

Vastbase G100 V2.2

(Build 5)

安装指南

【版权声明】

©2007-2022 北京海量数据技术股份有限公司 版权所有

本文档著作权归 **北京海量数据技术股份有限公司**（简称“海量数据”）所有，未经海量数据事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

北京海量数据技术股份有限公司保留所有的权利。

【商标声明】



及其它海量数据产品和服务相关的商标均为 **北京海量数据技术股份有限公司** 及其关联公司所有。

本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

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

目录

1. 安装概述	1
2. 安装准备	1
2.1. 操作系统版本要求	1
2.2. 服务器配置建议	1
2.2.1. 开发测试环境	1
2.2.2. 生产环境	1
3. 操作系统配置	2
3.1. 关闭防火墙	2
3.2. 设置时区和时间	3
3.3. 设置网卡 MTU 值	3
3.4. 设置 root 允许远程登录	3
3.5. 操作系统参数	3
3.6. 创建用户	4
3.7. 资源限制	5
3.8. 添加排序规则	5
4. DB Installer 图形化界面安装	5
4.1. 安装过程	5
4.1.1. 安装环境依赖	5
4.1.2. 单机数据库安装步骤	5
4.2. 启停数据库	8
4.3. 登录数据库	9
4.4. 卸载数据库	9
5. 字符安装	14
5.1. 安装过程	14
5.1.1. 注意事项	14
5.1.2. 校验安装包完整性	14
5.1.3. 安装步骤	14
● 不实例化数据库	14
● 实例化数据库	17
5.2. 初始化数据库运行环境	21
5.3. 启停数据库	21
5.4. 登录数据库	21
5.5. 执行卸载	21
6. 镜像安装	22
6.1. 镜像使用	22
6.1.1. 简介	22

6.1.2. 导入镜像.....	22
6.1.3. 启动容器.....	22
6.1.4. 进入容器.....	23
6.2. 数据库使用.....	23
6.2.1. 初始化数据库.....	23
● 切换到 vastbase 用户：su vastbase.....	23
● 创建数据库初始化目录.....	23
● 初始化数据库.....	23
● 检查 license 文件.....	24
● 启动数据库.....	24
● 连接数据库.....	24
6.2.2. 配置远程连接.....	25
● 在 pg_hba.conf 中添加要连接数据库的客户端 ip.....	25
● 在 postgresql.conf 中修改监听地址为*（表示监听所有请求）。.....	25
6.2.3. 注意事项.....	25
7. 升级说明.....	25
7.1. 旧版本升级.....	26
7.1.1. 升级准备.....	26
● 设置免密登录.....	26
● 修改 GUC 参数配置文件.....	26
● 将 om 脚本拷贝到 bin 目录下.....	26
● 将数据库安装目录更改为软连接.....	26
● 修改 version.cfg 和 upgrade_version.....	27
● 修改参数.....	27
7.1.2. 上传安装包和配置文件.....	27
● 创建新包目录.....	27
● 解压安装包.....	28
● 创建配置文件.....	29
7.1.3. 执行 gs_preinstall 准备升级.....	29
7.1.4. 执行 gs_upgradectl 进行升级.....	31
● 执行 gs_om.....	31
● 设置 OM_GAUSSLOG 环境变量.....	32
● 执行升级命令.....	32
● 升级验证.....	33
● 数据库回滚.....	33
● 提交升级.....	33
7.2. 新版本升级.....	34

7.2.1. 升级准备.....	34
● 设置免密登录.....	34
● 修改 GUC 参数配置文件.....	34
● 修改 version.cfg 和 upgrade_version.....	34
● 修改参数.....	35
7.2.2. 上传安装包和配置文件.....	35
● 创建新包目录.....	35
● 解压安装包.....	35
● 上传配置文件.....	36
7.2.3. 执行 gs_preinstall 准备升级.....	36
7.2.4. 执行 gs_upgradectl 进行升级.....	37
● 执行 gs_om.....	37
● 设置 OM_GAUSSLOG 环境变量.....	38
● 执行升级命令.....	38
● 升级验证.....	38
● 数据库回滚.....	39
● 提交升级.....	39

1.安装概述

Vastbase 支持单机部署和流复制集群两种方式。单机部署时，可在一个主机部署多个数据库实例，但为了数据安全，不建议用户这样部署。流复制集群部署支持一台主机和最少一台备机。

单机部署和流复制部署均支持原生 PostgreSQL 和兼容 Oracle 模式。原生 PG 模式下，您可以安装 Vastbase 数据库并正常使用。兼容模式是在原生 PG 模式的基础上做了兼容性增强，安装兼容包后会兼容 Oracle 数据库的语法，便于习惯主流数据库的用户使用。

2.安装准备

2.1.操作系统版本要求

LINUX 操作系统	处理器	版本
Red Hat Enterprise Linux	x86_64	7.3 及以上
CentOS	ARM,x86_64	7.3 及以上
Oracle Enterprise Linux	ARM	7.3 及以上
Kylin	ARM	10 及以上
Linx	x86_64	6.0.80

2.2.服务器配置建议

2.2.1.开发测试环境

名称	CPU	内存	本地存储	网络	机器数量
VastBase	2 核+	4GB+	SAS, 100G	千兆	1

2.2.2.生产环境

列出了 Vastbase 服务器应具备的最低配置硬件要求。在实际产品中，硬件配置的规划需考虑数据规模及所期望的数据库响应速度。请根据实际情况进行规划。

项目	配置描述
最小内存	功能调试 32GB 以上。 性能测试和商业部署时，单实例部署建议 128GB 以上。

	复杂的查询对内存的需求量比较高，在高并发场景下，可能出现内存不足。此时建议使用大内存的机器，或使用负载管理限制系统的并发。
CPU	<p>功能调试最小 1×8 核 2.0GHz。</p> <p>性能测试和商业部署时，单实例部署建议 1×16 核 2.0GHz。</p> <p>CPU 超线程和非超线程两种模式都支持。但是，Vastbase 各节点的设置需保持一致</p>
硬盘	<p>用于安装 Vastbase 的硬盘需最少满足如下要求：</p> <ul style="list-style-type: none"> ➤至少 1GB 用于安装 Vastbase 的应用程序包。 ➤每个主机需大约 300MB 用于元数据存储。 ➤预留 70%以上的磁盘剩余空间用于数据存储。 <p>建议系统盘配置为 Raid1，数据盘配置为 Raid5，且规划 4 组 Raid5 数据盘用于安装 Vastbase。有关 Raid 的配置方法在本手册中不做介绍。请参考硬件厂家的手册或互联网上的方法进行配置，其中 Disk Cache Policy 一项需要设置为 Disabled，否则机器异常掉电后有数据丢失的风险。</p> <p>Vastbase 支持使用 SSD 盘作为数据库的主存储设备，支持 SAS 接口和 NVME 协议的 SSD 盘，以 RAID 的方式部署使用。</p>

3.操作系统配置

3.1.关闭防火墙

为了在防火墙开启的状态下，确保 Vastbase 的正常使用。用户需要将同 Vastbase 相关的服务、协议、IP 以及端口添加到 Vastbase 各主机的防火墙白名单中。

步骤 1 关闭防火墙

```
[root@plat1 ~]# systemctl stop firewalld.service
```

步骤 2 修改/etc/selinux/config 文件中的“SELINUX”值为“disabled”

```
vim /etc/selinux/config
修改“SELINUX”的值“disabled”
SELINUX=disabled
```

步骤 3 永久关闭防火墙

```
[root@plat1 ~]# systemctl disable firewalld.service
```

3.2.设置时区和时间

将各数据库节点的时区设置为相同时区，可以将/usr/share/zoneinfo/目录下的时区文件拷贝为/etc/localtime 文件。

```
cp /usr/share/zoneinfo/$主时区/$次时区 /etc/localtime
```

使用 date -s 命令将数据库节点的时间设置相同，举例如下。

```
date -s "2020-08-03 14:15:00"
```

3.3.设置网卡 MTU 值

将各数据库节点的网卡 MTU 值设置为相同大小。MTU 值推荐 8192，要求不小于 1500

```
ifconfig 网卡编号 mtu 值
```

例如

```
Ifconfig eth0 mtu 8192
```

3.4.设置 root 允许远程登录

在安装 Vastbase 安装时需要 root 帐户远程登录访问权限，本章介绍如何设置使用 root 用户远程登录。

步骤 1 修改 PermitRootLogin 配置，允许用户远程登录。

```
打开 sshd_config 文件
vim /etc/ssh/sshd_config
修改权限配置，可以使用以下两种方式实现：
注释掉 "PermitRootLogin no"
#PermitRootLogin no
将 "PermitRootLogin" 改为 "yes"
PermitRootLogin yes
执行:wq 保存并退出编辑页面
```

步骤 2 步骤 2 重启 ssh 命令生效

```
service sshd restart
```

3.5.操作系统参数

```
# vi /etc/sysctl.conf

#在 linux6 和 7 上，使用 aio 的话，需要设置它，适应异步 IO，可以不需要重新修改该值
fs.aio-max-nr = 1048576
#在 linux6 和 7 上，允许打开文件数，pg 参数 max_files_per_process 对应，不重新修改该值
fs.file-max = 76724600
# 信号量，ipcs -l 或 -u 查看，每 16 个进程一组，每组信号量需要 17 个信号量。不重新修改该值
(四个参数，第一个参数*第四个参数= 第二个参数， 第一个参数和第三个参数相等)
第一个代表信号量， 第四个代表是组， 第三个参数设置大于等于 17
kernel.sem = 4096 2147483647 2147483646 512000
# 所有共享内存段相加大小限制 (建议内存的 80%，单位 page，查看 page 大小 getconf PAGE_SIZE)
```



```

# 例如: page 大小 4096, 在 128G 内存设置 80% 的值如下
kernel.shmall = 26843545    (需计算)
# 最大单个共享内存段大小 (建议为大于 shared_buffer 值), 单位 bytes.
kernel.shmmax = 51539607552 (需计算)
# 一共能生成多少共享内存段, 每个 PG 数据库集群至少 2 个共享内存段, 不重新修改该值
kernel.shmmni = 819200
# iptables 防火墙链表相关, 不重新修改该值
net.core.netdev_max_backlog = 10000
# 网络接收 buffer 大小, 单位 bytes, 不重新修改该值
net.core.rmem_default = 262144
# 允许最大网络接收 buffer 大小, 单位 bytes, 不重新修改该值
net.core.rmem_max = 4194304
# 网络传输 buffer 大小, 单位 bytes, 不重新修改该值
net.core.wmem_default = 262144
# 网络传输最大 buffer 大小, 单位 bytes, 不重新修改该值
net.core.wmem_max = 4194304
# socket 监听数, 默认 128, 不重新修改该值
net.core.somaxconn = 4096
# 加快僵尸进程回收速度, 不重新修改该值
net.ipv4.tcp_fin_timeout = 5
# 系统脏页到达这个值, 脏页刷到磁盘, 当前设置 390M, 如果磁盘 IO 能力是 512M/s
vm.dirty_background_bytes = 409600000 (需计算)
# 比这个值老的脏页, 将被刷到磁盘。3000 表示 30 秒。不重新修改该值
vm.dirty_expire_centisecs = 3000
# 如果系统进程刷脏页太慢, 使得系统脏页超过内存 80% 时, 则用户进程如果有写磁盘的操作 (如 fsync, fdatasync 等调用),
则需要主动把系统脏页刷出, 不重新修改该值
vm.dirty_ratio = 80
# 有效防止用户进程刷脏页, 在单机多实例, 并且使用 CGROUP 限制单实例 IOPS 的情况下非常有效。
# pdflush (或其他) 后台刷脏页进程的唤醒间隔, 100 表示 1 秒。 不重新修改该值
vm.dirty_writeback_centisecs = 50
# 在分配内存时, 设置为 0, vm.overcommit_ratio 参数可以不设置, 不重新修改该值
vm.overcommit_memory = 0
# 关闭交换分区, 数据库服务器不建议使用 swap, 不重新修改该值
vm.swappiness = 0
# 限制本地动态端口分配范围, 防止占用监听端口, 不重新修改该值
net.ipv4.ip_local_port_range = 40000 65535
# 建议 shared buffer 设置超过 64GB 时 使用大页, 查看页大小 more /proc/meminfo |grep Hugepagesize, 大于数据库
需要的 shared_buffer 内存即可。
vm.nr_hugepages = 66536    (需计算)
# 单个进程的打开句柄不能大于 fs.nr_open, 对于有很多对象 (表、视图、索引、序列、物化视图等) 的 PostgreSQL 数据库,
建议设置为 2000 万, 不重新修改该值
fs.nr_open=20480000

```

修改完成后执行 `sysctl -p` 使参数生效

3.6.创建用户

```

[root@mypgdb Packages]# groupadd -g 1005 vastbase
[root@mypgdb Packages]# useradd -g 1005 -u 1005 vastbase
[root@mypgdb Packages]# passwd vastbase

```

3.7.资源限制

```
vi /etc/security/limits.conf
vastbase soft nproc unlimited
vastbase hard nproc unlimited
vastbase soft stack unlimited
vastbase hard stack unlimited
vastbase soft core unlimited
vastbase hard core unlimited
vastbase soft memlock unlimited
vastbase hard memlock unlimited
```

3.8.添加排序规则

在初始化实例时，数据库会将当前操作系统所有 collation 语言排序规则信息（可以通过 Linux 命令 `locale -a` 查看）加入到 `pg_collation` 表中。如果语言排序规则是在初始化实例之后才加入系统的，则不能自动加入到 `pg_collation` 系统表中，在创建表等对象时如果指定了该排序规则，则会报该排序规则不存在。

例如 linux 操作系统默认安装时只有 C、POSIX、zh_CN.utf8 等语言排序规则，如果需要数据库支持 en_US.utf8 等其他语言排序规则，需要在初始化实例之前安装其他语言排序规则。如果是在初始化实例后再安装的语言排序规则，可以通过创建一个新的数据库，指定 `lc_collate`、`lc_ctype` 来使数据库支持新增加的语言排序规则。

操作系统添加排序规则命令：使用 root 用户执行

```
# 在 Linux 操作系统中，可以通过 dpkg-reconfigure locales 来安装新的语言排序规则
# 也可以使用以下命令安装 en_US.utf8
localedef -v -c -i en_US -f UTF-8 en_US.UTF-8
```

4.DB Installer 图形化界面安装

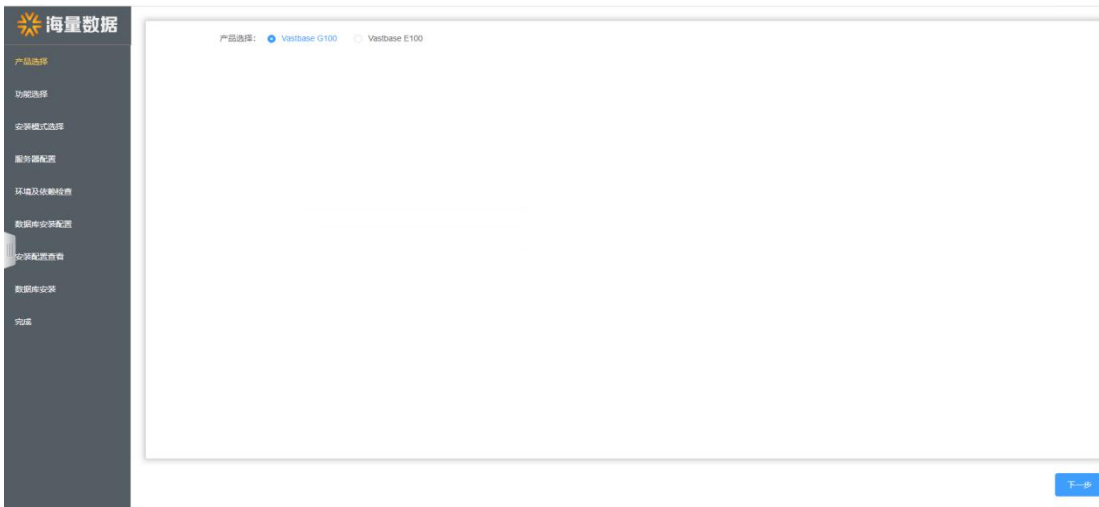
4.1.安装过程

4.1.1.安装环境依赖

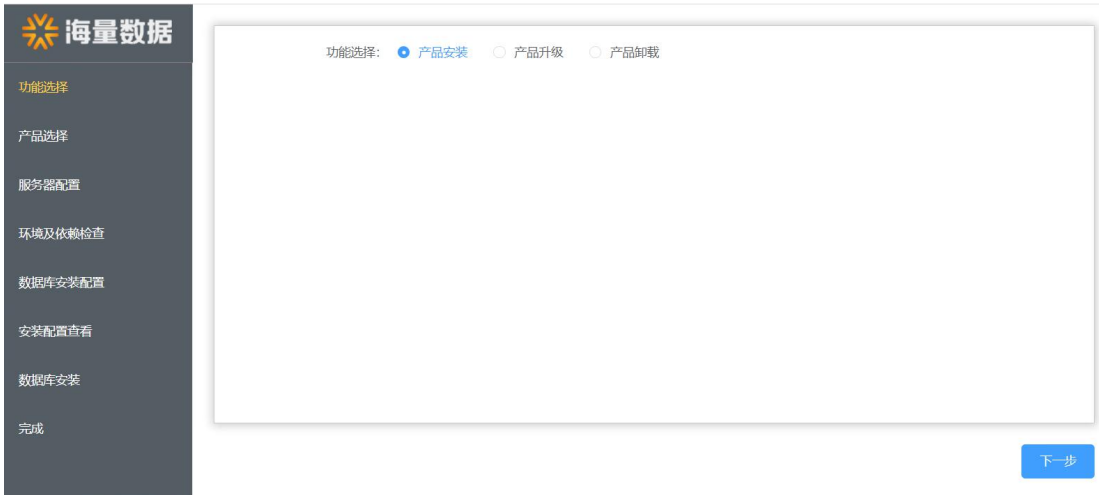
已安装 DB Installer 系统，详细安装步骤可参照《[VASTDATA]Vastbase G100 V1.1 图形化安装指南.docx》文档。通过浏览器访问 DB Installer 系统，例如:172.16.101.141:7070

4.1.2.单机数据库安装步骤

步骤 1 产品选择：可选 Vastbase G100 或 Vastbase E100，目前 G100 支持安装、卸载和升级，E100 只支持单机或集群的安装



步骤 2 功能选择：选择“产品安装”，点击下一步



步骤 3 产品选择：选择“数据库”，点击下一步



步骤 4 服务器配置：填写相应的服务器信息，测试 SSH 连接，连接通过后点击下一步。服务器用户名需要使用 root 等超级用户。

海量数据

功能选择
产品选择
服务器配置
环境及依赖检查
数据库安装配置
安装配置查看
数据库安装
完成

* 服务器地址: 172.16.103.82
* 服务器端口: 22
* 服务器用户名: root
* 服务器密码:

测试SSH连接 上一步 下一步

步骤 5 环境依赖性检查：检查通过后，点击下一步

海量数据

功能选择
产品选择
服务器配置
环境及依赖检查
数据库安装配置
安装配置查看
数据库安装
完成

-----安装包完整性检查-----
package check:ok

系统配置信息

操作系统 : Kylin Linux Advanced Server release V10 (Tercel)
CPU架构 : aarch64
CPU核数 : 128
内存大小 : 783943MB
当前用户名 : root

操作系统版本及CPU架构符合产品要求

依赖检查

zlib:
建议版本:1.2.7
实际安装版本: 1.2.11

重新检测 上一步 下一步

步骤 6 数据库安装配置：填写数据库安装的相关信息。必填项为实例用户密码，需要满足密码要求，即包含大小字母及数字，长度 8 至 16 位。下方的数据库配置如需更改配置，需先勾选“手动配置数据库参数”选项。配置完成后点击下一步。

海量数据

功能选择
产品选择
服务器配置
环境及依赖检查
数据库安装配置
安装配置查看
数据库安装
完成

* 安装路径: /usr/local/vastbase 是否创建实例:
* 创建实例用户: vastbase
* 实例用户密码:

* 数据库存放路径: /home/vastbase/vastbase_data
数据库日志存放路径:
手动配置数据库参数:

数据库配置
* 监听地址(listen_addresses): *

上一步 下一步

注意，单数据库安装可以填写*号，默认为*号。可以不勾选是否创建实例。

步骤 7 安装配置查看：显示所配置的实例选项，确认无误则点击下一步。

海量数据

- 功能选择
- 产品选择
- 服务器配置
- 环境及依赖检查
- 数据库安装配置
- 安装配置查看**
- 数据库安装
- 完成

```

-----安装选项-----
安装路径:/usr/local/vastbase
-----创建实例选项-----
创建用户:vastbase
数据库存放路径:/home/vastbase/vastbase_data
手动配置数据库参数:是
监听地址:*
监听端口:5432
最大连接数:500
最大缓存:7839MB
共享内存:3135MB
工作内存:4MB
预写日志缓冲区大小:-1MB
数据磁盘同步:是

```

上一步 下一步

步骤 8 数据库安装：安装完成后，可以点击“启动数据库”启动数据库或者点击下一步。

海量数据

- 功能选择
- 产品选择
- 服务器配置
- 环境及依赖检查
- 数据库安装配置
- 安装配置查看
- 数据库安装**
- 完成

```

ip:172.16.103.82
-----创建数据库用户vastbase-----
创建数据库用户vastbase成功!
-----安装Vastbase-----
正在解压...
安装完成。
-----配置用户vastbase环境-----
export PGUSER=vastbase
export PGDATA=/home/vastbase/vastbase_data
export LD_LIBRARY_PATH=/usr/local/vastbase/lib:$LD_LIBRARY_PATH
export GAUSSHOME=/usr/local/vastbase
export PATH=/usr/local/vastbase/bin:$PATH
用户环境配置完成
-----
vastbase installed:ok
-----
-----创建实例-----
The files belonging to this database system will be owned by user "vastbase".
This user must also own the server process.

The database cluster will be initialized with locale "en_US.UTF-8".
The default text search configuration will be set to "english".

```

启动数据库 上一步 下一步

步骤 9 完成：点击回到首页，完成安装

海量数据

- 功能选择
- 产品选择
- 服务器配置
- 环境及依赖检查
- 数据库安装配置
- 安装配置查看
- 数据库安装
- 完成**

```

-----数据库信息-----
数据库安装在:/usr/local/vastbase
-----实例选项-----
实例用户:vastbase
数据库实例路径:/home/vastbase/vastbase_data
数据库实例日志存放路径:
监听地址:*
监听端口:5432
恭喜您! 成功安装数据库!!! 请关闭安装页面。。。

```

回到首页

4.2.启停数据库

使用安装过程中创建的用户，执行命令启停数据库。

```
vb_ctl <start/stop/restart>
```

4.3.登录数据库

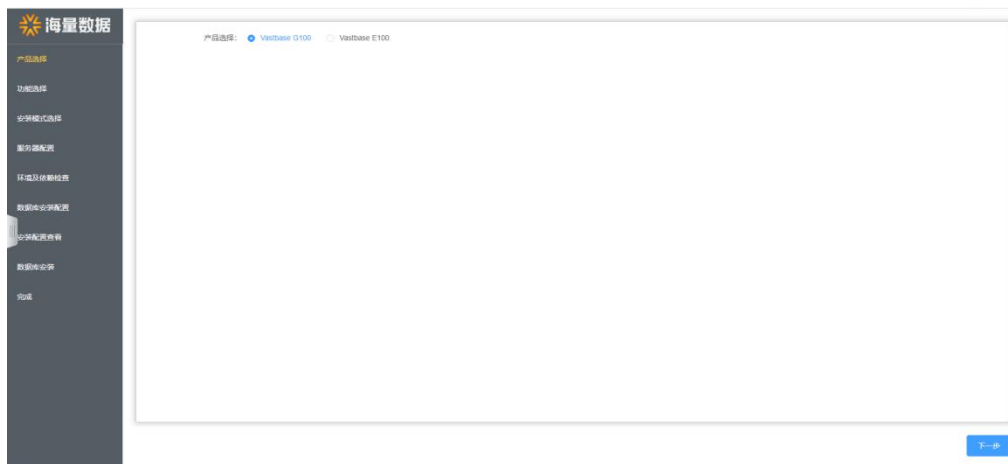
实例化数据库过程中会默认创建数据库超级用户 vastbase，执行命令登录数据库。

```
[vastbase@localhost vastbase]$ vsql -r -d vastbase
gsqldb ((Vastbase 1.0.0 build ) compiled at 2020-08-01 18:36:57 commit 0 last mr )
Non-SSL connection (SSL connection is recommended when requiring high-security)
Type "help" for help.

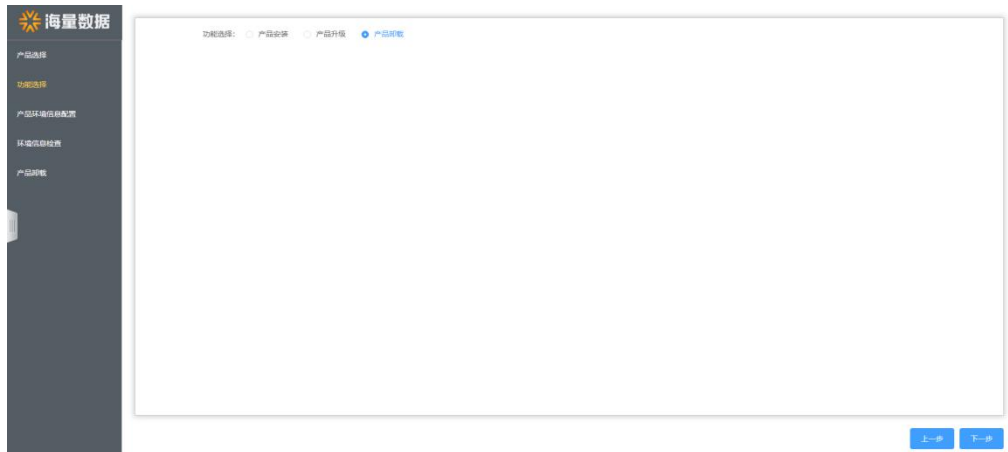
vastbase=# █
```

4.4.卸载数据库

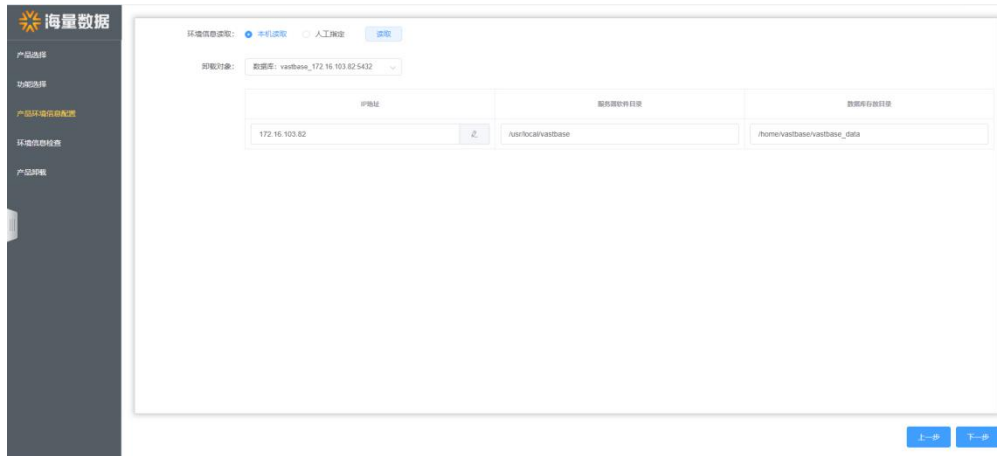
步骤 1 选择产品：选择 G100，点击下一步



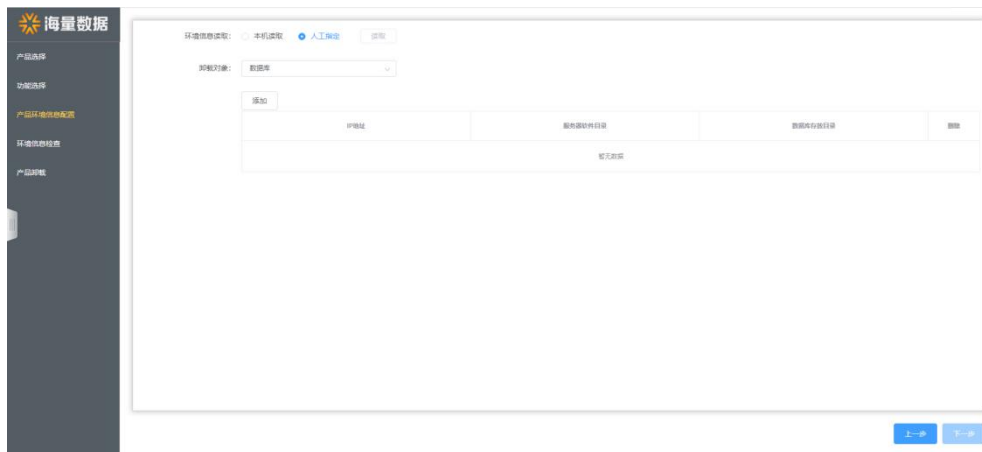
步骤 2 选择功能：产品卸载，点击下一步



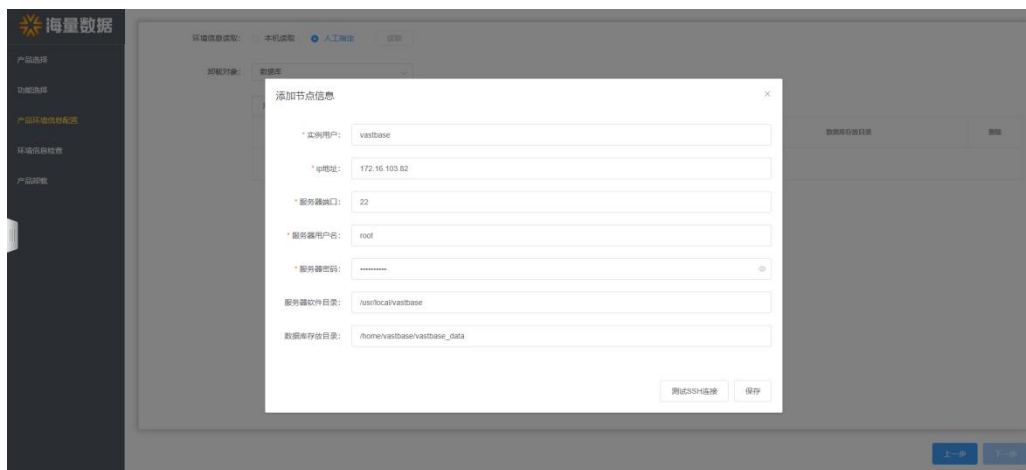
步骤 3 环境信息读取分为：本机读取和人工指定



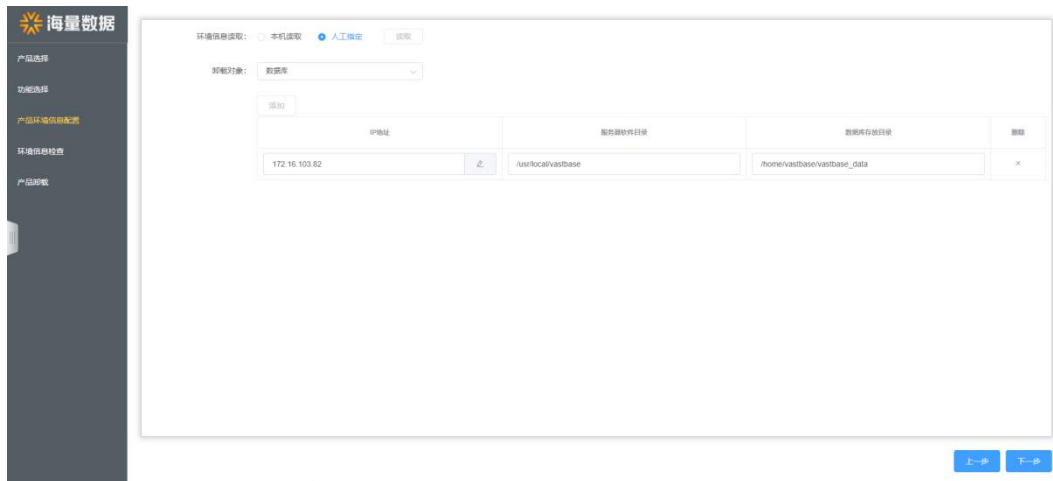
- 1) 选择本机读取后，点击读取，卸载对象可展示当前安装程序所安装的数据库或集群信息列表
- 2) 点击信息列表里“IP 地址”、“服务器软件目录”或“数据库存放目录”，支持对本机读取的 IP 地址、服务器端口、软件目录和数据库目录等信息进行修改，不允许为空



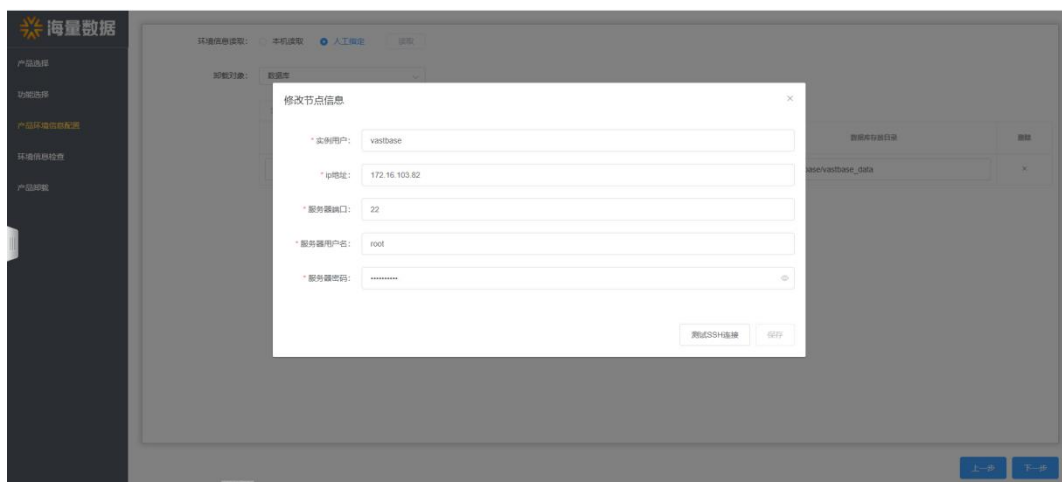
- 1) 选中人工指定，此时读取按钮不可用。
- 2) 卸载对象选择数据库
- 3) 点击添加，输入必填信息，实例用户即安装数据库时设置的实例用户。填写后，需要点击测试 ssh 连接，测试通过后，方可保存。如下图：



4) 保存信息后，如下图：



点击信息列表里“IP 地址”信息处的笔图标，支持对本机读取的 IP 地址、服务器端口等信息进行修改，不允许为空，如下图：



点击信息列表的“服务器软件目录”或“数据库存放目录”，支持对读取的结果进行修改，不允许为空。

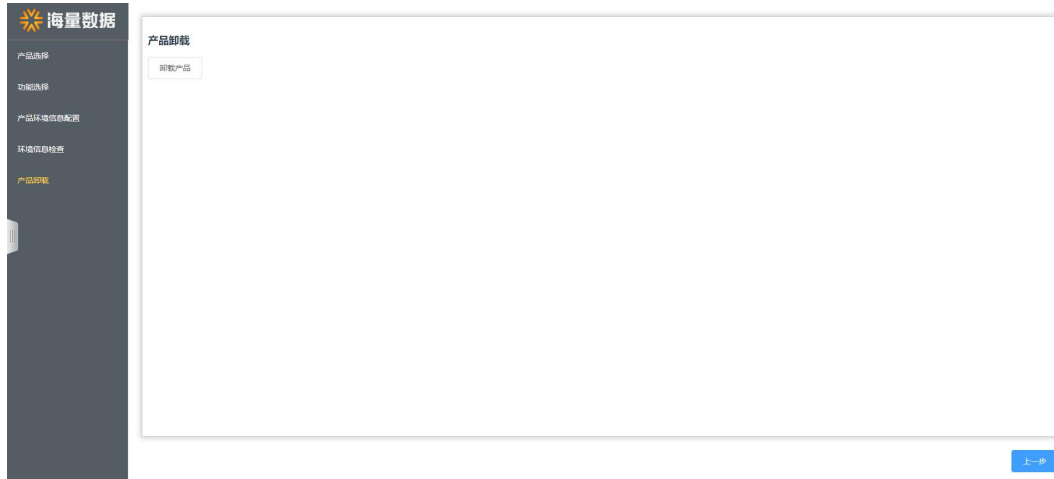
5) 点击下一步。

步骤 4 环境信息检查



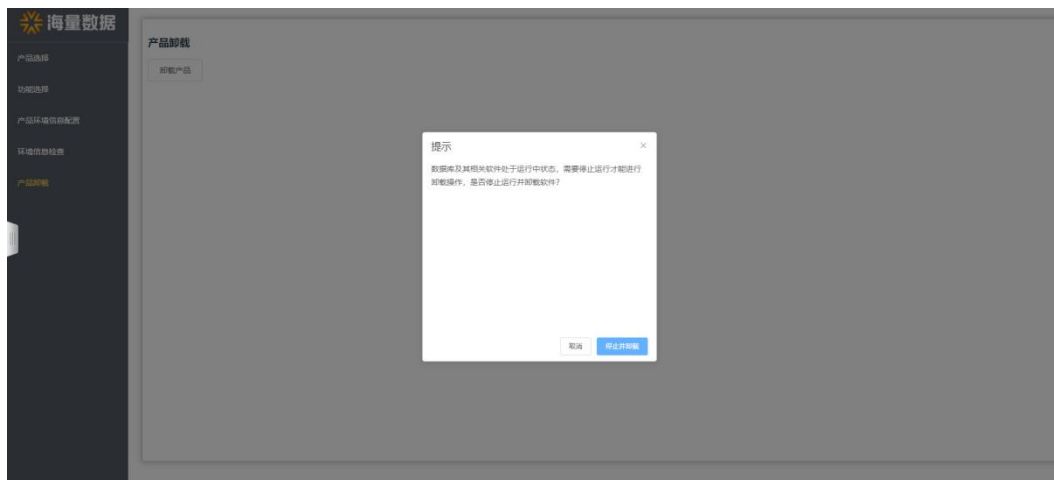
检查通过后，才可点击下一步。
点击下一步。

步骤 5 产品卸载

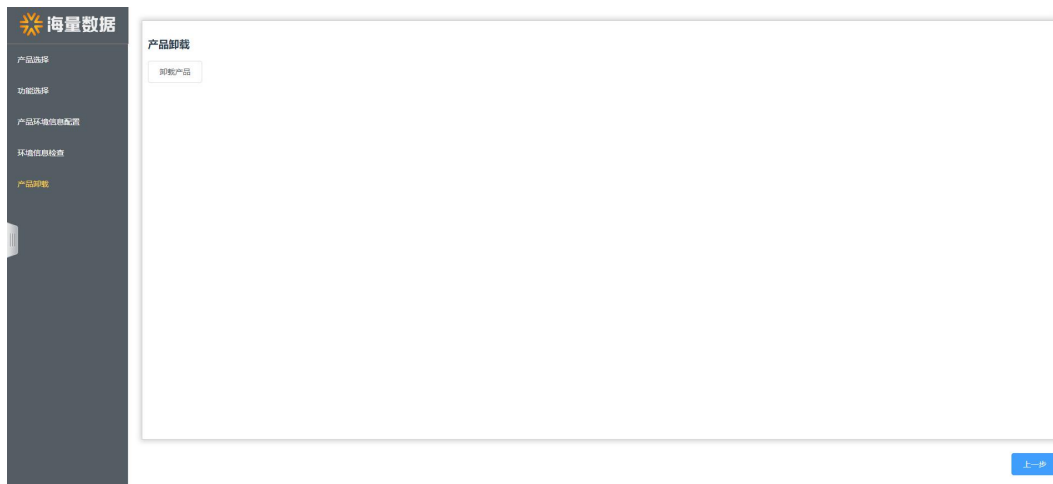


点击卸载：

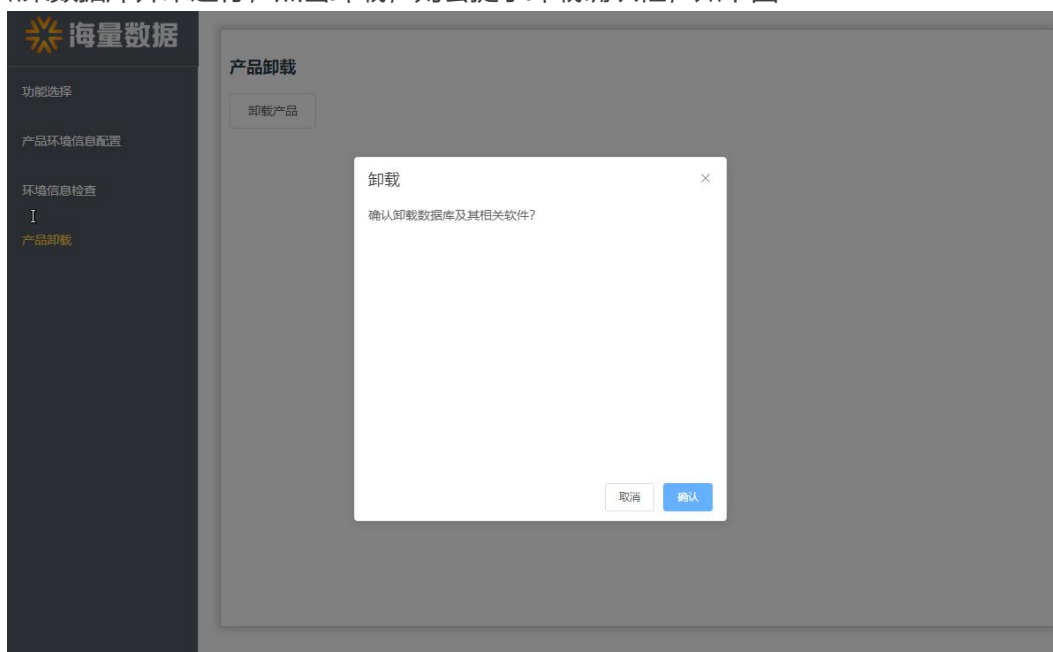
1) 如果数据库处于运行状态，点击卸载，会弹窗提示是否停止数据库并卸载的确认框，如下图：



点击取消，则不会进行卸载。此时点击停止并卸载，则会进入卸载页面，如下图：



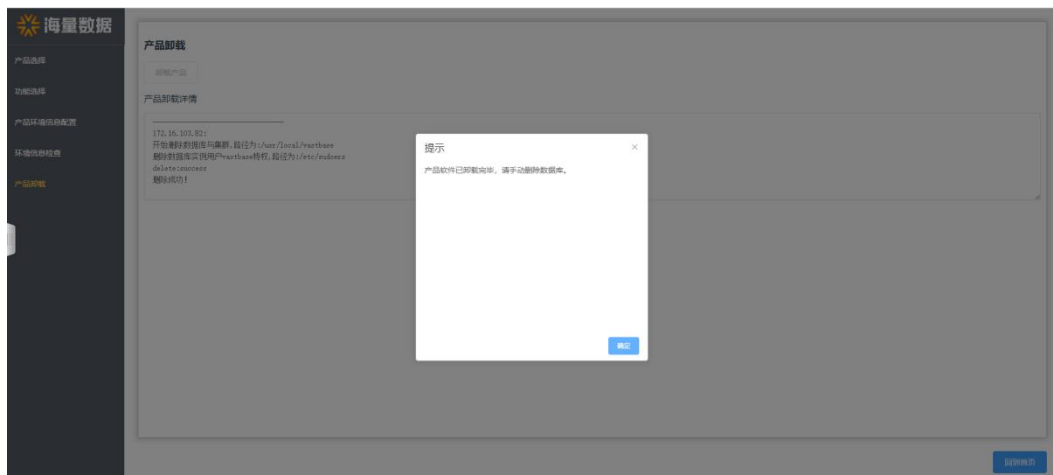
2) 如果数据库并未运行，点击卸载，则会提示卸载确认框，如下图：



点击取消，则不会进行卸载。此时点击确认，则会进入卸载页面

步骤 6 卸载完成

卸载完成后，会提示需要手动删除数据库的数据目录，如下图：



5. 字符安装

5.1. 安装过程

5.1.1. 注意事项

- ❖ 二进制安装程序与安装包需放于同一路径。
- ❖ 安装程序不能由操作系统超级用户执行。

5.1.2. 校验安装包完整性

可使用安装包相应的 sha256 文件对安装包进行 sha256 校验码校验，安装包文件和安装包名.sha256 需在同一文件夹里，执行以下命令：

```
sha256sum -c 安装包名.sha256
```

若校验成功，输出结果如下：

```
[vastbase225 lst@localhost ~]$ sha256sum -c Vastbase-G100-installer-2.2_Build_5.63-openeuler_20.03-aarch64_kunpeng-20220111.tar.gz.sha256
Vastbase-G100-installer-2.2_Build_5.63-openeuler_20.03-aarch64_kunpeng-20220111.tar.gz: OK
```

若校验失败，输出结果如下：

```
[vastbase225 lst@localhost ~]$ sha256sum -c Vastbase-G100-installer-2.2_Build_5.63-openeuler_20.03-aarch64_kunpeng-20220111.tar.gz.sha256
Vastbase-G100-installer-2.2_Build_5.63-openeuler_20.03-aarch64_kunpeng-20220111.tar.gz: FAILED
```

5.1.3. 安装步骤

- 不实例化数据库

步骤 1 创建用户和目录

安装数据库前需创建用户。例如：

```
useradd -m vastbase
```

步骤 2 切换到 vastbase 用户，使用 tar 命令解压安装包，此处以 kylin sp1 系统为例，不同操作系统包名有所区别。

```
tar xf vastbase-installer-kylin_v10sp1.aarch64-V2.2.4-20210623.tar.gz
```

步骤 3 安装环境检查

在 vastbase 用户下，进入 vastbase-installer 目录，执行脚本安装数据库。第一步是检查安装包是否完整（防篡改）。检查无误则输入<回车>继续。

安装包完整性校验成功如下图所示：

```
[vastbase@server82 vastbase-installer]$ ./vastbase_installer
=====
检查安装包是否完整
-----
ok
=====
欢迎安装vastbase
-----
输入<ENTER>继续：
=====
```

安装包完整性校验不成功如下图所示：

```
=====
检查安装包是否完整
-----
安装包不完整，可能已经被修改，请与管理员联系
=====
```

步骤 4 系统配置信息

打印系统配置信息，输入<回车>继续。

```
=====
系统配置信息
-----
      操作系统 : Kylin Linux Advanced Server V10 (Terce1)
      内存大小 : 783943 MB
      当前用户名 : vastbase
输入<ENTER>继续：
=====
```

步骤 5 依赖检查

检查服务器是否已经安装需要的依赖包，

```
=====
依赖检查
-----
      zlib : 1.2.11
      libuuid : 2.35.2
      readline : 7.0
      python2 : 2.7.16
      krb5-libs : 1.17
      libicu : 62.1
      cracklib : 2.9.7
      libxslt : 1.1.32
      tcl : 8.6.8
      perl : 5.28.0
      openldap : 2.4.50
      pam : 1.4.0
      systemd-libs : 243
      bzip2 : 1.0.8
      gettext : 0.20.1
      openssl : 1.1.1f
      libxml2 : 2.9.8
输入<ENTER>继续：
=====
```

若都已经安装，输入<回车>继续。

步骤 6 IPC 参数检查

若检查通过，自动跳转下一步，否则根据提示进行设置即可。

```
=====
IPC参数检查
-----
IPC参数检查完成
=====
```

步骤 7 安装数据库选项

是否需要实例化数据库(Y/N)可选择安装后是否自动实例化数据库；典型安装为安装过程的参数按默认值设置，如需自定义安装输入 2 并输入<回车>继续。

```
=====
安装数据库
-----
是否需要实例化数据库(Y/N): N
选择安装类型

典型安装      : 使用默认参数配置初始化数据库
自定义安装    : 手动配置安装参数和功能

-> 1- 典型安装
    2- 自定义安装

选择安装类型，或者输入<ENTER>选择默认值(1):
2
=====
```

步骤 8 设置数据库安装路径

输入前面创建的数据库安装路径，或者输入<回车>使用默认路径 (/home/vastbase/local/vastbase)。

```
=====
Vastbase软件安装目录
-----
Vastbase软件安装目录
 默认位置: /home/vastbase/local/vastbase

输入绝对路径(按下 ctrl+退格 进行回退)，或者输入<ENTER>使用默认路径:
=====
```

步骤 9 安装选项查看

查看设置的安装选项，输入<回车>继续。

```
=====
安装概要
-----
Vastbase软件安装目录:
/home/vastbase/local/vastbase

输入<ENTER>继续:
=====
```

步骤 10 安装

安装过程包括安装数据库、实例化数据库、配置数据库参数和配置用户环境变量。提示安装完成后输入<回车>退出。

```
正在安装,请稍后...
正在写入环境变量
=====
安装完成
-----

初始化客户端运行环境:
source ~/.bashrc

已安装完成,输入<ENTER>退出:
```

●实例化数据库

步骤 1 创建用户和目录

安装数据库前需创建用户,安装过程中需指定安装路径和实例化路径,用 root 用户提前创建并赋予 vastbase 用户操作的权限。例如:

```
useradd -m vastbase

mkdir -p /usr/local/vastbase
mkdir -p /data/vastbase

chown -R vastbase:vastbase /usr/local/vastbase /data/vastbase
chmod 0700 /usr/local/vastbase /data/vastbase
```

步骤 2 切换到 vastbase 用户,使用 tar 命令解压安装包,此处以 kylin sp1 系统为例,不同操作系统包名有所区别。

```
tar xf vastbase-installer-kylin_v10sp1.aarch64-V2.2.4-20210623.tar.gz
```

步骤 3 安装环境检查

在 vastbase 用户下,进入 vastbase-installer 目录,执行脚本安装数据库。第一步是检查安装包是否完整(防篡改)。检查无误则输入<回车>继续。

安装包完整性校验成功如下图所示:

```
[vastbase@server82 vastbase-installer]$ ./vastbase_installer
=====
检查安装包是否完整
-----

ok

=====
欢迎安装vastbase
-----

输入<ENTER>继续:
=====
```

安装包完整性校验不成功如下图所示:

```
=====
检查安装包是否完整
-----

安装包不完整,可能已经被修改,请与管理员联系
```

步骤 4 系统配置信息

打印系统配置信息,输入<回车>继续。

```
=====
系统配置信息
-----

      操作系统 : Kylin Linux Advanced Server V10 (Tercel)
      内存大小 : 783943 MB
      当前用户名 : vastbase

输入<ENTER>继续:
=====
```

步骤 5 依赖检查

检查服务器是否已经安装需要的依赖包，

```
=====
依赖检查
-----

      zlib : 1.2.11
      libuuid : 2.35.2
      readline : 7.0
      python2 : 2.7.16
      krb5-libs : 1.17
      libicu : 62.1
      cracklib : 2.9.7
      libxslt : 1.1.32
      tcl : 8.6.8
      perl : 5.28.0
      openldap : 2.4.50
      pam : 1.4.0
      systemd-libs : 243
      bzip2 : 1.0.8
      gettext : 0.20.1
      openssl : 1.1.1f
      libxml2 : 2.9.8

输入<ENTER>继续:
=====
```

若都已经安装，输入<回车>继续。

步骤 6 IPC 参数检查

```
=====
IPC参数检查
-----

IPC参数检查完成
=====
```

若检查通过，自动跳转下一步，否则根据提示进行设置即可。

步骤 7 安装数据库选项

是否需要实例化数据库(Y/N)可选择安装后是否自动实例化数据库,输入 Y 并输入<回车>继续
典型安装为安装过程的参数按默认值设置，如需自定义安装输入 2 并输入<回车>继续。

```
=====
安装数据库
-----

是否需要实例化数据库(Y/N): Y
选择安装类型

典型安装      : 使用默认参数配置初始化数据库
自定义安装    : 手动配置安装参数和功能

-> 1- 典型安装
    2- 自定义安装

选择安装类型，或者输入<ENTER>选择默认值(1):
2
=====
```

步骤 8 设置数据库超级用户密码

数据库操作用户密码由大小写字母加数字组成。


```
=====
数据库超级用户密码
-----

输入数据库超级用户 (vastbase) 密码: *****

请再次输入密码: *****

=====
```

步骤 9 设置数据库加密密钥设置，输入 1 并输入<回车>或直接输入<回车>使用数据库超级用户密码进行加密，或者输入 2 手动输入加密密钥

```
=====
数据库加密密钥 (PGENCRYPTIONKEY)
-----

数据库加密密钥设置:

-> 1- 使用数据库超级用户密码 (默认)
    2- 手动输入加密密钥
请选择数据库加密密钥设置方式, 或者输入<ENTER>选择默认值(1):

1

=====
```

步骤 10 设置数据库安装路径

输入前面创建的数据库安装路径，或者输入<回车>使用默认路径 (/home/vastbase/local/vastbase) 。

```
=====
Vastbase软件安装目录
-----

Vastbase软件安装目录
默认位置: /home/vastbase/local/vastbase

输入绝对路径, 或者输入<ENTER>使用默认路径:
/usr/local/vastbase

=====
```

步骤 11 设置数据库实例化路径

输入前面创建的数据库实例化路径，或者输入<回车>使用默认路径 (/home/vastbase/data/vastbase) 。

```
=====
数据库目录
-----

选择数据库目录
默认位置: /home/vastbase/data/vastbase

输入绝对路径, 或者输入<ENTER>使用默认路径:
/data/vastbase

=====
```

步骤 12 配置监听端口

输入监听端口，默认为 5432。

```
=====
监听端口
-----

输入监听端口, 或者输入<ENTER>选择默认值(5432):

=====
```

步骤 13 配置最大连接数

输入数据库最大连接数，默认为 500。


```
=====
最大连接数
-----

输入客户端最大连接数，或者输入<ENTER>选择默认值(500)：
=====
```

步骤 14 配置共享内存

输入共享内存大小，默认为（内存/4）MB。

```
=====
共享内存
-----

输入共享内存大小，单位MB，或者输入<ENTER>选择默认值(195985)：
5000
=====
```

步骤 15 磁盘 IO 调度算法检查

为避免数据库在运行过程中因为磁盘 IO 调度算法设置不合理导致系统性能了受影响，需在安装数据库时检查数据目录所在磁盘属性并做出合理选择。生产环境建议采用 deadline 策略。

```
=====
磁盘IO调度算法检查
-----

磁盘IO调度算法检查完成
=====
```

若检查出不是建议的策略，可按提示进行修改，再重新安装，或是直接跳过继续安装。

```
=====
磁盘IO调度算法检查
-----

WARNING: 当前IO调度算法为"cfq", 为避免影响数据库性能, 建议采用"deadline"算法
          可使用root用户通过以下命令进行临时修改或永久性修改:

-----> 1.临时修改: 执行命令echo deadline > /sys/block/sda/queue/scheduler
-----> 2.永久性修改: 执行命令grubby --update-kernel=ALL --args="elevator=deadline"并重启服务器
          提示: (1)执行完该命令可通过查看grub.cfg配置文件确认是否成功添加该参数后再重启服务器
              (2)grub.cfg配置文件因平台差异路径各不相同, 可通过全局查找来查看该文件
              (3)如果未安装grubby程序请自行安装后再进行修改

          可使用以下命令进行查询IO调度算法是否修改成功: cat /sys/block/sda/queue/scheduler
=====
```

步骤 16 安装选项查看

查看设置的安装选项，输入<回车>继续。

```
=====
安装概要
-----

Vastbase软件安装目录:
/usr/local/vastbase

数据库目录:
/data/vastbase

数据库超级用户:
vastbase

数据库初始化参数:
listen_addresses='*'
port=5432
max_connections=500
shared_buffers=5000MB
work_mem=4MB

输入<ENTER>继续:
=====
```

步骤 17 安装

安装过程包括安装数据库、实例化数据库、配置数据库参数和配置用户环境变量。安装时会随机生成数据库三个默认管理员的默认口令，请记录并及时修改。提示安装完成后输入<回车>退出。

```
正在安装，请稍后...
初始化数据库成功，数据目录： /data2/vastbase_lst/data/vastbase

数据库三个默认管理员vbaudit、vbssso、vbadmin的默认口令为：
system_admin[vbadmin] initial password: Zf9a877#
security_admin[vbssso] initial password: Y44d32e6
audit_admin[vbaudit] initial password: Vf5?0836

开始生成加密密钥文件
加密密钥文件生成成功

成功更新配置文件： /data2/vastbase_lst/data/vastbase/postgresql.conf

正在写入配置文件

正在写入cluster_config.xml文件

正在写入环境变量
=====
安装完成
-----

请先指定license路径(license路径错误会导致启动数据库失败):
请将license路径写入文件 /data2/vastbase_lst/data/vastbase/postgresql.conf，形式为 license_path='license路径'

初始化数据库运行环境：
source ~/.bashrc

启用、停止、重启数据库：
vb_ctl <start/stop/restart>

已安装完成，输入<ENTER>退出：
```

步骤 18 指定 license 路径

将 license 路径写入 postgresql.conf 配置文件，以确定 license 文件的位置，否则默认以 /etc/vastbase/.license 作为 license 文件位置

```
echo "license_path='/home/vastbase/license'" >> /data/vastbase/postgresql.conf
```

5.2.初始化数据库运行环境

```
source ~/.bashrc
```

5.3.启停数据库

```
vb_ctl <start/stop/restart>
```

5.4.登录数据库

实例化数据库过程中会默认创建数据库超级用户 vastbase，执行命令登录数据库。

5.5.执行卸载

卸载 Vastbase 的过程包含卸载 Vastbase 和对 Vastbase 服务器的环境做清理。

步骤 1 以操作系统用户 vastbase 登录数据库节点

步骤 2 进入安装程序所在目录

步骤 3 执行卸载命令

```
./vastbase_installer --uninstall
```

示例：使用安装程序中的卸载命令卸载运行中的 vastbase

```
./vastbase_installer --uninstall
```

```
=====
数据库卸载
-----
卸载操作将关闭数据库并删除数据库的服务端程序，是否确认卸载(Y/N):y
-----
检测到数据库正在运行，关闭数据库.....
-----
数据库关闭成功。
-----
数据库卸载完成。
=====
```

如果数据库正在运行，则先关闭数据库后进行卸载；否则直接进行卸载。卸载完成后，已安装的 Vastbase 程序（即\$GAUSSHOME 对应目录）和 Vastbase 相关环境变量将被删除，仅保留数据库实例。

错误排查：如果卸载失败请根据

\$GAUSSLOG/vastbase/gs_uninstall-YYYY-MM-DD_HHMMSS.log 中的日志信息排查错误。

6. 镜像安装

6.1. 镜像使用

6.1.1. 简介

提供的镜像包含了 vastbase 运行需要的基本环境，以及安装好的 vastbase。用户需要自己初始化数据库。

6.1.2. 导入镜像

以 Vastbase-G100_V2.2.5_openeuler_20.03sp2-aarch64_kunpeng.tar 为例。

首先导入镜像

```
docker load -i Vastbase-G100_V2.2.5_openeuler_20.03sp2-aarch64_kunpeng.tar
```

```
[root@localhost opt]# docker load -i Vastbase-G100_V2.2.5_openeuler_20.03sp2-aarch64_kunpeng.tar
98488fc3770a: Loading layer [=====>] 603.5MB/603.5MB
191a2bbd9701: Loading layer [=====>] 12.29kB/12.29kB
41db26318cbf: Loading layer [=====>] 314.4kB/314.4kB
Loaded image: vastbase:v1
```

6.1.3. 启动容器

```
docker run -dit --security-opt seccomp=unconfined --name vastbase vastbase:v1 /bin/bash
```

注意： --security-opt seccomp=unconfined 参数用于禁用 NUMA 否则启动数据库会出错

```
[root@localhost opt]# docker run -dit --security-opt seccomp=unconfined --name vastbase:v1 /bin/bash
cbf7fb67b1d38b84cb972be205cf7c5be208cb741a0902c110d2a9b71a28372
[root@localhost opt]# docker ps -a|grep vastbase
cbf7fb67b1d3 vastbase:v1 "/bin/bash" About a minute ago Up About a minute vastbase
```

6.1.4.进入容器

docker exec -it vastbase /bin/bash (也可以使用容器 id 号)。

```
[root@localhost opt]# docker exec -it vastbase /bin/bash

Welcome to 4.14.0-115.el7a.0.1.aarch64

System information as of time: Mon Dec 27 14:07:50 CST 2021

System load: 4.91
Processes: 6
Memory used: 47.1%
Swap used: 13.2%
Usage On: 84%
IP address: 172.17.0.11
Users online: 0

[root@cbf7fb67b1d3 /]#
```

6.2.数据库使用

6.2.1.初始化数据库

- 注意，飞腾+麒麟 v10 操作系统需在 root 下执行此步骤：yum install readline
- 切换到 vastbase 用户：su vastbase
- 创建数据库初始化目录

```
[vastbase@cbf7fb67b1d3 ~]$ pwd
/home/vastbase
[vastbase@cbf7fb67b1d3 ~]$ mkdir data
[vastbase@cbf7fb67b1d3 ~]$ ls
data local
[vastbase@cbf7fb67b1d3 ~]$
```

- 初始化数据库

```
vb_initdb -D data --nodename vastbase -w vbase@123
```

```
WARNING: enabling "trust" authentication for local connections
You can change this by editing pg_hba.conf or using the option -A, or
--auth-local and --auth-host, the next time you run vb_initdb.
```

```
Success. You can now start the database server of single node using:
```

```
vastbase -D data
or
vb_ctl start -D data -Z single_node -l logfile
```

●检查 license 文件

在 root 用户下查看 .license 文件所在位置

```
find / -name .license
```

```
[root@cbf7fb67b1d3 /]# find / -name .license
/home/vastbase/.license
```

修改数据库配置文件 postgresql.conf 中指定的 license_path 为上述路径

```
vi data/postgresql.conf
```

```
license_path='/home/vastbase/.license'
```

●启动数据库

```
vb_ctl start -D data -l logfile
```

其他参数可通过 vb_ctl --help 查看。

```
[vastbase@cbf7fb67b1d3 ~]$ vb_ctl start -D data -l logfile
[2021-12-27 14:23:12.405][413][][vb_ctl]: vb_ctl started,datadir is /home/vastbase/data
[2021-12-27 14:23:12.465][413][][vb_ctl]: waiting for server to start...
.
[2021-12-27 14:23:13.474][413][][vb_ctl]: done
[2021-12-27 14:23:13.474][413][][vb_ctl]: server started (/home/vastbase/data)
```

●连接数据库

```
vsq| vastbase
```



```
[vastbase@cbf7fb67b1d3 ~]$ vsql vastbase
NOTICE: Last successful login info
login time:2021-12-27 14:54:14
application name:gs_clean
Ip address:127.0.0.1
method:Trust

Failed 0 times since last successful login

vsql ((Vastbase G100 V2.2 (Build 5.39.4542)) compiled at 2021-12-20 12:14:59 commit 0 last mr
Non-SSL connection (SSL connection is recommended when requiring high-security)
Type "help" for help.

vastbase=# █
```

6.2.2.配置远程连接

- 在 pg_hba.conf 中添加要连接数据库的客户端 ip

```
# IPv4 local connections:
host all all 127.0.0.1/32 trust
host all all 172.16.12.187/32 md5
```

- 在 postgresql.conf 中修改监听地址为* (表示监听所有请求)。

```
listen_addresses = '*' # what IP address(es) to listen on;
# comma-separated list of addresses;
```

6.2.3.注意事项

可直接通过 `vb_ctl start /stop` 启动或停止数据库，但是需要指定 `PGDATA` 变量。我们提供了 `/home/vastbase/.Vastbase` 文件用于配置 `vastbase` 相关的环境变量，并在 `~/.bashrc` 文件中 `source`，建议将相关环境变量写入 `.Vastbase` 文件。

7.升级说明

Vastbase G100 数据库的升级通过升级工具 `gs_upgradectl` 完成，本文档主要介绍 Vastbase G100 数据库的升级过程。

由于本次升级工具的适配修改了安装程序和数据库打包脚本，因此被更新的数据库可分为旧版本和新版本。旧版本为 V2.2.5.55 及之前的版本，新版本为 V2.2.5.56 及之后的版本。V2.2.3 以及 V2.2.4 版本的升级仍采用重新初始化方法。旧版本升级和新版本升级的差异主要在于升级数据的准备，旧版本需要手动处理，新版本安装程序已经处理，无需手动处理。

7.1.旧版本升级

7.1.1.升级准备

以下步骤用数据库安装用户执行。

- 设置免密登录

(1) 执行 ssh-keygen

(2) 执行 ssh-copy-id -i ~/.ssh/id_rsa.pub 安装数据库用户@配置中的主机地址(如 ssh-copy-id -i ~/.ssh/id_rsa.pub vastbase@master)

- 修改 GUC 参数配置文件

修改 bin 目录下 cluster_guc.conf 配置文件，在配置项[coordinator/datanode]末尾添加以下内容：

```
vastbase_login_info|bool|0,0|NULL|NULL|
recovery_parallelism|int|1,2147483647|NULL|NULL|
audit_operation_result|int|0,2|NULL|NULL|
license_path|string|0,0|NULL|NULL|
```

```
disable_memory_protect|bool|0,0|NULL|NULL|
segment_buffers|int|16,1073741823|kB|NULL|
default_index_kind|int|0,2|NULL|NULL|
undo_zone_count|int|0,1048576|NULL|NULL|
unique_sql_clean_ratio|real|0,0.2|NULL|NULL|
enable_auto_clean_unique_sql|bool|0,0|NULL|NULL|
vastbase_login_info|bool|0,0|NULL|NULL|
recovery_parallelism|int|1,2147483647|NULL|NULL|
audit_operation_result|int|0,2|NULL|NULL|
license_path|string|0,0|NULL|NULL|
[cmserver]
log_dir|string|0,0|NULL|NULL|
log_file_size|int|0,2047|MB|NULL|
```

```
vi $GAUSSHOME/bin/cluster_guc.conf
wq!
```

注：若 license_path 使用默认路径，最后一行 license_path 相关内容可不添加。

- 将 om 脚本拷贝到 bin 目录下

将\$GAUSSHOME/om/script 文件夹拷贝到\$GAUSSHOME/bin。

```
cp -rf $GAUSSHOME/om/script $GAUSSHOME/bin
```

注：如果更早的版本没有 om 工具，可将新包中的 om 目录下的 script 文件夹拷贝到 \$GAUSSHOME/bin。

- 将数据库安装目录更改为软连接

(1) 执行 vastbase --version，查看数据库提交 ID，提交 ID 为如下字符串：

```
the configuration file.
[vbininstall@CentOS76-08 local]$ vastbase --version
vastbase (Vastbase G100 V2.2 (Build 5.39.4542)) compiled at 2021-12-20 15:35:54 commit 0 last mr
[vbininstall@CentOS76-08 local]$
```

(2) 进入到\$GAUSSHOME 的上一层目录，修改安装目录文件夹名为文件夹名+”_提交 ID”，创建与原文件夹同名的软连接，指向该目录。

```
mv vastbase vastbase_5.39_4542
ln -s vastbase_5.39_4542 vastbase
```

如下图所示：

```
drwxrwxr-x  5 vbininstall vbininstall 48 Dec 30 18:29 bin
lrwxrwxrwx  1 vbininstall vbininstall 41 Dec 30 19:22 vastbase -> /home/vbininstall/local/vastbase_5.39.4542/
drwx----- 10 vbininstall vbininstall 128 Dec 30 18:31 vastbase_5.39.4542
```

●修改 version.cfg 和 upgrade_version

到安装目录下，对比 vastbase -V 结果，按图片形式修改 version.cfg 和 upgrade_version。（如果内核版本号是 92.298，需修改为 92.299）

```
[vbininstall@CentOS76-12 vastbase]$
[vbininstall@CentOS76-12 vastbase]$ cat ./bin/upgrade_version
2.2.5.80
92.300
4878
[vbininstall@CentOS76-12 vastbase]$ vastbase -V
vastbase (Vastbase G100 V2.2 (Build 5.80.4878)) compiled at 2022-01-17 18:01:44 commit 0 last mr debug
[vbininstall@CentOS76-12 vastbase]$
```

```
[vbininstall@CentOS76-12 vastbase]$ cat version.cfg
Vastbase-G100-2.2.5.80
92.300
4878
[vbininstall@CentOS76-12 vastbase]$ vastbase -V
vastbase (Vastbase G100 V2.2 (Build 5.80.4878)) compiled at 2022-01-17 18:01:44 commit 0 last mr debug
[vbininstall@CentOS76-12 vastbase]$
```

```
vastbase -V
vi ./bin/upgrade_version
vi version.cfg
```

●修改参数

添加参数 enable_stream_replication=on 到 postgresql.conf 配置文件中。

修改参数 vastbase_login_info= off 到 postgresql.conf 配置文件中。

7.1.2.上传安装包和配置文件

该章节步骤用 root 用户执行。

●创建新包目录

以 root 用户登录，创建新包目录，并修改新包目录权限为 755，上传安装包到该目录下：


```
mkdir -p /opt/software/vastbase
chmod 755 /opt
chmod 755 /opt/software
chmod 755 /opt/software/vastbase
```

●解压安装包

- (1) 在新包目录下解压安装包，得到安装包和安装程序。

```
cd /opt/software/vastbase
tar xf Vastbase...tar.gz
```

```
total 123004
-rw-r--r-- 1 root root    13310 Dec 20 15:51 vastbase-centos_7-x86_64-V2.2_Build5.ReleaseNote.md
-rw-r--r-- 1 root root 122861304 Dec 20 15:51 Vastbase-G100-2.2_Build_5.50-centos_7-x86_64.tar.gz
-rwxr-xr-x 1 root root   3075120 Dec 20 15:51 vastbase_installer
```

- (2) 创建 vastbase 目录，继续解压安装包到该目录中，得到 OM 工具包。

```
mkdir /opt/software/vastbase/vastbase
cd vastbase-installer
tar xf Vastbase...tar.gz -C /opt/software/vastbase/vastbase
```

```
total 243490
-r----- 1 root root      65 Dec 30 18:06 upgrade_sql.sha256
-r----- 1 root root  134302 Dec 30 18:06 upgrade_sql.tar.gz
-rw-r--r-- 1 root root   13310 Dec 20 15:51 vastbase-centos_7-x86_64-V2.2_Build5.ReleaseNote.md
-r----- 1 root root      65 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.sha256
-r----- 1 root root 12457917 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.tar.gz
-r----- 1 root root      65 Dec 30 18:06 vastbase-G100-2.2_Build_5.50-CentOS-64bit.sha256
-r----- 1 root root 110774002 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit.tar.bz2
-rw-r--r-- 1 root root 122861304 Dec 20 15:51 Vastbase-G100-2.2_Build_5.50-centos_7-x86_64.tar.gz
-rwxr-xr-x 1 root root   3075120 Dec 20 15:51 vastbase_installer
[root@CentOS76-08 vastbase-installer]#
```

- (3) 解压 OM 工具包，会自动生成 script 目录已经在 script 目录下生成 gs_upgradectl 等脚本工具。

```
cd /opt/software/vastbase/vastbase
tar xf Vastbase-G100...om.tar.gz
```

```

drwxrwxr-x 14 root root      302 Dec 29 18:47 lib
-rw-rw-r-- 1 root root        31 Dec 29 18:47 om_version.cfg
drwxrwxr-x 6 root root       292 Dec 29 18:47 script
-r----- 1 root root         65 Dec 30 18:06 upgrade_sql.sha256
-r----- 1 root root      134302 Dec 30 18:06 upgrade_sql.tar.gz
-rw-r--r-- 1 root root     13310 Dec 20 15:51 vastbase-centos_7-x86_64-V2.2_Build5.ReleaseNote.md
-r----- 1 root root         65 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.sha256
-r----- 1 root root    12457917 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.tar.gz
-r----- 1 root root         65 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit.sha256
-r----- 1 root root    110774002 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit.tar.bz2
[root@CentOS76-08 vastbase]# cd script/
[root@CentOS76-08 script]# ll
total 308
drwxrwxr-x 2 root root      53 Dec 29 18:47 config
-rwxrwxr-x 1 root root     9059 Dec 29 18:47 gs_backup
-rwxrwxr-x 1 root root    64796 Dec 29 18:47 gs_check
-rwxrwxr-x 1 root root    60618 Dec 29 18:47 gs_checkos
-rwxrwxr-x 1 root root    10798 Dec 29 18:47 gs_checkperf
-rwxrwxr-x 1 root root    16260 Dec 29 18:47 gs_collector
-rwxrwxr-x 1 root root    31732 Dec 29 18:47 gs_om
-rwxrwxr-x 1 root root    19123 Dec 29 18:47 gs_preinstall
drwxrwxr-x 11 root root      150 Dec 29 18:47 gspylib
-rwxrwxr-x 1 root root     8267 Dec 29 18:47 gs_ssh
-rwxrwxr-x 1 root root    46153 Dec 29 18:47 gs_sshexkey
-rwxrwxr-x 1 root root    11128 Dec 29 18:47 gs_upgradectl
drwxrwxr-x 13 root root      200 Dec 29 18:47 impl
-rwxrwxr-x 1 root root         0 Dec 29 18:47 _init_.py
-rwxrwxr-x 1 root root     6189 Dec 29 18:47 killall
drwxrwxr-x 2 root root     4096 Dec 29 18:47 local
-rwxrwxr-x 1 root root     6580 Dec 29 18:47 transfer.py
[root@CentOS76-08 script]#

```

●创建配置文件

```

cd /opt/software/vastbase
vi cluster_config.xml

```

配置文件模板如下所示：（nodeNames 填写 hostname，gaussdbAppPath 填写安装目录，gaussdbLogPath 填写数据库日志目录，name 填写 hostname，dataNode1 填写数据目录，dataPortBase 填写数据库相应的端口，tmpMppdbPath、gaussdbToolPath 目录不要和其他目录重叠）

（注意，xml 配置文件需注意缩进正确，否则影响后续步骤）

```
<?xml version="1.0" encoding="UTF-8"?>
<ROOT>
  <!-- Vastbase整体信息 -->
  <CLUSTER>
    <!-- 数据库集群名称 -->
    <PARAM name="clusterName" value="dbCluster" />
    <!-- 数据库节点名称 (hostname) -->
    <PARAM name="nodeNames" value="CentOS76-08" />
    <!-- 数据库安装目录 -->
    <PARAM name="gaussdbAppPath" value="/home/vbinstall/local/vastbase" />
    <!-- om工具日志目录 -->
    <PARAM name="gaussdbLogPath" value="/home/vbinstall/data/vastbase/pg_log" />
    <!-- 临时文件目录 -->
    <PARAM name="tmpMppdbPath" value="/home/vbinstall/local/omTmp/tmp" />
    <!-- 数据库工具目录 -->
    <PARAM name="gaussdbToolPath" value="/home/vbinstall/local/omTmp/omToolPath" />
    <!-- 数据库core文件目录 -->
    <PARAM name="corePath" value="/home/vbinstall/local/omTmp/corefile" />
    <!-- 节点类型 -->
    <PARAM name="clusterType" value="single-inst"/>
  </CLUSTER>
  <!-- 每台服务器上的节点部署信息 -->
  <DEVICELIST>
    <!-- 节点上的节点部署信息 -->
    <DEVICE sn="1000001">
      <!-- 节点1的主机名称 -->
      <PARAM name="name" value="CentOS76-08" />
      <!-- 节点1的AZ及AZ优先级 -->
      <PARAM name="azName" value="AZ1" />
      <PARAM name="azPriority" value="1" />
      <!-- 如果服务器只有一个网卡可用，将backIp1和sshIp1配置成同一个IP -->
      <PARAM name="backIp1" value="127.0.0.1" />
      <PARAM name="sshIp1" value="127.0.0.1" />

      <!-- 节点数量 -->
      <PARAM name="dataNum" value="1" />
      <!-- 数据库端口号 -->
      <PARAM name="dataPortBase" value="35432" />
      <!-- 数据库实例目录 (数据目录) -->
      <PARAM name="dataNode1" value="/home/vbinstall/data/vastbase" />
    </DEVICE>
  </DEVICELIST>
</ROOT>
```

创建文件后将其上传到/opt/software/vastbase 目录，并将其所属用户修改为数据库安装用户。

```
chown vbinstall:vbinstall cluster_config.xml
```

注意：OM 工具是根据这份配置文件以及用这份配置文件生成的静态文件去识别我们的数据库的，因此该配置文件里面填的信息应当和我们待升级的数据库配置信息一致。

7.1.3. 执行 gs_preinstall 准备升级

由于 gs_preinstall 会修改 ~/.bashrc 环境变量配置文件，可能与旧版本的安装程序写入的 source ~/.Vastbase 这一行内容冲突（直接追加到行后，不换行），因此执行 gs_preinstall 之前最好在 ~/.bashrc 文件内容的最后加一个换行符。执行命令如下：

```
./gs_preinstall -U vbininstall -G vbininstall -X /opt/software/vastbase/cluster_config.xml
```

- -U 为数据库安装用户
- -G 为数据库安装用户所属的组名
- -X 为配置文件完整路径

执行结果如下图所示：

```
[root@CentOS76-08 script]# ./gs_preinstall -U vbininstall -G vbininstall -X /opt/software/vastbase/vastbase/cluster_config.xml
Parsing the configuration file.
Successfully parsed the configuration file.
Installing the tools on the local node.
Successfully installed the tools on the local node.
Setting pssh path
Successfully set core path.
Are you sure you want to create the user[vbininstall] and create trust for it (yes/no)? yes
Preparing SSH service.
Successfully prepared SSH service.
Checking OS software.
Successfully check os software.
Checking OS version.
Successfully checked OS version.
Creating cluster's path.
Successfully created cluster's path.
Setting SCTP service.
Successfully set SCTP service.
Set and check OS parameter.
Setting OS parameters.
Successfully set OS parameters.
Warning: Installation environment contains some warning messages.
Please get more details by "/opt/software/vastbase/vastbase/vastbase-installer/vastbase/script/gc_checkos -i A -h CentOS76-08 --detail".
Set and check OS parameter completed.
Preparing CRON service.
Successfully prepared CRON service.
Setting user environmental variables.
Successfully set user environmental variables.
Setting the dynamic link library.
Successfully set the dynamic link library.
Setting Core file
Successfully set core path.
Setting pssh path
Successfully set pssh path.
Set ARM Optimization.
No need to set ARM Optimization.
Fixing server package owner.
Setting finish flag.
Successfully set finish flag.
Preinstallation succeeded.
[root@CentOS76-08 script]#
```

7.1.4. 执行 gs_upgradectl 进行升级

切换到数据库安装用户。

- 执行 gs_om

执行 gs_om 生成升级中所用到的配置文件。

```
gs_om -t generateconf -X /opt/software/vastbase/cluster_config.xml --distribute
```

```
[vbininstall@CentOS76-08 local]$ gs_om -t generateconf -X /opt/software/vastbase/cluster_config.xml --distribute
Generating static configuration files for all nodes.
Creating temp directory to store static configuration files.
Successfully created the temp directory.
Generating static configuration files.
Successfully generated static configuration files.
Static configuration files for all nodes are saved in /home/vbininstall/local/omTmp/omToolPath/script/static_config_files.
Distributing static configuration files to all nodes.
Successfully distributed static configuration files.
```

●设置 OM_GAUSSLOG 环境变量

修改文件.Vastbase 和.bashrc，将 OM_GAUSSLOG 的值修改成 cluster_config.xml 配置文件中 gaussdbLogPath 的值。重新登录用户。

●执行升级命令

将数据库关闭，执行 gs_upgradectl 进行升级。目前只适配了就地升级方法，就地升级需要停止数据库。

```
gs_upgradectl -t auto-upgrade -X /opt/software/vastbase/cluster_config.xml
```

执行结果如下：

```
[vbininstall@CentOS76-08 ~]$ gs_upgradectl -t auto-upgrade -X /opt/software/vastbase/cluster_config.xml
Static configuration matched with old static configuration files.
Performing inplace rollback.
Rollback succeeded.
Checking upgrade environment.
Successfully checked upgrade environment.
Successfully started cluster.
Start to do health check.
Successfully checked cluster status.
Backing up current application and configurations.
Successfully backed up current application and configurations.
Backing up cluster configuration.
Successfully backup hotpatch config file.
Installing new binary.
Restoring cluster configuration.
Successfully restored cluster configuration.
Successfully started cluster.
Switch symbolic link to new binary directory.
Successfully switch symbolic link to new binary directory.
Successfully started cluster.
Start to do health check.
Successfully checked cluster status.
Upgrade main process has been finished, user can do some check now.
Once the check done, please execute following command to commit upgrade:

gs_upgradectl -t commit-upgrade -X /opt/software/vastbase/cluster_config.xml
```

报错部分为无权限访问一个操作系统配置文件，对升级暂无影响。

●升级验证

- (1) 版本查询。执行“vasebase --version” 查看是否为升级后正确版本。
- (2) 数据库状态检查。执行“gs_om -t status” 查看数据库当前状态是否正确。

●数据库回滚

回滚命令如下：

```
gs_upgradectl -t auto-rollback -X /opt/software/vastbase/cluster_config.xml
```

执行结果如下：

```
[vbininstall@CentOS76-08 ~]$ gs_upgradectl -t auto-rollback -X /opt/software/vastbase/cluster_config.xml
The requester's hostname: CentOS76-08, IP: 172.16.12.65, mac address:00:50:56:9b:e0:fd
The application is Vastbase G100
Static configuration matched with old static configuration files.
Performing inplace rollback.
Checking static configuration files.
Successfully checked static configuration files.
Successfully started cluster.
Restoring cluster configuration.
Successfully rollback hotpatch config file.
Successfully restored cluster configuration.
Switch symbolic link to old binary directory.
Successfully switch symbolic link to old binary directory.
Restoring application and configurations.
Successfully restored application and configuration.
Restoring cluster configuration.
Successfully rollback hotpatch config file.
Successfully restored cluster configuration.
Successfully started cluster.
Successfully cleaned new install path.
Rollback succeeded.
[vbininstall@CentOS76-08 ~]$
```

●提交升级

```
gs_upgradectl -t commit-upgrade -X /opt/software/vastbase/cluster_config.xml
```

执行结果如下：

```
[vbininstall@CentOS76-08 ~]$ gs_upgradectl -t commit-upgrade -X /opt/software/vastbase/cluster_config.xml
Successfully cleaned old install path.
Commit binary upgrade succeeded.
[vbininstall@CentOS76-08 ~]$
```

注：提交升级后将无法回滚。

7.2.新版本升级

7.2.1.升级准备

- 设置免密登录

(1) 执行 ssh-keygen

(2) 执行 ssh-copy-id -i ~/.ssh/id_rsa.pub 安装数据库用户@配置中的主机地址(如 ssh-copy-id -i ~/.ssh/id_rsa.pub vastbase@master)

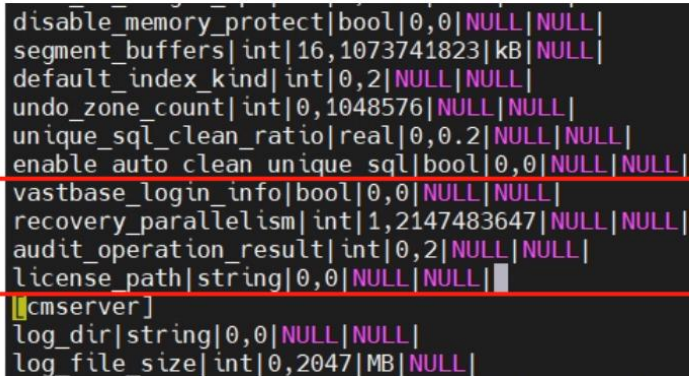
- 修改 GUC 参数配置文件

注：(1) 若 license_path 使用默认路径，可跳过这一步。

(2) 待升级版本为 2.2.5.73 及之后的，可跳过这一步。

修改 bin 目录下 cluster_guc.conf 配置文件，在配置项[coordinator/datanode]末尾添加以下内容：

```
vastbase_login_info|bool|0,0|NULL|NULL|
recovery_parallelism|int|1,2147483647|NULL|NULL|
audit_operation_result|int|0,2|NULL|NULL|
license_path|string|0,0|NULL|NULL|
```



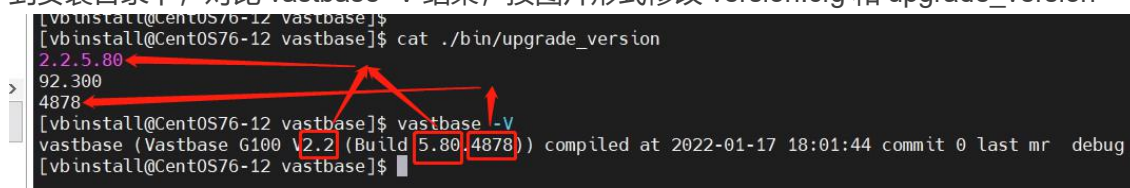
```
disable_memory_protect|bool|0,0|NULL|NULL|
segment_buffers|int|16,1073741823|kB|NULL|
default_index_kind|int|0,2|NULL|NULL|
undo_zone_count|int|0,1048576|NULL|NULL|
unique_sql_clean_ratio|real|0,0.2|NULL|NULL|
enable_auto_clean_unique_sql|bool|0,0|NULL|NULL|
vastbase_login_info|bool|0,0|NULL|NULL|
recovery_parallelism|int|1,2147483647|NULL|NULL|
audit_operation_result|int|0,2|NULL|NULL|
license_path|string|0,0|NULL|NULL|
[cmserver]
log_dir|string|0,0|NULL|NULL|
log_file_size|int|0,2047|MB|NULL|
```

```
vi $GAUSSHOME/bin/cluster_guc.conf
```

```
wq!
```

- 修改 version.cfg 和 upgrade_version

到安装目录下，对比 vastbase -V 结果，按图片形式修改 version.cfg 和 upgrade_version



```
[vbininstall@CentOS76-12 vastbase]$ cat ./bin/upgrade_version
2.2.5.80
92.300
4878
[vbininstall@CentOS76-12 vastbase]$ vastbase -V
vastbase (Vastbase G100 2.2 (Build 5.80.4878)) compiled at 2022-01-17 18:01:44 commit 0 last mr debug
[vbininstall@CentOS76-12 vastbase]$
```

```
[vbininstall@CentOS76-12 vastbase]$ cat version.cfg
Vastbase-G100-2.2.5.80
92.300
4878
[vbininstall@CentOS76-12 vastbase]$ vastbase -V
vastbase (Vastbase G100 V2.2) (Build 5.80.4878) compiled at 2022-01-17 18:01:44 commit 0 last mr debug
[vbininstall@CentOS76-12 vastbase]$
```

```
vastbase -V
vi ./bin/upgrade_version
vi version.cfg
```

- 修改参数

修改参数 `vastbase_login_info= off` 到 `postgresql.conf` 配置文件中。

7.2.2.上传安装包和配置文件

- 创建新包目录

以 root 用户登录，创建新包目录，并修改新包目录权限为 755，上传安装包到该目录下：

```
mkdir -p /opt/software/vastbase
chmod 755 /opt
chmod 755 /opt/software
chmod 755 /opt/software/vastbase
```

- 解压安装包

(1) 解压安装包，得到安装包和安装程序。

```
cd /opt/software/vastbase
tar xf Vastbase...tar.gz
```

```
total 123004
-rw-r--r-- 1 root root 13310 Dec 20 15:51 vastbase-centos_7-x86_64-V2.2_Build5.ReleaseNote.md
-rw-r--r-- 1 root root 122861304 Dec 20 15:51 Vastbase-G100-2.2_Build_5.50-centos_7-x86_64.tar.gz
-rwxr-xr-x 1 root root 3075120 Dec 20 15:51 vastbase_installer
```

(2) 创建 `vastbase` 目录，继续解压安装包到该目录中，得到 OM 工具包。

```
mkdir /opt/software/vastbase/vastbase
cd vastbase-installer
tar xf Vastbase...tar.gz -C /opt/software/vastbase/vastbase
```



```

total 243490
-r----- 1 root root      65 Dec 30 18:06 upgrade_sql.sha256
-r----- 1 root root    134302 Dec 30 18:06 upgrade_sql.tar.gz
-rw-r--r-- 1 root root    13310 Dec 20 15:51 vastbase-centos_7-x86_64-V2.2_Build5.ReleaseNote.md
-r----- 1 root root      65 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.sha256
-r----- 1 root root   12457917 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.tar.gz
-r----- 1 root root      65 Dec 30 18:06 vastbase-G100-2.2_Build_5.50-CentOS-64bit.sha256
-r----- 1 root root   110774002 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit.tar.bz2
-rw-r--r-- 1 root root  122861304 Dec 20 15:51 Vastbase-G100-2.2_Build_5.50-centos_7-x86_64.tar.gz
-rwxr-xr-x 1 root root   3075120 Dec 20 15:51 vastbase_installer
[root@CentOS76-08 vastbase-installer]#

```

(3) 解压 OM 工具包，会自动生成 script 目录已经在 script 目录下生成 gs_upgradectl 等脚本工具。

```

cd /opt/software/vastbase/vastbase
tar xf Vastbase-G100...om.tar.gz

```

```

drwxrwxr-x 14 root root      302 Dec 29 18:47 lib
-rw-rw-r-- 1 root root      31 Dec 29 18:47 om_version.cfg
drwxrwxr-x 6 root root      292 Dec 29 18:47 script
-r----- 1 root root      65 Dec 30 18:06 upgrade_sql.sha256
-r----- 1 root root    134302 Dec 30 18:06 upgrade_sql.tar.gz
-rw-r--r-- 1 root root    13310 Dec 20 15:51 vastbase-centos_7-x86_64-V2.2_Build5.ReleaseNote.md
-r----- 1 root root      65 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.sha256
-r----- 1 root root   12457917 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit-om.tar.gz
-r----- 1 root root      65 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit.sha256
-r----- 1 root root   110774002 Dec 30 18:06 Vastbase-G100-2.2_Build_5.50-CentOS-64bit.tar.bz2
[root@CentOS76-08 vastbase]# cd script/
[root@CentOS76-08 script]# ll
total 308
drwxrwxr-x 2 root root      53 Dec 29 18:47 config
-rwxrwxr-x 1 root root    9059 Dec 29 18:47 gs_backup
-rwxrwxr-x 1 root root   64796 Dec 29 18:47 gs_check
-rwxrwxr-x 1 root root   60618 Dec 29 18:47 gs_checkos
-rwxrwxr-x 1 root root   10798 Dec 29 18:47 gs_checkperf
-rwxrwxr-x 1 root root   16260 Dec 29 18:47 gs_collector
-rwxrwxr-x 1 root root   31732 Dec 29 18:47 gs_om
-rwxrwxr-x 1 root root   19123 Dec 29 18:47 gs_preinstall
drwxrwxr-x 11 root root     150 Dec 29 18:47 gspylib
-rwxrwxr-x 1 root root    8267 Dec 29 18:47 gs_ssh
-rwxrwxr-x 1 root root   46153 Dec 29 18:47 gs_sshkey
-rwxrwxr-x 1 root root   11128 Dec 29 18:47 gs_upgradectl
drwxrwxr-x 13 root root     200 Dec 29 18:47 impl
-rwxrwxr-x 1 root root      0 Dec 29 18:47 _init_.py
-rwxrwxr-x 1 root root    6189 Dec 29 18:47 killall
drwxrwxr-x 2 root root    4096 Dec 29 18:47 local
-rwxrwxr-x 1 root root    6580 Dec 29 18:47 transfer.py
[root@CentOS76-08 script]#

```

●上传配置文件

配置文件模板为位于\$GAUSSHOME/etc/om 的 cluster_config.xml 文件，将其拷贝到

/opt/software/vastbase 目录，修改后将文件所属用户修改为数据库安装用户。

```

chown vbininstall:vbininstall cluster_config.xml

```

7.2.3.执行 gs_preinstall 准备升级

执行命令如下：

```

./gs_preinstall -U vbininstall -G vbininstall -X /opt/software/vastbase/cluster_config.xml

```

■-U 为数据库安装用户

■-G 为数据库安装用户所属的组名

■-X 为配置文件完整路径

执行结果如下图所示:

```
[root@CentOS76-08 script]# ./gs_preinstall -U vbininstall -G vbininstall -X /opt/software/vastbase/cluster_config.xml
Parsing the configuration file.
Successfully parsed the configuration file.
Installing the tools on the local node.
Successfully installed the tools on the local node.
Setting pssh path
Successfully set core path.
Are you sure you want to create the user[vbininstall] and create trust for it (yes/no)? yes
Preparing SSH service.
Successfully prepared SSH service.
Checking OS software.
Successfully check os software.
Checking OS version.
Successfully checked OS version.
Creating cluster's path.
Successfully created cluster's path.
Setting SCTP service.
Successfully set SCTP service.
Set and check OS parameter.
Setting OS parameters.
Successfully set OS parameters.
Warning: Installation environment contains some warning messages.
Please get more details by "/opt/software/vastbase/vb_5.50/script/g_s_checks -i A -h CentOS76-08 --detail".
Set and check OS parameter completed.
Preparing CRON service.
Successfully prepared CRON service.
Setting user environmental variables.
Successfully set user environmental variables.
Setting the dynamic link library.
Successfully set the dynamic link library.
Setting Core file
Successfully set core path.
Setting pssh path
Successfully set pssh path.
Set ARM Optimization.
No need to set ARM Optimization.
Fixing server package owner.
Setting finish flag.
Successfully set finish flag.
Preinstallation succeeded.
[root@CentOS76-08 script]#
```

7.2.4.执行 gs_upgradectl 进行升级

切换到数据库安装用户。

●执行 gs_om

执行 gs_om 生成升级中所用到的配置文件。

```
gs_om -t generateconf -X /opt/software/vastbase/cluster_config.xml --distribute
```

执行成功结果如下:

```
[vbininstall@CentOS76-08 ~]$ gs_om -t generateconf -X /opt/software/vastbase/cluster_config.xml --distribute
Generating static configuration files for all nodes.
Creating temp directory to store static configuration files.
Successfully created the temp directory.
Generating static configuration files.
Successfully generated static configuration files.
Static configuration files for all nodes are saved in /home/vbininstall/local/omTmp/omToolPath/script/static_config_files.
Distributing static configuration files to all nodes.
Successfully distributed static configuration files.
[vbininstall@CentOS76-08 ~]$
```

●设置 OM_GAUSSLOG 环境变量

修改文件.Vastbase 和.bashrc，将 OM_GAUSSLOG 的值修改成 cluster_config.xml 配置文件中 gaussdbLogPath 的值。重新登录用户。

●执行升级命令

将数据库关闭，执行 gs_upgradectl 进行升级。目前只适配了就地升级方法，就地升级需要停止数据库。

```
gs_upgradectl -t auto-upgrade -X /opt/software/vastbase/cluster_config.xml
```

执行结果如下：

```
[vbininstall@CentOS76-08 ~]$ gs_upgradectl -t auto-upgrade -X /opt/software/vastbase/cluster_config.xml
Static configuration matched with old static configuration files.
Performing inplace rollback.
Rollback succeeded.
Checking upgrade environment.
Successfully checked upgrade environment.
Successfully started cluster.
Start to do health check.
Successfully checked cluster status.
Backing up current application and configurations.
Successfully backed up current application and configurations.
Backing up cluster configuration.
Successfully backup hotpatch config file.
Installing new binary.
Restoring cluster configuration.
Successfully restored cluster configuration.
Successfully started cluster.
Switch symbolic link to new binary directory.
Successfully switch symbolic link to new binary directory.
Successfully started cluster.
Start to do health check.
Successfully checked cluster status.
Upgrade main process has been finished, user can do some check now.
Once the check done, please execute following command to commit upgrade:

gs_upgradectl -t commit-upgrade -X /opt/software/vastbase/cluster_config.xml
```

报错部分为无权限访问一个操作系统配置文件，对升级暂无影响。

●升级验证

- (1) 版本查询。执行“vasebase --version”查看是否为升级后正确版本。
- (2) 数据库状态检查。执行“gs_om -t status”查看数据库当前状态是否正确。

(3) 检查被升级的功能能正常使用。

●数据库回滚

回滚命令如下：

```
gs_upgradectl -t auto-rollback -X /opt/software/vastbase/cluster_config.xml
```

执行结果如下：

```
[vbininstall@CentOS76-08 ~]$ gs_upgradectl -t auto-rollback -X /opt/software/vastbase/cluster_config.xml
The requester's hostname: CentOS76-08, IP: 172.16.12.65, mac address:00:50:56:9b:e0:fd
The application is Vastbase G100
Static configuration matched with old static configuration files.
Performing inplace rollback.
Checking static configuration files.
Successfully checked static configuration files.
Successfully started cluster.
Restoring cluster configuration.
Successfully rollback hotpatch config file.
Successfully restored cluster configuration.
Switch symbolic link to old binary directory.
Successfully switch symbolic link to old binary directory.
Restoring application and configurations.
Successfully restored application and configuration.
Restoring cluster configuration.
Successfully rollback hotpatch config file.
Successfully restored cluster configuration.
Successfully started cluster.
Successfully cleaned new install path.
Rollback succeeded.
[vbininstall@CentOS76-08 ~]$
```

●回滚验证

- (1) 版本查询。执行“vasebase --version” 查看是否为升级前的版本。
- (2) 数据库状态检查。执行“gs_om -t status” 查看数据库当前状态是否正确。
- (3) 检查被回滚的功能恢复到升级之前的状态。

●提交升级

```
gs_upgradectl -t commit-upgrade -X /opt/software/vastbase/cluster_config.xml
```

执行结果如下：

```
[vbininstall@CentOS76-08 ~]$ gs_upgradectl -t commit-upgrade -X /opt/software/vastbase/cluster_config.xml
Successfully cleaned old install path.
Commit binary upgrade succeeded.
[vbininstall@CentOS76-08 ~]$
```

注：提交升级后将无法回滚。



电话：010-82838118

地址：北京市海淀区学院路 30 号科大天工大厦 B 座 6 层

官网：www.vastdata.com.cn