

HP Performance Engineering 最佳实践系列

适用于性能工程师和经理

Performance Center of Excellence 最佳实践

文档发布日期：2014 年 12 月

软件发布日期：2014 年 12 月



法律声明

担保

HP 产品和服务的唯一担保已在此类产品和服务随附的明示担保声明中提出。此处的任何内容均不构成额外担保。HP 不会为此处出现的技术或编辑错误或遗漏承担任何责任。

此处所含信息如有更改，恕不另行通知。

受限权利声明

机密计算机软件。必须拥有 HP 授予的有效许可证，方可拥有、使用或复制本软件。按照 FAR 12.211 和 12.212，并根据供应商的标准商业许可的规定，商业计算机软件、计算机软件文档与商品技术数据授权给美国政府使用。

版权声明

© Copyright 2010 - 2011 Hewlett-Packard Development Company, L.P.

商标声明

Microsoft® 和 Windows® 是 Microsoft Corporation 在美国注册的商标。

Oracle 是 Oracle 和/或其子公司的注册商标。

文档更新

此文档的标题页包含以下标识信息：

- 软件版本号，用于指示软件版本。
 - 句点前面的数字标识主要发布号。
 - 句点后面的第一个数字标识次要发布号。
 - 句点后面的第二个数字标识次要的下一级发布号。
- 文档发布日期，该日期将在每次更新文档时更改。
- 软件发布日期，表示此版本软件的发布日期。

要检查是否有最新更新或验证所使用的文档是否为最新版本，请访问以下 URL：

<http://h20230.www2.hp.com/selfsolve/manuals>

此站点需要注册 HP Passport 才能登录。要注册 HP Passport ID，请访问：

<http://h20229.www2.hp.com/passport-registration.html>

或单击“HP Passport”登录页面上的“New users - please register”链接。

此外，如果订阅了相应的产品支持服务，则还会收到更新的版本或新版本。有关详细信息，请与您的 HP 销售代表联系。

支持

可以在以下地址访问 HP 软件支持网站：

www.hp.com/go/hpsoftwaresupport

此网站提供了联系信息，以及有关 HP 软件提供的产品、服务和支持的详细信息。

HP 软件联机支持提供客户自助解决功能。通过该联机支持，可快速高效地访问用于管理业务的各种交互式技术支持工具。作为尊贵的支持客户，您可以通过该支持网站获得下列支持：

- 搜索感兴趣的知识文档
- 提交并跟踪支持案例和改进请求
- 下载软件修补程序
- 管理支持合同
- 查找 HP 支持联系人
- 查看有关可用服务的信息
- 参与其他软件客户的讨论
- 研究和注册软件培训

大多数提供支持的区域都要求您注册为 HP Passport 用户再登录，很多区域还要求用户提供支持合同。要查找有关支持访问级别的详细信息，请访问以下 URL：

http://h20230.www2.hp.com/new_access_levels.jsp

要注册 HP passport ID，请访问以下 URL：

<http://h20229.www2.hp.com/passport-registration.html>

目录

简介	7
关于 HP Center of Excellence.....	7
读者	9
先决条件	9
结构	10
反馈	10
1 Performance CoE 简介	11
Performance Engineering 的重要性	11
为什么选择 CoE?	13
为什么有性能测试还不够	13
为什么不能落后于人	14
什么是 CoE?	15
CoE 阶段	17
组织成熟度	18
手动测试	18
项目测试	19
标准化	19
集中化	20
性能主导	20
2 构建成功 CoE	22
两种构建 CoE 的方法	24
自上而下	24
自下而上	25
结果	26
组织评估套件	27
概念论证	30
沟通	32
工具标准化	34
3 成功操作 CoE	36
项目计划	38
性能测试战略	38
性能测试计划	38

目标	39
范围	40
方法	40
假设	41
依赖关系	41
风险	41
计划	41
测试案例	42
制成品	43
缺陷	43
状态报告	44
测试执行报告	45
定期报告	46
摘要报告	46
4 招集 CoE 人员	48
角色	48
CoE 经理	49
解决方案架构师	50
测试实践主管	51
测试自动化工程师	51
业务分析师	52
运营经理	52
测试实验室专家	53
负载测试技能	54
应用程序知识	56
5 结论	58
A PCoE 调查表	60

欢迎使用本指南

简介

欢迎使用《HP Performance Center of Excellence 最佳实践》(HP Performance Center of Excellence Best Practices)。

本指南提供概念、准则以及各种组织在实施 Performance Centers of Excellence 过程中的实践示例。

关于 HP Center of Excellence

HP 是自动化性能测试的市场引领者。自动化性能测试是利用产品、人员和过程来最大程度地降低部署应用程序、升级或修补程序期间可能遇到的风险的一个专业领域。其核心功能是将生产工作负载应用于预部署系统，同时度量系统性能和最终用户体验。如欲构建结构合理的性能测试，请回答如下问题：

- 应用程序能够为预期的用户作出足够快速的响应吗？
- 应用程序可以处理预期的和更大的用户负载吗？
- 应用程序可以处理业务所需的事务数吗？
- 应用程序在预期和非预期的用户负载下都稳定吗？
- 确定上线当天用户会有良好的体验吗？

回答这些问题后，自动性能测试即可量化业务方面的变更带来的影响。进而明确地预测部署风险。有效的自动化性能测试过程帮助您进行更成熟的发布决策，防止出现系统停机和可用性问题。

实现自动化性能测试价值最大化的方法之一是建立内部 **Center of Excellence**。很多 IT 组织都已采用 **Center of Excellence (CoE)**模型作为一贯地、持续改进 IT 运行的可行办法。在 CoE 模型中，QA 基础结构、过程和最佳实践相合并并集中化。这为应用程序交付、领导作用、咨询和支持服务提供了管理和自动化平台，以帮助公司优化应用程序质量。

利用 CoE 还可使整个组织可以看到标准化度量和关键性能指标(KPI)。这有助于让所有项目干系人了解相关信息，并使应用程序保持与业务目标一致。

HP Performance Center 是允许本机实施 Performance CoE 的自动化性能测试产品。它是一种安装在企业的自有基础结构之上的企业全局性能测试工具。

- Performance Center 支持跨不同地理地域管理多个并发负载测试项目，无需在两地间穿行。
- Performance Center 管理所有内部负载测试需要。
- 使用 Performance Center，可以从可通过 Web 访问的中心位置管理大型负载测试项目的所有方面，包括资源分配和计划。
- Performance Center 可帮助简化测试过程，减少资源成本，提高运行效率。
- Performance Center 可帮助查明性能瓶颈。
- Performance Center 允许您确定测试中的应用程序可以扩展到的最大用户数。此信息为您提供线索，以找到增加应用程序负载容量的应对措施。

编写本书的目的是为了帮助 IT 组织评估其当前状况，成功地建立并维护自己的 Performance Center of Excellence。这些准则基于运用真实的最佳实践数据和各种来源的专业知识所做的深入研究，包括：

- HP 操作系统管理员
- HP 专业服务组织
- 由行业专家编写的技术文档和书籍
- 数家客户测试组织

使用这些准则有助于使初始创建过程生效，有助于获得 CoE 的最大价值。

读者

本指南的目标读者是：

- Performance CoE 经理
- 性能工程师
- QA 经理

先决条件

在使用本书之前，应当从性能工程师或开发者角度理解性能验证字段。最好具备软件开发生命周期（Software Development Life Cycle, SDLC）主要阶段的知识。还应当熟悉实际 IT 组织中的典型业务流程。

结构

本指南的组织方式如下：

- [欢迎使用本指南](#)
- [Performance CoE 简介](#)
- [构建成功 CoE](#)
- [成功操作 CoE](#)
- [招集 CoE 人员](#)
- [结论](#)
- [附录 — PCoE 约谈的调查表](#)

注：无论“CoE”一词用在哪里，都可认为它是用于 Performance Center of Excellence 上下文中。因此，整本书中既用到了 CoE 也用到了 PCoE。

反馈

如有问题、意见或宝贵的最佳实践信息要共享，请将邮件发送至以下电子邮件地址：

lt_cust_feedback@hp.com

1 Performance CoE 简介

Performance Center of Excellence 既是成熟 IT 组织的一种方法，也是发展形成的一个实体。虽然有很多方式可以执行性能测试，但客户的体验证明建立基于过程的中心组织，可以显著改进业务结果并节省时间。

Performance Engineering 的重要性

软件工程已经定义为用于控制软件开发过程并为以生产性方式构建高质量软件提供基础的过程、方法和工具。“软件质量”有很多种衡量标准，包括但不限于，功能性、易用性、灵活性、可扩展性、安全性和性能。很多软件工程方法都侧重于在确保软件达到功能要求的同时，又能按照严格的进度和预算进行开发。

但是，性能要求在变得愈加难以管理的同时又显得越发必要。进入到部署于分布式系统的客户端/服务器软件体系结构时，基于排队理论的传统分析就显得过时了。这主要是由于有了 Internet 服务后信息是基于“订阅”而不是“事务”发送给用户的。正如软件和计算机系统的性质发生了变化，用户社区的性质也发生了变化。

在很多情况下，系统性能在系统生命周期中都评估得太晚。最糟糕的情况是，致电时才发现性能问题，发怒的用户或客户抱怨着响应时间。稍好些的情形是，组织在部署前可能着手创建基准。但是，在部署前的阶段，应用程序已经完成了设计、编码和测试。因此，这样会增加计算和沟通资源的支出。

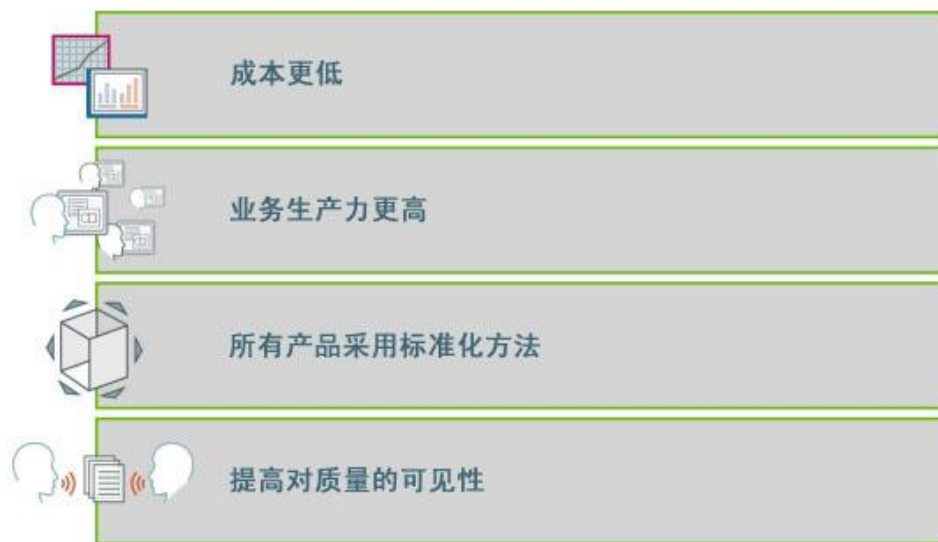
交付高质量应用程序的压力在持续上升，同时由于开发和部署计划压缩、组织地理上分散、外包服务以及熟练工的高速流动，使得应用程序测试更具挑战。即使一个业务单位有合适的技能，它的成果也不会运用于整个组织。很多公司的 IT 部门都面临着无共享、无通用过程和无性能意识的共同难题。

面对必须提高效率、同时应付多个项目和管理分散的不同项目团队的现实，很多组织正在采用测试管理方法，求助于自动化测试管理工具来帮助集中、组织和记录测试工作量和安排它们的优先级。

本文档探讨了测试管理的挑战和回报，并提供了可行方法来帮助您实现有组织、结构化的测试过程。

颇受关键客户和分析师等欢迎的方法之一是 Performance Center of Excellence (CoE)模型，它本质上是一个为改进质量、一致性、有效性和效率而在整个组织中推进标准化和过程的逻辑或物理实体。

CoE 业务价值



Performance CoE 无需使用不同工具，杜绝了效率低下的性能测试和不准确的分析。Performance CoE 还为整个组织提供性能管理的中心平台，使得交付的应用程序的关键性能参数可见，让每个人都能了解相关信息，并保持应用程序与业务目标一致。

为什么选择 CoE?

对此不太好回答的问题的简单答复是，Center of Excellence 会促成有价值的结果。许多分析师在报告中将 CoE 的重要性归结于显著提高组织符合或超过中心支持的目标的能力。当整个组织有管理权、支持结构、准则、度量和共享信息时，成功的几率要高得多。成功组织可以关注特定项目目标。想要取得成果应当是创建任何 Center of Excellence 的首要动机。没有它，组织无法施展其全部潜能。

为什么有性能测试还不够

性能测试是一组专门的技能。它需要应用程序、硬件和第三方系统的知识。它涉及从体系结构立场了解应用程序行为、建模技能和系统方法。并非所有测试人员都具备这些专业素质，它需要花数年的时间才能开发出一组合适的技能。如果性能测试人员呆在单个项目测试组中，则很难保证所有应用程序都按相同的标准进行正确的性能测试。而且，出现新技术时，这些完全不同的组并非全都有必需的专业知识来支持这些全新的技术。

在很多情况下，每个项目团队使用各成员恰巧喜欢的工具开发其自己的性能优化过程。结果造成各种未妥善记录、未正确集成和不可传输到其他组的工作流、实践、工具及技术混合在一起，无法管理。这不仅导致效率底下，还会大大增加成本和实践的复杂性，有可能影响士气和制约 IT 员工的技能培养。开发和交付过程的不同团队使用各自的性能优化方法，导致沟通失败、不一致，最终限制了其最佳性能。这种拼凑式的测试方式拆分 IT 和业务管理中的各个项目团队，模糊了项目目标。

谈到计划生产部署，很多公司都在部署过程的后期才创建系统基准。因此，系统是在次优配置中部署的，即仍有未发现的性能问题和可用性问题，且用于增强性能的不同工具集无法一起运行。

为什么不能落后于人

应用程序性能通常一直处于被忽视状态，因为它往往不是由任何一个业务团队所拥有，而事实是：

无论应用程序在功能上设计得如何完美，多么符合业务要求，如果其性能缓慢或不可靠，实际上它们对最终用户而言都是毫无用处的。

因为遗憾的是，用户更可能记住对公司最差的体验而不是最好的体验。因此，性能测试是任何部署的至关重要的组成部分，因为应用程序性能直接影响最终用户体验，而这一点又直接影响收入流。性能测试会在使客户满意和转向竞争对手之间作出区分。

纠正性能问题需要延长时间和额外增加工作量，因此性能问题常常严重影响用户体验。性能问题往往不是一夜之间就能解决的，性能修复通常需要彻底中断服务。

增强应用程序的速度和可用性可能很复杂，总会需要专业知识，而这些知识又需要花时间去获取。同时，每个阶段的应用程序性能优化由于平台复杂性加大（如 SOA、虚拟化、Web 2.0 及其他 IT 趋势）而更加专业化。这些复杂平台具有自己的特定技能集、工具、格式、协议和过程。通常，培养的技能以及各个项目团队为负载测试、代码优化、使用情况分析、调优和产能计划创建的过程，在 IT 部门内或各类业务之间得不到共享。

该问题与业务和 IT 预算限制以及持续业务变更（比如资产剥离、兼并收购和外包）堆积在一起。现代公司无法继续维持“一切如常”的景象，而它们在整个公司内分散资源和资金，充其量是在重复工作。

Internet 的迅速发展以及大多数最终用户熟悉 Web 应用程序性能这一事实，都对 IT 部门改进公司内部的用户体验施加了巨大压力。与过去相比，客户更不可能等待事务完成，甚至更复杂的事务也是如此。他们要求您严格遵守 SLA，其中通常包括预期性能条款。不令人满意的性能阈值导致客户满意度底下，甚至可能还会使用户放弃某些应用程序。

性能测试通过度量应用程序和体系结构的能力，在问题暴露给用户社区之前符合客户的性能预期，从而转移性能风险。性能测试不是最好有，而是必须有。

什么是 CoE？

Center of Excellence (CoE)一词还没有完全被公司接受，一些公司喜欢称之为“能力中心”、“集中测试”、“应用程序测试中心”或“测试组”。但是，Center of Excellence 一词在业界最受欢迎，而且不仅仅在性能测试领域，因此本书通篇都使用 Center of Excellence。

行业分析公司 **voke** 在其研究中引用了以下 PCoE 定义：

Performance CoE 是关注应用程序性能增强和验证以满足业务要求的内部能力中心。它为性能优化过程以及咨询服务、支持服务、领导和拥护作用提供管理和自动化平台，以帮助组织了解 CoE 模型性能验证和优化的重要性。通过 Performance CoE，项目团队能够利用 CoE 开发的所有专业技术、工具集和最佳实践。

此模型的优点是：

- 集中型基础结构和专业技术

测试实验室、工具、员工和最佳实践相整合，所有项目团队均可从单个源方便地进行访问，不必重复投入昂贵的资源（实际上，可以减少总劳动力和系统许可证）。随着中心团队开发的知识越来越多，整个组织将受益匪浅。中心团队促进更高的测试质量，并改进运行和分析这些测试的效率。

- 更快更好的过程

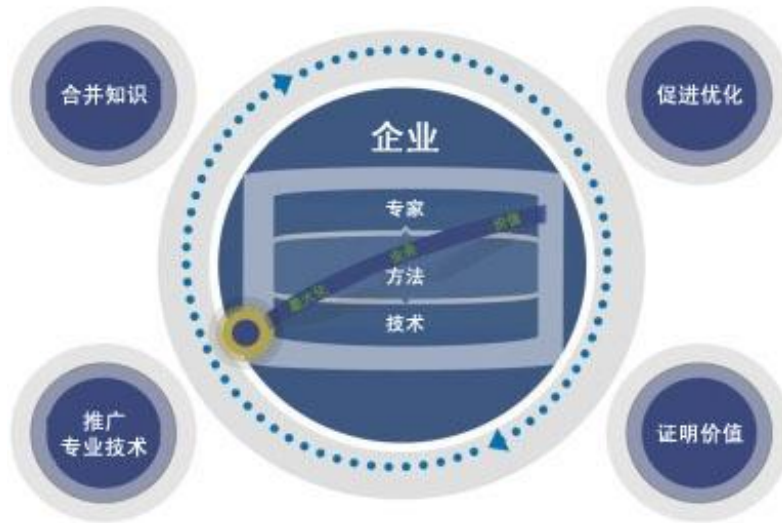
标准化工具集和过程支持一致、经济且快速地实施增强的过程。不必维护各台测试计算机或各个测试组中的所有脚本、监控器配置文件和场景，您可以将所有数据集中在一起使共享和重用更容易。它缩短创建测试的时间，提高了测试效率。

- 专注于业务

Performance CoE 模型从业务和最终用户结果的角度，而不仅仅是系统或组件的角度来度量性能。集中化测试系统后，每台计算机的停机时间缩短，并支持全天候测试，更易高效利用系统。

- 切实可行的目标

构建 Performance CoE 是一个切实可行的目标。您可以从小模型开始，利用现有资源，等价值体现出来时再扩大。以后当管理层和 IT 员工开始认识到模型的价值时，可以反复扩大资源、服务和能力。只要正确实现 CoE，它即会证明其价值。



通常，Performance CoE 可以改进性能和可用性，降低总拥有成本(TCO) ，并确保用最合适的人员为公司效力。

CIO 们常常会问类似以下的问题：“如果我们转换到 Performance COE，ROI 会如何呢？”性能测试往往被视作希望推动总体应用程序交付效率的一个瓶颈。问题在于任何 ROI 都取决于测试组织的初始状态。

无可否认，某些 COE 元素或集中测试组可能会改进效率。但是，大多数改进都不单单基于组织结构的更变，而是基于最佳实践的积累和组织行为的转变。

如果测试是瓶颈，则是因为“他们总是在项目后期查找错误”，他们只是在执行份内的工作。如果那些错误真的是由于要求未满足，Performance COE 则无能为力。

任何时候构建 Performance CoE 都不会太晚，本指南既面向无任何集中化测试的组织，也面向已实施其中部分措施的组织。

首先要确定企业的当前状态、定义目标、然后执行实现目标的必需步骤。

CoE 阶段

Performance CoE 模型非常容易实现。它允许您用最低的成本从小模型开始，往后逐渐递增。以后当您向管理层和 IT 员工证明其价值时，可以反复扩大资源、服务和能力。只要正确实现模型，成功率指日可待。

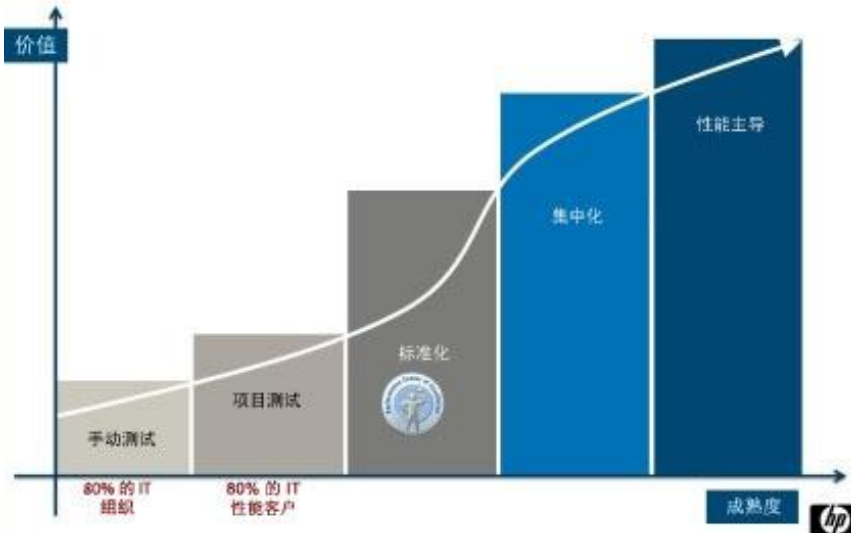
通常，有可能令公司研究建立 Performance CoE 的比较有说服力的事件是，关键性能问题、项目陷入困境或削减成本的需要，一般都处于部门层面。有时，该研究过程由性能专家在内部启动，性能专家希望解决测试中的应用程序日益复杂以及专业技术领域不断细分的问题。还有时，建立 CoE 的指示来自于测试组管理层，他们通常基于硬件合并和许可证重用等业务驱动程序的考虑，或迫于业务单位抱怨响应时间慢的压力。

无论由谁启动 CoE，它都将随着时间的推移从硬件和软件资产合并阶段逐渐过渡到性能主导阶段。在每个应用程序经过投产前的这些阶段时，专门技术得以保留供共享，测试方面的最佳实践也会记录在册并予以遵循。

组织成熟度

套用一句俗语，没有两个完全相同的 IT 企业。但是，不同的公司在性能测试方面却有着某些相似的管理方式。大多数公司经历相同的成熟轨迹，有时达到更高级别，但有时在平稳期作短暂停留。由于开发 PCoE 是一个持续过程，因此关键是要了解公司的当前成熟度，并评估测试组位置。即使团队才开始攀登生产力高峰的旅程，评估当前状况也会摆正过程的位置，有助于计划接下来的步骤。

性能测试团队发展过程



让我们来详细了解各种成熟度以及它们如何融入 CoE。

手动测试

这是最基本的测试类型。没有自动化测试，有零星的需要验证应用程序性能的工作量；没有业务责任制，有系统生产性能问题。遗憾的是，大部分 IT 组织都处于这一阶段，他们缺少实际性能工具、缺少度量测试业务结果的过程。

项目测试

这类测试可以被形容为迈向真正自动化测试的第一步。它通常基于部门，针对部门层面使用的特定应用程序。每个部门购买自己的一套工具和许可证，并根据自己的基础结构进行测试。一个部门可能有特定领域的专门技术（比如 **J2EE** 应用程序），另一个部门可能有常驻数据库专家，但组之间不共享知识。在大多数情况下，由开发人员执行测试，他们不具有业务客户的直接沟通渠道，因此无法洞察业务目标和应用程序临界状况。

此类测试导致有限结果，原因如下：

- 未重用基础结构和测试工具许可证
- 未重用人员和技能
- 实践和业务目标之间存在不一致

标准化

这是通往真正 **Performance CoE** 中的阶段。当组织发展到合并硬件资产和软件许可证时，通常是因为他们知道构建和维护类似生产的性能测试环境的成本高得令人望而却步。如果在各类业务之间复制测试实验室，这一成本甚至会更高。另一方面，无法建立准确的性能测试环境的后果可能是灾难性的。

Performance CoE 最初可以充当合并性能测试所用的软件和硬件的分析师，通过更好地利用资产和降低管理及维护成本来节省成本。在此阶段，**CoE** 将从关注部门的服务组织转变为实质上管理基础结构并支持用户的跨各类业务领域的资源。

至此，组织仍然没有意识到开发性能工程师的技能集和共享知识。当项目团队和管理层开始认识到 **Performance CoE** 模型的优点时，**CoE** 可以逐渐将其服务扩展到更广的性能问题。

集中化

这是通往性能测试团队更深入参与软件开发和部署生命周期各个阶段的下一个步骤。

在此阶段，Performance CoE 团队开始提供包括计划、编写脚本和执行测试的实际性能测试服务。传统上，多数业务领域受限于性能测试知识和行业最佳实践及过程，但有了 CoE 模型，他们可以获得性能专家的经验和建议。

此阶段有时称为 *服务实用工具* 模型，在许多要求实施 CoE 的客户组织中非常流行。CoE 的建立为培养专门、稀缺的性能工程知识创造了机会。它可以使员工跨业务领域工作，解决技术集成的交叉部分的问题，提高对体系结构协同工作方式的理解。所有这些都是吸引和留住人才的附加因素。

性能主导

CoE 成为 *性能主导* 时成熟度最高，即 CoE 成为应用程序开发、部署和操作的例程部分，助力于关注性能和成本效益的组织文化。在性能主导模型下，应用程序必须经历所有包括的针对于增强性能的持续过程，才能到达生产阶段。

性能主导履行以下基本原则，部分基本原则涵盖在 CoE 开发的早期阶段：

- 支持
对于其关注区域，Performance CoE 应该为各业务领域提供支持。这可能是以提供服务或主题内容专家的方式给予支持。
- 准则
提供准则的一些典型方式是标准、方法、最佳实践、工具和知识库。
- 共享知识
要鼓励共享知识，请提供培训和认证、技能评估、团队建设和形式化角色。

- 度量

CoE 应当能够展示其价值，通过输出度量的使用证明创建该模型的正确性。度量是 CIO 和业务可见性的基础。

- 治理

有效利用资源（资金、人员等）并发挥所有潜能是 CoE 的重要功能。这可确保组织投资最有价值的项目，创建规模符合服务项目的经济结构。而且，需要协调其他公司利益，以使 CoE 实现价值。

这些基本原则的正确履行可以使 IT 组织构建更有效的 CoE，致力于总体客户满意度，并达到或超越组织服务级别协议(SLA)的要求。而这又会提高 ROI，扩大性能主导的认可度。

2 构建成功 CoE

本章节介绍计划 Performance Center of Excellence 以及在您当前 IT 组织内建立这样一个实体的执行步骤的最佳实践。

性能测试是测试负载下的整个体系结构的性能。性能测试空间内应当应用的资源和您组织执行此类测试的总产能取决于以下几个问题的答案：

- 性能测试多久执行一次？
- 性能测试的严格程度如何？
- 生产中出现性能问题时会带来多大的风险和/或成本？
- 向性能测试分配多少技术资源，多久分配一次？

如果您的答案表明您的应用程序团队或公司正在付出极大的努力执行性能测试，而且这种努力会消耗您组织内现在或将来的多个资源，则应当认真考虑构建 Performance Center of Excellence (PCoE)。

正如前面的章节所提到过，Performance CoE 模型允许您从小规模开始，使用现有资源，实现立竿见影的切实收益。经过一段时间之后，当您证明了有价值的业务结果、在项目团队和管理层中赢得了更大的认可度时，就可以扩大到更高级别的成熟度和服务。

PCoE 促成性能测试最佳实践，定义并实施性能测试过程，向操作团队推出这些过程，帮助制定组织的战略方向。

PCoE 模型支持跨所有项目利用一组性能专家和技术性能测试组件。性能测试专家培训和补充每个项目团队内现有的资源，确保一致性。这要求极大的项目洞察力和稳定的人员配备。

PCoE 建立项目间共享管理和性能测试最佳实践的基础，以更高效地使用资源。**PCoE** 促进性能测试可交付结果的可重用性，比如测试计划、测试案例和缺陷报告。该组维护在每个项目中创建的可交付结果的存储库，为将来的测试工作提供框架和样本。

通常，**Performance CoE** 可以帮助您实现以下性能目标：

- 增强成功推出新代码、新软件和新硬件配置的信心；它允许您在投入生产前识别产能限制、并发问题和其他性能瓶颈。
- 通过应用程序调优提高现有代码性能、减少总资源需求、缩短总响应时间。
- 通过更好地理解您的体系结构性能配置文件及其可扩展性特性，改进正在进行的产能计划/管理。

两种构建 CoE 的方法

组织通常基于两个方向的压力，意识到需要建立 Performance CoE:

- 自上而下的方向，管理层强制要求构建 Center of Excellence
- 自下而上的方向，测试社区本身提出和沟通测试合并的迫切性

我们自己进行的客户和行业来源抽样调查结果表明，第一种影响的范围远远超过第二种。

自上而下

在自上而下的方法中，公司的应用程序开发副总裁或类似职位的人员授权测试组提出建议，以创建培养和集中测试资源及知识的面向过程的中心。当业务经理（比如 CIO 至或 CEO）接到无数投诉，抱怨任务关键的应用程序性能低劣时，通常会出台这一举措。有时，下达这样的指示可能是由于外部顾问声称，CoE 可以提高客户满意度，利用积累的知识，提供硬件和软件资源的更好的 ROI。有时，CIO 或应用程序开发副总裁可能接洽过性能工具供应商，或参加过宣传这样的概念的行业博览会：建立 Center of Excellence 会使公司大大受益。

无论事出何因，管理层都应当清楚地表明 Performance CoE 的目标、初始覆盖面和预计上线日期。建议从定义这一举措的任务声明开始。*任务声明* 一词这些年来遭遇极大的不光彩，因为组织（通常在顾问的提示下）举出了无数缺乏任何实质性内容的示例。有用的任务声明实际上应符合计划和操作。以我们为例，测试团队的任务是团队每天要完成的内容。

定义了此信息后，通常要求测试组创建详细提议，包括责任、时间线、报告流程和预期困难以及报酬。管理团队对提出建议及随后的讨论和实现过程的关注可能是决定项目成功与否的最重要因素。

自下而上

在某些组织中，强大的测试组在观察到各个开发团队工作不协调、缺乏业务领域的知识或低效使用贵重硬件和软件资源后，可以提出创建 Performance CoE 的建议。

虽然实现测试效率和提高客户满意度的目标与自上而下的方法是相同的，但通过自下而上方法创建 CoE 却比较少见。此方法通常涉及阐明业务案例、寻找管理层发起人和设立接受标准。

计划并执行项目的以下阶段至关重要：

- 定义预期目标

CoE 的建立应当肯定组织的业务目标。应当与组织的主要干系人商议后再定义 CoE 的范围。

- 定义当前测试模型

每个改进都应基于定义的基准模型。应该分析并记录现有测试过程，以便将来可以衡量测试中心的收益。

- 制定实现计划

定义范围后，制定基础结构要求、工具集和人员配备方面的具体计划。明确定义标准、策略和过程，一旦测试 CoE 上线就可以使用它们。

- 设立 CoE 服务

设立 CoE 服务后，多类业务(LOB)都可以针对质量保证需求使用测试 CoE 服务。

- 评估结果
按照预先确定的预期目标，评估 CoE 的效率。根据需要进行改进。
- 最佳实践和方法
创建各个知识领域的核心能力，并以标准化模板、标准、准则和检查清单的形式公开此知识。

结果

所有这些步骤都应支持确立构建 Performance Center of Excellence 的 *业务目标* 的主要目的。这可以通过回答以下问题来实现：“*性能验证方面的主要难点是什么？*”对该问题的回答促使您立即采取行动，定义更改范围和指定衡量进度的度量。

以下是某个成功构建 PCoE 的 HP 客户定义的目标列表：

- 促成最佳实践
用知识和负载测试最佳实践创建中心组。定义所有 IT 通用的统一框架和方法。
- 集中化工具管理
提供所有应用程序问题的单个内部联系点。
集中应用任何更新和修补程序。
提供更有效的许可证管理。
- 利用工具、资源和设备
通过跨企业共享人员、过程和技术，进一步节省成本。
- 对更多应用程序进行负载测试
在组织内更广泛地采用负载测试
- 最小化 IT 风险
对部署到生产系统的应用程序的性能具有更大信心。

组织评估套件

确立业务目标后，评估阶段评估组织对构建 Performance CoE 和完成前一步骤中定义的业务目标的准备情况。它还为组织的预期 ROI 提供 Performance CoE 的可能价值基础。

不同企业使用 Center of Excellence 的熟练程度不尽相同，并且看待 Center of Excellence 的战略重要性也会不同。组织成熟度模型是一种估算企业创建 Performance Center of Excellence 时要经历的阶段的好方法。这些模型当中最为人所熟知的可能是能力成熟度模型（CMM 或 CMMI）。

CMM 模型将组织的成熟度分为五个级别（逐级递增），与前面 [章节](#) 中提及的成熟度级别类似：

- 初始（混乱、非正式、临时、大型）—使用新过程的起点。在分类中称为 *手动测试*。

组织在开始意识到需要构建 CoE 之前，通常为初始阶段。一开始可能是特定的业务领域或个人具有能力。

- 可重复（有管理、有记录、有过程规定）—重复使用过程。在分类中称为 *项目测试*。

组织开始将 CoE 视为项目团队资产时，往往开始进入可重复阶段。借助于此项目中心视图，他们知道团队需要支持，团队正在寻找一个获取更深层次的所需技能的地方。他们开始跨项目协作，帮助计划和设定范围，监控他们负责构建的能力。但是，此阶段所产生的结果仍然很有限。大部分企业至少已发展到此阶段。

- 已定义（制度化、集成）—过程被定义/确认为标准业务过程。在分类中称为 *标准化*。

CoE 的下一个步骤是开始定义和记录能力标准和实践。在此阶段，他们以 wiki 形式捕获最佳实践，并开始更积极地管理相关风险和质量。他们建立基本培训，合并硬件和软件资源，在组织中更积极地交流能力。很多组织不需要履行成熟 CoE 的所有责任，并且可能不会跨越到更成熟级别。大多数组织会满足于达到成熟度的前几级。跨越到下一个更高级别需要较强的协调性，因此，需要各级管理人员的投入和大力支持。

- 有管理（战略性、已量化）—共享最佳实践，执行过程管理和度量。在分类中称为**集中化**。

得到管理层支持的组织可以开始构建有管理、战略性的 CoE。焦点应当不再在部门内，而涉及整个组织。在此阶段，测试组织可能开始支持公司目标和计划，与公司记分卡相结合，主动在使用这些服务的组织范围内管理举措组合。因为建立了更正式的职业通道，并提供 CoE 支持的能力培养和指导，Performance CoE 的能量开始真正得以释放。

- 优化（持续改进）—包括慎重的持续过程优化/改进。在分类中称为**性能主导**。

优化的 CoE 继续提升公司价值。在此级别，CoE 是整个组织认可的资产。CoE 员工对公司目标具有主人翁精神，通过支持这些目标保证项目成功。评估确保符合目标所需的能力得到一致应用和改善。CoE 负责公司记分卡及其相关值。审核、技能评估和认证会进一步改进能力。

通过前面的阐述，该如何实际衡量测试组织的成熟度和状态呢？最好是从评估当前软件工程和管理实践开始。该评估的结果是一份获取公司目前软件测试实践的强项和弱项的详细报告。通过以下方式进行分析：

- 管理层讨论
- 调查表（参见[附录 A](#)中的样本）
- 反馈
- 结构合理的约谈

- 分析
- 操作计划

公司的*业务目标*与 IT 组织的特定*应用程序性能目标*保持一致是构建 CoE 的关键。只有了解这些目标之间如何相互影响，IT 管理层才能开始确定一组衡量业务目标实现情况的 KPI。这些度量不仅定义可度量 Performance CoE 的成功标准，而且还构成用于确定 Performance CoE 经济价值的基础。例如，它们可以显示应用程序性能的提高如何直接降低客户服务成本。

在评估阶段，组织建立组织变更战略，制定实现变更的路线图。它还选择所需的服务以帮助使变更制度化。

此外，评估还合并以下过程：

- 设计 Performance CoE 的组织结构
- 概括构建 Performance CoE 的组织影响
- 编制现有硬件、软件和人力资源清单
- 指定其他需要的资源
- 定义全局目标
- 确定高级里程碑和时间线
- 计算 Performance CoE 在成本节省和过程改进方面的预计价值

概念论证

创建成功 Performance CoE 的一个重要步骤是，对概念项目进行论证。有时称为试验，其目的是：

- 验证设计阶段阐述的概念
- 论证建立 CoE 的初始价值
- 在组织内推广 Center of Excellence 的理念

要进行试验，组织应当确定 Performance CoE 的初始关注点，并定义特定实现迭代的确切内容。例如，可能选择让 Performance CoE 关注应用程序生命周期的一个特定阶段，比如应用程序开发，并实现支持该阶段所需的技术和资源。或可能关注特定活动，比如产能计划。或者可能选择关注特定业务单位的需求，并为该业务单位创建 Performance CoE 模型，以后可以推广到公司的其他领域。

准备试验阶段时，需要解决尽可能多的 CoE 方面，但范围较小。例如，如果试验的关注点是特定业务领域，建议让业务分析师和此 LOB 的开发人员参与：

- 计划试验
- 确定项目干系人
- 评估现有测试基础结构硬件
- 盘点使用中的软件测试工具，并选择其中一个工具作为首选工具
- 创建按严重性和优先级细分的应用程序列表，然后选取 2-3 个应用程序作为概念论证练习的目标
- 定义明确性能目标
- 定义试验团队成员，其中应配备来自提议的 CoE 的测试专家和 LOB 代表
- 制定试验项目的时间线，并确认交付点
- 记录每个步骤的进度

如果选择诸如 **J2EE**、**.Net** 或 **SAP** 的特定技术作为试验活动的关注点，则论证可度量结果的小组成员中必须有所选技术方面的专家。

从客户体验角度看，试验项目的持续时间一般是高强度工作的 **2-3** 个月，在此期间持续改进 **CoE** 的多种元素，以便为在组织中首次亮相做准备。即使执行小规模试验，也必须取得执行管理层的支持：**LOB** 管理层或应用程序开发副总裁，甚至 **CIO**。定期向管理层发起人报告进度和问题。

在过程结束时，最好召集试验团队及其客户（业务分析师、开发人员和所测试的应用程序的最终用户）召开分析会。广泛反馈可以防止将来发生不必要的错误，便于精简 **CoE**。如果可以的话，请邀请管理层发起人参加这个分析会。无论如何，摘要报告中都应当记录过程及结果，用作推出 **CoE** 的主计划。

沟通

建立 Performance Center of Excellence 时，最重要（通常会被忽视的）的因素之一是组织中各方之间的沟通，比如业务、财务和 IT 人员。

始终建议向项目干系人汇报情况，但这一点在建立 CoE 时愈显重要。我们已经提到过整个活动期间取得管理层支持的重要性。您应当定期用相关报告给予支持。报告应包括每个项目干系人根据其角色特别感兴趣的数据。

遗憾的是，性能测试人员往往：

- 不甚了解项目干系人想要知道哪些信息
- 不会删除不需要或不相关的信息（信息越多并不表示越好）
- 不擅长将技术问题转化为可能的业务影响
- 不善于以易于理解的方式提供数据

若要改善各方对 Performance Center of Excellence 的普遍态度，并有效地表达其价值，请付诸一些精力为公司众多项目干系人精心制作信息报告。下面按项目干系人角色讨论需要与不需要做的沟通：

- 管理层发起人
 - 需要做
 - 报告可能会延迟上线日期的问题
 - 报告上线后可能发生的问题
 - 不需要做
 - 提及您自己能够处理的问题
 - 列出任何技术详细信息或运行的测试
- 项目经理
 - 需要做
 - 报告项目计划风险
 - 报告项目预算风险
 - 提醒注意任何可能影响业务认可度的问题

- 报告所有计划完成的任务的状态
- 表达项目里程碑和与其他团队的依赖关系
- 不需要做
 - 提及每个遇到的小问题
 - 列出任何技术详细信息或运行的测试
- 应用程序架构师
 - 需要做
 - 表达正确测试方法
 - 提醒注意您的结果是否可能证明系统设计、功能设计或硬件要求是错误的
 - 报告应用程序符合其功能和性能目标的能力
 - 不需要做
 - 提及预算或资源限制
 - 列出计划冲突

报告内容不应只与角色相关联，因为各个组织的角色定位可能有所不同。

请遵循以下通用报告规则：

- 根据阅读报告的人的职能自定义报告
- 避免自动发送由性能工具生成的报告—请先审阅报告并删除不相关数据
- 以最重要信息开头，然后展开详细信息
- 只有大量的数据结果不利于传达您的信息—应该包括结论和分析，还可能包括如何改进性能的建议
- 最好突出几个主要业务事务的响应时间，而不应将响应时间页面直接粘贴到报告中
- 使用度量论证 CoE 的运营效率

- 书面报告固然重要，但还建议您召开会议讨论项目完成情况，以确保其结果得到正确理解
- 宣扬测试团队取得的价值，并乐于解释和宣传运营 CoE 的好处

工具标准化

正如本文档 [前面](#) 所介绍的，工具标准化是走向完全成熟的 Performance CoE 的前几个步骤之一。在这个基本级别，它包括编制各个业务领域的硬件资源和软件测试工具的清单，以查找共同的标准化工具。此清单有很多种可能的结果。

您可能会发现，因为公司以总购量折扣购买了软件，大多数项目都会使用相同测试软件。虽然这意味着积累了大量的该工具专门技术，但仍然要验证工具对所有业务领域目标的支持能力。确保它可以提供为自动化关键性能优化过程而设计的一整套应用程序，比如测试和调优、瓶颈识别、产能计划和诊断。

另一方面，清单检查可能会揭示公司使用两种或三种来自不同供应商的软件包，而且没有一种软件包明显优于另一种。在这种情况下，可能有必要对市场上出售的性能测试软件作出对比分析，并启动更换过程。

甚至在公司出于财务考虑决定继续使用现有工具集时，也建议将许可证移动到集中负责组。这会优化利用率，减少成本。

HP Performance Center 软件由于具有以下优点，可以用作最终性能软件包：

- 从单个控制点生成一致、可度量和可重复的负载测试
- 允许将实际用户替换为数千个虚拟用户，并跨所有层隔离性能瓶颈
- 管理所有性能测试资产和计划您的测试，所有操作都通过单个 Web 接口完成

与 HP Diagnostics 软件配合使用，允许技术特定的产品向下探究到代码级别，以解决 J2EE、.NET、Oracle、PeopleSoft、SAP、Siebel 等的性能问题。

HP Performance Center 附有 HP SiteScope 安装包和许可证，后者是生产和测试环境的先进监控解决方案。

至于硬件资源，将大多数或所有服务器、路由器及其他网络设备合并到一个位置，促进之前分散的团队之间加强合作，从而跟上紧张的测试计划。不是每个测试组都保留产品市场的一个部署领域的测试环境，而是根据总部署计划在需要时将测试环境展示出来。这不仅提高硬件利用效率，而且降低支持测试环境的总资源成本。

总的说来，工具标准化能够：

- 提高利用率
 - 对所有项目提供负载测试
 - 可见性促使负载测试无处不在
 - 更灵活的资产访问提高使用率
- 减少成本
 - 通过池许可证模型降低采购成本
 - 通过中心基础结构降低维护成本
 - 由于资源集中化和支持合同涵盖硬件或软件的所有实例，降低支持成本
 - 由于可以统一培训，降低人工成本

3 成功操作 CoE

任何软件测试阶段都需要以下一组核心测试可交付结果：

- 性能测试战略
- 性能测试计划
- 测试案例
- 缺陷收集
- 状态报告

这组交付需要性能测试团队完成过程的各个阶段，从计划到测试，到最终缺陷解决和状态报告。这并非是一组不可更改的测试可交付结果，但它可以帮助任何测试组织开始确定一组合适可交付结果的过程。

一个常见的误解是，这些内容必须以一组“物理”文档提供。有应用程序可以帮助实现这些可交付结果的目标，而不必创建一个或一组物理文档。目标就是尽可能简明地将所需内容收集到有用、一致的框架中。

在开始详细介绍各种测试可交付结果之前，有必要提一下 Performance CoE 员工在创建这些可交付结果时通常会碰到的问题（请参见[沟通](#)部分）。

性能工程师需要知道非测试人员难以理解测试概念。他们通常不明白用于完成测试的工具集细节。他们所知道的就是，想要的结果是可以自信地销售可靠的高质量软件。若要保证此结果，测试人员需要知道如何与不懂软件测试的客户、项目干系人、开发人员和经理沟通。

以下写作技巧可能会对您成功记录过程和与各性能测试方沟通有帮助：

- 确定谁是您的读者
- 了解他们想要知道些什么
- 忽略掉您的读者不需要知道的信息

- 直切主题。简洁明了。不要跑题
- 您的读者可能不会读完整篇文档—突出重要部分
- 清楚地指定操作项

项目计划

性能测试战略

很有可能，测试文档集中的第一个文档就是性能测试战略。虽然该文档适合于大型组织的大项目，但对定义性能测试的总体方法会很有用。该文档一般描述性能测试团队在测试阶段中执行的，影响计划、资源安排和预算的“重要”活动。理想情况下，它应回答有关测试周期的下列部分核心问题：

- 性能测试阶段在哪个测试阶段发生呢？
- 使用怎样的环境进行性能测试？
- 用什么工具进行测试？
- 测试团队的规模多大？
- 团队需要向谁寻求帮助？
- 该测试阶段的依赖关系如何？
- 进入和退出条件是什么？

性能测试计划

性能测试计划至少要表述测试本身：相应测试阶段或多个阶段的目标、范围、方法、假设、依赖关系、风险和计划。很多测试组织还使用测试计划描述软件测试阶段、测试技术、测试方法和任何测试工作的其他通用方面。建议您将此信息记录到“最佳实践”存储库中，这可避免提供给读者的信息有冗余和冲突，同时可使测试计划的重点放在即将执行的任务上——计划测试工作。在大多数情况下，测试计划的内容应当处理以下几个问题：

- 测试环境与生产环境是什么关系？
- 在两个环境中都应如何解释性能测试结果？

- 应当使用多少事务量？
- 多长的事务响应时间(TRT)令人满意？
- 数据创建和总体环境准备的计划是什么？
- 他们应何时开始脚本开发？
- 如何执行测试？
- 需要多少个测试执行周期？
- 何时及如何进行分析 and 报告？

以下部分详细介绍了测试计划的各部分内容。

目标

当前测试工作的目标需要表述清楚，以便软件测试团队和参与部署的任何其他组织都能正确理解。不应泛泛而谈，空洞地陈述测试“整个应用程序”——除非实际目标确实如此。主要测试目标应该与当前发布的目的相关。如果这是一个电子商务系统，当前发布的目的是提供其他付款功能，则目标（有时称为任务声明）可能是：

为了确保增强的付费功能按总体规范运行，请验证所有现有功能都在范围内。

测试目标描述测试工作的“理由”。测试工作的详细“内容”在测试计划的范围部分描述。再说一遍，任何总体测试目标都应记录在“最佳实践”存储库中。任何测试工作的总体或共同目标都可以包括展开测试案例回归套件、记录新要求、自动化测试案例和更新现有测试案例。

范围

在上面的目标部分，我们提到过测试整个应用程序的泛化表述通常让人无法掌握实际要旨。实际测试范围应包括应该测试（范围内）的和不应该测试（范围外）的项目列表。

- 范围内

这些是应该测试的系统的组件。这可以用详细列表的形式列出那些“范围内”的项目：要求、功能区域、子系统、业务功能或能向测试组织及参与部署的任何其他组织清楚描述系统范围的任何方面。测试计划的此部分应当回答“测试什么？”的问题——当前测试工作要涵盖的系统方面。

- 范围外

不测试的系统组件应明确定义为“范围外”。这并不意味着这些系统组件不会被执行或运用，只说明不会包括专门测试这些系统组件的测试案例。测试计划的此部分应当回答“不测试什么？”的问题。往往被人们忽略的是，测试计划的此部分开始处理所有测试组织必须解决的基于风险的计划：我可以不测试系统的哪些部分？这个问题应该在测试计划的测试方法部分予以解决。

方法

此部分根据当前测试阶段的应用程序定义要应用的测试活动。它解决如何完成系统的“范围内”方面的测试，还解决任何减轻因素，这些因素可能会降低将系统方面排除在范围外的风险。

方法应看作是待办任务列表，将在测试计划中再作详细说明。方法中应该清楚地说明要测试系统的哪些方面，以及如何测试：环境测试、界面测试、并行测试、回归测试、使用寿命测试、负荷和负载测试，以及适用于当前测试工作的任何其他测试方法。使用任何一组特定方法都应该描述其理由，通常从风险的角度进行描述。

假设

假设是测试团队认为真实的事实、说法和/或其他团队的预期。记录每个测试阶段特定的假设。这些是测试方法所依据的假设。如果证明假设错误，可能会对测试活动产生负面影响。在任何环境中，都有一组适用于任何既定发布的通用假设。这些通用假设应该记录在“最佳实践”存储库中。该假设文档应当只包括对当前测试工作唯一的假设，很可能应该记录对当前情况十分关键的通用假设。

依赖关系

依赖关系是指为继续任何既定测试活动而必须完成的事件或里程碑。这些依赖关系将在测试计划中呈现。在此部分，列出认为对测试工作十分关键的事件或里程碑，以及对测试计划的任何潜在影响或风险。

风险

风险是指可能对测试工作构成负面影响的因素。创建详细的风险列表，并描述对测试工作的潜在影响。项目计划中已逐条列出的风险不必在此重复，除非它们对测试工作的影响尚未表述清楚。

计划

测试计划定义执行测试活动的时间和人员。在这里使用收集的测试计划正文处的信息，再结合可用的资源池以确定测试计划。以前测试工作的经验和对当前测试目标的深入理解都有助于准确地制定测试计划。有几个计划和安排工具可以帮助您更轻松地构建和维护计划。例如，**HP Performance Center** 包括高级计划功能，允许您在一个项目中或两个项目之间有效共享测试资源。

测试案例

测试案例是测试案例设计的正式实现。任何一个或一组测试案例的目的都是为了检测被测试系统的缺陷。测试案例应该采用对当前和任何将来测试周期有用的方式进行记录。至少，每个测试案例都应当包括作者、名称、描述、步骤、预期结果和状态。

- 测试案例名称

名称或标题应该反映测试案例的本质，包括测试的功能区域和目的。使用通用命名约定有利于提高重用率和防止重复。请考虑在“最佳实践”存储库中定义命名约定。

- 测试案例描述

描述应当清楚说明测试案例要完成的业务事件的顺序。测试案例描述可以应用于一个或多个测试案例，若要完全测试应用程序的某个区域，往往需要多个测试案例。

- 测试案例步骤

每个测试案例步骤都应该清楚说明完成步骤所需的导航、数据和事件。使用通用描述方法有助于保持一致和提高重用率。关键字可以为测试案例设计提供最有效方法之一，并且可以应用于手动和自动测试案例。

- 预期结果

预期结果是在需要验证的任何测试案例步骤之后系统的预期行为。这可以包括屏幕弹出窗口、数据更新、显示更改，或执行测试案例步骤时希望发生的任何其他可辨别的事件或系统事务。

- 状态

这是指测试案例的操作状态。它是否已做好执行准备？

制成品

当性能测试执行完成时，下一步就是对监控和诊断阶段收集的数据进行分析。测试的结果以各种制成品的形式表现，比如检测到的缺陷和状态报告。

缺陷

测试的主要目的是在应用程序投产前检测应用程序中的缺陷。而且，缺陷大概是测试团队产生的、项目团队可以看到的唯一产品。请以对缺陷更正过程有帮助的方式记录缺陷。至少，每个缺陷都应该包括作者、名称、描述、严重度、影响区域和状态。

- 缺陷名称
名称或标题应该反映缺陷的本质，包括缺陷的功能区域和性质。
- 缺陷描述
描述应当清楚地说明导致缺陷的事件顺序。必要时可以包括错误的屏幕截图或打印输出。确保用非技术性语言表明业务影响
- 指明问题的频率或可能性，以便项目干系人可以确定严重度和优先级
- 复制方式
缺陷描述应该提供足够详细的说明，以便鉴别分类团队和修复缺陷的开发人员复制缺陷。
- 缺陷严重度
对缺陷的严重度分配取决于测试的阶段、缺陷对测试工作的影响，以及将缺陷引入生产会对业务带来的风险。
- 影响区域
系统的功能组件或功能区域可以参考影响区域。通常这两者都使用。

性能测试中发现的缺陷与功能测试人员发现的缺陷不同：

- 缺陷往往没有明确要求
- 开发团队若没有协助通常无法重现缺陷
- 性能缺陷实际上技术性很强，对非技术类管理人员来说很难理解。附加说明：非技术类管理人员是指分派开发人员修复这些缺陷的人。

在大多数组织中，都使用某种错误跟踪软件工具完成缺陷报告，比如 **HP Application Lifecycle Management (ALM)** 和 **Quality Center Edition**，允许您输入所有以上特征，并可以与 **HP ALM, Performance Center Edition** 无缝集成。

状态报告

Performance Center of Excellence 团队成员生成各种报告。虽然不是每个组织都生成各类报告，但最好定期记录测试结果，每天、每周或项目周期结束时记录一次。

所有状态报告的内容都应侧重于测试目标、范围和当前要解决的计划里程碑。您可以在每个状态报告开头说明各项内容，然后公布当前报告期间取得的成果或实现的目标，以及下一个报告期间将要实现的目标。任何会直接影响测试工作的已知风险都需要在这里逐条列出，尤其是会妨碍进一步测试系统的一个或多个方面的所有“瑕疵”。

- 报告期间

这是指当前状态报告中覆盖的期间。包括对任何应该审阅的之前状态报告的引用。

- 当前范围

指定测试的系统的组件（硬件、软件、中间件等）（“范围内”）和任何不测试的相关组件（“范围外”）。

- 计划里程碑

确定当前报告期间处理的所有计划里程碑，并说明它们的当前状态。已计划但当前报告期间未处理的里程碑则需要作为风险提出来。

- 风险

风险是指可能对当前测试工作构成负面影响的因素。逐条列出目前正影响测试工作的风险，并说明对测试工作的影响。

建议为项目干系人提供对报告及其原始数据的访问权限，例如，放到共享驱动器上：

- 这样使大家感觉可随意执行自己的分析
- 增加可信度，因为没有隐藏任何信息
- 提高参与度，因为每个人都可以使用熟悉的工具（比如 Excel）操作数据

测试执行报告

这是在每个测试执行后创建的简要、非正式文本，通常以电子邮件而不是 Word 文档的形式发送。这份报告的读者一般是技术类项目干系人，应该保留该文本以供 CoE 团队参考。

该文档应该表达有关测试执行的最重要事实：

- 测试的目的
- 配置更改（如果有更改）
- 测试的业务结果
- 调整或调查缺陷时的瓶颈的可能根源
- 对结果文件和场景文件的引用

记录测试执行并将它们发送给所需的各方会使测试人员对测试结果进行实际分析。

定期报告

即使不需要生成定期状态报告，仍建议这么做。在大多数情况下，定期报告（每天、每周、每月）的读者是项目经理，理想情况下应包括以下信息：

- 如何进行计划跟踪
- 任何可能影响计划、预算或项目资源安排的问题
- 此报告期间主要完成了哪些工作
- 下周计划完成哪些工作
- 没有必要列出已经解决的问题
- 不要包括影响报告阅读的不必要信息

摘要报告

这可能是项目或测试周期结束时总结系统性能和稳定性的最重要报告。即使对某些人来说可能技术性太强了，也应该每个项目干系人人拥有一份报告副本。

报告涉及测试周期的多个方面，比如：

- 执行了哪些测试，以及是否通过成功标准
在大多数性能测试周期中，结果比单单通过或失败状态更复杂。测试的主要目的是预测产品“上线”后的行为
- 清楚提供主要 KPI
它们应当包括响应时间、错误、事务量、系统资源利用率等。
- 范围更改
报告应当明确指出测试计划中所概括的范围和最终测试活动之间的任何差异

- 突出的问题
缺陷是唯一的测试结果材料，应当列出主要缺陷及其业务影响，如有可能按照优先级进行分类

- AUT 配置

报告中应提供范围定义以及所有相关信息，比如版本、关键设置、文件位置等

准备摘要报告时，请遵循以下准则：

- 提供结论和分析假设，不单单是原始测试结果
- 介绍测试中使用的最终设置。不需要列出调整设置的所有步骤
- 简要提及测试数据和其他准备活动的起点和状态
- 突出证明结论的非常重要的响应时间，其余的放在附录中

即使您遵循准则，并包括特定项目干系人感兴趣的相关信息，很多人也不会读完摘要部分。因此，除了书面摘要外，建议您召开一个会议，把最重要的结果和结论以 PowerPoint 等直观的形式传达给项目干系人。这将有助于您推行性能测试工作，提高大家对这项工作的认识和认可度。

4 招集 CoE 人员

建立 Performance Center of Excellence 的最重要动机可能是，利用在负载测试、脚本编写方面累积的知识，以及各种应用程序技术和领域的专业技术，最大程度利用可用的人力资源。

PCoE 首先是一组可以为多个约定提供测试知识、技术、方法和资源的测试专家和技术测试组件。测试专家培训和补充每个项目团队内现有的测试资源。该组成员通常维护创建的可交付结果，为将来的测试工作提供框架和样本。CoE 建立可以共享的最佳测试实践基础。CoE 促进测试可交付结果的可重用性，比如测试计划、测试案例、测试自动化和缺陷。此过程随着可重用组件的增多而提高测试工作效率。

角色

根据公司规模、组织结构、成熟度、甚至公司所属行业，Performance Center of Excellence 可以划分成多个部门和团队。在大多数情况下，中心包括以下核心角色：

- CoE 经理
- 解决方案架构师
- 测试实践主管
- 测试自动化工程师
- 业务分析师
- 运营经理
- 测试实验室专家

这些是角色和职责，不是资源。因此，一个资源可以在 PCoE 结构中履行多个角色。

此列表中包括环境专家角色，但在这里仅指与 PCoE 特别相关的那些活动。不包括为整个项目或组织而设立的任何更普通的环境活动或角色，比如配置管理。

CoE 经理

CoE 经理角色要有效领导测试团队。为此，经理在具备传统经理角色应有能力的基础上，还必须了解测试规定和如何有效实现测试过程。这意味着经理必须能够管理和实现或维护有效测试过程。这涉及创建支撑稳定通信的测试基础结构和经济有效的测试框架。

职责：

- 创立 Center of Excellence
- 定义 PCoE 在组织结构中的作用并宣传其重要性
- 在每个发布或交付上下文中定义测试范围
- 执行测试方法使用策略
- 帮助制定和维护服务级别协议(SLA)
- 向项目团队提供 SLA 知识
- 提出测试问题以供解决
- 实施和继续制定度量 PCoE 效能的合适 KPI
- 计划、部署和管理任何既定约定的测试工作
- 管理和持续更新满足测试委托所需的测试资产：
 - 团队成员
 - 测试工具
 - 测试过程
- 管理 CoE 人力资源和预算
- 向团队成员提供合适培训
- 留住有才能的性能测试人员

管理或领导测试团队可能是 IT 行业中最具挑战的职位之一。团队常常人手不足，缺少合适工具和资金。在这样的条件下，激励和留住骨干测试人才至关重要。有一些技巧可以帮助您应对这些挑战：

- 更新测试计划范围以适应不断变化的时间线
- 设定明确的测试团队和项目管理团队的预期
- 与开发人员和项目经理保持多种渠道的经常沟通
- 一旦相关就宣传测试团队的重要性和贡献
- 确保每个 Center of Excellence 团队成员都有一个明确的角色定位和清晰的职业发展通道
- 度量和传达 PCoE 投资回报
- 解释投资(ROI)而非成本方面的测试支出

解决方案架构师

测试解决方案架构师角色确定和维护 PCoE 的测试体系结构远景。该角色可能是任何 Center of Excellence 中最关键的角色，应该由在项目管理领域、测试实践领导领域或测试自动化工程领域拥有多个行业丰富经验的资深测试人员担任。测试解决方案架构师负责筛选和集成合适的工具、过程和步骤组合，以确保总体测试效率。测试架构师与 CoE 经理、测试实践主管、资深测试自动化工程师、软件供应商和质量保证团队紧密合作，共同制定并实施测试框架。

职责：

- 测试体系结构远景负责人
- 制定短期和长期测试体系结构目标
- 确保为满足这些目标选择合适的软件测试工具
- 将工具、过程和方法整合到统一结构中

- 向项目提供测试框架和模板
- 推动 PCoE 体系结构的改革

测试实践主管

测试实践主管角色提供支持有效测试设计和测试的过程、步骤和模板。它们应该包括测试设计、测试构建、测试执行、捕获测试结果、缺陷报告流程和测试覆盖面分析。

职责：

- 提供测试方法培训
- 与 QA 合作持续改进测试方法
- 提供有关方法实际应用的指导
- 提供有关组织测试项目的指导
- 评估测试战略和提出审批建议
- 确保技术测试过程应用正确的方法

测试自动化工程师

测试自动化工程师角色设计、建立、测试和部署有效测试自动化解决方案。若要履行此角色，测试工程师需要运用合适的自动化技术以满足测试组织的短期和长期目标。目标就是使用最少的代码/脚本尽量实现最多的测试工作自动化。其重点应该关注于优化测试工作，而不是测试覆盖面。如果一个手动测试案例或手动测试准备过程占用大量测试资源，则应该首先自动化该手动过程。

职责：

- 擅长测试工具的实际运用
- 培训最终用户使用测试工具
- 管理测试工具
- 提供升级的和新的测试工具的信息

- 执行各种测试，比如性能、负荷和产能

业务分析师

业务分析师(BA)有时称为主题内容专家(SME)，在 PCoE 层次结构中也许有或没有这一角色。大多数情况下，Performance Center of Excellence 是一个内部组织，业务分析师很可能是公司内个别实体的组成部分，或在各个业务领域间共用。如果业务分析师不是 PCoE 组织的一部分，PCoE 必须与 BA/SME 紧密合作，以确保要测试的业务过程得到清楚理解。如果 PCoE 是外包的外部组织，则 PCoE 应当包括与客户沟通并充当 PCoE 内业务联系人的 BA，以确保业务目标清晰，测试满足客户要求。

职责：

- 深入了解特定行业的现实
- 使方法包括趋势特定的过程
- 建议测试过程和测试阶段的适用性以支持行业趋势

运营经理

运营经理角色计划 PCoE 资源的使用，包括软件、硬件和人力资源。运营经理与 CoE 经理、解决方案架构师和实验室专家紧密合作，确保计划可以处理任何已计划或未计划的资源重叠。这是资源库的实现，资源库必须让所有业务领域和 IT 部门的所有 PCoE 合作者看到和了解。特别是在多个客户或合作者共享共同的资源池时，更是如此。

职责：

- 计划共享的项目资源—人力和环境资源
- 管理可重用测试业务组件，比如计划、案例、数据和脚本
- 公布测试计划

测试实验室专家

测试实验室专家角色确保 PCoE 硬件（和某种程度上的软件）可以支持所有计划的测试活动。

职责：

- 管理硬件、软件和网络，包括应用程序和系统
- 执行技术产能计划
- 支持各种测试，比如性能、负荷和产能
- 向技术支持组提供主要接口
- 向项目团队提供技术咨询
- 管理可重用技术测试组件
- 监控测试期间的技术环境

请记住很多 PCoE 员工都要履行多种角色。

负载测试技能

有效测试会在产品上市或支持内部业务功能之前，提供产品或产品增强的准确状态信息，从而产生重要业务优势。这允许组织在实施之前采取适当的措施，和计划合适的实施后支持。

但是，没有正确的负载测试设计和执行，就没有有效测试。性能测试是一组专门的技能。它需要应用程序、硬件和第三方系统的知识。它涉及从体系结构立场了解应用程序行为、建模技能和系统方法。并非所有测试人员都具备这些专业素质，它需要花数年的时间才能开发出一组合适的技能。

因此，Performance Center of Excellence 所面临的最重要挑战之一是，如何培训和留住性能专家。

这一挑战的第一个部分可以通过全面、持续的教育来实现。这里有份测试自动化工程师（每个 PCoE 的核心角色）的学习路径样本：

- 基本知识
 - 编程：至少熟悉一种面向对象的语言或 C 语言
 - 操作系统调优技能
 - 网络分析和调优专门技术
 - 数据库分析和调优技术
 - 系统的体系结构视图
 - 监控基础结构瓶颈的能力
 - 不借助工具而发现和解决系统瓶颈的能力
- 负载测试能力
 - 不借助软件测试工具而有效进行测试的能力
 - 抽象查看测试并正确识别软件测试和非软件测试的较差测试实践的能力

- 不借助于软件而执行性能测试的能力
- 用合适语言向技术类和非技术类读者报告发现的能力（请参见[沟通](#)）
- 超出基本要求
 - 参加供应商提供的认证课程。HP 服务安排的培训课程内容最一致，而且覆盖了在测试领域取得成功所需的大多数主题
 - 通过教辅安排开始积累专门技术：首先做初级测试员，然后与同事平起平坐，最后当一个或多个项目的主管
 - 成为其他人的辅导者

至于留住专家的挑战，则由 Performance CoE 通过为 IT 专业人士提供有吸引力的新职业机会来解决。这有助于组织招到并留住高级人才。若要支持此目标，PCoE 经理应该与团队保持联系，打开新视野，鼓励自学，定期奖励做出突出贡献的员工。

应用程序知识

在前面的主题中，我们讨论了从 CoE 经理一直到性能工程师的每个 Center of Excellence 角色的职责问题。还有适用于 CoE 成员的专门负载测试技能。

除了基本专业技能外，还有涉及了解测试中的应用程序(AUT)、技术、API、UI 等的其他特定门类的知识：

- 技术

由于错误一般是与技术相关的，或至少受系统技术影响的，因此性能工程师应当了解系统的构建方式，包括掌握基础的编程语言、系统体系结构、操作系统功能、联网、表示层、数据库功能和实施等等。

- 应用程序方面

其他错误较少源于系统的构建方式，而是源于系统应该要执行的操作。特别是很难预测正确结果应该如何。因此，工程师需要了解系统解决的业务、技术或科学问题。

虽然具备技术和应用程序方面的知识是显而易见的，但是拥有这些知识可能是性能验证项目成败的关键因素。例如，如果 AUT 通常是在行业领先的应用程序服务器上运行的 J2EE 应用程序，则测试团队中必须有 Java 专家。这位专家的知识可以帮助您查明 Java 特定的瓶颈，使用 Java 性能分析工具调优系统的鲜为人知的参数等等。

实际上每个 CoE 团队都需要的另一种常用知识就是数据库性能专家。大部分现代应用程序通过直接 SQL 调用或通过封装所谓的数据访问层的应用程序服务器对象，大量使用关系数据库中存储的数据。识别数据库相关的问题是一门很难得的、但每个 Center of Excellence 都应该拥有的必需技能。不管对于 Oracle、IBM DB2 还是 MS SQL Server，始终都需要数据库管理员，但他们中很少有人同时还会处理性能问题。如果您有幸拥有这样的人才，建议您设法鼓励他们开发有吸引力的职业发展通道，以帮助将他们留在团队里。

即使团队开始积累常规知识，这些知识对 ERP 和 CRM 这样的复杂企业系统来说，通常也不够用。这些系统需要非常精通总体应用程序体系结构，并熟悉某些模块、应用程序设置以及尤其是可能显著影响 AUT 行为的实施特定细节。

项目成员有时会遇到超出其能力范围的状况。他们不需要用更多人力来完成任务，只是需要更多的知识。在这种情况下，您应该在传统意义上的团队中加入顾问。顾问有不同的类型，但这时雇用所依据的基础特征即是 *专业技术*。

一般而言，为 CoE 配备的人员合适与否决定着方案是成功还是失败。务必通过提供持续富有挑战性的培训安排、提供专业论坛和活动以及认可成绩，来支持整个 CoE 内的高水平知识。

5 结论

Center of Excellence 有时称为能力中心，已经存在多年，但主要关注于特定技术，比如 J2EE 或 .Net。Performance Center of Excellence 关注性能验证和处理过程及工具，以增强测试效能。

PCoE 模型帮助创建一组测试专家和技术测试组件，可以在所有项目中利用测试知识、方法和资源。测试专家培训和补充每个项目团队内现有的测试资源。这样可以使用最少的人力资源和设备覆盖最多的项目。

成立和运营 PCoE 可以增加任何投资的短期和长期回报。当组织结构到位时，测试过程立即可以成为组织中建立的成熟过程。成熟的 Performance Center of Excellence 的好处包括：

- 提供在项目间利用环境资源、知识和人力资源的途径。
- 向所有项目团队直接或间接提供测试专家。
- 实现项目间测试一致性。
- 支持制成品的可重用性。
- 提高最终产品的质量。
- 允许为过程改进建立关键度量及其分析。
- 采用基于风险的测试、要求跟踪和知识管理技术提升测试组织业务价值。
- 帮助您兑现测试计划和预算承诺。
- 通过共享人力和技术资源，减少项目间的测试预算。
- 使测试工作可预测且准确。

在此框架中，战略资产—人员、过程和技术—共同构成强大的治理模型。这有助于增加效率、最大化测试投资回报和显著改进软件质量。实现 CoE 框架所节省的资源可以重新投资于能增加竞争力的新解决方案和服务。

已[提到过的](#) 某行业分析公司名为 **voke** 的研究结果表明 PCoE 模型能够提供持续的跟踪和满意度。提到建议时，**voke** 建议采取以下措施：

- 建立认同—获取强大管理层支持
- 与业务领域协作 — 谈论业务而非技术
- 自定义适合组织的方法 — 组织就像人一样构成各不相同
- 获取技能—以人为本
- 获取技术—合适的工具和教育很重要
- 关注客户满意度—视（包括内部）客户为上帝
- 进行试验以证明结果 — 一步一个脚印
- 推广结果—做您自己的营销人员
- 激励 CoE 员工—人才是您的最大资本

客户会将竞争力和性能等同起来—确保优化人员和资源的使用，从而满足客户的速度要求。

A PCoE 调查表

此附录列出了要向处于各个 Performance Center of Excellence 构建或运营阶段的客户调查的问题。这些问题可能有助于您对自己的业务成熟度进行自我评估。

1. 绘制常规组织结构图及性能团队在组织中的合适位置。
2. 描述性能测试组织的结构。什么角色（如果有）在此组织中发挥布局和外包作用？
3. 此组织的职责是什么？
4. 您是否将测试组织当做 Performance CoE？
 - a. 您的 PCoE 定义是什么？
 - b. 谁发起创建 CoE？
 - c. 您是否有目前未进行性能测试的项目，为什么？
 - d. 您是否使用标准化工具进行负载测试、监控、分析和诊断的？如果是，使用哪些工具？
 - e. 您是否有未使用标准化工具执行的项目？
 - f. 您开始涉及开发生命周期的哪个阶段？
 - g. 您是否有通用的性能测试方法？有一种以上这样的方法吗？
 - h. 组织是否将在发布产品进入生产前验证性能视为一项强制要求？
5. 是否有处理所有 LOB 的标准方法，还是方法因 LOB 而异？如果是后者，主要的区别在哪里？
6. 您为各个 LOB 提供哪些服务（性能验证、性能优化、产能计划和工作负载分析）？
7. PCoE 和 LOB 之间是否存在任何正式关系？您是否具有或执行 SLA？
8. 执行性能验证时，您需要提供正式报告或制成品吗？

9. 性能项目交付后会怎么办？会与开发团队进行交流吗？
10. 运营 CoE 时，从技术和软管理角度而言会面临哪些主要困难？
11. 未来 2-3 年您在 CoE 管理和运营方面的目标是什么？