



**认证备考指南**

**201801 版**

**特别致谢：**

文档第三部分，EXIN DevOps Professional 基础术语表，整合了 DOP 认证配套材料，以及核心教材 *DevOps Handbook* 中出现的术语与使用场景。由以下几位老师共同修订，特此声明，以表感谢。

以下排名不分先后：

术语提交&翻译：刘征，张乐，许峰，汪珺

术语整理&修订：卢梦纯

考试大纲&细则修订：卢梦纯

术语反馈：

如您对术语翻译有修改建议，或提交新的术语，请发送邮件至 [info.china@exin.com](mailto:info.china@exin.com)，经确认后，将在下一版本更新中进行修订。

Copyright © EXIN Holding B.V. 2019. All rights reserved.

EXIN® is a registered trademark.

DevOps Master™ is a registered trademark.

No part of this publication may be published, reproduced, copied or stored in a data processing system or circulated in any form by print, photo print, microfilm or any other means without written permission by EXIN.



# 目录

|            |    |
|------------|----|
| 1. 概述      | 4  |
| 2. 考试要求和规范 | 7  |
| 3. 基本概念列表  | 10 |
| 4. 文献      | 16 |

# 1. 概述

EXIN DevOps Professional (DEVOPSP.CH)

## 概述

DevOps 不仅在软件服务领域享有盛名，其法则也同样被应用到与快速交付可靠的产品和服务相关的所有环境。DevOps 一方面促进敏捷开发、服务管理和精益改进的协同作用，另一方面在持续交付流水线中确保安全性和保持控制，以此为整个组织的成功做出贡献。

本模块的主要目的是测试考生是否熟悉 DevOps 实践“三步工作法”：流、反馈、持续学习和实验。通过学习和考试，考生将理解这些组织层面和技术层面的变革对其日常工作的影响。

## 总结

DevOps 是“开发”和“运维”这两个词的缩写。DevOps 是一套最佳实践方法论，旨在在应用和服务的生命周期中促进 IT 专业人员（开发人员、运维人员和支持人员）之间的协作和交流，最终实现：

- 持续集成：每天数次将所有开发工作副本并入共享主线
- 持续部署：持续发布，或尽可能经常地发布
- 持续反馈：在生命周期的各个阶段寻求来自利益干系人的反馈

本认证涵盖的 DevOps 实践源自三步工作法：

第一步是从开发到运维再到客户，实现从左到右快速流动。

第二步是从所有利益干系人到价值流，实现从右到左快速反馈。

第三步是通过创建高度信任的实验和风险承担文化，促进学习。

此外，还涵盖了各个阶段至关重要的安全问题及在变更期间保持合规性。

本认证的研发，集结了 DevOps 工作领域的各方专家，协作完成。

## 背景信息

EXIN DevOps 认证项目：



## 目标群体

EXIN DevOps Professional 认证的目标人群是在 DevOps 环境中工作的所有人员或考虑向 DevOps 工作法过渡的组织中的所有员工。

目标群体包括（不限于）：

- 软件和网站开发人员
- 系统工程师
- DevOps 工程师
- 产品和服务负责人
- 项目经理
- 测试工程师
- IT 服务管理操作和支持人员
- 流程经理
- 精益 IT 从业人员
- Agile Scrum 从业者

## 认证要求

- 顺利通过 DevOps Professional 考试。
- 建议预先了解敏捷、精益和/或 IT 服务管理，例如通过 EXIN Agile Scrum Foundation 考试、LITA Lean IT Foundation 考试，或基于 ISO/IEC 20000 的 EXIN IT Service Management Foundation 考试。



## 布鲁姆级别

EXIN DevOps Professional 认证根据布鲁姆分类学修订版对考生进行布鲁姆 2 级和 3 级测试。

- 布鲁姆 2 级：理解——记忆（1 级）之上的步骤。理解表明考生能够理解呈现的内容，并能够评估如何将学习资料应用到个人所在的环境中。  
这类题目旨在证明考生能够整理、对比、说明并选择有关事实和想法的正确描述。
- 布鲁姆 3 级：应用——表明考生有能力在与学习环境不同的背景下使用所学信息。  
这类题目旨在证明考生能够以不同的方式或新的方式应用所掌握的知识、实例、方法和规则，在新的情况下解决问题。这类题目通常包含一个简短的场景。

## 考试细节

|                  |                  |
|------------------|------------------|
| 考试类型：            | 计算机考试 或 笔答考试，单选题 |
| 题目数量：            | 40               |
| 通过分数：            | 65%              |
| 是否开卷考试：          | 否                |
| 是否允许携带电子设备/辅助设备： | 否                |
| 考试时间：            | 90 分钟            |

EXIN 的考试规则 and 规定适用于此门考试。

## 培训

### 培训时长

建议本培训课程的培训时长为 16 小时，该时长包括学员分组、备考和短暂休息。该时长不包括家庭作业、考试和午餐休息时间。

### 建议个人学习量

60 小时，根据现有知识的掌握情况可能有所不同。

### 培训机构

您可以通过 [www.exin.com](http://www.exin.com) 查找该认证的授权培训机构。

## 2. 考试要求和规范

| 考试要求                 | 考试规范               | 比重           |
|----------------------|--------------------|--------------|
| <b>1.DevOps 应用</b>   |                    | <b>12.5%</b> |
|                      | 1.1 DevOps 的基本概念   | 2.5%         |
|                      | 1.2 三步工作法的原则       | 3.75%        |
|                      | 1.3 组织             | 6.25%        |
| <b>2.第一步：流</b>       |                    | <b>25%</b>   |
|                      | 2.1 部署流水线          | 12.5%        |
|                      | 2.2 自动化测试          | 5%           |
|                      | 2.3 持续集成           | 5%           |
|                      | 2.4 低风险发布          | 2.5%         |
| <b>3.第二步：反馈</b>      |                    | <b>30%</b>   |
|                      | 3.1 遥测             | 7.5%         |
|                      | 3.2 反馈             | 10%          |
|                      | 3.3 假设驱动开发和 A/B 测试 | 5%           |
|                      | 3.4 评审和协调          | 7.5%         |
| <b>4.第三步：持续学习和实验</b> |                    | <b>20%</b>   |
|                      | 4.1 学习             | 10%          |
|                      | 4.2 发现             | 10%          |
| <b>5.信息安全与变更管理</b>   |                    | <b>12.5%</b> |
|                      | 5.1 信息安全           | 7.5%         |
|                      | 5.2 变更管理           | 5%           |
| 合计                   |                    | 100%         |

## 考试规范

### 1 DevOps 应用

#### 1.1 DevOps 的基本概念

考生能够...

1.1.1 **说明** DevOps 的基本概念，如持续交付、敏捷基础架构、Kata、在制品、技术债务、前置时间。

#### 1.2 三步工作法的原则

考生能够...

1.2.1 **区分**流、反馈及持续学习和实验的原则。

1.2.2 **说明**交互系统(SoE)和记录系统(SoR)与 DevOps 之间的关系的区别。

#### 1.3 组织

考生能够...

1.3.1 **说明** DevOps 角色如何为了增加业务价值而协作。

1.3.2 **说明** I 型、T 型、E 型与 DevOps 之间的关系的区别。

1.3.3 **说明**如何将运维与日常开发工作相结合。

### 2 第一步：流

#### 2.1 部署流水线

考生能够...

2.1.1 **选择**解决部署流水线问题的方法，如基础架构即代码、容器。

2.1.2 **选择**优化价值流的最佳解决方案。

2.1.3 **评估**共享版本控制库的完整性。

2.1.4 **调整**完成的定义(DoD)，以反映 DevOps 原则。

2.1.5 **说明**如何将工具用于环境构建和配置自动化。

#### 2.2 自动化测试

考生能够...

2.2.1 **说明**非理想测试金字塔模型和理想测试金字塔模型之间的区别。

2.2.2 **选择**测试驱动开发在流中的使用目的。

#### 2.3 持续集成

考生能够...

2.3.1 **选择**最佳的分支策略。

2.3.2 **说明**技术债务对流的影响。

2.3.3 **说明**如何消除技术债务。

#### 2.4 低风险发布

考生能够...

2.4.1 **区分**不同的发布和部署模式，从而实现低风险发布。

2.4.2 **选择**可供使用的合适的架构原型。



### 3 第二步：反馈

#### 3.1 遥测

考生能够...

3.1.1 **描述**遥测如何促进价值流的优化。

3.1.2 **描述**监控框架组件。

3.1.3 **说明**自助使用遥测的附加值。

#### 3.2 反馈

考生能够...

3.2.1 使用前滚和回滚方法**解决**部署问题。

3.2.2 **变更**发布指导要求清单，使其与 DevOps 指导相符。

3.2.3 使用投产就绪评审(LRR)及交接就绪评审(HRR)**执行**安全检查。

3.2.4 **说明**如何将用户体验(UX)设计用作反馈机制。

#### 3.3 假设驱动开发和 A/B 测试

考生能够...

3.3.1 **说明**如何将 A/B 测试与发布和特性测试相结合。

3.3.2 **说明**假设驱动开发如何促进预期结果的交付。

#### 3.4 评审和协调

考生能够...

3.4.1 **检查**拉动请求流程的有效性。

3.4.2 **说明**评审方法、同行评审、观察者评审、结对编程、和工具辅助评审。

3.4.3 根据特定情况**选择**最合适的评审方法。

### 4 第三步：持续学习和实验

#### 4.1 学习

考生能够...

4.1.1 **区分**猴军团中不同种类的猴子，以改进学习方式。

4.1.2 **召开**免责事后分析会议。

4.1.3 **说明**如何通过在生产环境中嵌入故障来产生恢复能力。

4.1.4 **说明**何时使用游戏日。

#### 4.2 发现

考生能够...

4.2.1 **描述**如何使用（成文的）非功能性需求(NFR)进行运维设计。

4.2.2 **说明**如何将可重复使用的运维用户故事嵌入开发之中。

4.2.3 **说明**哪些对象应存储在单一共享源代码存储库中。

4.2.4 **说明**如何将局部发现转化成整体改进成果。

### 5 信息安全与变更管理

#### 5.1 信息安全

考生能够...

5.1.1 **说明**如何整合预防性安全控制措施。

5.1.2 **说明**如何在部署流水线中整合安全措施。

5.1.3 **说明**如何使用遥测增强安全。

#### 5.2 变更管理

考生能够...

5.2.1 **说明**如何在变更期间保持安全。

5.2.2 **说明**如何在变更期间保持合规性。

### 3. 基本概念列表

本场包含考生应熟知的术语和缩写。

请注意，单靠这些术语并不能满足考试要求。考生必须了解这些概念，并能够举例。

| 英文  | 中文        |
|---|-----------|
| A/B Testing                               | A/B 测试    |
| Acceptance Stage                          | 验收阶段      |
| Acceptance Test-Driven Development (ATDD) | 验收测试驱动开发  |
| Acceptance Tests                          | 验收测试      |
| Accident                                  | 事故        |
| Affinity                                  | 亲和        |
| Agile                                     | 敏捷        |
| Andon Cord                                | 安灯拉绳      |
| Anomaly Detection Techniques              | 异常探测技术    |
| Antifragility                             | 抗脆弱性      |
| Application Deployment                    | 应用部署      |
| Artifact Management                       | 构件制品库管理   |
| Artifacts                                 | 制品        |
| Automated Tests                           | 自动化测试     |
| Automation                                | 自动化       |
| Backlog                                   | 待办事项列表    |
| Bad Apple Theory                          | 坏苹果理论     |
| Bad Paths                                 | 失败路径      |
| Batch Sizes                               | 批次尺寸、批量大小 |
| Blame                                     | 责备        |
| Blameless Post Mortem                     | 免责事后分析    |
| Blamelessness                             | 免责        |
| Blue-Green Deployment                     | 蓝绿部署      |
| Blue-Green Deployment Pattern             | 蓝绿部署模式    |
| Branching Strategy                        | 分支策略      |
| Brownfield                                | 棕地        |
| Build                                     | 构建        |
| Business Value                            | 业务价值      |
| Canary Release                            | 金丝雀发布     |
| Canary Release Pattern                    | 金丝雀发布模式   |
| Card                                      | 卡片        |

|                                       |            |
|---------------------------------------|------------|
| Change Categories                     | 变更类别       |
| Change Schedules                      | 变更计划       |
| Cloud Computing                       | 云计算        |
| Cloud Configuration Files             | 云配置文件      |
| Cluster Immune System Release Pattern | 集群免疫系统发布模式 |
| Code Branch                           | 代码分支       |
| Code Review Forms                     | 代码审查表      |
| Codified Nfr                          | 成文的非功能需求   |
| Collaboration                         | 协作         |
| Commit Stage                          | 提交阶段       |
| Commit Code                           | 提交代码       |
| Compliance Requirement                | 合规性要求      |
| Compliance Checking                   | 合规性检查      |
| Compliance Officer                    | 合规检测官      |
| Configuration Management              | 配置管理       |
| Container(S)                          | 容器         |
| Continuous Deployment                 | 持续部署       |
| Continuous Integration                | 持续集成 (CI)  |
| Continuous Delivery                   | 持续交付 (CD)  |
| Conways Law                           | 康威定律       |
| Cycle Time                            | 周期时间       |
| Defect Tracking                       | 缺陷跟踪       |
| Definition Of Done (DoD)              | 完成的定义      |
| Dev Rituals                           | 开发仪式       |
| Developer                             | 开发人员       |
| Development                           | 开发         |
| Devops Transformation                 | DevOps 转型  |
| Downstream/Upstream                   | 下游/上游      |
| Downwards Spiral                      | 恶性循环       |
| E-Mail Pass-Around                    | 电子邮件轮查     |
| Expand/Contract Pattern               | 扩张/收缩模式    |
| Exploratory Test                      | 探索性测试      |
| Fast Feedback                         | 快速反馈       |
| Feature                               | 特性         |
| Feature Flag                          | 特性标志       |
| Feature Toggles                       | 特性开关       |
| Feedback/Feedback Loop                | 反馈/反馈回路    |
| Feedforward/Feedforward Loop          | 前馈/前馈回路    |
| Flow                                  | 流          |

|                                     |               |
|-------------------------------------|---------------|
| Gated Commit                        | 门控提交          |
| Gaussian Distribution               | 高斯分布          |
| Green Build                         | 绿色构建          |
| Greenfield                          | 绿地            |
| Handoff                             | 交接            |
| Hand-Off Readiness Review           | 交接就绪评审        |
| Happy Paths                         | 快乐路径          |
| Hypothesis-Driven Development       | 假设驱动开发        |
| Incident                            | 事件            |
| Information Radiators               | 信息辐射器         |
| Infosec                             | 信息安全          |
| Infrastructure Automation           | 基础架构自动化       |
| Infrastructure As Code              | 基础架构即代码       |
| Integration Tests                   | 集成测试          |
| I-Shaped, T-Shaped, E-Shaped        | I 型, T 型, E 型 |
| Iteration                           | 迭代            |
| Itsm (It Service Management)        | IT 服务管理       |
| Ji-Kotei-Kanketsu (JKK)             | 质量检查 (JKK)    |
| Just Culture                        | 公正文化          |
| Just-In-Time (JIT)                  | 准时制           |
| Kaizen (In Lean)                    | 持续改善          |
| Kaizen Blitz (Or Improvement Blitz) | 持续改善闪电战       |
| Kanban                              | 看板            |
| Kata                                | Kata          |
| Large Batch Size Merge              | 大批量合并         |
| Latent Defects                      | 潜在缺陷          |
| Lauching Guidance                   | 发布指导          |
| Launch Readiness Review             | 投产就绪评审        |
| Lead Time                           | 前置时间          |
| Lean                                | 精益            |
| Learning Culture                    | 学习文化          |
| Logging Levels                      | 日志级别          |
| Loosely Coupled Architecture        | 松耦合架构         |
| Micro-Services                      | 微服务           |
| Minimum Viable Product              | 最小化可行产品       |
| Monitoring Framework                | 监控框架          |
| Monolithic Applications             | 单体应用          |
| Monolytics                          | 单体应用          |
| MTTR                                | 平均恢复时间        |

|  |                   |
|--|-------------------|
| Non-Functional Requirement (NFR)         | 非功能性需求            |
| Non-Functional Requirement (NFR) Testing | 非功能需求测试           |
| (Non) Ideal Testing Pyramid              | (非) 理想测试金字塔模型     |
| One-Piece-Flow                           | 单件流               |
| Operations                               | 运维                |
| Operations Story                         | 运维故事              |
| Ops Liaison                              | 运维联络人             |
| Organisational Typology Model            | 组织结构模型            |
| Organization Archetypes                  | 组织原型              |
| Organizational Learning                  | 组织级学习             |
| Over-The-Shoulder                        | 观察者评审             |
| Packages                                 | 包                 |
| Pair Programming                         | 结对编程              |
| Peer Review                              | 同行评审              |
| Pilot                                    | 试点                |
| Pipeline                                 | 流水线               |
| Plan-Do-Check-Act Cycle (PDCA Cycle)     | 计划-实施-检查-改进 (戴明环) |
| Post-Mortem                              | 事后剖析              |
| Process Time                             | 处理时间              |
| Product Owner                            | 产品负责人             |
| Pull Request Process                     | 拉动请求流程            |
| QA                                       | 质量保证              |
| Reduce Batch Size                        | 降低批次尺寸            |
| Reduce Number Of Handoffs                | 减少交接次数            |
| Regression Test                          | 回归测试              |
| Release Branch                           | 发布分支              |
| Release Managers                         | 发布经理              |
| Release Patterns                         | 发布模式              |
| Retrospective                            | 回顾                |
| Rhythm                                   | 节奏                |
| Roll-Back                                | 回滚                |
| Sad Path                                 | 悲伤路径              |
| Safety Culture                           | 安全文化              |
| Safety Conditions                        | 安全条件              |
| Scaling                                  | 规模化               |
| Scrum                                    | Scrum             |
| Scrum Master                             | Scrum Master      |
| Security Testing                         | 安全测试              |
| Self Service Capability                  | 自服务能力             |

|  |   |
|--|---|
| Service Deployment   | 服务部署  |
| Service Level Agreement (SLA)  | 服务级别协议 (SLA)  |
| Shared Goals   | 共享目标  |
| Shared Operations Team (SOT)   | 共享运维团队  |
| Shared Version Control   | 共享版本控制  |
| Single Repository  | 单一存储库   |
| Smoke Testing  | 冒烟测试  |
| Sprint   | 冲刺  |
| Staging  | Staging   |
| Staging Environments, SIT  | 准生产环境   |
| Stakeholder  | 利益干系人   |
| Standard Deviation   | 标准差   |
| Standard Operations  | 标准运维  |
| Static Code Analysis   | 静态代码分析  |
| Swarm  | 聚集、聚焦、会诊、围观 (动词)  |
| Swarming   | 聚集  |
| System Of Engagement (SOE)   | 交互系统  |
| System Of Records (SOR)  | 记录系统  |
| Technical Debt   | 技术债务  |
| Technology Adaption Curve  | 技术适应曲线  |
| Technology Executives  | 技术主管  |
| Telemetry  | 遥测  |
| Test Coverage Analysis   | 测试覆盖率分析   |
| Test Story   | 测试故事  |
| Test-Driven Development  | 测试驱动开发  |
| The Downward Spiral - TDS  | 下行螺旋  |
| The Agile Manifesto  | 敏捷宣言  |
| The Lean Movement  | 精益运动  |
| The Simian Army:<br>Chaos Gorilla,<br>Chaos Kong,<br>Conformity Monkey,<br>Doctor Monkey,<br>Janitor Monkey,<br>Latency Monkey,<br>Security Monkey | 猿猴军团 (可靠性监控服务) :<br>混沌大猩猩 Chaos Gorilla,<br>混沌金刚 Chaos Kong,<br>一致性猴子 Conformity Monkey,<br>医生猴子 Doctor Monkey,<br>看门猴子 Janitor Monkey,<br>延迟猴子 Latency Monkey,<br>安全猴子 Security Monkey |
| The Three Ways   | 三步工作法   |
| Theory Of Constraints  | 约束理论  |
| Ticketing System   | 工单系统  |
| Tightly-Coupled  | 紧耦合   |
| Tool-Assisted Review   | 工具辅助评审  |
| Tools  | 工具  |

|                                  |        |
|----------------------------------|--------|
| Toyota Production System (TPS)   | 丰田生产系统 |
| Toyota Kata                      | 丰田套路   |
| Transformation Team              | 转型团队   |
| Trunk                            | 主干     |
| User Story                       | 用户故事   |
| Value Stream Mapping             | 价值流映射  |
| Value Stream                     | 价值流    |
| Velocity                         | 速率     |
| Version Control                  | 版本控制   |
| Virtualized Environment          | 虚拟化环境  |
| Visible                          | 可视的    |
| Visualisation                    | 可视化    |
| Waste                            | 浪费     |
| Waste Reduction                  | 减少浪费   |
| Waterfall                        | 瀑布式    |
| WIP (Work In Progress / Process) | 在制品    |
| WIP Limit                        | 在制品限制  |
| Work Center                      | 工作中心   |

## 4. 文献

### 必选教材

The knowledge required for the NAME exam is covered in the following literature:

- A. 作者: Gene Kim, Jez Humble, Patrick Debois, John Willis  
**The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations**  
出版社: IT Revolution Press; 第一版 (2016 年)  
国际标准书号: ISBN-10: 1942788002; ISBN-13: 978-1942788003

### 可选教材

- B. 作者: Bart de Best  
**DevOps Best Practices**  
出版社: Leonon Media (2017 年)  
国际标准书号: ISBN-13: 978-94-92618-07-8
- C. 作者: Gene Kim, Kevin Behr, George Spafford  
**The Phoenix Project 《凤凰项目》**  
出版社: IT Revolution Press (2013 年 1 月 10 日)  
国际标准书号: ISBN-10: 0988262576; ISBN-13: 978-0988262577

*该文献目前已有中文版图书《凤凰项目：一个运维的传奇故事》  
译者：成小留  
书号：ISBN 9787-7-115-40365-0*

- D. 其他来源:  
<http://newrelic.com/devops>  
<http://devops.com/Author>

### 备注

可选教材仅作为参考和其他深度学习使用



## 文献考点分布矩阵

| 考试要求                  | 考试规范               | 文献 | 文献参考章节               |
|-----------------------|--------------------|----|----------------------|
| <b>1.DevOps 应用</b>    |                    |    |                      |
|                       | 1.1 DevOps 的基本概念   | A  | 前言, 第一部分介绍, 第 1、21 章 |
|                       | 1.2 三步工作法的原则       | A  | 第 2、3、4、5 章          |
|                       | 1.3 组织             | A  | 第 26、7、8 章           |
| <b>2.第一步: 流</b>       |                    |    |                      |
|                       | 2.1 部署流水线          | A  | 第 5、6、7、8、9、11 章     |
|                       | 2.2 自动化测试          | A  | 第 10 章               |
|                       | 2.3 持续集成           | A  | 第 11、21、22 章         |
|                       | 2.4 低风险发布          | A  | 第 12、23 章            |
| <b>3.第二步: 反馈</b>      |                    |    |                      |
|                       | 3.1 遥测             | A  | 第 14、15 章            |
|                       | 3.2 反馈             | A  | 第 16 章               |
|                       | 3.3 假设驱动开发和 A/B 测试 | A  | 第 17 章               |
|                       | 3.4 评审和协调          | A  | 第 18 章               |
| <b>4.第三步: 持续学习和实验</b> |                    |    |                      |
|                       | 4.1 学习             | A  | 第 19 章、附录 9          |
|                       | 4.2 发现             | A  | 第 20 章               |
| <b>5.信息安全与变更管理</b>    |                    |    |                      |
|                       | 5.1 信息安全           | A  | 第 22 章               |
|                       | 5.2 变更管理           | A  | 第 23 章               |

**联系 EXIN**

[www.exin.com](http://www.exin.com)

