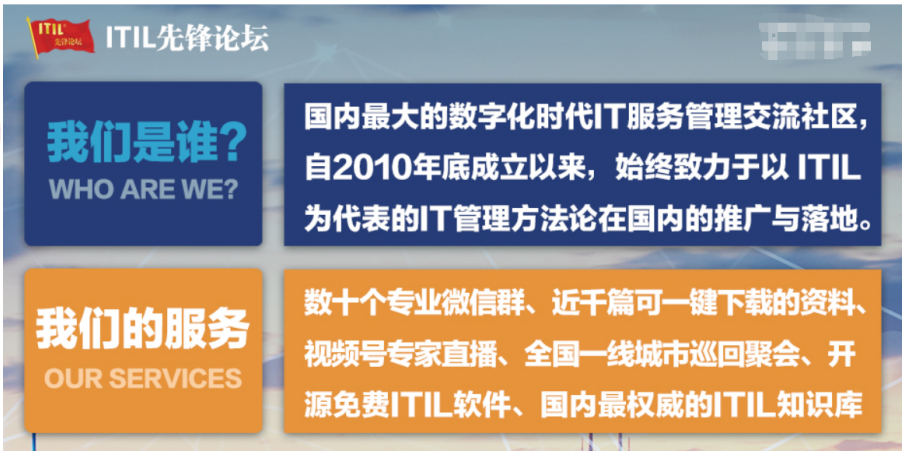
行业运维标准白皮书

行业应用日志运维参考规范



HUAWEI

目 录

1 日志概述 3

2 范围定义 4

3 日志框架参考 5

4 日志参考规范 6

4.1 日志分类 6

4.2 日志级别 6

4.3 典型字段 7

4.4 日志生成 8

4.4.1 用户日志 8

4.4.2 运行日志 8

4.4.3 安全日志 9

4.4.4 调试日志 9

4.5 日志内容 10

4.6 日志存储/转储 10

4.7 日志超限处理 11

4.8 海量日志抑制 11

5 日志上报 12

6 总结 13

附录 A：Syslog 协议标准 14

协议简介 14

日志级别 14

报文格式 14

搭建 Syslog 服务 14

附录 B：Logback 配置参考 15

1 日志概述

日志是应用最重要的基础运维数据之一，应用运行过程中的日志数据可以用于监控，故障分析，问题回溯等日常运维活动，也可用于挖掘、聚类等智能分析，有效支撑亚健康检测，故障预测等主动预防活动。

日志的特点：信息量大，基于时间序列生成，纯文本格式，生成后不再修改。

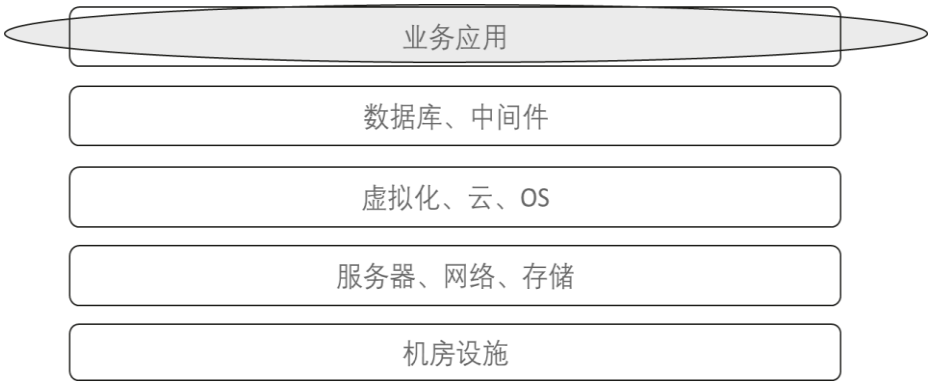
本文结合华为实践经验给出行业应用日志的参考规范。

2 范围定义

行业业务应用类场景多、厂商多，并且各应用对日志的实现差异较大。本文所述日志参考规范的使用范围为行业业务应用。

而其他各层的软件、硬件通常会有相对应的标准来规范化日志生成、记录、存储等活动，例如：rfc3164 或 rfc5424。

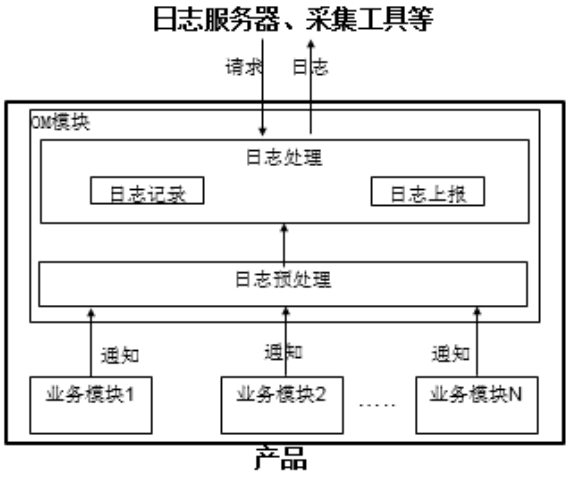
图2-1 范围定义



3 日志框架参考

参考 ISO/IEC ITU－T X.735 中定义的通用日志模型，一般日志处理框架如下图所示：

图3-1 日志处理框架



上图中：

1）业务模块：日志产生的源，各个业务模块产生日志原始信息并发送给 OM 日志预处理模块。

2）日志预处理：接收处理所有业务模块的日志通知（含大量日志消息时的缓冲、流控等处理），形成日志消息发送给日志处理模块，完成所有业务模块日志的集中存储。

3）日志处理：根据配置的日志格式，在文本文件、数据库中记录各类的日志信息，实现业务模块日志的集中存储；根据设置的过滤条件，日志上报模块将特定日志通过相关协议发送给日志服务器、采集工具等。

在行业应用开发中可以使用日志标准组件加快开发进度，如 Syslog 组件，Log4j，Log4j2，Logback 等。

4 日志参考规范

4.1 日志分类

日志范围比较广，凡是管理对象的相应事件和异常活动都可用日志形式记录下来。因此一般认为日志具有记录用户活动、管理系统安全的功能，同时也能为系统进行故障诊断和维护提供依据。

规则：日志按照使用基本场景划分为用户日志、运行日志、安全日志、调试日志等类别。

表1 日志分类

类别	定义
用户日志	记录实际用户的关键访问信息，例如登录、退出，关键操作，关键配置更改等。
运行日志	记录系统的运行状况或执行过程中的一些关键信息。
安全日志	记录系统管理用户登录、注销和鉴权，增加、删除用户，用户的锁定和解锁，角色权限变更等活动。
调试日志	记录调试级别的相关信息，一般用于跟踪运行路径，如记录函数的调用和返回值，大部分为代码级的信息输出，调试日志用于产品研发人员定位复杂的问题。

4.2 日志级别

日志级别用来表达日志所反映的事件的严重程度。级别的的分类方法较多，通常分为五级，可满足业务需要，也方便理解和记忆。

规则：日志级别分为 Critical（紧急）、Error（错误）、Warning（警告）、Informational（信息）、Debug（调试）五个级别。

说明：

1、 日志相关级别的说明如下：

表2 日志级别说明

级别	简写	含义

Critical	CRITICAL	导致系统业务严重受损或者完全不可用的紧急情况，规模性的用户受影响，需要运维人员紧急处理。
Error	ERROR	导致系统运行环境受损甚至丧失部分功能，或非预期的数据/事件影响模块功能实现的错误，需要运维人员关注和处理。
Warning	WARN	系统和预期的状态不一致，但是不影响整个系统的运行。
Informational	INFO	用于系统运行正常的信息记录，输出一些状态或状态变化的信息，例如当前系统的状态、数据库的连接状态等信息。
Debug	DEBUG	调试，用于跟踪运行路径，如跟踪函数的进入和退出等，记录调试信息。记载的信息比较全面，是给开发人员用于定位复杂的问题。

2、上述级别与 syslog（附录 A）中级别中的映射关系如下表：

表3 与syslog日志级别的映射

级别	对应 syslog 中的日志级别
Critical	Emergency、Alert、Critical
Error	Error
Warning	Warning
Informational	Notice、Informational
Debug	Debug

4.3 典型字段

日志通常包括如下典型字段。

字段	必选	说明
时间戳	必选	包括年月日时分秒信息。
主机名	可选	记录日志的主机名称或者主机 IP 地址。
进程名	可选	记录日志的进程名称。
模块名	可选	记录日志的软件模块对象。
级别	必选	日志的级别：CRITICAL,ERROR,WARN,INFO,DEBUG。

内容描述	必选	内容的详细描述。
处理建议	可选	Critical 级别优先给出处理建议。

4.4 日志生成

日志生成是指应用在其本地的日志文件增加日志记录的功能，也可记录在日志数据库或者同步发送给远端日志服务器。以下对各类日志时机、格式进行规定。

规则：同一个应用的不同模块要采用相同的日志格式。

建议：对于以文本文件记录的日志，为了便于字段内容的理解和解析，关键字段内容可采用固定分隔符进行分隔，如使用“|” “[ ]”作为分隔符。

规则：日志记录中的时间应包括年月日时分秒，优先使用本地时间，根据需要可使用 UTC 时间。

时间样例：2019-07-08 21:06:09

4.4.1 用户日志

规则：对用户日志的内容要求包括但不限于如下信息：

查询类操作，关键信息查询，查询结果是否成功；

设置类操作，需要包括设置对象标识、名称、相关属性名称和属性新值和旧值；

创建类操作，需要包括创建涉及对象标识、名称；

删除类操作，需要包括删除涉及对象标识、名称；

用户日志通常较多，是否记录日志可以结合实际情况考虑。

4.4.2 运行日志

规则：系统运行过程中的异常状态、异常动作、系统运行过程中的关键事件和系统资源占用的相关信息等需要记录运行日志。

说明：

运行日志记录创建的场景包括但不限于以下列出的几种情形：

- 记录系统运行过程中的异常状态和异常动作，如存储过程执行失败，系统死循环；

- 记录系统运行过程中的重要的状态变化，如服务状态变化；

- 记录系统进程运行过程中的关键事件，如系统启动、系统关闭、中断处理、任务切换、系统重启、进程吊死；

- 记录系统关键线程的运行情况，如线程启动、线程挂起、线程恢复、线程退出，线程吊死；

- 记录系统软硬件资源的相关信息，如服务器/CPU/内存/硬盘等器件的异常和恢复信息、资源破坏信息、资源操作错误信息、数据库表锁定等；

- 记录重要事件的处理轨迹；

- 记录交互中断、恢复信息，如业务模块连接数据库失败、模块间通信中断；

- License 加载成功/失败，配置文件加载成功/失败；

- 服务的加载、卸载、激活、去激活；

- 定时任务执行失败；

- 资源创建、存取、释放、大小改变、并发处理（如临界点）等；

- 业务相关资源统计处等；

- 跨模块的服务调用；

- 关键业务指标（如访问人数、访问成功率、访问时长等）可以通过周期性记录日志的方式保存为日志。

4.4.3 安全日志

规则：对于系统运行过程出现的登录认证、帐户管理等安全事件需要记录安全日志。

说明：

安全日志记录创建的场景包括以下列出的几种情形：

- 管理员登录：系统用户（指系统管理员、操作维护员、系统监控员等）登录、注销和鉴权等；

- 账户管理：增加、删除用户和用户属性（帐号、口令、权限范围等）的变更、用户的锁定和解锁，禁用和恢复等；

- 用户登陆鉴权：包括登录结果，失败原因，失败次数；需要包括鉴权用户名，失败的原因等信息；授权操作，需要包括被授权对象名，授权内容等。

4.4.4 调试日志

规则：以下场景（包括但不限于这些场景）可记录为调试级别的日志：

- 接口调用、函数调用等所有调用的入口、出口处；

- 任务/线程创建、启动、中止、暂停、恢复等处；

- 数据库连接建立、连接释放、SQL 提交、结果返回处；

- 文件的创建、读写、备份、转移、删除、重命名、获取文件属性等处；

- 操作入口处和设置预置条件处；
- 通信连接（如 Socket 连接）建立、断开处；
- 定时器启动、超时处；
- 状态设置、状态迁移条件判断前后处；
- 消息创建、编解码、收发处；
- 资源创建、存取、释放、大小改变、内容改变、并发处理处；
- 业务相关资源统计处、业务处理出入口、性能计算统计处；
- 处理失败、返回值异常等。

4.5 日志内容

日志内容是指日志输出的信息，以下为对日志描述内容的相关规定要求。

建议：除数据信息外，日志内容优先使用英文描述。

建议：单条日志的内容长度不要超过 1024 字节。

规则：日志内容不能存在语法、拼写错误，日志表达应完整准确，言简意赅。

规则：日志内容中涉及到度量信息需要标识相应的度量单位。

规则：非调试类日志内容中不出现魔鬼数字，任何带有程序逻辑意义的数值需被解释为有意义的可读字符串后再进行输出。

规则：敏感信息（如银行账号、密码、秘钥等）不能以明文形式记录在日志文件中，如需要输出则采用星号(*)形式。

建议：不要在非调试类日志中出现编码中的对象结构、方法接口名称等涉及代码内部实现的细节信息。

建议：不要在非调试类日志中输出 Java 异常堆栈。

规则：不要在非调试类日志中输出消息原始码流。

4.6 日志存储/转储

日志的存储主要描述日志记录在设备磁盘或数据库中的存储，转储是指将日志压缩存储到指定位置，同时删除这部分日志以释放存储空间用于记录新的日志。

规则：单一应用不同模块的日志，统一保存到一个日志文件，方便进行跨模块的问题分析。

建议：日志量较大时可以把日志文件分别记录为调试日志和非调试日志。

说明：调试日志内容较多，容易淹没非调试日志。

规则：当正在记录的日志文件大小超过设定门限，或文件记录时长达到设定的文件更替周期时，需新生成一个日志文件进行记录。

规则：单个日志文件大小不应超过 100M。

说明：日志文件太大，可能导致日志文件打开时间过长，影响整个故障处理时长，因此对日志文件大小进行了限制。

规则：日志分区空间全部被占用时，不能影响系统的正常运行。

说明：例如系统根分区空间占满时会影响系统运行，因此日志文件不能记录在系统根分区。

建议：对于记录在文件中的日志，支持文件压缩，以减少占用磁盘空间。

建议：日志建议使用文本文件进行存储，便于日志的解析。

说明：避免使用二进制文件保存日志，后端工具难以进行解析。

4.7 日志超限处理

日志超限，即日志记录占用空间已达到日志的最大容量，剩余存储空间将不能再存放一个日志文件的大小（100M），对于日志数据库记录指剩余存储空间将不能再存放一个日志记录。

规则：在日志超限时，最早的日志记录被删除以释放出空间来存储新的日志记录。例如删除最早记录的日志文件，或者删除数据库中最早的日志记录。

4.8 海量日志抑制

除了在所需要的地方记录日志外，也需要关注避免海量日志的情况。短时间内打印大量的日志，为高速海量日志，长时间周期性的打印相同的日志，为低速海量日志。海量日志带来的典型问题有：

1、异常情况下（例如在队列满或者数据库操作出现异常的时候），日志重复打印产生的持续海量日志输出，导致真正有用的日志信息被淹没，大大降低了日志的有效性。

2、海量日志导致日志难以快速传输、分析，给事故快速定界/恢复造成困难。

3、海量日志输出给系统性能、磁盘 IO 等带来了较大的压力，甚至影响正常的应用运行。

导致日志泛滥的根源在于日志设计中没有考虑日志信息是否会大量冗余输出，在日志的设计中要包含对日志泛滥抑制的分析设计，是避免日志泛滥的根本方法。

5 日志上报

日志上报是指应用以消息接口、文件接口、流式接口等将日志上报到日志服务器、网管、采集分析工具等。支持日志文件的压缩上报。

同一系统建议提供统一集中的日志上报/传输接口，各模块通过调用统一接口进行记录、上报功能。首选实时上报接口：如消息接口 syslog 等，或流式接口 logstash、filebeat 等。

6 总结

行业应用软件的可维护性主要依赖日志来实现，日志可以帮助运维工程师快速发现问题和解决问题，提升运维效率和降低成本。日志也是企业软件能力成熟度的衡量标准之一，需要加以重视并在实践中持续改进，提升日志质量。

