3-2023

A study of the impact of data intelligence on software delivery performance

Yongdong DONG
Singapore Management University, yddong.2017@ckdba.smu.edu.sg

Citation

DONG, Yongdong. A study of the impact of data intelligence on software delivery performance. (2023). 1-181.
Available at: https://ink.library.smu.edu.sg/etd_coll/478

A STUDY OF THE IMPACT OF DATA INTELLIGENCE ON SOFTWARE DELIVERY PERFORMANCE

DONG YONGDONG

SINGAPORE MANAGEMENT UNIVERSITY

2023

A Study of the Impact of Data Intelligence on Software Delivery Performance

Dong Yongdong

Submitted to Lee Kong Chian School of Business in partial fulfillment of the requirements for the
Degree of Doctor in Business Administration

Dissertation Committee:

Wang Heli (Supervisor/Chair)
Janice Bellace Professor of Strategic Management
Singapore Management University

Wang Yijiang (Co-Supervisor)
Professor of Economics
Cheung Kong Graduate School of Business

Guo Zhiling
Associate Professor of Information Systems
Singapore Management University

SINGAPORE MANAGEMENT UNIVERSITY
2023

I hereby declare that this DBA dissertation is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in this DBA dissertation.

This DBA dissertation has also not been submitted for any degree in any university previously.

Dong Yongdong

21 March 2023

A Study of the Impact of Data Intelligence on Software Delivery Performance

Dong Yongdong

Abstract

With the rise of big data and artificial intelligence, data intelligence has gradually become the focus of academia and industry. Data intelligence has two obvious characteristics: big data drive and application scene drive. More and more enterprises extract valuable patterns contained in data with prediction and decision analysis methods and technologies such as large-scale data mining, machine learning and deep learning and use them to improve the management and decision in complex practice, so as to promote changes of new business modes, organizational structures and even business strategies, and improve the operational efficiency of organizations. However, there are few studies on how data intelligence affects organizational performance. This dissertation points out three-dimensional elements for data intelligence to achieve iterative development: data, technology and prediction and decision-making capability, and makes a more in-depth exploration on the research and application of the three elements and organizational performance. Based on the relevant data of software enterprises, the author conducted empirical research and finally concluded the mechanism and path of data intelligence affecting the software delivery performance, providing an important reference for other industry organizations to improve the operational efficiency and organizational performance.

Key words: data intelligence, big data, decision making, software delivery performance, software project management

Contents

List of figures

List of tables

Acknowledgment

The doctoral study period is a very important experience, which has been a precious treasure for my whole life. Especially in the writing process of my doctoral thesis, the whole process from selecting the topic of the thesis, reading a lot of relevant literature, collecting a lot of data and perfecting the final writing is not only full of challenges and pressure, but also makes me deeply feel the fun of swimming in the ocean of knowledge.

Thanks to Professor Wang Heli, Chairman of the Committee of Supervisors, Professor Wang Yijiang, and Associate Professor Guo Zhiling for their kindness. Their profound knowledge and rigorous academic spirit have benefited me a lot. In the process of writing my doctoral thesis, they gave me careful guidance and valuable suggestions for many times, which enabled me to successfully complete my doctoral thesis and achieve today's results. Here I would like to express my deep gratitude to all the tutors. I would also like to thank Ms. Yao Wei, Ms. Li Linna and other teachers for their support and help during my doctoral study.

In addition, I would like to thank my colleagues and friends who helped me during the writing of my doctoral thesis. They actively cooperated with the questionnaire survey, carefully filled in relevant data, and provided sufficient data for the research of my thesis.

Finally, I want to give special thanks to my family. Their encouragement and support enable me to focus on my work and study and not be disturbed by the

trifles in life. During the writing of my doctoral thesis, I deeply realized the limitations and deficiencies of my knowledge structure. The completion of my doctoral thesis is just another new starting point in my knowledge learning. In the face of the rapidly changing digital era of science and technology, the journey of learning has just begun.

Thanks to all the teachers, colleagues, classmates and friends who have supported and paid attention to my doctoral study. Your support is the greatest motivation and help for my study and work. Thank you very much.

Chapter 1 Background

In today's world, scientific and technological innovation is advancing rapidly. In particular, big data, artificial intelligence, digitization, networking and intelligence are gaining momentum, and playing an increasingly important role in promoting economic and social development, modernizing the national governance system and governance capacity, and meeting people's ever-growing needs for a better life. With the arrival of the era of artificial intelligence, new technologies and methods such as data intelligence, computer vision and recognition, robotics and intelligent networking emerge in an endless stream. The concept that data intelligence can improve the operational efficiency of organizations has become a consensus. Helping enterprises promote or achieve the transformation of business modes, organizational structures and even business strategies by use of new-generation information technologies such as artificial intelligence has become a trend, and a breakthrough of application implementation. However, there is no final conclusion on how to use data intelligence to improve the operational efficiency of organizations accurately and how to establish the mechanism of data intelligence improving the operational efficiency of organizations.

For a software company (organization), software delivery performance is the main competitiveness of the organization. The software delivery performance in this dissertation includes the delivery performance of software products and customized software systems, which determines the value that software can bring to the organization. It is the core of the software enterprise and the main aspect impacting the organization's operational efficiency. This dissertation, taking the research and development and practice of the software company where the author works (Kedaguochuang Software Co., Ltd.) as an example, mainly studies how to use data intelligence to improve the software delivery performance and improve the operational efficiency of organizations. This dissertation, with the methods such as resource-based theory (RBT), regression analysis, and mediating effect analysis, and from the perspective of innovation management, comprehensively analyzes how software enterprises use data intelligence to improve the decision-making efficiency of organizations, improve the management, optimize the process, perfect the service, enhance the precision and control of the plan, and increase the value creation of enterprises, so as to improve the efficiency of software delivery, and finally enhance the operational efficiency of organizations. As a form of organization, software enterprises has some commonalities with other organizations, which provides an important reference for us to explore how data intelligence can improve the operational efficiency of organizations.

Chapter 2 Problems and significance

2.1 Problems

Software delivery performance refers to the efficiency of production software in continuously producing the effective value for users, reflected in effect, efficiency and sustainability. It is the dynamic balance among time, cost and quality. The main factors affecting software delivery performance are: task level evaluation, coordination, team structure, process control, process integration, and modularity. This dissertation aims to study how to use data intelligence to improve the evaluation quality, enhance the coordination ability, establish a decision-making organization, accurately control the process, optimize the management process, and make the modular system more advanced, so as to improve the delivery performance of software enterprises, and enable enterprises to develop in the long term and have lasting competitiveness. The specific problems are as follows:

- Analyze and study the connotation and characteristics of data intelligence and software delivery performance.
- What are the main factors through which data intelligence affects software delivery performance?
- What are the mechanism and path for data intelligence to improve software delivery performance?

2.2 Reasons and purposes

With the rapid development of 5G, artificial intelligence, IOT and other new technologies, enterprises develop business on software and the Internet. Software delivery performance determines the operational efficiency of software organizations. Due to the intensified market competition, enterprises have higher and higher expectations for software delivery performance, and software delivery capability has become the core competitiveness of enterprises. With the emergence of new business forms and new business modes, software becomes more and more complex, and the software delivery process also faces such problems as difficulty in collaboration among teams, lack of timely and accurate information in planning and control, increasing delivery costs, and delay in software delivery. How to effectively improve the software delivery performance has become a long-term important issue in the software industry.

Data intelligence is an interdisciplinary research field, which combines large-scale data processing, data mining, machine learning, human-computer interaction, visualization and other technologies to extract, explore and obtain revealing and operable information from data, and provide effective intelligent support for people in making decisions or executing tasks based on data. Data intelligence technologies are reshaping the traditional business analysis or business intelligence. Gartner serves more than 14,000 organizations in more than 100 countries and regions in the world. According to the survey of Gartner, a new "augmented analysis" mode is upending the old ways, and is expected to become the dominant driver of procurement of the business intelligence system within a few years. This "augmented analysis" mode is enabled by data intelligence technologies, providing core capabilities such as natural language query and narrative, augmented data preparation, automated advanced analysis, and visualization-based data exploration. This dissertation aims to study the impact of data intelligence on software delivery performance, analyze the main factors through which data intelligence affects software delivery performance, and give the mechanism and optimization path of data intelligence improving software delivery performance.

2.3 Research significance

2.3.1 Theoretical significance

The relevant papers at home and abroad have stated that data intelligence can improve the operational efficiency of organizations. It is currently a hot topic in the study on data intelligence, and has attracted extensive attention of scholars, computer experts, decision makers and entrepreneurs. However, there are few studies on how data intelligence can influence and improve the operational efficiency of organizations. This dissertation, taking software enterprises as an example, analyzes how data intelligence affects software delivery performance from the starting point that software delivery performance is the main factor affecting the operational efficiency of software organizations. This dissertation studies the mechanism and path of using data intelligence to improve software delivery performance, and the mediating effect of process integration and modularity on the impact of data intelligence on software delivery performance, and then discusses the positive impact of data intelligence on software delivery performance through process integration and modularity. The analysis on the impact of data intelligence on improving software delivery performance is helpful to deeply reveal the connotation and characteristics of data intelligence, and prove that data intelligence can improve the operational efficiency of organizations. On this basis, this dissertation provides the theoretical analysis method, which has certain guiding significance for platforms including government, army and enterprises to optimize the organizational structure and improve the operational efficiency.

2.3.2 Practical significance

Software delivery capability plays a crucial role in business development of the software industry. Finding ways to improve the productivity of software enterprises by studying the impact of data intelligence on improving software delivery performance is of overall significance for enterprises or organizations to reduce costs, increase efficiency and realize the strategic goal of scientific and technological empowerment. It is embodied in the following two aspects:

It has guiding significance for software enterprises to improve software delivery performance and operational efficiency. Analyzing and finding the indicators affecting software delivery performance by establishing a model for the effect of data intelligence on software delivery performance can effectively instruct software enterprises to evaluate software delivery capability, help software enterprises make full use of big data and other information technologies to improve their delivery performance, and actively play a role in continuously generating value for organizations. Meanwhile, the technical route of data intelligence improving software delivery performance can provide a set of feasible digital transformation paths for the software industry.

Software enterprises represent a kind of economic organizations offering services. Their operational efficiency mainly depends on software delivery performance. The path that data intelligence improves software delivery performance of software enterprises and thus improves the operational efficiency of organizations is also applicable to other organizations. The application of data intelligence technologies can also help other organizations improve the delivery performance of products or services, so as to improve the operational efficiency of other organizations. The discovery of this innovative path also has important practical guiding significance for improving the operational efficiency of other organizations in the industry and even in the society.

2.4 Innovations

In a nutshell, the main innovations of this study include:

Firstly, put forward the software delivery performance based on data intelligence technologies, and clarify the dimension and measurement system of software delivery performance. This study uses the dimension of data intelligence technologies to conduct the exploratory research on software delivery performance, and analyzes the content of software delivery performance, thus laying a foundation for the in-depth research on software delivery performance.

Secondly, explore the mechanism of data intelligence affecting software delivery performance, and construct the theoretical relationship model for the impact of data intelligence on software delivery performance. The previous studies have confirmed that big data and artificial intelligence technologies play a significant role in the software delivery performance, but they still fail to find the internal mechanism of data intelligence affecting software delivery performance, and lack the empirical test. By analyzing the influence mechanism, this study finds that process integration and modularity play an important mediating role in the process of data intelligence affecting software delivery performance; by conducting the empirical verification with structural equation modeling and other statistical methods, this study also finds that data intelligence not only directly affects organizational performance, but also indirectly affects software delivery performance through affecting process integration and modularity. Accordingly, this study constructs the multi-path model for the impact of data intelligence on software delivery performance, providing a reference for other organizations to use data intelligence to improve their operational efficiency.

Chapter 3 Literature review

3.1 Research on Resource-Based Theory

3.1.1 Enterprise resource-based theory

In 1920, Alfred Marshall created the theory of intrinsic growth of enterprises. Later, in the 1950s and 1960s, Penrose (1959) further improved this theory from the aspect of "knowledge accumulation within enterprises" through in-depth research, and believed that the function of enterprises was to acquire and organize human and non-human resources to provide products and services to the market for the purpose of profit. After the 1980s, the RBT was developed by Wernefelt, Barney and Hrebiniak on the basis of Penrose's theory. Wernefelt (1984) analyzed and explained the reasons for the success of some enterprises from the perspective of internal resources. The research results of Barney (1986) showed that the resources and capabilities of enterprises are the main reasons for maintaining product differentiation. Bar-ney (1991) further argued that the resources for maintaining sustainable competitive advantages of enterprises should be characterized by scarcity, value, inimitability and irreplaceability. Hrebiniak et al. (1985) believed that the performance of organizational decision making is a function of organizational structure, decision-making process and external environment. Enterprises should not only adapt to the changes of the external environment, but also strive to break through the constraints of external factors in order to keep enterprises in a favorable competitive position.

Amit and Shoemaker (1993) defined enterprise resources as "the stock of factors owned or controlled by an enterprise", including tradable know-how (such as patents and licenses), financial or material assets (such as property rights, factories and equipment), and human capital. They believed that enterprise resources can be transformed into final products or services by connecting with a wide range of other assets and mechanisms such as technology, management information system, incentive mechanism, and trust between managers and employees. Unlike the definition of resources, Amit and Shoamaker (1993) believed that capability refers to the ability of an enterprise to utilize resources (generally through integration and utilization processes) to effectively achieve the required results. It is a unique, information-based, and tangible or intangible process developed through long-term complex interactions among enterprise resources. It can be abstractly taken as "intermediate products" produced by an enterprise to improve its resource productivity, and strategic flexibility, and protect final products or services.

The RBT takes resources and their characteristics as its research objects. Teece (1997) further extended the dynamic thought on this capability and proposed the dynamic capability theory. The dynamic capability theory emphasizes that in order to adapt to the external environment changing rapidly, enterprises must constantly acquire and integrate the internal and external administrative organization technologies, resources and functions. Dynamic capability can continuously acquire new competitive advantages under the given path dependence and market position conditions. The dynamic capability theory is compatible with Schumpeter's "creative destruction". Innovation, especially the innovation of capability, is the real source of corporate profits. Stalk and Schulman (1992) of Boston Consulting Group put forward the capability theory based on process, believing that the key to the success of enterprises is not only the core capability, and successful enterprises pay great attention to organizational activities and business processes related to production capability, and take the improvement of these activities and processes as the primary strategic goal.

3.1.2 Data intelligence is an important resource for enterprises

Although researchers use different frameworks and angles for analyzing problems in the study on the RBT, their basic ideas are the same, that is, they believe that an enterprise is a collection of a series of capabilities, and the organic combination of resources, technologies and capabilities determines the competitive advantage of the enterprise. It is worth mentioning that among enterprise resources, a special kind of resources, namely, knowledge resources, attract more and more attention. Scholars have different understandings of knowledge. The mainstream view is that knowledge includes information and technologies. Kogut and Zander (1992) believed that knowledge includes information and know-how. Information is the knowledge that can be transferred without loss of integrity once the grammar has been known, including facts, axioms and symbols, solving the problem of "what"; know-how is the accumulated skills and expertise that make sure smooth and effective work, solving the problem of "how".

With information enriching and technology development, faced with a large amount of data and potential knowledge accumulated in the management process, data intelligence with the excellent ability to mine potential knowledge in data has become an important means of management decision making in the process of enterprise management. In combination with the direct impact of data intelligence on software delivery performance, this dissertation used the mediating effect theory, namely, the relationship between data intelligence and software delivery performance through mediating variables, deduces the correlation and degree of relation between the two. Among them, the mediating effect refers to the analysis on the influence of the independent variable X on the dependent variable Y. If the variable X affects the variable Y by influencing the variable M, the variable M is the mediating variable whose role is called the mediating effect. In this study, the author regards data intelligence as an important IT resource and studies its impact on enterprise performance.

3.1.3 Data intelligence promotes digital software delivery management

Digital transformation of software delivery management is to integrate digital technologies into software delivery process management, fundamentally improving software delivery capability, changing software delivery methods and project management modes, and providing the continuous value for customers. Digital transformation of software enterprises changes software delivery methods and management in the following three ways:

More effective and strategic communication between the team and the company. Cross-team collaboration is one of the most important and involved areas in which modern digital technologies are redefining project management. Collaborative work management software allows cross-department team members and colleagues to interact and connect in real time, thus greatly improving the efficiency of knowledge transfer between teams and the accuracy of information communication between managers, and enhancing the efficiency of project management.

Focus on results rather than process. As digital transformation automates workflows and coordinates traditional project management tasks such as planning, project managers will have more time to focus on strategy optimization and project delivery. With more digital tools and automatic processes, project managers begin to look for the best ways to align projects with their business strategies and goals and produce more successful results in the process.

More analysis to improve the project management process, results, and ROI (return on investment); Finally, digital transformation provides project managers with analysis technologies to make data-driven decisions, break down patterns and trends, make smarter decisions faster and easier, and ultimately improve project results and success rates. Strong analysis reports can help managers keep projects within budget through real-time cost and human analysis. Stakeholders and senior officers can also easily break down in-depth data sets, giving them accurate insight into the business impact and ROI of each project, and helping them plan future initiatives and make key strategic decisions.

To sum up, data intelligence is to process, analyze and mine massive data based on the big data engine and through large-scale machine learning, deep learning, knowledge mining and other technologies, extract the valuable information and knowledge contained in data, and make data "intelligent", so as to instruct organizations to conduct management improvement, process optimization, delivery collaboration, and service improvement, replace repeated decisions, enhance the decision-making efficiency and increase value creation. It has become one of the key technologies for promoting software enterprises to improve software delivery performance. At present, the studies on software delivery at home and abroad focus more on the improvement of software development process, but less on the direct relationship between data intelligence and software delivery performance, and the influence of data intelligence on the relevant intermediate variables to improve software delivery performance. Therefore, this dissertation takes the RBT as the guidance and uses the method of the mediating effect theory to carry out the research.

3.2 Research on software delivery performance

3.2.1 Performance of software organizations

(1) Basic concept of organizational performance

"Performance" originally refers to the effective and usable energy contained in things, mainly reflected in four aspects: ability, efficiency, quality and benefit. Management master Peter Drucker defined performance as the ability to choose appropriate goals and strive to achieve them.

When studying organization-related concepts, researchers deeply engaged in organizational studies hold that organizational performance is not a complete concept, but a "construct" formed through high-level abstraction after understanding and thinking of things in life. Therefore, organizational performance has always been the central issue in the field of organizational studies. It is difficult to define organizational performance. By collating a large number of literature, this dissertation summarizes the connotation of organizational performance given by domestic and foreign scholars, as shown in the table below.

Table 1 Definitions of organizational performance given by domestic and foreign scholars

Researcher	Connotation of organizational performance
Barnard (1964)	Performance is the achievement of an organization's goals.
Seashore S.E. (1965)	Organizational performance is a combination of standards to evaluate the operational quality of an organization, and a target system.
Venkatraman & Ramanujam (1986)	The core concept of organizational performance is organizational management, mainly including the output of the organization on product quality, employee satisfaction and social responsibility.
Steers (1977)	Organizational performance is a process of organizational operation. When organizational operation can minimize obstacles, the overall organizational performance can be increased.
Hall (1991)	Organizational performance is the ability of an organization to achieve its goals or maintain its functions to obtain rare and valuable resources from the environment.
Daniel R. Denison (2004)	Organizational performance is the comprehensive performance of an organization in all aspects of society, especially its output. It includes not only business performance, but also standards at the individual level within the organization, such as employees' and customers' perceptions.
Daft (2010)	Organizational performance refers to the extent to which an organization is able to achieve its goals. Organizational performance includes two aspects: one is performance, which is used to evaluate the extent to which an organization can achieve multiple goals; the other is efficiency, which means that the organization can achieve the desired output with less resource input.
Keeley (2015)	Organizational performance refers to the extent to which the interests of participants or constituent members of an organization are fairly reflected.
Li Zongyan (2002)	Organizational performance is the extent to which an organization manages to obtain the required resources, avoid the mistakes and obstacles in its internal operational process, and achieve processes and results that are more outstanding and more in line with organizational goals than other peers.

Researcher	Connotation of organizational performance
Luo Min (2009)	Organizational performance refers to the extent to which an organization achieves its goals. In the definition of organizational performance, we should pay attention to three core points: firstly, overall organizational performance, that is, organizational performance should explain the overall situation and performance of an enterprise from a more macro and comprehensive perspective; secondly, stated organizational goal, that is, the multi-level and inter-relationship among many measurement standards must be taken into account in the evaluation of operating activities of an organization; thirdly, general social expectation, that is, successful enterprises should not only focus on internal operation, but also effectively meet the requirements and expectations of different stakeholders.
Sun Shaobo (2012)	Organizational performance is based on organizational performance, but is more comprehensive than organizational performance in evaluating and measuring operating results of an organization. In addition, organizational performance and organizational performance are not completely separate, and organizational performance indicators should be used to measure organizational performance.
Yang Wenyin (2018)	Organizational performance is the extent to which the diversified goals of an organization are realized, and is a comprehensive reflection of the value of behaviors that all members or teams in an organization take to achieve the goals of the organization.

Source: collated by the author according to a large number of literature

It can be seen from the above table that researchers have different opinions on the meaning of organizational performance. Some scholars focus on elaborating the meaning of organizational performance from a certain aspect, while others are good at comprehensively defining organizational performance with the comprehensive research method. The relevant summary is shown in the table below:

Table 2 Summary of research perspectives on organizational performance

Research perspective	Representative figure
Goal achievement/organizational output	Barnard (1964), Seashore S.E.(1965), Daft(2010), Daniel R. Denison (2004)
Resource acquisition	Hall (1991), Luo Min (2009)
Quality of internal operation	Steers(1977), Sun Shaobo (2012), Yang Wenyin (2018)
Interest of stakeholder	Keeley (2015)
Comprehensive	Li Zongyan (2002), Venkatraman & Ramanujam (1986)

Source: collated by the author according to a large number of literature

It can be seen from the above summary that researchers' consensus on organizational performance mainly focuses on five aspects: firstly, from the perspective of organizational goal realization and output, researchers believe that organizational performance refers to the ability of an organization to select business goals and achieve them. Secondly, from the perspective of resource acquisition (such as input), researchers believe that the competitive advantage of an organization lies in its ability to acquire and use resources. Therefore, organizational performance is defined as the ability to acquire and utilize resources. Moreover, from the perspective of internal operation process, researchers believe that the operation of an organization is affected by a variety of factors. Among these factors, the harmonious relationship among members of the organization is the driving force for the development of the organization. Therefore, organizational performance is defined as the ability to enable members of an organization to work harmoniously. Fourthly, from the perspective of relevant stakeholders, researchers believe that meeting needs is a manifestation of organizational performance, so organizational performance is defined as the extent to which an organization meets the preferences of stakeholders. Finally, some researchers believe that organizational performance is a multi-dimensional goal collection, and a complex of the internal operation process and social relations of the system, and it should be studied in many aspects such as organizational operation and management.

This dissertation holds that, first of all, software enterprises refer to the enterprises that carry out business operations and obtain operating income with computer software products and services, and their means of production are mainly computers and all kinds of knowledge. Software enterprises are also labor-intensive enterprises, while software products are important means of production. In software enterprises, as workers, programmers and development teams need to process and recreate all kinds of knowledge in the production process. Software organizations are the organizations set up for operation within an enterprise that cooperate with each other, so the performance of software organizations mainly depends on the operational efficiency of software organizations or software teams.

Based on this, by analyzing the software development based on organizational performance and referring to the multi-dimensional viewpoints from research theories of several scholars, this dissertation puts forward that: the performance of software organizations refers to multiple effects produced by using allocated resources to carry out software development and production and services around goals of software organizations, including the realization ability and realization degree of goals. It can fully reflect the overall operation effect of software enterprises, and can be used to evaluate the overall organizational performance, and measure the operational efficiency of software organizations or teams and the buff that leads to sustainable development of enterprises in the future.

(2) Organizational efficiency and organizational performance

In the business operation of software enterprises or software organizations, efficiency and performance are very important. Efficiency is usually defined as the ability to produce work results within a specified period of time with limited resources, emphasizing the actual performance of an organization in a certain aspect. However, if a team or organization starts off in the wrong direction, even an efficient work state is ultimately futile and does not produce any performance.

(3) Organizational performance and organizational performance

Although many researchers equate organizational performance with organizational performance, more scholars still think that the two are not equal. For example, domestic scholar Lv Hongjiang (2013) believed that enterprise performance mainly includes three aspects, namely, economic representative performance, organizational performance and business performance. Through continuous research, scholars believe that organizational performance can affect enterprise performance. Scholar Zhang Meizi (2019) held that in the operation process of an organization, in addition to short-term performance, employees' work feelings and activeness as well as the enterprise's resource allocation to projects will have an impact on organizational performance.

The improvement of organizational performance of software enterprises is undoubtedly accompanied by value creation and accumulation, while value creation will involve personnel and advanced technologies. Software organizations or software teams use technical labor to carry out software (development) activities mainly for the diversified goal of corporate profit. From the perspective of dimension (internal business objectives of an organization or overall internal environment and external customer requirements of an enterprise), the main goal of software activities is to achieve fast, reliable, safe and sustainable delivery, that is, software organizations continuously produce (develop) valuable software within a short cycle (Chen L, 2015), and ensure the reliable software release at any time. Therefore, this dissertation considers that the important aspect of performance of software organizations is software delivery performance.

3.2.2 Software delivery performance

3.2.2.1 Connotation of software delivery performance

Based on the literature at home and abroad, over the years, domestic and foreign researchers have rarely conducted in-depth research on software delivery performance, but more research on organizational performance of software organizations or software enterprises. To delve into software delivery performance, we must first understand what software delivery means. Software delivery refers to the use of computer programming language to achieve a target system, namely, the activities and processes carried out for software products, with the purpose of realizing the integrated management and continuous execution of information-based business processes of business systems (Xu Pei, 2006).

Therefore, the process of software delivery involves the influence of multiple internal and external factors such as the development team, resources allocated to the team by an enterprise, and the value obtained by customers. In combination with the definition of performance of software organizations in the previous section, this dissertation believes that software delivery performance refers to the efficiency that production software can continuously produce the effective value for users.

In addition, foreign scholars usually refer to the concepts in software development research, and measure software delivery performance from two aspects, namely, process performance and product performance. Process performance (representing speed) measures the efficiency in the process from software development to delivery (J. G. Cooprider and J.C.Henderson, 1991), namely, whether a project is completed on budget and on time, while product performance (representing software quality) reflects the degree to which the system delivered meets the requirements (R. Pressman, 2001). Anandasivam Gopal (2011) used data from 83 software projects in nine Indian software companies to study how the coordination within a project team and between customers and supplier organizations affects two dimensions of performance of software projects -software quality and development speed. The results showed that both customers' (external) coordination and supplier teams' (internal) coordination have a positive impact on software quality. They also found that customers' communication barriers and customers' coordination have an impact on software quality, and the scale and coordination of software teams have a negative interactive impact on development speed.

Lianping Chen (2015) et al. pointed out in their article that adopting a continuous delivery method enables software teams to continuously produce valuable software in a short cycle and ensure the reliable software release at any time, and allows organizations to bring service improvements to market quickly, efficiently and reliably, and ultimately stay one step ahead in the competition. However, this method must use a team with strong ability, which will inevitably lead to a decline in the consistency and coordination within the team. Curtis B et al. proposed that "interruption of communication and coordination" is a major problem affecting software productivity and quality in large-scale system development. In large-scale system development, "interruption of communication and coordination" is a major problem affecting software productivity and quality, and such interruption often occurs at the boundary of organizations, which indicates that coordination among organizations is very important to the success of a project. With the gradual maturity of software projects, continuous coordination on technical details or other project information between project members and customers is required (Choudhury V, 2003; Kotlarsky J, 2005).

Therefore, the author believes that software delivery performance mainly includes: whether products can bring value to customers and companies; whether teams are able to quickly produce and release products, and quickly respond to changes in market demand and business objectives of enterprises; whether there is a sustainability as the scale of business and teams multiplies.

3.2.2.2 Analysis on factors related to software delivery performance

In the implementation process of software delivery, many factors will affect the software delivery performance, such as software quality problems and project schedule control (Zhang Xue, 2020). Low delivery performance leads to slow response to organizational demand, long delivery cycle, more accidents, and reduced service capability. Improving software delivery performance enables enterprises to quickly innovate and create, so as to achieve long-term development and maintain lasting competitiveness. In the 2018 Annual Report of DevOps titled Devops Report Focusing on Data-driven Statistical Analysis for High-performance Organizations", Dora put forward that the impact of software delivery performance has been studied and proven for many years, and proposed a set of metrics (deployment frequency, cycle & lead time, failure rate and MTTR (mean time to repair)) for software delivery performance, to measure an enterprise's software delivery level. Many different factors are considered to affect the efficiency of software development process, including programming language, use of formal methods, and CASE tools. (Aneesh Chinubhai, 2011).

The book of Nicole Forsgren titled Accelerate introduces four metrics for performance of software delivery, namely, deployment frequency, lead time to changes, change failure rate, and MTTR, to measure and visualize the speed and stability of application delivery. The 2019 Accelerate State of DevOps: Elite performance, productivity, and scaling explores practices to improve performance of software release, such as reducing the lead time to changes and increasing the deployment frequency through on-demand delivery so as to generate a strong business impact; In Improving the Software Delivery performance - A Preliminary Study of "On-demand Release", Yan Qilong et al. used lean thinking to find out all the obstacles to users in the process from demand confirmation to launch of a software project, and made adjustments according to demand from the business side, development side and testing side, so as to improve the overall efficiency of value flow and facilitate the smooth delivery of demand; in Practical Monitoring, Mike Julian highly recommended that monitoring metrics are established from the user perspective first, from which it is easier to draw conclusions than from underlying metrics: for example, response delay is felt by users; in Business-driven Lean and Agile Implementation, Alibaba proposed a framework of lean and agile research and development practice, clarifying that the direction of improving software delivery performance is to continuously and smoothly deliver the effective value with high quality, and established an actionable performance measurement system, including five metrics, namely, demand response cycle, continuous release ability, delivery throughput rate, delivery process quality and delivery quality.

From the above analysis, it can be seen that technologies and management measures applied by software organizations in the process of software development can affect the performance of software delivery.

For a long time, it has been found that small enterprises are different from large enterprises in terms of information technologies and software delivery capability. Unlike large enterprises, small enterprises are not able to simply improve their performance (Thong, J. Y, 1996). In general, small enterprises face considerably higher risks than large enterprises in implementation due to inadequate resources and limited understanding of information technologies. Although, in theory, appropriate information technologies can help enterprises develop the market, increase the sales turnover, and achieve a high profit rate, severe restrictions on financial and human resources cause small enterprises to lag behind large enterprises in the use of information technologies (Welsh, J. A,1981). In addition to resources, small enterprises often lack experience in computer technologies and don't have enough internal information technology experts. In order to solve the main factors affecting small enterprises' use of information technologies to improve software delivery capability and the impact thereof, Iacovou et al. conducted a number of case studies. The author believes that the reasons for enterprises' resistance to the impact of information technologies on software delivery capability are as follows: (1) due to insufficient utilization and lack of integration, the impact of information technologies on small enterprises is limited; (2) the complexity of information technologies of small enterprises is low; (3) the market position of small enterprises is weak and their technology is blocked. Based on a review of studies on improvement of software delivery performance and concepts and experience of small enterprises, three main factors influencing small enterprises' adoption of technologies to improve software delivery performance are identified. They are: perceived benefits; organizational preparation and external pressure.

Perceived benefits refer to the degree of recognition of comparative advantages that software delivery performance can provide for organizations. Therefore, they are divided into two types. The first type is direct benefits, which are mainly operating cost savings related to the internal efficiency of organizations; the second type is indirect benefits, which are mainly tactical and competitive advantages that have an impact on business processes and relationships. Organizational preparation refers to the level of a company's financial and technical resources. External pressure refers to the influence from the following aspects. External pressure refers to the influence from the external business environment. The former refers to the capability level of the industry where an enterprises is engaged in and its competitors, while the latter refers to the potential power and chosen influence strategies of trading partners seeking to adopt information technologies.

Melville, N and Kraemer, K proposed in 2004 that IT business value can be broadly defined as the influence of performance of IT organizations, including that at the intermediate process level and the entire organization level. Based on an analysis on the literature related to IT business value, two expressions of performance are described: efficiency and benefit, consistent with software delivery performance proposed in this dissertation. Efficiency emphasizes the use of indicators such as cost reduction and productivity improvement from an internal perspective, while benefit refers to the realization of organizational goals related to a company's external environment, which may be manifested as the realization of competitive advantages. Therefore, we can define the impact of software delivery performance as including operational impact and strategic impact. Operational impact represents the impact of software delivery performance on efficiency of business process, such as facilitating the cooperation between the internal and the external, reducing operating costs, and increasing productivity. Strategic impact represents the ability of IT technologies related to software delivery performance to support strategic objectives and create competitive advantage, such as product differentiation.

Based on the above analysis, and after reading a lot of relevant literature, the author summarizes the factors related to software delivery performance, as shown in the following table:

Table 3 Factor indicators for software delivery performance

Factor	Definition	Project	Source
Organizational performance	Refers to the actual output or results of an organization as measured against its expected output (or goals and purposes)	Improve the company's market condition Increase the company's sales Improve the company's profit rate.	Lin et al. (2013)
Perceived indirect benefits	Develop corporate strategies by building external relationships with customers and competitors	Conducive to the establishment of organizational image Increase competitive advantages Good for other business activities Improve services for customers Improve relationships with business partners	Iacovou et al.(1994)
Perceived direct benefits	Refers to an improvement of an organization's internal functions evident in daily activities.	Improve the accuracy of data Improve data security Improve operational efficiency	Iacovou et al.(1994)
Strategic performance	Strategic performance is the extent to which the IT department meets the strategic needs of the company.	Our IT infrastructure adapts easily to changes in the business process Our IT infrastructure adapts easily to technological changes We can conceive and realize applications faster than competitors	Tompson et al., 2013）

Factor	Definition	Project	Source
		We can design and realize more complex applications than competitors	
Operational performance	Refers to the measurable aspects of an organization's process results.	Provide more cost effective IT solutions for our organization. Provide more effective IT solutions for our organization. Better control the cost of IT operations. Reduce the risk of technology obsolescence. Increased the use of key information technologies.	Tompson et al., 2013）
IT performance	Improve the operational efficiency and validity of the organizational unit that deploys the target IT.	The application of the IT system improves the operational efficiency of our department. The application of the IT system improves the business process of our department. The use of the IT system improves the management of activities of our department. The application of the IT system improves the performance of our department.	Ragu-Nathan et al. (2004)

Source: collated by the author according to a large number of literature

3.2.2.3 Analysis on factors related to software delivery performance and their results

According to the actual situation of the enterprise where the author works and years of experience in the software industry, after further condensing and sublimating the main factors impacting software delivery performance, this dissertation believes that software delivery performance mainly includes two aspects: short-term performance and long-term performance.

Short-term performance is an important content of organizational performance. Organizational performance is the ultimate dependent variable of interest to researchers in almost all areas of management. Market competition for customers, inputs, and capital production makes organizational performance critical to the survival and success of modern enterprises. This structure has thus become a central object of modern industrial activities. Marketing, operations, human resources (HR), and strategy are ultimately judged according to their contributions to organizational performance (Richard et al., 2009).

Long-term performance include both perceived indirect benefits and strategic performance:

Perceived indirect benefits: among the benefits of EDI (electronic data interchange), some are strategic and some are operational. The former involves developing corporate strategy through external relationships with customers and competitors, which is classified by Iacovou et al. as "indirect interests", including improving organizational image, enhancing competitive advantage, benefiting from other business practices, improving customer service and improving relationships with business partners (Reekers et al., 1994).

Strategic performance: Strategic performance refers to the degree to which the IT department meets the strategic needs of the company. We therefore view strategic IT performance as IT's ability to make the company more flexible relative to its competitors in adapting to external pressures (Tompson et al., 2013).

3.3 Data Intelligence

3.3.1 Connotation of data intelligence

In recent years, "big data" and "artificial intelligence" have become one of the most discussed topics in academia and industry. In general, there are two perspectives on big data. One is data perspective, which believes that big data is multi-source heterogeneous information composed of tabular data, time series, space series, association networks, texts, images, and multimedia, etc., and brings about technology challenge in terms of volume, variety, velocity and veracity (Bizer et al., 2012). Therefore, big data processing technologies, especially NoSQL access technology (Cattell et al., 2011) and multimodal computing technology (Ngiam et al., 2011), have been widely studied and tried for productization. However, the big data analysis method remains at the level of data mining and visualization of analysis results for a long time. The other is decision-making perspective, namely, the so-called big data thinking (Provost et al., 2013). For example, big data should serve as a strategic assets of an enterprise; big data changes information asymmetry and drives the change of market equilibrium; big data spawns new industries and drives value creation. In particular, with respect to the management and decision-making problems under the big data environment, Chen Guoqing et al. from Tsinghua University pointed out that big data has the decision-making characteristics of granularity scaling, trans-boundary correlation and global view, and on this basis, proposed a new decision-making paradigm considering assumption transformation, trans-domain transformation and process transformation, and the PAGE framework of big data research including four dimensions, namely, paradigm, analytics, governance and enabling, greatly enriching the decision-making theory of big data (Chen Guoqing et al., 2018).

There is no accepted definition of artificial intelligence. It often refers to a machine's ability to learn from experience, adapt to new inputs, and perform human-like tasks. The terms AI and AI system were first introduced in the 1950s. Artificial intelligence is a representative of data-driven technologies (Yan Rong, 2021). With the rapid development of big data technologies, such as the improvement of computing and storage capacity and the emergence of ultra-high-speed data processing machines, the availability and energy of big data continue to contribute to the development of artificial intelligence (Duan et al., 2019). Thanks to the emergence of unstructured big data such as texts, images and multimedia in recent years, artificial intelligence can be continuously tuned and evolved in large-scale training, and even starts to be used to build and use domain general knowledge in the way of providing pre-training models (Lee et al., 2018).

To sum up, modern artificial intelligence technologies have been deeply embedded in the analysis and application of big data; the application of data-driven ideas and strategies has gradually become a consensus in practice; and the value of data has been fully demonstrated in different fields of scientific research and business. Under the big data environment, the realization of data-driven decision making is the core of the entire value chain (Barton D et al., 2012). According to a study conducted by the Massachusetts Institute of Technology, the more data-driven a company is, the better it performs in terms of financial and operational measures. In particular, the top three companies with "data-driven decision making" in the industry have the production efficiency 5% higher, and the profit rate 6% higher, on average than that of their competitors (McAfee et al., 2012). If the knowledge and information cannot be extracted from data and used effectively, data will not get its maximum value. In this context, "data intelligence" comes into being. Data intelligence comes from the evolution and collision of the word big data in the era of intelligence, and is the conceptual product of mutual integration of artificial intelligence technologies and big data technologies. Data intelligence is to realize intelligent judgment and decision making based on machine learning, data mining and big data analysis. Specifically, through data cleaning and transformation, feature extraction and integration, exploratory analysis of data and other approaches, problems are solved independently in the whole process from input to output (Yang Zhenghong et al., 2019).

From the perspective of management, Wu Junjie et al. defined data intelligence as processing and analyzing internal and external multi-source heterogeneous big data on real application scenarios through predictive analysis technologies such as large-scale data mining, machine learning and deep learning to extract valuable information or knowledge and use them to improve management and decision making in complex practical activities (Wu Junjie et al., 2020).

To sum up, data intelligence, as an interdisciplinary research field, combines large-scale data processing, data mining, machine learning, human-computer interaction, visualization and other technologies to extract, explore and obtain revealing and operable information from data, and provide effective intelligent support for people in making decisions or executing tasks based on data. Data intelligence obtains value by analyzing data, processes raw data into information and knowledge, and then converts them into decisions or actions.

3.3.2 Characteristics of data intelligence

Firstly, data intelligence is a predictive data analysis technology for big data, covering machine learning based on artificial features since the 1980s, data mining derived from databases since the 1990s, and deep learning since the early 2000s. Of course, it also includes traditional statistical analysis and visualization technologies - which need to be adapted and innovated for big data -- as well as big data acquisition and processing methods and technologies. Secondly, data intelligence is also a predictive data analysis technology for application scenarios, aiming to provide technical support for management and decision making in complex practical activities. Therefore, driven by the demand of application scenarios, some basic data analysis technologies will be innovated and integrated to form comprehensive data analysis technologies. For example, the recommendation system which has been widely used in e-commerce platforms is a typical comprehensive technology. Wu Junjie believed that big data drive and application scenario drive are the key features of data intelligence, and the key points that make data intelligence different from artificial intelligence, because the latter is not necessarily data-driven, and emphasizes more the study on general methods and technologies (Wu Junjie et al., 2020).

Business intelligence (Negash et al, 2008), which is similar to the concept of data intelligence, integrated data, algorithm and scenario when it was proposed, laying a good foundation for putting forward the concept of data intelligence in the era of big data and artificial intelligence. However, the author holds that algorithm and scenario are more inclined to be technology and prediction and decision-making capability, making it more convenient to measure the key characteristics of data intelligence.

Data (massive, and multi-type). From the perspective of data, data intelligence is applied to multi-source heterogeneous big data in the real sense, that is, big data emerging in such application fields as texts, networks, time series, and space series is the main analysis object.

According to a recent report of McKinsey, organizations in all industries except labor force, capital and land, regard data as an important factor in production. Organizations in the past focus on enterprise-specific structured data (namely, data that can be stored in relational databases) in making business decisions, while today's organizations tend to capture every bit of information, regardless of the scale of data, the structure of data, and the speed at which data is created (Manyika et al., 2011).

Technology (algorithm). Big data processing technologies used by data intelligence, including cutting-edge machine learning and deep learning methods, are the core technologies to realize big data analysis. New forms of data require new technologies to cope with the challenges posed by huge, diverse and fast-moving data. As a result, organizations have begun to move beyond traditional RDBMS method for storing and analyzing data (Gupta and George, 2016). In 2015, Apple acquired FoundationDB, a company that produces NoSQL databases, further emphasizing how important these new forms of technology are for organizations interested in exploring big data

(TechCrunch, 2015).

Prediction and decision-making capability (value). The expansion of data intelligence in the application field, such as intelligent city, intelligent finance, intelligent manufacturing, and intelligent medical treatment, is also the typical application scenario of data intelligence, and is the most important manifestation of the value of data intelligence. Data intelligence aims to help carry out prediction and decision making at the deep level, rather than stay at the level of analysis and presentation of data science (Yang Zhenghong et al., 2019). Prediction and decision-making capability is "the process of using a set of complex tools to develop models and estimate the future environment" (Wessler, 2013). It can apply multiple statistical analysis methods, modeling, machine learning, and data mining to both structured and unstructured data to determine future results (Y. Wang et al., 2019).

3.3.3 Application and research trend of data intelligence

Data intelligence technologies give the ability to explore the unknown parts of the data space, and spawn huge opportunities in different fields. Many new Internet-based business, including search engines, e-commerce and social media applications, are essentially built and operated on the basis of data intelligence. Data intelligence technologies are reshaping the traditional business analysis or business intelligence. According to the survey of Gartner, a new "augmented analysis" mode is upending the old ways, and is expected to become the dominant driver of procurement of the business intelligence system within a few years. This "augmented analysis" mode is enabled by data intelligence technologies, providing core capabilities such as natural language query and narrative, augmented data preparation, automated advanced analysis, and visualization-based data exploration.

In China, Baidu put forward the concept of data intelligence in 2014 that refers to the processing, analysis and mining of massive data based on big data engines and through large-scale machine learning, deep learning and other technologies to extract valuable information and knowledge contained in data, make data "intelligent", and establish models to seek solutions to existing problems an achieve prediction. Mu Hong, vice president of 360, also pointed out: 360's positioning is to take data intelligence as the core. Many ways and methods of artificial intelligence that can be used at present are related to big data. Many solutions mainly collect all kinds of data, make comprehensive research and judgment based on these data, and make prediction and early warning and provide models after the research and judgment. The key is to make enough detailed scenarios and get enough data. Alibaba Cloud believes: the more intelligent data is, the more intelligent business is. As data intelligence helps more and more enterprises gain more profits in business, data intelligence is of great significance in the current business development.

Data intelligence is widely used in such fields as business, electricity, meteorology, and hydrology. Big data intelligence has been seen as a disruptive technology that will reshape business intelligence, especially marketing intelligence (J Leon Zhao et al., 2014); one of the key factors determining the success of intelligent electric meters is data analysis of intelligent electric meters, including data acquisition, transmission, processing and interpretation (Damminda Alahakoon et al., 2015); data intelligence is most widely applied in the fields where data is used as the key factor in decision making such as meteorology and hydrology, and data intelligence models have obvious advantages (Fu Minglei et al., 2020; Zaher Mundher Yaseen et al., 2018; Khabat Khosravi et al., 2019).

In addition, the application of data intelligence in software industry management is gradually deepening. Big data technologies collect, organize, analyze and apply data and information in all links, comprehensively improving the management efficiency of software projects in the new era, and providing basic data guarantee for software project management (Zhang Jianying, Wang Gang et al., 2020). Big data technologies can make software project management more convenient and innovative (Ji Yuan, 2018). Jagtiani, J found in the research that open source big data can solve the problems of obtaining reliable data and relevant indicators in research, developing practical estimation models and influencing project improvement in the industry, and proposed that big data technologies can help improve traditional software project management practices (Jagtiani, J, 2018). Bakici, T improved the frequency of using big data in project cost management through BIM technologies, thus reducing project management costs and improving the project value (Bakici, T, 2018).

Through the literature analysis, it can be found that the previous research focused on the collection of big data, namely, data mining. However, with the gradual deepening of the research, more and more researches are devoted to data processing, such as data analysis, data visualization, predictive value of data and intelligent decision-making supported by data, divide the development process of data intelligence, and make it clear that the realization of data intelligence focuses more on the future-oriented value, that is, prediction of and help for making the next action decision that will also be the trend of data intelligence research.

3.4 Software project management

3.4.1 Connotation of software project management

3.4.1.1 Project management

In the sixth edition of A Guide to the Project Management Body of Knowledge published in 2017, the Project Management Institute (PMI) mentioned that a project is a temporary work carried out to create a unique product, service or result; the purpose of carrying out a project is to achieve the goal through the delivery of results. A project can be carried out at any level of an organization. A project may involve only one person or a group of people; it may involve only one organizational unit, or multiple units of multiple organizations.

From the perspective of business value, a project aims to move an organization to change from one state to another in order to achieve specific goals. Before a project begins, it is common to describe an organization at this time point as the "current state". Project-driven changes are made to achieve the desired result, namely, the "future state". Meanwhile, the factors such as laws, regulations or social requirements, requirements or needs of customers or related parties, implementation or change of business or technology strategies, creation or improvement or repair of products and processes and services will affect an organization's continuous operation and business strategies, so it has to find effective ways, enabling it to successfully make the necessary changes in response to these factors.

Project management is the application of knowledge, skills, tools and technologies to project activities to meet the requirements of a project. Project management is realized through the rational use and integration of the project management process required by a specific project. Project management enables an organization to carry out its projects effectively and efficiently. (PMBOOK, PMI, 2017). Project management involves nine aspects, namely, project scope management, project quality management, project cost management, project time management, project risk management, project communication management, project procurement management, and project comprehensive management, and five processes, namely, project starting process, planning process, implementation process, monitoring process and closing process (Song Lihong, 2006). It not only involves the demand and feasibility study in the early stage of a project, but also comprehensive management implemented in aspects such as project implementation time, project quality, cost and manpower in the whole life cycle from project planning to project completion and acceptance. Therefore, the process control, technologies and methods of project management can help software enterprises deal with complex problems encountered in software projects, and effectively help software enterprises improve the management performance of software projects.

3.4.1.2 Software project management

Generally speaking, there are two types of software products: software product development, and customized development of software products. The former means a software company creatively proposes a product that is better than the products on the market or has a specific function through its own conditions and in-depth study of software products on the market. This type of product is inherent to the software company and may appear in the software market to serve customers at any time. The latter means customized development based on customer needs, after making demand analysis to understand the customer's expected functions of the product. The latter is the system engineering from software design, code writing, software testing to delivery, operation and maintenance. For any kind of software products, external learning and demand analysis is required before software development. Therefore, this dissertation holds that a software project refers to a complete process from software development to delivery, including the preliminary research of the project, the confirmation of the customer's needs, the system engineering from software design, coding, and software testing to delivery, operation and maintenance, and the software product completed within the agreed time to achieves functions by means of implementation (Xie Wenjuan, 2020).

In the early stage of the rise of software, many software projects could not be completed according to the plan or according to the demand, mainly due to poor software project management as found through statistical data (Chen Xinxin, 2012), so the concept of software project management was born in the United States in the 1970s. A software project is a collection of practices established to achieve one or more predefined goals (Team, C. P., 2010), while software project management, known as the art and science of planning and leading software projects, requires knowledge of the entire software development life cycle: definition of visions, planning of tasks, gathering of staff, evaluation, schedule creation, monitoring, gathering of requirements, software design and programming, and testing of the final product (Stellman, A, 2015). Therefore, in order to make the products of software projects more reliable and stable, reduce unnecessary waste, and eliminate the influence of uncertain factors in the whole process, it is necessary to manage the whole life cycle of software projects.

Software project management is actually the management of different stage such as requirements, design, coding, software testing, and software launching, and is analysis and management activities carried out for the software organization or team, technology, and process in order to ensure that software projects can be completed and deliver high-quality software products to customers in accordance with the established target time and software quality (function). Software project management is mainly to maximize the utilization of resources, so that customers can obtain the functions they need and realize the expected value, and at the same time, software enterprises can also obtain successful experience and relevant benefits from projects. From this aspect, the connotation of software project management is that it is used to improve the software delivery performance.

3.4.2 Overview of software project management

In order to better manage software projects, many experts and scholars in the fields of management and software engineering have done a lot of research, and put forward some standards, models and methods for software projects, solving the problems encountered in the process of software project management to a large extent, such as project cost, project schedule, and software quality.

3.4.2.1 Standards for software project management

(1) ISO9000 management system

ISO9000 was first published by the International Organization for Standardization (ISO) in 1987, aiming to help software enterprises ensure that they meet the needs of customers and other stakeholders, and meet the requirements of laws and regulations for software products or projects, and standardize their organizational structure, resource allocation, technical capability, rules and regulations (Wang Xuyang, 2021). At present, in order to actively promote the standardization and institutionalization of software management, ISO9000 has derived a series of standards, such as ISO9001, ISO9002, ISO90003 and ISO9004.

ISO9001 specifies in detail the requirements that organizations need to meet to apply for the certification of software quality standards and the reference documents for measuring software quality in the process of software design, development, production and release.

ISO9002 provides quality assurance for software production, installation and service.

ISO9003 provides quality assurance for the final inspection and testing of software.

ISO9004 provides a quality guide for management systems, helping software companies identify and meet the requirements of customers and other stakeholders and improve software delivery capability and performance.

(2) CMMI certification system

CMMI (Capability Maturity Model Integration) was proposed by the Carnegie Mellon University in 1984 as the certification specifically aimed at the management standards and quality of software products (Zhuang Xiao, 2013). It can provide software enterprises with process improvement, improvement of product development efficiency and quality, and maintenance of product service capabilities.

Levels of CMMI

CMMI is divided into five levels: initial level (CMMI1), management level (CMMI2), definition level (CMMI3), quantitative management level (CMMI4), and optimization level (CMNI5). The greater the number is, the higher maturity the model has.

The initial level is characterized by the uncontrollable process of software development and project implementation. Many software enterprises in the early start-up stage are at this level. At this level, the success of an organization often depends on the individual capability of the project leader.

The management level is characterized by certain management capability. At this level, project requirements, process control and project results will be supervised, and project managers can know the progress and result of the project through the completion of tasks.

The definition level means that the process of software project management has been standardized or institutionalized. At this level, an organization can properly control each process according to the needs of project management.

The quantitative management level is to quantify the objectives of a software project, and establish process standards, to evaluate the benefits generated in the process of the project.

The optimization level is the highest level. At this level, software enterprises or organizations have very clear goals, make continuous optimization and improvement of the fourth level, and focus on improving the project process, enhancing the quality of software products, absorbing experience and avoiding mistakes in other projects.

2) Role of CMMI

Due to the small scale and limited resources, many software enterprises find it difficult to break through and improve the quality of their software products. Therefore, many enterprises adopt standard models to control and adjust the software process and improve the service quality of software products (Goncalves, 2018).

The improvement of software development process is an effective method to improve the quality of software products (Riaz M N, 2017). Although many studies and industry reports have shown that software process improvement plays a significant role in improving software products and quality, there are still problems and difficulties in the planning and implementation of process improvement in a multi-modal environment (Wang Wenjin, 2017), increasing the risk of failure especially for small and medium-sized enterprises. After the use of the CMMI model, many enterprises have found that it can solve the above problems and reduce the risk of failure, so it is widely used (Ardana I M S, 2017). According to the analysis on actual cases of objective and process improvement by organizations, the main improvement objectives of organizations in the process of software development improvement of enterprises focus on three basic aspects: improving quality, reducing development costs and reducing product defects, and the processes that need to be improved in such three aspects correspond to the Process and Product Quality Assurance (PPQA), Requirements Management (REQM) and Project Monitoring and Control (PMC) (Chavarria A E, 2016). According to the results of the implementation of this scheme, in the software development through the CMMI model, significant improvements were made in reducing quality costs and defects; at the same time, the risk of project failure was correspondingly reduced, and profitability and customer satisfaction were improved (Goncalves T G, 2016).

The above results show that the application of the CMMI model in the process of software development can effectively improve the development efficiency and development management level, showing broad application prospects of the CMMI model (Silva L S P, 2016).

3.4.2.2 Model and method for software project management

(1) Waterfall model

The "waterfall model" was first proposed by Winston Royce (1970) in Managing the Development of Larger Software Systems. Up to now, it is still used by many project managers of software enterprises in software development.

The core idea of the waterfall model is to divide the tasks of a software project and divide the whole life cycle of a software project into multiple stages (Lalman, 2013). As shown in the figure below, a project is carried out in the strict sequence from top to bottom.

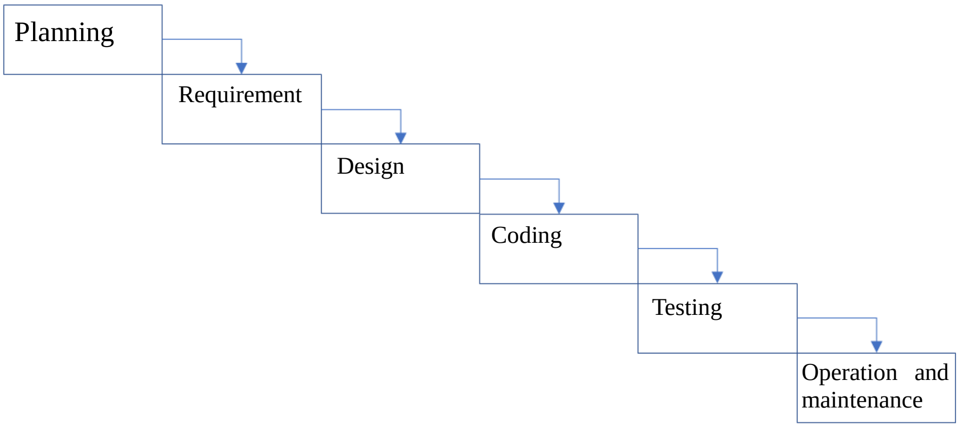


Figure 1 Waterfall development model

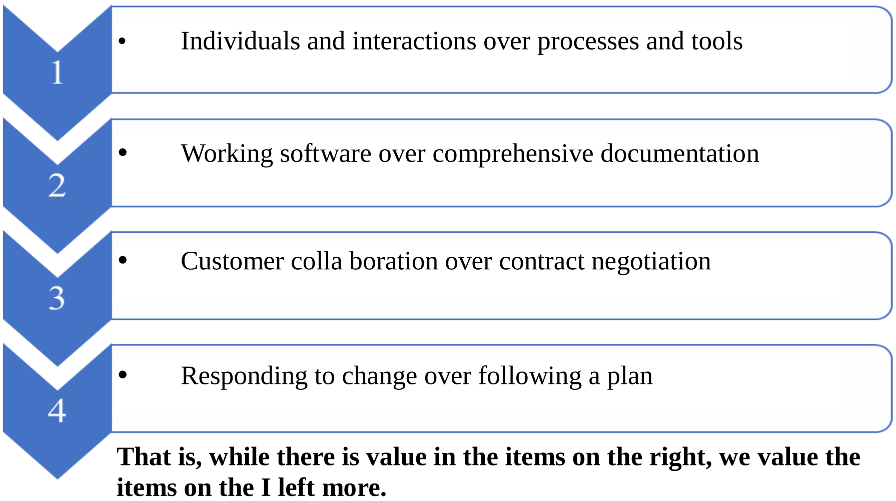
The waterfall model is the longest and most widely used project management model. It has the advantages of strong normalization, strong planning and clear responsibilities. Under the waterfall model, work must be carried out in order and only one activity can be executed at a time; the next phase of work cannot begin until the previous phase of work is completed; so, the work of team members can be restricted and planned, enabling them to perform their duties in accordance with the established plans and objectives.

These points also lead to some shortcomings of the waterfall model: when the requirements change, the change under the waterfall model will be very complicated, and the design process is too much, resulting in the low project efficiency; in addition, testing activities need to be performed after development activities are completed, which may delay error resolution.

(2) Agile development model

The agile development model is also known as agile development method. Since the 1990s, with the continuous refinement and change of market demand and customer demand, software development teams and software enterprises need to quickly adapt to changes in customer demand, market environment, products and processes (Ramin F, 2020). In 2001, senior experts in the software industry in the United States jointly discussed software development models, first proposed "agile development" and agreed to take "agile development" as a basic concept of software project management (ADBOK Writing Group of China Agile Software Development Alliance, 2013).

The core idea of agile development (Beck K, 2001) is shown below:


1
Individuals and interactions over processes and tools
2
Working software over comprehensive documentation
3
Customer colla boration over contract negotiation
4
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the I left more.

Figure 2 Core idea of agile development

Firstly, communication between individuals is far better than processes and tools. Communication between members within a team is more important than robust processes and advanced tools.

Secondly, available software is far better than complete documentation. Documentation in the software development process is only an auxiliary tool. A complete program or product delivered to customers is the most effective.

Thirdly, customer collaboration is better than negotiation. In the process of software project development and management, more attention should be paid to the communication and cooperation between the customer and the team, rather than focusing on contract negotiation. All parties should cooperate to ensure that the goals and expectations are consistent.

Fourthly, responding to changes is better than following a plan. In the implementation process of a software project, many factors such as policies and regulations, technologies, organizational structure, and customer requirements may change, which requires the manager to change the plan according to the actual situation until the planned goal is completed.

12 principles for agile development are as follows:

Table 4 Principles for agile development

NO.	Items
1	Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2	Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3	Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4 Business people and developers must work together daily throughout the project

5 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done

6 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation

7 Working software is the primary measure of progress.

8 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely

9 Continuous attention to technical excellence and good design enhances agility

10 Simplicity--the art of maximizing the amount of work not done--is essential

11 The best architectures, requirements, and designs emerge from self-organizing teams.

12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Source: Manifesto for agile software development, 2001.

3.4.2.3 Content and role of software project management

(1) Content of software project management

A software project is a complex system engineering. During software project development, compared with unmanaged or poorly managed teams, managed and coordinated teams or organizations can often accomplish more work faster, at higher quality, and at lower cost, and can bring better organizational performance, indirect benefits, and strategic performance. Our highest goal is to meet customer requirements through the rapid and continuous delivery of valuable software.

In combination with the above summary of the methods and models for software project management and the understanding of the connotation of software project management, the content of software project management mainly includes:

1) One of the challenges faced by software project management is to evaluate tasks of a project, so that the software enterprise can develop cost estimation of the project, and invest reasonable human and material resources;

2) For every software project management, developers must be used. Only developers form a small team to work together, can a perfect project is achieved. Software project management is to optimize the organizational structure of a software team, so that everyone has their own clear responsibilities and task arrangements, and jobs according to their abilities and fields of expertise, and can fulfill their duties and get twice the result with half the effort;

3) Within each software team, members strive to achieve multi-layer project goals, but sometimes these goals are incompatible or even in conflict with each other. Therefore, software project management is to coordinate the work between business analysts and front-line developers, motivate and lead team members, cultivate team spirit, and shape an efficient team;

4) At the beginning of a software project, there will be a requirement, which is also the goal of the project. During the development process, the requirement

may change. Software project management is to change and improve the plan according to the proposed requirement changes. Being good at adjusting team behaviors by taking advantage of requirement changes to help customers gain value can also improve the adaptability of our team.

5) In our organization, software project management is to make a detailed and thorough plan, so that the cost and schedule of our project can be effectively controlled;

6) The software project process generally goes through the planning stage, requirement stage, implementation stage and acceptance stage. Software project management is to optimize the work flow, clarify the objectives of each stage, and reduce unnecessary work as far as possible.

(2) Role of software project management

Although software project management has certain particularity compared with project management in other industries, it also belongs to project management. Therefore, in a software research and development project, if the project management is poor without the scientific and reasonable planning and implementation plan, the quality and delivery time of software products may not reach the expected, and the software project may even end in failure (Zhang Chengye, 2015). The role of software project management can be summarized in six aspects, as shown in the following figure.

Through analysis, it can be seen that software project management can directly affect the quality of software products, and customer satisfaction, etc. that are the embodiment of software delivery performance. Software delivery performance can directly affect the performance and long-term interests of software enterprises. Therefore, the good management and ability of software projects can improve the competitive advantage of software enterprises, the quality of services for customers, and the performance of enterprises and help the implementation of other business behaviors of enterprises.

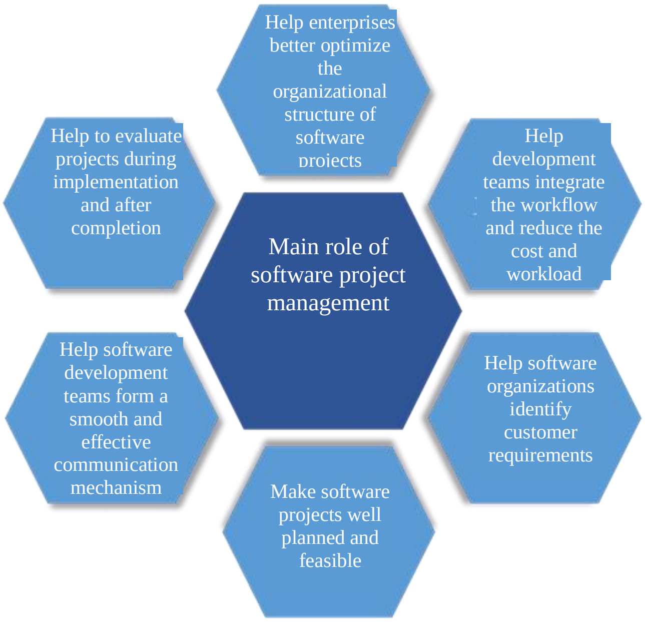

Help enterprises better optimize the organizational structure of software projects
Help to evaluate projects during implementation and after completion
Help development teams integrate the workflow and reduce the cost and workload
Main role of software project management
Help software development teams form a smooth and effective communication mechanism
Help software organizations identify customer requirements
Make software projects well planned and feasible

Figure 3 Main role of software project management

3.4.3 Core factors of software project management

By studying the literature on software delivery efficiency, data intelligence and application, the author finds that the factors affecting software delivery performance mainly include organizational structure, workflow, requirement management, schedule planning, staff motivation, corporate culture, and post-project assessment (Zhu Daogang, 2020; Xie Wenjuan, 2020; Zhao Yuehua, 2017; Song Huilin, 2017; Wang Zhaojuan, 2009; Liu Xuan, 2010). As a method and technology, data intelligence will support and drive the realization of application scenarios in all walks of life, including post-project assessment, organizational structure, production technology improvement, production process optimization, and intelligent comprehensive decision making (Shao Zhou, 2020; Sun Xinbo et al., 2019; Wang Xiwen, 2016; Zheng Shuquan et al., 2017). Therefore, in combination with the characteristics of software project management and software project delivery, as well as the research status of data intelligence and software delivery performance, this dissertation sets task level evaluation, coordination, team structure, process control, process integration and modularity as mediating variables.

(1) Task level evaluation

Task level evaluation is to provide multiple warnings for decision makers or managers of software projects to choose task levels and implementation plans, requiring objective and accurate collection and true and complete presentation of information related to software project development such as resources, technology, market, finance, economy and society to managers, so that software managers can keep an eye on the overall situation and make correct decisions. The significance of task level evaluation is to objectively evaluate all task-level work, thereby improving the performance of task-level delivery.

(2) Coordination

A software project involves personnel, resources, cost, time, and product quality, etc., so coordination in software development is not limited to the internal team, but involves five internal and external aspects of the entire organization, namely, individual, team, project, company and business environment.

(3) Team structure

The core of a software development project is team members. A complete project cannot be completed without the participation of team members. Similarly, for a software project team, it is impossible to complete a project in the absence of an important role in the development process. The completion of a software project requires the deployment of the team's work, the integration of business technologies, the improvement of the team's response speed, and the reasonable allocation of resources. Therefore, a complete and efficient team structure has a significant impact on software delivery.

(4) Process control

The process of software from development to delivery is an orderly work under the requirements of the institutional framework. Establishing an accurate and reasonable project management process enables software project managers to make a plan for what technology is needed to realize software functions before the development. By strengthening process control, software project managers can effectively utilize resources of software enterprises, reduce project risks, improve the efficiency and quality of project development, start and deliver projects on time, and make software better meet the requirements of customers and stakeholders at the time of delivery.

(5) Process integration

Process integration refers to a measure to maintain an enterprise's competitive advantage through the continuous development, improvement, and integration of business processes, including the process of sorting out, improving, and integrating existing workflows. It essentially optimizes and redesigns the business process in order to achieve breakthroughs in software quality, project cost, development speed, product quality and service. Software delivery process includes software development process and project management process. In the software delivery process, it needs to dynamically balance the relationship among time, cost, and quality around delivered software capabilities to achieve the goal of consistently generating the effective value for users. Therefore, the improvement of software delivery performance requires process integration according to the requirements of the delivered product and the software delivery capability of the software enterprise.

(6) Modularity

The principle and process of modularity is very simple. At its core is the replacement of formal and strict specifications with interactive and quickly established modules. Customers can provide real and concrete feedback for developers through the modularity of hands-on operation and test on computers.

Modularity: by showing customers the modularity of actual operations, customers can "see and touch" the system, and give very clear opinions on the system, improving the communication between customers and developers. Using modularity, customers play a leading role in the process of system development, and can help developers understand the real requirements of customers, and more effectively identify user requirements, which can not only shorten the time of requirement analysis, but also reduce developers' misunderstanding of user requirements, so as to significantly reduce the time of requirement change, improve the quality of requirement analysis and reduce the risk of system development. Based on the above, software enterprises can use modularity to improve the development speed of software projects or products, and thus enhance the software delivery performance.

3.5 Literature summary

3.5.1 Literature contribution

After reviewing extensive literature on resource-based theory, data intelligence, software project management, and software delivery performance, we regarded data intelligence, software project management methods and experiences including task level evaluation, coordination, team structure, process control, process integration, and modularity as an important resource for modern enterprises under the framework of resource-based theory, and obtained the following table of variables.

Table 5 Variables

3.5.2 Insufficient literature

Independent variable	Intermediate variable	Dependent variable
Data intelligence DI (Data D, Technology T, Prediction and decision- making capability P)	Task level evaluation TLE	Software delivery performance SDP (Short-term Performance SP, Long-term benefits LB)
	Coordination C	
	Team structure TS	
	Process control PC	
	Process integration PI	
	Modularity M	

- Data intelligence is a new concept, while the existing studies on data intelligence were mostly conducted from a technical perspective, and few examined its influence on management.
- Software delivery performance was obtained by measuring technical indicators in most of the studies, and there is a lack of management indicators.

3.5.3 Content of research

The research was carried out to examine the mechanism and path of data intelligence to improve software delivery performance and to extend it to organizations of other industries. The impact of data intelligence on organizational performance was investigated from an integrated perspective of technology and management.

Chapter 4 Theory and hypothesis

4.1 Application of RBT in this dissertation

Resource-based theory (RBT)

The core basis of the resource-based theory (RBT) is to assume that organizations compete with other organizations based on their own resources and capabilities and are rational when making decisions to select and accumulate resources. Organizational resources are considered as a set of attributes that can effectively help an organization compete in the market and achieve its vision, mission, strategy and goals.

In this research, IT technologies and methods (including data intelligence), management (software project management methods and experiences including task level evaluation, coordination, team structure, process control, process integration, modularity) as an important resource for modern enterprises. Through data analysis, a model was established to investigate how organizational resources can be transformed into a competitive advantage that is rare, difficult to imitate, and irreplaceable by other resources. Specifically, if data intelligence has a positive impact on software delivery performance, it will definitely form the competitiveness of an enterprise and realize the vision and strategic goals of the enterprise.

4.2 Model used to study the impact of data intelligence on software delivery performance

Therefore, this study sets data intelligence as an independent variable, sets task level evaluation, coordination, team structure, process control, process integration and modularity as intermediate variables, and sets software delivery performance as a dependent variable to analyze the direct impact of data intelligence on software delivery performance, and how data intelligence affects software delivery performance through intermediate variables, and then analyzes the relationship between data intelligence and software delivery performance. The research model is obtained as follows:

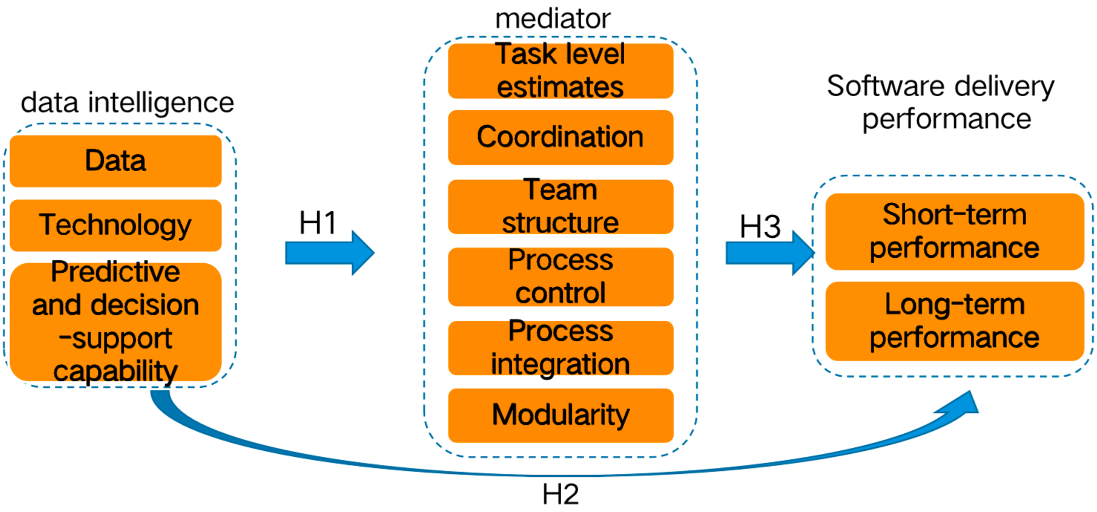


Figure 4 Model used to study the impact of data intelligence on software delivery performance

Note: Hypothesis H1: Data intelligence has a positive impact on mediating variables. Hypothesis H2: Data intelligence has a direct positive impact on software delivery performance. Hypothesis H3: Mediating variables have a positive impact on software delivery performance.

4.3 Hypotheses

Through literature review and theoretical research, we can see that data intelligence can play a positive role in improving the software delivery performance As there are many factors affecting software delivery performance, this study focuses on analyzing the direct impact of data intelligence on software delivery performance in light of the characteristics of software projects and software delivery process; and selects task level evaluation, coordination, team structure, process control, process integration, and modularity as the mediating variables between data intelligence and software delivery performance to analyze how data intelligence affects software delivery performance through mediating variables, and the relationship between data intelligence and software delivery performance.

4.3.1 Proposal of hypothesis

This dissertation mainly studies the relationship between data intelligence and software delivery performance. Based on literature review and theoretical research, we propose three hypotheses.

H1: data intelligence has a positive impact on mediating variables, including:

H1TLE: data intelligence has a positive impact on task level evaluation.

H1C: data intelligence has a positive impact on coordination.

H1TS: data intelligence has a positive impact on team structure.

H1PC: data intelligence has a positive impact on process control.

H1PI: data intelligence has a positive impact on process integration.

H1M: data intelligence has a positive impact on modularity.

H2: data intelligence has a positive impact on software delivery performance.

H3: mediating variables have a positive impact on software delivery performance, including:

H3TLE: task level evaluation has a positive impact on software delivery performance.

H3C: coordination has a positive impact on software delivery performance.

H3TS: team structure has a positive impact on software delivery performance.

H3PC: process control has a positive impact on software delivery performance.

H3PI: process integration has a positive impact on software delivery performance.

H3M: modularity has a positive impact on software delivery performance.

The list is as follows:

Table 6 Hypotheses on the impact of data intelligence on software delivery performance

H1	H2	H3
H1TLE: data intelligence has a positive impact on task level evaluation	H2: data intelligence has a positive impact on software delivery performance	H3TLE: task level evaluation has a positive impact on software delivery performance
H1C: data intelligence has a positive impact on coordination		H3C: coordination has a positive impact on software delivery performance
H1TS: data intelligence has a positive impact on team structure		H3TS: team structure has a positive impact on software delivery performance
H1PC: data intelligence has a positive impact on process		H3PC: process control has a positive impact on software

control		delivery performance
H1PI: data intelligence has a positive impact on process integration		H3PI: process integration has a positive impact on software delivery performance
H1M: data intelligence has a positive impact on modularity		H3M: modularity has a positive impact on software delivery performance

4.3.1.1 Data Intelligence has a positive impact on task level evaluation in software delivery

Task level evaluation is the systematical value evaluation for the organizational structure, function, environmental compatibility, operability and sustainability of tasks in the process of software delivery. Its main contents include investment necessity evaluation, construction condition evaluation, technology evaluation, and economic data evaluation. Task level evaluation is divided into pre-task, mid-task and post-task evaluations by the task stage and software delivery process. Pre-task evaluation means that when a task is started, the risk and cost faced by the task are evaluated according to task requirements, task construction contents, task stakeholders and other factors, which can improve the predictability of the expected effect of the task, provide a scientific basis for the task planning, and benefit the decision making of the organization. Mid-task evaluation means that in the requirement and overall design stage, the use of funds, progress, quality, and organizational management of the task are evaluated according to task requirements, overall design planning, actual construction situation of the task and difficulties faced by the task, so as to provide a scientific basis for the adjustment of the task plan. Post-task evaluation is carried out near the end or after the completion of a task to systematically and objectively analyze the achievement of the goal, implementation process, benefits, role and influence of the completed task, and provide effective process feedback, improving the decision making level of task management.

Software task evaluation under the digital transformation of software delivery management is the process of objectively and accurately collecting, analyzing and evaluating comprehensive data on the internal status such as progress, quality, cost, organization and management in the whole task cycle and the external status such as the achievement of goals, changes in the technology environment and the degree of emphasis and support of the undertaking entity by using the corresponding information technologies, methods and means, thus continuously improving the management decision-making level of the task implementer.

Based on the big data engine, data intelligence can obtain, process, analyze and mine the data involved in a task in a comprehensive and rapid way, and extract the valuable information contained in data, enabling decision makers to grasp the actual situation of the task accurately and timely. By using data intelligence for software tasks, task-related information and data can be obtained comprehensively and quickly, and the authenticity of information can be compared and judged, to avoid the delay in task progress or too high cost caused by information asymmetry; through big data and artificial intelligence technologies, multiple tools such as scenario analysis and Monte Carlo simulation can be well used to deeply analyze task risks, accurately judge risk points, and calculate the probability of financial indicators falling short of expectations (Zheng Honggen, 2018); in the process of software development, if there is no good evaluation model or calculation method, the project evaluation or estimate will be inaccurate, resulting in project delay or change, and increasing the risk of project implementation encountering problems. At present, the common project evaluation methods mainly include project cost evaluation and project risk evaluation. At first, the project cost evaluation method mainly relied on the expert system. Later, Symons proposed the function point method (1988), which, from the perspective of users, measures the software scale and estimates the software cost throughout the whole life cycle by quantifying system functions and clarifying the scale of function point data and code based on the system design. With the development of artificial intelligence technologies, methods based on machine learning have been applied to software cost estimation, mainly including neural network (Kaushik A, Soni A K, Soni R, 2014), (Rijwani P, Jain S, 2016), Support Vector Regression (SVR) (Song L, Minku Li, Yao X, 2013), Case Based Reasoning (CBR) (Mendes E, Watson I, Triggs C, 2002), genetic algorithm (Gharehchopogh F S, Pouralia, 2015), heuristic algorithm (Kumari S, Pushkar S, 2017; Rao, G. S., Krishna, C. V. P., & Rao, K. R.,2014), and fuzzy logic (Du W L, Ho D, Capretzl F, 2015; Mittal A, 2010; Du W L, Capretzlf; 2013), etc.

The application of these algorithms to software project evaluation requires a large amount of data, mainly from development teams in project management data (including the number of personnel, and the composition of personnel), products (function points, number of pages, number of database tables, and number of lines of code), and implementation performance requirements (number of users, number of concurrent users, and minimum response time).

According to the above discussion, software enterprises can use data intelligence to carry out task evaluation more effectively, accurately understand the matching degree and the technology matching degree between task organizations and tasks, and the rationality of schedule arrangement, clearly understand task risks, and improve the software delivery capability. Therefore, we propose the following hypothesis:

H1TLE: data intelligence has a positive impact on task level evaluation.

4.3.1.2 Data Intelligence has a positive impact on coordination of software delivery process

Coordination is to correctly handle all kinds of relations inside and outside organizations, create good conditions and environment for the normal operation of organizations, and promote the realization of organizational goals. The PMBOK Guide prepared by the PMI defines project coordination management as a process necessary for ensuring timely and appropriate generation, collection, dissemination, storage, retrieval, and final disposal of project information. It provides the critical connection between people and information necessary for success. The coordination management in the process of software delivery mainly refers to the information transmission and understanding between the project team and other organizations and between project team members. It runs through the whole life cycle of the software project and plays a very important role in the successful delivery of the project.

Healey et al. (2009) pointed out in their initial study that: "the responsibility for addressing many important challenges of our time has fallen on collaboration between different groups. Tasks as complex as drug development projects, complex engineering projects, and military operations can no longer be accomplished by just several persons, teams, or organizations, but rather by collaboration among groups with a common goal."

Hoegl et al. (2005) combined multi-team with project research and development in the study on multi-team interdependence in new products, and defined the multi-team system as "a system consisting of interdependent unit teams, each of which is only responsible for completing a certain component and cooperating and communicating with other unit teams to complete the entire product system".

At present, in order to reduce the management complexity caused by the large software scale, most software projects are implemented under the multi-team cooperation mode, and the realization of project objectives largely depends on the process coordination among teams.

The coordination in the process of software delivery is mainly divided into two aspects: one is the coordination in the delivery process, namely, the coordination of personnel involved in the process of software delivery from customer requirements to software production. With knowledge as the core content, the coordination aims to better process knowledge and improve the performance of the delivery process. The data required by this kind of collaboration is the knowledge required in the software production process, such as requirements, system architecture, detailed design, and program interface, mainly in the form of knowledge transfer into explanatory texts and related data (such as function points, and performance indicators). For example, coordination between customers and required personnel, and between requirements and designers. The other is management coordination, namely, the coordination between the production process and the manager in the process of software delivery. With information in the management process as the core content, including the process plan, personnel allocation and task allocation, and quality requirement data, the coordination aims to accurately reflect the progress, quality and cost data of the project to facilitate the response to the risks affecting the project delivery, and effectively control the implementation of the software delivery process to improve the performance of delivery management. For example, coordination between project managers and project team members, and between customers and project managers.

In process collaboration, the matching degree between tasks and collaborators directly affects the efficiency of software development. Rahman et al. (2015) proposed to model workers according to their individual factors (skills and wages) and group-based human factors (affinity in collaboration among workers) and allocate workers according to optimization goals. Pascanu et al. (2013) proposed an exploratory algorithm to understand workers' skills in advance and assign crowdsourcing tasks to workers according to their arrival time and the matching degree between skills and task attributes. Moayedikia (2018) et al. automatically classified workers according to the similarity of their behaviors, generated the corresponding automata, and used the corresponding automata to automatically allocate tasks to workers according to the classification results. Qiao et al. (2018) developed a new approach of collaborative crowdsourcing task allocation, under which workers who have completed tasks provide feedback to other workers, and models are built based on the feedback to assess workers' affinity and ability values, and the matching degree between specific tasks and workers is obtained based on the results of the model.

The performance of coordination in software delivery is based on the accuracy and timeliness of the information transfer process. The original coordination modes (oral and written) have ambiguous and distorted situations. Meanwhile, the data generated in the process of software development is characterized by large amount of data, various types, low value density, fast speed and high timeliness. The use of manpower to acquire data is inefficient and high-cost and is not conducive to the extraction and analysis of value information. Therefore, in order to ensure that information can be transmitted in real time, efficiently and truly in the process of coordination, information should be monitored and collected by relying on information technologies. By applying data intelligence, big data, artificial intelligence, deep learning and other technologies can be used to obtain data related to software project management comprehensively and efficiently, and case bases and knowledge bases can be established for learning of artificial intelligence. Data intelligence related technologies can not only collect data and information quickly and efficiently, but also use artificial intelligence to identify data authenticity. The identification of data authenticity is of great significance for coordination content analysis, result prediction and decision suggestion. Data intelligence can also build project communication case bases and knowledge bases. Artificial intelligence can promote the summary of massive information by learning knowledge bases, improve the algorithm on the basis of the summary, and then call information knowledge for analysis, prediction and suggestion, so as to make project coordination more effective. In addition, many scholars have studied the impact of data intelligence on project team coordination. For example, Chen Long et al. pointed out through research that a large amount of data will be generated in every link in the whole life cycle of software development; if such data can be recorded for analysis and displayed and fed back through visualization and other means, team management and project management can be further promoted, and development efficiency will be improved; by collecting and analyzing a large amount of data generated in every link in the software development process and extracting valuable information, the management of the agile software development process will be promoted (Chen Long, Ye Wei et al., 2016).

In terms of data processing, the engine based on the ETL (Extract-Transform-Load) method is a common way of data extraction and integration, widely used in project management. Using the ETL method to build the data conversion process and algorithm application, and defining data conversion rules by analyzing user and industry data, the accuracy of business data conversion will be improved. For example, in the process of project management, the project manager accurately obtains the work progress, quality and cost of the sub-project.

Based on the above discussion, it can be seen that software enterprises can use data intelligence to better and more timely reflect the information acquisition and processing in the process of software delivery, and accurately analyze the status and problems in the process of software delivery, so as to improve the team coordination in the process of software project management. So we propose the following hypothesis:

H1C: data intelligence has a positive impact on process coordination.

4.3.1.3 Data Intelligence has a positive impact on team structure in software delivery

Team structure of a software project is a structural system formed in terms of job scope, responsibilities and rights for division of labor and cooperation in the process of software project management in order to achieve team objectives. According to the definition of software team structure, we can see that evaluation indicators of team structure include: whether project objectives are clear, whether the division of team structure is reasonable, whether responsibilities and workflows of project personnel are clear, whether project personnel match with work contents, and whether the composition of project personnel is reasonable.

The relevant studies show that functional organizational structure, project-type organizational structure and matrix-type organizational structure are the most common project management modes (Luo Hanbin, 2008). In the functional organizational structure, each functional entity is responsible for completing the project work of its unit or specialty; the head of the functional department is responsible for project coordination; and the superior department can give instructions directly or indirectly to the subordinate department according to its own management needs.

The project-type organizational structure refers to the establishment of independent project teams according to the needs of project management. In the project-type organizational structure, project teams operate separately from other entities of the company and have their own technicians and managers, and traditional functional departments are eliminated with their professionals assigned to project teams. Meanwhile, the enterprise allocates certain resources to project teams directly led by the project leader.

In the matrix-type organizational structure mode, there are two different types of departments: vertical department and horizontal department. The intersection of pointing dashed lines of the vertical department and the horizontal department represents the work result. The work result is under guidance from two sources, namely, the vertical department and the horizontal department. If there is a conflict between instructions of the vertical department and the horizontal department, the project leader should coordinate or make a decision.

No matter which team structure is chosen for a software project, its purpose is to achieve organizational goals, so an enterprise should choose a project structure that can balance the project and meet overall organizational needs according to its own characteristics, organizational culture and project characteristics.

An enterprise needs to consider many factors in the selection of team structure. Project factors, such as the scale, the technology used, and the characteristics of project team members, are the dominant factors affecting the selection of structure. Moreover, the actual situation of the enterprise is also an important influencing factor. Therefore, factors that should be taken into account in choosing the team structure include: (1) project scale and structure; (2) project cycle; (3) enterprise management experience and level; (4) enterprise scale; (5) the software technology level of the enterprise; (6) the allocation of human and material resources of the enterprise. To sum up, the data in previous projects on the factors to be considered in the selection of team structure is required, so that data intelligence technologies can be used to analyze in the structure selection and select a reliable, reasonable and efficient structure.

With the rise of big data and new-generation information technologies, some new software delivery team modes also emerge, such as virtual team, platform team, and agile and lean team. These teams characterized by temporality and flexibility in dismissal improve the adaptability of enterprises in software development, delivery and partner collaboration. Among them, the agile and lean team mode integrates data analysis into the operation process and decision making of enterprises to agilely respond to requirements for user data analysis. It promotes enterprises to realize digital decision making in the operation process from product planning, production and manufacturing to precision marketing, and helps enterprises carry forward high-quality strategic development and refined operation management reform. The virtual team mode means that individuals cooperate in projects through information technologies. These individuals may be from different regions, and team members may be from the same organization, or from multiple organizations, and may never even meet in person.

Therefore, the application of data intelligence can help software enterprises better design and choose the team structure, and help managers formulate the most appropriate team structure according to the characteristics of software projects and the quality of employees.

In actual working scenarios, the construction of human resources of a team mainly depends on the internal selection. The structure of a software delivery team should be flexibly arranged after considering the specific situation. The requirements of human resources of a team depend on the matching between the abilities required by the specific work of the project and the individual abilities of team members (person-post matching). In this process, based on employees' personal conditions and specific requirements of posts, cyclic neural network, deep learning network and other technologies can be used to build a ability matching model to improve the post matching ability, clarify team members' composition and division of labor, and enhance the efficiency of team structure construction. Employees' individual abilities are quantified by means of fuzzy comprehensive evaluation and analytic hierarchy process for accurate screening and decision-making aid. In the delivery team construction and assignment decision making, the simulated annealing algorithm and multi-objective particle swarm optimization algorithm can be used to design the team model, so as to improve the ability of team construction; considering the consistency of the overall goal and the complexity of multiple sub-goals in a software delivery team, the conventional team work coordination and evaluation methods are not enough to deal with the multi-goal situation, so it is necessary to use the Myers-Briggs type indicators, and Kolber indicators to achieve the model transformation of team coordination goals. In order to verify the future work performance of construction of software delivery teams, it is necessary to build a team performance model with considering the influence of many factors, such as scenario and trust formation, and predict the accuracy of team construction schemes. The personal evaluation method for team members is to give the comprehensive score after considering individual work positions and actual performance, and realize the scientific evaluation of individual work through the application of artificial intelligence technologies and analysis algorithms according to individual work evaluation indicators (Zhang Xiang, 2013; Qin Chuan, 2021).

Under the technical requirements for building of software delivery teams, the capability requirement for enterprise extension is the capability requirement of data. The modeling technology is almost completely dependent on the realization of business data of enterprises, and the mathematical model must be built on the basis of data. In view of employees' personal data portraits, data on capability evaluation indicators for human resources, work data, and data on software projects, the big data environment is built based on the different requirements of technologies for the type and quantity of data, and the connection is realized through mathematical modeling, finally realizing the structure construction and optimization of the software delivery team system.

In addition, as an important means of digital transformation of enterprises, data intelligence also provides important capabilities for the design of team structure for software projects: firstly, data intelligence increases the capability of project management, resulting in the reduction of the role of the intermediate management level and the number of people, making the management level shrink, and promoting the team structure to be flat, the information communication between the upper and the lower to be fast and smooth, and the cooperation and coordination relationship to be strengthened; secondly, data intelligence provides a powerful platform for software delivery, enabling project teams to gradually realize the data sharing and integrated management of the whole team from procurement, production, sales, and services to accounting, finance, personnel, research and development, realize the combination of material flows and information flows, and promote the unified, continuous and integrated reform of the team's business process, and enabling software enterprise managers and project managers to make decisions by using shared data, and improve the accuracy and efficiency of decision making. Chen Long et al. pointed out that every link in the whole life cycle of software development will generate a large amount of data. If such data can be recorded for analysis and presented and fed back through visualization and other means, team management and project management can be further promoted and development efficiency can be improved. By collecting and analyzing a large amount of data generated in every link in the process of software development, valuable information is extracted to facilitate the management of agile software development process (Chen Long, Ye Wei et al., 2016). Yin Yilin studied the organizational form of multi-project management and proposed a special organizational structure -- PMO, under which a management office is established to optimize the organizational structure of the team, and effectively coordinate the internal operation and integrate and allocate resources of the team, so as to share experience and results, improve the project progress, reduce the project cost and enhance the core competitiveness of the enterprise (Yin Yilin, 2008). In the article on team structure optimization in the era of big data, Guo Xiaofang et al. pointed out that mining and analyzing the development of "big data" and "big data" related technologies in the human resource management system can form an information network with employees' information as the point, departments as the line and teams as the network, so as to better promote the coordinated development of personnel, affairs and teams (Guo Xiaofang, Liu Rong, Ma Xiaoyu, 2019).

Based on the above discussion, it can be seen that the application of data intelligence can better realize the rational allocation of team resources and create favorable conditions for team members to give full play to their subjective initiative and their own labor potential; it can help software project teams realize the comprehensive analysis on teams, posts, personnel, business and other aspects, and provide data support for optimizing the composition of personnel, affairs and teams in the process of software delivery and improving work efficiency. So we propose the following hypothesis:

H1TS: data intelligence has a positive impact on team structure.

4.3.1.4 Data intelligence has positive impact on process control in software delivery

Software project process refers to the process of providing a reasonable and feasible resource arrangement for the operation of software engineering and the management of software project activities. The project process provides a framework for the project leader to reasonably estimate the resources, funds and development schedule required for software project development, and control the implementation of the software project development process according to the process. Software project process is the process management of project scope process, project progress process, project quality process, project resource process, project communication process, project risk process, project procurement process, change control process and allocation management process. The most common realization stages of software process includes six stages: initial stage, formulation of software development process, examination and approval of software development process, implementation of software development process, measurement and evaluation of software development process, and modification of software development process.

At present, the specific process of software projects changes and develops on the basis of the classical software engineering theory. Software engineering is a subject that studies how to use systematic, standardized, quantitative and other engineering principles and methods to develop and maintain software. (Pfleege, S. L, 2003) On the basis of integrating classical software process models -- waterfall model, incremental model, iterative model and spiral model -- software capability maturity model and traditional project management theory, the process control of software projects of enterprises is connected with each link of the software project process, and divided into four dimensions, namely, time management, cost management, quality management and risk management throughout the whole process of software projects. These four dimensions are the main control directions of enterprises in the software project process. (Qi Zhichang, 2019, Liu Hai, 2020).

The preparation of software project process is the most complex activity. In order to make a realistic and practical process book, it is necessary to analyze the development cycle, project scale and cost of the software development project and control the implementation of the project. Due to the one-off characteristics of projects, project control is different from other management controls. In the whole process of process management, the project manager needs to evaluate the implementation results according to the latest information of the project and the amount of costs, manpower or other resources invested, and take the corresponding corrective measures by comparing, judging and negotiating with the benchmark process. The specific control management of projects mainly uses Gantt chart, critical path method, earned value analysis and other methods and tools.

The software development process and project management considered in software engineering are the scope of consideration in the software project process. Due to the complex, dynamic and systematic characteristics of software projects (Lai L. S, 1997), in analyzing each link in the software project process, it can be found that there are many specific uncertain factors in the control link of each process, such as the feedback system in software projects, a variety of nonlinear relations and time delays, and a large number of assumptions hardly defined and other factors (Ning Xiaoqian 2004; Yang Chen, 2018), which leads to a decrease in the controllability of the overall software process and has a negative impact on the delivery performance of software projects. In order to further eliminate all kinds of adverse factors in the software project process and improve the controllability of the software project process, the construction of the control ability and method of the software process containing the above uncertain factors is the key content of the control of the software project process.

Under the condition of considering the characteristics and processes of software projects, and in combination with the current experience of software projects of enterprises, through analyzing from the perspectives of time management, cost management, quality management and risk management of enterprises, the key links of project process control are mainly concentrated in five fields, namely, requirement control, cost estimation, project resource scheduling, software quality control and project risk management.

In terms of requirement control, due to the complexity of customers' requirements for software, process control of software projects focuses on the optimization of software requirements. Requirement optimization is to establish a relative selection order of selected elements by assigning different weights to different requirements to be optimized, that is, the complex qualitative problem is transformed into the mathematical problem based on ordering. Currently, the widely used requirement optimization technologies based on ordering mainly include Analytical Hierarchy Process (AHP), Quality Function Deployment (QFD) and Cumulative Voting (the 100-Dollar Test). The key of requirement selection focuses on four aspects: solving the clustering of large-scale requirements, solving the multi-employer conflict, dealing with the conflict between requirement dependence and requirement optimization, and solving the optimization of requirement change schemes (Garg N, 2017). (Dong Zhixiang, 2018) Through the analysis on the relevant research results at home and abroad, the current solutions to the four directions mainly rely on artificial intelligence technologies such as natural language processing technology, hypergraph technology, genetic algorithm, neural network and multi-objective analysis. It involves data de-redundancy, clustering and analysis, and requires the processing of a large amount of enterprise data (Srinivas N, 2014).

In terms of cost estimation, the main direction of control is cost control. Due to the characteristics that project cost estimation often produces large errors along with the change of real-time time, in combination with the uncertainty of multiple factors such as time, space, and resources of the software project process, the core control direction of enterprises is cost warning and correction strategy. At present, the main optimization methods focus on the realization of the mathematical model of cost deviation based on the control theory. Considering the cost attribute of cost control, the key points of calculation in different stages, and the complex relationship between data, the improved neural network algorithm, clustering algorithm, and group decision algorithm are used to realize the control process from cost estimation to decision change. Among them, although the clustering algorithms such as K-means have low dependence on sample data, neural networks and group decision algorithm models still need a large amount of data for learning to meet the actual requirements of cost control (Gao N, 2020; Sun Xiaokun, 2021).

In terms of project resource scheduling, restricted by the objective condition of limited resources enterprises can actually grasp, how to achieve the most efficient allocation of project resources through appropriate resource scheduling methods and make appropriate changes and controls along with the changes in the project process is the focus of project resource scheduling, which is a typical resource-constrained project scheduling problem (RCPSP). This problem is often solved by Memetic algorithm and other algorithms. In essence, it is a combination of genetic algorithm and local search strategy (simulated annealing algorithm, and tabu search algorithm, etc.), which finally obtains the calculation results by relying on input in operation of a large number of training data and constant iteration (Chen Di, 2016).

In terms of software quality control, due to the change of the software project development process, user use and new computer technologies, it is a constantly changing process in essence, and will always exist with the life cycle of software. Therefore, it is necessary to fully consider the control ability of the software project process and software itself. From the perspective of operability, the current mainstream control directions include software reliability evaluation, software defect detection and dynamic software update ability. The solution to the above problems is the integration of computer technologies and mathematical methods. Artificial intelligence technologies such as the construction of cyclic neural networks, transfer learning and machine learning methods are used to achieve the control ability in complex scenarios (Zhang Jie, 2019; Xu Xiaohui, 2011; Xu Zhou, 2019; Qiu Shaojian, 2019).

Project risks need to be controlled in every link of a software project. Under the condition of high project complexity and degree of change, due to the limitation of human cognitive ability and the change of the external environment, there are higher requirements for the level and ability of project risk assessment and management. In terms of risk identification, risk identification and evaluation models are constructed by the risk matrix, fuzzy comprehensive evaluation method and Broda ordinal value method; in terms of risk management, the control and optimization model for project risks is realized based on the earned value method and genetic algorithm. The above methods are implemented based on statistics and computer technologies, and a large number of enterprise data (Zhao Jinyuan, 2015; Shan Xiaohong, 2010).

Based on the above discussion, it can be seen that software enterprises can use data intelligence to monitor project status in real time, process and analyze the acquired data, and provide data support for evaluation and decision making of process control of software projects. Therefore, we propose the following hypothesis:

H1PC: data intelligence has a positive impact on process control.

4.3.1.5 Data Intelligence has a positive impact on process integration in software delivery

A complete business process is usually completed through coordination of a number of fine-grained business processes, which come from the existing independent systems within an enterprise. A business process integration scheme is mainly to connect these business processes across heterogeneous systems, making the overall business process become a closed loop from the perspective of the enterprise. Among them, information transfer is particularly important. Process integration involves the overall coordination of business processes and activities in a software development environment. It involves the effective task division of each post or unit, and the subsequent coordination of the execution of tasks within the enterprise and between the enterprise and customers (Van de Ven A H, 1976). When the uncertainty of a software project is high, a lot of mutual adjustments between different processes are required (Fairbank J F, et al, 2006).

In CMMI, there are the corresponding requirements for process integration and quantitative management of software enterprises. CMMI is divided into five levels: level 3 is definition level, where all processes in the software delivery process are identified and defined; software processes for management and engineering are documented and standardized, and standard software processes for the entire software organization are formed, with a focus on process. Level 4 is quantitative management level, where software process and product quality are understood and controlled quantitatively, with a focus on performance management of the delivery process, and quantitative project management is carried out. Level 5 is continuous integration level, which focuses on organizational renewal and deployment, reason analysis and decision making based on the quantification at level 4. The CMMI system does not advocate cross-level evolution, because from the second level on, the implementation of each lower level is the foundation for the implementation of the higher level, which also shows that quantitatively expressing the performance of the software delivery process in the form of data is the foundation for the integration of the software delivery process.

Wang Xiwen pointed out that the specific application of big data can realize the integration of the data-driven production mode, make the process within the factory and between cooperative factories more agile and efficient, and promote and realize the innovation of products, processes, and operation and business modes; big data can promote the innovation-driven development of the manufacturing industry through three aspects: production technology improvement, production process integration and energy consumption analysis (Wang Xiwen, 2016).

The use of big data and artificial intelligence technologies constitutes the cornerstone of information processing capability (Fairbank J F, 2006). Powerful technological means can facilitate the recording and retrieval of information about the software delivery process, and make it more feasible to formalize the process (Mithas S, Whitaker J, 2007). Formalized processes help software organizations improve the process efficiency, so that each process can operate in a unified, fast and stable manner. Big data and artificial intelligence can integrate processes well, so as to make better use of resources and reduce costs. In general, effective big data and artificial intelligence technologies can reduce the increase in project costs, slow progress and low quality of software caused by the increase of uncertainty. For example, big data technologies can be used to digitally input and archive documents generated in software projects and formulate unified digital labels for centralized classification, making it convenient to change and refer to when there is a change in requirements or reference in the later period; meanwhile, they can store and back up data in the cloud, reducing the complexity of paper versions, forming the digitization and integration of data, and solving the problems about storage time and space of data.

In addition, different data intelligence methods, artificial intelligence, optimization and online processes can be applied to every process of software delivery (Kulkarni, R. H, 2017). The process of artificial intelligence is a fully automated process that works like an expert's brain, so it can make the process planning for software delivery more accurate. Besides, it is important that the optimization process generates effective plans by searching for candidate solutions available in the surrounding environment, and the online processing of plan updates allows for more efficient and timely planning of software processes. Therefore, intelligent estimation, optimized scheduling, and online tracking are effectively integrated to achieve improved performance in software development, and product quality can be guaranteed through the estimation of results by intelligent agents.

With the continuous development of big data and other information technologies, as big data has the characteristics of multi-source heterogeneity, large quantity, variety, fast speed and authenticity (Tao et al, 2018), the accumulation of data prompts many companies to use big data analysis methods to convert data into useful information, with a view to improving decision making and supporting performance in software delivery (Hazen B T, 2014; Papadopoulos T, 2017). Therefore, through the application of data intelligence, software enterprises can comprehensively and quickly obtain all the data generated in the process of software delivery, provide data support for enterprise operations, reduce the variability of process performance of projects, and make projects enter the acceptable quantitative boundary, so as to achieve the overall control of products and project processes and the quantitative prediction of organizations' software process capability. At the same time, data intelligence provides analytical data for continuous process improvement of organizations and projects through real-time monitoring and intelligent analysis of data in the whole process, enabling managers to conduct process integration with new technologies and new methods. Sun Xinbo et al. pointed out through research that the rational application of big data is conducive to improving the supply chain agility of enterprises. The supply chain agility is divided into customer agility, process agility and partner agility (Sun Xinbo, Qian Yu et al., 2019).

The integration of the software delivery process aims to simplify complicated links, rearrange unreasonable links and make up for missing links according to the characteristics of delivered software products, and finally achieve the efficient process. Based on the above discussion, it can be seen that software enterprises can use data intelligence to better manage and control the process of organizations and projects, and continue to carry out process integration. Therefore, we propose the following hypothesis:

H1PI: data intelligence has a positive impact on process integration.

4.3.1.6 Data Intelligence has a positive impact on modularity in software delivery

Modularity refers to an experimental and simple application software with basic functions developed by system analysis and design personnel in cooperation with users on the basis of defining users' basic requirements in a short period of time. The modularity method is a system development method with the new design idea, process and method gradually formed on the basis of the database system, the fourth-generation program generation tool and system development and generation environments. With the modular system development method, after obtaining a set of basic requirement definitions, the visual development environment of advanced software tools can be used to quickly establish an initial version of the target system, and after the initial version is given to users for trial, supplement and modification, and develop a new version. This process is implemented repeatedly until the "exact solution" of the system is obtained, that is, until the user is satisfied. It can effectively solve the problem that it is difficult to accurately define software requirements of software projects, and improve the delivery performance of software enterprises.

The core of the modularity method is to give versions to customers, allow customers to comment on software products through presentation, and improve the communication between users and developers, so the most important thing is to let customers understand modular products. Therefore, human-computer interaction technologies can be used to improve the software design or development process. Interaction technologies are user-oriented, and support meaningful communication through circulation and collaboration between people and technologies. Successful interaction design has simple and clear goals, strong purposes, and intuitive interfaces. At the same time, automatic suggestion tools can be integrated into modular products to further improve the product development process and increase the delivery capability.

The user-oriented modularity method can identify user requirements more effectively, which can not only shorten the time of requirement analysis, but also reduce the misunderstanding of user requirements by developers, thus significantly reducing the time of requirement change, improving the quality of requirement analysis and reducing the risk of system development. In addition, some scholars have made an empirical study based on the hierarchical model established by the modularity method and in combination with the construction of actual software projects. The results show that this model can not only better solve the division of labor and cooperation among developers, customers and information management departments in software projects, but also better solve the requirement change management, so as to ensure the progress and quality of projects (Xiao Jin, 2009).

Therefore, software enterprises can use data intelligence to comprehensively and efficiently collect data related to user requirements for processing and analysis, and improve the matching degree between functions of the modular system and business requirements of users, and shorten the cycle of modularity correction and development; data intelligence can also use artificial intelligence and other technologies to carry out deep learning on software projects, dig out functional requirements and designs of similar projects, and provide users with more appropriate systems. Based on the above discussion, software enterprises can use data intelligence to better carry out requirement analysis and rapid development of modular systems. Therefore, we propose the following hypothesis:

H1M: data intelligence has a positive impact on modularity.

4.3.2 Data intelligence has a positive impact on software delivery performance

Driven by big data and application scenarios, data intelligence is an emerging subject that integrates data acquisition, processing, analysis and visualization technologies from multiple subjects to provide actionable criteria for complex management decision making practice in the real world. As the starting point of data intelligence, big data improves the training effect of models and directly drives the development of data intelligence. So big data analysis capability is the key ability of data intelligence. Big data analysis capability (BDAC) is broadly defined as the ability to provide business insights by use of data management, infrastructure (technologies) and talents' (personnel's) capabilities to transform business into competitiveness (Kiron et al., 2014). A large number of literature points out that BDAC has a positive impact on enterprise performance. (Wixom et al., 2013; Akter et al., 2016; Wamba et al., 2017; Gupta & George, 2016) and others demonstrated through the large sample survey that BDAC directly and significantly affects market performance and operational performance of enterprises. Xu Guohu (2017) carried out the research with 66 listed enterprises that have implemented the big data system as samples, showing that big data capability can improve operational performance and profit performance of enterprises.

Concerning specific enterprise performance indicators, the existing literature indicates from the perspectives of price optimization and profit maximization (Davenport and Harris, 2007; Schroeck et al., 2012), sales, profit rate and market share (Manyika et al., 2011), and return on assets (ROA) (Barton and Court, 2012; Columbus, 2014; McAfee and Brynjolfsson, 2012; Ramaswamy, 2013; Srinivasan and Arunasalam, 2013) that big data capability can help enterprises reduce costs or improve the quality of products and services, thus improving enterprise performance. Chen et al. (2015) studied the application of big data capability in the supply chain and pointed out that big data capability has a positive impact on productivity and business growth of enterprises.

Software delivery performance is an important part of performance of software enterprises, so we propose the following hypothesis:

H2: data intelligence has a positive impact on software delivery performance.

4.3.3 Relationship between intermediate variables and software delivery performance

The improvement of software delivery performance requires the efficient operation of software delivery organizations, and the division of labor and effective communication and coordination of delivery teams; adoption of engineering methods such as modularity and process integration according to the characteristics of delivered software; timely and accurate process control according to the status of project implementation; accurate evaluation of the delivery process. Therefore, this dissertation analyzes the actual relationship between intermediate variables, such as task level evaluation, coordination, team structure, process control, process integration and modularity, and software delivery performance.

4.3.3.1 Relationship between task level evaluation and software delivery performance

Task level evaluation is to provide decision makers with multi-aspect warnings for the selection of task levels and implementation schemes, and strive to objectively and accurately collect and present to decision makers the data and facts related to task level implementation such as resources, technology, market, finance, economy, and society in a true and complete manner, enabling them to be in a more favorable position and make the right and appropriate decisions based on facts, and laying the foundation for the implementation and comprehensive inspection of the investment task level.

The significance of task level evaluation is to objectively evaluate all the work at the task level, so as to form performance conclusions on the results brought by the task level and its members.

Task level evaluation of software is mainly to evaluate the requirements, organization, plan, cost, risk and quality involved in the task level through scientific and reasonable means and methods. The quality of task level evaluation directly affects the progress, cost control, risk prevention and quality of software projects, and at the same time, the evaluation results are applied to current and future software projects.

Definitive evaluation, often called bottom-up estimate, is a technology of breaking down work elements into smaller components. Estimates are prepared to meet the requirements of each lower and more detailed work, and summarized into the total number of work components (P.M. Institute, 2004). The accuracy of bottom-up estimate depends on the scale and complexity of the work identified at lower levels (P.M. Institute, 2004). The nature of this technology is divide and rule. Large deliverables are broken down into smaller sub-deliverables that are further divided into the work tasks required to develop such deliverables (P.M. Institute, 2004; Vicinanza, 1991). By summarizing such smaller and more accurate estimates, the predictability of project costs and the duration of deliverables can be increased.

Scholar Yang Li (2010) proposed an extension evaluation method for the possible risks of software projects, including a comprehensive evaluation method from the perspective of factors affecting software delivery, and an evaluation method for each event according to the probability of occurrence of each factor or risk and the possible consequences and losses, and verify the performance of the above methods through examples (Yang Li, 2010). In the process of software project development, each type of project uses different technologies, personnel, hardware, communication and coordination issues, and trust among personnel, etc. Khan, A. W. Et al. analyzed the weights of different factors in different projects through the AHP to evaluate the status of projects, providing a framework for dealing with problems about software reliability in software environment and playing a crucial role in determining the organizational success of software projects (Khan, A. W., 2021). Based on the above discussion, if enterprises can effectively conduct the task level evaluation, they will improve the project management from two main aspects, namely, cost and progress, and enhance the delivery performance of projects. So we propose the following hypothesis:

H3TLE: task level evaluation is positively correlated with software delivery performance.

4.3.3.2 Relationship between coordination and software delivery performance

A software project involves personnel, resources, cost, time, and product quality, etc., so coordination in software development is not limited to the internal team, but involves five internal and external aspects of the entire organization, namely, individual, team, project, company and business environment.

Software delivery management involves cost, time, scope and other targeted management, as well as quality, risk, procurement, coordination, human resources and integrated management. However, as software is a logical entity, which is intangible, invisible and difficult to monitor in essence, software quality problems are basically caused by human errors, and coordination is also an important factor to prevent human errors. In addition, a large part of the work in project management, such as time, cost, quality, manpower, risk and procurement, is coordination with people. Therefore, coordination management is highly valued in software development.

Coordination in software development is not limited to the internal of a team, but involves the internal and external of the entire organization. Curtis, through field research, presented a hierarchical structure of coordination of software organizations, including five levels: individual, team, project, company and business environment (Curtis, 1988).

Organizational structures, such as horizontal coordination devices, are placed in organizations to improve performance (Blake and Mouton, 1985). They form formal structures that improve organizational efficiency and also provide a small amount of communication and control (Nystrom, 1978).

In addition, Yang Kun et al. believed that communication and coordination among personnel and organizational factors in software development are important issues affecting the production efficiency and availability of software (Yang Kun et al., 2004). Curtis studied 17 large software development projects with the interview method, and proposed that interruption of communication and coordination is one of the main reasons for failure of software projects (Curtis, 1988).

Based on the above discussion, coordination plays an important role in software project management, and software organizations can enhance the software delivery performance by improving coordination in teams. So we propose the following hypothesis:

H3C: Coordination is positively correlated with software delivery performance.

4.3.3.3 Relationship between team structure and software delivery performance

Firstly, organizational structure is not only the way of existence of the material system, but also the basic attribute of the material system. It is the basis and premise for the system with integrity, hierarchy and functionality. Without a proper framework, there will be no clear form (Ci Hai, 1999). According to Sun Biao, the organizational structure includes three key factors: firstly, the organizational structure determines the formal reporting relationship of an enterprise, including the management range and hierarchical relationship of managers; secondly, how to combine individual employees into independent departments, and then combine departments into the whole enterprise is also determined by the organizational structure; thirdly, the organizational structure contains a set of process systems, guaranteeing the effective cooperation and communication among departments (Child, 1984; Sun Biao, 2011).

The core of a software development project is team members. A complete project cannot be completed without the participation of team members. Similarly, for a software project team, it is impossible to complete a project in the absence of an important role in the development process. The completion of a software project requires the deployment of the team's work, the integration of business technologies, the improvement of the team's response speed, and the reasonable allocation of resources. Therefore, a complete and efficient team structure has a significant impact on software delivery.

Traditional team structures for project management can be divided into three types: functional, project-type and matrix-type. The comparative analysis is shown in the figure below (see Table 7). Among them, the functional team structure is applied in a smaller range, mainly technology projects; the project-type team structure has advantages in large-scale construction projects, especially those with uncertain environment and complex technology; the matrix-type team structure can realize rational utilization of resources when managing large-scale projects.

Table 7 Pros and cons of three team structures for project management

Name	Pros	Cons
Project-type	（1）Control resources; （2）Responsible to customers.	（1）Lack of knowledge and information exchange;

Name	Pros	Cons
		（2）Lack of resource allocation.
Functional	（1） No duplication of effort; （2） Excellent functional activities.	（1）Not comprehensive enough; （2） Too slow to respond; （3） Lack of customer knowledge.

Matrix-type	（1） Rational use of resources;
	（2）Realize the application of all projects;
	（3）Good communication effect.
		（1） Multiple leadership;
		（2） Imbalance of rights;
		（3） Lack of coordination between projects.

Wang, Z proposed that in a Free/Libre and Open Source Software (FLOSS) project team, different roles make different types of contributions to the project, and proved that maintaining a reasonable team structure is of great significance to the potential growth and expansion of the project (Wang, Z, 2015). Therefore, we propose the following hypothesis:

H3TS: team structure is positively correlated with software delivery performance.

4.3.3.4 Relationship between process control and software delivery performance

The process of software from development to delivery is an orderly work under the requirements of the institutional framework. Establishing an accurate and reasonable project management process enables software project managers to make a plan for what technology is needed to realize software functions before the development. By strengthening process control, software project managers can effectively utilize resources of software enterprises, reduce project risks, improve the efficiency and quality of project development, start and deliver projects on time, and make software better meet the requirements of customers and stakeholders at the time of delivery.

According to the control theory, the tighter control over the processes used by an organization will increase productivity by reducing errors and rework (Carver and Scheier, 1981; Lord and Hanges, 1987). The feasible software delivery process is one of the key issues affecting the success or failure of a project. The delivery process control is the key to improve the management level of the delivery process. In software delivery management, it is necessary to effectively conduct process control management, analyze and identify factors, resource constraints and allocation that affect the project, effectively conduct the scientific and reasonable process preparation of progress, cost, quality and risk, constantly strengthen the process control level, optimize control measures, intensify the monitoring of the delivery process, and implement effective control over the delivery process, thus improving the delivery performance and market competitiveness of enterprises. Cao, P. proposed a new method for optimizing the process of software projects based on the rate of return of projects, established a control model for software projects and proposed an optimization algorithm. This method can change passive control into active control before the implementation of each process, and provide an effective tool for managers to implement process optimization, greatly promoting the possibility of success of software projects (Cao, P., 2009). Based on the study on the impact of the value management change theory on project changes, Wang, Q et al. improved the traditional process model for software development projects from two aspects: scope definition stage and requirement management of software development projects, after considering the actual requirements of customers. Key sub-modules such as requirement identification, pre-design, testing and requirement redefinition are added in the scope definition stage of projects to accurately define and understand the real requirements of customers, which helps to improve the process of software development projects and achieve project objectives (Wang, Q, 2010).

Based on the above discussion, strengthening the process control is beneficial for software enterprises to control the progress, cost, quality and risk in the process of software delivery, and improve the delivery performance of software projects by improving the progress of software projects, reducing the cost in the process of projects, and enhancing the quality management level of projects. Therefore, we propose the following hypothesis:

H3PC: process control is positively correlated with software delivery performance.

4.3.3.5 Relationship between process integration and software delivery performance

Process integration refers to a measure to maintain an enterprise's competitive advantage through the continuous development, improvement, and integration of business processes, including the process of sorting out, improving, and integrating existing workflows. It essentially optimizes and redesigns the business process in order to achieve breakthroughs in software quality, project cost, development speed, product quality and service.

For process integration, the appropriate process integration method should be selected according to the characteristics of the specific process. The proper process integration method can not only facilitate the process integration, but also bring the role of the process to the best play. In process integration, a single method can be selected first or multiple methods can be combined. The literature on process integration from the field of management mainly focuses on the integration of technology, information and business processes. Through the analysis, it is found that the process integration methods adopted can be generally divided into two categories: one is to use specific technologies or frameworks for process integration, such as SOA (Service-Oriented Architecture), EAI (Enterprise Application Integration) technology, and Petri Net; the other is to study the relationship between processes from knowledge, information and other aspects to achieve the integration of processes, such as ontology integration, and multi-level knowledge integration.

Process integration methods and tools are the basic method and strategy to study management process and process integration. Constantly developing, perfecting and integrating business process methods is of great significance to maintain the performance of business process implementation and maximize the results of process integration. At present, process integration methods include benchmarking method, DMAIC model, ESIA analysis method, ECRS analysis method, and SDCA cycle. The Capability Maturity Model (CMM) for software delivery of software enterprises is a standard that describes the development stages of a software organization in practices of defining, implementing, measuring, controlling, and improving its software process. The core of the CMM is to treat software development as a process, and under this principle, conduct process monitoring and research on software development and maintenance, so as to make software development more scientific and standardized and enable enterprises to better achieve business objectives. When the CMM reaches level 5, software enterprises can focus on improving processes, adopt new technologies and methods, and have the means to prevent defects, identify weak links and improve them, and can obtain and analyze statistical data on process performance to obtain the best method.

Through process integration, enterprises can improve the operational efficiency of organizations and reduce the overall operating cost, and can break the barriers between departments, and enhance horizontal collaboration; the management can effectively supervise and control the overall operation of the enterprise; and can ensure the effective implementation of corporate strategies, so as to support the realization of strategies. In general, the main risks developers face are failure to meet the delivery time or project requirements, which may be due to a lack of follow-up with key activities in the development process and uneven allocation of workload of the project. Therefore, LP Lopez-Arredondo et al. used the relevant theories of business process integration to redesign the software development process of a technical service company in Mexico and reallocate resources in a balanced way. The results showed significant improvements in both time and cost of the redesigned process (LP Lopez-Arredondo, 2019). In the research, Shang Huihua proposed a software quality management model based on process integration and improvement. The model takes the whole process of software development as the object, emphasizes the planning, control and improvement of the software development process, and meeting customer requirements through process inheritance, and provides specific control methods (Shang Huihua, 2011). When studying the level of information-based project management, Wu Gao proposed that designing the organizational structure and responsibilities of multi-project management, integrating the project management process and improving the project management system can reduce the labor cost of the project, shorten the project implementation cycle, and improve the utilization rate of effective working hours and the profit rate of the project (Wu Gao, 2018). Senapathi, M explored the implementation of DevOps in a product development organization in New Zealand and found that the support of some technological drivers, such as the implementation of automated pathways and cross-functional organizational structures, of DevOps (referring to a set of principles and practices for improving collaboration between development and IT operation) i critical to expected benefits. The benefits of adopting DevOps include increase in frequency of high-quality deployments and increase in collaboration between operational teams in development (Senapathi, M, 2018). Based on the above discussion, process integration of software enterprises can improve the operational efficiency of organizations, cross-departmental cooperation, and the supervision and control ability of managers, and enhance the efficiency of software delivery. Therefore, we propose the following hypothesis:

H3PI: process control is positively correlated with software delivery performance.

4.3.3.6 Relationship between modularity and software delivery performance

The principle and process of modularity is very simple. At its core is the replacement of formal and strict specifications with interactive and quickly established modules. Customers can provide real and concrete feedback for developers through the modularity of hands-on operation and test on computers.

Modularity: by showing customers the modularity of actual operations, customers can "see and touch" the system, and give very clear opinions on the system, improving the communication between customers and developers. Using modularity, customers play a leading role in the process of system development, and can help developers understand the real requirements of customers, and more effectively identify user requirements, which can not only shorten the time of requirement analysis, but also reduce developers' misunderstanding of user requirements, so as to significantly reduce the time of requirement change, improve the quality of requirement analysis and reduce the risk of system development. Based on the above, software enterprises can use modularity to improve the development speed of software projects or products, and thus enhance the software delivery performance.

Sanchez (1999) pointed out that by adopting modular product architectures and processes, enterprises can better manage organizational knowledge in the process. Modular products, processes and knowledge architectures enable companies to create more product categories, introduce technologically advanced products faster and bring them to market, reduce the cost of product creation and production, and create new market dynamics and vitality. Ellram et al. (2008) studied the offshore outsourcing of the service industry from the perspective of risk control, and proposed that under the premise of controllable risks, enterprises can better improve their innovation ability when focusing on the development of key core modules. The application of the modularity method can effectively dig out the potential value of internal and external resources of enterprises, so as to improve service innovation performance (Tao Yan, 2011).

Based on the above discussion, software enterprises can use technology modularity to improve the development progress and competitive advantage of software projects or products, thus enhancing the organizational performance.

So we propose the following hypothesis:

H3M: modularity is positively correlated with software delivery performance.

Chapter 5 Methodology

5.1 Methods

In this dissertation, variables are measured and analyzed by using the methods such as literature analysis, sampling questionnaire survey, mediating effect test, Akaike information criterion (AIC), bootstrap test, and multiple linear regression, and the scale is established on the basis of previous literature to measure and analyze the relevant data for research.

5.1.1 Sampling questionnaire survey

Based on the relevant hypotheses and conceptual models obtained through the theoretical research and in combination with the actual situation of the site, the measurement scale meeting the requirements of variables is designed; the questionnaire is designed and sampling survey is conducted to obtain data. The pre-test test questionnaire lays the foundation for large-scale sampling.

5.1.2 Statistical analysis

Descriptive statistical analysis, factor analysis, correlation analysis and multiple regression analysis are carried out for the effective data obtained to test the reliability and validity of the empirical data. The structural equation model integrates different statistical analysis technologies such as factor analysis and path analysis, and can analyze the structural relationship between potential variables. Therefore, this study uses the structural equation modeling method to verify the relationship between the model and variables.

5.1.3 Mediating effect test

In statistics, if a variable M is found when studying the effect of X on Y, and X can influence Y through M, then this variable M is called a mediating variable. The existence of a mediating variable indicates the existence of mediating effect.

5.1.4 AIC (Akaike information criterion)

Akaike information criterion (AIC) is a criterion to measure the goodness of fit of statistical models. The method is to find the model that can best interpret data but contains the least free parameters. Increasing the number of free parameters improves the goodness of fit. AIC encourages the goodness of fit of data but avoids overfitting as much as possible, so the model with the lowest AIC value should be given priority. Assuming that the choice is made among n models, the AIC value of n models can be calculated at one time, and the model corresponding to the minimum AIC value can be found as the selected object.

5.1.5 Bootstrap method in the mediating effect test

The Bootstrap method regards samples as the whole. Under the method, assuming that there is a sample with a sample size of N, the sampling with replacement (take out a case, put it back, and then take out the next case) is carried out on the sample until the number of extracted cases is equal to N, and these N cases are a sample. If the above process is repeated for K times, K samples are obtained. Each sample can be used to calculate an estimate of the mediating effect. Thus, the sampling distribution composed of the product of K coefficients can be obtained, and the confidence interval of the product of coefficients can be obtained, to test whether the ab effect exists and its proportion in the total effect, reflecting the extent of the mediating effect.

5.2 Logic and process

The research core of this dissertation is the mechanism and path of data intelligence influencing software delivery performance.

From an interdisciplinary perspective, and from the perspective of management (technology management, and strategy management), this dissertation adopts a variety of research methods, such as the combination of qualitative and quantitative research, empirical research and theoretical derivation, literature research and statistics (see the following figure), and uses deduction and induction to explore the decisive factors in the software development process of software enterprises under the environment of data intelligence, the role of data intelligence in the software industry and its impact on software delivery performance.

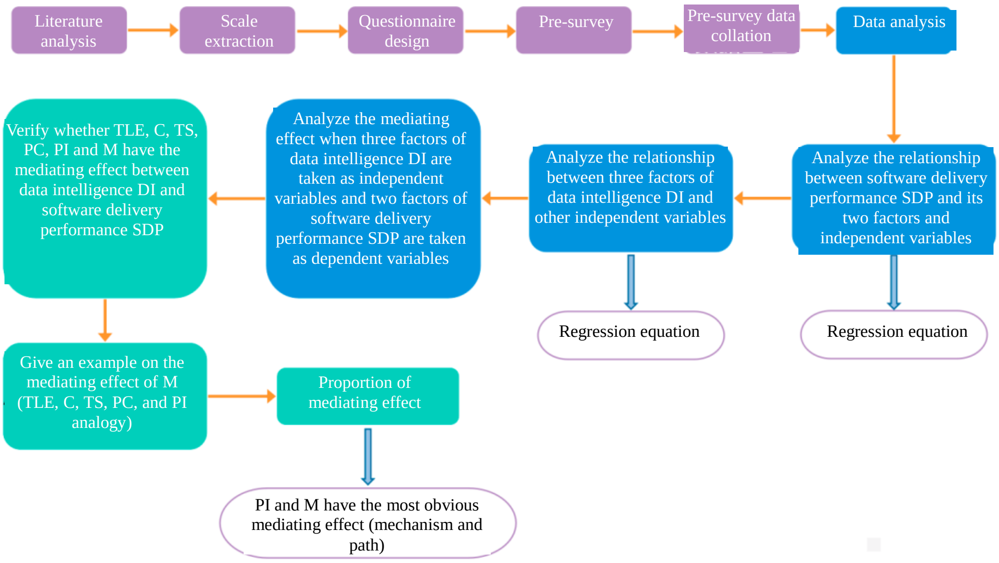

Literature analysis
Scale extraction
Questionnaire design
Pre-survey
Pre-survey data collation
Data analysis
Analyze the relationship between software delivery performance SDP and its two factors and independent variables
Regression equation
Analyze the relationship between three factors of data intelligence DI and other independent variables
Regression equation
Analyze the mediating effect when three factors of data intelligence DI are taken as independent variables and two factors of software delivery performance SDP are taken as dependent variables
Verify whether TLE, C, TS, PC, PI and M have the mediating effect between data intelligence DI and software delivery performance SDP
Give an example on the mediating effect of M (TLE, C, TS, PC, and PI analogy)
Proportion of mediating effect
PI and M have the most obvious mediating effect (mechanism and path)

Figure 5 Research method flow chart

5.3 Data collection method

5.3.1 Questionnaire design

Based on the hypothesis and conceptual model described in Chapter IV, this chapter uses empirical data to test the correctness of the theoretical hypothesis. The detailed arrangement is as follows: firstly, the questionnaire design process is explained. Based on the more mature measurement scale from predecessors and in combination with the concept of this study, the measurement scale of the questionnaire is formed to test the performance and reliability of the initial questionnaire.

Since the data required by this study cannot be obtained from materials disclosed by enterprises, the data collection method of this study is questionnaire survey. The questionnaire uses the Chinese scale. Respondents rate all the measurement questions using a 7-level Likert scale. The numerical score of the scale from 1 to 7 indicates "strongly disagree" to "strongly agree", and "4" indicates "general". The specific questionnaire design process in this study is as follows:

Firstly, collect a large amount of literature, conduct research on companies' data intelligence capability and software development status, and carry out the detailed, scientific and reasonable theoretical conception of variables, to ensure that the operational definition of each variable is clear and clear.

Secondly, collect the literature on data intelligence, software development, software delivery and organizational performance, search for measurement scales related to the measured object, and screen them according to the theme of this study and the reliability and validity of existing scales, to lay a foundation for measurement of variables.

Thirdly, categorize and summarize the screened scales, and select the validated more mature and effective questions. All the scales used in this study are from papers in internationally recognized core journals. For the original English scales used in the initial questionnaire, in order to avoid the distortion of the content meaning of such scales caused by the translation process, based on the views of Mr. Wang Zuoliang (1989) and Yun Seok Choi et al. (2010), this study adopts the cultural equivalent translation method to carry out equivalent translation from three aspects: content equivalence, semantic equivalence and research implementation equivalence, to ensure the reliability and validity of such scales in the cross-cultural context through such translation procedures.

Fourthly, without providing theoretical background introduction, send the questionnaire to software developers and consultants of enterprises, and recover questionnaire results.

5.3.2 Measurement of variables

The hypothesis of this study involves eight variables, namely, data intelligence, task level evaluation, coordination, team structure, process control, process integration, modularity, and software delivery performance.

The independent variable in this study is data intelligence. Data intelligence refers to extracting, exploring and acquiring revealing and operable information from data through machine learning, deep learning and other technologies, to provide effective intelligent support for people to make decisions or perform tasks based on data. It contains three core elements: data, technology, and prediction and decision-making capability. The measurement of data intelligence is complicated and difficult to quantify. At present, there is no special high-quality scale for measuring data intelligence at home and abroad. Therefore, the author compiles the scales by drawing on the relevant scales of business intelligence and big data technologies from Gupta and George (2016) and Wang et al. (2019), and in combination with the actual situation of the software industry. These scales are widely used by scholars at home and abroad and have higher reliability and validity. In this study, the scales are modified by the expression way of Chinese, and the following questions (as shown in the table) are used to measure data intelligence. The scale for data intelligence consists of 11 items in three dimensions: data, technology and prediction and decision-making capability. The questions in the scale are all forward questions.

Table 8 Scale for data intelligence

	Construct	Ordinal items	Resources
Data intelligence	Data	We have access to very large, unstructured, or fast-moving data for analysis.	Gupta. M., & George, J. F. (2016).
		We integrate data from multiple internal sources into a data warehouse or mart for easy access.	
		We integrate external data with internal to facilitate high-value analysis of our business environment.	
	Technology	We have explored or adopted parallel computing approaches (e.g., Hadoop) to data intelligence processing	
		We have explored or adopted different data visualization tools.	
		We have explored or adopted cloud-based services for processing data and performing analytics.	
	Predictive and decision-support capability	Discover patterns among specific variables of interest across departments.	Wang, Y., Kung, L., Gupta, S.,& Ozdemir, S.(2019).
		Analyse data from different sources and use the results to predict future trends.	
		Provide actionable insights from data in a format readily understood by customer.	

Mediating variables in this study include task level evaluation, coordination, process control, process integration and modularity. Most scholars at home and abroad measure these concepts in software project management in the form of questionnaire survey, forming some high-quality scales. By referring to scales from Henry et al. (2007), Akter et al. (2016), Nidumolu (1996), and Shamim et al. (2019), the author compiles the corresponding scales for task level evaluation, coordination, process control, process integration and modularity. These scales are widely used by scholars at home and abroad and have higher reliability and validity. In this study, the scales are appropriately modified according to the expression way of Chinese and in combination with the actual situation of the software industry, and the following question items (as shown in the table) are adopted for measurement. The questions in the scale are all forward questions.

Table 9 Scales for task level evaluation, coordination, process control, process integration and modularity

Task level estimates	when formulating project plans, I explicitly formulate cost estimates for individual project tasks.	Henry, R. M., McCray, G.E., Purvis, R.L.,& Roberts, T.L. (2007)
	when formulating project plans, I explicitly formulate time estimates for individual project tasks.	
Coordination	In our organization, business analysts and line people meet frequently to discuss important issues both formally and informally.	Akter, S., Wamba, S.F., Gunasekaran, A., Dubey, R., & Childe, S.J. (2016)
	In our organization, business analysts and line people from various departments frequently attend cross-functional meetings.	
	In our organization, business analysts and line people coordinate their efforts harmoniously.	
	In our organization, information is widely shared between business analysts and line people so that those who make decisions or perform jobs have access to all available know-how.	
Process control	Control over project costs.	Nidumolu (1996)
	Control over project schedule.	

	Adherence to auditability and control standards.	
	Overall control exercised over the project (overall item).	
Process integration	We have the ability to integrate the processes involved in the big data chain (i.e. data collection, preparation, analysis and decision making)	Shamim, S., Zeng, J., Shariq, S. M., & Khan, Z. (2019).
	The integration of the processes involved in the big data chain reduces the cost of big data use	
	The integration of the processes involved in the big data chain reduces the efforts necessary to analyse big data	

Mediating variables in this study also include team structure. The author believes that the team structure in software project management is different from that in traditional enterprise management, so the scale related to team structure in existing enterprise management is not used. However, domestic and foreign scholars have not made specific research on team structure in software project management, so, based on his/her own software practice, the author develops a new scale consisting of 4 items for measurement. The questions in the scale are all forward questions.

Table 10 Scale for team structure

Team structure	Our company has a reasonable organizational structure
	Our company's project personnel have clear responsibilities and division of labor
	Our company's project personnel match with job duties
	Our company has a reasonable composition of project personnel

The dependent variable in this study is software delivery performance. Software delivery performance mainly includes: whether the product can bring value to customers and the company; whether the team can quickly produce and release products, quickly respond to changes in market demand and business goals; whether it is sustainable with the exponential growth of business scale and team scale. It contains three core points: effect, efficiency and sustainability. The measurement of software delivery performance is complex. At present, there is no special high-quality scale for measuring software delivery performance at home and abroad. Therefore, here the author mainly draws on the relevant scales of enterprise performance, and prepares scales in combination with the actual situation of the software industry, and mainly by reference to Kevin et al.（2000）, Singh et al.（2019）, and Tompson et al.（2013）. These scales are widely used by scholars at home and abroad, and are highly reliable and valid. This research is modified according to the Chinese expression, and uses the following questions (as shown in the table) to measure software delivery performance. The data intelligence scale consists of 12 items, including two dimensions: short-term performance and long-term performance. Long-term performance include perceived indirect benefits and strategic performance. Short-term performance and long-term performance correspond to effect, efficiency and sustainability. The questions in the scale are all forward questions.

Table 11 Scale for software delivery performance

Software delivery performance	Short-term Performance		Improving firm's market situation.	Kevin et al. (2000)
			Increment in firm’s sales volume.	
			Increment in the firm's profit rate.	
	Long-term Performance	Perceived indirect benefits	Improve organization image.	Singh et al.(2019)
			Improve competitive advantage.	
			Benefit other business practices.	
			Improve customer services.	
			Improve relationship with business partners.	
		Strategic performance	Our IT infrastructure is easily adaptable to changes in business processes.	
			Our IT infrastructure is easily adaptable to leverage technology changes.	Tompson et al. (2013)
			We can conceptualize and implement applications much faster than our competitors.	
			We can design and implement more complex applications than our competitors.	

Chapter 6 Data analysis and discussion

6.1 Data collection

In this study, the questionnaire survey method is used to obtain the data needed for research. The large-scale distribution of questionnaires and data collection mainly include the following aspects:

1. Define the research matrix

In this study, the research matrix is defined as software enterprises.

2. Define the unit of analysis

This dissertation mainly analyzes the impact of data intelligence on software delivery performance from the perspective of employee perception. For the accuracy of the research, this study takes individuals as the unit of analysis.

3. Select samples

In order to ensure the smooth survey and improve the recovery rate of the questionnaire, the author chooses a software enterprise with social connections for survey.

4. Decide the sample size

Since the structural equation model (SEM) is adopted in this study as the analysis method in the confirmatory factor analysis, the sample size of this study is determined at the sample size required by the structural equation. This study adopts the view of Gorsuch and Venable (1983) that the ratio of measurement questions to respondents should be more than 1:5, and preferably 1:10. There are 38 questions in this study, so it is better to have more than 380 samples in total.

5. Send and recover questionnaires

A total of 785 questionnaires are sent out by e-mail, and 763 questionnaires are recovered with a recovery rate of 97.2%. The questionnaire survey lasted for 5 days from September 5, 2022 to September 10, 2022. The author screens the recovered questionnaires and eliminated the invalid ones. The principle of screening is as follows: ① Actual answering time was collected and the normal time needed for answering was estimated. If the former is much shorter than the latter, the respondent was considered answering casually instead of carefully reading the questions, and his/her questionnaire was deemed invalid; ② Respondents with the same IP address were regarded as one, and only the first questionnaire was deemed valid; ③ Questionnaires responded to according to obvious patterns or with excessive same options were deemed invalid; and ④ Questionnaires of respondents less than three years of working experience were deemed invalid. After selection, 400 effective questionnaires are obtained, with the recovery rate of effective questionnaires reaching 52%.

6. Statistics of respondents

Table 12 Classification and statistics of respondents

Classification	Description	Proportion	Remarks
Post	Project manager	28.5%	These four categories of personnel have extensive experience in software project management.
	Consultant	14%	
	R&D personnel	36.5%	
	Implementation personnel	21%	
Education background	Bachelor	77.5%	Software developers generally have a bachelor's degree or above.
	Master	19.5%	
	PhD	3%	
Age	25-29	46.5%	Software developers tend to be younger.
	30-34	30%	
	35-39	14.5%	
	40 and above	9%	
Years of work	3-5	57.5%	Software companies generally maintain stable work force.
	6-9	26.5%	
	10 and above	16%	

6.2 Data analysis process

Later, the author analyzes the data from the 400 questionnaires. Firstly, each item is rated comprehensively. There are eight items: data Intelligence, task level evaluation, coordination, team structure, process control, process integration, modularity, and software delivery performance.

DI: data intelligence TLE: task level evaluation C: coordination

TS: team structure PC: process control PI: process integration M: modularity

SDP: software delivery performance

The number of questions for each project varies, but all the questions are graded between [1,7]. It is assumed that each question in each item has the same impact on the item. Therefore, the average score of questions in each item is adopted as the actual effective score, abbreviated as s.

6.2.1 Impact of data intelligence and intermediate variables on software delivery performance

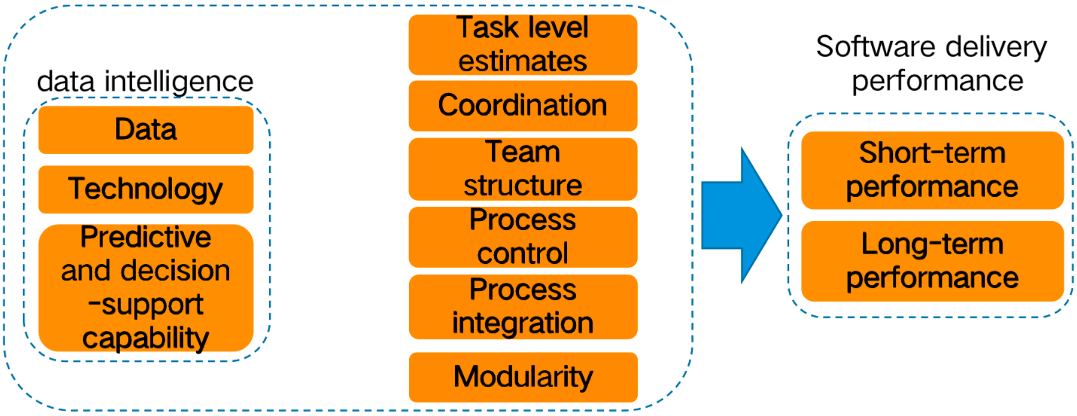


Figure 6 Impact of data intelligence and intermediate variables on software delivery performance

This study tries to add some variables that may affect software delivery performance into the independent variable. The author sets the model as follows for SDP and the three factors of SDP:

$$SDP=\beta_0+\beta_1D+\beta_2T+\beta_3P+\beta_4TLE+\beta_5C+\beta_6TS+\beta_7PC+\beta_8PI+\beta_9M+\mu$$

$$SP=\beta_0+\beta_1D+\beta_2T+\beta_3P+\beta_4TLE+\beta_5C+\beta_6TS+\beta_7PC+\beta_8PI+\beta_9M+\mu$$

$$LP=\beta_0+\beta_1D+\beta_2T+\beta_3P+\beta_4TLE+\beta_5C+\beta_6TS+\beta_7PC+\beta_8PI+\beta_9M+\mu$$

β_0, ... β_9 is the regression coefficient, and μ is the random error term. We can obtain the coefficients of the impact of data intelligence and intermediate variables on software delivery performance and its three factors and p values.

Table 13 Coefficients of the impact of data intelligence and intermediate variables on software delivery performance and its three factors and p values

Dependent variable	Software delivery performance SDP		Short-term performance SP		Long-term performance LP	
Independent variable	Coefficient	P value	Coefficient	P value	Coefficient	P value
Intercept	1.08	1.25e-06 ***	1.07	8.50e-06 ***	1.12	2.90e-06 ***
Data D						
Technology T					-0.13	0.0247
Prediction and decision-making capability P	0.17	0.0006 ***	0.11	0.0044 **	0.23	1.35e-05 ***
Task level evaluation TLE						
Coorporation C						
Team structure TS	0.08	0.0499 *	0.09	0.0283 *	0.09	0.0413 *
Process control PC						
Process integretion PI	0.22	1.34e-08 ***	0.24	1.63e-08 ***	0.22	1.81e-07 ***
Modularity M	0.39	<2e-16 ***	0.40	9.45e-16 ***	0.39	<2e-16 ***

(1) Here, an attempt was made to aggregate the three factors of software delivery performance as a dependent variable. The optimal linear model was determined according to the step function in R language and AIC and by running the code:

lm(SDP~P+TS+PI+M,data =data)

We find that, according to AIC, there is a linear relationship between SDP (software delivery performance) and P (prediction and decision-making capability), TS (team structure), PI (process integration) and M (modularity). We further test the significance of the coefficient of these four items. We find that the p values of P, TS, PI and M are smaller, being 0.0004, 0.04985, 1.34e-08 and <2e-16 respectively, statistically significant, especially P, PI and M. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit, and the goodness of fit R2 of the whole model is 0.6368, greater than 0.5, indicating that there is a significant linear correlation, and the model has a high goodness of fit to data. Therefore, the four main factors affecting SDP (software delivery performance) are P (prediction and decision-making capability), TS (team structure), PI (process integration), and M (modularity).

Then we get the regression equation: $SDP=\beta_0+ \beta_3 P+\beta_6 TS+\beta_8 PI+\beta_9 M$

i.e., $SDP=1.08+0.17P+0.08TS+0.22PI+0.39M$

(2) The optimal linear model was determined according to the step function in R language and AIC and by running the code:

```
lm(SP~P+TS+PI+M,data =data)
```

We find that, according to AIC, there is a linear relationship between SP (Short-term performance) and P (prediction and decision-making capability), TS (team structure), PI (process integration) and M (modularity). We further test the significance of the coefficient of these four items.

We find that p values of P, TS, PI and M are smaller, being 0.00443, 0.02830, 1.63e-08 and 9.45e-16 respectively, statistically significant, especially PI and

M. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit, and the goodness of fit R2 of the whole model is 0.5738. The decision coefficient of the goodness of fit R2 can explain the degree of the independent variable revealing the variation of the dependent variable. Edwards and Bagozzi (2000) believed that R2 greater than 0.5 indicates that there is a significant linear correlation, and the model has a high goodness of fit to data. Therefore, the author believes that the four main factors affecting SP (short-term performance) are P (prediction and decision-making capability), TS (team structure), PI (process integration), and M (modularity).

Then we get the regression equation: $SP=\beta_0+ \beta_3P+\beta_6TS+\beta_8PI+\beta_9M$

That is, SP=1.06+0.11P+0.08TS+0.23PI+0.39M

(3) Let's take a look at LP. With the same approach as SP, the optimal linear model is determined according to the step function in R language and AIC, and by running the code:

```
lm(LP~D+T+P+TS+PI+M,data =data)
```

According to AIC, there is a linear relationship between LP (long-term performance) and D (data), T (technology), P (prediction and decision-making capability), TS (team structure), PI (process integration) and M (modularity). We further test the significance of the coefficient of these 6 items. We find that p values of T, P, TS, PI, and M are smaller, being 0.0247, 1.35e-05, 0.0413, 1.81e-07, and <2e-16 respectively, statistically significant, especially P, PI, and M. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit, and the goodness of fit R2 of the whole model is 0.6009, greater than 0.5, indicating that there is a significant linear correlation, and the model has a high goodness of fit to data. Therefore, the five main factors affecting LP (long-term performance) are T (technology), P (prediction and decision-making capability), TS (team structure), PI (process integration), and M (modularity).

Then we get the regression equation: $LP=\beta_0+\beta_2T+\beta_3P+\beta_6TS+\beta_8PI+\beta_9M$

That is, $LP=1.12-0.13T+0.23P+0.09TS+0.22PI+0.39M$

According to the three regression equations, P, PI and M are always significant and closely related to SDP and its three factors.

6.2.2 Impact of data intelligence on intermediate variables

The above analysis can help us preliminarily understand the impact of data intelligence and intermediate variables on software delivery performance. In order to fully understand the relationship between variables and the influencing mechanism, the author analyzes the impact of data intelligence on intermediate variables. By understanding the impact of three factors of data intelligence on intermediate variables, we can understand the mechanisms by which data intelligence and intermediate variables affect software delivery performance. The specific method is to use the step function in R language and the AIC to determine the optimal linear model. We get the results of the impact of data intelligence factors on six intermediate variables.

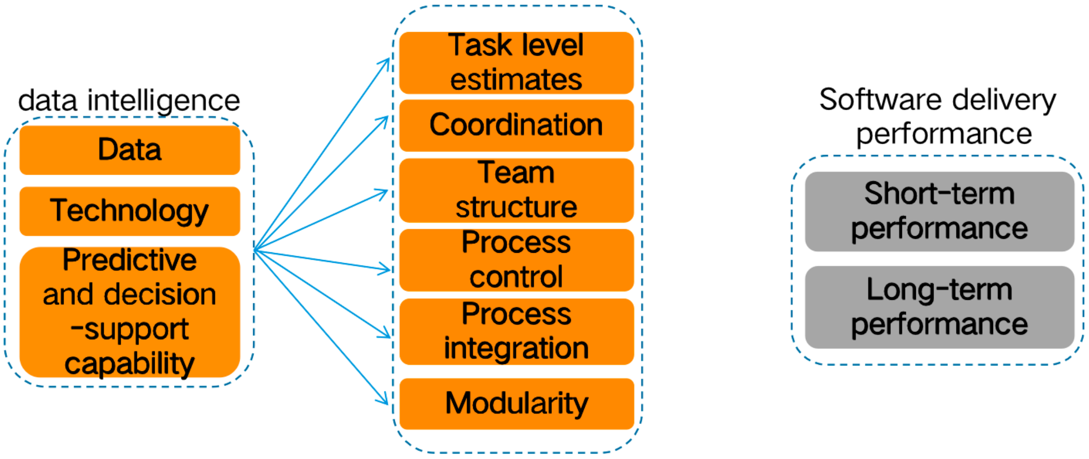
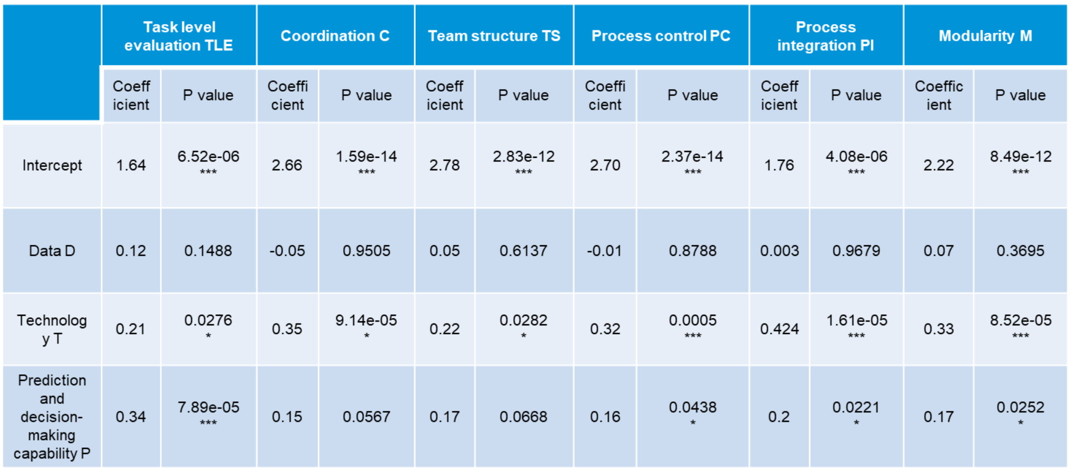


Figure 7 Impact of data intelligence on intermediate variables

In order to fully understand the relationship between variables and the influencing mechanism, we analyzed the impact of data intelligence on intermediate variables. By understanding the impact of three factors of data intelligence on intermediate variables, we were able to understand the mechanisms by which data intelligence and intermediate variables affect software delivery performance. Based on the step function in R language and the AIC, we got the coefficients of the impact of data intelligence on TLE (task level evaluation), C (coordination), team structure (TS), process control (PC), process integration (PI), and modularity (M) and the corresponding p-values, which were summarized as follows:

Table 14 Coefficients of the impact of data intelligence on intermediate variables and P values



	Task level evaluation TLE		Coordination C		Team structure TS		Process control PC		Process integration PI		Modularity M	
	Coefficient	P value	Coefficient	P value	Coefficient	P value	Coefficient	P value	Coefficient	P value	Coefficient	P value
Intercept	1.64	6.52e-06 ***	2.66	1.59e-14 ***	2.78	2.83e-12 ***	2.70	2.37e-14 ***	1.76	4.08e-06 ***	2.22	8.49e-12 ***
Data D	0.12	0.1488	-0.05	0.9505	0.05	0.6137	-0.01	0.8788	0.003	0.9679	0.07	0.3695
Technology T	0.21	0.0276 *	0.35	9.14e-05 *	0.22	0.0282 *	0.32	0.0005 ***	0.424	1.61e-05 ***	0.33	8.52e-05 ***
Prediction and decision-making capability P	0.34	7.89e-05 ***	0.15	0.0567	0.17	0.0668	0.16	0.0438 *	0.2	0.0221 *	0.17	0.0252 *

The optimal linear model is determined according to the step function in R language and AIC, and the optimal linear regression model for TLE is obtained by running the code:

lm (TLE~T+P,data =data)

That is, according to AIC, there is a linear relationship between TLE (task level evaluation) and T (technology) and P (prediction and decision-making capability). We further test the significance of the coefficient and find that p values of T and P are smaller, 0.0276 and 7.89e-05 respectively, statistically significant. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit. T (technology) and P (prediction and decision-making capability) have a significant influence on TLE (task level evaluation).

Then we get the regression equation: TLE=1.64+0.21T+0.34P.

The optimal linear model is determined according to the step function in R language and AIC, and the optimal linear regression model for C is obtained by running the code:

```
lm(C~T,data =data)
```

That is, according to AIC, there is a linear relationship between C (coordination) and T (technology). We further test the significance of the coefficient and find that the p value of T is smaller, 9.14e-05, statistically significant. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit. T (technology) has a significant influence on C (coordination).

Then we get the regression equation: $C=2.65+0.35T$.

The optimal linear model is determined according to the step function in R language and AIC, and the optimal linear regression model for TS is obtained by running the code:

```
lm(TS~T,data =data)
```

That is, according to AIC, there is a linear relationship between TS (team structure) and T (technology). We further test the significance of the coefficient and find that the p value of T is smaller, 0.0282, statistically significant. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit. T (technology) has a significant influence on TS (team structure).

Then we get the regression equation: $TS=2.77+0.22T$.

The optimal linear model is determined according to the step function in R language and AIC, and the optimal linear regression model for PC is obtained by running the code:

```
lm(PC~T+P,data =data)
```

That is, according to AIC, there is a linear relationship between PC (process control) and T (technology) and P (prediction and decision-making capability). We further test the significance of the coefficient and find that p values of T and P are smaller, respectively 0.000513 and 0.04377, statistically significant. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit. T (technology) and P (prediction and decision-making capability) have a significant influence on PC (process control).

Then we get the regression equation: $PC=2.7+0.32T+0.16P$.

The optimal linear model is determined according to the step function in R language and AIC, and the optimal linear regression model for PI is obtained by running the code:

```
lm(PI~T+P,data =data)
```

That is, according to AIC, there is a linear relationship between PI (process integration) and T (technology) and P (prediction and decision-making capability). We further test the significance of the coefficient and find that p values of T and P are smaller, respectively 1.61e-05 and 0.0221, statistically significant. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit. T (technology) and P (prediction and decision-making capability) have a significant influence on PI (process integration).

Then we get the regression equation: PI=1.72+0.42T+0.2P.

The optimal linear model is determined according to the step function in R language and AIC, and the optimal linear regression model for M is obtained by running the code:

```
lm(M~T+P,data =data)
```

That is, according to AIC, there is a linear relationship between M (modularity) and T (technology) and P (prediction and decision-making capability). We further test the significance of the coefficient and find that p values of T and P are smaller, respectively 8.52e-05 and 0.0252, statistically significant. Moreover, the p value of F statistic of the whole model is 2.2e-16, much lower than 0.001, indicating that the model has a goodness of fit. T (technology) and P (prediction and decision-making capability) have a significant influence on M (modularity), showing a linear relationship.

Then we get the regression equation: M=2.22+0.33T+0.17P.

In summary, T and P affect TLE, with the coefficient of 0.2099 and 0.3401 respectively, and P is more significant; T affects C, with the coefficient of 0.3480; T affects TS, with the coefficient of 0.2241; T and P affect PC, with the coefficient of 0.3162 and 0.1639 respectively, and T is more significant; T and P affect PI, with the coefficient of 0.4239 and 0.2002 respectively, and T is more significant; T and P affect M, with the coefficient of 0.3314 and 0.1683 respectively, and T is more significant.

To sum up, T and P have greater influence on intermediate variables.

6.2.3 Analysis on mediating effect

Above we analyzed the impact of data intelligence and intermediate variables on software delivery performance, and the relationship between data intelligence and intermediate variables. It can be seen that P in data intelligence has an impact on both software delivery performance and intermediate variables. In order to further verify the impact of mediating effect, we take the three factors of data intelligence as independent variables, the two factors of software delivery performance as dependent variables, and the six intermediate variables as mediating variables for mediating effect analysis.

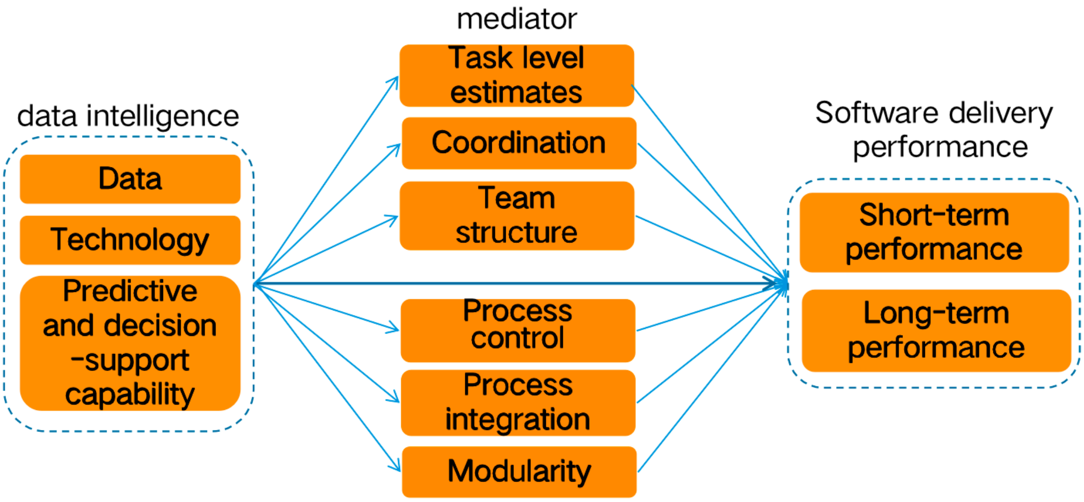


Figure 8 Impact of mediating effect

After analysis, we got the following table:

Table 15 Summary chart of mediating effect

Total effect	TLE effect	C effect	TS effect	PC effect	PI effect	M effect	Independent variable/dependent variable
0.4004	0.1752	0.1342	0.1441	0.1711	**0.2366**	**0.2728**	D/SP
0.4264	0.1491	0.1313	0.1314	0.1465	**0.2144**	**0.2585**	D/LP
0.4152	0.1603	0.1325	0.1368	0.1570	**0.2239**	**0.2666**	D/SDP
0.4641	0.1745	0.1526	0.1604	0.2062	**0.2874**	**0.3187**	T/SP
0.4487	0.1549	0.1563	0.1492	0.1792	**0.2667**	**0.3049**	T/LP
0.4553	0.1633	0.1547	0.1540	0.1908	**0.2756**	**0.3108**	T/SDP
0.4560	0.1847	0.1432	0.1563	0.1931	**0.2630**	**0.2924**	P/SP
0.4999	0.1467	0.1362	0.1394	0.1598	**0.2316**	**0.2662**	P/LP
0.4811	0.1630	0.1392	0.1467	0.1741	**0.2450**	**0.2774**	P/SDP

As can be seen from the figure, when D is the independent variable and SP is the dependent variable, the mediating effect of M and PI is the most significant, accounting for more than 50% of the total effect; when D is the independent variable and LP is the dependent variable, the mediating effect of M is the most significant, accounting for more than 50% of the total effect; when D is the independent variable and SDP is the dependent variable, the mediating effect of M and PI is the most significant, accounting for more than 50% of the total effect.

When T is the independent variable and SP is the dependent variable, the mediating effect of M and PI is the most significant, accounting for more than 50% of the total effect; when T is the independent variable and LP is the dependent variable, the mediating effect of M and PI is the most significant, accounting for more than 50% of the total effect; when T is the independent variable and SDP is the dependent variable, the mediating effect of M and PI is the most significant, accounting for more than 50% of the total effect.

When P is the independent variable and SP is the dependent variable, the mediating effect of M and PI is the most significant, accounting for more than 50% of the total effect; when P is the independent variable and LP is the dependent variable, the mediating effect of M is the most significant, accounting for more than 50% of the total effect; when P is the independent variable and SDP is the dependent variable, the mediating effect of M and PI is the most significant.

In conclusion, the mediating effect of PI and M is significant in all combinations, and the mediating effect of M is the most significant.

Chapter 7 Result and evaluation

7.1 Result

In this study, a large amount of literature is read, and studies on data intelligence and organizational performance are analyzed and summarized. Although the studies on data intelligence and organizational performance have attracted the attention of many scholars, it is difficult to measure and evaluate the relationship between the two. Therefore, based on previous studies and in combination with the actual situation of the enterprise where the author works, the author innovates the research idea and makes an integrated research on the relationship between software delivery performance, the core of performance of software organizations, and data intelligence. According to the existing knowledge theory, and in combination with the advanced research results at home and abroad, the author defines performance of software organizations and software delivery performance, proposes that factors of software delivery performance include three aspects, namely, organizational performance, perceived indirect benefits, and strategic performance, and proposes three factors of data intelligence, namely, data (massive, multi-type), technology (algorithm), and prediction and decision-making capability (value). Then, the author researches software project management, puts forward six intermediate variables affecting software delivery performance, and designs the measurement list of variables in this study. At the same time, on the basis of qualitative research of the relationship among the three, and in combination with the definition of connotation, the author builds the relationship model between variables, and proposes hypotheses to be verified for branch models. An electronic questionnaire is designed according to the model and the measurement list, and sent to individuals, obtaining the sample data on the relevant variables, and effective questionnaires are obtained through screening. According to the specific requirements and characteristics of problems, the multiple linear regression method and structural equation model are used to analyze the sample data, and the model is verified by examples. The results of the empirical test mainly include two aspects: on the one hand, it verifies the model for the impact of data intelligence on software delivery performance; on the other hand, it illustrates the positive effect of data intelligence on software delivery performance through six mediating variables: task level evaluation, coordination, team structure, process control, process integration and modularity. Based on the above analysis, this study mainly draws the following conclusions:

(1) Data intelligence has a significant impact on software delivery performance; in particular, the independent variable prediction and decision-making capability (P) has the most significant impact.

Through the empirical study based on samples, this study analyzes the degree of impact of data intelligence and mediating variables on software delivery performance in detail. Firstly, the structural equation model is constructed for the impact of three aspects of data intelligence on three factors of software delivery performance; the optimal linear model is obtained according to the step function in R language and AIC, and the optimal result is obtained by running the code:

1） Prediction and decision-making capability (P) in data intelligence is significantly correlated with short-term performance (SP) and long-term performance (LP); technology (T) in data intelligence is only significantly correlated with long-term performance (LP). It fully indicates that data intelligence (DI) has a positive impact on software delivery performance (SDP).

2）Data (D) in data intelligence can affect software delivery performance only through the prediction and decision-making capability shown by the algorithm.

The specific impact relationship and coefficient are shown in the figure:

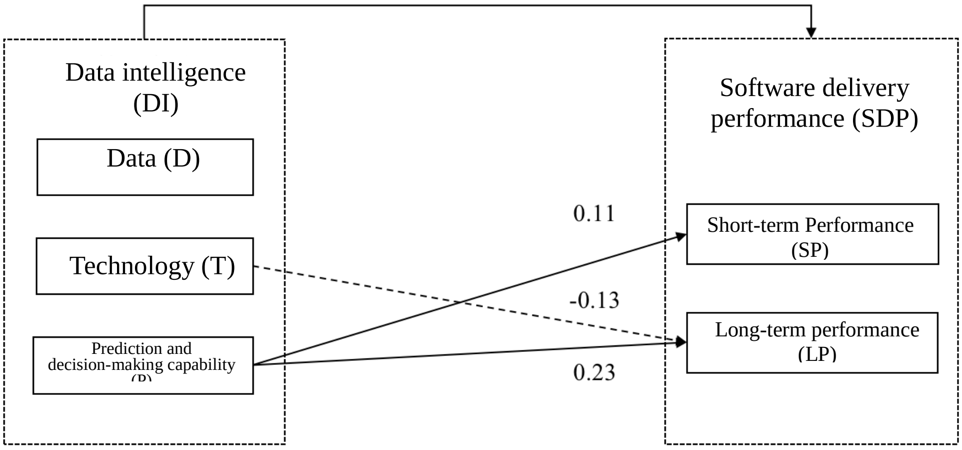

Data intelligence (DI)
Data (D)
Technology (T)
Prediction and decision-making capability (P)
Software delivery performance (SDP)
Short-term Performance (SP)
Long-term performance (LP)
0.11
-0.13
0.23

Figure 9 Chart of the impact relationship between factors of data

intelligence and factors of software delivery performance

(2) Data intelligence has a significant impact on mediating variables; in particular, technology (T) has the most significant impact on mediating variables

By analyzing the impact of three factors in data intelligence on six factors of mediating variables, it can be seen that:

1) Technology (T) in data intelligence has a significant impact on six mediating variables; prediction and decision-making capability (P) in data intelligence has a significant impact on task level evaluation (TLE), process control (PC), process integration (PI) and modularity (M).

2） Data (D) in data intelligence can affect mediating variables only through the prediction and decision-making capability shown by the algorithm.

The specific impact relationship and coefficient are shown in the figure:

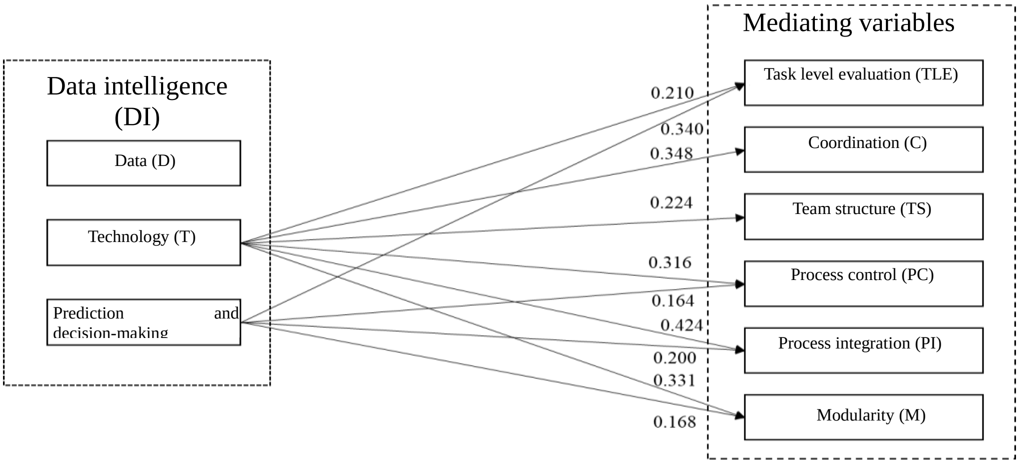


Figure 10 Chart of the impact relationship between factors of data

intelligence on mediating variables

(3) Mediating variables have a significant impact on software delivery performance; in particular, process integration (PI) and modularity (M) have the most significant impact.

Through the detailed analysis on samples, it can be seen that:

1）Process integration (PI), modularity (M) and team structure (TS) have a significant impact on short-term performance (SP) and long-term performance; process integration (PI) and modularity (M) have the most significant impact.

2）Task level evaluation (TLE), coordination (C) and process control (PC) have an insignificant linear relationship with three factors of software delivery performance.

The specific impact relationship and coefficient are shown in the figure:

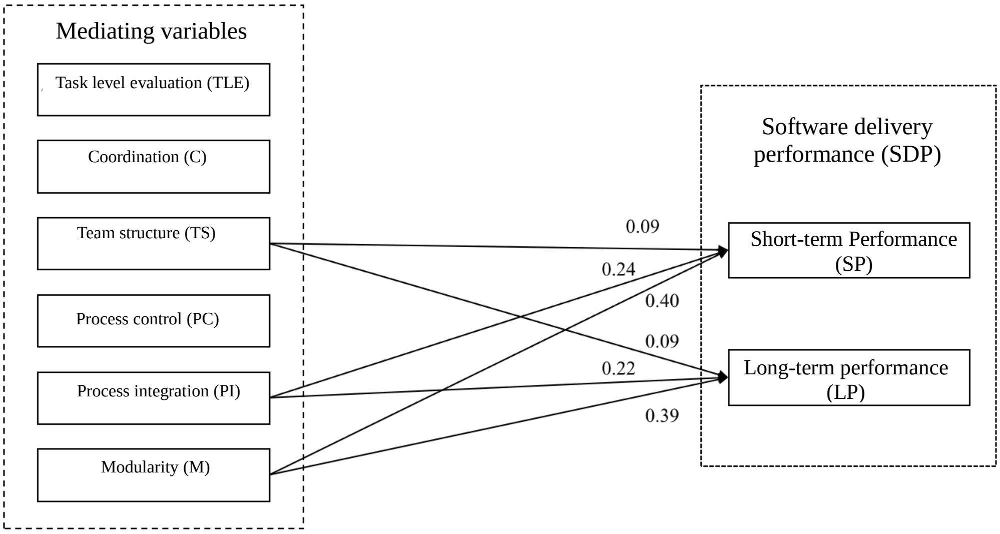


Figure 11 Chart of the impact relationship between mediating variables and factors of software delivery performance

(4) Process integration (PI) and modularity (M) have significant mediating effects, especially modularity (M).

In the analysis on the mediating effect of data intelligence on software delivery performance, we obtain 54 groups of data on influence coefficients (see Figure 12 for detailed values). It can be seen from data that in the impact process of the mediating effect, mediating variables are sequenced by the impact degree from high to low as follows: modularity (M), process integration (PI), process control (PC), task level evaluation (TLE), team structure (TS), and coordination (C), in which the mediation effect of modularity (M) and process integration (PI) accounts for more than 50% of the total effect. It can be seen that data intelligence improves software delivery performance primarily by influencing modularity (M) and process integration (PI). The specific mediating effect of data intelligence on software delivery performance is shown in the figure:

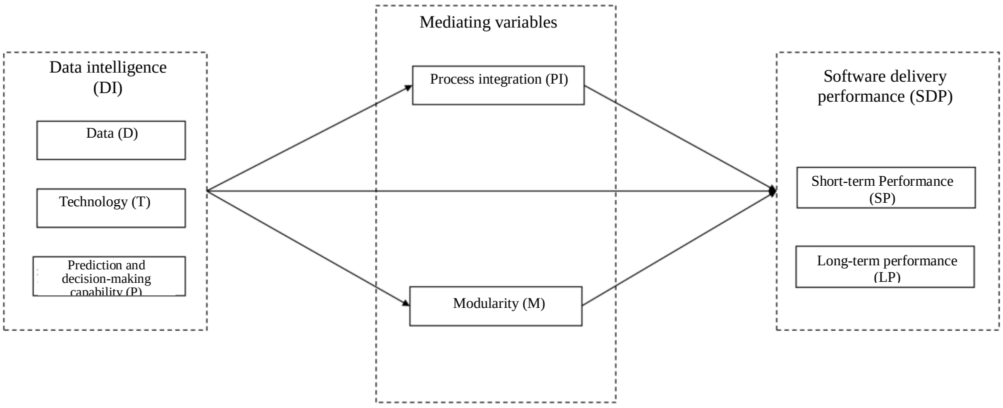

Mediating variables
Data intelligence (DI)
Data (D)
Technology (T)
Prediction and decision-making capability (P)
Process integration (PI)
Modularity (M)
Software delivery performance (SDP)
Short-term Performance (SP)
Long-term performance (LP)

Figure 12 Chart of the relationship of the mediating effect

Conclusion summary:

(1) Analyze and study the connotation and characteristics of data intelligence and software delivery performance.

According to the research in this dissertation, the characteristics of data intelligence are as follows: taking big data as the engine, data analysis and mining through machine learning and deep learning; the connotation of data intelligence is big data, algorithm, and prediction and decision-making capability; software delivery performance is characterized by the delivery ability to meet customer requirements and bringing value to customers; the connotation of software delivery performance is organizational and strategic performance of enterprises and perceived indirect benefits. The main factors of data intelligence affecting software delivery performance are modularity, process integration and team structure.

(2) What are the mechanism and path for data intelligence to improve software

delivery performance?

Through the research in this dissertation, the main mechanism through which data intelligence improves software delivery performance is embodied in three aspects: value chain, management and control mode, and capability. Value chain means that data intelligence directly improves software delivery performance, and has a positive impact on software delivery performance through modularity (M), process integration (PI), and team structure (TS); management and control mode means that data intelligence technologies and methods are used to affect modularity (M), process integration (PI) and team structure (TS), so as to improve the management and control of software delivery performance; capability is mainly reflected in the decision-making capability of improving software delivery by using big data and through machine learning and other algorithms.

Through the research in this dissertation, the path through which data intelligence improves software delivery performance is that the prediction and decision-making capability in data intelligence directly improves software delivery performance; data intelligence positively impacts software delivery performance through modularity, process integration, and team structure. The specific path is shown in the following figure.

Software delivery is mainly divided into software product delivery and software customized system delivery. The biggest difference between the two is that the requirements of software customization system are often unclear, especially for application software, which requires the delivery team to spend a lot of time and energy to communicate, understand the requirements, and constantly optimize and iterate. From the perspective of software development methods, they basically follow the same software engineering methods for development, there is little difference. From a technical point of view, data intelligence has a direct supporting effect on software products with clear needs because it integrates big data, machine learning, deep learning and other technologies. From the perspective of management, the delivery management of customized software system is more difficult. Due to the uncertainty of requirements, the boundary of software is difficult to define, which brings difficulties in software delivery and consumes a lot of manpower and material resources. Through data intelligence, we can improve the evaluation level of the early stage of the project, optimize the project management process and the business process in the software system, and accumulate more software modules. The mediation variables will exert more influence on the delivery process of software system and have a positive impact. The mediation variables will have a better supporting effect on the delivery of software customized system projects.

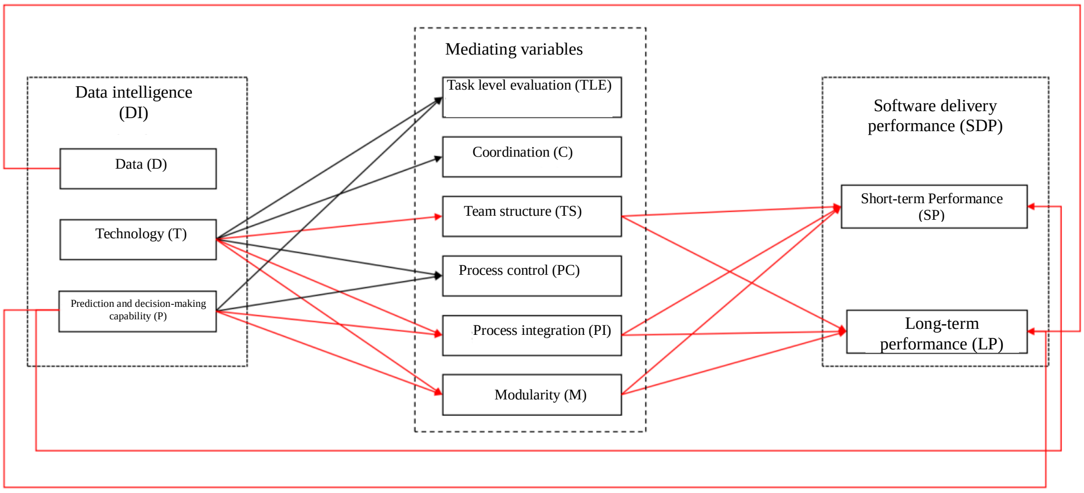

Mediating variables
Data intelligence (DI)
Data (D)
Technology (T)
Prediction and decision-making capability (P)
Task level evaluation (TLE)
Coordination (C)
Team structure (TS)
Process control (PC)
Process integration (PI)
Modularity (M)
Software delivery performance (SDP)
Short-term Performance (SP)
Long-term performance (LP)

Figure 13 Path through which data intelligence affects software delivery performance

(Black only indicates influence, while red indicates influence path)

7.2 Limitations and prospects

7.2.1 Limitations

Although the empirical research of this dissertation has obtained some important conclusions with theoretical and practical significance, due to the limitations of personal theoretical level and other material resources and time, there are still many deficiencies that need to be further explored in the future, specifically in the following aspects:

(1) The theory needs to be further studied. Since data intelligence and software project delivery are complex systems, involving technology, organization, management, resource allocation, third party and many other aspects, there are still many problems to be further improved. Based on the respective

perspectives of data intelligence and software delivery performance, this dissertation puts forward three aspects to measure the two, and the mediating variable of the impact, and conducts an empirical analysis according to the questionnaire. Although some valuable theories have been obtained, there are still some problems to be further studied, such as whether the selection of variables is perfect.

(2) The questionnaire survey is mainly based on subjective data, with insufficient attention to objective data, so it has certain limitations. The scope of the questionnaire needs to be further expanded. Due to factors such as time, manpower, and cost, the questionnaire only selected some employees of the enterprise where the author works, so the regional and industry distribution of the sample has certain limitations, which may reduce the external validity of this study to a certain extent, result in insufficient reasoning ability and may limit the generalizability of the findings. Therefore, in future explorations, we need to further expand the scope of sample selection and improve sampling techniques to enhance the representativeness, persuasiveness, and general applicability of research theories.

(3) This study focuses on the research on data intelligence technologies related to the organizational structure of the enterprise and mastered by the enterprise, software delivery performance and factors affecting delivery performance, without taking into account the development history of the enterprise, analysis

on core customer base, personal style of senior leaders, background of service-related industries and local external environment such as tax policy factors. Although this study proposes some new ideas on the basis of existing theories, its scope and depth are insufficient due to limited experience and energy.

7.2.2 Prospects

The author believes that the methods and ideas for research on the impact of data intelligence on software delivery performance can be applied and promoted in other industries or organizations, so as to use data intelligence to promote the digital transformation of Chinese economy. This study is aimed at improving software delivery performance of organizations in the software industry. This study finds the mechanism and way for software organizations to improve software delivery performance, so as to enhance the operational performance of software organizations. Under the current social background of the upsurge in the application of big data and artificial intelligence technologies, other industries or companies can use the ideas and methods formed in this dissertation to explore the factors influencing the operational performance, find mediating variables, build the mechanism through which data intelligence improves the operational capability, enhance the digital transformation capability of the industry, and promote the operational efficiency of organizations through digital transformation, so as to promote the economic efficiency of the whole society, and contribute to the digital transformation and the high-quality economic development of China.

Reference

Adamson, C. (2015). The age of big data modeling. www.tdwi.org

Akaike, H. (1974). A new look at the statistical model identification. IEEE transactions on automatic control, 19(6), 716-723.

Amit, R., & Schoemaker, P. J. (1993). Strategic assets and organizational rent. Strategic management journal, 14(1), 33-46.

Ardana, I. M. S. (2017, October). Software development evaluation process using CMMI-Dev on limited resources company. In 2017 3rd International Conference on Science in Information Technology (ICSITech) (pp. 319-324). IEEE

Bakici, T., Nemeh, A., & Hazir, Ö. (2021). Big Data Adoption in Project Management: Insights From French Organizations. IEEE Transactions on Engineering Management.

Barney, S., Aurum, A., & Wohlin, C. (2008). A product management challenge: Creating software product value through requirements selection. Journal of Systems Architecture, 54(6), 576-593.

Barney.J. B. (1986).Strategic Factor Markets: Expectations,Luck and Business Strategy, Management Science 32(10): PP.1231-1241.

Barney.J.B. (1991).Firm Resource and sustained CompetitiveAdvantage. Journal of Management,17: 99-120.

Barton, D., & Court, D. (2012). Making advanced analytics work for you. Harvard business review, 90(10), 78-83.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). Manifesto for agile software development.

Benesty, J., Chen, J., & Huang, Y. (2008). On the importance of the Pearson correlation coefficient in noise reduction. IEEE Transactions on Audio, Speech, and Language Processing, 16(4), 757-765.

Biao Sun. (2012). Research on Organizational Structure Reengineering of Large Commercial Banks in China (Doctoral dissertation, Beijing Jiaotong University).

Bizer, C., Boncz, P., Brodie, M. L., & Erling, O. (2012). The meaningful use of big data: four perspectives--four challenges. ACM Sigmod Record, 40(4), 56-60.

Bo Liu. (2008). Research on Key Technologies of XML Data Intelligent Management (Doctoral dissertation, Wuhan, Hubei Province, China: Central South University).

Cao, P., & Chen, F. (2009, December). A risk control optimization model for software project. In 2009 International Conference on Computational Intelligence and Software Engineering (pp. 1-4). IEEE.

Carver, C. S., & Scheier, M. F. (2012). Attention and self-regulation: A control-theory approach to human behavior. Springer Science & Business Media.

Cattell, R. (2011). Scalable SQL and NoSQL data stores. Acm Sigmod Record, 39(4), 12-27.

Chavarria, A. E., Ore, S. B., & Pastor, C. (2016). Quality assurance in the software development process using CMMI, TSP and PSP/Aseguramiento de la calidad en el proceso de desarrollo de software utilizando CMMI, TSP y PSP. RISTI (Revista Iberica de Sistemas e Tecnologias de Informacao), (20), 62-78.

Chen Yang. (2018). Software Architecture Assumptions and Management in Software Development (Doctoral dissertation, Wuhan University).

Chen, G., Wu, G., Gu, Y., Lu, B. J., & Wei, Q. (2018). The challenges for big data driven research and applications in the context of managerial decision-making: Paradigm shift and research directions. Journal of Management Sciences in China, 21(7), 1-10.

Chen, L. (2015). Continuous delivery: Huge benefits, but challenges too. IEEE software, 32(2), 50-54.

Child, J. (1972). Organization structure and strategies of control: A replication of the Aston study. Administrative science quarterly, 163-177.

Choudhury, V., & Sabherwal, R. (2003). Portfolios of control in outsourced software development projects. Information systems research, 14(3), 291-314.

Choudhury, V., & Sabherwal, R. (2003). Portfolios of control in outsourced software development projects. Information systems research, 14(3), 291-314.

Chuan Qin. (2021). Data Mining Methods for Intelligent Recruitment and Their Application (Doctoral dissertation, University of Science and Technology of China).

Coleman, W. (1986). The Managerial Grid III: The Key to Leadership Excellence-Blake, R. R., and Mouton, J. S.

Cooprider, J. G., & Henderson, J. C. (1990). Technology–process fit: perspectives on achieving prototyping performance. Journal of management information systems, 7(3), 67-87.

Cooprider, J. G., & Henderson, J. C. (1990). Technology–process fit: perspectives on achieving prototyping performance. Journal of management information systems, 7(3), 67-87.

Curtis, B., Krasner, H., & Iscoe, N. (1988). A field study of the software design process for large systems. Communications of the ACM, 31(11), 1268-1287.

Curtis, B., Krasner, H., & Iscoe, N. (1988). A field study of the software design process for large systems. Communications of the ACM, 31(11),

1268-1287.

Di Chen. (2017). Optimization Method for Project Scheduling and Multi-Skilled Human Resource Allocation (Doctoral dissertation, Northeastern University).

Di Vaio, A., Hassan, R., & Alavoine, C. (2022). Data intelligence and analytics: A bibliometric analysis of human–Artificial intelligence in public sector decision-making performance. Technological Forecasting and Social Change, 174, 121201.

Du, W. L., Ho, D., & Capretz, L. F. (2015). Improving software effort estimation using neuro-fuzzy model with SEER-SEM. arXiv preprint arXiv:1507.06917.

Duan, Y., Edwards, J. S., & Dwivedi, Y. K. (2019). Artificial intelligence for decision making in the era of Big Data–evolution, challenges and research agenda. International journal of information management, 48, 63-71.

Dutoit, A. H., & Bruegge, B. (1998). Communication metrics for software development. IEEE transactions on Software Engineering, 24(8), 615-628.

Fairbank, J. F., Labianca, G. J., Steensma, H. K., & Metters, R. (2006). Information processing design choices, strategy, and risk management performance. Journal of Management Information Systems, 23(1), 293-319.

Gao, N., & Touran, A. (2020). Cost overruns and formal risk assessment program in US rail transit projects. Journal of Construction Engineering and Management, 146(5), 05020004.

Garg, N., Sadiq, M., & Agarwal, P. (2017). GOASREP: Goal oriented approach for software requirements elicitation and prioritization using analytic hierarchy process. In Proceedings of the 5th international conference on frontiers in intelligent computing: theory and applications (pp. 281-287). Springer, Singapore.

Gharehchopogh, F. S., & Pourali, A. (2015). A new approach based on continuous genetic algorithm in software cost estimation. J. Sci. Res. Dev, 2(4), 87-94.

Goncalves, T. G., Oliveira, K., & Kolski, C. (2016, October). Verification and validation of interactive systems in the CMMI-DEV capability maturity model. In Actes de la 28ième conference francophone sur l'Interaction Homme-Machine(pp. 48-58).

Gonçalves, T. G., Oliveira, K. M., & Kolski, C. (2018). Identifying HCI approaches to support CMMI-DEV for interactive system development. Computer Standards & Interfaces, 58, 53-86.

Gopal, A., Espinosa, J. A., Gosain, S., & Darcy, D. P. (2011). Coordination and performance in global software service delivery: The vendor's perspective. IEEE Transactions on Engineering Management, 58(4), 772-785.

Gopal, A., Espinosa, J. A., Gosain, S., & Darcy, D. P. (2011). Coordination and performance in global software service delivery: The vendor's perspective. IEEE Transactions on Engineering Management, 58(4), 772-785.

Gopal, A., Mukhopadhyay, T., & Krishnan, M. S. (2002). The role of software processes and communication in offshore software development. Communications of the ACM, 45(4), 193-200.

Guide, A. (2017). Project management body of knowledge (pmbok® guide). In Project Management Institute.

Guowei Wei. (2019). Research on the Relationship among Competitive Environment, Dynamic Capability and Innovation Performance of Enterprises from the Perspective of Ecological Niche (Doctoral dissertation, Beijing University of Posts and Telecommunications).

Hai Liu, Yuanzhe Zhou, & Yan Chen. (2012). Software Project Management. China Machine Press.

Hazen, B. T., Boone, C. A., Ezell, J. D., & Jones-Farmer, L. A. (2014). Data quality for data science, predictive analytics, and big data in supply chain management: An introduction to the problem and suggestions for research and applications. International Journal of Production Economics, 154, 72-80.

Healey M P.,Hodgkinson G P., Teo S（2009）. Responding effectively to civil energencies: The role of transactive memory in the performance of multi-team systems.Proceedings of NDM9，the 9th International Conference on Naturalistic Decision Making. London, UK, 209,June.

Hoegl, M., & Weinkauf, K. (2005). Managing task interdependencies in Multi-Team projects: A longitudinal study. Journal of Management Studies, 42(6), 1287-1308.

Honggen Zheng. (2018). Research on the Application of Big Data and Artificial Intelligence Technologies by Enterprises to Improve the Evaluation Quality of Investment Projects. Chinese Science and Technology Journal Database (Full-text Edition) - Economic Management, (05), 00277-00279.

Hongjiang Lv, Hong Liu, & Kaicheng Chen. (2008). Relationship between Organizational Complexity and Organizational performance in the Context of Transformation: Impact of Environmental Uncertainty. Proceedings of the Third (2008) Annual Chinese Management Conference -- Entrepreneurship and SME Management.

Hongjiang Lv, Hong Liu, & Kaicheng Chen. (2008). Relationship between Organizational Complexity and Organizational performance in the Context of Transformation: Impact of Environmental Uncertainty. Proceedings of the Third (2008) Annual Chinese Management Conference -- Entrepreneurship and SME Management.

Hrebiniak,Lawence G.& William F.Joyce. (1985). organiz-tional Adaptation;Strategic Choice and Environmental Determination,Administrative Science Quarterly,(September) 。336-349.

Huihua Shang. (2011). Software Quality Management Model Based on Process Improvement. Computer Engineering and Design, 32(5), 5.

Jagtiani, J., Bach, C., & Huntley, C. (2018). Leveraging big data from open source to improve software project management. IEEE Engineering Management Review, 46(1), 65-79.

Jianying Zhang, Gang Wang, Huiming Gong, Dongyang Ma, & Deyan Zhao. (2020). Exploration of the New Thinking of Software Project Management under the Background of Big Data. XINXI JILU CAILIAO.

Jie Fang, & Minqiang Zhang. (2012). Point Estimation and Interval Estimation of Mediating Effect: Product Distribution Method, Nonparametric Bootstrap and MCMC Method. Acta Psychologica Sinica, 44(10), 1408.

Jie Zhang. (2019). Algebraic Methods and Empirical Research on Reliability Evaluation of Complex Software (Doctoral dissertation, Hefei University of Technology).

Jin Xiao. (2009). Research on Software Requirements Management Based on Use Cases. Nuclear Power Engineering (S2), 6.

Jingxue, L., & Qi, F. (2007). Research of intelligence data mining based on commanding decision-making. Journal of Systems Engineering and Electronics, 18(2), 275-280.

Jinyuan Zhao. (2015). Research on Risk Identification, Evaluation and Control of Manufacturing Software Projects (Doctoral dissertation, University of Science and Technology Beijing).

Junjian Gu, & Yang Xiao. (2020). Agile Models and Methods for Public Project Management. Project Management Review.

Junjie Wu, Guannan Liu, Jingyuan Wang, Yuan Zuo, Hui Bu, & Hao Lin. (2020). Data Intelligence: Trend and Challenge. System Engineering Theory and Practice, 40(8), 2116-2149.

Kaushik, A., Soni, A. K., & Soni, R. (2016). An improved functional link artificial neural networks with intuitionistic fuzzy clustering for software cost estimation. International Journal of System Assurance Engineering and Management, 7(1), 50-61.

Khan, A. W., Hussain, I., & Zamir, M. (2021). Analytic hierarchy process-based prioritization framework for vendor's reliability challenges in global software development. Journal of Software: Evolution and Process, 33(3), e2310.

Kotlarsky, J., & Oshri, I. (2005). Social ties, knowledge sharing and successful collaboration in globally distributed system development projects. European Journal of Information Systems, 14(1), 37-48.

Kotlarsky, J., & Oshri, I. (2005). Social ties, knowledge sharing and successful collaboration in globally distributed system development projects. European Journal of Information Systems, 14(1), 37-48.

Kraut, R. E., & Streeter, L. A. (1995). Coordination in software development. Communications of the ACM, 38(3), 69-82.

Kuan, K. K., & Chau, P. Y. (2001). A perception-based model for EDI adoption in small businesses using a technology–organization–environment framework. Information & management, 38(8), 507-521.

Kuan, K. K., & Chau, P. Y. (2001). A perception-based model for EDI adoption in small businesses using a technology–organization–environment framework. Information & management, 38(8), 507-521.

Kulkarni, R. H., & Padmanabham, P. (2017). Integration of artificial intelligence activities in software development processes and measuring performance of integration. Iet Software, 11(1), 18-26.

Kumar, S. A., & Kumar, T. A. (2011). Study the impact of requirements management characteristics in global software development projects: an ontology based approach. International Journal of Software Engineering & Applications, 2(4), 107.

Kumari, S., & Pushkar, S. (2017). Software cost estimation using cuckoo search. In Advances in Computational Intelligence (pp. 167-175). Springer, Singapore.

Kun Yang, & Erping Wang. (2004). Communication and Coordination of Software Development Organizations. Advances in Psychological Science, 12(6), 932-941.

Lai L. S. (1997) A synergistic approach to project management in information systems development. International Journal of Project Management. 15(3),173-179.

Larman. (2013). Agile & Iterative Development. Posts & Telecom Press.

Larson, D., & Chang, V. (2016). A review and future direction of agile, business intelligence, analytics and data science. International Journal of Information Management, 36(5), 700-710.

Lee, J. D. M. C. K., & Toutanova, K. (2018). Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Li Cui. (2013). Research on Integration of Implementation Processes of Multi-Emergency Response Based on Knowledge Element (Doctoral dissertation, Dalian University of Technology).

Li Yang. Research on Methods and Models of Risk Management of Software Projects. (Doctoral dissertation, Nanjing University of Aeronautics and Astronautics).

Lihong Song. Research on Multi-project Collaborative Management of Software Organizations. (Doctoral dissertation, Tianjin University).

Lin, R. J., Tan, K. H., & Geng, Y. (2013). Market demand, green product innovation, and firm performance: evidence from Vietnam motorcycle industry. Journal of Cleaner Production, 40, 101-107.

Lin, R. J., Tan, K. H., & Geng, Y. (2013). Market demand, green product innovation, and firm performance: evidence from Vietnam motorcycle industry. Journal of Cleaner Production, 40, 101-107.

Long Chen, Wei Ye, & Shikun Zhang. (2016). Onboard: A Data-Driven Agile Software Development Collaboration Tool. Journal of Computer Research and Development, 53(12), 2753-2767.

Lopez-Arredondo, L. P., Perez, C. B., Villavicencio-Navarro, J., Mercado, K. E., Encinas, M., & Inzunza-Mejia, P. (2019). Reengineering of the software development process in a technology services company. Business Process Management Journal, 26(2), 655-674.

Lord, R. G., & Hanges, P. J. (1987). A control system model of organizational motivation: Theoretical development and applied implications. Behavioral Science, 32(3), 161-178.

Maoshan Qiang, & Hanchen Jiang. (2018). Digital-driven: Engine of Deep Innovation in Project Management. Collections, 2.

McAfee, A., Brynjolfsson, E., Davenport, T. H., Patil, D. J., & Barton, D. (2012). Big data: the management revolution. Harvard business review, 90(10), 60-68.

Mendes, E., Watson, I., Triggs, C., Mosley, N., & Counsell, S. (2002, June). A comparison of development effort estimation techniques for web hypermedia applications. In Proceedings Eighth IEEE Symposium on Software Metrics (pp. 131-140). IEEE.

Mithas, S., & Whitaker, J. (2007). Is the world flat or spiky? Information intensity, skills, and global service disaggregation. Information systems research, 18(3), 237-259.

Mittal, A., Parkash, K., & Mittal, H. (2010). Software cost estimation using fuzzy logic. ACM SIGSOFT Software Engineering Notes, 35(1), 1-7.

Moayedikia, A., Ong, K. L., Boo, Y. L., & Yeoh, W. G. (2018). Task assignment in microtask crowdsourcing platforms using learning automata. Engineering Applications of Artificial Intelligence, 74, 212-225.

Negash, S., & Gray, P. (2008). Business intelligence. In Handbook on decision support systems 2 (pp. 175-193). Springer, Berlin, Heidelberg.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. Y. (2011, January). Multimodal deep learning. In ICML.

Nystrom, P. C. (1978). Managers and the hi-hi leader myth. Academy of management Journal, 21(2), 325-331.

Papadopoulos, T., Gunasekaran, A., Dubey, R., Altay, N., Childe, S. J., & Fosso-Wamba, S. (2017). The role of Big Data in explaining disaster resilience in supply chains for sustainability. Journal of Cleaner Production, 142, 1108-1118.

Park, Y., Park, H., Choi, H., & Baik, J. (2006, July). A study on the application of six sigma tools to PSP/TSP for process improvement. In 5th IEEE/ACIS International Conference on Computer and Information Science and 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse (ICIS-COMSAR'06) (pp. 174-179). IEEE.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013, May). On the difficulty of training recurrent neural networks. In International conference on machine learning (pp. 1310-1318). PMLR.

Pei Xu. (2006). Research on the performance of Software Development Project Teams. (Doctoral dissertation, Hohai University).

Penrose.E.T.(1959). The Theory of the Growth of firm Ox-ford: Basil Blackwell.

Preacher, K. J., & Hayes, A. F. (2008) . Asymptotic and resampling strategies for assessing and comparing indirect effects in multiple mediator models. Behavior Research Methods, 40(3), 879–891. [PDF]

Preacher, K. J., Rucker, D. D., & Hayes, A. F. (2007) . Addressing moderated mediation hypotheses: Theory, methods, and prescriptions. Multivariate Behavioral Research, 42, 185–227.

Pressman, R. S. (2005). Software engineering: a practitioner's approach. Palgrave macmillan.

Provost, F., & Fawcett, T. (2013). Data science and its relationship to big data and data-driven decision making. Big data, 1(1), 51-59.

Qiao, L., Tang, F., & Liu, J. (2018, May). Feedback based high-quality task assignment in collaborative crowdsourcing. In 2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA) (pp. 1139-1146). IEEE.

Qing Yang, Gaoning Wu, & Lizhen Wang. (2017). Big Data: A New Perspective of Data-driven Project Management. System Engineering Theory and Practice, 37(3), 710-719.

Rahman, H., Roy, S. B., Thirumuruganathan, S., Amer-Yahia, S., & Das, G. (2015, November). Task assignment optimization in collaborative crowdsourcing. In 2015 IEEE International Conference on Data Mining (pp. 949-954). IEEE.

Ramin, F. (2020, October). The role of egocentric bias in undergraduate Agile software development teams. In 2020 IEEE/ACM 42nd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)(pp. 122-124). IEEE.

Rao, G. S., Krishna, C. V. P., & Rao, K. R. (2014). Multi objective particle swarm optimization for software cost estimation. In ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India-Vol I (pp. 125-132). Springer, Cham.

Riaz, M. N. (2017, December). Factors affecting the transition time between capability maturity model integration (CMMI) levels in software industry of Pakistan: An empirical study. In 2017 International Conference on Information and Communication Technologies (ICICT) (pp. 90-96). IEEE.

Rijwani, P., & Jain, S. (2016). Enhanced software effort estimation using multi layered feed forward artificial neural network technique. Procedia Computer Science, 89, 307-312.

Rong Yan. (2021). Transient Stability Evaluation of Power Systems Based on Data Drive (Doctoral dissertation, Zhejiang University).

S.L. Pfleeger. (2003). Software Engineering: Theory and Practice. Higher Education Press.

Schlickman, J. J. (2003). ISO 9001: 2000 quality management system design. Artech House.

Senapathi, M., Buchan, J., & Osman, H. (2018, June). DevOps capabilities, practices, and challenges: Insights from a case study. In Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018 (pp. 57-67).

Shaojian Qiu. (2019). Research on Key Technologies of Cross-project Software Defect Prediction Based on Transfer Learning (Doctoral dissertation, South China University of Technology).

Shuquan Zheng (Ed.). (2017). Industrial Big Data: Architecture and Application. Shanghai ke xue ji shu chu ban she.

Silva, L. S. P., Sampaio, S. C., De Souza, E. R., Moreira, R. T., & Vasconcelos, A. M. (2016, September). Mapping between the guide of it solution contract and CMMI models: A qualitative analysis. In 2016 10th International Conference on the Quality of Information and Communications Technology (QUATIC) (pp. 150-153). IEEE.

Singh, S. K., & El-Kassar, A. N. (2019). Role of big data analytics in developing sustainable capabilities. Journal of cleaner production, 213, 1264-1273.

Singh, S. K., & El-Kassar, A. N. (2019). Role of big data analytics in developing sustainable capabilities. Journal of cleaner production, 213, 1264-1273.

Song, L., Minku, L. L., & Yao, X. (2013, October). The impact of parameter tuning on software effort estimation using learning machines. In Proceedings of the 9th international conference on predictive models in software engineering (pp. 1-10).

Sonnentag, S. (1995). Excellent software professionals: Experience, work activities, and perception by peers. Behaviour & Information

Technology, 14(5), 289-299.

Sonnentag, S. (2000). Excellent performance: The role of communication and cooperation processes. Applied Psychology, 49(3), 483-497.

Srinivas, N., & Deb, K. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary computation, 2(3), 221-248.

Stellman, A., & Greene, J. (2005). Applied software project management. " O'Reilly Media, Inc.".

Team, C. P. (2010). Improving processes for developing better products and services, CMMI RG for Development. Version 1.3. Technical report, Carnegie Mellon University, Pittsburgh.

Teo, T. S., & Bhattacherjee, A. (2014). Knowledge transfer and utilization in IT outsourcing partnerships: A preliminary model of antecedents and outcomes. Information & Management, 51(2), 177-186.

Teo, T. S., & Bhattacherjee, A. (2014). Knowledge transfer and utilization in IT outsourcing partnerships: A preliminary model of antecedents and outcomes. Information & Management, 51(2), 177-186.

Thong, J. Y., Yap, C. S., & Raman, K. S. (1996). Top management support, external expertise and information systems implementation in small businesses. Information systems research, 7(2), 248-267.

Van de Ven, A. H., Delbecq, A. L., & Koenig Jr, R. (1976). Determinants of coordination modes within organizations. American sociological review, 322-338.

Vicinanza, S. S., Mukhopadhyay, T., & Prietula, M. J. (1991). Software-effort estimation: an exploratory study of expert performance. Information systems research, 2(4), 243-262.

Wang, Q., & Song, Z. H. (2010, August). Research on optimization of software development project process based on theory of project management. In 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE) (Vol. 1, pp. V1-426). IEEE.

Wang, W., Lu, X., Jia, R., & Li, F. (2017, December). Development of mass spectrometer software project based on CMMI. In 2017 3rd IEEE International Conference on Computer and Communications (ICCC) (pp. 2508-2511). IEEE.

Wang, Z., & Perry, D. E. (2015, December). Role distribution and transformation in open source software project teams. In 2015 Asia-Pacific Software Engineering Conference (APSEC) (pp. 119-126). IEEE.

Wei, C., Ma, R., & Wang, X. D. (2014). Analysis on difficulties and measures of the age of big data brings to the enterprise competitive intelligence. In Advanced Materials Research(Vol. 1030, pp. 1886-1891). Trans Tech Publications Ltd.

Weigong Zhuang. (2020). Application and Development Trend of Computer Software Development Technology under the Background of Big Data. Technology Innovation and Application.

Welsh, J. A., White, J. F., & Dowell, P. (1981). A small business is not a little big business harvard business review. International Small Business Journal: Researching Entrepreneurship, 18.

Wernerfelt, B. (1984). A resource-based view of the firm. Strategic management journal, 5(2), 171-180.

Wernerfelt. (1984).A resource- based View of the Firm StrategicManagement journal,5:171-180.

Wu, J., Liu, G., Wang, J., Zuo, Y., Bu, H., & Lin, H. (2020). Data intelligence: trends and challenges. Syst. Eng.-Theory Pract, 40, 2116-2149.

Wu, Q., Wu, C., & Liu, Y. (2018). Research on the development of scientific and technological intelligence in big data environment. In ITM Web of Conferences (Vol. 17, p. 03018). EDP Sciences.

Xiang Zhang. (2013). Research on Cross-functional Team Construction from the Perspective of Team Performance (Doctoral dissertation, Tianjin University).

Xiao Yang. (2019). Research on Factors Affecting Software Quality in Software Development. Collections, 11.

Xiao Zhuang, Yizhen Zhang, Peng Ou, Hengxuan Miao, & Yuan Xia. (2013). Continuous Improvement of Software Process System Based on CMMI Framework. Computer Applications and Software, 29(1), 186.

Xiaohong Shan. (2010). Research on Issues concerning Response to Risks in Software Project Development. (Doctoral dissertation, Beijing University of Technology).

Xiaohui Xu. (2011). Research on Key Technologies and Formalization of Dynamic Updating of Component Software [D] (Doctoral dissertation, Shanghai Jiao Tong University).

Xiaokun Sun. (2021). Cost Deviation Control Method and Information System Design for Complex and large-scale Construction Projects (Doctoral dissertation, North China Electric Power University, Beijing).

Xiaoqian Ning. (2004). Software Development Project Management Based on System Dynamics (Doctoral dissertation, Fudan University).

Xinbo Sun, Yu Qian, Mingchao Zhang, & Jinzhu Li. (2019). Research on the Realization Mechanism of Enterprise Supply Chain Agility Driven by Big Data. Management World, 35(9), 133-151.

Xinxin Chen. (2012). Analysis of Software Project Management. CO-Operative Economy & Science, (9), 50-51.

Yangjun Chen. (2017). Research on the Influence Mechanism of Behavior Integration of the Construction Project Management Team on Team performance (Master's thesis, Kunming University of Science and

Technology).

Yilin Yin, & Pei Zhou. (2008). Pmo - A New Form of Multi-project Management Organization of Enterprises. Journal of Harbin University of Commerce: Social Science Edition (2), 3.

Yu Gao, & Zhenbo Bi. (2008). Development of Software Development Process Model. Computer Technology and Development, 18(7), 83-86.

Yuan Ji. (2018). New Thinking of Financial Management in the Era of Artificial Intelligence and Big Data. Collections, 34.

Yuanling Ye. (2013). Research on Technological Innovation Ability of Software Enterprises Based on Knowledge Sharing [D] (Doctoral dissertation, Peking University).

Zhangwen Liu. (1980). Akike Information Criterion (AIC) and its Significance. Journal of Mathematics in Practice and Theory, 3, 64-72.

Zhenghong Yang, Liangyue Guo, & Wei Liu. (2019). Introduction to Artificial Intelligence and Big Data Technologies (Vol.77). Tsinghua University Press (Song Bo).

Zhichang Qi, Qingping Tan, & Hong Ning. (2019). Software Engineering (4th Edition). Higher Education Press.

Zhixiang Tong. (2018). Research on Key Technologies of Search-based Software Requirements Optimization (Doctoral dissertation, Harbin Institute of Technology).

Zhonglin Wen, & Baojuan Ye. (2014). Analysis on Mediating Effect: Method and Model Development. Advances in Psychological Science, 22(5), 731.

Zhonglin Wen, Hongyun Liu, & Jietai Hou. (2012). Analysis on Regulating Effect and Mediating Effect. Education.

Zhou Xu. (2019). Research on Software Defect Prediction Methods in Different Scenarios (Doctoral dissertation, Wuhan University).

Appendix:Questionnaire

Evaluation Questionnaire for the Application of Data Intelligence Technologies and the performance of Software Development and Delivery

Dear Madam/Sir,

Thank you for taking time out of your busy schedule to answer this questionnaire. This questionnaire aims to understand the application of data intelligence technologies and the software delivery performance. This questionnaire is for academic purposes only and your data will be kept confidential.

Thank you for your support and cooperation!

Part I: Basic information

1. Gender

A. Male

B. Female

2. Age

A. Less than 20 years old

B. 21 to 24 years old

C. 25 to 29 years old

D. 30 to 34 years old

E. 35 to 39 years old

F. 40 years old or above

3.Years of working

A. Less than 3 years

B. 3-5 years

C. 5-10 years

D. More than 10 years

Please express your agreement with the following statements based on your working experience (7 points: strongly agree -> 1 point: strongly disagree).

1. The following are some statements about your **personal situation**. Please score according to your actual situation:	Strongly disagree	Disagree	Very disagree	General	Less agree	Agree	Strongly agree
D1 Data intelligence technologies analyze a large amount of data.	1						7
D2 Data intelligence technologies analyze unstructured data.							
D3 Data intelligence technologies analyze fast moving data.							
D4 Data intelligence technologies integrates data from multiple sources into data warehouses or marts for easy access.							
D5 Data intelligence technologies combine external data with internal data to enhance high-value analysis on the business environment. (Data**)**							
T1 Data intelligence technologies explore and adopt parallel computing methods (such as Hadoop) for intelligent data processing.							
T2 Data intelligence technologies explore machine learning, and deep learning, and use different data visualization tools.							
T3 Data intelligence technologies explore and adopt cloud services to process data and perform analysis. (Technology)							
P1 Data intelligence technologies can explore the correlation between core business indicators of each department.							
P2 Data intelligence technologies can analyze data from multiple sources and the results can be used to predict future							

1. The following are some statements about your **personal situation**. Please score according to your actual situation:	Strongly disagree	Disagree	Very disagree	General	Less agree	Agree	Strongly agree
trends. **P3** Data intelligence technologies can provide customers with easy-to-understand data forms, and thus provide customers with operable (business operation) consulting. **(**Predictive and dicision-support capability**)**							
TLE1 When developing project plans, our company explicitly sets a cost estimate for each project task.							
TLE2 When developing project plans, our company explicitly sets a time estimate for each project task. **(**Task level estimates**)**							
C1 In our company, members of organizations discuss important issues both formally and informally.							
C2 In our company, members of organizations attend cross-functional meetings together.							
C3 In our company, members of organizations coordinate the work cooperation among them.							
C4 In our company, information can be shared among members of organizations, so that decision makers and executors can access visual data resources (including data analysis, debugging, and application). **(**Coordination**)**							
TS1 Our company has a reasonable organizational structure.							
TS2 Our company's project personnel have clear responsibilities and division of labor.							
TS3 Our company's project personnel can match with their posts.							
TS4 Our company has a reasonable composition of project personnel. **(**Team structure**)**							
PC1 Our company can effectively							

1. The following are some statements about your **personal situation**. Please score according to your actual situation:	Strongly disagree	Disagree	Very disagree	General	Less agree	Agree	Strongly agree
control the project cost. **PC2** Our company can effectively control the project progress.							
PC3 Our company can effectively enforce auditability and control standards for projects.							
PC4 Our company can effectively control the project implementation. **(**Process control**)**							
PI1 Our company integrates the processes involved in the big data chain (i.e. data collection, preparation, analysis and decision-making).							
PI2 In our company, the integration of processes involved in the big data chain reduces the use cost of big data.							
PI3 In our company, the integration of processes involved in the big data chain reduces the workload needed to analyze big data. **(**Process integration**)**							
M1 In our company, reusable software modules are widely used to develop new analysis models.							
M2 In our company, object-oriented technologies can help customers create their own analysis applications.							
M3 In our company, object-oriented technologies can minimize the development time for new analysis applications.							
M4 In our company, applications can be adapted to meet requirements during task analysis. **(**Modularity**)**							
SP1 The company's market situation is improved.							

1. The following are some statements about your **personal situation**. Please score according to your actual situation:	Strongly disagree	Disagree	Very disagree	General	Less agree	Agree	Strongly agree
SP2 The company's sales is increased.							
SP3 The company's profit rate is improved. **(**Short-term Performance**)**							
PIB1 The company's image is improved.							
PIB2 The company's competitive advantage is strengthened due to improved software delivery .							
PIB3 The company's other business activities are running more smoothly.							
PIB4 The company improves its service to its customers.							
PIB5 The company establishes a good relationship with its business partners. **(**Perceived indirect benefits**)**							
SP1 The company's technical architecture can adapt to changes in business processes.							
SP2 The company's technical architecture can adapt to technological changes.							
SP3 The company can conceive and create applications faster than its competitors.							
SP4 The company can design and create more complex applications than its competitors. **(**Strategic performance**)**							