

信息系统协会

AIS电子图书馆 (AISEL)

---

经济资讯2022

进程轨道17: IT战略、管理&治理

---

1月17日上午12:00

## 分析和评估当今开源的力量: 开源价值画布

芭芭拉·斯蒂芬

德国多特蒙德大学, barbara.steffen@tu-dortmund.de

弗雷德里克·默勒

德国多特蒙德大学; 德国弗劳恩霍夫ISST, frederik.moeller@tu-dortmund.de

请访问以下网址进行此操作和其他工作: <https://aisel.aisnet.org/wi2022>

---

### 推荐引文

Steffen, Barbara和Möller, Frederik, “分析和评估当今开源的力量: 开源价值画布” (2022)。经济新闻2022年会议记录。6. [https://aisel.aisnet.org/wi2022/it\\_strategy/it\\_strategy/6](https://aisel.aisnet.org/wi2022/it_strategy/it_strategy/6)

本资料由AIS电子图书馆 (AISEL) 的Wirtschaftsinformation带给您。AIS电子图书馆 (AISEL) 的授权管理员已接受将其纳入《经济周刊2022年会刊》。如需更多信息, 请联系[library@aisnet.org](mailto:library@aisnet.org).

# 分析和评估当今开源的力量：开源价值画布

芭芭拉·斯蒂芬<sup>1</sup>和弗雷德里克·穆勒<sup>1,2</sup>

<sup>1</sup> TU Dortmund University, 德国多特蒙德I1, 2行业信息管理主席

{barbara.steffel, 弗雷德里克·莫勒}@t1, 2ortmund.e

德国多特蒙德弗劳恩霍夫ISIT <sup>2</sup>号

{frederik.moelllr}@isst.fraunhofer.de

**摘要。**社会和经济的数字化进程为开源范式带来了新的启示。以前，开源只是一种开发人员范式，用于公开共享代码并将其提供给他人。然而，鉴于创新和优化的需要，公司可以利用开源组件来使用开箱即用、在上面构建服务或替换可商品化的服务。随后，使用开源组件的公司有很大的潜力创造新的价值。为了帮助公司和研究人员实现这一目标，本文提出了开源价值画布，供公司协作和跨学科识别开源价值。它特别旨在从业务和IT的角度分析和调整开源潜力。我们从一个正在进行的研究项目中汲取了丰富的见解，该项目为欧洲物流行业提供量身定制的开源组件。

**关键词：**开源战略、开放式二元集成、开源价值画布、利益相关者对齐

## 1 介绍

社会和经济的持续数字化发展对公司提出了新的挑战[1]，例如基于数据识别和设计新的商机，以及总体上应对数字化转型[2]。产生数字创新的一种方法是使用开箱即用的开源（OS）组件，以取代可商品化的服务或在其上构建新的服务[3–5]。一般来说，OS是指具有开放代码库的所有软件组件，并允许检查源代码[6]。如今，90%的软件以某种方式集成了操作系统组件[7]，导致公司“（...）既是开源软件产品的重度用户，也是其开发的贡献者”[6 p. 649]。Fuerstenau等人[8]以Otto.de为例，报告了操作系统的利用率和社区的相应人群知识。该公司开放了部分平台，通过第三方开发人员利用外部投入。该案例是操作系统作为超越组织边界、利用知识、分担成本和改进现有产品和服务的工具的一个典型例子[9]。

2022年2月，德国纽伦堡，第17届国际电线大会

然而，公司有太多的决定因素来决定何时、如何以及从谁那里使用操作系统组件（例如，在治理和审批流程方面[10]）。例如，这些包括根据操作系统组件的活动和质量评估操作系统的需求以及操作系统组件社区的可信度和声誉[9, 11, 12]。考虑到可能集成质量低于标准的操作系统组件，降低公司产品/软件解决方案的整体质量[13]，在使用操作系统组件时考虑这些方面至关重要。

如今，集成操作系统组件是公司IT部门的一个独特问题，因为它具有战略业务意义。从公司管理的角度来看，必须分析和评估操作系统集成的成本效益比，以确保与单一的内部解决方案相比，质量/报价增强和/或内部成本/工作量减少。只有这样，才能确保公司的长期目标与向操作系统迈进保持一致[14]。因此，目的是确定混合潜力，即操作系统使用和内部开发的混合，例如开发缓解、商业模式整合或持续创新[15, 16]。识别和评估这些潜力需要跨学科的方法[3, 16-18]。

本文设计并验证了1, 2n源值画布（OSVC），该画布从多个角度分析和评估OS潜力。我们选择了一个可视化查询工具（也称为画布，我们在本文中使用的术语）作为指导跨学科和创新操作系统战略努力的人工制品。画布是一种结构化的协作工具，用于基于预定义的构建块设计感兴趣的东西（例如，商业模型画布（BMC）中的商业模型[19, 20]）[21]。它的特点是采用专门的跨学科方法，这应该产生对可视化问题领域的共同理解，并让用户能够创造性地提出解决方案[22, 23]。最近，学者们在多个领域提出了画布，例如，根据数据寻找商业模式潜力（例如，见[24]），设计研究（例如，参见[25, 26]），或支持构思（例如，见图[27]）和平台设计[28]。鉴于围绕操作系统的新机遇和风险，我们认为OSVC的共享创新设计是一项具有很高价值的努力，可以帮助从业者和研究人员概念化和评估操作系统集成的潜力。

鉴于`canvases`是人工制品（e. 1, 2见[19]），我们在一项设计科学研究（DSR）中开发了OSVC，该研究基于Hevner嵌入项目环境中的三周期视图[29]。OSVC解决了我们的项目，这是一个欧洲物流业正在进行的大型操作系统驱动项目，面临着协助物流公司寻找操作系统使用和集成的机会和风险的挑战。该项目为欧洲物流业开发了一个专用操作系统组件库，以促进全行业创新，制定标准，确保兼容性，并允许物流公司进入相应的平台经济。如今，几乎没有特定于物流的操作系统组件，因此该项目为解决公司的核心竞争力开辟了新的战略机遇。因此，OSVC支持对操作系统业务战略使用不熟悉的公司识别操作系统的潜力，并选择最合适的操作系统组件。因此，本文探讨了以下研究问题：

**研究问题：**如何通过识别和选择合适的操作系统组件，指导公司定义操作系统策略，从操作系统集成中创造价值？

本文的结构：在接下来的1,2节中，我们简要介绍了开源的价值机制。在第3节中，我们概述了我们的研究方法。第4节展示了我们论文的最终结果，OSVC，它的验证和讨论。第5节总结了本文的贡献、局限性和进一步研究的途径。

## 2 开源

从根本上说，1S是指软件的源代码<sup>1</sup>，2是开放的，给了其他人检查它的方法[6, 30]。也许最著名的操作系统许可证是通用公共许可证（GPL），例如Linux使用的GPL，它放弃了关于使用源代码和衍生产品的典型版权限制，而这些限制反过来也必须根据GPL许可证发布[31]。相比之下，专有解决方案只能将软件用作可执行的“黑匣子”程序[11]。然后将专有软件授权给客户，公司通常提供更新和维护[32]。操作系统与“闭源”或专有软件解决方案有很大不同[33]。例如，操作系统可以成为高质量标准的载体，但它也包括文档很少、没有或低于标准的项目，这使操作系统软件的工作和使用变得复杂[33]。

通常，操作系统范式与自由软件混为一谈，如“共享软件”或“免费软件”[34]。随后，操作系统可能需要“开放”，但它不一定是免费的，这促使众多学者分析操作系统的商业模式和商业化[34, 35]。例如，Okoli和Nguyen[35]基于德尔菲研究提出了八种自由和开源软件商业模式，包括广告、会员资格/捐赠或更新订阅。在这方面，自由意味着运行、重新分发、修改和重新分发源代码修改版本的自由[36]。随后，操作系统对公司有很多影响，因为它提供了各种潜在的优势。例如，向<sup>第</sup>三方开发人员开放访问可以利用外部创新来提供持续创新[37]，并在代码库披露的情况下确保更好的安全性[38]。虽然“开源”的决定充满了各种设计选择，但其优势包括更快地采用用户或潜在的新兴活跃社区，这有助于产品改进[39]。

有趣的是，人们对操作系统的理解和感知与公司的业务和IT观点截然不同。这两种观点都强调了不同的机会和威胁，因此通常不会达成共同的操作系统理解。在下文中，我们将根据项目观察和经验以及与IT和操作系统专家的对话，简要介绍操作系统的好处、风险和常见的误解。

一般来说，操作系统被认为是现成的、可集成的软件组件[40]。与单独的内部开发相比，它被视为减少对内部专业知识需求的杠杆，因此是一种外包机制，可以减少开发时间和成本，促进持续创新[41, 42]。此外，由于操作系统社区的高标准，源代码通常质量很高[41, 43]。操作系统代码可以查看，因此必须“美观”，经过良好测试（最好是由一个重要的用户组进行测试，其中所有用户都可以报告并提出错误修复）[40, 42]，并且易于理解，以便进行操作系统集成和修改。在这里，操作系统专家强调，只有满足这些标准，操作系统组件才有价值。

否则，太多的修改和太长的操作系统代码熟悉期是必要的，因为后者进入操作系统项目后，理解和贡献操作系统项目会变得更加困难。此外，单个供应商的独立性以及易于模块化集成和修改是重要的资产[40-43]。这使得全球范围内的技术转让和合作成为可能，使公司无法独立开发的商业模式成为可能[41]。

然而，操作系统的使用和贡献也伴随着风险[43]。操作系统组件的用户不享有保修权、承诺的组件功能保证以及长期维护和支持服务[41-43]。在这里，IT专家强调了对内部专家的依赖，以评估操作系统组件的质量，并选择和集成最合适的组件。此外，他们需要意识到所有潜在风险，并确保在社区活动停止的情况下，所有服务都能在内部得到补偿。因此，操作系统的使用并不会使内部专业知识变得多余，而是要求公司雇佣（罕见的）优秀的IT专家，他们担任IT协调专家，涵盖从开发和运营到（持续）部署的所有重要IT职能[41, 42]。为了实现这一目标，IT员工已经是操作系统专家，接受了所需的培训，或者得到了顾问的支持。这三种选择都是稀缺的、昂贵的，并且存在依赖性[42, 43]）。

总之，操作系统的使用可能非常有价值，但它的集成从来都不是免费的[44]。凭借正确的内部专业知识，操作系统的使用和贡献是内部员工队伍的重要补充，也是商业软件产品的竞争解决方案[44]。然而，如果缺乏这种内部专业知识，公司就会陷入一种非常危险的依赖状态，无法弥补。一般来说，业务专家往往高估了操作系统组件使用的质量和好处，而忽视了上述内部任务和责任。因此，预期的成本节约和缩短的开发时间通常小于预期[44]。

然而，操作系统的使用可以补充公司目前的产品和创新潜力[40]。由于操作系统组件的高质量和巨大的定制潜力，许多IT专家倾向于操作系统集成。与商业软件使用相比，这些为他们提供了很大的自由度。为了实现更现实的操作系统理解，并指导公司进行操作系统评估和潜在的集成，OSVC充当公司不同利益相关者的教育者和协调者。其主要重点在于协调业务和IT视角。如果公司是第一次考虑操作系统，或者对其战略杠杆是新的，因此跨学科竞争，这种支持尤其重要。第一个操作系统集成步骤可以是混合开发模型或混合业务模型。在这里，公司可以使用OSVC来设计内部和操作系统能力的一致性[15]。

### 3 研究设计

我们采用DSR方法，因为我们的预期结果是一个工件（即模型）[45, 46]。它使用户能够通过抽象来理解复杂的问题，并专注于所调查问题中最相关的部分[46]。我们的研究源于欧洲物流业的一个案例，该案例明确侧重于向物流公司（包括物流服务提供商）提供操作系统组件。这个

物流特定操作系统组件的采用、使用 and 商业化被认为是必不可少的，但仍然是一个尚未开发的领域（基于我们在项目环境中的工作经验）。由于缺乏对操作系统潜力和风险的基本了解（在公司实践中），我们得出了相应画布支持的必要性和要求，以实现跨学科（如IT和业务）和针对结构不良问题的创造性协作[21, 23]。

我们遵循Hevner的三周期观点，因为它非常适用于创建新的工件，并在最近一篇开发画布的论文中得到了应用（例如，见[25]）。该框架要求在相关性、设计和严谨性循环之间循环，使我们能够整合例如案例发现、从业者、专家、文献和理论。

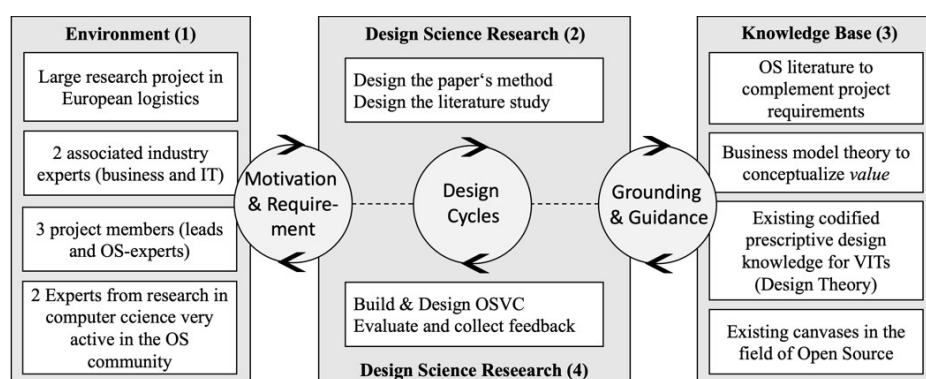


图1。基于Hevner三周期观的研究方法[29]

就**环境（1）**而言，我们的动机源于上述欧洲大型物流项目。由于我们是项目的一部分，我们可以通过与项目负责人、项目成员和物流行业的相关行业合作伙伴的互动（即非正式会谈、不同组合的定期会议、项目文档和演示）来了解对这样一个画布的需求。值得注意的是，目标是生成一个直观的工具，帮助从业者就是否应该集成操作系统以及如果应该集成哪些操作系统组件做出决策。为了适应战略性地利用操作系统组件的跨学科性，我们还邀请了来自计算机科学的外部专家，以提供涵盖技术方面的独特的IT重视角。

在**设计科学研究（2）**阶段，我们设计了论文的方法和目标及其文献综述研究，以开发OSVC的第一个版本。虽然文献综述并不全面，但我们的目标是确定初始构建块，以补充我们在项目设置中面临的要求。我们使用这些初始构建块作为作者之间讨论的基础，并促进对操作系统领域的共同理解。这些初始构建块作为迭代开发最终版本构建块的起点。

我们设计的**知识库（3）**有四个部分。首先，我们从现有的文献语料库中收集见解，搜索涉及操作系统组件或操作系统商业模式的使用及其相关价值的论文。为此，我们从AISel收集了论文，在标题和摘要中搜索了“开源”和“商业”

分别为“模型”和“价值”。我们搜索了涉及操作系统价值维度的论文，例如，就其商业模式（如[35]）或一般商业价值（如[47]）而言，并构建了一个初步的概念矩阵[48]。这些关键字帮助我们识别使用操作系统软件和组件的商业动机。搜索共产生了84篇论文，我们将其缩减为32篇论文的样本进行更深入的分析[48]。本文综述仅涵盖IS领域，因为它只是为了补充我们的案例和车间设计迭代。然而，对文献语料库的深入分析以及从其他数据库（如IEEE）收集额外文献是一项尚待完成的任务。

其次，由于我们努力实现商业价值，我们采用商业模式作为理论视角[49]，将我们的发现整合到设计项目中。特别是，我们绘制了典型的商业模式概念，如价值主张、价值评估和价值捕获，并将其与我们的操作系统特定发现相结合，以确保操作系统战略符合公司的整体战略（见第4.1节）。

第三，我们使用Avdi ji等人[21, 50]在视觉探究工具（画布）的设计理论中编纂为设计原则的现有规范性设计知识。在设计项目中，使用现有的设计知识对于减少冗余和创建成功的设计至关重要[51]。在这里，我们通过在画布中实例化画布来重用画布的设计原则[52]。

第四，谷歌搜索确定了覆盖操作系统的四个现有画布。所有四个都包含在概念矩阵中。分析表明，没有一个画布明确质疑和挑战操作系统的使用，也没有解决操作系统组件的价值导向利用问题，从而支持我们开发OSVC的动机。

在第二个**设计科学研究（4）**阶段，我们通过四个研讨会迭代改进OSVC的初始设计及其概念表，该概念表由OSVC的11个构建块中的每一个的指导性问题组成（见**图2**和**表1**）：1. 项目负责人2名。两个项目成员，3. 2名操作系统专家和4名。物流负责人（业务和IT专家）的两名代表。在信息系统研究和DSR中，研讨会对于在互动环境中收集从业者的见解非常有价值，例如，对于新工件设计的评估和迭代[53, 54]。

基于知识库的发现，例如文献和我们的案例经验，我们设计了OSVC的初始版本。然后，我们在连续的研讨会中评估和迭代了OSVC（设计）[53]。选择四个研讨会参与者的原因：第一和第二，项目负责人/成员可以从他们的项目专业知识中挑战OSVC，这些专业知识涵盖了行业需求、要求以及向用户传达操作系统价值潜力的困难。第三，IT（尤其是操作系统）专家确保IT视角的独特和现实的整合，例如，开发人员在选择和参与操作系统社区/组件时的典型决策过程。最后，行业专家在预期的跨学科环境中使用了OSVC，使我们能够收集复杂的行业反馈。

## 4 开源价值画布

在本节中，我们将介绍OSVC。它帮助用户设计创造性的解决方案，以利用操作系统的潜力，为潜在的陷阱做好准备并减少陷阱。

## 4.1 开源价值画布透视图

OSVC由11个构建块组成。他们共同设计公司的预期操作系统战略，并根据其质量和适用性支持操作系统组件的选择。使用商业模式研究中的现有术语，可以将构建块分配给三个归纳得出的公司级战略类别。这对于证明公司战略和操作系统使用之间的相互依存关系非常有价值。这些是：

- **价值主张**：描述公司在潜在商业模式中操作系统的潜在动机和使用选项及其预期收益[55]。
- **价值评估**：描述重要的考虑因素，以确保对操作系统项目和组件的潜在使用进行有根据的评估和评价。
- **价值捕获**：描述操作系统产生的成本和货币化潜力，从而直接风险和收益[56]。

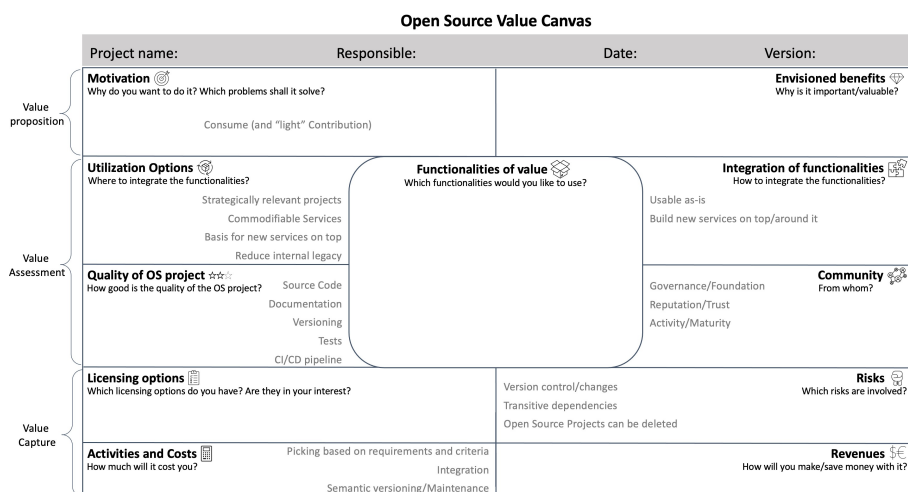


图2:OSVC的最终版本

大多数构建块通过两到五个相关子类别进一步细化（见图2和表1）。根据Avdi ji等人[21]的设计原则，并补充OSVC，我们得出了“使用说明”[21]，以简化画布的使用。我们在下面的概念表中将这些方向编入指导性问题，以支持用户正确应用OSVC。我们从构建块的总体主题和每个研讨会的评论、见解和反思中得出了指导性问题。随后，指导性问题是随着时间的推移而出现和成熟的迭代设计工件。

## 4.2 价值主张

价值主张涵盖了两个构建块，解决了公司在集成操作系统（组件）时的整体动机和预期收益。



**动机：**它代表了公司使用操作系统（组件）和/或为操作系统项目做出贡献的潜在动机。一般来说，OSVC的用户应该考虑在操作系统方面“为什么”和“操作什么”。研讨会参与者列举了几个原因：软件片段已经存在，代码库保持开放，或者目标是集成或利用现有的操作系统社区。作为补充，文献中提到了通过操作系统参与开放式协作和创新，例如共享资源、利用外部想法、节省成本或协同专业知识[4, 9]。因此，这些好处甚至为光贡献者带来了优势。它们可以影响操作系统组件的进一步发展和功能，并对组件和潜在用途有更好、更详细的了解。

**设想的好处：**它要求用户定义操作系统集成的好处，这使他们能够客观地衡量操作系统的使用是否成功。这些好处包括代码库的免费和可维护的可用性，以及现有组件的快速集成。这些可能会降低开发成本或提供公司边界之外的外部创新[57]。

### 4.3 价值评估

价值评估由五个构建块组成，支持用户根据几个标准评估和选择操作系统组件。五个构建模块中的三个侧重于选择、使用和集成操作系统组件，包括评估如何利用操作系统组件、哪些功能对公司有价值以及如何正确集成。另外两个构建块指导用户评估操作系统项目和围绕它构建的社区的质量。

**价值功能：**它强调识别为公司流程、产品和服务增加价值的操作系统项目和/或操作系统组件功能。这些服务的范围可以从特定领域的服务到一般应用程序（例如操作系统数据库）。在我们的项目中，跟踪和追踪等服务是有用的操作系统组件。这是一种商品化的服务，客户不愿意为此付费。

**利用率选项：**它描述了操作系统组件的利用率选项。根本问题是“如何处理”操作系统组件。最主要和最直接的利用选项是在开发过程中将现有的操作系统组件集成到软件中。公司应考虑集成操作系统的内部原因及其替换当前使用的软件和/或开发新软件的好处。

**功能集成：**接下来，问题是“在哪里”集成操作系统组件。它在哪里增值？整合起来有多容易？用户必须评估操作系统组件的可集成性，并分析围绕它的成本效益比。主要问题是，与自己开发功能相比，开发人员集成它是否需要花费时间。例如，可能需要雇佣负责管理操作系统任务的全职员工。

**操作系统项目的质量：**在选择用于重用和集成的操作系统组件时，一个重要的问题是操作系统项目质量。在这里，研讨会的参与者强调，他们通常会根据不同的指标来筛选一个操作系统项目的承诺功能（例如，它是否解决了问题）和质量，例如GitHub上的星级、可靠性、提交者数量、开放票与封闭票之比以及项目分叉频率。可以说，其他指标很难

例如，评估代码或文档的质量。评估操作系统的质量可能比专有软件的质量更复杂[38]。

**社区：**操作系统组件的一个核心要素是它们所源自或嵌入的社区。从公司的角度评估操作系统社区应包括社区治理的活动/成熟度和可信度。例如，使用操作系统组件的一个常见陷阱是操作系统项目可能会消失或被放弃[58]。然而，通常可以继续按原样使用源代码，将维护和开发留给用户。活跃的社区也是操作系统项目持续创新的纽带[16]。

#### 4.4 价值捕获

价值捕获类别包括四个构建块，支持用户评估操作系统组件的使用权限、风险、成本和收益。这一步对于评估组件的业务契合度尤为重要，特别是与已经定义的价值主张相关的部分（见第4.2节）。

**许可：**许可是操作系统分发的重要组成部分，决定了操作系统组件的使用[17]。有各种许可证可供选择，附带不同的权利、义务和需要遵守的禁令，因为可能会产生法律后果[7, 17, 59]。例如，*copyleft*许可证要求操作系统组件的用户遵守与初始操作系统组件（例如GNU）相同的许可协议[34, 58]。仔细分析许可证（选项）至关重要。一位行业专业人士告诉我们，有些许可证根本是不允许的，因为它们不能“自由”使用这在文献中被称为“不可接受的许可条款”[59]。**风险：**利用操作系统组件会遇到各种风险。例如，虽然操作系统为公司提供了获取新软件组件的合理途径，但依赖各种项目可能会导致可传递的依赖关系，这些依赖关系将连接到根库的依赖关系纠缠在一个依赖关系层次结构中[60]。其他风险包括操作系统项目可能被放弃或中止，这意味着用户必须确保控制和维护操作系统组件本身的能力。另一个问题是，用户必须确保遵守许可协议，防止非法使用操作系统组件和担心责任。

**活动和成本：**操作系统组件免费提供；然而，在公司的产品/服务中选择、集成和维护这些组件需要IT部门的时间和成本资源。此外，我们的一位研讨会参与者回忆起一次不成功的投标，即为现有的操作系统项目捐款并成为社区的一部分。

**收入：**只有当公司通过支持操作系统的产品或服务赚钱或省钱，或者从长期创新潜力中受益时，使用操作系统才有商业意义[55]。如果一家公司希望基于操作系统组件产生收入（而不是“仅”节省成本），概述潜在的收入模式至关重要。当然，这也包括一些问题，例如决定用操作系统组件构建的软件的一部分是否也可以作为操作系统组件/项目使用。如果是这样，Rajala[34]概述了各种操作系统收入模式，如亏损领先、配件化或支持销售。如果软件是专有提供的，则适用软件应用程序的典型收入模式。

**表1：**使用OSVC的指导问题

建筑砌块		指导性问题
价值主张	动机	<ul style="list-style-type: none"> <li>你为什么想使用开源？</li> <li>它应该解决哪些问题（例如，缺少专业知识/资源）？</li> <li>为什么你想为现有的操作系统项目做出贡献（例如，设置自己的标准）？</li> </ul>
	预期收益	<ul style="list-style-type: none"> <li>您获得了哪些好处（例如，减少时间、节省成本、提高质量）？</li> <li>为什么它对您有价值？</li> </ul>
价值评估	价值的功能	<ul style="list-style-type: none"> <li>您想使用哪些功能？</li> </ul>
	使用选项	<ul style="list-style-type: none"> <li>您在哪里集成功能（例如，在产品中）？</li> <li>这些利用方案是否具有战略相关性？                             <ul style="list-style-type: none"> <li>它是否支持战略项目？</li> <li>这是新报价的基础吗？</li> <li>它使商品自动化了吗？</li> <li>它是否减少了内部遗留问题？</li> </ul> </li> </ul>
	功能整合	<ul style="list-style-type: none"> <li>如何整合这些功能？</li> <li>它们现在可用吗？</li> <li>您是否会在其之上/周围构建服务（例如垂直整合）？</li> </ul>
	操作系统项目质量	<ul style="list-style-type: none"> <li>功能的的质量如何？</li> <li>源代码的质量如何？</li> <li>它的表现如何？</li> <li>它安全吗？</li> <li>测试是否集成？</li> <li>是否包括持续集成和部署？</li> <li>版本控制是否透明？</li> <li>文档的质量如何？</li> </ul>
	社区	<ul style="list-style-type: none"> <li>来自谁（包括操作系统项目的传递依赖关系）？</li> <li>社区/OS项目基础值得信赖吗？</li> <li>是否存在潜在的利益冲突？</li> <li>社区是如何治理的？</li> <li>谁决定整合/接受什么？</li> <li>他们如何确保质量？</li> <li>社区是否活跃/成熟？</li> </ul>
价值捕获	许可选项	<ul style="list-style-type: none"> <li>您有哪些许可选项？</li> <li>它们是否符合您的利益（例如，利用率）？</li> </ul>
	风险	<ul style="list-style-type: none"> <li>涉及哪些风险？</li> <li>是否存在与其他操作系统项目的依赖关系？（传递依赖关系）？</li> <li>你能处理版本控制/更改吗？</li> <li>如果操作系统项目被删除：你能维护/开发它吗？</li> <li>你准备好降低这些风险了吗？</li> </ul>
	活动和成本	<ul style="list-style-type: none"> <li>需要哪些活动才能使其发挥作用？</li> <li>搜索所需功能</li> <li>根据质量和社区可信度等选择操作系统项目</li> <li>你要花多少钱？</li> <li>您是否有操作系统预算（例如，投资于二手社区）？</li> </ul>
	收入	<ul style="list-style-type: none"> <li>你将如何用它赚钱？</li> <li>你将如何用它省钱？</li> </ul>

## 4.5 评估

OSVC在所有四个研讨会结束时都进行了评估，导致在迭代之间不断进行，最终版本如**图2**所示。我们询问了所有参与者的总体印象，构建块是否可以理解，它们是否涵盖了所有基本方面，以及构建块和/或子类别是否多余。所有参与者都喜欢OSVC与概念表的结合，因为它们易于理解，外观直观，可以快速采用。没有一个构建块被认为是不必要的，参与者也没有提出额外的构建块。关于OSVC的适用性，每个研讨会都分析了一个特定的操作系统项目，以使用和评估OSVC来挑战其结构、构建块和指导。操作系统场景表明，OSVC有效、有用，并且没有显示出任何逻辑不足。研讨会参与者背景的多样性（计算机科学、商业、研究和从业者）确保了对操作系统使用和集成的商业和IT观点的评估。在这里，我们验证并添加了新的输入，涵盖了选择操作系统组件时需要考虑的必要方面。

项目内部研讨会（研讨会1和2）强调了使操作系统有形化的好处。这种透明度简化了行业合作伙伴的协作和教育，并使共同理解成为可能。第三次研讨会由两名IT/OS专家参加，详细介绍了构建块，并补充了另外12个子类别。在这里，我们添加了三个额外的组件质量评估标准“版本控制”、“测试”和“CI/CD管道”，并添加了构建块风险、活动和成本的所有子类别。这些补充尤其有价值，因为它们确保了非操作系统专家“至少”在操作系统组件的评估和选择过程中分析这些标准，从而降低了操作系统使用的风险。特别是，像“传递依赖关系”和“可以删除开源项目”这样的子类别是必须考虑的，因为操作系统的使用会导致组件依赖关系。考虑到停止操作系统开发和支持的最坏情况，公司必须在内部承担这些任务和责任。因此，这些标准确保了操作系统集成不会被轻视。第四次研讨会与物流行业的IT和市场领导者的业务专长一起确认了最终的子类别。

总之，OSVC被认为是专业、可理解和适用的。没有提出进一步的构建块建议，最终的子类别被评估为足以进行首次操作系统使用/组件筛选。主要功能是一眼就能看到、设计和调整操作系统战略和操作系统集成的最相关方面，确保跨专业知识的一致性（特别是业务和IT视角）。

## 5 贡献、限制和未来工作

本文开发了OSVC来评估集成和/或贡献操作系统的商业价值，并为两者选择最适用的操作系统项目。在补偿中，我们努力设计一个画布，帮助跨学科团队（特别是专注于

协调业务和IT对操作系统的看法），讨论并决定操作系统集成潜力是否符合公司战略。

OSVC在研究和实践方面做出了各种贡献。我们的工作为**研究人员**提供了一种指导操作系统组件结构化分析和评估的工具。随后，研究人员可能会使用我们的画布或构建块的底层结构来扩展、改进或使其适应特定行业。例如，OSVC可以明确地针对特定的许可选项、工业部门或技术（例如，工业4.0或物联网设备的操作系统组件）进行定制。画布概述了操作系统的价值组成部分，通过工具指导和要求跨学科合作来扩展当前的研究状态，以防止片面的操作系统理解。

**从业者**可以使用我们的工具来评估操作系统的价值、操作系统组件的集成和选择，以及“轻松”为项目/社区做出贡献的决定。画布的一个显著优势是在“一张纸”上提供浓缩的信息。使用OSVC，从业者不会忽视操作系统评估和操作系统组件选择和集成的基本方面。例如，OSVC包括重要的操作系统方面，如操作系统组件质量和许可，并迫使用户考虑潜在的动机、必要的活动和相关成本。此外，由于我们有意从跨学科的角度看待操作系统元素，即IT和业务，因此OSVC充当了学科之间的桥梁。

当然，我们的工作受到**限制**。首先，我们只考虑了现有文献语料库的快照，以指导其严格设计OSVC。然而，整合更多的数据库和更广泛的文献样本可以为新的构建块或进一步的构建块改进提供见解。此外，我们还从行业专业人士和计算机科学家那里了解了我们的项目环境。通过整合进一步的专家反馈，可以评估该工具的普遍适用性。

计划中的**未来工作**包括对OSVC进行额外测试，并不断检查其适用性和范围的适当性。例如，随着越来越多的从业者进入该项目，OSVC可以应用于更多场景，拥有更广泛的行业专家领域。相应地，未来的工作应考虑通过额外的研究方法对构建块进行三角测量。虽然目前OSVC是基于狭窄的文献样本和研讨会构建的，但通过深入访谈收集专家反馈，甚至通过问卷定量收集反馈，可以在未来大大提高每个构建块的有效性。此外，OSVC预计将得到额外的操作系统组件评估支持的补充，更详细地解决子类别问题，并允许对其进行定量评估。这将支持从业者做出正确的操作系统组件评估和选择决策。此外，目标是将OSVC和附加分析功能集成为IT工具支持的动态画布[61]。

## 致谢

“硅经济物流生态系统”项目由联邦交通和数字基础设施部资助。

## 工具书类

1. Bharadwaj, A., El Sawy, O.A., Pavlou, P.A., Venkatraman, N.: 数字商业战略: 迈向下一代洞察。《管理信息系统季刊》, 第37卷, 471-482 (2013)。doi:10.25300/MISQ/2013/37:2.3
2. Veit, D., Clemons, E., Benlian, A., Buxmann, P., Hess, T., Kundisch, D., Leimeister, J.M., Loos, P., Spann, M.: 商业模式。信息系统研究议程。商业与信息系统工程, 第6卷, 45-53 (2014)。doi:10.1007/s1259-013-0308-y
3. Harutyunyan, N., Riehle, D., Sathya, G.: 企业开源的行业最佳实践。《第53届夏威夷国际系统科学会议论文集》(2020年)。doi:10.24251/HICSS.2020.716
4. Link, G., Qureshi, S.: 开源社区在发展中的作用: 对政府的政策影响。《第51届夏威夷国际系统科学会议论文集》(2018)。doi:10.24251/HICSS.2018.302
5. 奥: 构建开源ERP销售宣传——寻找解释性曲目。发表于: 第八届地中海信息系统会议论文集 (2014)
6. Krogh, G.von, Haefliger, S., Spaeth, S., Wallin, M.W.: 胡萝卜和彩虹: 开源软件开发中的动机和社会实践。《管理信息系统季刊》, 第36卷, 649-676 (2012)。doi:10.2307/41703471
7. Harutyunyan, N.: 为什么以及如何管理开源供应链? 《计算机》, 第53卷, 第77-81页 (2020年)。doi:10.109/MC.2020.2983530
8. Fuerstenau, D., Rothe, H., Baiyere, A., Schulte-Althoff, M., Masak, D., Schewina, K., Anisimova, D.: 数字平台的生长、复杂性和生成性: Otto.de的案例。载于: 第40届国际信息系统会议论文集 (2019)
9. Morgan, L., Feller, J., Finnegan, P.: 开源创新网络: 探索高和低密度模型。来源: PACIS 2012 (2012)
10. Harutyunyan, N., Riehle, D.: FLOSS治理中组件批准的行业最佳实践。发表于: 《2020年欧洲公共政策论坛论文集》(2020年)
11. Atiq, A., Tripathi, A.: 开源软件开发人员的货币奖励。收录: 第35届国际信息系统会议论文集 (2014)
12. Pitts, C.: 混合开源公司的产品在商业上是如何成功的? 《南方信息系统协会会议论文集》(2014)
13. Maki Asiala, P., Matilasi, M.: 开源组件的质量保证: 集成商的观点。第30届国际计算机软件与应用年会 (COMPSAC'06), 第2卷, 第189-194页 (2006年)。doi:10.109/COMPSAC.2006.153
14. Steffen, B. 问为什么。第十届形式化方法应用国际研讨会, ISoLA 2021 (2021)
15. Shaikh, M.: 开源中的数字混合: Linux中的混合危机。在: Nagle, F., Wissel, J., Zaggl, M., Lifshitz Assaf, H., Shaikh, M., Tambe, P., Wang, S. (编辑) 开源软件开发和组织 (2019)

16. Riehle, D.: 商业开源商业模式。会议: 第十五届美洲信息系统会议记录, 第卷 (2009年)
17. Riehle, D.: 开源的创新。《计算机》, 第52卷, 第59-63页 (2019年)。doi:10.109/MC.2019.2898163
18. Lakka, S., Varoutas, D., Martakos, D.: Oss对软件市场的影响——评估。载于: MCIS 2009会议记录 (2009)
19. Osterwalder, A.: 商业模式本体论。设计科学方法中的一个命题 (2004)
20. Osterwalder, A., Pigneur, Y.: 商业模式生成。远见者、游戏规则改变者和挑战者手册。威利 (2013)
21. Avdiji, H., Elikan, D.A., Missonier, S., Pigneur, Y.: 视觉探究工具的设计理论。信息系统协会杂志, 第21卷, 695-734 (2020)。doi:10.17705/ljais.00617
22. Reijnen, C., Overbeek, S., Wijers, G., Sprokholt, A.: 数字化转型的共同愿景: 运营模型画布方法的编码。第26届欧洲信息系统会议论文集 (2018)
23. Elikan, D.A., Pigneur, Y.: 品牌识别的视觉查询工具。第52届夏威夷国际系统科学会议论文集 (2019)
24. Hunke, F., Seebacher, S., Thomsen, H.: 请告诉我该怎么做——在数据丰富的服务系统中引导关键活动的编排。In: Hofmann, S., Müller, O., Rossi, M. (编辑)《数字化转型设计》。与公民和工业共同创造服务, 第426-437页。施普林格国际出版社, 商会 (2020)
25. Schoormann, T., Behrens, D., Fellmann, M., Knackstedt, R.: 《On Your Mark, Ready, Search: 一个构建信息系统文献检索策略的框架》。《第16届经济新闻国际会议论文集》 (2021年)
26. Möller, F., Schoormann, T., Otto, B.: 谨慎-正在构建的原则-开发设计原则的视觉查询工具。载于: 钱德拉·克鲁斯, L., 塞德尔, S. (主编)《2021年可持续发展》, 223-235 (2021)
27. Lecuna, A., Thoring, K., Mueller, R.: 概念弧: 为模糊概念设计视觉画布。第52届夏威夷国际系统科学会议论文集 (2019)。doi:10.24251/HICSS.2019.068
28. Steffen, B., Möller, F., Nowak, L.: 物流行业的变革者——使物流公司能够通过数字平台脱颖而出。第55届夏威夷国际系统科学会议论文集 (2022)
29. Hevner, A.: 设计科学研究的三个周期观。斯堪的纳维亚信息系统杂志, 第19卷, 87-92 (2007)
30. Schryen, G., Kadura, R.: 开源与闭源软件: 迈向安全衡量。In: Shin, S.Y., Osowski, S. (编辑) 2009年ACM应用计算研讨会论文集-SAC 09, 第2016-2023页。ACM出版社, 美国纽约, 纽约 (2009)。doi:10.1145/1529282.1529731
31. Fitzgerald, B.: 开源软件的转型。《管理信息系统季刊》, 第30卷, 587-598页 (2006年)。doi:10.2307/25148740
32. Sarraf, M., Rehman, O.M.H.: 基于软件质量特征的开源软件选择的实证研究。工程软件进展, 第69卷, 1-11 (2014)。doi:10.1016/j.advengsoft.2013.12.001

33. Aberdour, M.: 在开源软件中实现质量。IEEE软件。 , 第24卷, 第58-64页 (2007年)。 doi:10.109/MS.2007.2
34. Rajala, R., Nissilä, J., Westerlund, M.: 开源软件收入模式选择的决定因素。《第14届欧洲信息系统会议论文集》, 第1839-1850页 (2006年)
35. Okoli, C., Nguyen, J.: 自由和开源软件的商业模式: 来自德尔菲研究的见解。《第21届美洲信息系统会议论文集》 (2015年)
36. Stallman, R.: GNU操作系统和自由软件运动。收录: DiBona, C., Ockman, S., Stone, M. (编辑)《开源》。来自开源革命的声音, 第31至38页。O'Reilly&Associates, 塞瓦斯托波尔 (1999)
37. Tiwana, A.: 平台生态系统中的进化竞争。信息系统研究, 第26卷, 266-281 (2015)。 doi:10.287/iser.2015.0573
38. Asundi, J., Carare, O., Dogan, K.: 商业公司为什么要公开其产品的源代码? 载于: MCIS 2009会议记录 (2009)
39. Riehle, D.: 单一供应商开源公司。《计算机》, 第53卷, 第68-72页 (2020年)
40. Sherlock, J., Muniswamaiah, M., Clarke, L., Cicoria, S.: 开源软件的机会和风险。arXiv预印本arXiv:1812.1697, 第 (2018) 卷
41. Morgan, L., Finnegan, P.: 超越自由软件: 探索战略性开源的商业价值。《战略信息系统杂志》, 第23卷, 226-238 (2014)
42. Gentemann, L., Termer, F.: 开源监视器。2019年学习
43. Renner, T., Vetter, M., Rex, S. 和 Kett, H.: 开源软件。潜力和经济。弗劳恩霍夫研究所。斯图加特2005, 卷 (2005)
44. Ven, K., Verelst, J., Mannaert, H.: 你应该采用开源软件吗? IEEE软件。 , 第25卷, 第54-59页 (2008年)
45. Poepelbuss, J., Lubarski, A.: 模块化画布——一个可视化服务模块化潜力的框架。发表于: 第十四届经济新闻国际会议论文集 (2019)
46. March, S.T., Smith, G.F.: 信息技术的设计与自然科学研究。决策支持系统, 第15卷, 251-266 (1995)。 doi:10.1016/0167-9236 (94) 00041-2
47. Chengalur Smith, I., Nevo, S., Demertzoglou, P.: 开源基础设施技术商业价值的实证分析。《信息系统协会杂志》, 第11卷, 第3期 (2010年)
48. 韦伯斯特, J., 沃森, R.T.: 分析过去为未来做好准备: 撰写文献综述。管理信息系统季刊:《管理信息系统》, 第26卷, xiii-xxiii (2002)
49. Niederman, F., March, S.: “理论透镜”概念: 我们都知道它意味着什么, 但我们都知同样事情吗? 公社。信息系统协会。 , 第44卷, 第1-33页 (2019年)。 doi:10.17705/1CAIS.04401
50. Avdiji, H., Elikan, D.A., Missonier, S., Pigneur, Y.: 集体解决结构不良问题的设计工具。《第51届夏威夷国际系统科学会议论文集》 (2018)。 doi:10.24251/HICSS.2018.053
51. Iivari, J., Rotvit Perlth-Hansen, M., Haj Bolouri, A.: 设计原则最小可重用性评估建议。《欧洲信息系统杂志》, 第1-34卷 (2020年)。 doi:10.1080/0960085X.2020.1793697



52. Schoormann, T., Möller, F., Hansen, M.R.P.: 研究人员如何（重新）使用设计原则：累积研究的归纳分析。收录：钱德拉·克鲁斯, L., 塞德尔, S. (编辑) 《2021年的命运》, 188-194 (2021)
53. Thoring, K., Mueller, R., Badke Schaub, P.: 作为研究方法的研讨会：通过第53届夏威夷系统科学会议论文集设计和评估人工制品的指南。夏威夷：美国 (2020)。doi:10.24251/HICSS.2020.620
54. Ørngreen, R., Levinsen, K.: 研讨会作为一种研究方法。电子学习杂志, 第15卷, 70-81 (2017)
55. Osterwalder, A., Pigneur, Y., Tucci, C.L.: 澄清商业模式：概念的起源、现在和未来。信息系统协会通讯, 第16卷, 1-25 (2005)。doi:10.17705/1CAIS.01601
56. Avesa, P., Haefliger, S., Rossi, A., Baden-Fuller, C.: 从商业模式到商业建模：模块化和操纵。商业模式与建模, 第33卷, 151-185 (2015)。doi:10.108/S0742-33222015000000033022
57. Feller, J., Finnegan, P., Hayes, J.: 开源网络：商业模式和敏捷性问题的探索。第14届欧洲信息系统会议论文集 (2006年)
58. Colazo, J.A., Fang, Y., Neufeld, D.: 开源软件项目的开发成功：探索copyleft许可证的影响。《第十一届美洲信息系统会议论文集》 (2005年)
59. Gangadharan, G.R., Paoli, S.de, D'Andrea, V., Weiss, M.: 自由和开源软件中的许可证合规问题。MCIS 2008会议记录, 第2卷 (2008年)
60. Pashchenko, I., Plate, H., Ponta, S.E., Sabetta, A., Massacci, F.: 易受攻击的开源依赖：计算那些重要的依赖。ESEM 18: 第12届ACM/IEEE实证软件工程与度量国际研讨会论文集 (2018)
61. Steffen, B., Möller, F., Rotgang, A., Ryan, S., Margaria, T.: 走向活罐头。参见：Margaria, T., Steffen, B. (编辑) 《利用形式化方法的应用，验证和确认》，第95-116页。施普林格国际出版社，昌 (2021)