

香草进化式 IT 服务规范 v1.0



北京香草信息技术有限公司

这是一个关于 IT 进化的规范，也是一个进化中的规范，它将会被不停地更新。

此规范的目的在于提供指导性的原则，以实现进化式的 IT 服务。此规范中的很多思路来自于 XP Programing（极限编程）的思想和《人月神话》、《人件》、《与熊共舞—软件项目风险管理》等经典著作。

一、定义



进化式 IT 服务是：服务提供者和客户，基于共同的长期利益，在一个长时间的持续的过程中，通过不断的磨合、改善、调整，共同努力，以尽可能小的代价使客户的 IT 系统始终良好的适合客户所处的环境。

适合环境，这正是进化服务的最主要含义。

二、目标

进化式 IT 服务的最大目标是：在服务提供者和客户之间一起形成一个良性互动的团队，这个团队应达到以下目标：

- I 相互赏识
- I 相互信任
- I 相互激励



我们认为这样一个团队是双方最为宝贵的财富，也是实现客户 IT 进化的根本保障。

二、原则

意识

首先是意识，客户和技术人员共同的意识，对进化的意识。具体而言，双方要在以下几方面达成共识：

1. 没有最好的，只有最合适的
2. IT 的改进需要长期的积累
3. 共同寻找改进的时机，和创造改进的机会
4. 坚信存在改进的可能

长期

实现 IT 的进化是为了追求相对长远的利益，虽然这并不排斥带来短期的利益，但我们相信只有基于一个相对长期的计划，才有可能考虑 IT 的进化，也更符合客户的利益。

即使双发的合同暂时是短期的，双发也应尽可能考虑更长远的计划，这样做不会带来任何坏处。



信任

双方之间的相互信任极为重要。信任是相互的，因此我们要努力做到：

1. 对客户的信任，其中包括：
 - a) 不怀疑客户提出的需求的必要性
 - b) 不怀疑客户报告的错误或使用上的问题的真实性
 - c) 充分理解客户在需求认知上的变化

上述要求虽然简单，却是许多技术人员所不能真正做到的。

2. 努力建立起客户对我们的信任

要做到这一点，需要在以下几方面进行努力：

- a) 让客户理解技术上的问题
即使问题需要相当的专业知识，也应该尽力解释
- b) 提供透明的进度报告，让客户理解并共同安排进度
- c) 不隐瞒已知的缺陷，或可能存在的问题
- d) 努力工作，并让客户知道
- e) 让客户理解创新或克服障碍可能存在的风险
- f) 征求客户的同意，允许为了做得更好，而在一定程度上出错的可能



沟通



保持良好的沟通非常重要，我们应该做的：

1. 确认双方对问题保持相同的理解，发现任何不确定立即联系
2. 及时将变化反馈
3. 定期的例行报告，及时在双方之间更新信息

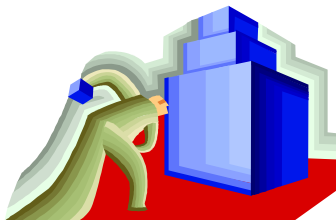
主动

不能发挥主动性的技术人员是糟糕的，我们要在以下几方面发挥主动：

1. 主动检查、分析和确认需求，而非盲目接受，发现问题立即反馈
2. 主动更正客户提供的文档、材料中显而易见的错误（如：文字拼写、明显的前后文冲突），而非置之不理
3. 主动承担额外的工作，帮助客户（如：数据处理、格式转换），而非袖手旁观
4. 主动发掘潜在的风险，并提供分析
5. 鼓励多管闲事



保持激情

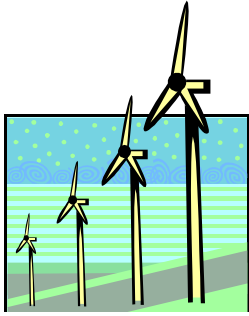


要保持工作的激情，充分相信我们工作的意义所在。

要铭记“人不知而不愠，不亦君子乎”，不为别人可能的不理解或不知道我们的工作而烦恼。

保持乐观，总是精神抖擞的进行工作，并以此影响他人。

鼓励



要做到有意识的自我鼓励和相互鼓励:

1. 鼓励客户对需求的深入分析，鼓励客户对软件的深入理解和使用
2. 请求客户对我们的鼓励
3. 对重要里程碑举行庆祝仪式

三、实施

启动

我们认为在双方的合作之初，应该采取措施实现前述目标：建立团队。可能采取的措施包括：

1. 双方相关人员的相互认识和熟悉，可以酌情考虑以下方案
 - a) 一起聚餐，野餐或家庭式聚餐尤为可取
 - b) 举行某运动的友谊赛
 - c) 一起拓展训练
 - d) 开会
 - e) 上门拜访
2. 阅读相同的书籍并讨论，推荐的书目包括：
 - a) 《人月神话》
 - b) 《人件》
 - c) 《与熊共舞—软件项目风险管理》
 - d) 《你的灯亮着吗？》
3. 建立起双方合作的技术平台，根据具体情况考虑以下手段：
 - a) 建立网站、讨论组
 - b) 建立 VPN 连接



4. 讨论并议定操作流程，包括：

- a) 需求的管理
- b) 计划进度的管理
- c) 文档的管理
- d) 发行版本的管理
- e) 紧急事件的处理机制
- f) 风险的共同防范

共同计划

参考《与熊共舞》



在项目之初，双方即以风险图的形式确定项目重要里程碑的边界。

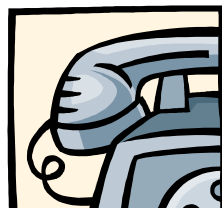
在项目进行过程中，根据实际进度和需求变化情况，举行会议重新讨论计划，并将新的计划以明确的方式发布。在此过程中要注意：

- 1. 确保所有人拥有共同的计划
- 2. 将计划明确并分解到个人，避免某件事情被所有人忽略
- 3. 确定日期之前充分征求各方意见，避免一厢情愿
- 4. 充分允许事先的质疑或过程中的变化请求

沟通

要实现良好的沟通，注意以下细节：

- 1. 不怕麻烦，随时确认，确保相同的理解
- 2. 保持愉快的语气
- 3. 微笑
- 4. 理解和尊重别人
- 5. 如果别人不理解，则要耐心
- 6. 注意寻找共同的话题



重构

虽然重构对于客户往往不可见，但技术人员应充分理解重构的重要性。本规范并不详细说明如何重构，仅指出实现重构前的步骤：



1. 阅读重构有关书籍并讨论
2. 了解并配备相关的软件工具
3. 有资深人员的指导
4. 注重设计

小步迭代

在进化式的 IT 开发中，小步迭代是取得成功的重要保证。我们实现小步迭代，要做到以下几个方面：

1. 短发布周期，建议以一周为周期
2. 增量开发，确定需求优先级次序
3. 划分出需求的最小元单位
4. 风险取决于价值，确定每一步迭代的风险和价值



附件

附件一 敏捷软件开发宣言（摘录自《敏捷软件开发—原则、模式与实践》）

我们正在通过亲身实践以及帮助他人实践，揭示更好的软件开发方法。通过这项工作，我们认为：

个体和交互	胜过	过程和工具
可以工作的软件	胜过	面面俱到的文档
客户合作	胜过	合同谈判
响应变化	胜过	遵循计划

虽然右项也具有价值，但我们认为左项具有更大的价值。

Kent Back James Grenning Robert C. Martin Mike Beedle
Jim Highsmith Steve Mellor Arie van Bennekum Andrew Hunt
Ken Schwaber Alistair Cockburn Ron Jeffries Jeff Sutherland
Ward Cunningham Jon Kern Dave Thomas Martin Fowler Brain Marick