



## 白皮书

# 敏捷服务创建和扩展的云原生 NFV 体系架构

编制

Roz Roseboro  
Heavy Reading 高级分析师  
[www.heavyreading.com](http://www.heavyreading.com)

代表



[www.mellanox.com](http://www.mellanox.com)

2016 年 1 月

## 简介

虽然网络功能虚拟化 (NFV) 最初的关注点通常是提升效率和节约成本，但是更成熟的通信服务提供商 (CSP) 已积极地利用 NFV 进行敏捷服务创建和扩展，并最终提高组织的营收和竞争力。为了实现这个目标，虚拟网络功能 (VNF) 供应商与 CSP 合作，确保 VNF 不仅能够保持其功能和性能，而且为云原生，可解决应用层的敏捷性、可靠性和可伸缩性。现在，大多数服务都能够在商用现货 (COTS) 服务器上的虚拟环境中运行。但是，最终状态将是全自动云环境，因为近乎所有 VNF 最终都将从云端提供。

如今，大多数 VNF 都不是“云原生”，必须重新架构才能在云环境中支持自动配置、自动扩展和从错误状态中自动修复。这包括许多不同的方面。一个方面是将 VNF 分解为微服务，这提供了集中常见功能并去除不必要功能的机会，同时可通过软件模块化来提高应用的整体敏捷性、可维护性和弹性，以及更好的功能分发。另一个潜在好处是可以从应用事务处理中分离状态，以便方便扩展。最后，开放的应用编程接口 (API) 和开源有助于在创建服务时轻松集成和推动更快速的创新。

VNF 供应商正在与其 CSP 客户积极合作，以便让他们了解为什么对于敏捷服务创建和扩展来说切换到云原生 VNF 已势在必行，这种转换需要什么以及它对云基础架构本身的影响。较超前的 CSP 都理解重新架构 VNF 的必要性，而其他人则对是否转到云体系架构保持谨慎，因为担心它的弹性。但是通过适当的监控和分析工具，CSP 应该会相信：云的灵活性、敏捷性和成本优势比眼下对性能和可靠性的任何担心更加重要。

此报告的结构如下：

- **第 2 部分**解释了转向敏捷服务创建和扩展时，考虑到 VNF 的多样性和独特性，需要如何改变网络功能的体系架构。它还重点介绍了 CSP 日益增长的意识，即需要重新架构 VNF 以及开放 API 和开源在 NFV 演化中所发挥的作用。
- **第 3 部分**描述了在 VNF 供应商过渡到云端时他们所采取的步骤。它详细说明了为什么将 VNF 分解为微服务对于帮助 CSP 提高服务敏捷性来说非常重要，并且讨论了需要考虑的弊端和影响。
- **第 4 部分**讨论了来自云模型的自动化优势，以及从垂直扩展到水平扩展的过渡。它重点介绍了 VNF 管理器和云编排系统在帮助进行自动扩展上所发挥的作用，以及 VNF 的服务无状态性的适用性。
- **第 5 部分**为 CSP 就如何与其 VNF 供应商合作转到基于云的 NFV 模型提出了建议。

## 网络功能需要改变

### 超越成本优化，进入敏捷服务创建和扩展

CSP 正在努力应对来自互联网视频服务 (OTT) 提供商带来的压力。一方面，基本语音和数据服务的价格逐渐下滑，这直接影响了 CSP 的营收。另一方面，OTT 提供商的增值服务已经从数字转换中获得了大部分价值。这些服务的增长正在驱使网络容量扩充，从而增加了 CSP 的运营成本，并且对他们的利润进一步施压。除了成本控制以外，CSP 也在积极地加快创新步伐，更快地交付和扩展服务，以及提高基础架构的效率。这驱使他们转向 NFV 并且在云中进行服务交付。

NFV 最初的关注点需要利用标准 IT 虚拟化技术来整合许多网络设备类型，并在行业标准大容量服务器、交换机和存储平台上以虚拟机 (VM) 形式运行网络服务，以便减少基本成本。这些平台往往依赖于商业芯片而不是专用 ASIC，这会限制像过去一样直接加速媒体/数据包处理的能力。另外，数字信号处理器 (DSP) 等专用芯片的设计与 COTS 芯片明显不同，需要强制应用进行改变以适应这些区别。多线程核心变得越来越常见，这有助于减轻一些潜在影响。大多数人都认为，在更加分离的模型中使用标准化平台所带来的优势超越了对性能的担忧。

更加成熟的 CSP 正在开始考虑成本优化以后的问题，将更多的重点放在服务敏捷性和可伸缩上。NFV 的未来必须融合到云原生体系架构，以便以全自动的方式处理 VNF 配置、扩展和错误恢复。

### 敏捷服务的创建和扩展需要云原生 NFV

云原生体系架构通常属于横向扩展设计，并且涉及到在 VNF、NFV MANO 和 NFV 基础架构之间进行紧密的团队合作，以交付优越的敏捷性、可伸缩性和弹性。它需要开放 VNF 分解为微服务并将状态从事务处理中分离出来，NFV MANO 智能监控 VNF 状态并关闭反馈回路进行服务资源调整，以及高效 NFV 基础架构可有效支持云原生 VNF 和 MANO 层。

预期是，CSP 将能够混合并匹配微服务以创建新型服务，并以一种更加自动化和简单的方式做这件事情。在促进这种更加敏捷的服务创建上，NFV MANO 将发挥重要的作用。它也将云环境中支持资源自动扩展。

在此新范式中，需要重新考虑弹性和高可用性，以便顾及标准服务器、交换机和存储平台没有专用硬件平台那么可靠。这导致将关注点从在硬件级别提供高可用性转移到在服务级别提供高可用性。对于实现高服务可用性的责任所在，VNF 供应商的意见不尽相同。许多供应商表示需要将弹性设计到应用中，但是有些供应商则表示云管理系统应该对此负责，而不是应用。VNF 管理器 (VNF-M) 监控 VNF 的运行状况，并在出现故障时重新实例化虚拟机，或者在发生降级时采取行动。有些 VNF 已经设计为可适应故障，尤其是传统上依赖于许多设备盒来交付的那些。其他人则认为 COTS 平台可能没有一些人想像的那么不可靠，并且在任何情况下，大多数人都希望平台的可靠性会随着时间的推移而改善。

使用云，资源将不像专用硬件那样受到约束。通过共用资源，它们不受任何限制，至少理论上如此。这与基于硬件的环境截然相反，在这种环境中，设计应用时需要了解固定资源的约束。这也意味着，要想让资源保持积极备用状态，代价会非常高。在云环境中，拥有备用资源将变得更加便宜/更加高效，因此应用应该考虑到这一变化。

## 电信功能在重要的方面不同于 IT 应用

许多电信应用不同于 IT 应用，因为除了控制、信令和媒体处理功能，它们还有数据平面/数据包处理功能。与典型 IT 应用相比，它们每一个都有不同的性能、延迟和容量要求，在许多情况下，这些要求更加严格。CSP 正在推动他们的供应商提高性能，但他们似乎愿意因为云的敏捷性和灵活性优势而接受小幅性能下降。

实时应用明显对延迟很敏感，这使得必须尽可能快地将数据包传输到网络上，并且能够在故障条件下重新配置网络流量。对数据平面敏感的网络功能（例如路由）的特点是，同时需要具有高容量（一次可以支持多少会话/多少流量）和高性能（每秒可以处理多少数据包/会话/事务）。控制（例如会话边界控制器）和信令（例如 PCRF、IMS）功能对性能就没有那么敏感，但同样需要容量和可伸缩性。许多电信应用都需要维护状态，这在分布式体系架构中特别具有挑战。

## “开放”是重新架构 VNF 的一个重要元素

CSP 正在推动他们的 VNF 供应商采用更加开放的软件开发方法。他们看到了 Web 级互联网公司已经实现的成本和面市时间优势，打算实现同样的优势。

在这里，“开放”可以指开放的 API，它们对于解决集成相关问题非常关键，而集成又一直是 CSP 过渡到基于云的基础架构的最大挑战之一。“开放”也可以指“开源”，CSP 发现它非常重要，因为它可以降低成本，提高开发速度，并且可以利用社区的智慧加速创新。事实上，大多数 VNF 供应商都参加了无数的开源项目，并且正在开始利用开源和引入更多的开放 API 来直接回应这个新出现的 CSP 要求。

## CSP 逐渐意识到需要重新架构 VNF

CSP 的理解程度各不相同，因为他们处于转换到云端的不同阶段中。较为先进的 CSP 的理解是，需要重新架构大多数的 VNF 才能在云中实现真正的敏捷性、可伸缩性和最佳性能。他们看到了云模型提供的机会，并且正在促使他们的供应商转向云。他们意识到，如果没有云提供的自动化和灵活性，那么他们就不会得到 NFV 的所有优势。“明白这一点”的那些人认识到了云的颠覆性，并且将它看作解决竞争挑战的关键部分。他们重视从基础硬件中分离应用所带来的灵活性和选择。他们预期云可以支持自助模型，并帮助产生新的收入流。

尽管如此，向云的转移仍处于早期阶段。大部分活动都处于概念证明状态，4-7 层设备正在成为首批试用 VNF。CSP 及其 VNF 供应商正在一起学习如何在虚拟环境中运行功能，并且刚刚开始了解它们如何在云中运行。供应商正在帮助他们的 CSP 客户了解 I/O、CPU、内存和存储如何影响 VNF 的性能 – 以前一直是 IT 组织的问题。

# 云原生 NFV 体系架构的关键里程碑

## 移植到虚拟的大容量服务器

对于许多网络功能来说，第一步是将软件从专用平台移植到虚拟的大容量服务器。要实现此里程碑，网络功能所有者通常必须应对两个主要挑战：缺乏专门的硬件帮助以及与虚拟化关联的性能损失。对于数据平面密集型 VNF，高效的服务器 I/O 可能会一石二鸟：既提高数据包的处理性能，又使系统能够通过 CPU 卸载更高效地运行。

有关更多详细信息，请参见[使用硬件提高 NFV 性能](#)。

## 分解

大多数网络功能都部署为一个庞然大物，所有组件都绑在一起形成一个实体。这可能会带来维护挑战，并且会减慢新技术的试用速度。因此，分解是将 VNF 演化为云原生并使其更加敏捷和可扩展的重要一步。应该为绝大多数的数据平面、控制和信令 VNF 执行这一步。（有些面向连接的 VNF 因为资源无法共享而不会从云中运行得到好处，所以不太可能执行这一步。）

在这个背景下，我们将分解定义为将庞大的 VNF 分为一组协作的服务，通常称之为“微服务”。每一个服务都实施相关的一组功能，例如会话管理、订阅者计费、统计信息收集等。服务通常会维护自己的状态并通过明确定义的 API 彼此通信，但是每个服务都可以独立进行开发和部署。值得注意的是，分解的目标不是简单地为了分解而分为微小的服务；它的目标是解决上述庞大体系架构的问题和限制。有些服务可能会很小，但是有些则会大很多。

### *将庞大 VNF 分解为微服务的优势和挑战*

首先，微服务提供了彼此独立地进行扩展的能力。而且，每个服务都可以部署到最适合其资源要求的硬件上。这与使用庞大体系架构的情况完全不同，在庞大体系架构中，具有截然不同资源要求的组件（例如 CPU 密集型和 I/O 密集型）必须部署在一起。CSP 重视这种方法的灵活性和效率，认为它将帮助他们最大程度地利用资源，并且更快地响应不断变化的市场和业务状况。如果服务之间的接口稳定且已标准化，则复杂度较低的较小元素也更容易集成。

其次，微服务体系架构缩小了故障域，并且改进了故障隔离，从而帮助轻松地进行故障排除和获得高可用性。例如，一个服务中的内存泄漏仅影响该服务；其他服务将继续正常处理请求。相比之下，庞大体系架构中一个组件行为不正常就会让整个系统崩溃。

第三，分解还允许从应用的核心逻辑中剥离常见功能，例如负载均衡和 DPI。这让应用本身能够更加轻型，这让它们能够更轻松更快速地进行开发、管理和部署（也许作为容器），并且能够集中核心功能并以“即服务”模型进行操作。CSP 希望能够重复使用常见功能，但是又不必为它付费多次。分解过程还让 VNF 所有者有机会从应用逻辑中去除不需要的功能，例如对已淘汰协议的支持。

这种方法也有关联的挑战，因为它明显地让运作更加复杂。在生产中必须管理更多的移动部分，即不同类型服务的多个实例。要成功地执行此操作，CSP 需要在 NFV MANO 层中设计更高度的自动化。另外，当供应商因为关键功能而被迫与另一个供应商的产品交互时，也可能引起潜在的互操作性问题。

将 VNF 分解为微服务后，CSP 能够将它们混合并匹配到新的不同服务。4-7 层 VNF 往往已经是分散的功能，因此某些 4-7 层系统供应商认为，在转移到云端之前，可能不需要分解过多功能。另外，IMS 和 SBC 等功能是如此专业化，以致某些移动 VNF 供应商认为让其子功能在其他地方重复使用毫无价值。总的来说，CSP 确实重视通过微服务进行服务创建提升灵活性和敏捷性。

轻型 VNF 更适合部署在分布式体系架构中，从而将应用功能部署到作为集群管理的多台服务器上。如今，许多 VNF 都能够以这种方式运行，但不是全部。CSP 正在考虑用分布式体系架构来实现灵活性、冗余和弹性。

VNF 供应商正在努力寻找分解和集中之间的适当平衡。他们需要确保产生的问题不会比正在解决的问题更多。更具体来说，他们的目标是：

- 最大程度地减少需要接触或移动数据包的次数以及对中央数据存储的调用次数，以便 I/O 不会显著增加。
- 创建如此多的微服务，也不会导致集成、编排和管理状态变得难以管理和/或成本过高，并且东西向流量不会压垮架顶式交换机。
- 最大程度地减少跳跃次数，以便不会引入延迟。这在云中的重要性更大，因为可能会将虚拟机自动移到不同的数据中心，从而失去对延迟的控制。

## 自动化与云伴随左右

自动化是敏捷服务创建、交付和扩展的主要促成因素，并且是基本虚拟化与云原生 NFV 体系架构之间的关键区别。许多 CSP 正在推动他们的供应商支持此交付模型。VNF 必须与 VNF 管理器和云编排系统一起使用才能实现自动配置、自动扩展和弹性。

### 配置

VNF-M 和云编排系统负有管理 VNF 生命周期的主要责任。VNF-M 提取 VNF 要求的详细信息，然后编排器使用这些信息，根据需要实例化新虚拟机。VNF 自己必须找出 VNF-M 已经分配了什么，然后自己进行配置。它还监控 VNF 的运行状况，并且可以根据需要向上或向下扩展。

分析工具将理清从网络进入的所有数据，从而让编排系统可以实时做出更加明智的选择。尽管如此，CSP 仍需要通过工具来监控和验证执行的操作，这一点非常重要，尤其是在早期，运作团队需要熟悉在自动化环境中的工作。

### 扩展

自动扩展资源（也称为“弹性资源”）是采用云原生体系架构的主要优势之一。但是，可通过两种可能的方法来增加容量：向上扩展，即向给定虚拟机添加更多资源；横向扩展，即创建更多虚拟机来支持给定功能。与本白皮书中讨论的大多数因素相同，每一种方法都有不同的考虑事项和权衡取舍，并且决定不一定就取决于 VNF 类型。

使用向上扩展，可通过增加更多资源来提高性能。但是，VNF 体系架构需要提供有更多内核可供它利用的可能性。它也提供了达到主机物理容量的可能性。

使用横向扩展，通过将功能散布在多个虚拟机来提高冗余。但是，下文将讨论到，跨这些独立的事务处理虚拟机管理状态可能会带来挑战。除非使用集群，否则由于需要考虑更多的元素，因此它可能也会带来管理挑战并让服务链复杂化。

虽然 VNF 供应商及其 CSP 客户仍在了解每一种方法的影响和权衡取舍，但是横向扩展似乎是大多数 VNF 的首选选项。吞吐量密集型供应商更可能向上扩展，而会话/容量密集型供应商看似可能会横向扩展。事实上，一些供应商可能既需要向上扩展又需要横向扩展，例如 EPC，它可能需要部署在更多的位置，每一个实例处理更多的会话。可以使用分析工具，根据预期网络行为帮助建议一种方法。

## 弹性

提高弹性是分布式体系架构的主要驱动因素之一。应用功能分散在以集群方式管理的多台服务器上的虚拟机之间。这种方法消除了单点故障，同时也限制了发生故障时产生的冲击半径。这样一来，只有一少部分用户会受到故障影响，而不是数百万的用户。

VNF 在进行不同的架构时必须支持水平扩展而不是垂直扩展。云的弹性需要具有短暂性。我们不仅必须能够快速创建新应用实例，而且我们还必须能够快速安全地处理它们。为了实现高效的水平扩展，通常使用服务无状态性原则，尽可能从状态数据中分离服务处理。无状态应用可以快速创建和销毁，并且可以与外部状态管理器快速连接和断开，从而提高我们响应需求变化的能力。当然，这也需要外部状态管理器本身具有可扩展性。根据访问状态所需的性能不同，可以将分布式容错状态存储实施为内存内数据网格、缓存和各种介质上的永久对象存储，例如闪存/SSD 或硬盘。

如本报告之前所述，维护状态在电信领域尤其重要，这在云中扩展 VNF 提出了一个更大的挑战。VNF 供应商对管理状态的最佳方式意见不同，但是大多数供应商都同意将状态从应用事务处理逻辑本身中分离出来更可取。VNF 领域的一些思想领袖已经证明，电信应用架构为无状态，因此几乎可线性扩展以满足任何负载要求。

将状态和处理服务分离后，可通过多种方式管理状态信息。最常见和直观的一种方式是在数据库中存储和管理状态，从而使得在发生故障时，或者在应用需要扩展或收缩时，任何虚拟机都可以访问状态。有许多分布式容错数据库解决方案可供选择，并且它们中的许多都是开源数据库，例如 Cassandra 或 HBase。大多数平台即服务 (PaaS) 产品都将数据库作为一项基本服务包含在内。

从积极的一面来看，VNF 本身可以免于管理状态的复杂性，从而重点处理其他服务处理逻辑，这可以大大缩短 VNF 开发时间。在这些数据库解决方案中，有一些是在 Web 级公司（例如 Facebook）发起的，因此它们已较为成熟，且已证明可在高度可伸缩的环境中发挥作用。但是，当数据库中与其对应微服务关联的信息需要同步时，可能会带来复杂性。当大量数据包导致缓存未命中，并且需要回调到不同的位置来获取状态信息时，可能会带来性能问题。对于必须以线速运行的数据平面应用来说，这一挑战最为显著，但是可通过使用一个分布式内存对象缓存解决方案（例如 Memcached）获得缓解。

其他人则认为，将状态复制到无数虚拟机更有效，这样任何一个虚拟机都可以在虚拟机发生故障时进行接管。它的优势是明显降低了故障切换时间，因为备用虚拟机有进行操作所需的一切。它也提供了冗余，这对灾难恢复来说非常有用。但是，让所有内容保持同步会带来开销，并且会构成一个风险因素。对于这种方法，扩展和收缩 VNF 也会带来挑战，它需要在负载均衡器模块中具有许多智能才能将流量引导至适当的虚拟机。用来管理状态的代码也与特定实现绑定在一起，这让代码的重复使用变得几乎不可能，从而减慢新功能的开发速度。

## 这对 CSP 意味着什么

CSP 应该鼓励其 VNF 供应商越过虚拟化，将 VNF 分解为微服务。这种方式几乎适用于所有 VNF。分解后，通过消除不必要的功能和集中常见功能，让 VNF 变得更加轻型，从而给予 CSP 更多的灵活性和敏捷性，并且能够更快且风险更低地创建新类型的服务。

他们还应该建议其供应商谨慎地在分解与复杂性之间进行平衡。微服务的数量将影响对数据中心内网络资源的集成、编排和需求。为了处理由于虚拟机数量增加而产生的预期东西向流量的增加，CSP 需要小心地计划其数据中心网络容量，以容纳预期的流量。他们还应该考虑服务链中的跳跃数，并且最大程度地减少每个跳跃处的延迟，以维护服务的 SLA。

在可以预见的未来，状态管理依然是一项挑战，它们可能会限制特定功能的分布和伸缩程度。如果 CSP 对 VNF 具有高弹性需求预期，或者打算通过将多个 VNF 时分复用到同一基础架构来提升基础架构的效率，那么他们应该鼓励其 VNF 供应商尽可能从服务处理中分离状态，以便服务实例可以轻松地进行收缩和扩展。

CSP 应该利用该功能来共用资源并自动按比例增加或减少资源，以实现云资源的最大利用率。他们应该使用分析工具来模仿网络行为，以便更好地了解如何在云中优化 VNF 功能，并认识到不同的 VNF 具有不同的资源要求，在向上扩展和横向扩展时必须将它们考虑在内，因此，一个人不能为所有 VNF 使用同一方法。

为了实现服务无状态性并根据分析实时获得见解，CSP 需要小心地设计其云基础架构，特别是要选择适当存储选项，满足状态访问和数据使用要求。