

# LaTeX 双栏排版方法介绍

CSDN，繁星间漫步，陆巍的博客

2022 年 8 月 18 日

# 目录

第一章 pdfcolparallel 宏包实现双栏排版	1
第二章 parcolumns 宏包实现双栏排版	3
第三章 paracol 宏包实现双栏排版	5

# 第一章 pdfcolparallel 宏包实现双栏排版

Agile is a buzzword of software development, and so all DoD software development projects are, almost by default, now declared to be “agile.” The purpose of this document is to provide guidance to DoD program executives and acquisition professionals on how to detect software projects that are really using agile development versus those that are simply waterfall or spiral development in agile clothing (“agile-scrum-fall”).

## Principles, Values, and Tools

Experts and devotees profess certain key “values” to characterize the culture and approach of agile development. In its work, the DIB has developed its own guiding maxims that roughly map to these true agile values:

敏捷是软件开发的一个热门词汇，因此国防部所有的软件开发项目现在几乎在默认情况下都被宣布为“敏捷”。本文的目的是为国防部项目主管和采购专业人员提供指导，指导他们如何识别真正使用敏捷开发的软件项目，以及那些披着敏捷外衣的瀑布式或螺旋式开发（“agile-scrum-fall”）。

## 原则、价值观和工具

专家和拥护者声称某些关键的“价值观”是敏捷开发文化和方法的特征。在其工作中，DIB（国防创新委员会）制定了自己的指导准则，大致符合这些真正的敏捷价值观：

Agile Value / 敏捷价值观	DIB maxim / DIB 格言
Individuals and interactions over processes and tools / 个人和互动高于过程和工具	“Competence trumps process” / “能力胜过程”
Working software over comprehensive documentation / 可工作的软件胜过全面的文档	“Minimize time from program launch to deployment of simplest useful functionality” / “尽量缩短从程序启动到部署最简单有用功能的时间”
Customer collaboration over contract negotiation / 客户合作胜过合同谈判	“Adopt a DevSecOps culture for software systems” / “在软件系统上采用 DevSecOps 文化”
Responding to change over following a plan / 响应改变胜过遵循计划	“Software programs should start small, be iterative, and build on success - or be terminated quickly” / “软件程序应该从小处着手，进行迭代，并建立在成功的基础上——或者迅速终止”

Key flags that a project is not really agile:

- Nobody on the software development team is

一个项目不是真正敏捷的关键标志：

- 软件开发团队中没有人是在与软件用户交谈和

<sup>1</sup>Acceptable substitutes for talking to users: Observing users working, putting prototypes in front of them for feedback, and other aspects of user research that involve less talking.

<sup>2</sup>Dev is short for development, ops is short for operations

<sup>1</sup>与用户交谈的可接受替代方法：观察用户的工作情况，将原型放在他们面前以获得反馈，以及用户研究的其他涉及较少交谈的方面。

<sup>2</sup>Dev 是开发的简称，OPS 是运营的简称。

talking with and observing the users of the software in action; we mean the actual users of the actual code.<sup>1</sup> (The PEO does not count as an actual user, nor does the commanding officer, unless she uses the code.)

- Continuous feedback from users to the development team (bug reports, users assessments) is not available. Talking once at the beginning of a program to verify requirements doesn't count!
- Meeting requirements is treated as more important than getting something useful into the field as quickly as possible.
- Stakeholders (dev, test, ops, security, contracting, contractors, end-users, etc.)<sup>2</sup> are acting more-or-less autonomously (e.g., 'it's not my job.' )
- End users of the software are missing-in-action throughout development; at a minimum they should be present during Release Planning and User Acceptance Testing.
- DevSecOps culture is lacking if manual processes are tolerated when such processes can and should be automated (e.g., automated testing, continuous integration, continuous delivery).

观察他们的行为；我们是指实际代码的实际用户。<sup>1</sup>（PEO（项目执行官）不算是实际用户，指挥官也不算，除非她使用代码。）

- 没有用户对开发团队的持续反馈（错误报告、用户评估）。至于说在程序开始时与用户交谈过一次以验证需求，那不算数！
- 满足需求被视为比尽快将有用的东西投入现场更重要。
- 利益相关者（开发、测试、运营、安全、承包、承包商、最终用户等）或多或少都在自主行动（例如，”这不是我的工作”。）<sup>2</sup>
- 软件的最终用户在整个开发过程中处于缺失状态；他们至少应该在发布计划和用户验收测试期间出现。
- 如果允许手动过程，而这些过程可以并且应该是自动化的（例如，自动化测试、持续集成、持续交付），则缺乏 DevSecOps 文化。

## 第二章 parcolumns 宏包实现双栏排版

Agile is a buzzword of software development, and so all DoD software development projects are, almost by default, now declared to be “agile.” The purpose of this document is to provide guidance to DoD program executives and acquisition professionals on how to detect software projects that are really using agile development versus those that are simply waterfall or spiral development in agile clothing (“agile-scrum-fall”).

### Principles, Values, and Tools

Experts and devotees profess certain key “values” to characterize the culture and approach of agile development. In its work, the DIB has developed its own guiding maxims that roughly map to these true agile values:

敏捷是软件开发的一个热门词汇，因此国防部所有的软件开发项目现在几乎在默认情况下都被宣布为“敏捷”。本文的目的是为国防部项目主管和采购专业人员提供指导，指导他们如何识别真正使用敏捷开发的软件项目，以及那些披着敏捷外衣的瀑布式或螺旋式开发（“agile-scrum-fall”）。

### 原则、价值观和工具

专家和拥护者声称某些关键的“价值观”是敏捷开发文化和方法的特征。在其工作中，DIB（国防创新委员会）制定了自己的指导准则，大致符合这些真正的敏捷价值观：

Agile Value / 敏捷价值观	DIB maxim / DIB 格言
Individuals and interactions over processes and tools / 个人和互动高于过程和工具	“Competence trumps process” / “能力胜过程”
Working software over comprehensive documentation / 可工作的软件胜过全面的文档	“Minimize time from program launch to deployment of simplest useful functionality” / “尽量缩短从程序启动到部署最简单有用功能的时间”
Customer collaboration over contract negotiation / 客户合作胜过合同谈判	“Adopt a DevSecOps culture for software systems” / “在软件系统上采用 DevSecOps 文化”
Responding to change over following a plan / 响应改变胜过遵循计划	“Software programs should start small, be iterative, and build on success - or be terminated quickly” / “软件程序应该从小处着手，进行迭代，并建立在成功的基础上——或者迅速终止”

Key flags that a project is not really agile:

一个项目不是真正敏捷的关键标志：

<sup>1</sup> Acceptable substitutes for talking to users: Observing users working, putting prototypes in front of them for feedback, and other aspects of user research that involve less talking.

<sup>2</sup> Dev is short for development, ops is short for operations

<sup>1</sup> 与用户交谈的可接受替代方法：观察用户的工作情况，将原型放在他们面前以获得反馈，以及用户研究的其他涉及较少交谈的方面。

<sup>2</sup> Dev 是开发的简称，OPS 是运营的简称。

- Nobody on the software development team is talking with and observing the users of the software in action; we mean the actual users of the actual code.<sup>1</sup> (The PEO does not count as an actual user, nor does the commanding officer, unless she uses the code.)
- Continuous feedback from users to the development team (bug reports, users assessments) is not available. Talking once at the beginning of a program to verify requirements doesn't count!
- Meeting requirements is treated as more important than getting something useful into the field as quickly as possible.
- Stakeholders (dev, test, ops, security, contracting, contractors, end-users, etc.)<sup>2</sup> are acting more-or-less autonomously (e.g., 'it's not my job.' )
- End users of the software are missing-in-action throughout development; at a minimum they should be present during Release Planning and User Acceptance Testing.
- DevSecOps culture is lacking if manual processes are tolerated when such processes can and should be automated (e.g., automated testing, continuous integration, continuous delivery).
- 软件开发团队中没有人正在与软件用户交谈和观察他们的行为；我们是指实际代码的实际用户。<sup>1</sup>（PEO（项目执行官）不算是实际用户，指挥官也不算，除非她使用代码。）
- 没有用户对开发团队的持续反馈（错误报告、用户评估）。至于说在程序开始时与用户交谈过一次以验证需求，那不算数！
- 满足需求被视为比尽快将有用的东西投入现场更重要。
- 利益相关者（开发、测试、运营、安全、承包、承包商、最终用户等）或多或少都在自主行动（例如，”这不是我的工作”。）<sup>2</sup>
- 软件的最终用户在整个开发过程中处于缺失状态；他们至少应该在发布计划和用户验收测试期间出现。
- 如果允许手动过程，而这些过程可以并且应该是自动化的（例如，自动化测试、持续集成、持续交付），则缺乏 DevSecOps 文化。

## 第三章 paracol 宏包实现双栏排版

Agile is a buzzword of software development, and so all DoD software development projects are, almost by default, now declared to be “agile.” The purpose of this document is to provide guidance to DoD program executives and acquisition professionals on how to detect software projects that are really using agile development versus those that are simply waterfall or spiral development in agile clothing ( “agile-scrum-fall” ).

### Principles, Values, and Tools

Experts and devotees profess certain key “values” to characterize the culture and approach of agile development. In its work, the DIB has developed its own guiding maxims that roughly map to these true agile values:

敏捷是软件开发的一个热门词汇，因此国防部所有的软件开发项目现在几乎在默认情况下都被宣布为“敏捷”。本文的目的是为国防部项目主管和采购专业人员提供指导，指导他们如何识别真正使用敏捷开发的软件项目，以及那些披着敏捷外衣的瀑布式或螺旋式开发（“agile-scrum-fall”）。

### 原则、价值观和工具

专家和拥护者声称某些关键的“价值观”是敏捷开发文化和方法的特征。在其工作中，DIB（国防创新委员会）制定了自己的指导准则，大致符合这些真正的敏捷价值观：

Agile Value / 敏捷价值观	DIB maxim / DIB 格言
Individuals and interactions over processes and tools / 个人和互动高于过程和工具	“Competence trumps process” / “能力胜过程”
Working software over comprehensive documentation / 可工作的软件胜过全面的文档	“Minimize time from program launch to deployment of simplest useful functionality” / “尽量缩短从程序启动到部署最简单有用功能的时间”
Customer collaboration over contract negotiation / 客户合作胜过合同谈判	“Adopt a DevSecOps culture for software systems” / “在软件系统上采用 DevSecOps 文化”
Responding to change over following a plan / 响应改变胜过遵循计划	“Software programs should start small, be iterative, and build on success - or be terminated quickly” / “软件程序应该从小处着手，进行迭代，并建立在成功的基础上——或者迅速终止”

Key flags that a project is not really agile:

一个项目不是真正敏捷的关键标志：

- Nobody on the software development team is
- 软件开发团队中没有人是在与软件用户交谈和观

talking with and observing the users of the software in action; we mean the actual users of the actual code.<sup>1</sup> (The PEO does not count as an actual user, nor does the commanding officer, unless she uses the code.)

- Continuous feedback from users to the development team (bug reports, users assessments) is not available. Talking once at the beginning of a program to verify requirements doesn't count!
- Meeting requirements is treated as more important than getting something useful into the field as quickly as possible.
- Stakeholders (dev, test, ops, security, contracting, contractors, end-users, etc.)<sup>2</sup> are acting more-or-less autonomously (e.g., 'it's not my job.' )
- End users of the software are missing-in-action throughout development; at a minimum they should be present during Release Planning and User Acceptance Testing.
- DevSecOps culture is lacking if manual processes are tolerated when such processes can and should be automated (e.g., automated testing, continuous integration, continuous delivery).

---

<sup>1</sup>Acceptable substitutes for talking to users: Observing users working, putting prototypes in front of them for feedback, and other aspects of user research that involve less talking.

<sup>2</sup>Dev is short for development, ops is short for operations

察他们的行为；我们是指实际代码的实际用户。<sup>1</sup> (PEO (项目执行官) 不算是实际用户，指挥官也不算，除非她使用代码。)

- 没有用户对开发团队的持续反馈 (错误报告、用户评估)。至于说在程序开始时与用户交谈过一次以验证需求，那不算数!
- 满足需求被视为比尽快将有用的东西投入现场更重要。
- 利益相关者 (开发、测试、运营、安全、承包、承包商、最终用户等) 或多或少都在自主行动 (例如, "这不是我的工作")。<sup>2</sup>
- 软件的最终用户在整个开发过程中处于缺失状态; 他们至少应该在发布计划和用户验收测试期间出现。
- 如果允许手动过程, 而这些过程可以并且应该是自动化的 (例如, 自动化测试、持续集成、持续交付), 则缺乏 DevSecOps 文化。

---

<sup>1</sup>与用户交谈的可接受替代方法: 观察用户的工作情况, 将原型放在他们面前以获得反馈, 以及用户研究的其他涉及较少交谈的方面。

<sup>2</sup>Dev 是开发的简称, OPS 是运营的简称。