



**openEuler**  
**20.03 LTS**

## **A-Tune 用户指南**

发布日期      2020-03-26

# 目 录

法律申明 .....	iv
前言 .....	v
<b>1 认识 A-Tune .....</b>	<b>6</b>
1.1 简介 .....	6
1.2 架构 .....	7
1.3 支持特性与业务模型 .....	7
<b>2 安装与部署 .....</b>	<b>10</b>
2.1 软硬件要求 .....	10
2.2 环境准备 .....	10
2.3 安装 A-Tune .....	10
2.3.1 安装模式介绍 .....	10
2.3.2 安装操作 .....	11
2.4 部署 A-Tune .....	12
2.4.1 配置介绍 .....	12
2.5 启动 A-Tune .....	14
<b>3 使用方法 .....</b>	<b>15</b>
3.1 总体说明 .....	15
3.2 查询负载类型 .....	16
3.2.1 list .....	16
3.3 分析负载类型并自优化 .....	17
3.3.1 analysis .....	17
3.4 自定义模型 .....	17
3.4.1 define .....	17
3.4.2 collection .....	18
3.4.3 train .....	19
3.4.4 undefine .....	20
3.5 查询 profile .....	20
3.5.1 info .....	20
3.6 更新 profile .....	22

---

3.6.1 update .....	22
3.7 激活 profile .....	22
3.7.1 profile .....	22
3.8 回滚 profile .....	23
3.8.1 rollback .....	23
3.9 更新数据库 .....	23
3.9.1 upgrade .....	23
3.10 系统信息查询 .....	24
3.10.1 check .....	24
3.11 参数自调优 .....	24
3.11.1 tuning .....	24
<b>4 常见问题与解决方法 .....</b>	<b>31</b>
<b>5 附录 .....</b>	<b>32</b>
5.1 术语和缩略语 .....	32

---

## 法律申明

---

版权所有 © 2020 华为技术有限公司。

您对“本文档”的复制，使用，修改及分发受知识共享(Creative Commons)署名一相同方式共享 4.0 国际公共许可协议(以下简称“CC BY-SA 4.0”)的约束。为了方便用户理解，您可以通过访问 <https://creativecommons.org/licenses/by-sa/4.0/> 了解 CC BY-SA 4.0 的概要 (但不是替代)。CC BY-SA 4.0 的完整协议内容您可以访问如下网址获取：  
<https://creativecommons.org/licenses/by-sa/4.0/legalcode>。

### 商标声明

A-Tune、openEuler 为华为技术有限公司的商标。本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

### 免责声明

本文档仅作为使用指导，除非适用法强制规定或者双方有明确书面约定，华为技术有限公司对本文档中的所有陈述、信息和建议不做任何明示或默示的声明或保证，包括但不限于不侵权，时效性或满足特定目的的担保。

# 前言

## 概述



本文档介绍 openEuler 系统性能自优化软件 A-Tune 的安装部署和使用方法，以指导用户快速了解并使用 A-Tune。

## 读者对象

本文档适用于使用 openEuler 系统并希望了解和使用 A-Tune 的社区开发者、开源爱好者以及相关合作伙伴。使用人员需要具备基本的 Linux 操作系统知识。

## 符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 须知	用于传递设备或环境安全警示信息。如不可避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
 说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

# 1 认识 A-Tune

- 1.1 简介
- 1.2 架构
- 1.3 支持特性与业务模型

## 1.1 简介

操作系统作为衔接应用和硬件的基础软件，如何调整系统和应用配置，充分发挥软硬件能力，从而使业务性能达到最优，对用户至关重要。然而，运行在操作系统上的业务类型成百上千，应用形态千差万别，对资源的要求各不相同。当前硬件和基础软件组成的应用环境涉及高达 7000 多个配置对象，随着业务复杂度和调优对象的增加，调优所需的时间成本呈指数级增长，导致调优效率急剧下降，调优成为了一项极其复杂的工程，给用户带来巨大挑战。

其次，操作系统作为基础设施软件，提供了大量的软硬件管理能力，每种能力适用场景不尽相同，并非对所有的应用场景都通用有益，因此，不同的场景需要开启或关闭不同的能力，组合使用系统提供的各种能力，才能发挥应用程序的最佳性能。

另外，实际业务场景成千上万，计算、网络、存储等硬件配置也层出不穷，实验室无法遍历穷举所有的应用和业务场景，以及不同的硬件组合。

为了应对上述挑战，openEuler 推出了 A-Tune。

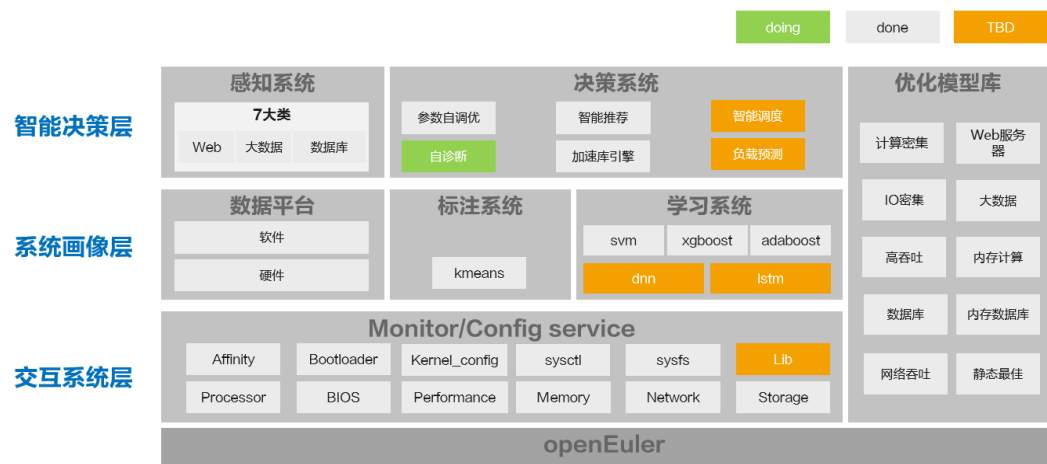
A-Tune 是一款基于 AI 开发的系统性能优化引擎，它利用人工智能技术，对业务场景建立精准的系统画像，感知并推理出业务特征，进而做出智能决策，匹配并推荐最佳的系统参数配置组合，使业务处于最佳运行状态。



## 1.2 架构

A-Tune 核心技术架构如下图，主要包括智能决策、系统画像和交互系统三层。

- 智能决策层：包含感知和决策两个子系统，分别完成对应用的智能感知和对系统的调优决策。
- 系统画像层：主要包括标注和学习系统，标注系统用于业务模型的聚类，学习系统用于业务模型的学习和分类。
- 交互系统层：用于各类系统资源的监控和配置，调优策略执行在本层进行。



## 1.3 支持特性与业务模型

### 支持特性

A-Tune 支持的主要特性、特性成熟度以及使用建议请参见表 1-1。

表1-1 特性成熟度

特性	成熟度	使用建议
----	-----	------

特性	成熟度	使用建议
七大类 11 款应用负载类型 自动优化	已测试	试用
自定义负载类型和业务模型	已测试	试用
参数自调优	已测试	试用

## 支持业务模型

根据应用的负载特征，A-Tune 将业务分为七大类，各类型的负载特征和 A-Tune 支持的应用请参见表 1-2。

表1-2 支持的业务类型和应用

负载模型	业务类型	负载特征	支持的应用
default	默认类型	CPU、内存带宽、网络、IO 各维度资源使用率都不高	N/A
webserver	https 应用	CPU 使用率高	Nginx
big_database	数据库	<ul style="list-style-type: none"><li>关系型数据库 读：CPU、内存带宽、网络使用率高 写：IO 使用率高</li><li>非关系型数据库 CPU、IO 使用率高</li></ul>	MongoDB、MySQL、PostgreSQL、MariaDB
big_data	大数据	CPU、IO 使用率较高	Hadoop、Spark
in-memory_computing	内存密集型应用	CPU、内存带宽使用率高	SPECjbb2015
in-memory_database	计算+网络密集型应用	CPU 单核使用率高，多实例下网络使用率高	Redis
single_computer_intensive_jobs	计算密集型应用	CPU 单核使用率高，部分子项内存带宽使用率高	SPECint2006
communication	网络密集型应用	CPU、网络使用率高	Dubbo



负载模型	业务类型	负载特征	支持的应用
	用		
idle	系统 idle	系统处于空闲状态，无任何应用运行	N/A

# 2 安装与部署

本章介绍如何安装和部署 A-Tune。

- [2.1 软硬件要求](#)
- [2.2 环境准备](#)
- [2.3 安装 A-Tune](#)
- [2.4 部署 A-Tune](#)
- [2.5 启动 A-Tune](#)

## 2.1 软硬件要求

### 硬件要求

- 鲲鹏 920 处理器

### 软件要求

- 操作系统：openEuler 20.03 LTS

## 2.2 环境准备

安装 openEuler 系统，安装方法参考《openEuler 20.03 LTS 安装指南》。

## 2.3 安装 A-Tune

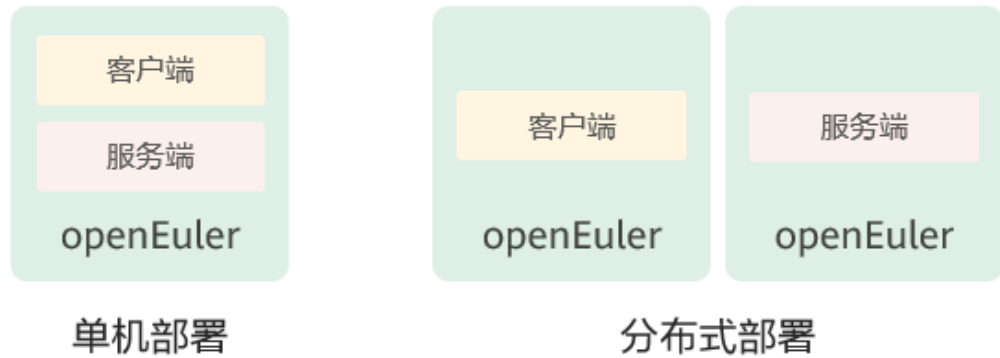
本章介绍 A-Tune 的安装模式和安装方法。

### 2.3.1 安装模式介绍

A-Tune 支持单机模式和分布式模式安装：

- 单机模式  
client 和 server 安装到同一台机器上。
- 分布式模式  
client 和 server 分别安装在不同的机器上。

两种安装模式的简单图示如下：



## 2.3.2 安装操作

安装 A-Tune 的操作步骤如下：

步骤 1 挂载 openEuler 的 iso 文件。

```
# mount openEuler-20.03-LTS-aarch64-dvd.iso /mnt
```

步骤 2 配置本地 yum 源。

```
# vim /etc/yum.repos.d/local.repo
```

配置内容如下所示：

```
[local]
name=local
baseurl=file:///mnt
gpgcheck=0
enabled=1
```

步骤 3 安装 A-Tune 服务端。

### 📖 说明

本步骤会同时安装服务端和客户端软件包，对于单机部署模式，请跳过步骤 4。

```
# yum install atune -y
```

步骤 4 若为分布式部署，请安装 A-Tune 客户端。

```
# yum install atune-client -y
```

步骤 5 验证是否安装成功。

```
# rpm -qa | grep atune
atune-client-xxx
atune-db-xxx
atune-xxx
```

有如上回显信息表示安装成功。

----结束

## 2.4 部署 A-Tune

本章介绍 A-Tune 的配置部署。

### 2.4.1 配置介绍

A-Tune 配置文件/etc/atuned/atuned.cnf 的配置项说明如下：

- A-Tune 服务启动配置  
可根据需要进行修改。
  - protocol: 系统 grpc 服务使用的协议，unix 或 tcp，unix 为本地 socket 通信方式，tcp 为 socket 监听端口方式。默认为 unix。
  - address: 系统 grpc 服务的侦听地址，默认为 unix socket，若为分布式部署，需修改为侦听的 ip 地址。
  - port: 系统 grpc 服务的侦听端口，范围为 0~65535 未使用的端口。如果 protocol 配置是 unix，则不需要配置。
  - rest\_port: 系统 restservice 的侦听端口，范围为 0~65535 未使用的端口。
  - sample\_num: 系统执行 analysis 流程时采集样本的数量。
- system 信息  
system 为系统执行相关的优化需要用到的参数信息，必须根据系统实际情况进行修改。
  - disk: 执行 analysis 流程时需要采集的对应磁盘的信息或执行磁盘相关优化时需要指定的磁盘。
  - network: 执行 analysis 时需要采集的对应的网卡的信息或执行网卡相关优化时需要指定的网卡。
  - user: 执行 ulimit 相关优化时用到的用户名。目前只支持 root 用户。
  - tls: 开启 A-Tune 的 gRPC 和 http 服务 SSL/TLS 证书校验，默认不开启。开启 TLS 后 atune-adm 命令在使用前需要设置以下环境变量方可与服务端进行通讯：
    - export ATUNE\_TLS=yes
    - export ATUNE\_CLICERT=<客户端证书路径>
  - tlsservercertfile: gPRC 服务端证书路径。
  - tlsserverkeyfile: gPRC 服务端秘钥路径。
  - tlshttplibcertfile: http 服务端证书路径。
  - tlshttplibkeyfile: http 服务端秘钥路径。
  - tlshttplibcacertfile: http 服务端 CA 证书路径。
- 日志信息  
根据情况修改日志的路径和级别，默认的日志信息在/var/log/messages 中。

- **monitor 信息**  
为系统启动时默认采集的系统硬件信息。

## 配置示例

```
##### server #####
# atuned config
[server]
# the protocol grpc server running on
# ranges: unix or tcp
protocol = unix

# the address that the grpc server to bind to
# default is unix socket /var/run/atuned/atuned.sock
# ranges: /var/run/atuned/atuned.sock or ip
address = /var/run/atuned/atuned.sock

# the atuned grpc listening port, default is 60001
# the port can be set between 0 to 65535 which not be used
port = 60001

# the rest service listening port, default is 8383
# the port can be set between 0 to 65535 which not be used
rest port = 8383

# when run analysis command, the numbers of collected data.
# default is 20
sample num = 20

# Enable gRPC and http server authentication SSL/TLS
# default is false
# tls = true
# tlsservercertfile = /etc/atuned/server.pem
# tlsserverkeyfile = /etc/atuned/server.key
# tlshttplibcertfile = /etc/atuned/http/server.pem
# tlshttplibkeyfile = /etc/atuned/http/server.key
# tlshttplibcacertfile = /etc/atuned/http/cacert.pem

##### log #####
# Either "debug", "info", "warn", "error", "critical", default is "info"
level = info

##### monitor #####
[monitor]
# With the module and format of the MPI, the format is {module} {purpose}
# The module is Either "mem", "net", "cpu", "storage"
# The purpose is "topo"
module = mem topo, cpu topo

##### system #####
# you can add arbitrary key-value here, just like key = value
# you can use the key in the profile
[system]
# the disk to be analysis
disk = sda
```

```
# the network to be analysis
network = enp189s0f0

user = root
```

## 2.5 启动 A-Tune

A-Tune 安装完成后，需要启动 A-Tune 服务才能使用。

- 启动 atuned 服务：

```
# systemctl start atuned
```

- 查询 atuned 服务状态：

```
# systemctl status atuned
```

若回显为如下，则服务启动成功。

```
● atuned.service - A-Tune Daemon
   Loaded: loaded (/usr/lib/systemd/system/atuned.service; disabled; vendor preset: disabled)
   Active: active (running) since Sat 2019-12-21 19:28:47 CST; 7s ago
     Main PID: 94790 (atuned)
        Tasks: 19 (limit: 104856)
       Memory: 132.2M
      CGroup: /system.slice/atuned.service
              └─94790 /usr/bin/atuned
                  └─94797 python3 /usr/libexec/atuned/analysis/app.py /etc/atuned/atuned.cnf
                      └─94801 /usr/bin/python3 -c from multiprocessing.semaphore_tracker import main;main(3)
```

# 3 使用方法

用户可以通过命令行客户端 `atune-adm` 使用 A-Tune 提供的功能。本章介绍 A-Tune 客户端包含的功能和使用方法。

- 3.1 总体说明
- 3.2 查询负载类型
- 3.3 分析负载类型并自优化
- 3.4 自定义模型
- 3.5 查询 `profile`
- 3.6 更新 `profile`
- 3.7 激活 `profile`
- 3.8 回滚 `profile`
- 3.9 更新数据库
- 3.10 系统信息查询
- 3.11 参数自调优

## 3.1 总体说明

- `atune-adm` 支持的命令可以通过 `atune-adm help/--help/-h` 查询。
- 使用方法中所有命令的使用举例都是在单机部署模式下，如果是在分布式部署模式下，需要指定服务器 IP 和端口号，例如：

```
# atune-adm -a 192.168.3.196 -p 60001 list
```
- `define`、`update`、`undefine`、`collection`、`train`、`upgrade` 不支持远程执行。
- 命令格式中，`[ ]` 表示参数可选，`<>` 表示参数必选，具体参数由实际情况确定。
- 命令格式中，各命令含义如下：
  - `WORKLOAD_TYPE`: 用户自定义负载类型的名称，负载支持的类型参考 `list` 命令查询结果。

- PROFILE\_NAME: 用户自定义 profile 的名称
- PROFILE\_PATH: 用户自定义 profile 的路径

## 3.2 查询负载类型

### 3.2.1 list

#### 功能描述

查询系统当前支持的 workload\_type 和对应的 profile，以及当前处于 active 状态的 profile。

#### 命令格式

**atune-adm list**

#### 使用示例

```
# atune-adm list

Support WorkloadTypes:
+-----+-----+-----+
| WorkloadType          | ProfileName          | Active |
+=====+=====+=====+
| default                | default              | true   |
+-----+-----+-----+
| webserver              | ssl webserver        | false  |
+-----+-----+-----+
| big database           | database              | false  |
+-----+-----+-----+
| big data               | big data              | false  |
+-----+-----+-----+
| in-memory computing    | in-memory computing  | false  |
+-----+-----+-----+
| in-memory database     | in-memory database   | false  |
+-----+-----+-----+
| single computer intensive jobs | compute-intensive   | false  |
+-----+-----+-----+
| communication          | rpc communication    | false  |
+-----+-----+-----+
| idle                   | default              | false  |
+-----+-----+-----+
```

#### 说明

Active 为 true 表示当前激活的 profile，示例表示当前激活的是 default 类型对应的 profile。



## 3.3 分析负载类型并自优化

### 3.3.1 analysis

#### 功能描述

采集系统的实时统计数据进行分析负载类型识别，并进行自动优化。

#### 命令格式

**atune-adm analysis** [OPTIONS]

#### 参数说明

- OPTIONS

参数	描述
--model, -m	用户自训练产生的新模型

#### 使用示例

- 使用默认模型进行分类识别

```
# atune-adm analysis
```

- 使用自训练的模型进行识别

```
# atune-adm analysis --model /usr/libexec/atuned/analysis/models/new-model.m
```

## 3.4 自定义模型

A-Tune 支持用户定义并学习新模型。定义新模型的操作流程如下：

步骤 1 用 **define** 命令定义 **workload\_type** 和 **profile**

步骤 2 用 **collection** 命令收集 **workload\_type** 对应的画像数据

步骤 3 用 **train** 命令训练得到模型

----结束

### 3.4.1 define

#### 功能描述

添加用户自定义的 **workload\_type**，及对应的 **profile** 优化项。

## 命令格式

**atune-adm define** <WORKLOAD\_TYPE> <PROFILE\_NAME> <PROFILE\_PATH>

## 使用示例

新增一个 workload type，workload type 的名称为 test\_type，profile name 的名称为 test\_name，优化项的配置文件为 example.conf。

```
# atune-adm define test_type test_name ./example.conf
```

example.conf 可以参考如下方式书写（以下各优化项非必填，仅供参考），也可通过 **atune-adm info** 查看已有的 profile 是如何书写的。

```
[main]
# list its parent profile
[tip]
# the recommended optimization, which should be performed manually
[check]
# check the environment
[affinity.irq]
# to change the affinity of irq
[affinity.task]
# to change the affinity of tasks
[bios]
# to change the bios config
[bootloader.grub2]
# to change the grub2 config
[kernel config]
# to change the kernel config
[script]
# the script extension of cpi
[sysctl]
# to change the /proc/sys/* config
[sysfs]
# to change the /sys/* config
[systemctl]
# to change the system service config
[ulimit]
# to change the resources limit of user
```

## 3.4.2 collection

### 功能描述

采集业务运行时系统的全局资源使用情况以及 OS 的各项状态信息，并将收集的结果保存到 csv 格式的输出文件中，作为模型训练的输入数据集。

#### 说明

- 本命令依赖采样工具 perf, mpstat, vmstat, iostat, sar。
- CPU 型号目前仅支持鲲鹏 920，可通过 dmidecode -t processor 检查 CPU 型号。

## 命令格式

**atune-adm collection** <OPTIONS>

## 参数说明

- OPTIONS

参数	描述
--filename, -f	生成的用于训练的 csv 文件名: 名称-时间戳.csv
--output_path, -o	生成的 csv 文件的存放路径, 需提供绝对路径
--disk, -b	业务运行时实际使用的磁盘, 如/dev/sda
--network, -n	业务运行时使用的网络接口, 如 eth0
--workload_type, -t	标记业务的负载类型, 作为训练时使用的标签
--duration, -d	业务运行时采集数据的时间, 单位秒, 默认采集时间 1200 秒
--interval, -i	采集数据的时间间隔, 单位秒, 默认采集间隔 5 秒

## 使用示例

```
# atune-adm collection --filename name --interval 5 --duration 1200 --output_path  
/home/data --disk sda --network eth0 --workload_type test_type
```

### 3.4.3 train

## 功能描述

使用采集的数据进行模型的训练。训练时至少采集两种 workload\_type 的数据, 否则训练会出错。

## 命令格式

**atune-adm train** <OPTIONS>

## 参数说明

- OPTIONS

参数	描述
--data_path, -d	存放模型训练所需的 csv 文件的目录
--output_file, -o	训练生成的新模型

## 使用示例

使用 data 目录下的 csv 文件作为训练输入，生成的新模型 new-model.m 存放在 model 目录下。

```
# atune-adm train --data_path /home/data --output_file  
/usr/libexec/atuned/analysis/models/new-model.m
```

## 3.4.4 undefine

### 功能描述

删除用户自定义的 workload\_type。

### 命令格式

**atune-adm undefine <WORKLOAD\_TYPE>**

### 使用示例

删除自定义的负载类型 test\_type。

```
# atune-adm undefine test_type
```

## 3.5 查询 profile

### 3.5.1 info

#### 功能描述

查看 workload\_type 对应的 profile 内容。

#### 命令格式

**atune-adm info <WORKLOAD\_TYPE>**

#### 使用示例

查看 webserver 的 profile 内容：

```
# atune-adm info webserver  
  
*** ssl webserver:  
  
#  
# webserver tuned configuration  
#  
[main]  
#TODO CONFIG
```

```
[kernel config]
#TODO CONFIG

[bios]
#TODO CONFIG

[sysfs]
#TODO CONFIG

[sysctl]
fs.file-max=6553600
fs.suid dumpable = 1
fs.aio-max-nr = 1048576
kernel.shmmax = 68719476736
kernel.shmall = 4294967296
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
net.ipv4.tcp tw reuse = 1
net.ipv4.tcp syncookies = 1
net.ipv4.ip local port range = 1024      65500
net.ipv4.tcp max tw buckets = 5000
net.core.somaxconn = 65535
net.core.netdev max backlog = 262144
net.ipv4.tcp max orphans = 262144
net.ipv4.tcp max syn backlog = 262144
net.ipv4.tcp timestamps = 0
net.ipv4.tcp synack retries = 1
net.ipv4.tcp syn retries = 1
net.ipv4.tcp fin timeout = 1
net.ipv4.tcp keepalive time = 60
net.ipv4.tcp mem = 362619      483495      725238
net.ipv4.tcp rmem = 4096      87380      6291456
net.ipv4.tcp wmem = 4096      16384      4194304
net.core.wmem default = 8388608
net.core.rmem default = 8388608
net.core.rmem max = 16777216
net.core.wmem max = 16777216

[systemctl]
sysmonitor=stop
irqbalance=stop

[bootloader.grub2]
selinux=0
iommu.passthrough=1

[tip]
bind your master process to the CPU near the network = affinity
bind your network interrupt to the CPU that has this network = affinity
relogin into the system to enable limits setting = OS

[script]
openssl hpre = 0
prefetch = off
```

```
[ulimit]
{user}.hard.nofile = 102400
{user}.soft.nofile = 102400

[affinity.task]
#TODO CONFIG

[affinity irq]
#TODO CONFIG

[check]
#TODO CONFIG
```

## 3.6 更新 profile

用户根据需要更新已有 profile。

### 3.6.1 update

#### 功能描述

将 workload\_type 原来的优化项更新为 new.conf 中的内容。

#### 命令格式

**atune-adm update** <WORKLOAD\_TYPE> <PROFILE\_NAME> <PROFILE\_FILE>

#### 使用示例

更新负载类型为 test\_type，优化项名称为 test\_name 的优化项为 new.conf。

```
# atune-adm update test_type test_name ./new.conf
```

## 3.7 激活 profile

### 3.7.1 profile

#### 功能描述

手动激活 workload\_type 对应的 profile，使得 workload\_type 处于 active 状态。

#### 命令格式

**atune-adm profile** <WORKLOAD\_TYPE>

## 参数说明

WORKLOAD\_TYPE 支持的类型参考 list 命令查询结果。

## 使用示例

激活 webserver 对应的 profile 配置。

```
# atune-adm profile webserver
```

## 3.8 回滚 profile

### 3.8.1 rollback

#### 功能描述

回退当前的配置到系统的初始配置。

#### 命令格式

**atune-adm rollback**

#### 使用示例

```
# atune-adm rollback
```

## 3.9 更新数据库

### 3.9.1 upgrade

#### 功能描述

更新系统的数据库。

#### 命令格式

**atune-adm upgrade <DB\_FILE>**

#### 参数说明

- DB\_FILE  
新的数据库文件路径

#### 使用示例

数据库更新为 new\_sqlite.db。

```
# atune-adm upgrade ./new_sqlite.db
```

## 3.10 系统信息查询

### 3.10.1 check

#### 功能描述

检查系统当前的 cpu、bios、os、网卡等信息。

#### 命令格式

**atune-adm check**

#### 使用示例

```
# atune-adm check
cpu information:
  cpu:0  version: Kunpeng 920-6426  speed: 2600000000 HZ  cores: 64
  cpu:1  version: Kunpeng 920-6426  speed: 2600000000 HZ  cores: 64
system information:
  DMIBIOSVersion: 0.59
  OSRelease: 4.19.36-vhulk1906.3.0.h356.eulerosv2r8.aarch64
network information:
  name: eth0          product: HNS GE/10GE/25GE RDMA Network Controller
  name: eth1          product: HNS GE/10GE/25GE Network Controller
  name: eth2          product: HNS GE/10GE/25GE RDMA Network Controller
  name: eth3          product: HNS GE/10GE/25GE Network Controller
  name: eth4          product: HNS GE/10GE/25GE RDMA Network Controller
  name: eth5          product: HNS GE/10GE/25GE Network Controller
  name: eth6          product: HNS GE/10GE/25GE RDMA Network Controller
  name: eth7          product: HNS GE/10GE/25GE Network Controller
  name: docker0       product:
```

## 3.11 参数自调优

A-Tune 提供了最佳配置的自动搜索能力，免去人工反复做参数调整、性能评价的调优过程，极大地提升最优配置的搜寻效率。

### 3.11.1 tuning

#### 功能描述

使用指定的项目文件对参数进行动态空间的搜索，找到当前环境配置下的最优解。



命令格式

说明

- 在运行命令前，需要满足如下条件：
- 1. 编辑好服务端 yaml 配置文件，且需要服务端管理员将该配置文件放到服务端的 /etc/atuned/tuning/目录下。
  - 2. 编辑好客户端 yaml 配置文件并放在客户端任一目录。

```
atune-adm tuning [OPTIONS] <PROJECT_YAML>
```

参数说明

OPTIONS

参数	描述
--restore, -r	恢复 tuning 优化前的初始配置
--project, -p	指定需要恢复的 yaml 文件中的项目名称

说明

当使用参数时，上述两个参数需要同时使用，且 -p 参数后需要跟具体的项目名称。

- PROJECT\_YAML：客户端 yaml 配置文件。

配置说明

表3-1 服务端 yaml 文件

配置名称	配置说明	参数类型	取值范围
project	项目名称。	字符串	-
startworkload	待调优服务的启动脚本。	字符串	-
stopworkload	待调优服务的停止脚本。	字符串	-
maxiterations	最大调优迭代次数，用于限制客户端的迭代次数。一般来说，调优迭代次数越多，优化效果越好，但所需时间越长。用户必须根据实际的业务场景进行配置。	整型	>10

配置名称	配置说明	参数类型	取值范围
object	需要调节的参数项及信息。 object 配置项请参见表 3-2。	-	-

表3-2 object 项配置说明

配置名称	配置说明	参数类型	取值范围
name	待调参数名称	字符串	-
desc	待调参数描述	字符串	-
get	查询参数值的脚本	-	-
set	设置参数值的脚本	-	-
needrestart	参数生效是否需要重启业务	枚举	"true", "false"
type	参数的类型，目前支持 discrete, continuous 两种类型，对应离散型、连续型参数	枚举	"discrete", "continuous"
dtype	该参数仅在 type 为 discrete 类型时配置，目前支持 int 和 string 两种类型	枚举	int, string
scope	参数设置范围，仅在 type 为 discrete 且 dtype 为 int 时或者 type 为 continuous 时生效	整型	用户自定义，取值在该参数的合法范围
step	参数值步长，dtype 为 int 时使用	整型	用户自定义

配置名称	配置说明	参数类型	取值范围
items	参数值在 scope 定义范围之外的枚举值，dtype 为 int 时使用	整型	用户自定义，取值在该参数的合法范围
options	参数值的枚举范围，dtype 为 string 时使用	字符串	用户自定义，取值在该参数的合法范围
ref	参数的推荐初始值	整型或字符串	用户自定义，取值在该参数的合法范围

表3-3 客户端 yaml 文件配置说明

配置名称	配置说明	参数类型	取值范围
project	项目名称，需要与服务端对应配置文件中的 project 匹配	字符串	-
iterations	调优迭代次数	整型	$\geq 10$
benchmark	性能测试脚本	-	-
evaluations	性能测试评估指标 evaluations 配置项请参见表 3-4	-	-

表3-4 evaluations 项配置说明

配置名称	配置说明	参数类型	取值范围
name	评价指标名称	字符串	-
get	获取性能评估结果的脚本	-	-
type	评估结果的正负类型，positive 代表最小化性能值，negative 代表最大	枚举	"positive", "negative"

配置名称	配置说明	参数类型	取值范围
	化对应性能值		
weight	该指标的权重百分比, 0-100	整型	0-100
threshold	该指标的最低性能要求	整型	用户指定

## 配置示例

服务端 yaml 文件配置示例:

```
project: "example"
maxiterations: 10
startworkload: ""
stopworkload: ""
object :
-
  name : "vm.swappiness"
  info :
    desc : "the vm.swappiness"
    get : "sysctl -a | grep vm.swappiness"
    set : "sysctl -w vm.swappiness=$value"
    needrestart: "false"
    type : "continuous"
    scope :
      - 0
      - 10
    ref : 1
-
  name : "irqbalance"
  info :
    desc : "system irqbalance"
    get : "systemctl status irqbalance"
    set : "systemctl $value sysmonitor;systemctl $value irqbalance"
    needrestart: "false"
    type : "discrete"
    options:
      - "start"
      - "stop"
    dtype : "string"
    ref : "start"
-
  name : "net.tcp min tso segs"
  info :
    desc : "the minimum tso number"
    get : "cat /proc/sys/net/ipv4/tcp min tso segs"
    set : "echo $value > /proc/sys/net/ipv4/tcp min tso segs"
    needrestart: "false"
    type : "continuous"
```

```
scope:
  - 1
  - 16
ref : 2
-
name : "prefetcher"
info :
  desc : ""
  get : "cat /sys/class/misc/prefetch/policy"
  set : "echo $value > /sys/class/misc/prefetch/policy"
  needrestart: "false"
  type : "discrete"
  options:
    - "0"
    - "15"
  dtype : "string"
  ref : "15"
-
name : "kernel.sched min granularity ns"
info :
  desc : "Minimal preemption granularity for CPU-bound tasks"
  get : "sysctl kernel.sched min granularity ns"
  set : "sysctl -w kernel.sched min granularity ns=$value"
  needrestart: "false"
  type : "continuous"
  scope:
    - 5000000
    - 50000000
  ref : 10000000
-
name : "kernel.sched latency ns"
info :
  desc : ""
  get : "sysctl kernel.sched latency ns"
  set : "sysctl -w kernel.sched latency ns=$value"
  needrestart: "false"
  type : "continuous"
  scope:
    - 10000000
    - 100000000
  ref : 16000000
```

客户端 yam1 文件配置示例:

```
project: "example"
iterations : 10
benchmark : "sh /home/Benchmarks/mysql/tunning mysql.sh"
evaluations :
-
  name: "tps"
  info:
    get: "echo -e '$out' |grep 'transactions:' |awk '{print $3}' | cut -c 2-"
    type: "negative"
    weight: 100
    threshold: 100
```

## 使用示例

- 进行 tuning 调优

```
# atune-adm tuning example-client.yaml
```

- 恢复 tuning 调优前的初始配置，example 为 yaml 文件中的项目名称

```
# atune-adm tuning --restore --project example
```

# 4 常见问题与解决方法

**问题 1: train 命令训练模型出错，提示 “training data faild”。**

原因: collection 命令只采集一种类型的数据。

解决方法: 至少采集两种数据类型的数据进行训练。

**问题 2: atune-adm 无法连接 atuned 服务。**

可能原因:

1. 检查 atuned 服务是否启动，并检查 atuned 侦听地址。

```
# systemctl status atuned  
# netstat -nap | atuned
```

2. 防火墙阻止了 atuned 的侦听端口。
3. 系统配置了 http 代理导致无法连接。

解决方法:

1. 如果 atuned 没有启动，启动该服务，参考命令如下:

```
# systemctl start atuned
```

2. 分别在 atuned 和 atune-adm 的服务器上执行如下命令，允许侦听端口接收网络包，其中 60001 为 atuned 的侦听端口号。

```
# iptables -I INPUT -p tcp --dport 60001 -j ACCEPT  
# iptables -I INPUT -p tcp --sport 60001 -j ACCEPT
```

3. 不影响业务的前提下删除 http 代理，或对侦听 IP 不进行 http 代理，命令如下:

```
# no_proxy=$no_proxy, 侦听地址
```

**问题 3: atuned 服务无法启动，提示 “Job for atuned.service failed because a timeout was exceeded.”。**

原因: hosts 文件中缺少 localhost 配置

解决方法: 在/etc/hosts 文件中 127.0.0.1 这一行添加上 localhost

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
```

# 5 附录

5.1 术语和缩略语

## 5.1 术语和缩略语

表5-1 术语表

术语	含义
workload_type	负载类型，用于标记具有相同特征的一类业务
profile	优化项集合，最佳的参数配置