

云硬盘

操作指南

产品文档



腾讯云

【版权声明】

©2013-2018 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

文档目录

操作指南

- 创建云硬盘

- 将弹性云硬盘连接到 CVM 实例

- 启动新实例时自动挂载数据盘

 - 云硬盘分区、格式化及创建文件系统

 - Windows系统分区、格式化、联机及创建文件系统

 - Linux 系统分区、格式化、挂载及创建文件系统

- 扩容云硬盘

 - 扩容文件系统

 - 扩容Windows文件系统

 - 扩容Linux文件系统

- 续费云硬盘

- 卸载云硬盘

- 删除云硬盘

操作指南

创建云硬盘

最近更新时间：2018-09-20 15:28:33

您可以创建云硬盘并将它连接到同一个可用区内的任何 CVM 实例。通过块储存设备映射，云硬盘被云服务器实例识别并使用。同时，用户也可基于先前创建的快照来创建新的云硬盘。有关更多信息，请参阅 [快照创建云硬盘](#)，云硬盘在创建后不需要预热即可达到其最高性能。

您可以根据业务使用需要选择创建不同类型的云硬盘，有关云硬盘类型的更多内容请参考 [云硬盘的分类](#)。

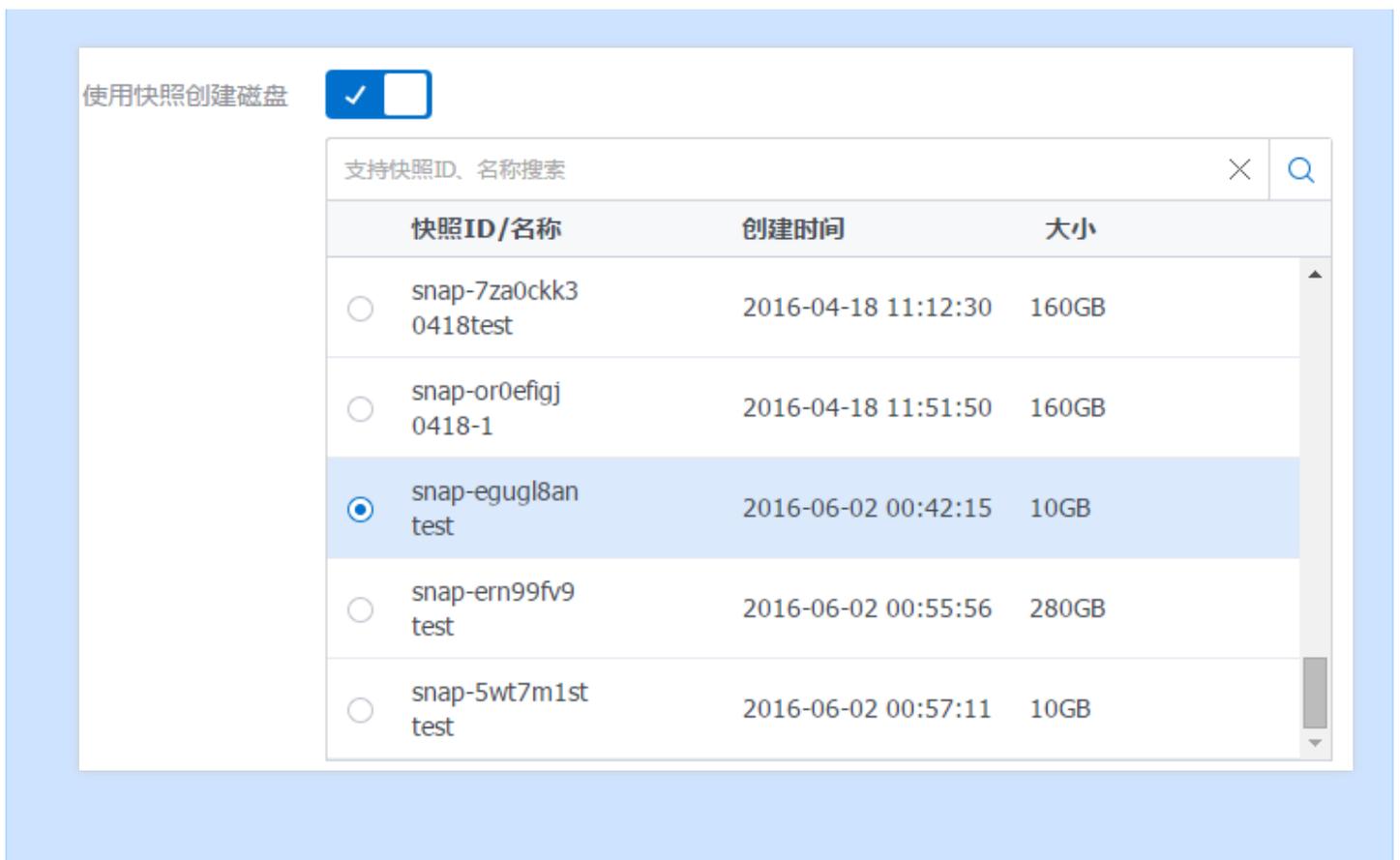
使用控制台创建云硬盘

+ 新建

- 1) 登录 [云硬盘控制台](#)，单击  即可开始购买。
- 2) 在弹出框中选择地域/可用区、计费模式、容量、购买数量和购买时长等信息，单击【确定】按钮。
- 3) 进入支付页面完成支付或下单后，您可在 [云硬盘列表页](#) 查看已购买的云硬盘，您刚刚购买的弹性云盘默认未命名，并显示为**待挂载**状态。

注：

- 弹性云盘可以在同一可用区内的云服务器之间自由挂载和卸载；
- 如果您需要在新创建的磁盘中保留数据盘快照数据，那么您可以在该页面开启【使用快照创建磁盘】，并选择您要使用的快照。选择快照后，磁盘容量默认等于快照大小，您可以调整容量大于默认值。



使用快照创建云硬盘

如果您需要在新创建的磁盘中保留数据盘快照数据，则可以选择这种方式。

1) 登录 [快照控制台](#)，在您要使用的快照后，单击【新建云硬盘】即可开始购买。

<input type="checkbox"/>	ID/名称	状态	大小	快照类型	关联磁盘	创建时间	操作
<input type="checkbox"/>	snap-5wt7m1st test	正常	10G	数据盘	disk-d8tq6q0y	2016-06-02 00:57:11	回滚 删除 改名 新建云硬盘

2) 在弹出框中选择地域/可用区、计费模式、容量、购买数量和购买时长等信息，单击【确定】按钮。

注意：新购买的磁盘容量默认等于快照大小，您可以调整容量大于默认值

3) 进入支付页面完成支付或下单后，您可在 [云硬盘列表页](#) 查看已购买的云硬盘，您刚刚购买的弹性云硬盘默认名称为From snap-xxxxxxx，并显示为待挂载状态。

使用 API 创建云硬盘

请参考 [CreateDisks](#) 接口。

将弹性云硬盘连接到 CVM 实例

最近更新时间：2018-09-20 15:28:52

非弹性云硬盘(即生命周期跟随 CVM 实例的云硬盘)在创建时将自动连接至创建的 CVM 实例上，且不可以更改；同时用户可以手动将弹性云盘挂载到同一可用区中的任意实例上，可以在挂载时确定每个实例还能挂载多少云硬盘。有关挂载数量的更多信息，请参阅 [使用约束](#)。有关弹性云盘和非弹性云硬盘的更多内容，请参阅 [云硬盘的分类](#)。

使用控制台将云硬盘连接到实例

目前支持对作为**数据盘**的普通弹性云硬盘云盘进行挂载，不可挂载系统盘。

- 1) 登录 [腾讯云控制台](#)。
- 2) 进入【云服务器】-【云硬盘】选项卡。
- 3) 在云硬盘列表页，单击状态为**待挂载**、**支持挂载/卸载**的云硬盘后的【更多】-【挂载】按钮进行单盘挂载；或在云硬盘列表页，勾选状态为**待挂载**、**支持挂载/卸载**的云硬盘，单击顶部【挂载】按钮进行批量挂载。
- 4) 在弹出框中选择需要挂载到的云服务器，单击【确定】按钮，等待挂载完毕即可登录云服务器查看云硬盘挂载状况。

云硬盘在挂载完后并不能马上使用，需要进行分区、格式化等一系列操作。具体操作方式请见：[Windows 系统分区、格式化及创建文件系统](#)、[Linux 系统分区、格式化、挂载及创建文件系统](#)。

使用 API 将云硬盘连接到实例

请参考 [AttachDisks 接口](#)。

部分已创建的云服务器实例无法识别云硬盘的解决方案

目前提供的所有镜像已经支持弹性云盘的连接/解挂操作。**请注意，拔盘（卸载）前请先执行umount（Linux）或脱机（Windows）操作，否则可能出现再次挂载时无法识别的问题。**

但若您在此时间之前购买了下面类型的云服务器并计划向云服务器中添加弹性云盘时：

云服务器操作系统类型	版本
CentOS	5.11 64位
	5.11 32位

	5.8 64位
	5.8 32位
Debian	6.0.3 32位
Ubuntu	10.04 64位
	10.04 32位
OpenSuse	12.3 64位
	12.3 32位

建议您在购买弹性云盘之前在实例中执行如下命令添加驱动来获得热插拔功能：

```
modprobe acpiphp
```

另外，当您在关机或者重新启动该云服务器后，仍然需要再次加载 `acpiphp` 驱动模块，建议您将 `acpiphp` 模块设置成开机自动加载，各个系列的设置方法如下：

CentOS 5系列

执行以下命令创建文件：

```
vi /etc/sysconfig/modules/acpiphp.modules
```

并在文件中添加如下内容：

```
#!/bin/bash  
modprobe acpiphp >& /dev/null
```

执行以下命令添加可执行权限，设置完成后此脚本即可开机加载：

```
chmod a+x /etc/sysconfig/modules/acpiphp.modules
```

Debian 6系列、Ubuntu 10.04系列

执行以下命令修改文件：

```
vi /etc/modules
```

并写入以下内容：

```
acpihp
```

OpenSUSE 12.3系列

执行以下命令修改文件：

```
vi /etc/sysconfig/kernel
```

并写入以下内容：

```
MODULES_LOADED_ON_BOOT="acpihp"
```

启动新实例时自动挂载数据盘

最近更新时间：2017-11-24 15:53:46

使用自定义镜像及数据盘快照启动新实例时自动挂载数据盘 (linux)

在启动新的云服务器实例时，如果用户指定 **自定义镜像** 及 **数据盘快照**，腾讯云云硬盘可以支持启动云服务器实例后自动挂载（即不需要进行一系列的添加、分区、格式化等操作可直接读写数据盘）。用户需要在制作自定义镜像和数据盘快照前在原实例上进行一些操作，下文将详细描述。

在 Linux 系统下如果用户希望指定数据盘快照生产出来的云硬盘能够自动挂载至新的云服务器实例，指定的自定义镜像和数据盘快照必须满足以下要求：

- 数据盘在制作快照前 **必须** 已经格式化过，也即在原云服务器上已经 mount 成功。
- 系统盘在制作自定义镜像前，需要在 `/etc/rc.local` 文件中添加以下命令，将数据盘挂载点写入文件中：

```
mkdir -p <mount-point>
mount <device-id> <mount-point>
```

其中：`<mount-point>` 请填入文件系统的挂载点如 `/mydata`，`<device-id>` 请填入用户的实际文件分区位置，如 `/dev/vdb`(无分区有文件系统的设备名) 和 `/dev/vdb1`(有分区有文件系统)。

只有同时满足以上两个条件才能保证新启动的 Linux 云服务器实例数据盘可以被自动识别和挂载。

使用自定义镜像及数据盘快照启动新实例时自动挂载数据盘 (windows)

在启动新的云服务器实例时，如果用户指定 **自定义镜像** 及 **数据盘快照**，腾讯云云硬盘可以支持启动云服务器实例后自动挂载（即不需要进行一系列的添加、分区、格式化等操作可直接读写数据盘）。用户需要在制作自定义镜像和数据盘快照前在原实例上进行一些操作，下文将详细描述。

在 Windows 系统下如果用户希望指定数据盘快照生产出来的云硬盘能够自动挂载至新的云服务器实例，指定的自定义镜像和数据盘快照必须满足以下要求：

- 自定义镜像中的 SAN 策略为：`onlineAll`。腾讯云目前提供的 Windows 公有镜像已默认进行相关设置，但仍建议用户在制作自定义镜像前检查下此配置，检查方法如下：

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> diskpart

Microsoft DiskPart 版本 6.1.7601
Copyright (C) 1999-2008 Microsoft Corporation.
在计算机上: 10_135_6_116

DISKPART> san

SAN 策略: 使共享磁盘脱机

DISKPART> san policy=onlineall

DiskPart 已成功更改用于当前操作系统的 SAN 策略。

DISKPART> san

SAN 策略: 全部联机
```

- 数据盘在制作快照前必须已经被格式化为 `ntfs` 或 `fat32` 格式。

只有同时满足以上两个条件才能保证新启动的 Windows 云服务器实例数据盘可以被自动识别和联机。

云硬盘分区、格式化及创建文件系统

Windows系统分区、格式化、联机及创建文件系统

最近更新时间：2018-07-04 10:17:52

实例可以识别连接的云硬盘并将其视为普通硬盘。用户可以使用任何文件系统将云块存储设备格式化、分区及创建文件系统。此后任何写入文件系统的数据均写入云硬盘中，并且对使用该设备的应用程序是透明的。数据盘默认为脱机状态，未做分区和格式化时无法使用。本教程将引导您进行 Windows 系统分区以及格式化。

虽然不同的Windows版本（win2012,win2008,win2003等）在进入“磁盘管理”界面的路径不同，但进入磁盘管理界面后对于磁盘分区格式化的操作基本一致。

本文将从 Windows 2012，Windows 2008 两种系统来引导用户进行数据盘挂载、分区和格式化。

前提条件

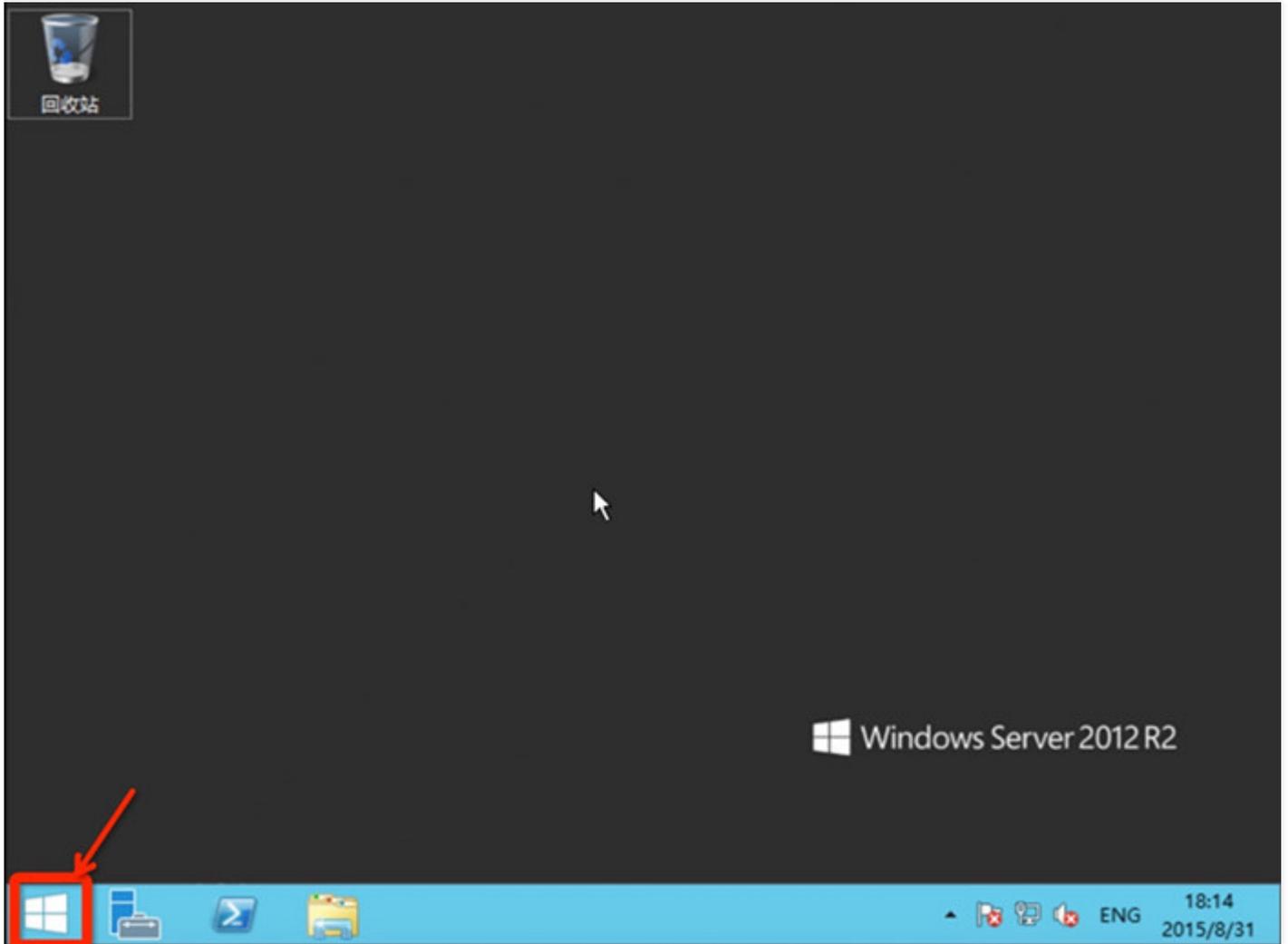
- 请确保您已进行 [将云硬盘连接到 CVM 实例](#) 及 [登录 Windows 实例](#) 操作。
- **格式化后，数据盘中的数据将被全部清空。请在格式化之前，确保数据盘中没有数据或对重要数据已进行备份。为避免服务发生异常，格式化前请确保云服务器已停止对外服务。**
- 当用户购买了多块云硬盘时，建议您对存放重要数据的弹性云盘设置自定义名称，并设置自动续费，防止因为没有及时续费导致弹性云盘到期对您的业务产生影响。
- 用户可以在[云硬盘控制台](#)中根据自定义名称或者关联的云服务器内网 IP 快速查找云硬盘。

Windows 2012 磁盘联机、分区和格式化

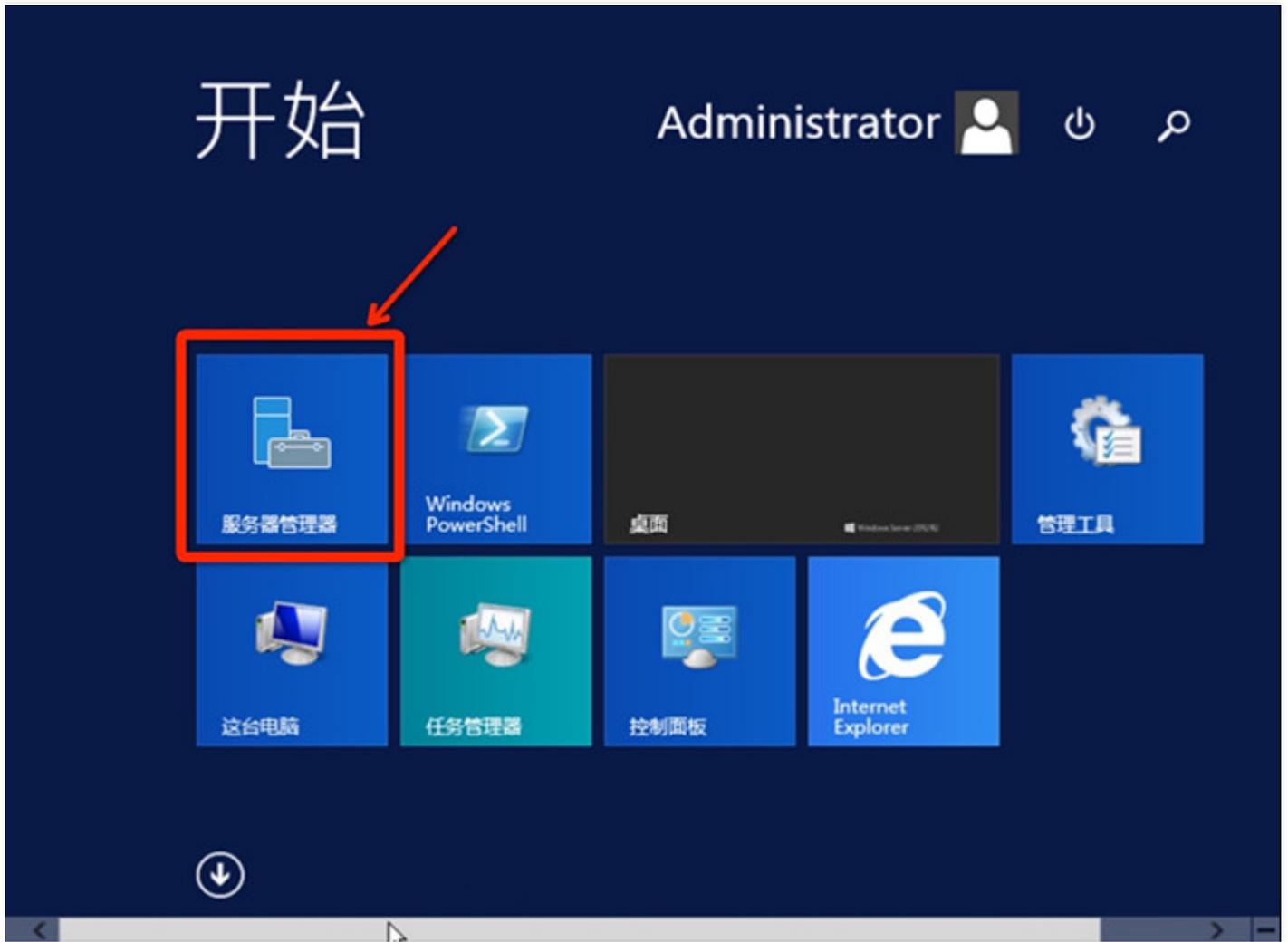
联机磁盘

Win 2012 进入磁盘管理的路径为：【开始】 - 【服务器管理】 - 【工具】 - 【计算机管理】 - 【磁盘管理】。

单击【开始】按钮：



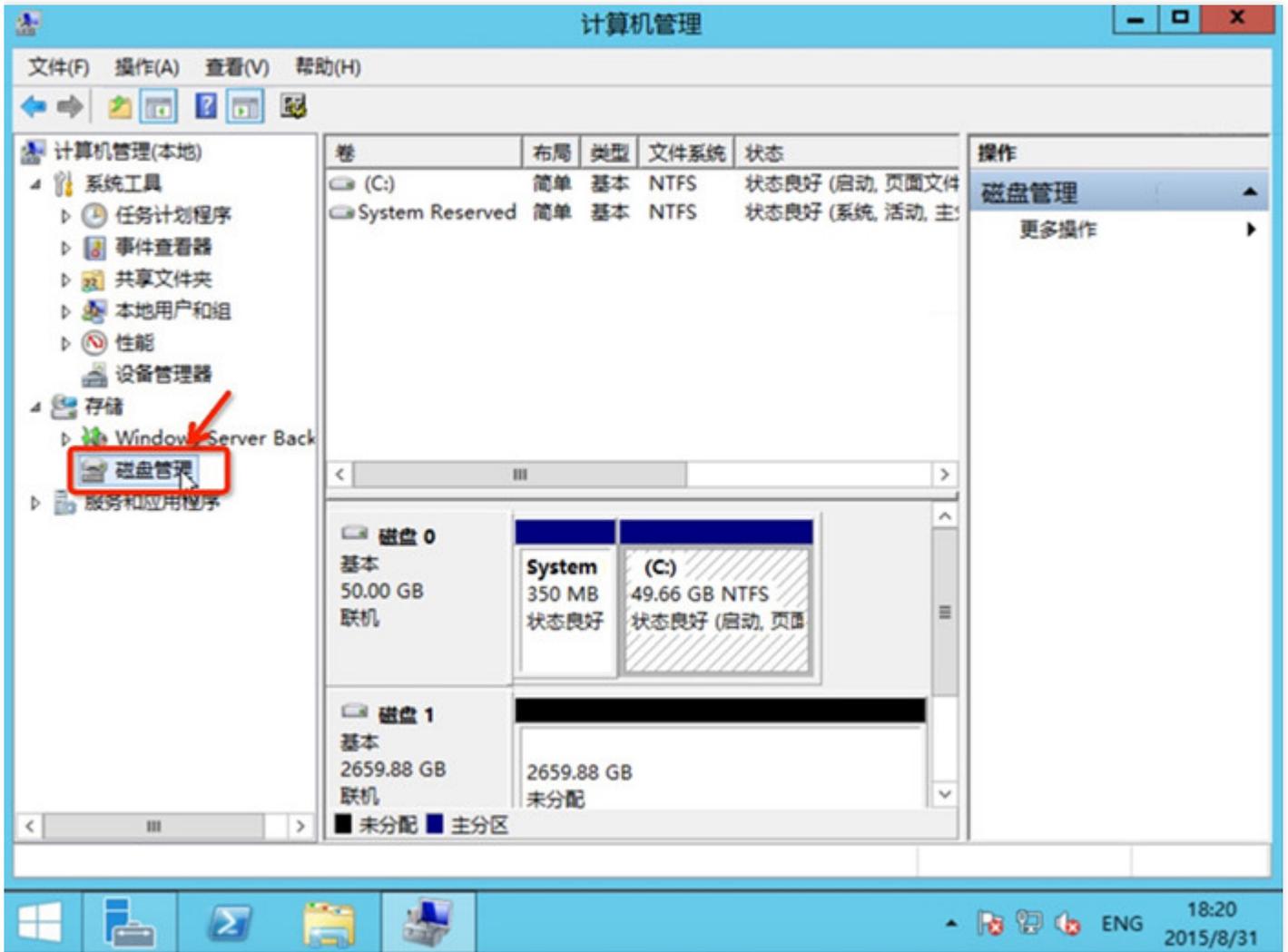
单击【服务器管理】：



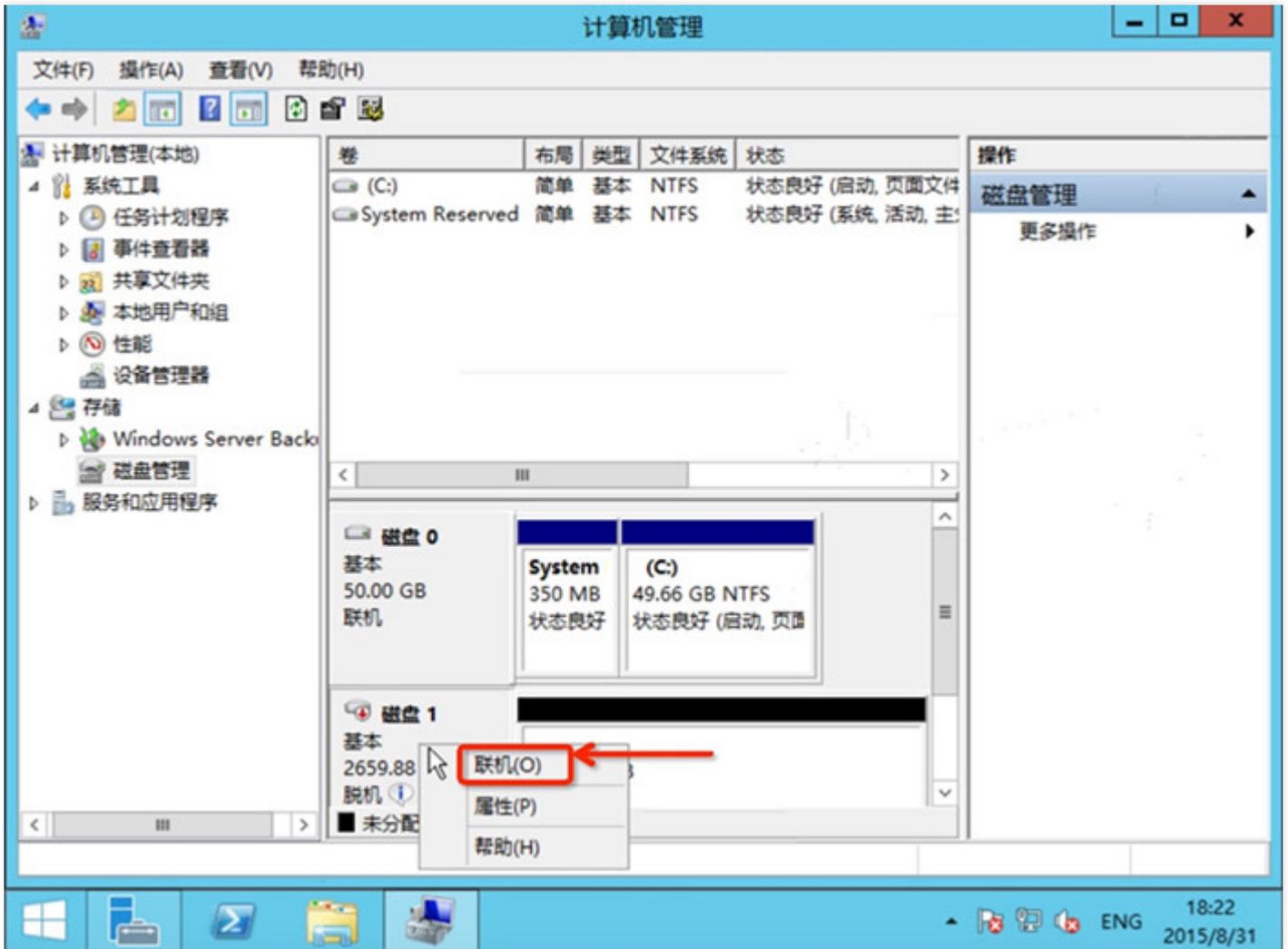
单击【工具】-【计算机管理】：



单击【磁盘管理】：



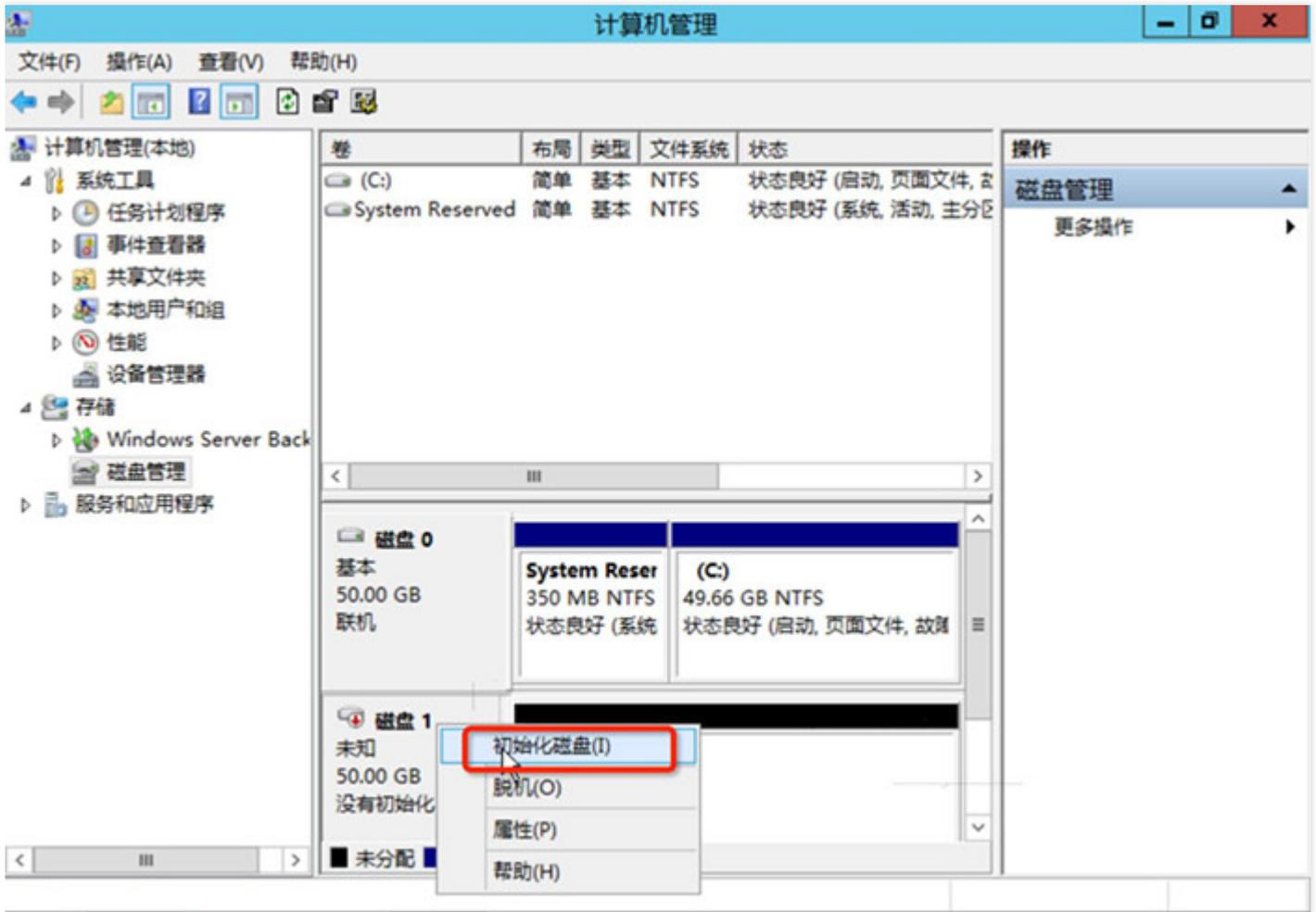
如下图所示：在磁盘1上右键单击，选择【联机】：



若磁盘上已经有数据（即不为空白磁盘情况下），用户可忽略下面的操作。对一个非空磁盘进行重新格式化或分区将清除所有原有数据

(可选) 格式化磁盘

右键单击，选择【初始化磁盘】：



根据分区方式的不同，选择【GPT】或【MBR】，单击【确定】按钮：

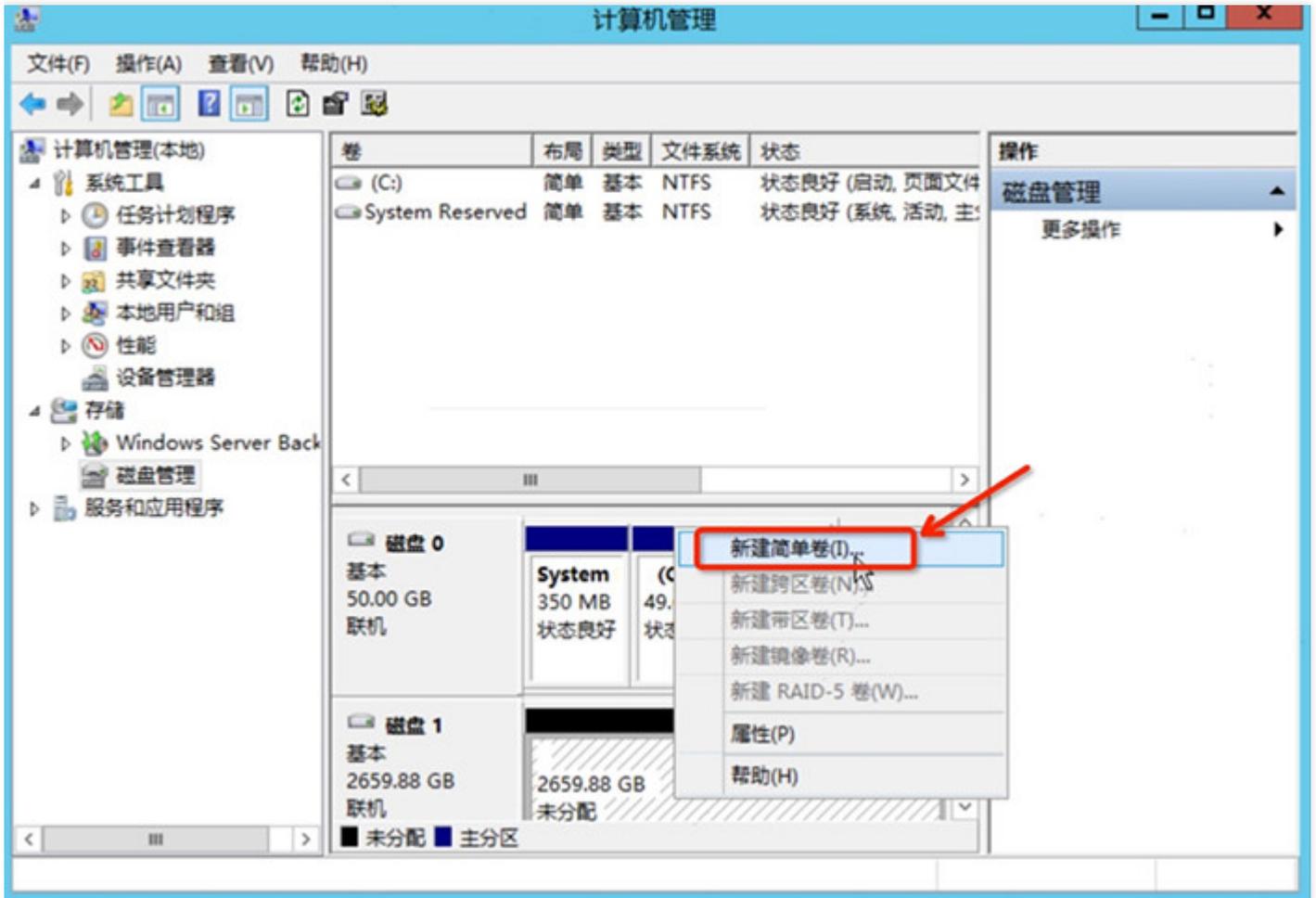


注意：

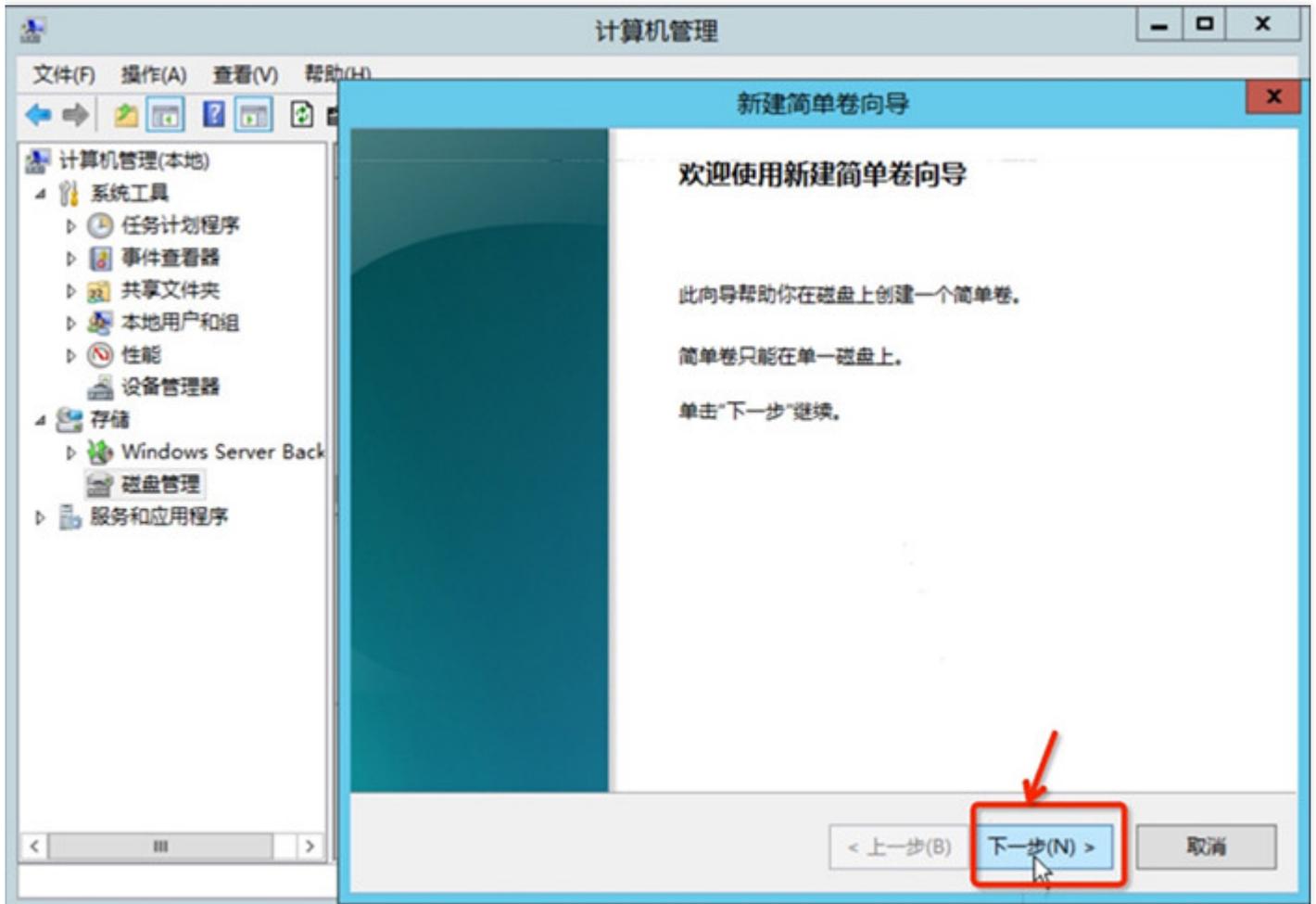
磁盘大于 2TB 时仅支持 GPT 分区形式。若您不确定磁盘后续扩容是否会超过该值，则建议您选择 GPT 分区；若您确定磁盘大小不会超过该值，则建议您选择 MBR 分区以获得更好的兼容性。

(可选) 对磁盘分区

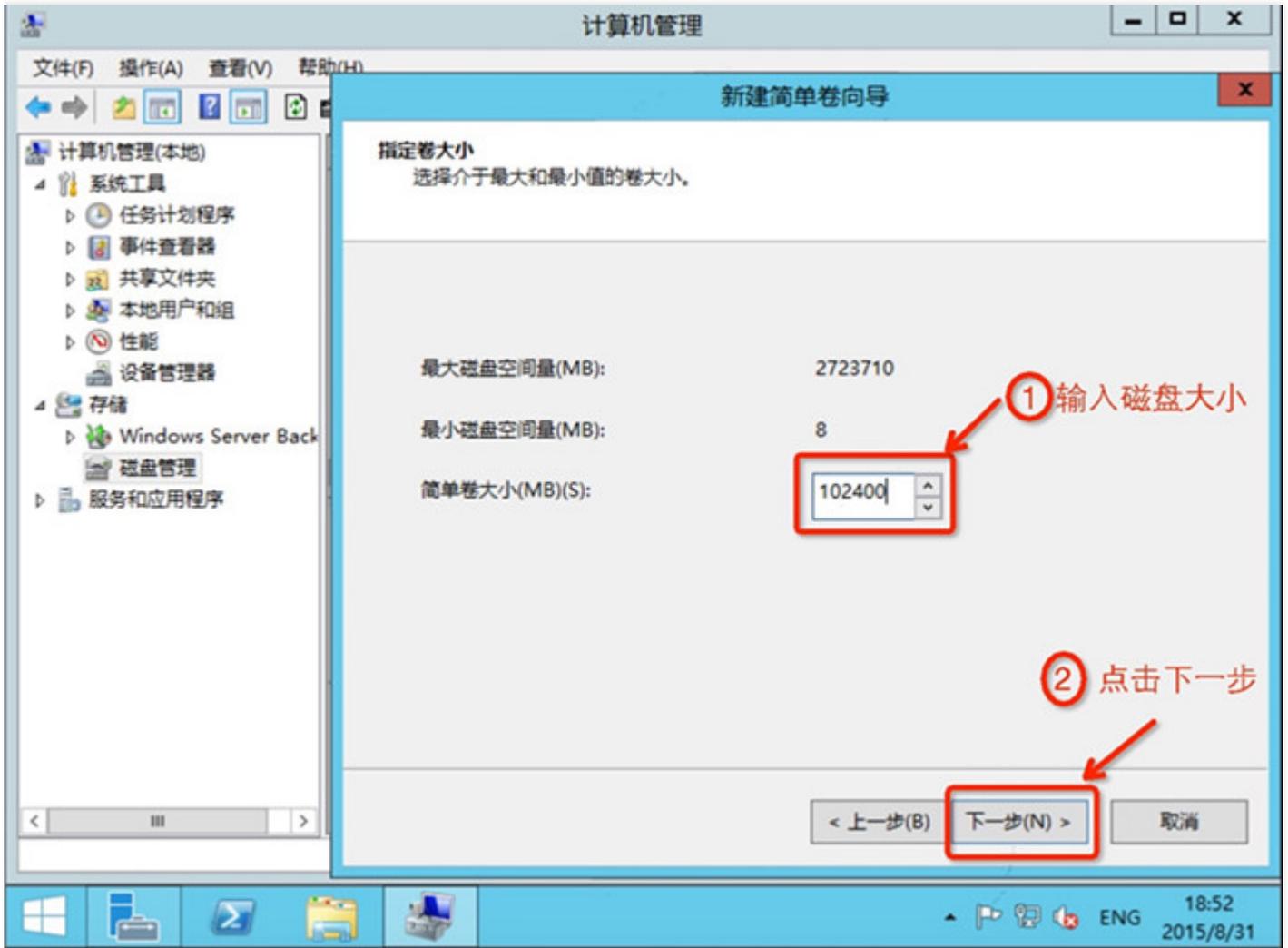
在未分配的空间处右击，选择【新建简单卷】：



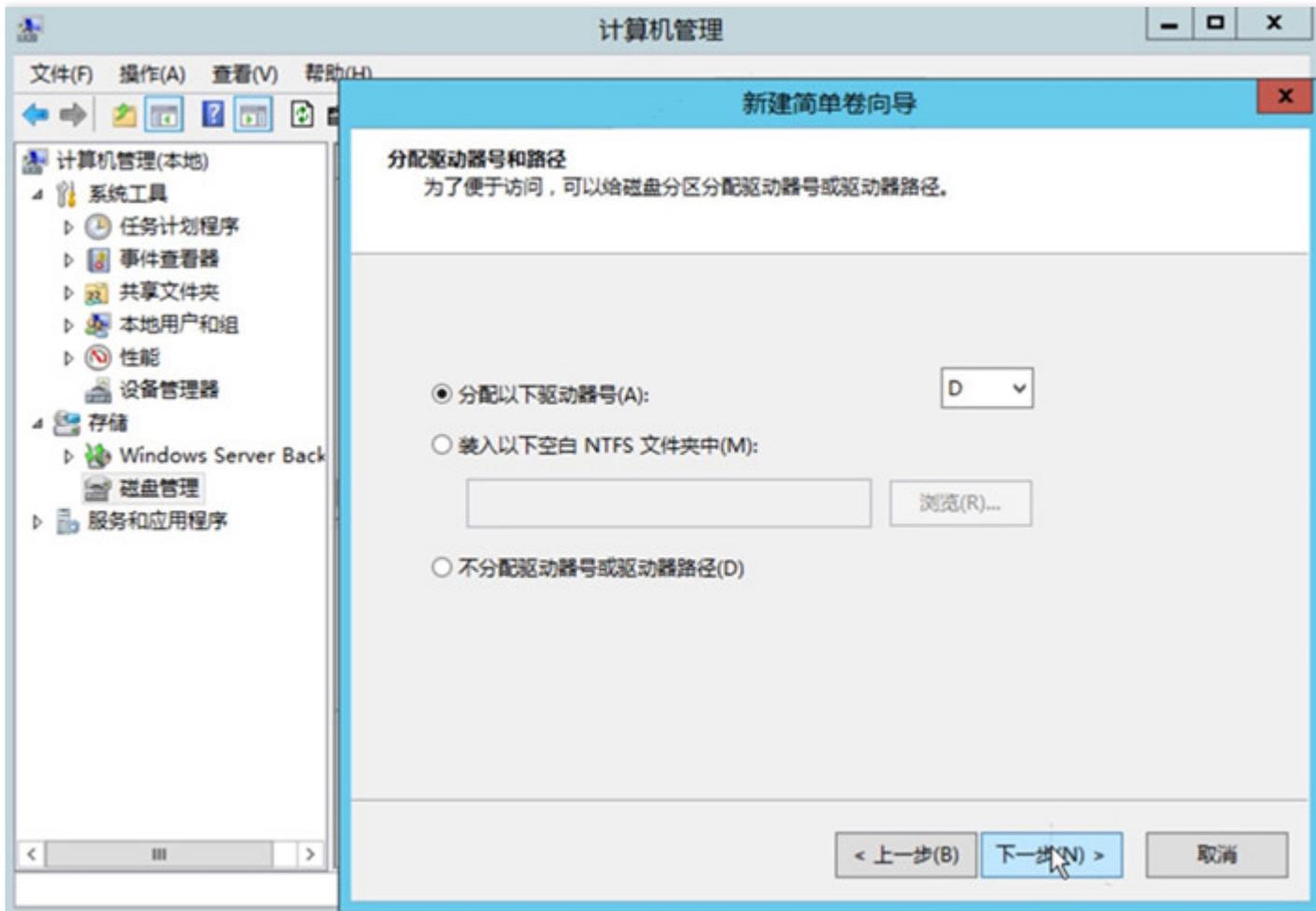
在弹出的“新建简单卷向导”窗口中，单击【下一步】：



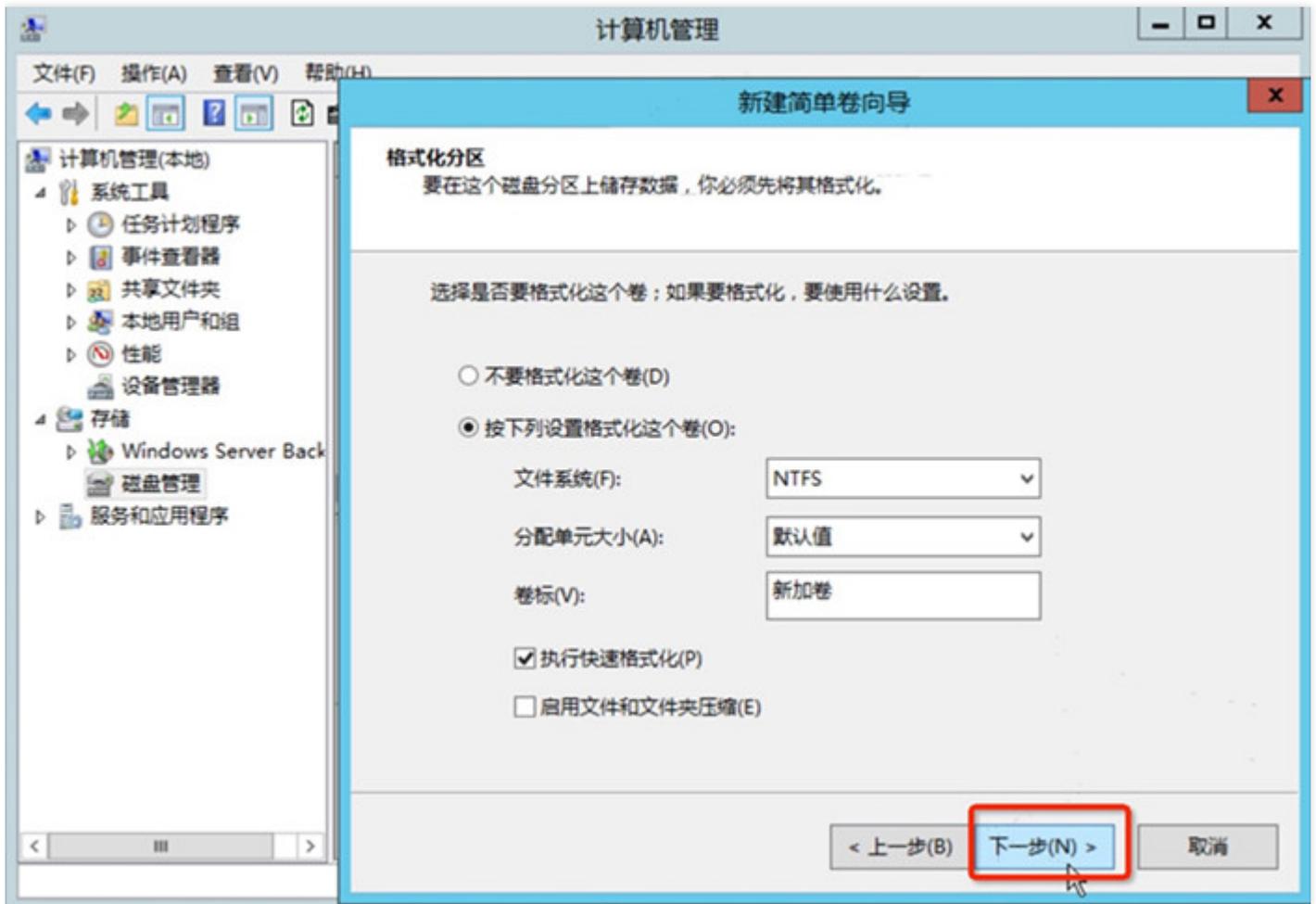
输入分区所需磁盘大小，单击【下一步】：



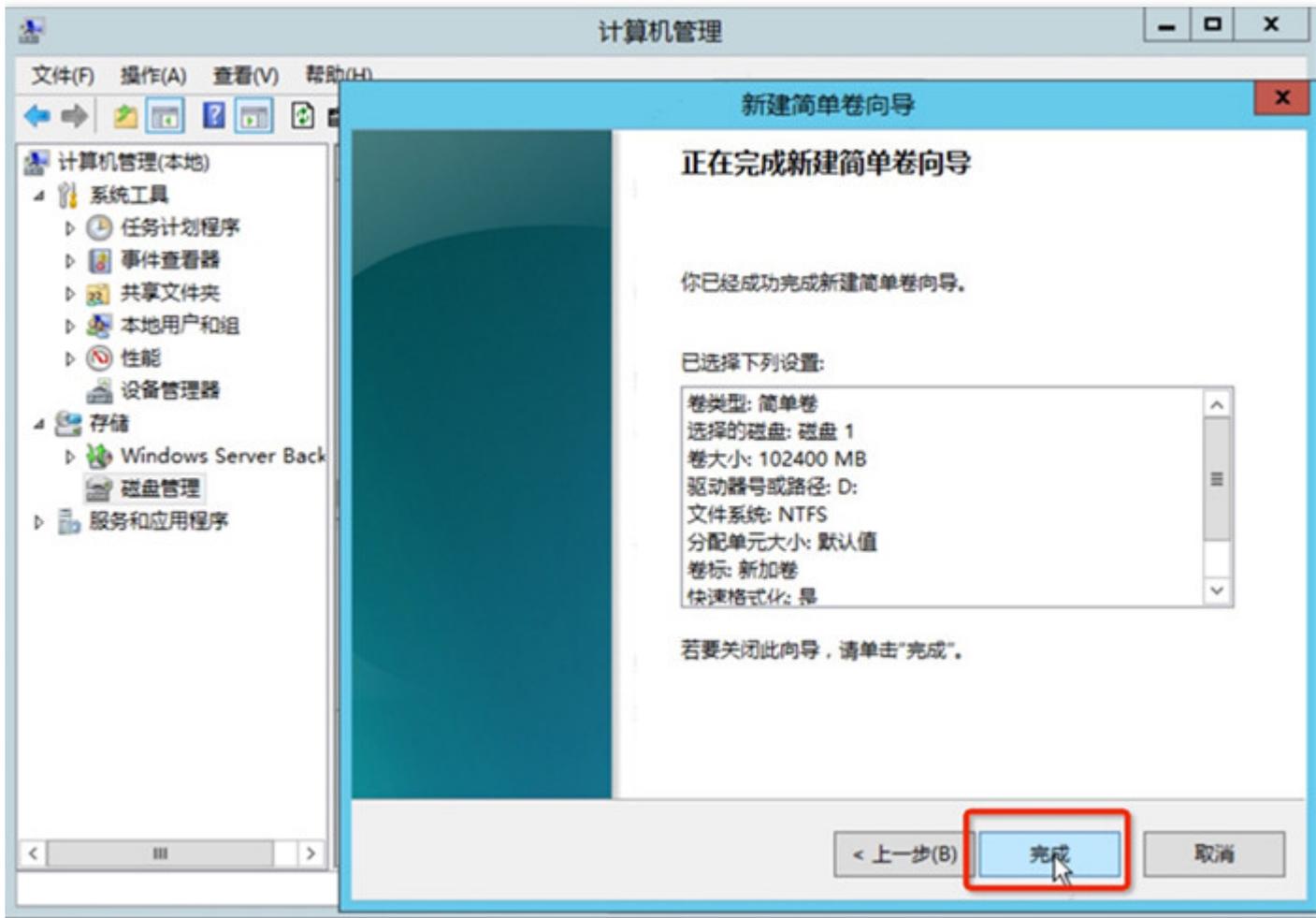
输入驱动器号，单击【下一步】：



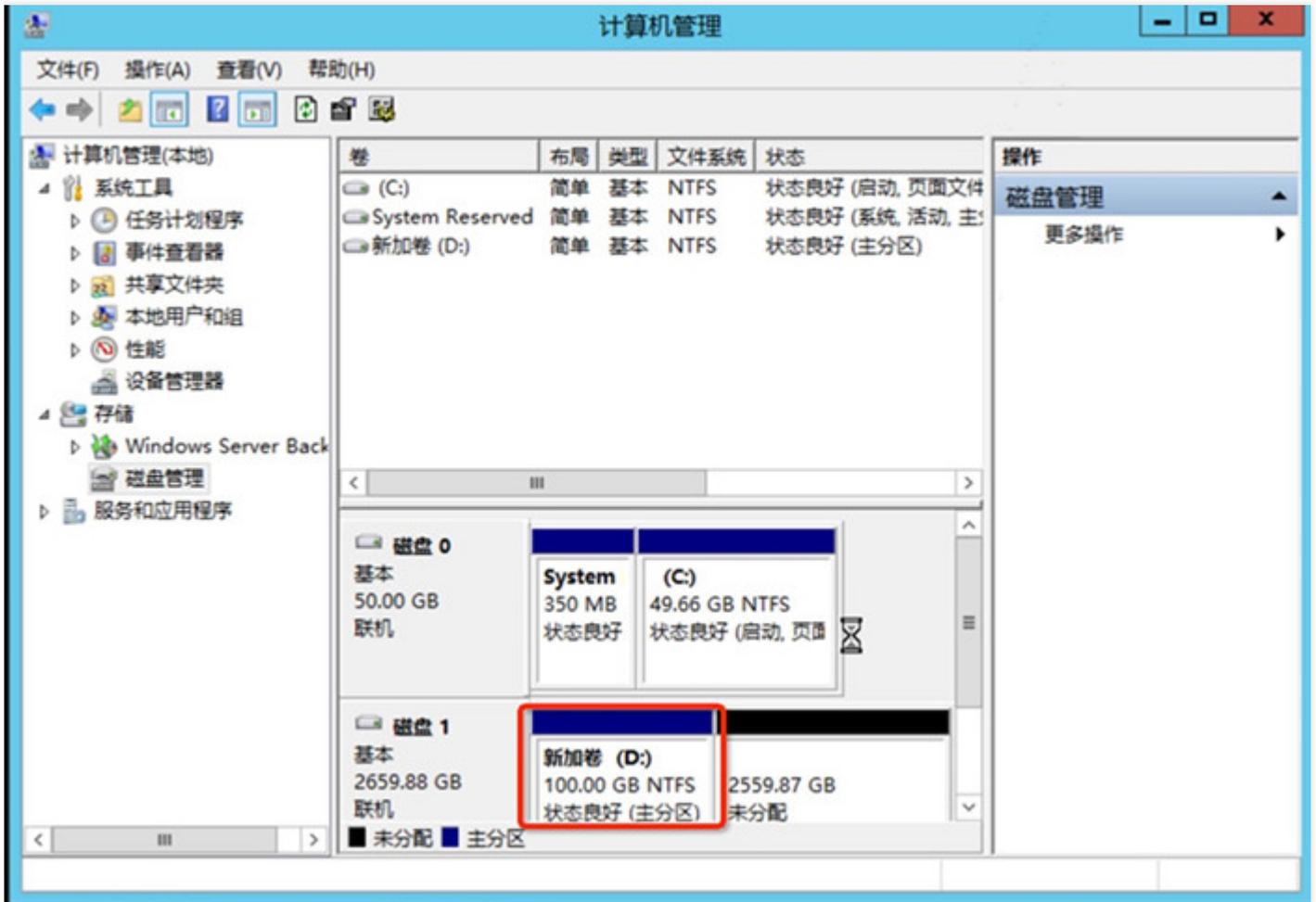
选择文件系统，格式化分区，单击【下一步】：

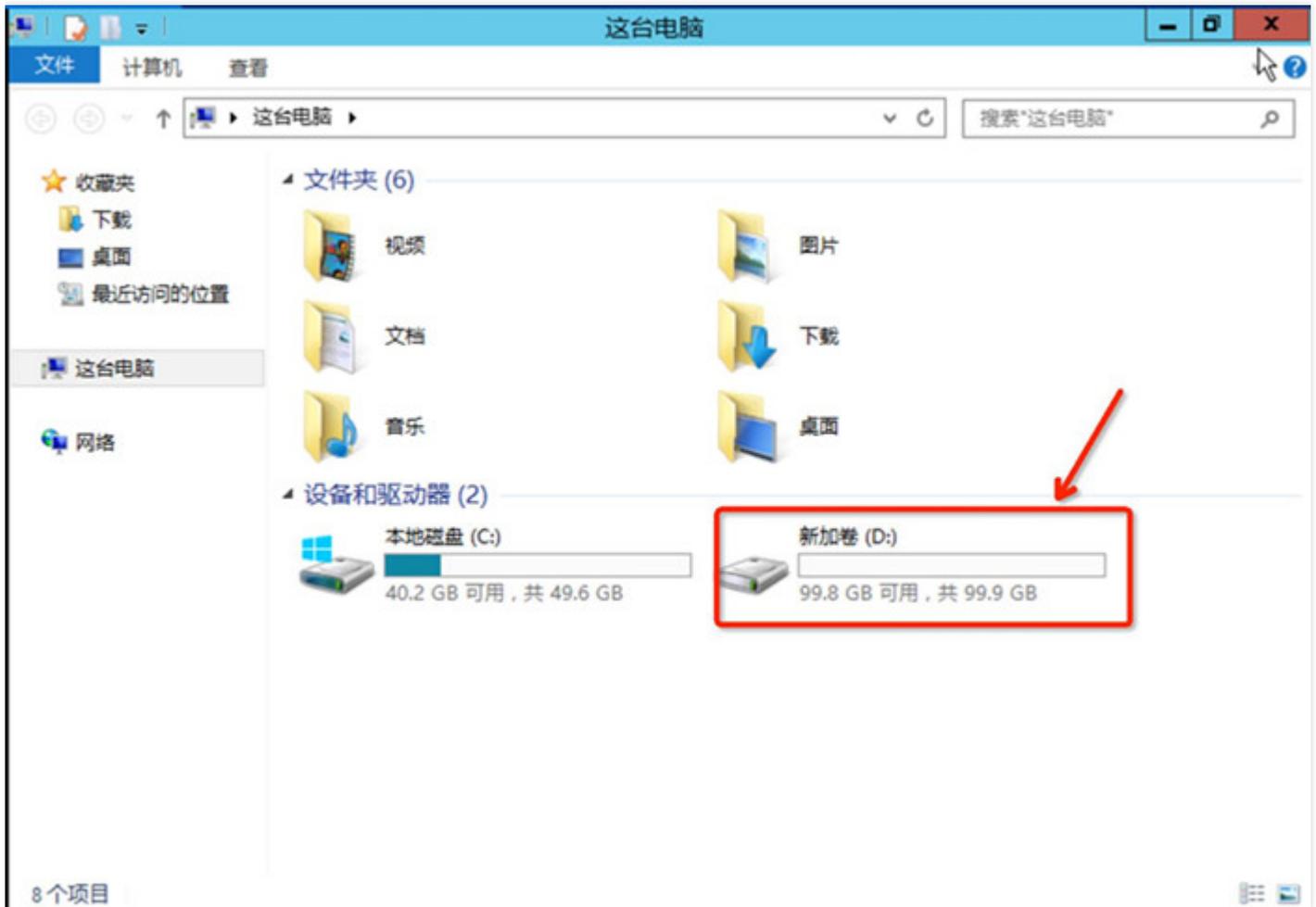


完成新建简单卷，单击【完成】：



查看新分区：



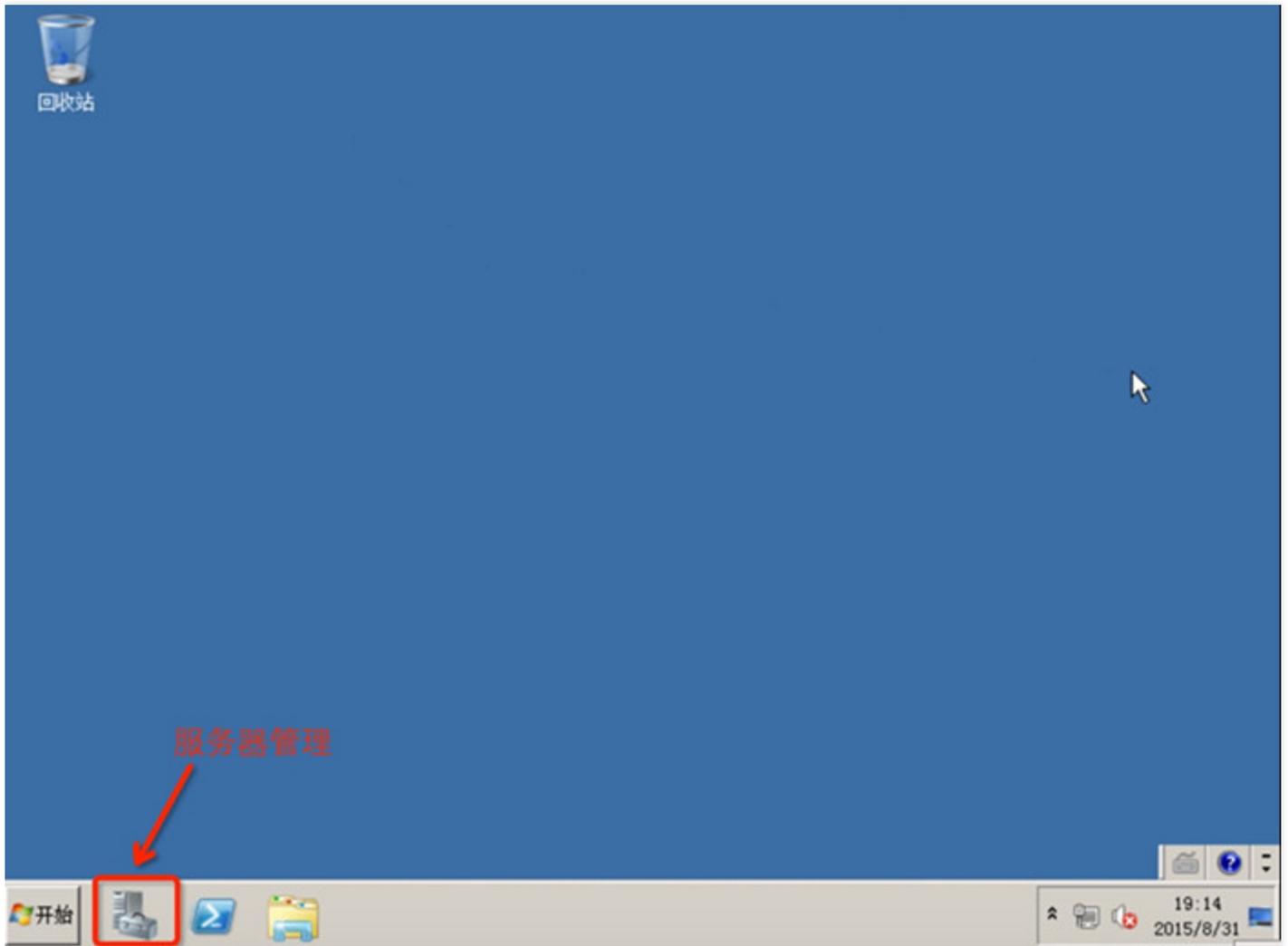


Windows2008磁盘分区和格式化

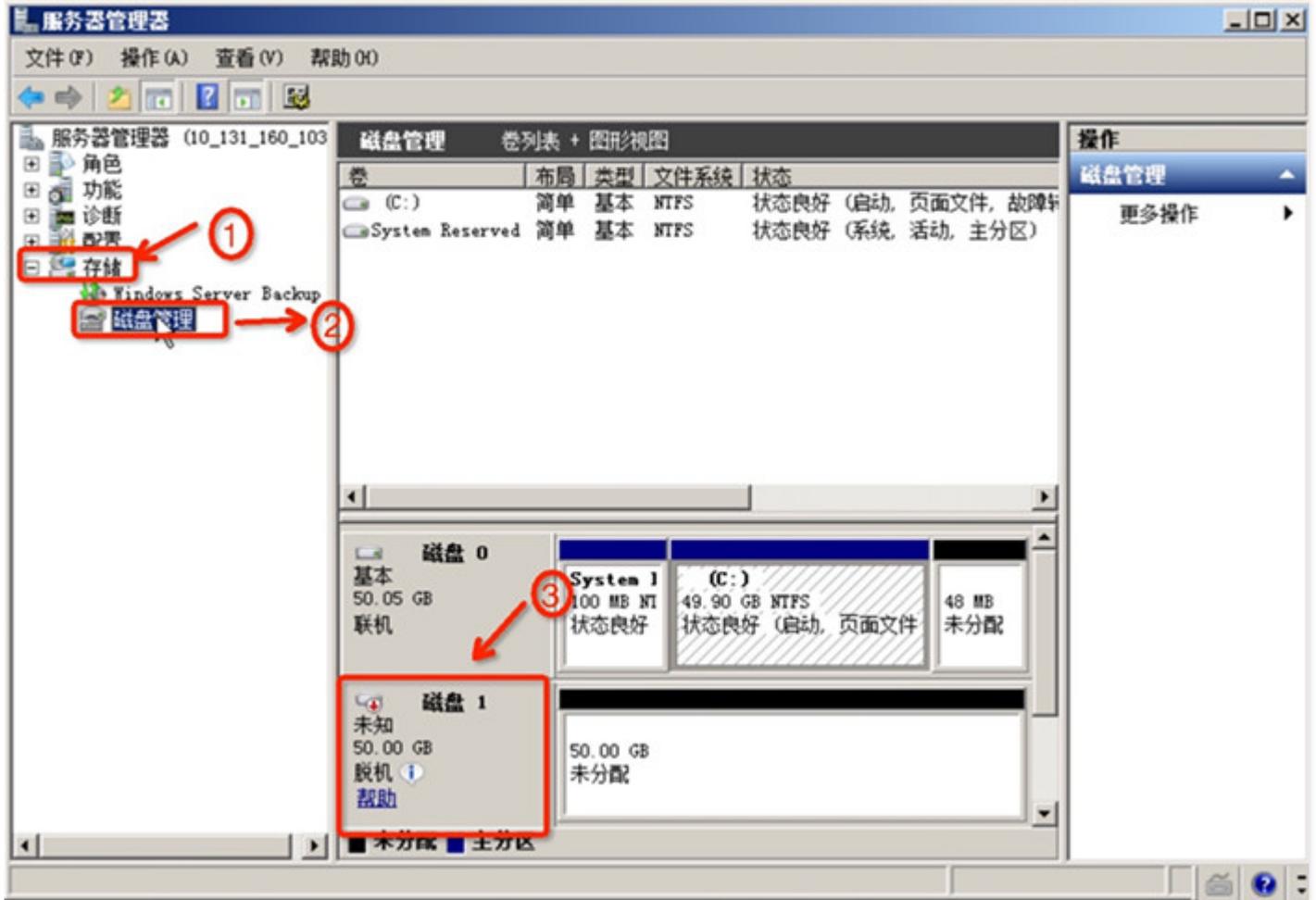
联机磁盘

Windows 2008 进入磁盘管理方法与 Windows 2012 不同，通过【服务器管理】-【存储】-【磁盘管理】的路径进入磁盘管理。

单击【服务器管理】：



单击【存储】-【磁盘管理】：

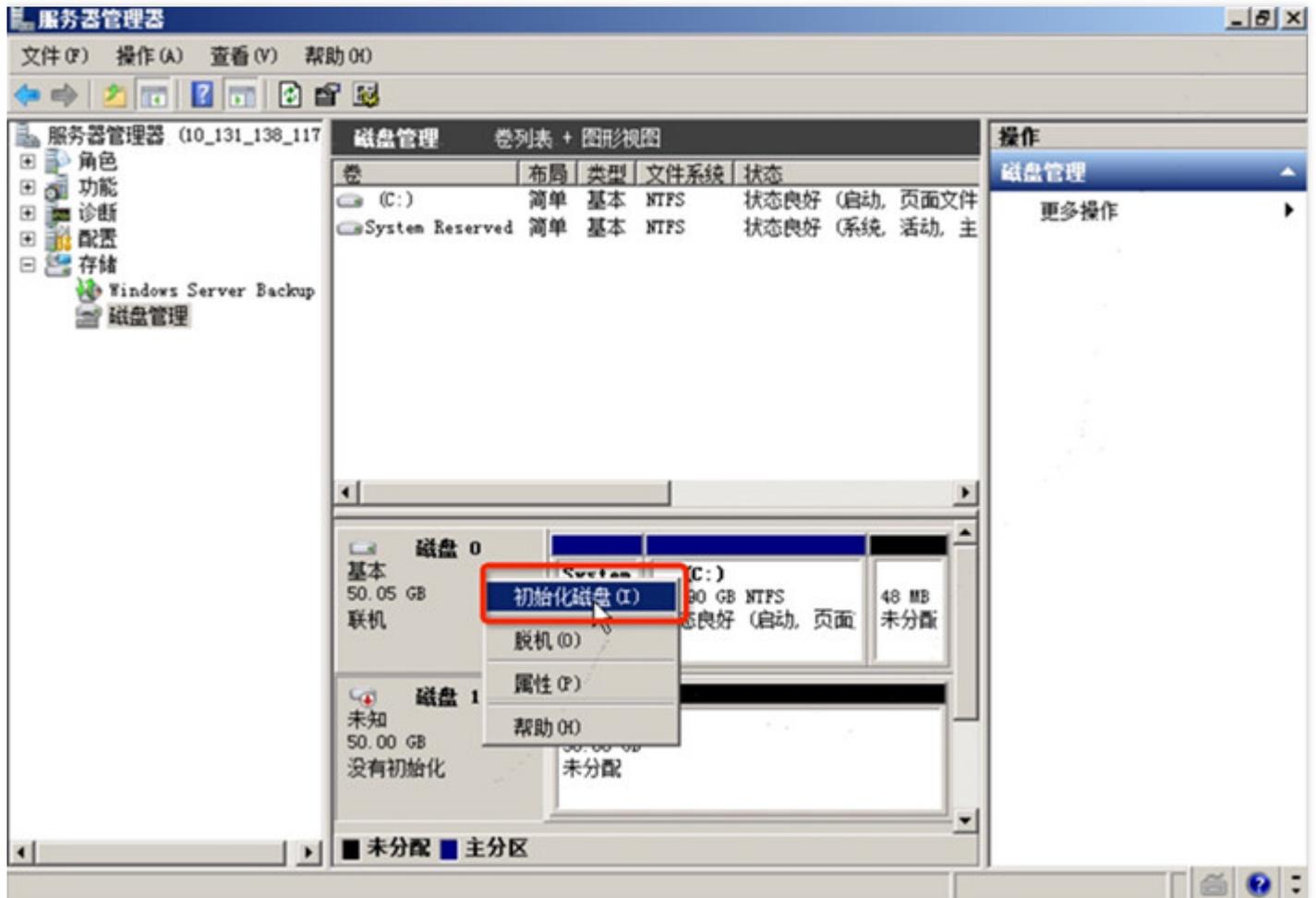


“磁盘1”初始情况下未联机，右键单击“磁盘1”，在弹出的菜单里单击【联机】：

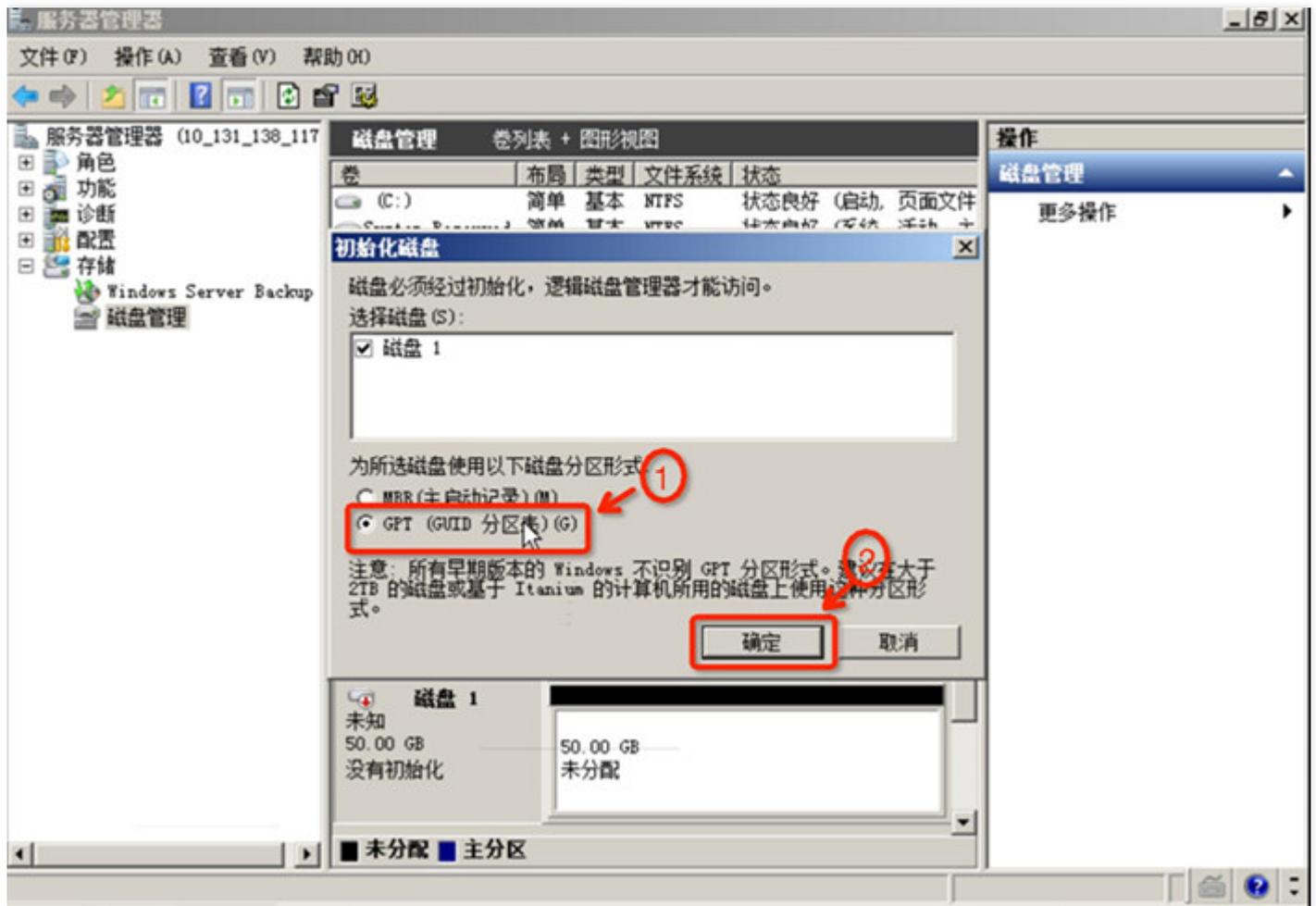


(可选) 格式化磁盘

再次右键单击“磁盘1”，在弹出的菜单里单击“初始化磁盘”：



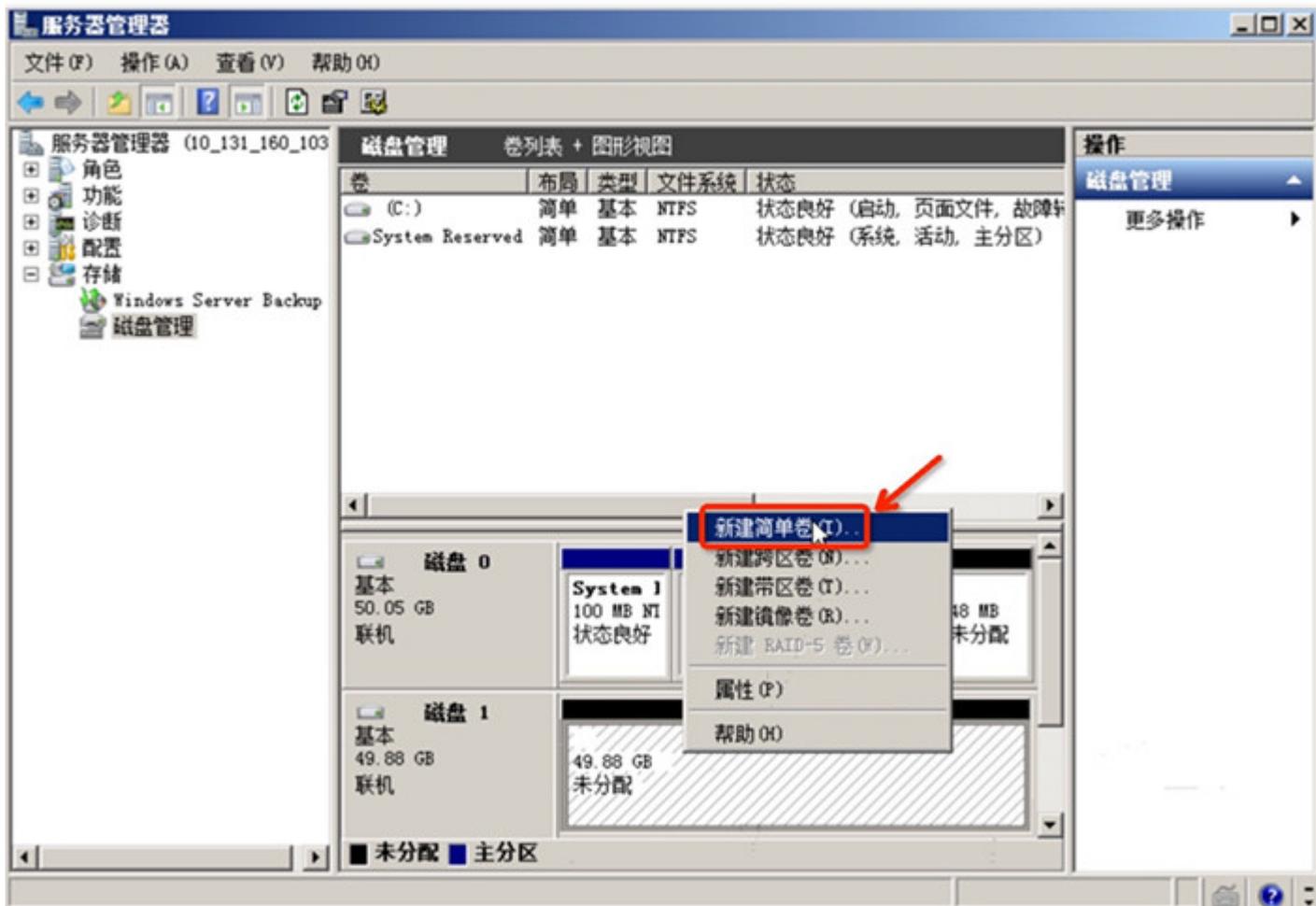
选择GPT的初始化方式，单击【确定】按钮：



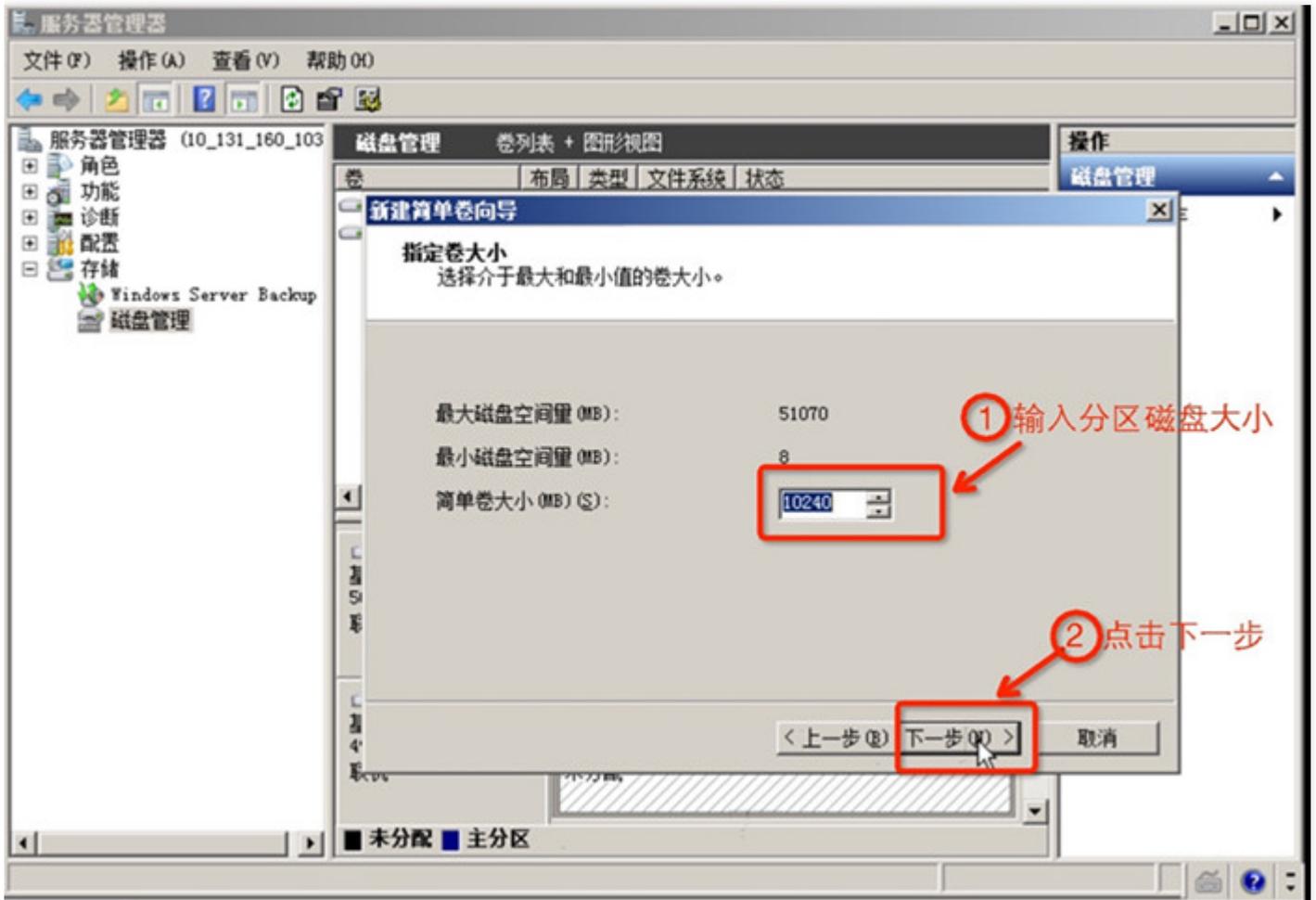
注：磁盘大于2TB时一定要选择GPT分区形式。

(可选) 对磁盘分区

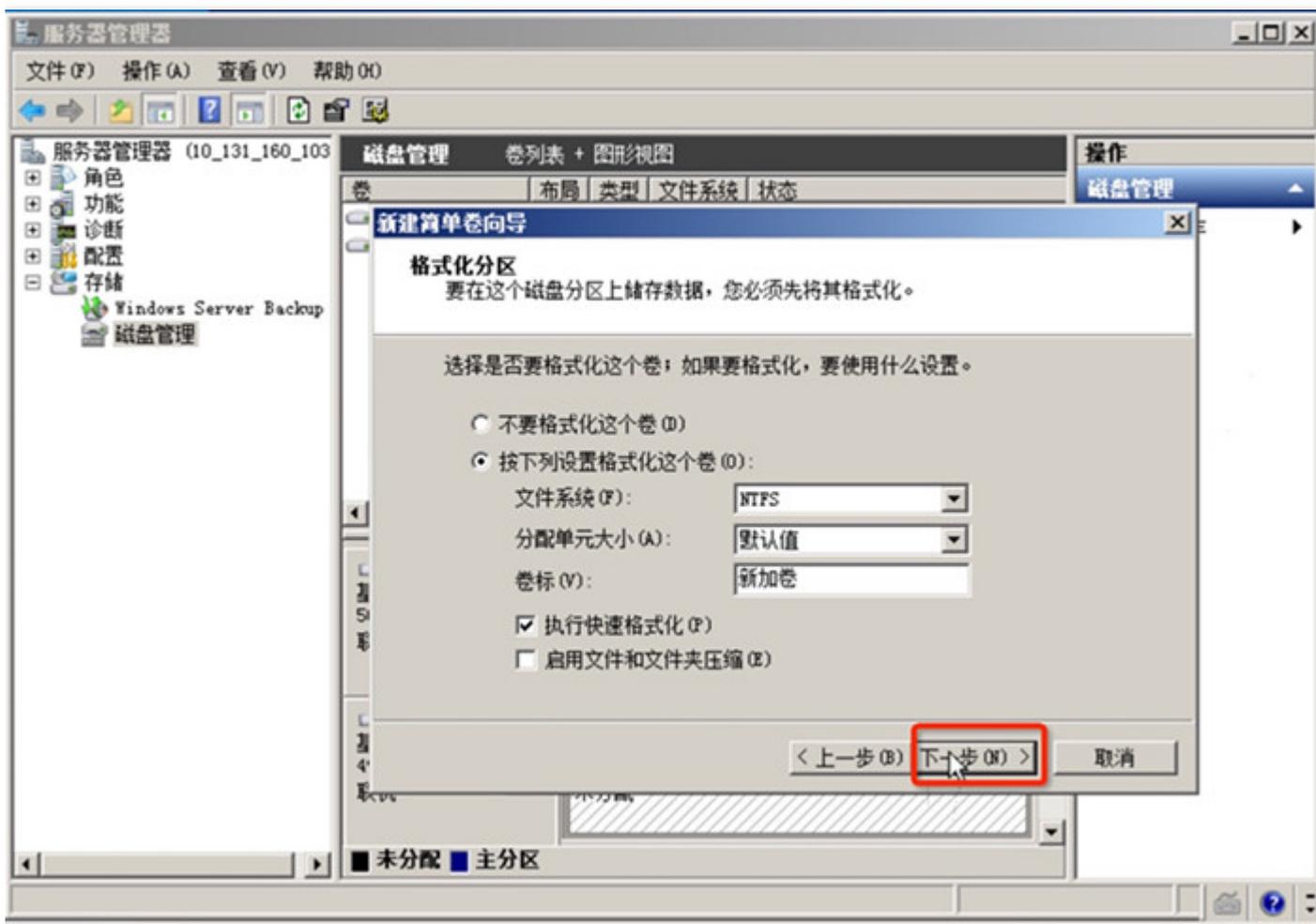
右键单击“磁盘1”后未分配的区域，在弹出的快捷菜单中选择【新建简单卷】：



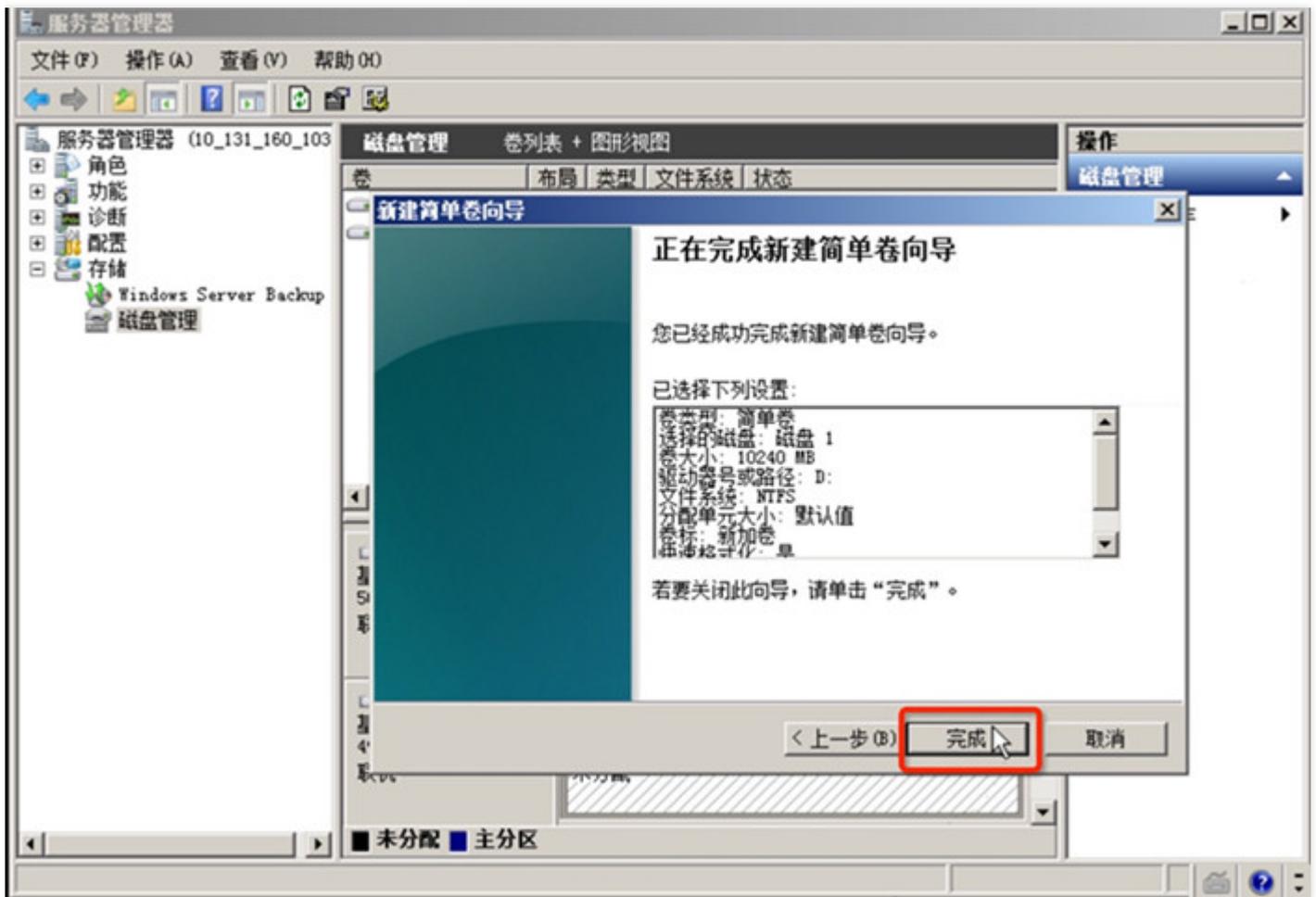
根据向导提示进行操作，输入分区磁盘的大小，单击【下一步】：



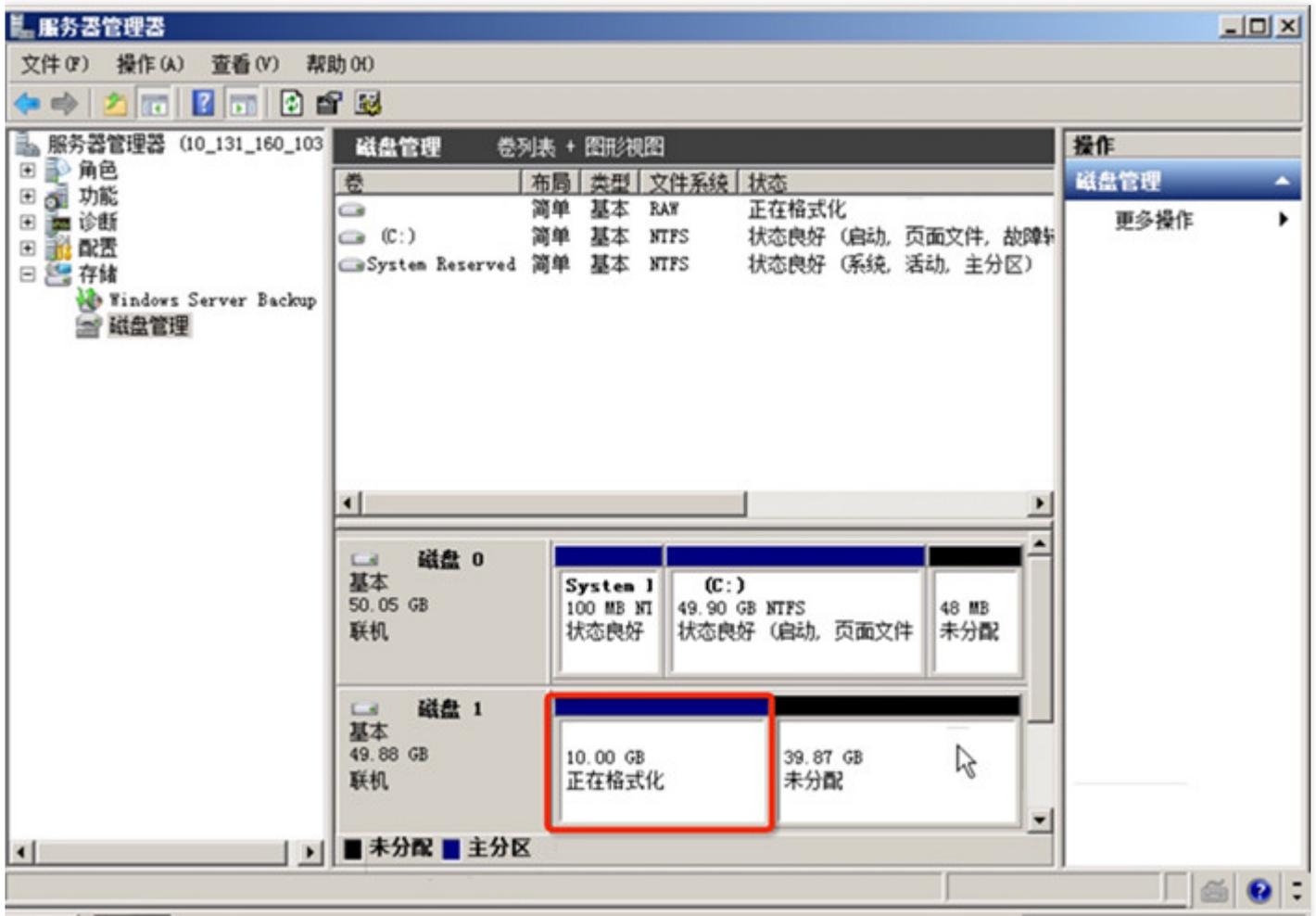
选择文件系统，格式化分区，单击【下一步】：



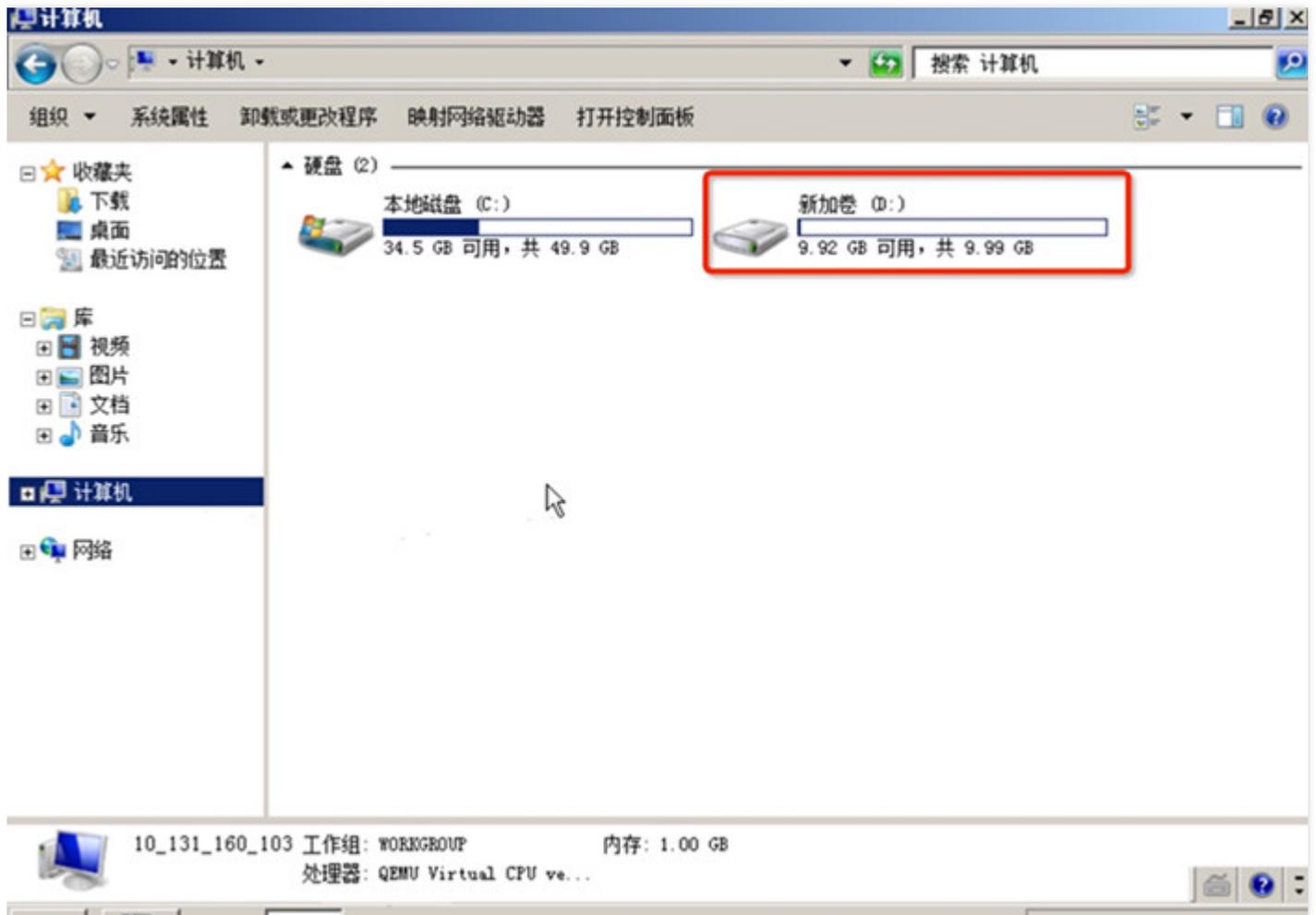
完成新建简单卷，单击【完成】按钮：



显示正在格式化：



在计算机界面可以看到新分区的数据盘：



注：请勿将基本硬盘转换到动态硬盘，倘若因此操作造成数据丢失，我们将不承担责任。

联机设置

在 Windows 操作系统下，常需要在磁盘管理中设置联机。为了方便您更好的使用弹性云硬盘，建议您对操作系统执行以下修改：

进入cmd命令行，执行以下命令

```
diskpart
san policy=onlineall
```

```
PS C:\Users\Administrator> diskpart

Microsoft DiskPart 版本 6.1.7601
Copyright (C) 1999-2008 Microsoft Corporation.
在计算机上: 10_141_55_49

DISKPART> san

SAN 策略: 使共享磁盘脱机

DISKPART> san policy=onlineall

DiskPart 已成功更改用于当前操作系统的 SAN 策略。

DISKPART> san

SAN 策略: 全部联机

DISKPART> c
```

操作后，当此弹性云硬盘重新挂载到 Windows 云服务器上后，如果弹性云硬盘包含有效的文件系统，用户则可以无需操作直接使用此弹性云硬盘了。

使用自定义镜像及数据盘快照启动新实例时自动挂载数据盘

在启动新的云服务器实例时，如果用户指定 **自定义镜像** 及 **数据盘快照**，腾讯云云硬盘可以支持启动云服务器实例后自动挂载（即不需要进行一系列的添加、分区、格式化等操作可直接读写数据盘）。用户需要在制作自定义镜像和数据盘快照前在原实例上进行一些操作，下文将详细描述。

在 Windows 系统下如果用户希望指定数据盘快照生产出来的云硬盘能够自动挂载至新的云服务器实例，指定的自定义镜像和数据盘快照必须满足以下要求：

- 自定义镜像中的 SAN 策略为：`onlineAll`。腾讯云目前提供的 Windows 公有镜像已默认进行相关设置，但仍建议用户在制作自定义镜像前检查下此配置，检查方法如下：

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> diskpart

Microsoft DiskPart 版本 6.1.7601
Copyright (C) 1999-2008 Microsoft Corporation.
在计算机上: 10_135_6_116

DISKPART> san

SAN 策略: 使共享磁盘脱机

DISKPART> san policy=onlineall

DiskPart 已成功更改用于当前操作系统的 SAN 策略。

DISKPART> san

SAN 策略: 全部联机
```

- 数据盘在制作快照前必须已经被格式化为 `ntfs` 或 `fat32` 格式。

只有同时满足以上两个条件才能保证新启动的 Windows 云服务器实例数据盘可以被自动识别和联机。

Linux 系统分区、格式化、挂载及创建文件系统

最近更新时间：2018-07-02 16:41:22

实例可以识别连接的云硬盘并将其视为普通硬盘。用户可以使用任何文件系统将云块存储设备格式化、分区及创建文件系统。此后任何写入文件系统的文件均写入云硬盘中，并且对使用该设备的应用程序是透明的。本文档通过一个示例演示如何在云服务器中使用块存储设备，并对一些需要注意的事项进行了详细的说明。您还可以从 [Windows 系统分区、格式化、联机及创建文件系统](#) 获得有关 Windows 实例上使用云硬盘的指引。

本示例使用的云服务器启动时一并购买了一块 20GB 的系统盘和一块 30GB 的数据盘（即非弹性云硬盘），而后又在云硬盘控制台购买了一块 10GB 的弹性云盘。

前提条件

- 请确保您已进行 [将云硬盘连接到 CVM 实例](#) 及 [登录 Linux 实例](#)。
- **格式化后，数据盘中的数据将被全部清空。请在格式化之前，确保数据盘中没有数据或对重要数据已进行备份。为避免服务发生异常，格式化前请确保云服务器已停止对外服务。**
- 当用户购买了多块云硬盘时，建议您对存放重要数据的弹性云盘设置自定义名称，并设置自动续费，防止因为没有及时续费导致弹性云盘到期对您的业务产生影响。
- 用户可以在 [云硬盘控制台](#) 中根据自定义名称或者关联的云服务器内网 IP 快速查找云硬盘。

本示例中，弹性云盘 ID 为 `ins-kjo6azag`，名称为 `弹性云盘使用演示`。

查看已挂载的硬盘

- 1) 运行 `fdisk -l` 命令查看硬盘信息。

此时可以看到随云服务器创建的非弹性云硬盘 `vdb` 和刚刚挂载的弹性云盘 `vdc` 都尚未格式化。

```
brw-rw---- 1 root disk    253, 16 Jul 21 16:25 vdb
brw-rw---- 1 root disk    253, 32 Jul 21 16:43 vdc
[root@VM_28_66_centos ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       20G   1.3G   18G   7% /
devtmpfs        912M   0    912M   0% /dev
tmpfs           921M   24K   921M   1% /dev/shm
tmpfs           921M   8.3M   912M   1% /run
tmpfs           921M   0    921M   0% /sys/fs/cgroup
tmpfs           185M   0    185M   0% /run/user/0
[root@VM_28_66_centos ~]# fdisk -l

Disk /dev/vda: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x0005fc9a

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1 *          2048     41943039     20970496   83   Linux

Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 62914560 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/vdc: 10.7 GB, 10737418240 bytes, 20971520 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

2) 执行 `ls -l /dev/disk/by-id/` 命令，可以在此处看到弹性云盘与设备名的对应关系。注意，非弹性云盘目前不会在这里显示任何信息。

```
tmpfs           185M   0    185M   0% /run/user/0
[root@VM_28_66_centos ~]# ll /dev/disk/by-id/
total 0
lrwxrwxrwx 1 root root 9 Jul 21 16:43 virtio-disk-bm42ztpm -> ../../vdc
[root@VM_28_66_centos ~]#
```

硬盘从未进行初始化时，您需要先创建文件系统，然后才能够使用它。从快照创建的云硬盘中可能已经含有文件系统，**如果您在现有的文件系统上创建新的文件系统，则将覆盖原有的全部数据。**

使用 `file -s device` 命令可列出特殊信息，例如文件系统类型。

```
sudo file -s /dev/xvdf
/dev/xvdf: data
```

如果前面的命令的输出仅显示该设备的 data，则说明设备上没有文件系统，您需要创建一个文件系统。您可继续下面的所有步骤。如果在包含文件系统的设备上运行此命令，则输出将有所不同，若返回形如 Linux rev 1.0 ext4 filesystem data 的输出则说明此硬盘上已经创建了文件系统，您可以跳过分区和格式化操作。

- 若磁盘已经初始化，可从 System 字段判断是否需要创建新的文件系统。如果 System 字段显示 EXT3、EXT4 等文件系统类型，则不需新建文件系统，可以跳过分区和格式化操作步骤。

(可选) 对磁盘进行分区操作

1) 对云硬盘进行分区操作。当然用户也可以无需分区直接进行格式化操作。这里我们演示了将弹性云盘划分为两个分区使用。执行以下命令：

```
fdisk /dev/vdb
```

按照界面的提示，依次输入“n”(新建分区)、“p”(新建主分区)、“1”(使用第1个主分区)，两次回车(使用默认配置)，输入“wq”(保存分区表)，回车开始分区。

这里是以创建 1 个分区为例，开发者也可以根据自己的需求创建多个分区。

```
[root@VM_124_230_centos ~]# fdisk /dev/vdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel with disk identifier 0x2d8cd07a.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
switch off the mode (command 'c') and change display units to
sectors (command 'u').

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
  d
Partition number (1-4): 1
First cylinder (1-104025, default 1):
Using default value 1
Last cylinder, +cylinders or +size{K,M,G} (1-104025, default 104025):
Using default value 104025

Command (m for help): wq
The partition table has been altered!

calling ioctl() to re-read partition table.
syncing disks.
[root@VM_124_230_centos ~]#
```

使用“fdisk -l”命令，即可查看到，新的分区 vdb1 已经创建完成。

```
[root@VM_124_230_centos ~]# fdisk -l
Disk /dev/vda: 8589 MB, 8589901824 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xcd6e8236

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1 *          1         1044     8385898+  83  Linux

Disk /dev/vdb: 53.7 GB, 53687091200 bytes
16 heads, 63 sectors/track, 104025 cylinders
Units = cylinders of 1008 * 512 = 516096 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x2d8cd07a

   Device Boot      Start         End      Blocks   Id  System
/dev/vdb1           1        104025     52428568+  83  Linux

Disk /dev/vdc: 2147 MB, 2147483648 bytes
16 heads, 63 sectors/track, 4161 cylinders
Units = cylinders of 1008 * 512 = 516096 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/vdc doesn't contain a valid partition table
[root@VM_124_230_centos ~]#
```

```
Command (m for help): p

Disk /dev/vdc: 10.7 GB, 10737418240 bytes, 20971520 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x53d03ce6

   Device Boot      Start         End      Blocks   Id  System
/dev/vdc1           2048        12584959     6291456   83  Linux
/dev/vdc2          12584960        20971519     4193280    5  Extended
/dev/vdc5          12587008        20971519     4192256   83  Linux

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@VM_28_66_centos ~]#
```

2) 分区后执行 `ls -l /dev/disk/by-id` 命令，可以看到以下内容：

```
[root@VM_28_66_centos ~]#  
[root@VM_28_66_centos ~]# ls -l /dev/disk/by-id/  
total 0  
lrwxrwxrwx 1 root root 9 Jul 21 16:58 virtio-disk-bm42ztpm -> ../../vdc  
lrwxrwxrwx 1 root root 10 Jul 21 16:58 virtio-disk-bm42ztpm-part1 -> ../../vdc1  
lrwxrwxrwx 1 root root 10 Jul 21 16:58 virtio-disk-bm42ztpm-part2 -> ../../vdc2  
lrwxrwxrwx 1 root root 10 Jul 21 16:59 virtio-disk-bm42ztpm-part5 -> ../../vdc5  
[root@VM_28_66_centos ~]#
```

（有条件）硬盘格式化

此步骤假定您在处理初始化的硬盘。如果该硬盘已经包含数据（如，从快照创建的云硬盘），请勿使用 `mkfs`（而应跳到下一步）。否则，您会格式化并删除现有数据。

运行 `mkfs.ext4 device_name` 命令格式化并创建 `ext4` 文件系统。用设备名称（例如，`/dev/vdb`）替换 `device_name`。根据应用程序的要求或操作系统的限制，您可以选择其他文件系统类型，如 `ext3` 或 `XFS`。

```
[root@VM_28_66_centos ~]#  
[root@VM_28_66_centos ~]# mkfs.ext3 /dev/vdc1  
mke2fs 1.42.9 (28-Dec-2013)  
Filesystem label=  
OS type: Linux  
Block size=4096 (log=2)  
Fragment size=4096 (log=2)  
Stride=0 blocks, Stripe width=0 blocks  
393216 inodes, 1572864 blocks  
78643 blocks (5.00%) reserved for the super user  
First data block=0  
Maximum filesystem blocks=1610612736  
48 block groups  
32768 blocks per group, 32768 fragments per group  
8192 inodes per group  
Superblock backups stored on blocks:  
    32768, 98304, 163840, 229376, 294912, 819200, 884736  
  
Allocating group tables: done  
Writing inode tables: done  
Creating journal (32768 blocks): done  
Writing superblocks and filesystem accounting information: done  
  
[root@VM_28_66_centos ~]# mkfs.ext3 /dev/vdc5  
mke2fs 1.42.9 (28-Dec-2013)  
Filesystem label=  
OS type: Linux  
Block size=4096 (log=2)  
Fragment size=4096 (log=2)  
Stride=0 blocks, Stripe width=0 blocks  
262144 inodes, 1048064 blocks  
52403 blocks (5.00%) reserved for the super user  
First data block=0  
Maximum filesystem blocks=1073741824  
32 block groups  
32768 blocks per group, 32768 fragments per group  
8192 inodes per group  
Superblock backups stored on blocks:  
    32768, 98304, 163840, 229376, 294912, 819200, 884736  
  
Allocating group tables: done  
Writing inode tables: done  
Creating journal (16384 blocks): done  
Writing superblocks and filesystem accounting information: done
```

挂载硬盘

```
mkdir /data/part1 -p # 创建示例挂载点  
mkdir /data/part5 -p # 创建示例挂载点  
mount /dev/vdc1 /data/part1 # 将vdc1挂载到/data/part1处  
mount /dev/vdc5 /data/part5 # 将vdc5挂载到/data/part5处  
touch /data/part1/disk-bm42ztpm-part1.txt # 创建一个空文件用于后续演示  
touch /data/part5/disk-bm42ztpm-part5.txt # 创建一个空文件用于后续演示  
yum install tree -y # 安装一个用于展示目录结构的工具  
tree /data # 查看/data目录结构
```

此时可以看到如下的结构树：

```
[root@VM_28_66_centos ~]#
[root@VM_28_66_centos ~]# tree /data
/data
|-- part1
|   |-- disk-bm42ztpm-part1.txt
|   |-- lost+found
|-- part5
|   |-- disk-bm42ztpm-part5.txt
|   |-- lost+found

4 directories, 2 files
[root@VM_28_66_centos ~]#
```

执行 `lsblk -f` 命令查询文件系统UUID和挂载点信息：

```
[root@VM_28_66_centos ~]#
[root@VM_28_66_centos ~]# lsblk -f
NAME      FSTYPE LABEL UUID                                MOUNTPOINT
vda
`-vda1    ext3      49f819fd-e56d-48a4-86d3-7ebe0a68ec88 /
vdb       ext3      e95f9d9c-d968-4da4-822a-f8e2ef140e4e
vdc
|-vdc1    ext3      970da23c-52bb-4208-b989-4b717a89c7d3 /data/part1
|-vdc2
`-vdc5    ext3      2a5294d9-ba68-4025-a288-f13ca8003b0a /data/part5
[root@VM_28_66_centos ~]#
```

(可选) 设置自动挂载

如果此时重启云服务器

，可以发现刚刚挂载点已经消失。如果希望云服务器在重启或开机时能自动 mount 数据盘，必须将分区信息添加到 `/etc/fstab` 中。如果没有添加则云服务器重启或重新开机后都不能自动挂载数据盘。在 `/etc/fstab` 配置文件中可以使用三种不同的方法使文件系统可以找到 mount 点：

自动mount方法	优点	缺点
使用设备名称		假如您将云服务器上的弹性云盘解挂后再次挂载（例如迁移数据时），该名称有可能会发生变化，因此有可能会使您的自动挂载设置失效
使用文件系统 UUID		与文件系统相关，重新格式化文件系统后，UUID将会发生变化，因此有可能会使您的自动挂载设置失效
使用弹性云盘软链接	与设备名及文件系统无关，与实际使用的云硬盘唯一对应的名称	只有弹性云盘才会有此软链接，无法感知到分区的格式化操作

从下图可以看出，重新格式化文件系统后 UUID 发生了变化：

```
[root@VM_28_66_centos ~]# umount /dev/vdc5
[root@VM_28_66_centos ~]# lsblk -f
NAME FSTYPE LABEL UUID MOUNTPOINT
vda
├─vda1 ext3 49f819fd-e56d-48a4-86d3-7ebe0a68ec88 /
├─vdb ext3 e95f9d9c-d968-4da4-822a-f8e2ef140e4e
└─vdc
   ├─vdc1 ext3 970da23c-52bb-4208-b989-4b717a89c7d3 /data/part1
   ├─vdc2
   └─vdc5 ext3 2a5294d9-ba68-4025-a288-f13ca8003b0a
[root@VM_28_66_centos ~]# mkfs.ext3 /dev/vdc5
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
262144 inodes, 1048064 blocks
52403 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=1073741824
32 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

[root@VM_28_66_centos ~]#
[root@VM_28_66_centos ~]# lsblk -f
NAME FSTYPE LABEL UUID MOUNTPOINT
vda
├─vda1 ext3 49f819fd-e56d-48a4-86d3-7ebe0a68ec88 /
├─vdb ext3 e95f9d9c-d968-4da4-822a-f8e2ef140e4e
└─vdc
   ├─vdc1 ext3 970da23c-52bb-4208-b989-4b717a89c7d3 /data/part1
   ├─vdc2
   └─vdc5 ext3 cac64424-b399-4244-a876-477b2a91ba51
[root@VM_28_66_centos ~]#
```

同理，从下图可以看出，在控制台卸载弹性云盘并重新挂载后，设备名称发生了变化：

```
[root@VM_28_66_centos ~]#  
[root@VM_28_66_centos ~]# ls -l /dev/ | grep vd  
brw-rw---- 1 root disk    253,  0 Jul 21 17:32 vda  
brw-rw---- 1 root disk    253,  1 Jul 21 17:32 vda1  
brw-rw---- 1 root disk    253, 16 Jul 21 17:32 vdb  
brw-rw---- 1 root disk    253, 32 Jul 21 17:32 vdc  
brw-rw---- 1 root disk    253, 33 Jul 21 17:32 vdc1  
brw-rw---- 1 root disk    253, 34 Jul 21 17:32 vdc2  
brw-rw---- 1 root disk    253, 37 Jul 21 17:32 vdc5  
[root@VM_28_66_centos ~]#  
[root@VM_28_66_centos ~]# ls -l /dev/ | grep vd  
brw-rw---- 1 root disk    253,  0 Jul 21 17:32 vda  
brw-rw---- 1 root disk    253,  1 Jul 21 17:32 vda1  
brw-rw---- 1 root disk    253, 16 Jul 21 17:32 vdb  
[root@VM_28_66_centos ~]#  
[root@VM_28_66_centos ~]# ls -l /dev/ | grep vd  
brw-rw---- 1 root disk    253,  0 Jul 21 17:32 vda  
brw-rw---- 1 root disk    253,  1 Jul 21 17:32 vda1  
brw-rw---- 1 root disk    253, 16 Jul 21 17:32 vdb  
brw-rw---- 1 root disk    253, 48 Jul 21 17:37 vdd  
brw-rw---- 1 root disk    253, 49 Jul 21 17:37 vdd1  
brw-rw---- 1 root disk    253, 50 Jul 21 17:37 vdd2  
brw-rw---- 1 root disk    253, 53 Jul 21 17:37 vdd5  
[root@VM_28_66_centos ~]#
```

综上，我们建议您始终使用第三种方式实现自动挂载弹性云盘。创建 `/etc/fstab` 文件的备份，保证意外损坏或删除文件的情况下，可以使用该备份。

```
cp /etc/fstab /etc/fstab.backup
```

打开 `/etc/fstab` 文件。

```
vi /etc/fstab
```

使用弹性云盘软链接（推荐）

输入：`device_name mount_point file_system_type fs_mntops fs_freq fs_passno`

示例：

```
/dev/disk/by-id/virtio-disk-bm42ztpm-part1 /data/part1 ext3 defaults,nofail 0 1
```

```
/dev/disk/by-id/virtio-disk-bm42ztpm-part5 /data/part5 ext3 defaults,nofail 0 1
```

最后三个字段分别是文件系统安装选项、文件系统转储频率和启动时的文件系统检查顺序。一般使用示例中的值（`defaults,nofail 0 2`）即可。有关 `/etc/fstab` 条目的更多信息，在命令行上输入 `man fstab` 即可查看。

运行 `mount -a` 命令，如果运行通过则说明文件正常，刚刚创建的文件系统会在下次启动时自动安装。

（可选）使用自定义镜像及数据盘快照启动新实例时自动挂载数据盘

在启动新的云服务器实例时，如果用户指定 **自定义镜像** 及 **数据盘快照**，腾讯云云硬盘可以支持启动云服务器实例后自动挂载（即不需要进行一系列的添加、分区、格式化等操作可直接读写数据盘）。用户需要在制作自定义镜像和数据盘快照前在原实例上进行一些操作，下文将详细描述。

在 Linux 系统下如果用户希望指定数据盘快照生产出来的云硬盘能够自动挂载至新的云服务器实例，指定的自定义镜像和数据盘快照必须满足以下要求：

- 数据盘在制作快照前 **必须** 已经格式化过，也即在原云服务器上已经 mount 成功。
- 系统盘在制作自定义镜像前，需要在 `/etc/rc.local` 文件中添加以下命令，将数据盘挂载点写入文件中：

```
mkdir -p <mount-point>
mount <device-id> <mount-point>
```

其中：`<mount-point>` 请填入文件系统的挂载点如 `/mydata`，`<device-id>` 请填入用户的实际文件分区位置，如 `/dev/vdb`(无分区有文件系统的设备名) 和 `/dev/vdb1`(有分区有文件系统)。

只有同时满足以上两个条件才能保证新启动的 Linux 云服务器实例数据盘可以被自动识别和挂载。

扩容云硬盘

最近更新时间：2018-06-22 17:09:51

云硬盘是云上可扩展的存储设备，用户可以在创建云硬盘后随时扩展其大小，以增加存储空间，同时不失去云硬盘上原有的数据。要达到扩容并使用扩容空间的目的，用户需要扩容实体云硬盘大小，然后扩展其上的文件系统以识别新近可用的空间。

如果云硬盘的最大容量（4T）都无法满足您的需求，您可以使用 RAID 跨多个物理硬盘来创建一个逻辑上的超大空间。有关更多信息，请参阅 [配置云硬盘 RAID 组](#)。

如果您的硬盘分区为 MBR 格式，扩容到超过 2TB 大小时 MBR 分区格式已不支持。建议您新建一块数据盘，使用 GPT 分区后通过复制的方式将数据拷贝至新盘上。

扩容类型为云硬盘的数据盘

通过云硬盘控制台扩容CBS数据盘

- 1) 打开云服务器 [CVM控制台](#)。
- 2) 单击导航窗格中的【云硬盘】。
- 3) 只有显示为【未挂载】状态且【支持挂载/卸载】的硬盘可以扩容，单击末尾【更多】-【扩容】按钮，选择需要的新大小（必须大于或等于当前大小）并支付即完成了实体硬盘的扩容操作。

对于已经连接到了实例的弹性云盘请先执行 [卸载云硬盘](#) 操作

通过API扩容CBS数据盘

请参考 [ResizeCbsStorage 接口](#)。

通过云服务器控制台扩容CBS数据盘

- 1) 打开云服务器 [CVM控制台](#)。
- 2) 单击导航窗格中的【云主机】。
- 3) 只有显示为【关机】状态且系统盘、数据盘均为云硬盘的实例可以扩容，单击末尾【更多】-【云主机设置】-【调整硬盘】按钮，选择需要的新大小（必须大于或等于当前大小）并支付即完成了实体硬盘的扩容操作。

对于正在运行的系统盘、数据盘均为云硬盘的实例，需要扩容请先执行 [实例关机](#) 操作。

扩容类型为云硬盘的系统盘

系统盘类型为云硬盘时，支持对系统盘的扩容，但仅可以通过对云服务器进行重装系统操作，详细内容请参考[重装系统](#)。

扩容文件系统

扩容Windows文件系统

最近更新时间：2018-08-27 09:42:10

Windows 云硬盘扩容有以下两种场景：

- 对于新增的容量空间，建立独立的新分区，老的分区保持不变。
- 扩容旧的分区至新增的容量空间，并且保持老分区的数据不丢失。

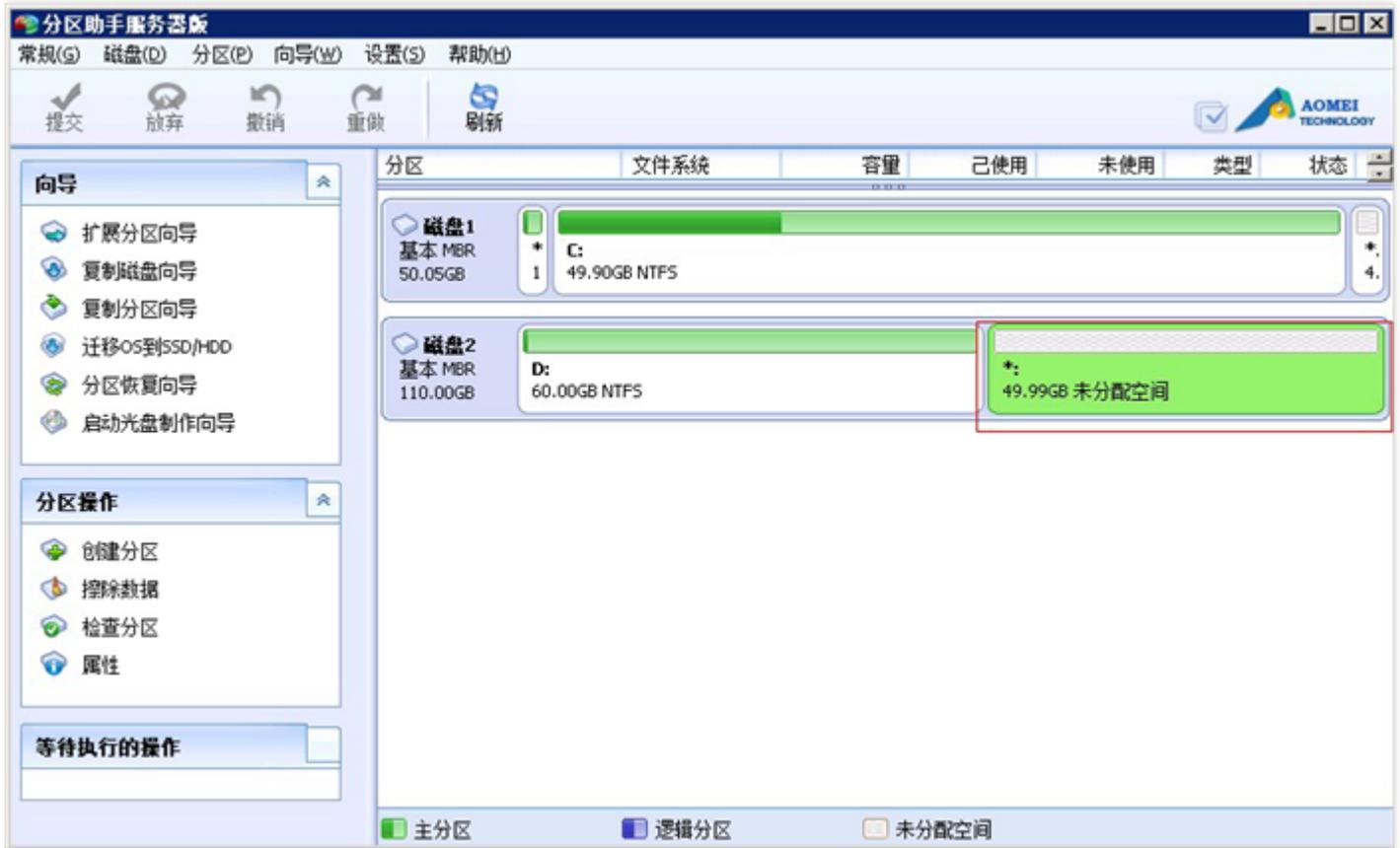
以上两种场景在您的 Windows 云硬盘升级成功之后（看到云硬盘容量变化），都可以通过 Windows 下的分区扩容工具分区助手完成分区扩容，并且保证原数据不会丢失。

前提条件

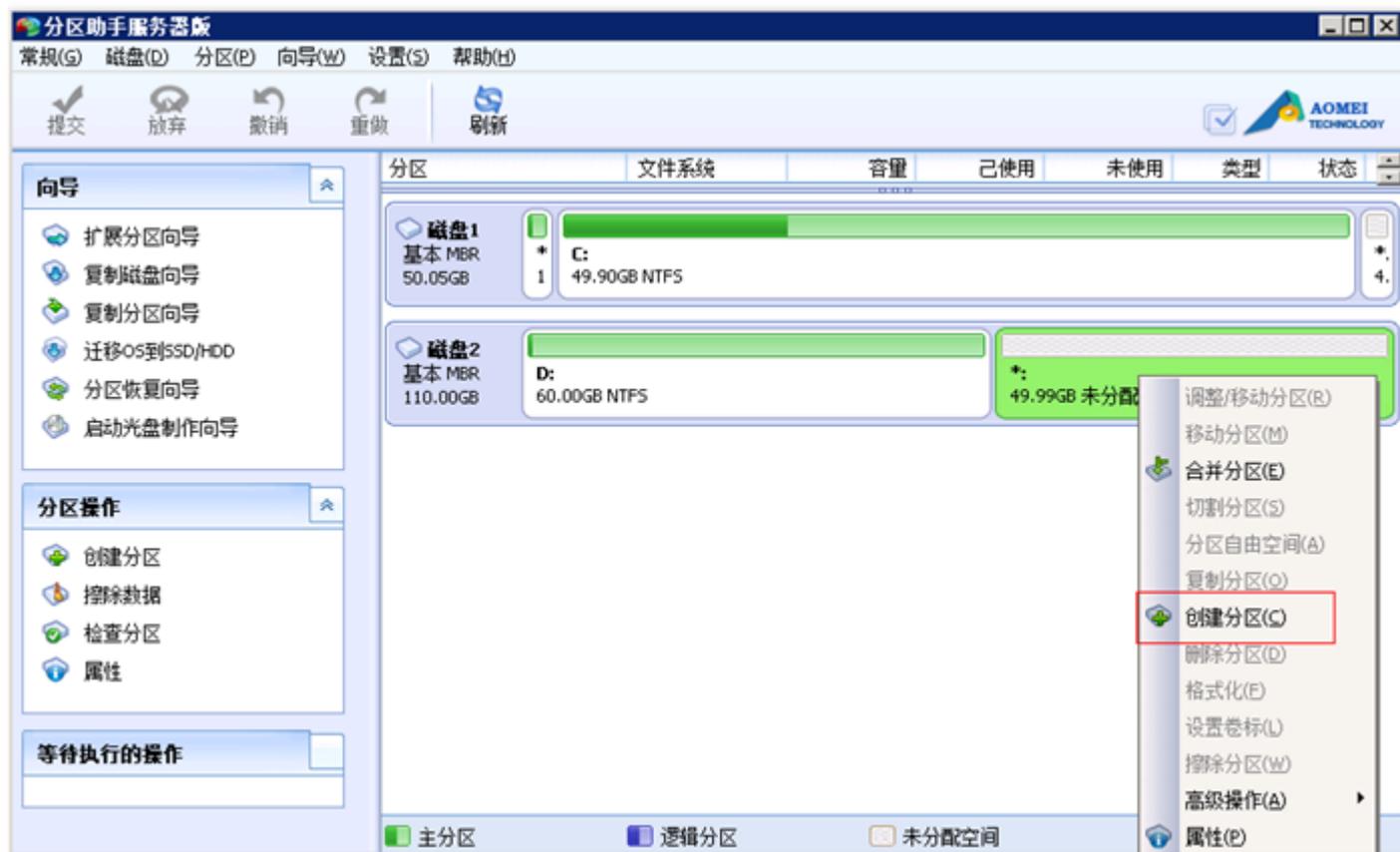
- 用户需要先下载 [分区助手](#)。
- 然后完成 [扩容实体云硬盘](#) 操作。
- 若此云硬盘上没有经过格式化和创建文件系统，直接在原有空白云硬盘基础上增加了容量，用户可以直接参考 [Windows 系统分区、格式化及创建文件系统](#) 相关操作。

新空间格式化成一个独立分区

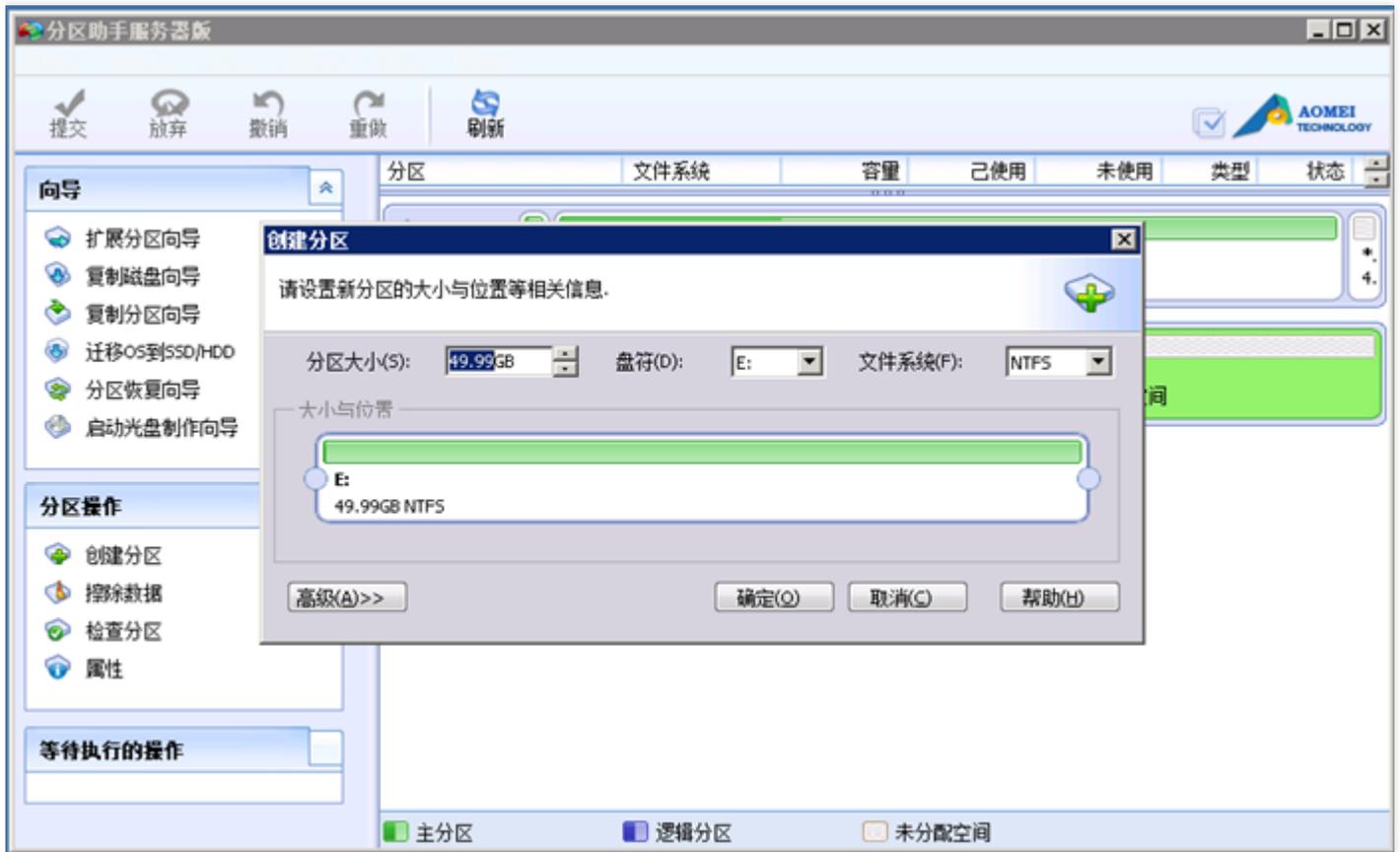
打开分区助手可以看到新扩容未使用的磁盘空间：



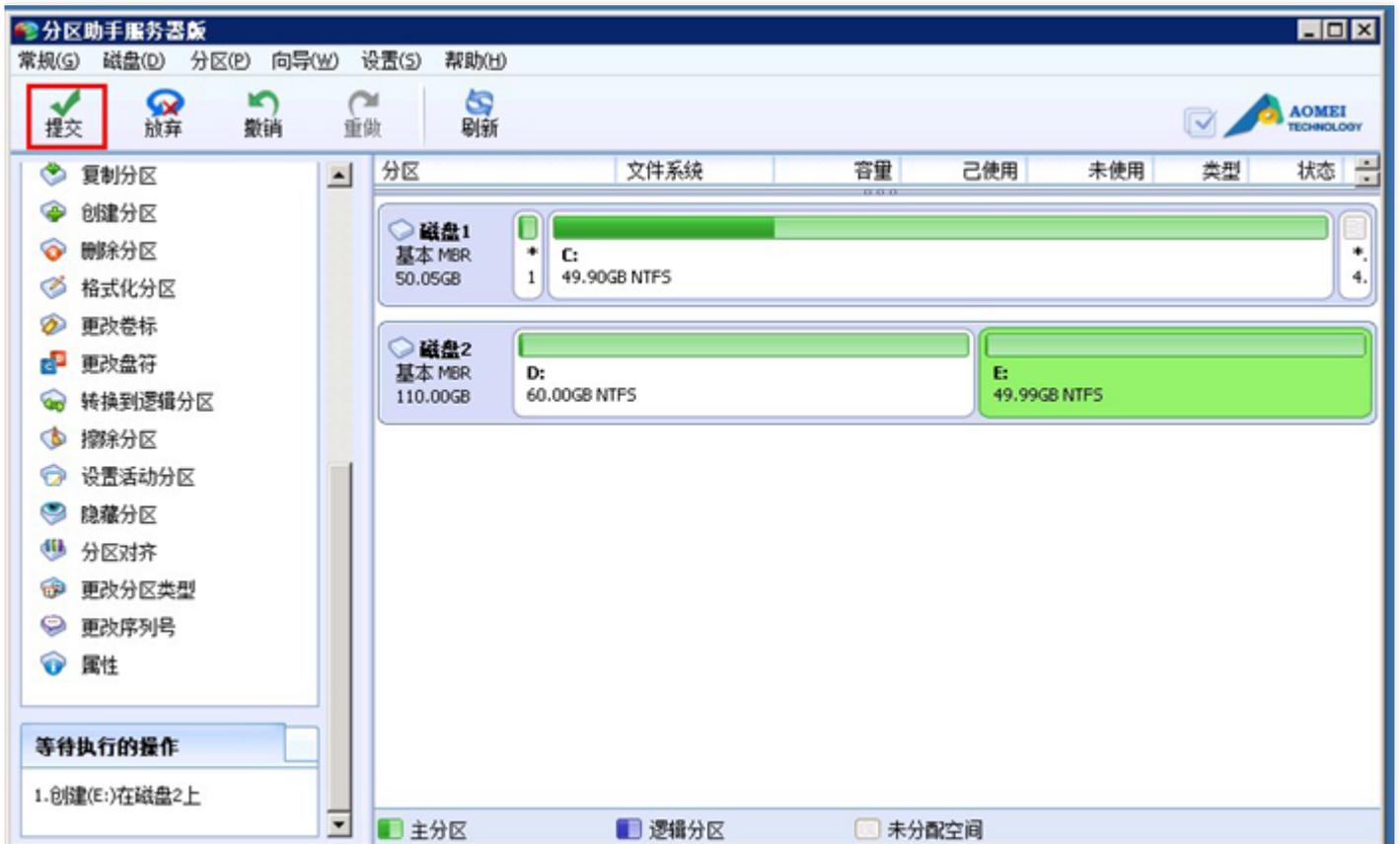
右键选中未使用的磁盘空间，选择【创建分区】：



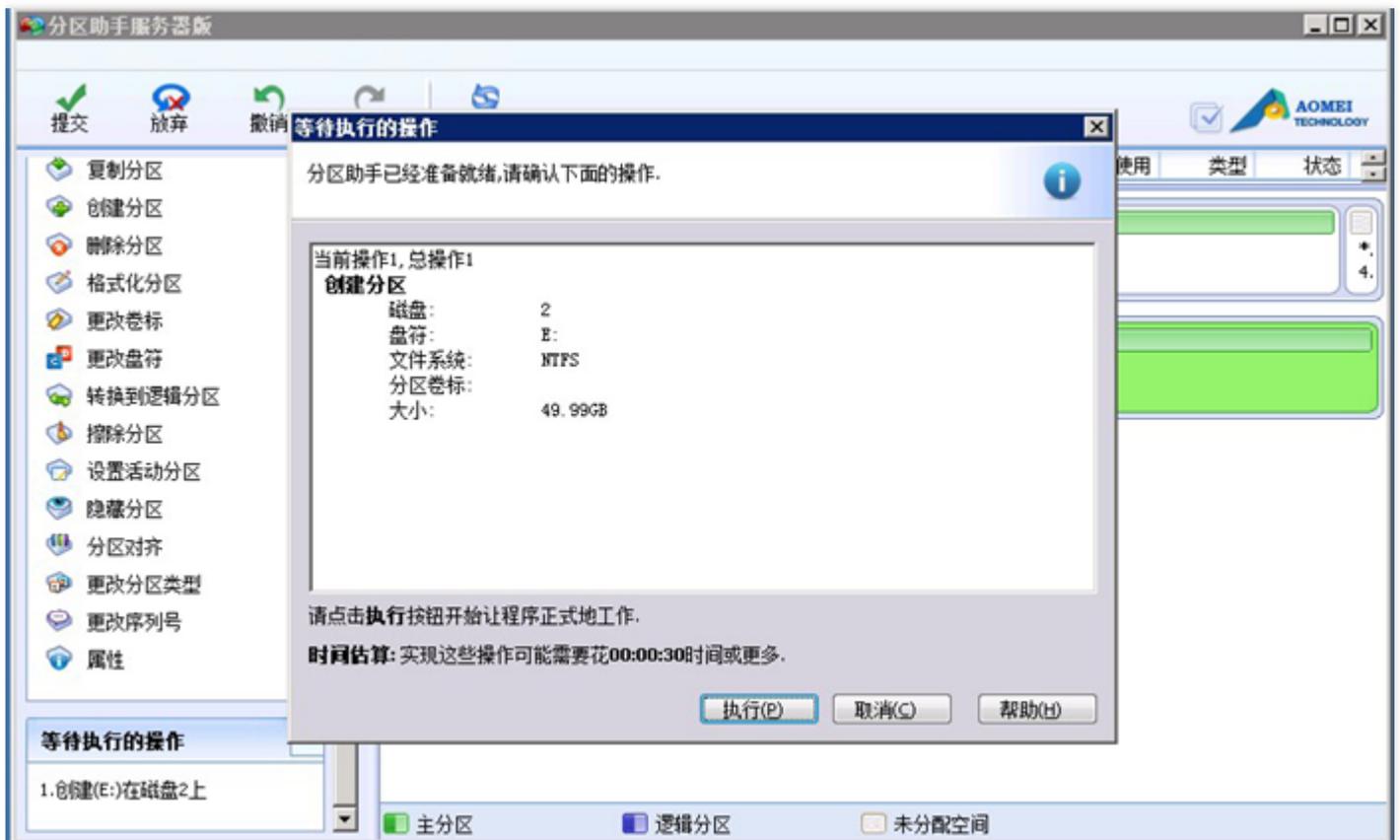
在弹出框中，输入需要的分区大小、盘符和文件系统，然后单击【确定】按钮：



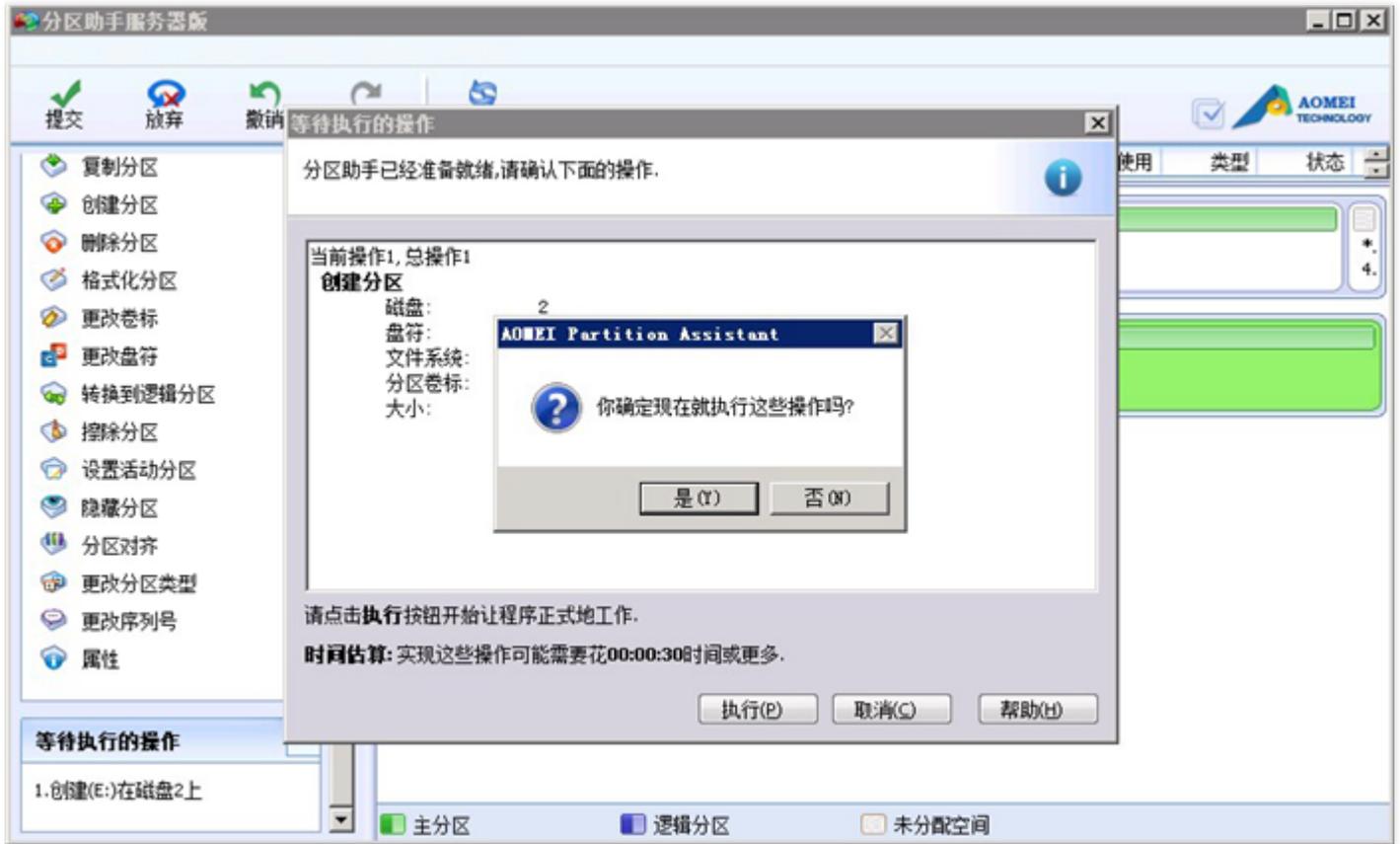
在左上角单击【提交】任务按钮：



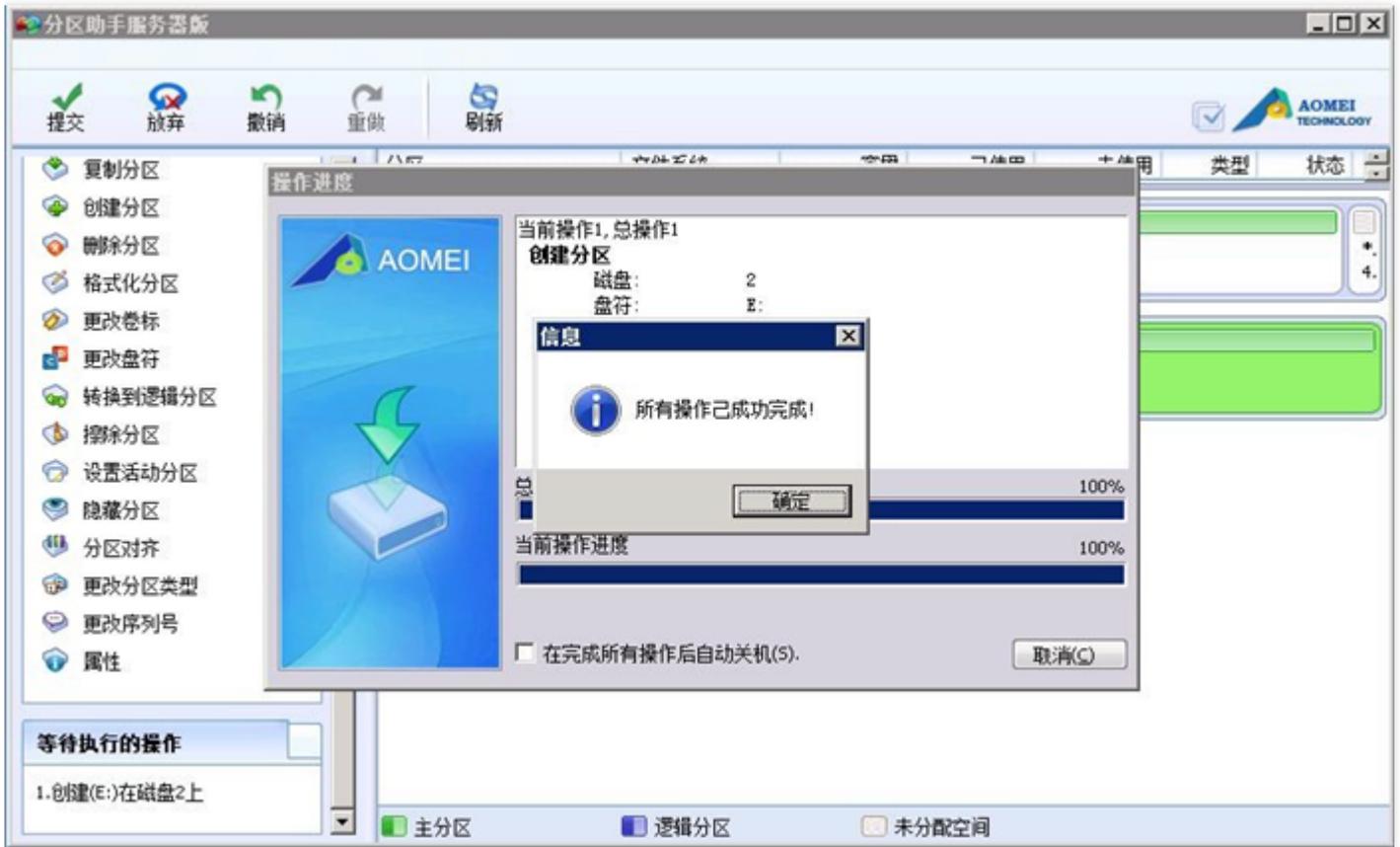
在弹出框中，确认格式分区的信息无误后，单击【执行】按钮：



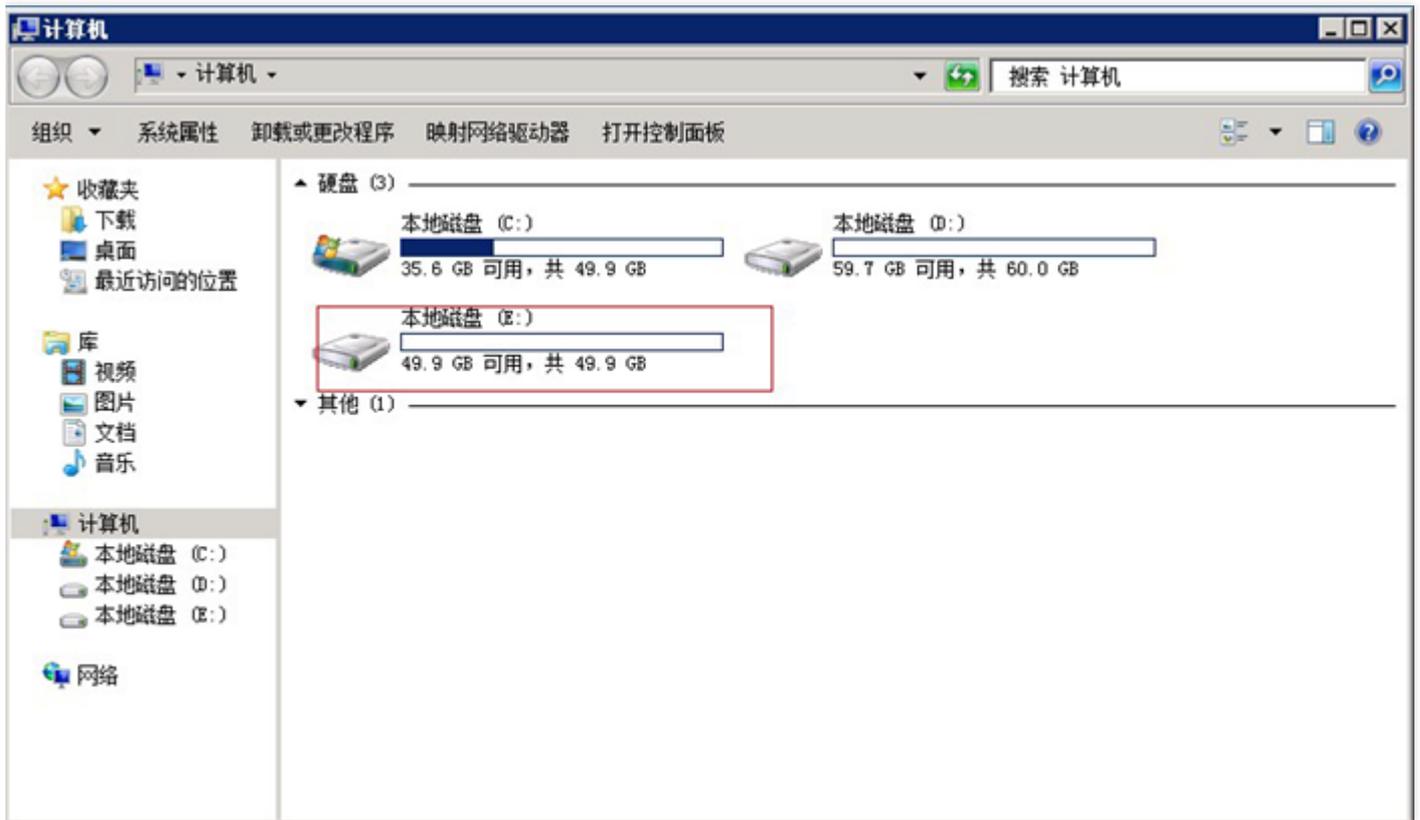
再次确认格式分区的信息无误后，在弹出框中单击【是】按钮：



创建完成后，单击【确认】按钮：

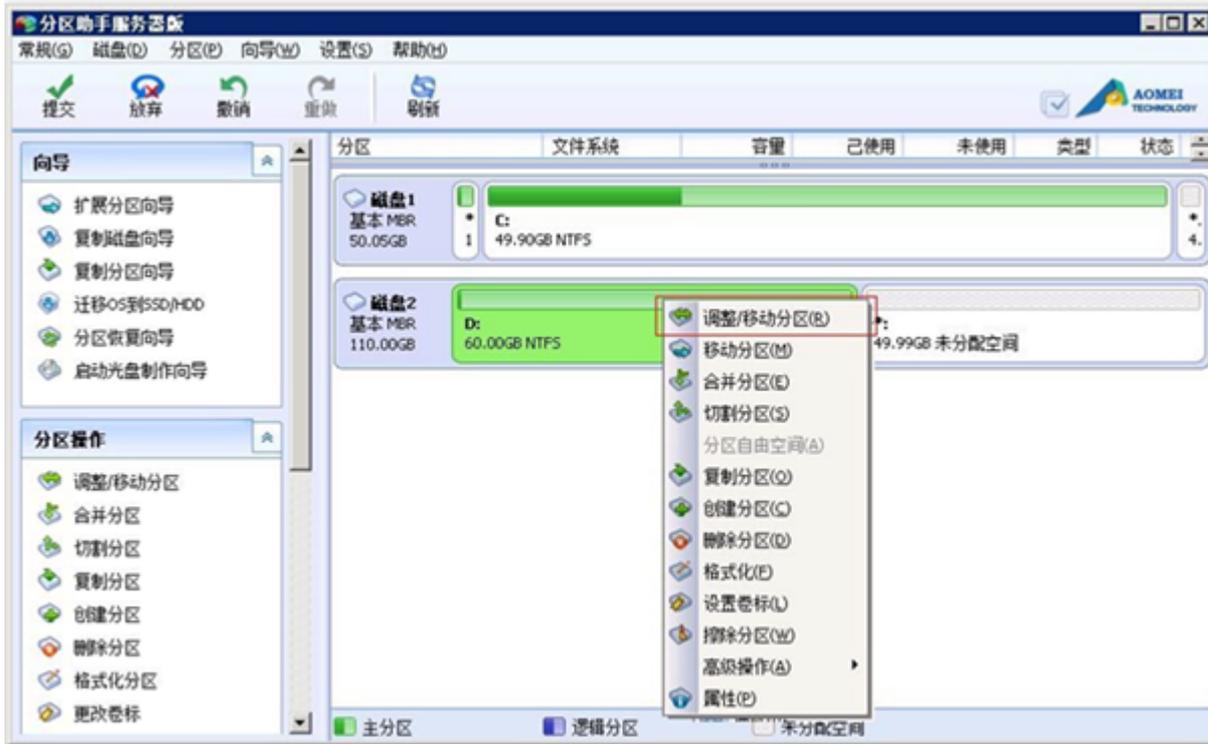


打开【我的电脑】可以看到新创建的磁盘分区(此例中新创建的是E盘)：

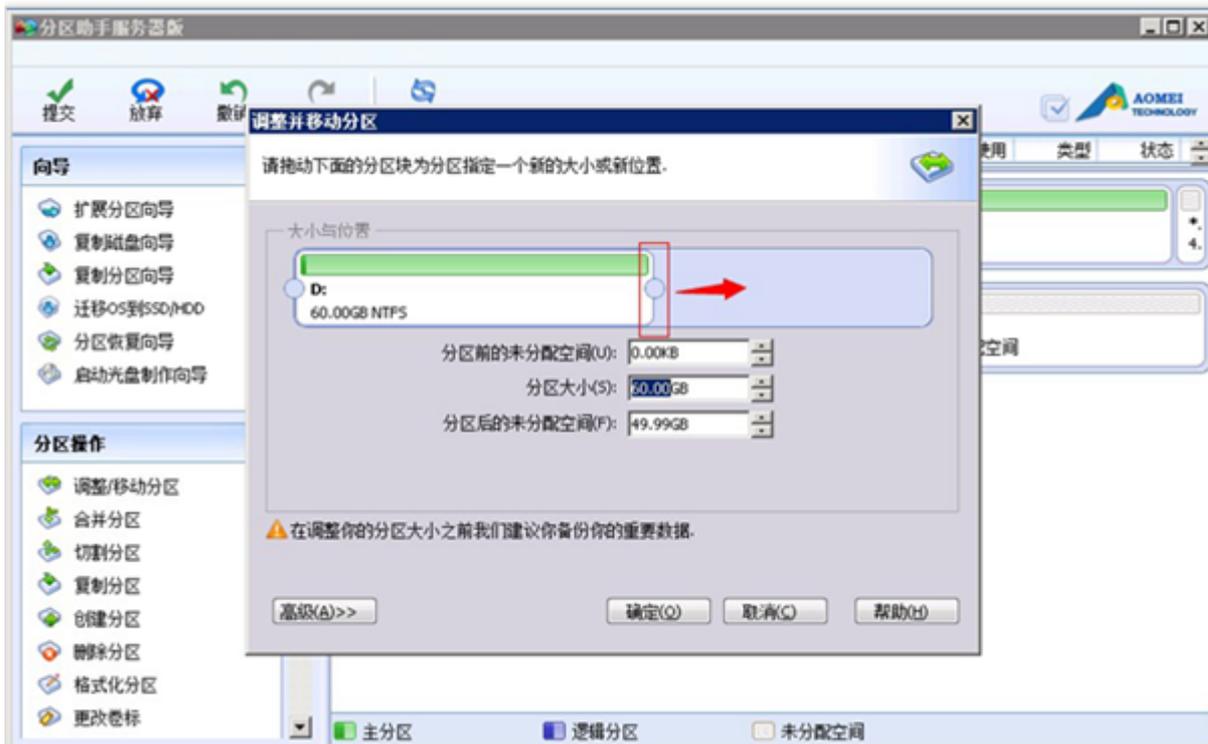


新空间增加到已有分区空间中

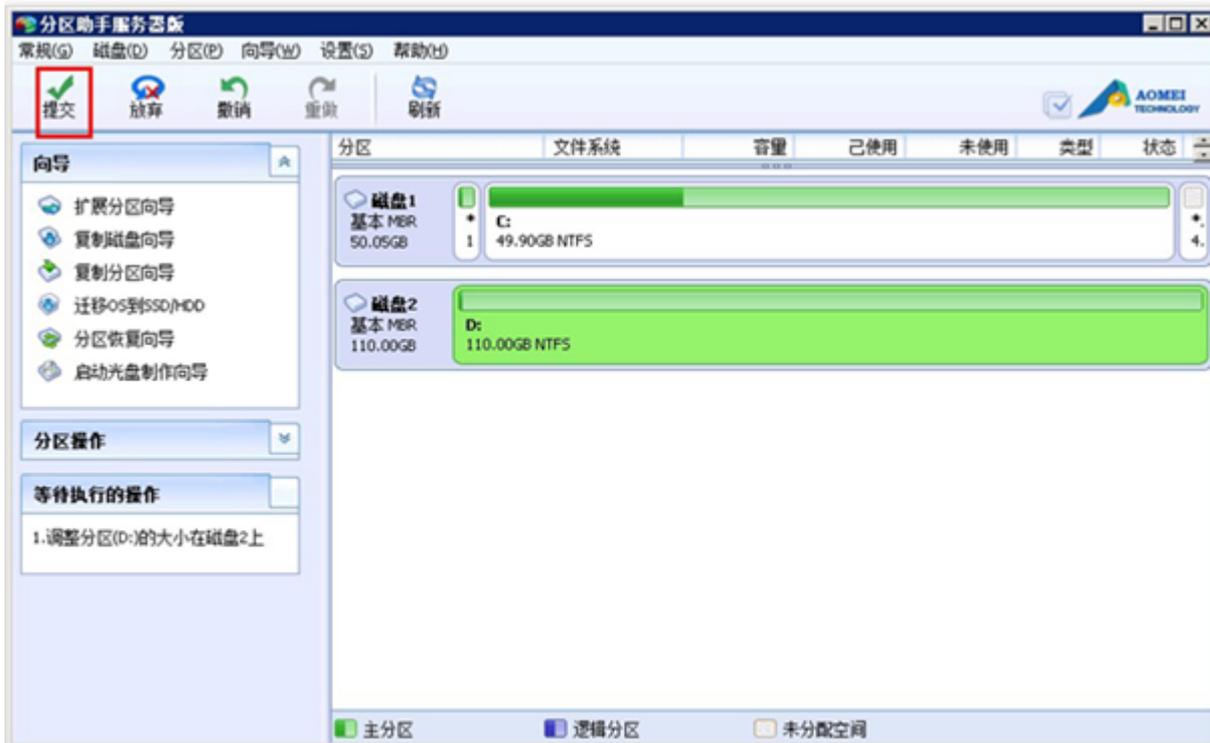
右键单击需要扩容的分区，选择【调整/移动分区(R)】：



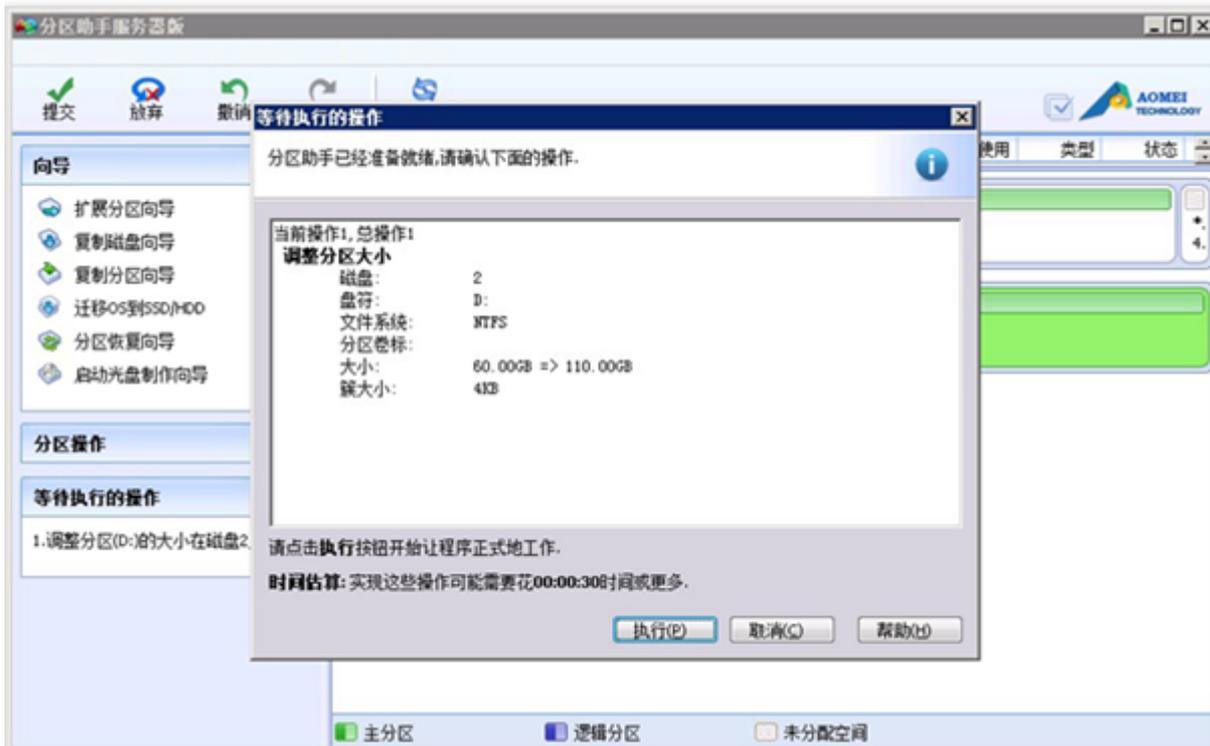
在弹出框中，如图所示，向右拖动小箭头调整分区需要的空间大小，然后单击【确认】按钮：



在左上角单击【提交】按钮：



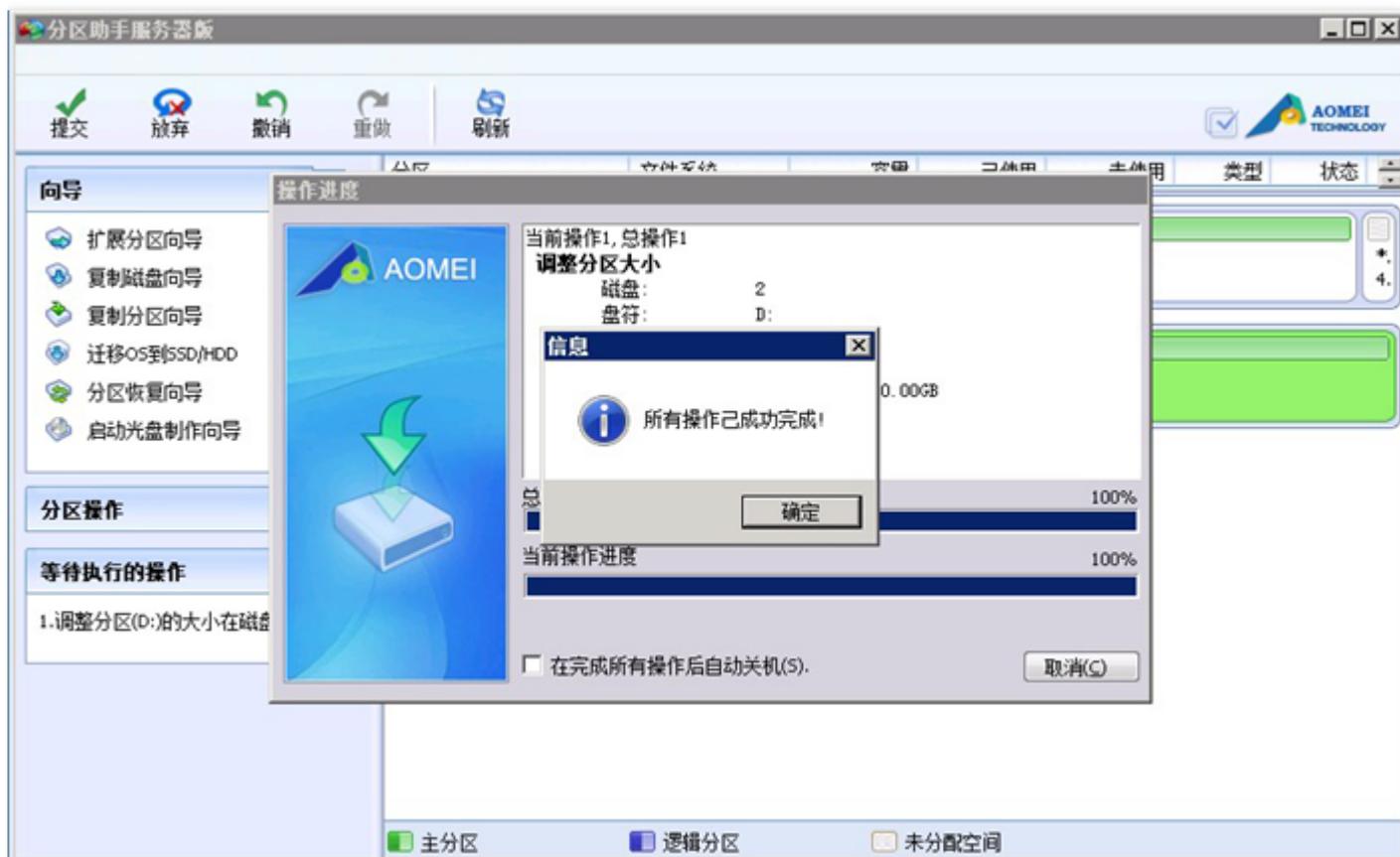
在弹出框中，确认分区扩容信息正确无误后，单击【确认】按钮：



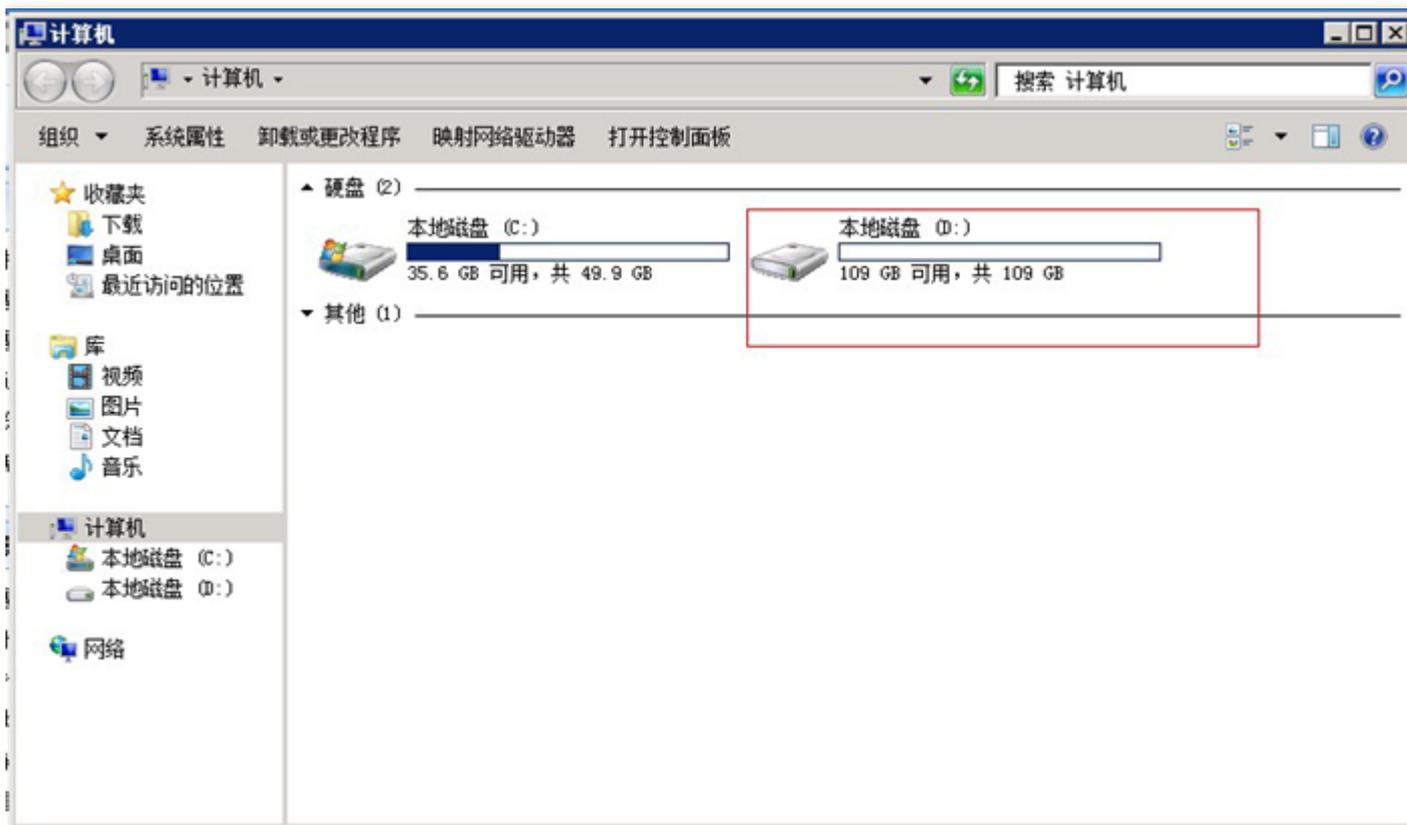
再次确认分区扩容信息后，单击【是】按钮：



等待分区扩容完成后，单击【确定】：



打开【我的电脑】，可以查看到扩容后的分区变化(此例中扩容的是D盘，由60G扩容至109G)：



扩容Linux文件系统

最近更新时间：2018-08-29 11:27:10

云硬盘是云上可扩展的存储设备，用户可以在创建云硬盘后随时扩展其大小，以增加存储空间，同时不失去云硬盘上原有的数据。要达到扩容并使用扩容空间的目的，用户在[扩容实体云硬盘](#)大小之后，还需要扩展其上的文件系统以识别新近可用的空间。您可以根据下面的步骤进行：

1) 扩容实体云硬盘大小

2) 扩容分区

- 确定文件系统分区表形式
- 扩容分区

3) 扩容文件系统

检查文件系统大小

在执行了扩容实体云硬盘大小操作后，用户可以通过检查文件系统大小来查看实例是否识别了更大的云硬盘空间。在Linux上，可以使用 `df -h` 命令检查文件系统大小。

如果没有看到云硬盘大小变成扩容后的值，则需要扩容此文件系统，以便实例可以使用新的空间。

确认分区表形式

使用以下命令确认云硬盘在扩容前使用的分区表形式：

```
fdisk -l
```

若结果如下两图所示（根据操作系统不同略有不同），则说明云服务器扩容前为GPT分区方式，后续操作请参见[GPT分区云硬盘扩容后修改分区指南](#)。

```
Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 7864320 sectors
Units = sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disk label type: dos
Disk identifier: 0x00000000
```

Device	Boot	Start	End	Blocks	Id	System
/dev/vdb1		1	2621439	10485756	ee	GPT

WARNING: GPT (GUID Partition Table) detected on '/dev/vdb'! The util fdisk doesn't support GPT. Use GNU Parted.

若结果如下图所示（根据操作系统不同略有不同），则说明云服务器扩容前为MBR分区方式，后续操作请参见MBR分区云硬盘扩容后修改分区指南。

```
[root@VM_20_49_tlinux ~]# fdisk -l
```

```
Disk /dev/xvda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xf5a25329
```

Device	Boot	Start	End	Blocks	Id	System
/dev/xvda1	*	1	1044	8385898+	83	Linux

```
Disk /dev/xvdc: 107.4 GB, 107374182400 bytes
255 heads, 63 sectors/track, 13054 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x06a4a875
```

Device	Boot	Start	End	Blocks	Id	System
/dev/xvdc1		1	6526	52420063+	83	Linux

GPT分区云硬盘扩容后修改分区指引

新空间格式化成一个独立GPT分区

查看数据盘信息

执行命令 `parted 磁盘路径 print` 命令来确认云硬盘的容量变化。如在过程中收到如下提示,请输入 `Fix` :

```
Error: The backup GPT table is not at the end of the disk, as it should be.
This might mean that another operating system believes the disk is smaller.
Fix, by moving the backup to the end (and removing the old backup)?
Fix/Ignore/Cancel? Fix
Warning: Not all of the space available to /dev/vdb appears to be used, you can
fix the GPT to use all of the space (an extra 188743680 blocks) or continue with
the current setting?
Fix/Ignore? Fix
```

```
[root@VM_164_30_centos ~]# parted /dev/vdb print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 107GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name      Flags
  1      1049kB  10.7GB 10.7GB ext3         primary
```

这里扩容后的云硬盘大小为107GB,已有分区的大小为10.7GB。

卸载已挂载数据盘

执行以下命令确认该云硬盘是否还有分区已挂载:

```
mount | grep '磁盘路径'
```

```
[root@VM_164_30_centos ~]# mount | grep '/dev/vdb'
/dev/vdb1 on /data type ext3 (rw)
```

这里云硬盘上有一个分区(vdb1)挂载在/data上,需要将其解挂。

使用以下命令解挂:

```
umount 挂载点
```

本例中即执行 `umount /data` 进行卸载。

注：要将云硬盘上所有分区文件系统都解挂，如vdb1、vdb2.....

再次使用 `mount | grep '/dev/vdb'` 命令来确认此硬盘上所有分区文件系统都已解挂。

```
[root@VM_164_30_centos ~]# mount | grep '/dev/vdb'
[root@VM_164_30_centos ~]#
```

数据盘分区

确认云硬盘所有分区均已卸载后，执行以下命令新建一个分区：

`parted` 磁盘路径

这里输入 `parted /dev/vdb`。

接下来输入 `print` 来查看分区信息，记住已有分区的End值，以此值作为下一个分区的起始偏移值：

```
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 107GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start  End  Size  File system  Name  Flags
  1      1049kB 10.7GB 10.7GB ext3         primary
```

接下来执行以下命令新建一个主分区，此分区将从已有分区的末尾开始，覆盖硬盘所有的新增空间。：

`mkpart primary start end`

本例使用 `mkpart primary 10.7GB 100%`

再次执行 `print` 可发现新分区已经新建成功，键入 `quit` 即可退出 `parted` 工具：

```
(parted) print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 107GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name      Flags
  1      1049kB  10.7GB  10.7GB  ext3         primary
  2      10.7GB  107GB   96.6GB
```

格式化新建分区

执行以下命令格式化上述新建的分区，用户可以自行决定文件系统的格式，如 `ext2`、`ext3` 等。

```
mkfs.[fstype] [分区路径]
```

这里使用命令 `mkfs.ext3 /dev/vdb2` 对新分区进行格式化，文件系统为 `EXT3`。

新空间增加到已有分区中(GPT分区格式)

查看数据盘信息

执行命令 `parted 磁盘路径 print` 命令来确认云硬盘的容量变化。如在过程中收到如下提示,请输入 `Fix`：

```
Error: The backup GPT table is not at the end of the disk, as it should be.
This might mean that another operating system believes the disk is smaller.
Fix, by moving the backup to the end (and removing the old backup)?
Fix/Ignore/Cancel? Fix
Warning: Not all of the space available to /dev/vdb appears to be used, you can
fix the GPT to use all of the space (an extra 188743680 blocks) or continue with
the current setting?
Fix/Ignore? Fix
```

```
[root@VM_164_30_centos ~]# parted /dev/vdb print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 107GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name      Flags
  1      1049kB  10.7GB 10.7GB  ext3         primary
```

这里扩容后的云硬盘大小为107GB，已有分区的大小为10.7GB。

卸载已挂载数据盘

执行以下命令确认该云硬盘是否还有分区已挂载：

```
mount | grep '磁盘路径'
```

```
[root@VM_164_30_centos ~]# mount | grep '/dev/vdb'
/dev/vdb1 on /data type ext3 (rw)
```

这里云硬盘上有一个分区(vdb1)挂载在/data上，需要将其解挂。

使用以下命令解挂：

```
umount 挂载点
```

本例中即执行 `umount /data` 进行卸载。

注：要将云硬盘上所有分区的文件系统都解挂，如vdb1、vdb2.....

再次使用 `mount | grep '/dev/vdb'` 命令来确认此硬盘上所有分区的文件系统都已解挂。

```
[root@VM_164_30_centos ~]# mount | grep '/dev/vdb'
[root@VM_164_30_centos ~]#
```

数据盘分区

确认云硬盘所有分区均已卸载后，执行以下命令，将原分区删除并以同样的起始偏移新建一个分区：

```
parted [磁盘路径]
```

接下来输入 `unit s`，将显示和操纵单位变成sector（默认为GB），输入 `print` 来查看分区信息，记住已有分区的Start值。删除分区并新建后，Start值必须与这个相同，否则数据将会丢失。

```
(parted) unit s
(parted) print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 209715200s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
```

Number	Start	End	Size	File system	Name	Flags
1	2048s	20969471s	20967424s	ext3	primary	

执行以下命令删除原有分区：

```
rm [分区Number]
```

由上图可知云硬盘上有一个分区，Number号为“1”，执行 `rm 1` 结果如下图：

```
(parted) rm 1
(parted) print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 209715200s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
```

Number	Start	End	Size	File system	Name	Flags
--------	-------	-----	------	-------------	------	-------

输入 `mkpart primary [原分区起始扇区] 100%` 新建一个主分区。本例中使用 `mkpart primary 2048s 100%`，此主分区从第2048个扇区开始（必须与删除之前的分区一致），100%表示此分区到磁盘的最末尾。

如果出现如图状态请输入 `Ignore`：

```
Warning: The resulting partition is not properly aligned for best performance.
Ignore/Cancel? Ignore
```

再次输入 `print` 可发现新分区已经新建成功，输入 `quit` 即可退出 `parted` 工具：

```
(parted) mkpart primary 2048s 100%
(parted) print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 209715200s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
```

Number	Start	End	Size	File system	Name	Flags
1	2048s	209713151s	209711104s	ext3	primary	

检查扩容后分区的文件系统

使用以下命令检查扩容后的分区：

```
e2fsck -f 分区路径
```

前述步骤中本例已新建了分区1，使用 `e2fsck -f /dev/vdb1` 进行操作。结果如下：

```
[root@VM_164_30_centos ~]# e2fsck -f /dev/vdb1
e2fsck 1.41.12 (17-May-2010)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/vdb1: 11/655360 files (0.0% non-contiguous), 79696/2620928 blocks
```

扩容 EXT 文件系统

执行以下命令进行新分区上 EXT 文件系统的扩容操作：

`resize2fs` 分区路径，如/dev/vdb1

```
[root@VM_164_30_centos ~]# resize2fs /dev/vdb1
resize2fs 1.41.12 (17-May-2010)
Resizing the filesystem on /dev/vdb1 to 26213888 (4k) blocks.
The filesystem on /dev/vdb1 is now 26213888 blocks long.
```

扩容 XFS 文件系统

执行以下命令进行新分区上 XFS 文件系统的扩容操作：

`xfs_growfs` 分区路径，如/dev/vdb1

挂载新分区

执行以下命令挂载分区：

`mount` 分区路径 挂载点

这里通过 `mount /dev/vdb1 /data` 命令手动挂载新分区，并使用 `df -h` 命令查看，出现以下信息说明挂载成功，即可以查看到数据盘了。

```
[root@VM_164_30_centos ~]# mount /dev/vdb1 /data
[root@VM_164_30_centos ~]# df -hl
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       7.9G  902M  6.6G  12% /
tmpfs           1.9G   0    1.9G   0% /dev/shm
/dev/vdb1       99G   163M  94G   1% /data
```

MBR分区云硬盘扩容后修改分区指引

MBR分区的云硬盘进行扩容后，您可以选择：

- 将新增的容量空间建立成独立的新分区同时原有分区保持不变
- 扩容原有分区至新增的容量空间（包括未分区直接格式化的场景），并且保持原有分区的数据不丢失。

以上两种场景，在您的Linux云服务器的云硬盘升级成功之后，都可以通过Linux下的分区扩容工具（`fdisk/e2fsck/resize2fs`），执行一系列命令，完成分区扩容，并且保证原数据不会丢失。需要注意的是，不管是添加

新分区还是扩容到已有分区都需要先将此磁盘的所有已挂载分区umount再执行扩容操作，这样内核才能识别出新的分区表。

请注意，由于MBR的限制，选择任何一种方式时，请保持任意分区的大小不超过2TB（若您扩容后的空间已经大于2TB则不可选择第二种方式。

新空间格式化成一个独立分区

查看数据盘信息

执行命令 `df -h` 查看已挂载的数据盘分区信息，以及命令 `fdisk -l` 查看数据盘扩容后未分区的信息：

```
[root@VM_20_49_tlinux ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda1      7.9G  2.1G  5.5G  28% /
tmpfs           4.0G   0  4.0G   0% /dev/shm
/dev/xvdc1      50G   180M  47G   1% /data
[root@VM_20_49_tlinux ~]#
```

```
[root@VM_20_49_tlinux ~]# fdisk -l

Disk /dev/xvda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xf5a25329

   Device Boot      Start         End      Blocks   Id  System
/dev/xvda1 *          1         1044     8385898+   83  Linux

Disk /dev/xvdc: 107.4 GB, 107374182400 bytes
255 heads, 63 sectors/track, 13054 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x06a4a875

   Device Boot      Start         End      Blocks   Id  System
/dev/xvdc1          1         6526     5242063+   83  Linux
```

卸载所有已挂载的分区

执行以下命令卸载所有已挂载的分区：

umount 挂载点

这里使用 `umount /data` 卸载所有已挂载分区。

数据盘分区

确认云硬盘所有分区均已卸载后，执行以下命令新建一个新分区：

fdisk [硬盘路径]

本例使用 `fdisk /dev/xvdc` 命令，按照界面的提示依次输入“p”(查看现有分区信息)、“n”(新建分区)、“p”(新建主分区)、“2”(新建第2个主分区)，两次回车(使用默认配置)，输入“w”(保存分区表)，开始分区：

```
[root@VM_20_49_tlinux /]#
[root@VM_20_49_tlinux /]# fdisk /dev/xvdc

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
switch off the mode (command 'c') and change display units to
sectors (command 'u').

Command (m for help): p

Disk /dev/xvdc: 107.4 GB, 107374182400 bytes
255 heads, 63 sectors/track, 13054 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x06a4a875

   Device Boot      Start         End      Blocks   Id  System
/dev/xvdc1         1           6526     52420063+  83  Linux

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (6527-13054, default 6527):
Using default value 6527
Last cylinder, +cylinders or +size{K,M,G} (6527-13054, default 13054):
Using default value 13054

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@VM_20_49_tlinux /]#
```

这里是以创建1个分区为例，用户也可以根据自己的需求创建多个分区。

查看新分区

使用以下命令查看新分区

```
fdisk -l
```

```
[root@VM_20_49_tlinux /]# fdisk -l

Disk /dev/xvda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xf5a25329

   Device Boot      Start         End      Blocks   Id  System
/dev/xvda1 *          1         1044     8385898+  83  Linux

Disk /dev/xvdc: 107.4 GB, 107374182400 bytes
255 heads, 63 sectors/track, 13054 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x06a4a875

   Device Boot      Start         End      Blocks   Id  System
/dev/xvdc1          1         6526     52420063+  83  Linux
/dev/xvdc2         6527        13054     52436160   83  Linux
```

这里新的分区xvdc2已经创建完成。

格式化新分区并创建文件系统

在进行分区格式化时，用户可以自行决定文件系统的格式，如ext2、ext3等。这里以“ext3”为例，使用命令 `mkfs.ext3 /dev/xvdc2` 对新分区进行格式化。

```
[root@VM_20_49_tlinux /]# mkfs.ext3 /dev/xvdc2
mke2fs 1.41.12 (17-May-2010)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
3278576 inodes, 13109040 blocks
655452 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
401 block groups
32768 blocks per group, 32768 fragments per group
8176 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424

Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 20 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
[root@VM_20_49_tlinux /]#
```

挂载新分区

使用以下命令创建新的挂载点

```
mkdir 新挂载点
```

并执行以下命令挂载新分区到新挂载点上：

```
mount 新分区路径 新挂载点
```

这里使用命令 `mkdir /data1` 创建data1目录，再通过 `mount /dev/xvdc2 /data1` 命令手动挂载新分区后，用 `df -h` 命令查看，出现以下信息说明挂载成功，即可以查看到数据盘了：

```
[root@VM_20_49_tlinux /]# mkdir /data1
[root@VM_20_49_tlinux /]# mount /dev/xvdc2 /data1
[root@VM_20_49_tlinux /]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda1      7.9G  2.1G  5.5G  28% /
tmpfs           4.0G   0  4.0G   0% /dev/shm
/dev/xvdc1       50G  180M   47G   1% /data
/dev/xvdc2       50G  180M   47G   1% /data1
[root@VM_20_49_tlinux /]#
```

添加新分区信息

如果希望云服务器在重启或开机时能自动挂载数据盘，必须将分区信息添加到/etc/fstab中。如果没有添加，则云服务器重启或开机后都不能自动挂载数据盘。

执行以下命令添加信息：

```
echo '/dev/xvdc2 /data1 ext3 defaults 0 0' >> /etc/fstab
```

执行 `cat /etc/fstab` 命令查看，出现以下信息表示添加分区信息成功：

```
[root@VM_20_49_tlinux /]# echo '/dev/xvdc2 /data1 ext3 defaults 0 0' >> /etc/fstab
[root@VM_20_49_tlinux /]# cat /etc/fstab
/dev/xvda1 / ext3 noatime,acl,user_xattr 1 1
/dev/xvdb swap swap defaults 0 0
/dev/xvdc1 /data ext3 noatime,acl,user_xattr 1 2
tmpfs /dev/shm tmpfs defaults 0 0
proc /proc proc defaults 0 0
sysfs /sys sysfs noauto 0 0
debugfs /sys/kernel/debug debugfs noauto 0 0
devpts /dev/pts devpts mode=0620,gid=5 0 0
/dev/xvdc2 /data1 ext3 defaults 0 0
[root@VM_20_49_tlinux /]#
```

将新空间增加到已有分区空间中

若原有的硬盘分区为一个MBR分区(可以看到vdb1,vdc1等字样)，同时在此分区上制作了文件系统。或原有的硬盘没有分区，直接在此硬盘上制作了文件系统。这两种情况都可以选择使用自动扩容工具进行扩容。

自动扩容工具适用于Linux操作系统，用于将扩容时新扩的云硬盘存储空间添加到已存在的文件系统中，扩容能够成功必须满足下面3个条件：

- 文件系统是ext2/ext3/ext4
- 当前文件系统不能有错误
- 扩容后的磁盘大小不超过2TB

下面介绍自动扩容工具的使用方法。

卸载正在使用的硬盘分区

执行以下命令卸载分区：

`umount` 挂载点

```
[root@VM_136_143 ~]# umount /data/
[root@VM_136_143 ~]# df
Filesystem      1K-blocks      Used Available Use% Mounted on
/dev/vda1        8254240    6832388    1002560   88% /
[root@VM_136_143 ~]#
```

下载一键扩容工具

执行以下命令下载工具：

```
wget -O /tmp/devresize.py https://raw.githubusercontent.com/tencentyun/tencentcloud-cbs-tools/master/devresize/devresize.py
```

执行扩容工具

执行以下命令进行扩容：

```
python /tmp/devresize.py 硬盘路径
```

请注意，这里硬盘路径是需要扩容的云硬盘，而不是分区名。若您的文件系统在vdb1上，则应执行 `python /tmp/devresize.py /dev/vdb`

```
[root@VM_136_143 ~]# python /tmp/devresize.py /dev/vdb
It will resize (/dev/vdb1).
This operation may take from several minutes to several hours, continue? [Y/n]
Y
Backup MBR to /tmp/MBR_2015-05-22_16:56:18_bak
e2fsck 1.41.12 (17-May-2010)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/vdb1: 591/656640 files (0.0% non-contiguous), 87719/2621440 blocks
resize2fs 1.41.12 (17-May-2010)
Resizing the filesystem on /dev/vdb1 to 13107198 (4k) blocks.
The filesystem on /dev/vdb1 is now 13107198 blocks long.
```

若输出“The filesystem on /dev/vdb1 is now XXXXX blocks long.”则表示扩容成功。

若输出的是“[ERROR] - e2fsck failed!!”，请先用fsck对文件系统所在分区进行修复，可以执行以下命令进行自动修复：

```
fsck -a 分区路径
```

请注意这里与前一个命令不同，需要填写的是文件系统所在分区。若您的文件系统在vdb1上，则应执行 `fsck -a /dev/vdb1`。

修复成功后，再使用 `python /tmp/devresize.py 硬盘路径` 来使用扩容工具进行扩容。

重新挂载扩容后的分区

执行以下命令挂载扩容后的分区：

mount 分区路径 挂载点

并通过以下命令查看扩容后的分区容量：

```
df -h
```

这里通过 `mount /dev/vdb1 /data` 命令手动挂载扩容后的分区(如果原先是没有分区的，执行 `mount /dev/vdb /data`)，用 `df -h` 命令查看，出现以下信息说明挂载成功，即可以查看到数据盘了：

```
[root@VM_136_143 ~]# mount /dev/vdb1 /data/
[root@VM_136_143 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       7.9G  6.6G  930M  88% /
/dev/vdb1       50G   2.2G  45G   5% /data
[root@VM_136_143 ~]# ll /data/
total 2050024
drwx----- 2 root root      16384 May 25 17:45 lost+found
-rw-r--r--  1 root root 1048576000 May 25 17:46 test1.txt
-rw-r--r--  1 root root 1048576000 May 25 17:46 test2.txt
```

再执行 `ll /data` 命令，可以查看到，扩容后原分区的数据没有丢失，新增加的存储空间已经扩容到文件系统中。

续费云硬盘

最近更新时间：2018-08-27 09:59:27

及时对云硬盘续费可以有效避免因到期引起的数据丢失，建议用户对重要的数据设置定期提醒。

续费弹性云盘

使用控制台续费弹性云盘

在弹性云盘到期前，您可对其进行续费，以防止到期后硬盘卸载、无法读写的情况：

- 1) 打开[云硬盘控制台](#)。
- 2) 对需要续费的弹性云盘，在右侧操作栏中，单击【续费】。
- 3) 在续费弹出框中，选择需要续费的时间，单击【确定】。
- 4) 进行支付后即可对弹性云盘续费。

使用 API 续费弹性云盘

用户可以使用 RenewCbsStorages 接口续费弹性云盘，具体内容请参考[续费弹性云盘接口](#)。

续费非弹性云硬盘

使用控制台续费非弹性云硬盘

非弹性云硬盘跟随云服务器实例的生命周期，如果您需要对其进行续费，请直接[续费云服务器实例](#)。

使用 API 续费非弹性云硬盘

非弹性云硬盘跟随云服务器实例的生命周期，用户可以使用 RenewInstance 接口续费非弹性云盘，具体内容请参考[续费实例（包年包月）接口](#)。

卸载云硬盘

最近更新时间：2018-09-20 15:23:01

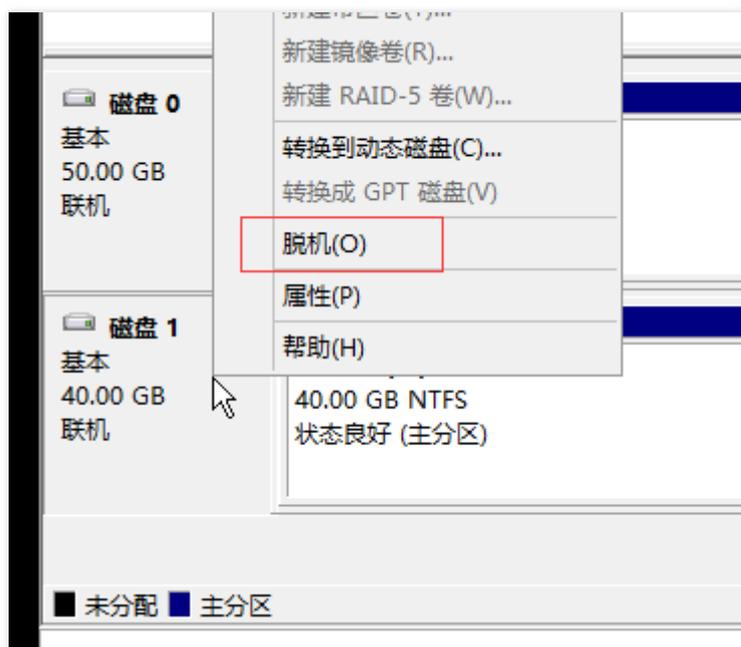
当您需要将 **弹性云盘** 挂载到另一台实例上使用时，您可以主动地从实例断开该弹性云盘，并将其连接到其他实例上。**卸载弹性云盘并不会清除该硬盘上的数据。**

使用控制台卸载云硬盘

目前支持对作为数据盘的普通弹性云盘进行卸载，不可卸载系统盘。

卸载数据盘时，请确保您了解以下事项：

- 在 Windows 操作系统下，为了保证数据完整性，建议您暂停对该磁盘的所有文件系统的读写操作，否则未完成读写的数据会丢失。**解挂弹性云盘时需要先将磁盘设为脱机状态，否则在不重启云服务器器的情况下，您可能将无法再次挂载弹性云盘。**



- 在 Linux 操作系统下，您需要先登录实例，并对需要卸载的弹性云硬盘进行 `umount` 操作，命令执行成功后再进入控制台对磁盘进行卸载操作。若未进行 `umount` 操作直接被强制解挂后，关机时和开机时可能会出现以下问题：

```
Checking filesystems
/dev/vda1: clean, 35630/524288 files, 335690/2096474 blocks
fsck.ext3: Unable to resolve 'UUID=dabe8ee8-221b-44c7-9074-4d3f8fc4ae44'
fsck.ext3: No such file or directory while trying to open /dev/disk/by-id/virtio
-disk-ezy5q5l6-part5
/dev/disk/by-id/virtio-disk-ezy5q5l6-part5:
The superblock could not be read or does not describe a correct ext2
filesystem.  If the device is valid and it really contains an ext2
filesystem (and not swap or ufs or something else), then the superblock
is corrupt, and you might try running e2fsck with an alternate superblock:
    e2fsck -b 8193 <device>

                                     [FAILED]

*** An error occurred during the file system check.
*** Dropping you to a shell; the system will reboot
*** when you leave the shell.
Give root password for maintenance
(or type Control-D to continue):
```

```
Checking filesystems
/dev/vda1: clean, 35630/524288 files, 335690/2096474 blocks
fsck.ext3: Unable to resolve 'UUID=dabe8ee8-221b-44c7-9074-4d3f8fc4ae44'
fsck.ext3: No such file or directory while trying to open /dev/disk/by-id/virtio
-disk-ezy5q5l6-part5
/dev/disk/by-id/virtio-disk-ezy5q5l6-part5:
The superblock could not be read or does not describe a correct ext2
filesystem.  If the device is valid and it really contains an ext2
filesystem (and not swap or ufs or something else), then the superblock
is corrupt, and you might try running e2fsck with an alternate superblock:
    e2fsck -b 8193 <device>

                                     [FAILED]

*** An error occurred during the file system check.
*** Dropping you to a shell; the system will reboot
*** when you leave the shell.
Give root password for maintenance
(or type Control-D to continue):
```

- 1) 登录 [腾讯云控制台](#)。
- 2) 进入【云服务器】-【云硬盘】选项卡。
- 3) 在云硬盘列表页，单击状态为**已挂载**、**支持挂载/卸载**的云硬盘后的【更多】-【卸载】按钮进行单盘卸载；或在云硬盘列表页，勾选状态为**已挂载**、**支持挂载/卸载**状态的云硬盘，单击顶部【卸载】按钮进行批量卸载。
- 4) 在弹出的对话框中确认警告事项，单击【确认】按钮。

使用 API 卸载云硬盘

用户可以使用 DetachDisks 接口卸载云硬盘，具体内容请参考 [解挂云硬盘接口](#)。

删除云硬盘

最近更新时间：2018-06-22 17:02:52

由于云硬盘的生命周期根据类型不同而有差异：弹性云盘的生命周期独立于云服务器实例，因此可以独立于 CVM 实例而销毁；非弹性云硬盘的生命周期跟随云服务器，只能在云服务器销毁时被删除。

删除弹性云盘

弹性云盘的生命周期独立于云服务器实例。当前包年包月类型的弹性云盘生命周期结束于其到期一段时间后被系统自动销毁。弹性云盘将在到期当日自动解除与实例的关联关系并进入回收站并保留7个自然日，期间您可选择续费。7个自然日后仍未续费则这块云硬盘将被销毁，数据将彻底丢失。同时，在到期前您也可以为其进行续费，以防止到期后销毁导致的数据丢失。

弹性云盘的回收

腾讯云回收站是一种云服务的回收机制及系统，弹性云盘到期后会进入回收站保留一段时间，用户可在回收站中找到已经超过使用期的部分云服务并对其进行续费。避免了云服务直接被系统清除导致的数据丢失等重大损失。

若您的弹性云盘在到期前（包括到期当天）未进行续费，系统将在到期时间点开始将其从挂载的云服务器实例卸载，保留数据并进入回收站。在进入回收站的7个工作日内，您仍可以在回收站对这块云盘进行续费找回。若在这段时间内该弹性云盘未被续费，系统将该资源释放，**数据将被清除且不可恢复**。

进入回收站内的弹性云盘，在销毁前用户可以对回收对象的操作，仅为**续费恢复**。

恢复弹性云盘

弹性云盘到期后的七天内，您可进入回收站对其续费恢复：

打开[回收站控制台](#)，找到列表内，您要恢复的弹性云盘，单击操作按钮【恢复】，完成续费支付即可到[云硬盘控制台](#)找到刚刚恢复的资源。

或对多块已经过期的弹性云盘续费：

打开[回收站控制台](#)，找到列表内，您要恢复的弹性云盘，勾选您要恢复的资源，单击【批量恢复】操作按钮，完成续费支付即可到[云硬盘控制台](#)找到刚刚恢复的资源。

删除非弹性云盘

非弹性云盘的生命周期跟随创建的云服务器实例，因此非弹性云盘在挂载的实例销毁时随之销毁，有关实例销毁的更多内容，可以参考[包年包月实例到期](#)和[销毁按量计费实例](#)。