



# 将 VTune (TM) Performance Analyzer 样本采集器应用于移动网络设备 (MID)

## 用户指南

应用于移动网络设备 (Mobile Internet Device, MID) 的 VTune™ Performance Analyzer 的样本数据采集器是一个独立的命令行工具，它可以在本地系统中提供基于事件采样 (event-based sampling, EBS) 的功能。这种基于硬件的采样是一种低开销、系统级的剖析，通过在操作系统和应用上逐项查看，它可以帮助您确定最消耗时间的模块和功能。本工具可以帮助您配置数据采集，进行系统级的剖析，并且将结果储存在文件中。

本手册的目的是对样本采集器的命令行选项做一个全面的介绍，指导使用者应用这些选项获取更多有用的内容。

## 目录

1	样本采集器命令行选项 .....	4
1.1	通用采集器选项 .....	4
	[-d   -duration <in seconds>] .....	4
	[-nb   -non-blocking] .....	4
	[-c   -count] .....	4
	[-ce   -count-emon] .....	4
	[-rc   -rerun-based-on-count] .....	5
	[-si   -sampling-interval <interval in milliseconds>] .....	5
	[-sb   -sample-buffer-size <size in kilobytes>] .....	6
	[-sd   -sampling-delay <delay in seconds>] .....	6
	[-msc   -max-samples-to-collect <maximum number of samples to collect>] .....	6
	[-sm   -sampling-method <ebs tbs>] .....	6
	[-sp   -start-paused] .....	6
	[-cm   -cpu-mask <"text to specify the cpu mask">] .....	7
	[-out   -output-file <file name>] .....	7
	[-of   -options-from-file <file name>] .....	7
	[-app <full-path-to-the-application>] .....	8
	[-em   -event-multiplexing] .....	8
	1.2 特定于事件的选项 .....	9
	[-ec   -event-config] .....	9



		[-dc   -data-config .....	9
		语法 10	
	1.2.1	通用事件修改器 .....	10
		:sa   sample-after = <sample after value> .....	10
	1.2.2	P6 处理器家族的事件修改器 .....	10
2		将 VTune 分析工具的样本采集器应用于移动网络设备(MID) .....	12
	2.1	启动数据收集 .....	12
	2.1.1	输出 .....	12
	2.2	关于期限的事项 .....	12
	2.2.1	暂停 .....	12
	2.2.2	延期开始.....	13
	2.2.3	零期限和多轮执行 .....	13
	2.3	关于数据采集的事项 .....	13
	2.3.1	通过样本采集器启动应用程序 .....	13
	2.3.2	计算评估后样本.....	14
	2.3.3	事件计数并基于计数重新执行 .....	14
	2.3.4	事件计数文件格式 .....	14
	2.3.5	固定的计数器支持 .....	15
	2.4	用法示例 .....	15
3		查看采集的数据 .....	17
	3.1	通过 vtl 查看抽样数据 .....	17
	3.2	通过 vtlec 查看抽样数据 .....	17



## 免责和法律信息

本文件中提供的信息专门针对 Intel 产品提供。本文件未以禁止反言或以其他方式授予任何知识产权的许可，无论是明示的还是暗示的。除非有有关该产品的其他 Intel 销售条款规定，否则 Intel 不承担由此导致的任何责任，也不承认任何有关该产品销售权与/或者产品使用权的明示或暗示的授权，其中包括以特殊目的、以营利为目的的授权，或者对专利权、版权、或其他知识产权的侵害。

除非另有书面协议，Intel 产品不得用于可能导致人身伤害或死亡发生的情况。

Intel 可能随时更改规范或产品说明，恕不另行通知。标有“保留”或“不确定”字样的任何功能或指令，其存在性与特性均不确定，设计人员切勿对此有所依赖。Intel 保留这些供将来定义之用，对于因将来对它们的更改而导致的任何冲突或不兼容现象，Intel 概不承担任何责任。此处提供的所有信息可能随时更改，恕不另行通知。切勿将此信息作为确定最终方案的依据。

本文所述软件可能包含某些设计缺陷或错误，一经发现将收入勘误表，因此可能导致产品与已出版的规格有所差异。目前已确定的软件缺陷勘误表可应客户要求提供。

订购 Intel 产品之前，请先与当地经销商或分销商联系以获取最新产品规格手册。

本文中提及的含有订单号的文档以及其他 Intel 文档可通过访问 Intel 网站获取，也可以致电 1-800-548-4725 索取。

Intel 处理器编号不作为衡量处理器性能的标准。处理器编号主要用于区分某一处理器家族内不同处理器的特性，不同处理器家族之间的处理器编号不具备可比性。欲了解更多信息，请访问[http://www.intel.com/products/processor\\_number](http://www.intel.com/products/processor_number)。

BunnyPeople、Celeron、Celeron Inside、Centrino、Centrino 徽标、Core Inside、FlashFile、i960、InstantIP、Intel、Intel 徽标、Intel386、Intel486、Intel740、IntelDX2、IntelDX4、IntelSX2、Intel Core、Intel Inside、Intel Inside 徽标、Intel Leap ahead、Intel Leap ahead 徽标、Intel NetBurst、Intel NetMerge、Intel NetStructure、Intel SingleDriver、Intel SpeedStep、Intel StrataFlash、Intel Viiv、Intel vPro、Intel XScale、IPLink、Itanium、Itanium Inside、MCS、MMX、Oplus、OverDrive、PDCharm、Pentium、Pentium Inside、skool、Sound Mark、The Journey Inside、VTune、Xeon 和 Xeon Inside 是 Intel 公司在美国和其他国家或地区的商标。

\*其他名称和商标属于各自所有者的资产。

Copyright © 2004–2007, Intel Corporation。

本产品包括 Apache Software Foundation (<http://www.apache.org>) 开发的软件。以下软件许可证适用于 Apache Software Foundation 开发的软件。

Copyright (c) 2000-2004, Apache Software Foundation。保留所有权利。

在满足下列条件的前提下，允许再发布并使用经过或未经过修改的、源代码或二进制形式的本软件：

源代码的再发布，必须保留上述的版权声明、本许可条件列表和下面的免责声明。

二进制形式的再发布，必须在随同提供的文档和其它文字材料中，复制上述版权声明、本许可条件列表和下面的免责声明。

如果再发布时随同提供最终用户文档，则该文档必须包括以下的答谢。

“本产品包括 Apache Software Foundation (<http://www.apache.org>) 开发的软件。”

答谢可能出现在软件本身中任何可能出现第三方答谢的地方。

未经提前书面许可，“Apache”和“Apache Software Foundation”字样均不得用于支持或宣传从既有软件派生的产品。如需获得书面许可，请联系[apache@apache.org](mailto:apache@apache.org)。

未经 Apache Software Foundation 的提前书面许可，从此软件派生出的其他产品不得被命名为“Apache”，产品名称中也不得出现“Apache”的字样。

本软件按“原样”提供，对于产品的使用(包括但不限于适合特定用途以及适销性)不做任何明确或暗示的担保。在任何情况下，由于使用此软件造成的直接、间接、连带、特别、惩戒或因此而造成的损害(包括但不限于获得替代品及服务，无法使用，丢失数据，盈利损失或业务中断)，无论此类损害是如何造成的，基于何种责任推断，是否属于合同范畴，严格赔偿责任或民事侵权行为(包括疏忽和其他原因)，即使预先被告知此类损害可能发生，Apache Software Foundation 和其贡献者均不承担任何责任。

根据由“自由软件”发表的 GNU 通用公共许可证和 GNU 次要公共许可证的条款，开源 GNU C++ 运行库的源文件可以在 <ftp://ftp.gnu.org/pub/gnu/gcc> 下载。

如需查看条款细则，请登录[www.gnu.org](http://www.gnu.org)或向 Free Software Foundation, Inc (59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.) 索取。

Intel(R) 希望以上提供的链接有效，但这些站点并非由 Intel(R) 控制，所以 Intel(R) 不会对以上站点中的内容以及以上站点中包含的其他链接承担任何责任。用户自行承担登录本文提供的任何第三方站点链接存在的风险。



# 1 样本采集器命令行选项

本章详细介绍应用于移动网络设备(MID)的 VTune 分析器样本采集器的命令行选项。

---

## 1.1 通用采集器选项

---

### [-d | -duration <in seconds>]

为样本采集指定一个期限。

---

使

用 **-d** 选项可为样本采集指定一个期限。默认值是 20 秒。要在未确定的期限内进行采集，请将期限设定为 0，直到用 **sep -stop** 命令明确终止采集为止。

---

### [-nb | -non-blocking]

切换到 *non-blocking* 模式

---

使用 **-nb** 选项将样本命令行采集器切换到 *non-blocking* 模式。采集器在后台开始运行。您可以在数据采集开始后恢复控制。当样本采集器在后台运行时，默认行为是 “**blocking**”。

---

### [-c | -count]

计算选择的事件数

---

使用 **-c** 选项计算选择的事件数。计数的结果可以在导出的文件中查看，该导出文件的扩展名为 **.txt**。以下是 **XXX.txt** 文件的格式：

event, cpu #, count, duration

此文件中每个事件或处理器都对应一个独立的行。

---

### [-ce | -count-emon]

以 *EMON* 风格计算选择的事件数。

---



## **[-ce | -count-emon]**

使用**-ce** 选项计算选择的事件数，并且以 **EMON** 风格导出，使用**-q** 或**-u**（二者等价）选项在 **EMON**（精简模式）中切换。使用**-out** 选项指定输出的文件。输出文件的格式是：

Event Timestamp	Count (cpu0)	Count (cpuN)
-----------------	--------------	--------------

此文件中每个事件都有它自己的行。在当前时间戳之后，每个处理器的计数都被打印出来。接下来的一行是每次运行时的合计/用户/系统/空闲的次数。所有的字段都以空格隔开。每次运行的事件以破折号被隔开。

您也可以使用此选项和**-si** 选项一起进行间隔计数。**-si** 选项后指定的数值就是间隔的长度（以毫秒计算）。当指定间隔计数时，按每个间隔输出的文件与前面设定的文件格式相同，只是每个处理器的计数值是自上次间隔后收集起来的数值（不累加）。每个间隔以等号隔开。计数期结束之后，累计的计数值在最后显示，用另外一个破折号与最后一个间隔的计数值隔开。

## **[-rc | -rerun-based-on-count]**

### *运行评估后样本计算*

通过使用**-rc** 选项设定样本采集器，然后生成事件计数信息，并且应用这些信息为每个事件计算一个合理的评估后样本（Sample After value, SAV）。

计算 SAV 之后，通过计算出的 SAV，样本采集器重新进行数据采集并收集样本。

## **[-si | -sampling-interval <interval in milliseconds>]**

### *指定样本之间的时间量*

使用**-si** 选项为采集的样本间指定一个时间量。在基于时间的采样（TBS）中，样本间的实际时间是由操作系统规定的。在基于事件的采样（EBS）中，您可以使用**-rerun-based-on-count** 选项来让样本采集器决定一个评估后样本，用于计算每个事件每秒所采集样本的平均数。默认值是 1 毫秒。

在 **EMON** 输出模式中，此选项也可以被用于间隔计数。有关更多信息，请参见**-ce |count-emon** 选项。



**注：** Linux\*平台不支持 TBS。

---

## **[-sb | -sample-buffer-size <size in kilobytes>]**

*指定储存样本数据的缓冲区容量*

---

使用**-sb** 选项指定缓冲区的大小，样本在保存到磁盘之前先临时储存到这里。默认值是 2000KB。

---

## **[-sd | -sampling-delay <delay in seconds>]**

*指定数据采集延迟*

---

在执行应用程序期间，使用**-sd** 选项来指定延迟采样的秒数。默认值是 0 秒。启动延迟是一段独立的时间间隔，而非采集期限的一部分。例如，如果您有一个期限为 60 秒的操作，而启动延迟为 10 秒，样本采集器将在 10 秒后开始采集样本，并且运行 60 秒，那么总共需要的时间为 70 秒。

---

## **[-msc | -max-samples-to-collect <maximum number of samples to collect>]**

*在停止数据采集前指定样本的总数。*

---

使用**-msc** 选项指定在停止数据采集前要采集样本的总数。默认是“无最大值”。

---

## **[-sm | -sampling-method <ebs|tbs>]**

*选择基于事件采样或基于时间采样*

---

使用**-sm** 选项选择是基于事件采样（EBS）还是基于时间采样（TBS）。默认是基于事件采样（EBS）。

**注：** Linux\*平台不支持 TBS。

---

## **[-sp | -start-paused]**

*在暂停模式下启动数据采集*

---

在暂停模式下使用**-sp** 选项启动数据采集。使用**-resume** 选项再次启动采集。



---

## **[`-cm` | `-cpu-mask` <"text to specify the cpu mask">]**

*指定从哪个处理器采集数据*

---

使用**`-cm`** 选项指定一个 **`cpu-mask`** 来定义您希望从中采集数据的处理器。输入处理器编号或处理器范围，以逗号隔开。例如：

```
-cm <2-5,10, 12-14>
```

在此例当中，采样的处理器为2、3、4、5、10、12、13、14

---

## **[`-out` | `-output-file` <file name>]**

*为输出文件指定文件名*

---

数据被采集到输出文件中，为输出文件指定基础文件名。样本采集器输出文件的扩展名要与数据被采集的方式（采样或计数）相一致。如果是采样则扩展名是**`.tb5`**，如果是计数则扩展名是**`.txt`**。

如果选项未定，则输出文件的文件名将被随机选择。除了上述情况之外，如果多项操作同时进行，则输出的样本文件的文件名需要添加能显示每项操作的序列号。

例如，文件名 **`foo`** 被保存为 **`foo_001.tb5`**, **`foo_002.tb5`**。

---

## **[`-of` | `-options-from-file` <file name>]**

*从一个文件中读取样本采集器选项*

---

使用**`-of`** 选项指定一个可以从中读取采样选项的文件。采集器从指定文件中读取选项并应用。

在该文件中指定的选项为命令行选项。这些选项可以被指定在同一行，也可以在多个行中。

例如，这是 **`my_clocks.txt`** 文件的内容：

```
-d 10  
-ec Clockticks:sa=1000000  
-out clock_out
```

您可以通过以下两种命令得到相同的结果：



将 VTune (TM) Performance Analyzer 的样本采集器应用于移动网络设备(MID)

```
sep -start -of my_clocks.txt
```

或者

```
sep -start -d 10 -ec Clockticks:sa=1000000 -out clock_out
```

**注：** 命令行选项优先于从文件中读取的选项。

---

## **[`-app <full-path-to-the-application>`]**

为数据采集启动的应用程序

---

指定通过样本采集器启动的应用程序。您需要指定该应用程序的完整路径。

如果这个样本任务配置为多轮执行或使用了 `-rerun...|-rc` 命令, 或者存在不能在单独执行的操作中同时收集的事件, 则这个应用程序将无法启动。

**注：** 此选项中不支持 `-d | -duration` 选项。样本数据采集将不定期地延续, 直到启动的应用程序中断或用 `sep -stop` 命令终止采集。

---

## **[`-em | -event-multiplexing`]**

多事件采样

---

通过 `-em` 选项支持多事件采样。多事件采样是在一个单独的取样操作中抽取多组事件的能力。

多事件采样在定时器模式下工作。当使用这种模式时, 样本采集器在明确的时间间隔下采集数据。您可以定义时间间隔 (`dts`) 和被计数的事件列表。如果未指定 `dts`, 默认值是 50 毫秒。

例如:

```
sep -start -em dts=100 -ec "Instructions Retired (Flex)", "Branch Instructions Executed", "Clockticks (Flex)", "L2 Cache Request Misses"
```

这个采样操作每 100 毫秒采集一次以下事件的样本: 指令执行的数目 (`Flex`)、执行的分枝指令、时钟 (`Flex`) 和 L2 缓存请求遗漏。





## 1.2 特定于事件的选项

本节描述了用于选择事件的选项。

---

### **[-ec | -event-config]**

*配置被采样的事件*

---

事件配置选项以 **-ec | -event-config** 转换器作为开始。指定监控的事件，并且在单括号中嵌入事件名。

**[modifier=val]** 选项使您能够为一个特定的平台指定不同的事件修改器和它们各自的值。修改器可以是一个事件通用的，也可以是特定于限定（或者一个事件限定词）的。特定于限制的修改器在 **[/constraint=]** 之后出现。修改器的值可以是 10 进制的也可以是 16 进制的。只有特定的修改器接受字符串形式的值。

每个事件规范都以逗号隔开。

---

### **[-dc | -data-config]**

*配置采集的数据*

---

**-dc | -data-config** 语句能够在 **-ec** 开关中指定。**-dc** 语句允许您指定一个额外的列表，列表中是每个样本采集的数据值。这类数据的一个例子是 “power-states”。指定 “**-dc power-states**” 选项使得系统中电源状态的数据采集能够支持电源状态。

如果 **-dc** 语句在转换器中被指定为一个事件名，只有在该事件取样时可选的数据值被采集。但是如果没有事件名被 **-dc** 语句指定，那么它适用于数据采集期间被抽样的所有事件。VTune 分析器样本采集器的当前版本只支持后一种使用模式。

**-dc** 语句并不影响在抽样采集期间默认采集的数据值。例如，IP 地址和时间戳的值在任何样本中均会被采集。



## 语法

```
[ -ec | -event-config [ -dc | -data-config <optional-data1>,<optional-data2>... ] ' <event name1> ':modifier1=val:modifier2=val/constraint1={:modifier3=val:modifier4=va l}, ' <event-name2> '...]
```

可选的数据值将在下面的章节中列出。

### 1.2.1 通用事件修改器

**:sa | sample-after = <sample after value>**

事件的评估后样本 (SAV) 显示了样本采集后事件的数目。参见“计算评估后样本”章节了解关于如何计算 SAV 的信息。以下为如何计算 SAV 的说明，之后是处理器的修改器。

下面的属性值可以为 10 进制或 16 进制格式。

### 1.2.2 P6 处理器家族的事件修改器

下表列出了适用于 P6 处理器家族的事件修改器，并对每个修改器进行了简短的描述。



表 1: 适用于 P6 处理器家族的事件修改器

修改器	描述
Event_Select=<value>	选择要监控的事件
UMASK=<value>	单元屏蔽，进一步限定选择的事件
USR=<yes/no>	只有当处理器在优先级为 1、2 或 3 时才计算事件数。此标记可用于与 OS 标记相关联。
OS=<yes/no>	只有当处理器在优先级为 0 时才计算事件数。此标记可用于与 OS 标记相关联。
E=<yes/no>	允许进行事件边缘检测
PC=<yes/no>	允许折叠 PM <sub>i</sub> #。
INT=<yes/no>	允许进行 APIC 中断处理
EN=<yes/no>	允许执行监控的计数器
INV=<yes/no>	设置时转换反码比较的结果，这样不管是多于还是少于都可以比较出来。
CMASK=<yes/no>	允许设定反码
PRECISE=<yes/no>	对于允许 PEBS 的事件启用 PEBS 特性。 <b>注：</b> PEBS 特性只有在运行于代号为 Silverthorne 的 Intel 处理器的 Intel®64 位体系结构上才能应用。

表 2: 适用于代号为 Silverthorne 的 Intel 处理器上的事件修改器

修改器	描述
ANYTHR=<yes/no>	任何处理器中的核心线程在 C0 电源状态，使计数能够继续进行。



## 2 将 VTune 分析器的样本采集器应用于移动网络设备(MID)

---

本章说明如何将命令行样本采集器应用于移动网络设备。

### 2.1 启动数据采集

抽样采集启动时（使用 **sep -start** 选项），该调用的默认行为是阻塞。这意味着只有在抽样采集结束后，才能重新控制。

#### 2.1.1 输出

一次成功的 EBS 执行将给出以下信息：

```
SEP is collecting samples based on the following events<事件名由逗号隔开>
```

一次不成功的 EBS 执行将给出以下信息：

```
SEP failed to start sampling collection due to one of the following  
reason(s) :<错误信息>
```

### 2.2 关于期限的事项

在以下几个小节中，将说明抽样活动期限内可能出现的几种事项。

#### 2.2.1 暂停

抽样操作暂停时（使用 **sep -pause** 选项），执行的期限不变。例如，如果一个期限为 60 秒的抽样操作启动，在大约 20 秒后暂停，那么这个抽样活动将仍然在 60 秒之后结束，但是只在暂停之前的前 20 秒内采集数据。



## 2.2.2 延期开始

采集延迟是一段独立的时间间隔，而并非采集期限的一部分。例如，如果您有一个期限为 60 秒的操作，而启动延迟为 10 秒，则样本采集器将在 10 秒后开始采集样本，并且运行 60 秒，那么总共需要的时间为 70 秒。

## 2.2.3 零期限和多轮执行

如果将期限设定为 0，这意味着样本采集器将不定期地继续进行抽样操作。在这种情况下，抽样活动不应该进行多轮执行。

样本采集器进行多轮执行是出于以下原因：

- 选定的事件无法在一次单独执行中被分组。这样就需要进行多轮执行，以便能够从所有事件中抽样。
- 选择了 `-rc | -rerun-based-on-count` 选项，至少需进行两次采集操作：一次是在采集运行期间获取事件的计数；另一次是为了实际的数据采集，使用第一次操作中的事件计数来帮助生成评估后样本。

因此，在多轮采集执行中，期限不得设定为 0。

## 2.3 关于数据采集的事项

### 2.3.1 通过样本采集器启动应用程序

样本采集器允许使用 `-app` 开关来启动一个应用程序。样本采集器启动工作量并且采集数据，直到该应用程序结束（或被强制终止），或者通过使用 `sep -stop` 命令将采集器停止时为止。

通常，一个活动可以通过以下方法之一（但不是全部）控制：

- 期限：样本采集器在指定的期限内运行。一个不同步的任务可以通过将期限设定为 0 的方法启动。
- 应用程序：通过样本采集器启动一个应用程序时，直到该应用程序启动后采集器才会运行。只有当应用程序终止或通过使用 `sep -stop` 命令将操作停止后，数据采集任务才会停止。

以下是关于这个使用模型的一些附加限制：

- 当样本采集器停止时，应用程序不会终止。通常，样本采集器只是启动应用程序但从强制终止它。



- 只有将采集器被配置为独立运行才能被指定一个应用程序。在多轮执行的情况下（或者由于 `-rc` | `-re-run-based-on-count` 选项，或者由于选定的事件不能在一次单独执行中分组），应用程序启动项是无效的。

## 2.3.2 计算评估后样本

VTune 分析工具样本采集器最初为默认的事件（时钟报时、CPU 循环和完成的指令）计算一个默认的评估后样本，具体如下：

默认事件的  $SAV = CPU \text{ 频率} \times \text{样本间隔}$ ，以毫秒为单位。

为任一事件计算评估后样本，需按以下操作进行：

1. 计算目标的（或预期的）样本数量：  
目标样本数量 = (抽样时间 / 抽样间隔) \* 处理器数量
2. 为一个单独的处理器的计算事件计数的平均数：  
平均事件计数 = 所有 CPU 的总事件计数 / CPU 数量
3. 最后，计算评估后样本 (SAV)：  
评估后样本 (SAV) = 平均事件计数（见操作 2）/ 目标样本数量（见操作 1）。

SAV 的最小值为 1。它的值不能为 0 也不能是负数。

## 2.3.3 事件计数并基于计数重新执行

`-count` 选项被指定时，一个扩展名为 `.txt` 的文件（`XXX.txt`）被创建，并且关于 `sep -start -count` 操作（可能为多轮执行）的所有计数都被添加到这个文件中。即使一个抽样活动生成多个 `tb5` 文件，每个抽样操作也总是生成唯一的 `XXX.txt` 文件。对于 `-count` 选项（没有 `[-rc | -rerun-based-on-count]` 选项）和一次单独运行，期限允许为 0，因为可以在 `sep -stop` 时写入 `XXX.txt` 输出文件。

由于这是一个常规的取样操作，0 期限/多轮执行限制也同样适用于 `-count` 选项。

## 2.3.4 事件计数文件格式

指定了 `-count` 选项时，样本采集器生成一个文件。这个文件包含了从一个采样任务的所有操作中选出的所有事件的计数信息。每个采样任务只生成唯一的文件。计数文件包含两种类型的输出内容：信息和数据。



- 每次信息输出总是以#sep 选项 开始：关键字作为第一个自变量 (\$1 Perl 符号)。信息输出的下一个关键字显示了信息的类型。
- 当前支持的信息输出类型有以下几种：
  - 版本——样本采集器的版本
  - 标题——事件计数数据的格式
  - 日期——任务开始的日期和时间（在v1中可能无法使用）
- 事件计数数据在信息输出之后。应该在信息输出中指定数据输出的格式（通常在“标题”关键字中）

### 2.3.5 固定的计数器支持

在代号为 Silverthorne 的 Intel 处理器上，3 个固定的计数器分别用于以下事件的计数：CPU\_CLK\_UNHALTED.CORE（时钟报时）、INST\_RETIRED.ANY（已完成的指令）和 CPU\_CLK\_UNHALTED.REF（循环引用）。当其中一个或多个事件在事件配置中被指定时，相应的固定计数器将投入使用。此外 2 个通用计数器仍然在其他 PMU 事件计数时使用。因此，您可以在一次单独执行时最多进行 5 个事件的计数：3 个固定事件和 2 个通用事件。

在下面的例子中，样本采集器在一次单独执行中进行了全部 5 个事件的计数：

```
sep -start -d 20 -ec "CPU_CLK_UNHALTED.CORE", "INST_RETIRED.ANY",  
"CPU_CLK_UNHALTED.REF", "DTLB_MISSES.ANY", "MEM_LOAD_RETIRED.L2_MISS" -out myData
```

## 2.4 用法示例

本节中提供了几个样本命令行的例子，以演示样本采集器的用法。

表 3： 命令用法

操作	命令行
启动数据采样	sep -start <"collector options">
暂停数据采样	sep [-pause]
重新启动一个暂停的数据采样会话	sep [-resume]
停止数据采样	sep [-stop]
获得平台支持的事件列表	sep [-el   -event-list]



将 VTune (TM) Performance Analyzer 的样本采集器应用于移动网络设备(MID)

获得帮助信息	<code>sep [-help   /?]</code>
获得版本或建立信息	<code>sep [-version]</code>





## 3 查看采集的数据

如需在 MID Linux 平台上查看采集的数据，则需要另外一个由 VTune analyzer 9.0 Update 6 支持的非 MID Linux 平台(参见产品发布手册中的支持平台列表)。在非 MID Linux 平台上安装 VTune analyzer 9.0 Update 6，并从 VTune analyzer command-line tool (vtl)或 Eclipse\* GUI tool (vtlec)中选择查看器。

在 MID Linux 机器上，使用命令行样本采集器来采集数据。例如：

```
sep -start -d 10 -out myData
```

这将生成一个样本数据文件 myData.tb5。下一步是将这个 tb5 文件复制到 non-MID 机器上进行导入。

### 3.1 通过 vtl 查看抽样数据

为了通过 vtl 命令行工具导入并查看数据，需要将 tb5 文件复制到 non-MID Linux 机器并使用以下命令：

vtl import myData.tb5	# 将 tb5 文件导入一个新的活动中
vtl show	# 显示导入的活动列表
vtl view -ar f1 -processes	# 打开进程，在文本模式下查看 f1 活动
vtl view -ar f1 -gui	# 在 GUI 模式下显示 f1 活动的结果
man vtl	# 打开手册页查看 vtl

### 3.2 通过 vtlec 查看抽样数据

如需通过 vtlec GUI 工具导入并查看数据，按以下步骤操作：



1. 将tb5文件复制到 non-MID Linux机器上。
2. 启动vtlec。
3. 进入**File->Open File...**并指定要导入的tb5文件。  
在 Eclipse 工作区的 **Tuning Browser** 窗口中创建了一个新的项目图标。
4. 双击导入的抽样结果图标查看tb5数据。

**注:** 使用 **vtl** 或 **vtlec** 向下钻取到热点或源文件时，您可能需要将其他一些文件（如二进制应用程序、库和源代码）复制到非 MID 机器上。