描述UIT日志系统，并简要说明使用方法

潘荣涛

UIT日志系统和运行时环境

德赛西威

[ 运行时环境 3](#_Toc26266)

[ Log系统 4](#_Toc9284)

[ 结构 4](#_Toc27092)

[ 通道属性 4](#_Toc18538)

[ Console Channel 5](#_Toc24071)

[ File Channel 5](#_Toc2950)

[ API用例 5](#_Toc28852)

[ 配置文件 6](#_Toc3605)

[ 图示 6](#_Toc19813)

# 运行时环境

core/Runtime.h提供统一获取运行时环境的接口，包括家目录、库目录、配置目录、可执行文件目录、临时目录、共享目录、资源目录、数据目录等。进程不需要关心不同平台的运行时配置细节，直接使用api获取即可。

getUitHomeDirectory：获取uit的运行时家目录

getUitLibDirectory：获取uit的运行时库路径

getUitEtcDirectory：获取uit的运行时配置路径

getUitBinDirectory：获取uit的运行时可执行文件路径

getUitTempDirectory：获取uit的运行时共享目录

getUitResourceDirectory：获取uit的运行时资源路径

getUitDataDirectory：获取uit的运行时数据目录

# Log系统

## 结构

property

Log API

Console

Channel

File

Channel

Console

File

property

UIT Log系统目前支持两种通道方式：Console Channel和File Channel。每个使用UIT Log的进程都可以配置启用和关闭其中的一个或多个。也就是说，你可以同时使用Console Chanel和File Channel来输出日志到控制台和文件中，也可以选择关闭任何一个。

## 通道属性

日志通道有多种属性，用以控制各种输出参数，属性可以通过api或者配置文件进行设置。

### Console Channel

控制台通道拥有唯一的的level属性，level可以控制日志输入的条件等级。Level的等级分为debug, information, warning, error, fatal, none，等级依次由低到高。只要低于level的日志，都将被过滤。比如，设置level为warning的情况下，debug和information等级的log都将被过滤不予以输出。等级为none表示关闭该通道。

### File Channel

文件通道也同样拥有level属性，与控制台通道一致。额外的，文件通道还有用它特定的几种属性，下面是对这些属性的描述。[]表示可用的值。

"path" : ""，表示 log文件的输出位置。注意：不要把所有的进程的文件日志通道的path属性都设置为同一路径，否则会导致多进程读写同一log文件而挂死！因此，File Channel的path默认是空的，所以File Channel默认不开启，知道你给path属性设置了一个值。

"rotation" : [ "n", "n K", "n M", "24:00", "daily", "weekly", "monthly", "n hours", "n weeks", "n months", “never”]，表示日志循环方式，超过尺寸或满足一定时间条件将建立/覆盖（如果文件数达上限）新的log文件。其中，n表示整数。

"archive" : ["timestamp", "number"]，表示归档日志的目录名后缀。

"times" : ["local", "utc"] ，日期格式。

"compress" : ["true", "false"] ，是否自动压缩。

"purgeAge" : ["n minutes", "n hours", "n days", "n weeks", "n months"] ，文档期限差值，超过的文件将被删除 ，比如5 days，n表示整数。

"purgeCount" : "n"， 文件数量限制，超过后最早的文件将被删除。

## API用例

* configFromFile与UIT\_LOG\_USE\_SYSTEM\_CONFIG宏

Uit::Log::configFromFile(“”);

configFromFile可以让你的进程已配置文件的方式来设置通道的属性。然而，一般而言，你只需要在你的程序初始化时使用UIT\_LOG\_USE\_SYSTEM\_CONFIG宏即可，它表示使用统一的初始化配置文件（指向uit::Runtime::getUitEtcDirectory() + "uit.log.conf"）来设置日志系统的通道属性。

* static void setLevel(int channels, Log::Level level, bool perpetual = false);

setLevel可以控制通道的level属性，低于level的日志都将被过滤。比如，设置了level为warning，当使用debug和info来输出log时，都将被过滤。当perpetual为true时，这个设置将被保留到configFromFile指定的系统配置文件中，如果未指定过，这个参数会比忽略。

* setFileChannelProperty和getFileChannelProperty

设置和获取文件通道的属性，属性值请参考上述中的【通道属性】。

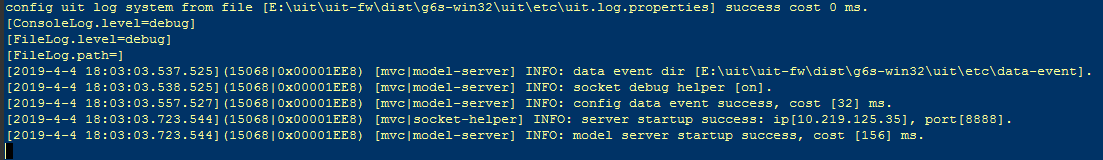
* debug, info, warn, error, fatal

打印日志信息，这些指令能否最终得到输出，收到level属性的影响。

## 配置文件

Log日志系统支持以配置文件的方式来进行初始化属性，默认的配置文件位于etc/uit.log.conf。你也可以使用configFromFile来指定自己的配置文件路径。然而，如无特别需要，仅使用UIT\_LOG\_USE\_SYSTEM\_CONFIG即可，表示使用系统默认的配置文件。配置文件中涉及到各通道的各个属性配置，详情请查看etc/uit.log.conf文件。

## 图示



如图所示，日志的格式为：

[日期时间](进程ID|线程ID) [主模块|子模块] Level: 消息体