



最终用户计算

权威指南

以混合多云方式部署虚拟桌面和应用

目录

- 2 作者**
- 2 关于本电子书**
- 3 前言**
- 4 EUC 部署架构**
 - 传统的本地代理
 - 云代理
 - 混合部署
 - 用例
- 9 架构原则**
 - 进入点
 - 可扩展性
 - 高性能
 - 容量
 - 监控
- 14 构造组件**
 - 虚拟化管理程序
- 17 基础架构方案**
 - 自建
 - 融合基础架构
 - 超融合基础架构
- 23 存储要求**
- 26 存储类型**
 - 传统的分层架构
 - 全闪存
 - 混合闪存
- 28 GPU**
 - 专用 GPU
 - 共享 GPU
 - GRID 许可
- 30 文件服务**
 - Nutanix 文件服务
- 31 确定计算规模**
 - 物理内存
 - CPU 时钟速度计算
 - CPU 比率
- 35 虚拟化集群设计**

作者

Brian Suhr 在企业基础架构设计、实施和管理方面拥有超过二十年的 IT 工作经验。他致力于在各种虚拟化、数据中心和基于云的项目中提供架构和工程方面的专业知识，同时与高效的技术团队在全球性的环境中开展合作。Brian 是 DataCenter-Zombie 和 VirtualizeTips 博客的独立作者，他专注于虚拟化、自动化、基础架构方面的内容，并热衷于宣传有利于技术社区的产品和服务。Brian 的 Twitter 账号为：@bsuhr

关于本电子书

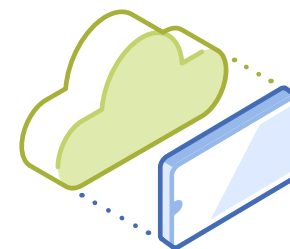
本书的主题是 VDI 和最终用户计算 (EUC) 环境的基础架构设计。本书中的内容改编自《设计最终用户计算解决方案并搭建架构》(Architecting and Designing End User Computing Solutions) (可在亚马逊上购买) 中以基础架构为重点的章节。

简介

为交付 EUC 服务和应用选择了合适的策略和软件提供商之后，应用和桌面虚拟化项目的下一个重大决策是部署架构和基础架构的选择。

控制平面在哪里运行，以及如何运行，这个决定越来越重要，因为选择越来越多。基础架构是构建服务的基础。与水和电类似，这是我们赖以生存的东西 — 打开水龙头就应该出水，按下开关就应该通电。基础架构可以作为一种服务租用，否则必须有人来维护。

如果没有一个稳定、高度可用、高性能的控制平面和基础架构，IT 部门在 EUC 项目的部署和运营阶段会面临各种挑战。虽然基础架构起着至关重要的作用，但 IT 部门不应花费大量的时间来部署和维护它 — 这种情况屡见不鲜。合格的基础架构应该始终顺畅运行，让架构师和工程师有时间专注于提供 EUC 服务和应用。



EUC 部署架构

最佳的部署架构取决于组织的要求，而这一决定反过来又影响到组织对 EUC 控制平面的选择。下文列出了不同类型的控制平面和 EUC 中介性代理 (broker) 的部署选项。顾名思义，控制平面用于处理置备、功耗和中介性代理，是管理员使用的主要界面。控制平面通常也充当 API 接口的功能。

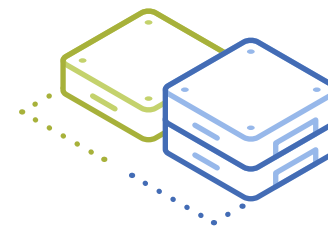
EUC 控制平面通常为用户与应用或桌面的连接提供中介性代理。对于应用，它们通过应用表示（通常是基于 RDSH）进行中介性代理，而对于桌面，它们可以使用托管共享桌面 (HSD) 或虚拟桌面基础架构 (VDI) 会话。

让我们了解一下三种主要的选择，以及一些使用案例。

传统的本地代理

对于这种 EUC 中介性代理 (broker) 方案，组织通常以每用户为基础购买许可，并安装在他们的数据中心内。该类别下有若干个提供商和产品，其中 Citrix Virtual Apps and Desktops (CVAD) 和 VMware Horizon 是迄今为止使用最广泛的两个。

在构建本地 EUC 部署时，架构和部署由组织负责，也可以交给外包解决，包括确定要部署的 broker 代理软件的所有组件，如控制器服务器、数据库服务器、许可服务器、边缘安全服务器或设备，以及任何其他所需的支持服务。作为部署这些项目的一部分，您还要确定每个服务有哪些高可用性 (HA) 选项，并选择对您的设计要求有意义的选项。



大多数这些服务通常还有负载和规模方面需要考虑。您需要适当地确定每个服务可以处理多少个连接，从而确定您需要多少台控制器服务器，例如，5000 个带有 HA 选项的连接。然后，您可以决定是否要为这 5000 个连接预先部署所有的控制器服务器，或者在扩展部署时再添加。如果您的扩展计划远不止这个数字，您可以使用这些信息来继续扩展不同的服务，并确保始终符合最佳实践和支持的数量上限。

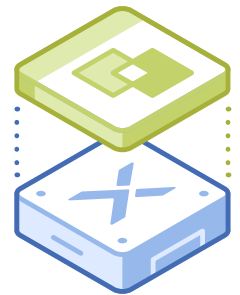
下一个考虑因素是操作，在这里有必要了解如何修补和升级部署的 EUC 中介性代理中的所有服务。除了我们已经讨论过的服务（通常包含在所有部署的服务器虚拟机中），您还需要维护自主性代理 (agent) 和客户端。agent 代理包含在镜像中，镜像用于部署用户可以连接到的应用服务器和桌面池。如何更新代理取决于您如何配置这些池。客户端位于用户用来连接这些服务的端点上，这些端点根据端点的风格有很大的不同。

抛开名称不谈，您实际上可以在公有云中部署一个传统的本地 broker 代理。在绝大多数情况下，在本地部署 broker 代理更有意义。

云代理

云代理 (cloud broker) 在典型的软件即服务 (SaaS) 模式中作为一项服务提供，最常被称为桌面即服务 (DaaS) 产品。该类别下也有许多提供商和产品。公有云提供商有自己的产品，然而，由于 Citrix 是 EUC 市场当之无愧的领导者，要想体验 DaaS 的好处，Citrix Virtual Apps and Desktops Service (CVADS) 无疑是个不错的选择。大多数产品提供的功能与传统本地厂商提供的功能不相上下，但这并不是说它们自身没有优势和局限性。

软件提供商、云提供商和服务提供商都拥有 DaaS 产品。除了 VDI 和应用表示等主打功能时，它们所提供的服务差别很大，这就是为什么在做决定时，了解您的用例和用例的要求很重要。这些要求能够为您指引方向；如果您青睐的选项不符合某些要求，您必须确定，是否可以接受在没有这些要求的情况下工作，直到要求得到满足。

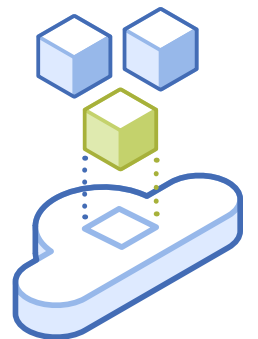


作为一项订阅服务，这些产品有几个值得一提的特点。首先，您通常按每用户每月为单位支付服务费用，因此成本很容易计算和跟踪。该成本也属于运营成本预算，这对一些公司来说是一项好处，因为他们被迫转向了云消费模式。您可以与这些 DaaS 产品签订长期协议，这些协议可能反过来提供更优惠的价格，以换取更长的签约期限；不过，订阅后，您可以选择在需求减少或消失时退出订阅。在传统的许可方式下，企业通常会购买许可，然后支付永久的支持费用。

鉴于它是一种服务，因此组织可以按需消费。您不必操心代理层的所有设计、部署和扩展方面。另外，您通常在 DaaS 产品旁边的公有云中运行应用和桌面，所以也不必管理运行这些应用和桌面的基础架构，因此您也不必担心这些代理层的监控和升级。这些 DaaS 产品可以使您的项目更快启动，而且从操作方面来说，这也会让您更省力。

虽然 DaaS 服务涵盖了代理和基础架构层，但仍有操作任务需要执行。包括镜像构建和更新、应用的安装和更新、用户数据、VPN 等等。

云提供的另一个好处是能够使用世界各地的数据中心，在一组用户附近提供应用或桌面池。这一好处并不总是奏效。例如，如果用户对位置遥远的数据中心的数据或特定的应用有严格的依赖，这种能力就毫无用武之地。

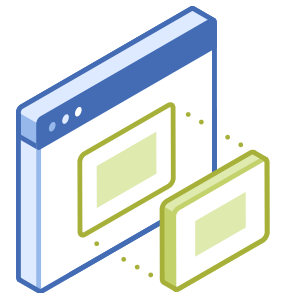


混合部署

我们在前面已经讨论了 EUC 中介性代理的不同架构，现在让我们来看看如何部署。尽管有人预测所有的工作负载都将迁移到公有云，但很明显，大多数的部署仍然位于本地，而且大多数的组织正在改用混合架构。在我们的讨论中，混合架构只是指混合使用本地部署和公有云。每种方式所占的百分比各不相同，具体取决于许多因素。当涉及到 EUC 领域的混合时，有一些方法既适用于传统环境，也适用于 DaaS 产品。让我们首先看看各种架构是否适合混合环境，然后深入研究一些使用案例。

DaaS 解决方案是提供混合产品最常见的方法，并不是所有的 DaaS 提供商都能提供在混合架构中部署的能力。对于 DaaS 的混合部署，控制平面通常仍然在云中，这类产品可以控制和管理私有资源池，方法通常是在您的本地集群上部署一个本地虚拟机，作为本地连接器使用。通过这些云连接器虚拟机，DaaS 供应商可以置备虚拟机，控制其功耗状态，并以中介性代理的方式连接到这些虚拟机。连接器虚拟机很容易设置，可以成对部署，以提高可用性。这就是 Citrix CVADS、Horizon Cloud 和 Nutanix Frame 支持的架构。

总的来说，DaaS 的混合部署在设计、部署和操作上仍然远没有传统方法复杂。您仍然要负责管理本地虚拟机所运行的基础架构，但您不必管理 broker 代理层。



用例

让我们来了解一下适合混合部署的不同使用场景和用例。如果使用得当，混合型设计可以在成本和功能方面提供很大的灵活性。大多数用例分为三种不同的使用场景。

- **灾难恢复 (DR)**。该使用场景可能适合大多数需要业务连续性 (BC) 的传统本地部署。历史上，灾难恢复涉及建立或租用一个辅助站点，用于部署 EUC 资源以支持故障切换。公有云为传统方法提供了一种非常有吸引力的方案，因为您可以预留一些云容量，然后在发生灾难时将其突增至全容量。在这种方法中，您要付费运行 broker 代理基础架构虚拟机和一个小型的桌面池，并为复制的用户数据和配置文件提供存储容量。然后，一旦发生故障，您可以迅速根据需求将该桌面池扩展到任何规模，并将用户引导至该桌面池，让他们可以重新开始工作。当一切大功告成后，您可以缩减到稳定状态的规模，从而再次降低成本。
- **突增**。有一些用例或项目只在短时间内需要消耗资源，为它们预留本地空间可能没有意义。在这种情况下，您可以在（通常是）公有云中启用这些资源，当需求消失时再将其销毁。突增可以有效满足很多使用场景的需求。学生实验室是一个常见的用例，因为需求是季节性的，取决于学生的日程安排。学生也可能有 GPU 使用需求，而这在本地无法提供。特殊项目和临时性的工作也是常见的突增用例。
- **真正的混合部署**。大多数其他用例都属于这种形式，因为它们不是临时需求，这意味着您经常在本地上运行一些工作负载，在云中运行另一些工作负载，并根据对组织和设计有意义的参数来决定部署位置。例如，您可能有一个需要使用 GPU 的用例，而这在本地无法提供；您可能需要在亚太地区部署，而不是让本地用户绕远访问北美站点；或者您可能只是在本地环境中没有容量可用了。



架构原则

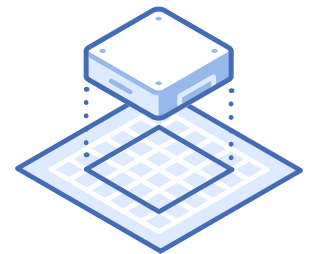
在 EUC 基础架构的设计过程中，应考虑到几个重要的因素。根据这些因素和组织的要求，有助于更好地构思架构的方案。在评估 EUC 项目的架构选择和提供商选择时，应该考虑以下因素：

- 进入点
- 可扩展性
- 性能
- 监控
- 容量

进入点

基础架构的进入或起点往往是一个项目的决定性因素。该因素决定了，基于不同的起点规模，组织需要何种规模的基础架构，以及需要花费多少成本来启动应用或桌面的虚拟化和交付部署。

如果项目计划在全面部署后覆盖 10,000 个用户，起始部署阶段为 5,000 个用户，那么组织可能不太会在意起始成本。因为，根据所选择的基础架构的类型，每个用户的成本可能在部署了几千个用户之后才有意义。



另一方面，如果一个组织要部署 10,000 个用户，但打算仅从 500 个用户开始，并在项目期限内以稳定的速度逐渐增加用户数量。他们将密切关注 500 个用户的初始基础架构部署的成本，而不是扩大初始规模。在这一规模下，每用户成本的发展曲线可能随着环境的扩展而保持稳定，也可能在刚开始时看起来非常倾斜，因为初始基础架构的成本较高。

虽然每用户成本非常模糊，几乎与决定基础架构成本的因素无关，但当您试图向企业推销项目或向领导证明您的基础架构选择是合理的时，他们就会问这个问题。如果您选择了一个前期用户成本较高的方案，就需要做好准备，解释其中的细节。要评估那些您认为更适合您的环境的解决方案。否则就要好好解释一番您的决定，证明花费这些成本将带来何种成效。图 1 这的例子分别对应这两种情况。

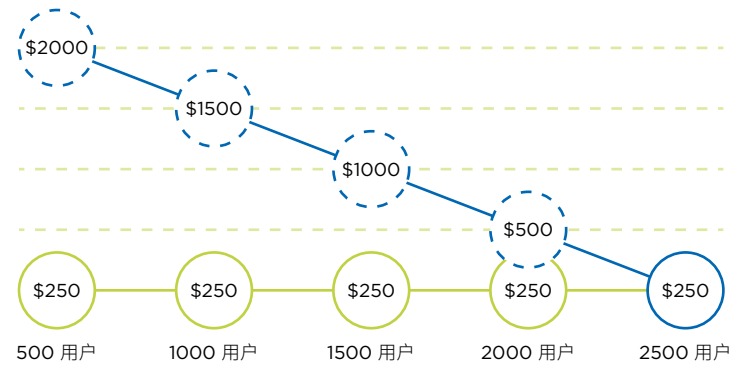


图 1:
每桌面进入点



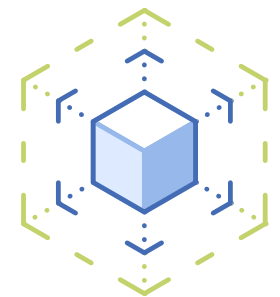
可扩展性

架构的可扩展性是评估项目可行性的一个重要因素。架构师需要了解不同方案的起始规模选择；这又回到了刚才讲到的进入点问题。该方案是否可以轻松让设计从较小的规模起步，或者组织是否需要购买更多的基础架构资源才能满足项目的起始规模——并且在项目发展到一定程度之前无法利用所有的资源？

除了知道方案的起始规模有多小，还有一点也同样重要，那就是方案可以扩展到的最大规模有多大。如果组织希望从 500 个用户起步，并且之后能够扩展到 10,000 个用户，那么该方案能否同时满足？组织是否会对可扩展的最低和最高点感到满意，或者对两者都很满意？

可扩展性问题不仅仅关乎存储。它也适用于计算、网络，以及设计中可能用到的其他层。如果对计算层的配置进行调整，让每台主机服务器达到较小的虚拟机密度，这在扩展时将对不同的设计选择有何影响？例如，如果最初的主机设计从每台主机 128GB 内存开始，而最终扩展到 256GB 或更大的容量，那么组织就需要确保使用正确尺寸的 DIMM，以便可以在未来扩展配置。如果为了节约成本而做出错误的选择，由于尺寸限制，将会影响到虚拟机密度，或者从长远来看，会由于 DIMM 无法被重复使用而花费更多金钱。

架构师应该重点关注解决方案如何能够从小规模起步，以及是否能够扩展到最高点。但也不能忽视这中间的所有点，因为根据扩展部署的方式，在开始和结束之间可能还有许多扩展点。理想的做法是寻找相应的方案，让设计可以轻松根据项目需求，按用户数量成批扩展，同时还要符合部署的时间要求和能力。对于一个项目来说，理想的扩展批次大小可能是以 500-1000 个用户为增量。但是，如果所选择的架构方案的扩展规模大于这个数字，要了解这对成本和部署有什么影响。



高性能

以最终用户体验来衡量的 EUC 性能始终是一个首要考虑因素。您选择的架构必须能够满足项目任何阶段的要求。对于一些解决方案来说，这可能比较棘手。如果缩小解决方案的规模，以满足用户的最低起始需求，那么如果不能线性扩展，最终可能会牺牲性能。架构师不希望为了满足最低起始要求而在架构上做出妥协，这可能会影响解决方案的整体最大性能选项。如果您在前期花时间做出了正确的决定，就可以避免以后出现问题。

EUC 解决方案的设计通常会有许多不同的性能要求。要选择一个足够灵活的架构方案，以满足单一选择中的所有性能要求。无论设计将提供多种类型的 EUC 服务，还是只关注应用和桌面虚拟化，都必须考虑到多种性能需求。了解每个选择如何能够或不能满足个别的性能要求，将在很大程度上影响您的评估和设计过程。

容量

关于容量的讨论与性能类似。在 EUC 设计中，需要满足许多不同的容量要求。解决方案将要求为架构运行服务器虚拟机、桌面虚拟机、应用、用户配置文件和用户数据。设计中的每一层对容量的要求都可能大相径庭。有些层要使用大量的数据，但这些数据通常可以有效去重。其他部分，如用户配置文件和数据，每个用户产生的数据很少，但乘以成千上万的用户，最终的体积既然很大。

过去几年中一个较为突出的问题是容量购买过多或过少，同时还要努力达到所需的性能水平。在设计阶段要密切关注架构的方案，看看它们如何能够提供所需的容量，同时确保最低性能要求也得到满足。该方案不应提供 2-3 倍或更多的容量来满足存储性能要求，也不应大幅增加额外性能以满足容量要求。



理想的解决方案应具有充分的灵活性，以类似的速度扩展性能和容量，所以两者的节奏不会相差太远。

在过去，这方面的问题层出不穷。许多组织由于容量扩展得比性能更快，而使自己陷入性能和容量规划的困境。仅仅因为解决方案有 5TB 的自由空间，并不意味着能够再扩展 500 个用户。这种情况可能会导致性能大打折扣。管理员和 IT 领导层如果对解决方案的扩展方式没有深入的了解，就会落入这个陷阱。

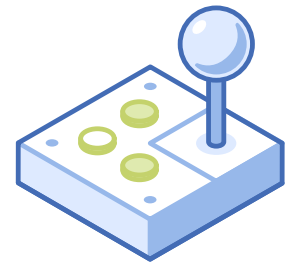
监控

监控非常重要，但经常被忽视。当涉及到监控 EUC 环境中的基础架构时，管理员通常关注性能方面。他们需要能够了解哪些功能正常运行，以及有哪些问题没有得到解决。

监控应该易于使用，同时也要能够提供大量的详细信息。对于许多制造商来说，情况并非如此，所以应该仔细观察每个方案的监控体验。

另一个要求是能够在虚拟机层面提供性能监控。不幸的是，大多数基础架构提供商仍然不能提供这种水平的虚拟化环境可见性。一流的基础架构性能监控应该让管理员能够快速查看存储层，并确定存储性能问题是全局性的，还是仅限于某个主机、一组虚拟机或只是某个单一的虚拟机出现了故障。

通过在虚拟机层面管理存储性能，可以使用类似的方法在主机层面管理虚拟机的 CPU 和内存性能。管理员需要知道某个虚拟机是否临时使用了额外的性能，或者它消耗的资源是否一直比典型用户更多。这有助于了解什么时候出现了需求高峰，什么时候需要进一步分析以确定问题。



构造组件

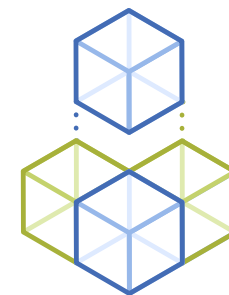
构造组件是一套预定义的基础架构，映射到特定的资源数量或用户数量。这种方法是实现基础架构设计与最终用户计算的最佳方式之一。

通过使用这种方法，可以开发一个架构，提供一个可预测的成本、性能和容量扩展模型。在确定构造组件的尺寸时，要选择您需要以何种增量来扩展用户，以及基础架构的选择如何能与之相适应。例如，组织可能希望以 50 到 100 个用户为增量来扩展用户，但基础架构的选择并不能以这么小的增量来进行扩展。这可能会迫使设计以 500 或 1000 个用户的较大增量进行扩展。如果基础架构的选择以较大的增量进行扩展，我们可以紧密配合这种节奏，也可以接受事实，即基础架构的成本不会像用户部署那样同步增长。这意味着，企业将以 1000 个用户为单位购买基础架构，而只以 50 到 100 个用户为单位进行部署。

当购买大规模的基础架构来部署少量用户时，确实会让虚拟桌面或用户会话的成本看起来十分昂贵。而当组织按照计划将全部用户部署完毕时，成本便会回到合理水平。

构造组件式的架构在任何设计项目中都有用武之地，但 EUC 的部署总是有共同的用户群和用例，它们的特征相似，并按群组部署。继续以 100 个用户的规模为例，通过了解 100 个用户的资源需求，可以确保这组基础架构能够提供这些用户所需要的一切。

如果每个用户在稳定状态下需要 15 IOPS 和 30GB 的存储容量，以及 2GB 的内存和 200MHZ 的 CPU，那么架构师就知道构造组件必须提供 1500 IOPS、3TB 的容量、200GB 的内存和 20GHZ 的 CPU。架构师可以设计构造组件以包含额外的资源，但不能低于这些值。我们也希望避免在每个构造组件中包含太多我们无法利用的额外资源，因为这会造成的浪费。



有了这种方法和精细的设计，我们现在可以以 100 个用户为一组来扩展环境。为此可以采用缓慢、稳定的节奏，并提供可预测的值，以便组织可以围绕部署、性能、容量和成本进行规划。如果组织想要以更快的速度和更大的步伐进行扩展，他们只需一次购买多个构造组件。

最后，事实和证明，构造组件式的方法很有吸引力，因为大多数客户都喜欢从小规模开始部署，然后逐步扩展。这种“从小规模起步，按需支付”的模式使他们能够在前期投入较少的资金，并随着部署规模的增加而积累经验。下一节中，我们将介绍当今不同类型的基础架构，以及它们是否支持构造组件式的方法。

虚拟化管理程序

虚拟化管理程序是基础架构设计中一个很重要的层。它在很大程度上直接影响到解决方案的性能、可用性、恢复能力和可管理性。

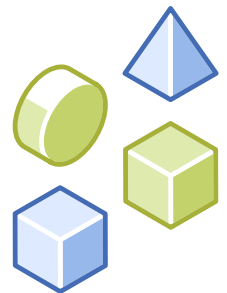
虚拟化管理程序的态势

在当下的虚拟化环境中，有很多虚拟化程序可以供企业选择。其中下述虚拟化程序可以部署 EUC 和 CDI 用例：

- Citrix Hypervisor (XenServer)
- Nutanix AHV
- VMware vSphere (ESXi)
- Microsoft Hyper-V

为什么考虑更换

对于更换虚拟化程序，企业有很多理由，从简化虚拟化程序层的架构和操作，到提高安全性，再到减少厂商锁定和消除成本。



评估标准

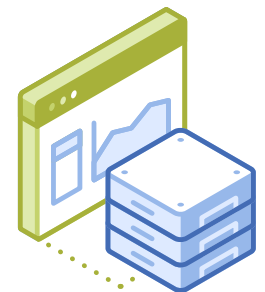
虚拟化管理程序是一种复杂的软件，由数百种可能的功能组成。基本功能包括资源调度、高可用性、虚拟网络和多操作系统兼容性。这些现在都是有实用价值的功能，上述虚拟化管理程序都可以轻松提供。

除此以外，企业应该关注每种方案为企业提供的价值，以及它将如何改变或改善运营。

为此，首先要看一下不同的虚拟化管理程序的设计阶段。将最近的项目或即将到来的项目的设计要求用作解决方案，并为之搭建架构。了解为所选的项目设计一个解决方案所需的工作量。要留意一下，由于设计选择的复杂性，这个阶段是否耗费了几天或几周的精神力来设计解决方案的虚拟化管理程序层。理想情况下，虚拟化管理程序应该提供所有需要的特性和功能，同时将设计阶段简化为几个清晰而简单的选择。

接下来一定要了解部署工作的具体流程。根据评估中选择的设计，确定部署解决方案需要做哪些工作，以及这与基本部署有什么不同（如果有的话）。理想情况下，就像设计阶段一样，在部署阶段工程师需要花费的精神力应该是很少的，并且流程应该是高度自动化的。部署一个新的集群不应该需要忙活几天到几周。

最后，要仔细看看虚拟化管理程序的操作，以便了解这与您目前的工作有何不同。修补和升级虚拟化管理程序一向需要耗费较多的精力。为了在操作和升级的可靠性方面简化这一过程，有没有哪种虚拟化管理程序能够提供任何优势？除了升级之外，还需要对虚拟机进行哪些日常管理工作？这是否可以通过一个简单的界面来完成？



基础架构方案

目前有三种主要的应用和桌面虚拟化架构选择，也就是广义的 EUC 解决方案。分别是自建 (BYO)、融合基础架构 (CI) 和超融合基础架构 (HCI)。

自建

顾名思义，自建基础架构指的是架构师或团队自主选择他们喜欢的或认为是最好的产品。选择该方案会导致前期规划和研究时间大幅增加，因为团队必须单独评估每个产品，以及它们如何可能或不可能一起工作。

使用这种方案，也让我们可以选择和遵循提供商为正在构建的解决方案类型所发布的参考架构。这些参考架构通常由一个提供商发布，重点面向他们自己的产品。这些 DIY 参考架构可以节省时间并减少一些风险，但它们并不总是适用于您的设计要求、用例和环境。

至少，如果 BYO 方案面向基于 EUC 的设计，其中将包含计算和存储资源。您可能能够使用现有的网络连接，所以网络连接可能不是该方案的组成部分。图 2 中的例子展示了 BYO 方案的各个部分。由于扩展较为灵活，成本的可预测性很高；唯一的例外是在存储方面。根据设计的最大规模和所选择的存储，您可能需要多个存储阵列或设备。当您扩展存储并需要添加新的阵列或设备时，成本将会激增。



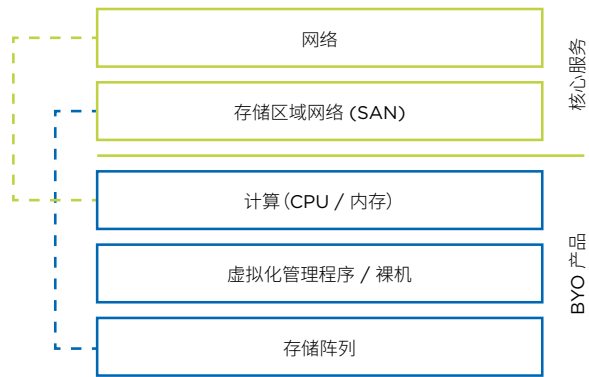
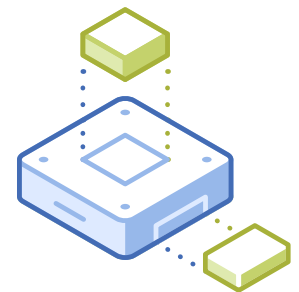


图 2:
自建 (BYO) 基础架构

无论何时，如果您要组装来自同一个提供商或多个提供商的产品，但事先没有经验，那么就会产生额外的风险。在购买并以架构方式部署实际的基础架构之前，解决方案的性能和可靠性具有一定程度的不确定性。

如果组织能够接受未知因素和额外的风险，自建方案确实可以最大限度地提高灵活性。只要不同的产品可以协同运行，组织就能够自由选择各种厂商和产品，因此，您可以继续与表现良好的厂商合作，也可以从其他领域的新厂商那里购买产品。

BYO 方案能够独立地扩展计算和存储资源。对扩展方式或最大规模的唯一限制在于对选择个别产品的限制。由于产品是单独购买的，所以产品扩展的最小单位或固定数量没有限制。这也体现了前面所介绍的“构造组件”方法所具有的灵活性。



融合基础架构

融合基础架构 (CI) 是 2010 年左右问世的一种架构。融合基础架构提供的产品通常也可能被选为自建方案的一部分, 并将它们打包成一个产品化的解决方案。这意味着一个 CI 提供商将在他们的产品中纳入计算机、存储和网络。通常情况下, 大多数 CI 产品包含来自多个厂商的产品, 并作为单一产品的一部分, 或者一个厂商也可以通过他们自己的产品线涵盖 CI 产品的所有层次。图 3 中是一个融合基础架构方案的简单例子。

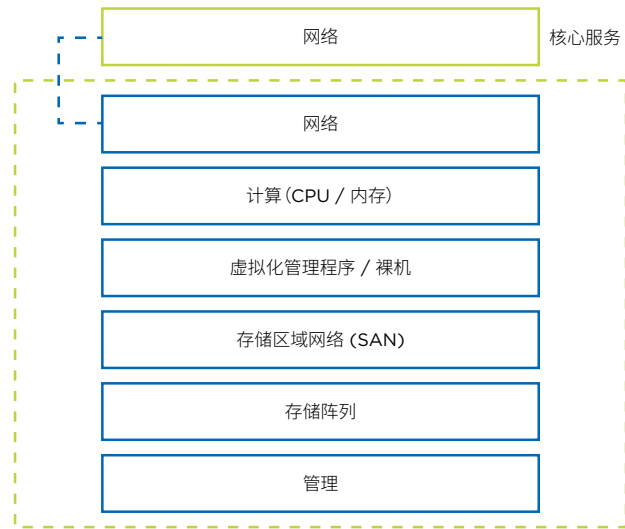
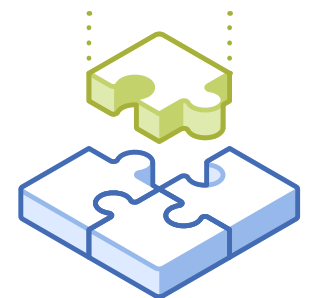


图 3:
融合基础架构

融合基础架构产品将使您能够购买熟悉的产品, 并将其打包成一个单一的解决方案。可以将它看成一个可以作为产品购买的参考架构。根据评估的具体 CI 产品, 与在自建方案中单独购买产品相比, 此类产品可能提供也可能不提供任何额外的融合。



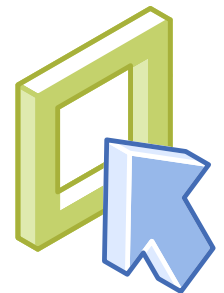
通常情况下，大多数 CI 厂商和产品支持通过单一的产品型号购买所有基础架构组件。CI 厂商应该能够为整个 CI 解决方案提供相同的电话支持，这意味着 CI 厂商可以支持解决方案中的所有产品。这是一个额外的好处，因为它允许客户在故障排除过程中不需要与多个厂商合作。

在大多数 CI 产品中，解决方案中提供的产品数量有限。这使得 CI 供应商能够预先测试和验证所有的部件和组件，确保它们能够一起正常工作，从而消除了 BYO 方案中的大部分风险。

即使 CI 产品已经在市场上销售了几年，此类产品管理起来依然比较麻烦。CI 产品中包含与 BYO 产品相同的产品，组织通常会以类似的、分散的方式管理这两种产品。这种方案可能会融合购买或一些产品，但它通常不会融合解决方案的日常操作管理。

融合基础架构产品应该能够独立扩展其中的资源。这意味着，您可以只增加计算资源，尽管扩展的增量可能会很小。在 CI 产品中可以扩展的另一个资源是存储，这在很大程度上取决于存储解决方案的类型 — 它是 CI 产品一部分。融合基础架构产品有最大规模限制，这意味着它可以支持的服务器数量是有上限的，基于所包含的存储阵列，存储容量也是有上限的。

CI 产品的扩展限制通常是相当大的，但在某些时候，随着 CI 产品内资源的不断扩展，将达到最大限度。此时要想继续扩展，就需要购买一个额外的 CI 产品。在扩展过程的不同阶段，这将导致基础架构成本激增 — 具体取决于设计的最大规模是多少。



超融合基础架构

超融合架构是在融合基础架构问世大约一年之后推出的。真正的超融合架构是通过将计算资源、存储资源和管理层融合到一个单一产品中来实现的。可以在纯软件（SWO）模式或在包含专用硬件的设备模式中部署超融合解决方案。

通过将硬件设备作为产品的一部分，厂商现在可以将基础架构的管理和其他资源的融合一起纳入产品中。图 4 是超融合基础架构方案的一个简单例子。

纯软件架构可以在硬件层提供很大的灵活性，允许用户自主选择平台进行部署。在考虑 SWO 方案时，它们依然应该提供类似于设备的体验。这与具有松散耦合的硬件兼容性列表的产品不同，这些产品只包含了最低程度的测试。

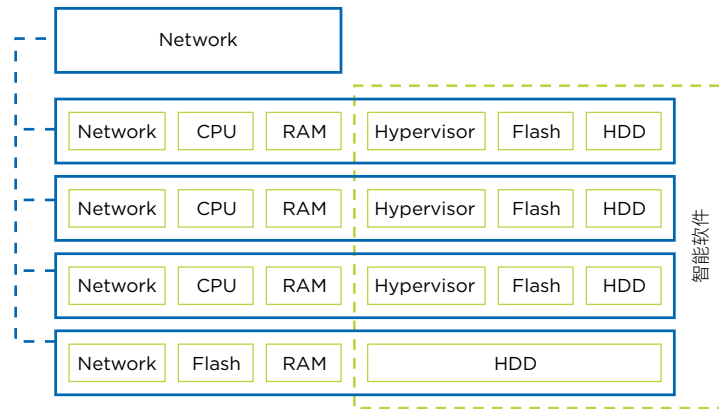
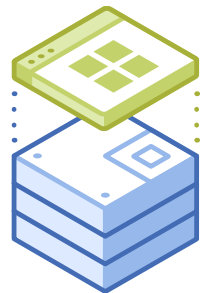


图 4：
超融合基础架构



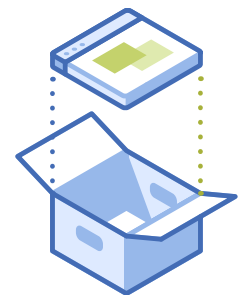
安装简单 — 领先的 HCI 产品应该使用高度自动化的流程，在几分钟或几小时内安装完节点，而不是耗费几天或几周。

易于扩展 — HCI 产品应易于横向或纵向扩展。应该可以通过管理界面，轻松而快速地在环境中增加新的节点。

现代化管理 — 现代化的管理界面必须以虚拟机 (VM) 作为管理点。管理员必须能够查看虚拟机的性能、每台虚拟机消耗了多少资源、是否有任何事件或错误，并能够轻松地根据虚拟机拉取报告。

可扩展性 — 您必须能够轻松地将基础架构与解决方案的其他部分集成，并以编程方式加以控制。这需要 HCI 产品提供一个 API，或者提供其他方法，如 PowerShell cmdlets。有了 API，您就可以自动实现产品间的通信和控制，进一步减少工作量，并提高环境的准确性。

在 HCI 的所有优势中，我们有意忽略了性能，因为我们都希望现代化的混合解决方案或基于闪存的解决方案能提供良好的性能。HCI 旨在创建一个简单而高效的基础架构层。它使团队不必再花时间手动操作按钮，而是能够在自动化或应用层面为业务提供额外的价值。



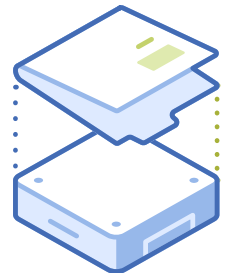
存储要求

任何 EUC 设计都具有一些不同的存储资源要求。需要考虑到基于服务器的虚拟机、用户数据和虚拟桌面基础架构 (VDI)。VDI 的存储要求是整个环境中最苛刻的，也是导致大多数 VDI 项目部署失败或体验不佳的原因。

为此，在本电子书的存储部分，我们主要关注解决方案中 VDI 服务的需求。每个虚拟桌面的需求往往看起来都很小，微不足道，但当随着存储规模的扩大而将许多虚拟桌面组合成大规模群组时，未能针对这些需求而进行合理设计的存储将无法随之而来的性能需求。

如果每个虚拟桌面平均达到 15 IOPS，预计有 2000 个并发用户，这相当于 30,000 IOPS。这个数字是相当大的，可能会使一般的存储阵列不堪重负。但是，我们不能简单地设计存储解决方案来满足环境的平均 I/O 工作负载，设计必须考虑到需求高峰，包括桌面启动和用户登录事件。

虚拟桌面工作负载的 I/O 需求峰值非常尖锐，这使得它们与一般企业数据中心内的其他类型的工作负载非常不同。例如，在一个会话中第一次打 Outlook 这样的应用，可以为该用户会话生成超过 1000 IOPS。这远远超出了前面讨论的平均 15 IOPS。图 5 中的例子展示了不同的应用 IOP 影响。



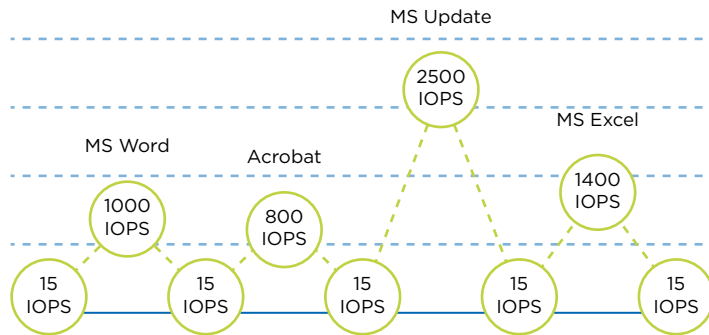
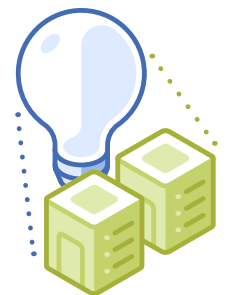


图 5:
VDI IOPS

其他部署和操作项目，如修补和环境刷新，也会产生很高的 IOPS 峰值，如果不加以考虑和做出相应的规划，将会影响性能。如果我们又部署了 50 个虚拟桌面，这会产生一个很高的 I/O 峰值。由于这些原因，存储架构的设计必须能够适应来自维护操作的峰值 IOPS。

有许多方法可以构建具有完全克隆或共享镜像表示的 VDI 解决方案，每种方法都会对容量和性能方面的存储需求产生不同的影响。由于完全克隆会消耗额外的容量和存储，重复数据删除非常重要的。完全克隆也必须进行独立修补，这将增加进行这些操作时的 I/O。

Citrix 用 MCS 或 PVS 提供的共享映像方法，以及 VMware 用链接克隆提供的共享映像方法，都带来了不同的 I/O 挑战。从本质上讲，这些共享映像方法需要的存储容量较少，因为父映像是共享的，每个虚拟桌面只为其独特的数据消耗较少的空间。共享镜像与典型的虚拟机有不同的性能要求。这个镜像现在被成百上千的虚拟桌面使用，必须能够生成大量的 IOPS。如果共享镜像是一个瓶颈，所有使用它的虚拟桌面都会受到负面影响，用户体验会很差。



考虑到峰值和不同类型的应用桌面虚拟化架构，我们必须选择和设计一个能够满足环境的峰值启动、登录和稳定状态需求的存储解决方案。为了了解设计的存储要求，组织应该对现有的物理 PC 环境进行桌面评估。该评估将从用户群中收集真实的性能和容量细节，从而应用到设计计算中。

关于应用/桌面虚拟化相关的存储需求，最后一点是，除了在 I/O 方面难以预测之外，桌面工作负载的写入操作非常频繁。与许多服务器工作负载不同 — 它们主要是读取数据并将其提供给用户，台式机通常花更多的时间写入磁盘。对存储阵列来说，写入比读取更加密集。一个典型的服务器工作负载可能包括 80% 的读取和 20% 的写入，而稳定状态下虚拟桌面工作负载的情况可能恰恰相反。当评估您的存储选择时，一定要密切关注存储解决方案是如何缓冲和提交写入操作的，而不是关注一些空洞的承诺，例如在缓存经常读取的块方面做得“非常好”。



存储类型

有许多不同类型的存储。目前，市场上的主流存储方案包括传统的分层存储阵列、混合闪存阵列和全闪存阵列。每种方案都通过不同的方法来为工作负载提供性能和容量。在每一种方案中，厂商会采取不同的方法来打造他们的产品，下面让我们简要了解一下各个存储类型。

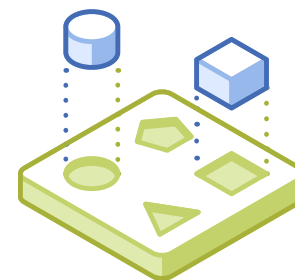
传统的分层架构

此类存储是传统的企业阵列，在过去 10-20 年里一直用于基于服务器的工作负载。它们通常基于双控制器的架构，在过去的十年中经过改造，可以在架构中包含多个层级的性能和容量磁盘。提供不同层级的磁盘是为了满足分散式工作负载的容量和性能需求。此类存储有两种选择：可以通过为工作负载创建专用的高性能磁盘池来设计性能，但这种方法可能成本高昂，且具有限制性。另一个选择是尝试利用添加到该架构中的分层结构，要求阵列根据需求提升或降低数据块的层级。这种自动分层架构的问题在于，对于 VDI 工作负载来说，做出这些决定往往需要耗费太长的时间。

全闪存

全闪存存储阵列是完全由基于闪存的存储组成的。在这些存储阵列中，有许多不同类型的闪存可供使用。现代全闪存阵列的设计是为了利用闪存的特性，也就是说，操作系统和文件系统在设计时都考虑到了闪存。一些产品采用了传统的阵列设计，并简单地用全闪存取代了旋转的磁盘。虽然这仍然要比过去的传统磁盘要快，但最终的产品并不是为这个目的而设计的。

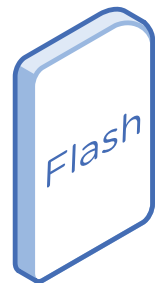
全闪存存储阵列的速度非常快，产品性能仅分为一个级别。为了确保全闪存存储阵列也能以合理的价格提供设计所需的容量，您应该寻找能提供重复数据删除和压缩功能的阵列。虽然几乎每一种现代全闪存阵列都比它们的传统同类产品更容易管理，但它们并不总是像许多混合闪存产品那样，能够易于管理，并且可以按照每个虚拟机进行管理。



混合闪存

混合存储阵列采用了现代化的架构，旨在有效地使用闪存驱动器和旋转磁盘的组合。厂商在各自的混合存储阵列产品中采取了不同的架构方法来使用容量和性能，但最终的结果是相似的。它们都能够以较小的闪存提供不凡的性能，同时通过在阵列中的大型旋转磁盘上存储数据，还能够提供较大的容量。理想的混合存储架构方案会搭载原生智能功能，根据需求在闪存和磁盘驱动器之间对数据进行自动分层，消除了手动调整的需要和潜在的性能隐患。

最适合现代 VDI 设计的架构是混合闪存和全闪存架构。这些架构能够提供 VDI 环境所需的性能，通常还能提供前面讨论的现代管理体验。VDI 工作负载在本质上是难以预测的，如果您的存储解决方案必须要等待一段时间才能做出存储决策或将块推到缓存层，那么在这之前，性能需求早已消失，体验也会大打折扣。



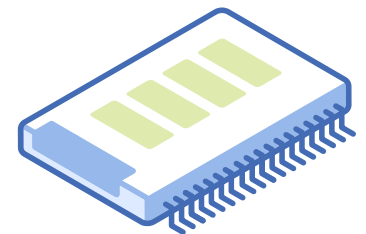
GPU

图形处理单元 (GPU) 是现代桌面体验的一个重要部分。实体台式电脑和笔记本电脑一直以来都会配备一块 GPU，但它并不重要，除非您想玩电子游戏或从事高端的图形密集型工作。虽然这些是事实，但现代操作系统和新的工作负载，如机器学习 (ML) 和人工智能 (AI)，也在利用 GPU 来提高性能。

在 VDI 领域，GPU 通常不是必需的，一般要根据使用情况进行评估，看看是否需要提高性能，比较增加的成本和低要求用例中可以体验到的价值。Windows 10 或观看在线视频属于要求较低的用例，可以从 GPU 中受益，但相较于在没有 GPU 的情况下，以较低的总成本提供可接受的用户体验，GPU 可能无法提供足够的价值。然而，与 CPU 制造一样，GPU 的性价比正在不断提高，在设计 EUC 环境时，在整体评估中也需要评估从 GPU 中获得的价值。

在桌面虚拟化专用显卡制造商当中，英伟达 (NVIDIA) 的知名度最高，性能也最强。AMD 的显卡只适用于某些用例，而且无法提供 NVIDIA 显卡的优化功能。

NVIDIA GRID 显卡支持 GPU 虚拟化技术，使多个用户能够共享一块显卡。GPU 虚拟化不仅支持更高的用户密度，还能在访问虚拟桌面时提供本地性能。NVIDIA GPU 还拥有一个用于 H.264 编码的引擎，可以从 CPU 卸载进程，从而进一步提高硬件上的用户密度。NVIDIA GRID 显卡通常搭载多块 GPU，从而提高了扩展性。



专用 GPU

借助 GPU 直通技术，您可以创建一个具有专用 GPU 的虚拟机。这种配置提供的用户体验能够与搭载高端显卡的“胖客户机”相媲美。然而，将一个 GPU 核心分配给一个单一的虚拟机，无论是托管共享桌面 (SBC) 还是托管私人桌面 (VDI)，都限制了可扩展性。

共享 GPU

GRID 技术让多个虚拟桌面可以共享一个 GPU，同时提供与本地 GPU 相同的用户体验。这种共享通常被称为 vGPU，是 GRID 软件和虚拟化管理程序的一项功能。例如，NVIDIA GRID M10 显卡有四个物理 GPU 核心，每个核心可以托管 16 个用户，因此每个 M10 卡可以托管 64 个用户，每个用户有一个 vGPU 驱动的桌面。

GPU 直接处理虚拟机的图形命令，这意味着用户可以享受高端显卡的好处，而不会因为虚拟化管理程序的干扰而导致性能下降。vGPU 比直通技术的可扩展性更高，因为我们为用户分配了 vGPU 配置文件，从而在同一块显卡上支持更多的用户。

vGPU 配置文件通过 vGPU Manager 提供专用的图形内存，vGPU Manager 为每个桌面分配配置的内存。vSphere 安装包 (VIB) 将 vGPU Manager 安装在虚拟化管理程序上。在 AHV 中，这项任务由 RPM 包管理器完成。每个 VDI 实例都有基于应用需求的预设资源。

GRID 许可

NVIDIA 的独特之处在于，要利用 vGPU 的功能，需要先获得许可。对于用于基于 RDSH 的解决方案的虚拟应用，有不同级别的许可。大多数 VDI 用例中的高级用户和设计师会使用各种高端应用，如 Adobe 应用、工程设计和 CAD 应用。GRID 许可通过命名或并发用户选项提供。

文件服务

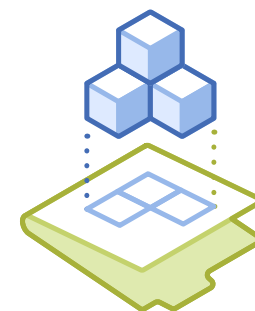
文件服务的应用非常广泛，是 VDI 解决方案架构中的一个重要层次。文件服务传统上是由 NAS 设备或基于 Windows 服务器的设备提供的。该服务具体上是一个 SMB 共享，不同的服务和客户操作系统设备会使用共享或私人访问权限。

在 VDI 设计中，通常对文件服务有几个不同的要求。管理和捕获用户配置文件对于提供持久的用户体验非常重要，大多数配置文件管理解决方案都将数据存储存储在 SMB 共享上。将文件夹作为用户配置文件的一部分重定向到 SMB 共享，这种做法也非常常见。最后，用户创建的数据，如文档、媒体文件和图像，都存储在 SMB 共享的私有或共享文件夹中。

Nutanix 文件服务

Nutanix 提供了 Nutanix Files，将其作为 Nutanix 云平台的一项原生功能。Files 是一个集成的横向扩展文件服务平台，通过 Prism 管理（连同所有其他 Nutanix 功能）。这样就可以实现简单的一键式部署和无间断升级。这种简化使 VDI 团队可以在需要时尽可能全面管理整个解决方案。

Files 提供了一个高度可扩展的架构，只需点击鼠标即可向文件服务虚拟机添加额外的容量，从而增加容量或用户连接。如果需要，Files 实例可以和您的服务器或 VDI 虚拟机在同一个集群中运行，也可在一个专门的集群中运行。为了实现灵活连接，Files 支持 SMB 2.1、SMB 3.0 以及 NFS V3 和 V4 连接。



确定计算规模

可以采用不同的方法来确定设计中计算层的规模。一种是纵向扩展，即使用少量的大型主机来提供资源，而横向扩展则使用大量的小型主机来提供资源。首选的方法介于这两种方法之间，利用 2 个 socket 主机，在不违反整合率的情况下（这是设计的一部分），使这些主机尽可能密集。本电子书重点介绍如何确定 VDI 工作负载的计算资源规模。

在确定设计中计算资源的规模时，有三个主要的计算项目：每台主机的物理内存大小、CPU 时钟速度和 CPU 核心数量，以及 CPU 比率。

物理内存

首先，在 VDI 设计中，我们不应该过度配置内存。过度配置不会带来多少好处，只会导致环境中出现性能问题。

CPU 时钟速度计算

CPU 时钟速度的计算在很大程度上取决于之前桌面评估中收集的详细数据。我们可以通过评估报告，了解用户会话在平均需求水平和需求高峰时使用的 CPU 数量。我们将使用这些信息，以及从评估中得到的内存数据来进行计算。



主机利用率限制

对于其他一些主机和虚拟化集群，建议永远不要超过 80% 的主机利用率，并始终以 N+1 的方式确定集群规模。80% 的主机利用率限制不仅仅适用于应用/桌面虚拟化部署，也适用于在虚拟化管理程序上运行的任何工作负载。根据集群的规模，如果您的主机在运行时超过了 80% 的利用率，就会没有多少空间来应对需求高峰，也可能没有足够的资源来应对主机故障。

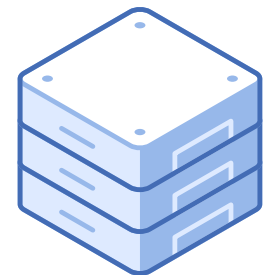
始终以 N+1 的方式确定集群规模

以 N+1 的方式确定集群规模，是为了确保集群中有足够的资源来应对单一的主机故障，以确保所有的虚拟机可以继续运行，出现故障的虚拟机将重新启动，并消除故障。单一主机故障是最常见的恢复能力水平；有一小部分客户需要 N+2 来实现更高的 SLA 要求。

CPU 比率

确定计算规模的最后一个要点是 CPU 比率，即虚拟 CPU 与物理 CPU 的数量之比（vCPU : pCPU）。该比率非常重要，如果太高，最终会出现 CPU 调度问题，极大地影响性能和用户体验。当 vSphere 主机上发生 CPU 调度问题时，CPU 就绪时间会增加，这表示调度器在将所有 vCPU 调度到 pCPU 上时遇到了困难。这意味着 vCPU 将不得不等待，尽管它已经准备就绪。对于在 VMware 集群上被虚拟化的各种工作负载来说，CPU 比率相差很大。通常情况下，服务器和数据库工作负载的 CPU 比率要小得多，而 VDI 工作负载的 CPU 比率则很高。

vCPU 的使用并不是线性计算，也就是说，如果所有的虚拟机都只有一个 vCPU，我们就可以构建一个具有高整合率的主机。当许多虚拟机有两个或更多的 vCPU 时，这将影响计算结果。如果有两倍数量的 vCPU，计算方法不是除以 2 那么简单。事实证明，图 6 中展示的范围适用于适用于的客户部署。进行合成测试的制造商拥有更高的比率。要谨慎进行计算，因为计算结果并不总是适用于实际的设计。



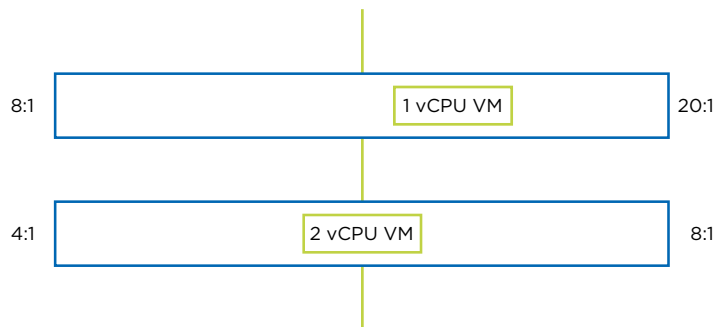
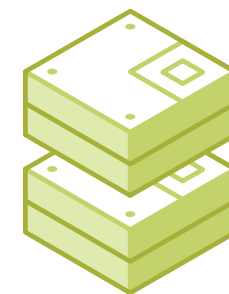


图 6: VDI 的整合在很大程度上取决于配置虚拟桌面时使用的 vCPU 比率。本图中展示的是经过验证的安全范围。

对于单个 vCPU 而言，虚拟桌面正常运行的比率范围是 8:1 和 20:1 之间。这是一个很大的范围，在其中处于哪个位置是由不同的选择所决定的，包括主机的规模，每台主机的虚拟机数量，以及客户对这一数量的满意程度。让我们以一台搭载双 18 核 CPU 的双 socket 主机为例。如果您有适当的内存和足够的时钟速度，便可以容纳 700 多个虚拟机。一般来说，在一台主机上有这么多的虚拟机会让大多数客户敬而远之。因此，在这种情况下有两个选择，第一是添加人为限制，选择较低的密度。如果选择较低的比率，在同一台主机上将容纳 288 个虚拟机。第二个选择是选择内核较少的 CPU，但选择一个中等的比率。如果选择 12 个核心的 CPU，并使用 12:1 的比率，将容纳 288 个虚拟机。要做出这个决定，通常要综合考虑客户反馈、架构师的建议和基础架构的价格。选择不同的物理 CPU 配置，可能会节省大量的成本。

双 vCPU 虚拟桌面的计算方法类似，只是现在我们面对的是双倍的 vCPU 数量。此时正常运行的比率范围是 4:1 和 8:1 之间。一些厂商可能会更高，但具体要取决于实际的客户部署。我们应该根据前面的例子如法炮制，只是 CPU 比率范围有所不同而已。

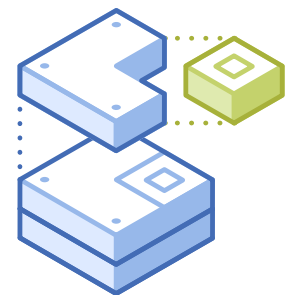


还有一点需要记住，如果您在选择了中等的 CPU 比率，只要环境继续在容忍范围内执行，您就可以向上扩展整合密度。需要注意的一点是，目前市面上的任何其他工具，都无法提供调整 CPU 比率的设置选项。该属性必须在设计中声明，并成为在管理和扩展环境时需要考虑的数据点。就像内存和时钟速度一样，在决定向集群添加更多虚拟机，以及何时向集群添加另一台主机以提供更多资源，也需要计算 CPU 比率。

我们可以收集数据进行人工计算，从而管理 CPU 比率。一些管理员使用 PowerShell 脚本来收集数据，并通过脚本输出 CPU 比率。脚本可以定时每天执行，以确保遵循 CPU 比率，避免将任何集群置于危险之中。

在确定计算规模时，也会用到 RAM 或内存总线频率。在确定内存大小时，经验法则是在预算允许的情况下，以最高密度和最快的总线速度为目标。内存经常面临的挑战是，较慢的内存可能会导致 CPU 周期闲置，等待对 RAM 的读/写事务完成。

将 GPU 纳入 VDI 集群和设计通常也会影响每台主机的用户密度。这与 GPU 显卡的数量和每张显卡搭载的 GPU 数量直接相关，这些显卡可以安装在您的主机中，然后为您的用户选择 vGPU 配置文件。有一个简单的例子：一台主机可以安装两块 GPU 显卡，每块显卡搭载一个 GPU。然后选择一个 vGPU 配置文件，允许每块显卡支持 16 个用户，这意味着该主机上只有 32 个需要使用 GPU 的用户。如果主机上有可用的 CPU 和内存，仍然可以运行非 GPU 虚拟机。使用合适的 vGPU 配置文件对于有效确定规模非常重要。

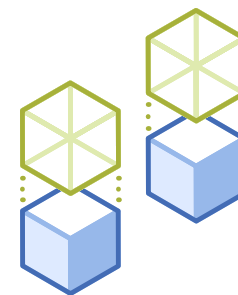


虚拟化集群设计

出于很多原因，在 EUC 设计中要创建不同的虚拟化集群。拥有不同的集群通常是由不同的工作负载和集群规模决定的。本电子书不打算用很多篇幅来讨论这个问题，但我们总结了其他书中和网上的一些话题，并提供了一些建议。

首先，在构建拥有超过几百个用户的 VDI 设计时，必须将虚拟化管理基础架构与 VDI 工作负载分开。这意味着所有的管理服务器、VDI broker 代理、文件服务器、应用管理服务器以及其他任何不属于虚拟桌面的功能都应该在不同的集群中运行。

管理集群是否需要是一个专门用于 EUC 设计的集群，这要取决于环境的规模。如果设计规模比较小，可以在现有的服务器虚拟化集群中运行管理虚拟机。可以将此类虚拟桌面集群的规模扩大到 16-32 台主机。在这一范围内，可以创建一个更大的资源池供虚拟机使用，同时也会推动大多数客户采用比其典型规模更大的集群。最近虚拟化管理程序经过了更新，允许集群扩大到 64 台主机，但许多架构师和客户需要花费一些时间才能适应这么大的规模。如果环境规模大到主机数量超过该范围，就需要部署一个以上的 VDI 集群。



除了环境规模，设计多个虚拟化集群的另一个原因是为了运行不同的工作负载。在 VDI 集群内有不同的工作负载。如果有大量的 1 vCPU 和 2 vCPU 的虚拟桌面，我们应该为每个虚拟桌面设计一个单独的集群。图 7 展示了多集群的设计方法。通过这种方法，我们可以在每个集群中以不同的方式管理 CPU 比率，从而使设计更容易管理。如果要混合使用不同的 CPU 配置，就需要计算新的混合比率，而这只会一切更加混乱。

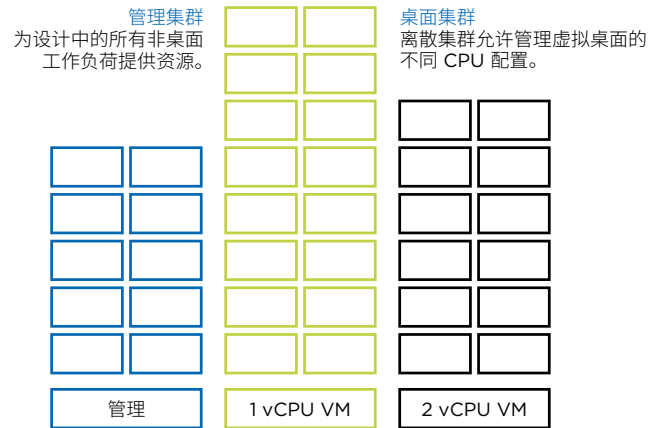
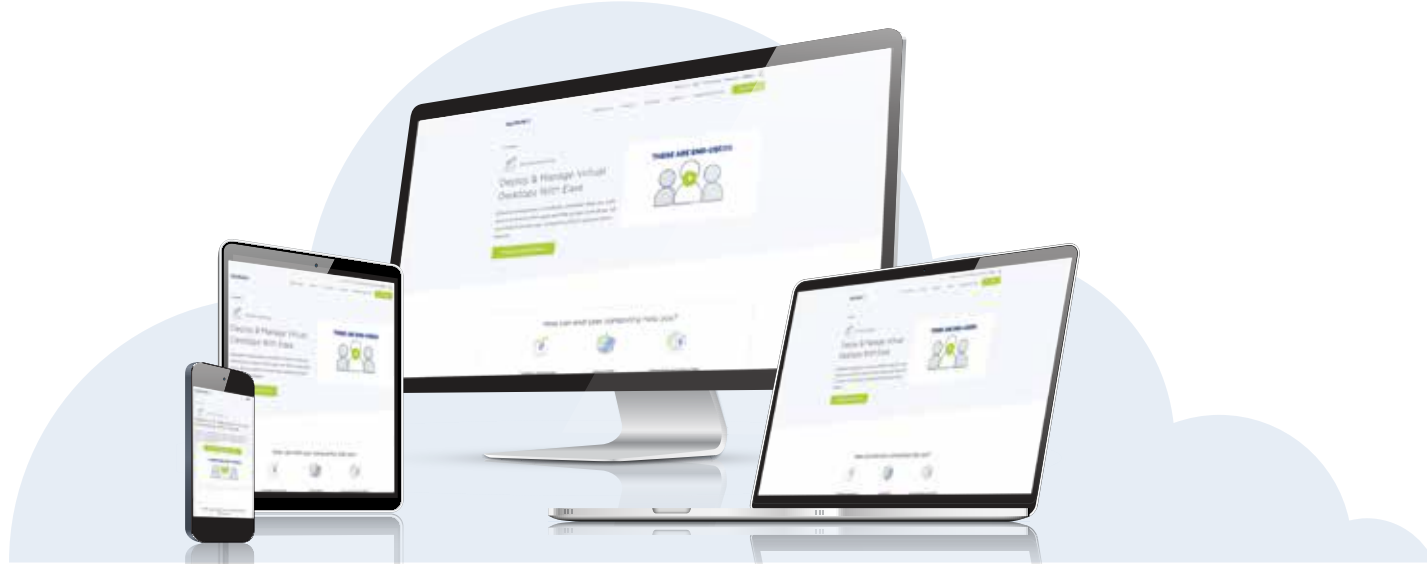


图 7:
管理和桌面集群

使用 GPU 是考虑为 GPU 用户提供专用集群的另一个原因。您可以将 GPU 和非 GPU 用户混合部署在同一个集群中，但这可能会使这些用户的操作和调度变得更加复杂。如果您的 GPU 用户足够多，可以满足一个小型集群的最低要求，那么提供专用集群是值得的。

无论部署在何处， 都能获得出色的 最终用户体验



要了解 Nutanix 的最终用户计算解决方案，请访问 nutanix.cn/euc