systems

MDPI

Article

Open-Source Collaboration and Technological Innovation in the Industrial Software Industry: A Multi-Case Study

Xiaohong Chen [1] and Yuan Zhou [2,*]

[1] School of Humanities and Social Science, University of Science and Technology Beijing, Beijing 100083, China; chenxiaohong@ustb.edu.cn
[2] School of Public Policy and Management, Tsinghua University, Beijing 100084, China
* Correspondence: zhou_yuan@tsinghua.edu.cn

Abstract: Open-source collaboration, as both an open and cooperative software development paradigm and a novel production model in the era of the industrial internet, plays a pivotal role in overcoming technological bottlenecks in the industrial software industry. However, previous studies have often treated open-source collaboration as a single unified concept and have not explored the specific types of open-source collaboration and their differential effects on technological innovation. To address these gaps, this study aims to answer two core research questions: (1) What are the different types of open-source collaboration models based on their characteristics? (2) How do these collaboration models influence technological innovation in the industrial software industry? Drawing upon four representative collaboration cases in the industrial software domain, this study conducts within-case and cross-case comparative analyses to propose a typological framework based on the dimensions of coreness and complementarity. The analysis identifies four distinct open-source collaboration models: (1) single-core with high complementarity, (2) single-core with low complementarity, (3) multi-core with high complementarity, and (4) multi-core with low complementarity. The formation of these models is shaped by three key factors: strategic intentions, resource endowments, and technological capabilities. Moreover, different collaboration types exert varied impacts on organizational characteristics, innovation strategies, and technological impacts. Theoretically, this study makes an original contribution by opening the “black box” of open-source collaboration and revealing the internal mechanisms through which it shapes innovation dynamics. Practically, the findings offer targeted insights for enterprises, policymakers, and open-source communities in selecting appropriate collaboration models that align with innovation goals, thereby supporting technological upgrading and ecosystem resilience in the industrial software industry.

Keywords: open-source collaboration; industrial software industry; technological innovation; multi-case study

check for updates

Academic Editors: Wen-Hsiang Lai and Sheng-Tung Chen

Received: 1 April 2025
Revised: 18 May 2025
Accepted: 22 May 2025
Published: 3 June 2025

Citation: Chen, X.; Zhou, Y. Open-Source Collaboration and Technological Innovation in the Industrial Software Industry: A Multi-Case Study. *Systems* **2025**, *13*, 433. https://doi.org/10.3390/systems13060433

1. Introduction

Technological innovation management systems are key to gaining competitive advantages in rapidly changing market environments [1], and they also serve as an essential support for improving innovation performance [2]. With the development of digitalization, the focus of corporate technological innovation is gradually shifting from individual products and services to more complex value propositions [3]. These complex value propositions depend not only on the R&D activities of a single enterprise but also require cross-industry and cross-field cooperation and innovation [4]. As a special asset of a company, the technological innovation management system encompasses various fields, including technology, production, processes, knowledge, experience, and organization [5]. Technological innovation capability is driven by scientific research [6] and is the process of developing, introducing, and applying new products, services, and production processes [7]. Technological progress not only comes from the combination of existing components [8] but also relies on a more comprehensive perspective and resources from multiple actors to drive breakthrough innovation [9].

In the modern process of technological innovation, open-source collaboration has gradually become one of the key factors influencing technological innovation management systems. Open-source collaboration breaks traditional company boundaries, facilitating the open flow and sharing of external technological resources and knowledge [10]. The flow of elements across industries generates relevant interrelations among different sectors [11], and due to the input-output relationships between industries, intellectual resources can also generate spillover effects within industrial clusters [12].The latest industrial revolution also means that even small enterprises can engage in global trade and cooperation through the development of information and communication technologies, artificial intelligence, and the internet [13], thus influencing cross-industry technological innovation. In the process of open-source collaboration, the characteristics of complementarity and coreness are particularly evident. The former refers to the mutual complementarity between different collaborators in terms of technology, knowledge, resources, etc. Complementary resources and capabilities can create unique value, promote learning, and generate new knowledge and innovations [14]. Coreness, on the other hand, reflects the leading role of certain core enterprises or technologies in the collaboration network. These two fundamental features combine to jointly promote the iteration and breakthrough of technological innovation management systems.

In the field of industrial software, technological innovation faces numerous inherent challenges, particularly in the context of cross-industry collaboration and knowledge integration. On the one hand, differences in standards across industries may hinder the effectiveness of collaboration. Knowledge barriers in cross-sector cooperation often stem from disparities in knowledge domains, experience, professional language, perspectives, organizational culture, and stakeholder interests [15]. These divergences can result in inconsistencies in technical protocols, data interface standards, and cybersecurity regulations, thereby impeding effective technological convergence. On the other hand, industrial software innovation frequently encounters incompatibilities in the integration of heterogeneous knowledge systems. Semantic barriers and pragmatic barriers are commonly observed in cross-industry cooperation. Semantic barriers arise from varying interpretations of terms and technical concepts, while pragmatic barriers reflect misalignments in actual implementation, engineering practices, and collaboration mechanisms [16]. These issues are further exacerbated by the intrinsic complexity of industrial software itself, which typically involves multi-source heterogeneous data, high real-time performance requirements, and deep integration between hardware and software systems. This complexity significantly raises the bar for system compatibility and amplifies the difficulty of achieving unified technological evolution across diverse industrial domains.

Given these challenges, open-source collaboration has emerged as a particularly necessary and effective mechanism for driving innovation in industrial software. Unlike traditional closed or bilateral models, open-source collaboration enables cross-organizational and cross-sectoral knowledge pooling through transparent codebases, community-based modular development, and decentralized governance structures. It reduces entry barriers for diverse participants, accelerates the convergence of heterogeneous standards, and fosters iterative innovation across industries. Moreover, open-source ecosystems provide a platform for absorbing and reconfiguring technological spillovers across industries [17]. In such ecosystems, contributors from different sectors can embed industry-specific know-how into shared software modules, thereby enhancing the transferability and scalability of innovations. Open-source practices also facilitate the accumulation of common protocols and interface standards through collaborative testing and peer review, helping to resolve semantic and pragmatic barriers in real-world industrial applications. Therefore, in the face of fragmented innovation paths, siloed industrial knowledge systems, and increasingly complex integration demands, open-source collaboration is not only beneficial but essential for accelerating industrial software innovation and achieving systemic compatibility in the era of intelligent manufacturing.

In this process, the characteristics of complementarity and coreness in open-source collaboration provide enterprises with more diversified innovation paths and resource support, especially when integrating cross-disciplinary knowledge and technologies required in industrial software development. By strengthening collaboration among various departments, the pervasive penetration of intelligence promotes the flow of knowledge elements [18], accelerates the recombination and creation of knowledge, and thus drives technological innovation [19]. Technological innovation based on industrial software applications can enhance the digital capabilities of production systems, optimize production costs, more accurately manage production processes, and improve the flow of production information, supported by big data analytics capabilities [20], digital transformation [21], and emerging mobile information technologies [22]. These developments further promote technological innovation in the manufacturing industry.

However, current research on the impact of open-source collaboration on the technological innovation management systems of industrial software enterprises still has the following limitations: First, the concept of open-source collaboration has not been deeply unpacked. Existing studies have acknowledged the positive impact of open-source collaboration on technological innovation, but tend to treat open-source collaboration as a monolithic construct. Previous studies have not explored the specific types of open-source collaboration and the lack of typological clarity hinders a nuanced understanding of how different collaboration configurations affect innovation processes. Second, existing studies have examined the role, definition, and characteristics of complementarity in the collaboration process [23], there remains a gap in examining how variations within open-source collaboration differentially influence the technological innovation of industrial software enterprises. Finally, there is a lack of sufficient case studies and empirical research on the influence of open-source collaboration on the technological innovation management systems of industrial software. Based on this, the research questions of this paper are as follows:

(1) What are the different types of open-source collaboration models based on their characteristics?
(2) How do these collaboration models influence the technological innovation in the industrial software industry?

This study makes distinct contributions by addressing the lack of typological clarity in the field of open-source collaboration within industrial software innovation. Unlike previous studies that often treat open-source collaboration as a monolithic concept, this research constructs a novel analytical framework that classifies collaboration types based on two fundamental dimensions: the number of core entities and the degree of complementarity. This dual-axis typology comprising single-core with high complementarity, single-core with low complementarity, multi-core with high complementarity, and multi-core with low complementarity models offers a systematic lens for differentiating open-source cooperation patterns. Furthermore, through multi-case analysis of global industrial software firms, the study advances existing scholarship by unpacking the organizational characteristics, strategic logics, and technological impacts associated with each collaboration type. Theoretically, it bridges gaps between ecosystem governance theory and innovation management in the open-source context. Practically, it provides actionable guidance for enterprises, policymakers, and open-source communities to align collaboration models with strategic innovation goals and resource configurations.

The structure of this study is as follows. Section 2 provides a systematic literature review, elaborating on the concept and characteristics of open-source collaboration, the technological innovation features of industrial software, and the impact of open-source collaboration on industrial software innovation. This chapter establishes the theoretical foundation for the study. Section 3 discusses the research design and methodology, explaining the rationale for adopting a multiple-case study approach, detailing the data collection process, and justifying case selection criteria. This chapter lays the groundwork for subsequent empirical analysis. Section 4 presents the case analysis and findings, offering a detailed examination of selected cases through both within-case and cross-case analyses. It classifies different types of open-source collaboration, assesses their impact on industrial software innovation, and synthesizes a refined research framework. Section 5 concludes the study by summarizing the key research findings, outlining theoretical contributions, providing practical implications for different stakeholders, and discussing research limitations and future research directions.

2. Literature Review

2.1. Open-Source Collaboration

Open-source collaboration, in a narrow sense, refers to the collaboration between various entities such as developers, companies, communities, or foundations in the co-development of open-source software [24]. In a broader sense, it is not limited to the open-source software domain but emphasizes an open, inclusive, and shared mode of collaboration among multiple entities. Based on the characteristics of open-source collaboration, this paper divides its connotation into two dimensions: the complementarity of collaborative attributes and the coreness of collaborative structure.

Complementarity was introduced by Teece [25], emphasizing that companies need to collaborate with external partners during the innovation process to gain competitive advantages by acquiring complementary resources or capabilities. Recent studies highlight that complementarity refers to the relationship between two entities, where the value of one entity increases through cooperation with the other. With the deepening of the integration of digital and physical industries, research on complementarity has gradually expanded beyond traditional material resource complementarity and capability complementarity to include multidimensional elements such as data complementarity, technological complementarity, and institutional complementarity. For example, in the Chinese industrial software ecosystem, the collaboration between Yonyou and Huawei Cloud offers a concrete example of technological complementarity. Yonyou, as a leading enterprise software provider in China, has strong capabilities in ERP, industrial MES, and financial management software tailored for manufacturing and service sectors. However, to meet the needs of state-owned and high-security sectors, Yonyou began adapting its core product suite to Huawei's Kunpeng architecture and openEuler operating system. Huawei Cloud provided the underlying cloud infrastructure and computing ecosystem, while Yonyou contributed application-level industrial software capabilities. This demonstrates technological and infrastructural complementarity, where Yonyou focused on business logic, application modules, and industrial scenarios, and Huawei supplied performance-optimized chips, OS, and cloud-native environments.

Complementarity is characterized by varying degrees of intensity, which stem primarily from the level of interdependence. Depending on the extent of this dependency, complementarity can be categorized into high and low levels [26]. High complementarity refers to a situation where the partners have significant differences in resources, capabilities, knowledge, and technology, and there is a strong dependency. When the value of X depends on Y, and X cannot achieve its value without Y, high complementarity exists. If X and Y depend on each other in a similar manner, a bidirectional high complementarity occurs [27]. For example, the collaboration between Microsoft and OpenAI exhibits typical high complementarity characteristics. The two parties promote mutual benefits through strategic cooperation in three aspects. First, Microsoft increases investment in the development and deployment of dedicated supercomputing systems to accelerate OpenAI's pioneering independent AI research. Second, Microsoft actively deploys OpenAI's large language models in its consumer and enterprise products, enhancing the digital technology experience of OpenAI's models through continuous simulation training. Third, Microsoft's Azure platform serves as OpenAI's independent cloud provider, promoting the application development of AI technologies and tools, including OpenAI's API and GPT-3. The highly complementary technological vision and strategy of both companies have significantly contributed to technological innovation and transformation in the AI field [28].

Low complementarity occurs when X still has value without Y, representing weak complementarity. When substitutes exist, the exit of one party does not undermine the value of the other. Thus, low complementarity occurs when X and Y create value together, but their dependency on each other is relatively weak. The cooperating entities exhibit some degree of complementarity, but their business segments are relatively independent, and there may be overlaps in capabilities, resources, and knowledge, making it easier for the cooperation to shift into competition or coexist in a co-opetition relationship. For example, the collaboration between Microsoft and Nokia is a typical case of low complementarity. Although they had some technological overlap, Nokia excelled in hardware manufacturing but lacked strong software development capabilities, as represented by the Symbian system, which fell behind in the mobile internet era. Microsoft, despite having the Windows Phone operating system, did not match market demand well, and their collaboration did not result in mutual benefits. Ultimately, it ended with Microsoft's acquisition of Nokia's mobile phone business [29].

Coreness, derived from the core-periphery structure theory in social network analysis [30]. It pointed out that network connections can be categorized into strong ties and weak ties, with core nodes having a greater influence due to their strong connections, while peripheral nodes are more dependent on the core nodes' resources for information transmission. For example, Kingdee Cloud offers a typical example of a single-core, high-complementarity collaboration model in China's enterprise and industrial software ecosystem. Kingdee, as the platform leader, provides the core PaaS layer, middleware, development environment, and application design standards. Numerous ecosystem partners—including IoT device providers, industrial app developers, consulting firms, and digital transformation integrators—develop domain-specific modules or services based on Ciangqiong's cloud-native infrastructure. These partners depend on Kingdee's standardized API, low-code development tools, and compliance framework to deliver industry-specific ERP, MES, and SCM solutions. In this structure, Kingdee serves as the core node. It controls the architectural backbone, sets security and interface protocols, and provides certification for ecosystem partners. This concentration of decision-making and technical influence illustrates high coreness, while the modularity of partner contributions reflects complementarity around the central platform.

Based on the number of cores, coreness is typically divided into single-core and multi-core. Single-core refers to a cooperative network where resources, decision-making, and full authority are concentrated in a single core node, with peripheral participants collaborating around this core node. In a single-core structure, the core node plays a dominant role in setting strategic directions, allocating resources, and disseminating information. This core node can be a company, individual, organization, or project, and it exerts relatively stronger control compared to others. This model is commonly applied in environments where resource concentration and decision-making efficiency are emphasized, and it often appears in scenarios such as supply chain management, large project coordination, and certain open-source software communities. For example, in the Linux open-source software community, although there are numerous developers and contributors, the core team and its technical committee hold a single-core position, controlling the project's direction, code, and community development [31]. Another example is the Apple-dominated supply chain model, where Apple, as the core node, has absolute control over parts manufacturers, assembly factories, and other stakeholders in the supply chain. Through centralized management, Apple ensures high levels of production control and market responsiveness [32].

Multi-core refers to a cooperative network with multiple core nodes, each independent but with high autonomy and decision-making power. While cooperation progresses, the distribution of rights and resources is relatively decentralized, and no single core node controls the entire network. In multi-core structures, the relationships between collaborators are relatively equal, and they work together toward common goals through organizational coordination mechanisms. This type of collaboration is often found in global cooperative R&D networks and distributed innovation networks. For example, Aksnes & Sivertsen [33] found that from 1980 to 2021, the proportion of international collaboration between countries increased from 4.7% to 25.7%. Specifically, in the case of the International Nucleotide Sequence Database Collaboration, Japan's DNA Data Bank (DDBJ), Europe's European Nucleotide Archive (ENA), and the United States' GenBank have cooperated to ensure standardized data formats and protocols, enabling global scientists to submit and access data according to specific standards [34].

2.2. Industrial Software Technological Innovation

Industrial software refers to software primarily used or specifically designed for the industrial sector, aimed at enhancing industrial enterprises' research and development, manufacturing, operations management, and equipment performance [35]. Its core essence lies in the encoding of industrial knowledge, encapsulating the knowledge used in industrial production in a graphical engineering form within software, thus improving industrial production efficiency [36]. What differentiates industrial software from general application software is its unique attributes. First, it has a higher technical threshold. Industrial software often requires the integration of industrial knowledge and technical experience into code and algorithms, which is particularly challenging in fields like manufacturing, aerospace, and energy. These industries require dealing with highly complex demands, with stringent requirements for physical modeling, process control, and simulation technologies. Second, industrial software requires greater research and development investment. For example, Siemens' R&D investment in 2023 amounted to €6.2 billion, accounting for approximately 8% of its total revenue, with funds primarily allocated to advancing high-end technologies such as industrial digitalization, artificial intelligence, and digital twins. Third, the development cycle is longer. Industrial software demands higher stability and reliability during development compared to general application software, requiring repeated testing and iteration. Typically, it takes 3–5 years to develop a complete and stable industrial software product. Furthermore, once the software is applied in an industrial setting, it must undergo validation in various scenarios before it can be considered true industrial software.

Generally speaking, industrial software can be classified into four categories: (1) Research and design software, which is primarily used for product design, research, and optimization to help enterprises improve product innovation and design efficiency, such as CAD, CAE, and PLM; (2) Production control software, which is used for real-time monitoring, production scheduling, and automation control in production processes to ensure high efficiency and consistency, such as MES, SCADA, and DCS; (3) Business management software, which refers to enterprise resource planning, production scheduling, and supply chain management, providing comprehensive management and decision-making support for enterprises, such as ERP, CRM, and SCM; (4) Operation and maintenance services software, which is used for the management, monitoring, maintenance, and optimization of equipment to improve equipment utilization, extend service life, and reduce failure rates through predictive maintenance, such as EAM and CMMS. Among these, the technological innovation difficulty of research and design software is the highest [37].

Among these, research and design software poses the greatest technological innovation challenge. This is due to its high reliance on domain-specific algorithms, numerical modeling accuracy, and deep integration with physics-based simulation and engineering logic. Software in this category often requires long development cycles, high-performance computing capabilities, and specialized knowledge from fields such as mechanics, materials science, or fluid dynamics. Furthermore, many of these core software tools are dominated globally by a few proprietary vendors, making innovation and substitution even more difficult. In contrast, production control software, while also technically demanding, typically involves more modular and interface-based development. Its focus is on real-time signal processing, process logic, and control system integration—domains where the maturity of sensor networks, PLCs, and industrial protocols provides more established development paths. Moreover, innovation in this category is often application-driven and can rely on incremental improvements to adapt to specific factory or process needs. In terms of business management or maintenance service software, they emphasize data integration, workflow optimization, and service customization in dealing with the challenges.

Moreover, beyond the differences in innovation difficulty across software categories, industrial software enterprises also face systemic innovation management challenges that span the entire value chain. The challenge in managing technological innovation in industrial software lies in its technological complexity and the involvement of multiple links, requiring collaboration across the upstream and downstream of the industry chain. Upstream, in addition to operating systems, open-source technologies, and other development environments and platforms, there is also a hidden upstream—industrial technology. The software implementation of industrial technology is highly challenging and requires talent with a combination of industrial and software expertise. However, the shortage of such interdisciplinary professionals prevents industrial enterprises and software companies from fully integrating their respective advantages. Downstream, referring to industrial equipment and other hardware terminals, the primary challenge lies in the limited generalizability of industrial software. Typically, the value of industrial software stems from its ability to standardize and streamline industrial knowledge into structured processes. However, due to the high degree of customization required by certain enterprises, a single industrial software solution often fails to accommodate the diverse production needs of different companies. As a result, industrial software companies face a dilemma: on one hand, they seek to establish technological barriers through proprietary, closed-source software; on the other hand, they lack the capability to independently develop comprehensive, universal industrial software solutions that can attract widespread industry adoption. Under these circumstances, adopting an open-source collaborative innovation strategy emerges as the optimal approach for upgrading the technological innovation management system of industrial software enterprises [38].

2.3. The Impact of Open-Source Collaboration on Technological Innovation in Industrial Software

Open-source collaboration plays an increasingly important role in modern industrial software technological innovation management systems. From an overall effect perspective, open-source collaboration helps improve innovation efficiency, reduce R&D costs, and accelerate market share growth. According to transaction cost theory, the value of business collaboration lies in reducing transaction costs, information search costs, and learning costs [39]. In industrial engineering, breakthroughs in innovation co-developed by participants with heterogeneous knowledge can significantly enhance the appeal of innovation to customers [40]. Through open-source collaboration, industrial software companies can not only quickly acquire external knowledge but also avoid the high costs associated with independent development. Companies can leverage each other's resources and capabilities to rapidly expand platform structures, enrich product matrices, and improve technological ecosystems, thereby enhancing market competitiveness. Additionally, existing research introduces the concept of digital legitimacy, which suggests that when innovation does not depend on a single actor, capital investors, policymakers, society, and regulatory bodies are more likely to accept the innovation [41]. Therefore, the support of diverse participants in industrial software innovation provides additional legitimacy for breakthrough innovations, making ecosystem-embedded breakthrough innovations develop faster than those supported by a single company.

The complementarity characteristic of open-source collaboration occupies a key position in technological innovation in industrial software enterprises, especially in terms of the complementarity of technology and resources. Complementarity spans across industry boundaries, meaning that different participants, knowledge bases, business models, and production logics are involved in the development, diffusion, and operation of the same technology [42]. At the same time, the innovation ecosystem is influenced by complementarity, as neglecting technological complementarity can lead to bottlenecks when forming the innovation ecosystem's value proposition [43]. Overall, companies providing complementary components participate as interdependent complementors in the value co-creation of industrial enterprises [44].

The coreness characteristic of open-source collaboration is another significant dimension influencing technological innovation in industrial software enterprises. Coreness not only represents the dominant position of certain key technologies or companies in collaboration but also reflects their leadership role within the entire innovation ecosystem. Core companies are capable of effectively integrating innovation resources from different domains, ensuring the continuity and breadth of technological breakthroughs. Typically, companies with a single core have greater influence in resource allocation, decision-making, and standard implementation [45]. When multiple collaborators are involved, particularly in cases where collaborators are easily replaceable, the core position and dominant role of the focal company tend to become more prominent. This also indicates that coreness in open-source collaboration possesses a certain level of dynamism, with the position of core companies potentially changing as collaboration relationships and market environments evolve. This dynamic shift in coreness further reveals the complexity of interdependencies and competition among companies in open-source collaboration.

3. Research Design

3.1. Research Method

This study adopts a multiple case study approach, selecting four different industrial software collaboration cases as research subjects to explore how various open-source collaboration models impact technological innovation of Industrial Software Industry. The primary goal is to provide theoretical insights and practical guidance for strategic collaboration in industrial software and, more broadly, in the information industry. The choice of the multiple case study method is based on several considerations. First, existing research on this topic lacks systematic and comprehensive theoretical exploration, and the inductive reasoning of case study methodology can bridge this research gap through in-depth description and analysis of cases [46]. In the intersection of technological innovation and open-source collaboration, although scholars have explored collaboration models in open-source communities [47] and the evolution mechanisms of open-source software ecosystems [48], there is still a lack of detailed case analysis regarding the specific mechanisms of industrial software collaboration models in technological innovation management systems. Therefore, the case study approach helps deepen the understanding of this topic at the theoretical construction level. Second, the research questions of this paper aim to explore how different open-source collaboration models specifically function, which involves unpacking the "conceptual black box" of open-source collaboration and revealing the operational mechanisms of different collaboration models. Case study methodology has unique advantages in identifying complex collaboration model characteristics and analyzing their underlying mechanisms [49]. Finally, the multiple case study method has a strong replication logic, which allows for the development of robust and generalizable theoretical propositions through within-case analysis and cross-case comparison. In technological innovation management research, multiple case study methods have been widely used to explore business collaboration models [50], innovation paths [51], and technology evolution [52]. By synthesizing and comparing different cases, this method can effectively extract common patterns, thus forming a theoretical framework that provides better explanatory power for real-world situations. To provide a more intuitive understanding of the study's structure, Figure 1 presents the overall research framework, which integrates theoretical foundation, case selection and collection, multi-case analysis, and research contributions.

3.2. Case Selection

This study selects four different industrial software companies involved in technological innovation collaborations as case studies. The selection criteria are as follows. First, the theoretical representativeness and alignment with research questions are essential. The selected cases must illustrate the four distinct types of open-source collaboration models proposed in the study—categorized by the degree of coreness and complementarity. Each case exemplifies a specific configuration, thereby enabling comparative analysis across collaboration structures. This approach aligns with the logic of theoretical sampling, as advocated by Eisenhardt [53], where cases are selected not for statistical generalizability but for their potential to illustrate theoretical constructs and refine emerging frameworks. Second, industry influence and completeness of information were key practical considerations. The cases involve multinational or industry-leading companies that play central roles in the industrial software domain. These cases are well-documented through official reports, press releases, academic studies, and enterprise publications, etc. ensuring both the accessibility and validity of data for cross-case analysis. Third, each case reflects a distinct stage or domain within the industrial software innovation chain, ranging from platform-based IoT services to simulation software integration and cloud-infrastructure partnerships. This variety enhances the scope and generalizability of the study's conclusions.

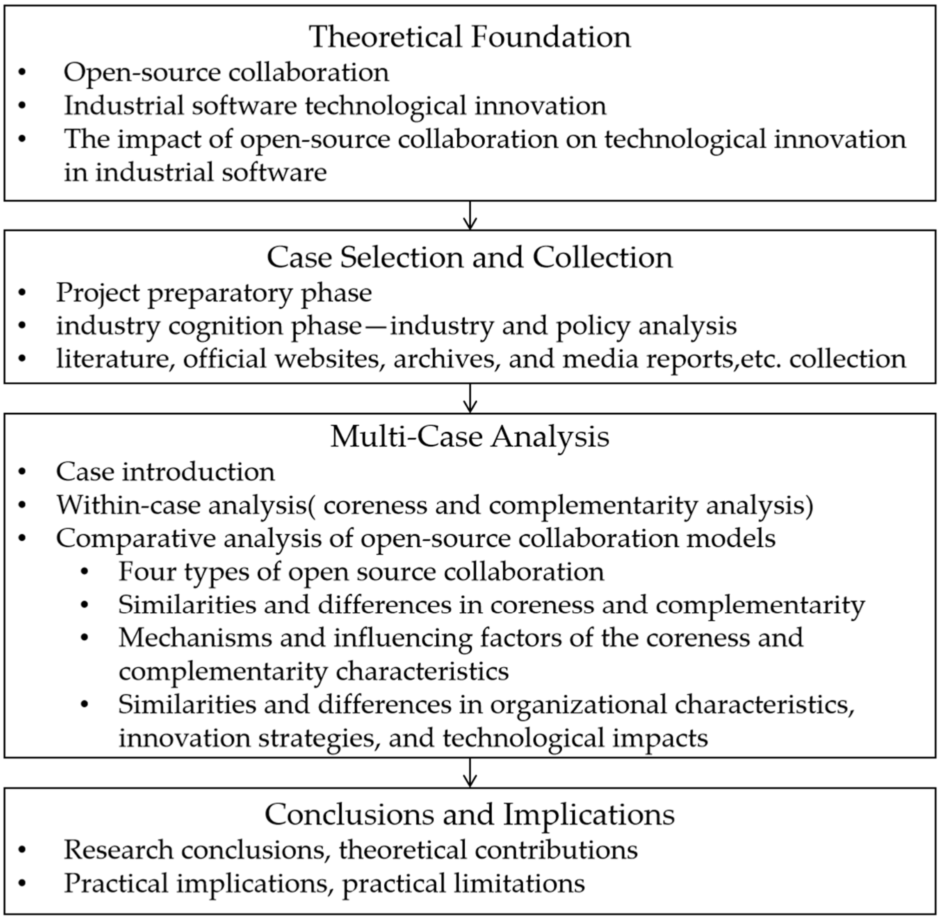

Figure 1. Research Framework of This Study.

Overall, the case selection criteria in this study align with the logic of theoretical sampling, while also ensuring the generalizability and credibility of the research conclusions. The basic information of the four cases is shown in Table 1.

Case A: Siemens and MindSphere Collaboration

This case represents a single-core with high-complementarity model. Siemens AG, founded in 1847 and headquartered in Munich, Germany, is a global leader in automation, digitalization, and smart manufacturing, ranked among the world's top 500 technology companies. It has long been committed to providing digital transformation solutions for the industrial sector, with business coverage spanning industries including industrial, home, infrastructure, healthcare, and transportation. MindSphere, launched by Siemens at the Hannover Messe in April 2016, is an industrial software based on a cloud-based open Internet of Things (IoT) operating system. It is a platform-as-a-service (PaaS) industrial IoT software, with its business model primarily based on providing cloud computing services such as server platforms or development environments. This case was selected for its maturity, standard-setting influence, and modular ecosystem construction—critical for illustrating centralized open-source collaboration in practice.

Table 1. Basic Information of Four Cases of Open-Source Collaboration. Among Industrial Software Companies.

	Case A	Case B	Case C	Case D
Basic Information	Siemens (Germany, est. 1847), a global leader in automation, digitalization, and smart manufacturing (Fortune Global 500)	Dassault Systèmes (France, est. 1981), a top industrial software firm specializing in 3D design and Product Lifecycle Management (PLM)	General Electric (GE) (USA, est. 1892), one of the world's largest industrial companies	Schneider Electric (France, est. 1836), a global leader in energy management and automation
	MindSphere (est. 2016), Siemens' industrial IoT platform	Exalead (est. 2000), a company specializing in enterprise search and big data analytics	Microsoft (est. 1975, USA), one of the world's most valuable tech companies	AVEVA (est. 1967, UK), a prominent industrial software company
Business Model	Siemens provides industrial automation and digitalization solutions, including MindSphere, a cloud-based industrial IoT platform (PaaS).	Dassault Systèmes specializes in 3D modeling, digital twins, and PLM, offering the 3DEXPERIENCE platform for digital solutions.	GE focuses on industrial internet, renewable energy, and smart manufacturing, with Predix, an industrial IoT platform for data analysis and optimization.	Schneider Electric operates in industrial automation and energy management, offering EcoStruxure, an AI-enabled industrial IoT solution for smart manufacturing and digital factories.
	MindSphere as IoT platform	Exalead offers enterprise search, NLP, data analytics	Microsoft offers cloud computing, AI, OS, enterprise software, etc.	AVEVA offers engineering design, digital lifecycle solutions, etc.
Technological Innovation Strategy	Siemens focuses on developing an open, scalable industrial IoT platform to address data silos, high maintenance costs, and low production efficiency.	Dassault enhances smart manufacturing and Industry 4.0 through PLM and big data technologies.	GE advances the industrial internet using cloud computing, AI, and big data to optimize energy and manufacturing efficiency.	Schneider Electric promotes smart manufacturing, industrial system intelligence, and sustainability through digital platforms.
	MindSphere enhances data analytics via its open IoT platform.	Exalead improves enterprise data management and search efficiency.	Microsoft builds a smart cloud and edge computing ecosystem.	AVEVA facilitates industrial digital transformation and production efficiency through software solutions.

Case B: Dassault Systèmes and Exalead Collaboration

This case is a classic example of single-core with low-complementarity collaboration, structured around an acquisition. Dassault Systèmes, founded in 1981 and headquartered in Paris, France, is a leading global industrial software company. Its core business and solutions involve 3D design, digital modeling, simulation, and product lifecycle management (PLM) software, deeply participating in the implementation of smart manufacturing, Industry 4.0, and digital twins. For instance, in the aerospace sector, Airbus uses Dassault's 3DEXPERIENCE platform to provide digital continuity from design to operation using a unified data model, helping Airbus in its digital transformation and laying the foundation for its aerospace ecosystem. In the automotive sector, Dassault provides support across multiple stages, from concept design to production operations, for renowned car manufacturers such as BMW and Tesla. Exalead is a French company founded in 2000, focused on enterprise search and big data analytics. Its core products and technologies are based on its powerful search engine and data management systems, covering enterprise search, big data analytics, semantic search, natural language processing (NLP), and cloud solutions, with industry applications spanning manufacturing, financial services, healthcare, and life sciences. This case illustrates how innovation consolidation through platform-centric logic can reflect low partner autonomy and unidirectional resource absorption, supporting this typology.

Case C: General Electric and Microsoft Collaboration

This collaboration showcases a multi-core with high-complementarity model. General Electric Company (GE), founded in 1892 and headquartered in Boston, USA, is one of the world's largest technology providers and business services multinational corporations, ranked in the Fortune Global 500. Its operations span over 100 countries and cover various fields such as aviation, energy, and healthcare, with particular leadership in industrial internet and renewable energy. Microsoft Corporation, founded in 1975 and headquartered in Redmond, Washington, is one of the highest-valued companies in the world, also listed in the Fortune Global 500. Primarily focused on the computer software industry, Microsoft's core business spans operating systems, office software, cloud computing, artificial intelligence, and hardware devices, significantly influencing the digital transformation of both individuals and enterprises. This case typifies equal strategic partnership and horizontal knowledge integration between tech giants across domains.

Case D: Schneider Electric and AVEVA Collaboration

This case illustrates a multi-core with low-complementarity model with limited technological convergence but clear market synergy. Schneider Electric SA, founded in 1836 and headquartered in Rueil, France, is a global leader in energy management and automation, with operations in over 100 countries. Its AI-powered, full-lifecycle industrial IoT solutions cover various sectors including residential, buildings, infrastructure, data centers, and industrial markets. AVEVA Group plc, founded in 1967 and headquartered in Cambridge, UK, is a prominent computer software company listed on the London Stock Exchange. AVEVA primarily provides full-lifecycle solutions and services for industries such as marine engineering, shipbuilding, oil, gas, power, and chemicals. This case helps demonstrate how open-source principles are adopted in contexts of low technical interdependence but strong brand and market linkage.

By selecting these four cases, the study ensures full coverage of the open-source collaboration typology, enhances external validity through industry diversity, and supports in-depth comparative analysis of innovation outcomes across collaboration models.

3.3. Data Collection and Analysis

The first step: the preparatory phase. The preliminary foundation of this research began in November 2022 with an open-source industrial software project led by a senior academician. Based on this project, the research team developed an initial understanding and analytical direction concerning core-dominant models (single-core) and multi-agent collaboration structures (multi-core) in the open-source development process. At the same time, under the guidance of this paper's lead professor, the team conducted extensive readings on theories of complementarity and core-periphery structures, which laid a theoretical foundation for further in-depth investigation in this field.

The second step: the industry cognition phase—industry and policy analysis. In 2023, the research team drafted policy recommendations related to open-source collaboration in industrial software. During this process, authoritative industry documents were systematically reviewed, including the Development Report of the China Industrial Technology Software Industry Alliance, the White Paper on the Development of China's Industrial Software, and the Beijing Software Cost Evaluation Technology Alliance Analytical Report, etc. These materials offered a comprehensive understanding of the current state and key challenges facing domestic industrial software in China. The analysis revealed that across the four major categories of industrial software, Chinese firms still lag significantly behind international leaders, particularly in open collaboration mechanisms, ecosystem development, and technical resource integration. Therefore, the study initially decided to focus its case analysis on internationally leading companies engaged in open-source collaboration, considering both the representativeness of the cases and the availability of transparent and accessible data.

The third step: the data collection phase—literature, official websites, archives, and media reports, etc. Since 2024, the research team has collected and reviewed a wide range of materials. From the literature side, 176 Chinese and international academic articles were reviewed using keywords such as “industrial software”, “complementarity”, “coreness”, “technological innovation”, and “innovation catch-up” through Web of Science, Google Scholar, and CNKI databases. On the enterprise and documentation side, in-depth study was conducted on the official websites and archives of Siemens MindSphere, Dassault and Exalead, General Electric, Microsoft Azure, Schneider Electric, and AVEVA. In addition, the team reviewed developer communities such as CSDN and technical news portals. Altogether, 36 pieces of highly relevant secondary data were collected and verified, resulting in a reference base of more than 200 materials including literature, news reports, and industry documents.

The fourth step: the data analysis phase. This study follows the logic of multi-case research, first conducting detailed within-case analyses of each collaboration case. Each case was classified using a dual-dimensional framework of coreness and complementarity to identify the corresponding open-source collaboration model. Based on this, a cross-case comparison was carried out to uncover the similarities and differences among the four typical collaboration models in terms of organizational structure, strategic intent, and technological innovation pathways. Although the study did not adopt software-assisted coding techniques, the analysis followed a structured thematic framework derived from the literature. Researchers conducted iterative reading, summary mapping, and cross-comparison across dimensions such as coreness, complementarity, and innovation outcomes. Through the combination of comparative and inductive reasoning, this study not only revealed the internal mechanisms behind different collaboration models but also provided empirical support and a reusable analytical structure for future policy recommendations.

4. Case Analysis

4.1. Case Introduction

4.1.1. Case A: Siemens and MindSphere Collaboration

Siemens launched the MindSphere industrial Internet of Things (IoT) platform primarily to address the demands of Industry 4.0 and the digital transformation of manufacturing, as well as to respond to the intensifying global competition in the industrial IoT market. Around 2015, General Electric (GE) launched Predix, IBM released Watson IoT, and Amazon AWS entered the industrial IoT market, prompting major technology companies to accelerate their strategies. As a global leader in industrial automation and digital solutions, Siemens sought to maintain its competitive advantage and drive its own digital transformation. In 2016, Siemens officially released MindSphere, aiming to create an open and scalable industrial IoT platform to solve pain points faced by enterprises such as data silos, high equipment maintenance costs, and low production efficiency. Since its launch, the MindSphere platform has gone through several key development stages. In 2016, the 1.0 version was released, focusing on providing device connectivity, data collection, and basic analytics functionalities. In 2017, Siemens partnered with Amazon AWS to provide public cloud services globally. In 2018, Siemens, in collaboration with several partners, founded the MindSphere World Association to promote the development of the IoT ecosystem. In 2020, Siemens partnered with IBM and Red Hat to optimize industrial IoT solutions using hybrid cloud technology. In 2021, a private cloud version of MindSphere was launched for the Chinese market to meet local enterprises’ needs for data security and compliance.

4.1.2. Case B: Dassault Systèmes and Exalead Collaboration

In June 2010, Dassault Systèmes acquired the French search technology company Exalead for €135 million. The acquisition aimed to enhance Dassault Systèmes' capabilities in search technology and information intelligence to support the development of its Product Lifecycle Management (PLM) solutions. Exalead was renowned for its powerful search platform and search engine technology, with its flagship product, CloudView, offering advanced search and information access functionalities for enterprises. After the acquisition, Exalead was integrated as one of Dassault Systèmes' brands and merged with the NETVIBES brand, providing customers with advanced search capabilities and semantic processing features. This integration enabled Dassault Systèmes to offer more convenient information access services to industries such as banking, retail, publishing, business services, life sciences, and consumer services.

4.1.3. Case C: General Electric and Microsoft Collaboration

Amid the wave of digital transformation, industrial enterprises face the challenge of efficiently utilizing data to improve operational efficiency and innovation capabilities. To address this demand, General Electric (GE) and Microsoft officially announced a strategic partnership in July 2016, planning to integrate GE's Predix industrial IoT platform into Microsoft's Azure cloud platform. Predix, developed by GE, is an industrial data analytics platform focused on device data collection and predictive maintenance, while Azure is a leading global cloud computing platform providing powerful computing capabilities and global coverage. The collaboration aimed to combine GE's industrial expertise with Microsoft's cloud computing power to create an integrated industrial IoT solution, helping enterprises better analyze asset data, optimize business processes, and enhance operational efficiency. As the collaboration progressed, both parties set clear development stages. By the end of 2016, the developer preview version of Predix on Azure was launched, allowing developers to test and provide feedback. In the second quarter of 2017, the enterprise version of Predix was officially released, offering commercial services to global industrial customers and supporting digital upgrades across industries.

4.1.4. Case D: Schneider Electric and AVEVA Collaboration

As a leader in energy management and automation, Schneider Electric is committed to providing interconnected products, controls, software, and services from the edge to the cloud. AVEVA, with many years of experience in the industrial software sector, offers full-lifecycle solutions covering engineering, operations, and maintenance. In 2018, the two companies formally announced a strategic collaboration, aiming to integrate Schneider Electric's energy management and automation technologies with AVEVA's industrial software capabilities. Together, they would provide integrated digital solutions to help enterprises optimize engineering design, enhance operational performance, and promote sustainability through intelligent means. As the collaboration advanced, the two companies deepened their cooperation around product and solution integration. Schneider Electric's EcoStruxure™ architecture seamlessly integrated with AVEVA's industrial software system, offering digital life cycle support from design to operations for clients across multiple industries.

4.2. Within-Case Analysis

4.2.1. Coreness Analysis

(1) Case A: Siemens and MindSphere Collaboration—Single Core

The collaboration in Case A involves three main levels of participants: Siemens, the ecosystem built around MindSphere, and manufacturing enterprises using MindSphere for digital transformation. This collaboration model centers around the MindSphere industrial

IoT platform and involves multiple stakeholders, from technology development to ecosystem collaboration, forming an open-source collaboration structure with Siemens at the core. Firstly, Siemens, as the developer and operator of the MindSphere platform, is responsible for core technology design, data management, and security standards, and determines the platform's functional expansion and technical standards. As the core architecture provider, Siemens not only sets the foundational framework but also exercises effective control over the entire ecosystem through standardization strategies, ensuring that all collaborators' development and integration follow its technical guidelines. Secondly, the MindSphere ecosystem is built by multiple industry participants, including (1) Hardware suppliers, such as sensor and industrial equipment manufacturers, providing foundational hardware support for the industrial IoT. (2) Software developers, including data analysis and AI algorithm providers, driving intelligent applications for industrial data. (3) Integration service providers, businesses offering specific application solutions based on MindSphere assisting the digital upgrades of various manufacturing industries. These collaborators develop based on Siemens' established technical standards, achieving interconnectivity and building various industrial IoT applications on the platform. Additionally, manufacturing enterprises, as the end-users of MindSphere, use the platform's real-time data analysis and intelligent optimization functions to enhance production processes and improve efficiency. For instance, companies like BMW and Airbus use MindSphere for real-time monitoring of manufacturing equipment, optimizing smart manufacturing processes to improve production quality and reduce operational costs. Since MindSphere adopts Siemens' technical architecture as the standardized core, all collaborators' development and integration on the platform must align with Siemens' technical framework. This open-source collaboration model, centered around a single core enterprise, places Siemens in a dominant position in the global industrial IoT platform development, profoundly influencing industry technical standards and ecosystem structures.

(2) Case B: Dassault Systèmes and Exalead Collaboration—Single Core

The collaboration in Case B involves Dassault Systèmes and Exalead. In 2010, Dassault acquired Exalead, taking full control after the acquisition, exercising absolute authority in technology development, company operations, and market positioning, establishing a single-core collaboration model with Dassault at the center. The primary goal of this strategic acquisition was to leverage Exalead's technological strengths in enterprise search and data analysis to enhance Dassault's 3DEXPERIENCE platform's data management and search capabilities to meet the demands of industrial digital transformation and smart manufacturing. The 3DEXPERIENCE platform is Dassault's core product lifecycle management (PLM) solution, designed to help companies manage data across the entire lifecycle from product design and manufacturing to maintenance. As industrial digitalization accelerates, the volume of data handled by PLM systems has dramatically increased, and enterprises require efficient data integration and search capabilities to support complex product management needs. Exalead's CloudView technology, with its powerful data integration and analysis capabilities, can extract structured and unstructured data from multiple data sources (such as documents, databases, and enterprise information systems) and perform intelligent searches. Dassault integrated Exalead's technology into the 3DEXPERIENCE platform, enabling users to access all product-related information on a single platform, thus improving data utilization efficiency. In terms of technical integration, Dassault not only introduced Exalead's core search engine technology but also standardized data interfaces, functional modules, and security protocols to ensure seamless integration of Exalead's solution into the 3DEXPERIENCE ecosystem. Furthermore, in terms of market and brand management, Dassault downplayed Exalead's independent brand, gradually incorporating it into the overall architecture of 3DEXPERIENCE and promoting it as an internal search and data management component. Thus, in this acquisition, Dassault Systèmes not only led the technological integration of Exalead but also deeply integrated business strategies and brand positioning, fully embodying a single-core dominant acquisition integration model.

(3) Case C: General Electric and Microsoft Collaboration—Multi Core

In Case C, both parties possess independent technological and market resources, with the collaboration focused on GE's Predix platform and Microsoft's Azure platform. Both are Platform-as-a-Service (PaaS) solutions, but their core technological advantages are complementary, with each maintaining a dominant position in the industrial IoT and cloud computing fields. Predix, developed by GE with billions of dollars in investment, is an industrial IoT platform focused on aggregating sensor, IoT, and factory equipment data to enable real-time monitoring, data collection, predictive maintenance, and production process optimization. It holds a dominant position in industrial applications in sectors such as manufacturing, energy, and aviation, granting GE core control over industrial equipment data management in this collaboration. Meanwhile, Azure, as Microsoft's cloud computing and big data analytics platform, possesses core competencies in cloud storage, artificial intelligence, machine learning, and other domains, thereby dominating the cloud computing and data management areas. Since the technical and market resources of the two parties do not overlap, the collaboration model is not one of acquisition and integration, but rather one that maintains independent operations through open interfaces and data sharing, promoting the collaborative development of industrial IoT based on technological interconnection. Therefore, Case C represents a multi-core collaboration structure in which GE and Microsoft retain leadership in their respective core technological fields.

(4) Case D: Schneider Electric and AVEVA Collaboration—Multi Core

Case D illustrates the multi-core collaboration model between Schneider Electric and AVEVA, where both companies maintain core leadership in their respective fields. Schneider Electric leads in energy management and automation control, while AVEVA holds core advantages in industrial software, particularly in design and operational management software. The two companies did not fully integrate their technologies or businesses, but instead operate their platforms and solutions independently while achieving collaborative synergy. In the collaboration model, Schneider Electric's EcoStruxure and AVEVA's industrial software operate as independent core systems. Customers can use EcoStruxure for energy and automation management or choose AVEVA software for design and operational optimization. While there are complementary functions in some areas, overall, the two systems evolve in parallel, maintaining their respective core functionalities and application scenarios, without being integrated into a single platform architecture. Additionally, the independence of branding and market positioning is a key characteristic of this collaboration. Schneider Electric and AVEVA each retain their brand identity and market positioning, maintaining a dominant position in the energy management, automation control, and industrial software markets. The collaboration has not led to the fusion or dilution of their market systems; rather, it has enhanced the overall value through synergistic cooperation while allowing each company to independently develop. Customers can clearly distinguish the core competencies of both companies and choose the most suitable solution based on their specific needs. Therefore, Case D fully embodies the parallel collaboration model under a multi-core architecture between Schneider Electric and AVEVA.

4.2.2. Complementarity Analysis

(1) Case A: Siemens and MindSphere—High Complementarity

The collaboration between Siemens, MindSphere, and its ecosystem partners demonstrates a high level of complementarity, primarily manifested in three areas: application boundary resources, development boundary resources, and social boundary resources [54].

Siemens optimizes platform functionality through these boundary resources, promotes multi-party collaboration, and enhances the openness and adaptability of the Industrial Internet of Things (IIoT) ecosystem. First, application boundary resources refer to the protocols and tools that allow third-party applications to interact with the platform. MindSphere is compatible with Siemens' proprietary protocols while supporting multiple open protocols, enabling different hardware suppliers and machine operators to collaborate in building IIoT monitoring solutions regardless of the devices used. Furthermore, MindSphere continuously expands its compatibility with cloud infrastructure, initially supporting only SAP data center hosting, then adding support for AWS (Amazon Web Services), Azure (Microsoft Cloud), and Alibaba Cloud, ultimately extending to on-premise installations. This flexible architecture significantly enhances data security and alleviates industrial enterprises' concerns about data compliance. Second, development boundary resources refer to the tools that support developers in creating and expanding applications. In April 2018, Siemens launched the MindSphere Java SDK, providing open development resources that enhanced third-party application compatibility. Additionally, MindSphere utilizes Cloud Foundry, an open-source PaaS, supporting cross-cloud vendor application deployment, allowing developers to freely choose infrastructure without being tied to a single cloud provider. This not only enhances cross-platform compatibility but also reduces developers' workload, optimizing the software development process. Third, social boundary resources refer to resources that facilitate knowledge transfer and coordinated development. Siemens established two core projects to enhance the complementarity of the MindSphere ecosystem. One is the MindSphere Partner Program (launched on 29 November 2017), which targets software developers and system integrators, offering webinars, training resources, and an open MindSphere app store to enhance partner software sales capabilities, while providing paid developer membership discounts to lower collaboration barriers and increase platform stickiness. Another initiative is the Hackathons, where Siemens organizes industry partners to participate in MindSphere platform hackathons, offering use cases, data integration, and hardware resources to promote multi-party collaboration, accelerate industrial application prototype development, and enhance the efficiency and quality of knowledge sharing. Overall, Siemens has driven the high complementarity of the MindSphere ecosystem through technical standardization, an open platform strategy, and multi-layered collaboration mechanisms, enhancing the innovation capacity and scalability of the IIoT ecosystem.

(2) Case B: Dassault Systèmes and Exalead Collaboration—Low Complementarity

Compared to the high complementarity between Siemens and MindSphere, the complementarity between Dassault Systèmes and Exalead is relatively low. Dassault Systèmes' core business focuses on 3D design, simulation, and Product Lifecycle Management (PLM), while Exalead primarily specializes in enterprise search and data analytics. Although Exalead's technology enhances Dassault' s PLM platform in terms of data management and search capabilities, it is not an essential component of the core architecture but rather an additional technology aimed at improving the user experience. In this collaborative relationship, Dassault's reliance on Exalead's technology is minimal. Dassault could easily invest in or acquire other companies to achieve similar data search and management capabilities. Exalead' s technology primarily enhances the data retrieval efficiency and user experience of the PLM system but does not bring fundamental innovation to Dassault' s core product architecture or business model. Given the significant disparity in the comprehensive strengths of both companies, Exalead functions more as a tool-like resource for Dassault rather than a deeply complementary partner. Therefore, this case reflects a functional supplement rather than a highly complementary collaboration model.

(3) Case C: General Electric and Microsoft Collaboration—High Complementarity

General Electric (GE)'s Predix platform has deep technological accumulation and resources in the industrial sector, allowing for efficient management of industrial production and operational data, thereby optimizing manufacturing processes, improving production efficiency, and reducing operational costs. By partnering with the world-leading cloud computing platform, Azure, Predix leverages Azure's advantages in AI, machine learning, hybrid cloud, and data sovereignty to further enhance equipment maintenance and production management capabilities, enabling the intelligent upgrade of industrial data. For Microsoft' s Azure platform, its core strengths lie in enterprise-level customer management, global cloud computing infrastructure, and data analytics capabilities. However, Azure has relatively limited market resources in the industrial IoT domain, while the massive industrial customer base and high-value data generated by the Predix platform are precisely what Microsoft lacked. By collaborating with GE, Azure quickly expanded into the industrial application market, using GE's customer base to build new IIoT solutions and further strengthening its cloud computing, data analytics, and operational capabilities. Therefore, the collaboration between Predix and Azure is a multi-core collaboration model based on complementary core technologies and mutual empowerment. It helps GE deepen its position in the industrial intelligence sector and also drives Microsoft's market expansion in the industrial IoT domain.

(4) Case D: Schneider Electric and AVEVA Collaboration—Low Complementarity

Firstly, the collaboration between the two companies is based on functional division rather than strong interdependence. The collaboration between Schneider Electric and AVEVA is one of functional division, where Schneider is responsible for automation control, real-time data collection, and energy management, while AVEVA handles engineering design and operational management. AVEVA's software does not rely on Schneider's hardware control, and Schneider's EcoStruxure platform can operate independently without AVEVA' s software. This means that the technologies between the two are not highly complementary; rather, they can function independently and only cooperate in specific application scenarios. Furthermore, both companies operate relatively independently, without forming exclusive dependencies. Even without Schneider' s EcoStruxure platform, AVEVA' s software can still serve clients in engineering design and operational management. Similarly, Schneider' s EcoStruxure can independently provide energy and automation management in industrial applications, without necessarily requiring AVEVA' s software. This relative independence means the collaboration is characterized by low complementarity, with neither company being the exclusive or absolute technological support for the other. Finally, the technological innovation collaboration model between the two is highly flexible. Schneider and AVEVA provide open interfaces and APIs in their collaboration, allowing customers to choose to combine products from both companies, but this combination is not mandatory. Customers can select either independent or integrated solutions according to their actual needs. This openness further reduces the technological complementarity between the two, making the collaboration more flexible, rather than deeply binding or integrated.

4.3. Comparative Analysis of Open-Source Collaboration Models

Based on the characteristics of the four cases analyzed above, this study further decouples the open-source collaboration models into four distinct types: single-core with high complementarity, single-core with low complementarity, multi-core with high complementarity, and multi-core with low complementarity. This decoupling is based on two dimensions: the degree of complementarity in open-source collaboration (high complementarity/low complementarity) and the number of core entities (single-core/multi-core), as shown in Figure 2. The analysis and summary of how these four open-source collaboration models impact the technological innovation management systems of industrial software companies are as follows.

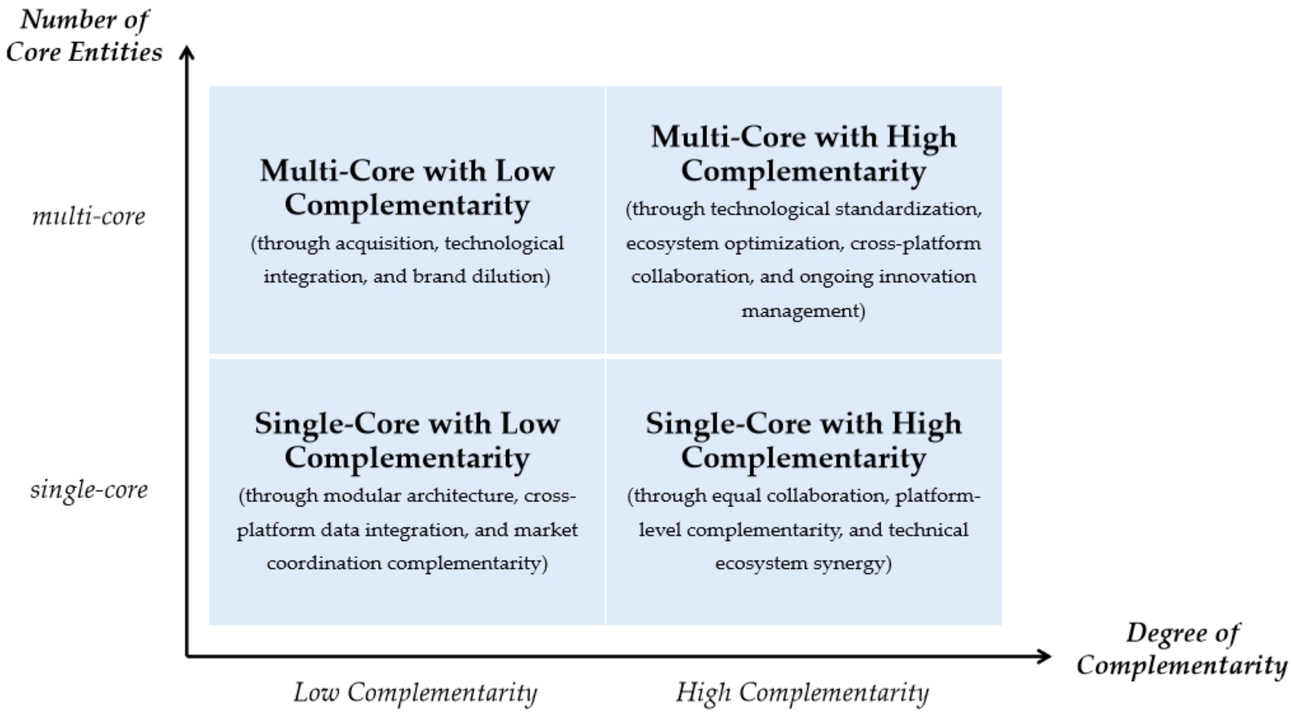

Figure 2. Matrix of Open-source Collaboration Models based on Degree of Complementarity and Number of Core Entities.

4.3.1. Single-Core with High Complementarity

The Single-Core with High Complementarity collaboration model refers to a scenario where multiple partners collaborate around a single core enterprise. The core enterprise controls key resources, standards, and decision-making authority, while peripheral products or enterprises rely on the core platform to provide essential functions. There is a high degree of interdependence between the parties, but the dependence of peripheral enterprises or products on the core enterprise is stronger [55]. This collaboration model has an incubation characteristic, where the core enterprise incubates platform-level products under its own system or standards [56]. In the Siemens-MindSphere ecosystem, this model continuously drives the upgrade of Siemens' industrial IoT technology innovation management system and ecosystem expansion through technological standardization, ecosystem optimization, cross-platform collaboration, and ongoing innovation management.

First, technological standardization ensures the stability and scalability of the ecosystem. Siemens established the core architecture, data interfaces, and security protocols for MindSphere, and all collaborators' product developments must comply with the established technical framework. This standardization not only reduces the issue of technological fragmentation, improves the compatibility of the platform, and allows third-party enterprises to seamlessly integrate into the MindSphere ecosystem, but also enhances the efficiency of technological coordination [57]. It enables industrial enterprises to quickly deploy IIoT solutions during their digital transformation without being restricted by underlying system adaptations.

Second, ecosystem optimization promotes the efficient integration of innovation resources. Siemens established a multi-tier collaboration system through the MindSphere Partner Program, involving sensor manufacturers, data analytics firms, and AI solution providers. Peripheral enterprises develop specific application modules around MindSphere, while Siemens ensures these modules integrate seamlessly into the overall architecture. This allows the platform to maintain the core enterprise's technological leadership while fully utilizing the expertise and innovation capacity of external companies, making the entire ecosystem more adaptable and scalable.

Additionally, cross-platform collaboration extends the flexibility of technology and market adaptability. MindSphere adopts an open architecture, making it compatible with leading cloud platforms such as AWS, Azure, and Alibaba Cloud. This strategy not only enhances the operability of enterprises across different IT infrastructures but also reduces the risk of vendor lock-in, thus providing more flexibility.

Finally, continuous innovation management ensures the long-term competitiveness of the ecosystem. Siemens uses developer incentive programs, technical competitions (such as hackathons), and other methods to encourage external developers to contribute new technologies and applications to the MindSphere ecosystem. This not only increases the platform's innovation capacity but also ensures the ecosystem continuously adapts to industry changes and market demands, thereby maintaining its leadership in the IIoT domain [58].

In this model, if the collaboration between the core enterprise and the platform under its standard system were to be disrupted for any reason, it could lead to incomplete functionality of the core enterprise, limiting core services, making it difficult for the platform to operate independently, and rendering the overall solution incomplete. Key functions may be missing, and the solution could require migration to a new integrated platform [59].

4.3.2. Single-Core with Low Complementarity

The Single-Core with Low Complementarity collaboration model refers to a situation where multiple partners with low complementarity collaborate around a single core enterprise. Resources, technology, and decision-making authority are primarily concentrated within the core enterprise, and peripheral enterprises may supplement the core enterprise's additional functions, often through acquisition [60]. However, the supplementary functions generally do not constitute the core competitive advantage of the core enterprise, and the acquired peripheral company is, to some extent, replaceable. This model is typically used when the core enterprise seeks to quickly enhance or improve its additional functions. In the collaboration between Dassault Systèmes and Exalead, this model is primarily driven by acquisition, technological integration, and brand dilution, which together facilitate the upgrading and enhancement of the technology innovation management system.

First, the core enterprise acquires capabilities quickly through mergers and acquisitions [61]. Dassault did not develop its enterprise search and data analytics capabilities through internal R&D, as this was not its core strength. Instead, it opted to acquire Exalead to fill this gap. However, the acquired unit was not part of Dassault's core products. Second, the core enterprise enhances its product competitiveness through technological integration [62]. Dassault optimized its PLM system with Exalead's data analytics capabilities, enabling manufacturing clients to achieve faster data access and intelligent decision-making, thus improving user experience and the digitalization of manufacturing processes. However, since Exalead's technology is highly replaceable, Dassault's innovation management system focuses primarily on optimizing its own PLM ecosystem, rather than on deep collaborative innovation with its partners. Finally, the core enterprise influences the independence of the acquired company through brand and market dilution. After acquiring Exalead, Dassault integrated its technology into the 3DEXPERIENCE platform and promoted it as part of Dassault's product portfolio rather than as an independent software solution. This means the peripheral enterprise gradually loses market influence in the collaboration, and its technological direction is entirely driven by the core enterprise [63].

This model limits Exalead's potential for technological breakthroughs in broader industries but strengthens Dassault's overall advantage in the Industry 4.0 competitive landscape. In this model, if the collaboration between the core enterprise and the peripheral enterprise is interrupted for any reason, the core enterprise may lose its supplementary functions, but its core services remain unaffected. The partner may lose access to the main platform's market and user base, and the overall customer service experience may decline, although the core product's service experience is likely to remain intact, with only some additional features or secondary modules being lost.

4.3.3. Multi-Core with High Complementarity

The Multi-Core with High Complementarity collaboration model refers to a situation where multiple core entities maintain independence while collaborating based on high complementarity to achieve mutually beneficial goals. In this model, each core enterprise maintains independent authority in its domain, with complementary technologies but without being under the control of a single entity. This model is suitable for collaboration projects that involve high technological complexity, strong cross-domain integration, and relatively complex organizational structures [64]. In the collaboration between General Electric and Microsoft, this model primarily influences the technological innovation management system through equal collaboration, platform-level complementarity, and technical ecosystem synergy, maintaining the technological independence of each core enterprise while improving overall innovation efficiency through high complementarity in collaboration.

First, equal collaboration and technological openness promote cross-domain innovation. General Electric and Microsoft, both powerful in their respective fields, do not constrain each other; instead, they collaborate based on open interfaces (APIs) and interoperability standards. Predix and Azure facilitate flexible solution deployment across different platforms via data sharing, open APIs, and cloud compatibility, which not only enhances the flexibility of technological innovation but also reduces the technological barriers to cross-platform migration [65]. Second, multi-platform complementarity enhances ecosystem resilience [66]. Each party maintains its core competitiveness in the collaboration, with GE's Predix dominating the IIoT field, while Microsoft's Azure continues to expand its cloud computing and AI ecosystem. This platform-to-platform dependency allows both companies to leverage their strengths without affecting their own technological paths, resulting in business gains. They can continue independent growth in future technological iterations, avoiding vendor lock-in. Finally, collaborative innovation drives the expansion of the technical industry ecosystem [67]. Predix and Azure are compatible with different industrial applications, allowing developers to create cross-industry smart manufacturing solutions across platforms. This open ecosystem enables client enterprises to integrate both platforms to achieve broader industry coverage, thereby improving the overall effectiveness of the innovation management system. In this model, if the collaboration between the two parties were to end for any reason, the new business segments resulting from their cooperation would be affected, and customers would no longer enjoy the complete joint solution. They might need to seek other similar platforms, affecting the overall customer experience.

4.3.4. Multi-Core with Low Complementarity

The Multi-Core with Low Complementarity collaboration model refers to a situation where multiple partners with low complementarity collaborate while maintaining their independence, each focusing on their own core strengths. This model is typically suitable for cooperation relationships that are highly modular, with clear business boundaries and low interdependence. In such collaborations, companies assume different responsibilities and tasks in the process to achieve a common goal. Compared to the multi-core with high complementarity model, this model involves weaker technological collaboration between core enterprises, with a greater focus on market-level complementarity. In the collaboration between Schneider Electric and AVEVA, this model primarily influences the technological innovation management system through modular architecture, cross-platform data integration, and market coordination complementarity.

First, modular architecture ensures the independence of technological innovation [68]. Schneider Electric's EcoStruxure platform focuses on automation control and energy management, while AVEVA primarily offers industrial software and engineering management solutions. Although there is a synergy in the digital transformation process, the technological systems are relatively independent, allowing customers to use either company's products independently without relying on the other. This characteristic enables both parties to maintain their market positioning in the collaboration and avoid excessive dependence on a single technological framework [69]. Second, loose coupling structure promotes cross-platform data integration [70]. Schneider Electric and AVEVA adopt open interfaces (APIs) and data interoperability standards, allowing customers to freely combine their products according to their needs. The platform-to-platform loose dependency in this model allows enterprises to adapt their respective products to different industrial scenarios without being bound to each other's technological architecture due to the collaboration. Finally, market coordination complementarity drives the widespread expansion of technology applications [71]. Schneider Electric and AVEVA maintain their brand and market independence while expanding each other's market boundaries through collaboration. For example, Schneider Electric enhances its data integration and intelligent analysis capabilities with AVEVA's software, while AVEVA expands its market share in the energy management and automation sectors by leveraging Schneider's industry influence. However, compared to the multi-core with high complementarity model, the technological synergy in this model is weaker, with innovation primarily manifested in market expansion and customer penetration rather than deep integration of the technology itself. In this model, if the collaboration between the two parties is interrupted for any reason, their respective products and services will continue to operate independently without affecting the core innovation paths of the companies. Customers will only lose some of the cross-platform convenience, and the overall impact will be relatively small.

Based on the analysis of the impact of these four different open-source collaboration models on the technological innovation of industrial software companies, a comprehensive comparison is summarized in terms of coreness, complementarity, applicable scenarios, elements affecting technological innovation, and the impact on partners and customers after the collaboration is interrupted (Table 2). It is important to note that no single model can be conclusively stated as having the best impact; a comprehensive consideration of the actual circumstances and business needs of the collaborating parties is necessary.

Table 2. Comparison of Coreness and Complementarity Across Four Collaboration Models.

No.	Open Source Collaboration	Coreness Characteristics	Complementarity Characteristics
1	Single-Core with High Complementarity	Centralized platform governance, dominant actor dictates architecture and standards	High, partners contribute interoperable, non-redundant modules tightly integrated with the platform

Table 2. *Cont.*

No.	Open Source Collaboration	Coreness Characteristics	Complementarity Characteristics
2	Single-Core with Low Complementarity	Centralized, post-acquisition hierarchy, minimal architectural collaboration	Low, functional integration is narrow and substitutive
3	Multi-Core with High Complementarity	Distributed, co-governed between multiple platform owners	High, partners contribute domain-specific, synergistic capabilities
4	Multi-Core with Low Complementarity	Decentralized, loosely coordinated governance, independent roadmaps	Low, capabilities are overlapping and loosely coordinated

4.3.5. Comparative Analysis Framework

To synthesize insights across the four open-source collaboration cases, this section develops a comparative analysis framework based on two theoretical dimensions: coreness and complementarity. Building on the prior case descriptions, this section addresses three analytical aims: (1) comparing the structural configurations of the four models; (2) explaining the underlying drivers of their coreness and complementarity patterns; and (3) analyzing their differential impacts on organizational characteristics, innovation strategies, and technological outcomes. This multi-layered comparison enhances the study's explanatory power and contributes to theoretical advancement in innovation collaboration.

(1) Similarities and Differences in Coreness and Complementarity

As shown in Table 2, the four open-source collaboration models exhibit distinct configurations. The single-core with high complementarity model, such as Siemens and MindSphere, features centralized platform control combined with highly modular and interoperable external contributions. In contrast, the single-core with low complementarity model, represented by Dassault and Exalead, also has high coreness but weak interdependence, as functional integration is narrow and based on acquisition.

The multi-core with high complementarity model, exemplified by GE and Microsoft, is a co-governed architecture with partners contributing domain-specific yet synergistic modules. Finally, the multi-core with low complementarity model (Schneider and AVEVA) demonstrates loosely coupled interaction between independent actors with overlapping capabilities and low integration intensity.

Despite these differences, all models reflect attempts to leverage external resources to enhance innovation. However, they vary considerably in interface openness, governance symmetry, and resource orchestration, revealing important contrasts in their structural logics.

(2) Underlying Mechanisms and Influencing Factors of the Coreness and Complementarity Characteristics

The formation of coreness and complementarity characteristics across the four collaboration models is not random; rather, it is shaped by specific strategic intentions, resource endowments, and technological capabilities of the participating enterprises. These factors jointly determine how governance power is allocated, how functionally interdependent the collaborators are, and how architectural integration unfolds. This section analyzes each case accordingly.

In the single-core with high complementarity model, exemplified by Siemens and MindSphere collaboration, the dominant core structure was a direct manifestation of Siemens's trategic vision to become an orchestrator of industrial Internet of Things (IIoT) platforms. Siemens sought to construct a scalable and standardized ecosystem, which required strong centralized control over platform architecture, API governance, and developer access rights. Its global presence in industrial automation, embedded control systems, and engineering software endowed it with unparalleled technological and market leverage. This resource concentration enabled Siemens to attract a wide range of complementary contributors—such as edge analytics firms, predictive maintenance providers, and device manufacturers—who could modularly integrate their services into the MindSphere platform through open interfaces and developer toolkits. Therefore, the high degree of coreness was underpinned by platform orchestration logic, while the high complementarity emerged from modular openness and non-redundant partner capabilities. This case illustrates that strong strategic ambition combined with asymmetric resources and open platform infrastructure fosters both dominant coreness and strong complementarity.

In the single-core with low complementarity model, represented by Dassault and Exalead case, the mechanism of collaboration was fundamentally different. Dassault Systèmes' strategic goal was to fill a functionality gap in its PLM portfolio through the acquisition of Exalead's search technology. The collaboration was not oriented towards co-development or ecosystem integration but was a unidirectional capability patching exercise. Dassault's resource superiority in R&D, market channels, and platform control ensured a hierarchical relationship, wherein Exalead's semantic search functions were embedded as auxiliary modules. Due to a lack of mutual architectural redesign, and limited openness to other partners, the functional integration was shallow and non-essential to the broader platform. This case shows that when strategic intent is acquisition-driven and resources are asymmetrically distributed, high coreness may coexist with weak complementarity due to limited openness and one-sided integration.

The multi-core with high complementarity model, as seen in GE and Microsoft partnership, was underpinned by a mutual strategic alignment to jointly shape the industrial digital transformation space. Both parties brought distinct and high-value resource endowments: GE contributed its operational technology expertise and the Predix industrial cloud, while Microsoft provided infrastructure, AI services, and ecosystem access through Azure. Neither firm aimed for unilateral control; instead, they coordinated platform interoperability via shared APIs, security standards, and joint technical roadmaps. The technical architecture was modular but interoperable, and strategic intent was based on mutual domain expansion rather than acquisition or dominance. This balance allowed each core firm to retain sovereignty while co-developing solutions with strong functional interdependence. This case demonstrates that when firms possess non-redundant resources and aligned strategic goals, they can sustain co-governed structures with high complementarity through joint architecture design.

In the multi-core with low complementarity model, exemplified by the Schneider Electric and AVEVA partnership, the collaboration mechanism was based more on market-side synergy than on technological convergence. Although Schneider eventually acquired AVEVA, both companies continued to operate relatively independently, maintaining their own product lines, development priorities, and governance systems. Their overlapping expertise in SCADA, DCS, and simulation tools created resource redundancy rather than synergy. Moreover, their platforms lacked open interface standards for tight integration, which further constrained co-innovation. The strategic intention behind the collaboration appeared more commercially motivated—expanding customer access and bundling solutions—than aimed at driving systemic technical integration. This case reveals that even under formal ownership structures, strategic fragmentation, overlapping resources, and low technical modularity can lead to loosely coupled collaborations with low coreness and low complementarity.

These comparative insights suggest that coreness is fundamentally shaped by a firm's strategic ambition to lead or co-lead a technological ecosystem and its corresponding control over platform governance. In contrast, complementarity is more dependent on whether the involved entities possess distinct, non-overlapping resources and whether their technical interfaces permit collaborative modularity. The evolution of open-source collaboration structures thus reflects a combination of intentional governance design and resource-technical alignment.

(3) Similarities and Differences in Organizational Characteristics, Innovation Strategies, and Technological Impacts of different collaboration models

Open-source collaboration models differ not only in their structural configurations of coreness and complementarity, but also in their organizational forms, innovation strategies, and technological impacts. These differences significantly influence how industrial software enterprises organize cooperation, pursue innovation, and generate technological value. Therefore, Table 3 provides a summary comparison across these three dimensions.

Table 3. Comparison of Organizational Characteristics, Innovation Strategies, and Technological Impacts.

No.	Open Source Collaboration	Organizational Characteristics	Innovation Strategy	Technological Impact
1	Single-Core with High Complementarity	Platform-centric, centralized governance, modularized contributors	Exploitative innovation, platform optimization and rapid scaling	Strong ecosystem convergence, standardized modular innovation
2	Single-Core with Low Complementarity	Hierarchical structure, asymmetric power from acquisition	Substitutive innovation, capability patching via acquisition	Isolated enhancement, low synergy and limited system integration
3	Multi-Core with High Complementarity	Balanced authority, distributed platform sovereignty, co-governance	Exploratory innovation, cross-domain co-development and experimentation	Interoperable co-innovation, collaborative technical evolution
4	Multi-Core with Low Complementarity	Loose coupling, independent product and governance structures	Market-driven innovation, incremental coordination with minimal integration	Non-systemic output, limited technical spillover, stable coexistence

In terms of organizational characteristics, the single-core with high complementarity model—exemplified by Siemens and MindSphere collaboration—adopts a platform-centric architecture. The dominant firm governs core technical standards, interface specifications, and platform governance mechanisms, while peripheral contributors are modular participants with limited strategic input. This arrangement enhances platform efficiency and consistency but centralizes decision-making power. In the single-core with low complementarity model (e.g., Dassault and Exalead), organizational structure is even more hierarchical, stemming from a post-acquisition integration logic. The acquiring firm absorbs specific functionalities into the dominant product line without establishing a long-term co-governance structure. The collaboration tends to be temporary and functionally narrow. By contrast, the multi-core with high complementarity model, such as GE and Microsoft partnership, features distributed organizational authority. Each firm maintains operational autonomy while jointly coordinating integration layers, protocols, and platform alignment. This co-governance model supports cross-sectoral institutional collaboration and dynamic organizational learning, though it also requires stronger relational governance and inter-firm trust. In the multi-core with low complementarity model, such as Schneider and AVEVA, collaboration occurs between relatively symmetric and independent entities. The organizational interaction is minimal, usually confined to customer-facing integration or commercial bundling. R&D pipelines, product planning, and ecosystem standards remain separate, and organizational interdependence is low.

The collaboration models also vary in their innovation logics. The single-core with high complementarity model favors exploitative innovation, emphasizing incremental improvements, platform optimization, and integration of third-party functionalities under a unified architecture. This model enables rapid scaling and standard consolidation but may suppress diverse experimentation. In contrast, the multi-core with high complementarity model is more aligned with exploratory innovation, where partners engage in joint knowledge generation across technical domains. Due to mutual autonomy and trust-based coordination, these collaborations promote architectural experimentation and open-ended problem-solving. Innovation is emergent, driven by the recombination of cross-domain capabilities. The single-core with low complementarity model typically pursues functionally substitutive innovation. The primary firm fills strategic or technical gaps by acquiring narrowly focused firms, enabling rapid deployment but limited architectural evolution. This results in low integration depth and minimal impact on long-term innovation capacity. Meanwhile, multi-core with low complementarity configurations often adopt market-driven innovation. The partners' goal is to co-exist in the market ecosystem without deep integration, relying on loose coordination to align sales channels or expand joint offerings. Innovation in such contexts is incremental, commercially oriented, and often modularly redundant.

From a technological impacts perspective, different collaboration models result in varying degrees of system integration, knowledge spillovers, and innovation scalability. The single-core with high complementarity model often achieves strong ecosystem convergence, as a unified platform attracts diverse external contributions that comply with standardized protocols. However, such centralization may lead to partner lock-in and reduced resilience to architectural change. The multi-core with high complementarity model tends to generate interoperable co-innovation. Cross-platform compatibility, joint technical architecture, and open interfaces enable distributed innovation and broader ecosystem participation. The diversity of contributors and balance of influence allow for sustained system evolution and collaborative standard-setting. In single-core with low complementarity collaborations, technological impacts are localized and bounded. Innovation is typically confined to specific functional modules, and there is little long-term synergy. The absorbed technologies may enhance core capabilities, but their integration is isolated and often non-replicable across platforms. Finally, in multi-core with low complementarity cases, technological spillovers are limited. Each party continues to iterate its existing stack, and collaboration outcomes are frequently non-systemic. The modular coexistence avoids disruption but also restricts the emergence of novel technological trajectories.

5. Conclusions and Implications

5.1. Research Conclusions

This study addresses two core research questions: (1) What are the different types of open-source collaboration models based on their characteristics? and (2) How do these collaboration models influence technological innovation in the industrial software industry?

To answer these questions, the study adopts a multiple case study approach, selecting four representative industrial software collaboration projects (Case A: Siemens and MindSphere, Case B: Dassault and Exalead, Case C: GE and Microsoft, and Case D: Schneider and AVEVA) as empirical samples. The research process began by developing a typological framework built upon two key structural dimensions: coreness, referring to the degree of control and centralization in the collaboration, and complementarity, representing the extent of functional interdependence among participating actors. Based on this dual-dimensional framework, the study first conducted within-case analysis to examine the features of each collaboration structure, and then employed cross-case comparison to identify similarities, differences, and their implications. A combination of literature-informed conceptualization, empirical documentation from secondary data, and inductive interpretation was used to derive grounded and generalizable insights. This integrative design ensures that the findings are both theoretically informed and empirically validated.

The main findings are summarized in three key aspects: First, the study identifies four distinct types of open-source collaboration models based on the interaction between coreness and complementarity: (1) Single-Core with High Complementarity, as seen in Siemens and MindSphere collaboration, characterized by centralized platform control and deep modular integration of complementary partners. (2) Single-Core with Low Complementarity, exemplified by Dassault and Exalead, featuring a dominant firm absorbing peripheral functionalities through acquisition with limited mutual interdependence. (3) Multi-Core with High Complementarity, as in GE and Microsoft partnership, where co-equal firms maintain autonomous platforms while contributing interoperable capabilities for cross-sector innovation. (4) Multi-Core with Low Complementarity, illustrated by Schneider and AVEVA, characterized by decentralized, loosely coordinated collaborations with overlapping technical resources and minimal integration. This classification contributes a new structural lens to the literature on open-source collaboration by delineating how collaboration models differ systematically based on internal governance and external synergy.

Second, the study reveals the underlying mechanisms that shape the formation of these four models. The formation of coreness and complementarity characteristics across the four collaboration models is shaped by specific strategic intentions, resource endowments, and technological capabilities of the participating enterprises. For example, the high complementarity in the Siemens and GE cases is enabled by modular platforms and non-redundant technical capabilities, while the low complementarity in the Dassault and Schneider cases reflects overlapping functions, shallow integration, or asymmetrical objectives. These findings clarify why different structural configurations emerge, providing a theoretical basis for understanding the evolution and governance of open-source collaboration.

Third, the study demonstrates that each type of collaboration model has differentiated impacts on organizational structures, innovation strategies, and technological impacts. Single-core with high complementarity collaborations enable platform standardization and ecosystem convergence but may centralize innovation power and limit flexibility. Single-core with low complementarity models offer short-term functional enhancement for the lead firm but generate limited spillovers. Multi-core with high complementarity supports cross-domain co-innovation and collaborative ecosystem evolution, although it requires high governance capability and mutual trust. Multi-core with low complementarity allows for market flexibility and parallel product evolution but lacks deep technological integration or systemic innovation output. These differentiated outcomes offer strategic insights into how firms may align their innovation goals with appropriate collaboration structures, especially in highly complex and modularized sectors such as industrial software.

Overall, by bridging theoretical typology with real-world industrial cases, this study enriches the literature on open-source innovation management and ecosystem governance. It also offers actionable implications for firms and policymakers seeking to harness open collaboration to enhance technological innovation in the rapidly evolving industrial software landscape.

5.2. Theoretical Contributions

On the one hand, this study constructs a classification framework for open-source collaboration models based on the number of core entities and the degree of complementarity, expanding the theoretical foundation of organizational management in open-source collaboration. Existing research on open-source collaboration primarily focuses on how enterprises participate in open-source communities with limited attention to the systematic classification of open-source collaboration models and their impacts. By incorporating core entity count (single-core vs. multi-core) and complementarity degree (high complementarity vs. low complementarity) as two key dimensions, this study categorizes open-source collaboration into four distinct models and conducts an in-depth analysis of their organizational characteristics, collaboration dynamics, and applicable scenarios. This classification framework not only fills the research gap in open-source collaboration models but also provides a novel theoretical perspective for future studies, thereby enriching the organizational management literature in open-source collaboration.

On the other hand, this study reveals the impact of different open-source collaboration models on technological innovation in the industrial software industry, advancing the theoretical understanding of open-source innovation management. While existing research generally acknowledges the role of open-source models in fostering innovation, it lacks a nuanced exploration of how different collaboration models shape innovation pathways. This study systematically compares the organizational characteristics, innovation strategies, and technological impacts of different models, and further explores their underlying formation mechanisms such as strategic intention, resource endowment, and technological modularity. These findings not only deepen the theoretical foundation of open-source innovation management but also offer practical insights for enterprises in selecting appropriate open-source collaboration models to enhance technological innovation.

5.3. Practical Implications

This study provides valuable insights for various stakeholders in navigating open-source collaboration, optimizing technological innovation strategies, and mitigating risks associated with collaboration termination. Enterprises, policymakers, open-source communities, and industry partners all play distinct roles in shaping the effectiveness and sustainability of open-source collaboration.

For enterprises, selecting an appropriate open-source collaboration model is crucial to aligning technological innovation with business strategy. Different models cater to distinct innovation needs and competitive landscapes. Specifically, for firms seeking platform standardization and centralized control, the single-core with high complementarity model is suitable. These enterprises can invest in building proprietary platforms with open APIs, actively orchestrate third-party modules, and develop strong ecosystem governance protocols. For enterprises emphasizing cross-domain innovation and technological co-creation, the multi-core with high complementarity model is more applicable. These firms can engage in joint R&D alliances, sign IP-sharing and joint governance agreements, and build shared cloud platforms to facilitate knowledge recombination and reduce interface costs. In implementing any model, firms must also anticipate collaboration termination risks. This includes creating contingency transition plans, modularizing dependencies, and adopting dual-sourcing strategies to avoid vendor lock-in. Enterprises engaged in multi-core with low complementarity collaborations should emphasize independent product planning while maintaining flexible data-level interoperability to quickly adjust to ecosystem shifts.

For policymakers, fostering a robust and secure open-source ecosystem requires differentiated policy instruments tailored to the needs of various collaboration models. In single-core with high complementarity settings, policy tools should focus on platform standardization, such as promoting interoperability certification, open standard setting consortia, and national-level pilot platforms to accelerate ecosystem consolidation. For example, establishing a national industrial software interoperability index can encourage companies to comply with core standards. In multi-core with high complementarity collaboration, governments should promote cross-sector collaboration subsidies, interdisciplinary joint labs, and open data infrastructure, particularly for complex, high-risk industrial software innovation. To mitigate risks of collaboration breakdown, policymakers can introduce open continuity protocols—legal or funding mechanisms that ensure software, interfaces, and documentation remain accessible in the event of termination. Moreover, policies should support talent mobility, ensuring experienced developers can migrate between platforms while preserving software continuity. Finally, regional governments can provide sandbox environments and regulatory testbeds that allow enterprises to trial different collaboration models under controlled and policy-supported conditions.

For open-source communities, the governance architecture must adapt to different open-source collaboration types to ensure inclusiveness, scalability, and continuity. In single-core with high complementarity models, communities should formalize code contribution workflows, implement IP protection agreements, and provide API development toolkits to align enterprise needs with community standards. For example, creating a core-periphery governance board comprising both enterprise leaders and community maintainers can enhance mutual trust and reduce coordination costs. In multi-core models, communities need to strengthen federated governance models, such as interoperability working groups and shared architecture committees, to balance contributions and resolve interface disputes. Additionally, all open-source communities should establish collaboration fallback mechanisms, such as community mirror repositories, developer transfer protocols, and platform-neutral user forums, to ensure innovation momentum even when collaboration falters.

For industry users and customers, understanding and managing dependency risks in open-source collaboration is critical. In single-core with high complementarity ecosystems, customers should assess platform stickiness by monitoring the openness of APIs, licensing terms, and long-term platform viability. Risk-averse firms should negotiate data portability rights and functional redundancy guarantees in supplier contracts. For clients using solutions developed under multi-core models, it is advisable to promote multi-vendor integration testing, adopt open-standard procurement frameworks, and maintain internal capacity for modular customization. This ensures that system resilience and compatibility can be maintained even under changing collaboration conditions. Moreover, industry users are encouraged to actively participate in open-source communities—either via direct contribution or advisory participation—to influence development direction, obtain early access to tools, and secure training or support resources that reduce operational uncertainty.

5.4. Research Limitations

Despite its contributions, this study has certain limitations that provide avenues for future research. First, while the classification framework of open-source collaboration models offers a structured perspective, it lacks quantitative validation of its impact on domestic industrial software innovation catch-up. Future research could employ empirical methods to measure the extent to which different collaboration models contribute to innovation outcomes and explore the role of external moderating variables in shaping these effects. Second, this study does not fully assess the effectiveness of government policies in fostering open-source collaboration. Future research could conduct policy impact evaluations, identifying key challenges in policy implementation and proposing improvements to optimize regulatory support for open-source innovation.

Author Contributions: Conceptualization: X.C. and Y.Z.; methodology: X.C. and Y.Z.; software: X.C.; validation: X.C.; formal analysis: X.C.; resources: X.C. and Y.Z.; data curation: X.C. and Y.Z.; writing—original draft preparation: X.C.; writing—review and editing: X.C. and Y.Z.; visualization: X.C.; supervision: Y.Z.; project administration: X.C.; funding acquisition: X.C. and Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was funded by the Humanities and Social Science Youth Fund of Ministry of Education of China (Grant: 23YJC630014); National Natural Science Foundation of China (Grant: 71974107); National Natural Science Foundation of China (Grant: L2424237); the National Social Science Foundation of China; Humanities and Social Sciences Fund of Ministry of Education of China (Grant: 16JDGC011).

Data Availability Statement: The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest: The authors declare no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

1. Ryu, D.; Baek, K.H.; Yoon, J. Open innovation with relational capital, technological innovation capital, and international performance in SMEs. *Sustainability* **2021**, *13*, 3418. [CrossRef]
2. Hertog, P.D. Knowledge-intensive business services as co-producers of innovation. *Int. J. Innov. Manag.* **2000**, *4*, 491–528. [CrossRef]
3. Jacobides, M.G. How to compete when industries digitize and collide: An ecosystem development framework. *Calif. Manag. Rev.* **2022**, *64*, 99–123. [CrossRef]
4. Palmié, M.; Wincent, J.; Parida, V.; Caglar, U. The evolution of the financial technology ecosystem: An introduction and agenda for future research on disruptive innovations in ecosystems. *Technol. Forecast. Soc. Chang.* **2020**, *151*, 119779. [CrossRef]
5. Mortazavi Ravari, S.S.; Mehrabanfar, E.; Banaitis, A.; Banaitienė, N. Framework for assessing technological innovation capability in research and technology organizations. *J. Bus. Econ. Manag.* **2016**, *17*, 825–847. [CrossRef]
6. Chen, X.; Mao, J.; Ma, Y.; Li, G. The knowledge linkage between science and technology influences corporate technological innovation: Evidence from scientific publications and patents. *Technol. Forecast. Soc. Chang.* **2024**, *198*, 122985. [CrossRef]
7. Yoo, Y.; Henfridsson, O.; Lyytinen, K. Research commentary—The new organizing logic of digital innovation: An agenda for information systems research. *Inf. Syst. Res.* **2010**, *21*, 724–735. [CrossRef]
8. Jin, Y.; Shao, Y. Power-leveraging paradox and firm innovation: The influence of network power, knowledge integration and breakthrough innovation. *Ind. Mark. Manag.* **2022**, *102*, 205–215. [CrossRef]
9. Xiao, T.; Makhija, M.; Karim, S. A knowledge recombination perspective of innovation: Review and new research directions. *J. Manag.* **2022**, *48*, 1724–1777. [CrossRef]
10. Chen, X.H.; Zhou, Y. The Essence of Innovative Collaboration and Theoretical Evolution of Open Source Software from Cooperation and Competition Perspectives. *Sci. Sci. Manag. S. T.* **2024**, *45*, 13–30.
11. Zhou, Y.; Liu, W.; Lv, X.; Chen, X.; Shen, M. Investigating interior driving factors and cross-industrial linkages of carbon emission efficiency in China's construction industry: Based on Super-SBM DEA and GVAR model. *J. Clean. Prod.* **2019**, *241*, 118322. [CrossRef]
12. Delgado, M.; Porter, M.E.; Stern, S. Clusters, convergence, and economic performance. *Res. Policy* **2014**, *43*, 1785–1799. [CrossRef]
13. Fernández-Portillo, A.; Almodóvar-González, M.; Hernández-Mogollón, R. Impact of ICT development on economic growth. A study of OECD European union countries. *Technol. Soc.* **2020**, *63*, 101420. [CrossRef]
14. Dyer, J.H.; Singh, H.; Hesterly, W.S. The relational view revisited: A dynamic perspective on value creation and value capture. *Strateg. Manag. J.* **2018**, *39*, 3140–3162. [CrossRef]
15. Dingler, A.; Enkel, E. Socialization and innovation: Insights from collaboration across industry boundaries. *Technol. Forecast. Soc. Chang.* **2016**, *109*, 50–60. [CrossRef]
16. Bø Lyng, H.; Brun, E.C. Innovating with strangers; managing knowledge barriers across distances in cross-industry innovation. *Int. J. Innov. Technol. Manag.* **2020**, *17*, 2050008. [CrossRef]
17. Spithoven, A.; Merlevede, B. The productivity impact of R&D and FDI spillovers: Characterising regional path development. *J. Technol. Transf.* **2023**, *48*, 560–590.
18. Zhao, X.; Yang, S. Does intelligence improve the efficiency of technological innovation? *J. Knowl. Econ.* **2023**, *14*, 3671–3695. [CrossRef]

19. Liu, J.; Chang, H.; Forrest, J.Y.L.; Yang, B. Influence of artificial intelligence on technological innovation: Evidence from the panel data of china's manufacturing sectors. *Technol. Forecast. Soc. Chang.* **2020**, *158*, 120142. [CrossRef]
20. Feng, N.; Tu, S.; Guo, F. Big-data analytics capability, value creation process, and collaboration innovation quality in manufacturing enterprises: A knowledge-based view. *Comput. Ind. Eng.* **2024**, *187*, 109804. [CrossRef]
21. Wen, H.; Zhong, Q.; Lee, C.C. Digitalization, competition strategy and corporate innovation: Evidence from Chinese manufacturing listed companies. *Int. Rev. Financ. Anal.* **2022**, *82*, 102166. [CrossRef]
22. Liu, C.; Ji, H.; Ji, J. Mobile information technology's impacts on service innovation performance of manufacturing enterprises. *Technol. Forecast. Soc. Chang.* **2022**, *184*, 121996. [CrossRef]
23. Agarwal, S.; Kapoor, R. Value creation tradeoff in business ecosystems: Leveraging complementarities while managing interdependencies. *Organ. Sci.* **2023**, *34*, 1216–1242. [CrossRef]
24. Baldwin, C.; Von Hippel, E. Modeling a paradigm shift: From producer innovation to user and open collaborative innovation. *Organ. Sci.* **2011**, *22*, 1399–1417. [CrossRef]
25. Teece, D.J. Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy. *Res. Policy* **1986**, *15*, 285–305. [CrossRef]
26. Yang, M.J. The interdependence imperative: Business strategy, complementarities, and economic policy. *Oxf. Rev. Econ. Policy* **2021**, *37*, 392–415. [CrossRef]
27. Thomas, L.D.W.; Ritala, P.; Karhu, K.; Heiskala, M. Vertical and horizontal complementarities in platform ecosystems. *Innovation* **2024**, 1–25. [CrossRef]
28. Badr, M. Unleashing the Power of AI: The Microsoft and OpenAI Partnership. April 2023. Available online: http://hdl.handle.net/1773/49810 (accessed on 27 December 2024).
29. Alcacer, J.; Khanna, T.; Snively, C. The rise and fall of Nokia. *Harv. Bus. Sch. Case* **2014**, *714*, 428.
30. Granovetter, M.S. The strength of weak ties. *Am. J. Sociol.* **1973**, *78*, 1360–1380. [CrossRef]
31. Baldwin, C.Y.; Clark, K.B. The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Manag. Sci.* **2006**, *52*, 1116–1127. [CrossRef]
32. Gereffi, G.; Humphrey, J.; Sturgeon, T. The governance of global value chains. *Rev. Int. Political Econ.* **2005**, *12*, 78–104. [CrossRef]
33. Aksnes, D.W.; Sivertsen, G. Global trends in international research collaboration, 1980–2021. *J. Data Inf. Sci.* **2023**, *8*, 26–42.
34. Arita, M.; Karsch-Mizrachi, I.; Cochrane, G. The international nucleotide sequence database collaboration. *Nucleic Acids Res.* **2021**, *49*, D121–D124. [CrossRef]
35. Zhou, X.W.; Xiao, Y.L.; Yang, X.H. Industrial software innovation breakthrough path and incentive mechanism for domestic replacement. *Sci. Technol. Rev.* **2023**, *41*, 34–46.
36. Gao, L.; Li, P.G.; Huang, P.; Yang, Z.; Gao, J. Development Strategies of Industrial Software for Digital Design. *Strateg. Study CAE* **2023**, *25*, 254–262.
37. Shao, Z.; Zhao, Y.; Wang, C.; Feng, X.; Wang, J.; Xiong, H. Development path of China's industrial software industry in the new era. *Strateg. Study Chin. Acad. Eng.* **2022**, *24*, 86–95. [CrossRef]
38. Zhang, H. Research on Collaborative Innovation in Industrial Software Development Based on Tripartite Evolutionary Game Theory. In Proceedings of the 4th International Conference on Artificial Intelligence and Computer Engineering, Dalian, China, 17–19 November 2023; pp. 811–817.
39. Haefliger, S.; Von Krogh, G.; Spaeth, S. Code reuse in open source software. *Manag. Sci.* **2008**, *54*, 180–193. [CrossRef]
40. Adner, R. Match your innovation strategy to your innovation ecosystem. *Harv. Bus. Rev.* **2006**, *84*, 98. [PubMed]
41. Carayannis, E.G.; Campbell, D.F.J. 'Mode 3' and 'Quadruple Helix': Toward a 21st century fractal innovation ecosystem. *Int. J. Technol. Manag.* **2009**, *46*, 201–234. [CrossRef]
42. Jorge, J. Strategic complementarities, geographical agglomeration, and firm investment. *Eur. J. Financ.* **2023**, *29*, 135–154. [CrossRef]
43. Kapoor, R.; Furr, N.R. Complementarities and competition: Unpacking the drivers of entrants' technology choices in the solar photovoltaic industry. *Strateg. Manag. J.* **2015**, *36*, 416–436. [CrossRef]
44. Kapoor, R.; Agarwal, S. Sustaining superior performance in business ecosystems: Evidence from application software developers in the iOS and Android smartphone ecosystems. *Organ. Sci.* **2017**, *28*, 531–551. [CrossRef]
45. Conaldi, G.; De Vita, R. At the core of innovation: Network reconfiguration during radical and incremental innovation episodes in an open source software project. *Int. J. Inf. Decis. Sci.* **2011**, *3*, 5–25. [CrossRef]
46. Yin, R.K. *Case Study Research and Applications: Design and Methods*; Sage: Atlanta, GA, USA, 2018.
47. Luo, Y.; Jin, Y.; Ji, Y. Explore an open-source value co-creation framework: A multiple case study. *PLoS ONE* **2024**, *19*, e0310516. [CrossRef]

48. Schueller, W.; Wachs, J.; Servedio, V.D.P.; Thurner, S.; Loreto, V. Evolving collaboration, dependencies, and use in the rust open source software ecosystem. *Sci. Data* **2022**, *9*, 703. [CrossRef]
49. Eisenhardt, K.M.; Graebner, M.E. Theory building from cases: Opportunities and challenges. *Acad. Manag. J.* **2007**, *50*, 25–32. [CrossRef]
50. Lin, Y.; Zhao, X.; Chen, L. A multi-case study of business innovation models for manufacturing capacity sharing platforms, based on a resource orchestration perspective. *Eng. Manag. J.* **2024**, *36*, 92–102. [CrossRef]
51. Fitzky, I.; Baltes, G.H. The Path to Innovation: Exploring the Interplay of Multiple Corporate Entrepreneurship Units. In Proceedings of the 2024 IEEE International Conference on Engineering, Technology, and Innovation (ICE/ITMC), Funchal, Portugal, 24–28 June 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–7.
52. Massa, L.; Ardito, L.; Petruzzelli, A.M. Brokerage dynamics in technology transfer networks: A multi-case study. *Technol. Forecast. Soc. Chang.* **2022**, *183*, 121895. [CrossRef]
53. Eisenhardt, K.M. Building theories from case study research. *Acad. Manag. Rev.* **1989**, *14*, 532–550. [CrossRef]
54. Petrik, D.; Herzwurm, G. iIoT ecosystem development through boundary resources: A Siemens MindSphere case study. In Proceedings of the 2nd ACM SIGSOFT International Workshop on Software-Intensive Business: Start-Ups, Platforms, and Ecosystems, Tallinn, Estonia, 26 August 2019; pp. 1–6.
55. Yu, S.; Alqahtani, F.; Tolba, A.; Lee, I.; Jia, T.; Xia, F. Collaborative team recognition: A core plus extension structure. *J. Informetr.* **2022**, *16*, 101346. [CrossRef]
56. Schermuly, L.; Schreieck, M.; Wiesche, M.; Krcmar, H. Developing an industrial IoT platform–Trade-off between horizontal and vertical approaches. In Proceedings of the 14th International Conference on Wirtschaftsinformatik, Siegen, Germany, 24–27 February 2019.
57. Petrik, D.; Herzwurm, G. Towards an understanding of iIoT ecosystem evolution-MindSphere case study. In Proceedings of the Software Business: 10th International Conference, ICSOB 2019, Jyväskylä, Finland, 18–20 November 2019; Proceedings 10. Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 46–54.
58. Novikov, S.V.; Sazonov, A.A. Application of the open operating system 'MindSphere' in digital transformation of high-tech enterprises. *Econ. J.* **2019**, *1*, 20–26. [CrossRef]
59. Petrik, D.; Herzwurm, G. Complementor satisfaction with boundary resources in IIoT ecosystems. In Proceedings of the Business Information Systems: 23rd International Conference, BIS 2020, Colorado Springs, CO, USA, 8–10 June 2020; Proceedings 23. Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 351–366.
60. Suo, L.; Yang, K.; Ji, H. The impact of technological mergers and acquisitions on enterprise innovation: A review. *Sustainability* **2023**, *15*, 12883. [CrossRef]
61. Hanif, N. Unleashing synergy: Exploring the impact of experiential learning and component technology on integration implementation strategy in cross-border technological acquisitions. *Rev. Int. Bus. Strategy* **2024**, *34*, 387–411. [CrossRef]
62. Basana, S.; Malelak, M.; Suprapto, W.; Siagian, H.; Tarigan, Z.J.H. The impact of SCM integration on business performance through information sharing, quality integration and innovation system. *Uncertain Supply Chain. Manag.* **2024**, *12*, 435–448. [CrossRef]
63. Chiu, W.H.; Shih, Y.S.; Chu, L.S.; Chen, S.L. Merger and acquisitions integration, implementation as innovative approach toward sustainable competitive advantage: A case analysis from Chinese sports brands. *Front. Psychol.* **2022**, *13*, 869836. [CrossRef] [PubMed]
64. Wang, Y.; Ma, J. Innovative Research and Development of Strategic Alliance of Core Technologies in The Emerging Technology Supply Chain. *Front. Bus. Econ. Manag.* **2023**, *8*, 155–159. [CrossRef]
65. Reznikov, R. The economic impact of cloud technologies on the industry 4.0 development. *Econ. Her. Donbas* **2023**, *4*, 74. [CrossRef]
66. Mohamed, M.; Ahokangas, P.; Pikkarainen, M. Complementors' coopetition-based business models in multiplatform ecosystems. *J. Bus. Models* **2023**, *11*, 68–77. [CrossRef]
67. Vosman, L.; Coenen, T.B.J.; Volker, L.; Visscher, K. Collaboration and innovation beyond project boundaries: Exploring the potential of an ecosystem perspective in the infrastructure sector. *Constr. Manag. Econ.* **2023**, *41*, 457–474. [CrossRef]
68. Habib, T.; Kristiansen, J.N.; Rana, M.B.; Ritala, P. Revisiting the role of modular innovation in technological radicalness and architectural change of products: The case of Tesla X and Roomba. *Technovation* **2020**, *98*, 102163. [CrossRef]
69. de Beer, J.; Depew, C. The role of process engineering in the digital transformation. *Comput. Chem. Eng.* **2021**, *154*, 107423. [CrossRef]

70. Ceniceros, J.A.R.; Aguilar-Calderón, J.A.; Espinosa, R.; Tripp-Barba, C. The external and data loose coupling for the integration of software units: A systematic mapping study. *PeerJ Comput. Sci.* **2021**, *7*, e796. [CrossRef] [PubMed]
71. Feizabadi, J.; Alibakhshi, S. Synergistic effect of cooperation and coordination to enhance the firm's supply chain adaptability and performance. *Benchmarking Int. J.* **2022**, *29*, 136–171. [CrossRef]