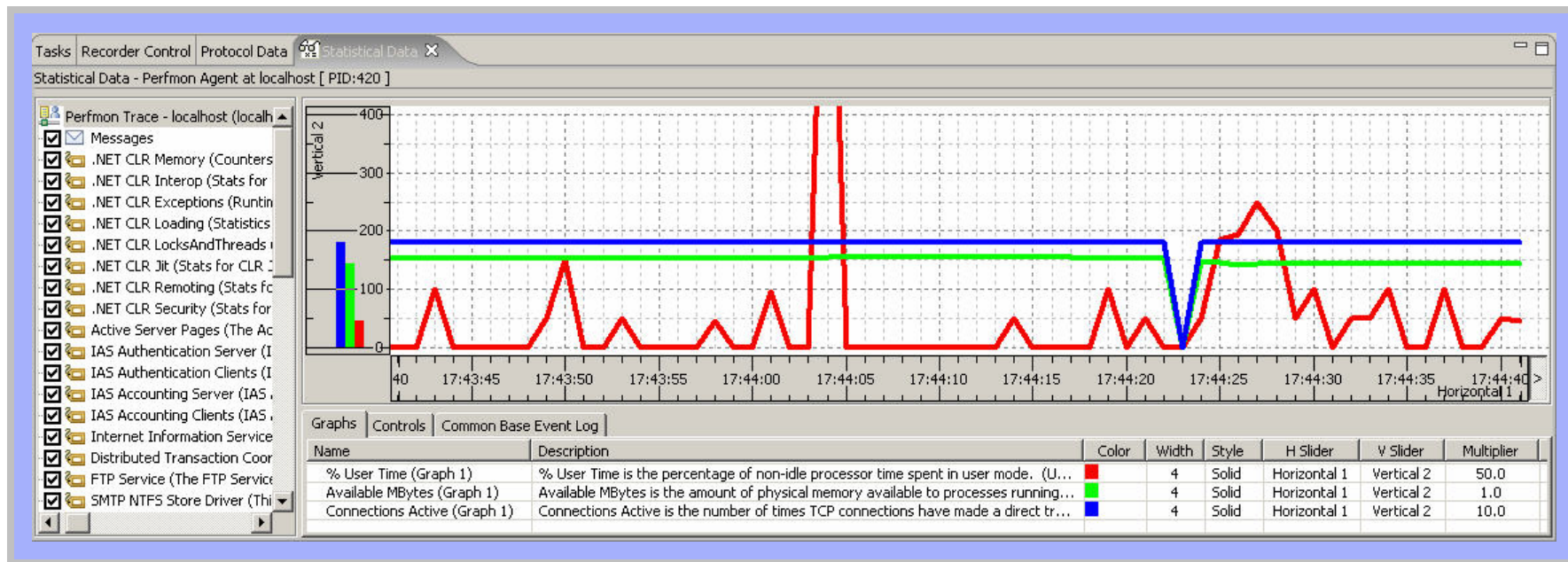


找到响应慢的页面



- 很容易就能定位站点中响应慢的页面
 - ▶ 在执行过程中实时的报告以及执行结束后运行分析

确定资源瓶颈



- 测试执行过程中实时的资源利用率的监控
 - ▶ 捕获并分析性能监控统计
 - ▶ 确定资源瓶颈



议程

- 性能测试综述
- 如何进行性能测试
- IBM性能测试解决方案
- 总结



Rational 软件质量保证解决方案

业务

软件质量保证解决方案

测试管理

需求

- 捕获需求
- 描述用例
- 量化需求

测试

- 管理测试活动
- 测试和业务需求关联
- 跟踪测试进度和变更
- 监控质量

变更

- 跟踪变更
- 变更统计分析

缺陷

- 缺陷跟踪
- 缺陷统计分析

测试资产管理

测试执行

开发测试

- 静态分析
- 内存分析
- 性能分析
- 代码覆盖
- 组件测试

功能测试

自动

- 脚本自动化生成
- 自动执行测试

手工

- 完成业务测试
- 测试脚本模块化

性能测试

- 模拟用户负载
- 发现并定位性能问题
- 优化系统配置

质量度量

测试进度报告

产品质量报告

团队绩效报告

开发

测试



性能测试的关键特性

- 保证工作生产力
 - ▶ 创建没有代码的测试
 - ▶ 隐藏复杂度使工作完成

- 先进的数据访问和处理
 - ▶ 创建没有代码的工作负载计划
 - ▶ 数据变量及同步

- 低成本的性能测试
 - ▶ 低资源需要
 - ▶ 桌面开发环境集成



下一步

IBM

Home | Products & services | Support & downloads | My account

Software > Rational >

Software quality

Test smarter, finish faster. Improve testing scope and depth while shortening test time with Rational testing products, online tester resources and services.

Ease the team into test automation by first using IBM Rational® TestManager to manage manual testing and reporting, and IBM Rational® Robot to automate regression tests. Take advantage of integrated defect tracking, test management and requirements traceability in the IBM Rational® Team Unifying Framework to uncover more defects, earlier, when they are least expensive to fix. Enhance testers' automation skills on-the-job via the Rational Developer University, Rational University web-based courses, and the browser-based Rational Process.

Rational offers the following products for automation:

- [IBM Rational Functional Tester for script maintenance by creating tests with ScriptAssure™.](#)
- [IBM Rational Functional Tester Applications: End-to-end test automation.](#)
- [IBM Rational Functional Tester for test automation.](#)

Related links:

- DB2 Information Management
- Lotus
- Tivoli
- WebSphere
- Warranty Info

■ 联系您的IBM代表

▶ <http://www.ibm.com/contact/cn>

■ 在 IBM.com 上的产品信息

▶ www.ibm.com/software/rational

▶ 成功案例

▶ 评估计划: Trials, Demos, Betas

▶ Events: Webinars, Seminars, Conferences & Trade Shows

▶ Literature, Technical Resources & Support



THANK YOU

