基于分布式云计算节点搭建超低延迟边缘网络应用

Akamai Generalized Edge Computing ('Gecko') for Low-Latency Web Applications

吴琼

Akamai 解决方案工程师 | 资深运维专家

InfoQ 极客传媒

极客邦科技 2024 年会议规划

促进软件开发及相关领域知识与创新的传播

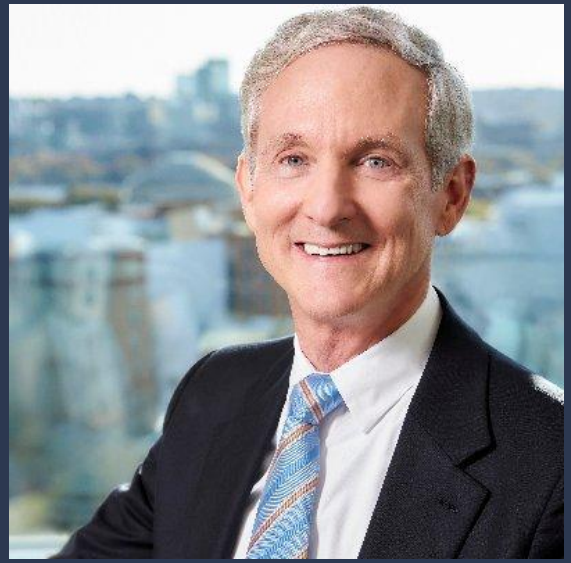
访问大会官网

参会咨询

起点：
一切都始于一个想法

1998 年

2022 年



AKAMAI HISTORY

全球数以千计分布最广的边缘节点

Edge

- CDN
- Serverless computing

1,200+
Networks

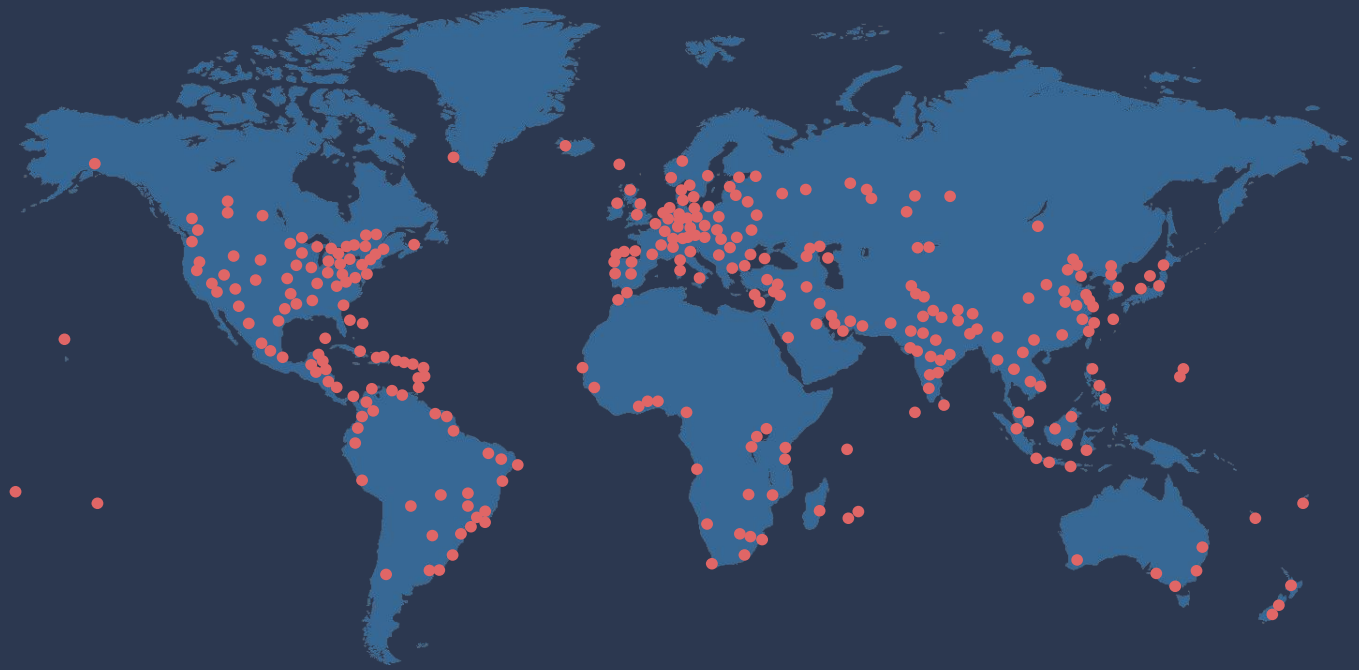
4,000+
Edge PoPs

1,000
TBPS of capacity

Edge PoPs - CDN | Serverless

AKAMAI HISTORY

将安全防护扩展到边缘



10+

TBPS of defense capacity

- WAF
- API
- DDoS

1,200+

Edge PoPs

130

Countries

5+

24/7 SOCC Locations

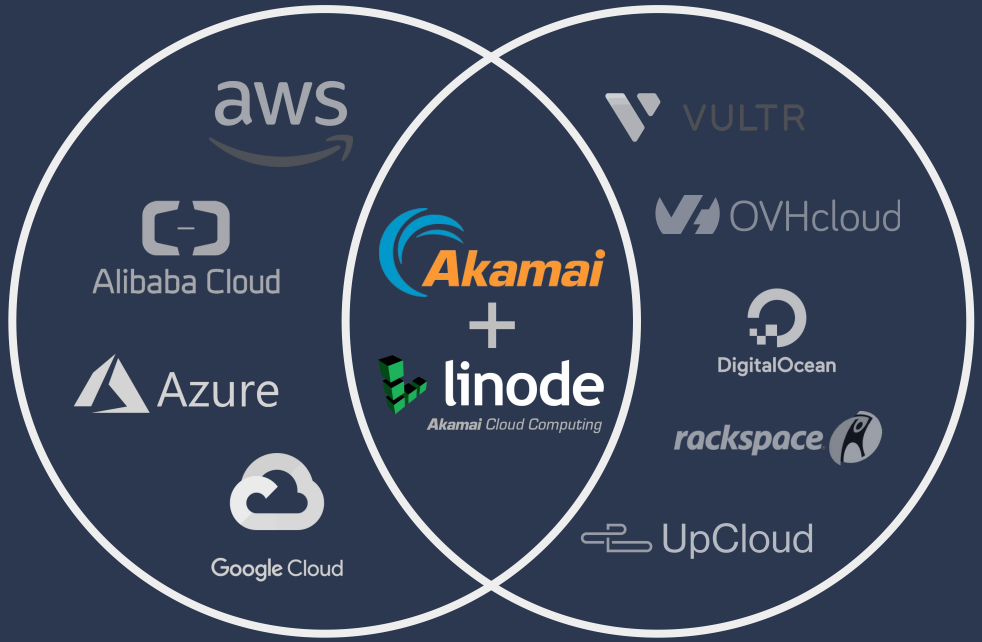
Edge PoPs - CDN | Serverless | WAF | App & API Protection | DDoS

Akamai Connected Cloud

为开发者构建, 为企业规模化

Cloud Hyperscalers

- 平台功能全面
- 突出企业级优势(规模性、稳定性)
- 需要专用工具



Alternative Clouds

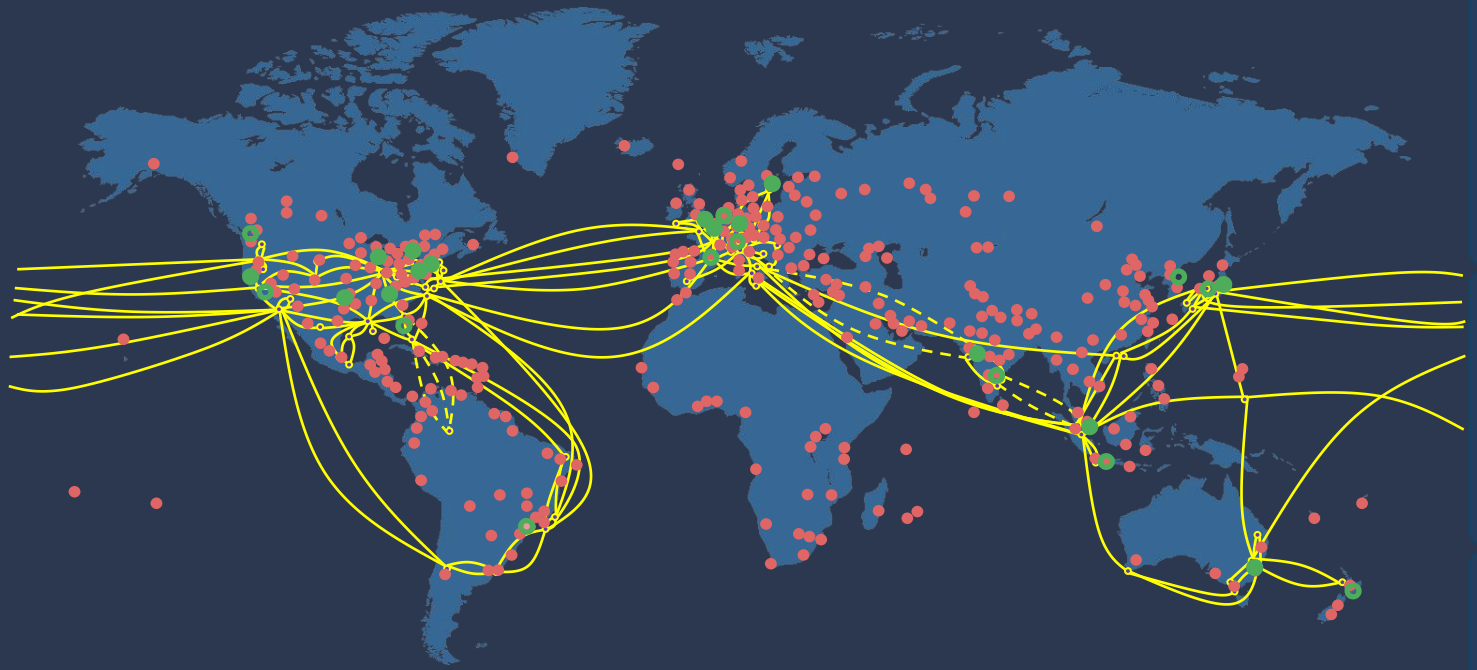
- 快速简单
- 价格透明且富有竞争力
- 多云有好
- 与客户本地部署之间的“非竞争关系”

全功能,规模化

简单, 价格“亲民”, 可访问

AKAMAI HISTORY

连接Akamai的全球骨干网的云计算平台



Regions
Full stack compute & storage

20+
Core compute regions

15+
Countries

Built for
Distributed workloads

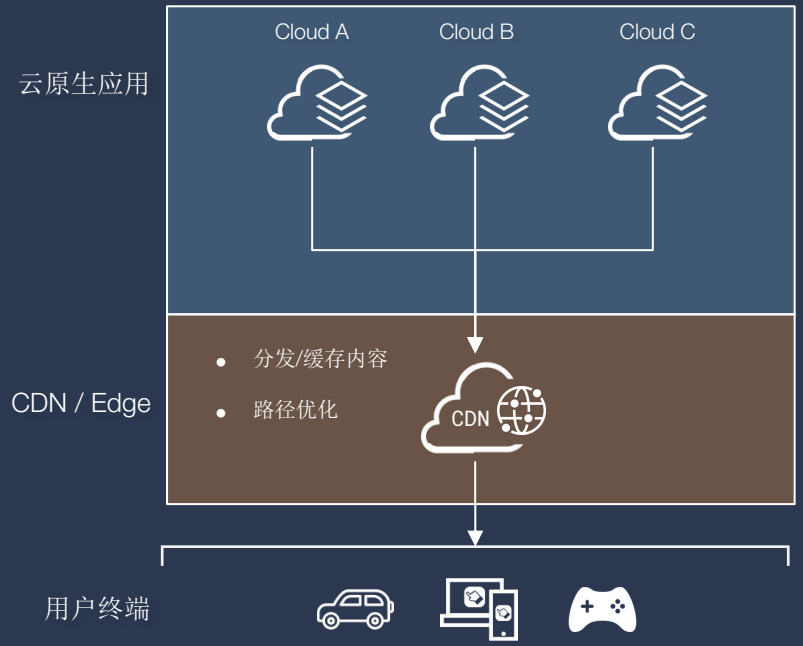
Edge PoPs - CDN | Serverless | WAF | App & API Protection | DDoS

Core Compute Regions

动机

为什么提出边缘云计算节点的构想？

当今主流的部署方式



挑战

- “大集中”式的计算资源部署在主要业务覆盖地区
- 仅有轻量化的计算需求可以被移至边缘
- 大量的“交互”流量
- 运维任务随着“多云”而变得日益复杂

无法逾越的“距离”

云计算中心

处理复杂且的海量计算任务
具有丰富的云计算服务

>150ms
平均连接延时

CDN边缘

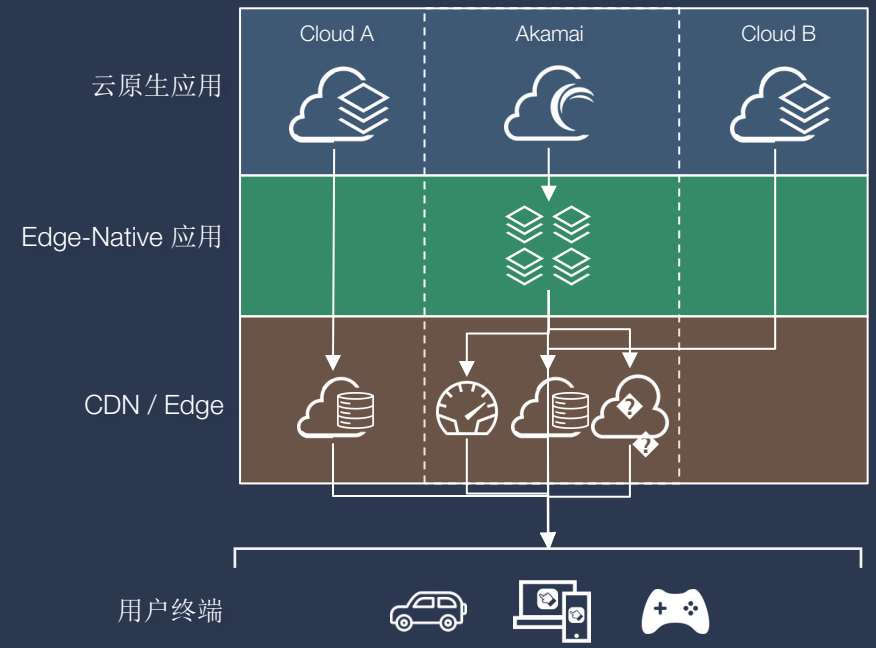
广泛分布，往往近具备KV数据库和轻量化FaaS处理能力

<20ms
平均连接延时

设计

边缘云计算节点的实现方式

Akamai Edge-Native 的计算连续体



优势

- 大规模分布
- 计算资源可以根据需求在不同位置、层次进行部署
- 超低延迟
- 优化云资源使用成本

运行环境实现方式

- 云计算站点运行一个名为VBIN的软件包，也称为“主机引擎”，这是一个软件应用程序，它在主机上管理虚拟机。
- Akamai网络运行ALSI，或Akamai Linux Server Install，能够部署到并管理全球超过400,000台主机。
- Guest IP的分配器会从数据库中的表中选择一个空闲本地原生IP分配给需要分配IP的Gecko VM。

```
clutzer@bos-mp2sp ~ % ssh root@1
+------------------------------------------------------------------+
| This is a vbin host.  No unauthorized use is permitted.          |
| if you are not an authorized user, please log off now.           |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
| Do not execute commands that will impact the system without      |
| validating the  action with vbin-sysops or vbin SREs.            |
|                                                                  |
| The Guest VMs shown below may be affected by your actions.       |
+------------------------------------------------------------------+
vbin guests          :
  linode506953       : Runnir
  linode506964       : Runnir
  linode507571       : Stoppe
```

```
region 43989 {
    name=SDN-OSAg
    provider=Akamai_Technologies_NetEng
    # servesto=public
    ipv6_netblock=2600:140b:1e00:6::/64
    guest_ipv4_ips=23.53.117.192/26          # Actual IPs may be a subset of this CIDR
    guest_ipv6_ips=2600:1:2:3::1743:4ef4/126 # In reality: 2600:???::1743:4ef1-4efc
    ecp_security_zone=1500695
    vbin_datacenter_id=76
    vbin_datacenter_slug=es-osa1
    vbin_net_version=3

    23.53.117.212 [ VBIN ] 31
    23.53.117.213 [ VBIN ] 32
    23.53.117.214 [ VBIN ] 33
    23.53.117.215 [ VBIN ] 34
    23.53.117.216 [ VBIN ] 35
    23.53.117.217 [ VBIN ] 36
}
```

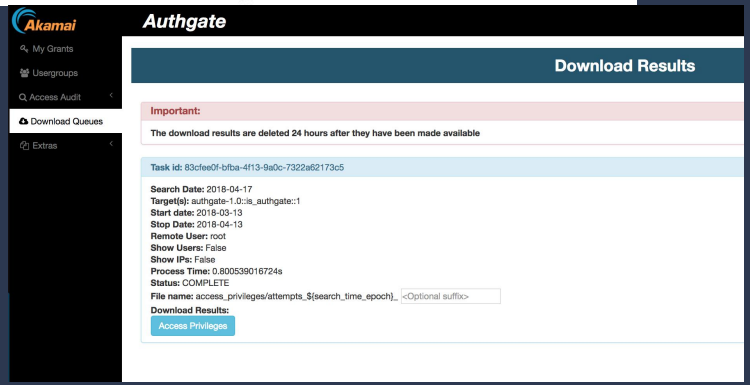
密钥管理

- Secret管理，包括主机上的运行时秘密保护以及秘密的中央配置、分发和轮换，都由Akamai的KMI来处理。 在Gecko主机上的KMI秘密只对root可用，并挂载到ramfs文件系统，这意味着只要主机的RAM子系统有电，它们就存在。

```
root@2.16.29.194:~# ls -l /etc/linode/vbin.yaml
lrwxrwxrwx 1 root root 32 Nov 22 17:48 /etc/linode/vbin.yaml -> /run/vbin/gecko_vbin_db/vbin.yaml
root@2.16.29.194:~# ls -l /run/vbin/gecko_vbin_db/vbin.yaml
-rw------- 1 root root 167 Nov 22 17:57 /run/vbin/gecko_vbin_db/vbin.yaml
root@2.16.29.194:~# mount | grep ramfs
none on /run/vbin type ramfs (rw,noatime,mode=750)
```

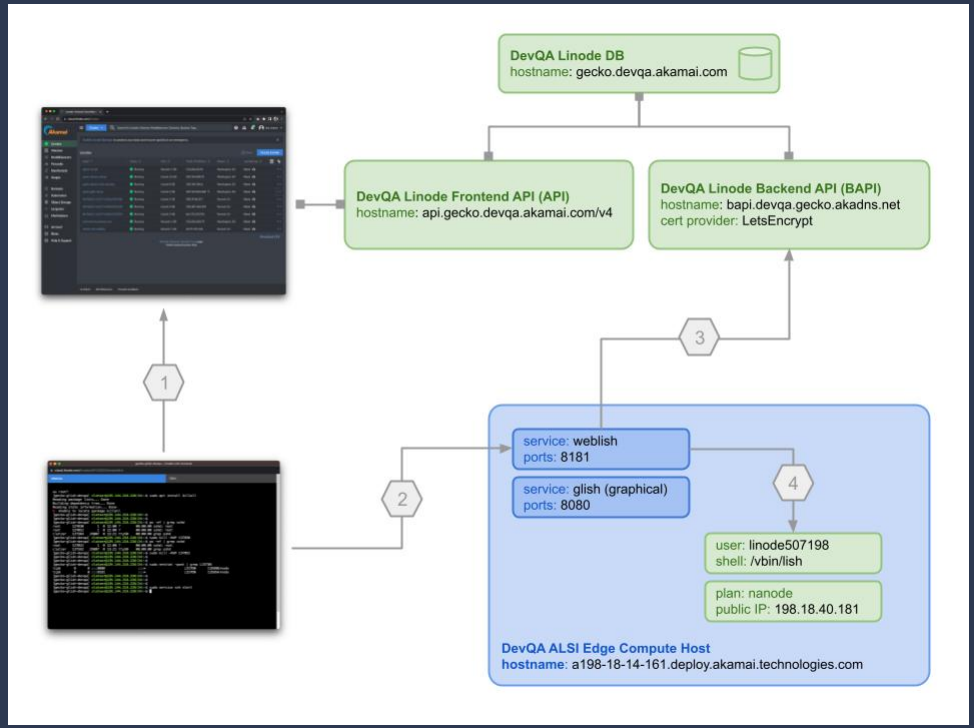
The following are some secrets that are currently managed by KMI:

- /etc/linode/vbin.yaml
  - Contains DB credentials
  - Contains CDN-based Distribution delivery Auth Token 2.0 key
- /etc/linode/private-stackscript.pem
  - Contains the private PKI key used by VBIN to decrypt StackScript decryption keys in the DB
- /etc/linode/vbin-images.s3cfg
  - Contains S3 credentials for the S3 Storage Box



用户界面

- MLISH是一套客户可以用来连接到他们的虚拟机控制台的带外访问方法。 MLISH由3个产品组成：
  - LISH over SSH
  - 显示映射（Graphical LISH）
  - LISH over Web（HTTP）
- 为了突出Gecko站点的访问性能优势，对于MLISH的访问架构，需要进行最小化的站点部署。而不是调度Akamai云平台核心站点的访问功能。我们将原有基于Perl的调度中心代码用GoLang重构了。



技术价值的驱动因素

大规模分布，具有更具吸引力的位置、成本和服务质量

超大规模分布式云计算能力，提供：

- 在特定低延迟网络位置的云服务
- 一致的低延迟性能和吞吐量等优势能力

边缘就绪的云解决方案

对于开发者来说，使用超大型云商的解决方案或第三方平台集成边缘计算能力更为困难，因为这需要他们管理网络、编排和自动化，尤其是跨不同的技术提供商平台

更精确边缘网络位置

应用的所有者越来越希望能够在他们的网络边缘，提供云计算中心级别的平台服务容量

技术价值的驱动因素

数据合规

敏感数据管理需要本地数据中心遵守法规。尽管超大规模云提供商正在稳步扩张，但仍有许多未服务的市场。

更开放的开发者体验

当所有的超大规模云开发和测试工具一起使用时，完整的环境可能运行成本高昂，限制了开发者的选择。

实践

构建边缘云计算节点遇到的技术难题

搭建过程中遇到的挑战

全球分布的需求

- 将云计算的核心能力带到以往任何一家云计算平台都无法到达的位置

网络架构

- 边缘计算节点互联
- 与云计算中心互联

安全防护

- DDoS高防
- WAF
- 东西向与南北向防护

其他技术问题

当从核心站点与Gecko站点之间进行互相迁移时，由于密钥管理机制的不同，导致迁移工作流非常不安全

Gecko主机将需要像核心站点主机一样的全局私有StackScript密钥，以便解密客户的StackScripts

Akamai CDN PoP 的 IP Reputation

应用

案例分享

理想应用场景

- 对延迟、经济型和数据管理有高要求
- 更灵活的IaaS架构，包括虚机、容器、负载均衡等工作负载
- 高容量或QoS敏感的网络流量
- “边缘原生” 的应用程序

应用场景举例

游戏

游戏匹配

启用依赖于接近性的数字体验，以实现等待时间、高性能和自适应决策制定

游戏对战服

启用实时响应，这对于竞技游戏体验至关重要

社交媒体

用户生成直播

最小化延迟以提供最佳的交互体验，例如聊天、反应等功能

WebRTC

启用直接在用户之间的低延迟通信，尤其是对于地理位置接近的用户

流媒体

媒体清单操作

最大化视频质量，实现无缝广告插入，并根据实时边缘和设备特性改善用户体验

直播流

启用最佳的流媒体性能，为靠近边缘服务器的观众减少延迟

人工智能

AI 推理

启用近用户通用计算，以全球范围内提供LLMs提供的服务

数据分布

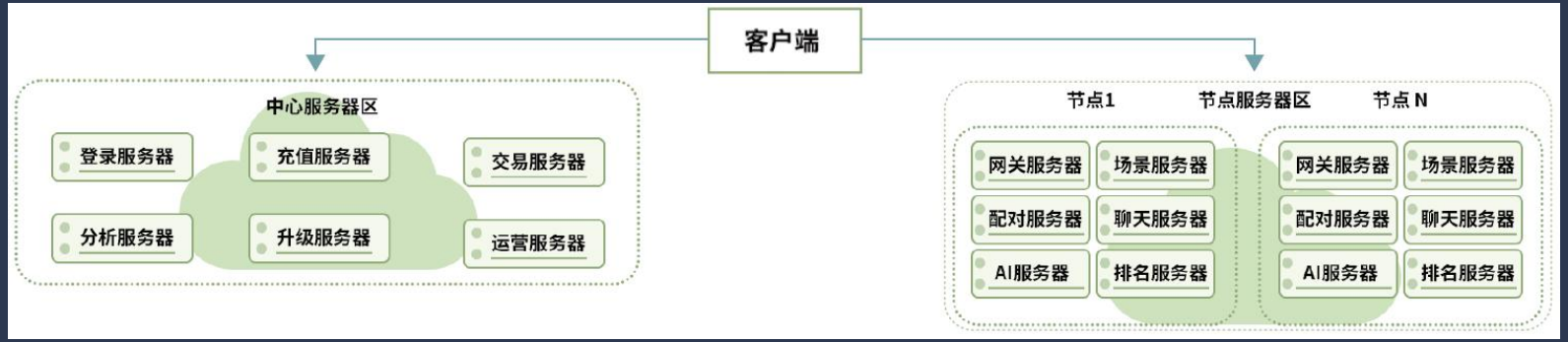
启用全球大规模数据分布，以支持实时决策和边缘的可扩展计算

游戏 | 对战服

游戏部署一般架构举例

从玩家体验来看，卡顿是破坏游戏体验的关键因素。

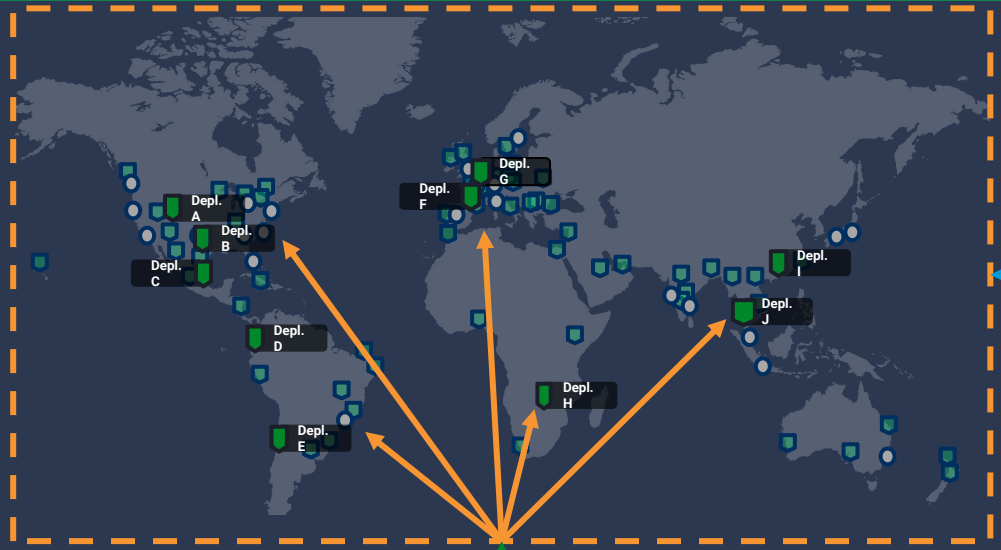
- <50ms, 体验较好
- >100ms, 明显延迟。 据统计，对战游戏每增加100ms延时，会导致14%的客户体验降低



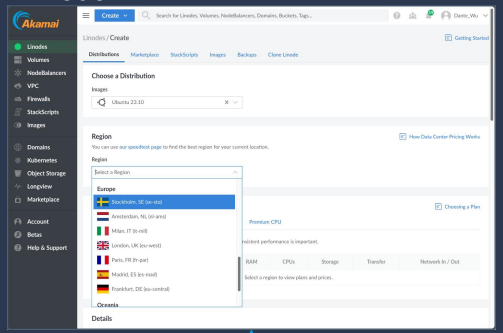
经长期服务全球游戏发行商的经验总结，Akamai发现理想游戏服务器的标准：

- 位置
- 硬件
- 联网

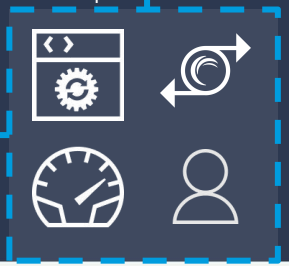
High-Level Architecture



Linode API



DevOps



社交媒体 | WebRTC

WebRTC 简述

WebRTC (Web Real-Time Communications) 是一项实时通讯技术，它允许网络应用或者站点，在不借助中间媒介的情况下，建立浏览器之间点对点（Peer-to-Peer）的连接，实现视频流和（或）音频流或者其他任意数据的传输。

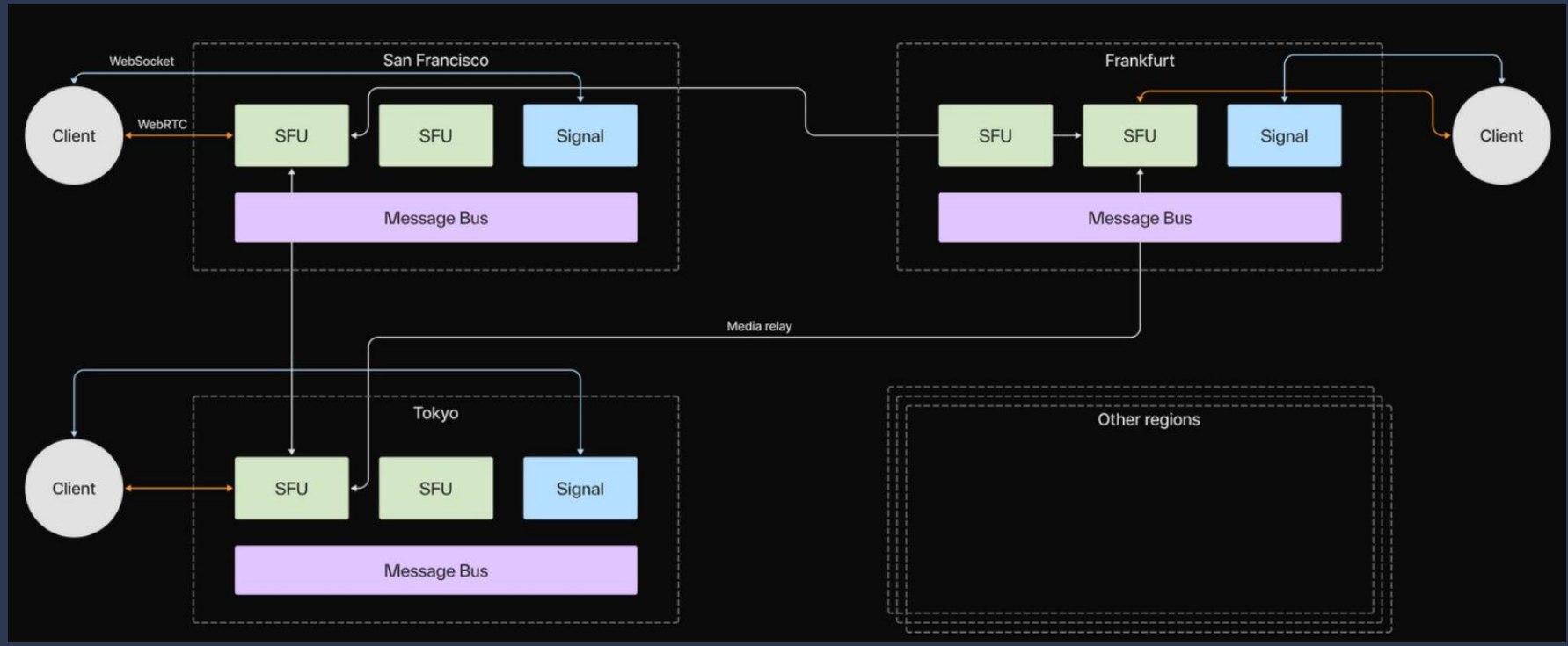
WebRTC的主要应用：

- 点对点通信
- 多方会议
- 屏幕共享
- 远程协作
- 在线教育、医疗
- 物联网

实践当中的网络问题定位：

- 带宽
- 抖动
- 丢包
- 拥塞
- **传输时间和RTT**

Sample Architecture



Terraform Code Sample

全球广泛部署WebRTC应用的挑战:

- 手动在不同的区域部署和管理超过100个实例是很耗时的。
- 定义了实例配置，例如，2个磁盘和网络配置。
- IPv6需要手动配置。
- 手动部署这些VM实例可能会非常容易出错。我们需要一种方法来保持部署过程的一致性和幂等性。

```
module "region-deployment" {
  source = "./modules/region-deployment"
  for_each = var.regions

  region = each.value.region
  instances_count = each.value.instances_count
  ipam_address_prefix = each.value.ipam_address_prefix
  ipam_address_start = each.value.ipam_address_start
  instance_type = each.value.instance_type
  disk_root_size = each.value.disk_root_size
  disk_data_size = each.value.disk_data_size
  os_image = each.value.os_image
  instance_label_prefix = each.value.instance_label_prefix

  ipam_address_netmask = 24

  booted = true
  enable_network_helper = true

  authorized_keys = var.authorized_keys
}

locals {
  regions_instances = merge([
    for k,v in var.regions: {
      for i in range(v.instances_count): "${k}-${i}" => {
        region = k
        idx = i
      }
    }
  ]...)
}

resource "terraform_data" "demo" {
  for_each = local.regions_instances

  connection {
    type = "ssh"
    user = "root"
    private_key = "${file(var.private_key)}"
    # host = "${each.value.instance_ip_addrs}"
    host = "${module.region-deployment[each.value.region].instances[each.value.idx].ip_address}"
  }

  # Init script
  provisioner "remote-exec" {
    inline = [
      "echo 'provisioner done. ${module.region-deployment[each.value.region].instances[each.value.idx].ipv6}' > /root/test",
    ]
  }
  depends_on = [ module.region-deployment ]
}
```

```
authorized_keys = ["ssh public key",]
token = "xxxxxx"
private_key = "/root/private_key"
regions = {
  eu-central = {
    region = "eu-central"
    instances_count = 1
    ipam_address_prefix = "10.40.1."
    ipam_address_start = 1
    instance_type = "g6-nanode-1"
    disk_root_size = 10000
    disk_data_size = 10000
    os_image = "linode/debian9"
    instance_label_prefix = "demo"
  }
  us-east = {
    region = "us-east"
    instances_count = 1
    ipam_address_prefix = "10.40.2."
    ipam_address_start = 1
    instance_type = "g6-nanode-1"
    disk_root_size = 10000
    disk_data_size = 10000
    os_image = "linode/debian10"
    instance_label_prefix = "demo"
  }
}
```

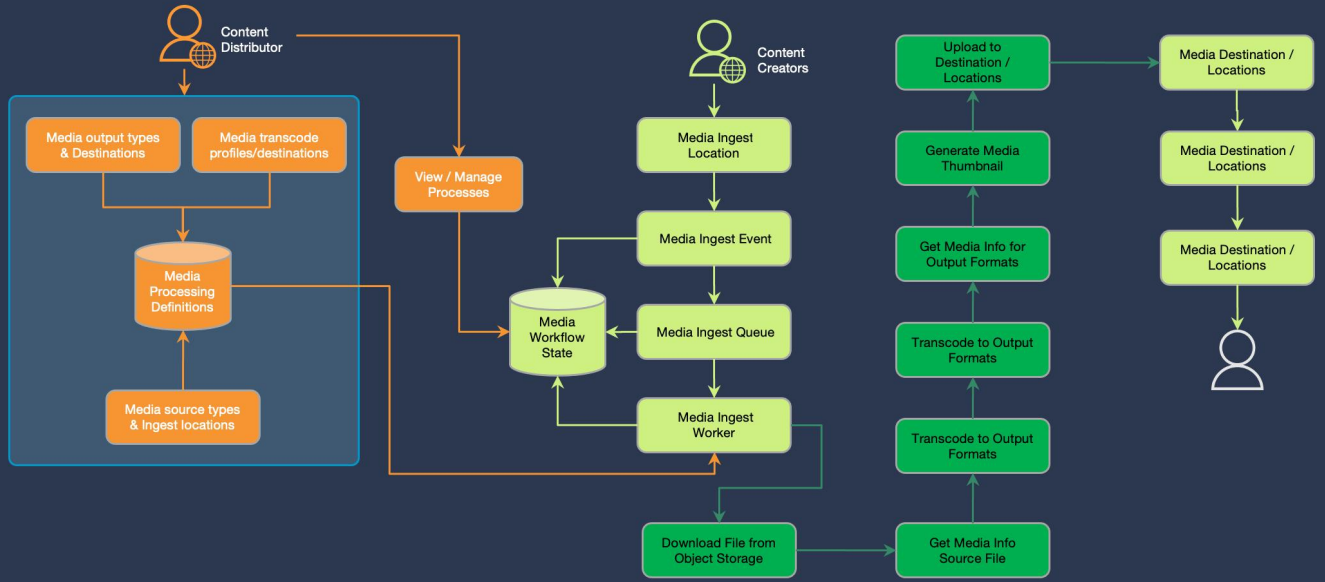
流媒体

直播转码

直播转码的边缘部署优势

Gecko边缘原生部署在直播转码中的应用主要有以下几个优势：

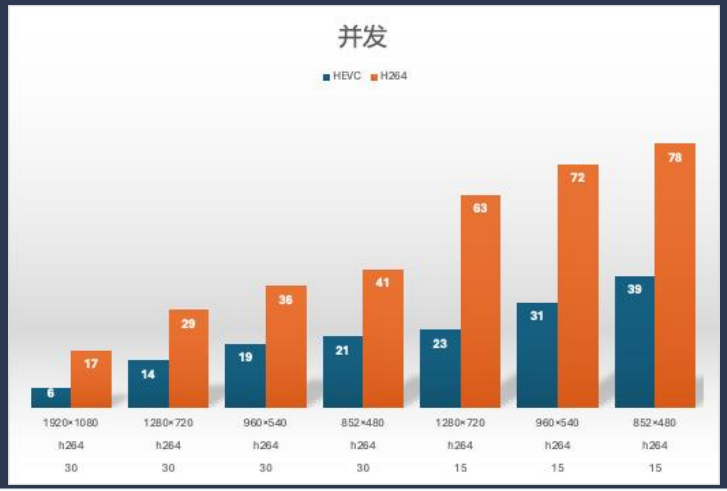
- 低延迟
- 高效利用资源
- 优化网络带宽
- 提升用户体验



视频转码性能压测

压测使用的是部署在Gecko站点当中的48vCore 96GB 的VM，部署了开源的转码方案。输入素材：

- 分辨率1920×1080
- 均为h.264格式
- 压测视频时长25s
- avg.CPU使用率 = 68%



Inbound FPS	Codec	Resolution	Concurrent Tasks
30	hevc	1920×1080	6
30	hevc	1280×720	14
30	hevc	960×540	19
30	hevc	852×480	21
15	hevc	1280×720	23
15	hevc	960×540	31
15	hevc	852×480	39
30	h264	1920×1080	17
30	h264	1280×720	29
30	h264	960×540	36
30	h264	852×480	41
15	h264	1280×720	63
15	h264	960×540	72
15	h264	852×480	78

人工智能

AI 推理

什么是推荐引擎

综合利用用户的行为、属性，对象的属性、内容、分类，用户对内容或商品的喜好，以及用户之间的社交关系等等，挖掘用户的喜好和需求，主动向用户推荐其感兴趣或者需要的内容和商品。

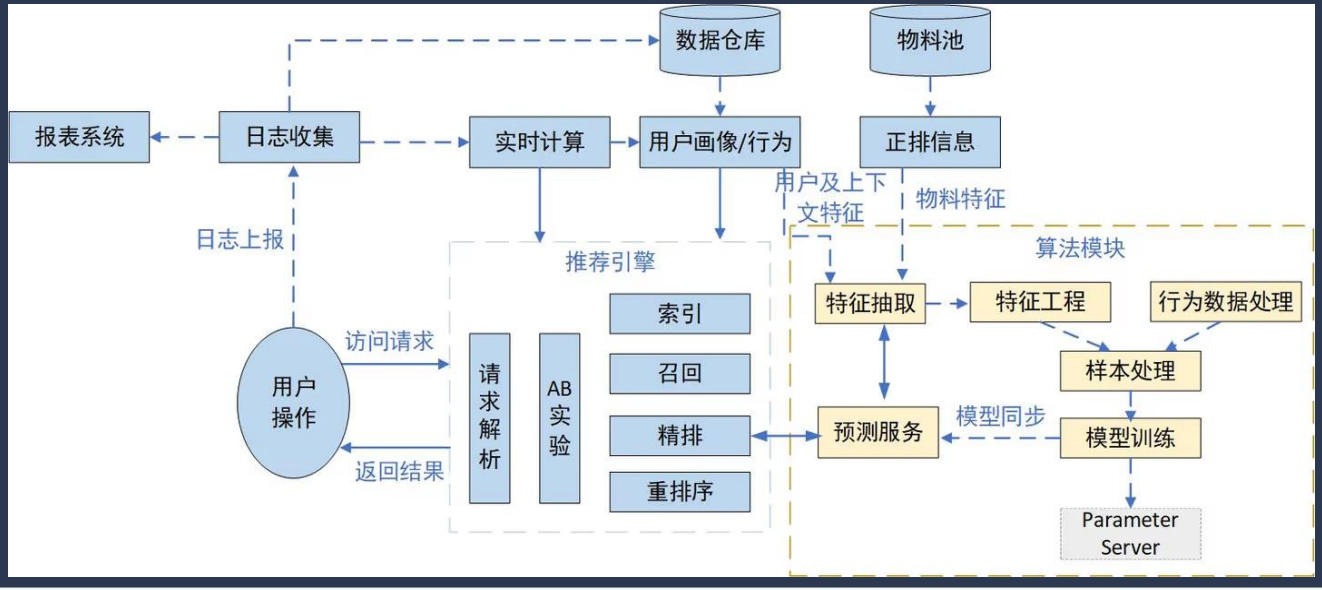
- **输出**
  - 为用户推荐其感兴趣或者需要的对象
- **数据源**
  - 用户:行为、属性
  - 对象:属性、内容、分类用户、
  - 对象间:偏好
  - 用户间:社交关系、信任
- **处理**
  - 挖掘用户喜好

- **推荐引擎通常包括：**
  - 召回：从海量物品库中快速查找
  - 粗排：通过用户、物品特征对召回排序
  - 精排：推荐环节的重中之重
  - 重排：打散或满足业务运营的特定强插需求

网络性能对推荐引擎的影响

推荐引擎对网络访问时延有要求的原因主要有以下几点：

- 用户体验
- 实时性
- 系统效率
- 数据一致性

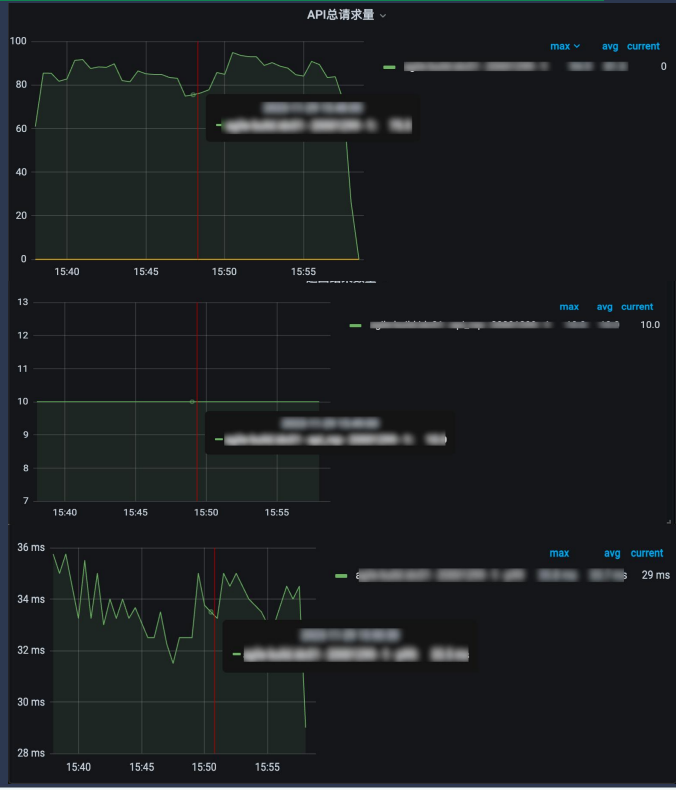


AI 推理性能压测

压测部署：

- 流量来源： Mock
- 压测时长：30min
- 参数：
  - QPS: 100，200
  - 候选场景数量: 1,000
  - 召回策略：4路召回，高热+模型召回
  - 排序：使用一个精排ctr模型
  - 精排物品数量：50
  - 一刷推荐物品数量：5

```
{
    "user":{
        "uid":"uid",
        "deviceInfo":{
            "deviceId":"device_id",
            "platform":"ios"
        },
        "age":"25",
        "gender":"female",
        "province":"Jiangsu",
        "city":"Nanjing",
        "country":"China"
    },
    "context":{
        "spm":"13$##$购买$##$130",
        "extra":{
            "fresh_mode":"2"
        }
    }
}
```



开放 | 云合作伙伴

Akamai解决方案 合作伙伴招募

我们正在寻找有特点的云原生方案：

- 能解决特定行业需求
- 有性能、安全性、可扩展性上有明显优势
- 与Akamai产品和服务优势互补，能共同打造差异化市场竞争能力

海外市场支持能力

- 具备跨文化沟通的经验
- 能理解不同市场的要求
- 能提供本地化服务
- 可通过全球网络和资源服务海外客户

高性价比

- 提供具有竞争力的价格
- 帮助客户降低成本
- 提升市场吸引力
- 能与Akamai共同推动云计算技术的普及和应用

如果您符合Akamai解决方案招募计划的条件，成为Akamai的伙伴，您将获得

全球强大的
边缘云网络部署能力

快速覆盖主流
出海用户客群渠道

市场推广和品牌建设的机会
提升品牌知名度

Akamai 已有解决方案合作伙伴

Distributed Database

Macrometa / HarperDB

Time-Series Database & Analytics

hydrolix

Video Transcoding

CAPELLA ATEME mediaexcel

Video Packaging

Unified Streaming / SCALSTRM

WebRTC & Interactive Media

LiveSwitch / LiveKit

Game Server Orchestration & Fleet Management

EDGEGAP

InfoQ 极客传媒

极客邦科技 2024 年会议规划

促进软件开发及相关领域知识与创新的传播

访问大会官网

参会咨询

THANKS

大模型正在重新定义软件

Large Language Model Is Redefining The Software