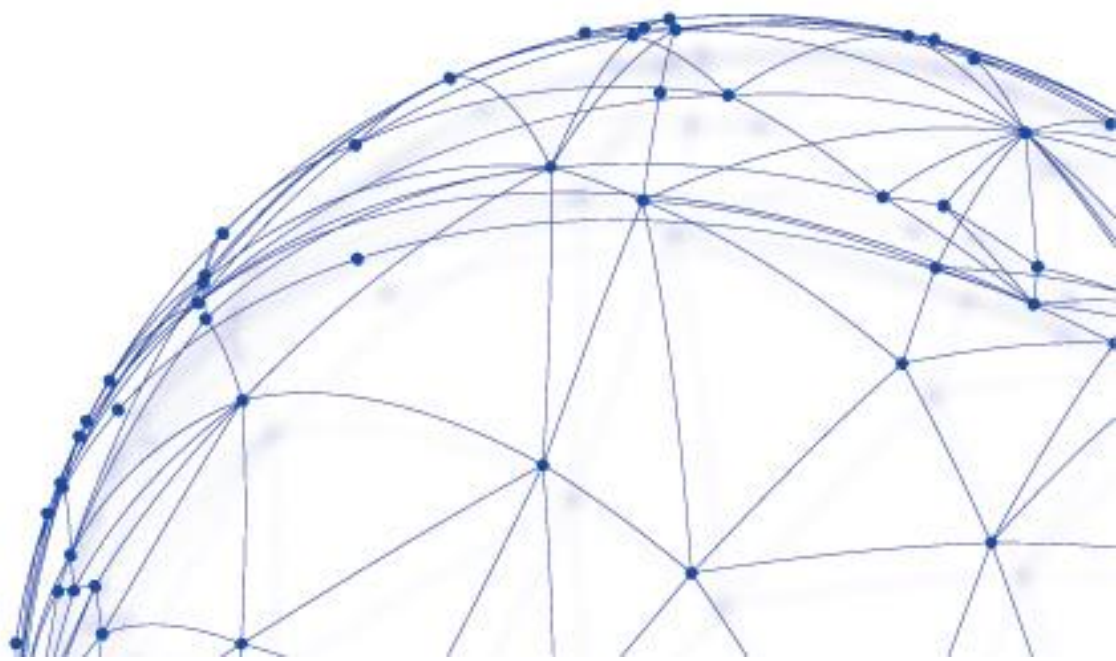


达梦技术手册

DM8_dmagent 使用手册

Service manual of DM8_dmagent



前言

概述

本文档主要介绍 DM Web 版管理工具 DEM 部署在远程主机上的代理 dmagent 的配置方法、使用方法以及注意事项等。

读者对象





本文档主要适用于 DM 数据库的：

- 开发工程师
- 测试工程师
- 技术支持工程师
- 数据库管理员

通用约定

在本文档中可能出现下列标志，它们所代表的含义如下：

表 0.1 标志含义

标志	说明
 警告：	表示可能导致系统损坏、数据丢失或不可预知的结果。
 注意：	表示可能导致性能降低、服务不可用。
 小窍门：	可以帮助您解决某个问题或节省您的时间。
 说明：	表示正文的附加信息，是对正文的强调和补充。

在本文档中可能出现下列格式，它们所代表的含义如下：

表 0.2 格式含义

格式	说明
宋体	表示正文。
Courier new	表示代码或者屏幕显示内容。
粗体	表示命令行中的关键字（命令中保持不变、必须照输的部分）或者正文中强调的内容。 标题、警告、注意、小窍门、说明等内容均采用粗体。
<>	语法符号中，表示一个语法对象。
::=	语法符号中，表示定义符，用来定义一个语法对象。定义符左边为语法对象，右边为相应的语法描述。
	语法符号中，表示或者符，限定的语法选项在实际语句中只能出现一个。
{ }	语法符号中，大括号内的语法选项在实际的语句中可以出现 0...N 次（N 为大于 0 的自然数），但是大括号本身不能出现在语句中。
[]	语法符号中，中括号内的语法选项在实际的语句中可以出现 0...1 次，但是中括号本身不能出现在语句中。
关键字	关键字在 DM_SQL 语言中具有特殊意义，在 SQL 语法描述中，关键字以大写形式出现。 但在实际书写 SQL 语句时，关键字既可以大写也可以小写。

访问相关文档

如果您安装了 DM 数据库，可在安装目录的“\doc”子目录中找到 DM 数据库的各种手册与技术丛书。

您也可以通过访问我们的网站 www.dameng.com 阅读或下载 DM 的各种相关文档。

联系我们

如果您有任何疑问或是想了解达梦数据库的最新动态消息，请联系我们：

网址：www.dameng.com

技术服务电话：400-991-6599

技术服务邮箱：dmtech@dameng.com

目录

1 简介.....	1
2 使用 dmagent.....	2
2.1 获取 dmagent.....	2
2.2 配置 dmagent.....	2
2.2.1 agent.ini 配置项.....	2
2.2.2 配置示例.....	8
2.3 启动与停止	10
2.3.1 start 启动脚本	10
2.3.2 服务.....	13
3 注意事项.....	17

1 简介

dmagent 是 DM Web 版管理工具 DEM 部署在远程主机上的代理。DEM 通过 dmagent 可以监控远程主机的相关信息，也可以在远程主机部署 DM MPP、达梦数据守护和读写分离集群等集群系统。

同时，dmagent 也提供了 SQL 命令行查询工具和 TPCC 测试工具。

2 使用 dmagent

2.1 获取 dmagent

可以通过 2 种方式获取到 dmagent 程序。下面分别介绍：

1. 达梦数据库安装目录的 tool 下存有 dmagent。例如 D:\dmdbms\tool\dmagent。

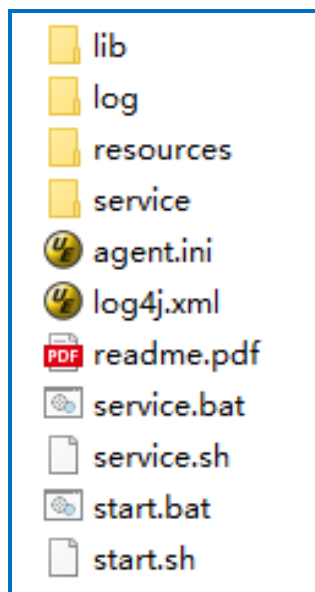


图 2.1 dmagent 目录

2. 登录部署好的 DEM，在智能运维的资源包模块中下载 dmagent。拷贝 dmagent 到要部署的机器上。

2.2 配置 dmagent

用户需要通过 agent.ini 配置文件来配置 dmagent，agent.ini 配置文件存放在 AGENT_HOME 目录下（注：AGENT_HOME 目录表示 dmagent 程序的所在目录）。

2.2.1 agent.ini 配置项

agent.ini 配置项如下所示：

1. general（常规）

参数名	默认值	描述
-----	-----	----

gen_id_policy	0	dmagent 的 ID 生成策略。0:使用 MAC 和 IP 生成; 1:使用 IP 生成
center_url	http://192.168.2.10:8080/dem	dem 的 url 地址
ip_list	[]	指定 dmagent 的 ip 地址列表, 主要应用于存在网络地址映射的环境
collect_command_data	false	是否启用操作系统命令收集服务器资源数据(仅支持 Linux 系统)。 true:启用; false:关闭

2. ap (外部函数功能)

参数名	默认值	描述
ap_enable	true	是否启用外部函数功能。true:启用;false:关闭
ap_port	6363	外部函数端口。使用外部函数时要和 dm.ini 中的 EXTERNAL_JFUN_PORT 保持一致

3. isql (SQL 命令行工具)

参数名	默认值	描述
isql_auto_commit	true	是否自动提交
isql_rs_batch_show	true	结果集是否分批显示
isql_rs_batch_show_size	50	结果集分批显示时, 每批显示的行数
isql_col_max_width	50	每列最大显示字符数, 超过用部分...显示
isql_col_auto_wrap	true	当列字符数超过 isql_col_max_width 时, 是否自动换行以显示全部内容
isql_col_escape	true	是否将查询结果中的 \t \r \n 特殊字符转为空格
isql_highlight_sql	true	sql 语句是否对关键字进行高亮标注

4. build

参数名	默认值	描述
build_config_path	build.xml	编译的配置文件路径

5. service (远程调用服务)

参数名	默认值	描述
service_enable	true	是否启用 dmagent 远程调用服务功能。true:启用; false:关闭
service_port	6364	服务端口号

6. gather (采集数据)

参数名	默认值	描述
gather_enable	true	是否启用 dmagent 信息收集服务功能。 true:启用; false:关闭
gather_center_servlet	/dem/dma_agent	dem 信息收集的 servlet 地址
gather_offline_save_local	false	是否启用信息收集通信失败本地保存功能。true:启用; false:关闭
gather_offline_data_directory	gather	本地保存目录
gather_offline_flush_freq	60	通信失败时数据刷新频率
gather_offline_send_freq	300	通信失败时数据再次发送频率
gather_mf_info_freq	[60]	主机基本信息收集频率(秒)
gather_mf_stat_freq	[60]	收集主机状态信息的频率(秒)
gather_mf_disk_freq	[60]	收集主机磁盘信息的频率(秒)
gather_mf_exec_freq	[60]	收集主机自定义可执行程序的频率(秒)
gather_mf_exec_timeout	10	主机可执行程序超时时间(秒)
gather_mf_process_freq	[60]	收集主机进程信息的频率(秒)
gather_db_info_freq	[60]	数据库基本信息收集频率(秒)
gather_db_stat_freq	[60]	收集数据库状态信息的频率(秒)
gather_db_deadlock_freq	[60]	收集数据库死锁信息的频率(秒)
gather_db_sqlslow_freq	[60]	收集数据库慢 SQL 信息的频率(秒)
gather_db_sqlslow_time	1000	监控中对于慢 SQL 的界定时间(毫秒)
gather_db_sqlcount_freq	[60]	收集数据库高频 SQL 信息的频率(秒)
gather_db_sqlerror_freq	[60]	收集数据库执行 SQL 出错信息的频率(秒)

gather_db_event_freq	[60]	收集数据库事件信息的频率(秒)
gather_db_session_freq	[60]	收集数据库会话信息的频率(秒)
gather_db_tablespace_freq	[60]	收集数据库表空间信息的频率(秒)
gather_db_user_freq	[60]	收集数据库用户锁定信息的频率(秒)
gather_db_table_freq	[60]	收集数据库表数据信息的频率(秒)
gather_db_udsql_freq	[60]	收集数据库自定义 SQL 信息的频率(秒)
gather_db_udsql_timeout	10	数据库自定义 SQL 执行超时时间(秒)
gather_db_audit_record_freq	[0]	收集数据库审计记录的频率(秒)
gather_db_log_history_freq	[60]	收集数据库运行日志记录的频率(秒)
gather_db_rlog_history_freq	[60]	收集数据库事务日志记录的频率(秒)
gather_db_backup_freq	[60]	收集数据库备份信息的频率(秒)

7. upgrade (升级)

参数名	默认值	描述
upgrade_enable	true	是否启用 dmagent 自动升级功能。true:启用; false:关闭
upgrade_center_servlet	/dem/dma_agent	升级 Servlet 路径
upgrade_version_check_freq	600	dmagent 检查升级频率, 单位: 秒。设置为 0 表示不检查

8. smoke (冒烟)

参数名	默认值	描述
smoke_test_case_path		冒烟测例文件目录
smoke_config_path	smoke.xml	冒烟测试配置文件路径
smoke_terminals	1	冒烟测试并行线程数
smoke_terminate_on_error	true	冒烟测试遇到错误是否终止程序
smoke_online_log	false	冒烟测试日志是否发送至远程数据库

9. tpcc

参数名	默认值	描述
tpcc_step	[create_table,load_data,create_index,run_tpcc]	TPCC 执行步骤，分别为创建表、加载数据、创建索引、执行测试。步骤可选，但必须保证顺序
tpcc_db	dameng	数据库类型
tpcc_driver	dm.jdbc.driver.DmDriver	数据库驱动
tpcc_url	jdbc:dm://localhost:5236	数据库 URL
tpcc_user	SYSDBA	数据库用户名
tpcc_password	SYSDBA	数据库密码
tpcc_warehouses	100	TPCC 仓库数量
tpcc_load_workers	10	TPCC 加载数据工作线程数
tpcc_terminals	5	TPCC 终端数
tpcc_terminal_warehouse_fixed	false	TPCC 终端仓库是否固定。 true:固定; false:不固定
tpcc_run_mins	5	TPCC 运行时间，单位：分钟
tpcc_run_trxs_per_terminal	0	TPCC 每个终端执行的事务数
tpcc_limit_trxs_per_min	0	TPCC 每分钟事务数限制
tpcc_new_order_weight	45	TPCC 新订单事务数权重
tpcc_payment_weight	43	TPCC 支付事务数权重
tpcc_order_status_weight	4	TPCC 订单状态事务数权重
tpcc_delivery_weight	4	TPCC 交付事务数权重
tpcc_stock_level_weight	4	TPCC 库存状态事务数权重
tpcc_file_location		TPCC 加载数据文件的路径
tpcc_result_directory_enable	false	TPCC 执行信息和执行结果是否保存。true:保存;

		false:不保存
tpcc_os_collector_script		系统信息收集脚本
tpcc_os_collector_interval	0	系统信息收集间隔
tpcc_os_collector_sshaddr		系统信息收集 SSH 地址
tpcc_os_collector_devices		系统信息收集的设备
tpcc_csv_null_value		TPCC 信息 CSV 文件默认空 值
tpcc_gui_enable	false	是否启用 TPCC GUI。 true:启用; false:关闭
tpcc_gui_interval	1	TPCC GUI 数据点采集时间 间隔, 单位: 秒
tpcc_gui_max_data_number	720	TPCC GUI 面板最大容纳数 据点个数, 达到最大个数之 后保存面板之后重绘

10. command(批量执行 shell)

参数名	默认值	描述
id		批量执行 id, 不可重复
dependency_id		此批量执行依赖的其他批 量执行 id 集合, 格 式:id1,id2,...
user	test	Linux 环境下, 执行 SHELL 脚本的用户名
password	test	Linux 环境下, 执行 SHELL 脚本用户的密码
agent_service_url	rpc://localhost:6364	目标主机的 dmagent 服务 地址
working_directory		SHELL 脚本进程的工作目 录

timeout	0	执行 SHELL 脚本的超时时间
commands		待执行的 SHELL 脚本

11. sql (批量执行 sql)

参数名	默认值	描述
id		批量执行 id, 不可重复
dependency_id		此批量执行依赖的其他批量执行 id 集合, 格式: id1, id2, ...
user	SYSDBA	数据库登录用户名
password	SYSDBA	数据库登录密码
url	jdbc:dm://localhost:5236	Jdbc 连接串, 格式: jdbc:dm://host:port[?propertyName=propertyValue1][&propertyName2=propertyValue2]...
continue_when_error	false	执行 SQL 语句出错是否继续
timeout	0	执行 SQL 语句的超时时间
sqls		待执行的 SQL 语句

2.2.2 配置示例

本章节主要介绍以下两种功能的配置方法:

- 部署与监控功能
- 外部函数功能

部署与监控功能

配置部署与监控功能需要 dmagent 与 DEM 进行连接, 连接 DEM 需要修改 agent.ini 配置文件的 center_url 参数 (参数介绍请参见 [2.2.1 agent.ini 配置项](#)), 将连接地址修改为 DEM 的 URL 地址。

示例如下:

```
#general

gen_id_policy = 0    #agent id general policy, 0: mac & ip, 1: ip

center_url = http://192.168.1.60:8080/dem

ip_list = []
```

agent.ini 配置文件中其他关于部署与监控功能的参数根据功能需求进行调整, 修改完成后启动 dmagent, 登录 DEM 后即可进行部署与监控。

外部函数功能

配置外部函数功能需要打开外部函数功能开关, 即修改 agent.ini 配置文件中的参数 ap_enable 和 ap_port (参数介绍请参见 [2.2.1 agent.ini 配置项](#))。其中, 参数 ap_enable 需设置为 true, 参数 ap_port 的值需与 dm.ini 文件中的参数 EXTERNAL_JFUN_PORT 的值保持一致。

示例如下:

```
#ap

ap_enable = true    #whether enable ap plugin

ap_port = 6363      #ap listen tcp/ip port
```

agent.ini 配置文件修改完成后启动 dmagent, 即可使用 dmagent 的外部函数功能。

2.3 启动与停止

dmagent 存在两种启动模式。

1. 运行 start.bat(sh) 脚本启动 dmagent。
2. 运行 service.bat(sh) 脚本安装 dmagent 服务后，通过系统服务命令或者 service.bat(sh) 脚本启动 dmagent 服务。

2.3.1 start 启动脚本

在 AGENT_HOME 目录下存在启动脚本 start.bat 和 start.sh, 运行 start 脚本可以以命令行的方式启动 dmagent。

start.bat 专门在 Windows 平台下使用, start.sh 专门在 Linux (Unix) 平台下使用。

针对不同的平台, start 启动脚本的启动参数略有不同, 下面分别介绍。

2.3.1.1 Windows 环境

下面介绍 Windows 环境下 dmagent 启动脚本 start.bat 的使用方法。

语法如下:

```
start.bat [<命令参数>] <agent.ini 路径>
```

```
<命令参数>::= smoke|isql|tpcc|tpcc_table|tpcc_load|tpcc_index|tpcc_run
```

参数介绍:

<agent_ini 路径>: 指定 agent.ini 配置文件的路径。

<命令参数>: 运行 dmagent 各种功能, 具体如下所示:

- smoke: 运行冒烟测试, 关于冒烟测试的配置参数请参考 [2.2.1 agent.ini 配置项](#)。
- isql: 连接数据库并打开命令行查询工具 ISQL, 关于 ISQL 工具的相关配置信息请参考 [2.2.1 agent.ini 配置项](#)。
- tpcc: 运行 TPCC 测试程序, 关于 TPCC 测试程序的相关配置信息请参考 [2.2.1 agent.ini 配置项](#), 按照其中的配置参数 tpcc_step 执行相应步骤。
- tpcc_table: 运行 TPCC 测试程序的建表步骤。

- tpcc_load: 运行 TPCC 测试程序的装载数据步骤。
- tpcc_index: 运行 TPCC 测试程序的创建索引步骤。
- tpcc_run: 运行 TPCC 测试程序的测试步骤。

通过 start.bat 脚本启动 dmagent，并运行 dmagent 各种功能的示例如下：

1. 启动 dmagent。

```
D:\dmagent>start.bat D:\dmagent\agent.ini
Agent-V5.0.1.5667-2021.02.08-DEV starting...
_
```

2. 启动 dmagent 命令行查询工具 isql。

```
D:\dmagent>start.bat isql agent.ini
Agent-V5.0.1.5892-2021.03.08-DEV starting...
ISQL> _
```

3. 运行 TPCC 功能。

```
D:\dmagent>start.bat tpcc agent.ini
Agent-V5.0.1.5892-2021.03.08-DEV starting...

Worker 000: Loading ITEM
Worker 001: Loading Warehouse      1
Worker 002: Loading Warehouse      2
Worker 003: Loading Warehouse      3
Worker 004: Loading Warehouse      4
Worker 005: Loading Warehouse      5
Worker 006: Loading Warehouse      6
Worker 007: Loading Warehouse      7
Worker 008: Loading Warehouse      8
Worker 009: Loading Warehouse      9
Worker 000: Loading ITEM done
Worker 000: Loading Warehouse     10
```

4. 打印 dmagent 帮助信息。

```
D:\dmagent>start.bat
Agent-V5.0.1.5667-2021.02.08-DEV

usage as: start [command] agent.ini
command
- "smoke"           - run smoke test
- "isql"            - start sql command
- "tpcc"            - tpcc, run steps configed in agent.ini
- "tpcc_table"      - tpcc, run create table
- "tpcc_load"       - tpcc, run load data
- "tpcc_index"      - tpcc, run create table index
- "tpcc_run"        - tpcc, run test
```

2.3.1.2 Linux(Unix)环境

下面介绍 Linux (Unix) 环境下 dmagent 启动脚本 start.sh 的使用方法。

语法如下：

```
start.sh [-d] [<命令参数>] <agent.ini 路径>
```

或

```
start.sh [--list|-l]
```

```
<命令参数>::= smoke|isql|tpcc|tpcc_table|tpcc_load|tpcc_index|tpcc_run
```

参数介绍：

--list|-l: 打印正在运行的 dmagent 进程相关信息。

-d: 后台启动 dmagent 进程。

<命令参数>、<agent.ini 路径>: 请参考 [2.3.1.1 Windows 环境](#)。

通过 start.sh 脚本启动 dmagent，并运行 dmagent 各种功能的示例如下：

1. 启动 dmagent。

```
[root@localhost dmagent]# ./start.sh /opt/dmagent/agent.ini
Agent-V5.0.1.5667-2021.02.08-DEV starting...
```

2. 后台启动 dmagent。

```
[root@localhost dmagent]# ./start.sh -d /opt/dmagent/agent.ini
Starting dmagent...
.....
The dmagent(pid: 29307) started successfully.
[root@localhost dmagent]#
```

3. 打印正在运行 dmagent 进程。

```
[root@localhost dmagent]# ./start.sh --list
dmagent process list:
root      29307      1 99 08:55 pts/0    00:05:05 java -Xms64m -Xmx2048m -cp ./opt/dmagent/lib/*
```

4. 启动 dmagent 命令行查询工具 isql。

```
[test@test208 dmagent]$ ./start.sh isql ./agent.ini
Tip: if using the deployment feature, it is recommended to start dmagent with the '-d' parameter.
Agent-V5.0.1.5892-2021.03.08-DEV starting...
ISQL>
```

5. 运行 TPCC 功能。


```
[test@test208 dmagent]$ ./start.sh tpcc agent.ini
Tip: if using the deployment feature, it is recommended to start dmagent with the '-d' parameter.
Agent-V5.0.1.5892-2021.03.08-DEV starting...

Worker 000: Loading ITEM
Worker 001: Loading Warehouse      1
Worker 002: Loading Warehouse      2
Worker 003: Loading Warehouse      3
Worker 004: Loading Warehouse      4
Worker 005: Loading Warehouse      5
Worker 006: Loading Warehouse      6
Worker 007: Loading Warehouse      7
Worker 008: Loading Warehouse      8
Worker 009: Loading Warehouse      9
```

6. 打印 dmagent 帮助。

```
[root@localhost dmagent]# ./start.sh
Tip: if using the deployment feature, it is recommended to start dmagent with the '-d' parameter.
Agent-V5.0.1.5683-2021.02.09-DEV

usage as: start [-d] [command] agent.ini
          start [--list|-l]
-d          start agent in the background
--list | -l show agent process list
command
|- "smoke"    - run smoke test
|- "isql"     - start sql command
|- "tpcc"     - tpcc, run steps configed in agent.ini
|- "tpcc_table" - tpcc, run create table
|- "tpcc_load" - tpcc, run load data
|- "tpcc_index" - tpcc, run create table index
|- "tpcc_run"  - tpcc, run test
[root@localhost dmagent]#
```

2.3.2 服务

在 AGENT_HOME 目录下存在 service 脚本 service.bat 和 service.sh。

service.bat 专门在 Windows 平台下使用，service.sh 专门在 Linux (Unix) 平台下使用。

运行 service 脚本可以注册、删除、启动、停止 dmagent 的服务，dmagent 的默认服务名为 DmAgentService。

service 脚本使用方法。

语法如下：

```
SERVICE.BAT {INSTALL|DELETE|START|STOP|STATUS|HELP}
```

或

```
SERVICE.SH {INSTALL|DELETE|START|STOP|STATUS|HELP}
```

参数介绍：

INSTALL: 注册。

DELETE: 删除。

START: 启动。

STOP: 停止。

STATUS: 查看状态。

HELP: 查看帮助信息。

2.3.2.1 Windows 环境

下面介绍 Windows 环境下 service 脚本 service.bat 的使用方法。

1. 注册服务。

```
d:\dmagent>service.bat install
input agent home [d:\dmagent] :
input agent.ini path [d:\dmagent\agent.ini] :
Installation the service DmAgentService completed.
```

2. 删除服务。

```
d:\dmagent>service.bat delete
Delete the service DmAgentService completed.
```

3. 启动服务。

```
d:\dmagent>service.bat start
DmAgentService 服务正在启动。
DmAgentService 服务已经启动成功。
```

4. 停止服务。

```
d:\dmagent>service.bat stop
DmAgentService 服务正在停止。
DmAgentService 服务已成功停止。
```

5. 查看服务状态。

```
d:\dmagent>service.bat status
The service 'DmAgentService' is running.
```

6. 打印 service 脚本帮助信息。

```
d:\dmagent>service.bat help
Usage: service.bat { start | stop | delete | status | restart | install }
service.bat start          #start agent service
service.bat stop           #stop agent service
service.bat delete         #delete agent service
service.bat restart        #restart agent service
service.bat status         #show agent service's status
service.bat install        #install agent service
```



注意: 操作 **dmagent** 服务 (**DmAgentService**) 需要管理员权限。

2.3.2.2 Linux(Unix)环境

下面介绍 Linux (Unix) 环境下 **service** 脚本 **service.sh** 的使用方法。

1. 注册服务。

```
[root@localhost dmagent]# ./service.sh install
input agent home [/opt/dmagent] :
input agent.ini path [/opt/dmagent/agent.ini] :
Installation the service DmAgentService completed.
[root@localhost dmagent]#
```

2. 删除服务。

```
[root@localhost dmagent]# ./service.sh delete
Redirecting to /bin/systemctl stop DmAgentService.service
Delete the service DmAgentService completed.
[root@localhost dmagent]#
```

3. 启动服务。

```
[root@localhost dmagent]# ./service.sh start
Redirecting to /bin/systemctl start DmAgentService.service
[root@localhost dmagent]#
```

4. 停止服务。

```
[root@localhost dmagent]# ./service.sh stop
Redirecting to /bin/systemctl stop DmAgentService.service
[root@localhost dmagent]#
```

5. 查看服务状态。

```
[root@localhost dmagent]# ./service.sh status
Redirecting to /bin/systemctl status DmAgentService.service
● DmAgentService.service - Dm Agent Service
   Loaded: loaded (/usr/lib/systemd/system/DmAgentService.service; enabled; vendor preset
   Active: active (running) since 二 2021-02-09 09:18:16 CST; 33s ago
   Process: 31149 ExecStart=/opt/dmagent/service/DmAgentService start (code=exited, status
   Main PID: 31195 (java)
   Tasks: 44
   CGroup: /system.slice/DmAgentService.service
           └─31195 java -Xms64m -Xmx2048m -cp ./opt/dmagent/lib/* -Ddameng.log.file=/opt

2月 09 09:18:11 localhost.localdomain systemd[1]: Starting Dm Agent Service...
2月 09 09:18:16 localhost.localdomain DmAgentService[31149]: Starting dmagent[ 确定 ]
2月 09 09:18:16 localhost.localdomain systemd[1]: Started Dm Agent Service.
[root@localhost dmagent]#
```

6. 打印 service 脚本帮助信息。

```
[root@localhost dmagent]# ./service.sh help
Usage: service.sh { start | stop | delete | status | restart | install }
   service.sh start           #start agent service
   service.sh stop            #stop agent service
   service.sh delete          #delete agent service
   service.sh restart         #restart agent service
   service.sh status          #show agent service's status
   service.sh install         #install agent service

[root@localhost dmagent]#
```



注意：操作 dmagent 服务 (DmAgentService) 需要管理员权限。

3 注意事项

1. 在远程主机使用 dmagent，需首先手动将 dmagent 拷贝到远程主机；然后根据不同的平台，选用 service (服务方式) 或 start (命令行方式) 运行 dmagent。

2. dmagent 未自带 JAVA 运行时环境，所以运行 dmagent 需要用户提前设置好 JRE_HOME 或 JAVA_HOME 的环境变量，JAVA 运行时环境需要 1.8 及以上的版本。

3. 在 agent.ini 中 collect_command_data 参数设置为 true 时，Linux 环境下数据监控功能通过操作系统命令实现，启用数据监控功能前请通过 com.dameng.common.util.system.SystemInfoTest 类功能进行检查。下面详细介绍：

1) SystemInfoTest 类功能介绍：

检测监控收集功能涉及的命令和文件是否可用。

打印命令输出和文件内容。

打印操作系统信息和硬件信息。

2) SystemInfoTest 类使用方法

方法如下：

```
SystemInfoTest [options]
```

```
Options:
```

```
-c, --check      检查命令和文件是否可用
-d, --data       打印命令输出和文件内容
-s, --systeminfo 打印操作系统信息和硬件信息
-h, --help       打印帮助信息
```

3) 使用 check 功能进行检查：

通过 java 命令调用 agent.jar 执行 SystemInfoTest 类的 check 功能，检查监控相关命令及文件是否可以正常运行或读取。若输出结果全为 success 则表示可正常执行监控功能，若出现 fail 则表示检查失败会影响监控功能的正常使用，请咨询系统管理员。

例如，假设 agent 部署目录为 /opt/dmagent

```
java -cp /opt/dmagent/lib/*:/opt/dmagent/lib/agent-6.0.1.jar
com.dameng.common.util.system.SystemInfoTest -c
```

输出结果：

```
checking command df.....success!  
checking command getconf.....success!  
checking command getent.....success!
```

咨询热线：400-991-6599

技术支持：dmtech@dameng.com

官网网址：www.dameng.com



武汉达梦数据库股份有限公司
Wuhan Dameng Database Co.,Ltd.

地址：武汉市东湖新技术开发区高新大道999号未来科技大厦C3栋16—19层

16th-19th Floor, Future Tech Building C3, No.999 Gaoxin Road, Donghu New Tech Development Zone,Wuhan,Hubei Province,China

电话：(+86) 027-87588000 传真：(+86) 027-87588810
