在亚马逊上部署 SQL Server 的最佳实践 EC2

AWS 规范性指导

AWS 规范性指导: 在亚马逊上部署 SQL Server 的最佳实践 EC2

Table of Contents

在亚马逊上部署微软 SQL Server 的最佳实践 EC2

Abhishek Soni 和 Sagar Patel、Amazon Web Services (AWS)

2023 年 12 月（文档历史记录）

本指南的目的是确保在亚马逊网络服务（EC2）云上将 Microsoft SQL Server 部署或迁移到亚马逊弹性计算云 (亚马逊AWS) 后获得一致的体验。它提供了配置数据库和服务器的最佳实践，以帮助优化基础设施、调整性能，并避免在部署或迁移后遇到意外问题。

本指南适用于计划将 Microsoft SQL Server 从本地环境迁移到亚马逊，或者想要在亚马逊上优化新 SQL Server 部署的数据库架构师 EC2、系统和数据库负责人以及管理员 EC2。

Amazon 在 AWS 云中 EC2提供可扩展的计算容量。在亚马逊 EC2 上使用 SQL Server 与在本地运行 SQL Server 类似。Amazon EC2 让您可以完全控制您的基础设施和数据库环境。您可以从 AWS 云的规模、性能和弹性中受益，但您负责配置和调整所有组件，包括 EC2 实例、存储卷、文件系统、网络和安全。本指南提供的信息可帮助您优化配置并最大限度地提高 AWS上 SQL Server 的性能。它详细讨论了服务器和存储设置以及最佳实践。它还说明了如何在适用的情况下自动进行设置，并讨论了数据库级别的配置更改。

> **Note**
>
> AWS 还提供了将本地 SQL Server 数据库迁移到托管服务的选项，例如适用于 SQL Server 的亚马逊关系数据库服务 (Amazon RDS)。有关迁移选项的讨论，请参阅 AWS 规范性指南网站上的关系数据库迁移策略。

配置计算和存储设置

在 Amazon 上迁移或部署 SQL Server 之前 EC2，您可以配置您的 EC2 实例和存储设置以提高性能并降低成本。以下各节提供了优化技巧和最佳实践。

主题

使用 Amazon EBS 优化的实例类型

如果您的 SQL Server 数据库处理 I/O 密集型工作负载，那么配置 Amazon Elastic Block Store (Amazon EBS) 优化实例将有助于提高性能。

Amazon EBS 优化型实例使用经过优化的配置堆栈，并为 Amazon EBS I/O 提供额外的专用容量。这种优化通过最小化 Amazon EBS I/O 与来自您实例的其他流量之间的争用，为您的 EBS 卷提供最佳性能。

优化磁盘布局或文件分发

一个卷用来存储数据和日志文件，另一个卷存储 tempdb 工作负载，Cold HDD (`sc1`) or Throughput Optimized HDD (`st1`) 卷用于备份。

如果您遇到与 I/O 相关的问题，并且想要将数据和日志文件的工作负载分开，请考虑使用不同的卷。如果您的工作负载需要将特定数据库分开，请考虑为每个数据库使用专用卷。

通常，tempdb 是最高 I/O 的目标，因此，如果不将该工作负载分开，它可能会成为瓶颈。这种分离还有助于将 tempdb 与用户数据库的数据和日志文件隔离开来。您可以使用成本相对较低的存储进行备份，以优化成本。

将 NTFS 分配单元大小设置为 64 KB

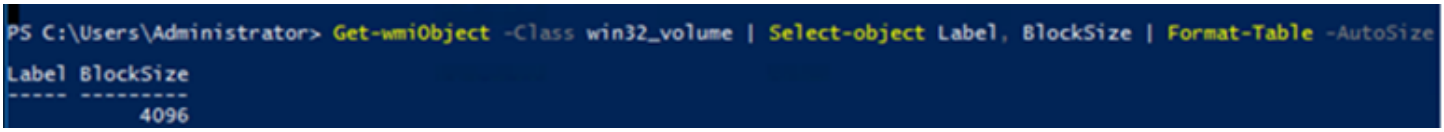
SQL Server 中的原子存储单元是一个页面，大小为 8 KB。八个物理上连续的页面构成了一个数据区（大小为 64 KB）。SQL Server 使用数据区来存储数据。因此，在 SQL Server 计算机上，用于托管 SQL 数据库文件（包括 tempdb）的 NTFS 分配单位大小应为 64 KB。

要检查驱动器的群集（NTFS 分配）大小，可以使用 PowerShell 或命令行。

使用 PowerShell：

```
Get-wmiObject -Class win32_volume | Select-object Label, BlockSize | Format-Table -
AutoSize
```

下图显示了的输出示例 PowerShell。



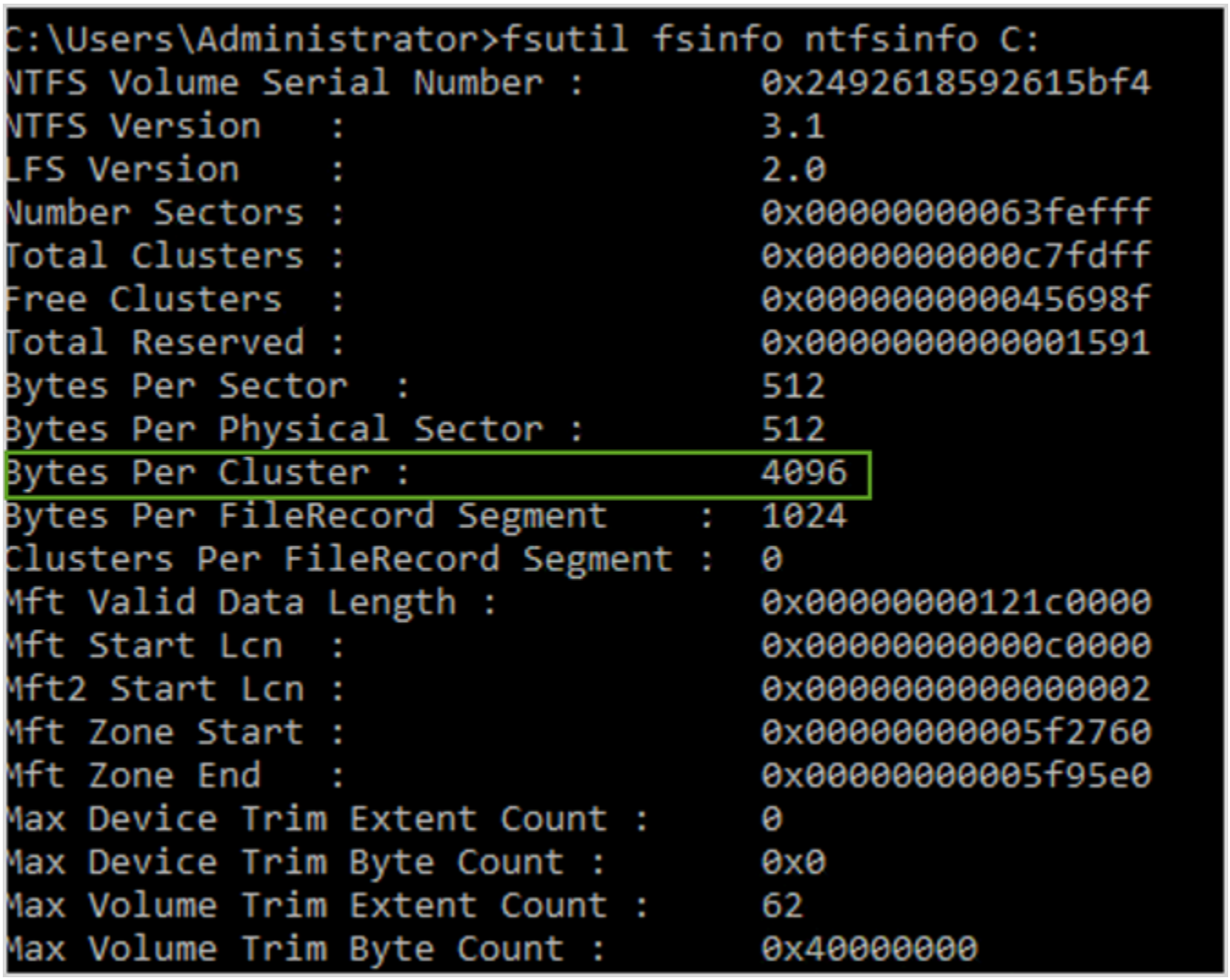
或者使用：

```
$wmiQuery = "SELECT Name, Label, BlockSize FROM win32_volume WHERE FileSystem='NTFS'"
Get-wmiObject -Query $wmiQuery -ComputerName '.' | Sort-Object Name | Select-Object
 Name, Label, BlockSize
```

使用命令行：

```
$ fsutil fsinfo ntfsinfo C:
```

下图显示了命令行的示例输出。每个集群的字节数值以字节为单位显示格式大小。示例输出显示了 4096 个字节。对于托管 SQL Server 数据库文件的驱动器，此值应为 64 KB。



在某些情况下，当您在亚马逊上使用固态硬盘存储时，SQL Server 的性能并不取决于块大小 EC2。有关更多信息，请参阅博客文章 SQL Server 存储的 64KB 代码块大小是否让 AWS 客户受益？

将 tempdb 放在实例存储中

当您使用 Amazon EC2 实例存储时，请将实例存储卷用于 tempdb。实例存储为您的实例提供临时性（短暂的）块级存储。我们建议您在实例存储卷上放置 tempdb，原因有两个：速度和成本。Tempdb 通常是使用最频繁的数据库，因此它受益于最快的可用驱动器。将 tempdb 放在实例存储中的另一个好处是可以节省成本，因为您无需为实例存储的 I/O 单独付费。

每当您重新启动 SQL Server 时，都会重新创建 Tempdb，因此停止或终止实例不会导致数据丢失。但是，在另一台主机上启动虚拟机时，实例存储卷会丢失，因为临时磁盘是在本地连接到计算机的，因此请谨慎规划。

当您使用实例存储卷时：

- 在 SQL Server 服务启动之前初始化卷。否则，SQL Server 启动过程将失败。
- 向 SQL Server 启动账户明确授予对实例存储卷的权限（完全控制权）。

将 tempdb 移至实例存储

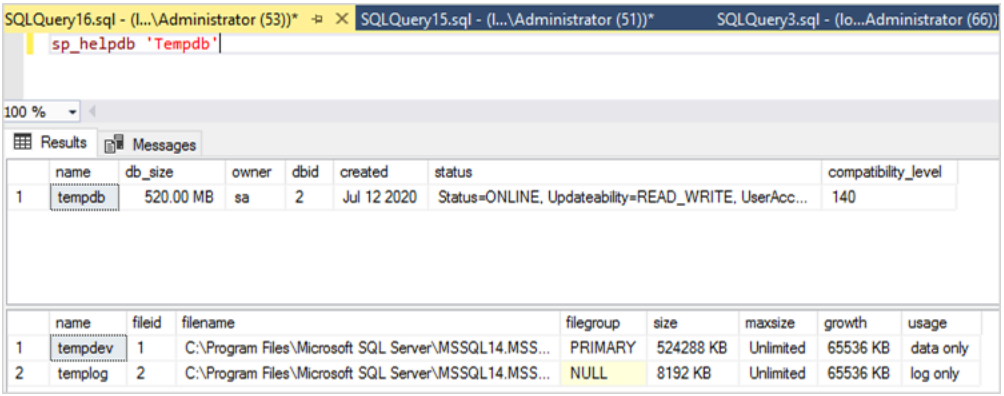
要将 tempdb 移动到实例存储卷，请执行以下操作：

1. 在 Windows 中，以管理员身份运行 `diskmgmt.msc` 以打开“磁盘管理”系统实用程序。
2. 初始化一个新磁盘。
3. 右键单击菜单，然后选择 New Simple Volume (新建简单卷)。
4. 使用以下设置格式化卷，完成提示：
   - 文件系统：NTFS
   - 分配单位大小：64K
   - 卷标：tempdb

   有关更多信息，请参考 Microsoft 网站上的 Disk Management（磁盘管理）文档。
5. 连接到 SQL Server 实例，然后运行以下命令以记下 tempdb 数据库的逻辑和物理文件名：

```
$ sp_helpdb 'tempdb'
```

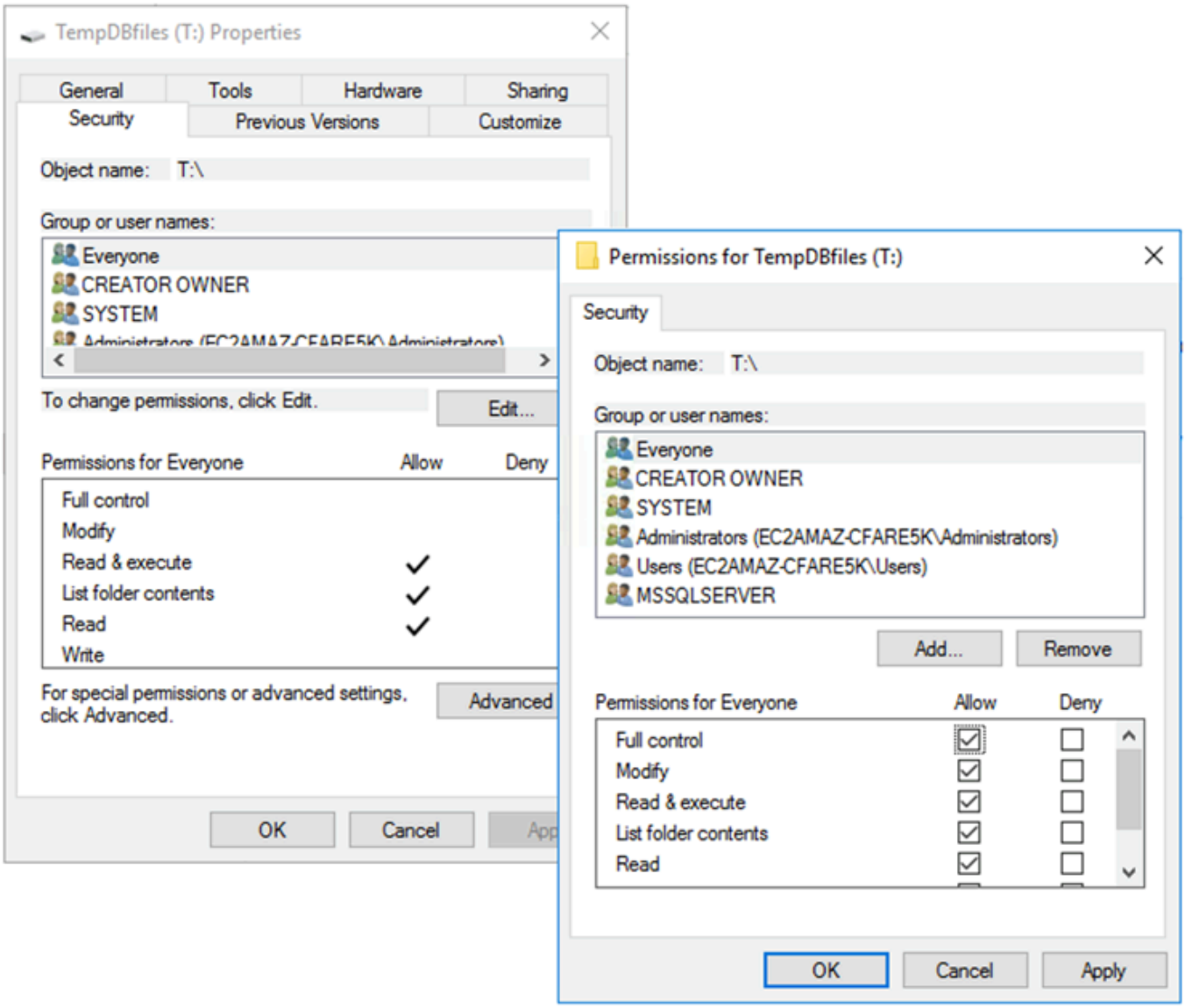
下列屏幕截图展示了该命令及其输出。



6. 将 tempdb 文件移到新位置。请记住将所有 tempdb 数据库文件设置为相同的初始大小。以下示例 SQL Server 脚本将 tempdb 文件移动到驱动器 T 并将数据文件设置为相同大小。

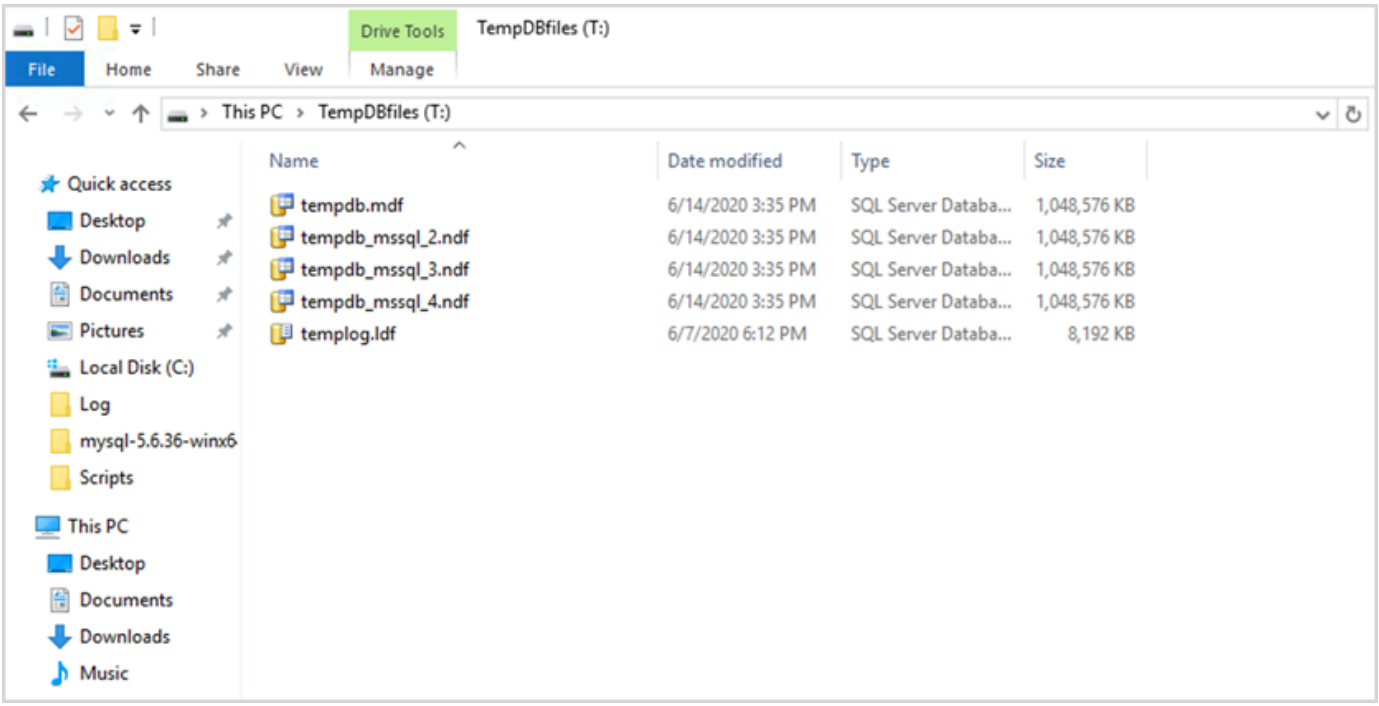
```
USE master
GO
ALTER DATABASE TempDB MODIFY FILE (NAME = tempdev, FILENAME = 'T:\tempdb.mdf',SIZE
 = 524288KB)
GO
ALTER DATABASE TempDB MODIFY FILE (NAME = temp2, FILENAME = 'T:
\tempdb_mssql_2.ndf',SIZE = 524288KB)
GO
ALTER DATABASE TempDB MODIFY FILE (NAME = temp3, FILENAME = 'T:
\tempdb_mssql_3.ndf',SIZE = 524288KB)
GO
ALTER DATABASE TempDB MODIFY FILE (NAME = temp4, FILENAME = 'T:
\tempdb_mssql_4.ndf',SIZE = 524288KB)
GO
ALTER DATABASE TempDB MODIFY FILE (NAME = templog, FILENAME = 'T:\templog.ldf')
GO
```

7. 向 SQL Server 启动帐户授予访问 tempdb 数据库新位置的权限，使其可以创建 tempdb 文件，如以下屏幕截图所示。



8. 重新启动 SQL Server 以使用 tempdb 的新位置。

   您将看到在新位置创建的 tempdb 文件，如以下屏幕截图所示。



9. 从原来位置删除 tempdb 文件。

在实例重启或启动/停止的情况下，要确保在 SQL Server 启动之前初始化实例存储卷，请按照下一节中的步骤进行操作。否则，由于 tempdb 未初始化，SQL Server 启动将失败。

初始化实例存储

要初始化数据存储，请执行以下操作：

1. 打开 Windows 服务管理器 (`services.msc`)，将 SQL Server 及其相关服务（例如 SQL Server 代理）设置为手动启动。（当实例存储卷准备就绪时，您将使用脚本启动它。）
2. 创建 PowerShell 脚本以用户数据形式传递给 Amazon EC2 实例。该脚本执行以下操作：
   - 检测临时存储并为其创建 tempdb 驱动器（示例中为驱动器 T）。
   - 如果 EC2 实例停止并重新启动，则刷新临时磁盘。
   - 授予 SQL Server 启动帐户对新初始化的 tempdb 卷的完全控制权。该示例假设一个默认实例，因此它使用 `NT SERVICE\MSSQLSERVER`。对于命名实例，通常默认值为 `NT SERVICE\MSSQL$<InstanceName>`。
   - 将脚本保存在本地卷上（示例中的 `c:\scripts`），并为其分配一个文件名（`InstanceStoreMapping.ps1`）。

- 使用 Windows Task Scheduler 创建计划任务。此任务在启动时运行 PowerShell 脚本。
- 在执行之前的操作之后启动 SQL Server 和 SQL Server Agent。

以下脚本来自 MS-SQL 可用性组研讨会的 第二个实验，但有一些更改。启动 EC2 实例时将脚本复制到用户数据字段，并根据需要对其进行自定义。

```
<powershell>
# Create pool and virtual disk for TempDB using the local NVMe, ReFS 64K, T: Drive
    $NVMe = Get-PhysicalDisk | ? { $_.CanPool -eq $True -and $_.FriendlyName -eq "NVMe
 Amazon EC2 NVMe"}
    New-StoragePool -FriendlyName TempDBPool -StorageSubsystemFriendlyName "Windows
 Storage*" -PhysicalDisks $NVMe
    New-VirtualDisk -StoragePoolFriendlyName TempDBPool -FriendlyName TempDBDisk -
ResiliencySettingName simple -ProvisioningType Fixed -UseMaximumSize
    Get-VirtualDisk -FriendlyName TempDBDisk | Get-Disk | Initialize-Disk -Passthru
 | New-Partition -DriveLetter T -UseMaximumSize | Format-Volume -FileSystem ReFS -
AllocationUnitSize 65536 -NewFileSystemLabel TempDBfiles -Confirm:$false
    # Script to handle NVMe refresh on start/stop instance
    $InstanceStoreMapping = {
    if (!(Get-Volume -DriveLetter T)) {
        #Create pool and virtual disk for TempDB using mirroring with NVMe
        $NVMe = Get-PhysicalDisk | ? { $_.CanPool -eq $True -and $_.FriendlyName -eq
 "NVMe Amazon EC2 NVMe"}
        New-StoragePool -FriendlyName TempDBPool -StorageSubsystemFriendlyName "Windows
 Storage*" -PhysicalDisks $NVMe
        New-VirtualDisk -StoragePoolFriendlyName TempDBPool -FriendlyName TempDBDisk -
ResiliencySettingName simple -ProvisioningType Fixed -UseMaximumSize
        Get-VirtualDisk -FriendlyName TempDBDisk | Get-Disk | Initialize-Disk -Passthru
 | New-Partition -DriveLetter T -UseMaximumSize | Format-Volume -FileSystem ReFS -
AllocationUnitSize 65536 -NewFileSystemLabel TempDBfiles -Confirm:$false
         #grant SQL Server Startup account full access to the new drive
        $item = gi -literalpath "T:\"
        $acl = $item.GetAccessControl()
        $permission="NT SERVICE\MSSQLSERVER","FullControl","Allow"
        $rule = New-Object System.Security.AccessControl.FileSystemAccessRule
 $permission
        $acl.SetAccessRule($rule)
        $item.SetAccessControl($acl)
        #Restart SQL so it can create tempdb on new drive
        Stop-Service SQLSERVERAGENT
        Stop-Service MSSQLSERVER
```

```
        Start-Service MSSQLSERVER
        Start-Service SQLSERVERAGENT
        }
    }
    New-Item -ItemType Directory -Path c:\Scripts
    $InstanceStoreMapping | set-content c:\Scripts\InstanceStoreMapping.ps1
# Create a scheduled task on startup to run script if required (if T: is lost)
    $action = New-ScheduledTaskAction -Execute 'Powershell.exe' -Argument 'c:\scripts
\InstanceStoreMapping.ps1'
    $trigger =  New-ScheduledTaskTrigger -AtStartup
    Register-ScheduledTask -Action $action -Trigger $trigger -TaskName "Rebuild
 TempDBPool" -Description "Rebuild TempDBPool if required" -RunLevel Highest -User
 System
</powershell>
```

使用缓冲池扩展

如果您计划使用缓冲池扩展，也可以考虑将其放在临时卷上。但是，我们强烈建议在实施之前对其进行全面测试。避免将相同的卷用于缓冲池扩展和 tempdb。

> **Note**
>
> 尽管缓冲池扩展在某些情况下可能很有用，但它不能取代 RAM。在您决定使用它之前，请查看 Microsoft 网站上提供的详细信息。

避免 CPU 内核不匹配

选择内核数高于许可范围的服务器可能会导致 CPU 偏斜和 CPU 功耗浪费。这是因为逻辑核心和实际核心之间的映射关系。当您使用带有客户端访问许可证 (CAL) 的 SQL Server 时，有些调度器程序将是 `VISIBLE ONLINE`，其余的将是 `VISIBLE OFFLINE`。由于调度器节点未得到最佳利用，这可能会导致非均匀内存访问 (NUMA) 拓扑的性能问题。

例如，如果您在 `m5.24xlarge` 实例上运行 SQL Server，它将检测到两个具有 24 个核心的套接字，每个套接字 48 个逻辑处理器，因此总共有 96 个逻辑处理器。如果您只有 48 个核心的许可证，则会在 SQL Server 错误日志中看到类似于以下内容的消息：

2020-06-08 12:35:27.37 Server SQL Server 检测到 2 个套接字，每个套接字 24 个核心，每个套接字 48 个逻辑处理器，共有 96 个逻辑处理器；根据 SQL Server 许可，使用 48 个逻辑处理器。这是一条信息性消息；无需用户进行任何操作。

如果您发现核心总数与 SQL Server 使用的核心数之间存在差异，请检查 CPU 使用率是否不平衡，或者使用许可证支持的核心数相同的服务器类型。

CPU 偏斜： 对于我们示例 (`m5.24xlarge`) 中的实例类型，默认情况下，SQL Server 会创建八个 NUMA 节点。这些节点中只有四个节点（父节点 ID 0、1、2、3）的调度器状态为 `VISIBLE ONLINE`。剩下的计划都是 `VISIBLE OFFLINE`。调度器之间的这种差异可能导致性能降级。

要查看调度器信息和状态，请使用：

```
$ select * from sys.dm_os_schedulers
```

如果您想要使用的服务器实例的核心数高于 SQL Server 许可证所支持的数量，请考虑按照 Amazon EC2 文档中为实例指定 CPU 选项中的说明自定义内核数。

测试磁盘性能

我们建议您使用类似的工具检查磁盘性能DiskSpd。在运行特定于 SQL Server 的测试之前，此工具可为您提供磁盘速度的估算值。评估磁盘性能非常重要，因为在本地环境中，EBS 卷的工作方式与传统 SAN 不同。缺乏适当的性能测试可能会导致迁移后出现意想不到的性能下降。您也可以使用运行自定义测试 DiskSpd。

启用即时文件初始化

在 SQL Server 中，使用执行卷维护任务设置启用即时文件初始化，除非您跟随监管限制。此选项可显著提高文件自动增长性能。

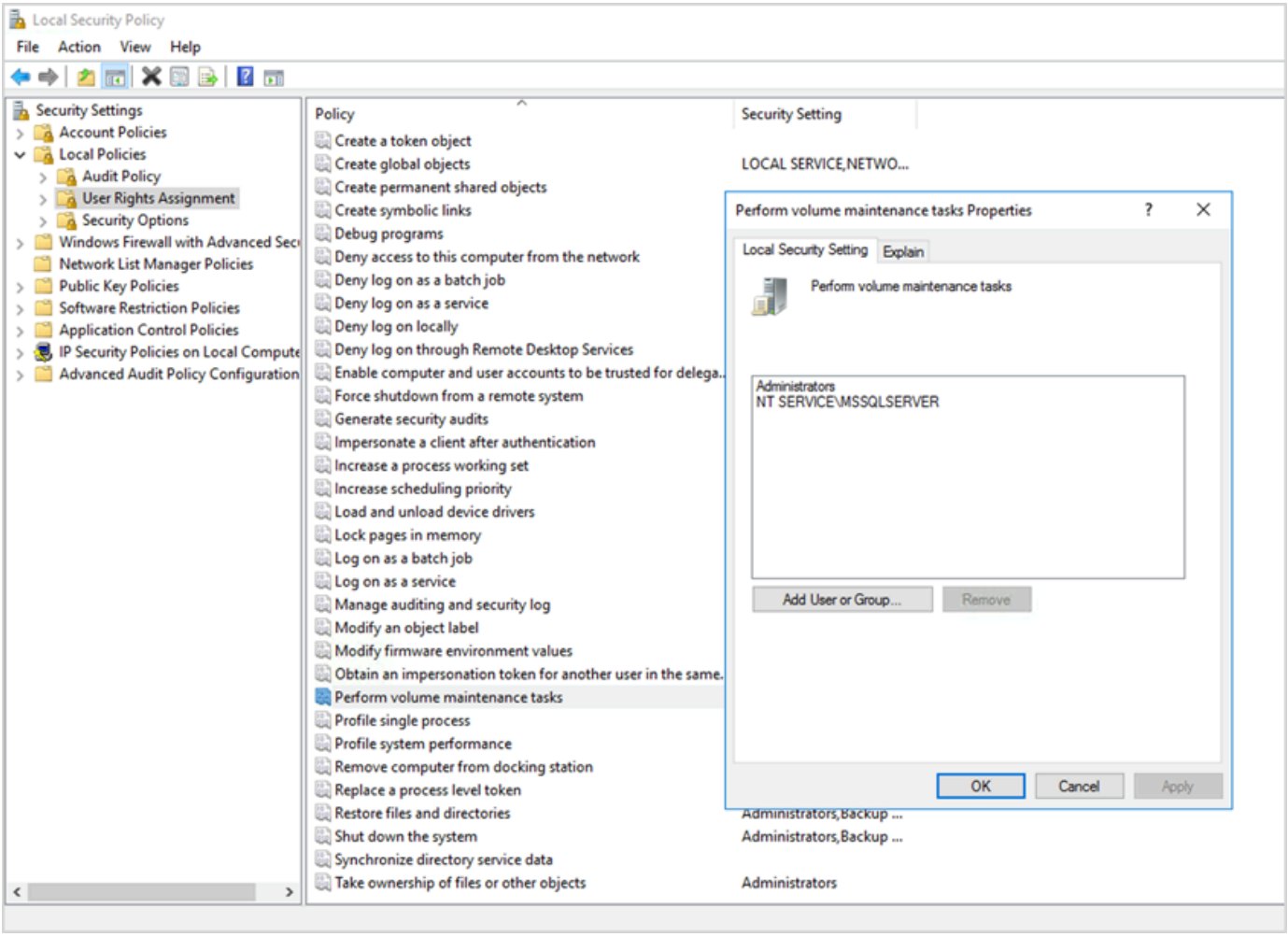
此设置会跳过数据文件的归零操作。也就是说，数据文件在初始化时不会填充零值 (`0x0`)，这可能需要很长时间。只有在向磁盘写入新数据时，磁盘上的当前版本内容才会被覆盖。

> **Note**
>
> 日志文件不能获得即时文件初始化的优势。

启用即时文件初始化：

1. 在开始屏幕上，运行 `secpol.msc` 以打开本地安全策略控制台。

2. 选择本地策略、用户权限分配、执行卷维护任务，然后添加 SQL Server 服务帐户，如以下屏幕截屏所示。



3. 重新启动 SQL 服务器实例以使更改生效。

有关即时文件初始化的更多信息，请参阅 Microsoft 网站上的 SQL Server 文档。

> **安全说明**
>
> 使用即时文件初始化时，只有在向文件中写入新数据时，磁盘才会被覆盖，因此可以读取已删除的内容。
>
> 当驱动器附加到实例时，文件上的自由访问控制列表 (DACL) 可降低信息泄露风险，因为它只允许访问 SQL Server 服务帐户和本地管理员。但是，分离文件后，可以对其进行访问。如果担心已删除内容的泄露问题，则应禁用 SQL Server 实例的即时文件初始化。

在内存中锁定页面

启用 SQL Server 启动帐户的在内存中锁定页面选项，以确保操作系统不会修剪 SQL Server 工作集。

要检查此选项是否已启用，请使用以下SQL 查询：

```
SELECT sql_memory_model, sql_memory_model_desc
FROM sys.dm_os_sys_info;
```

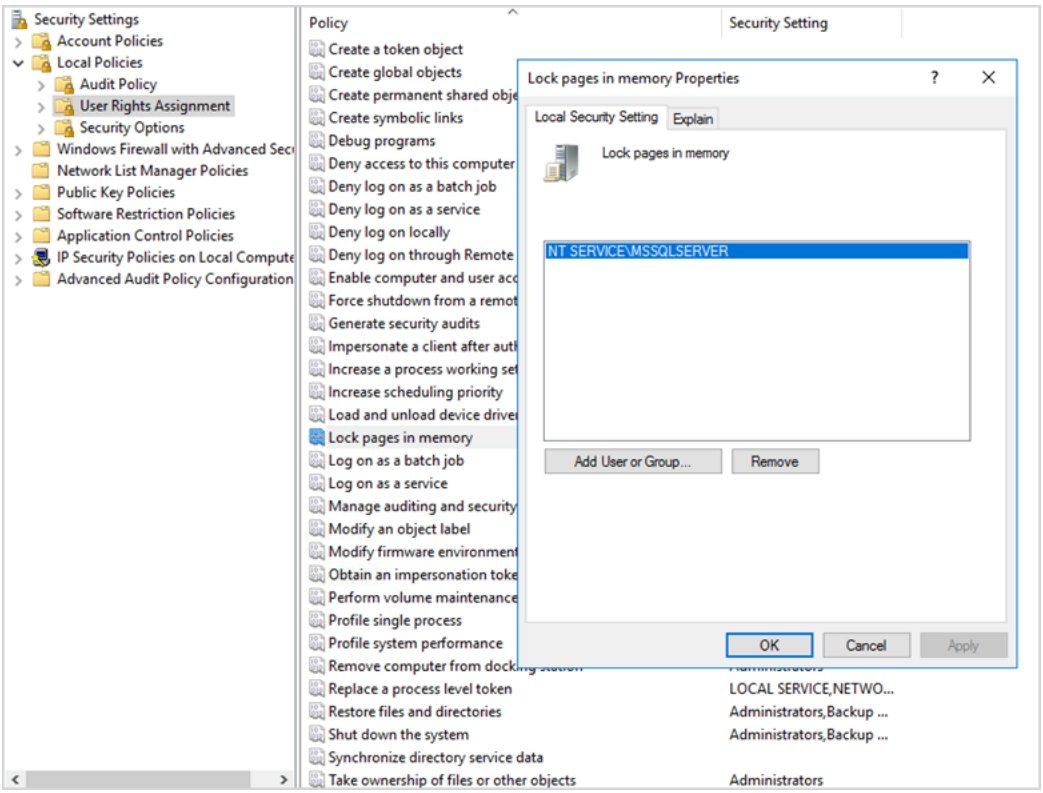
输出：

```
sql_memory_model     sql_memory_model_desc
1                   CONVENTIONAL

"CONVENTIONAL" means it’s not enabled.
```

要启用在内存中锁定页面选项，请执行以下操作：

1. 在开始屏幕上，运行 `secpol.msc` 以打开本地安全策略控制台。
2. 选择本地策略、用户权限分配、在内存中锁定页面，然后添加 SQL Server 服务帐户，如以下截屏所示。



3. 重新启动 SQL 服务器实例以使更改生效。
4. 使用以下 SQL 查询来确认在内存中锁定页面选项已启用：

```
SELECT sql_memory_model, sql_memory_model_desc
FROM sys.dm_os_sys_info;
```

输出：

```
sql_memory_model     sql_memory_model_desc
2                    LOCK_PAGES

"LOCK_PAGES" means it’s enabled.
```

有关 SQL Server 内存模型的更多信息，请参阅 Microsoft 网站上 sys.dm_os_sys_info 文档中的 `sql_memory_model` 和 `sql_memory_model_desc`。

禁用 TCP 卸载和 RSS 设置

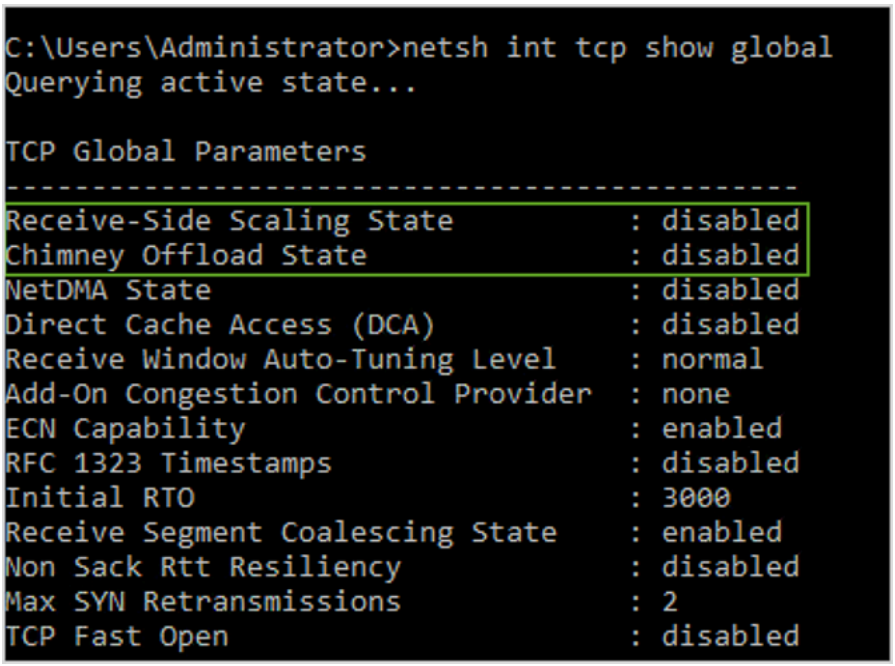
如果您在运行 SQL 工作负载时观察到随机连接问题，例如传输级错误或数据包传输错误，则可能需要禁用 TCP 卸载和 RSS 设置。

- TCP 卸载（TCP 烟囱卸载功能）将 TCP/IP 数据包的处理从处理器转移到网络适配器，从而腾出 CPU 来执行其他任务。
- 接收端扩展 (RSS) 对于在多处理器系统上分配对传入网络流量的处理有帮助。它可以在网络处理之间进行高效的负载平衡 CPUs。

要检查当前设置，请在命令提示符处运行 netsh 命令：

```
$ netsh int tcp show global
```

以下是该命令的示例输出。在此示例中，接收端扩展状态和烟囱卸载状态均处于禁用状态。



要获取有关特定连接的任务卸载信息，请在命令提示符下运行：

```
netstat -t
```

并检查卸载状态列的值。

要禁用 Windows Server 2008 和 2012 的 TCP 卸载和 RSS，请在命令提示符下运行以下命令：

```
netsh int ip set global taskoffload=disabled
netsh int tcp set global chimney=disabled
netsh int tcp set global rss=disabled
netsh int tcp set global netdma=disabled
```

有关这些设置的更多信息，请参阅：

- Microsoft 网站上的 TCP 烟囱卸载、接收侧缩放和网络直接内存访问功能以及接收侧缩放简介
- 亚马逊 EC2 文档中的 TCP 卸载
- Amazon EC2 文档中的 PV 驱动程序疑难解答

> **Important**
>
> 请勿使用IPsec 任务卸载或 TCP 烟囱卸载。根据 Microsoft 文档，这些卸载功能已在 Windows Server 2016 中弃用，未来的版本可能不支持这些卸载功能。使用这些功能可能会对性能产生不利影响。

确定您的 IOPS 和吞吐量要求

使用 Windows 性能监视器获取有关 IOPS 和吞吐量的信息。

要打开 Windows 性能监视器，请在命令提示符下运行 perfmon。IOPS 和吞吐量数据由以下性能计数器提供：

- 磁盘 reads/sec + disk writes/sec = IOPS
- 磁盘读取 bytes/sec + disk write bytes/sec = 吞吐量

我们建议您获取高峰使用时间和典型工作负载周期的 IOPS 和吞吐量数据，以便更好地估计您的需求。请确保您为 SQL Server 选择的实例类型支持这些 I/O 要求。

正确估算值很重要。否则，您可能会过度配置资源，这可能会导致资源未得到充分利用，或者资源配置不足，从而可能导致严重的性能问题。

使用条带化来绕过 IOPS 和吞吐量限制

当您的 SQL Server 应用程序需要超过 EBS 卷上可用的最大 IOPS 和吞吐量时，可以考虑对您的 EBS 卷进行条带化以克服这些限制。

条带化卷 (RAID) 可帮助您满足 IOPS 和吞吐量要求，并且受特定实例支持的最大 IOPS 和带宽的限制。有关条带化选项的更多信息，请参阅 Amazon EC2 文档中的 RAID 配置。您也可以在独立服务器上使用存储空间。有关更多信息，请参阅 Microsoft 文档。

从防病毒软件中排除 SQL Server 文件

配置防病毒软件设置时，请确保将 SQL Server 文件和目录排除在病毒扫描范围之外。有关详细信息以及要排除的文件和目录列表，请参阅 Microsoft 网站上的如何选择要在运行 SQL Server 的计算机上运行的防病毒软件。

如果您不排除这些 SQL Server 文件，则当 SQL Server 需要使用它们时，它们可能会被防病毒软件损坏或隔离。不排除这些文件也可能导致性能问题。

配置 SQL Server

本节提供了配置 SQL Server 数据库以微调性能、避免常见陷阱并满足安全性和可用性要求的最佳实践。您可以在将数据库迁移到 Amazon 之前或之后实施这些更改 EC2。以下各节提供了配置提示和最佳实践。

主题

配置 tempdb 以减少争用

我们建议您使用多个大小相等且增长因子相等的数据文件配置 tempdb。

在经常使用 tempdb 的繁忙数据库服务器上，当服务器遇到沉重负载时，您可能会注意到严重阻塞。您可能会注意到，任务正在等待指向 tempdb 中页面的资源。这些页面可能是页面可用空间 (PFS) 和共享全球分配地图 (SGM) 页面，格式为 2:*x*:*x*（例如 2:1:1 或 2:1:2）。

要提高 tempdb 的并发性，可以增加 tempdb 中的数据文件数量，以最大限度提高磁盘带宽并减少分配结构中的争用。下面是一些指导方针：

- 如果逻辑处理器的数量等于或小于 8：使用相同数量的数据文件和逻辑处理器。
- 如果逻辑处理器的数量大于 8：使用 8 个数据文件。

如果争用仍然存在，则以 4 的倍数增加数据文件数，直到争用得到纠正，最多为服务器上的逻辑处理器数量。这将有助于避免 tempdb 中出现争用 SGAM。如果您使用的是 SQL Server 2014 或更早版

本，则还需要启用跟踪标志 1118。此标志强制在统一范围而不是混合区上分配页面，这样可以最大限度地减少 SGAM 页面上的扫描并减少争用。

从 SQL Server 2016 (13.x) 开始，此行为由 ALTER DATABASE 的 AUTOGROW_SINGLE_FILE 和 AUTOGROW_ALL_FILES 选项控制。例如：

```
alter database <database name> MODIFY FILEGROUP [PRIMARY] AUTOGROW_ALL_FILES
```

有关这些选项设置的更多信息，请参阅 Microsoft SQL Server 文档。

设置 MAXDOP 以获得最佳性能

最大并行度 (MAXDOP) 是一个服务器配置选项，用于在多个服务器上运行 SQL Server。 CPUs它控制用于在并行计划执行中运行单个语句的处理器数量。原定设置值为 0，这将使 SQL Server 使用所有可用的处理器。这可能会影响性能，而且对于大多数用例来说并不是最佳选择。

在为 SQL Server 配置最大并行度 (MAXDOP) 值时，请遵循以下准则。

NUMA 节点	逻辑处理器	最大并行度 (MAXDOP) 值
单列排序	≤ 8	4、2 或内核数量（用于一个或两个内核）
单列排序	> 8	8、4 或 2
多个	≤ 16	8、4 或 2
多个	> 16	16、8、4 或 2

> **Note**
>
> 在大多数用例中，将最大并行度 (MAXDOP) 设置为 2、4 或 8 通常可以获得最佳结果。我们建议您测试工作负载并监控任何与并行度相关的等待类型，例如 CXPACKET。

您可以使用以下查询来收集 SQL Server 2016 及更高版本的当前 NUMA 配置：

```
select @@SERVERNAME,
```

```
SERVERPROPERTY('ComputerNamePhysicalNetBIOS'),
cpu_count,
hyperthread_ratio,
softnuma_configuration,
softnuma_configuration_desc,
socket_count,
numa_node_count
from
sys.dm_os_sys_info
```

其中：

- cpu_count指系统 CPUs 中逻辑的数量。
- hyperthread_ratio 是一个物理处理器暴露的内核数量之比。
- softnuma_configuration 为 0、1 或 2:
  - 0 (OFF): 默认值
  - 1 (automated)：基于软件的 NUMA
  - 2 (manual)：基于软件的 NUMA
- softnuma_configuration_desc 为 OFF、ON 或 MANUAL:
  - OFF 表示基于软件的 NUMA 功能已关闭。
  - ON 表示 SQL Server 会自动决定 NUMA 节点的大小。
  - MANUAL 表示基于软件的 NUMA 为手动配置。
- socket_count 是处理器插槽的数量。
- numa_node_count 是系统中可用的 NUMA 节点的数量。

要检查当前最大并行度 (MAXDOP) 值，请使用：

```
$ sp_configure 'max_degree_of_parallelism'
```

有关最大并行度 (MAXDOP) 的更多信息，请参阅 Microsoft SQL Server 文档。

更改并行度的成本阈值

并行性的开销阈值决定了哪些查询适合并行执行。此属性的默认值为 5，这意味着如果串行计划的成本大于 5（这是指抽象的成本单位，而不是估计的时间），则优化器会切换到并行计划。我们建议您设置此属性为更高的数值。

当处理器价格高、处理能力低且查询处理速度比现在慢时，默认值是恰当的。如今的处理器要快得多。因此，相对较小的查询（例如，给定成本阈值为 32）不会从并行执行中获益太多，特别是考虑到与协调并行执行相关的开销。

在大多数情况下，将并行度的成本阈值设置为 50 是一个不错的起点。以下是有关如何配置并行度成本阈值的示例：

```
USE sampledb;
GO
EXEC sp_configure 'show advanced options', 1 ;
GO
RECONFIGURE
GO
EXEC sp_configure 'cost threshold for parallelism', 50 ;
GO
RECONFIGURE
GO
```

针对临时工作负载进行优化

启用针对临时工作负载进行优化选项，以提高包含许多一次性临时批处理的工作负载的计划缓存效率。最初运行即席查询时，数据库引擎会缓存已编译的计划存根，而不是完整的执行计划，从而节省计划缓存中的空间。如果您再次运行临时批处理，则数据库引擎会识别出之前已经运行过该批处理，并将已编译的计划存根替换为计划缓存中的完整已编译计划。

要检查此选项是否已启用，请使用以下查询：

```
$ sp_configure 'optimize for ad hoc workloads'
```

有关针对临时工作负载进行优化的更多信息，请参阅 Microsoft SQL Server 文档。

使用跟踪标志提高性能

考虑使用适用于您环境的 SQL Server 跟踪标志来增强性能。例如：

- 4199：启用 SQL Server 累积更新 () 和 Service Pack () 中发布的查询优化器 (QOCUs) 更改。SPs
- 8048：将 NUMA 分区的内存对象转换为 CPU 分区的内存对象。
- 9024：将全局日志池内存对象转换为 NUMA 分区的内存对象。

以下示例说明了如何在 Amazon 上开启和关闭 SQL Server 的跟踪标志 EC2。如果您在启用跟踪功能时遇到任何问题，请确保您拥有该账户的相应权限。

要开启跟踪标志 4199，请运行：

```
dbcc traceon (4199, -1);
```

要检查跟踪标志的状态，请运行：

```
dbcc tracestatus (4199);
```

要关闭跟踪标志 4199，请运行：

```
dbcc traceoff (4199, -1);
dbcc tracestatus (4199);
```

有关跟踪标志的完整列表，请参阅 微软 SQL Server 文档。

安装最新补丁

从 SQL Server 2017 开始，微软已停止发布 Service Pack (SPs)。它仅发布累积更新 (CUs) 和关键更新 (GDRs)。

SPs 包括针对 SQL Server 的重要修复程序，因此请确保已安装最新的 SP。另外，如果可能，请安装最新的 CU 软件包。

有关最新 SQL Server 更新的信息，请参阅微软网站上的微软 SQL Server 最新更新。

设置最大服务器内存上限以避免内存压力

出于性能考虑，SQL Server 不会释放已经分配的内存。启动 SQL Server 时，它会慢慢占用 min_server_memory 选项下指定的内存，然后继续增长，直到达到 max_server_memory 选项中指定的值。（有关这些选项设置的更多信息，请参阅 SQL Server 文档中 Server 内存配置选项。）

SQL Server 内存由两个部分组成：缓冲池和非缓冲池（也称为待留内存或 MTL）。max_server_memory 选项的值确定 SQL Server 缓冲池的大小，该缓冲池由缓冲区缓存、过程缓存、计划缓存、增益结构和其他缓存组成。

从 SQL Server 2012 开始，min_server_memory 和 max_server_memory 会考虑所有缓存的所有内存分配，包括

SQLGENERAL、SQLBUFFERPOOL、SQLQUERYCOMPILE、SQLQUERYPLAN、SQLQUERYEXEC、SQLOPTIMIZ
和 SQLCLR。有关 max_server_memory 下的内存管理员的完整列表，请参阅微软 SQL Server 文档中的 sys.dm_os_memory_clerks。

要检查当前的 max_server_memory 值，请使用以下命令：

```
$ sp_configure 'max_server_memory'
```

我们建议您将 max_server_memory 的上限设置为一个不会造成全系统内存压力的值。没有适用于所有环境的通用公式，但我们在本节中提供了一些指南。max_server_memory 是一个动态选项，因此可以在运行时对其进行更改。

首先，您可以按如下方式确定 max_server_memory：

```
max_server_memory = total_RAM - (memory_for_the_OS + MTL)
```

其中：

- 操作系统的内存为 1-4 GB。
- MTL（待留内存）包括堆栈大小，在 64 位计算机上，每个工作线程为 2 MB，计算方法如下：MTL = stack_size * max_worker_threads

或者，您也可以使用：

```
max_server_memory = total_RAM - (1 GB for the OS
 + memory_basis_amount_of_RAM_on_the_server)
```

其中 RAM 的内存基础量按如下方式确定：

- 如果服务器上的 RAM 介于 4 GB 到 16 GB 之间，则每 4 GB 内存留出 1 GB。例如，对于容量为 16 GB 的服务器，留出 4 GB。
- 如果服务器上的 RAM 超过 16 GB，则每 4 GB 的 RAM 留出 1 GB 不超过 16 GB，16 GB 以上每 8 GB RAM 留出 1 GB。

例如，如果一台服务器有 256 GB 的 RAM，则计算结果为：

- 操作系统为 1 GB
- 最大 16 GB RAM：16/4 = 4 GB

- 剩余 RAM 超过 16 GB：(256-16) /8 = 30
- 剩下的总 RAM：1 + 4 + 30 = 35 GB
- max_server_memory：256 - 35 = 221 GB

初始配置后，监视在典型工作负载持续时间内可以腾出的内存，以确定是否需要增加或减少分配给 SQL Server 的内存。

> **Note**
>
> Windows 以 96 MB 的速度发出低内存资源通知信号，因此您需要一个缓冲区，但您可以将具有 256 GB 或更高内存的大型服务器上的 可用兆字节设置为 1 GB 以上。

有关更多信息，请参阅 Microsoft SQL Server 文档中的内存管理架构指南。

使用最高的数据库兼容性级别

请检查并确保您使用的是当前数据库兼容性级别，以便利用 SQL Server 的最新改进。检查这一点很重要，因为当您将数据库从较低版本恢复到较高版本时，它将保持较低版本的兼容级别。只有当您将数据库兼容性设置为适用于所安装引擎版本的最新级别时，某些最新的数据库增强功能才会生效。

要检查当前的数据库兼容性，请使用：

```
$ select name, compatibility_level from sys.databases
```

有关数据库兼容级别的更多信息，请参阅 Microsoft SQL Server 文档。

控制数量 VLFs

预先分配数据和日志文件的最大大小。为了获得更好的性能，请通过预分配空间和更正日志文件的自动增长（自动增长VLFs）设置来控制虚拟日志文件的数量 ()。

通常，8 GB 的自动增长系数在大多数生产环境中效果良好。考虑将事务日志文件按照 8 GB 的区块进行扩展。较高的数量 VLFs 可能会延长数据库的备份和恢复时间，并且可能导致任何需要浏览日志文件的操作（例如复制）出现性能问题。

有关 VLF 创建和增长算法的更多信息，请参阅SQLskills 博客。

检查数据库自动增长设置

任何需要数据或日志文件增长的事务都包括文件增长操作所花费的时间。文件按照 FILEGROWTH 选项定义的增量大小增长。您可以在 SQL Server 概要分析器跟踪中查找文件增长事件。如果文件增长需要很长时间，您可能会看到像 ASYNC_IO_COMPLETION 这样的等待类型，这种情况发生在数据处理非常慢的时候。此类等待类型不仅会影响性能，还可能导致事务超时。如果该事务锁定了其他事务所寻求的资源，则超时将导致严重的服务器阻塞问题。

为此，我们建议您非常仔细地配置自动增长设置。请记住以下事项：

- 在 SQL Server 中，文件增长是成本最高的操作之一。
- 小组块中频繁的自动增长可能导致磁盘碎片。
- 日志文件中频繁的自动增长会导致大量的虚拟日志文件 (VLFs) 并影响性能，如上一节所述。

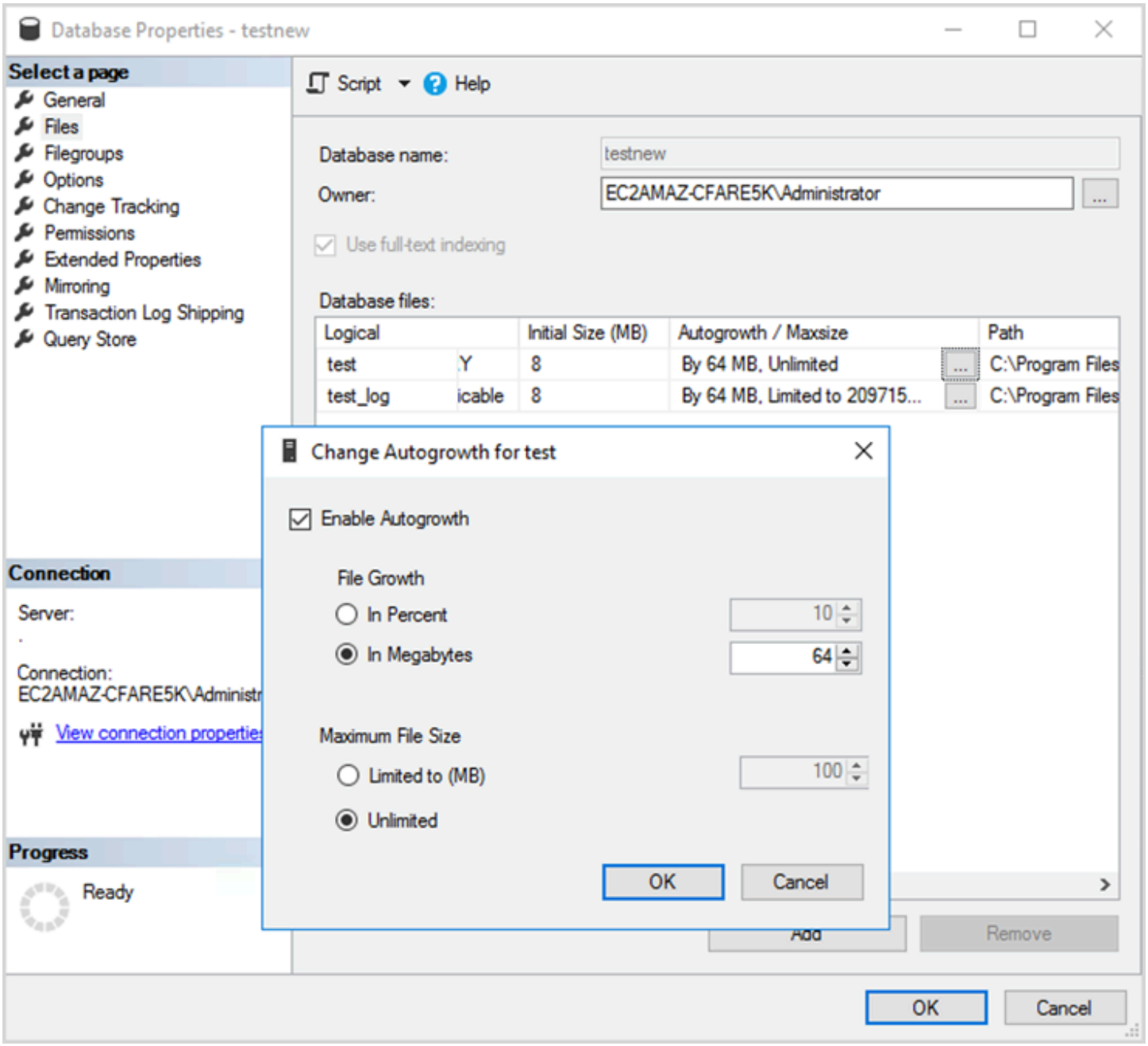
所有这些原因都可能导致数据库启动缓慢以及备份和恢复时间增加。

理想情况下，您应该在定期监控的基础上主动预先增长文件。在将自动增长设置为百分比或静态值（以 MB 为单位）之间进行谨慎选择。通常，将自动增长设置为文件大小的八分之一是一个不错的起点，但这可能不是正确的选择。（例如，如果您的数据文件大小为几 TBs 个，则此百分比会过高。）

在大多数情况下，1024 MB 的自动增长值适用于大多数大型数据库中的数据文件。对于日志文件，512 MB 是一个不错的起点。为了采取应急措施，我们强烈建议您设置自动增长值，但要根据过去的趋势手动将文件增长几个月。

> **Note**
>
> 设置自动增长应该是一种应急措施，因此应在为文件预分配存储空间之后进行设置。

您可以使用 SQL Server Management Studio (SSMS) 或 Transact-SQL 来更改自动增长设置。下图显示了 SSMS 中的自动增长设置。



对数据和日志文件使用 FILEGROWTH 选项时，请谨慎选择将其设置为百分比或静态值（以 MB 为单位）。设置百分比会导致文件增长不断增加，因此您可能更喜欢使用静态大小来更好地控制增长率。

- 在 SQL Server 2022 (16.x) 之前的版本中，事务日志无法使用即时文件初始化，因此延长日志增长时间尤其重要。
- 从 SQL Server 2022（16.x，所有版本）开始，即时文件初始化可以使高达 64 MB 的事务日志增长事件受益。新数据库的默认自动增长大小增量为 64 MB。大于 64 MB 的事务日志文件自动增长事件无法从即时文件初始化中受益。

配置 Always On 可用性组

如果您使用的是 SQL Server 版本 2012 及更高版本的本机客户端库和.NET Framework 4.5 库，则可以使用MultiSubnetFailover参数来更改连接行为。建议您将该参数设置为 TRUE。这将使用 Always On 可用性组实现更快的故障转移。

> **Note**
>
> 如果您的旧应用程序无法使用该MultiSubnetFailover参数，则可以在您的 SQL Server 实例前面放置一个 Network Load Balancer。平衡器使用运行状况检查来确定哪个 SQL Server 数据库处于活动状态，并将流量发送到当前托管该数据库的实例。负载均衡器跨越一个或多个可用区。您可以使用专用端口（例如 59999）进行运行状况检查，然后修改集群组参数以响应该端口。这使您无需使用MultiSubnetFailover参数即可将 SQL Server 故障转移时间缩短至大约一分钟。有关详细说明，请参阅博客文章使用 Network Load Balancer 缩短 Amazon EC2 实例上 SQL Server 的故障转移时间。

以下两个设置会影响可用性组侦听器在 DNS 中注册的方式：RegisterAllProvidersIP 和 HostRecordTTL。

使用 “始终开启” 可用性组时，将 RegisterAllProviders IP 设置为 true

我们建议您将 RegisterAllProvidersIP 设置为 1 (t rue)。在创建可用性组侦听器时将 RegisterAllProvidersIP 设置为时1，该监听器的所有 IP 地址都将注册到 DNS 中。当 RegisterAllProvidersIP 设置为0（假）时，仅注册一个活动 IP。

在故障转移的情况下，当主副本从一个子网移动到另一个子网时，将取消注册旧 IP 地址，并注册新 IP 地址。当可用性组侦听器联机时，DNS 将使用新 IP 进行更新。但是，在当前缓存的条目到期之前，客户端系统不会将侦听器名称解析为新的 IP 地址。

使用 Always On 可用性组时，将 HostRecord TTL 设置为 60 或更小

HostRecordTTL 设置控制缓存 DNS 条目的生存时间 (TTL)。默认值为 1200 秒。我们建议您将 HostRecordTTL 更改为低得多的设置（60 秒或更短）。这会导致缓存的值更快地过期，因此在故障转移的情况下，客户端系统可以更快地解析新 IP。

禁用 Always On 集群组的自动故障恢复

确认 Windows 集群管理器中的 Always On 可用性组已禁用自动故障恢复。

配置备份

如优化磁盘布局或文件分发部分所述，我们建议您将本地 SQL Server 备份发送到单独的驱动器。还要考虑对备份文件所在的 EBS 卷进行定时快照。

改善数据库优化

本节提供了在使用 SQL Server 查询优化器时提高性能的最佳实践。它讨论了定期重建索引和更新表统计信息如何帮助优化执行计划。以下各节提供了配置提示和最佳实践。

主题

重建索引

为了使查询优化器生成最佳查询计划并使用正确的索引，索引不应分段。根据更新、插入或删除速率，索引会随着时间的推移而变得分散。确保定期对表进行重新索引。重建频率取决于数据库处理数据操纵语言 (DML) 操作的速率。

一个好的起点是重建碎片率超过 30% 的索引，重组碎片率低于 30% 的索引。30% 的值适用于大多数用例，但是如果您仍然看到由于未使用的索引而导致查询计划不佳，则可能需要重新考虑这个百分比。

使用如下查询来检查碎片：

```
SELECT OBJECT_NAME(OBJECT_ID), index_id,index_type_desc,index_level,
avg_fragmentation_in_percent,avg_page_space_used_in_percent,page_count
FROM sys.dm_db_index_physical_stats
(DB_ID(N'<your_database>'), NULL, NULL, NULL , 'SAMPLED')
ORDER BY avg_fragmentation_in_percent DESC
```

建议您创建维护任务，以定期重建索引。

更新统计数据

与碎片索引一样，如果优化器没有关于表列的键值分布（统计 up-to-date信息）的信息，则无法生成最佳执行计划。建议您定期更新所有表的统计信息。更新的频率取决于数据库处理 DML 操作的速率，但通常在非高峰时段每周运行两次。但是，请避免在重建索引的当天更新统计数据。有关更新统计数据的详细信息，请参阅 Microsoft SQL Server 文档。

为了优化数据库，我们建议使用索引和统计信息维护脚本。有关示例，请参阅 SQL Server 维护解决方案网站上提供的 SQL Server 索引和统计信息维护脚本。

EC2 使用 La AWS unch Wizard 在亚马逊上优化 SQL Server 部署

AWS Launch Wizard 是在亚马逊上部署 SQL Server 单实例和高可用性 (HA) 的主要方法 EC2。Launch Wizard 部署基于 AWS Well-Architected Framework，并针对安全性、可靠性、性能效率和成本节省进行了优化。

Launch Wizard 简化了您的 SQL Server 部署，还使配置 SQL Server 变得更加容易。其功能包括：

- 自动选择 AWS 资源 — Launch Wizard 可以根据您的虚拟 CPU (vCPU)、内存和网络要求推荐最佳实例类型。它还可以根据存储驱动器和吞吐量推荐卷类型。
- 一键监控 — Launch Wizard 与 Amazon App CloudWatch lication Insig hts 集成，可以设置对 SQL Server HA 部署的监控 AWS选择此选项后，Application Insights 会自动设置相关指标、日志和警报 CloudWatch，并开始监控新部署的工作负载。
- 应用程序资源组便于发现 — Launch Wizard 会为您的 SQL Server 应用程序创建的所有 AWS 资源创建一个资源组。您可以从 AWS Systems Manager 控制台管理、修补和维护 SQL Server 应用程序。

Launch Wizard 为您提供可重复使用的 AWS CloudFormation 代码模板。这些模板可以作为后续应用程序部署的基准。要了解更多信息，请参阅 AWS Launch Wizard 概述和用户指南。

后续步骤

本指南介绍了在亚马逊上配置和运行 Microsoft SQL Server 工作负载的一些最佳实践 EC2。在迁移过程的规划和实施阶段遵循这些准则将有助于您在生产环境中建立稳定的服务器。

有关这些配置任务的更多信息，请参阅每个部分中提供的链接，并访问其他资源一节中列出的网页。

其他资源

相关策略、指南和模式

- 关系数据库的迁移策略
- SQL Server 模式：
  - 所有模式
  - 重新托管模式（将 SQL Server 迁移到亚马逊 EC2）
  - 更换平台模式（从 SQL Server 迁移到 Amazon RDS for SQL Server）
  - 重新架构模式（将 SQL Server 迁移到开源和 AWS 云原生数据库）
- AWS 规范性指导网站

AWS resources

- AWS 文档
- AWS 一般参考
- AWS 术语表

AWS 服务

- Amazon EBS
- Amazon EC2

其他资源

- 如何将 Micrsoft SQL Server tempdb 移动到亚马逊上的实例/临时磁盘 EC2
- 启动时对 Windows 临时磁盘进行条带化
- 我的 SQL Server 实际需要多少内存？
- SQL Server 等待统计（或者请告诉我哪里有问题…）
- 多子网可用性组中的连接超时
- 规划缓存并针对临时工作负载进行优化
- Windows 中的 RAM、虚拟内存、页面文件和内存管理
- 如何确定适用于 64 位版本的 Windows 的相应页面文件大小

文档历史记录

下表介绍了本指南的一些重要更改。如果您希望收到有关未来更新的通知，可以订阅 RSS 源。

变更	说明	日期
更正的信息	更正了有关并行度属性的成本阈值的信息。它的价值是用成本而不是时间单位来衡量的。	2023 年 12 月 4 日
更新指南	更新有关设置 NTFS 分配单元大小、在内存中锁定页面、使用任务卸载功能、使用条带化、更改并行度成本阈值、使用跟踪标志和使用数据库自动增长设置的部分。	2023 年 8 月 8 日
新增指导	添加了有关对无法使用该MultiSubnetFailover参数的旧版应用程序使用 Network Load Bal ancer 的信息。	2022 年 11 月 11 日
固定代码	更正了初始化实例存储部分中的 PowerShell 代码。	2022 年 6 月 27 日
新增部分	添加了 AWS Launch Wizard for SQL Server 的信息。	2021 年 8 月 18 日
初次发布	—	2020 年 7 月 21 日

AWS 规范性指导词汇表

以下是 AWS 规范性指导提供的策略、指南和模式中的常用术语。若要推荐词条，请使用术语表末尾的提供反馈链接。

数字

7 R

将应用程序迁移到云中的 7 种常见迁移策略。这些策略以 Gartner 于 2011 年确定的 5 R 为基础，包括以下内容：

- 重构/重新架构 - 充分利用云原生功能来提高敏捷性、性能和可扩展性，以迁移应用程序并修改其架构。这通常涉及到移植操作系统和数据库。示例：将您的本地 Oracle 数据库迁移到兼容 Amazon Aurora PostgreSQL 的版本。
- 更换平台 - 将应用程序迁移到云中，并进行一定程度的优化，以利用云功能。示例：在中将您的本地 Oracle 数据库迁移到适用于 Oracle 的亚马逊关系数据库服务 (Amazon RDS) AWS Cloud。
- 重新购买 - 转换到其他产品，通常是从传统许可转向 SaaS 模式。示例：将您的客户关系管理 (CRM) 系统迁移到 Salesforce.com。
- 更换主机（直接迁移）- 将应用程序迁移到云中，无需进行任何更改即可利用云功能。示例：在中的 EC2 实例上将您的本地 Oracle 数据库迁移到 Oracle AWS Cloud。
- 重新定位（虚拟机监控器级直接迁移）：将基础设施迁移到云中，无需购买新硬件、重写应用程序或修改现有操作。您可以将服务器从本地平台迁移到同一平台的云服务。示例：将Microsoft Hyper-V应用程序迁移到 AWS。
- 保留（重访）- 将应用程序保留在源环境中。其中可能包括需要进行重大重构的应用程序，并且您希望将工作推迟到以后，以及您希望保留的遗留应用程序，因为迁移它们没有商业上的理由。
- 停用 - 停用或删除源环境中不再需要的应用程序。

A

ABAC

请参阅基于属性的访问控制。

抽象服务

参见托管服务。

ACID

参见原子性、一致性、隔离性、耐久性。

主动-主动迁移

一种数据库迁移方法，在这种方法中，源数据库和目标数据库保持同步（通过使用双向复制工具或双写操作），两个数据库都在迁移期间处理来自连接应用程序的事务。这种方法支持小批量、可控的迁移，而不需要一次性割接。与主动-被动迁移相比，它更灵活，但需要更多的工作。

主动-被动迁移

一种数据库迁移方法，在这种方法中，源数据库和目标数据库保持同步，但在将数据复制到目标数据库时，只有源数据库处理来自连接应用程序的事务。目标数据库在迁移期间不接受任何事务。

聚合函数

一个 SQL 函数，它对一组行进行操作并计算该组的单个返回值。聚合函数的示例包括SUM和MAX。

AI

参见人工智能。

AIOps

参见人工智能操作。

匿名化

永久删除数据集中个人信息的过程。匿名化可以帮助保护个人隐私。匿名化数据不再被视为个人数据。

反模式

一种用于解决反复出现的问题的常用解决方案，而在这类问题中，此解决方案适得其反、无效或不如替代方案有效。

应用程序控制

一种安全方法，仅允许使用经批准的应用程序，以帮助保护系统免受恶意软件的侵害。

应用程序组合

有关组织使用的每个应用程序的详细信息的集合，包括构建和维护该应用程序的成本及其业务价值。这些信息是产品组合发现和分析过程的关键，有助于识别需要进行迁移、现代化和优化的应用程序并确定其优先级。

人工智能（AI）

计算机科学领域致力于使用计算技术执行通常与人类相关的认知功能，例如学习、解决问题和识别模式。有关更多信息，请参阅什么是人工智能？

人工智能操作 (AIOps)

使用机器学习技术解决运营问题、减少运营事故和人为干预以及提高服务质量的过程。有关如何在 AIOps AWS 迁移策略中使用的更多信息，请参阅操作集成指南。

非对称加密

一种加密算法，使用一对密钥，一个公钥用于加密，一个私钥用于解密。您可以共享公钥，因为它不用于解密，但对私钥的访问应受到严格限制。

原子性、一致性、隔离性、持久性（ACID）

一组软件属性，即使在出现错误、电源故障或其他问题的情况下，也能保证数据库的数据有效性和操作可靠性。

基于属性的访问权限控制（ABAC）

根据用户属性（如部门、工作角色和团队名称）创建精细访问权限的做法。有关更多信息，请参阅 AWS Identity and Access Management (I AM) 文档 AWS中的 AB AC。

权威数据源

存储主要数据版本的位置，被认为是最可靠的信息源。您可以将数据从权威数据源复制到其他位置，以便处理或修改数据，例如对数据进行匿名化、编辑或假名化。

可用区

中的一个不同位置 AWS 区域 ，不受其他可用区域故障的影响，并向同一区域中的其他可用区提供低成本、低延迟的网络连接。

AWS 云采用框架 (AWS CAF)

该框架包含指导方针和最佳实践 AWS ，可帮助组织制定高效且有效的计划，以成功迁移到云端。AWS CAF将指导分为六个重点领域，称为视角：业务、人员、治理、平台、安全和运营。业务、人员和治理角度侧重于业务技能和流程；平台、安全和运营角度侧重于技术技能和流程。例如，人员角度针对的是负责人力资源（HR）、人员配置职能和人员管理的利益相关者。从这个角度来看，AWS CAF 为人员发展、培训和沟通提供了指导，以帮助组织为成功采用云做好准备。有关更多信息，请参阅 AWS CAF 网站和 AWS CAF 白皮书。

AWS 工作负载资格框架 (AWS WQF)

一种评估数据库迁移工作负载、推荐迁移策略和提供工作估算的工具。 AWS WQF 包含在 AWS Schema Conversion Tool (AWS SCT) 中。它用来分析数据库架构和代码对象、应用程序代码、依赖关系和性能特征，并提供评测报告。

B

坏机器人

旨在破坏个人或组织或对其造成伤害的机器人。

BCP

参见业务连续性计划。

行为图

一段时间内资源行为和交互的统一交互式视图。您可以使用 Amazon Detective 的行为图来检查失败的登录尝试、可疑的 API 调用和类似的操作。有关更多信息，请参阅 Detective 文档中的行为图中的数据。

大端序系统

一个先存储最高有效字节的系统。另请参见字节顺序。

二进制分类

一种预测二进制结果（两个可能的类别之一）的过程。例如，您的 ML 模型可能需要预测诸如“该电子邮件是否为垃圾邮件？”或“这个产品是书还是汽车？”之类的问题

bloom 筛选条件

一种概率性、内存高效的数据结构，用于测试元素是否为集合的成员。

蓝/绿部署

一种部署策略，您可以创建两个独立但完全相同的环境。在一个环境中运行当前的应用程序版本（蓝色），在另一个环境中运行新的应用程序版本（绿色）。此策略可帮助您在影响最小的情况下快速回滚。

自动程序

一种通过互联网运行自动任务并模拟人类活动或互动的软件应用程序。有些机器人是有用或有益的，例如在互联网上索引信息的网络爬虫。其他一些被称为恶意机器人的机器人旨在破坏个人或组织或对其造成伤害。

僵尸网络

被恶意软件感染并受单方（称为机器人牧民或机器人操作员）控制的机器人网络。僵尸网络是最著名的扩展机器人及其影响力的机制。

分支

代码存储库的一个包含区域。在存储库中创建的第一个分支是主分支。您可以从现有分支创建新分支，然后在新分支中开发功能或修复错误。为构建功能而创建的分支通常称为功能分支。当功能可以发布时，将功能分支合并回主分支。有关更多信息，请参阅关于分支（GitHub 文档）。

破碎的玻璃通道

在特殊情况下，通过批准的流程，用户 AWS 账户 可以快速访问他们通常没有访问权限的内容。有关更多信息，请参阅 Well -Architected 指南中的 “实施破碎玻璃程序” 指示 AWS 器。

棕地策略

您环境中的现有基础设施。在为系统架构采用棕地策略时，您需要围绕当前系统和基础设施的限制来设计架构。如果您正在扩展现有基础设施，则可以将棕地策略和全新策略混合。

缓冲区缓存

存储最常访问的数据的内存区域。

业务能力

企业如何创造价值（例如，销售、客户服务或营销）。微服务架构和开发决策可以由业务能力驱动。有关更多信息，请参阅在 AWS上运行容器化微服务白皮书中的围绕业务能力进行组织部分。

业务连续性计划（BCP）

一项计划，旨在应对大规模迁移等破坏性事件对运营的潜在影响，并使企业能够快速恢复运营。

C

CAF

参见AWS 云采用框架。

金丝雀部署

向最终用户缓慢而渐进地发布版本。当你有信心时，你可以部署新版本并全部替换当前版本。

CCoE

参见云卓越中心。

CDC

请参阅变更数据捕获。

更改数据捕获（CDC）

跟踪数据来源（如数据库表）的更改并记录有关更改的元数据的过程。您可以将 CDC 用于各种目的，例如审计或复制目标系统中的更改以保持同步。

混沌工程

故意引入故障或破坏性事件来测试系统的弹性。您可以使用 AWS Fault Injection Service (AWS FIS) 来执行实验，对您的 AWS 工作负载施加压力并评估其响应。

CI/CD

查看持续集成和持续交付。

分类

一种有助于生成预测的分类流程。分类问题的 ML 模型预测离散值。离散值始终彼此不同。例如，一个模型可能需要评估图像中是否有汽车。

客户端加密

在目标 AWS 服务 收到数据之前，对数据进行本地加密。

云卓越中心 (CCoE)

一个多学科团队，负责推动整个组织的云采用工作，包括开发云最佳实践、调动资源、制定迁移时间表、领导组织完成大规模转型。有关更多信息，请参阅 AWS Cloud 企业战略博客上的 CCoE 帖子。

云计算

通常用于远程数据存储和 IoT 设备管理的云技术。云计算通常与边缘计算技术相关。

云运营模型

在 IT 组织中，一种用于构建、完善和优化一个或多个云环境的运营模型。有关更多信息，请参阅构建您的云运营模型。

云采用阶段

组织迁移到以下阶段时通常会经历四个阶段 AWS Cloud：

- 项目 - 出于概念验证和学习目的，开展一些与云相关的项目
- 基础 — 进行基础投资以扩大云采用率（例如，创建着陆区、定义 CCo E、建立运营模型）
- 迁移 - 迁移单个应用程序
- 重塑 - 优化产品和服务，在云中创新

Stephen Orban在 AWS Cloud 企业战略博客的博客文章《云优先之旅和采用阶段》中定义了这些阶段。有关它们与 AWS 迁移策略的关系的信息，请参阅迁移准备指南。

CMDB

参见配置管理数据库。

代码存储库

通过版本控制过程存储和更新源代码和其他资产（如文档、示例和脚本）的位置。常见的云存储库包括GitHub或Bitbucket Cloud。每个版本的代码都称为一个分支。在微服务结构中，每个存储库都专门用于一个功能。单个 CI/CD 管道可以使用多个存储库。

冷缓存

一种空的、填充不足或包含过时或不相关数据的缓冲区缓存。这会影响性能，因为数据库实例必须从主内存或磁盘读取，这比从缓冲区缓存读取要慢。

冷数据

很少访问的数据，且通常是历史数据。查询此类数据时，通常可以接受慢速查询。将这些数据转移到性能较低且成本更低的存储层或类别可以降低成本。

计算机视觉 (CV)

人工智能领域，使用机器学习来分析和提取数字图像和视频等视觉格式的信息。例如，Amazon SageMaker AI 为 CV 提供了图像处理算法。

配置偏差

对于工作负载，配置会从预期状态发生变化。这可能会导致工作负载变得不合规，而且通常是渐进的，不是故意的。

配置管理数据库（CMDB）

一种存储库，用于存储和管理有关数据库及其 IT 环境的信息，包括硬件和软件组件及其配置。您通常在迁移的产品组合发现和分析阶段使用来自 CMDB 的数据。

合规性包

一系列 AWS Config 规则和补救措施，您可以汇编这些规则和补救措施，以自定义合规性和安全性检查。您可以使用 YAML 模板将一致性包作为单个实体部署在 AWS 账户 和区域或整个组织中。有关更多信息，请参阅 AWS Config 文档中的一致性包。

持续集成和持续交付（CI/CD）

自动执行软件发布过程的源代码、构建、测试、暂存和生产阶段的过程。 CI/CD is commonly described as a pipeline. CI/CD可以帮助您实现流程自动化、提高生产力、提高代码质量和更快地交付。有关更多信息，请参阅持续交付的优势。CD 也可以表示持续部署。有关更多信息，请参阅持续交付与持续部署。

CV

参见计算机视觉。

D

静态数据

网络中静止的数据，例如存储中的数据。

数据分类

根据网络中数据的关键性和敏感性对其进行识别和分类的过程。它是任何网络安全风险管理策略的关键组成部分，因为它可以帮助您确定对数据的适当保护和保留控制。数据分类是 Well-Architecte AWS d Framework 中安全支柱的一个组成部分。有关详细信息，请参阅数据分类。

数据漂移

生产数据与用来训练机器学习模型的数据之间的有意义差异，或者输入数据随时间推移的有意义变化。数据漂移可能降低机器学习模型预测的整体质量、准确性和公平性。

传输中数据

在网络中主动移动的数据，例如在网络资源之间移动的数据。

数据网格

一种架构框架，可提供分布式、去中心化的数据所有权以及集中式管理和治理。

数据最少化

仅收集并处理绝对必要数据的原则。在中进行数据最小化 AWS Cloud 可以降低隐私风险、成本和分析碳足迹。

数据边界

AWS 环境中的一组预防性防护措施，可帮助确保只有可信身份才能访问来自预期网络的可信资源。有关更多信息，请参阅在上构建数据边界。 AWS

数据预处理

将原始数据转换为 ML 模型易于解析的格式。预处理数据可能意味着删除某些列或行，并处理缺失、不一致或重复的值。

数据溯源

在数据的整个生命周期跟踪其来源和历史的过程，例如数据如何生成、传输和存储。

数据主体

正在收集和处理其数据的个人。

数据仓库

一种支持商业智能（例如分析）的数据管理系统。数据仓库通常包含大量历史数据，通常用于查询和分析。

数据库定义语言（DDL）

在数据库中创建或修改表和对象结构的语句或命令。

数据库操作语言（DML）

在数据库中修改（插入、更新和删除）信息的语句或命令。

DDL

参见数据库定义语言。

深度融合

组合多个深度学习模型进行预测。您可以使用深度融合来获得更准确的预测或估算预测中的不确定性。

深度学习

一个 ML 子字段使用多层人工神经网络来识别输入数据和感兴趣的目标变量之间的映射。

defense-in-depth

一种信息安全方法，经过深思熟虑，在整个计算机网络中分层实施一系列安全机制和控制措施，以保护网络及其中数据的机密性、完整性和可用性。当你采用这种策略时 AWS，你会在 AWS

Organizations 结构的不同层面添加多个控件来帮助保护资源。例如，一种 defense-in-depth方法可以结合多因素身份验证、网络分段和加密。

委托管理员

在中 AWS Organizations，兼容的服务可以注册 AWS 成员帐户来管理组织的帐户并管理该服务的权限。此账户被称为该服务的委托管理员。有关更多信息和兼容服务列表，请参阅 AWS Organizations 文档中使用 AWS Organizations的服务。

后

使应用程序、新功能或代码修复在目标环境中可用的过程。部署涉及在代码库中实现更改，然后在应用程序的环境中构建和运行该代码库。

开发环境

参见环境。

侦测性控制

一种安全控制，在事件发生后进行检测、记录日志和发出警报。这些控制是第二道防线，提醒您注意绕过现有预防性控制的安全事件。有关更多信息，请参阅在 AWS上实施安全控制中的侦测性控制。

开发价值流映射 (DVSM)

用于识别对软件开发生命周期中的速度和质量产生不利影响的限制因素并确定其优先级的流程。DVSM 扩展了最初为精益生产实践设计的价值流映射流程。其重点关注在软件开发过程中创造和转移价值所需的步骤和团队。

数字孪生

真实世界系统的虚拟再现，如建筑物、工厂、工业设备或生产线。数字孪生支持预测性维护、远程监控和生产优化。

维度表

在星型架构中，一种较小的表，其中包含事实表中有关定量数据的数据属性。维度表属性通常是文本字段或行为类似于文本的离散数字。这些属性通常用于查询约束、筛选和结果集标注。

灾难

阻止工作负载或系统在其主要部署位置实现其业务目标的事件。这些事件可能是自然灾害、技术故障或人为操作的结果，例如无意的配置错误或恶意软件攻击。

灾难恢复 (DR)

您用来最大限度地减少灾难造成的停机时间和数据丢失的策略和流程。有关更多信息，请参阅 Well-Architected Fr ame AWS work 中的 “工作负载灾难恢复：云端 AWS 恢复”。

DML

参见数据库操作语言。

领域驱动设计

一种开发复杂软件系统的方法，通过将其组件连接到每个组件所服务的不断发展的领域或核心业务目标。Eric Evans 在其著作领域驱动设计：软件核心复杂性应对之道（Boston: Addison-Wesley Professional, 2003）中介绍了这一概念。有关如何将领域驱动设计与 strangler fig 模式结合使用的信息，请参阅使用容器和 Amazon API Gateway 逐步将原有的 Microsoft ASP.NET（ASMX）Web 服务现代化。

DR

参见灾难恢复。

漂移检测

跟踪与基准配置的偏差。例如，您可以使用 AWS CloudFormation 来检测系统资源中的偏差，也可以使用 AWS Control Tower 来检测着陆区中可能影响监管要求合规性的变化。

DVSM

参见开发价值流映射。

E

EDA

参见探索性数据分析。

EDI

参见电子数据交换。

边缘计算

该技术可提高位于 IoT 网络边缘的智能设备的计算能力。与云计算相比，边缘计算可以减少通信延迟并缩短响应时间。

电子数据交换 (EDI)

组织之间自动交换业务文档。有关更多信息，请参阅什么是电子数据交换。

加密

一种将人类可读的纯文本数据转换为密文的计算过程。

加密密钥

由加密算法生成的随机位的加密字符串。密钥的长度可能有所不同，而且每个密钥都设计为不可预测且唯一。

字节顺序

字节在计算机内存中的存储顺序。大端序系统先存储最高有效字节。小端序系统先存储最低有效字节。

端点

参见服务端点。

端点服务

一种可以在虚拟私有云（VPC）中托管，与其他用户共享的服务。您可以使用其他 AWS 账户 或 AWS Identity and Access Management (IAM) 委托人创建终端节点服务， AWS PrivateLink 并向其授予权限。这些账户或主体可通过创建接口 VPC 端点来私密地连接到您的端点服务。有关更多信息，请参阅 Amazon Virtual Private Cloud（Amazon VPC）文档中的创建端点服务。

企业资源规划 (ERP)

一种自动化和管理企业关键业务流程（例如会计、MES 和项目管理）的系统。

信封加密

用另一个加密密钥对加密密钥进行加密的过程。有关更多信息，请参阅 AWS Key Management Service (AWS KMS) 文档中的信封加密。

环境

正在运行的应用程序的实例。以下是云计算中常见的环境类型：

- 开发环境 — 正在运行的应用程序的实例，只有负责维护应用程序的核心团队才能使用。开发环境用于测试更改，然后再将其提升到上层环境。这类环境有时称为测试环境。
- 下层环境 — 应用程序的所有开发环境，比如用于初始构建和测试的环境。

- 生产环境 — 最终用户可以访问的正在运行的应用程序的实例。在 CI/CD 管道中，生产环境是最后一个部署环境。
- 上层环境 — 除核心开发团队以外的用户可以访问的所有环境。这可能包括生产环境、预生产环境和用户验收测试环境。

epic

在敏捷方法学中，有助于组织工作和确定优先级的功能类别。epics 提供了对需求和实施任务的总体描述。例如， AWS CAF 安全史诗包括身份和访问管理、侦探控制、基础设施安全、数据保护和事件响应。有关 AWS 迁移策略中 epics 的更多信息，请参阅计划实施指南。

ERP

参见企业资源规划。

探索性数据分析（EDA）

分析数据集以了解其主要特征的过程。您收集或汇总数据，并进行初步调查，以发现模式、检测异常并检查假定情况。EDA 通过计算汇总统计数据和创建数据可视化得以执行。

F

事实表

星形架构中的中心表。它存储有关业务运营的定量数据。通常，事实表包含两种类型的列：包含度量的列和包含维度表外键的列。

失败得很快

一种使用频繁和增量测试来缩短开发生命周期的理念。这是敏捷方法的关键部分。

故障隔离边界

在中 AWS Cloud，诸如可用区 AWS 区域、控制平面或数据平面之类的边界，它限制了故障的影响并有助于提高工作负载的弹性。有关更多信息，请参阅AWS 故障隔离边界。

功能分支

参见分支。

特征

您用来进行预测的输入数据。例如，在制造环境中，特征可能是定期从生产线捕获的图像。

特征重要性

特征对于模型预测的重要性。这通常表示为数值分数，可以通过各种技术进行计算，例如 Shapley 加法解释（SHAP）和积分梯度。有关更多信息，请参阅使用机器学习模型的可解释性 AWS。

功能转换

为 ML 流程优化数据，包括使用其他来源丰富数据、扩展值或从单个数据字段中提取多组信息。这使得 ML 模型能从数据中获益。例如，如果您将“2021-05-27 00:15:37”日期分解为“2021”、“五月”、“星期四”和“15”，则可以帮助学习与不同数据成分相关的算法学习精细模式。

少量提示

在要求法学硕士执行类似任务之前，向其提供少量示例，以演示该任务和所需的输出。这种技术是情境学习的应用，模型可以从提示中嵌入的示例（镜头）中学习。对于需要特定格式、推理或领域知识的任务，Few-shot 提示可能非常有效。另请参见零镜头提示。

FGAC

请参阅精细的访问控制。

精细访问控制 (FGAC)

使用多个条件允许或拒绝访问请求。

快闪迁移

一种数据库迁移方法，它使用连续的数据复制，通过更改数据捕获在尽可能短的时间内迁移数据，而不是使用分阶段的方法。目标是将停机时间降至最低。

FM

参见基础模型。

基础模型 (FM)

一个大型深度学习神经网络，一直在广义和未标记数据的大量数据集上进行训练。 FMs 能够执行各种各样的一般任务，例如理解语言、生成文本和图像以及用自然语言进行对话。有关更多信息，请参阅什么是基础模型。

G

生成式人工智能

人工智能模型的一个子集，这些模型已经过大量数据训练，可以使用简单的文本提示来创建新的内容和工件，例如图像、视频、文本和音频。有关更多信息，请参阅什么是生成式 AI。

地理封锁

请参阅地理限制。

地理限制（地理阻止）

在 Amazon 中 CloudFront，一种阻止特定国家/地区的用户访问内容分发的选项。您可以使用允许列表或阻止列表来指定已批准和已禁止的国家/地区。有关更多信息，请参阅 CloudFront 文档中的限制内容的地理分布。

GitFlow 工作流程

一种方法，在这种方法中，下层和上层环境在源代码存储库中使用不同的分支。Gitflow 工作流程被认为是传统的，而基于主干的工作流程是现代的首选方法。

金色影像

系统或软件的快照，用作部署该系统或软件的新实例的模板。例如，在制造业中，黄金映像可用于在多个设备上配置软件，并有助于提高设备制造运营的速度、可扩展性和生产力。

全新策略

在新环境中缺少现有基础设施。在对系统架构采用全新策略时，您可以选择所有新技术，而不受对现有基础设施（也称为棕地）兼容性的限制。如果您正在扩展现有基础设施，则可以将棕地策略和全新策略混合。

防护机制

一项高级规则，可帮助管理各组织单位的资源、策略和合规性 (OUs)。预防性防护机制会执行策略以确保符合合规性标准。它们是使用服务控制策略和 IAM 权限边界实现的。侦测性防护机制会检测策略违规和合规性问题，并生成警报以进行修复。它们通过使用 AWS Config、、Amazon、 AWS Security Hub GuardDuty AWS Trusted Advisor、Amazon Inspector 和自定义 AWS Lambda 支票来实现。

H

HA

参见高可用性。

异构数据库迁移

将源数据库迁移到使用不同数据库引擎的目标数据库（例如，从 Oracle 迁移到 Amazon Aurora）。异构迁移通常是重新架构工作的一部分，而转换架构可能是一项复杂的任务。AWS 提供了 AWS SCT 来帮助实现架构转换。

高可用性 (HA)

在遇到挑战或灾难时，工作负载无需干预即可连续运行的能力。HA 系统旨在自动进行故障转移、持续提供良好性能，并以最小的性能影响处理不同负载和故障。

历史数据库现代化

一种用于实现运营技术 (OT) 系统现代化和升级以更好满足制造业需求的方法。历史数据库是一种用于收集和存储工厂中各种来源数据的数据库。

抵制数据

从用于训练机器学习模型的数据集中扣留的一部分带有标签的历史数据。通过将模型预测与抵制数据进行比较，您可以使用抵制数据来评估模型性能。

同构数据库迁移

将源数据库迁移到共享同一数据库引擎的目标数据库（例如，从 Microsoft SQL Server 迁移到 Amazon RDS for SQL Server）。同构迁移通常是更换主机或更换平台工作的一部分。您可以使用本机数据库实用程序来迁移架构。

热数据

经常访问的数据，例如实时数据或近期的转化数据。这些数据通常需要高性能存储层或存储类别才能提供快速的查询响应。

修补程序

针对生产环境中关键问题的紧急修复。由于其紧迫性，修补程序通常是在典型的 DevOps 发布工作流程之外进行的。

hypercare 周期

割接之后，迁移团队立即管理和监控云中迁移的应用程序以解决任何问题的时间段。通常，这个周期持续 1-4 天。在 hypercare 周期结束时，迁移团队通常会将应用程序的责任移交给云运营团队。

我

IaC

参见基础设施即代码。

基于身份的策略

附加到一个或多个 IAM 委托人的策略，用于定义他们在 AWS Cloud 环境中的权限。

空闲应用程序

90 天内平均 CPU 和内存使用率在 5% 到 20% 之间的应用程序。在迁移项目中，通常会停用这些应用程序或将其保留在本地。

IIoT

参见工业物联网。

不可变的基础架构

一种为生产工作负载部署新基础架构，而不是更新、修补或修改现有基础架构的模型。不可变基础架构本质上比可变基础架构更一致、更可靠、更可预测。有关更多信息，请参阅 Well-Architected Framework 中的使用不可变基础架构 AWS 部署最佳实践。

入站（入口）VPC

在 AWS 多账户架构中，一种接受、检查和路由来自应用程序外部的网络连接的 VPC。AWS 安全参考架构建议设置您的网络帐户，包括入站、出站和检查， VPCs 以保护您的应用程序与更广泛的互联网之间的双向接口。

增量迁移

一种割接策略，在这种策略中，您可以将应用程序分成小部分进行迁移，而不是一次性完整割接。例如，您最初可能只将几个微服务或用户迁移到新系统。在确认一切正常后，您可以逐步迁移其他微服务或用户，直到停用遗留系统。这种策略降低了大规模迁移带来的风险。

工业 4.0

该术语由克劳斯·施瓦布（Klaus Schwab）于2016年推出，指的是通过连接、实时数据、自动化、分析和人工智能/机器学习的进步实现制造流程的现代化。

基础设施

应用程序环境中包含的所有资源和资产。

基础设施即代码（IaC）

通过一组配置文件预置和管理应用程序基础设施的过程。IaC 旨在帮助您集中管理基础设施、实现资源标准化和快速扩展，使新环境具有可重复性、可靠性和一致性。

工业物联网 (IIoT)

在工业领域使用联网的传感器和设备，例如制造业、能源、汽车、医疗保健、生命科学和农业。有关更多信息，请参阅制定工业物联网 (IIoT) 数字化转型战略。

检查 VPC

在 AWS 多账户架构中，一种集中式 VPC，用于管理对 VPCs（相同或不同 AWS 区域）、互联网和本地网络之间的网络流量的检查。AWS 安全参考架构建议设置您的网络帐户，包括入站、出站和检查， VPCs 以保护您的应用程序与更广泛的互联网之间的双向接口。

物联网（IoT）

由带有嵌入式传感器或处理器的连接物理对象组成的网络，这些传感器或处理器通过互联网或本地通信网络与其他设备和系统进行通信。有关更多信息，请参阅什么是 IoT？

可解释性

它是机器学习模型的一种特征，描述了人类可以理解模型的预测如何取决于其输入的程度。有关更多信息，请参阅使用机器学习模型的可解释性 AWS。

IoT

参见物联网。

IT 信息库（ITIL）

提供 IT 服务并使这些服务符合业务要求的一套最佳实践。ITIL 是 ITSM 的基础。

IT 服务管理（ITSM）

为组织设计、实施、管理和支持 IT 服务的相关活动。有关将云运营与 ITSM 工具集成的信息，请参阅运营集成指南。

ITIL

请参阅 IT 信息库。

ITSM

请参阅 IT 服务管理。

L

基于标签的访问控制 (LBAC)

强制访问控制 (MAC) 的一种实施方式，其中明确为用户和数据本身分配了安全标签值。用户安全标签和数据安全标签之间的交集决定了用户可以看到哪些行和列。

登录区

landing zone 是一个架构精良的多账户 AWS 环境，具有可扩展性和安全性。这是一个起点，您的组织可以从这里放心地在安全和基础设施环境中快速启动和部署工作负载和应用程序。有关登录区的更多信息，请参阅设置安全且可扩展的多账户 AWS 环境。

大型语言模型 (LLM)

一种基于大量数据进行预训练的深度学习 AI 模型。法学硕士可以执行多项任务，例如回答问题、总结文档、将文本翻译成其他语言以及完成句子。有关更多信息，请参阅什么是 LLMs。

大规模迁移

迁移 300 台或更多服务器。

LBAC

请参阅基于标签的访问控制。

最低权限

授予执行任务所需的最低权限的最佳安全实践。有关更多信息，请参阅 IAM 文档中的应用最低权限许可。

直接迁移

见 7 R。

小端序系统

一个先存储最低有效字节的系统。另请参见字节顺序。

LLM

参见大型语言模型。

下层环境

参见环境。

M

机器学习（ML）

一种使用算法和技术进行模式识别和学习的人工智能。ML 对记录的数据（例如物联网（IoT）数据）进行分析和学习，以生成基于模式的统计模型。有关更多信息，请参阅机器学习。

主分支

参见分支。

恶意软件

旨在危害计算机安全或隐私的软件。恶意软件可能会破坏计算机系统、泄露敏感信息或获得未经授权的访问。恶意软件的示例包括病毒、蠕虫、勒索软件、特洛伊木马、间谍软件和键盘记录器。

托管服务

AWS 服务 它 AWS 运行基础设施层、操作系统和平台，您可以访问端点来存储和检索数据。亚马逊简单存储服务 (Amazon S3) Service 和 Amazon DynamoDB 就是托管服务的示例。这些服务也称为抽象服务。

制造执行系统 (MES)

一种软件系统，用于跟踪、监控、记录和控制在车间将原材料转化为成品的生产过程。

MAP

参见迁移加速计划。

机制

一个完整的过程，在此过程中，您可以创建工具，推动工具的采用，然后检查结果以进行调整。机制是一种在运行过程中自我增强和改进的循环。有关更多信息，请参阅在 Well-Architect AWS ed 框架中构建机制。

成员账户

AWS 账户 除属于组织中的管理账户之外的所有账户 AWS Organizations。一个账户一次只能是一个组织的成员。

MES

参见制造执行系统。

消息队列遥测传输 (MQTT)

一种基于发布/订阅模式的轻量级 machine-to-machine (M2M) 通信协议，适用于资源受限的物联网设备。

微服务

一种小型的独立服务，通过明确的定义进行通信 APIs ，通常由小型的独立团队拥有。例如，保险系统可能包括映射到业务能力（如销售或营销）或子域（如购买、理赔或分析）的微服务。微服务

的好处包括敏捷、灵活扩展、易于部署、可重复使用的代码和恢复能力。有关更多信息，请参阅使用 AWS 无服务器服务集成微服务。

微服务架构

一种使用独立组件构建应用程序的方法，这些组件将每个应用程序进程作为微服务运行。这些微服务使用轻量级通过定义明确的接口进行通信。 APIs该架构中的每个微服务都可以更新、部署和扩展，以满足对应用程序特定功能的需求。有关更多信息，请参阅在上实现微服务。 AWS

迁移加速计划（MAP）

AWS 该计划提供咨询支持、培训和服务，以帮助组织为迁移到云奠定坚实的运营基础，并帮助抵消迁移的初始成本。MAP 提供了一种以系统的方式执行遗留迁移的迁移方法，以及一套用于自动执行和加速常见迁移场景的工具。

大规模迁移

将大部分应用程序组合分波迁移到云中的过程，在每一波中以更快的速度迁移更多应用程序。本阶段使用从早期阶段获得的最佳实践和经验教训，实施由团队、工具和流程组成的迁移工厂，通过自动化和敏捷交付简化工作负载的迁移。这是 AWS 迁移策略的第三阶段。

迁移工厂

跨职能团队，通过自动化、敏捷的方法简化工作负载迁移。迁移工厂团队通常包括运营、业务分析师和所有者、迁移工程师、开发 DevOps 人员和冲刺专业人员。20% 到 50% 的企业应用程序组合由可通过工厂方法优化的重复模式组成。有关更多信息，请参阅本内容集中有关迁移工厂的讨论和云迁移工厂指南。

迁移元数据

有关完成迁移所需的应用程序和服务器的信息。每种迁移模式都需要一套不同的迁移元数据。迁移元数据的示例包括目标子网、安全组和 AWS 账户。

迁移模式

一种可重复的迁移任务，详细列出了迁移策略、迁移目标以及所使用的迁移应用程序或服务。示例： EC2 使用 AWS 应用程序迁移服务重新托管向 Amazon 的迁移。

迁移组合评测（MPA）

一种在线工具，可提供信息，用于验证迁移到的业务案例。 AWS Cloud MPA 提供了详细的组合评测（服务器规模调整、定价、TCO 比较、迁移成本分析）以及迁移计划（应用程序数据分析和数据收集、应用程序分组、迁移优先级排序和波次规划）。所有 AWS 顾问和 APN 合作伙伴顾问均可免费使用 MPA 工具（需要登录）。

迁移准备情况评测（MRA）

使用 AWS CAF 深入了解组织的云就绪状态、确定优势和劣势以及制定行动计划以缩小已发现差距的过程。有关更多信息，请参阅迁移准备指南。MRA 是 AWS 迁移策略的第一阶段。

迁移策略

用于将工作负载迁移到的方法 AWS Cloud。有关更多信息，请参阅此词汇表中的 7 R 条目和动员组织以加快大规模迁移。

ML

参见机器学习。

现代化

将过时的（原有的或单体）应用程序及其基础设施转变为云中敏捷、弹性和高度可用的系统，以降低成本、提高效率和利用创新。有关更多信息，请参阅中的应用程序现代化策略。 AWS Cloud

现代化准备情况评估

一种评估方式，有助于确定组织应用程序的现代化准备情况；确定收益、风险和依赖关系；确定组织能够在多大程度上支持这些应用程序的未来状态。评估结果是目标架构的蓝图、详细说明现代化进程发展阶段和里程碑的路线图以及解决已发现差距的行动计划。有关更多信息，请参阅中的评估应用程序的现代化准备情况 AWS Cloud。

单体应用程序（单体式）

作为具有紧密耦合进程的单个服务运行的应用程序。单体应用程序有几个缺点。如果某个应用程序功能的需求激增，则必须扩展整个架构。随着代码库的增长，添加或改进单体应用程序的功能也会变得更加复杂。若要解决这些问题，可以使用微服务架构。有关更多信息，请参阅将单体分解为微服务。

MPA

参见迁移组合评估。

MQTT

请参阅消息队列遥测传输。

多分类器

一种帮助为多个类别生成预测（预测两个以上结果之一）的过程。例如，ML 模型可能会询问“这个产品是书、汽车还是手机？”或“此客户最感兴趣什么类别的产品？”

可变基础架构

一种用于更新和修改现有生产工作负载基础架构的模型。为了提高一致性、可靠性和可预测性，Well-Architect AWS ed Framework 建议使用不可变基础设施作为最佳实践。

O

OAC

请参阅源站访问控制。

OAI

参见源访问身份。

OCM

参见组织变更管理。

离线迁移

一种迁移方法，在这种方法中，源工作负载会在迁移过程中停止运行。这种方法会延长停机时间，通常用于小型非关键工作负载。

OI

参见运营集成。

OLA

参见运营层协议。

在线迁移

一种迁移方法，在这种方法中，源工作负载无需离线即可复制到目标系统。在迁移过程中，连接工作负载的应用程序可以继续运行。这种方法的停机时间为零或最短，通常用于关键生产工作负载。

OPC-UA

参见开放流程通信-统一架构。

开放流程通信-统一架构 (OPC-UA)

一种用于工业自动化的 machine-to-machine（M2M）通信协议。OPC-UA 提供了数据加密、身份验证和授权方案的互操作性标准。

运营级别协议（OLA）

一项协议，阐明了 IT 职能部门承诺相互交付的内容，以支持服务水平协议（SLA）。

运营准备情况审查 (ORR)

一份问题清单和相关的最佳实践，可帮助您理解、评估、预防或缩小事件和可能的故障的范围。有关更多信息，请参阅 Well-Architecte AWS d Frame work 中的运营准备情况评估 (ORR)。

操作技术 (OT)

与物理环境配合使用以控制工业运营、设备和基础设施的硬件和软件系统。在制造业中，OT 和信息技术 (IT) 系统的集成是工业 4.0 转型的重点。

运营整合（OI）

在云中实现运营现代化的过程，包括就绪计划、自动化和集成。有关更多信息，请参阅运营整合指南。

组织跟踪

由此创建的跟踪 AWS CloudTrail ，用于记录组织 AWS 账户 中所有人的所有事件 AWS Organizations。该跟踪是在每个 AWS 账户 中创建的，属于组织的一部分，并跟踪每个账户的活动。有关更多信息，请参阅 CloudTrail文档中的为组织创建跟踪。

组织变革管理（OCM）

一个从人员、文化和领导力角度管理重大、颠覆性业务转型的框架。OCM 通过加快变革采用、解决过渡问题以及推动文化和组织变革，帮助组织为新系统和战略做好准备和过渡。在 AWS 迁移策略中，该框架被称为人员加速，因为云采用项目需要变更的速度。有关更多信息，请参阅 OCM 指南。

来源访问控制（OAC）

在中 CloudFront，一个增强的选项，用于限制访问以保护您的亚马逊简单存储服务 (Amazon S3) 内容。OAC 全部支持所有 S3 存储桶 AWS 区域、使用 AWS KMS (SSE-KMS) 进行服务器端加密，以及对 S3 存储桶的动态PUT和DELETE请求。

来源访问身份（OAI）

在中 CloudFront，一个用于限制访问权限以保护您的 Amazon S3 内容的选项。当您使用 OAI 时，CloudFront 会创建一个 Amazon S3 可以对其进行身份验证的委托人。经过身份验证的委托人只能通过特定 CloudFront 分配访问 S3 存储桶中的内容。另请参阅 OAC，其中提供了更精细和增强的访问控制。

ORR

参见运营准备情况审查。

OT

参见运营技术。

出站（出口）VPC

在 AWS 多账户架构中，一种处理从应用程序内部启动的网络连接的 VPC。AWS 安全参考架构建议设置您的网络帐户，包括入站、出站和检查， VPCs 以保护您的应用程序与更广泛的互联网之间的双向接口。

P

权限边界

附加到 IAM 主体的 IAM 管理策略，用于设置用户或角色可以拥有的最大权限。有关更多信息，请参阅 IAM 文档中的权限边界。

个人身份信息 (PII)

直接查看其他相关数据或与之配对时可用于合理推断个人身份的信息。PII 的示例包括姓名、地址和联系信息。

PII

查看个人身份信息。

playbook

一套预定义的步骤，用于捕获与迁移相关的工作，例如在云中交付核心运营功能。playbook 可以采用脚本、自动化运行手册的形式，也可以是操作现代化环境所需的流程或步骤的摘要。

PLC

参见可编程逻辑控制器。

PLM

参见产品生命周期管理。

policy

一个对象，可以在中定义权限（参见基于身份的策略）、指定访问条件（参见基于资源的策略）或定义组织中所有账户的最大权限 AWS Organizations （参见服务控制策略）。

多语言持久性

根据数据访问模式和其他要求，独立选择微服务的数据存储技术。如果您的微服务采用相同的数据存储技术，它们可能会遇到实现难题或性能不佳。如果微服务使用最适合其需求的数据存储，则可以更轻松地实现微服务，并获得更好的性能和可扩展性。有关更多信息，请参阅在微服务中实现数据持久性。

组合评测

一个发现、分析和确定应用程序组合优先级以规划迁移的过程。有关更多信息，请参阅评估迁移准备情况。

谓词

返回`true`或的查询条件`false`，通常位于子`WHERE`句中。

谓词下推

一种数据库查询优化技术，可在传输前筛选查询中的数据。这减少了必须从关系数据库检索和处理的数据量，并提高了查询性能。

预防性控制

一种安全控制，旨在防止事件发生。这些控制是第一道防线，帮助防止未经授权的访问或对网络的意外更改。有关更多信息，请参阅在 AWS上实施安全控制中的预防性控制。

主体

中 AWS 可以执行操作和访问资源的实体。此实体通常是 IAM 角色的根用户或用户。 AWS 账户有关更多信息，请参阅 IAM 文档中角色术语和概念中的主体。

通过设计保护隐私

一种在整个开发过程中考虑隐私的系统工程方法。

私有托管区

一个容器，其中包含有关您希望 Amazon Route 53 如何响应针对一个或多个 VPCs域名及其子域名的 DNS 查询的信息。有关更多信息，请参阅 Route 53 文档中的私有托管区的使用。

主动控制

一种安全控制措施，旨在防止部署不合规的资源。这些控件会在资源配置之前对其进行扫描。如果资源与控件不兼容，则不会对其进行配置。有关更多信息，请参阅 AWS Control Tower 文档中的控制参考指南，并参见在上实施安全控制中的主动控制 AWS。

产品生命周期管理 (PLM)

在产品的整个生命周期中，从设计、开发和上市，到成长和成熟，再到衰落和移除，对产品进行数据和流程的管理。

生产环境

参见环境。

可编程逻辑控制器 (PLC)

在制造业中，一种高度可靠、适应性强的计算机，用于监控机器并实现制造过程自动化。

提示链接

使用一个 LLM 提示的输出作为下一个提示的输入，以生成更好的响应。该技术用于将复杂的任务分解为子任务，或者迭代地完善或扩展初步响应。它有助于提高模型响应的准确性和相关性，并允许获得更精细的个性化结果。

假名化

用占位符值替换数据集中个人标识符的过程。假名化可以帮助保护个人隐私。假名化数据仍被视为个人数据。

publish/subscribe (pub/sub)

一种支持微服务间异步通信的模式，以提高可扩展性和响应能力。例如，在基于微服务的 MES 中，微服务可以将事件消息发布到其他微服务可以订阅的频道。系统可以在不更改发布服务的情况下添加新的微服务。

Q

查询计划

一系列步骤，例如指令，用于访问 SQL 关系数据库系统中的数据。

查询计划回归

当数据库服务优化程序选择的最佳计划不如数据库环境发生特定变化之前时。这可能是由统计数据、约束、环境设置、查询参数绑定更改和数据库引擎更新造成的。

R

RACI 矩阵

参见 “负责任、负责、咨询、知情”（RACI）。

RAG

请参见检索增强生成。

勒索软件

一种恶意软件，旨在阻止对计算机系统或数据的访问，直到付款为止。

RASCI 矩阵

参见 “负责任、负责、咨询、知情”（RACI）。

RCAC

请参阅行和列访问控制。

只读副本

用于只读目的的数据库副本。您可以将查询路由到只读副本，以减轻主数据库的负载。

重新架构师

见 7 R。

恢复点目标 (RPO)

自上一个数据恢复点以来可接受的最长时间。这决定了从上一个恢复点到服务中断之间可接受的数据丢失情况。

恢复时间目标 (RTO)

服务中断和服务恢复之间可接受的最大延迟。

重构

见 7 R。

区域

地理区域内的 AWS 资源集合。每一个 AWS 区域 都相互隔离，彼此独立，以提供容错、稳定性和弹性。有关更多信息，请参阅指定 AWS 区域 您的账户可以使用的账户。

回归

一种预测数值的 ML 技术。例如，要解决“这套房子的售价是多少？”的问题 ML 模型可以使用线性回归模型，根据房屋的已知事实（如建筑面积）来预测房屋的销售价格。

重新托管

见 7 R。

版本

在部署过程中，推动生产环境变更的行为。

搬迁

见 7 R。

更换平台

见 7 R。

回购

见 7 R。

故障恢复能力

应用程序抵御中断或从中断中恢复的能力。在中规划弹性时，高可用性和灾难恢复是常见的考虑因素。 AWS Cloud有关更多信息，请参阅AWS Cloud 弹性。

基于资源的策略

一种附加到资源的策略，例如 AmazonS3 存储桶、端点或加密密钥。此类策略指定了允许哪些主体访问、支持的操作以及必须满足的任何其他条件。

责任、问责、咨询和知情（RACI）矩阵

定义参与迁移活动和云运营的所有各方的角色和责任的矩阵。矩阵名称源自矩阵中定义的责任类型：负责 (R)、问责 (A)、咨询 (C) 和知情 (I)。支持 (S) 类型是可选的。如果包括支持，则该矩阵称为 RASCI 矩阵，如果将其排除在外，则称为 RACI 矩阵。

响应性控制

一种安全控制，旨在推动对不良事件或偏离安全基线的情况进行修复。有关更多信息，请参阅在AWS上实施安全控制中的响应性控制。

保留

见 7 R。

退休

见 7 R。

检索增强生成（RAG）

一种生成式人工智能技术，其中法学硕士在生成响应之前引用其训练数据源之外的权威数据源。例如，RAG 模型可以对组织的知识库或自定义数据执行语义搜索。有关更多信息，请参阅什么是 RAG。

轮换

定期更新密钥以使攻击者更难访问凭据的过程。

行列访问控制 (RCAC)

使用已定义访问规则的基本、灵活的 SQL 表达式。RCAC 由行权限和列掩码组成。

RPO

参见恢复点目标。

RTO

参见恢复时间目标。

运行手册

执行特定任务所需的一套手动或自动程序。它们通常是为了简化重复性操作或高错误率的程序而设计的。

S

SAML 2.0

许多身份提供商 (IdPs) 使用的开放标准。此功能支持联合单点登录 (SSO)，因此用户无需在 IAM 中为组织中的所有人创建用户即可登录 AWS Management Console 或调用 AWS API 操作。有关基于 SAML 2.0 的联合身份验证的更多信息，请参阅 IAM 文档中的关于基于 SAML 2.0 的联合身份验证。

SCADA

参见监督控制和数据采集。

SCP

参见服务控制政策。

secret

在中 AWS Secrets Manager，您以加密形式存储的机密或受限信息，例如密码或用户凭证。它由密钥值及其元数据组成。密钥值可以是二进制、单个字符串或多个字符串。有关更多信息，请参阅 Secret s Manager 密钥中有什么？ 在 Secrets Manager 文档中。

安全性源于设计

一种在整个开发过程中考虑安全性的系统工程方法。

安全控制

一种技术或管理防护机制，可防止、检测或降低威胁行为体利用安全漏洞的能力。安全控制主要有四种类型：预防性、侦测、响应式和主动式。

安全加固

缩小攻击面，使其更能抵御攻击的过程。这可能包括删除不再需要的资源、实施授予最低权限的最佳安全实践或停用配置文件中不必要的功能等操作。

安全信息和事件管理（SIEM）系统

结合了安全信息管理（SIM）和安全事件管理（SEM）系统的工具和服务。SIEM 系统会收集、监控和分析来自服务器、网络、设备和其他来源的数据，以检测威胁和安全漏洞，并生成警报。

安全响应自动化

一种预定义和编程的操作，旨在自动响应或修复安全事件。这些自动化可作为侦探或响应式安全控制措施，帮助您实施 AWS 安全最佳实践。自动响应操作的示例包括修改 VPC 安全组、修补 Amazon EC2 实例或轮换证书。

服务器端加密

在目的地对数据进行加密，由接收方 AWS 服务 进行加密。

服务控制策略（SCP）

一种策略，用于集中控制组织中所有账户的权限 AWS Organizations。 SCPs 定义防护措施或限制管理员可以委托给用户或角色的操作。您可以使用 SCPs 允许列表或拒绝列表来指定允许或禁止哪些服务或操作。有关更多信息，请参阅 AWS Organizations 文档中的服务控制策略。

服务端点

的入口点的 URL AWS 服务。您可以使用端点，通过编程方式连接到目标服务。有关更多信息，请参阅 AWS 一般参考 中的 AWS 服务 端点。

服务水平协议（SLA）

一份协议，阐明了 IT 团队承诺向客户交付的内容，比如服务正常运行时间和性能。

服务级别指示器 (SLI)

对服务性能方面的衡量，例如其错误率、可用性或吞吐量。

服务级别目标 (SLO)

代表服务运行状况的目标指标，由服务级别指标衡量。

责任共担模式

描述您在云安全与合规方面共同承担 AWS 的责任的模型。 AWS 负责云的安全，而您则负责云中的安全。有关更多信息，请参阅责任共担模式。

SIEM

参见安全信息和事件管理系统。

单点故障 (SPOF)

应用程序的单个关键组件出现故障，可能会中断系统。

SLA

参见服务级别协议。

SLI

参见服务级别指标。

SLO

参见服务级别目标。

split-and-seed 模型

一种扩展和加速现代化项目的模式。随着新功能和产品发布的定义，核心团队会拆分以创建新的产品团队。这有助于扩展组织的能力和服务，提高开发人员的工作效率，支持快速创新。有关更多信息，请参阅中的分阶段实现应用程序现代化的方法。 AWS Cloud

恶作剧

参见单点故障。

星型架构

一种数据库组织结构，它使用一个大型事实表来存储交易数据或测量数据，并使用一个或多个较小的维度表来存储数据属性。此结构专为在数据仓库中使用或用于商业智能目的而设计。

strangler fig 模式

一种通过逐步重写和替换系统功能直至可以停用原有的系统来实现单体系统现代化的方法。这种模式用无花果藤作为类比，这种藤蔓成长为一棵树，最终战胜并取代了宿主。该模式是由 Martin Fowler 提出的，作为重写单体系统时管理风险的一种方法。有关如何应用此模式的示例，请参阅使用容器和 Amazon API Gateway 逐步将原有的 Microsoft ASP.NET（ASMX）Web 服务现代化。

子网

您的 VPC 内的一个 IP 地址范围。子网必须位于单个可用区中。

监控和数据采集 (SCADA)

在制造业中，一种使用硬件和软件来监控有形资产和生产操作的系统。

对称加密

一种加密算法，它使用相同的密钥来加密和解密数据。

综合测试

以模拟用户交互的方式测试系统，以检测潜在问题或监控性能。你可以使用 Amazon S CloudWatch ynthetics 来创建这些测试。

系统提示符

一种向法学硕士提供上下文、说明或指导方针以指导其行为的技术。系统提示有助于设置上下文并制定与用户交互的规则。

T

tags

键值对，充当用于组织资源的元数据。 AWS 标签可帮助您管理、识别、组织、搜索和筛选资源。有关更多信息，请参阅标记您的 AWS 资源。

目标变量

您在监督式 ML 中尝试预测的值。这也被称为结果变量。例如，在制造环境中，目标变量可能是产品缺陷。

任务列表

一种通过运行手册用于跟踪进度的工具。任务列表包含运行手册的概述和要完成的常规任务列表。对于每项常规任务，它包括预计所需时间、所有者和进度。

测试环境

参见环境。

训练

为您的 ML 模型提供学习数据。训练数据必须包含正确答案。学习算法在训练数据中查找将输入数据属性映射到目标（您希望预测的答案）的模式。然后输出捕获这些模式的 ML 模型。然后，您可以使用 ML 模型对不知道目标的新数据进行预测。

中转网关

一个网络传输中心，可用于将您的网络 VPCs 和本地网络互连。有关更多信息，请参阅 AWS Transit Gateway 文档中的什么是公交网关。

基于中继的工作流程

一种方法，开发人员在功能分支中本地构建和测试功能，然后将这些更改合并到主分支中。然后，按顺序将主分支构建到开发、预生产和生产环境。

可信访问权限

向您指定的服务授予权限，该服务可代表您在其账户中执行任务。 AWS Organizations 当需要服务相关的角色时，受信任的服务会在每个账户中创建一个角色，为您执行管理任务。有关更多信息，请参阅 AWS Organizations 文档中的AWS Organizations 与其他 AWS 服务一起使用。

优化

更改训练过程的各个方面，以提高 ML 模型的准确性。例如，您可以通过生成标签集、添加标签，并在不同的设置下多次重复这些步骤来优化模型，从而训练 ML 模型。

双披萨团队

一个小 DevOps 团队，你可以用两个披萨来喂食。双披萨团队的规模可确保在软件开发过程中充分协作。

U

不确定性

这一概念指的是不精确、不完整或未知的信息，这些信息可能会破坏预测式 ML 模型的可靠性。不确定性有两种类型：认知不确定性是由有限的、不完整的数据造成的，而偶然不确定性是由数据中固有的噪声和随机性导致的。有关更多信息，请参阅量化深度学习系统中的不确定性指南。

无差别任务

也称为繁重工作，即创建和运行应用程序所必需的工作，但不能为最终用户提供直接价值或竞争优势。无差别任务的示例包括采购、维护和容量规划。

上层环境

参见环境。

V

vacuum 操作

一种数据库维护操作，包括在增量更新后进行清理，以回收存储空间并提高性能。

版本控制

跟踪更改的过程和工具，例如存储库中源代码的更改。

VPC 对等连接

两者之间的连接 VPCs ，允许您使用私有 IP 地址路由流量。有关更多信息，请参阅 Amazon VPC 文档中的什么是 VPC 对等连接。

漏洞

损害系统安全的软件缺陷或硬件缺陷。

W

热缓存

一种包含经常访问的当前相关数据的缓冲区缓存。数据库实例可以从缓冲区缓存读取，这比从主内存或磁盘读取要快。

暖数据

不常访问的数据。查询此类数据时，通常可以接受中速查询。

窗口函数

一个 SQL 函数，用于对一组以某种方式与当前记录相关的行进行计算。窗口函数对于处理任务很有用，例如计算移动平均线或根据当前行的相对位置访问行的值。

工作负载

一系列资源和代码，它们可以提供商业价值，如面向客户的应用程序或后端过程。

工作流

迁移项目中负责一组特定任务的职能小组。每个工作流都是独立的，但支持项目中的其他工作流。例如，组合工作流负责确定应用程序的优先级、波次规划和收集迁移元数据。组合工作流将这些资产交付给迁移工作流，然后迁移服务器和应用程序。

蠕虫

参见一次写入，多读。

WQF

参见AWS 工作负载资格框架。

一次写入，多次读取 (WORM)

一种存储模型，它可以一次写入数据并防止数据被删除或修改。授权用户可以根据需要多次读取数据，但他们无法对其进行更改。这种数据存储基础架构被认为是不可变的。

Z

零日漏洞利用

一种利用未修补漏洞的攻击，通常是恶意软件。

零日漏洞

生产系统中不可避免的缺陷或漏洞。威胁主体可能利用这种类型的漏洞攻击系统。开发人员经常因攻击而意识到该漏洞。

零镜头提示

向法学硕士提供执行任务的说明，但没有示例（镜头）可以帮助指导任务。法学硕士必须使用其预先训练的知识来处理任务。零镜头提示的有效性取决于任务的复杂性和提示的质量。另请参阅 few-shot 提示。

僵尸应用程序

平均 CPU 和内存使用率低于 5% 的应用程序。在迁移项目中，通常会停用这些应用程序。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。