

编者说明：

软件过程管理中的一个很重要的工作就是制定项目、组织的过程规范，它是软件开发组织行动的准则与指南。该文档就是一个实际的过程规范的实例，通过该实例，相信大家根据自身情况制定符合要求的项目过程规范、组织过程规范有很好的借鉴作用。

# 1.总则

最大限度提高Q&P（质量与生产率），提高Q&P的可预见性，是每一个软件开发机构的最大目标。而Q&P依赖于三个因素：过程、人和技术，因此要实现Q&P的提高，除了加强技术能力，引进、培育更多优质技术人才之外，规范、改进机构的过程是一个十分重要的手段。我们希望通过在制定软件过程规范标准，并在软件开发实践中不断地完善、修订，提高Q&P和Q&P的可预见性。

本规范采用CMM（软件过程成熟度模型）的指导，吸收RUP、XP、MSF、PSP、TSP等过程规范指南的思想、方法及实践，充分结合xxx技术开发部的实际情况，引入先进的技术、方法、工具，为公司的软件开发工作提供一部详细、可操作的过程指南。在本规范的第一版本中，主要包括管理过程和开发过程两个部分，管理过程中包括项目管理过程、需求变更管理过程、配置管理过程。对于软件开发项目中的其它的一些过程将在实践中逐步补充、完善。

## 2.项目管理过程规范

项目管理过程主要包括三个阶段：项目立项与计划、项目实施、项目关闭。

### 2.1 项目立项与计划

参与人员：技术开发部指定的项目负责人（包括前期负责人、正式的**项目经理**）、立项申请人、[相关最终客户]以及实施该项目的开发组队成员；

入口准则：接到经公司总经理或副总经理批准的市场部门的《软件开发立项申请表》；

出口准则：立项申请人签字确认了经修订正后的正式《软件项目计

划》，并通过《工作任务卡》下达了开发任务，开发工作正式开始；  
输入：经审批的《软件开发立项申请表》、与需求相关的业务资料；  
输出：《软件项目计划》、《软件需求规格说明书》、《开发任务卡》；

活动：

接到《软件开发立项申请表》后，技术开发部经理指定前期负责人，并告知立项申请人；

前期负责人阅读《软件开发立项申请表》后，通过与立项申请人的沟通、阅读立项申请人提交的材料、通过立项申请人与客户直接交流等方式，了解项目目标、范围与基本需求；并形成最初的《软件需求规格说明书》；

前期负责人会同技术开发部经理以及其它相关人员，制定最初的《软件项目计划》，并组织评审；

向立项申请人提交最初的《软件项目计划》；

最初的《软件项目计划》通过立项申请人的确认后，项目经理计划安排需求分析；

需求分析完成后，形成正式的《软件需求说明书》，提交立项申请人确认；（需求分析过程参见开发过程规范部分）

根据立项申请人确认后的《软件需求说明书》，项目经理组织进行软件高层设计，并对工作任务进行分解，并根据实际需要向技术开发部经理申请资源，组建项目组队；

项目经理根据工作任务分解，下发《工作任务卡》，并协同组队成员进行任务估算；

注：工作任务包括模块开发任务、其它任务（如安装）；模块开发任务主要包括：详细设计、编码和单元测试

任务估算完成后，组队成员向项目经理提交《个人进度安排》（以甘特图的形式表示），项目经理根据每个组队成员的《个人进度安排》修订《软件项目计划》（必须包括总的计划甘特图），并提交立项申请人确认；

立项申请人确定后，项目经理根据软件项目计划基线，补充《工作任务卡》，下发到每个组队成员，开发工作开始。

项目立项与计划过程的工作流程如下图所示：

图表 1 项目立项与计划工作流程图

相关模板：

《软件需求规格说明书》、《软件项目计划》、《工作任务卡》

说明：如果计划确认、需求确认未通过，立项申请人与项目经理进行协商，进行修正，无法达成共识的，提交部门经理、总经理协调；

## 2.2 项目实施

参与人员：项目经理，项目组成员；

入口准则：项目计划基线已建立，并通过立项申请人确定，带有工作进度要求的《工作任务卡》已下发到每个项目成员；

出口准则：立项申请人在《验收报告》上签字确认；

输入：《软件需求规格说明书》、《软件项目计划》、《工作任务卡》；

输出：经验收测试的可交付的程序、源代码及相关文档。

活动：

在开发期间，项目成员每周需上交一份《时间日志》、《缺陷日志》，每天向项目经理汇报工作任务进度；

在开发期间，项目经理负责填写《项目进度周报》报于技术开发部经理、立项申请人（格式不同，交予立项申请人的只需周报的第一页，报予技术开发部经理的项目进度周报的第二页为“跟踪甘特图”）；

项目经理必须根据实际的进度情况，及时调整项目计划，若发现进度延误，需采取措施。

相关模板：

《软件项目计划》、《开发任务卡》、《时间日志》、《缺陷日志》、《项目进度周报》

## 2.3 项目关闭

参与人员：技术开发部经理或经理助理、项目经理，项目组成员、立项申请人、[相关客户、公司总经理、公司副总经理]；

入口准则：立项申请人在《验收报告》上确认；

出口准则：形成《项目总结》，完成项目绩效考核，项目数据存入“过程数据库”；

输入：《时间日志》、《缺陷日志》、《项目开发计划》；

输出：《项目总结》、已完成的《项目绩效考核表》、过程数据库中的该项目记录；

活动：

项目经理主持召开项目总结会，交流项目实施过程中的心得体会，对项目实施中的成功处、不足处进行总结，并由项目经理形成《项目总结》；

由技术开发部经理组织对该项目进行绩效考核，并填写相应的《项目绩

效考核表》；

**项目经理**组织所有成员对项目过程中的文档、源程序等资料进行整理、归档；

由**项目经理**根据过程数据库的需要，整理相应的数据，提交技术开发部经理，存入过程数据库。

相关模板：

《项目总结》、《项目绩效考核表》

## 3.开发过程规范

开发过程是提炼用户需求，设计、构建和测试满足这些需求的软件并最终将其交付给客户的过程。是软件过程中的主体过程之一。当开发新的应用或计划为现有的应用进行重要的增强时，需使用本规范所定义的开发过程执行。

项目管理过程是对开发过程进行计划、监控/管理、总结的辅助过程，但由于项目管理是保证进度、质量的重要手段，因此在软件项目中也是十分重要的过程之一。而需求管理过程与配置管理过程则是次重要的辅助过程，需求管理过程是一个需求变更管理的过程，以对变更进行统一的管理；配置管理过程的最重要工作就是版本控制，使得开发过程中的各种交付物能够有机地形成一个整体。

因此以上四个过程是交织进行的，均是为成功完成软件项目的保障过程。

### 3.1 过程总述

现在比较通行的开发过程模型包括：瀑布模型、演化模型、原型模型、螺旋模型等。根据公司的项目特点、队伍规模、组队情况等实际因素，决定选择最为简单、易于掌握的瀑布模型为基础，根据公司特点，进行合理的修改，使其成为公司本阶段的软件开发过程。

正如下图所示，本规范将整个开发过程分为：需求分析、高层设计、详细设计、编码和单元测试、集成计划与测试、系统测试、验收测试与安装、维护等八个阶段。

图表 2 开发过程总图

注：SRS：软件需求规格      HLD：高层设计      DD：详细设计

     SRC：代码      UT Plan：单元测试计划

注：“归档”在配置管理过程统一说明。

## 3.2 需求分析阶段

需求分析的主要目的是生成一个正确说明客户所有需求的文档。换言之，软件需求规格（Software Requirement Specification, SRS）文档是该阶段的主要输出。正确的需求分析和确定需求规格对一个项目的成功是非常关键的。许多在系统和验收测试时发现的缺陷是在需求阶段产生的。在验收阶段去掉需求阶段产生的一个错误将比在需求阶段本身去掉该错误要多花100多倍的费用。很明显，在执行这阶段时，正确地生成具有最少缺陷的SRS是非常必要的。

参与人员：**项目经理**，[分析员]，立项申请人，[客户，最终用户]；

入口准则：项目立项，最初的项目计划已得到立项申请人的确认。

注：这里所说明的需求分析阶段是进行开发过程的需求分析阶段，在技术开发部出具初步的项目计划之前的需求沟通工作，不是该过程规范所定义的。最初的需求沟通工作可以参考本过程规范。

出口准则：立项申请人、[客户]在《软件需求规格说明书》上签字确认；

输入：《项目立项申请表》、最初的《项目计划》，需求相关的资料；

输出：经确认的《软件需求规格说明书》；

活动：

整个需求分析过程主要包括以下几个步骤：

图表 3 需求分析阶段活动总图

首先，**项目经理**与分析员一块，做好需求分析的准备，包括阅读相关的背景资料，熟悉客户的实际情况，准备用户访谈计划，准备会谈问题清单等；

然后通过面谈、专题讨论会等形式与客户进行沟通，采集需求的详细内容，澄清每一个需求点；从而界定出系统的目标和范围；

对所采集和澄清的需求进行分析，构建需求模型，从功能性、非功能性两个方面进行需求分析，深入领会客户需求；

形成《软件需求规格说明书》，建立软件需求基线，并为软件需求评审做好准备；

由**项目经理**安排软件需求评审，协同立项申请人、[客户]进行需求评审；

立项申请人[或客户]在《软件需求规格说明书》上确认。

相关模板：

《软件需求规格说明书》



## 3.3 高层设计阶段

高层设计是软件开发过程中的一个重要阶段，在这个阶段将从计算机实现的逻辑角度开发针对用户需求的解决方案。这一解决方案是一个高级的抽象方案。高层设计要设计出各主要部分，并说明他们在技术上如何工作：1) 相互间的协作；2) 所需外在的硬件和软件环境；3) 内在环境。也就是说，高层设计确定了组成产品的构件，定义了每个构件的功能任务，并且定义了构件间的接口及构件到运行环境的外部接口。

参与人员：项目经理，项目组员（设计团队）；

入口准则：《软件需求规格说明书》已通过立项申请人的确认；

出口准则：形成高层设计，实现任务分解，所有的问题得到解决；

输入：《软件需求说明书》

输出：《高层设计说明书》（功能与数据库设计）、详细设计、编码、文档和用户接口标准；

活动：

制定详细设计、编码、文档和用户接口的标准；

根据项目特点选择运行的目标平台和开发工具；

制定软件的体系结构，定义逻辑和物理的对象模型，包括确定类、类的属性、类方法、类之间的关系和对象间的动态交互。若采用结构化设计，则该活动应为功能设计；

从需求规格说明书中的数据模型中得到物理数据库结构，进行物理数据库设计：包括确定表/记录类型、域和其他部分。

生成高层设计说明书，并组织设计评审。

相关模板：

《高层设计说明书》

## 3.4 详细设计阶段

在详细设计阶段，高层设计阶段开发出的整体应用被分成几个模块（或构件）和程序。为每个程序（或构件）进行逻辑设计，然后归档作为程序规格，同时为每个程序（或构件）生成一个单元测试计划。详细设计阶段的重要活动包括通用例程和程序的确定、框架程序的开发以及用于提高生产率的实用程序和工具的开发。

在详细设计阶段负责每个程序、模块（或构件）的内部设计，确定其程序流程，并且可以通过使用设计语言、图形流程图（如活动图、状态图）等，或通过简单地写叙述而将设计文档化。

参与人员：每个模块（或构件）的任务承担人；  
入口准则：《高层设计说明书》已通过评审；  
出口准则：完成详细设计，所有的问题得到解决，详细设计与单元测试计划文档化；  
输入：《软件需求规格说明书》、《高层设计说明书》、详细设计标准  
输出：《详细设计说明书》、《单元测试计划》  
活动：  
将高层设计中的每个程序（或构件）细分成小的组件；  
对每个小组件进行详细设计，包括确定调用方法、输入和输出、程序逻辑、数据结构等；  
根据组件的逻辑，制定单元测试计划，包括确定单元测试环境、测试用例、测试数据等；  
向**项目经理**（或高层设计者）提交详细设计与单元测试计划；  
相关模板：  
《详细设计说明书》、《单元测试计划》  
剪裁说明：对一些小项目，详细设计阶段的活动1、2可以省略。

## 3.5 编码和单元测试

在编码子阶段，根据详细设计用编程语言编写所需的程序。这个阶段根据合适的编码规范产生源代码、可执行代码以及数据库（如果使用了数据库）。这个阶段的输出是随后测试和验证的主体。而单元测试子阶段则是根据详细设计阶段所制定出来的单元测试计划进行测试，验证每一个组件正确、可用。

参与人员：每个模块（或构件）的任务承担人；  
入口准则：《详细设计说明书》已通过批准，编码规范已建立；  
出口准则：成功执行所有单元测试计划中的测试用例；  
输入：《软件需求规格说明书》、《高层设计说明书》、《详细设计说明书》、《单元测试计划》编码、用户接口标准；  
输出：测试数据、源代码、可执行代码、《单元测试报告》  
活动：  
根据详细设计，按照编码、用户接口规范编写程序；  
对程序进行代码复查、编译、调试，直到程序运行通过，符合详细设计的要求；  
根据单元测试计划进行单元测试，生成单元测试报告。  
相关模板：  
《单元测试报告》

## 3.6 集成计划与测试

集成是把设计阶段制定的，已通过单元测试的模块构建成一个完整软件结构的系统方法。可采用很多方式进行集成，集成计划必须指定模块集成的顺序。在该阶段，同时进行测试，以发现与接口相关的缺陷。集成按照集成计划中制定的顺序进行，并执行每个集成阶段的相应测试用例。集成计划描述了集成顺序、额外需要的软件、测试环境和资源需求。集成计划与集成测试计划通常一起完成。

参与人员：项目经理，集成团队；

入口准则：经批准的《高层设计说明书》；

出口准则：集成计划和集成测试计划经过评审和授权；

输入：《高层设计说明书》、源程序

输出：《集成计划》、《集成测试计划》

活动：

确定集成所需的环境，包括硬件的物理特性、通信和系统软件、使用模式等；

决定集成规程，确定将要集成的关键模块，集成的顺序，需要测试的接口等；

开发集成测试计划，确定测试用例和执行用例的规程，确定测试数据，确定期望输出等。

相关模板：

《集成计划》、《集成测试计划》

剪裁说明：对一些小项目，集成计划与测试阶段可以省略。

## 3.7 系统测试

系统测试是依据需求规格验证软件产品有效性的活动。这个阶段是为了发现那些只有通过测试整个系统才能暴露的缺陷。就像外部接口、性能、安全、配置敏感性、共存、恢复以及可靠性等属性只有在这个阶段才能判断其是否有效。可以使用具有不同测试目的的一系列测试来验证所有系统元素都已经正确地集成，系统能够执行所有功能并满足所有非功能需求。系统测试开始之前，必须在系统测试计划阶段详细地制定计划。

系统测试计划工作从需求分析结束后就可以开始，一直到编码时结束。

参与人员：项目经理，系统测试团队；

入口准则：经确认的《软件需求规格说明书》和经批准的《高层设计说



明书》；

出口准则：系统测试计划经过评审和授权，成功执行所有系统测试计划中的测试用例；；

输入：《软件需求规格说明书》、《高层设计说明书》

输出：《系统测试计划》、《系统测试报告》

活动：

决定所需的测试环境；

决定系统测试的规程，包括：确定测试特性，如用户接口、软硬件接口、通信接口、主要业务过程；确定不需要测试的重要特性以及不测试的原因；确定关键测试；

开发测试用例，包括确定每个测试用例以及执行它的规程，确定每个输入、输出数据的要求，确定预期的结果。

相关模板：

《系统测试计划》、《系统测试报告》

剪裁说明：对一些小项目，系统测试阶段可以省略，直接准备验收测试，在验收测试之前，开发组队按验收测试计划做一次没有立项申请人、[客户]参加的预测试。

## 3.8 验收测试与安装

验收测试和安装阶段的主要任务是将软件产品集成到它的操作环境中，并在这个环境中经受测试，以确保它按需求执行。这个阶段包括两个基本任务：使软件得以验收和客户处安装软件。验收指的是由立项申请人、[客户]根据早期准备的《验收报告》而进行正式的测试，并对测试结果进行分析，以确定系统是否满足验收准则。当分析结果满足验收测试时，用户接受软件。安装指的是把接受的软件置于实际产品环境中。

注：《验收报告》应附有验收测试计划

参与人员：项目经理，安装团队、立项申请人、[客户]；

入口准则：成功地完成了系统测试（或成功地完成了验收预测试）；

出口准则：立项申请人或客户在《验收报告》上签署确认意见；

输入：《软件需求说明书》、测试后的软件和《验收报告》

输出：签署了确认意见的《验收报告》和安装后的软件；

活动：

根据《软件需求说明书》，编写验收报告；

与立项申请人、[客户]一起按《验收报告》执行验收测试，包括：在验收环境下安装软件、进行实况运行、协助客户进行验收测试、改正验收缺陷、更新文档以反映所有变更、获得客户的验收确认；

执行安装，包括：在产品环境下安装软件、搭建产品环境、载入软件和数据、进行实况运行、修改安装缺陷、执行用户培训。

相关模板：

《验收报告》

## 3.9 维护

维护支持阶段是指已安装的应用得到支持，直至其在生产环境中稳定运行的阶段。

参与人员：项目经理，系统安装人员；

入口准则：软件在生产中运行；

出口准则：合同中指定的维护支持阶段终止；

输入：安装后的应用、用户文档和《软件维护申请表》；

# 4.需求变更管理过程规范

需求变更，这是个永恒的真理。需求变更的一个重要原因是系统周围的世界在变化，从而要求系统适应这个变化。在项目生命周期的任何时候或者项目结束之后都可以有需求变更。与其希望变更不会来临，不如希望初始的需求在某种程度上做得很好而使得没有变更需求，最好是项目准备时想到对付这些变更，以防变更真的到来。不管做多少准备和计划都不可能阻止变更，说项目在需求冻结后再开始不过是个神话罢了。

## 4.1 过程总述

需求变更管理过程定义了一系列活动，当有新的需求或对现有需求进行变更（我们可以称它们都是需求变更）时就会执行这些活动。需求变更可以在项目执行的任何一个点上发生。需求变更会影响项目进度，甚至会影响已经生产出来的产品。越是在生命周期后期的需求变更，对项目的影​​响越严重。不可控的需求变更导致对成本、进度以及项目质量的负面影响，这些极可能严重危害项目成功的概念。

需求变更管理过程用来控制需求变更并减少他们对项目的影响。这个目标需要理解需求变更请求的隐含意义，以及变更带来的总影响。同样，也需要立项申请人、[客户]意识到变更对项目影响的后果，使得可以友好地将变更反映到协商好的条款中。需求变更管理过程，从某种程序上说，试图保证在需求变更影响下项目依然可以成功。

需求变更管理有两个方面，一方面与立项申请人、[客户]就怎样处理变更达成一致，一方面是实际进行变更的过程。处理变更的整体方法必须与立项申请人、[客户]达成一致。一般来说，它制定怎样进行变更请求，当需要正式的批准时，为处理变更估计留出冗余空间等等。在整个方法的背景下，当需求变更到来时，需要执行需求变更管理过程。

## 4.2 过程规范

参与人员：项目经理，立项申请人、[客户]、开发团队；

注：项目经理对将变更纳入项目中所需的过程执行负主要责任。立项申请人、[客户]以及开发队伍也需要参与这个过程。

入口准则：收到立项申请人提交的《需求变更请求单》

出口准则：变更已列入新的《软件需求说明书》，并体现在新的《软件项目计划中》；

输入：《需求变更请求单》

输出：根据《需求变更请求单》，在充分协商与的基础上，提交新的《软件需求说明书》，并提交《软件项目计划变更表》；

活动：

记录需求变更请求，记录项中应包括变更请求数、变更的简要描述、变更的影响、变更请求的状态和关键数据；

分析变更请求对工作的影响；

估计变更请求需要的工作量；

修改项目计划，重新估计交付时间；

对总的成本花费的影响进行估计；

将修改过的项目计划提交立项申请人，并获得确认。

相关模板：

《项目计划变更表》

## 5. 配置管理过程规范

软件项目在其执行过程会产生大量的工件，包括各种文档、程序、数据和手册。所有这些工件都是易于改变的。这是软件一个独有的特点。正如“需求变更管理”章节中所述，在软件项目中，在项目执行过程中的任何时候，需求本身都会发生变更。为避免项目在变更时失控，正确控制和管理变更是很必要的。配置管理（Configuration Management，CM）又称为软件配置管理，是项目管理中专用于关注系统地控制项目进

行中发生的变更的那些部分，由用来识别机构软件产品并控制其修改的一系统活动构成。

配置管理需要满足项目基本目标之一：为客户提交高质量的软件产品。这个提交的产品，包括各种资源以及构成资源或目标代码的目标文件，还包括以这些文件来构建工作系统的脚本以及相关文档。在项目中，资源和文档通常以很多独立文件的方式来维护。

当项目进展时，文件发生了改变，产生了不同的版本。在这种情况下，即使将项目的各部分组合起来，构建成系统，也是很困难的任务，怎样保证合并的是源程序的正确版本以及没有遗漏任何源程序？还有，怎样保证传送的文档的版本是正确的，该版本和最终交付的软件是一致？对于这类型的情况，必须正确跟踪软件开发过程中的各种中间产品、其版本以及软件产品的版本。没有这些信息，交付最终系统就成为繁重的任务。这个活动不是由开发过程完成的，而需要一个独立的过程，那就是配置管理过程。

## 5.1 配置管理的目标

配置管理过程，需要达到以下目标：

能够随时给出程序的最新版本；

能够处理并发的文档、程序的更新/修改请求；

能够根据需要撤消程序的修改；

能够有效防止未授权的程序员对文档、程序进行变更或删除；

能够有效地显示变更的情况。

## 5.2 配置管理过程规范

配置管理过程包括两个主要阶段：配置管理计划、实施配置管理。

### 5.2.1 配置管理计划

参与人员：**项目经理**，配置管理团队；

入口准则：《软件需求规格说明书》已经确认；

出口准则：完成项目配置管理计划；

输入：《软件需求规格说明书》

输出：《配置管理计划》

活动：

识别配置项，配置项的典型例子包括需求规格、设计文档、源代码、测试计划、测试脚本、测试规程、测试数据、项目使用的编码、用户接口规范、验收报告等；

定义为配置项命名和编号的计划：如果使用CM工具，那么有时由工具处理版本编号，否则，在项目中必须明确地进行版本编号；

定义CM所需的目录结构；

定义访问控制；

定义变更控制规程；

确定CM工作人员的责任和权利；

定义跟踪配置项状态的方法；

定义备份制度

定义发布制度；

确定将配置项转移到基线的原则。

相关模板：

《软件配置管理计划》

## 5.2.2 实施配置管理

参与人员：项目经理，配置管理团队、开发项目组队成员；

入口准则：《软件配置管理计划》已批准，项目开始；

出口准则：项目结束；

输入：《软件配置管理计划》

活动：

接受变更请求；

Check out需要变更、修改的配置项，并进行修改；

Check in变更、修改过的配置项。

## 6. 附件

附件包括各种文档模板与工作指南。所有附件以单独的文档形式存储，文档名为xxxx模板、xxxx工作指南。具体包括：

### 6.1 文档模板

#### 6.1.1 项目管理类



《软件项目计划模板》、《工作任务卡模板》、《时间日志模板》、《缺陷日志模板》、《项目进度周报模板》、《项目总结模板》、《项目绩效考核表模板》、《项目计划变更表模板》、《软件配置管理计划》

## 6.1.2 开发过程类

《软件需求规格说明书模板》、《高层设计说明书模板》、《详细设计说明书模板》、《单元测试计划模板》、《单元测试报告模板》、《集成计划模板》、《集成测试计划模板》、《集成测试报告模板》、《系统测试计划模板》、《系统测试报告模板》、《验收测试报告模板》。

## 6.2 工作指南

《软件需求分析工作指南》、《软件项目计划工作指南》、《软件需求管理工作指南》、《软件配置管理工作指南》

网站定量评估的度量指标

度量指标	描述	获得方法	评估注意事项
点击数	访问服务器上某个文件的请求	日志文件	目前该指标的有效性大大降低，因为一个页面可能伪造成多个点击。
页面访问次数	访问服务器上一个HTML页面的请求	日志文件	注意主机、代理服务器和缓存可能会误报页面访问次数
唯一用户数	有不同IP地址或Cookie的用户	日志文件/数据库分析	动态分配IP会导致统一数大于实际数，而代理服务器会导致统一数小于实际数
用户会话数	与网站连接的时间超过30分钟而且没有中断的对话	日志文件	可以在网络服务器上设定超时时间，默认设置为30分中。然而通过修改这个设置，使采集的结果产生偏差
用户会话	平均的用户会话长度	日志文件	由于多个用户共享一台计算机或由于使用代理

			服务器上网，因此很难判断一个用户产生了会话，期间他离开了，而由另一个用户接替这个会话
访问网站的顶级路径	当用户访问网站时，大多数人访问页面的顺序	日志文件	如果使用框架的话，可能会到“无人访问的结果”
进入和离开页面	大多数用户进入或离开网站的页面	日志文件	有助于了解用户是否从外部的链接进入网站，或从一个标签页进入，而非从网站主要进入
与其它网站互相链接的点击数	常常用来测量广告链条的情况	日志文件(广告服务器)	用来评测一个广告条的吸引力，不能用来评测品牌的知名度或链接的效果
涉及网站	联系最紧密的网站	日志文件	用来了解网站最主要的流量从哪里来
繁忙时间段	网站网络服务器使用率最高的时刻	日志文件	用于计划网站升级和对网络进行维护、管理的依据
客户/服务错误类别	客户或服务产生错误的详细情况	日志文件	很多错误不一定是真正的错误
用户浏览器和操作系统	用户使用浏览器及操作系统的类型和版本	日志文件	用来了解设计的技术规格和实际情况间的差距，可以对未来的规划提供依据
用户地域分布	地域分布统计	日志文件	基于国家域名后缀的统计不是很可靠
网站响应时间	页面下载时间和数据搜索时间	性能监控软件	是一个很好的基准，网站随着流量的增加，可以了解其性能变化
服务器正常运行时间	网站可用时间的百分比	性能监控软件	应该尽量使这个指标接近100%，对于那些以广告为主要业务的网站更重要

注册用户数	带有用户个人详细信息的注册用户总数	数据库	要求用户使用时需注册和登录，操作更麻烦
个性化应用	专家报告功能	数据库分析	包括分析购买趋势、销售转化率、用户生活方式等。