**新域信息**

**项目管理标准化**

**开发项目管理规范**

［V1.0版］

项目部

2017年02月28日

**变更记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 作者 | 日期 | 变更索引 | 审核者 |
| V1.0 | 王辰 | 2017-02-28 | 创建开发项目管理规范 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**重要声明：**

***本文件是一份机密文件，任何阅读者均应承担保密义务，只在客户的雇员、有关专家、伙伴在获得授权对本文件进行评估时方可解密。本文件中的有关内容未经书面许可，本文件任何内容不得被复制或抄袭用于任何目的。请阅读者本着诚信的原则保证本文件的保密性，我们表示衷心感谢！***

目 录

[项目过程描述 4](#_Toc475456702)

[1. 开发项目过程图 4](#_Toc475456703)

[2. 每阶段详细文档见标准化文档，如下图： 5](#_Toc475456704)

[项目人员角色定义 5](#_Toc475456705)

[1. 项目经理（必选） 5](#_Toc475456706)

[2. 架构师（可选） 5](#_Toc475456707)

[3. 测试经理（可选） 5](#_Toc475456708)

[4. 项目成员（开发/测试/系统/UI工程师）（必选） 6](#_Toc475456709)

[5. 质量小组（可选） 6](#_Toc475456710)

[6. PMO（项目管理办公室）小组（可选） 6](#_Toc475456711)

[项目阶段划分 7](#_Toc475456712)

[1. 项目交底 7](#_Toc475456713)

[2. 需求分析 7](#_Toc475456714)

[3. 项目规划（项目启动） 9](#_Toc475456715)

[4. 项目设计 10](#_Toc475456716)

[5. 项目计划 13](#_Toc475456717)

[6. 项目开发 13](#_Toc475456718)

[7. 项目测试 15](#_Toc475456719)

[8. 上线部署及培训 16](#_Toc475456720)

[9. 项目验收 17](#_Toc475456721)

[10. 项目运维 17](#_Toc475456722)

[11. 项目管理 18](#_Toc475456723)

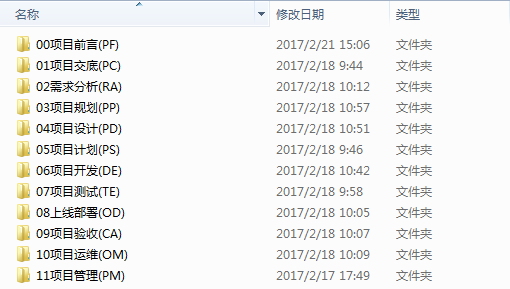
# 项目过程描述

## 开发项目过程图

项目管理总体上采用瀑布模型，但在需求分析和概要设计阶段借鉴螺旋模型，允许在项目过程中根据实际情况借鉴敏捷开发和极限编程的一些做法。如下图：



## 每阶段详细文档见标准化文档，如下图：

。

# 项目人员角色定义

## 项目经理（必选）

1. 负责项目交付工作，负责编写项目计划/项目周报等项目文档
2. 负责项目质量控制、风险控制、客户协调工作
3. 负责分配项目任务，执行项目/员工绩效考核工作

## 架构师（可选）

1. 负责项目技术管理工作，负责编写需求分析文档、详细设计文档等项目文档
2. 负责系统架构设计工作
3. 配合项目经理完成项目执行，配合测试经理完成系统测试

## 测试经理（可选）

1. 负责项目测试管理工作，制定测试计划、完成测试报告、审核测试用例等项目文档
2. 负责系统上线/版本发布工作
3. 建立测试环境
4. 配合项目经理完成项目执行，及时报告发现的错误
5. 对项目测试作项目总结，为日后的工作提供经验和教训
6. 分析错误的成因，估计错误的重要程度，评估系统质量，完成测试报告

## 项目成员（开发/测试/系统/UI工程师）（必选）

1. 严格执行项目经理工作任务安排
2. 参与项目需求分析
3. 参与项目工作量评估
4. 严格执行架构师方案设计
5. 开发工程师负责单元测试
6. 配合测试经理完成功能测试/压力测试/安全测试

## 质量小组（可选）

1. 审核概要设计
2. 审核详细设计
3. 审核项目代码，包括编码规范、安全性、可维护性、性能问题等。
4. 审核压力测试报告
5. 审核安全测试报告

## PMO（项目管理办公室）小组（可选）

1. 审核需求分析文档，发现并纠正不合理的需求。
2. 检查项目必备文档、里程碑、项目计划、项目周报等。
3. 审核压力测试报告
4. 审核安全测试报告
5. 审核系统上线计划、系统应急预案
6. 为项目经理的工作提供支持

注：PMO(Project Management Office)一般称为项目管理办公室

**特别说明：小的项目可能没有其他角色，但一定有项目经理和工程师，这两个必选角色可能是一人。**

# 项目阶段划分

## 项目交底

* 1. **说明**

销售部已签订合同的项目向项目部交接客户情况介绍、客户内部关系、客户口头承诺、客户前期需求点、客户对接人及客户提供的书面材料，并确定项目组成员及角色。

* 1. **交付物**

《项目交底备忘录》

## 需求分析

* 1. **说明**

业务分析阶段是一个项目的开始，它从业务确定立项开始，以完成需求分析并通过需求评审为结束标志。在这个阶段，项目组必须始终以用户的角度考虑问题，以用户的语言描述问题，不要过早的受到技术实现的限制，力争能够得到翔实、准确的业务执行情况。

**需求调研**

需求调研阶段并不需要考虑技术实现之类的后续问题，保持用户导向、加强用户的参与是至关重要的，对业务模式有一定理解后，再逐层细化具体的业务模块。需求调研阶段我们主要对客户进行引导型的提问，使我们尽量多的了解业务情况和客户期望，把客户所持的假设和含混的地方理解清楚。尽量理解用户用于表达他们需求的思维过程，研究用户执行任务时作出决策的过程，并提取出潜在的逻辑关系，来揭示用户真正关心的问题。

需求调研必须要给予足够的重视，同时也一定要认识到需求调研工作不是一次就能完成的，需要经过了解，总结，再了解的多次反复。

**需求分析**

需求分析阶段是在前一阶段需求调研的基础上，对调研所了解的业务情况和用户需求进行分析整理以及优化，从而明确项目所要解决的问题，并将分析整理优化的结果作为项目业务解决方案加以纪录。需求分析对于项目的成功具有至关重要的作用，经过需求分析，项目所要处理的业务，所面临的环境等都要被明确，没有经过一个有效的需求分析，你可能会做出一个优秀的技术方案却完全没有解决业务问题。

需求分析阶段和需求调研阶段一样，项目组必须从业务角度出发考虑问题，不要让技术方面的因素过早的约束了对业务的分析。更进一步的说：需求分析不是要得出项目技术解决方案，如功能模块、组件服务、用户界面定义等，需求分析是试图使项目组把握用户实际的业务处理过程和对系统的要求，达到清楚地了解业务进行的程度。

|  |  |
| --- | --- |
| **工作** | **任务** |
| 组建团队 | 确定核心团队 |
| 确定合作伙伴团队 |
| 确定客户参与配合的方式和人员 |
| 确定功能范围 | 了解客户业务环境和技术环境 |
| 确定客户对系统的要求 |
| 定义系统提供的功能特性 |
| 制定项目建设计划 | 确定项目开发计划 |
| 确定系统联调计划 |
| 确定工程实施计划 |
| 确定系统测试计划 |
| 确定项目沟通计划 |
| 确定风险管理计划 |
| 确定系统用户培训计划 |
| 确定项目变更控制计划 |
| 进度控制 | 明确各项任务的负责人和时间安排 |
| 风险管理 | 执行项目风险管理计划 |
| 沟通管理 | 执行沟通管理计划-和客户的沟通 |
| 执行沟通管理计划-和合作伙伴的沟通 |
| 执行沟通管理计划-项目组内部的沟通 |
| 阶段工作同步 | 文档整理提交 |

注意：在测试人员完全理解需求文档前不得进入开发阶段（可进入设计阶段）。

* 1. **交付物**

1. 《需求分析说明书》
2. 《需求变更申请》
   1. **角色职责**
3. 项目经理：协调召集需求调研、讨论，撰写需求分析文档。召开需求调研讨论会、需求分析文档审核会、需求分析确认会。
4. 架构师：参与需求调研、分析，撰写需求规格说明书。
5. 测试经理：参与需求调研、分析，编写测试计划。
   1. **会议**
6. 需求调研讨论会：项目经理组织，客户、架构师、测试经理参加，记录交流内容。
7. 需求分析文档审核会：项目经理组织，架构师、测试经理、项目销售、PMO参加，形成对需求分析文档的批注。
8. 需求分析确认会：项目经理组织，架构师、测试经理、项目销售、客户项目经理、客户其他人员参加，完成需求分析文档和需求规格说明书定稿。

## 项目规划（项目启动）

* 1. **说明**

项目规划是在项目组了解了项目的必要信息后，组织双方项目组成员及双方领导共同参加项目启动汇报大会，项目经理需汇报项目的蓝图设计、项目计划、及双方的权利和义务。

* 1. **交付物**

1. 《双方项目组成员名单》
2. 《客户项目组成员职责》
3. 《项目总体计划》
   1. **角色职责（详细职责见项目规划阶段文档）**
4. 项目经理：撰写项目启动会PPT、项目章程，编排项目计划，召开项目启动会。
5. 架构师：协助项目经理编排项目计划，细化任务项，评估工作量。
6. 团队成员：参加项目启动会，了解工作说明和项目计划。
   1. **会议**

项目启动会：项目经理、项目销售组织，架构师、PMO、客户项目经理、客户其他人员、双方高层领导参加。

项目启动会议的目标：

* 让整个项目组的成员相互认识
* 建立项目的工作关系和沟通关系
* 让大家明确团队的工作目标
* 让大家了解项目的当前状态
* 一起审阅项目计划
* 找出项目的难点或可能出问题的环节
* 分配小组和个人的角色与责任
* 获得小组和个人的承诺

## 项目设计

* 1. **说明**

将客户的需求通过文档或者原型设计的方式呈现出来，作为项目开发的依据。

* 1. **概要设计**

通过概要设计准则我们可以根据实际情况确定概要设计的具体内容，但是一般来说，概要设计的内容可以包含系统构架、模块划分、系统接口、数据设计4个主要方面的内容。

**1.系统构架**

系统构架包含系统使用的软硬件技术平台及相关技术列表、系统的物理架构、物理器件类型、数据库管理系统类型、服务器类型、子系统划分及部署方式、系统的软件架构、第三方软件平台列表等。

**2.模块划分**

模块划分的主要工作是结合业务领域以及系统的要求，通过拆分、合并、分组等方法，将各功能划分到子系统和功能模块中，并细化到各机能组件和功能模块上。模块划分可以通过2步来实现，层次及模块设计和流程设计。

第1步层次设计，对于复杂系统需要对系统进行多层的划分，然后一层层地设计。层次划分的方法可以结合业务、部署、运用功能等因素考虑，首先进行分类，然后进行抽象分析，整理共性要素，将功能归纳到模块。制作形式可以采用图或表。

第2步进行流程划分，进行模块间的流程设计。层次、模块划分完成后，需要对整理描述各模块间的关系，这些关系可以包括时序关系、状态关系、数据关系等等。在实际设计中，可以根据情况采用不同的描述方法来制作对应的设计文档，比如以时序为主的系统可以采用时序图。

**3.接口设计**

软件系统结合业务、功能、部署等等因素将软件系统逐步分解到模块，那么模块与模块间就必须根据各模块的功能定义对应的接口。概要中的接口设计主要用于子系统/模块之间或内部系统与外部系统进行各种交互。接口设计的内容应包含:功能描述、接口的输入输出定义、错误处理等。软件系统接口的种类以及规范很多，可以有API、服务接口、文件、数据库等等，所以设计的方法也有很大的差异。但是总体来说接口设计的内容包含：通讯方法、协议、接口调用方法、功能内容、输入出参数、错误/例外机制等。从成果上来看，接口一览（表）以及详细设计资料是必须的资料。

**4.对象设计**

在面向对象的设计中，应该是先建立对现实世界中事务的抽象模型（对象），然后再考虑这些对象是否需要持久化，以及如何持久化（数据库、文件等形式）。所以，与传统的软件开发规范不同，我们不要求在概要设计中完成数据库设计，但必须包含主要对象、接口的设计，包括主要对象的名称、属性、方法、继承关系、多态实现等。

* 1. **详细设计**

详细设计的是概要设计的延续，这个延续体现在设计粒度上。一般来说，详细设计根据业务要求以及概要设计的内容，确定细化各类组件的功能、逻辑流程、业务要求、组件接口。 从过程上基本遵循设计的4个主要活动，具体操作一般可以概括为模块组件分割、模块组件设计、设计共通功能3个步骤，其中组件分割是比较关键的。一般的信息系统，主要的组件分类可以包括画面、报表、服务、共同组件。

**1.画面设计**

画面设计是交互性设计的主要内容之一。常见的设计内容包括画面一览对应的画面迁移关系，详细的画面功能。画面迁移主要描述画面间的动作关系、定义所有画面的出入口等内容。

**2.服务设计**

服务组件对应软件系统主要通过软件接口实现功能的模块，详细设计中服务设计主要内容包括对应程序模块（package）设计、Class设计、处理逻辑（时序关系）、内外接口、限制条件等等。

**3.共同设计**

详细设计中关键的步骤就是进行共同设计，共同设计的多少往往极大的影响后续开发的工作量。比如一个系统抽取40%的共同和只抽取10%的共同，开发、测试工作量差异很大。建议具体项目中应尽早以及有效的管理共同设计。共同设计可以大体分为业务共同、技术共同。

* 业务共同是同业务相关的，可以根据业务的特征识别并抽象出不同的模型进行设计，不过一些常见共同比较容易识别，比如权限管理、基础数据管理、用户管理、安全管理等等。
* 技术共同一般是和计算机技术相关的，比如线程管理、画面各组件、LOG管理、内存管理、文件处理、文字处理、配置管理、加解密、日志、消息、异常、错误处理等。

|  |  |
| --- | --- |
| **工作** | **任务** |
| 设计系统技术实现方案 | 确定系统设计方案 |
| 确定系统测试方案 |
| 确定系统工程实施方案 |
| 确认相关系统接口实现方案 |
| 确定性能测试方案 |
| 验证技术关键点可行性 |
| 制定部署/割接方案 | 制定系统部署方案 |
| 制定系统割接方案 |
| 完成开发建设准备 | 准备开发环境 |
| 准备测试环境 |
| 制定开发规范 |
| 制定项目日常管理规范 |
| 组建开发测试实施队伍 |
| 进度控制 | 制定详细的开发/测试进度计划 |
| 制定详细的工程实施进度计划 |
| 协调确认相关系统接口实现的进度计划 |
| 制定系统联调进度计划 |
| 质量控制 | 进行项目检查 |
| 风险管理 | 执行项目风险管理计划 |
| 沟通管理 | 执行沟通管理计划-和客户的沟通 |
| 执行沟通管理计划-和合作伙伴的沟通 |
| 执行沟通管理计划-项目组内部的沟通 |
| 阶段工作同步 | 文档整理提交 |

* 1. **交付物**

1. 《设计原型》
2. 《概要设计说明书》
3. 《详细设计说明书》
4. 《数据字典》
   1. **角色职责**
5. 架构师：撰写概要设计、详细设计。召开概要设计讨论会、详细设计讨论会、详细设计确认会。
6. 测试经理：编排测试计划。
   1. **会议**
7. 概要设计讨论会：项目经理组织，架构师、测试经理、公司质量小组参加，形成对概要设计文档的批注。
8. 详细设计讨论会：项目经理组织，架构师、测试经理、开发人员、**公司质量小组**参加，形成对详细设计文档的批注。
9. 详细设计确认会：项目经理组织，架构师、测试经理、客户项目经理、客户架构师、客户其他人员参加，完成详细设计文档定稿。

## 项目计划

* 1. **说明**

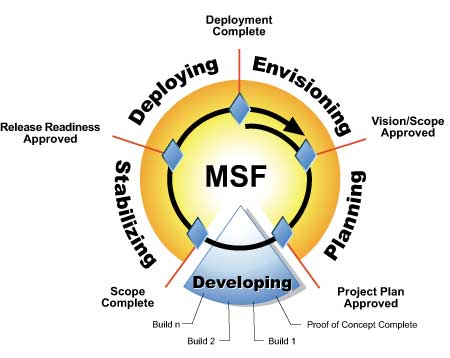
制定项目实施的进度计划，该计划确定后将直接导入到项目管理系统中。

* 1. **交付物**

1. 《项目开发计划》

## 项目开发

* 1. **说明**

系统开发阶段即程序编码和单元测试阶段。在这个阶段，项目组致力于将设计方案成为最终的可发布系统。

系统开发阶段是整个项目管理周期内最为琐碎的阶段。所有的业务和技术细节，不可预计因素对进度的影响，前期阶段没有发现的问题等等都使得必须对这个阶段进行大量的工作。这些工作除了保证代码开发的进度和质量外，还涉及到保障跟踪机制的运行，变动因素的管理等等日常项目管理工作。同时，在这一阶段，用户方将需要着手进行发布部署的准备，包括人员组织上的准备，物理资源的准备，以及必要的文档和培训工作的准备。

毫无疑问，代码编写是整个项目开发过程中工作强度最大的一个阶段，但如果在之前经过了良好的计划和严格的管理，代码编写应该是难度较小的一个阶段。在代码编写阶段，最重要的不是编写代码的难度，而是管理的难度。必须解决好在代码开发中出现的种种在计划阶段很难考虑到的因素，并尽一切可能保证编码工作的顺利进行。代码编写阶段的核心就是质量保证和变动控制。解决好了这两个因素，系统的按时按质发布就有了必要的保证。

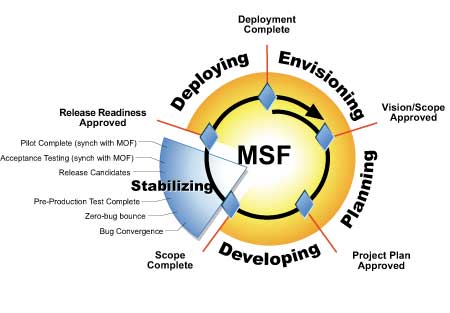
|  |  |
| --- | --- |
| **工作** | **任务** |
| 代码开发 | 标准化系统框架代码 |
| 标准化代码的配置管理 |
| 系统分阶段交付 |
| 标准化开发-测试流程 |
| 系统测试 | 业务模块测试 |
| 接口联调测试 |
| 系统性能测试 |
| 制定系统部署/割接方案 | 开发相关安装程序/脚本 |
| 开发数据割接程序/脚本 |
| 系统培训 | 准备用户培训工作 |
| 编写用户操作手册 |
| 进度控制 | 定期检查开发进度状况 |
| 定期检查工程实施进度状况 |
| 定期了解相关系统接口开发进度状况 |
| 质量控制 | 代码Bug跟踪 |
| 代码检查 |
| 风险管理 | 执行项目风险管理计划 |
| 沟通管理 | 执行沟通管理计划-和客户的沟通 |
| 执行沟通管理计划-和合作伙伴的沟通 |
| 执行沟通管理计划-项目组内部的沟通 |
| 阶段工作同步 | 文档整理提交 |

* 1. **交付物**

1. 《项目开发约定》
2. 《开发记录文档》
3. 《源代码》
   1. **角色职责**
4. 项目经理：分配具体开发任务工作，督促工作进度，管理项目风险、需求变更，每1~2个月组织一次代码审核会（如项目周期短于1个月，则必须在开发工作开始两周内组织代码审核会）。
5. 架构师：构建项目模块，定义关键接口、方法签名，关键技术难点攻关。
6. 开发工程师：撰写具体模块设计文档，完成具体开发及单元测试工作。
   1. **会议**
7. 代码审核会：项目经理组织，架构师、开发人员、公司质量小组参加，形成代码审核意见记录，公司质量小组负责确保所有问题按时修正。

## 项目测试

* 1. **说明**

由于我们的失误和对业务功能理解的欠缺，在开发过程中很可能引入各种各样的错误，测试是项目质量保证的关键元素，我们的目标是能够系统的揭示不同类型的错误，并且耗费最少的时间与工作量，可以把测试看作是摧毁性的而不是建设性的，毫无疑问开发人员最了解系统结构，开发人员也经常能够敏锐的发现一些问题，但是无论如何开发人员对系统的心理预期都是希望系统取得成功，与测试的目标不能达成一致。

我们进行了一系列技术层面的测试后进入集成测试，集成测试的重点在于验证系统功能是否满足实际的业务需要，其次还需要查找错误并验证系统与运行环境的适应情况。测试的主要手段是运用实际的业务流程和数据来测试系统是否满足需要，虽然测试并不能说明系统是完美的，但测试至少能够用一个量化的指标说明软件满足业务使用的情况，说明系统在实际业务环境下可以完成要求的功能。

系统通过测试并得到用户认可后我们并不能确认系统能够满足实际需要，通常我们先在小范围内投入使用，经过一段系统的并行，收集用户反馈，使系统问题及早暴露出来，防止直接投入使用造成大面积后果。试运行阶段的诸多工作需要事先准备，如建立符合要求的硬件和网络环境、用户培训、宣传推广等，试运行阶段技术不应该是主要的问题，主要调查用户使用系统的感受。

|  |  |
| --- | --- |
| **工作** | **任务** |
| 业务测试/业务试点 | 验证系统业务执行的正确性 |
| 制定系统部署/割接方案 | 准备系统安装配置工作 |
| 组建部署实施团队 |
| 确定系统部署/割接执行方案 |
| 建立系统运维实施组织 | 建立运维团队 |
| 建立运维规范 |
| 系统培训 | 组织用户培训 |
| 进度控制 | 检查系统环境的准备情况 |
| 质量控制 | 考察系统质量状况 |
| 风险管理 | 执行项目风险管理计划 |
| 沟通管理 | 执行沟通管理计划-和客户的沟通 |
| 执行沟通管理计划-和合作伙伴的沟通 |
| 执行沟通管理计划-项目组内部的沟通 |

* 1. **交付物**

1. 《功能测试报告》
   1. **角色职责**
2. 测试经理：分配具体测试工作，跟踪问题列表，提交相关文档，召开测试用例评审会、压力测试结果讨论会、安全测试结果讨论会，对项目交付质量负责。
3. 测试工程师：编写测试用例及相关文档，跟踪发现错误。
   1. **会议**
4. 测试用例评审会：测试经理组织，测试工程师、项目经理、架构师参加，形成《测试用例评审意见》。
5. 压力测试结果讨论会：测试经理组织，测试工程师、项目经理、架构师、客户项目经理、客户架构师参加，记录交流内容并报告PMO、公司质量小组。
6. 安全测试结果讨论会：测试经理组织，测试工程师、项目经理、架构师、客户项目经理、客户架构师参加，记录交流内容并报告PMO、公司质量小组。

## 上线部署及培训

* 1. **交付物**

1. 《系统部署手册》
2. 《环境准备检查表》
3. 《系统上线部署计划》
4. 《数据迁移计划》
5. 《操作手册》
6. 《培训计划》/《培训签到表》
   1. **角色职责**
7. 项目经理：制定部署计划与方案，提出部署请求，安排部署工作。
8. 架构师：保障部署工作顺利进行。
9. 部署工程师：部署应用程序，对系统进行初始化、数据导入。
10. 测试经理：保证应用程序符合预期质量，可以交付客户。测试经理有权拒绝不合格的版本发布。
11. 测试工程师：完成测试工作。
12. 项目经理：制定培训计划与方案，协调客户培训人员、时间、场地、器材等。
13. 讲师：准备培训资料，完成培训工作并达到预期效果。培训材料准备时间一般是培训时间的3倍以上。

## 项目验收

* 1. **交付物**

1. 《项目验收报告》
   1. **角色职责**
2. 项目经理：提交相关文档，配合销售完成项目验收。
3. 销售：完成项目验收。

## 项目运维

* 1. **说明**

作为完整的IT生命周期的一部分，系统运行和维护的生命周期中需要保障系统的高可用性、性能、安全性、可伸缩性等等。而系统管理的科学性日益成为维护系统成功平稳运行的重要因素。在运维过程中，从应用发布后，通过系统维护，应用维护进行每日系统运行中的支持工作，使用户能够使用系统；通过用户支持工作，得到用户反馈；一直到质量控制人员从整个系统的角度对系统进行SLM服务级别管理、系统容量管理、可靠性管理、灾难预警管理等，并通过提交系统优化可行性分析报告，促使达到优化现有系统，整个系统达到螺旋式上升的模式，不断提高系统功能和性能。微软公司在自身IT系统运行与开发的过程中总结开发出了一套保障系统运营与维护的一套方法学，称为微软运行框架(Microsoft Operation Framework，即MOF)。

MOF是一个衡量、管理和改善IT运行的框架，可以专门地用于微软环境。它基于ITIL，使用ITIL通用的语言和结构。因此它特别可以专门用于微软的环境。我们想通过ITIL提供高级别的认识，也提供了细节的、底层的信息使客户可以在可管理的环境中使用微软产品。随着时间的推移，将越来越专门性，使微软的产品与操作架构捆绑起来。MOF的目标在于：

1. 强调微软平台上的规程、组织、技术操作的规范化
2. 减少消耗时间和复杂性
3. 从运维管理的要求和实际情况考虑，本运行管理框架需要完成如下四类维护任务：系统维护、应用维护、用户支持及质量控制。
4. 系统维护。保障接入平台运行在一个稳定、安全、可靠的系统环境中。主要包括操作系统管理、网络管理、数据库管理、安全管理等。
5. 应用维护。确保应用程序的稳定运行及更新。主要包括程序的部署、发布、变更、配置、监控管理。
6. 用户支持。为用户提供服务支持及解决平台自身出现的各种问题及隐患。包括帮助台支持、故障管理、问题管理。
7. 质量控制。整体评估系统运行质量，包括服务级别管理、灾难预警管理、系统容量及可靠性管理。
   1. **交付物**
8. 《问题清单》
   1. **角色职责**
9. 测试经理：负责系统配置更新发布。
10. 项目经理/架构师：负责系统服务质量保障
11. 运维人员：负责系统运维

## 项目管理

* 1. **说明**

项目沟通、汇报的记录。

* 1. **交付物**

1. 《项目周报》
2. 《会议纪要》