

阿里云 云服务器 ECS

块存储

文档版本：20190613

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 禁止： 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告： 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明： 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定 。
<code>courier</code> 字体	命令。	执行 <code>cd /d C:/windows</code> 命令，进入Windows系统文件夹。
##	表示参数、变量。	<code>bae log list --instanceid Instance_ID</code>
[]或者[a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ }或者{a b}	表示必选项，至多选择一个。	<code>swich {stand slave}</code>

目录

法律声明.....	I
通用约定.....	I
1 什么是块存储.....	1
2 块存储性能.....	2
3 云盘.....	9
3.1 云盘和共享块存储.....	9
3.2 云盘三副本技术.....	11
3.3 ECS 云盘加密.....	12
3.4 创建云盘.....	15
3.4.1 创建按量付费云盘.....	15
3.4.2 创建预付费云盘.....	17
3.4.3 使用快照创建云盘.....	20
3.5 挂载云盘.....	22
3.6 分区格式化数据盘.....	25
3.6.1 Windows格式化数据盘.....	25
3.6.2 Linux格式化数据盘.....	26
3.6.3 分区格式化大于2 TiB云盘.....	30
3.7 扩容云盘.....	37
3.7.1 扩容概述.....	37
3.7.2 扩容云盘容量.....	39
3.7.3 扩展分区和文件系统_Windows.....	40
3.7.4 扩展分区与文件系统_Linux系统盘.....	45
3.7.5 扩展分区与文件系统_Linux数据盘.....	48
3.8 更换系统盘.....	56
3.8.1 更换系统盘（非公共镜像）.....	56
3.8.2 更换系统盘（公共镜像）.....	60
3.9 回滚云盘.....	65
3.10 重新初始化云盘.....	67
3.11 转换云盘的计费方式.....	73
3.12 查看云盘监控信息.....	73
3.13 卸载数据盘.....	75
3.14 释放云盘.....	78
4 共享块存储.....	80
5 本地盘.....	82
6 块存储FAQ.....	87

1 什么是块存储

概念

块存储是阿里云为云服务器ECS提供的块设备类型产品，具备高性能、低时延等特性。您可以像使用物理硬盘一样格式化并建立文件系统来使用块存储，可满足绝大部分通用业务场景下的数据存储需求。

阿里云为您的云服务器ECS提供了丰富的块存储产品类型，包括基于分布式存储架构的云盘、共享块存储产品，以及基于物理机本地硬盘的本地盘产品。其中：

- **云盘**，是阿里云为云服务器ECS提供的数据块级别的块存储产品，采用 **多副本的分布式机制**，具有低时延、高性能、持久性、高可靠等性能，可以随时创建或释放，也可以随时扩容。
- **共享块存储**，是一种支持多个ECS实例并发读写访问的数据块级存储设备，与云盘类似，采用 **多副本的分布式机制**，具有支持多实例并发访问、低时延、高性能、高可靠等特性，适用于shared-everything架构下对块存储设备的共享访问场景。
- **本地盘**，是指基于云服务器ECS所在物理机（宿主机）上的本地硬盘设备，为对存储I/O性能、海量存储性价比有极高要求的业务场景而设计的产品类型，为实例提供本地存储访问能力，根据具体产品类型的不同，具有低时延、高随机IOPS、高吞吐量、高性价比等产品能力。

不同块存储的性能，请参考 **块存储性能**。

块存储、对象存储、文件存储的区别

阿里云目前主要提供三种数据存储产品，分别是块存储、**对象存储#OSS#** 和 **文件存储#NAS#**。

三者区别如下：

- **块存储**：是阿里云为云服务器ECS提供的块设备，高性能、低时延，满足随机读写，您可以像使用物理硬盘一样格式化并建立文件系统来使用块存储。块存储可用于绝大部分通用业务场景下的数据存储。
- **对象存储（OSS）**：可以理解为一个海量的存储空间，适合存储互联网上产生的图片、短视频、音频等海量非结构化数据。您可以通过API在任何时间、任何地点访问对象存储里的数据。常用于互联网业务网站搭建、动静资源分离、CDN加速等业务场景。
- **文件存储（NAS）**：类似于对象存储，适合存储非结构化的海量数据。但是您需要通过标准的文件访问协议访问这些数据，比如Linux系统需要使用Network File System（NFS）协议，Windows系统需要使用SMB（又称为CIFS）协议。您可以通过设置权限让不同的客户端同时访问同一份文件。文件存储适合企业部门间文件共享、广电非线性编、高性能计算、容器服务等业务场景。

2 块存储性能

本文介绍云盘、共享块存储和本地盘的性能衡量指标，以及性能对比和性能测试方式。

性能指标

衡量块存储产品的性能指标主要包括：IOPS、吞吐量和访问时延。

· IOPS

IOPS（Input/Output Operations per Second）是指每秒能处理的I/O个数，表示块存储处理读写（输出/输入）的能力。如果要部署事务密集型应用，比如数据库类业务应用等典型场景，需要关注IOPS性能。

常用的IOPS性能指标是随机操作和顺序操作，如下表所示。

IOPS性能指标	描述	
总IOPS	每秒执行的I/O操作总次数。	
随机读IOPS	每秒执行的随机读I/O操作的平均次数	对硬盘存储位置的不连续访问。
随机写IOPS	每秒执行的随机写I/O操作的平均次数	
顺序读IOPS	每秒执行的顺序读I/O操作的平均次数	对硬盘存储位置的连续访问。
顺序写IOPS	每秒执行的顺序写I/O操作的平均次数	

· 吞吐量

吞吐量是指单位时间内可以成功传输的数据数量。

如果要部署大量顺序读写的应用，比如Hadoop离线计算型业务等典型场景，需要关注吞吐量。

· 访问时延

访问时延是指块存储处理一个I/O需要的时间。

如果您的应用对时延比较敏感，比如数据库（过高的时延会导致应用性能下降或报错），建议您使用ESSD云盘、SSD云盘、SSD共享块存储或本地SSD盘类产品。

如果您的应用更偏重存储吞吐能力，对时延相对不太敏感，比如Hadoop离线计算等吞吐密集型应用，建议您使用本地HDD盘类产品，如d1或d1ne大数据型实例。

性能对比

块存储按照二进制单位计算，单位为GiB、KiB、TiB或者MiB。



说明:

二进制单位用于表示1024进位的数据大小。例如, 1 GiB = 1024 MiB。

- 云盘性能

四种云盘的性能对比如下表所示。

参数	ESSD云盘	SSD云盘	高效云盘	普通云盘***
单盘最大容量	32768 GiB	32768 GiB	32768 GiB	2000 GiB
最大IOPS	1000000	25000*	5000	数百
最大吞吐量	4000 MBps	300 MBps*	140 MBps	30–40 MBps
单盘性能计算公式**	$IOPS = \min\{1800 + 50 * \text{容量}, 1000000\}$	$IOPS = \min\{1800 + 30 * \text{容量}, 25000\}$	$IOPS = \min\{1800 + 8 * \text{容量}, 5000\}$	无
	$\text{吞吐量} = \min\{120 + 0.5 * \text{容量}, 4000\} \text{ MBps}$	$\text{吞吐量} = \min\{120 + 0.5 * \text{容量}, 300\} \text{ MBps}$	$\text{吞吐量} = \min\{100 + 0.15 * \text{容量}, 140\} \text{ MBps}$	无
数据可靠性	99.9999999%	99.9999999%	99.9999999%	99.9999999%
API名称	cloud_essd	cloud_ssd	cloud_efficiency	cloud

参数	ESSD云盘	SSD云盘	高效云盘	普通云盘***
典型应用场景	- OLTP数据库：如MySQL、PostgreSQL、Oracle、SQL Server等关系型数据库 - NoSQL数据库：如MongoDB、HBase、Cassandra等非关系型数据库 - ElasticSearch分布式日志：ELK（Elasticsearch、Logstash和Kibana）日志分析等	- PostgreSQL、MySQL、Oracle、SQL Server等中大型关系数据库应用 - 对数据可靠性要求高的中大型开发测试环境	对数据可靠性要求高、中度性能要求的中小型开发测试应用	- 数据不被经常访问或者低I/O负载的应用场景（如果应用需要更高的I/O性能，建议使用SSD云盘） - 需要低成本并且有随机读写I/O的应用环境

* SSD云盘的性能因数据块大小而异，数据块越小，吞吐量越小，IOPS越高，如下表所示。只有挂载到I/O优化的实例时，SSD云盘才能获得期望的IOPS性能。挂载到非I/O优化的实例时，SSD云盘无法获得期望的IOPS性能。

数据块大小	IOPS最大值	吞吐量
4 KiB	约25000	约100 MBps
16 KiB	约17200	约260 MBps
32 KiB	约9600	约300 MBps

数据块大小	IOPS最大值	吞吐量
64 KiB	约4800	约300 MBps

** 单盘性能计算公式说明：

- 以单块SSD云盘最大IOPS计算公式为例说明：起步1800 IOPS，每GiB增加30 IOPS，最高25000 IOPS。
- 以单块SSD云盘最大吞吐量计算公式为例说明：起步120 MBps，每GiB增加0.5 MBps，上限为 300 MBps的吞吐量。

不同云盘的单路随机写访问时延如下：

- ESSD云盘：0.2 ms
- SSD云盘：0.5-2 ms
- 高效云盘：1-3 ms
- 普通云盘：5-10 ms

*** 普通云盘属于上一代云盘产品，目前已经逐步停止售卖。请根据业务实际需求，选择高效云盘、SSD云盘等其他类型的云盘产品。

· 共享块存储性能

2种共享块存储的性能对比如下表所示。

参数	SSD共享块存储	高效共享块存储
最大容量	- 单盘：32768 GiB - 单个实例：最大128 TiB	- 单盘：32768 GiB - 单个实例：最大128 TiB
最大随机读写IOPS**	30000	5000
最大顺序读写吞吐量**	512 MBps	160 MBps
单盘性能计算公式**	$IOPS = \min\{1600 + 40 * \text{容量}, 30000\}$	$IOPS = \min\{1000 + 6 * \text{容量}, 5000\}$
	$\text{吞吐量} = \min\{100 + 0.5 * \text{容量}, 512\} \text{ MBps}$	$\text{吞吐量} = \min\{50 + 0.15 * \text{容量}, 160\} \text{ MBps}$

参数	SSD共享块存储	高效共享块存储
典型应用场景	- Oracle RAC - SQL Server - 故障转移集群 - 服务器高可用	服务器高可用架构

* 最大IOPS和吞吐量是在2个或2个以上实例同时压测裸设备能达到的性能数值。

** 单盘性能计算公式说明：

- 以单块SSD共享块存储最大IOPS计算公式为例：起步1600 IOPS，每GiB增加40 IOPS，最高30000 IOPS。
- 以单块SSD共享块存储最大吞吐量计算公式为例：起步100 MBps，每GiB增加0.5 MBps，上限为512 MBps的吞吐量。

不同共享块存储的单路访问时延如下：

- SSD共享块存储：0.5–2 ms
- 高效共享块存储：1–3 ms
- 本地盘性能

本地盘的性能信息，请参见[本地盘](#)。

性能测试

Linux实例和Windows实例都推荐使用FIO工具测试块存储性能。



说明：

您也可以使用其他工具测试块存储性能，但不同工具测试出来的硬盘基准性能会有差异，如dd、sysbench、iometer等工具可能会受到测试参数配置和文件系统影响，难以反映真实的磁盘性能。

本文以Linux实例和FIO为例，说明如何使用FIO测试块存储性能。在进行测试前，请确保块存储设备已经4 KiB对齐。



警告：

测试裸盘可以获得真实的块存储盘性能，但直接测试裸盘会破坏文件系统结构，请在测试前提前做好数据备份。建议您只在新购无数据的ECS实例上使用工具测试块存储性能，避免造成数据丢失。

- 测试随机写IOPS，运行以下命令：

```
fio -direct=1 -iodepth=128 -rw=randwrite -ioengine=libaio -bs=4k -
size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -
name=Rand_Write_Testing
```

- 测试随机读IOPS，运行以下命令：

```
fio -direct=1 -iodepth=128 -rw=randread -ioengine=libaio -bs=4k -
size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -
name=Rand_Read_Testing
```

- 测试顺序写吞吐量，运行以下命令：

```
fio -direct=1 -iodepth=64 -rw=write -ioengine=libaio -bs=1024k -size
=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name
=Write_PPS_Testing
```

- 测试顺序读吞吐量，运行以下命令：

```
fio -direct=1 -iodepth=64 -rw=read -ioengine=libaio -bs=1024k -size=
1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=
Read_PPS_Testing
```

下表以测试随机写IOPS的命令为例，说明命令中各种参数的含义。

参数	说明
-direct=1	表示测试时忽略I/O缓存，数据直写。
-iodepth=128	表示使用AIO时，同时发出I/O数的上限为128。
-rw=randwrite	表示测试时的读写策略为随机写（random writes）。作其它测试时可以设置为： · randread（随机读random reads） · read（顺序读sequential reads） · write（顺序写sequential writes） · randrw（混合随机读写mixed random reads and writes）
-ioengine=libaio	表示测试方式为libaio（Linux AIO，异步I/O）。应用程序使用I/O通常有两种方式： · 同步 同步的I/O一次只能发出一个I/O请求，等待内核完成才返回。这样对于单个线程iodepth总是小于1，但是可以透过多个线程并发执行来解决。通常会用16-32根线程同时工作将iodepth塞满。 · 异步 异步的I/O通常使用libaio这样的方式一次提交一批I/O请求，然后等待一批的完成，减少交互的次数，会更有效率。

参数	说明
-bs=4k	表示单次I/O的块文件大小为4 KB。未指定该参数时的默认大小也是4 KB。 测试IOPS时，建议将bs设置为一个比较小的值，如本示例中的4k。 测试吞吐量时，建议将bs设置为一个较大的值，如本示例中的1024k。
-size=1G	表示测试文件大小为1 GiB。
-numjobs=1	表示测试线程数为1。
-runtime=1000	表示测试时间为1000秒。如果未配置，则持续将前述-size指定大小的文件，以每次-bs值为分块大小写完。
-group_reporting	表示测试结果里汇总每个进程的统计信息，而非以不同job汇总展示信息。
-filename=iotest	指定测试文件的名称，比如iotest。测试裸盘可以获得真实的硬盘性能，但直接测试裸盘会破坏文件系统结构，请在测试前提前做好数据备份。
-name=Rand_Write_Testing	表示测试任务名称为Rand_Write_Testing，可以随意设定。

3 云盘

3.1 云盘和共享块存储

云盘和共享块存储，是阿里云为云服务器ECS提供的数据块级别的块存储产品，具有低时延、高性能、持久性、高可靠等特点，采用 [分布式三副本机制](#)，为ECS实例提供99.9999999%的数据可靠性保证。支持在可用区内自动复制您的数据，防止意外硬件故障导致的数据不可用，保护您的业务免于组件故障的威胁。就像硬盘一样，您可以对挂载到ECS实例上的云盘和共享块存储做分区、格式化、创建文件系统等操作，并对数据进行持久化存储。

您可以根据业务需要随时扩容云盘和共享块存储。具体操作，请参见 [扩容数据盘](#) 和 [扩容云盘容量](#)。您也可以为云盘和共享块存储创建快照，备份数据。关于快照的更多信息，请参见 [#unique_18](#)。

云盘和共享块存储的主要区别，在于是否可同时挂载到多台ECS实例上进行读写访问。区别如下：

- 云盘：一块云盘只能挂载到同一地域、同一可用区的一台ECS实例。
- 共享块存储：一块共享块存储最多可以同时挂载到同一地域、同一可用区的8台ECS实例。



说明：

共享块存储目前处于公测阶段，您可以 [提交工单](#) 申请公测资格。

云盘

- 根据性能分类
 - ESSD云盘：阿里云全新推出的超高性能云盘产品。基于新一代分布式块存储架构，结合25GE网络和RDMA技术，为您提供单盘高达100万的随机读写能力和更低的单路时延能力。ESSD云盘当前处于公测阶段，更多信息，请参见 [ESSD云盘FAQ](#)。
 - SSD云盘：具备稳定的高随机读写性能、高可靠性的高性能云盘产品。
 - 高效云盘：具备高性价比、中等随机读写性能、高可靠性的云盘产品。
 - 普通云盘：具备高可靠性、一般随机读写性能的云盘产品。

- 根据用途分类

- 系统盘：生命周期与系统盘所挂载的ECS实例相同，随实例一起创建和释放。不可共享访问。系统盘可选的容量范围与实例所选的镜像有关：

- Linux（不包括CoreOS）+ FreeBSD：20 GiB ~ 500 GiB

- CoreOS：30 GiB ~ 500 GiB

- Windows：40 GiB ~ 500 GiB

- 数据盘：可以与ECS实例同时创建，也可以 [单独创建](#)，不可共享访问。与ECS实例同时创建的数据盘，生命同期与实例相同，随实例一起创建和释放。单独创建的数据盘，可以 [单独释放](#)，也可以设置为随ECS实例一起释放。数据盘的性能由云盘类型决定，详细信息，请参见 [块存储性能](#)。

作数据盘用时，云盘与共享块存储共享数据盘配额，即一台实例最多挂载16块数据盘。

共享块存储

共享块存储是一种支持多台ECS实例并发读写访问的数据块级存储设备，具备多并发、高性能、高可靠等特性，数据可靠性可以达到 99.9999999%。单块共享块存储最多可以同时挂载到8台ECS实例。

共享块存储只能作数据盘用，只能单独创建，可以共享访问。您可以设置共享块存储与挂载的ECS实例一起释放。

根据性能不同，共享块存储可以分为：

- SSD共享块存储：采用固态硬盘作为存储介质，能够提供稳定的高随机I/O、高数据可靠性的高性能存储。
- 高效共享块存储：采用固态硬盘与机械硬盘的混合介质作为存储介质。

挂载到实例上时，共享块存储与云盘共享数据盘配额，即一台实例最多挂载16块数据盘。

更多共享块存储的信息，请参见 [共享块存储FAQ](#)。

计费

共享块存储目前处于免费公测阶段，不收费。

云盘的计费方式与创建方式有关：

- 随包年包月实例一起创建的云盘，需要先付费再使用。更多计费信息，请参见 [包年包月](#)。
- 随按量付费实例一起创建的云盘，以及单独创建的云盘，均采用按量付费方式计费。更多计费信息，请参见 [按量付费](#)。

您可以使用不同的方式转换云盘的计费方式，如下表所示。

计费方式转换	功能	生效时间	适用的云盘
包年包月—>按量付费	续费降配	在新的计费周期生效	包年包月实例上挂载的包年包月数据盘。不能变更系统盘的计费方式
按量付费—>包年包月	升级配置	立即生效	包年包月实例上挂载的按量付费数据盘。不能变更系统盘的计费方式
	按量付费转预付费		按量付费实例上挂载的系统盘和数据盘

相关操作

您可以对云盘执行以下操作：

- 如果您 [单独创建了云盘作为数据盘](#)，需要先在控制台上 [挂载云盘](#)，再登录ECS实例 [分区格式化数据盘](#)。
- 如果您想加密存储在云盘上的数据，可以 [加密云盘](#)。
- 如果您的系统盘容量不足，可以 [扩容云盘容量](#)。
- 如果您的数据盘容量不足，可以 [扩容数据盘](#)。
- 如果您想更换操作系统，可以 [更换系统盘](#)。
- 如果您想备份云盘的内容，可以为云盘 [手动创建快照](#) 或者 [为硬盘设置自动快照策略](#) 按时自动创建快照。
- 如果您想在一台实例上使用另一台实例的操作系统和数据环境信息，可以 [使用后者的系统盘快照创建自定义镜像](#)。
- 如果您想将云盘恢复到某份快照时的状态，可以使用快照 [回滚云盘](#)。
- 如果您想将云盘恢复到创建时的状态，可以 [重新初始化云盘](#)。
- 如果您不再需要一块作数据盘用的按量付费云盘，可以 [卸载数据盘](#) 并 [释放云盘](#)。
- 如果您不再需要一块预付费云盘，可以将它 [转为按量付费云盘](#) 后，再 [卸载数据盘](#) 并 [释放云盘](#)。

更多云盘的操作，请参见《用户指南》的 [云盘](#) 章节。

3.2 云盘三副本技术

阿里云分布式文件系统为ECS提供稳定、高效、可靠的数据随机访问能力。

您对云盘的读写最终都会被映射为对阿里云数据存储平台上的文件的读写。阿里云提供一个扁平的线性存储空间，在内部会对线性地址进行切片，一个分片称为一个Chunk。对于每一个Chunk

，阿里云会复制出三个副本，并将这些副本按照一定的策略存放在集群中的不同节点上，保证您数据的可靠。

三份副本的原理

在阿里云数据存储系统中，有三类角色，分别称为Master、Chunk Server和Client。您的一个写操作，经过层层转换，最终会交由Client来执行，执行过程简要说明如下：

1. Client计算出这个写操作对应的Chunk。
2. Client向Master查询该Chunk的三份副本的存放位置。
3. Client根据Master返回的结果，向这3个Chunk Server发出写请求。
4. 如果三份都写成功，Client向您返回成功；反之，Client向您返回失败。

Master的分布策略会综合考虑集群中所有Chunk Server的硬盘使用情况、交换机的分布情况、电源供电情况、机器负载情况，尽量保证一个Chunk的所有副本分布在不同机架下的不同Chunk Server上，有效防止由于一个Chunk Server或一个机架的故障导致的数据不可用。

数据保护机制

当有数据节点损坏，或者某个数据节点上的部分硬盘发生故障时，集群中部分Chunk的有效副本数就会小于3。一旦发生这种情况，Master就会发起复制机制，在Chunk Server之间复制数据，使集群中所有Chunk的有效副本数达到3份。

综上所述，对云盘上的数据而言，所有用户层面的操作都会同步到底层三份副本上，无论是新增、修改还是删除数据。这种模式，能够保障您数据的可靠性和一致性。

如果ECS实例由于病毒感染、人为误删除或黑客入侵等软故障原因造成数据丢失，需要采用备份、[快照](#)等技术手段来解决。任何一种技术都不可能解决全部的问题，因地制宜地选择合适的数据保护措施，才能为您宝贵的业务数据筑起一道坚实的防线。

3.3 ECS 云盘加密

本文中的云盘指代 云盘 和 共享块存储。下文中统一简称为 云盘，除非特别指出。

什么是 ECS 云盘加密

当您的业务因为业务需要或者认证需要，要求您对您存储在云盘上的数据进行加密，阿里云 ECS 云盘加密功能能对云盘加密，为您提供了一种简单的安全的加密手段，能够对您新创建的云盘进行加密处理。您无需构建、维护和保护自己的密钥管理基础设施，您也无需更改任何已有的应用程序和运维流程，无需做额外的加解密操作，云盘加密功能对于您的业务是无感的。

加密解密的过程对于云盘的性能几乎没有衰减。关于性能测试方式，请参见 [块存储性能](#)。

在创建加密云盘并将其挂载到 ECS 实例后，将对以下类型的数据进行加密：

- 云盘中的数据
- 云盘和实例间传输的数据（实例操作系统内数据不再加密）
- 加密云盘创建的所有快照（加密快照）

加密解密是在 ECS 实例所在的宿主机上进行的，对从 ECS 实例传输到云盘的数据进行加密。

ECS 云盘加密支持所有在售云盘（普通云盘、高效云盘、SSD 云盘和 ESSD 云盘）和共享块存储（高效共享块存储和 SSD 共享块存储）。

ECS 云盘加密支持所有在售的实例规格。所有地域都支持云盘的加密。

ECS 云盘加密的依赖

ECS 云盘加密功能依赖于同一地域的密钥管理服务（Key Management Service, KMS），但是您无需到密钥管理服务控制台做额外的操作，除非您有单独的 KMS 操作需求。

首次使用 ECS 云盘加密功能（在 ECS 实例售卖页或者独立云盘售卖页）时，需要根据页面提示授权开通密钥管理服务（KMS），否则将无法购买带有加密云盘的实例或者加密的独立云盘。

如果通过 API 或者 CLI 使用 ECS 云盘加密功能，比如 [#unique_39](#)、[#unique_40](#)，您需要先在阿里云网站上开通密钥管理服务。

当您在同一地域第一次使用加密盘时，ECS 系统会为您在密钥管理服务（KMS）中的使用地域自动创建一个专为 ECS 使用的用户主密钥（Customer Master Key, CMK），这个用户主密钥，您将不能删除，您可以在密钥管理服务控制台上查询到该用户主密钥。

ECS 云盘加密的密钥管理

ECS 云盘加密功能会为您处理密钥管理。每个新创建云盘都使用一个唯一的 256 位密钥（来自于用户主密钥）加密。此云盘的所有快照以及从这些快照创建的后续云盘也关联该密钥。这些密钥受阿里云密钥管理基础设施的保护（由密钥管理服务提供），这将实施强逻辑和物理安全控制以防止未经授权的访问。您的数据和关联的密钥使用行业标准的 AES-256 算法进行加密。

您无法更改与已经加密了的云盘和快照关联的用户主密钥（CMK）。

阿里云整体密钥管理基础设施符合(NIST) 800-57 中的建议，并使用了符合 (FIPS) 140-2 标准的密码算法。

每个阿里云 ECS 账号在每个地域都具有一个唯一的用户主密钥（CMK），该密钥与数据分开，存储在一个受严格的物理和逻辑安全控制保护的系统中。每个加密盘及其后续的快照都使用云盘粒度唯一的加密密钥（从该用户该地域的用户主密钥创建而来），会被该地域的用户主密钥（CMK）加

密。云盘的加密密钥仅在您的 ECS 实例所在的宿主机的内存中使用，永远不会以明文形式存储在任
何永久介质（如云盘）上。

费用

ECS 不对云盘加密功能收取额外的费用。

ECS 为您在每个地域创建的用户主密钥（CMK）属于服务密钥，不收取额外费用，也不占用您在
每个地域的主密钥数量限制。



说明：

您对云盘的任何读写操作（例如 mount/umount、分区、格式化等）都不会产生费用。凡是涉及
云盘本身的管理操作（见下面列表），无论是通过 ECS 管理控制台还是通过 API，均会以 API 的
形式使用到密钥管理服务（KMS），将会记入到您在该地域的 KMS 服务 API 调用次数。

对加密云盘的管理操作包括：

- 创建加密盘（[#unique_39](#) 或 [#unique_40](#)）
- 挂载（[#unique_41](#)）
- 卸载（[DetachDisk](#)）
- 创建快照（[#unique_43](#)）
- 回滚云盘（[#unique_44](#)）
- 重新初始化云盘（[#unique_45](#)）

如何创建加密的云盘

目前，ECS 云盘加密功能只支持云盘。您可以通过不同渠道创建加密云盘：

- 通过购买实例页面：
 - 勾选加密选项，创建加密的空盘。
 - 选择加密快照来创建云盘。
- 通过 API 或 CLI：
 - 指定参数 `DataDisk.n.Encrypted`（[#unique_39](#)）或者 `Encrypted`（[#unique_40](#)）为 `true`。
 - 在 `CreateInstance` 或 `CreateDisk` 中，指定加密快照的 `SnapshotId`。

数据加密状态的转换

已经存在的 非加密盘，不能直接转换成 加密盘。同样的，已经存在的 加密盘，不能直接转换成 非
加密盘。

已经存在的 非加密盘产生的快照，不能直接转换成 加密快照。同样的，已经存在的 加密盘产生的快照，不能直接转换成 非加密快照。

所以，如果您需要对现有数据 非加密状态 转换为 加密状态，阿里云推荐用 Linux 下的 `rsync` 命令或者 Windows 下的 `robocopy` 命令将数据从 非加密盘 上复制到（新创建的）加密盘 上。

如果您需要对现有数据 加密状态 转换为 非加密状态，则用 Linux 下的 `rsync` 命令或者 Windows 下的 `robocopy` 命令将数据从 加密盘 上复制到（新创建的）非加密盘 上。

限制

ECS 云盘加密有如下限制：

- 只能加密云盘，不能加密本地盘。
- 只能加密数据盘，不能加密系统盘。
- 已经存在的非加密盘，不能直接转换成加密盘。
- 已经加密的云盘，也不能转换为非加密云盘。
- 已经存在的非加密盘产生的快照，不能直接转换成加密快照。
- 加密快照不能转换为非加密快照。
- 不能共享带有加密快照的镜像。
- 不能跨地域复制带有加密快照的镜像。
- 不能导出带有加密快照的镜像。
- 每个地域每个用户无法自己选择用户主密钥 CMK，由系统为您生成。
- 每个地域 ECS 系统创建的用户主密钥（CMK），用户不能删除，但不收费用。
- 不支持在云盘加密后更换该云盘用于加解密的关联的用户主密钥。

3.4 创建云盘

3.4.1 创建按量付费云盘

您可以通过控制台云盘管理页面或者API接口单独创建云盘，增加系统的存储空间。本文介绍如何在ECS控制台上单独创建一个按量付费的云盘。

注意事项

创建云盘前，您需要了解以下信息：

- 通过 [ECS管理控制台](#) 云盘 页面只能创建数据盘，不能创建系统盘。



说明：

您可以在创建按量计费ECS实例时同时创建云盘作为数据盘。通过这种方式创建的云盘，计费方式与实例的计费方式相同。

- 您可以创建全新的空云盘，也可以 [使用快照创建云盘](#)。
- 每个账号在所有地域创建作数据盘用的按量付费云盘限额 = 该账号在所有地域的按量付费实例数量 * 5。详细信息，请参见 [使用限制](#)。
- 目前云服务器ECS不支持合并多块云盘。云盘创建后，每块云盘都是独立个体，无法通过格式化将多块云盘空间合并到一起。建议您提前做好云盘数量和容量的规划。
- 已创建的多块云盘，不建议做LVM（Logical Volume Manager）等逻辑卷。因为快照只能针对独立的每个云盘创建，使用LVM后，使用快照回滚云盘时会造成数据差异。
- 创建后，您可以转换云盘的计费方式：
 - 如果云盘挂载在包年包月实例上，通过 [预付费实例升级配置](#) 将按量付费云盘转换为包年包月云盘。
 - 如果云盘挂载在按量付费实例上，通过 [按量付费转预付费](#) 将按量付费云盘转为包年包月云盘。
- 通过这种方式创建的云盘，只要未转换计费方式，您能随时 [卸载数据盘](#) 并 [释放云盘](#)。

前提条件

创建云盘前，您必须确认以下信息：

- 需要挂载云盘的实例所在的地域和可用区。[挂载云盘](#) 时，实例与云盘必须在同一个地域和可用区。
- 云盘是否需要加密。详细信息，请参见 [ECS云盘加密](#)。

操作步骤

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏中，单击 存储 > 云盘。
3. 在 磁盘列表 页右上角，单击 创建云盘 进入 创建 页面。
4. 选择地域和可用区。



说明：

挂载云盘时，实例与云盘必须在同一个地域和可用区。

5. 选择云盘的类型、大小，确认是否加密，并设置购买量。您也可以选择 [使用快照创建云盘](#)。



6. 确认当前配置和配置费用。

7. 单击 确认订单，并根据页面引导完成创建。

创建成功后，回到 磁盘列表 页，刷新列表，您可以看到新建的云盘，磁盘状态显示为 待挂载。

后续操作

[挂载云盘](#)。

相关 API

单独创建云盘：[#unique_40](#)

随实例一起创建云盘：[#unique_51](#) 或 [#unique_39](#)

3.4.2 创建预付费云盘

您可以在控制台为预付费实例单独创建预付费云盘来增加系统的存储空间。

注意事项

创建云盘前，您需要了解以下信息：

- 通过ECS管理控制台的实例页面为预付费实例创建的预付费云盘，此云盘采用[预付费#包年包月#](#)方式计费，只能作数据盘使用。



说明：

您可以在创建预付费ECS实例的同时添加云盘作为数据盘。通过这种方式创建的云盘，计费方式与实例的计费方式相同。

- 您可以创建全新的空云盘，也可以用快照单独创建云盘。具体操作，请参见 [创建按量付费云盘](#)和[使用快照创建云盘](#)。
- 目前ECS不支持合并多块云盘。云盘创建后，每块云盘都是独立个体，无法通过格式化将多块云盘空间合并到一起。建议您提前做好云盘数量和容量的规划。

- 已创建的多块云盘，不建议做LVM（Logical Volume Manager）等逻辑卷。因为快照只能针对独立的每个云盘创建，使用LVM后，使用快照回滚云盘时会造成数据差异。
- 通过这种方式创建的云盘，不支持直接[卸载数据盘](#)。云盘到期时间和实例一致，随实例一起释放。



说明：

如果想要释放云盘，您可以先将预付费云盘转换为按量付费云盘，再卸载并释放云盘。

操作步骤

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏中，单击实例。



3. 找到需要添加预付费云盘的预付费实例，在操作栏中，单击更多 > 资源变配 > 添加预付费云盘。



4. 在云盘区域框中，填写以下几项：

- 云盘类型：在下拉框中选择需要新建的[云盘类型](#)。
- 云盘容量：在文本框中输入要添加的云盘容量。可填容量范围为20GiB到32768GiB。
- 云盘加密：如果您需要为添加的[#unique_54](#)，勾选加密。
- 购买量：在文本框中输入要添加的云盘的数量。



说明：

单实例数据盘数量（包括云盘和共享块存储）最多为16块。

- 云盘名称：可选项。可在文本框中输入要添加的云盘名称。名称长度为 [2, 128] 个英文或中文字符，可以包含数字、点号 (.)、半角冒号 (:)、下划线 (_) 或者连字符 (-)。
- 云盘描述：可选项。可在文本框中输入要添加的云盘的描述信息。描述信息长度为 [2, 256] 个英文或中文字符，不能以 http:// 和 https:// 开头。
- 如果您需要[使用快照创建云盘](#)，单击用快照创建磁盘。

The screenshot shows the 'Create Cloud Disk' (创建云盘) interface. At the top left is a '云盘' (Cloud Disk) icon. The main form includes a dropdown menu for '云盘类型' (Cloud Disk Type) set to 'SSD云盘' (SSD Cloud Disk), a text input for '容量' (Capacity) set to '20', and a unit selector set to 'GiB'. To the right of the capacity input are '2400 IOPS' and a checkbox for '加密' (Encryption). A blue link '用快照创建磁盘' (Create disk from snapshot) is on the far right. Below these fields is a link '如何选择 SSD 云盘 / 高效云盘 / 普通云盘，请看 详细说明>' (How to choose SSD cloud disk / High-performance cloud disk / General cloud disk, see detailed explanation>). The '购买量' (Purchase quantity) section shows a text input set to '1' and the unit '块' (Pieces), with a note '还可以挂载 16 块，已挂载 0 块' (Can still mount 16 pieces, 0 pieces mounted). The '名称' (Name) section has a text input with placeholder '请输入云盘名称' (Please enter cloud disk name) and a note '2-128个字符，以大小写字母或中文开头，可包含数字、"."、"_"、":"或"-"' (2-128 characters, starting with uppercase or lowercase letters or Chinese, can contain numbers, ".", "_", ":", or "-"). The '描述' (Description) section has a text input with placeholder '请输入云盘描述' (Please enter cloud disk description) and a note '长度为2-256个字符，不能以http://或https://开头' (Length 2-256 characters, cannot start with http:// or https://).

5. 勾选《云服务器ECS服务条款》。
6. 单击确认订单。
7. 单击创建订单。
8. 选择支付方式并单击确认支付以完成创建。
9. 单击管理控制台，在实例列表中单击添加了预付费云盘的实例名称。

10. 单击本实例磁盘，在磁盘列表中可找到新添加的预付费云盘。



预付费云盘已自动挂载在该实例中，并处于使用中状态。

后续操作

分区并格式化新创建的云盘。具体步骤请参见[Linux格式化数据盘](#)或[Windows格式化数据盘](#)。

3.4.3 使用快照创建云盘

本文介绍如何在云服务器ECS控制台上使用一份快照创建一块云盘，快照对应的历史云盘可以是系统盘或数据盘。创建的云盘可以挂载到同一地域同一可用区下的任何一台ECS实例上。

应用场景

使用快照创建云盘可以读取已有云盘或者误释放云盘的数据，例如：

- 如果您需要获取快照中的某些数据，且不希望通过[回滚云盘](#)的方式实现，您可以使用快照创建一块新云盘，然后从已有云盘中读取数据。
- 当系统盘出现故障导致无法运行实例时，您可以使用系统盘已有的快照创建一块新云盘，再作为数据盘挂载到健康的实例上，从而读取系统盘数据。

注意事项

- 使用快照创建的云盘默认采用[按量付费](#)方式计费，并且只能作数据盘用。如果您需要转换云盘的计费方式，请参见下文[转换计费方式](#)。
- 由于快照存储在[对象存储OSS](#)中，在使用快照创建云盘时，云服务器ECS从OSS读取数据并写入云盘，该过程需要一段时间，会造成首次访问时云盘性能下降。建议您在正式使用新云盘前，等待云盘完成所有数据块的读取，避免初期性能下降带来的不便。
- 您在所有地域创建的按量付费数据盘不能超过您在所有地域的按量付费实例数量的五倍。更多详情，请参见[使用限制](#)。
- 不支持合并多块云盘。云盘创建后，每块云盘都是独立个体，无法通过格式化将多块云盘空间合并到一起。建议您提前做好云盘数量和容量的规划。

- 已创建的多块云盘，不建议制作LVM（Logical Volume Manager）逻辑卷。因为快照只能备份单块云盘的数据，使用LVM后，回滚云盘时会造成数据差异。

前提条件

您已经为历史系统盘或数据盘创建了快照，并确认快照ID。具体操作，请参见[创建快照](#)。

操作步骤

在ECS管理控制台上创建云盘的操作路径如下所示。您也可以通过API [CreateDisk](#)完成操作。

1. 登录[ECS管理控制台](#)。
2. 在左侧导航栏，选择存储与快照 > 云盘。
3. 在磁盘列表页右上角，单击创建云盘。
4. 选择地域和可用区。



说明：

由于挂载云盘时，实例与云盘必须在同一个地域同一个可用区，请根据实例部署情况选择云盘的可用区。

5. 在创建云盘页面，完成以下选购配置。
 - a. 选择云盘类型，可选项有高效云盘、ESSD云盘以及SSD云盘。
 - b. 单击用快照创建磁盘，并选择一份快照。
 - c. 设置云盘的容量，新容量必须大于20 GiB，小于32768 GiB。



说明：

- 如果您不设置云盘容量，系统会根据快照对应的源云盘大小自动设置云盘大小。
- 如果您设置的容量大于快照大小，您必须重新分区才能使用全部容量。
- 如果您的快照容量小于2048 GiB，但您希望设置的新云盘容量大于2048 GiB，请提前确认快照对应的历史云盘采用的是GPT（Globally Unique Identifier Partition Table，GUID分区表）分区格式。否则，建议您设置为小于2048 GiB，避免配置分区容量时带来数据丢失的风险。更多详情，请参见[分区格式化大于2 TiB云盘](#)。

- d. 设置购买量。
 - e. 勾选《云服务器 ECS 服务条款》。
6. 确认当前配置和配置费用。
 7. 单击确认订单，根据页面引导完成创建。

回到磁盘列表页后，刷新列表，您可以看到新建的云盘，磁盘状态显示为待挂载。

下一步

挂载云盘

转换计费方式

使用快照创建的云盘默认采用按量付费计费方式。创建后，您可以转换云盘的计费方式：

- 如果云盘挂载在预付费实例上，可以通过[预付费实例升级配置](#)转换为预付费云盘。
- 如果云盘挂载在按量付费实例上，可以通过[按量付费转预付费](#)转换为预付费云盘。

您也可以[在创建ECS实例](#)的同时使用系统盘或数据盘的快照创建云盘。通过这种方式创建的云盘，计费方式与实例相同。

3.5 挂载云盘

您可以将从云盘管理页面单独创建的云盘（作数据盘用）挂载到ECS实例上。您可以选择从实例管理页面挂载云盘，也可以从云盘管理页面挂载云盘。

注意事项

在挂载云盘前，您需要了解以下注意事项：

- 您只能挂载作数据盘用的云盘，不需要挂载作系统盘用的云盘。
- 挂载云盘时，实例必须满足以下条件：
 - 实例状态必须为 运行中（Running）或者 已停止（Stopped），不能为 已锁定（Locked）。
 - 实例不欠费。
- 挂载云盘时，云盘的状态必须为 待挂载（Available）。
- 云盘只能挂载到同一地域下同一可用区内的实例上，不能跨可用区挂载。
- 一台ECS实例最多能挂载16块云盘作数据盘用，同一时刻，一块云盘只能挂载到一个实例上。
- 通过 [ECS管理控制台](#) 上的 云盘页面独立创建的云盘能挂载到同一地域下同一可用区的任意实例上（包括预付费和按量付费的实例）。

前提条件

您已经在同一可用区创建了ECS实例和云盘。详细操作，请参见 [使用向导创建实例](#) 或 [创建按量付费云盘](#)。

在实例管理页面挂载云盘

若要在同一ECS实例上挂载多块云盘，在实例管理页面操作比较方便。按以下步骤在选定的实例上挂载云盘：

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏中，单击 实例。
3. 选择地域。
4. 找到需要挂载云盘的实例，单击实例ID。
5. 在左侧导航栏中，单击 本实例磁盘，并在磁盘列表右上方单击 挂载云盘。
6. 在弹出对话框中，完成以下设置：
 - 目标磁盘：只能选择同一可用区里状态为 待挂载 的云盘。
 - 磁盘随实例释放：如果选中，当实例释放时，该云盘也会同时释放。
 - 自动快照随磁盘释放：如果选中，当云盘释放时，该云盘创建的所有自动快照都会一起释放，但手动快照不会释放。建议您不要选择该选项，保留备份数据。

完成设置后，单击 确定，并单击 执行挂载。

7. 刷新本实例磁盘列表。

如果该云盘的状态变为 使用中，表示挂载成功。

8. 根据云盘的内容和ECS实例的操作系统，您必须执行不同的操作使云盘可用。如下表所示。

云盘的内容	实例的操作系统	后续操作
全新的空云盘	Linux	Linux格式化数据盘 。大于2 TiB的云盘，请参见 分区格式化大于2 TiB云盘 。
	Windows	Windows格式化数据盘 。大于2 TiB的云盘，请参见 分区格式化大于2 TiB云盘 。
使用快照创建的云盘	Linux	远程连接实例，并执行 <code>mount <分区> <挂载点></code> 命令，将需要挂载的分区挂载到合适的挂载点上，才能正常使用云盘。
	Windows	不需要执行其他操作，能直接使用云盘。

在云盘管理页面挂载云盘

若要将多块云盘挂载到不同的ECS实例上，在云盘管理页面操作比较方便。按以下步骤将选中的云盘挂载到实例上：

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏中，选择 存储 > 云盘。
3. 选择地域。
4. 找到 待挂载 云盘，在 操作 列中，选择 更多 > 挂载。

5. 在弹出对话框中，完成以下设置：

- 目标实例：只能选择同一可用区的ECS实例。
- 磁盘随实例释放：如果选中，当实例释放时，该云盘也会同时释放。
- 自动快照随磁盘释放：如果选中，当云盘释放时，该云盘创建的所有自动快照都会一起释放，但手动快照不会释放。建议您不要选择该选项，保留备份数据。

完成设置后，单击 执行挂载。

6. 刷新云盘列表。

如果该云盘的状态变为 使用中，表示挂载成功。

7. 根据云盘的内容和ECS实例的操作系统，您必须执行不同的操作使云盘可用。如下表所示。

云盘的内容	实例的操作系统	后续操作
全新的空云盘	Linux	Linux格式化数据盘 。大于2 TiB的云盘，请参见 分区格式化大于2 TiB云盘 。
	Windows	Windows格式化数据盘 。大于2 TiB的云盘，请参见 分区格式化大于2 TiB云盘 。
使用快照创建的云盘	Linux	连接实例并执行 mount 命令挂载分区后，才能正常使用云盘。
	Windows	不需要执行其他操作，能直接使用云盘。

后续操作

使用云盘一段时间后，根据业务需求，您能执行以下不同的操作：

- 您可以通过 [重新初始化云盘](#)，将云盘回复到初始状态。
- 如果需要更大容量的云盘，您可以扩容。具体操作，请参见 [扩展分区和文件系统-Linux数据盘](#) 或者 [扩展分区和文件系统-Windows](#)。
- 您可以通过 [#unique_30](#) 备份云盘的数据。或者，[为磁盘设置自动快照策略](#)，创建自动快照。
- 如果您需要将云盘恢复到某个时刻的状态，可以使用快照 [回滚云盘](#)。
- 如果您的实例不再需要一块云盘，为避免产生不必要的费用，可以 [卸载数据盘](#) 并 [释放云盘](#)。

相关API

[#unique_41](#)

3.6 分区格式化数据盘

3.6.1 Windows格式化数据盘

本文描述如何为Windows实例的全新数据盘创建一个单分区并挂载文件系统。您也可以根据业务需要，对数据盘进行多分区配置。

风险提示

- 磁盘分区和格式化是高风险行为，请慎重操作。本文描述如何处理一块全新的数据盘，如果您的数据盘上有数据，请务必为数据盘[创建快照](#)，避免数据丢失。
- 云服务器ECS仅支持数据盘分区操作，不支持系统盘分区操作。如果您强行使用第三方工具对系统盘做分区操作，可能引发系统崩溃和数据丢失等未知风险。仅允许在扩容系统盘后做扩展分区或新增分区操作，具体操作请参见[扩展分区和文件系统-Windows](#)。

自动分区操作步骤

无论是随实例一起购买的数据盘，还是单独购买或创建的数据盘，无论是不大于2 TiB还是大于2 TiB的数据盘，都可以通过系统自动完成分区格式化操作。

- 随实例一起购买的数据盘：创建实例后，您可以[远程连接实例](#)，并在实例内部的磁盘管理页面查看分区情况。
- 单独购买或创建的数据盘：
 1. [创建数据盘](#)。
 2. [挂载数据盘](#)。
 3. 在ECS控制台或通过API（[RebootInstance](#)）[重启ECS实例](#)。
 4. [远程连接实例](#)。
 5. 在实例内部的磁盘管理页面查看分区情况。

数据盘自动分区的执行结果是一个单分区，挂载的是NTFS文件系统。

手动分区操作步骤

本操作步骤仅适用于不大于2 TiB的数据盘，大于2 TiB的数据盘，请参见[分区格式化大于2 TiB云盘](#)。示例步骤以Windows Server 2012 R2 64位操作系统为例，分区并格式化一块20 GiB的数据盘。

1. [远程连接实例](#)。
2. 在Windows Server桌面，右键单击开始图标，选择磁盘管理。
3. 查找到未格式化分区的数据盘（如磁盘 2），其处于脱机状态。

4. 右键单击磁盘 2 周边的空白区，在弹出菜单中，选择联机。

联机后，磁盘 2 的状态显示为没有初始化。

5. 右键单击磁盘 2 周边的空白区，在弹出菜单中，选择初始化磁盘。

6. 在初始化磁盘 对话框里，选择磁盘2，并选择磁盘分区形式：

- MBR 目前仍是最常用的分区形式，但是，MBR 只支持处理不大于 2 TiB 的数据盘，而且，只支持分 4 个主区，如果您要将磁盘分成更多的区，需要将某个主区作为扩展区并在其中创建逻辑分区。
- GPT 是一种新的分区形式，早期版本的 Windows 不能识别这种分区形式。GPT 能处理的数据盘容量由操作系统和文件系统决定。在 Windows 操作系统里，GPT 最多可以支持 128 个主分区。

在本示例中，我们选择 MBR 分区形式，并单击确定。

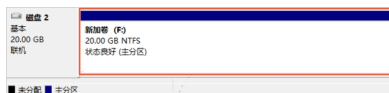
7. 在 磁盘管理 窗口，右键单击磁盘 2 的未分配区域，选择新建简单卷。

8. 在新建简单卷向导中，完成以下操作：

- 单击下一步。
- 指定卷大小：指定简单卷大小。如果您只要创建一个主区，使用默认值。单击下一步。
- 分配驱动器号和路径：选择一个驱动器号（即盘符），如本示例中选择 F。单击下一步。
- 格式化分区：选择格式化设置，包括文件系统、分配单元大小和卷标，确认是否执行快速格式化和启用文件和文件夹压缩。这里使用默认设置。单击下一步。
- 开始创建新简单卷。当向导对话框里出现以下截图中的信息时，说明已经完成新简单卷的创建。单击完成关闭新建简单卷向导。

执行结果

格式化分区完成后，磁盘管理中磁盘 2 的状态如下图所示。



您可以在这台电脑中查看到新建的驱动器新加卷 (F:)。至此，您就可以开始使用这个数据盘了。

3.6.2 Linux 格式化数据盘

本文描述如何为 Linux 实例的全新数据盘创建一个单分区并挂载文件系统。您也可以根据业务需要，对数据盘进行多分区配置。



说明：

本文操作仅适用小于等于2 TiB的数据盘。大于2 TiB的数据盘，请参见[分区格式化大于2 TiB云盘](#)。

风险提示

- 磁盘分区和格式化是高风险行为，请慎重操作。本文描述如何处理一块全新的数据盘，如果您的数据盘上有数据，请务必为数据盘[创建快照](#)，避免数据丢失。
- 云服务器ECS仅支持数据盘分区操作，不支持系统盘分区操作。如果您强行使用第三方工具对系统盘做分区操作，可能引发系统崩溃和数据丢失等未知风险。仅允许在扩容系统盘后做扩展分区或新增分区操作，具体操作请参见[扩展分区与文件系统-Linux系统盘](#)。

准备工作

- 单独[购买的数据盘](#)必须先[挂载数据盘](#)才能格式化。随实例一起购买的数据盘，无需挂载。
- 在ECS控制台的云盘页面中，选择更多 > 修改属性查看数据盘的设备名。



说明：

数据盘的设备名默认由系统分配，I/O优化实例的数据盘设备名从`/dev/vdb`递增排列，包括`/dev/vdb`–`/dev/vdz`。如果数据盘设备名为`dev/xvd*`（*是a–z的任意一个字母），表示您使用的是非I/O优化实例。

操作步骤

本示例采用一块新的20 GiB数据盘，设备名为`/dev/vdb`，创建一个单分区数据盘并格式化为ext4文件系统。使用的是I/O优化实例，操作系统为CentOS 7.6。

1. [远程连接实例](#)。
2. 运行`fdisk -l`命令查看实例上的数据盘。



说明：

执行命令后，如果不存在`/dev/vdb`，表示您的实例没有数据盘。确认数据盘是否已挂载。

3. 依次执行以下命令以创建一个单分区数据盘：
 - a. 运行`fdisk -u /dev/vdb`：分区数据盘。
 - b. 输入p：查看数据盘的分区情况。本示例中，数据盘没有分区。
 - c. 输入n：创建一个新分区。
 - d. 输入p：选择分区类型为主分区。



说明：

本示例中创建一个单分区数据盘，所以只需要创建主分区。如果要创建四个以上分区，您应该创建至少一个扩展分区，即选择e（extended）。

- e. 输入分区编号并按回车键。本示例中，仅创建一个分区，输入1。
- f. 输入第一个可用的扇区编号：按回车键采用默认值2048。
- g. 输入最后一个扇区编号：本示例仅创建一个分区，按回车键采用默认值。
- h. 输入p：查看该数据盘的规划分区情况。
- i. 输入w：开始分区，并在分区后退出。

```
[root@ecshost~ ]# fdisk -u /dev/vdb
Welcome to fdisk (util-linux 2.23.2).
Changes will remain in memory only, until you decide to write
them.
Be careful before using the write command.
Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x3e60020e.

Command (m for help): p
Disk /dev/vdb: 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: 0x3e60020e
Device Boot Start End Blocks Id System

Command (m for help): n
Partition type:
p primary (0 primary, 0 extended, 4 free)
e extended
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-41943039, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-41943039, default
41943039):
Using default value 41943039
Partition 1 of type Linux and of size 20 GiB is set

Command (m for help): p

Disk /dev/vdb: 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: 0x3e60020e
Device Boot Start End Blocks Id System
/dev/vdb1 2048 41943039 20970496 83 Linux

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

Calling ioctl() to re-read partition table.
```



```
Syncing disks.
```

4. 运行命令 `fdisk -lu /dev/vdb` 查看新分区。

如果出现以下信息，表示新分区 `/dev/vdb1` 创建成功。

```
[root@ecshost~]# fdisk -lu /dev/vdb

Disk /dev/vdb: 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: 0x3e60020e

Device Boot Start End Blocks Id System
/dev/vdb1 2048 41943039 20970496 83 Linux
```

5. 运行命令 `mkfs.ext4 /dev/vdb1` 在新分区上创建一个文件系统。

本示例中，创建一个 `ext4` 文件系统。您也可以根据自己的需要，选择创建其他文件系统，例如：如果您需要在 Linux、Windows 和 Mac 系统之间共享文件，可以使用 `mkfs.vfat` 创建 VFAT 文件系统。



说明：

创建文件系统所需时间取决于数据盘大小。

```
[root@ecshost~]# mkfs.ext4 /dev/vdb1

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
1310720 inodes, 5242624 blocks
262131 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2153775104
160 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, 1605632,
2654208,
4096000

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

6. （建议）运行命令 `cp /etc/fstab /etc/fstab.bak` 备份 `etc/fstab` 文件。

7. 运行命令 `echo /dev/vdb1 /mnt ext4 defaults 0 0 >> /etc/fstab`向`/etc/fstab`写入新分区信息。



说明:

Ubuntu 12.04系统不支持barrier, 您需要运行命令`echo '/dev/vdb1 /mnt ext4 barrier=0 0 0' >> /etc/fstab`。

如要把数据盘单独挂载到某个文件夹, 例如单独用来存放网页, 则将命令中`/mnt`替换成所需的挂载点路径。

8. 运行命令`cat /etc/fstab`查看`/etc/fstab`中的新分区信息。

```
[root@ecshost~]# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Wed Dec 12 07:53:08 2018
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=d67c3b17-255b-4687-be04-f29190d37396 / ext4 defaults 1 1
/dev/vdb1 /mnt ext4 defaults 0 0
```

9. 运行命令`mount /dev/vdb1 /mnt`挂载文件系统。

10. 运行命令`df -h`查看目前磁盘空间和使用情况。

出现新建文件系统的信息, 表示挂载成功, 您不需要重启实例即可以使用新的文件系统。

```
[root@ecshost~]# df -h

Filesystem Size Used Avail Use% Mounted on
/dev/vda1 40G 1.6G 36G 5% /
devtmpfs 234M 0 234M 0% /dev
tmpfs 244M 0 244M 0% /dev/shm
tmpfs 244M 484K 244M 1% /run
tmpfs 244M 0 244M 0% /sys/fs/cgroup
tmpfs 49M 0 49M 0% /run/user/0
/dev/vdb1 20G 45M 19G 1% /mnt
```

3.6.3 分区格式化大于2 TiB云盘

如果您要分区格式化一块大于2 TiB的作数据盘用的云盘(本文统一称为 大容量数据盘, 小于2 TiB的数据盘统称为 小容量数据盘), 您必须采用GPT分区形式。本文档描述了如何在不同的操作系统里分区格式化一块大容量数据盘。



说明:

如果您要分区格式化一块小于2 TiB的数据盘，请参见 [Linux格式化数据盘](#) 和 [Windows格式化数据盘](#)。

注意事项

分区格式化大容量数据盘时，需要注意以下事项：

- 大容量数据盘支持的分区工具和文件系统如下表所示。

操作系统	分区工具	文件系统
Linux	parted	ext4或xfs
Windows	磁盘管理	NTFS

- 不建议使用小容量数据盘的快照创建大容量数据盘

理论上，您可以使用一块小容量数据盘的快照创建一个大容量数据盘，但是我们建议创建空的大容量数据盘，或者使用大容量数据盘的快照创建大容量数据盘。原因如下：

- 使用小容量数据盘的快照创建大容量数据盘时，系统只完成块设备级的磁盘扩容，并没有实现分区格式和文件系统的自动转换。
- 如果小容量数据盘快照中使用的是MBR分区格式，以上提到的分区工具（Linux上的 parted 和Windows上的 磁盘管理）都不能在保留数据的前提下将分区形式从MBR转换为GPT。所以，即使您使用小容量数据盘的快照创建了大容量数据盘，在分区格式化时，您都需要删除原有数据，再按照GPT格式分区。如果您已经用小容量数据盘的快照创建了大容量数据盘，请参见 [Windows里分区格式化由小容量数据盘的快照创建的大容量数据盘](#)。



说明：

如果小容量数据盘快照本身就是GPT分区格式，或者您另有强大的分区工具，则不在此列。您可以根据自身情况来选择。

- 数据盘快照的影响

大容量数据盘的数据量很大，但是创建快照的速度和小容量数据盘是一样的，所以每天创建快照的时间会与数据量成比例增长。创建快照的速度和数据的增量成正比，脏数据越多，创建快照耗时越久。

Windows里分区格式化空的大容量数据盘

这部分以Windows Server 2008 R2 64位系统为例，说明如何在Windows实例中分区格式化一块大容量数据盘。假设需要处理的数据盘是一个4 TiB的空盘。

前提条件

数据盘已经挂载到实例上。具体操作，请参见 [挂载云盘](#)。

操作步骤

按以下步骤分区格式化一块大容量数据盘：

1. [远程连接Windows实例](#)。

2. 在任务栏里，单击  图标。

3. 在 服务器管理器 的左侧导航栏里，选择 存储 > 磁盘管理。

4. 找到需要分区格式化的磁盘（本示例中为 磁盘 4）。磁盘状态显示为 脱机。

5. 右击磁盘 4 周边空白处，单击 联机。

联机后，磁盘 4 的状态显示为 没有初始化。

6. 右键单击磁盘 4 周边的空白区，在弹出菜单中，选择 初始化磁盘。

7. 在 初始化磁盘 对话框里，选择 磁盘 4，并选择磁盘分区形式为 GPT。

8. 在 磁盘管理 窗口，右键单击磁盘 4 的 未分配 区域，选择 新建简单卷，创建一个 4 TiB 的 NTFS 格式的卷。

9. 在 新建简单卷向导 中，完成以下操作：

a. 单击 下一步。

b. 指定卷大小：指定简单卷大小。如果您只要创建一个主区，使用默认值。单击 下一步。您也可以把 磁盘 4 分成多个分区来使用。



说明：

64

NTFS 卷上的最大尺寸，理论上，32 NTFS 的最大卷包含 $2^{32}-1$ 个簇。实际上，WinXP Pro 中，NTFS 卷的最大限制是 $2^{32}-1$ 个簇。举例来说，如果是 64 KiB 的簇，那 NTFS 卷的最大尺寸就是约 256 TiB。如果选择 4 KiB 的簇，那 NTFS 卷的最大尺寸就是约 16 TiB。NTFS 会根据磁盘的容量来自动选择簇的大小。

c. 分配驱动器号和路径：选择一个驱动器号（即盘符），如本示例中选择 G。单击 下一步。

d. 格式化分区：选择格式化设置，包括文件系统、分配单元大小和卷标，确认是否 执行快速格式化 和 启用文件和文件夹压缩。这里仅选择 执行快速格式化。单击 下一步。

e. 开始创建新简单卷。当向导对话框里显示已经完成新简单卷的创建时，单击 完成，关闭 新建简单卷向导。

格式化分区完成后，磁盘管理 中 磁盘 4 的状态如下截图所示。

Windows里分区格式化由小容量数据盘的快照创建的大容量数据盘

如果您使用一个小容量数据盘的快照创建了一块大容量数据盘，您需要先将数据盘的分区形式从MBR转为GPT，再格式化数据盘，原来快照的数据将无法保存，所以我们不建议您使用小容量数据盘的快照创建大容量数据盘。

如果您确实创建了这样的大容量数据盘，按以下步骤分区格式化这块数据盘。本示例中的操作系统是Windows Server 2012 R2 64位，假设需要处理的数据盘容量为3 TiB。

前提条件

数据盘已经 [挂载](#) 到实例上。

操作步骤

按以下步骤分区格式化一块大容量数据盘：

1. [远程连接Windows实例](#)。
2. 在Windows Server桌面，右键单击 开始 图标，选择 磁盘管理。

未格式化分区的数据盘（如本示例中的磁盘 2）处于 脱机 状态。
3. 右键单击磁盘 2周边的空白区，在弹出菜单中，选择 脱机。
4. 右键单击一个简单卷，在弹出菜单中，选择 删除卷。
5. 右键单击磁盘 2周边的空白区，在弹出菜单中，选择 转换成GPT磁盘。
6. 在 磁盘管理 窗口，右键单击磁盘 2的 未分配 区域，选择 新建简单卷,创建一个3 TiB的NTFS格式的卷。
7. 在 新建简单卷向导 中，完成以下操作：
 - a. 单击 下一步。
 - b. 指定卷大小：指定简单卷大小。如果您只要创建一个主区，使用默认值。单击 下一步。您也可以把 磁盘 2 分成多个分区来使用。



说明：

64

NTFS卷上的最大尺寸，理论上， 2^{32} NTFS的最大卷包含 $2^{32}-1$ 个簇。实际上，WinXP Pro中，NTFS卷的最大限制是 $2^{32}-1$ 个簇。举例来说，如果是64 KiB的簇，那NTFS卷

的最大尺寸就是约256 TiB。如果选择4 KiB的簇，那NTFS卷的最大尺寸就是约16 TiB。NTFS会根据磁盘的容量来自动选择簇的大小。

- c. 分配驱动器号和路径：选择一个驱动器号（即盘符），如本示例中选择E。单击 下一步。
- d. 格式化分区：选择格式化设置，包括文件系统、分配单元大小和卷标，确认是否 执行快速格式化和 启用文件和文件夹压缩。这里仅选择 执行快速格式化。单击 下一步。
- e. 开始创建新简单卷。当向导对话框里显示已经完成新简单卷的创建时，单击 完成，关闭 新建简单卷向导。

格式化分区完成后，磁盘管理 中 磁盘 4 的状态如下截图所示。

Linux里分区格式化大容量数据盘

对于Linux实例上挂载的大容量数据盘，采用GPT分区形式。Linux系统里，大容量数据盘一般采用xfs或者ext4文件系统。

这部分以CentOS 7.4 64位系统为例，说明如何在Linux实例上使用 parted 和 e2fsprogs 工具分区并格式化一个大容量数据盘。假设需要处理的数据盘是一个新建的3 TiB的空盘，设备名为 `/dev/vdd`。

前提条件

您的Linux实例上已经安装了 parted。如果未安装，运行命令 `yum install -y parted`。

您的Linux实例上已经安装了 e2fsprogs。如果未安装，运行命令 `yum install -y e2fsprogs`。

数据盘已经挂载到实例上。详细信息，请参见 [挂载云盘](#)。

操作步骤

按以下步骤分区格式化大容量数据盘，并挂载文件系统：

1. 运行命令 `fdisk -l` 查看数据盘是否存在。返回结果应包括如下所示的信息。如果没有，表示您未挂载数据盘。

```
Disk /dev/vdd: 3221.2 GB, 3221225472000 bytes, 6291456000 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. 运行命令 `parted /dev/vdd` 开始分区：

- a. 运行命令 `mklabel gpt`，将MBR分区形式转为GPT分区形式。
- b. 运行命令 `mkpart primary ext4 <StartSector> <EndSector>`，划分一个采用ext4文件系统的主分区，并设置分区的开始位置和结束位置。如果一个数据盘只分一个分区，则运行命令 `mkpart primary ext4 0 -1`。



说明：

您也可以使用xfs文件系统。

- c. 运行命令 `print`，查看分区表。

```
(parted) mkpart primary ext4 0 -1
Warning: The resulting partition is not properly aligned for best
performance.
Ignore/Cancel? ignore
(parted) print
Model: Virtio Block Device (virtblk)
Disk /dev/vdd: 3221GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:
Number Start End Size File system Name Flags
1 17.4kB 3221GB 3221GB primary
```

- d. 运行命令 `quit`，退出 `parted` 操作。

3. 运行命令 `partprobe`，使系统重读分区表。

4. 运行以下命令，创建一个ext4文件系统，并使 `/dev/vdd1` 分区使用ext4。

```
mke2fs -O 64bit,has_journal,extents,huge_file,flex_bg,uninit_bg,
dir_nlink,extra_isize /dev/vdd1
```



说明：

- 如果您要关闭ext4文件系统的lazy init功能，避免该功能对数据盘I/O性能的影响，可以参考 [附录2#关闭lazy init功能](#)。
- 如果数据盘的容量为16 TiB，需要使用指定版本的e2fsprogs工具包格式化，请参考 [附录1#升级e2fsprogs工具包](#)。
- 如果您要创建一个xfs文件系统，运行命令 `mkfs -t xfs /dev/vdd1`。

5. 运行命令 `mkdir /test`，创建一个名为 `/test` 的挂载点。

6. 运行命令 `mount /dev/vdd1 /test`，将分区 `/dev/vdd1` 挂载到 `/test`。

7. 运行命令 `df -h`，查看目前磁盘空间和使用情况。

如果返回结果里出现新建文件系统的信息，说明挂载成功，可以使用新的文件系统了。挂载完成后，不需要重启实例即可开始使用新的文件系统。

```
[root@izXXXXX ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/vda1 40G 6.4G 31G 18% /
devtmpfs 487M 0 487M 0% /dev
tmpfs 497M 0 497M 0% /dev/shm
tmpfs 497M 364K 496M 1% /run
tmpfs 497M 0 497M 0% /sys/fs/cgroup
tmpfs 100M 0 100M 0% /run/user/0
/dev/vdd1 2.9T 89M 2.8T 1% /test
```

8. (可选) 向 `/etc/fstab` 写入新分区信息，启动开机自动挂载分区。

- (可选) 运行命令 `cp /etc/fstab /etc/fstab.bak`，备份 `etc/fstab`。
- 运行命令 `echo /dev/vdd1 /test ext4 defaults 0 0 >> /etc/fstab`，向 `/etc/fstab` 里写入新分区信息。
- 运行命令 `cat /etc/fstab`，查看 `/etc/fstab` 的信息。

如果返回结果里出现了写入的新分区信息，说明写入成功。

至此，您已经成功分区并格式化了一个 3 TiB 数据盘。

附录1：升级 `e2fsprogs` 工具包

如果数据盘容量为 16 TiB，您需要使用 1.42 及以上版本的 `e2fsprogs` 工具包完成 `ext4` 文件系统格式化。如果 `e2fsprogs` 版本太低（比如：`e2fsprogs 1.41.11`等），会出现如下错误信息：

```
mkfs.ext4: Size of device /dev/vdd too big to be expressed in 32 bits
using a blocksize of 4096.
```

您需要按以下方式安装高版本的 `e2fsprogs`，如本示例中使用的 1.42.8：

1. 运行命令 `rpm -qa | grep e2fsprogs` 检查 `e2fsprogs` 当前的版本。

```
$sudo rpm -qa | grep e2fsprogs
e2fsprogs-libs-1.41.12-3
e2fsprogs-1.41.12-3
e2fsprogs-libs-1.39-33.1.alios5
e2fsprogs-devel-1.39-33.1.alios5
```

如果当前版本低于 1.42，按以下步骤安装软件。

2. 运行以下命令下载 1.42.8 版本的e2fsprogs。您可以在 [e2fsprogs](#) 找到最新的软件包。

```
wget https://www.kernel.org/pub/linux/kernel/people/tytso/e2fsprogs/v1.42.8/e2fsprogs-1.42.8.tar.gz
```

3. 依次运行以下命令，编译高版本的工具。

```
tar xvzf e2fsprogs-1.42.8.tar.gz
cd e2fsprogs-1.42.8
./configure
make
make install
```

4. 运行命令 `rpm -qa | grep e2fsprogs` 检查是否成功安装高版本软件。

附录2：关闭lazy init功能

ext4文件系统的lazy init功能，默认开启。该功能开启时，系统后台会发起一个线程持续地初始化ext4文件系统的metadata，从而延迟metadata初始化。所以在刚格式化数据盘的一段时间内IOPS会受到影响，比如，数据盘的IOPS性能测试的数据会明显偏低。

如果要在格式化以后马上测试数据盘性能，您需要运行以下命令在格式化文件系统时关闭lazy_init功能。

```
mke2fs -O 64bit,has_journal,extents,huge_file,flex_bg,uninit_bg,
dir_nlink,extra_isize -E lazy_itable_init=0,lazy_journal_init=0 /dev
/vdd1
```

关闭lazy init功能后，格式化的时间会大幅度地延长，格式化32 TiB的数据盘可能需要10-30分钟。

请您根据自身的需要选择是否使用lazy init功能。

3.7 扩容云盘

3.7.1 扩容概述

在阿里云，随着业务的发展和应用数据增长，您能随时扩展云盘容量。

扩容场景

您可以通过以下方式增加单台实例的存储容量：

- [扩容已有云盘](#)，您需要自行扩展已有分区或者扩展新建分区。
- [创建一块新云盘](#)，作为数据盘[挂载](#)到实例上，并需要自行[分区格式化](#)。
- [更换系统盘](#)的同时指定更高的系统盘容量。

扩容云盘系列文档主要讲解扩容已有云盘的操作步骤和注意事项。

扩容云盘相关操作

根据云盘是否挂载以及是否分区，其操作选项如下表所示。

云盘状态分类	扩容未挂载未分区的云盘	扩容已挂载未分区的云盘	扩容已挂载并分区的云盘
扩容操作步骤	1. 在控制台或通过API (ResizeDisk) 扩容云盘。 2. 在控制台或通过API (AttachDisk) 挂载云盘。 3. 分区扩容或者格式化操作： - 分区格式化小于2 TiB的Windows数据盘 - 分区格式化小于2 TiB的Linux数据盘 - 分区格式化大于2 TiB云盘	1. 在控制台或通过API (ResizeDisk) 扩容云盘。 2. 在控制台或通过API (RebootInstance) 重启实例。 3. 执行分区格式化操作： - 分区格式化小于2 TiB的Windows数据盘 - 分区格式化小于2 TiB的Linux数据盘 - 分区格式化大于2 TiB云盘	1. 在控制台或通过API (ResizeDisk) 扩容云盘。 2. 在控制台或通过API (RebootInstance) 重启实例。 3. 分区扩容已挂载的系统盘或数据盘： - 扩展文件系统_Windows - 扩展文件系统_Linux系统盘 - 扩展文件系统_Linux数据盘

系统盘扩容上限

新值必须大于系统盘现有容量，小于等于500 GiB。系统盘容量限制和镜像的关系如下表所示。

镜像	扩容限制 (GiB)
Linux (不包括CoreOS) + FreeBSD	[max{20, 系统盘当前容量}, 500]
CoreOS	[max{30, 系统盘当前容量}, 500]
Windows	[max{40, 系统盘当前容量}, 500]

例如，一台CentOS实例扩容前系统盘容量为35 GiB。更换系统盘后，容量必须大于等于35 GiB，小于等于500 GiB。

数据盘扩容上限

新值必须大于数据盘现有容量。不同云盘类别的数据盘扩容上限如下表所示。

云盘类别	扩容前容量 (GiB)	扩容上限 (GiB)
普通云盘	< 2000	2000

云盘类别	扩容前容量 (GiB)	扩容上限 (GiB)
SSD云盘、高效云盘或ESSD云盘	< 2048	2048
SSD云盘、高效云盘或ESSD云盘	≥ 2048	不支持扩容

3.7.2 扩容云盘容量

随着业务的发展和应用数据增长，您可以参照本文描述扩容云盘容量，包括系统盘和数据盘。

使用说明

系统限制

- 系统盘和数据盘的扩容上限请参见[扩容概述](#)。
- 扩容云盘只是扩大存储容量，而不会扩容文件系统。请扩容后自行分配存储空间，更多详情，请参见[下一步](#)。

不支持项

- 不支持扩容正在创建快照的云盘。
- 包年包月实例[续费降配](#)后，当前计费周期的剩余时间内，不支持扩容实例的包年包月云盘。
- Windows Server 2003实例不支持系统盘扩容。

支持项

- 支持状态为使用中（In Use）的云盘，且所挂载的实例状态为运行中（Running）或已停止（Stopped）。
- 支持扩容类型为普通云盘、高效云盘、SSD云盘和ESSD云盘的云盘。
- 对于Windows实例，仅支持NTFS文件系统扩容。

准备工作

为防止扩容过程中误操作导致数据丢失，请[创建快照](#)以备份数据。

操作步骤

在ECS管理控制台上扩容云盘的操作路径如下。

1. 登录[ECS管理控制台](#)。
2. 选择地域。
3. 在左侧导航栏，单击存储 > 云盘。
4. 找到需要扩容的云盘，在操作列表中选择更多 > 磁盘扩容。
5. 设置扩容后的容量，变更后的容量不允许小于当前容量。

6. 确认费用，阅读服务条款，然后单击确定扩容。
7. 完成支付。
8. 在控制台或者使用API ([RebootInstance](#)) 重启实例使操作生效。

相关API：您也可以使用[ResizeDisk](#)完成磁盘扩容操作。

下一步

根据云盘是否挂载以及是否分区，扩容后的下一步操作选项如下表所示。

云盘状态分类	未挂载未分区的云盘	已挂载未分区的云盘	已挂载并分区的云盘
下一步	如果您的云盘是待挂载 (Available) 状态的数据盘，支付完成后扩容操作直接生效。您可以按以下步骤操作： 1. 在控制台或通过API (AttachDisk) 挂载云盘。 2. 分区扩容或者格式化操作： · 分区格式化小于2 TiB的Linux数据盘 · 分区格式化小于2 TiB的Windows数据盘 · 分区格式化大于2 TiB云盘	您可以按以下步骤操作： 1. 执行分区格式化操作： · 分区格式化小于2 TiB的Linux数据盘 · 分区格式化小于2 TiB的Windows数据盘 · 分区格式化大于2 TiB云盘 2. 执行文件系统扩容操作： · Linux实例请参见扩展文件系统-Linux系统盘或扩展文件系统-Linux数据盘 · Windows实例请参见扩展文件系统-Windows	执行分区扩容和文件系统扩展操作： · Linux实例请参见扩展文件系统-Linux系统盘或扩展文件系统-Linux数据盘 · Windows实例请参见扩展文件系统-Windows

3.7.3 扩展分区和文件系统_Windows

扩容云盘只是扩大存储容量，不会扩容文件系统。您需要参照本文描述自行格式化新增的存储容量，以便扩展存储空间。

使用限制

本文仅适用于状态为使用中的云盘，且所挂载的实例状态为运行中。对于待挂载数据盘，请参见[挂载云盘](#)和[分区格式化数据盘](#)。

准备工作

1. 通过ECS控制台或者API扩容云盘。
2. [创建快照](#)以备份数据。
3. 云盘已挂载到实例上，实例已处于运行中状态。
4. 数据盘已完成分区格式化。具体操作请参见[分区格式化数据盘](#)。

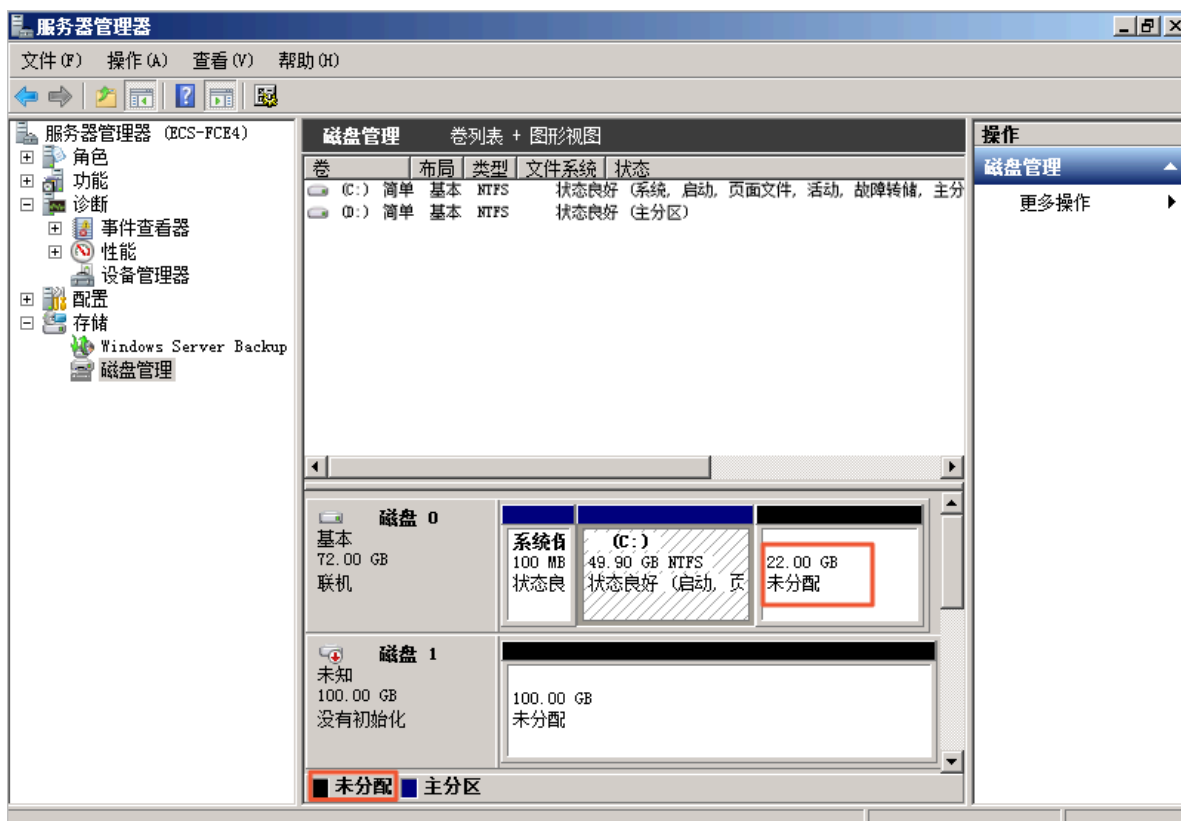
扩展系统盘分区

在ECS控制台上扩容系统盘后，对应系统盘分区的文件系统并未扩容。您需要连接实例扩容文件系统。本示例中以Windows Server 2008 R2企业版64位中文版操作系统为例。扩容前的系统盘容量为50 GiB，扩容为72 GiB，文件系统类型为NTFS。

1. [远程连接Windows实例](#)。
2. 打开服务器管理器。
3. 在左侧导航栏中，选择存储 > 磁盘管理。
4. 选择操作 > 刷新或操作 > 重新扫描磁盘。



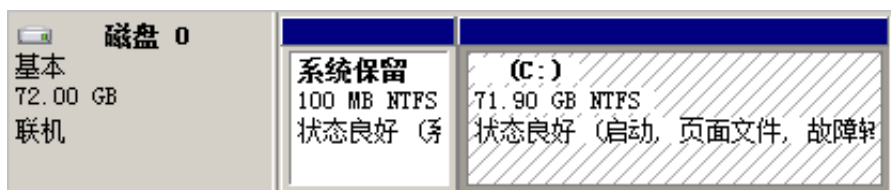
5. 在磁盘管理区域，查看未分配容量。本例中，磁盘 0 是扩容的系统盘。



6. 右键单击磁盘 0 主分区的空白处，并选择扩展卷。

7. 根据扩展卷向导的指示完成扩展卷操作。

完成后，新增空间会自动合入原来的卷中，如下图所示。



扩展数据盘分区

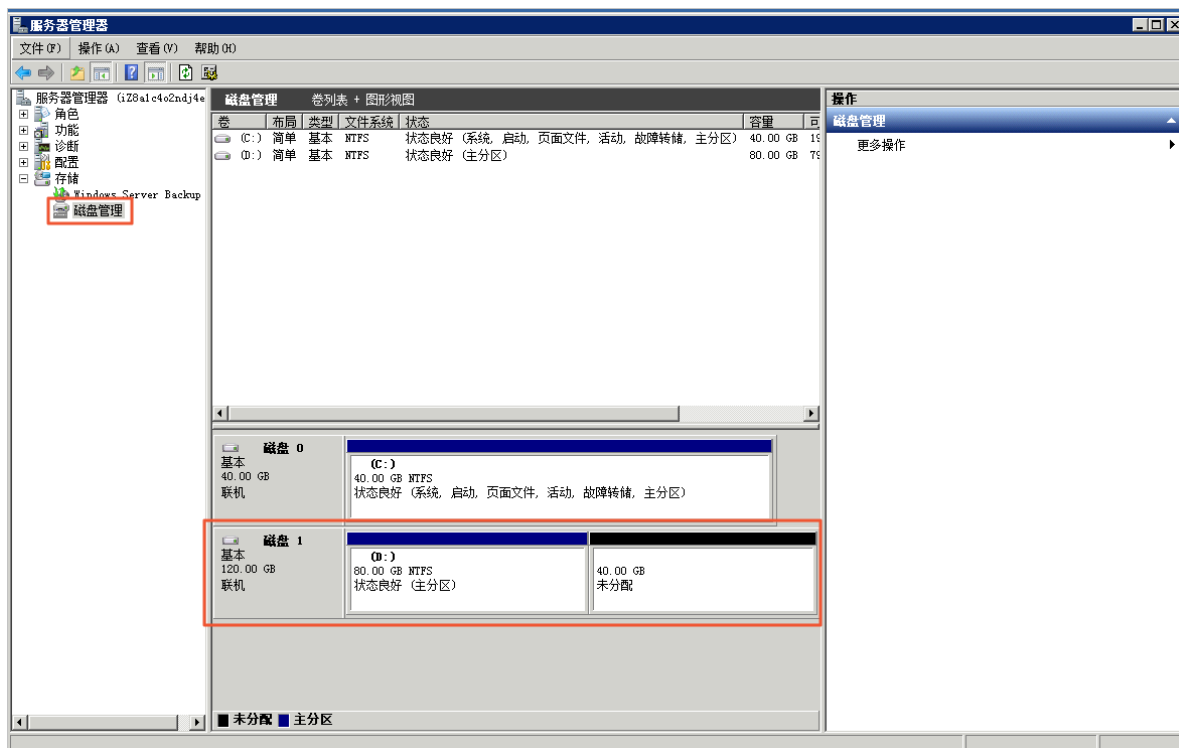
在ECS控制台上扩容数据盘后，数据盘的每个分区的文件系统并未扩容。您需要连接实例扩容文件系统。本示例中以Windows Server 2008 R2企业版64位中文版操作系统为例。扩容前的数据盘容量为80 GiB，扩容为120 GiB，文件系统类型为NTFS。

1. [远程连接Windows实例](#)。
2. 打开服务器管理器。
3. 在左侧导航栏中，选择存储 > 磁盘管理。

4. 选择操作>刷新或操作>重新扫描磁盘。



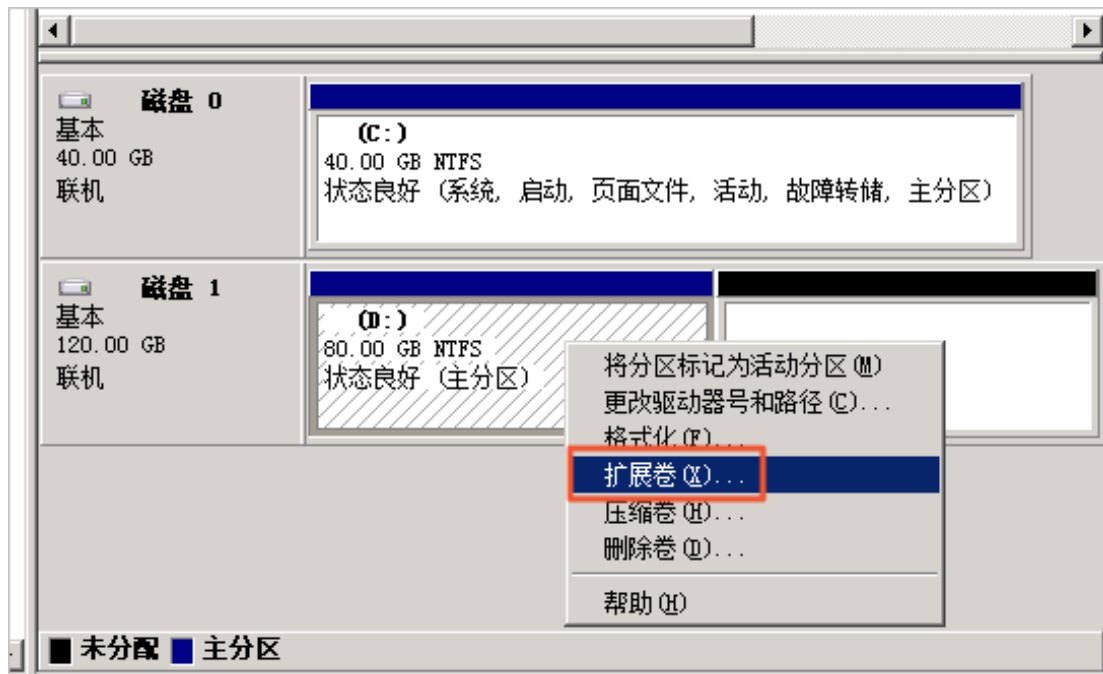
5. 在磁盘管理区域，查看未分配的数据盘容量。本例中，磁盘 1 是扩容的数据盘。



6. 扩展磁盘 1。

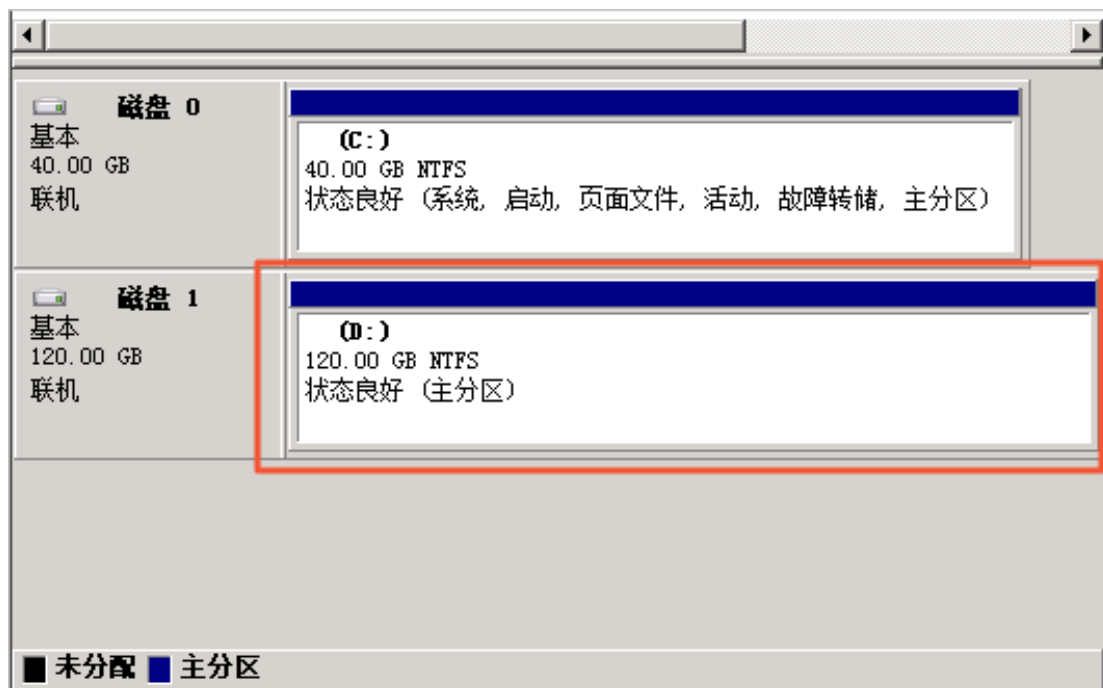
- 选项一：如果新磁盘空间用于扩容已有的分区，按照以下步骤完成扩容：

a. 右键单击磁盘 1 主分区的空白处，并选择扩展卷。



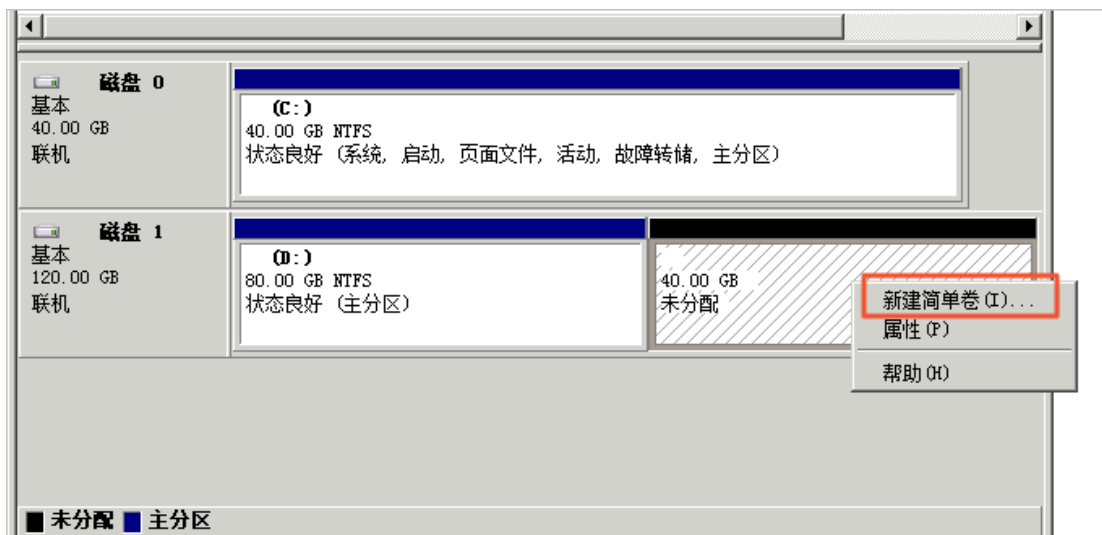
b. 根据扩展卷向导的指示完成扩展卷操作。

完成后，新增的数据盘空间会自动合入原来的卷中。



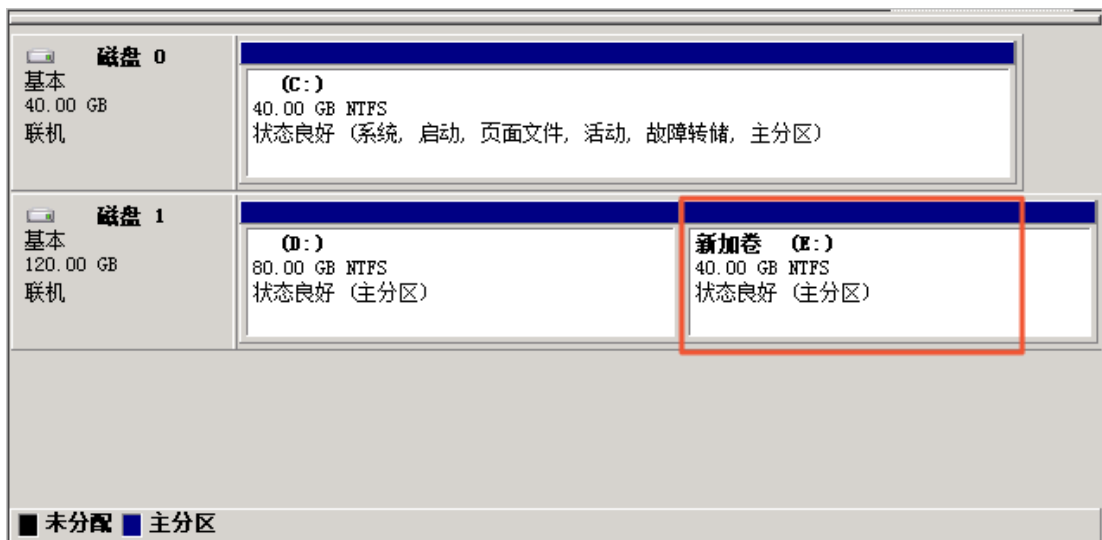
- 选项二：如果新磁盘空间用于增加新的分区，按照以下步骤完成扩容：

a. 右键单击磁盘 1 未分配区的空白处，并选择新建简单卷。



b. 根据新建简单卷向导的指示完成扩展卷操作。

完成后，新增的数据盘空间会新建一个分区。



3.7.4 扩展分区与文件系统_Linux系统盘

本文提供了如何使用growpart和resize2fs工具完成Linux系统盘分区扩容及文件系统扩展的操作指导。

适用范围

本文的操作步骤适用于以下分区和文件系统格式的云盘：

- 分区格式支持：mbr、gpt
- 文件系统支持：ext、xfs、btrfs、ufs

准备工作

1. 通过ECS控制台或者API扩容云盘。

2. [创建快照](#)以备份数据。
3. 实例已处于运行中状态。连接方式请参见[连接方式导航](#)。
4. 根据操作系统安装growpart扩容格式化工具。

- CentOS 7、Aliyun Linux:

```
yum install cloud-utils-growpart
```

- Ubuntu 14、Ubuntu 16、Ubuntu 18、Debian 9:

```
apt install cloud-guest-utils
```

- Debian 8、OpenSUSE 42.3、OpenSUSE 13.1、SUSE Linux Enterprise Server 12 SP2: 请使用上游版本 (upstream) 的growpart工具

5. 检查实例的内核版本, 如通过uname -a查看内核版本。

- 内核版本大于3.6.0, 则无需重启reboot便能完成扩容分区和文件系统。该情况请参见[高内核版本的操作步骤](#)。
- 内核版本小于3.6.0, 如CentOS 6、Debian 7和SUSE Linux Enterprise Server 11 SP4等发行版, 需要经过一次重启reboot才能完成分区扩容。该情况请参见[低内核版本的操作步骤](#)。

高内核版本的操作步骤

此处以CentOS 7操作系统为例演示分区扩展的步骤。

1. 运行fdisk -l查看现有磁盘大小。示例返回磁盘 (/dev/vda) 容量是100 GiB。

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

Disk /dev/vda: 107.4 GB, 107374182400 bytes, 209715200 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: 0x0008d73a

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

2. 运行df -h查看磁盘分区大小。示例返回分区容量 (/dev/vda1) 是20 GiB。

```
[root@localhost ~]# df -h

Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       20G   1.5G   18G   8% /
devtmpfs        7.8G     0   7.8G   0% /dev
tmpfs           7.8G     0   7.8G   0% /dev/shm
tmpfs           7.8G  344K   7.8G   1% /run
tmpfs           7.8G     0   7.8G   0% /sys/fs/cgroup
```

```
tmpfs          1.6G      0  1.6G    0% /run/user/0
```

- 运行 `growpart <DeviceName> <PartionNumber>` 调用 `growpart` 为需要扩容的磁盘和对应的第几个分区扩容。示例命令表示为系统盘的第一个分区扩容。

```
[root@localhost ~]# growpart /dev/vda 1

CHANGED: partition=1 start=2048 old: size=41940992 end=41943040 new
: size=209710462,end=209712510
```

- 运行 `resize2fs <PartitionName>` 调用 `resize2fs` 扩容文件系统。示例命令表示为系统盘的 `/dev/vda1` 分区扩容文件系统。

```
[root@localhost ~]# resize2fs /dev/vda1

resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/vda1 is mounted on /; on-line resizing required
old_desc_blocks = 2, new_desc_blocks = 7
The filesystem on /dev/vda1 is now 26213807 blocks long.
```

- 运行 `df -h` 查看磁盘分区大小。返回分区 (`/dev/vda1`) 是 100 GiB，表示已经成功扩容。

```
[root@localhost ~]# df -h

Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       99G   1.6G   93G   2% /
devtmpfs        7.8G     0   7.8G   0% /dev
tmpfs           7.8G     0   7.8G   0% /dev/shm
tmpfs           7.8G  500K   7.8G   1% /run
tmpfs           7.8G     0   7.8G   0% /sys/fs/cgroup
tmpfs           1.6G     0   1.6G   0% /run/user/0
```

低内核版本的操作步骤

此处以 CentOS 6 操作系统为例演示分区扩展的步骤。

- 安装 `dracut-modules-growroot` 工具。

```
[root@AliYunOS ~]# yum install -y dracut-modules-growroot
```

如果您使用的是其他软件包管理器，请将 `yum` 修改为对应的命令。

- 覆盖已有的 `initramfs` 文件。

```
[root@AliYunOS ~]# dracut -f
```

- 运行 `fdisk -l` 查看现有磁盘大小。示例返回磁盘 (`/dev/vda`) 容量是 100 GiB。

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

Disk /dev/vda: 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: 0x0003a7b4
```

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1	*	1	2611	20970496	83	Linux

4. 运行 `df -h` 查看磁盘分区大小。示例返回分区容量 (/dev/vda1) 是 20 GiB。

```
[root@AliYunOS ~]# df -h

Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1        20G   1.1G   18G   6% /
tmpfs            7.8G     0   7.8G   0% /dev/shm
```

5. 运行 `growpart <DeviceName><PartionNumber>` 调用 `growpart` 为需要扩容的磁盘和对应的第几个分区扩容。示例命令表示为系统盘的第一个分区扩容。

```
[root@AliYunOS ~]# growpart /dev/vda 1

CHANGED: partition=1 start=2048 old: size=41940992 end=41943040 new
: size=209710462,end=209712510
```

6. 重启实例。

```
[root@AliYunOS ~]# reboot
```

7. 再次远程连接实例。

8. 运行 `resize2fs <PartitionName>` 调用 `resize2fs` 扩容文件系统。示例命令表示为系统盘的 /dev/vda1 分区扩容文件系统。

```
[root@AliYunOS ~]# resize2fs /dev/vda1

resize2fs 1.41.12 (17-May-2010)
Filesystem at /dev/vda1 is mounted on /; on-line resizing required
old desc_blocks = 2, new_desc_blocks = 7
Performing an on-line resize of /dev/vda1 to 26213807 (4k) blocks.
The filesystem on /dev/vda1 is now 26213807 blocks long.
```

9. 运行 `df -h` 查看磁盘分区大小。返回分区 (/dev/vda1) 是 100 GiB，表示已经成功扩容。

```
[root@AliYunOS ~]# df -h

Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1        99G   1.1G   93G   2% /
tmpfs            7.8G     0   7.8G   0% /dev/shm
```

3.7.5 扩展分区与文件系统_Linux数据盘

扩容云盘只是扩大存储容量，不会扩容文件系统。您需要参照本文描述自行格式化新增的存储容量，以便扩展存储空间。

使用限制

本文仅适用于扩容前已挂载到实例的数据盘。对于待挂载数据盘，请参见[挂载云盘](#)和[分区格式化数据盘](#)。

扩容指引

表 3-1: Linux实例数据盘的不同扩容场景

扩容场景		相关操作
数据盘已挂载到实例上	数据盘已分区并格式化	参考本文档的操作步骤完成扩容。
	空数据盘，未分区，未格式化	在控制台扩容磁盘空间后， 分区并格式化数据盘 ，然后参考本文档的操作步骤完成扩容。
	数据盘已格式化，未分区	在控制台扩容磁盘空间后， 扩容文件系统 。
数据盘未挂载到实例上		您可以选择离线扩容数据盘，或者挂载数据盘到实例后再扩容。

本示例中，数据盘为高效云盘，ECS实例的操作系统为CentOS 7.5 64 位。数据盘挂载点为/dev/vdb，已有分区为/dev/vdb1。扩容前的数据盘容量为20 GiB，扩容为40 GiB。已挂载到实例上并操作过分区格式化的数据盘有如下选项。

- 如果您需要扩展数据盘已有分区，请参见[选项一#扩展已有分区](#)。
- 如果新磁盘空间用于增加新分区，请参见[选项二#新增并格式化分区](#)。

准备工作

1. 通过ECS控制台或者API扩容云盘。
2. [创建快照](#)以备份数据。
3. 实例已处于运行中状态。连接方式请参见[连接方式导航](#)。

确认分区和文件系统

1. 运行 `fdisk -lu <DeviceName>` 确认磁盘是否分区。

本示例中，原有的磁盘空间已做分区/dev/vdb1。

```
[root@localhost ~]# fdisk -lu /dev/vdb

Disk /dev/vdb: 42.9 GB, 42949672960 bytes, 83886080 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: 0x9277b47b

Device Boot Start End Blocks Id System
```

```
/dev/vdb1 2048 41943039 20970496 83 Linux
```

2. 运行 `blkid <PartitionName>` 确认文件系统的类型。

本示例中，`/dev/vdb1` 的文件系统类型为 `ext4`。

```
[root@localhost ~]# blkid /dev/vdb1
/dev/vdb1: UUID="e97bf1e2-fc84-4c11-9652-73554546c324" TYPE="ext4"
```



说明：

未分区并且未格式化的数据盘，以及已分区但未格式化的数据盘，不会返回结果。

3. 运行以下命令确认文件系统的状态。

- `ext` 系列文件系统：`e2fsck -n <dst_dev_part_path>`
- `xfs` 文件系统：`xfs_repair -n <dst_dev_part_path>`

本示例中，文件系统状态为 `clean`。如果状态不是 `clean`，请排查并修复。

```
[root@localhost ~]# e2fsck -n /dev/vdb1
e2fsck 1.42.9 (28-Dec-2013)
Warning! /dev/vdb1 is mounted.
Warning: skipping journal recovery because doing a read-only
filesystem check.
/dev/vdb1: clean, 11/1310720 files, 126322/5242624 blocks
```



说明：

如果检查后发现数据盘还未操作过分区格式化，请完成分区初始化工作。具体操作步骤参见[格式化数据盘-Linux](#)。

选项一：扩展已有分区

为了防止数据丢失，不建议扩容已挂载（`mount`）的分区和文件系统。请先取消挂载（`umount`）分区，完成扩容并正常使用后，重新挂载（`mount`）。如果确实需要扩容已挂载（`mount`）的分区，则操作如下：

- 实例内核版本 < 3.6 ：先取消挂载该分区，再修改分区表，最后扩容文件系统。
- 实例内核版本 ≥ 3.6 ：先修改对应分区表，再通知内核更新分区表，最后扩容文件系统。

如果新磁盘空间用于扩容已有的分区，按照以下步骤在实例中完成扩容：

步骤一：修改分区表

1. 运行 `fdisk -lu /dev/vdb`，并记录旧分区的起始和结束的扇区位置。

本示例中，`/dev/vdb1` 的起始扇区位置是2048，结束扇区位置是41943039。

```
[root@localhost ~]# fdisk -lu /dev/vdb

Disk /dev/vdb: 42.9 GB, 42949672960 bytes, 83886080 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: 0x9277b47b

Device Boot Start End Blocks Id System
/dev/vdb1 2048 41943039 20970496 83 Linux
```

2. 使用 `fdisk` 命令删除旧分区。

a. 运行 `fdisk -u /dev/vdb`：分区数据盘。

b. 输入 `p`：打印分区表。

c. 输入 `d`：删除分区。

d. 输入 `p`：确认分区已删除。

e. 输入 `w`：保存修改并退出。

```
[root@localhost ~]# fdisk -u /dev/vdb

Welcome to fdisk (util-linux 2.23.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
Command (m for help): p
Disk /dev/vdb: 42.9 GB, 42949672960 bytes, 83886080 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: 0x9277b47b
Device Boot Start End Blocks Id System
/dev/vdb1 2048 41943039 20970496 83 Linux
Command (m for help): d
Selected partition 1
Partition 1 is deleted
Command (m for help): p
Disk /dev/vdb: 42.9 GB, 42949672960 bytes, 83886080 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: 0x9277b47b
Device Boot Start End Blocks Id System
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
WARNING: Re-reading the partition table failed with error 16: Device
or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
```

Syncing disks.

3. 使用fdisk命令新建分区。

- a. 运行 `fdisk -u /dev/vdb`：分区数据盘。
- b. 输入 `p`：打印分区表。
- c. 输入 `n`：新建分区。
- d. 输入 `p`：选择分区类型为主分区。
- e. 输入 `<###>`：选择分区号。本示例选取了1。



警告：

- 新分区的起始位置，必须和旧分区的起始位置相同。
- 新分区的结束位置，必须大于旧分区的结束位置，否则会导致扩容失败。

- f. 输入 `w`：保存修改并退出。

本示例中，将 `/dev/vdb1` 由20 GiB扩容到40 GiB。

```
[root@localhost ~]# fdisk -u /dev/vdb

Welcome to fdisk (util-linux 2.23.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
Command (m for help): p
Disk /dev/vdb: 42.9 GB, 42949672960 bytes, 83886080 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: 0x9277b47b
Device Boot Start End Blocks Id System
Command (m for help): n
Partition type:
p primary (0 primary, 0 extended, 4 free)
e extended
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-83886079, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-83886079, default
83886079):
Partition 1 of type Linux and of size 30 GiB is set
Command (m for help): p
Disk /dev/vdb: 42.9 GB, 42949672960 bytes, 83886080 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: 0x9277b47b
Device Boot Start End Blocks Id System
/dev/vdb1 2048 62916607 31457280 83 Linux
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
```



```
Syncing disks.
```

4. 运行 `lsblk /dev/vdb` 确保分区表已经增加。

5. 运行 `e2fsck -n /dev/vdb1` 再次检查文件系统，确认扩容分区后的文件系统状态为 `clean`。

步骤二：通知内核更新分区表

运行 `partprobe <dst_dev_path>` 或者 `partx -u <dst_dev_path>`，以通知内核磁盘的分区表已经修改，需要同步更新。

步骤三：扩容文件系统

- ext 系列文件系统（例如 `ext3` 和 `ext4`）：运行 `resize2fs /dev/vdb1`。

```
[root@localhost ~]# resize2fs /dev/vdb1
resize2fs 1.42.9 (28-Dec-2013)
Resizing the filesystem on /dev/vdb1 to 7864320 (4k) blocks.
The filesystem on /dev/vdb1 is now 7864320 blocks long.
```

- xfs 文件系统：先运行 `mount /dev/vdb1 /mnt/` 命令，再运行 `xfs_growfs /dev/vdb1`。

```
[root@localhost ~]# mount /dev/vdb1 /mnt/
[root@localhost ~]# xfs_growfs /dev/vdb1

meta-data=/dev/vdb1          isize=512    agcount=4, agsize=
1310720 blks
                =          sectsz=512    attr=2, projid32bit=1
                =          crc=1        finobt=0 spinodes=0
data        =          bsize=4096      blocks=5242880,
imaxpct=25
                =          sunit=0      swidth=0 blks
naming      =version 2        bsize=4096  ascii-ci=0 ftype=1
log         =internal        bsize=4096  blocks=2560, version=2
                =          sectsz=512   sunit=0 blks, lazy-
count=1
realtime    =none            extsz=4096   blocks=0, rtextents=0
data blocks changed from 5242880 to 7864320
```

选项二：新增并格式化分区

如果新磁盘空间用于增加新的分区，按照以下步骤在实例中完成扩容：

1. 运行 `fdisk -u /dev/vdb` 命令新建分区。

本示例中，为新增的 20GiB 新建分区，作为 `/dev/vdb2`。

```
[root@localhost ~]# fdisk -u /dev/vdb

Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): p

Disk /dev/vdb: 42.9 GB, 42949672960 bytes, 83886080 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: 0x2b31a2a3

   Device Boot      Start         End      Blocks   Id  System
/dev/vdb1           2048     41943039     20970496   83   Linux

Command (m for help): n
Partition type:
   p   primary (1 primary, 0 extended, 3 free)
   e   extended
Select (default p): p
Partition number (2-4, default 2): 2
First sector (41943040-83886079, default 41943040):
Using default value 41943040
Last sector, +sectors or +size{K,M,G} (41943040-83886079, default
83886079):
Using default value 83886079
Partition 2 of type Linux and of size 20 GiB is set

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

Calling ioctl() to re-read partition table.
Syncing disks.
```

2. 运行命令 `lsblk /dev/vdb` 查看分区。

```
[root@localhost ~]# lsblk /dev/vdb

NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
vdb          253:16  0   40G  0 disk
├─vdb1       253:17  0    20G  0 part
└─vdb2       253:18  0    20G  0 part
```

3. 格式化新的分区。

- 创建ext4文件系统: `mkfs.ext4 /dev/vdb2`

```
[root@localhost ~]# mkfs.ext4 /dev/vdb2

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
1310720 inodes, 5242880 blocks
262144 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2153775104
160 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,
    1605632, 2654208,
    4096000

Allocating group tables: done
```

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

```
[root@localhost ~]# blkid /dev/vdb2
/dev/vdb2: UUID="e3f336dc-d534-4fdd-af71-b6ff1a55bdbb" TYPE="ext4"
```

- 创建ext3文件系统: `mkfs.ext3 /dev/vdb2`

```
[root@localhost ~]# mkfs.ext3 /dev/vdb2

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
1310720 inodes, 5242880 blocks
262144 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
160 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,
    1605632, 2654208,
    4096000

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

[root@localhost ~]# blkid /dev/vdb2
/dev/vdb2: UUID="dd5be97d-a630-4593-9b0f-5056def914ea" SEC_TYPE="
ext2" TYPE="ext3"
```

- 创建xfs文件系统: `mkfs.xfs -f /dev/vdb2`

```
[root@localhost ~]# mkfs.xfs -f /dev/vdb2

meta-data=/dev/vdb2            isize=512    agcount=4, agsize=
1310720 blks
                        =               sectsz=512    attr=2, projid32bit=
1                               =               crc=1      finobt=0, sparse=0
data                =               bsize=4096    blocks=5242880,
imaxpct=25          =               sunit=0      swidth=0 blks
naming              =version 2      bsize=4096    ascii-ci=0 ftype=1
log                 =internal log   bsize=4096    blocks=2560, version
=2                      =               sectsz=512    sunit=0 blks, lazy-
count=1              =               extsz=4096    blocks=0, rtextents=
realtime =none                                     blocks=0, rtextents=
0

[root@localhost ~]# blkid /dev/vdb2
```

```
/dev/vdb2: UUID="66251477-3ae4-4b44-8b21-5604420dbecb" TYPE="xfs"
```

- 创建trfs文件系统: `mkfs.btrfs /dev/vdb2`

```
[root@localhost ~]# mkfs.btrfs /dev/vdb2

btrfs-progs v4.9.1
See http://btrfs.wiki.kernel.org for more information.

Label:                      (null)
UUID:                      6fb5779b-57d7-4aaf-bf09-82b46f54a429
Node size:                  16384
Sector size:                4096
Filesystem size:            20.00GiB
Block group profiles:
  Data:                     single                8.00MiB
  Metadata:                 DUP                    1.00GiB
  System:                   DUP                    8.00MiB
SSD detected:               no
Incompat features:          extref, skinny-metadata
Number of devices:          1
Devices:
   ID     SIZE  PATH
   1     20.00GiB /dev/vdb2

[root@localhost ~]# blkid /dev/vdb2
/dev/vdb2: UUID="6fb5779b-57d7-4aaf-bf09-82b46f54a429" UUID_SUB="
9bdd889a-ab69-4653-a583-d1b6b8723378" TYPE="btrfs"
```

4. 运行 `mount /dev/vdb2 /mnt` 挂载文件系统。

5. 运行 `df -h` 查看目前磁盘空间和使用情况。

显示新建文件系统的信息，表示挂载成功，您可以使用新的文件系统。

```
[root@localhost ~]# df -h

Filesystem Size Used Avail Use% Mounted on
/dev/vda1 40G 1.6G 36G 5% /
devtmpfs 3.9G 0 3.9G 0% /dev
tmpfs 3.9G 0 3.9G 0% /dev/shm
tmpfs 3.9G 460K 3.9G 1% /run
tmpfs 3.9G 0 3.9G 0% /sys/fs/cgroup
/dev/vdb2 9.8G 37M 9.2G 1% /mnt
tmpfs 783M 0 783M 0% /run/user/0
```

3.8 更换系统盘

3.8.1 更换系统盘（非公共镜像）

更换系统盘是指为您重新分配一个系统盘（系统盘ID会更新），原来的系统盘会被释放。如果在创建ECS实例时选择了错误的操作系统，或者随着业务发展，需要扩大系统盘容量或者使用其他操作系统，您都能通过更换系统盘实现。

您能将系统盘的镜像更换为公共镜像、共享镜像、自定义镜像或镜像市场里的任一种镜像。

**说明:**

微软已经停止支持Windows Server 2003。为了保证您的数据安全，阿里云不推荐ECS实例继续使用Windows Server 2003系统，也不再提供这个镜像。更多信息，请参见[关于阿里云不再支持Windows Server 2003系统镜像](#)。

更换系统盘后，

- 您的实例会被分配一个新的系统盘，系统盘ID会更新，原系统盘被释放。
- 系统盘的云盘类型不能更换。
- 实例的IP地址和MAC地址不变。
- 为了保证有足够的快照额度完成新系统盘的自动快照策略，您可以[删除不需要的旧系统盘快照](#)。

本文介绍如何将系统盘的镜像更换为非公共镜像。如果您需要使用公共镜像，请参见[更换系统盘#公共镜像#](#)。

注意事项

更换系统盘是一个高风险操作，所以，在更换系统盘之前，请务必认真阅读以下注意事项。

风险

更换系统盘存在如下风险：

- 原系统盘的会被释放，建议您提前[创建快照](#)备份数据。
- 更换系统盘需要停止实例，因此会中断您的业务。
- 更换完成后，您需要在新的系统盘中重新部署业务运行环境，有可能会对您的业务造成长时间的中断。
- 更换系统盘是重新为您的实例分配一个系统盘，磁盘ID会改变，所以基于旧的系统盘创建的快照将不能用于回滚新的系统盘。

**说明:**

更换系统盘后，您手动创建的快照不受影响，您仍能用这些快照创建自定义镜像。如果您为旧的系统盘设置了自动快照策略，而且设置了自动快照随磁盘释放，则自动快照策略不再适用于新的系统盘，而且，旧系统盘的自动快照会自动删除。

跨操作系统更换注意事项

跨操作系统更换是指互相更换Windows与Linux系统。

**说明:**

非中国大陆的地域暂不支持Linux和Windows系统的互换，仅支持Linux不同发行版本或Windows不同版本之间的更换。

跨操作系统更换时，数据盘的文件系统格式可能会无法识别。

- 如果您的数据盘没有重要数据，建议您[重新初始化数据盘](#)，再将其格式化为相应系统的默认文件系统。
- 如果您的数据盘上有重要数据，您需要做以下操作：
 - Windows系统更换为Linux系统：需要单独安装软件识别，例如NTFS-3G等，因为Linux缺省情况下无法识别NTFS格式。
 - Linux系统更换为Windows系统：需要单独安装软件识别，例如Ext2Read、Ext2Fsd等，因为Windows缺省情况下无法识别ext3、ext4、XFS等文件系统格式。

如果您将操作系统从Windows系统换为Linux系统，您能选择使用密码认证或SSH密钥对认证。

准备工作

将系统盘的镜像更换为非公共镜像之前，您必须先完成以下工作：

- 如果更换为自定义镜像：
 - 如果要使用某台ECS实例上的镜像，您必须[对指定实例的系统盘创建快照](#)，并[使用快照创建自定义镜像](#)。如果这台实例与您要更换系统盘的实例不在同一个地域，您需要[复制镜像](#)。
 - 如果要使用本地物理镜像文件，您需要[在控制台导入镜像](#)或者[使用Packer创建并导入本地镜像](#)。镜像所在地域必须与您的实例相同。
 - 如果要使用其他地域的镜像，您需要先[复制镜像](#)。



说明：

通过上述方式得到的镜像，在更换系统盘时，都出现在自定义镜像的下拉列表里。

- 如果要使用其他阿里云账号的镜像，您必须先[共享镜像](#)。
- 如果更换为Linux系统，而且选择SSH密钥对认证，您必须先[创建SSH密钥对](#)。
- 因为更换系统盘是一个高危操作，可能会造成数据丢失或业务停止，为了最大限度地减少更换系统盘对您业务的影响，建议在更换系统盘之前先为当前系统盘[创建快照](#)。
- 如果要对Windows系统盘做更换，请确保系统盘有足够的空间。建议预留1 GiB的空间，否则系统盘更换后，系统可能无法正常启动。



说明：

避免在您的业务高峰期创建快照，以免影响业务。一个40 GiB的快照，第一次创建大约需要40分钟。因此请预留出足够的时间。而且，创建快照可能会降低块存储I/O性能（一般在10%以内），出现短暂瞬间变慢。

操作步骤

1. 登录[ECS管理控制台](#)。
2. 在左侧导航栏，选择实例与镜像 > 实例。
3. 在顶部状态栏处，选择地域。
4. 找到需要更换系统盘的实例，在操作列中，选择更多 > 实例状态 > 停止。

当实例状态变为已停止时，表示实例已经成功停止。

5. 在操作列中，选择更多 > 磁盘和镜像 > 更换系统盘。
6. 在弹出的对话框里，仔细阅读更换系统盘注意事项后，单击确定，更换系统盘。
7. 在更换系统盘页面上，
 - a. 镜像类型：选择自定义镜像、共享镜像或镜像市场，并选择需要的镜像。
 - b. 系统盘：不能更换系统盘的云盘类型，但是您能根据业务需求和新镜像的需求扩容系统盘，最大容量为500 GiB。扩容时能设置的最小容量与系统盘当前容量和镜像有关，如下表所示。

镜像	扩容的容量限制（GiB）
Linux（不包括CoreOS）+ FreeBSD	[Max{20, 系统盘当前容量}, 500]
CoreOS	[Max{30, 系统盘当前容量}, 500]
Windows	[Max{40, 系统盘当前容量}, 500]



说明：

如果您做过[续费降配](#)操作，进入下一个计费周期前，您不能修改系统盘容量。

- c. 安全设置：
 - 如果新的操作系统是Windows系统，您只能使用密码认证。
 - 如果您的实例是I/O优化实例，而且新的操作系统是Linux系统，您能选择使用密码认证或SSH密钥对认证。您必须设置登录密码或者绑定SSH密钥对。
- d. 确认配置费用：包括镜像的价格与系统盘的价格。请详见[云产品价格页](#)。
- e. 确认无误后，单击确定更换。

登录ECS控制台监控系统状态，完成操作系统更换大概需要10分钟。完成后，实例会自动启动。

后续操作

更换系统盘后，您可能需要做以下操作：

- （可选）[为新的系统盘设置自动快照策略](#)。自动快照策略与磁盘ID绑定。更换了新的系统盘后，旧磁盘上应用的自动快照策略自动失效。您需要对新系统盘设置自动快照策略。
- 如果更换前后都是Linux系统，而且，实例上原来挂载了数据盘并设置了开机自动挂载分区：更换系统盘后，原来系统盘中的数据盘分区挂载信息丢失。您必须在新系统盘的/etc/fstab文件写入新分区信息，并挂载分区，不需要对数据盘格式化并分区。操作步骤如下，具体的操作命令参考[Linux格式化和挂载数据盘](#)：

1. （建议）备份etc/fstab。
2. 向/etc/fstab写入新分区信息。
3. 查看/etc/fstab中的新分区信息。
4. 运行mount命令挂载分区。
5. 查看文件系统空间和使用情况：运行命令df -h。

挂载操作完成后，不需要重启实例即可开始使用新的数据盘。

相关API

[ReplaceSystemDisk](#)

3.8.2 更换系统盘（公共镜像）

更换系统盘是指为您重新分配一块系统盘，系统盘ID会更新，原来的系统盘会被释放。

应用场景

如果您在创建ECS实例时选择了错误的操作系统，或者随着业务发展，需要使用其他操作系统，您能通过更换系统盘来更换操作系统。您能将系统盘的镜像更换为公共镜像、共享镜像、自定义镜像或镜像市场里的任一种镜像。

更换系统盘后：

- 为实例重新分配一块系统盘，磁盘ID会更新，原系统盘被释放。
- 系统盘的云盘类型不变。
- 实例的IP地址和MAC地址不变。
- 为了保证有足够的快照额度完成新系统盘的自动快照策略，您可以[删除不需要的旧系统盘快照](#)。

本文介绍如何将系统盘的镜像更换为公共镜像。如果您需要使用非公共镜像，请参见[更换系统盘#非公共镜像#](#)。

注意事项



说明:

微软已经停止支持Windows Server 2003。为了保证您的数据安全，阿里云不推荐ECS实例继续使用Windows Server 2003系统，也不再提供该镜像。更多信息，请参见[关于阿里云不再支持Windows Server 2003公共镜像](#)。

更换系统盘是高风险操作，在更换系统盘之前，请务必认真阅读以下注意事项。

风险

- 原系统盘的会被释放，建议您提前[创建快照](#)备份数据。
- 更换系统盘需要停止实例，因此会中断您的业务。
- 更换完成后，您需要在新的系统盘中重新部署业务运行环境，有可能会对您的业务造成长时间的中断。
- 更换系统盘后，磁盘ID会改变，历史系统盘快照不能用于回滚新的系统盘。



说明:

更换系统盘后，您手动创建的快照不受影响，您仍能用这些快照创建自定义镜像。如果您为旧的系统盘设置了自动快照策略，而且设置了自动快照随磁盘释放，则自动快照策略不再适用于新的系统盘。而且，旧系统盘的自动快照会自动删除。

跨操作系统更换注意事项

跨操作系统更换是指互相更换Windows与Linux系统。



说明:

非中国大陆的地域暂不支持Linux和Windows系统的互换，仅支持Linux不同发行版本或Windows不同版本之间的更换。

跨操作系统更换时，数据盘的文件系统格式可能会无法识别。

- 如果您的数据盘没有重要数据，建议您[重新初始化数据盘](#)，再将其格式化为相应系统的默认文件系统。
- 如果您的数据盘上有重要数据，您需要做以下操作：
 - Windows系统更换为Linux系统：需要单独安装软件识别，例如NTFS-3G等。Linux缺省情况下无法识别NTFS格式。
 - Linux系统更换为Windows系统：需要单独安装软件识别，例如Ext2Read、Ext2Fsd等。Windows缺省情况下无法识别ext3、ext4、XFS等文件系统格式。

准备工作

- 如果要对Windows系统盘做更换，请确保系统盘有足够的空间。建议预留1 GiB的空间，否则系统盘更换后，系统可能无法正常启动。
- 如果要更换为Linux系统，而且选择SSH密钥对认证，您必须先[创建SSH密钥对](#)。
- 更换系统盘是高危操作，可能会造成数据丢失或业务停止，为了最大限度地减少更换系统盘对您业务的影响，建议在更换系统盘之前先为当前系统盘[创建快照](#)。



说明：

避免在您的业务高峰期创建快照，以免影响业务。一份40 GiB的快照，第一次创建大约需要40分钟。因此请预留出足够的时间。

操作步骤

1. 登录[ECS管理控制台](#)。
2. 在左侧导航栏，选择实例与镜像 > 实例。
3. 在顶部状态栏处，选择地域。
4. 找到需要更换系统盘的实例，在操作列中，选择更多 > 实例状态 > 停止，并按页面提示停止实例。



说明：

对于按量付费的VPC类型实例而言，[停机不收费](#)模式下，更换系统盘后可能无法成功启动实例。建议您停止实例时关闭停机不收费。

5. 实例停止后，在操作列中，选择更多 > 磁盘和镜像 > 更换系统盘。
6. 在弹出的对话框里，仔细阅读更换系统盘注意事项后，单击确定，更换系统盘。
7. 在更换系统盘页面上，
 - a. 镜像类型：选择公共镜像，并选择需要的镜像版本。



说明：

如果您需要使用其他镜像，请参见[更换系统盘#非公共镜像#](#)。

- b. 系统盘：不能更换系统盘类型，但是您可以根据业务需求和新镜像的需求扩容系统盘，最大容量为500 GiB。扩容时能设置的最小容量与系统盘当前容量和镜像有关，如下表所示。

镜像	扩容的容量限制（GiB）
Linux（不包括CoreOS）+ FreeBSD	[Max{20, 系统盘当前容量}, 500]
CoreOS	[Max{30, 系统盘当前容量}, 500]

镜像	扩容的容量限制（GiB）
Windows	[Max{40, 系统盘当前容量}, 500]



说明:

如果您的实例已经做过续费降配操作，进入下一个计费周期前，您不能修改系统盘容量。

c. 安全设置：

- 如果新的操作系统是Windows系统，您只能使用密码认证。



- 如果您的实例是I/O优化实例，而且新的操作系统是Linux系统，您能选择使用密码认证或SSH密钥对认证。您必须设置登录密码或者绑定SSH密钥对。



d. 确认配置费用：包括了镜像的价格与系统盘的价格。系统盘价格详见[云产品价格页](#)。

e. 确认无误后，单击确定更换。

登录ECS管理控制台监控系统状态。完成操作系统更换大概需要10分钟。完成后，实例会自动启动。

后续操作

更换系统盘后，您可能需要做以下操作：

- （可选）[为新的系统盘设置自动快照策略](#)。自动快照策略是与磁盘ID绑定的。更换了新的系统盘后，旧磁盘上应用的自动快照策略自动失效。您需要对新系统盘设置自动快照策略。
- 如果更换前后都是Linux系统，而且，实例上原来挂载了数据盘并设置了开机自动挂载分区：更换系统盘后，原来系统盘中的分区挂载信息丢失。您必须在新系统盘的/etc/fstab文件写入新分区信息，并挂载分区，不需要再次分区并格式化数据盘。操作步骤如下，具体的操作命令参见[Linux格式化和挂载数据盘](#)：

1. （建议）备份/etc/fstab。
2. 向/etc/fstab写入新分区信息。
3. 查看/etc/fstab中的新分区信息。
4. 运行mount命令挂载分区。
5. 查看文件系统空间和使用情况：运行命令df -h。

挂载操作完成后，不需要重启实例即可开始使用新的数据盘。

相关API

[ReplaceSystemDisk](#)

3.9 回滚云盘

如果当前系统出现问题，您希望将一块云盘的数据恢复到之前的某一时刻，而且您已经为云盘 [#unique_30](#)，您可以使用 回滚磁盘 功能实现。

注意事项

回滚云盘前，请注意以下事项：

- 回滚云盘是不可逆操作，一旦回滚完成，原有的数据将无法恢复，请谨慎操作。
- 回滚云盘后，从快照的创建日期到回滚云盘时这段时间内的数据会丢失。
- 回滚系统盘后，默认自动绑定相应实例当前的密钥对或密码。

前提条件

回滚云盘前必须确认以下信息：

- 您已经为云盘 [#unique_30](#)，而且要回滚的云盘当前没有正在创建的快照。
- 云盘未被释放。

- 云盘必须已经 [挂载到某台ECS实例上](#)，而且已经 [停止实例](#)。



说明:

如果是按量付费的VPC类型实例，而且已经开启了 [按量付费实例停机不收费](#) 功能，停止实例时，在提醒弹出框中，单击 [确定](#) 之后，在 [停止实例](#) 对话框中，选择 [停止后仍旧保留实例并继续收费](#)。如果使用 [停机不收费](#) 模式，回滚云盘后可能无法成功启动实例。

停止实例

您所选的 **1 个实例** 将执行停止操作，您是否确认操作？

停止方式：

☒ 停止
☐ 强制停止

停止模式：

☒ **停止后仍旧保留实例并继续收费**

包年包月实例停止后不会改变到期时间

如果您停止实例是为了更换系统盘、重新初始化磁盘、更改实例规格、修改私网IP等操作，建议您勾选 "停止后仍旧保留实例并继续收费" 选项，避免启动失败。

确定

取消

操作步骤

按以下步骤回滚云盘：

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏中，单击 [实例](#)。
3. 选择地域。
4. 找到需要回滚云盘的实例，进入 [实例详情](#) 页。
5. 在左侧导航栏中，单击 [本实例快照](#)。
6. 选择需要的快照，在 [操作](#) 列中，单击 [回滚磁盘](#)。
7. 在弹出的提示框中，单击 [确定](#)。



说明:

如果您选择 [回滚后立即启动实例](#)，云盘回滚成功后，实例会自动启动。

相关API

[#unique_44](#)

后续操作

如果在数据盘创建快照后，您做过扩容操作，回滚云盘后，您需要登录实例重新扩容文件系统。具体操作参见：

- [扩展分区和文件系统-Linux数据盘](#)
- [扩展分区和文件系统-Windows](#)

3.10 重新初始化云盘

当云盘挂载在ECS实例上时，您可以使用 **重新初始化磁盘** 功能将作系统盘或数据盘用的云盘恢复到创建时的状态。重新初始化后：

- 系统盘会恢复到刚创建时的状态。比如，如果您用公共镜像Windows Server 2008 R2创建实例，那么，重新初始化系统盘后，操作系统依然是最初的Windows Server 2008 R2，但是创建实例后安装的应用程序和产生的数据都会被删除。



说明：

更换系统盘后再执行初始化，初始化的是更换后的系统盘，而不是更换前的系统盘。

- 作数据盘用的云盘会恢复到初始状态：
 - 如果原来是空盘，则恢复为空盘。
 - 如果是 [通过快照创建的云盘](#)，则会恢复到快照的状态。
- 如果云盘应用了自动快照策略，重新初始化后，设置仍然有效，不需要重新设置。
- 重新初始化磁盘不会更改ECS实例的IP地址、云盘ID。
- 重新初始化磁盘不会删除云盘的自动快照或手动快照，您可以使用这些快照 [回滚云盘](#)。



警告：

- 重新初始化磁盘前必须先停止ECS实例，该操作会中断您的服务，请务必谨慎操作。
- 重新初始化磁盘后，云盘上的数据将会丢失，请务必提前做好数据备份，例如为云盘创建快照，详细操作，请参见 [#unique_30](#)。

重新初始化系统盘

前提条件

初始化Linux实例的系统盘时，如果使用密钥对验证，您必须已经 [创建 SSH 密钥对](#) 或 [导入 SSH 密钥对](#)。

操作步骤

按以下步骤重新初始化系统盘：

1. 登录 [ECS管理控制台](#)。
2. 选择地域。
3. 在左侧导航栏中，单击 **实例**。
4. 找到需要初始化系统盘的实例，单击实例ID进入 **实例详情** 页。
5. 在页面右上角，单击 **停止**，停止实例。



说明：

如果是按量付费的VPC类型实例，而且已经开启了 [按量付费实例停机不收费](#) 功能，在 **提醒** 弹出框中，单击 **确定**。在 **停止实例** 对话框中，选择 **停止后仍旧保留实例并继续收费**。如果使用 **停机不收费** 模式，重新初始化系统盘后可能无法成功启动实例。

停止实例

?

您所选的 1 个实例 将执行停止操作，您是否确认操作？

停止方式：

☒ 停止
☐ 强制停止

停止模式：

☒ 停止后仍旧保留实例并继续收费

包年包月实例停止后不会改变到期时间

如果您停止实例是为了更换系统盘、重新初始化磁盘、更改实例规格、修改私网IP等操作，建议您勾选 "停止后仍旧保留实例并继续收费" 选项，避免启动失败。

确定

取消

6. 实例进入 **已停止** 状态后，在左侧导航栏中，单击 **本实例磁盘**。
7. 找到系统盘，在 **操作** 列中，单击 **重新初始化磁盘**。

8. 在弹出的 重新初始化磁盘 对话框里，完成以下设置：

a. 设置验证方法：

- Windows实例：重新设置实例登录密码。可以使用旧的密码，也可以指定新的密码。

重新初始化磁盘

您确认重新初始化以下磁盘吗：

系统盘： d-bp1itotpmd9ntjhgcl1b

*登录密码：

.....

8-30个字符，必须同时包含下面四项中的三项：大写字母、小写字母、数字、和特殊字符（仅支持下列特殊字符：() ` ~ ! @ # \$ % ^ & * - + = | { } [] : ; ' < > , . ? / ）

*确认密码：

.....

安全加固：

☒ 免费开通

启动实例策略：

☒ 重置磁盘后启动

提示：1. 您所选择的系统盘，重新初始化磁盘后将恢复到镜像的初始状态。

2. 重新初始化磁盘时您需要重新设置实例的登录密码。

注：因您没有备份相关个人数据而造成的数据丢失，阿里云不承担责任。

确认重新初始化磁盘

取消

- Linux实例：选择 设置密钥 或 设置密码，并绑定密钥对或重新设置密码。可以使用旧的密码，也可以指定新的密码。

重新初始化磁盘

您确认重新初始化以下磁盘吗：

系统盘：IniSys

安全设置：☒ 设置密钥 ☐ 设置密码

密钥对：

SSHTest

安全加固：☒ 免费开通

启动实例策略：☒ 重置磁盘后启动

提示：1. 您所选择的系统盘，重新初始化磁盘后将恢复到镜像的初始状态。

2. 重新初始化磁盘时您需要重新设置实例的登录密码。

注：因您没有备份相关个人数据而造成的数据丢失，阿里云不承担责任。

确认重新初始化磁盘

取消

b. （可选）安全加固：选择 免费开通。开通安全加固功能后，您的实例会自动免费加载云服务器安全组件，提供网站后门检测、异地登录提醒、暴力破解拦截等安全功能。

c. （可选）启动实例策略：选择 重置磁盘后启动。完成初始化后，实例会自动启动。

d. 单击 确认重新初始化磁盘。

9. 仅适用于Linux实例：如果初始化前挂载了数据盘，您需要 [重新创建挂载点信息并挂载文件系统](#)。



说明：

在Linux实例中，重新初始化系统盘不会改变数据盘里的内容，但是数据盘的挂载信息会丢失，所以需要执行这步操作。

系统盘重新初始化后，您需要重新部署业务（包括应用和配置）从而尽快恢复业务。

重新初始化数据盘

重新初始化后，数据盘的状态根据云盘最初的状态以及实例的操作系统不同而异：

- Windows实例：无论数据盘是从快照创建而来，还是一个全新的空盘，重新初始化后，数据盘会恢复到创建后的初始状态，但是可以直接使用，不需要执行其他操作。

· Linux实例：

- 如果数据盘创建时是一个空盘：数据盘的文件系统及内容完全丢失，您必须重新格式化、分区数据盘，并挂载数据盘分区。



说明：

在这种情况下，如果重新初始化前您在 `/etc/fstab` 文件中写入了开机自动挂载分区的命令，那么，重新初始化后，开机自动挂载会失效，启动时一般会出现异常。所以，在重新初始化数据盘之前，您必须先在 `/etc/fstab` 中注释掉开机自动挂载数据盘分区的信息。

- 如果数据盘是从快照创建而来：数据盘会恢复到原来快照的状态，不需要重新挂载数据盘分区，但是数据盘创建成功后产生的所有数据都会丢失。

本文的步骤描述中用 `/dev/vdb1` 表示数据盘分区，`/InitTest` 表示挂载点。您可以根据实际情况替换。

前提条件

云盘已经挂载到实例上。详细操作，请参见 [挂载云盘](#)。

操作步骤

按以下步骤重新初始化数据盘：

1. 仅适用于Linux实例：如果数据盘创建时是一个空盘，而且您在 `/etc/fstab` 中添加了开机自动挂载数据盘分区的命令，那么，您必须在 `/etc/fstab` 里注释掉开机挂载数据盘分区的命令。操作步骤如下：

- a. [远程连接Linux实例](#)。
- b. 运行 `vim /etc/fstab`。
- c. 按 `i` 键进入编辑模式。
- d. 找到数据盘分区挂载信息，加 `#` 注释掉挂载信息，如下所示。

```
# /dev/vdb1 /InitTest ext3 defaults 0 0
```

- e. 按 `Esc` 键退出编辑模式，输入 `:wq` 保存并退出。

2. 登录 [ECS管理控制台](#)。
3. 在左侧导航栏中，单击 **实例**。
4. 选择地域。
5. 找到需要初始化数据盘的实例，单击实例ID进入 **实例详情** 页。

6. 在页面右上角，单击 停止，停止实例。



说明：

如果是按量付费的VPC类型实例，而且已经开启了 [按量付费实例停机不收费](#) 功能，在 提醒 弹出框中，单击 确定。在 停止实例 对话框中，选择 停止后仍旧保留实例并继续收费。如果使用 停机不收费 模式，重新初始化数据盘后可能无法成功启动实例。

停止实例

?

您所选的 1 个实例 将执行停止操作，您是否确认操作？

停止方式：

☒ 停止
☐ 强制停止

停止模式：

☒ 停止后仍旧保留实例并继续收费

包年包月实例停止后不会改变到期时间

如果您停止实例是为了更换系统盘、重新初始化磁盘、更改实例规格、修改私网IP等操作，建议您勾选 "停止后仍旧保留实例并继续收费" 选项，避免启动失败。

确定

取消

7. 实例进入 已停止 状态后，在左侧导航栏中，单击 本实例磁盘。

8. 找到数据盘，在 操作 列中，单击 重新初始化磁盘。

9. 在弹出的 重新初始化磁盘 对话框里，认真阅读提示后，单击 确认重新初始化磁盘。

10. 在左侧导航栏中，单击 实例详情。

11. 在页面右上角，单击 启动，启动实例，完成数据盘初始化。

12. 仅适用于Linux实例：如果数据盘创建时是一个空盘，您必须 [Linux格式化数据盘](#)。

数据盘重新初始化后，您需要重新部署业务（包括应用和配置）从而尽快恢复业务。

相关API

[#unique_45](#)

3.11 转换云盘的计费方式

云盘的计费方式与创建方式有关：

- 随包年包月实例一起创建的云盘，需要先付费再使用。更多计费信息，请参考 [包年包月](#)。
- 随按量付费实例一起创建的云盘，以及通过[ECS管理控制台](#)云盘页面创建的云盘，均采用按量付费方式计费。更多计费信息，请参考 [按量付费](#)。

您能使用不同的方式转换云盘的计费方式，如下表所示。

计费方式转换	转换方式	适用的云盘	生效时间
包年包月—>按量付费	续费降配	包年包月实例上挂载的包年包月云盘。不能变更系统盘的计费方式。	在新的计费周期生效
按量付费—>包年包月	升级配置	包年包月实例上挂载的按量付费数据盘。不能变更系统盘的计费方式。	立即生效
	按量付费转包年包月	按量付费实例上挂载的系统盘和数据盘。	

3.12 查看云盘监控信息

使用云盘时，您需要注意以下2个性能指标：

- IOPS：指Input/Output Operations per Second，即每秒能处理的I/O个数，用于表示块存储处理读写（输出/输入）的能力。如果要部署事务密集型应用，需要关注IOPS性能。
- 吞吐量：是指单位时间内可以成功传输的数据数量，单位为 MBps。如果要部署大量顺序读写的应用，需要关注吞吐量。

您可以在ECS控制台上查看指定时间段内单块云盘的IOPS和吞吐量的监控信息。如果您为云盘所挂载的ECS实例 安装了云监控插件，也可以在云监控中监控云盘的使用率、IOPS和吞吐量。

在ECS管理控制台上查看指定云盘的IOPS和吞吐量的监控信息，步骤如下：

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏中，选择 存储 > 云盘。
3. 选择地域。
4. 找到要查看监控信息的云盘，单击磁盘ID，进入 磁盘详情 页。
5. 在左侧导航栏中，单击 磁盘监控信息。

6. 在 磁盘监控信息 页上，单击  图标，设置监控信息的起止时间。您能查看最多15天间隔的云盘性能监控信息。

监控信息

2018-03-18 14:25 - 2018-03-18 15:25 

IOPS(个)

开始时间：

2018-03-18 

14  

：

25  

结束时间：

2018-03-18 

15  

：

25  

1小时

6小时

1天

7天

时间间隔最大15天

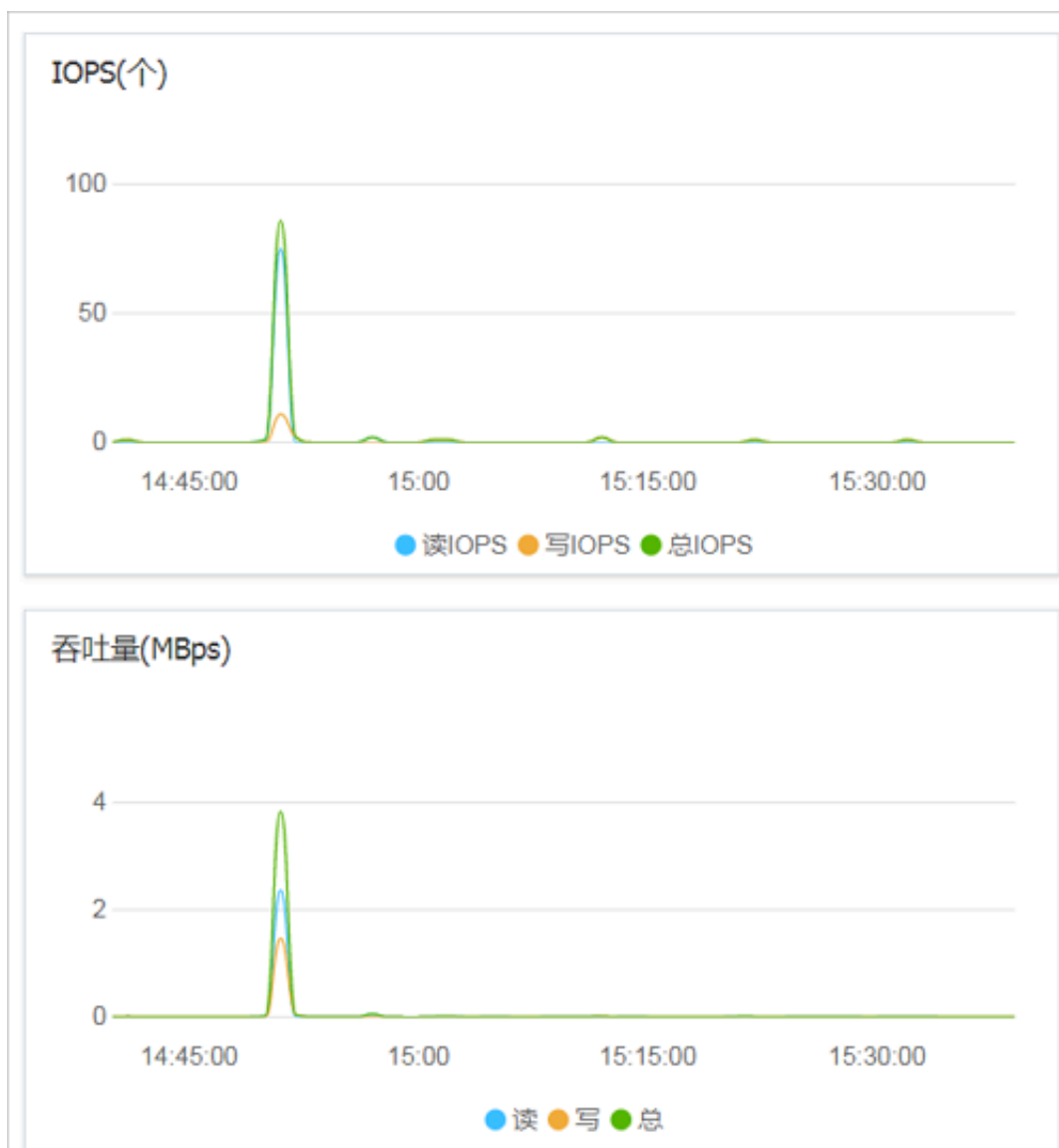
确定

7. 查看云盘的IOPS和吞吐量信息。



说明：

您可以单击图表中的图例查看单个指标的信息。



3.13 卸载数据盘

当按量付费云盘作数据盘用时，您可以卸载云盘。但是，云盘作系统盘用时，您不能卸载系统盘。

卸载云盘时，需要注意以下事项：

- 只能卸载状态为 使用中，而且磁盘属性为 数据盘 的按量付费云盘。
- 不能卸载本地存储。
- 在Windows操作系统下，您需要注意以下事项：
 - 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。
 - 在ECS控制台上卸载云盘前，必须先 [远程连接实例](#)，在 磁盘管理 中对磁盘执行脱机操作。

· 在Linux操作系统下，您需要注意以下事项：

- 在ECS控制台上卸载云盘前，必须先 [远程连接实例](#)，运行 `umount` 命令卸载磁盘分区。
- 如果您已经在 `/etc/fstab` 文件中设置了启动实例时自动挂载磁盘分区，在卸载云盘前，您必须先先在 `/etc/fstab` 文件中删除磁盘分区的挂载信息，否则实例重启后您将无法连接实例。

您可以根据需要采用不同的方式卸载云盘，如下表所示。

场景	适用的卸载操作
您要从某台实例上卸载一块或多块云盘	在实例管理页面卸载云盘
您已经明确知道要卸载哪块云盘	在云盘管理页面卸载云盘

在实例管理页面卸载云盘

您可以在实例的管理页面上卸载一台实例上挂载的一块或多块云盘。

前提条件

云盘已经 [挂载到实例上](#)，状态为 使用中。

如果是卸载Linux实例上的某块云盘，而且已经在 `/etc/fstab` 文件中写入磁盘分区挂载信息，必须先删除这个配置信息。

操作步骤

在实例管理页面卸载云盘，按如下步骤操作：

1. 远程连接实例，在实例内部卸载磁盘或分区。根据操作系统不同，您需要执行不同的操作，如下表所示。

操作系统	操作
Linux	运行 <code>umount [磁盘分区]</code> 。比如： <code>umount /dev/vdb1</code> 。
Windows	在 磁盘管理 窗口，右键单击磁盘名称（如下图所示的 磁盘2），单击 脱机。

2. 登录 [ECS管理控制台](#)。
3. 在左侧导航栏中，单击 实例。
4. 选择地域。
5. 找到需要操作的实例，单击实例ID，进入实例管理页面。
6. 在左侧导航栏中，单击 本实例磁盘。

7. 找到需要卸载的云盘，在 操作 栏中，选择 更多 > 卸载。

支持卸载的云盘必须具备以下属性：

- 磁盘状态 必须为 使用中。
- 可卸载 必须为 支持。
- 磁盘属性 必须为 数据盘。

8. 在弹出对话框中，单击 确认卸载。

9. （可选）如果需要卸载多块云盘，重复第7步和第8步操作。

当云盘的状态变为 待挂载 时，表明您已经成功卸载云盘。

在云盘管理页面卸载云盘

您可以在云盘管理页面上将某块云盘从实例上卸载。

前提条件

云盘已经 [挂载到实例上](#)，状态为 使用中。

如果是卸载Linux实例上的某块云盘，而且已经在 `/etc/fstab` 文件中写入磁盘分区挂载信息，必须先删除这个配置信息。

操作步骤

在云盘管理页面卸载云盘，按如下步骤操作：

1. 远程连接实例，在实例内部卸载磁盘或分区。根据操作系统不同，您需要执行不同的操作，如下表所示。

操作系统	操作
Linux	运行 <code>umount [磁盘分区]</code> 。比如： <code>umount /dev/vdb1</code> 。
Windows	在 磁盘管理 窗口，右键单击磁盘名称（如下图所示的 磁盘2），单击 脱机。

2. 登录 [ECS管理控制台](#)。

3. 在左侧导航栏中，选择 存储 > 云盘。

4. 选择地域。

5. 找到需要卸载的云盘，在 操作 栏中，选择 更多 > 卸载。

支持卸载的云盘必须具备以下属性：

- 磁盘状态 必须为 使用中。
- 可卸载 必须为 支持。
- 磁盘属性 必须为 数据盘。

6. 在弹出对话框中，单击 确认卸载。

当云盘的状态变为 待挂载 时，表明您已经成功卸载云盘。

相关API

[DetachDisk](#)

后续操作

如果不再需要该云盘，您可以 [释放云盘](#)。

3.14 释放云盘

如果您不再需要某块云盘，卸载后，可以将其释放，以节省费用。

注意事项

单独释放云盘时，您需要注意以下事项：

- 只有处于 待挂载 状态的云盘，才能单独释放。其他云盘（比如系统盘和作数据盘用的包年包月云盘）都只能随实例一起释放。如果云盘仍处于 使用中 状态，请先 卸载数据盘。
- 默认情况下，云盘的自动快照会随云盘一起释放。但是手动快照不受影响。您可以在 挂载云盘 时设置自动快照不随云盘一起释放。



说明：

每块云盘有64个快照额度。为了保证拥有足够的快照额度来顺利完成周期性的自动快照策略，建议您删除不必要的手动快照和自动快照。

- 释放前确认云盘中的内容是否仍然有效，完成数据备份，比如 创建快照。

操作步骤

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏中，选择 存储 > 云盘。
3. 选择地域。
4. 找到需要释放的云盘（状态为 待挂载），在 操作 列中，选择 更多 > 释放。

5. 在弹出的对话框中，确认信息后，单击 确认释放。

相关API

[#unique_118](#)

4 共享块存储

共享块存储是一种支持多台ECS实例并发读写访问的数据块级存储设备，具备多并发、高性能、高可靠等特性，数据可靠性可以达到 99.9999999%。



说明：

共享块存储目前处于公测阶段，您可以[提交工单](#)申请公测资格。

共享块存储采用分布式三副本机制，为ECS实例提供99.9999999%的数据可靠性保证。支持在可用区内自动复制您的数据，防止意外硬件故障导致的数据不可用，保护您的业务免于组件故障的威胁。就像硬盘一样，您可以对挂载到ECS实例上共享块存储做分区、格式化、创建文件系统等操作，并对数据进行持久化存储。



说明：

- 一块共享块存储最多可以同时挂载到同一地域、同一可用区的8台ECS实例。
- 挂载到实例上时，共享块存储与云盘共享数据盘配额，即一台实例最多挂载16块数据盘。
- 共享块存储只能单独创建，并且只能作数据盘用，可以共享访问。

适用场景

在传统集群架构中，需要多个计算节点能够访问同一份数据，防止当一个或多个计算节点故障时导致业务中断，保证整个高可用集群对外可以持续提供业务服务。因此，重要的数据文件需要放置在共享块存储上，并通过集群文件系统对共享块存储进行统一管理。在前端多个计算节点并发进行读写访问时，保持数据在多个节点间的一致性。

共享块存储专为企业级用户的核心业务高可用架构而设计，适用于shared-everything架构下对块存储设备的共享访问场景，例如，政府、企业和金融行业常用的Oracle RAC数据库高可用架构以及服务器High-availability cluster高可用架构。

共享块存储类型

根据性能不同，共享块存储可以分为：

类型	说明
SSD共享块存储	采用固态硬盘作为存储介质，能够提供稳定的高随机I/O、高数据可靠性的高性能存储。
高效共享块存储	采用固态硬盘与机械硬盘的混合介质作为存储介质。

共享块存储性能

共享块存储的性能，请参见[块存储性能](#)。

计费

共享块存储目前处于免费公测阶段，不收费。

使用共享块存储

共享块存储的使用与云盘类似。

共享块存储产品本身并不提供集群文件系统，需要您自行安装集群文件系统来管理共享块存储。

正确使用共享块存储的方式是采用集群文件系统进行块设备的统一管理，例如GFS、GPFS等。典型Oracle RAC业务场景中推荐采用ASM统一管理存储卷和文件系统。

如果只是将共享块存储挂载到多台ECS实例，但依旧使用常规文件系统来管理时，会造成以下两个问题：

- 磁盘空间分配冲突

如果一块共享块存储挂载到多个实例上，当实例A在写文件时，会查询文件系统和可用的磁盘空间，文件写入后会修改实例A上的空间分配记录，但不会修改其他实例的记录。因此，当实例B写入文件时，可能会将实例A已经分配出去的磁盘空间再次分配，造成磁盘空间分配冲突。

- 数据文件不一致

当实例A读取数据并记录在缓存中后，实例A上另一个进程来访问同样的数据就会直接从缓存中进行读取。但如果此时实例B修改了同样的数据，而实例A并不知道，依旧从缓存中读取数据，则会造成业务数据不一致。

5 本地盘

本地盘是云服务器ECS所在宿主机上的本地硬盘设备。本地盘能够为实例提供本地存储访问能力，具有低时延、高随机IOPS、高吞吐量和高性价比的优势。

存储类型

本地盘适用于对存储I/O性能、海量存储性价比有极高要求的业务场景。目前，阿里云提供以下两种本地盘：

- NVMe SSD本地盘：搭配使用的实例规格族包括i2、i1、gn5。其中，i1和i2实例规格族适用于以下场景：
 - 网络游戏、电商、视频直播、媒体等在线业务。满足I/O密集型应用对块存储的低时延和高I/O性能需求。
 - 对存储I/O性能有较高要求，同时具备应用层高可用架构的业务场景，如NoSQL非关系型数据库、MPP数据仓库和分布式文件系统等。
- SATA HDD本地盘：搭配使用的实例规格族包括d1ne和d1。适用于互联网行业、金融行业等有大数据计算与存储分析需求的行业，进行海量数据存储和离线计算的业务场景。充分满足以Hadoop为代表的分布式计算业务类型对实例存储性能、存储容量和内网带宽的多方面要求。

本文主要描述当前与实例一起销售的本地盘的信息。有关本地盘实例和大数据型实例的性能详情，请参见[实例规格族](#)。如果您使用的是已经停售的上一代本地SSD盘，请参见[上一代磁盘 - 本地SSD盘](#)。

NVMe SSD本地盘性能

NVMe SSD本地盘性能如下表所示。

NVMe SSD指标	单盘性能	综合实例性能
最大容量	1456 GiB	2912 GiB
最大IOPS	240000	480000
写IOPS *	min{165 * 容量, 240000}	2 * min{165 * 容量, 240000}
读IOPS *		
最大读吞吐量	2 GBps	4 GBps
读吞吐量 *	min{1.4 * 容量, 2000}MBps	2 * min{1.4 * 容量, 2000}MBps
最大写吞吐量	1.2 GBps	2.4 GBps
写吞吐量 *	min{0.85 * 容量, 1200}MBps	2 * min{0.85 * 容量, 1200}MBps

NVMe SSD指标	单盘性能	综合实例性能
访问时延	微秒级 (μs)	

* 单盘性能计算公式说明：

- 以单块NVMe SSD本地盘写IOPS计算公式说明：每GiB为165 IOPS，最高240000 IOPS。
- 以单块NVMe SSD本地盘写吞吐量计算公式说明：每GiB为0.85 MBps，最高1200 MBps。

为压测出阿里云提供的标准性能数据及本地盘QoS（Quality of Service，服务质量），您可以根据本文测试NVMe SSD本地盘在带宽、IOPS以及延迟等性能指标上的表现。参见下文[NVMe SSD本地盘性能测试](#)。

SATA HDD本地盘性能

SATA HDD本地盘性能如下表所示。

SATA HDD指标	单盘性能	综合实例性能
最大容量	5500 GiB	154000 GiB
最大吞吐量	190 MBps	5320 MBps
访问时延	毫秒级 (ms)	

计费方式

本地盘的费用包括在本地盘实例的费用里。关于实例的计费方式，请参见[预付费#包年包月#和按量付费](#)。

使用说明

- 本地盘来自单台宿主机，数据可靠性取决于宿主机的可靠性，存在单点故障风险。建议您在应用层做数据冗余，保证数据的可用性。



警告：

使用本地盘存储数据有丢失数据的风险，例如宿主机发生硬件故障时。请勿在本地盘上存储需要长期保存的业务数据。如果您的应用无数据可靠性架构设计，强烈建议您在实例中同时使用[云盘](#)，提高数据可靠性。

- 购买了带本地盘的实例后，请登录ECS实例自行[分区并格式化本地盘](#)。

- 本地盘不支持的操作有：
 - 单独创建全新本地盘
 - 使用快照创建本地盘
 - 挂载本地盘
 - 单独卸载并释放本地盘
 - 扩容本地盘
 - 重新初始化本地盘
 - 为本地盘创建快照
 - 使用快照回滚本地盘

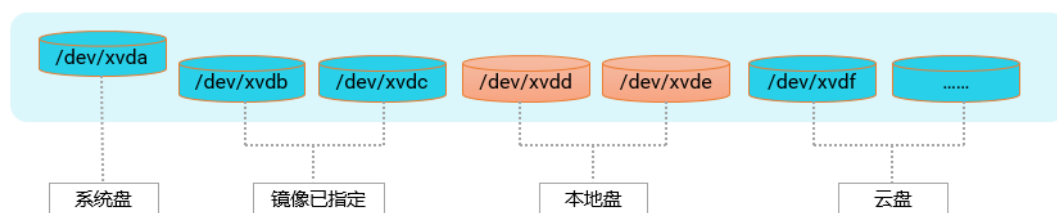
磁盘初始化顺序

创建带本地盘的实例时，所有磁盘的初始化顺序遵循以下规则：

- 规则一：如果指定的镜像不带有数据盘快照，则按照本地盘优先、随实例创建的云盘其次的顺序排列。
- 规则二：如果指定的镜像带有数据盘快照，由于制作镜像时，会同时记录数据盘设备名，优先保留镜像中的数据盘快照所对应的磁盘顺序，其余排列顺序遵循规则一。

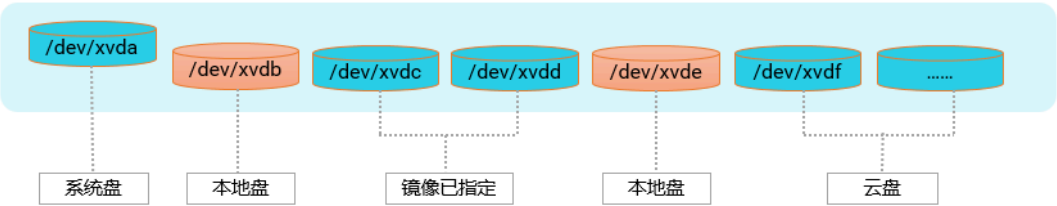
以Linux类型镜像中包含两块数据盘快照的场景为例，为您讲解规则二的排序原理：

- 假设两块数据盘的原设备名分别是/dev/xvdb和/dev/xvdc：在初始化本地盘实例时，阿里云优先将/dev/xvdb和/dev/xvdc分配给镜像中指定的数据盘使用。则磁盘初始化顺序为系统盘、镜像已指定的数据盘1、镜像已指定的数据盘2、本地盘1、本地盘2、云盘1、云盘2、…。如下图所示。



- 假设两块数据盘的原设备名分别是/dev/xvdc和/dev/xvdd：在初始化本地盘实例时，阿里云优先将/dev/xvdc和/dev/xvdd分配给镜像中指定的数据盘使用。剩下的设备名位置再以本地盘优

先的方式填充。则磁盘初始化顺序为系统盘、本地盘1、镜像已指定的数据盘1、镜像已指定的数据盘2、本地盘2、云盘1、云盘2、…。如下图所示。



生命周期

本地盘的生命周期与它所挂载的本地盘实例相同，更多详情，请参见[实例生命周期介绍](#)。

实例操作对本地盘数据的影响

操作本地盘实例对本地盘数据的影响如下表所示。

实例操作	保留本地盘数据状态	保留本地盘数据
操作系统重启/ 控制台重启 /强制重启	是	是
操作系统关机/ 控制台停止 /强制停止	是	是
实例自动恢复	否	否
释放实例	否	否
预付费实例到期停机或账号欠费，释放实例前	是	是
预付费实例到期停机或账号欠费，释放实例后	否	否

NVMe SSD本地盘性能测试

Linux实例和Windows实例都推荐使用FIO工具测试本地盘性能。本文以测试裸盘/dev/vdx为例，说明如何使用FIO测试本地盘块存储性能。部分测试命令参数解释请参见[块存储性能](#)。

 **警告：**

测试裸盘可以获得真实的本地盘性能，但直接测试裸盘会破坏文件系统结构，请在测试前提前做好数据备份。本文均为裸盘测试，请务必了解写操作会覆盖本地盘数据。建议您只在新购无数据的ECS实例上测试本地盘性能。

- NVMe SSD本地盘带宽性能测试

- 测试读带宽，运行以下命令：

```
fio -direct=1 -iodepth=128 -rw=read -ioengine=libaio -bs=128k -  
numjobs=1 -time_based=1 -runtime=1000 -group_reporting -filename=/  
dev/vdx
```

- 测试写带宽，运行以下命令：

```
fio -direct=1 -iodepth=128 -rw=write -ioengine=libaio -bs=128k -  
numjobs=1 -time_based=1 -runtime=1000 -group_reporting -filename=/  
dev/vdx
```

- NVMe SSD本地盘IOPS性能测试

- 测试随机读IOPS，运行以下命令：

```
fio -direct=1 -iodepth=32 -rw=randread -ioengine=libaio -bs=4k -  
numjobs=4 -time_based=1 -runtime=1000 -group_reporting -filename=/  
dev/vdx
```

- 测试随机写IOPS，运行以下命令：

```
fio -direct=1 -iodepth=32 -rw=randwrite -ioengine=libaio -bs=4k -  
numjobs=4 -time_based=1 -runtime=1000 -group_reporting -filename=/  
dev/vdx
```

- NVMe SSD本地盘延迟性能测试

- 测试随机读延迟，运行以下命令：

```
fio -direct=1 -iodepth=1 -rw=randread -ioengine=libaio -bs=4k -  
numjobs=1 -time_based=1 -runtime=1000 -group_reporting -filename=/  
dev/vdx
```

- 测试随机写延迟，运行以下命令：

```
fio -direct=1 -iodepth=1 -rw=randwrite -ioengine=libaio -bs=4k -  
numjobs=1 -time_based=1 -runtime=1000 -group_reporting -filename=/  
dev/vdx
```

- 测试顺序读延迟，运行以下命令：

```
fio -direct=1 -iodepth=1 -rw=read -ioengine=libaio -bs=4k -numjobs  
=1 -time_based=1 -runtime=1000 -group_reporting -filename=/dev/vdx
```

- 测试顺序写延迟，运行以下命令：

```
fio -direct=1 -iodepth=1 -rw=write -ioengine=libaio -bs=4k -  
numjobs=1 -time_based=1 -runtime=1000 -group_reporting -filename=/  
dev/vdx
```

6 块存储FAQ

什么是ESSD云盘？

ESSD云盘，又称增强型（Enhanced）SSD云盘，是阿里云全新推出的超高性能云盘产品。ESSD云盘基于新一代分布式块存储架构，结合25GE网络和RDMA技术，为您提供单盘高达100万的随机读写能力和更低的单路时延能力。

ESSD云盘的可靠性如何？

ESSD云盘的可靠性与SSD云盘、高效云盘一致，提供99.9999999%的可靠性设计。

ESSD云盘适用于哪些应用场景？

ESSD云盘本身具备低时延、高性能的特点，比较适合应用在以下时延敏感或者I/O密集型业务场景。

- 部署OLTP数据库。例如，MySQL、PostgreSQL、Oracle或者SQL Server等关系型数据库。
- 部署NoSQL数据库。例如，MongoDB、HBase或者Cassandra等非关系型数据库。
- 运行ElasticSearch分布式日志。例如，ELK（Elasticsearch、Logstash和Kibana）日志分析。

ESSD云盘与SSD云盘或者高效云盘有哪些异同之处？

共同点：ESSD云盘和SSD云盘和高效云盘类似，都是基于分布式块存储架构的云盘类产品，具备高可靠、弹性扩容等特性，支持快照、加密等数据功能。

差异点：ESSD云盘的性能相对SSD云盘和高效云盘都有大幅提升，详情请参见[块存储性能](#)。

ESSD云盘的性能指标是如何计算的？

ESSD云盘的性能与容量线性相关，容量越大性能越高。相比SSD云盘有大幅的性能提升。详情请参见[块存储性能](#)。

如何测试ESSD云盘的性能？100万IOPS如何测试出来？应该如何配置ECS实例？

您可以尝试通过FIO工具测试ESSD云盘性能，具体步骤请参见[如何压测ESSD云盘的性能](#)。

ESSD云盘性能和实例性能有什么关系？

部分实例维度的存储I/O性能和实例规格成正比线性关系。例如，云服务器ECS存储增强型实例g5se的存储I/O性能跟实例规格成线性关系，实例规格越高可获得的存储IOPS和吞吐量越高。

当您创建某个规格的g5se实例并挂载ESSD云盘之后：

- 如果ESSD云盘的性能总和不超过实例规格族所对应的存储I/O能力，实际存储性能以ESSD云盘性能为准。
- 如果ESSD云盘的性能总和超过了实例规格族所对应的存储I/O能力，实际存储性能以该实例规格对应的存储I/O能力为准。

例如，当您创建了ecs.g5se.xlarge 16 GiB规格实例后，该实例最大存储IOPS为6万。如果挂载了1块存储I/O能力为2 TiB的ESSD云盘（单盘IOPS为101800），该实例最大存储IOPS只能为6万，而无法达到2 TB ESSD云盘的101800 IOPS。

其他云服务器ECS存储增强型实例g5se的规格及性能见下表。

实例规格	vCPU（核）	内存（GiB）	存储IOPS（万）	存储吞吐量（MBps）
ecs.g5se.large	2	8	3	118
ecs.g5se.xlarge	4	16	6	235
ecs.g5se.2xlarge	8	32	12	470
ecs.g5se.4xlarge	16	64	23	900
ecs.g5se.6xlarge	24	96	34	1350
ecs.g5se.8xlarge	32	128	45	1800
ecs.g5se.16xlarge	64	256	90	3600
ecs.g5se.18xlarge	70	336	100	4000

ESSD云盘如何计费？

支持预付费和按量付费。具体价格请参见 [价格详情页](#)。

当前哪些地域或者可用区开放了ESSD云盘资源？

当前您可以在以下可用区申请购买ESSD云盘：

- 华北 2（北京）可用区 G
- 华东 2（上海）可用区 E、可用区 F
- 华东 1（杭州）可用区 H、可用区 I
- 华南 1（深圳）可用区 E

- 华北 3（张家口）可用区 A

ESSD云盘目前能挂载到哪些实例规格族上？

目前ESSD云盘产品处于公测期间，只有25GE网络的实例规格族（c5、ic5、g5、r5、g5se）支持挂载ESSD云盘，其他实例规格族暂时不支持。

磁盘有哪些常用操作？

关于磁盘的常见操作，请参见以下文档：

- [Windows格式化数据盘](#)
- [Linux格式化数据盘](#)
- [卸载数据盘](#)
- [重新初始化云盘](#)
- [更换系统盘#非公共镜像#](#)

SSD云盘具备怎样的I/O性能？

请参见[块存储性能](#)。

SSD云盘适合的应用场景有哪些？

SSD云盘具有高性能、高可靠性等特点，非常适合数据可靠性要求高的I/O密集型应用，如MySQL、SQL Server、Oracle、PostgreSQL等中小型关系数据库场景，同时，对于数据可靠性要求高的中小型开发测试环境同样适用。

I/O优化是什么概念？能将存量的ECS实例升级为I/O优化的实例吗？

I/O优化为实例与云盘之间提供更好的网络能力，可保证SSD云盘存储性能的发挥。对于I/O优化的实例，挂载SSD云盘时能够获得SSD云盘的全部存储性能。

已购买的存量ECS实例不支持I/O优化。对于非I/O优化的实例，不支持通过升级的方式转变为I/O优化的实例。

是否支持将原普通云盘更换成SSD云盘？

由于SSD云盘采用全SSD的存储介质实现，不支持将存量的普通云盘直接更换为SSD云盘。

如何购买SSD云盘，I/O优化的实例及SSD云盘的价格是多少？

请参见SSD云盘及I/O优化实例的 [价格详情页](#)。

有些自定义镜像不支持创建I/O优化的实例，我该如何操作？

目前部分自定义镜像不支持创建I/O优化的实例。如果想要使用这类自定义镜像来创建I/O优化的实例，请提工单申请（需指定镜像名称）。

购买SSD云盘后是否支持升级？

支持升级和扩容。

使用了I/O优化实例和SSD云盘之后，Linux系统在分区挂载的时候报错，为什么？

SSD云盘在Linux系统内挂载点显示为/dev/vdx，与普通实例的挂载点/dev/xvdx有所区别，所以在执行之前的工具脚本时会出错。建议您在手动操作时写正确的挂载点/dev/vdx。

为什么我用FIO测试性能时，会导致实例宕机？

FIO测试工具支持裸盘分区、文件系统两种方式测试I/O性能。如果是直接测试裸盘分区，可能导致裸盘分区中的文件系统元数据被破坏掉，访问该裸盘分区中的文件时会失败，导致实例宕机。使用FIO的文件系统方式测试性能时，不存在上述问题，因此我们推荐使用FIO文件系统的方式测试I/O性能。

测试工具及方法

请参见[块存储性能](#)。

什么是共享块存储？

阿里云共享块存储是一种支持多台ECS实例并发读写访问的数据块级存储设备，具备多并发、高性能、高可靠等特性，单块共享块存储最多支持同时挂载到8台ECS实例。

为什么需要共享块存储？

在传统集群架构中，需要多个计算节点能够访问同一份数据，防止当一个或多个计算节点故障时导致业务中断，保证整个高可用集群对外可以持续提供业务服务。因此，重要的数据文件需要放置在共享块存储上，并通过集群文件系统对共享块存储进行统一管理。在前端多个计算节点并发进行读写访问时，保持数据在多个节点间的一致性。

共享块存储如何申请公测资格？

共享块存储正处于公测阶段，您需要[申请试用](#)。

共享块存储的售卖模式和公测范围

公测期间使用按量付费模式，但是免费使用。公测结束后支持按量付费模式和预付费模式。

公测期间，全地域均支持使用共享块存储。

共享块存储适用于哪些行业和业务场景？

共享块存储产品专为企业级客户的核心业务高可用架构而设计，适用于Shared-everything架构下对块存储设备的共享访问场景，比如政府、企业和金融行业常用的Oracle RAC数据库高可用架构，服务器High-availability cluster高可用架构。

如何正确使用共享块存储？

共享块存储产品本身并不提供集群文件系统，需要您自行安装集群文件系统来管理共享块存储。

正确使用共享块存储的方式是采用集群文件系统进行块设备的统一管理，例如GFS、GPFS等。典型Oracle RAC业务场景中推荐采用ASM统一管理存储卷和文件系统。

如果只是将共享块存储挂载到多台ECS实例，但依旧使用常规文件系统来管理时，会造成磁盘空间分配冲突和数据文件不一致两个问题，具体如下：

- 磁盘空间分配冲突

如果一块共享块存储挂载到多个实例上，当实例A在写文件时，会查询文件系统和可用的磁盘空间，文件写入后会修改实例A上的空间分配记录，但不会修改其他实例的记录。因此，当实例B写入文件时，可能会将实例A已经分配出去的磁盘空间再次分配，造成磁盘空间分配冲突。

- 数据文件不一致

当实例A读取数据并记录在缓存中后，实例A上另一个进程来访问同样的数据就会直接从缓存中进行读取。但如果此时实例B修改了同样的数据，而实例A并不知道，依旧从缓存中读取数据，则会造成业务数据不一致。

可以跨地域挂载共享块存储吗？

不能。您只能将共享块存储挂载到同一地域、同一可用区的ECS实例上。

一台实例上最多挂载多少块共享块存储？

作数据盘用时，共享块存储与云盘共享数据盘配额，即，一台实例上最多只能挂载16块数据盘。

共享块存储的产品规格和性能是什么？

当前支持SSD共享块存储和高效共享块存储，具体规格和性能请参见[块存储性能](#)。

如何测试共享块存储性能？



说明：

使用FIO进行性能压测时，多个客户端iodepth数值总计不能超过384（4台实例同时压测，每个客户端iodepth不能超过96）。

参与压测的实例数量	性能测试项目	测试命令
2	测试随机写IOPS	<pre>FIO -direct=1 -iodepth=128 -rw=randwrite -ioengine=libaio -bs=4k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Rand_Write_Testing</pre>
	测试随机读IOPS	<pre>FIO -direct=1 -iodepth=128 -rw=randread -ioengine=libaio -bs=4k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Rand_Read_Testing</pre>
	测试写吞吐量	<pre>FIO -direct=1 -iodepth=64 -rw=write -ioengine=libaio -bs=64k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Write_PPS_Testing</pre>
	测试读吞吐量	<pre>FIO -direct=1 -iodepth=64 -rw=read -ioengine=libaio -bs=64k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Read_PPS_Testing</pre>
4	测试随机写IOPS	<pre>FIO -direct=1 -iodepth=96 -rw=randwrite -ioengine=libaio -bs=4k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Rand_Write_Testing</pre>
	测试随机读IOPS	<pre>FIO -direct=1 -iodepth=96 -rw=randread -ioengine=libaio -bs=4k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Rand_Read_Testing</pre>
	测试写吞吐量	<pre>FIO -direct=1 -iodepth=64 -rw=write -ioengine=libaio -bs=64k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Write_PPS_Testing</pre>
	测试读吞吐量	<pre>FIO -direct=1 -iodepth=64 -rw=read -ioengine=libaio -bs=64k -size=1G -numjobs=1 -runtime=1000 -group_reporting -filename=iotest -name=Read_PPS_Testing</pre>

- 如果您的ECS实例之间需要低网络访问延时，建议您将ECS实例创建在同一个可用区下。

单独购买的按量付费云盘只能挂载到同一可用区的ECS实例。

什么是独立云磁盘？

独立云磁盘，即单独购买的按量付费数据盘，而不是随实例创建的数据盘。它可以在同一个可用区内的不同ECS实例之间自由挂载和卸载，但不能跨地域、跨可用区挂载。独立云磁盘必须挂载到实例上，并分区格式化，才能使用。详情请参见[创建按量付费云盘](#)。

一块云盘可以挂载到多台ECS实例上吗？

不能。一块数据盘只能挂载到同一可用区的一台实例，不支持跨可用区挂载。

购买了按量付费云盘并挂载到ECS实例后，还需要执行挂载分区操作吗？

单独购买了按量付费数据盘之后，你需要将该数据盘挂载到实例上，并分区格式化。详情请参见[Linux格式化数据盘](#)和[Windows格式化数据盘](#)。

单独购买的按量付费数据盘如何收费？

按量收取费用，如果账号余额不足，会停止该数据盘的服务。

单独购买的按量付费数据盘能够挂载到包年包月实例上吗？

可以。

包年包月实例上的数据盘能卸载吗？

包年包月实例不支持直接卸载数据盘。数据盘到期时间和实例一致，随实例一起释放。如果想要释放数据盘，您可以先将预付费数据盘转换为按量付费数据盘，再卸载并释放数据盘。如何转换云盘的计费方式，请参见[转换云盘的计费方式](#)。

我已经操作过续费变配，在续费变配期内是否还能将预付费云盘转为按量付费云盘？

不能。您可以等到续费变配期结束后，再选择续费变配，转化为按量付费云盘。

为什么我单独创建的云盘和我的实例一起释放了？

单独创建了数据盘后，在挂载时可以选择该云盘是否随实例释放。您可以随时通过控制台或API更改这个选项。

为什么挂载云盘时找不到我想挂载的实例？

请确保您的实例没有被释放，并且实例和云盘处于同一个地域的同一个可用区。

删除云盘的时候，快照会被保留吗？

手动快照不会被删除。自动快照是否会被删除，取决于您是否设置了自动快照随磁盘释放，详情请参见[自动快照随磁盘释放](#)。

为什么我的云盘没有自动快照了？

云盘只有在使用中的状态（挂载到实例时）才会执行创建新的自动快照的策略。快照数量达到快照额度上限后，系统会自动删除最早创建的自动快照，手动快照不受影响。

一台实例能挂载多少块云盘？

作数据盘用时，云盘与共享块存储共享数据盘配额，即一台实例最多挂载16块数据盘。

卸载云盘（数据盘）时，云盘数据会丢吗？

- 在Windows操作系统下，为了保证数据完整性，强烈建议您暂停对该云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。
- 在Linux操作系统下，您需要登录实例中对该云盘执行umount命令，命令执行成功后再进入控制台卸载云盘。

系统盘能够卸载吗？

不能卸载系统盘，但您可以更换系统盘。详情请参见[更换系统盘#非公共镜像#](#)。

什么是设备名（挂载点）？

设备名（挂载点）是云服务器ECS上磁盘在磁盘控制器总线上的位置。所选配的设备名，在Linux操作系统下与磁盘设备号对应，在Windows操作系统下与磁盘管理器中的磁盘顺序一致。

重新初始化云盘时，我的快照会丢失吗？

不会。手动快照和自动快照都不会丢失。

更换系统盘时，我的快照会丢失吗？

不会。但是请注意，更换系统盘相当于对您的系统盘进行更换，磁盘ID会发生变化，而您原来的系统盘的快照（手动快照和自动快照）将不能用于回滚新的系统盘。原系统盘的手动快照仍旧保留（自动快照会被删除），可以创建自定义镜像。

可以使用快照单独创建云盘吗？

可以。在单独创建按量付费云盘时，选择已有的快照创建即可。新创建的数据盘与快照的容量大小及数据完全相同，容量大小不可更改。

数据盘挂载问题导致数据无法访问，应该如何排查？

如果无法访问Linux数据盘的数据，您可以参考以下排查思路：

1. 找到数据所在的云盘，并确认数据盘是否挂载在对应的ECS实例。

- 在控制台查看。详情请参见[查看云盘监控信息](#)
- 在实例内部，使用 `fdisk -l` 命令查看数据盘的分区信息是否正常，并使用 `df -h` 和 `mount | grep "<devpath>"` 命令查看挂载情况。

2. 用 `cat` 命令查看 `/etc/fstab` 文件，是否将两块云盘挂载到同一目录。

- 如果挂载在同一目录，先挂载的云盘会被后挂载的云盘替换，导致数据无法访问。建议将其中一块云盘挂载到另外一个目录下。
- 如果没有挂载在同一个目录下，但挂载信息仍显示在同一目录，使用 `ll` 命令查看挂载的两个目录是否存在连接。如果存在连接，建议用 `mkdir` 命令新建一个目录挂载，并测试是否能访问数据。

如何扩容系统盘？

您可以通过控制台或者调用 [ResizeDisk](#) 扩容系统盘。您也可以通过更换系统盘来变更系统盘的大小。

更换系统盘应该注意什么问题？

更换系统盘之前要创建快照和自定义镜像。在创建快照和自定义镜像时，请确保系统盘有足够的空间，建议预留 1 GiB 的空间，否则更换系统盘后，系统可能无法正常启动。

所有块存储都支持系统盘扩容吗？有地域限制吗？

高效云盘、SSD云盘、ESSD云盘支持系统盘扩容。所有的地域都支持扩容系统盘。

包年包月和按量付费的ECS实例都支持扩容系统盘吗？

包年包月和按量付费的ECS实例都支持扩容系统盘。

系统盘的容量范围是多少？最大能扩容到多少？

操作系统不同，容量范围不同。详情请参见[扩容概述](#)。

云服务器ECS续费变配后，不支持更换系统盘时指定系统盘容量吗？

包年包月实例完成[续费降配](#)操作后，只有进入新计费周期后才能扩容系统盘，当前计费周期内不支持扩容系统盘。

系统盘扩容之后是否支持再缩容？

系统盘扩容之后不支持再缩容。

数据盘能否再次划分出一个分区用作数据存储？

支持。您可以根据需求对数据盘进行多分区配置，建议使用系统自带的工具进行分区操作。

划分了多个分区的磁盘，做快照时是针对该分区的，还是针对磁盘的？

磁盘快照针对的是整块磁盘，而不是该磁盘的某几个分区。

磁盘二次分区有哪些注意事项？

出于数据安全考虑，操作前请[创建快照](#)备份目标磁盘，出现意外时可以进行及时回滚。

数据盘进行二次分区后，此时回滚快照后，数据盘是几个分区？

快照回滚是回滚到快照时间点，如果该时间点尚未进行二次分区，则没有多个分区。

如何利用快照创建云盘实现无损扩容数据盘？

在扩容数据盘时，若遇到磁盘原因导致无法无损地扩容时，可以通过临时购买一块按量付费云盘来存放数据，然后将数据盘彻底格式化来解决。操作步骤如下：

1. 为当前数据盘创建快照。
2. 前往[云盘购买页](#)，选择和ECS实例相同的地域及可用区，并选择用快照创建磁盘，使用上一步生成的快照。
3. 在ECS控制台中，将刚才购买的数据盘挂载到ECS实例上。
4. 登录ECS实例，运行mount命令将这块云盘挂载到系统中，并查看其中的文件是否和数据盘中的一致。
5. 确定数据一致后，先使用fdisk删除原有的分区表，然后再使用fdisk、mkfs.ext3等命令重新对数据盘执行分区格式化操作，使数据盘的可用空间变为扩容之后的大小。
6. 运行cp -R命令将单独购买的数据盘中的数据全部拷回原数据盘。可以加上--preserve=all参数，复制时保留文件的属性。
7. 运行umount命令卸载单独购买的数据盘。
8. 登录ECS控制台，从实例上卸载这块单独购买的数据盘，并释放它。

普通云盘和SSD云盘添加挂载信息时需要注意什么？

Linux系统添加数据盘时，按照[Linux格式化数据盘](#)步骤添加分区信息时要注意，`/dev/xvdb1`是普通云盘的挂载点，如果您使用的是SSD云盘，挂载点为`/dev/vdb1`。如果添加了错误的信息，使用mount -a命令将无法挂载。解决方法如下：

1. 运行fdisk -l查看一下具体的数据盘信息。
2. 查看/etc/fstab里面添加的信息是否对应。请注意不要重复添加挂载信息，这样会导致开机重复挂载时系统无法启动。
3. 使用vim修改/etc/fstab文件。
4. 将错误的信息注释或者删除，添加对应的挂载信息后，再运行mount -a测试是否能挂载成功。

Linux实例初始化系统盘后如何重新挂载数据盘？

在Linux实例中，重新初始化系统盘不会改变数据盘里的内容，但是数据盘的挂载信息会丢失，因此重启Linux实例后，请参考以下步骤创建新的挂载点信息并挂载数据盘分区。

假设初始化系统盘之前，实例上挂载的数据盘分区名称为`/dev/vdb1`，挂载点名称为`/InitTest`。

1. 运行命令`mount`查看数据盘挂载信息。返回结果中没有`/dev/vdb1`的信息。
2. 运行命令`fdisk -l`查看数据盘分区信息。
3. 运行命令`mkdir /InitTest`重新创建数据盘分区的挂载点。这里的挂载点名称必须与系统盘初始化之前`/dev/vdb1`的挂载点名称保持一致。您可以通过`cat /etc/fstab`命令查看原来的挂载点名称。
4. 运行命令`mount /dev/vdb1 /InitTest`重新挂载数据盘分区。
5. 运行命令`df -h`查看挂载结果。
6. 检测`/dev/vdb1`是否能自动挂载。
 - a. 运行命令`umount /dev/vdb1`卸载`/dev/vdb1`。
 - b. 运行命令`mount`查看挂载信息。如果卸载成功，返回结果中不会有`/dev/vdb1`的信息。
 - c. 运行命令`mount -a`自动挂载`/dev/vdb1`。
 - d. 运行命令`mount`查看挂载信息。如果自动挂载成功，返回结果中会有`/dev/vdb1`的信息。

重启ECS Linux实例、初始化系统后数据盘不见了怎么办？

问题现象：Linux实例重启或初始化系统之后，再登录实例器运行`df -h`查看磁盘挂载，发现数据盘不见了。

问题原因：

- 重启实例：因为重启之前，在挂载数据盘的时候没有将挂载信息写入到`/etc/fstab`，导致重启实例之后系统没有自动挂载数据盘。
- 初始化系统：因为初始化系统盘会重置`/etc/fstab`文件，导致初始化之后系统启动没有自动挂载数据盘。

解决方案：

1. 再次手动运行`mount /dev/xvdb1`命令挂载数据盘。
2. 运行`mount`命令查看数据盘分区`/dev/xvdb1`的文件格式。

3. 本示例中，查看到/dev/xvdb1是ext3格式。执行以下命令将数据盘挂载信息写入到/etc/fstab中：

```
echo '/dev/xvdb1 /data ext3 defaults 0 0' >> /etc/fstab
```

操作完成后，再重启实例，不会再出现数据盘未加载的情况。但是，下一次再初始化系统之后仍旧需要执行上述操作。

重启ECS Linux实例后数据丢失怎么办？

问题现象：重启实例后发现/alidata目录所有数据丢失。

问题原因：运行df -h查看，发现没有进行挂载。

解决方案：



说明：

本方案以I/O优化实例为例。如果是非I/O优化实例，磁盘分区为/dev/xvdb1，请您根据实际情况进行操作。

1. 运行fdisk -l命令查看到有一块未挂载的硬盘。
2. 运行命令mount /dev/vdb1 /alidata进行挂载。

挂载后恢复正常。

为避免下次出现同样问题，您也可以在/etc/fstab中设置开机自动挂载。

ECS Linux扩容云盘提示Bad magic number in super-block while trying to open /dev/xvdb1

- 问题现象

e2fsck -f /dev/xvdb扩容格式化磁盘时提示报错Bad magic number in super-block while trying to open /dev/xvdb1

- 问题原因

可能是由于没有磁盘分区导致。

- 解决方案

分别执行e2fsck -f /dev/xvdb和resize2fs /dev/xvdb实现扩容。然后将磁盘挂载使用。

回滚磁盘时，报错“执行回滚磁盘需要停止实例，并且确保当前磁盘没有创建中的快照和没有更换过操作系统。”

- 问题现象

回滚云盘时，报错“执行回滚磁盘需要停止实例，并且确保当前磁盘没有创建中的快照和没有更换过操作系统。”

- 问题原因

通常是由于磁盘属性或磁盘状态不符等原因导致。

- 解决方案

- 检查相关快照所关联的实例是否做过[更换系统盘#非公共镜像#](#)操作。

如果更换了操作系统，实例的系统盘会随新镜像自动重新创建，系统盘ID会改变。所以，实例原有的快照将无法再用于回滚。但您可以基于相关快照[使用快照创建自定义镜像](#)，然后再通过[更换系统盘#非公共镜像#](#)方式指定该自定义镜像，将实例切换到相应的快照状态。

- 检查实例是否已经正常停止。

只有已停止的实例才能进行磁盘回滚操作。您可以在控制台的实例列表页查看实例状态。

- 检查待回滚快照归属的磁盘是否有正在进行的快照任务。

为保障数据一致性，如果快照归属磁盘有正在进行的快照任务，将不允许进行回滚操作。您可以在实例详情页的本实例快照中，查看是否存在进度不是100%，状态处于未完成的快照任务。

如果您需要强制终止相关快照任务以便尽快进行磁盘回滚操作，可以勾选相应快照，并单击删除快照终止快照任务。

如何跨实例拷贝数据？

根据不同的操作系统，您可以参考以下方法跨ECS实例拷贝数据：

- Linux实例之间数据拷贝

- lrzsz方式

登录实例，安装lrzsz，然后执行rz上传，sz下载。

您也可以先用sz下载到本地，再rz上传到另一台实例上。

- FTP方式

使用sftp工具，建议用root账号登录进行下载和上传。

- wget命令

把文件或者文件夹压缩后，放在web目录下生成一个下载URL，在另一台实例上用wget命令进行下载。

- Linux与Windows之间数据拷贝

建议在Linux实例上使用sftp，先下载到本地，然后使用FTP上传到Windows实例。

- Windows之间数据拷贝

- 使用FTP的方式

- 两台实例都登录阿里旺旺进行传输

云盘和实例在不同的可用区，可以挂载吗？

不可以。您只能在同一可用区内的不同ECS实例间自由挂载和卸载按量付费云盘。

为Linux实例购买了数据盘，但是系统中看不到怎么办？

如果是单独购买的按量付费数据盘，您需要[分区格式化](#)、[挂载](#)后才能使用和看到空间。

如何压测ESSD云盘的性能？

由于压测云盘的性能时，云盘本身以及压测条件都起着重要的作用，因此本文提供了如何配置合适的条件压测ESSD云盘性能的示例，并测试出100万IOPS。为了充分发挥出多核多并发的系统性能，压测出100万IOPS性能指标，您可以参考以下建议测试ESSD云盘的性能。



警告：

测试裸盘可以获得真实的块存储盘性能，但直接测试裸盘会破坏文件系统结构，请在测试前提前[创建快照](#)做好数据备份。建议您只在新购无数据的ECS实例上使用工具测试块存储性能，避免造成数据丢失。

镜像：您可以使用阿里云官方镜像中高版本的Linux镜像版本。例如，CentOS 7.4/7.3/7.2 64位和AliyunLinux 17.1 64位操作系统。由于对应的驱动还不够完善，不推荐使用其他低版本的Linux镜像和Windows镜像。

工具：您可以使用FIO作为云盘性能标准的测试工具。

以实例规格ecs.g5se.18xlarge为例，同时ESSD云盘的设备名为/dev/vdb，示范通过随机写（randwrite）测试ESSD云盘的性能。

1. 远程连接Linux实例。
2. 运行以下命令安装libaio和FIO。

```
sudo yum install libaio -y
sudo yum install libaio-devel -y
sudo yum install fio -y
```

3. 运行cd /tmp切换路径。
4. 运行vim test100w.sh新建脚本文件，并粘贴以下内容，脚本为随机写randwrite IOPS性能测试示例。

```
function RunFio
{
    numjobs=$1 # 实例中的测试线程数，如示例中的 8
    iodepth=$2 # 同时发出I/O数的上限，如示例中的 64
    bs=$3      # 单次I/O的块文件大小，如示例中的 4K
    rw=$4      # 测试时的读写策略，如示例中的 randwrite
    filename=$5 # 指定测试文件的名称，如示例中的 /dev/vdb
    nr_cpus=`cat /proc/cpuinfo |grep "processor" |wc -l`
    if [ $nr_cpus -lt $numjobs ];then
        echo "Numjobs is more than cpu cores, exit!"
        exit -1
    fi
    let nu=$numjobs+1
    cpulist=""
    for ((i=1;i<10;i++))
    do
        list=`cat /sys/block/vdb/mq/*/cpu_list | awk '{if(i<=NF) print $i;}' i="$i" | tr -d ',' | tr '\n' ','`
        if [ -z $list ];then
            break
        fi
        cpulist=${cpulist}${list}
    done
    spincpu=`echo $cpulist | cut -d ',' -f 2-{$nu}`
    echo $spincpu
    fio --ioengine=libaio --runtime=30s --numjobs=${numjobs} --iodepth=
${iodepth} --bs=${bs} --rw=${rw} --filename=${filename} --time_based
=1 --direct=1 --name=test --group_reporting --cpus_allowed=$spincpu
--cpus_allowed_policy=split
}
echo 2 > /sys/block/vdb/queue/rq_affinity
sleep 5
RunFio 8 64 4k randwrite /dev/vdb
```



说明：

· 因测试环境而异，脚本中您需要修改的命令行有：

- 命令行 `list=`cat /sys/block/vdb/mq/*/cpu_list | awk '{if(i<=NF) print $i;}' i="$i" | tr -d ',' | tr '\n' ' '`` 中的 `vdb`。
- 命令行 `RunFio 8 64 4k randwrite /dev/vdb` 中的 `8`、`64`、`4k`、`randwrite` 和 `/dev/vdb`。

· 直接测试裸盘会破坏文件系统结构。如果云盘上的数据丢失不影响业务，可以设置 `filename=[设备名, 如本示例中的/dev/vdb]`。否则，请设置为 `filename=[具体的文件路径, 比如/mnt/test.image]`。

5. 运行 `sh test100w.sh` 开始测试 ESSD 云盘性能。

```
[root@... ~]# sh test100w.sh
4,8,12,16,20,24,28,1
test: (g=0): rw=randwrite, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth=64
...
fio-3.0
Starting 8 processes
Jobs: 8 (f=8): [w(8)][100.0%][r=0KiB/s,w=3965MiB/s][r=0,w=1015k IOPS][eta 00m:00s]
test: (groupid=0, jobs=8): err= 0: pid=9565: Thu Jan 11 17:37:00 2018
write: IOPS=1006k, BW=3931MiB/s (4122MB/s)(115GiB/30047msec)
    slat (usec): min=2, max=52179, avg= 4.22, stdev= 9.78
    clat (usec): min=82, max=624296, avg=500.05, stdev=814.37
    lat (usec): min=86, max=624300, avg=504.46, stdev=814.43
    clat percentiles (usec):
      1.00th=[ 174], 5.00th=[ 210], 10.00th=[ 235], 20.00th=[ 273],
      30.00th=[ 306], 40.00th=[ 338], 50.00th=[ 383], 60.00th=[ 441],
      70.00th=[ 523], 80.00th=[ 660], 90.00th=[ 930], 95.00th=[ 1139],
      99.00th=[ 1582], 99.50th=[ 1909], 99.90th=[ 3458], 99.95th=[ 4424],
      99.99th=[17957]
    bw ( KiB/s): min=256936, max=526637, per=12.53%, avg=504445.98, stdev=33299.37, samples=480
```

脚本解读：

· 块设备参数

测试实例时，脚本中的命令 `echo 2 > /sys/block/vdb/queue/rq_affinity` 是将 ECS 实例中的块设备中的参数 `rq_affinity` 值修改为 2：

- 参数 `rq_affinity` 的值为 1 时，表示块设备收到 I/O 完成 (I/O Completion) 的事件时，这个 I/O 被发回处理这个 I/O 下发流程的 vCPU 所在 Group 上处理。在多线程并发的情况下，I/O Completion 就可能集中在某一个 vCPU 上执行，这样会造成瓶颈，导致性能无法提升。
- 参数 `rq_affinity` 的值为 2 时，表示块设备收到 I/O Completion 的事件时，这个 I/O 会在当初下发的 vCPU 上执行。在多线程并发的情况下，就可以完全充分发挥各个 vCPU 的性能。
- 绑定对应的 vCPU
 - 普通模式下，一个设备 (Device) 只有一个请求列表 (Request-Queue)。在多线程并发处理 I/O 的情况下，这个唯一的 Request-Queue 就是一个性能瓶颈点。
 - 最新的多队列 (Multi-Queue) 模式下，一个设备 (Device) 可以拥有多个处理 I/O 的 Request-Queue，可以充分发挥后端存储的性能。如果您有 4 个 I/O 线程，您需要将 4 个线

程分别绑定在不同的Request-Queue对应的CPU Core上，这样就可以充分利用Multi-Queue提升性能。

为了充分发挥设备（Device）的性能，需要将I/O线程分发到不同的Request-Queue上处理。[脚本](#)中通过以下命令，分别将几个jobs绑定不同的CPU Core上，其中vd为您的云盘设备名，例如，/dev/vdb。

```
fio -ioengine=libaio -runtime=30s -numjobs=${numjobs} -iodepth=${iodepth} -bs=${bs} -rw=${rw} -filename=${filename} -time_based=1 -direct=1 -name=test -group_reporting -cpus_allowed=$spincpu -cpus_allowed_policy=split
```

FIO提供了参数cpusallowed以及cpus_allowed_policy来绑定vCPU。以上命令一共运行了几个jobs，分别绑定在几个CPU Core上，分别对应着不同的Queue_Id。

关于如何查看Queue_Id绑定的cpu_core_id，您可以：

- 运行ls /sys/block/vd/mq/查看设备名为vd云盘的Queue_Id，例如vdb。
- 运行cat /sys/block/vd/mq//cpu_list查看对应设备名为vd*云盘的Queue*绑定到的cpu_core_id。