EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO

Tecnológico Nacional de México

Centro Nacional de Investigación y Desarrollo Tecnológico

Ph.D Thesis

Assessing and improving the maturity of engineering project management, by exploiting software traces

Presented by

MC. Jorge Francisco Ruiz Lopez

as a requirement for obtaining the degree of

Ph.D. in Computer Science

Thesis Director

Dr. Javier Ortiz Hernández

Thesis Co-Director

Dr. Eric Bonjour

Cuernavaca, Morelos, México. Agosto 2025.

cenidet

Centro Nacional de Investigación y Desarrollo Tecnológico

cenidet® Centro Nacional de Investigación y Desarrollo Tecnológico	**ACEPTACIÓN DE IMPRESIÓN DEL DOCUMENTO DE TESIS DOCTORAL**	**Código: CENIDET-AC-006-D20**
		Revisión: O
	Referencia a la Norma ISO 9001:2008 7.1, 7.2.1, 7.5.1, 7.6, 8.1, 8.2.4	**Página 1 de 1**

Cuernavaca, Mor., a 10 de junio de 2025

DR. CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO
P R E S E N T E

AT'n: DR. JUAN GABRIEL GONZÁLEZ SERNA
PRESIDENTE DEL CLAUSTRO DOCTORAL DEL
DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

Los abajo firmantes, miembros del Comité Tutorial del estudiante **M.C. Jorge Francisco Ruiz López** manifiestan que después de haber revisado el documento de tesis titulado **"Evaluación y Mejora de la Madurez de la Gestión de Proyectos de Ingeniería Basada en el Análisis de Trazas Informáticas"**, realizado bajo la dirección del **Dr. Javier Ortiz Hernández** y la codirección del **Dr. Eric Bonjour,** el trabajo se ACEPTA para proceder a su impresión.

A T E N T A M E N T E
Excelencia en Educación Tecnológica®
"Conocimiento y Tecnología al Servicio de México"

DR. JAVIER ORTIZ HERNÁNDEZ
TECNM/CENIDET

DR. ERIC BONJOUR
UNIVERSITÉ DE LORRAINE

DRA. MARÍA YASMÍN HERNÁNDEZ PÉREZ
TECNM/CENIDET

DRA. ALICIA MARTÍNEZ REBOLLAR
TECNM/CENIDET

DR. HUGO ESTRADA ESQUIVEL
TECNM/CENIDET

DR. JEAN-PIERRE MICAELLI
UNIVERSITÉ DE LYON III

c.c.p: C Verónica Sotelo Boyas/ Jefa del Departamento de Servicios Escolares
c.c.p: C Nóe Alejandro Castro Sánchez / Jefe del Departamento de Ciencias Computacionales
c.c.p: Expediente

Educación
Secretaría de Educación Pública

TECNOLÓGICO NACIONAL DE MÉXICO

Centro Nacional de Investigación y Desarrollo tecnológico
Subdirección Académica

Cuernavaca Mor, 11/junio/2025
Oficio No. SAC/137/2025
Asunto: Autorización de impresión de tesis

JORGE FRANCISCO RUIZ LOPEZ
CANDIDATO AL GRADO DE DOCTOR
EN CIENCIAS DE LA COMPUTACIÓN
P R E S E N T E

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado **"Evaluación y Mejora de la Madurez de la Gestión de Proyectos de Ingeniería Basada en el Análisis de Trazas Informáticas"**, ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

A T E N T A M E N T E
Excelencia en Educación Tecnológica®
"Conocimiento y Tecnología al Servicio de México"

CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO

c.c.p. Departamento de Ciencias Computacionales
Departamento de Servicios Escolares

CMAZ/lmz

Interior Internado Palmira S/N, Col. Palmira,
C. P. 62490, Cuernavaca, Morelos Tel. 01 (777) 3627770, ext. 4104,
e-mail: acad_cenidet@tecnm.mx tecnm.mx | cenidet.tecnm.mx

Dedication

To God, for His infinite mercy and guidance throughout my personal and professional life, which has allowed me to reach this point.

To my mother, Gina Lopez, who has been a true example of courage and strength in facing challenges despite adversity. Her conversations and advice have been a source of comfort and wisdom in difficult times.

To my father, Francisco Ruiz, who has shaped me with strong values, good habits, and compassion. His words, conversations, and above all, his example in life have given me strength during challenging moments.

To my beloved partner, Sylvie, whose support, understanding, and advice have helped me stay motivated throughout this journey.

Finally, I dedicate this thesis to the memory of my grandfather, whose example of faith has inspired me to remain steadfast in my devotion to God.

Acknowledgments

I would like to express my gratitude to all the people who have contributed during these four years of my thesis.

First of all, my deepest thanks to my thesis advisors, Dr. Javier Ortiz Hernández and Dr. Eric Bonjour, and my co-advisor, Prof. Jean-Pierre Micaelli, for their expertise, guidance, and the rigor they instilled in me throughout my research training.

To the members of my thesis review committee in Mexico: Dr. Alicia Martínez Rebollar, Dr. María Yasmín Hernández Pérez, and Dr. Hugo Estrada Esquivel, for their valuable insights, advice, and comments that helped improve this research work over the four years.

To the members of the follow-up committee in France: Dr. Armelle Brun and Dr. Laurent Geneste, for their expertise, advice, and comments that helped enhance the quality of this research throughout these four years.

I would particularly like to thank the Secretariat of Science, Humanities, Technology, and Innovation for funding this research over four years through a scholarship granted to me. I would also like to express my gratitude to the National Technological Institute of Mexico, CENIDET campus, and to the research laboratory in France, *Équipe de Recherche sur les Processus Innovatifs*, for providing the facilities to carry out my research activities.

My thanks also go to my colleagues who helped me with online presentations: Daza, Eduardo, Juan José, and Nohema.

I would also like to thank my colleagues in France: Brice, Fanny, Marie, John, Neal, Marine, Oussama, and Corentin, for their support and for making me feel welcome during the three years of my thesis. Also to my colleagues: Ignacio, Sebastian, Ziadany, Ally, Lorena, Catalina and Christian for the interesting discussions and conversations during the breaks.

These years of thesis work were a crucial period both personally and professionally. I also want to thank all the other people I met along the way who contributed something valuable, whether directly or indirectly involved.

Finally, I am infinitely grateful to my entire family, especially my parents, Francisco Ruiz and Gina Lopez, who gave me unconditional support throughout this journey. I would also like to give very special thanks to my girlfriend, for her patience, support, and advice over these four years. Also I want to thanks to my uncle Dr. Jaime Arau Roffiel, for the advices, talks, support, comments that helps me to deal with the research challenges.

Resumen de tesis

La tesis sostenida a lo largo de este trabajo de investigación es la capacidad de las ciencias computacionales para mejorar la gestión de proyectos, basándose en las trazas generadas por los actores involucrados en el proceso dentro de pequeñas y medianas empresas (PYMEs). Esta investigación doctoral en gestión de proyectos y ciencias computacionales se llevó a cabo en colaboración entre el Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) de México y la Université de Lorraine en Francia, en el marco del convenio No. 2021-02386 firmado entre ambas instituciones.

Esta investigación aborda un desafío crítico en el campo de la gestión de proyectos, particularmente en las PYMEs: la evaluación de la madurez en la gestión de proyectos. El problema industrial radica en que las expectativas asociadas a estas evaluaciones a menudo superan las capacidades y recursos disponibles en este tipo de organizaciones. Este desafío se agrava aún más por la creciente necesidad de certificación y la adopción de tecnologías de la información en los Sistemas de Gestión de Proyectos (SGP), es decir, sistemas que incluyen los procesos y prácticas desarrolladas para la gestión de proyectos. Frente a estas problemáticas, surge la necesidad de investigar soluciones que permitan a las PYMEs realizar evaluaciones de madurez de forma efectiva y alineadas con sus capacidades. El proceso de evaluación de madurez generalmente implica tres actores principales: el directivo superior o medio (director general, jefe de área, etc.), el auditor y el equipo del proyecto. Es imprescindible la participación de un experto en TI cuando se trabaja con SGP.

El principal problema de investigación que orienta este estudio es el siguiente: ¿Pueden utilizarse las trazas generadas por los SGP como datos, pruebas e indicadores confiables para evaluar la madurez en la gestión de proyectos? Para abordar esta cuestión, se propone un enfoque integral que combina una revisión sistemática de la literatura, una metodología analizada con teoría de grafos y validaciones empíricas en contextos específicos. Este enfoque integrado proporciona una base tanto para avances teóricos como para implementaciones prácticas en el ámbito de la evaluación de la madurez en la gestión de proyectos. Las tres principales contribuciones de esta investigación pueden describirse de la siguiente manera:

La **primera contribución** consiste en una revisión sistemática de la literatura, cuyo objetivo es identificar y analizar los modelos de madurez en gestión de proyectos publicados. Esta revisión sirve como base para el desarrollo de un marco conceptual fundamentado en redes de trazas, con un enfoque en actividades de ingeniería que apoyan los sistemas de software. El marco ofrece una perspectiva novedosa sobre cómo aprovechar las trazas para inferir niveles de madurez en las prácticas de gestión de proyectos. La originalidad de esta contribución radica en su enfoque innovador para vincular la trazabilidad con los modelos de madurez, una conexión que ha sido escasamente explorada en investigaciones previas. La validez académica de este trabajo está respaldada por publicaciones que destacan la relevancia y aplicabilidad del marco en el dominio de la gestión de proyectos.

La **segunda contribución** se centra en el diseño y desarrollo de un proceso de transformación de trazas, que reduce la granularidad de las mismas para permitir su explotación efectiva en la evaluación de la madurez. Este proceso facilita la estructuración y análisis de la información generada por los SGP, mejorando la comprensión y utilización de las trazas. El proceso de transformación fue implementado en un prototipo de sistema de gestión de trazas capaz de recolectar, transformar y visualizar trazas, y fue validado empíricamente con datos reales de proyectos. La originalidad de esta contribución reside en su capacidad para abordar desafíos técnicos como la granularidad de las trazas y la falta de documentación en los sistemas basados en trazas (TBS, por sus siglas en inglés). Al superar estos retos, se abren nuevas vías para el procesamiento de datos en la gestión de proyectos. La validación empírica demostró la efectividad del prototipo para ofrecer información clave para la evaluación de la madurez, mostrando su aplicabilidad en escenarios reales. Además, esta contribución

representa un avance significativo en el campo de la ingeniería de software basada en trazas, al introducir herramientas prácticas para analizar e interpretar trazas de gestión de proyectos.

La **tercera contribución** propone un proceso para determinar si los sistemas de software poseen todas las funcionalidades necesarias para generar trazas y llevar a cabo una evaluación de madurez (estudio de factibilidad). Este proceso involucra dos aplicaciones de la teoría de grafos como herramienta de mapeo. En primer lugar, se emplea para mapear las funcionalidades del sistema y las mejores prácticas a evaluar. En segundo lugar, se utiliza para mapear las mejores prácticas reportadas en la literatura y aquellas descritas en un marco de referencia. La aplicación de la teoría de grafos permite generar un criterio formal de comparación considerando la complejidad de las redes generadas en el software (Gephi), responsable de producir el diagrama de red. Gracias a este criterio, es posible determinar la viabilidad de una herramienta para realizar un análisis de madurez basado en trazas. La originalidad de esta contribución radica en su integración de la teoría de grafos para el análisis funcional de sistemas basados en trazas, presentando una perspectiva innovadora que combina metodologías computacionales con aplicaciones prácticas.

Además de estas contribuciones, esta investigación profundiza en la intersección entre la gestión de proyectos y los sistemas basados en trazas (TBS), proporcionando un marco innovador para la evaluación de la madurez. Al aprovechar los datos de trazas generados por los sistemas de gestión de proyectos, este trabajo ofrece un enfoque novedoso que responde a la creciente demanda de toma de decisiones basada en datos en el ámbito de la gestión de proyectos. Las metodologías y herramientas propuestas no solo mejoran la comprensión de los niveles de madurez, sino que también ofrecen soluciones prácticas adaptadas a entornos con recursos limitados. Las implicaciones de esta investigación van más allá del ámbito inmediato de la gestión de proyectos. La integración de metodologías basadas en trazas y teoría de grafos abre nuevas posibilidades para aplicaciones interdisciplinarias, especialmente en campos que requieren sistemas robustos para el análisis de datos y el soporte a la toma de decisiones. Asimismo, los hallazgos de este estudio contribuyen al discurso actual sobre la adopción de tecnologías de la información (TI) en los procesos organizacionales, subrayando la necesidad de herramientas escalables y eficientes que cumplan con los estándares de la industria.

En resumen, esta investigación doctoral representa un avance significativo en el campo de la evaluación de la madurez en la gestión de proyectos. La combinación de revisiones sistemáticas de la literatura, procesos innovadores de transformación de trazas y modelos basados en teoría de grafos ofrece un marco integral que une conocimientos teóricos y aplicaciones prácticas. La originalidad de estas contribuciones radica no solo en su novedad metodológica, sino también en su aplicabilidad a los desafíos reales que enfrentan las PYMEs. Los resultados de este estudio contribuyen a la comprensión académica de los modelos de madurez y ofrecen soluciones viables para las PYMEs, especialmente en contextos con recursos limitados. Al abordar los desafíos críticos del uso de trazas y la evaluación de madurez, este trabajo establece las bases para futuras investigaciones y desarrollos en la gestión de proyectos y los sistemas basados en trazas, asegurando su relevancia e impacto en los años venideros.

Si bien los avances son significativos, las limitaciones incluyen la dependencia de la calidad de los datos generados por los SGP, la cual puede variar según la implementación y prácticas de cada organización. La capacidad para generalizar los resultados a diferentes contextos de PYMEs también es una limitación, ya que las características de cada empresa pueden influir en la efectividad de la evaluación. Además, los modelos basados en grafos y la transformación de trazas presentan desafíos técnicos que podrían dificultar su adopción en entornos con recursos limitados. Desde una perspectiva económica, las PYMEs pueden enfrentar barreras para adoptar nuevas tecnologías debido a restricciones presupuestarias. Finalmente, desde una perspectiva ética, surgen cuestiones relacionadas con la privacidad y confidencialidad de los datos en los SGP. A pesar de estas limitaciones, este trabajo sienta las bases para futuras investigaciones y aplicaciones prácticas en el campo. Los objetivos principales de esta investigación fueron resumidos en los párrafos anteriores

.

Thesis Abstract

The thesis asserted through this research work is the capacity of computational sciences to enhance project management based on traces generated by actors involved in the process within small and medium-sized enterprises (SMEs). This doctoral research in project management and computer science was conducted in collaboration between the National Center for Research and Technological Development in Mexico and the Université de Lorraine in France, under the framework of agreement No. 2021-02386 signed between the parties.

This research addresses a critical challenge in the field of project management, particularly in SMEs: the assessment of project management maturity. The industrial issue arises from the fact that the expectations associated with these assessments often exceed the capacities and resources available in such organizations. This challenge is further exacerbated by the growing need for certification and the adoption of information technologies within Project Management Systems (PMS), i.e. (system that includes the project management processes and practices developed)[1]. In light of these issues, there is a need to investigate solutions that enable SMEs to carry out maturity assessments effectively and in alignment with their capabilities. The routine assessment process involves in general three main actors: top or middle manager (CEO, Area Manager, etc.), the auditor and the project team. It is imperative to involve an IT expert when dealing with PMS.

The key research issue that guides this study is as follows: Could the traces generated by PMS be used as reliable data, proofs and indicators to assess project management maturity? To address this issue, a comprehensive approach will be proposed, combining a systematic literature review, a methodology analyzed with a graph theory, and empirical validations in specific contexts. This integrated approach provides a foundation for both theoretical advancements and practical implementations in the domain of project management maturity assessment. The three main contributions of this research can be outlined as follows:

The first contribution is constituted by a systematic literature review, which aims to identify and analyze published project management maturity models. This review forms the basis for the development of a conceptual framework grounded in trace networks, with a focus on engineering activities that support software systems. The framework provides a novel perspective on leveraging traces to infer maturity levels in project management practices. The originality of this contribution lies in its innovative approach to linking traceability with maturity models, a connection that has been scarcely explored in prior research. The scholarly validity of this work is substantiated by the publication of articles that underscore the relevance and applicability of the framework within the domain of project management.

The second contribution is focused on the design and development of a trace transformation process, which reduces the granularity of traces to enable their effective exploitation to assess the maturity. This process facilitates the structuring and analysis of information generated by PMS, enhancing the understanding and utilization of traces. The trace transformation process was implemented in a prototype trace management system capable of collecting, transforming, and visualizing traces, and was empirically validated using real project data. The contribution's originality lies in its ability to address technical challenges such as trace granularity and the lack of documentation in trace-based systems (TBS). By addressing these challenges, the contribution paves the way for new methods of

data processing in project management. Empirical validation demonstrated the prototype's effectiveness in providing key insights for maturity assessment, showcasing its applicability in real-world scenarios. Furthermore, this contribution significantly advances the field of trace-based software engineering by introducing practical tools for analyzing and interpreting project management traces.
The third contribution proposes a process to determine whether software systems possess all the required functionalities to generate traces and conduct a maturity assessment (feasibility study). This process involves two applications of graph theory as a mapping tool. Firstly, it is employed to map the system's functionalities and the best practices to be evaluated. Secondly, it is used to map the best practices reported in the literature and those described in a framework. The application of graph theory enables the generation of a formal comparison criterion by considering the complexity of the networks generated in the software (Gephi), which is responsible for producing the network diagram. Thanks to this criterion, it is possible to determine the feasibility of a tool for conducting a trace-based maturity analysis. The originality of this contribution lies in its integration of graph theory for the functional analysis of trace-based systems to assess maturity, presenting an innovative perspective that combines computational methodologies with practical applications.

In addition to these contributions, this research delves into the intersection of project management and TBS, providing an innovative framework for maturity assessment. By leveraging trace data generated by project management systems, this work offers a novel approach that aligns with the growing demands for data-driven decision-making in project management. The proposed methodologies and tools not only enhance the understanding of maturity levels and provide practical solutions tailored to environments with limited resources. The broader implications of this research extend beyond the immediate domain of project management. The integration of trace-based methodologies and graph theory opens new avenues for interdisciplinary applications, particularly in fields that require robust systems for data analysis and decision support. Furthermore, the findings of this study contribute to the ongoing discourse on the adoption of information technologies (ITs) in organizational processes, emphasizing the need for scalable and efficient tools to meet industry standards.

In summary, this doctoral research represents a significant advancement in the field of project management maturity assessment. The combination of systematic literature reviews, innovative trace transformation processes, and graph theory-based models provides a comprehensive framework that bridges theoretical insights and practical applications. The originality of these contributions lies not only in their methodological novelty but also in their applicability to real-world challenges faced by SMEs. The findings of this study contribute to the academic understanding of maturity models and offer actionable solutions for SMEs, particularly in contexts where resources are limited. By addressing the critical challenges of trace utilization and maturity evaluation, this work establishes the foundation for future studies and developments in both project management and trace-based systems, ensuring its relevance and impact in the years to come.

While the advancements are significant, the limitations include the dependence on data quality generated by PMS, which can vary depending on the implementation and practices of each organization. The ability to generalize the results to different SME contexts is also a limitation, as the characteristics of each company may influence the effectiveness of the assessment. Additionally, graph-based models and trace transformation present technical challenges that could hinder adoption in resource-constrained environments. From an economic perspective, SMEs may face barriers to adopting new technologies due to budget constraints. Finally, from a ethical perspective, issues arise regarding privacy and data confidentiality in PMS. Despite these limitations, this work lays the foundation for future research and practical applications in the field. The main objectives of this research were summarized in the previous paragraphs.

Content table Pag.

Figure List

Table List

List of abbreviations

TP2Ms: Traditional Process Maturity Models

P2M: Process Maturity Model

PMMM: Project Management Maturity Models

CP2Ms: Computer-based Process Maturity Models

CBCN: a Cone-Based Conceptual Network

PCS: preliminary conceptual structure

TBS: Trace-Based System

NIAM: Nijssen Information Analysis Method

KPIs: Key Performance Indicators :

CMM: Capability Maturity Model

SEI: Software Engineering Institute

PMBOK: Project Management Body of Knowledge

General Introduction

The thesis asserted through this research work is the capacity of computational sciences to enhance project management based on traces generated by actors involved in the process within small and medium-sized enterprises (SMEs). This doctoral research in project management and computer science was conducted in collaboration between the National Center for Research and Technological Development in Mexico and the Université de Lorraine in France, under the framework of agreement No. 2021-02386 signed between the parties. In a period of four years, Between 2021 and 2025, the first year was completed at a Mexican institution, followed by three years at a French institution.

This research addresses a critical challenge in the field of project management, particularly in SMEs: the assessment of project management maturity. The industrial issue arises from the fact that the expectations associated with these assessments often exceed the capacities and resources available in such organizations. This challenge is further exacerbated by the growing need for certification and the adoption of information technologies within Project Management Systems (PMS), i.e. (system that includes the project management processes and practices developed)[1]. In light of these issues, there is a need to investigate solutions that enable SMEs to carry out maturity assessments effectively and in alignment with their capabilities. The routine assessment process involves in general three main actors: top or middle manager (CEO, Area Manager, etc.), the auditor and the project team. It is imperative to involve an IT expert when dealing with PMS.

The key research issue that guides this study is as follows: Could the traces generated by PMS be used as reliable data, proofs and indicators to assess project management maturity? To address this issue, a comprehensive approach will be proposed, combining a systematic literature review, a methodology analyzed with a graph theory, and empirical validations in specific contexts. This integrated approach provides a foundation for both theoretical advancements and practical implementations in the domain of project management maturity assessment. The three main contributions of this research can be outlined as follows:

The first contribution is constituted by a systematic literature review, which aims to identify and analyze published project management maturity models. This review forms the basis for the development of a conceptual framework grounded in trace networks, with a focus on engineering activities that support software systems. The framework provides a novel perspective on leveraging traces to infer maturity levels in project management practices. The originality of this contribution lies in its innovative approach to linking traceability with maturity models, a connection that has been scarcely explored in prior research. The scholarly validity of this work is substantiated by the publication of articles that underscore the relevance and applicability of the framework within the domain of project management.

The second contribution is focused on the design and development of a trace transformation process, which reduces the granularity of traces to enable their effective exploitation to assess the maturity. This process facilitates the structuring and analysis of information generated by PMS, enhancing the understanding and utilization of traces. The trace transformation process was implemented in a prototype trace management system capable of collecting, transforming, and visualizing traces, and was empirically validated using real project data. The contribution's originality lies in its ability to address technical challenges such as trace granularity and the lack of documentation in trace-based systems (TBS). By addressing these challenges, the contribution paves the way for new methods of data processing in project management. Empirical validation demonstrated the prototype's effectiveness in providing key insights for maturity assessment, showcasing its applicability in real-world scenarios. Furthermore, this contribution significantly advances the field of trace-based software engineering by introducing practical tools for analyzing and interpreting project management traces.

The third contribution proposes a process to determine whether software systems possess all the required functionalities to generate traces and conduct a maturity assessment (feasibility study). This process involves two applications of graph theory as a mapping tool. Firstly, it is employed to map the system's functionalities and the best practices to be evaluated. Secondly, it is used to map the best practices reported in the literature and those described in a framework. The application of graph theory enables the generation of a formal comparison criterion by considering the complexity of the networks generated in the software (Gephi), which is responsible for producing the network diagram. Thanks to this criterion, it is possible to determine the feasibility of a tool for conducting a trace-based maturity analysis. The originality of this contribution lies in its integration of graph theory for the functional analysis of trace-based systems to assess maturity, presenting an innovative perspective that combines computational methodologies with practical applications.

In addition to these contributions, this research delves into the intersection of project management and TBS, providing an innovative framework for maturity assessment. By leveraging trace data generated by project management systems, this work offers a novel approach that aligns with the growing demands for data-driven decision-making in project management. The proposed methodologies and tools not only enhance the understanding of maturity levels and provide practical solutions tailored to environments with limited resources. The broader implications of this research extend beyond the immediate domain of project management. The integration of trace-based methodologies and graph theory opens new avenues for interdisciplinary applications, particularly in fields that require robust systems for data analysis and decision support. Furthermore, the findings of this study contribute to the ongoing discourse on the adoption of information technologies (ITs) in organizational processes, emphasizing the need for scalable and efficient tools to meet industry standards.

In summary, this doctoral research represents a significant advancement in the field of project management maturity assessment. The combination of systematic literature reviews, innovative trace transformation processes, and graph theory-based models provides a comprehensive framework that bridges theoretical insights and practical applications. The originality of these contributions lies not only in their methodological novelty but also in their applicability to real-world challenges faced by SMEs. The findings of this study contribute to the academic understanding of maturity models and offer actionable solutions for SMEs, particularly in contexts where resources are limited. By addressing the critical challenges of trace utilization and maturity evaluation, this work establishes the foundation for future studies and developments in both project management and trace-based systems, ensuring its relevance and impact in the years to come.

While the advancements are significant, the limitations include the dependence on data quality generated by PMS, which can vary depending on the implementation and practices of each organization. The ability to generalize the results to different SME contexts is also a limitation, as the characteristics of each company may influence the effectiveness of the assessment. Additionally, graph-based models and trace transformation present technical challenges that could hinder adoption in resource-constrained environments. From an economic perspective, SMEs may face barriers to adopting new technologies due to budget constraints. Finally, from a ethical perspective, issues arise regarding privacy and data confidentiality in PMS. Despite these limitations, this work lays the foundation for future research and practical applications in the field. The main objectives of this research were summarized in the previous paragraphs. We now proceed to provide a detailed explanation of the research context, problems, objectives, and the structure of the thesis document.

I.I Research Context

Projects are foundational to the value creation process for organizations, including SMEs. To enhance their organization and manage their projects, these entities frequently utilize computer tools that are widely available, including task planners, project management platforms, collaboration systems, and data repositories. Additionally, they often refer to various project management guides and standards, such as the project management book of knowledge (PMBOK®) Guide, which was developed by the Project Management Institute (PMI, https://www.pmi.org/). Project management experts posit that best practices enhance the chances of success for all types of projects, whether technical or non-technical, routine, or innovative, sequential, or agile, medium-sized or mega projects, etc. However, several studies [2], [3], [4] have identified challenges related to effectiveness and performance, particularly related to the lack of maturity in project management.

In the domain of project management the time, the cost, and the scope are regarded as control variables, which are to be monitored throughout all stages of the project. The specific stages a project undergoes are determined by its application area; however, they include initiation, development, and closure as overarching phases. Each stage involves specific tasks, some of which may lead to problems affecting the process and impact the control variables. Delays in the scheduled times for project activities impact project timelines. Cost overruns resulting from inadequate planning and management of product or service prices used in the project affect project costs. When issues arise that impact project time or cost, the manager may need to adjust the scope, leading to uncertainty about the project being currently managed. These issues represent just a few examples of the challenges and hurdles currently faced in project management.

In the context of project management, computational tools and methodologies have emerged as instrumental mechanisms for enhancing outcomes. Generally, computational tools aim to facilitate the management of multiple projects, with functionalities including task assignment, team workload balancing, cost management, risk management, task management, and others specific to the software. As for methodologies, they serve as guides for managing and organizing the process into execution stages. Maturity models are tools used to assess the maturity level of companies, by comparing the current state with the ideal state, either for a specific area or the entire company. These models are used to evaluate project management, and the type of model used depends on the project type (software or engineering). While these tools and methodologies were created with the sole purpose of improving project management, this doesn't apply universally to all companies, as their implementation typically requires substantial financial and human resources. SMEs generally do not invest in developing a specific project management area, and in some cases, the individuals responsible for project execution also have other duties within the organization.

SMEs account for between 96% and 99% of the total number of businesses in member countries of the Organization for Economic Cooperation and Development (OECD) [5]. In Mexico, according to figures from the national survey on productivity and competitiveness, over 4.1 million companies are considered SMEs. Despite this, nowadays, theoretical and computational tools to improve processes are not specifically tailored for SMEs, as they are generally focused on companies with significant human and financial resources. The industry acknowledges maturity models as a valuable tool for assessing the status of specific or general processes within the organization and determining if these processes have room for improvement. Despite the benefits of their implementation, maturity models present significant complexity because they lack a standardized manual for application and may use terminology that is not familiar to those implementing them. Initially, these models emerged as a tool for software development companies to help improve the process by evaluating interacting areas and increasing the likelihood of successful application development. On a global scale, there have been concerted national and international efforts to refine and adapt software maturity models. These efforts include ESSI in Europe, MoProSoft, and EvalProSoft in Mexico [6].

Despite the difficulty of implementation, software maturity models have led to the development of project management maturity assessment models. These models are implemented through surveys administered to personnel working in project management areas, and based on the results, the maturity level of the area is determined. It's important to note that, in some cases, the surveys are designed based on selected maturity models and input from experts in the field, which further complicates the evaluation process. However, using surveys to assess the maturity of an organization or a process is not always effective because the individuals being evaluated may lack the basic knowledge to answer the questions clearly, resulting in a subjective assessment. Similarly, it was identified that in some cases, the persons being assessed via surveys left questions unanswered, which hindered the maturity assessment [7].

Therefore, it's easy to assume that SMEs not only encounter summarized problems related to resources and clarity in maturity assessment models but also face challenges regarding basic knowledge about best practices in project management. In [8], It is recognized that most models are developed for large enterprises and only partially considered small and medium-sized businesses. With the present research, the aim is to establish a tool that allows engineering SMEs to evaluate their maturity based on best practices and improve their project management process. The research context provides an understanding of the challenges and issues faced by project management and maturity models during their implementation. The following section describes the problems that were identified.

I.II Research Problem

This section examines two primary problems associated with project management maturity models. These issues are analyzed to highlight their implications for practical implementation and overall effectiveness.

I.II.I Lack of resources

Currently, there are two types of organizations: those referred to as "large" and those as "small-medium," each with a specific structure and set of financial and human resources. Large enterprises often engage private sector firms to conduct maturity assessments, which entails a significant investment of financial resources and time. On the other hand, small and medium-sized enterprises (SMEs) can also conduct these assessments through contracted services from the private sector, but some of them may be constrained in terms of available resources [9], [10].

Indeed, maturity models for project management implemented in organizations often require a significant amount of financial and human resources for their proper execution [11]. The amount of these resources depends on the size and capabilities of the company where the evaluation is conducted, so it is crucial to have a detailed understanding of the financial and personnel capabilities before initiating the evaluation process. One of the processes that demand more resources in project management maturity models is the evaluation process, which utilizes surveys to collect data [6]. However, the implementation of surveys among unqualified personnel may result in incomplete or unsatisfactory responses, which affects the quality of the results obtained. Also, implementing surveys among unqualified personnel can result in incomplete or unsatisfactory responses, thus affecting the quality of the results obtained.

I.II.II Lack of knowledge/ concision

The lack of explicitness regarding best practices, the evaluation process, and evaluation questions are some of the elements that generate great complexity when conducting a maturity assessment. Those tasked with conducting the evaluation often face confusion about how to proceed, as while models describe best practices in each domain, such as project management, they do not detail the necessary process to fulfill those practices. This lack of clarity can significantly hinder the evaluation process and impede the attainment of accurate and useful results [12]. Although project management frameworks sometimes describe best practices, they may not always detail the optimal process for implementing these practices. For instance, the Gantt chart is considered a best practice, but its methodology for development is not explicitly outlined in the frameworks.

Furthermore, the formulation of questions to assess best practices is often limited. While some maturity models proposed by the scientific community include a list of questions in their appendices, these questions may not be very comprehensive. In contrast, the industry utilizes a simpler type of assessment called "Self-assessment," which is often paid and provides limited information for the evaluation process and associated questions. In some models, the evaluation process is not detailed; rather, only maturity levels and best practices are specified as criteria for categorizing a company at a particular level. Some authors emphasize the importance of having conceptual knowledge when conducting a maturity assessment to do so effectively [13], [14].

I.II.III Research question

Given the two limitations mentioned that project management maturity models face, the following research question was formulated as described in Figure 1:

> Could the traces generated by project management software be used as valid and reliable indicators to assess project management maturity?

Figure 1. Research question

In this section, the research problems and questions are addressed to highlight the challenges associated with the use of project management maturity models. The next section outlines the research objectives aimed at tackling these two problems.

I.III Research objectives

Considering the research question presented in Section 1.2, four key research objectives have been identified and categorized into two primary domains: Project Management and Computer Science. These domains reflect the interdisciplinary nature of the study, addressing both the practical and technical challenges associated with project management maturity models. The general research objective was defined in alignment with the thesis title, serving as a guiding framework for the activities and goals of the study. The General objective is described in Figure 2.

> Assess and enhance project management maturity in engineering through the analysis of software-generated traces.

Figure 2. General Research Objective

The specific objectives (so) within the Project Management domain focus on improving methodologies, frameworks, and processes, while those within the Computer Science domain emphasize technological advancements, data analysis, and tool development. A detailed breakdown of these specific objectives, organized by domain, is provided in Table 1.

Table 1. Specific objectives.

Project Management	*Computer Science*
S.O. 1: Generate a comparative analysis of the PMM Models	**S.O. 3**: Generate a conceptual model that represents related to Project Management, Trace-based systems, and maturity evaluation
S.O 2: Propose a process for exploiting computer traces to evaluate PMM	**S.O. 4**: Propose the specification of a project management software architecture adapted for maturity assessment.

The previous table divides the specific objectives of this thesis by domains, aiming to address the complexity of project management, the number of resources, and the knowledge required to apply maturity models. Specific objective one and two focus on considering project management as the basis for their realization. Specific objective 1 aims to characterize the proposals in the literature on project management maturity models, identifying their differences, advantages, and disadvantages, and establishing a solid proposal that truly addresses the current challenges in project management maturity models. Specific objective 2 to develop a process that enables the exploitation of software traces to evaluate the maturity of project management.

The goals three and four consider computer science as a fundamental basis for their development. Specific Objective 3 aims to create a conceptual guide for individuals interested in implementing project management maturity models and software traces, facilitating the implementation of these models. Finally, specific objective 4 seeks to propose a specification of architecture to build a project management software adapted for maturity assessment. The objective of this research has already been outlined in the previous section, providing a clear understanding of the focus and purpose of the study. In the following section, the structure of the thesis document is presented to guide the reader through its organization. This section offers a comprehensive overview of the chapters and their respective contents, illustrating how the research process is systematically structured. Each chapter addresses specific aspects of the research, beginning with the general introduction, followed by the first part (research positioning), the second part (contributions), and concluding with the general conclusions and perspectives. This structure ensures a logical and coherent flow throughout the thesis.

I.IV Thesis structure

The first part of this thesis delineates the scientific framework within which the present doctoral research is situated. Chapter 1 introduces the fundamental key concepts: project management, maturity models, and project management, among others. Chapter 2 reviews the literature related maturity models, project management maturity models, use of computational traces and trace transformation process.

The second part of the thesis is dedicated to the contributions, which are focused on addressing the problems identified in the state-of-the-art review. Chapter 3 presents a general model that describes the domains and their conceptual interrelations. This model is required when conducting a maturity assessment and determining the level of maturity in a domain, considering the user traces. Chapter 4 proposes a general trace transformation process aimed at reducing the granularity level of the computer traces obtained from a management activity software. The objective of this process is to enable the utilization of traces for the evaluation of an engineering activity's maturity. Chapter 5, proposes a process to analyze the management tool and determine its feasibility for performing a trace-based maturity assessment and propose an architecture adapted, in this case applied to OpenProject.

The third part corresponds to the general conclusions and perspectives. This section describes the resume conclusion of the thesis with the findings reached after achieving the objectives set at the beginning of the research. To illustrate this structure, Figure 3 was designed.

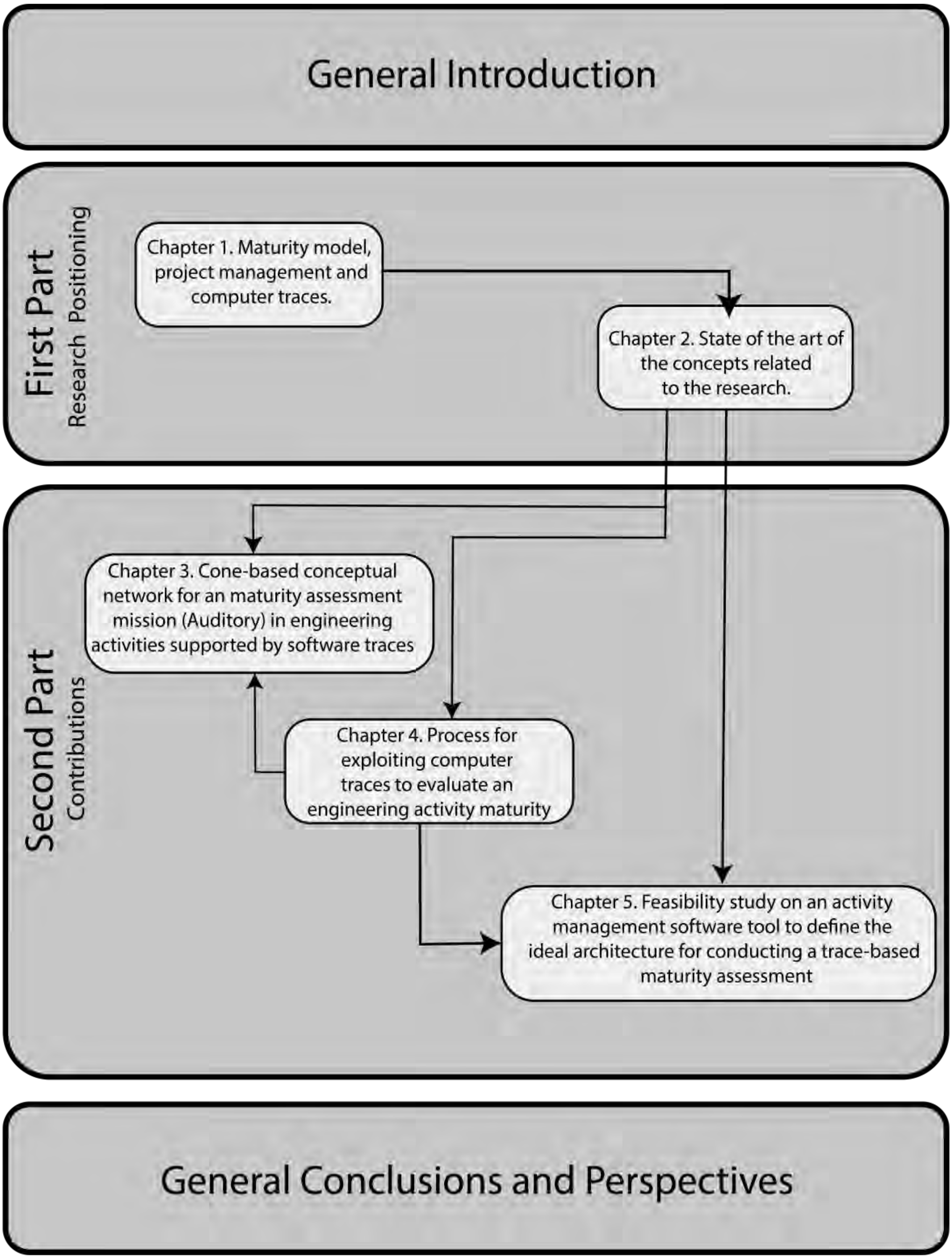


Figure 3. Thesis Structure

I.V Research summary

The present study addresses the issue that assessing project management maturity in small and medium-sized enterprises (SMEs) may exceed their capabilities and available resources. This issue gives rise to the following general research question: Could the data generated by project management systems be used as reliable indicators to assess project management maturity? In order to answer this question, a literature review is conducted focusing on two key questions: (1) What are the project management maturity models described in the scientific literature? and (2) What are the processes or methodologies for transforming traces as described in the scientific literature? These questions help establish a theoretical foundation that supports the development of the study. As a result, five main contributions are proposed.

First, a systematic review of project management maturity models is conducted (C1). This is followed by the development of a conceptual network based on a conoidal model to assess maturity in engineering activities (auditing), supported by software traces (C2). Additionally, a trace transformation process (C3) is defined to facilitate its application in maturity assessment. A feasibility study is then carried out on activity management tools to define the ideal architecture for trace-based maturity assessment (C4).Finally, an architecture specification is established (C5) to serve as a foundation for future implementations.

A variety of verification strategies are employed to validate the proposed contributions, including academic validations conducted through the publication of scientific articles, formal validations that utilize the NIAM method and the application of graph theory, instance validations performed by applying the models to the field of project management and tools such as OpenProject, and computational validation conducted by developing a prototype trace management system. The overarching objective of this study is to establish a conceptual and methodological framework for assessing project management maturity through the analysis of traces generated by management systems. This will contribute to a more objective and automated evaluation process for SMEs. Figure 4, delineates the issues, statement, main goal, understanding, contributions, and validations.

Finally, the following section describes the research positioning where the first two chapters of this thesis are addressed. On the one hand, the conceptual framework of the key concepts and on the other hand, the review of the state of the art of the research..

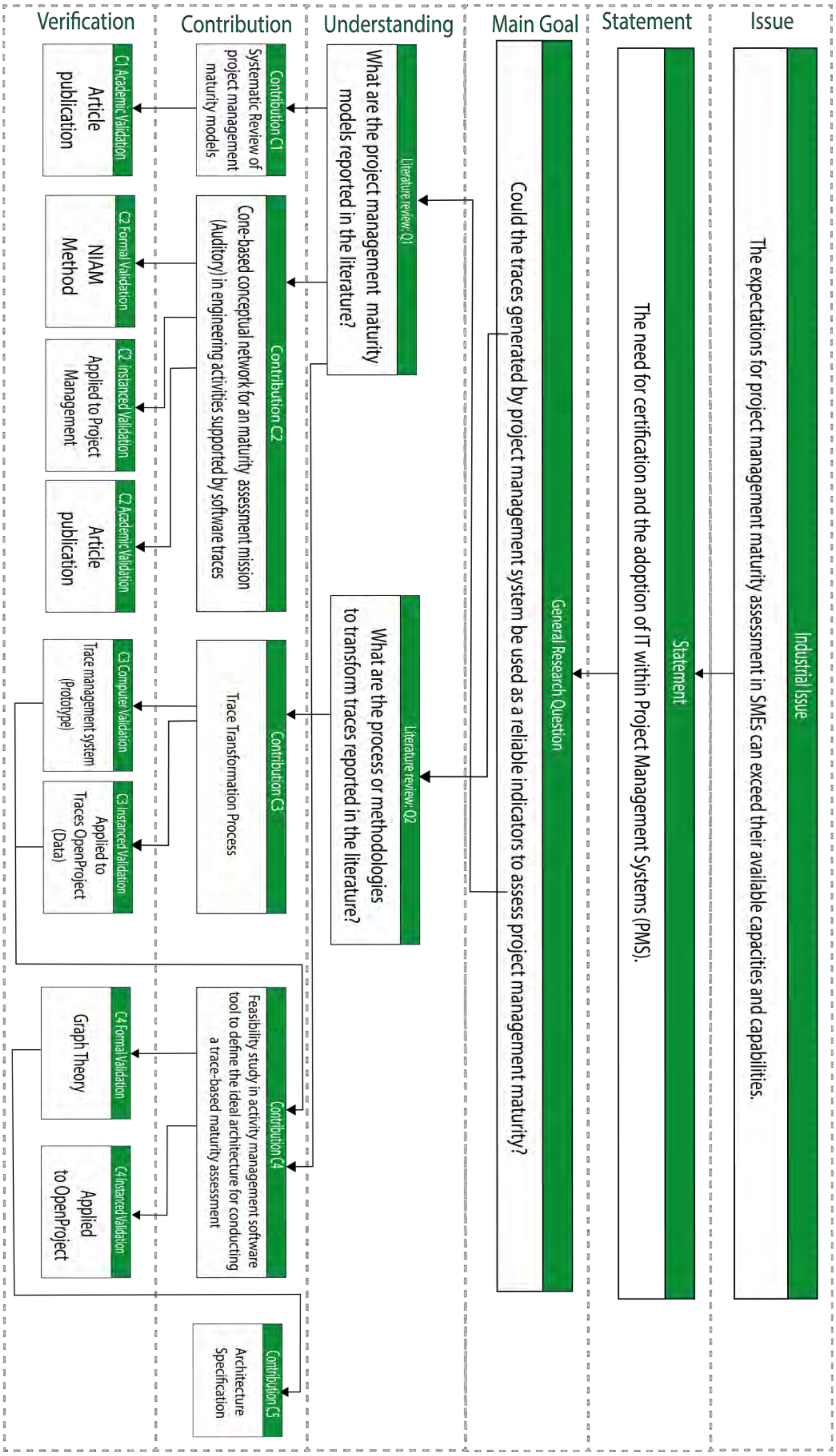


Figure 4. Research summary

Part 1: Research positioning

Introduction to part 1:

This first section of the thesis delineates the scientific positioning, thereby facilitating a conceptual understanding of the primary components underpinning the research on utilizing maturity models to enhance project management. In addition, it provides a comprehensive review of the current literature, with the objective of identifying extant research on maturity models, project management, and the application of traces. This section comprises two chapters: the first outlines the fundamental conceptual elements central to this study, while the second presents an in-depth state-of-the-art review on maturity models in project management and the use of IT traces. Moreover, complementary investigations were undertaken to further enrich the state of the art, encompassing trace transformation processes, the advantages of project management, and the tools employed in project management.

Chapter 1. Maturity model, project management and computer traces. Conceptual definitions

In the contemporary global context, organizations function within a sophisticated network of interconnected elements and relationships, necessitating a comprehensive understanding of these dynamics to leverage novel tools and theories aimed at enhancing performance and achieving objectives. This chapter aims to define and delineate the conceptual framework addressed in the research. Maturity models are a concept of considerable relevance across various industrial sectors, particularly within the scientific community. The interdisciplinary nature of their application poses a challenge in providing precise definitions. Project management has been successfully implemented in these two sectors, with methodologies, tools, and models undergoing different adaptations across domains. However, international efforts have been made to standardize project management, resulting in guidelines and standards being regularly updated to address current needs. Lastly, computational traces have gained significance due to significant advancements in the field of informatics and the vast amount of data generated within organizations. These traces play a crucial role in extracting knowledge and information to enhance organizational performance. Therefore, it is essential to define, delineate, and explicitly state these fundamental concepts to establish a succinct and robust foundation. The chapter is organized as follow: In Section 1.1, the key concepts of the research PMMM, project management, and computer trace are described. In Section 1.2, a brief overview of project management and one of the principal open-source tools used to manage project activities is provided. Also a complementary sections: 1.3 Chapter 1 Summary and 1.4 Chapter 1 Discussion are addressed in the end of this chapter.

1.1 Principal definitions: Maturity models, project management, and computer traces

1.1.1 Project Management Maturity Models

In general terms, a maturity model can be defined as a collection of best practices that assist organizations in improving their processes[15]. According to [16], in 1986, the Software Engineering Institute began the development of a process maturity framework aimed at helping improve its software process. The first maturity model was published in 1988 [17] and called the Capability Maturity Model (CMM) by the Software Engineering Institute at Carnegie Mellon University in the United States, funded by the U.S. Department of Defense. The goal of this model was to assess the quality and capability of software companies providing services to the U.S. Department of Defense.

Due to its great utility, the models have been adopted by other domains such as processes, organizational management, software development, human resources, quality, project management, product development, and supply chain. In general, organizations recognize models as useful tools for assessing the status of specific or general processes. They allow for determining whether processes have room for improvement. Maturity models present significant complexity because they lack a standardized manual for their application and may involve terminological biases that are not commonly used by those implementing them [18]. Models also require a considerable number of resources and time to be implemented properly and obtain a meaningful evaluation [11].

In general, maturity models describe five or six maturity levels [19]. They typically outline a set of best practices that help position the company and determine its level of maturity. The definition of the best practices is provided by the domain referential in which the model is applied. For example, in project management, the PMBOK referential defines them as *a general agreement that the correct application of skills, tools, and techniques can increase the chances of success for a wide variety of different projects*. Then, if we remove the word 'project management,' we would get the following definition.: *a general agreement that the correct application of skills, tools, and techniques can increase the chances of reaching the organizational goals.* Although determining the level at which companies are may seem evident, selecting the optimal level is often a challenging task.

General structure of PMMM

The implementation of maturity models is typically characterized by a three-phase process: namely, preparation, execution, and result analysis. This process is intended to facilitate continuous improvements. In the preparatory phase, the individual tasked with the evaluation selects the model to be utilized, taking into consideration the available resources, the stakeholders involved, and the requisite questionnaires. Subsequently, in the execution phase, interaction with the stakeholders occurs through interviews based on the previously designed questionnaire.

The responses obtained are then compared to ascertain the current maturity level. The result analysis phase is where the company's maturity level is established, and the necessary actions to improve this level are identified. This cyclical process concludes with a new evaluation to measure the impact of the implemented improvements. Additionally, various questions associated with each stage are addressed throughout the process, as shown in Figure 5.

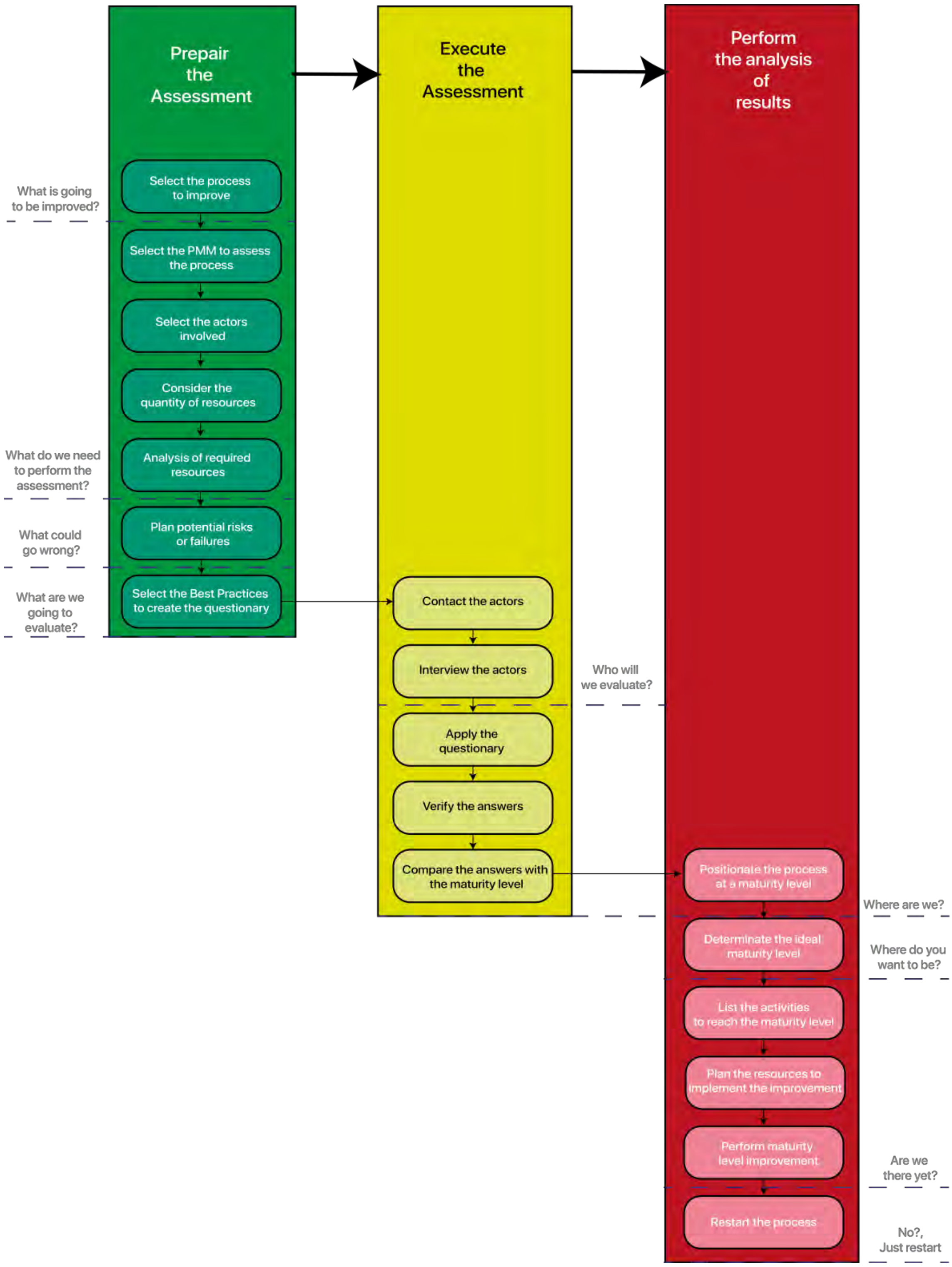


Figure 5. Traditional PMMM general structure and example of process

1.1.2 Project Management

Project management is defined as the application of knowledge, skills, tools, and techniques to project activities to meet project requirements [20]. The knowledge required to carry out a project will depend on the application area where the project is being undertaken; similarly, the skills, tools, and techniques will be specific. However, certain generalities do not depend on the application domain and are considered guidelines or recommendations for project management, particularly those identified in the Project Management Body of Knowledge (PMBOK) proposed by the Project Management Institute (PMI).

Various types of projects are recognized within both the industrial and scientific communities. However, to refine the focus of this research, engineering and software projects were selected. While both categories progress along a temporal continuum through a sequence of activities, each necessitates distinct management approaches. In engineering projects, adherence to original requirements may prove pivotal, particularly concerning safety certification. Conversely, in software projects, prioritizing the assessment of process quality at each stage and the flexibility to adjust requirements are paramount for ensuring the success of the developed software, typically following an agile methodology.

Best Practices of Project Management

The subset of project management fundamentals is generally recognized as best practices, and, on the other hand, best practices are defined as: "a general agreement that the correct application of skills, tools, and techniques can increase the chances of success for a wide variety of different projects" [21]. In other words, best practices are those activities suggested by a guideline or standard, whose correct application increases the likelihood of project success.

1.1.3 Computer Traces

A computer trace or log is a sequence of events that may have various relationships with different entities within the system[22]. These events represent the interactions between a user and the system. Entities, on the other hand, are objects that exist within the environment and are modified by an event within the system. In this research, a computational trace is a record or footprint generated using software. Specifically, in project management software, such traces are generated when managers and project leaders utilize the module of this software, and the information is stored in its database.

Trace-Based System

It's a system designed to collect, process, and visualize traces generated from the interaction between a user and a computer system. [23]. This system is commonly used for the exploitation of computational traces for a specific purpose. During the collection phase, the system extracts raw traces from the system, which typically exhibit a high degree of granularity (detail). In the transformation phase, the system reduces the level of granularity and enables the recognition of user actions within the traces. Finally, in the visualization phase, the system can generate a view of all the actions performed by each user.

A TBS is a technological concept referring to an intermediary system between raw data captured and knowledge about a computer-based activity. Several criteria can be used to classify instances of TBS. TBSs are either modules embedded in existing software or stand-alone applications processing data, often offline. They focus on a specific process, or they may concern any of them. TBSs are applications with varying technological readiness levels, even though most systems in the literature are at the laboratory stage. Furthermore, TBSs can be approached from two points of view, data-oriented vs. activity-oriented. One stream of the literature is focused on the content and the characteristic of the activity to trace whereas the second one seeks to reveal the most completed forms or patterns of raw data possible, as data analysts do in their day-to-day work. In both cases, we end up with a triangle linking raw data, TBS, and knowledge of the activity of interest, but the first peak studied is different. To simplify matters, we suggest distinguishing between three types of TBSs: those targeting routine processes with exploitation features (routine TBSs); those concerning exploratory processes, e.g. learning (open TBSs), and those aiming at knowledge generalization (expanding TBSs).

Routine TBSs.

TracED is a tool facilitating the delineation of a particular engineering domain and capturing the dynamics of product development, mostly software [24]. TracED is considered mature enough for experimental use, and integration with existing computer-aided design tools is suggested to improve its usability. In the field of user assistance, the kernel for the Trace-Based System (kTBS) uses the Resource Description Framework (RDF) to provide an environment for trace modeling, where data captured during user-software interaction can be represented in a structured format [25]. In the field of contextual recommendations, the web application Wanaclip is concerned by interaction-traces to recommend actions to users creating video clips. The importance of evaluating accuracy, responsiveness, and user satisfaction is also highlighted by the designers of this mature TBS [26].

Open TBSs.

The Trace-Based Indicator Management System (TB-IMS) and the Multi-Agent System for Computing Human Learning Indicators (IC-MAS) are TBSs implemented to compute human learning activities considering modeled traces [27]. The importance of traces lies in their ability to provide knowledge about learning processes, curricula customization, understanding, and improvement of learning activities. In the field of computing, SGBTm (Modelized Traces Management System) is a TBS devoted to the construction of knowledge in the case of learning; it integrates a collaborative ontology [28]. For its part, Rainbow is a TBS prototyped by the Computer Supported Collaborative Learning (CSCL) community to analyze teaching and learning interactions where pairs debate open-ended questions [29]. Rainbow is useful for retrospectively and contextually coding and analyzing debates using videos, computer traces, and chat dialogues, then heterogeneous data.

Expanding TBSs.

In the field of knowledge exploration, the Transmute system is presented, designed to visualize, transform, and assist in the interpretation of traces, including Knowledge Extraction from Data (KED) functionalities for data preprocessing, mining, and post-processing. Its architecture includes various components, such as Samotraces for customizable trace visualization and DisKit for implementing KED process, making it a versatile TBS. Interactivity in Transmute allows it to analyze the effects of users' actions and select the most relevant mining results, with the possibility of iterating the process with new constraints [23], [25]. DISKIT allows the formulation of various queries on the data, even in educational contexts. Additionally, FINEPIO is an approach to integrate context into trace analysis, although its implementation is still under development [30].

Relevant points for the research were found, including approaches to computer traces, systems based on computer traces, and trace-based recommendation systems. Current approaches to traces encompass collaborative activity, design process, learning process, computing process, and knowledge exploration. However, TBSs technological readiness is quite low. One expects TBSs' in existing software even though most systems in the literature are proofs of concept at the laboratory stage.

Trace Transformation Process

The trace transformation process comprises a series of activities aimed at reducing the granularity level of raw traces to a point where user actions can be observed without additional information. Achieving granularity reduction necessitates knowledge of functionalities, user types, permissions, and architecture to generate an ontology that accurately translates raw traces into user actions.

Type of traces

In this research work we have considered interaction traces, which are generated by a user interacting with a software. However, there are also other types of traces, such as calculated traces, which are derived from non-computer resources, such as video interviews or the capture of a user's activities through recordings. On the other hand, enriched traces, which result from the transformation of traces through a usage model, are described in [31].

1.2 Referential and tools

1.2.1 PMBOK

Since its inception in 1996, the Project Management Book of Knowledge (PMBOK) has evolved significantly to become an essential reference point for the project management community. The first six editions focused on the interaction between process groups and knowledge areas, delineating the key stages of project management and the competency domains necessary for project managers. These areas cover fundamental aspects such as integration, scope, time, cost, quality, human resources, communications, risks, procurement, and stakeholder management. With the 2021 edition, the structure of the PMBOK underwent a radical transformation, shifting its focus to the skills and activities of the project team leader. Principles and performance domains in project management were introduced instead of process groups and knowledge areas. This transition reflects a trend towards a more competency-focused approach, recognizing the crucial importance of the project leader's role in project success.

Furthermore, across the different versions, changes have been observed in the criteria for including material in the PMBOK, with the introduction of the term "Best Practice" from version 1 onwards. Although fundamental concepts such as project definition and project management have remained constant, adjustments and refinements have been made over time. Similarly, project management skills have evolved, with a more detailed and comprehensive description starting from version 5 of the PMBOK. This ongoing evolution reflects the PMBOK's adaptation to changing needs and growing maturity in the field of project management.

1.2.2 Project Management Software Tools

Project management software tools play a crucial role in orchestrating project activities, providing project managers and teams with clarity on objectives, deadlines, and scopes for each project activity. There are different types of project management tools: Open-Source tools and paid applications. Paid tools are typically used by large organizations that require maximum security and support capacity in case of system failures. In this research, a search was conducted on the website https://project-management.zone/, which allows a comparison of both types of tools. The objective was to select the best tool for this research, focusing on Open-Source tools that allow access to user logs within the system, as the research centers on these logs, called traces. The top-ranked tool, rated number 1, was OpenProject.

OpenProject

OpenProject is a web application that can be installed either in the cloud or locally via a server. This application is open source, released under version 3 (GPLv3) of the General Public License. OpenProject comprises 20 modules and approximately 39 APIs that generate user logs with each activity performed within the system. The modules group a set of system functionalities responsible for project management. On the other hand, the APIs are small applications that enable these activities within the modules. Generally, OpenProject consists of a Frontend and a Backend. Figure 6 illustrates the architecture of the application.

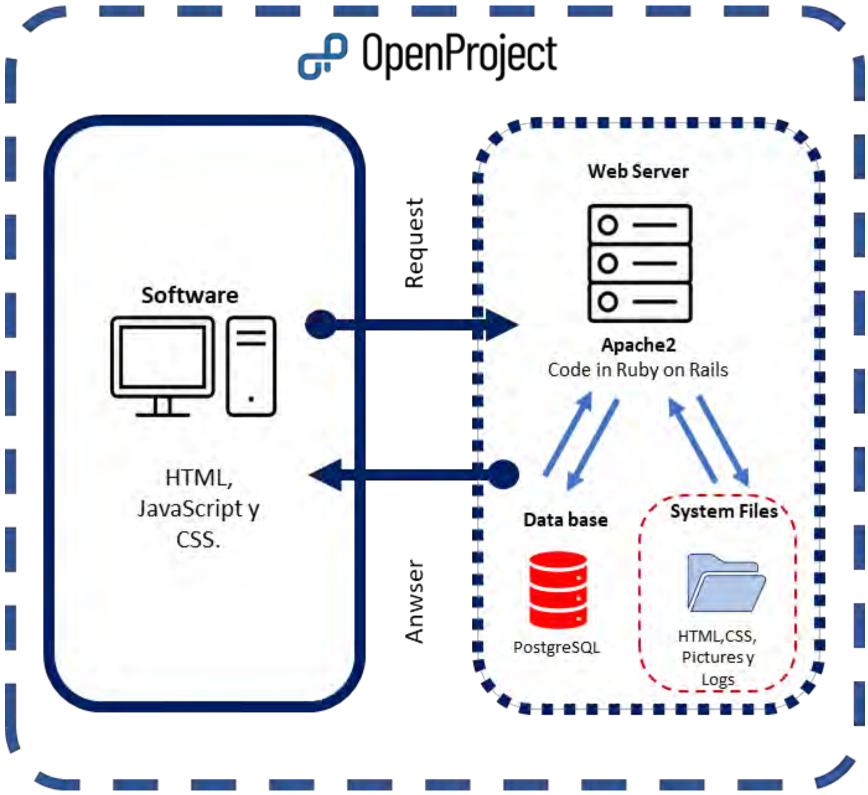


Figure 6. General OpenProject Structure

The elements described in Figure 6, show the frontend which is programmed in HTML, JavaScript and CSS. It also shows the WEB server which is responsible for containing the web application which stores data in a database and generates user traces. Additionally, it was identified that in the system files are stored all user traces. OpenProject is managed considering user permissions to access system functionalities. Currently, these user permissions are distributed by default among the following user profiles: project admin, staff and project manager, reader, and member. The project admin profile is responsible for managing a project, program, or portfolio, allowing the creation, modification, and deletion of a project. The project manager and staff profile can check the status of a project and request information, create, and modify it if necessary. The reader and member profile can only view the project they are involved in. In any case, the software provides the opportunity to generate new user profiles. Figure 7 describes the modules, rights, and user profile types.

Module	Global			Project									
User/ Rights	Create a project	Create and edit users	Create, edit, and delete placeholder users	Edit project	Select project modules	Manage members	View members	Manage versions	Select types	Select custom fields	Create subprojects	Copy projects	Manage overview
Project admin				√	√	√	√	√	√	√	√	√	√
Staff and project manager	√	√	√										
Reader													
Member							√						

Module	Work Package																		
User/ Rights	View work packages	Add work packages	Edit work packages	Move work packages	Add notes	Edit notes	Edit own notes	Manage work package categories	Export work packages	Delete work packages	Manage work packages relations	Manage work packages hierarchies	Manage public views	Save views	View watchers list	Add watchers	Delete watchers	Assign versions	Become Assigne/Responsible
Project admin	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Staff and project manager																			
Reader	√				√		√							√					
Member	√	√	√	√	√		√		√		√	√	√	√	√	√	√	√	√

Module	Boards		BackLogs				Budgets		Calendar		Documents		File storages		
User/ Rights	View boards	Manageboards	View Master backlog	View Taskboard	Select done statuses	Update sprints	View budgets	Edit budgets	View calendars	Manage calendars	View documents	Manage documents	View file links	Manage file links	Manage file storage
Project admin	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Staff and project manager															
Reader	√		√	√					√		√		√		
Member	√		√	√			√		√		√	√	√	√	

Module	Forums						Github	Meetings											News	
User/ Rights	Manage forums	Post messages	Edit messages	Edit own messages	Delete messages	Delete own messages	Show Github Content	View meetings	Create meetings	Edit meetings	Delete meetings	Invite users to meeting	Manage agendas	Close agendas	Send review notifications for agendas	Send meeting agenda as calendar entry	Manage minutes	Send review notifications for minutes	Manage news	Comment news
Project admin	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Staff and project manager																				
Reader		√		√		√		√											√	
Member		√		√		√		√	√	√	√		√	√	√	√	√	√	√	

Module	Repository					Team Planner	
User/ Rights	Read-only access to repository (Browser and checkout)	Read/Write access to repository (commit)	Manage repository	View repository revisions in openproject	View commit author statistics	View team planner	Manage team planner
Project admin	√	√	√	√	√	√	√
Staff and project manager							
Reader	√			√		√	
Member	√	√		√	√	√	

Module	Time and Cost														
User/ Rights	View spent time	View own spent time	Log own time	Log time for others users	Manage project activities	View own hourly rate	View all hourly rates	Edit own hourly rates	Edit hourly rates	View cost rates	Book unit cost for oneself	Book unit cost	Edit own booked unit costs	Edit booked unit costs	View own booked costs
Project admin	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Staff and project manager															
Reader															
Member	√	√		√			√			√	√		√		√

Module	Wiki											
User/ Rights	View wiki	Lst attachements	Manage wiki	Manage wiki menu	Rename wiki pages	Change parent wiki page	Delete wiki pages	Export wiki pages	View wiki history	Edit wiki pages	Delete Attachements	Protect Wiki pages
Project admin	√	√	√	√	√	√	√	√	√	√	√	√
Staff and project manager												
Reader	√	√						√				
Member	√	√	√	√	√	√	√	√	√	√	√	√

Figure 7. User Profile-Rights OpenProject

1.3 Chapter 1 Summary

In this first chapter of the thesis, the key and relevant concepts that constitute the fundamental theoretical framework for understanding this research work are described. Although maturity models, project management, and trace-based systems have been defined independently, together they form a body of knowledge that enables the understanding of this thesis's objective: optimizing the organizational project management process through a systematic and data-driven approach (traces). Precision in evaluation, detailed analysis, and the implementation of best practices are key elements that allow organizations to continuously improve and achieve their objectives more efficiently and effectively. The integration of these three elements ensures that all areas work together to achieve continuous and sustained improvement in organizational performance and efficiency.

Furthermore, knowledge of the tools responsible for managing the process is indispensable for administering activities within the domain, in this case, projects. Therefore, this thesis has selected the open-source software tool called OpenProject. This tool not only offers robust functionality for project management but also, thanks to its open structure and extensive documentation, allows for a detailed analysis of the user-generated logs. This facilitates the identification of usage patterns and the evaluation of system performance, providing a solid foundation for implementing improvements based on concrete data. The choice of OpenProject is justified by its ability to generate detailed traces, its flexibility to adapt to different types of projects, and its potential to integrate into a systematic approach to organizational optimization.

In summary, this chapter establishes the theoretical and practical foundation necessary to understand how the combination of maturity models, project management, and trace analysis can lead to effective optimization of organizational processes. The selection of OpenProject as the study tool illustrates the practical application of these concepts, demonstrating how a data-driven approach can significantly enhance project management and, ultimately, organizational performance.

1.4 Chapter 1 Discussion

The conceptual framework established in this chapter highlights underscores the convergence of maturity models, project management methodologies, and trace-based systems as key concepts for optimizing maturity assessment process. While each of these concepts has been studied, their integration presents both opportunities and challenges that require deeper analysis. The following four key issues are explored below:

Issues Related to the Implementation of Maturity Models: One of the primary challenges associated with maturity models is the lack of a standardized approach to their application. As discussed in this chapter, these models often require significant resources, expertise, and time to yield meaningful assessments. Furthermore, terminological biases inherent to different industries can hinder the effective implementation of maturity models across various sectors. Despite these challenges, the structured evaluation provided by maturity models remains a valuable tool for organizations striving for continuous improvement.

Issues Related to the Shift in the versions of PMBOK Approach: Project management, as described, is a fundamental discipline that has evolved through standardized frameworks such as the PMBOK. However, the shift in the latest edition of the PMBOK from process groups and knowledge areas to principles and performance domains reflects a broader trend toward adaptability and competency-based management. This evolution aligns with the increasing complexity of projects, particularly in technology-driven fields where rapid changes demand agile methodologies and flexible management approaches. Consequently, integrating best practices from both traditional and agile project management is essential for organizations seeking to remain competitive in dynamic environments.

Issues Related to the Use of Computer Traces to Improve Project Management: As organizations generate vast amounts of data through project activities, leveraging these computer traces can provide valuable insights into workflow efficiency, bottlenecks, and process optimization opportunities. The classification of trace-based systems (TBS) into routine, open, and expanding categories further underscores the diverse applications of trace analysis across different contexts. However, as noted, the technological readiness of many of these systems remains relatively low, with most still in the experimental stage. The challenge lies in transitioning from conceptual models to fully integrated, practical solutions that can be widely adopted by organizations.

Issues Related to Project Management Tool Selection: The selection of OpenProject as the project management tool for this research further emphasizes the importance of trace analysis in optimizing project workflows. The open-source nature of OpenProject and its extensive logging capabilities provide a unique opportunity to systematically analyze user interactions. However, the success of implementing this tool depends on the ability to extract meaningful insights from raw data and translate them into actionable improvements. This highlights the need to develop robust methodologies or processes for trace transformation and analysis, ensuring that extracted information contributes to the overarching goal of enhancing organizational performance.
In conclusion the integration of maturity models, project management methodologies, and trace-based systems presents a promising approach to improving organizational efficiency. However, the practical application of these concepts requires addressing challenges related to standardization, adaptability, and technological readiness. The findings of this research will contribute to a deeper understanding of how data-driven project management strategies can enhance decision-making processes and drive continuous improvement in organizations. Future research should focus on refining the implementation methodologies for maturity models, advancing the technological capabilities of TBS, and further exploring the intersection of these elements in real-world project management scenarios.

Finally, The following Chapter 2 presents a review of the state of the art considered for the generation of the proposed solution presented in the introductory part of this thesis. Chapter 2 considers a review of the work previously done in the ERPI Lab, general maturity models, project management maturity models, approaches to the use of traces and trace transformation processes.

Chapter 2. State of the art of the research.

This Second chapter identifies two primary sources of information on maturity models, project management, and computational traces. On the one hand, the scientific community disseminates its research in indexed scientific journals, while on the other hand, organizations share their contributions through documents known as "White Papers." Often, both sources are synthesized in "Survey" articles, which compile and categorize publications into a single document. To conduct the state-of-the-art review, two methodologies were considered: the first methodology is a systematic review, which allows for the identification of the most relevant articles prior to their reading, using filtering criteria such as publication year, language, number of citations, and keywords. The second methodology is a classical review, which involves the reading of articles to determine their relevance to the research. The central purpose of this second chapter of the thesis is twofold: first, to analyze the current state of these concepts based on the theoretical framework established in the previous chapter, and second, to present additional research conducted. This analysis seeks to complement the existing literature by highlighting the differences between what is described in the literature and the specific contributions of this research in terms of maturity models, project management, and computational traces. The chapter is organized as a follow: In Section 2.1, previous work conducted in the ERPI laboratory on maturity models is described. The subsequent section, Section 2.2, encompasses the broader maturity models that have been identified in the extant literature. Section 2.3 delves into the project management maturity models that have been identified through a systematic review. Section 2.4 explores contemporary approaches to the use of computer traces, while the final section, Section 2.5, describes and analyzes the processes that enable the transformation of traces. Also, a complementary sections: 2.6 Chapter 2 Summary and 2.7 Chapter 2 Discussion are addressed in the end of this chapter.

2.1 Background in the laboratory ERPI

In the context of the background research, previous work on maturity models conducted at the University of Lorraine by Prof. Eric Bonjour and Dr. Felipe Sanchez was identified. In this Ph.D. work, Sanchez Garzon focused on the evaluation of maturity criteria in project management [32]. The objective of the thesis was to create an architecture of a project performance prediction system based on the evaluation of project management maturity criteria. The research includes an analysis of mission statements from Sopra Steria Consulting and a review of available literature on project management and maturity models. It was identified that current maturity models lack clarity regarding their application in companies. As a response to this lack of clarity, a proposal was made to standardize certain categories. These standardized categories were incorporated into a model known as the Invariant Based Maturity Model or IB2M. Furthermore, a proposed causal model demonstrates a meaningful relationship between project management maturity and the risk of cost overruns. It highlights how the maturity of the project management process influences the occurrence of cost overruns. This thesis work is relevant to the research as it aims to enhance the field of project management through the utilization of maturity assessment tools and by its methodological proposal to group and conceptualize the maturity assessment criteria in a clear and precise way.

Additionally, previous Ph.D. works in the laboratory ERPI related to maturity models and the use of computer traces were identified. Darwich focuses on formalizing the enterprise's best practices process, called the "experience business process," using traces recorded by computer tools during collaborative activities in product development studies. Her work includes a holistic approach to managing these processes, an ontology named "OntoProcess" to organize organizational aspects, and a trace-based system to construct the operational process[33] . On the other hand, Girodón proposes a framework for organizational evolution based on the operational management of knowledge and competencies. Organizational mechanisms for sharing and utilizing knowledge and competencies in the design environment are enhanced, proposing models, methods, and tools to improve the performance of actors and organizations. Four key elements are developed, including an enterprise modeling method based on the KROM conceptual model, a global framework for organizational evolution, an organizational approach for designing Multi-Agent Systems (MAS) called DOCK, and an assessment of the value of the implemented framework. This evaluation is based on the potential to integrate the evolution framework into a global steering approach to evolve the organization, as well as experiments conducted in the Engineering department of Essilor[34].

Gálvez-Manríquez addresses one of the central challenges in contemporary industrial policy: fostering innovation in small and medium-sized enterprises (SMEs). Galvez highlights the lack of a suitable evaluation framework for innovation capacity due to its complexity. In response, he proposes an innovative approach developed in the ERPI laboratory, which includes creating a new index, the Innovation Potential Index (PII). His thesis presents an optimization model that replicates this evaluation approach, generating a specific action plan to improve business innovation systems [35]. Galvez also considers several key factors, such as the company's particular situation, associated economic efforts, and the entrepreneur's objectives. His research offers three optimization strategies and simulates various scenarios to evaluate their effectiveness. Finally, Dethine proposes tailored tools and methods to assist industrial SMEs in their digital transformation process. By developing a digital maturity assessment model and a decision support method, Dethine addresses the lack of specific guidance for these enterprises. The assessment model evaluates organizational, human, and digital resources through maturity grids, while the decision support method helps prioritize improvement areas based on the assessment results. Tested with French industrial companies, these contributions offer practical support for defining and implementing digital transformation strategies [36]. A summary of the thesis works considered in this research is described in Table 2.

Table 2 Ph.D. Works related

Year	Autor	Title of the thesis	Proposal
2014	**Hind Darwich**	Agent-based organizational approach for knowledge synthesis and reuse in collaborative engineering	• Trace-Based System application • Ontology of the process
2015	**Julien Girodón**	Proposal of an approach to improve the performance of organizations through the operational management of their knowledge and skills.	• Characterization and assessment of the maturity of organizations in their knowledge • Skills management practices. ACKME Method
2015	**Daniel Gálvez-Manríquez**	Evaluation of the capacity to innovate: a self-assessment and monitoring approach supported by dynamic multicriteria analysis	• An optimization model that replicates the process of evaluating the capacity for innovation of the SMEs
2019	**Felipe Sanchez-Garzon**	Supporting the transformation of a company's project management by elaborating an invariant-based project management maturity model and a causal predictive model between maturity criteria and project performance	• Project Management Maturity Model (IBM2) • A causal model is proposed which demonstrates that the maturity criteria and project performance are so related
2023	**Benjamin Dethine**	Towards a decision support method to define the digital strategy of industrial SMEs	• Maturity model for measuring numerical transition capability

2.2 General maturity models

Although the origin of maturity models begins with CMM, they have been applied across various domains. The following models are examples of maturity models identified during a literature review focused on project management.

2.2.1 Process-focused maturity models

The Industrial Process Maturity Model (IPMM)

The IPMM integrates concepts from software industry models, including CMM, with notions from traditional organizational environments [37].. For evaluation, the model proposes a statistical analysis of survey responses to determine the maturity level. It employs ANOVA as the statistical analysis technique and a Pearson evaluation.

The SMES Process Maturity Model (SPMM)

The SPMM integrates the concepts described in ISO/IEC 15504-7 for maturity levels but adapts those outlined in ISO 12207:2008 to fit small development teams and SMEs [6]. It proposes six maturity levels (0-5) to assess the organization, where level 0 indicates a lack of effective implementation, and level 5 indicates that one or more processes have a capability level of five. The main elements considered in this model are outcomes, processes, activities, and process attributes.

2.2.2 Organizational management-focused maturity models

The Business Process Maturity Model (BPMM)

The BPMM was developed by co-authors of CMM and CMMI [38]. It follows the principles of the Process Maturity Framework [17]. This model consists of five maturity levels. Level 1 describes the business process as inconsistently developed and unable to predict outcomes. Conversely, level 5 describes a proactive and opportunistic organization capable of seeking innovative actions to bridge the gap between current and ideal capacity. In this model, each area of the organization presents a set of best practices indicating what should and should not be done. For data collection, the BPMM employs a survey administered to staff and managers. Finally, it evaluates whether the best practices are being implemented or not in each process area.

The Value-based Process Maturity Model (VPMM)

The VPMM model determines the maturity of the business process practices taking into consideration the value creation capacity [39].. This model helps to prioritize their business process to develop the capacity requirements to accomplish business values. The model describes five maturity levels, 23 KPAs, 52 specific goals, and five general goals. The KPAs (Key Process

Areas) are introduced within the maturity levels to assess the capability of each business process. The model seeks to evaluate the business process according to the KPAs. However, the model does not explicitly describe the evaluation process or the tool for collecting information.

2.2.3 Software development-focused maturity models

The Capability Maturity Model (CMM)

The CMM model was developed to improve, control, and measure the software development process. The model describes 5 levels of process maturity: initial, repeatable, defined, managed, and optimizing [17]. For maturity assessment, the model requires an internal evaluation through a questionnaire aimed at evaluating the best software development practices. The model enables positioning within a maturity level to focus on those items that facilitate reaching the next level of maturity.

2.2.4 Human resources-focused maturity models

The People Capability Maturity Model (P-CMM)

By using the P-CMMM model, organizations can transform undisciplined activity domains into disciplined activity domains to be able to predict results [40]. The model describes maturity levels from one to five, this last one is recognized as the maximum level that indicates that the organization is in a mature state where the cultural characteristic is excellence. Also, the model describes four maturity threads that are the areas with a significant role in managing and developing the workforce: a) developing individual capability, b) building workgroups and culture, c) motivating and managing performance, and last, d) shaping the workforce. To implement this model a certified lead assessor is required.

2.2.5 Quality-focused maturity models

The European Forum for Quality Management`s “Business Excellence” model

The business excellence model is considered as a generic model that allows to assess the quality management maturity [41]. The model considers nine dimensions that the authors call criteria that include 32 sub-criteria. The criteria are leadership, people, policy and strategy, partnerships and resources, processes, people results, customer results, society results, and key performance results. These are grouped in two dimensions: enablers (24 criteria) and results (eight criteria). Also, the model provides 173 examples of each sub-criteria that help to understand how to address and measure. The maturity assessment could be done by the organization as a self-assessment but requires nine steps divided into two parts (preparation and assessment). In some cases, the RADAR Measurement System of the EFQM approach to perform the analysis of the assessment.

2.3 Project Management Maturity Models

Given the abundance of articles related to project management maturity models, the use of a state-of-the-art review methodology known as systematic review, proposed by [42], was considered. This approach comprises three main stages: 1) review planning, 2) review implementation, and finally, 3) systematic review reporting. These reviews allow for the filtering of articles according to inclusion and exclusion criteria to determine their relevance before proceeding with their reading. Below is a description of each phase and the activities carried out in each of the stages.

2.3.1 Phase 1: Planning the review.

In this phase, the following aspects were identified and integrated as relevant for this precursor phase to the implementation of the review: rationale, approach, research question, and criteria.

Identification of the need for a systematic review

For the characterization of the justification, the particular interest in identifying the key aspects of the project management maturity models was considered to compare them and identify the areas of opportunity and the relevance of each model identified.

Development of the review protocol

A first version of the protocol was developed and analyzed to determine if it complied with two important aspects: completeness and consistency with the objectives of the review. The protocol was composed of the following parts: background, strategy for formulating the research question, strategy for selecting primary studies, selection criteria, strategy for establishing quality assessment criteria, data extraction strategy, synthesis strategy, dissemination strategy, and establishment of a schedule of activities.

2.3.2 Phase 2: Implementation of the review.

In this phase, the systematic review protocol is used to develop the tasks established in the methodology for this phase. Previously the protocol went through a process of refinement and revision.

Identification of the research

In this section, the research question is formulated to help to identify project management maturity models. For the formulation of this question, the great importance of engineering project management maturity models was taken into consideration. Considering this research, the question is defined: What are the project management maturity models reported in the literature?

Selection of primary studies

For the selection, the strategy designed in the systematic review protocol was followed. It consists of searching for publications using search strings in the search engines and repositories to which the National Center for Research and Technological Development (CENIDET) and the Université de Lorraine has access. The research strings are listed next: 1) Improvement and project and Management and engineering; 2) Evaluation and maturity and project and engineering; 3) *Evaluación and madurez and proyectos and ingeniería* (research string used in Spanish), 4) Maturity evaluation and Project management, 5) *Mejora and proyecto and gestión and ingeniería* (research string used in Spanish), 6) Maturity evaluation and project management and SMES, 7) Maturity evaluation and project management and systematic review. It was identified that the main keywords of the search string were as follows: Maturity and Project Management. Likewise, within this section, the strategy established in the protocol was applied to determine the search criteria. These criteria are title, year of publication, number of citations, and language. A total of 1423 articles were reviewed considering only the title and keywords, after this review only 78 articles were considered primary studies. The inclusion and exclusion criteria in this phase of the process were only the title and keywords.

Characterization of the quality of publications

At this point, the following information was extracted for each publication: name of the article, number of pages, focus, number of citations, year of publication, and keywords, in addition to checking the availability of the article. After characterization, inclusion, and exclusion criteria were applied to each publication, only 32 articles were recognized as relevant articles for this research.

Data extraction and synthesis

This step was carried out only based on the publications that passed the quality filters and were identified as relevant. In the synthesis task, each of the relevant publications was analyzed to identify project management maturity models, and a summary was produced because of this analysis.

2.3.3 Phase 3: Systematic review report

In this phase, a quantitative and qualitative analysis of the literature on project management maturity models was carried out. Both analyses are in the following.

The existence of many articles related to project management maturity models involves a large amount of analysis time and depends on the subjectivity of the reader to determine their relevance. The main objective of systematic reviews is to apply a methodology to identify relevant papers reported in the literature that answer one or more research questions. To answer these questions, it is necessary to perform a qualitative and quantitative analysis of the articles and identify areas of opportunity not reported in the literature.

Quantitative analysis

A) Publications by year

With the quantitative analysis of this section, it was possible to identify that in 2014 there were a total of seven relevant publications on project management maturity models, and in 2018 a total of 5 publications.

Figure. 8 shows a trend in the increase in the number of publications highlighting the importance of maturity models and project management for the scientific community and industry.

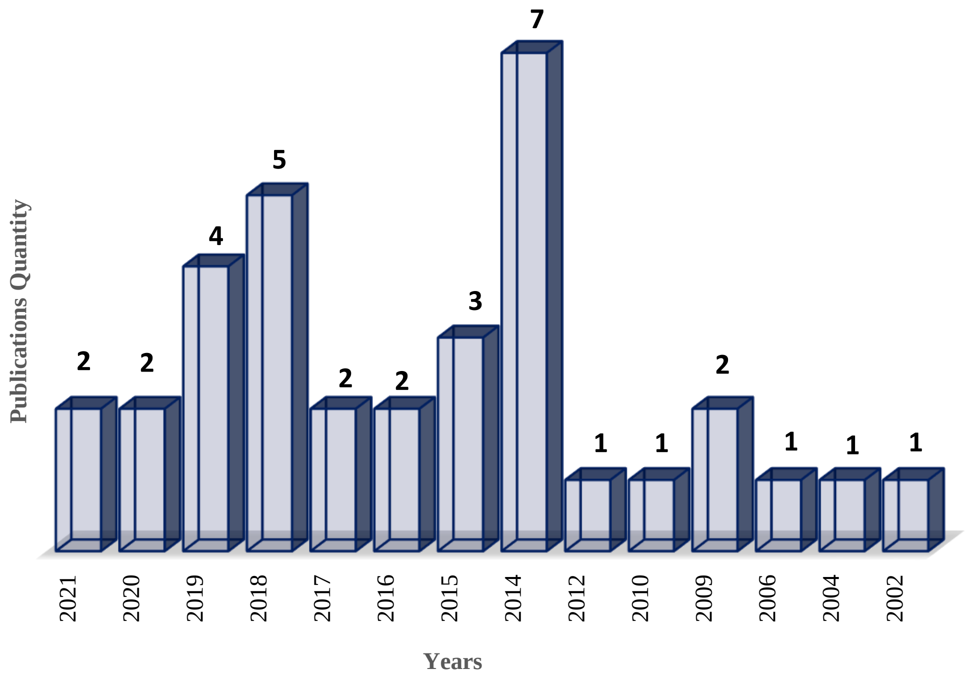


Figure. 8. Number of relevant publications on project management maturity models

B) Publications by search engine

Selecting the search engines was taken into consideration in the reports in the literature on digital libraries and indexing systems highly adopted by the scientific community. The most recognized search engines are Google Scholar and WorldWideScience, which is also highlighted in the scientific community for being the only search engine powered by Deep Technologies, a company dedicated to extracting scientific knowledge from the Deep Web. Figure 9, a pie chart shows the proportionality of the number of publications identified by title and keyword. In blue, the search engine Google Scholar with 61 publications; in orange the search engine WorldWideScience with five publications.

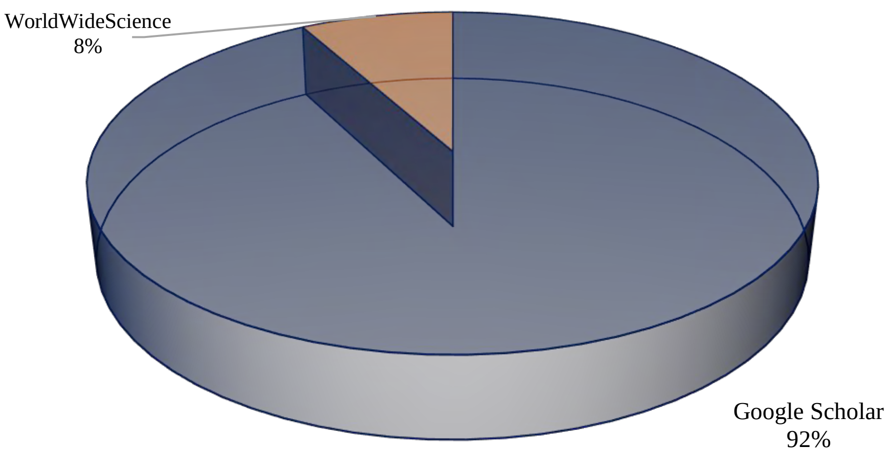


Figure 9. Distribution of publications of project management maturity models by a search engine.

C) Publications by repository

The repositories consulted for this research were ScienceDirect and ACM digital library. However, other repositories were also identified where at least one publication on maturity models and project management was found. Figure 10 shows the number of publications identified by the repository in a bar chart. The repositories with the highest number of publications identified were Science Direct and ACM digital library.

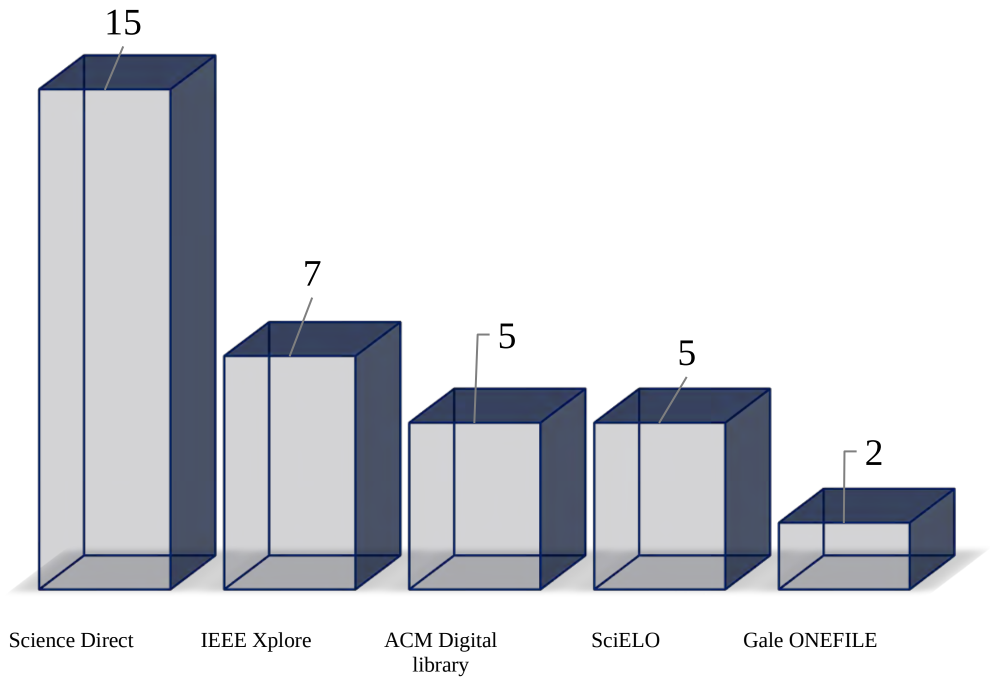


Figure 10. Number of publications of project management maturity models by repository

D) Most cited publications

The collection of the number of citations made it possible to identify the most relevant publications on project management maturity models. Figure. 11 describes the 10 most cited publications after applying the quality criteria to determine their relevance.

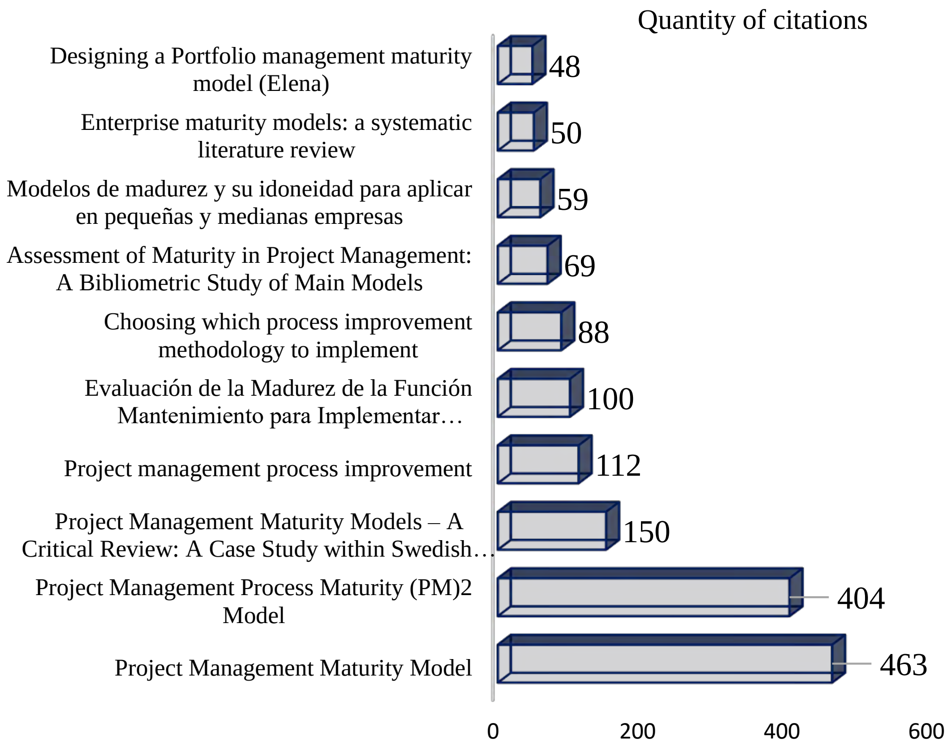


Figure. 11. Ranking of publications of project management maturity models.

E) Keywords

The analysis of the keywords of the publications on project management maturity models was useful to identify the current trends in this subject. Publications were identified that mixed the keywords: project management, maturity models, and software development, with which, it is easy to identify that at the beginning the maturity models were intended to evaluate in a general way to software development companies, which caused it to be implemented in project management of these types of companies. Consequently, due to their usefulness, the models began to be used in other fields of application such as engineering. Figure 12 describes the five most repeated keywords in the publications on project management maturity models.

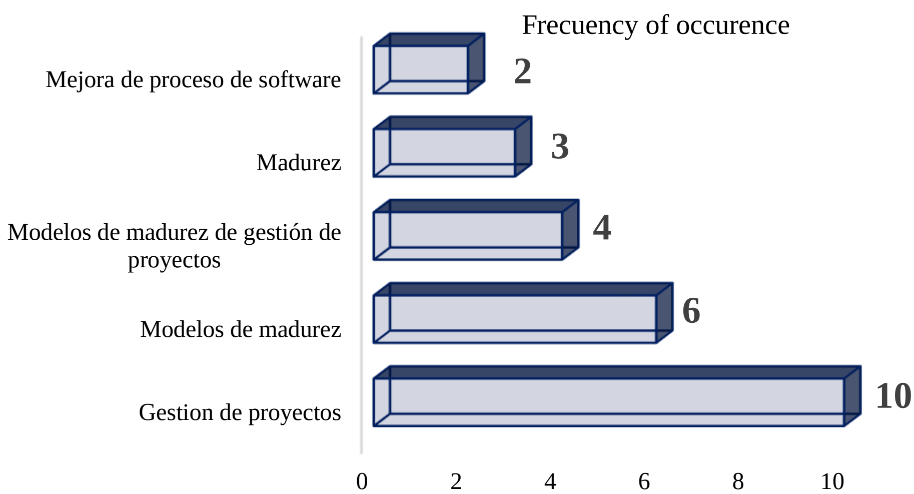


Figure 12. Frequency of occurrence of keywords in selected articles on project management maturity models.

F) Search Chains

The search strings were constructed based on the research question, to identify relevant publications on project management maturity models. Thanks to this, publications with great relevance were identified not only because they propose maturity models for the project area, but also because they present cases where such maturity models were implemented to evaluate the company's project management maturity level. Two search strings are recognized as the most effective since both individually generated a total of 19 articles potentially useful for the research. It should be noted that in this section we evaluated only the effectiveness of the search string in generating articles that contain in the title, abstract, or keywords some of the words contained in the string. Figure 13 shows the number of articles generated with each search string after applying the first quality filter.

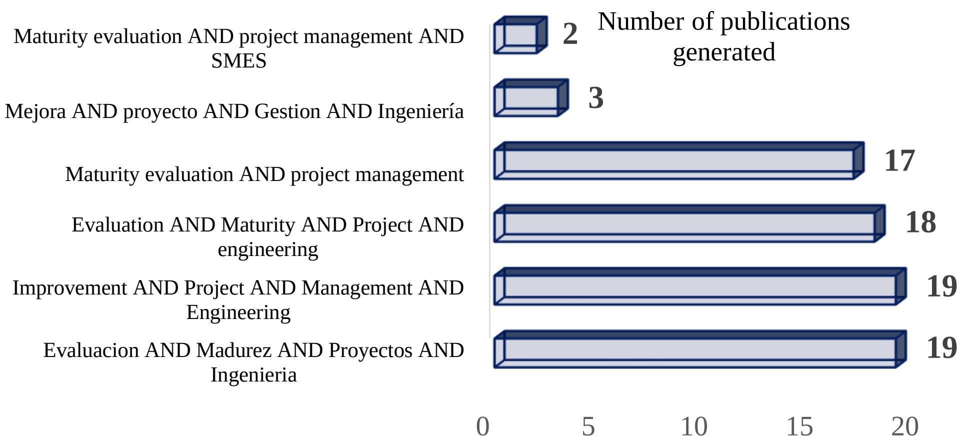


Figure 13. Effectiveness of search strings used in the systematic review.

Qualitative analysis

A) Keywords of publications

Figure 14 shows a cloud diagram of the keywords contained in the 32 articles identified as relevant. The words that were repeated the most are shown with a larger size, and those that were not so often repeated, but were key, are shown with a smaller size.

Figure 14. Keywords of publications.

B) Summary of publications

1) Elaboration of cross-training plans for key personnel in projects: the case of a multidisciplinary engineering company.

Gonzales et al. [43], propose a strategy to elaborate cross-training plans to implement them in Engineering companies that carry out multidisciplinary projects. For this purpose, empirical data were collected through a case study to analyze them and determine the activities and tasks that specialists perform as complementary activities. Likewise, this case study used the theoretical basis of some related disciplines (industrial engineering, project management, maturity models) to evaluate the maturity of the personnel and therefore of the areas where the study was developed using specific questionnaires.

2) Methodological proposal to measure the level of maturity of project management in engineering companies

A methodology is proposed in [7], to evaluate the maturity of project management in small Engineering companies, based on a generic standard which is adjusted considering the opinion of experts to establish a specific evaluation criterion and a rating scale to evaluate the best practices of project management. The methodology proposed is described as follows: a) Select a maturity model (CMMI, P3M3, and OPM3), b) Identify experts in project management, c) Domain selection, d) Best practices selection, e) Evaluation scale design, f) Test design and surveys g) Results analysis.

3) An application of ISO/IEC 15504 for the maturity-level assessment of SMEs and small development teams

The great investment (cost, time, and resources), the complexity of the recommendations, and the long-term return of the investment is considered a limitation for implementing the CMMI and ISO 15504 models in SMEs to evaluate their maturity. Considering this, an adaptation of the ISO 15504 model for maturity assessment in SMEs and small working groups is proposed in [6]. Likewise, the author recognizes other national and international efforts to narrow down and adapt software maturity models: ESSI in Europe, MoProSoft, and EvalProSoft in Mexico. In this adaptation, 6 maturity levels are established to classify organizations from levels 0 to 5. The authors highlight the importance of selecting the process level capability to determine the maturity level. This proposal is considered to face the limitations of CMMI and ISO 15504 to be applied in the SMEs.

4) Maturity models and their suitability for application in small and medium-sized companies

Poorly qualified personnel, lack of procedures for implementation, and a deficient organizational structure hinder the implementation of maturity models in SMEs mentioned in [44]. This lack of maturity models adapted to maturity is addressed in this article. The maturity models allow the assessment of the growth process, tracing clear strategies to implement the improvements to reach the organizational goals and guide the improvement program.

This article provides maturity models and standards comparisons to select the most suitable for assessing the maturity level of SMEs. The model PEMM was created to guide an improvement program based on the process. This model proposed by Hammer, considers two main characteristics of organizational efficient performance: the facilitators (design, executors, responsible, infrastructure, and indicators) and the capabilities (leadership, culture, past experiences, and governability). For the assessment process, the model contains a matrix elaborated with the characteristics of each maturity level.

5) Evaluation of the Maturity of the Maintenance Function for Implementing Innovations in its Management

In [45], a validated model implemented in Excel is presented to assess the maturity and the results of the maturity assessment of the maintenance area of a company. For the maturity analysis, the model proposed by Clarke and Garside is considered, which contains 5 aspects: commitment, culture, communication, tools, and methodologies, and, finally, conflict management. These aspects will be defined based on the area of application. Generally, the maturity assessment aims to identify the state of the organizations.

6) Maturity assessment models and Industry 4.0 readiness

A systematic review of the state of the art of maturity/readiness assessment models is presented in [46] to facilitate research on the subject, seeking to transmit knowledge to Mexico and Latin America. It also highlights that maturity models are tools used in the early stages of business transformation when companies seek to identify where they can make changes. In general, Industry 4.0 has encouraged organizations to motivate changes in the business through more technological environments that allow automating operations and blurring the boundary between virtual and real in the industry. The authors mention that maturity models are important for the organizational level and are useful for creating improvement paths within the organization. Similarly, the models are useful to cost reduction, quality improvement, and gaining a competitive advantage over other companies.

7) Project management maturity of a housing construction company: an analysis in Colombia.

The authors report in [47] the Maturity assessment of a Colombian construction company using three different maturity models: the Organizational Project Maturity Model (OPM3), the Colombian Project Management Maturity Model (CP3M V5.0), and the Construction Industry Macro Maturity Model (CIM3). The evaluation process was divided in three parts: in the first, the theoretical foundations of each of the models OPM3, CP3M V5.0 and CIM3 were reviewed; in the second, the company was characterized through a documentary survey of the project management processes in place at that time; and finally, in the third, the maturity models were applied in the following order: 1.- CP3M V5.0: The assessment with this model was carried out in 4 phases (planning, application, processing and analysis and presentation of results) and 8 sequential steps (process inventory, assessment programming, organization/project characterization, assessment of the level of formalization of practices, information processing, definition of the process capability level, definition of the maturity level, analysis and presentation of results); 2. - OPM3: This model contemplates 5 steps (preparation of the assessment, carrying out the assessment, planning of improvements, implementation of improvements and repetition of the process) and 3.- CIM3: This model is divided into 2 stages: weights of importance and capacity of the organization.

As a result of the assessment, it was determined that the company *Caso Construcciones S.A.S.* has a low maturity level of project management considering the maturity models (CP3M V5.0, OPM3, and CIM3) considered in this evaluation. There is an important lack of risk management practice. The authors identified a lack of resources to implement processes to analyze, plan, and control all the risk factors. It was considered an important factor in the success of the implementation of tools to project management is to create conscience in the CEO to destiny resources for the implementation.

8) Status of Project Management in SMEs in the Graphic Arts Sector and Proposed Guidelines for Improvement

In [9], the results of the diagnosis of the state of project management in 30 companies of the Colombian art sector considered MSMEs are presented. The OPM3 and P3M3 maturity models were used to assess maturity through personal interviews with project managers and production managers. In addition, the authors highlight the most specific characteristics of each model despite their similarities. For the evaluation, the authors followed the following methodology: 1.- Analysis for the identification of companies to be studied and evaluated, 2.- Definition of thematic categories as the central axis of project management maturity in organizations, 3. - Design of maturity scale based on the suitability of the organizations for the study, 4.- Construction of the complete hierarchical framework, 5.- Fieldwork (Surveys), 6.- Collection of information, 7.- Analysis of information (Main components, detailed and clusters).

Derived from the application of the proposed maturity assessment methodology to the set of companies, it was identified that in general in terms of project management, they do not reach level 3. The authors consider that for an SME to improve its level of maturity it must acquire a culture that involves innovation, strategic planning, and a long-term framed project approach.

9) Research on Application of Project Management Maturity Model in Risk Management of Electrical Engineering

A three-dimensional project risk management maturity model and a project risk management evaluation index system are presented in [48], which were applied to the private company H in Shanghai, China. For the generation of the model, the authors considered the research conducted on the PMMM, CMMI, P3M3, and OP3M models. The proposed model contemplates 3 dimensions: the first dimension is the key indicator of project risk management elements; the second is the project risk management process, and the third is the project risk management maturity level. This includes 5 levels of project risk management maturity: level 1: initial level, level 2: managed, level 3: standardized, level 4: specified, and level 5: optimized management.

On the other hand, the index system contemplates 5 steps for the establishment of the maturity evaluation index system: 1.- Summary of the factors that affect the risk management maturity of electrical engineering projects, 2.- Conducting surveys through questionnaires and statistics of the relevant personnel, 3.- Determination of the project management risk evaluation index, 4.- Determination of the weight correspondence index with each evaluation, and 5.- Establishment of the project evaluation system index.

As a result of the application of the project management maturity model in Company H in Shanghai, it was identified that project risk management and senior management support are relatively high. The organizational structure and corporate culture reach the standard level and it's possible to improve these to reach the next level. The low maturity level identified is related to the use of tools for project risk management and methods.

10) Research on the maturity of engineering project informatization management-As an example of Anhui San-Jian Engineering Co., LTD.

The analysis of the current situation and shortcomings of engineering project informatization management of Anhui Sanjian, Engineering Co., LTD is presented in [49]. This paper mainly analyzes the engineering project informatization from the management perspective and constructs a comprehensive evaluation of the engineering project informatization maturity management system based on the CMM maturity model. As a result of the analysis, the management index was identified as the factor that most affects the maturity of computerization in engineering projects, followed by the technical index and finally the effectiveness index.

11) Building Information Modeling Implementation through Maturity Evaluation and Critical Success Factors Management.

The problem of building information modeling implementation in architecture, engineering, and construction is addressed in [50]. The complexity of system implementation is also considered, and the recent introduction of BIM prevents organizations from basing their experience on recognized standards and procedures. For this reason, the development of a BIM implementation assistance model is proposed, which will help to better integrate and use BIM, considering the different maturity levels of each organization.

The authors recognize CMM as a provider of the maturity index through an organized assessment of BIM usage. The maturity assessment was divided into 11 criteria (data enrichment, life cycle views, roles, and disciplines, change management, business process, timeline/response, delivery method, information graphics, spatial capability, precise information, and interoperability) marked from 1 to 10, each rank describing a particular condition of the company for the evaluated criterion. As a result of this analysis, a model was developed to assess the maturity of the companies where BIM was implemented.

12) Project-based maturity assessment model for smart transformation in Taiwanese enterprises

The gap between maturity models and project management is addressed in [8], by designing an effective assessment framework for smart transformation, which adopts the smart industry readiness index. This index was created by the Singapore development council as a maturity assessment model to analyze the smart transformation of enterprises and formulate project management strategies. This model is useful for examining the maturity level of their transformation and assessing the scope for improvement of their strategies and implementation barriers, as well as being easy to use for SMEs. The authors mention that the existing models are designed for large enterprises and the SMEs only take it partially into consideration in the organizational dimension assessment.

The research focused on Taiwanese companies using data collected from 165 valid questionnaires and subjected to an aggregate analysis. For the evaluation process, the company used a project-based maturity model to perform a statistical analysis of the maturity of Taiwanese companies, identifying implementation obstacles and determining the key factors of the projects according to managers' opinions. The research flow involved the following activities: a) sampling and data collection, b) enterprise assessment with a project-based maturity model, c) data analysis to determine the difficulty of transformation projects, and d) industry survey and results. The project-based maturity assessment model comprises four blocks (organization intelligence, transformation project, and equipment intelligence and process intelligence) and 11 pillars (information exchange, management, demand and execution, resource allocation, required capability, storage and communication, robots self-monitoring, operation equipment, production and warehouse, supply chain and quality, service, and analysis). As a result of the maturity analysis, it was identified that most of the surveyed companies were immature or had a medium level of maturity for smart transformation and had transformation projects in 2021.

13) An Approach Based on Bayesian Network for Improving Project Management Maturity: An Application to Reduce Cost Overrun Risks in Engineering Projects

The development of a general framework and method for estimating the impact of project management maturity models (PMM) on project performance is presented in [51]. For this purpose, Bayesian networks are used to formalize the knowledge of project management experts and extract knowledge from databases of past projects. The data used for the research came from a company in the gas industry to which access was obtained. The research methodology includes the following 7 steps: 1.- Define a common framework to project management maturity assessment, 2.- Define how the input nodes will measure maturity, 3.- Define and classify the slippage factors, 4.- Define the aggregation of rules for the slippage factors, 5.- Select, clean and structure the database, 6.- Define the target node and 7.- Train and test the model.

To validate the model, the Sopra Steria gas company was applied in a case study and as a result, it was identified that the relationship between the increase in the maturity level and the reduction of the probability of incurring cost overruns can be modeled using a Bayesian network. It should be noted that, for the maturity assessment, the levels were modeled as a set of 5 output nodes (from level 1 to level 5) for each maturity criterion with two states YES or NO. In other words, it was evaluated if an activity complied with the established maturity level, and if this was the case, a maturity level was determined for each activity.

14) Project Management Maturity Models - A Critical Review: A Case Study within Swedish Engineering and Construction Organizations

The authors show an analysis of why and how to apply Project Management Maturity Models (PM3s) within the organization presented in [11]. This research has an exploratory nature since it aims to clarify and define the use of PM3s and why they are not applied more extensively. According to the authors, the application of PM3s requires a great amount of time and resources.

The present study considered selecting L.K.A.B., which is considered a large company due to the number of employees it employs and the number of projects per year. In the research, the following project management maturity assessment models are recognized: a) OPM3, b) SPICE, c) PM2, d) P3M3, and other models proposed by international organizations such as The

Baldridge National Quality Award (BNQA) and The European Forum for Quality Management`s "Business Excellence" model (EFQM).

15) New design for calculating Project Management Maturity (PMM)

The authors present in [52], a framework for calculating project management maturity within organizations, inspired by a literature review of existing models. The proposal is a new way to measure maturity, considering the existence or not of characteristics (function or capability) by dimension and area. The research methodology is divided into three important parts: 1) identification of the research objective, b) description of the framework, and c) selection of the reference model for the assessment.

The authors should note the four dimensions on which the evaluation should be performed: Methodology, Communication, IT resources, and management skills. Each dimension mentioned above contains the main functions and capabilities that should be measured. Once the maturity level has been identified, organizations can begin to improve their maturity through the dimensions and stop at the desired level according to market requirements. According to the authors, the model was built using data from market companies and for this reason, the proposed model is contemplated to be used exclusively within them.

16) *Project complexity as an influencing factor on the balance of costs and benefits in project management security modeling*

The question of what factors and parameters influence or determine the ideal maturity level for organizations is addressed in [53]. The question is addressed because academics and practitioners considered that when maturity models are used to assess a company it should reach the maximum level, since this may not be the optimal level for the organization. After all, the cost-benefit is not high. To try to answer this question, the authors conducted a quantitative analysis of case studies in industrial organizations in the energy and automotive sectors.

The analysis used the PjM3 model to interview managers and leaders of each organization to assess the maturity level of each process perspective (organization, control management, financial management, stakeholder engagement, risk management, organizational governance, resource management, and project management). In the results obtained, it was observed that the maturity of the organizations evaluated is homogeneous, with a difference of at most 2 points between the lowest and highest levels. Likewise, it was identified that according to the opinion of the managers and leaders of the results obtained, it is considered that each organization should have an optimum level, and the highest maturity level to which the organizations should aspire should be based on a previous evaluation.

17) Improving Project Management Performance through Capability Maturity Measurement

The results of a survey of 25 companies in the ICT sector in Slovakia that have a standardized project management improvement methodology are presented in [16]. Also, the authors analyzed the factors affecting performance and proposed a solution to mitigate the impact on projects and their results through maturity assessment. And finally, a systematic approach is suggested which allows continuous improvement of project management performance. As part of the research, detailed surveys were conducted with top managers of companies of all sizes in Slovakia to identify the level of maturity and its outcome.

Some relevant aspects of the methodologies and their project management processes were identified, in general, the managers have their process under control, however, they consider that there is an opportunity to improve it. On the other hand, according to the PMMM model, 42% of the organizations were identified as being at level 2; however, only 16% of the companies evaluated are seeking to improve their maturity level. In general, the companies have a standard project management methodology and are seeking to improve this process.

18) Project Management Process Maturity (PM)2 Model

The project management process maturity model is presented in [54], which allows organizations to position themselves above others in project management. The proposal considered project management models and practices reported in the literature to generate this model. This maturity model spread the project management process and practices in nine project management areas (Integration, Scope, Time, Cost, Quality, Communication, Human Resources, Risk, and Procurement) and five processes (Initiation, Planning, Executing, Controlling, and Closing) considering the PMBOK. The objective is to assess the project management practices and determine the weak practices that need to be improved.

This model follows a systematic approach to determine the level of the project management area within the organization and contemplates 5 maturity levels which are described in detail by the authors (1.-Basic project management process, 2.-Individual project planning, 3.-Systematic project planning, and control, 4.-multi-project integration planning and control, and 5.-Continuous improvement of the project management process).

19) A project management improvement program according to ISO/IEC 29110 and PMBOK®

The results obtained, and lessons learned from implementing the international standard for software process improvement ISO/IEC 29110 in four small software development companies, which are part of a process improvement program, are presented in [55]. The initial objective of the research was to help a group of software organizations in the definition of good practices that they should follow in their software development process and project management; however, for this research, the implementation of the project management process was considered the only objective, and the ISO/IEC 29110 standard was used for this purpose.

In terms of lessons learned and results, the following stand out: a) companies are focused day by day on solving the problems they face; therefore, they do not focus on defining processes b) standards and guidelines are unknown, however, the existence of the PMP certification provided by the PMI is recognized c) small companies need specific procedures that describe in detail the work to be done, with a clear set of best practices and a set of assets that help to carry them out. The PMBOK was used to complement ISO/IEC 29110 to provide a detailed understanding of the inputs, outputs, and best practices for each project management task.

20) Choosing which process improvement methodology to implement

This research describes in [56], how process improvement emerged from Total Quality Management (TQM) and how process improvement methodologies have evolved from it, which initially sought to improve quality and with the evolution also sought to increase the efficiency of the area where they are applied.

The author describes and lists the characteristics of some of the process improvement methodologies: Six Sigma, Lean Management, Lean Six Sigma, Agile Management, Re-engineering, Total Quality Management, Just-In-Time, Kaizen, Hoshin Planning, Poka-Yoke, Design of Experiments, and Process Excellence. Likewise, the author recognizes that all the methodologies described share the philosophy of process improvement as well as similarities in their implementation. Lean and Six Sigma are the best approaches developed to date, with satisfactory results.

21) Project management process improvement

The best practices of CMM (Capability Maturity Model) and PMBOK (Project Management Body of Knowledge) are presented in [57]. The *de facto* standards for process improvement and project management are collected to show project managers how to ensure that their project plans are not only perfect on paper but also in practice. Practitioners learn how to identify the underlying reasons why projects are not progressing as planned and discover the steps to create a solid process improvement program that leads to project success.

22) Project management maturity: an assessment of maturity for developing pilot plants

A study was conducted in South African mining and engineering plants in which the maturity of project management for developing pilot plants was assessed [58]. To assess the maturity, the PMSolutions model was used, and the Delphi technique was used for the surveys, which were applied to a panel of experts in the mining and engineering plants.

It should be noted that the project management evaluation consisted of evaluating the nine areas of knowledge proposed by the PMI: a) integration, b) scope, c) time, d) cost, e) quality, f) human resources, g) communication, h) risk, i) procedure and j) organizational maturity level. The average maturity level of the organizations is 2, however, compared to the level required for the organization, they are below the ideal by 1 level.

23) Assessment of Maturity in Project Management: A Bibliometric Study of Main Models

A bibliometric study of the main project management maturity models reported in the literature proposed by the scientific community is presented in [59]. The study considered the Web of Science, Scopus, and Scielo databases to perform the searches, and the year of the publications was limited to 2010 and 2014. The methodology used in this research included the following steps: a) definition of keywords, b) establishment of a Boolean search strategy, c) definition of criteria for article selection, d) database search, e) article selection, f) export of articles to Zotero software, g) reading of selected articles, h) data entry in Excel and j) descriptive analysis.

The main models identified were: CMMI, OPM3, KPMMM, PMMM, and MMGP. It should be noted that the CMMI model served as the basis for all the others and that the OPM3 model is the only one that presents 4 maturity levels instead of 5, this is because this model adopts the documentation and standardization process within level 1, which normally represents level 2. Among the results, the following points stand out: a) 81% of the articles were found in the Scopus database, b) the countries with the highest number of publications are China and Brazil, and c) the most studied model in academia is the CMM followed by OPM3, d) the OPM3, KPMMM, and PMMM models are the most used in China, e) the most repeated words in the articles analyzed were in the areas of IT, construction, airline, and engineering.

24) Enhancing Organizational Project Management Maturity: a framework based on the value focused thinking model

An organizational project management maturity model is presented in [60], which maintains the advantages of previous models seeking to fix the problems and improve them, in addition, the portfolio management evaluation model has 3 dimensions and offers four ways to evaluate. As well as a review of the literature, which provides information about the harsh criticisms and bad opinions about maturity models. In general, there is talk of inefficiency, ambiguity in understanding them, unestablished evaluation processes, subjectivity in the evaluations, etc. On the other hand, there are also claims of positive changes when implementing maturity models.

The proposed model considers the following parts for the maturity assessment process: 1.- Maturity assessment, 2.- Creating alternatives to improve maturity, 3.- Planning and monitoring the implementation of initiatives. Additionally, a case study is shown where the proposed framework was implemented in a Brazilian company. This produced some useful insights that aided the discussion on how managers could make better and more holistic decisions using the currently available PMMMs to maximize the return on their investments.

25) Designing a Portfolio management maturity model (Elena)

This article [61], provides some definitions of maturity models in general and their great relevance and impact on organizations and project management. Some relevant parts about maturity models are mentioned below: a) a model is a framework that describes an ideal progression towards the desired improvement using some scenarios or levels, b) maturity models are seen as models that reflect certain aspects of reality called capabilities and define qualitative attributes which are used to rank competency objectives within one of several defined areas, c) organizations with high maturity levels expect to be successful in terms of effectiveness and efficiency, d) the most mature organizations show an 85% reduction in defects and 75% in cost reduction.

The proposed model is distinguished by the following characteristics: a) having both a continuous and discrete (staged) approach to assessing maturity, b) adaptability, c) simple and practical constraints, d) assessing three dimensions of maturity including process, concepts, and organizational documents and e) general purpose and with the ability to be applied to all types of organizations and industries of different nature. In addition, this model presents 5 phases to manage the portfolio: 1.-Principles, 2.-Concepts, 3.- Processes, 4.- Tools, and 5.- Adaptation.

26) Project management maturity in the age of big data

The authors mention in [62], that project management maturity models lend two points of view both in practice and in research as to the object of assessment, the first is the process point of view which defines project management maturity as the degree of coding, measuring and controlling the process and on the other hand, the organizational point of view defines maturity as the responsiveness of organizations to project management and examines the integration of projects and organizational process. In addition, organizational maturity assessments are said to be conducted using structured data, which emphasizes the process rather than the organizational viewpoint. Recently, the understanding of maturity models has increased due to the advances in data collection and analysis, since now it is not only said what to measure but how to measure it.

It is also mentioned that thanks to big data's ability to evaluate unstructured data, it provides an opportunity to apply new analytical approaches to create deeper insights. The authors recognize that two analytical techniques have great potential to help in the task of assessing project management maturity, social network analysis (organizational coexistence between areas) and text analysis or text mining (this approach aims to assess in depth how the project is implemented within the organizations which involve the analysis of text including document and emails).

27) Maturity analysis in project management in Colombian universities

The authors describe in [63], that for the maturity assessment, the Kerzner maturity model (PMMM) was selected after an analysis of the models presented in the theoretical framework. It was also identified that it is a model focused on all types of organizations and that it consists of 5 maturity levels, which allows better implementation of the model to assess the maturity of the university's project management. The assessment process is described in four steps: a) maturity model selection, b) stakeholder management plan, c) maturity level assessment, and d) improvement strategies.

For the evaluation, the group of stakeholders involved in project management was taken into consideration, the selected stakeholders are 1.- dean, 2.- department directors, 3.- program coordinators, 4.- professors, 5.- faculty council, 6.- academic vice rectory, 7.- research vice rectory, 8.- social vice rectory, 9.- interaction, 10.- planning office and 11.- "*Colciencias*". For the evaluation of maturity, a 7-section questionnaire was adapted by Manuel Alvarez and applied to maturity, methodologies, tools, competencies, portfolio, multi-projects, and project owner. For the implementation of improvement strategies, the result of the maturity assessment, which is at level 2, was taken into consideration.

28) National project management maturity: A conceptual framework

The authors highlight in [64], the importance of good project management within organizations for the generation of value. The existence of two major project management organizations is recognized: PMI (U.S.A) and IPMA (Europe). According to Standish Group, two-thirds of all registered projects fail due to a lack of good management. PMI states that 12% of investments are spent unnecessarily due to poor project management performance. On the other hand, they also mention how the concept of project management has been used in different areas by different governments worldwide: Great Britain, Scotland, Germany, Stuttgart, and a few more. These examples of the application of project management within governments give rise to a question:

What can a country do to increase project management competence not only at the governmental level but throughout the national economy, thus contributing to the agility and national and international success of its national companies?

Derived from the question posed, the authors propose a framework that not only allows measuring the level at the organizational level but also at the national level; this model was entitled National Project Management Maturity (NPMM). The literature review identified the existence of different maturity models with different approaches and some divided opinions on the use of project management. Some of these opinions highlight the high complexity to make efficient assessments and the only tacit direction of project management knowledge, in other cases, the opinions focus on highlighting the great importance of the use of maturity models. The proposed NPM3 model considers 4 maturity levels: 1) Level 1: Birth, 2) Development, 3) Adolescent, and 4) Mature. The model also compares project KPIs across governments and identifies the most mature governments in project management.

29) Project management maturity model: the case in an automotive industry in Brazil

In [65] the introductory part, the authors highlight the importance of project management for organizations and mention some authors who refer to the benefits of a correct implementation within the organization. It is also recognized that every aspect of project management has two dimensions: 1) The technical dimension and 2) The human dimension. The technical dimension focuses on the groups of practices or processes that are integrated into project management; the human dimension focuses not only on the people who perform the project management process but also on their experience of the process. The need for maturity models arises from the need of organizations to identify weaknesses and measure the level of efficiency and effectiveness of project management methodologies. In the final part of the introductory section, some maturity models are described in a general way, and more specifically the Darci Prado model is described. This paper describes that the qualitative and quantitative analysis was performed through a survey developed by Darci Prado to identify the status of project management maturity of a company dedicated to the automotive industry located in Brazil. The results of the maturity assessment were: 2.59 points. This result compared to the general level of private companies established in Rio de Janeiro is low. Likewise, it was identified that the maturity of the organization's projects is not the same in all cases, the quality area was the one with the highest maturity level of 3.5, purchasing at 2.66, bus planning at 2.19, and van planning at 2.4.

30) Proposal and verification of project management methods and tools-oriented maturity model

In [66] the introductory part, the authors highlight that the success rate of projects is highly related to the maturity and appropriate use of tools and methods for project management. In addition, it is recognized that the models that help to measure the maturity of project management are called Project Management Maturity Models (PMMM) and are defined as tools that serve to evaluate the level of maturity and identify areas of opportunity within the organization. The most recognized international standards are PMI, IPMA, and PRINCE 2. This article aims to compare the current project management maturity models and generate a new, more user-friendly model using the tools and methods as evaluation criteria.

According to the literature, a total of 29 project management maturity models are recognized, with different theoretical bases (PMI, PRINCE 2, IPMA, own basis, and combination of these) and different approaches (Organizational or process-oriented). In addition to these, the authors are aware of 8 project management models with a specific approach. As part of the theoretical analysis for the generation of a new model, two analyses were performed: The primary analysis was related to what is reported in the literature on project management maturity models and the secondary analysis was a data analysis by research quality. In the secondary analysis, a group of representatives from the chemical industry was interviewed for 2 hours and grouped to generate an outline of their maturity model.

As a result of the analysis of the literature and the data generated by a group of experts (best-fit models), the methods and tools-oriented project management maturity model was generated. This model contemplates 4 prerequisites for its implementation: A) Project life cycle,

B) Division or stages of a project, C) Tools and methods adapted for project management for each stage of the project, and D) Organizational and software support. The maturity model contemplates a survey-type checklist applied to know the total number of tools used in each area of project management. According to this, it acquires a score that serves as a basis for the maturity assessment.

31) Project Management Maturity: Case study analysis using the OPM3® model in manufacturing industry

In [67], the introductory part, it is recognized that in the literature some authors acknowledge that higher levels of maturity are related to better costs and schedule results. Also, it is recognized that some investments in the project management area do not have a direct impact on the organization's profits and this is perceived in this way since they are not directly involved in the product development process which makes it less noticeable. The following project management maturity models are recognized: OPM3, P3M3, and The Berkeley Project management maturity model. These models are implemented through numerous surveys and may lack scientific validity, however, despite this, the most recognized is OPM3.

The authors recognize that maturity models are sometimes too general for industries that perform very specific projects, which causes a limitation in applying the models. Examples include metallurgy or machine manufacturing, which have specific processes. In these types of organizations, project management becomes a key factor in reducing costs, and increasing sales and product quality. It is also noted that only one of the total case study organizations recognized good project management practices. The basis for the maturity assessment was the OPM3 self-assessment model, converted to an Excel sheet.

C) General overview of project management maturity assessment

It was noted that maturity models emerged as a solution to enhance software development quality by systematizing processes and ensuring reproducibility within the field of application. Subsequently, their application extended to areas significantly impacting software development. Project management was acknowledged as pivotal, given that virtually every development process is framed as a project, possessing distinct initiation, execution, and closure phases.

Following a thorough review and analysis of numerous publications, 12 project management maturity models were identified (OPM3, P3M3, CP3M, CIM3, PMMM, KPM3, MMGP, PM2, NPM3, PM2TOM2, P2MM, and P2CMM). These models have become instrumental tools within project management, aiding in process improvement and the identification of both favorable and unfavorable practices within the domain, thereby gauging their alignment with ideal standards. Employing questionnaires as an evaluation method, these models determine the maturity level of project management based on their outcomes. Although some models feature 5 maturity levels while others include 4, the discrepancies among proposed levels are generally minimal.

However, the utilization of project management maturity models is primarily confined to large corporations due to their access to substantial financial and human resources, facilitating a more comprehensive evaluation of maturity levels. Consequently, small, and medium-sized enterprises (SMEs) encounter limitations in their adoption. Yet, it's crucial to recognize that SMEs represent a significant portion, comprising between 90 and 98% of the global business landscape. Given their substantial presence, SMEs play a vital role in various industries, often serving as suppliers to larger enterprises, thus underscoring the importance of achieving a robust project management maturity level within this sector.

To visualize the differences between the maturity models identified, a comparison in Table 3 was built. The criteria considered to compare the maturity models are: 1) Approach, 2) Domain, 3) Maturity levels, 4) Number of best practices, 5) Evaluation process, 6) Number of survey questions, 7) Dimensions, 10) Referential and 11) Generated by the industry/ scientific community.

These criteria are important to identify the important information for our approach. In the beginning, it was considered the tools used to obtain the data, but we identified that the survey was the only way to get the information on the process to be assessed. Also, it was identified that the maturity models are focused on traditional project management.

Table 3. Project Management Maturity Models reported in the literature.

Model	Approach	Domains	Maturity levels	Number of best practices	Evaluation process	Number of questions in the survey	Dimensions	Based on	Generated by
OPM3: Project Management Maturity Model	Project Management	Project Management Program Management Portfolio Management	Level 1: Ignorance Level 2: Medium Level 3: Maximum	600	Step 1: Preparation for the evaluation Step 2: Assessment Step 3: Improvement plan Step 4: Implement improvement Step 5: Repeat the process	150 (Self-assessment)	1.- Domains 2.- Process improvement steps	PMBOK	Industry
P3M3: Program and Project Management Maturity Model	Organizational performance of the project management area	Project Management Program Management Portfolio Management	Level 1: process knowledge Level 2: a repeatable process Level 3: defined process Level 4: managed process Level 5: optimized process	Not identified	Step 1: Planning the assessment Step 2: Selection of the P3M3 model to be used Step 3: Establishing the scope Step 4: Selecting the data structure Step 5: Selecting the approach Step 6: Planning the checklist Step 7: Understanding results Step 8: Improving planning	9((Self-assessment)	Not identified	Not identified	Industry
CP3M V5.0: Colombian Project Management Maturity Model	Project Management	Project Management Program Management Portfolio Management	Level 1: Inconsistency Level 2: Planning and Control Level 3: Integration Level 4: Strategic alignment Level 5: Innovation and optimization	Not identified	Step 1: Process Inventory Step 2: Evaluation scheduling Step 3: Organization/Project Characterization Step 4: Assessment of the level of formalization of practices Step 5: Information processing Step 6: Definition of process capability level Step 7: Maturity level definition Step 8: Analysis and presentation of results	The evaluation method is based on the criteria established for each maturity level	1.- PMBOK 2.- Strategic alignment 3.- Learning 4.- Adaptability 5.- Life Cycle	PMBOK	Scientific Community
CIM3: Construction Industry Macro Maturity Model	Construction project management	Project Management	Level 1: Immaturity (0) Level 2: Immaturity (1/3) Level 3: Traditional maturity (2/3) Level 4: Maturity (1)	They propose key practices that are generated to achieve the organizational objective. For this reason, maturity levels are established based on the existence of key practices.	Step 1: Establishing the importance of key practice areas Step 2: Organizational capability	The evaluation method considers the assessment of the existence of key practices	1.- Cost 2.- Quality 3.- Health and safety 4.- Human resources	CMMM	Industry
PMMM: PM Solutions Project Management Maturity Model	Project management process in companies	Project Management Program Management Portfolio Management	Level 1: Initial process Level 2: Structured and standardized process Level 3: Organizational standards and institutionalized processes Level 4: Process managed Level 5: Process optimized	Not identified	Step 1: Analyze the status of the company. Step 2: Characterization of the company's practices. Step 3: Analysis of surveys built based on PMMM Step 4: Determination of maturity level	52	1.- Scope management 2.- Time management 3.- Cost management	PMBOK	Industry / Scientific Community
KPM3: Kerzner Project Management Maturity Model	Project management process in companies	Project Management Program Management Portfolio Management	Level 1: Common language Level 2: Common process Level 3: Unique methodology Level 4: Benchmarking Level 5: Continuous Improvement	Not identified	Step 1: Analyze the current status of the company. Step 2: Characterization of the company's practices. Step 3: Analysis of surveys built based on PMMM Step 4: Determination of maturity level	183	Not identified	CMMM	Scientific Community

Model	Approach	Domains	Maturity levels	Number of best practices	Evaluation process	Number of questions in the survey	Dimensions	Based on	Generated by
MMGP: Project Management Maturity Model- Darci Prado	Project Management	Project Management	Level 1: Initial Level 2: Knowledge Level 3: Standardized Level 4: Managed Level 5: Optimized	Not identified	Not identified	40	1.- Strategic alignment 2.- Behavioral competence Organizational structure 4.- Computerization 5.- Methodology 6.- Technical competence	PMBOK	Scientific Community
PM2: Project Management Process Maturity Model	Project management process in companies	Project Management	Level 1: Basic Project Management Process Level 2: Individual project planning Level 3: Systematic project planning and control Level 4: Multi-project integration planning and control Level 5: Continuous improvement of the project management process.	Not identified	Not identified	148	Not identified	Not identified	Scientific Community
NPM3: National Project Management Maturity Model	Project Management	Project Management	Level 1: Emerging Level 2: Developing Level 3: Adolescent Level 4: Maturity	Not identified	Not identified	The number of questions for the maturity assessment is not explicitly described.	Not identified	Not identified	Scientific Community
PM2TOM2: Project management methods and tools-oriented maturity model	Project Management	Project Management	1.- Low Management Maturity Level 2.- Lower Medium Management Maturity Level 3.- Medium Management Maturity Level 4.- Advanced Management Maturity Level 5.- High Management Maturity Level	Not used, the model uses 43 methods and tools to assess	Step 1: Collection of data in the organization carry out the projects Step 2: Analysis of the data obtained using the model designed Step 3: Evaluation of project management maturity Step 4: Use of the evaluation results	The number of questions for the maturity assessment is not explicitly described.	1.- Time management 2.- Resource management 3.- Cost management 4.- Risk management 5.- Scope Management 6.- Organizational support of the project 7.-Staff training, and project management support based on literature review	Based on the literature review	Scientific Community
P2MM: Prince 2 maturity model	Project Management	Project Management Program Management Portfolio Management	1.- Awareness of the process 2.- Repeatable process 3.- Defined process 4.- Managed process 5.- Optimized process	Not identified	Not identified	The number of questions for the maturity assessment is not explicitly described.	1.- Management Control 2.- Benefits Management 3.- Financial Management 4.- Stakeholder Engagement 5.- Risk Management 6.- Organizational Governance 7.- Resource Management	Prince 2	Scientific Community
P2CMM: Portfolio Management Maturity Model	Project Management	Project Management	1.- Cognitive level 2.- Repeatable level 3.- Management level 4.- Integration level 5.- Continuous level	This model merges 45 sub-processes of PRINCE2 into 25 sub-processes	The authors describe a process to use the data after the evaluation and describe the type of questionnaire used	The number of questions for the maturity assessment is not explicitly described.	1.-Starting Up a project 2.- Initiation a project 3.- Directing a project 4.- Controlling a stage 5.- Managing product delivery 6.-Managing stage boundaries 7.- Closing a project 8.- Planning	Prince 2	Government

After comparing maturity models, we identified that the current proposal for maturity assessment based on traces has not been explicitly presented in the literature. Project maturity models are developed by both the scientific community and the industry. These models generally consist of between 5 and 6 maturity levels in which organizations are positioned, considering a set of best practices as evaluation criteria. Therefore, industries are compelled to use questionnaires to gather information on whether the individuals responsible for the processes apply them.

It is important to note that the literature emphasizes that maturity models must undergo a preparation process before being implemented in organizations. Likewise, a post-evaluation process is required, which includes analyzing the results and providing feedback. Both processes must be based on the evaluation of best practices through surveys that ask questions about practical cases or, in some cases, direct inquiries about their application. One area of opportunity identified in project management maturity models is the lack of a framework that not only describes what best practices are but also how they should be carried out. This is observed since most models extract best practices from the PMBOK or expert surveys, but do not explicitly describe the process that enables compliance with the best practice requirement.

2.3.4 Summary of the systematic review of project management maturity models

Through this systematic review, a large number of studies related to project management maturity models have been identified, reflecting the growing interest of the scientific and industrial community in this field. The application of a systematic methodology has allowed filtering and analyzing the most relevant studies, providing a comprehensive view of the approaches, domains, maturity levels, and evaluation criteria used in different sectors.

Quantitative results indicate an upward trend in the publication of articles on project management maturity models, especially in the last decade, highlighting the need for tools that facilitate the continuous improvement of organizational processes. The analysis of search engines and repositories consulted reveals that Google Scholar and ScienceDirect are the most commonly used sources by researchers to access relevant literature. From a qualitative perspective, it has been identified that project management maturity models have evolved from their initial application in software development to other sectors, such as engineering, construction, and manufacturing. However, despite their potential, the adoption of these models is limited in small and medium-sized enterprises (SMEs) due to a lack of financial and human resources for their effective implementation. This is concerning, considering that SMEs represent a significant proportion of the global business landscape and play a crucial role in the supply chains of large corporations.

Regarding the comparison of models, 12 main models were identified, among which OPM3, P3M3, and PMMM stand out. These models provide different maturity levels and evaluation criteria but share a common approach based on best practices from the PMBOK guide. However, an identified opportunity area is the lack of a clear methodological framework that not only describes what best practices are but also how they should be effectively implemented within organizations. Additionally, no articles were identified that propose the use of traces to evaluate maturity, nor a conceptual guide that allows individuals to implement these models in a structured manner. Likewise, no methods or tools are proposed to determine whether a software system possesses the minimum necessary functionalities to generate traces and evaluate maturity, which represents a significant limitation in the applicability of these models.

Finally, this systematic review highlights the importance of developing more accessible and adaptable maturity models to meet the needs of SMEs, as well as the need for approaches that integrate emerging technologies, such as data analysis and artificial intelligence, to improve the accuracy and effectiveness of maturity evaluations. These findings provide a solid foundation for future research in the field of project management, focusing on closing the identified gaps and maximizing the positive impact of maturity models across a wide range of industries.

2.4 Use of computer traces to improve project management

To search for specific articles on the use of traces to enhance project management, a classical literature review methodology was employed. This review aimed to identify whether traces have been utilized to improve project management or what their purpose has been in the literature. Google Scholar was used as the search engine, applying ten search strings and conducting three author searches: "Improvement" AND "project management" AND "computer traces", "improvement" AND "project management" AND "computer traceability", "improvement" AND "project management" AND "Information technology traces", "project management" AND "computer traces", "project management" AND "computer traceability", "Mejora" AND "gestión de proyectos" AND "usando datos", "Trace-Based" AND "System" AND "Project Management", "Trace-Based" AND "System" AND "Project Management" AND "engineering projects", "Trace-Based" AND "System" AND "Project improvement", "Trace Based System" AND "project management". The consulted authors were Alain Mille, Raafat Zarka, and Béatrice Fuchs, as the literature suggested they actively work with generated traces.

After implementing the search strings and author searches, a total of 659 articles were examined based on their titles and keywords. Out of these, only 31 publications were considered relevant for the research, based on their content. Overall, a trend in the scientific community is observed, focusing on analyzing experiences or traces generated through human interaction with computing devices, as well as identifying patterns in this behavior. Lathoud [68] discusses the variability in the use of traces generated by software, highlighting the challenge of finding a precise definition and appropriate term for this activity, requiring extensive research. For this study, it was determined that the key term best describing this activity is "Trace-Based System".

2.4.1 Relevant publications identified

1) Formalization of the processing of electronic traces

In [68], the need to formalize the processing of electronic traces in the field of information systems (IS) management is addressed. It highlights the importance of closer collaboration between forensic investigators and IS managers, proposing a generic framework that benefits both parties. The proposed framework includes understanding forensic concepts, designing a trace management process, strategically integrating it into the organization's security policy, and selecting the most relevant traces to enhance system efficiency and security. Potential benefits in terms of security, quality, and legal processes are suggested, along with the possibility of a progressive and modular implementation of the proposed framework.

2) Benefits of a research methodology from organizational sciences for analyzing design interactions

In [29], the increasing complexity of manufactured products and the need for collaboration among designers from various disciplines throughout the design project lifecycle is addressed. The authors emphasize the importance of understanding the multidisciplinary cooperative design process, especially during synchronous phases and design reviews, and propose a research methodology to support this understanding. The relevance of traces to structure and support designers' activities is mentioned, and the interaction analysis framework is introduced to study design activities in the context of Volvo Trucks. The authors have considered Rainbow as an analysis tool, which is an analytical framework developed in the Computer Supported Collaborative Learning (CSCL) community to analyze pedagogical interactions where pairs debate open-ended questions. This framework comprises 7 categories covering activities outside the task, non-task-focused activities, and task-focused activities. Rainbow is useful for retrospectively and contextually coding and analyzing debates using videos, computer traces, and chat dialogues.

3) From case-based reasoning to traces based reasoning

In [69], Two important paradigms in artificial intelligence are presented: Case-Based Reasoning (CBR) and Trace-Based Reasoning (TBR). CBR aims to reuse problem-solving schemes from past experiences to solve new problems in new situations, while TBR relies on the use of event traces to find solutions to new problems. The importance of traces lies in their ability to capture temporally situated human experience, enabling the adaptation of knowledge to new dynamic situations. CBR involves capturing and analyzing user interactions with a computer system to identify useful patterns that can be reused in similar situations in the future. In summary, the article addresses how to amalgamate human experience and computerized systems to solve problems, emphasizing the importance of the CBR and TBR paradigms in this human-machine cooperation process.

4) Providing assistance by reusing episodes stored in traces: a case study with SAP Business Objects Explorer

In the master thesis of [70], the challenge of developing techniques to assist users in their tasks without disrupting their main workflow is addressed. It proposes building an assistant based on traces using a Trace-Based System. Traces are used as a knowledge source to discover useful information and reuse past experiences. The main idea is to identify useful episodes in interaction traces and reuse them to provide users with contextualized help. This approach was implemented in partnership with SAP-BO, specifically in the SAP-BO Explorer project, which aims to provide business users with easy access to the information they need to make confident, up-to-date decisions. The main contribution of the work is an algorithm that enables the retrieval of past episodes corresponding to a "task signature" using finite-state machines.

5) Supporting Activity Modeling from Activity Traces

In [71], the article introduces a new method and tool for modeling activities through qualitative analysis of sequential data. It addresses the construction of a symbolic abstract representation of an activity from an activity trace. The importance of traces lies in their ability to assist analysts in understanding unknown aspects of an activity, constructing meaningful symbolic representations, and testing and supporting the created models. The proposed trace methodology is pragmatic and evolutionary, guided by the analyst's modeling objectives and research questions. Knowledge engineering is used to build an activity ontology and define transformation rules to process raw traces into abstract traces based on the ontology. This tool and approach enable the discovery of innovative ways to represent activities, such as lane changing on highways, from activity traces collected from an instrumented vehicle.

6) Trace-based framework for experience management and engineering

In [72], the article addresses experience management in computer-mediated environments, focusing particularly on complex tasks whose support relies more on experience than on knowledge. An approach based on the concept of "trace use" is presented to investigate the "reflexivity of activity" as a first step in experience management and sharing and reusing experiences as potential applications. Furthermore, the framework supporting the creation of trace-based systems is outlined. Traces, trace models, and the trace lifecycle are formally defined. The main parts of the framework architecture are detailed: collection systems, transformation, visualization, and querying. In summary, the article discusses the importance of traces as a tool for understanding and managing experience in complex computerized environments, offering a methodology for their use and application in experience management.

7) TracED: A tool for capturing and tracing engineering design processes

In [24], the article addresses the need for tools that support the capture and tracking of engineering design processes, where activity is creative and sometimes unstructured. It emphasizes the generation of multiple models during these processes, with different levels of abstraction. Therefore, TracED is introduced as a tool designed to capture and trace the design process, allowing for the definition of a specific design domain and capturing the transformation of products under development throughout the process. It focuses on software architecture design and highlights the importance of computational tools in facilitating the capture of complex processes. Additionally, a future extension is proposed to address conflict management in collaborative design. Despite its limitations, TracED is considered robust enough for experimental use, and integration with existing tools is suggested to improve its usability. Additionally, the article provides a comprehensive overview of the TracED environment, detailing the functionality of its Domain Editor and Versions Manager components, which allow domain experts to define design domains and designers to execute various design projects by leveraging predefined modeling concepts and operations.

8) Rule-Based Impact Propagation for Trace Replay

In [73], the article addresses the topic of providing contextual assistance to application users using trace-based reasoning. Interaction traces, which record the actions performed by a user in a system, are crucial for capturing user experiences and enabling the reuse of those experiences to solve new problems. The importance of traces lies in their ability to provide a detailed record of user actions, allowing not only the replay of those actions but also the exploration, modification, and optimization of user interaction with the system. A proposed trace methodology in the article is "trace replay," which allows users to rewind to previous system states and replay their past interactions. This mechanism is implemented by defining a trace model and applying impact rules to manage dependencies between system elements and the consequences of changes in traces. The implementation includes the collection and visualization of interaction traces, as well as the ability to reuse those traces to assist users during interaction with the system. In summary, the article addresses how interaction traces can be used as a basis for improving user assistance and optimizing the user experience of applications.

9) Cots Products To Trace Method Enactment: Review And Selection

In [74], The article introduces the iAMF project, which aims to integrate Situational Method Engineering (SME) techniques into software development to enhance its transparency and quality. It tackles the challenge of adapting theoretical methods into practice, focusing on stakeholders' actual activities to automatically generate customized methods based on the Agile Method Framework (AMF). The use of trace tools is proposed to record stakeholders' activities and perform intention mining to discover the actual methods used in projects. Additionally, it details the process of selecting appropriate tools using the MADISE decision-making methodology. This methodology, adaptable to the iAMF project context, employs a pairwise comparison matrix to evaluate selection criteria and assign weights, thereby facilitating the identification of the most suitable tool. In summary, the article discusses the importance of traces in decision-making for selecting event-logging tools and presents the MADISE methodology as an effective approach for this purpose.

10) Trace-based reasoning for user assistance and recommendations

In the Ph.D. work of [75], the challenge of developing systems that enable users to share and reuse their experiences in digital environments is addressed. It focuses on the overarching problem of contextual recommendations for specific web applications in particular contexts, such as complex tasks, large amounts of data, and diverse user types. The importance of traces lies in their role as knowledge containers in Trace-Based Reasoning (TBR), an artificial intelligence paradigm inspired by Case-Based Reasoning. Interaction traces reflect the context of user activity and provide suitable and appropriate knowledge for experience-based assistance. This thesis work describes a methodology for representing and manipulating modeled traces, including a method for calculating similarity measures between modeled traces, implemented in a framework called T Store. This framework is a management system for traces, storing M-Traces and responding to user queries to aid in effective task completion. It comprises six modules: Storage, Querying, Transformer, Visualization, Similarity Measure, and Security, leveraging a database management system for performance and storage. It proposes new additions, such as similarity measures and a security manager, to enhance the architecture of trace management systems. Additionally, a trace replay mechanism is presented, enabling users to revert to a particular state of the application, along with a trace-based recommendation approach, facilitating knowledge sharing among user communities and improving the quality of recommendations.

11) Using interaction traces for evolutionary design support

In [76], an approach to controlling the design of online learning devices through the analysis and visualization of interaction traces is discussed. Traces are real-time user action logs collected during their interaction with the system, enabling the representation of user activity and the extraction of knowledge from it. The significance of traces lies in their ability to tailor the design of the learning device to the changing needs of users, as they provide detailed information on how different actors interact with the system. The mentioned trace methodology involves the collection, transformation, and visualization of traces. In the collection phase, user actions are observed and stored; in the transformation phase, high-level representations of significant traces for the user are computed; and in the visualization phase, traces are presented interactively for different actors in the learning environment. This methodology is based on interaction trace theory and aims to enhance the design and quality of online learning devices.

12) Competences Network Based on Interaction Data for Recommendation and Evaluation Aims

In [77], the authors focus on developing a benchmarking system for competency recommendation and evaluation based on interaction data. It highlights the importance of interaction traces, i.e., data generated by user interactions on digital platforms such as professional social networks or emails, which provide valuable information for identifying competencies. The proposed system utilizes these traces to recommend competency profiles to users and evaluate the relevance of these recommendations, which is crucial for enhancing efficiency and effectiveness in competency management within professional environments. In summary, the article addresses the significance of interaction traces as a source of information for identifying and evaluating competencies in a digital context, emphasizing their relevance in optimizing competency recommendation and evaluation processes.

13) Conclusive On-the-fly Validation of High-Level Functional Tests

In [78], the importance of structural testing as an acceptance criterion for safety-critical embedded systems, such as those in transportation, aerospace, and critical infrastructure is addressed. It highlights the difficulty and cost associated with achieving comprehensive coverage of both specified requirements and code deployment, especially in integrated and critical systems. A solution is proposed that validates the conclusiveness of high-level functional tests in integrated safety-critical applications in a non-intrusive manner through real-time online analysis of processor trace data. The significance of traces lies in their ability to provide deep insights into the completeness of tests and their underlying requirements, significantly reducing the effort needed to achieve and justify conclusive coverage statements. The article also discusses technical challenges and opportunities associated with online analysis of execution traces, as well as the design of custom platforms to facilitate the testability of embedded systems.

14) Generating relational database transactions from EB 3 attribute definitions

In [79], is presented the formal language eb3, designed for specifying information systems. This language utilizes recursive functions over valid traces of external events to define attributes of entities and associations independently. An algorithm is presented that generates transactions to update the values of affected attributes in the representation of the relational database. The significance of traces lies in their central role in specification and automatic transaction generation, simplifying the design and implementation of information systems while allowing for the replacement of complex SQL queries with simpler eb3 constructions. This facilitates the development process and enhances efficiency in building information systems.

15) From traces to indicators for serious games: case study concerning a new way to introduce product lifecycle management

In [80], the use of learning games in educational settings, highlighting their potential compared to traditional teaching methods is addressed. It discusses how these games allow students to learn at their own pace, while gaming environments also offer assessment and entertainment features that make them more engaging than Learning Management Systems (LMS). Additionally, it explores the challenges of enhancing the social dimension in these environments and how learning games can address this issue. A Massive Multiplayer Online Learning Game (MMOLG) approach is mentioned, along with its potential to address complex pedagogical goals. The significance of traces in these environments is highlighted as a powerful tool for assessing progress and adapting pedagogical objectives. It describes how traces can be transformed into meaningful indicators for evaluating and adapting the learning scenario, using specific examples of Product Lifecycle Management (PLM) system implementations in Small and Medium Enterprises (SMEs). Dynamic adaptation of the learning scenario is shown as an effective way to enhance the user learning experience. In summary, the article underscores the importance of learning games and trace analysis in addressing complex pedagogical goals and promoting change in educational and business environments.

16) Modeling and exploitation of the traces of interactions in the collaborative working environment

The Ph.D. work of [81], is focused on the importance of collaboration in web-based working environments, specifically in Collaborative Working Environments (CWE), and how these collaborations generate traces or records of activities. These traces not only represent knowledge and experience but are also essential for understanding and analyzing interactions between individuals and systems. Furthermore, a model is proposed to define, model, and leverage these collaborative traces, highlighting their utility in enhancing group experience and facilitating collaboration in areas such as SWOT analysis, group recommendations, and project planning. The implementation of this model in the context of the collaborative platform E-MEMORAe2.0 demonstrates its effectiveness in improving both individual and group work, although some considerations are noted such as the need for many traces, the influence of group size and structure, and the dependence on previous collaborative activities.

17) Trace Based System in TEL Systems: Theory and Practice

In [27], the use of activity traces in Technology-Enhanced Learning (TEL) systems to diagnose and provide appropriate assistance to students, tutors, or teachers during the learning process is discussed. It highlights the difficulty of personalizing tracing and diagnostic processes in most distance education systems, limiting the provision of customization services and remedies during learning activities. Several tracing systems are mentioned, including the Trace-Based System (TBS), which manages modeled activity traces (m-traces) structured as objects and proposes transformation operators to obtain other transformed m-traces. Also presented are the Trace-Based Indicator Management System (TB-IMS) and the Multi-Agent System for Computing Human Learning Indicators (IC-MAS). These systems use activity traces to calculate indicators of human learning and offer concrete methods and tools for their implementation and management. The importance of traces lies in their ability to provide knowledge about learning processes, enabling customization, understanding, and improvement of educational activities, which has significant implications for the efficiency and ethics of online learning.

18) Trace-based contextual recommendations

In [26], the topic of contextual recommendation systems using interaction traces is addressed. It highlights the importance of context in making recommendations, especially in activities like selecting trips in different seasons. It shows how interaction traces can capture user experience on the web and be used for various purposes such as task automation, user assistance, and recommendations. The concept of Trace-Based Reasoning (TBR) is introduced, which leverages past experiences to solve new problems. A trace system called T Store is mentioned, which stores, processes, and exploits interaction traces. Additionally, an example of recommendation in the web application Wanaclip is presented, where interaction traces are used to recommend actions to users while they create video clips. The importance of evaluating accuracy, responsiveness, and user satisfaction is also highlighted in the article. In summary, the article provides an innovative approach to context-based recommendations using interaction traces, demonstrating its effectiveness in the Wanaclip application, and suggesting areas for future research, such as the development of indices to accelerate recommendations.

19) Facilitate sharing of training experience by exploring behavior discovery in trainees traces

In [82], the crucial importance of harnessing and sharing the knowledge of expert tutors in the handling of training simulators, particularly in critical environments such as the training of nuclear plant operators, where operator certification heavily relies on the skills of the tutors and allows no room for errors is explored. It presents a methodology for visual synthesis of activity to analyze data collected during simulation sessions, detailing four levels of traces. Additionally, a generic trace model is outlined, stipulating that each simulation M-Trace must have a specific identifier to be locatable and usable over time. In addition, a transformation model is described, enabling the generation of an objective trace of level n from one of level n-1. A prototype called Define, Discover, and Disseminate Knowledge from Observation to Develop Expertise (D3KODE) is introduced, designed to facilitate the visualization and analysis of learners' activities.

20) Traces and activity: a case study of a joint writing process mediated by a digital environment.

In [83], the importance of interaction traces in computer environments where collective work oriented towards human learning takes place. It emphasizes the relevance of analyzing both the direct communication flow between actors and the digital traces generated by interaction with computer hardware. These traces are written registers of the sequence of events on the hardware, such as keyboard, screen, and network interactions. It is mentioned that certain digital instrumentation systems automatically save interaction traces that are more meaningful than simple conventional log files. The importance of these traces lies in their role in users' appropriation of the digital environment. The research of the Silex team is highlighted, which focuses on studying how users utilize computer interaction traces. In summary, the article addresses the importance of interaction traces in human learning and their role in users' appropriation of the digital environment.

21) Vers des traces numériques comme objets informatiques de premier niveau

In [28], the importance of digital traces in computing processes and their role in measuring and understanding human activities mediated by computer environments is discussed. Traces are digital traces of activities that are constantly generated during the use of computer systems, ranging from documents and emails to application and server activity logs. The importance of traces lies in their ability to provide detailed information about computing processes and human interactions, enabling a deep analysis of the activities performed. The article mentions a trace system called SGBTm (Modelized Traces Management System), which facilitates the manipulation and transformation of digital traces, providing tools for interpreting them at different levels and application contexts. This system allows for a formalized management of digital traces, which is essential for the dynamic construction of knowledge and its application in various fields such as education, knowledge management, and collaborative ontology engineering, among others.

22) An approach to User-Centric Context-Aware Assistance based on Interaction Traces

In [22], the development of a contextual assistance system for users, proposing an innovative approach based on modeling user interaction traces is addressed. It argues that traditional methods of context modeling have limitations, such as the difficulty in predicting context in advance and the loss of information when applying the same model to all situations. In contrast, the proposal focuses on modeling user interactions with the environment and reusing these traces to provide relevant contextual assistance. A system called Trace-Based System (TBS) is described for collecting, transforming, and reusing traces interactively. The importance of this approach is highlighted for capturing the uniqueness of context in each situation and its greater relevance for user contextual assistance. Additionally, it discusses how the system can provide recommendations based on identifying recurring patterns in user interaction traces, as well as offering narratives of similar past stories to enhance their understanding of the current context.

23) Assister l'utilisateur à expliciter un modèle de trace avec l'analyse de concepts formels

In [25], the importance of traces in studying and understanding complex phenomena, particularly the numerical interaction traces that record users' actions when performing tasks with computer systems is discussed. It highlights the increasing availability of tools for trace management, which facilitate their production, collection, and subsequent exploitation for various purposes such as analysis, visualization, and knowledge discovery. The concept of a Modelized Traces-Based System (SBT) is introduced, which is dedicated to the modulation and manipulation of numerical traces, associating them with a model for interpretation. The kernel for the Trace-Based System (kTBS) system is mentioned, which uses the Resource Description Framework (RDF) formalism to describe traces and their models. In practical terms, KTBS provides an environment for trace modeling, where data captured during user-system interaction can be represented in a structured format. Additionally, the trace collection process is described, highlighting the complexity of integrating multiple heterogeneous sources and the need for manual modeling. The use of Formal Concept Analysis (FCA) is proposed to assist in this process, illustrated with the case study of the Tamagocours game, demonstrating how an interactive trace model is generated using this methodology.

24) Transmute: un outil interactif pour assister l'extraction de connaissances à partir de traces

In [23], the process of Knowledge Extraction from Data (KED), focusing on the interpretation stage and the importance of traces in this process is addressed. Traces are sequences of temporally observed elements associated with a model that describes their attributes and relationships. It highlights the relevance of trace-based systems, such as the kTBS framework, which facilitate the collection, processing, and visualization of traces. The Transmute system is presented, designed to visualize, transform, and assist in the interpretation of traces, including KED functionalities for data preprocessing, mining, and post-processing. This system was originally conceived to aid in trace interpretation. Furthermore, it has been enriched with Knowledge Extraction from Data (KED) functionalities. Its architecture includes various modules, such as Samotraces for customizable trace visualization and DisKit for implementing the KED process, making it a versatile tool for trace-based data analysis. Interactivity in Transmute allows the analyst to see the impact of their actions and select the most relevant mining results, with the possibility of iterating the process with new constraints.

25) Interprétation Interactive de connaissances à partir de traces

In [84], Knowledge Discovery from Data (KDD), an interactive and iterative process to uncover new knowledge in large volumes of data using non-trivial methods is discussed. It emphasizes the importance of traces in this process, which are temporally situated sequences of observed elements, and proposes a trace-based system to assist in interpreting the results of data mining. One of the mentioned trace systems is the kTBS (Kernel for Trace Based System) framework, which facilitates the collection, manipulation, and visualization of traces. This system includes a pre-processing and mining process, where a sequence is constructed from a trace selected by the analyst, and mining algorithms are used to search for patterns in these sequences. Subsequently, a post-processing and interactive interpretation stage is conducted, where the mining results are presented to the analyst, who interprets and selects the most relevant ones.

26) A Trace-Based Approach for Managing Users Experience

In [85], the concept of digital traces left by users in computer systems and their importance as containers of knowledge highlights their relevance in knowledge dynamics, and user assistance in various tasks is addressed. It proposes a modeled trace approach aimed at studying knowledge dynamics based on traces left during individual or collective activities in digital environments. It presents a framework for interactive knowledge discovery based on traces, which includes a trace-based system (TBS) for managing and manipulating these traces. The framework is exemplified with two applications: musical analysis and active video reading, emphasizing the system's capability to assist in complex and temporal analysis. The principles of the modeled trace approach, the framework's structure, and future research perspectives are discussed. Regarding the importance of traces, their utility in pattern detection, user activity analysis, and knowledge management is highlighted. As an example of a trace system, the TBS is mentioned, which provides basic functionalities for managing and transforming digital traces, enabling collection, manipulation, and analysis of these traces for various purposes.

27) Extraction d'épisodes séquentiels à partir de Traces : application au jeu Tamagocours

In [86], the article focuses on trace analysis within the framework of an interactive approach to Knowledge Extraction from Data (KED), aiming to assist the analyst during interpretation. It emphasizes the importance of traces in detecting regularities, such as sequential episodes and sequential rules, which are recurrent patterns of actions in activity traces. The DISKIT system and the DMT4SP1 prototype are mentioned as tools for extracting these patterns. The article highlights the importance of involving humans in the knowledge discovery process through visual and interactive approaches, as in the Tamagocours game, where a Trace-Based System (TBS) and an interactive graphical interface called TRANSMUTE are employed. Various techniques and constraints for trace analysis are mentioned, including pre-processing, support, syntactic, temporal, and post-processing constraints, to reduce redundant results and facilitate interpretation. In summary, the article addresses the importance of trace analysis in knowledge extraction, emphasizing the need for interactive and visual approaches to assist analysts in interpreting the results. Tamagocours is a serious game used for learning the rules of digital resource diffusion.

28) Focaliser l'extraction d'épisodes séquentiels à partir de traces par le contexte

In [30], the importance of analyzing digital learning or gaming traces, which describe users' actions in a digital environment and are crucial for understanding user difficulties and enhancing available tools is addressed. Knowledge extraction from data (KED) is highlighted as a key method for understanding these traces, with a focus on extracting sequential episodes to identify behavior patterns. The Tamagocours game is mentioned as an example, where difficulties were encountered due to inadequate consideration of action context. The importance of traces lies in their ability to provide valuable information about user behavior in digital environments, which can enhance assistance and learning tools. One of the mentioned trace systems is DISKIT, which allows the formulating of various queries on the data, even in educational contexts. Additionally, the FINEPIO model is mentioned as an approach to integrate context into trace analysis, although its implementation is still under development.

29) Théorie de la trace, application à l'analyse musicale

In [87], the article focuses on researching user interaction with computer tools as a system that co-learns and co-evolves during activities. It highlights the use of knowledge dynamics engineering to leverage activity inscriptions in the form of traces, which are traces of performed activity. The importance of traces lies in their ability to assist the user in understanding knowledge through the application of formal models and associated manipulation tools. A trace-based system (TBS) is proposed for the storage and manipulation of these traces, with an open architecture and formally defined operations. An explored application area is musical analysis, where scores are modeled as traces and interactive tools are proposed to explore and discover knowledge within them. The main goal is to provide a knowledge management environment for musical analysts, where traces derived from scores are transformed and analyzed from various perspectives, thus facilitating an interactive process of discovery and rewriting into more abstract forms.

2.4.2 Comparison of the relevant publications identified

Relevant points for the research were identified, including approaches to computer traces, systems based on computer traces, and trace-based recommendation systems. Current approaches to computer traces encompass learning projects, information management, personality analysis, incident detection in software development companies, computer forensics, software reuse, and knowledge management. Additionally, various systems mentioned in the literature were identified, such as TracED, model generation systems through traces, trace-based assistants, unstructured trace reuse systems, frameworks for exploiting collaborative traces, the Tstore recommendation system, trace-based user interaction assistance systems, user context modeling through computer traces, and finally, data mining. Also, a trace-based language called EB3 was identified. To summarize the research works identified Table 4 is generated.

Table 4. Approach of Trace publications identified.

Article	Trace Approach	Autor (s)	Publication Year
Formalization of the processing of electronic traces	Cyber-security	Bertrand Lathoud	2007
Benefits of a research methodology from organizational sciences for analyzing design interactions	Collaborative Work	Guy Prudhomme, Kristine Lund , Jean-Laurent Cassier	2010
From case-based reasoning to traces based reasoning	Case-based Reasoning	Alain Mille	2006
Providing assistance by reusing episodes stored in traces: a case study with SAP Business Objects Explorer	Software assistant development	Raafat Zarka	2010
Supporting Activity Modeling from Activity Traces	Trace-based Modeling Activity Ontology	Olivier L. Georgeon, Alain Mille, Thierry Bellet, Benoit Mathern, Frank E. Ritter	2011
Trace-Based framework for experience management and engineering	User-experience Management	Julien Laflaquière, Lotfi S. Settouti, Yannick Prié & Alain Mille	2006
TracED: A tool for capturing and tracing engineering design processes	Engineering Desing Process	ML Roldán, S Gonnet, H Leone	2010
Rule-Based Impact Propagation for Trace Replay	Past user experiences	Raafat Zarka, Amélie Cordier, Elöd Egyed-Zsigmond & Alain Mille	2011
Cots Products to Trace Method Enactment: Review And Selection	Information Systems Engineering Methods	K Ghazaleh, H Charlotte	2013
Trace-based reasoning for user assistance and recommendations	User assistance and recommendations	Raafat Zarka	2013
Using interaction traces for evolutionary design support	Online learning devices	K Sehaba, SMV Metz	2011
Competences Network Based on Interaction Data for Recommendation and Evaluation Aims	Competency recommendation and evaluation	S Yahiaoui, C Courtin, P Maret, L Tabourot	2018

Article	Trace Approach	Autor (s)	Publication Year
Conclusive On-the-fly Validation of High-Level Functional Tests	Safety-critical embedded systems,	TB Preußer, A Weiss	2020
Generating relational database transactions from EB 3 attribute definitions	SQL improvement	F Gervais, M Frappier, R Laleau	2009
From traces to indicators for serious games: case study concerning a new way to introduce product lifecycle management	Learning improvement and product lifecycle management	P Pernelle, JC Marty, T Carron	2013
Modeling and exploitation of the traces of interactions in the collaborative working environment	Collaborative Working Environments	Qiang Li	2014
Trace Based System in TEL Systems: Theory and Practice	Learning process	T Djouad, A Mille	2018
Trace-based contextual recommendations	Recommendation systems	R Zarka, A Cordier, E Egyed-Zsigmond	2016
Facilitate sharing of training experience by exploring behavior discovery in trainees' traces	Knowledge Discovery	O Champalle, K Sehaba, A Mille	2015
Traces and activity: a case study of a joint writing process mediated by a digital environment	Collaborative Work	M Ollagnier-Beldame, C Brassac, A Mille	2014
Vers des traces numériques comme objets informatiques de premier niveau	Computing Process	PA Champin, A Mille, Y Prié	2013
An approach to User-Centric Context-Aware Assistance based on Interaction Traces	Contextual Assistance	D Cram, B Fuchs, Y Prié, A Mille	2008
Assister l'utilisateur à expliciter un modèle de trace avec l'analyse de concepts formels.	Contextual Assistance	B Fuchs	2017
Transmute: un outil interactif pour assister l'extraction de connaissances à partir de traces	Knowledge Extraction	PL Barazzutti, A Cordier, B Fuchs	2016
Interprétation Interactive de connaissances à partir de traces	Knowledge Discover Data	A Cordier, B Fuchs	2016
A Trace-Based Approach for Managing Users Experience	Knowledge Discovery	B Fuchs, A Belin	2012
Extraction d'épisodes séquentiels à partir de Traces : application au jeu Tamagocours	Visual and Interactive	B Fuchs	2017
Focaliser l'extraction d'épisodes séquentiels à partir de traces par le contexte	Knowledge extraction	B Fuchs	2018
Théorie de la trace, application à l'analyse musicale	Musical Theory	B Fuchs	2012

2.4.3 Summary of the classical literature review of the use of computer traces

The range of applications is extensive. Traces have been utilized in a variety of domains, including cybersecurity, software engineering, collaborative design, user experience management, digital learning environments, and contextual assistance. Specific systems have been developed to capture, model, and analyze traces in contexts such as education, trace-based systems in TEL, professional competency management, and engineering process design. However, no application or use of traces to assess maturity in project management was identified, representing a gap in the current literature and an opportunity for future research. Two key approaches to reasoning were identified: case-based reasoning (CBR) and trace-based reasoning (TBR). The latter allows for the capture of human experience in a contextual and temporal manner, facilitating the adaptation of solutions to new situations. The combination of both paradigms has proven effective in the development of recommendation and user assistance systems.

The formalization and management of traces is a critical aspect. The need to formalize trace management in information systems has been highlighted as a crucial element to ensure security, quality, and legal compliance. The proposed models address everything from trace collection to transformation and visualization, enabling a better understanding of business and design processes. Trace-based recommendation systems have been developed to improve user assistance in business and educational applications. Systems such as T Store and TB-IMS allow the reuse of past experiences to enhance real-time decision-making.

Several challenges have been identified, including the lack of standardization in terminology. A review of the literature reveals several challenges, including the lack of standardization in terminology and methodology of trace analysis. The selection of relevant traces and their accurate interpretation represent significant challenges requiring interdisciplinary approaches and advanced data analysis tools. Privacy and security in the collection and management of traces are critical aspects that should be considered in future research. Emerging trends indicate a growing preference for data mining and machine learning techniques to analyze traces, offering the potential to generate more in-depth and customized insights. The application of traces in collaborative and knowledge management environments is expanding, with a particular emphasis on enhancing data-driven decision-making.

In summary, a comprehensive review of the literature suggests that traces are a potent instrument for enhancing project management, facilitating data-driven decision-making, personalizing the user experience, and promoting continuous process improvement. However, their effective implementation necessitates addressing challenges related to interpretation, privacy, and standardization.

2.5 Trace Transformation process

Trace-based systems typically involve three stages: collection, transformation, and visualization of transformed traces. Collection is the process of gathering traces generated as a result of interaction between humans and software, requiring identification of where the traces are stored and how the software generates them. Transformation, on the other hand, is a more complex process involving knowledge of software architecture, functionalities, user permissions, software type, operating system, and approach. Finally, the visualization process considers transformed traces with a comprehensible low-level granularity capable of describing user actions rather than detailed data.

Given the complexity of the trace transformation process, a rapid systematic review was conducted to identify methodologies, processes, or methods of trace transformation and their potential uses in the literature. The classical literature review considered the research question: "Is there a general methodology, process, or method of trace transformation focused on project management?" From this question, the following search strings were generated in Spanish and

English and implemented in Google Scholar: "Transformación" AND "trazas," "Transformation" AND "Traces," "Logs" AND "Transformation," "Logs" AND "Transformation" AND "Methodology," "Logs" AND "Transformation" AND "Process." As a result of the search, a total of 32 articles were identified. After reading the articles only 24 were considered relevant.

2.5.1 Relevant publications identified

1) Odoo module for extracting and transforming event log traces for Process Mining

In the thesis of [88], the complex task of analyzing traces generated by an ERP software like Odoo, and recognizing the inherent challenges in their exploitation is addressed. These challenges include identifying events within the traces, managing convergence and divergence issues between activities, and properly crafting the event log to ensure its usefulness in analysis. Despite the existence of various event management tools such as Eventifier, XESame, and Xtract2 tool, each has its advantages and disadvantages. Following a thorough analysis, the authors highlight that XESame and the SAUXE module are the only ones capable of exporting traces in XES format, with the latter being particularly relevant for Odoo as it leverages a module called Odoo Connector for real-time event capture. However, despite these available options, the authors propose a new module that integrates identified best practices to optimize event management and prepare data for subsequent analysis using process mining techniques.

2) Uncovering Object-centric Data in Classical Event Logs for the Automated Transformation from XES to OCEL

In [89] the approach of automatic information discovery using flat event logs, following the OCEL format and object-oriented orientation is addressed. Traditional data mining techniques assume that each event in a log corresponds to a single case, represented by a clear and unique case notion. However, complexity arises when the process allows for the concurrency of different types of objects with complex interrelationships, making it difficult to define a single-case notion. This leads to the loss of information about other objects when recording an event from this perspective. To address this issue, the authors have adopted an approach of object-centered event logs, capable of capturing multiple concurrent objects in the process. The proposed process combines semantic analysis of textual attributes in flat event logs with data profiling and extraction of control-flow-based relationships. The transformation follows the steps of 1) Action and object type extraction, 2) Identifier and object type matching, 3) Object instance discovery, 4) Property type assignment, and 5) Instance event assignment.

3) Component for extraction and transformation of event logs from data-aware web systems

In [90], the article describes that information systems are fundamental tools for companies and organizations, enabling them to manage operations, interact with customers, and compete in the market. These systems record events that occur within the system, and by applying process mining to these traces, valuable insights can be gained to identify opportunities for improvement in organizational processes. Process mining is used to discover, monitor, and implement process improvements through knowledge extraction. The article aims to propose a software component that extracts and transforms traces for process mining in a data-oriented web system. The extraction and transformation process involves defining processes, activities, and additional attributes, capturing events, and generating traces in XES format. Subsequently, ProM and the Visual Miner module are used to visualize the discovery process. This component can be integrated into any web management system and configure organizational processes. The proposal includes the use of a data model to reorganize traces into database tables and transform traces into XES format for applying process mining techniques.

4) Quality evaluation in transformation of event logs into visual representations

In the master thesis of [91], software quality is explored as an essential factor in software products; however, its definition poses a difficult challenge as it involves various perspectives and contexts. Currently, efforts are underway to define it using the ISO-9126 Software Product Evaluation Characteristics standard, which allows for quality characterization. Critical security systems play a crucial role, as failure in such systems can pose a high risk of human loss. These systems require a significantly high level of quality, necessitating greater attention to quality management. This research focuses specifically on defining and evaluating the quality of security-related programs used in crisis management activities. Crisis management involves a series of activities carried out to handle an incident across different organizations. This master thesis proposes the development of an interactive web application with collaborative crisis management and quality assessment support, applying the ISO/IEC 9126 standard. This application focuses on transforming textual event logs generated during a crisis into visual representations in the form of maps. To achieve this, the authors propose the development of a prototype that utilizes positioning data, with the main challenge being to ensure quality evaluation.

5) LogEvent2vec: LogEvent-to-Vector based anomaly detection for large-scale logs in internet of things

In the article [92], the Internet of Things (IoT) has led to a significant increase in the amount of data collected from information systems, mainly through sensors that record a wide range of states and events during their operation. These traces, being semi-structured textual data, represent a valuable source of information for both research and industry, allowing for the identification of behavior patterns and the detection of potential system errors. To conduct effective analysis, various methods have been developed, such as regular expressions, log analysis, and machine learning models, aimed at structuring and processing this abundant information. However, these approaches often entail considerable computational costs. To address this issue, an innovative model called LogEvent2Vec is proposed, designed to extract relevance between log events and vectorize them directly. This approach not only significantly reduces the time required for training and coordinate transformation, but also demonstrates superior effectiveness compared to other methods in the state of the art. Experimental results show improved performance of the Random Forest machine learning model when used in conjunction with LogEvent2Vec compared to approaches based on neural networks and Naive Bayes.

6) Extracting event logs for process mining from data stored on the blockchain

In [93], The implementation of blockchain in inter-organizational execution offers a new perspective, providing participants with the updated version of the blockchain with each new block following the consensus algorithm. There are advantages to its implementation, such as avoiding the need for a central orchestrator and promoting trust among interacting parties, and decentralized databases, as well as increasing the ease of retrieving data on interactions, interaction time and duration, and blockchain execution costs. These advantages enable process monitoring and evaluation through a mining process. However, to utilize the data, manual effort is required to transform it into an appropriate format, which poses several challenges for implementation.

The data obtained from the blockchain is represented in hexadecimal and numerical formats, which have approximate timestamps because they are organized in blocks. Therefore, a complete understanding of the blockchain structure is required to transform the data. To address the complexity of transforming blockchain data into event logs, a research methodology in design sciences was used to create a framework that retrieves process data from blocks and transforms them into event logs in a format according to the IEEE Extensible Event Stream (XES) standard. To evaluate the proposal, a case study was conducted in which, after converting the data to XES format, the ProM tool was used.

7) Root cause analysis with enriched process logs

In [94], the prioritization of risk management in business strategy for top-level executives and CEOs, highlighting the significance of risk mitigation through the identification of key factors that trigger risk events is emphasized. Root cause analysis involves pinpointing the explanation for risk incidents, preferably involving the fewest possible factors. Utilizing stored traces for root cause analysis proves highly beneficial, as data serves as a reliable source reflecting reality rather than a mere perspective of it. Here, post-event data serves as a starting point for comparing characteristics, saving attributes, and identifying useful and useless cases. However, post-event traces, or event logs, may lack the necessary attributes or characteristics to explain potential root causes of an incident. Manipulating these traces to obtain relevant attributes is primarily done on a case-by-case basis, highlighting the lack of a generic systematic approach to attribute extraction. Therefore, this article proposes a systematic approach that begins with existing event traces and is supplemented with relevant contextual information to explain potential root causes. The proposed solution approach consists of the following steps: 1.- Determining response and prediction variables 2.- Enriching raw event traces 3.- Case logging 4.- Ready record classification 5.- Data mining analysis; Enriching traces and case logging are informed/impacted by response and prediction variables, which in turn receive input from ready record classification.

8) Process mining by using event logs

In [95], process mining is defined as a technique that involves the use of tools and methods to discover, monitor, and control organizational data to enhance real processes by extracting knowledge from system logs. Its applications span across various domains such as medicine, customer relationship management, product development, workflow management, and enterprise resource planning, among others. In this context, data is sourced from real-time processes and utilized to reconstruct process models, focusing on process analysis and optimization. Notably, process mining acts as a bridge between process management and data mining. While event logs are valuable for extracting process models, real-life processes often lack structure, resulting in ambiguous and complex process models. Traditional solution approaches propose clustering specific process instances, where each resulting process cluster corresponds to sets of process instances that can eventually be adequately represented by a process model. This involves identifying various process variants based on clustering event logs. The proposed approach entails a series of steps: 1) transformation of log files (String Tokenizer, Raw Detailed Filtering), 2) clustering (ActiTrac Algorithm), 3) comparison of clusters with mining process definition, 4) generation of mining model, and 5) prediction generation for mining process models.

9) Discovering web access patterns and trends by applying olap and data mining technology on web logs

In [96], the authors describe how web server logs capture various information in each entry (such as requested URL, IP, and timestamp), reflecting user interaction with the website. This interaction grows daily, leading to an increase in the volume of raw logs captured by servers, posing a non-trivial task to identify significant and useful information. The benefits of using web server logs depend on the nature of the business, including performance analysis, system design enhancement, understanding web traffic, and user reactions and motivations. The success of web log analysis applications relies on the validity and viability of the knowledge discovered in large clusters of information.

With over 30 solutions in the market, many prove slow, inflexible, and difficult to maintain. Therefore, this article proposes a tool called WebLogMiner for mining log files, benefiting from online analytical processing, data mining, and multidimensional data cubes to extract implicit knowledge interactively from vast volumes of web logs. While individual parameters recorded in logs may be irrelevant, combining different parameters can yield relevant information. For the WebLogMiner proposal, the authors outline four stages: data filtering and storage in a database, construction of a data cube using available dimensions, online analytical processing to process the log cube, and implementation of mining techniques on the log cube to predict, classify, and describe interesting correlations. As part of the transformation process, the authors group a series of user actions by event to reduce the number of logs. An example of this grouping is as follows: If a user performs multiple actions related to purchasing a product, all actions are labeled as "payment," indicating that the user completed the payment.

10) Mining Moodle Logs for Grade Prediction

In [97], the authors address the growing relevance of learning analytics, driven by the increasing availability of data generated from educational tools. These data are derived from students' interactions with the system, such as Moodle or other teaching-focused systems, applications, or platforms. While some studies have focused on social networks and graph analysis using Moodle as a reference, others have implemented interaction analysis to generate visual graphs. Here, the authors present a methodology combining data mining and machine learning (training, testing, and tuning) to predict grades.

Additionally, this proposed methodology was applied in a practical case study using 33k traces from 161 students enrolled in the "Technical Communication" course on the Moodle platform. The methodology entails the following steps: 1) obtaining data from the Moodle platform, 2) data preparation, 3) data statistics (statistical analysis), 4) exploratory data analysis, and 5) model generation (identification, creation, and training). The practical case study yielded insights into student behaviors regarding course information access and class attendance. Trends were observed indicating a tendency to avoid errors and a low rate of result updates, suggesting a lack of interest in improving previous work but rather a focus on completing tasks.

11) Data Transformation and Semantic Log Purging for Process Mining

In [98], process mining is oriented towards the analysis of execution event logs with two approaches: process model knowledge discovery and comparative analysis of process models. Event logs are data that are generally distributed across various processes and are not always found in a single location. Due to this variety of data sources within applications, it is necessary to develop adequate methods for data transformation and define this process as part of the preprocessing phase within a process mining project. Some of the challenges faced at this stage include incorrect or incomplete data, data distributed across parallel branches, infrequent traces, and ad-hoc constructed log data instances of change.

This article proposes addressing the challenges of the preprocessing stage by involving data quality. It presents a method based on data collection queries and temporal application data transformation into process logs and a data cleaning method based on semantic log purging. To evaluate this, the authors applied the proposed methods to a realistic dataset from a higher education domain. The methodology applied to compare process mining with semantic logs is as follows: a) data transformation, b) semantic log purging, c) mining process, and d) evaluation. For data transformation, a method was designed that extracts and transforms data based on functional queries. The transformation process consists of 4 steps: 1) data transformation into internal object representation, 2) persistence of internal object representation, 3) querying within internal object representation, and 4) transformation of internal object representation.

12) Abstracting process mining event logs from process-state data to monitor the control-flow of industrial manufacturing processes

In [99], the applicability of process mining methodologies in combination with data-driven digital twin concepts to discover and monitor process control flow is explored. With the rise of Industry 4.0, the use of data sensors in manufacturing has significantly increased, providing massive data sources. The digital twin concept serves as a driver for digital transformation, focusing on enhancing competitiveness, productivity, and efficiency by creating a virtual representation of a production system integrating data sensors, mathematical models, and real-time data processing. Consequently, due to the vast amount of data generated in manufacturing processes today, there's a growing need for methods to extract knowledge from this data. Process mining operates on well-defined event logs, such as the major steps in a process. The authors propose an abstraction step to identify recurring process states and map them to high-level activity logs to facilitate the use of process mining methodologies.

13) From Low-Level-Event-Logs to High-Level-Business-Process-Model-Activities: An advanced framework based on machine learning and flexible bpmn model translation

In [100], the authors propose a machine-learning approach to assign low-level event logs to high-level activities in business processes. The approach consists of two phases: automatic labeling and machine learning-based classification. For automatic labeling, a modification of k-prototypes clustering is used, and the labels from the examples are employed for training different machine learning classifiers. In recent years, there has been a shift from data-centric to process-centric management approaches. Business processes aim to divide a sequence of activities where different individuals are responsible for different tasks. Efficient and well-organized process management significantly impacts a company's success and deliverables. Proper business process management involves modeling, discovering, and analyzing processes for improvement.

The integration of process mining into process management is an emerging research area focusing on extracting insights from event logs in systems. Mapping event logs to a process model entails different levels of granularity, as event logs typically have higher granularity than models. Some approaches map processes and event logs one-to-one, substituting event names with activity names. This granularity difference requires an efficient algorithm to map low-level event logs to high-level process activities for process model creation. This article implements an artificial intelligence model capable of learning the mapping between event logs and process activity models in two phases: automatic labeling and automatic classification. The proposed approach includes the following stages: a) automatic labeling, b) clustering of sequences with associated labels, c) training phase, d) machine learning-based classification, e) event sequence labeling, and f) creation of high-level events.

14) The ONPROM Toolchain for Extracting Business Process Logs using Ontology-based Data Access

In [101], the ONPROM tool is proposed to assist in the preparation and extraction of traces from legacy information systems. It aids users in semi-automatically extracting event logs from the system and reflecting them in views across different related processes, thus facilitating process mining. This tool is based on a data paradigm grounded in ontology, comprising three components: the UML editor, notation editor, and trace extractor. Each component can be integrated with the ProM framework, ensuring compatibility with the XES standard format. Process mining has gained popularity for bridging the gap between process models and event data generated during execution. It combines model-based analysis and data-driven analysis to derive valuable insights into real-world business processes. However, for process mining to be effective, data quality and structure are essential, ensuring suitability for analysis algorithms. Organizational information systems often have specific and complex configurations that do not explicitly present data, necessitating data restructuring post-extraction. To address the complexity of data acquisition and preparation, this article proposes the ONPROM methodology. This methodology enables data and process analysts to explore concepts and their relationships by extracting event traces. The authors outline a methodology proposed in two previous articles, consisting of the following steps: creating a conceptual data schema, establishing mappings, selecting a perspective, creating event data annotations, obtaining data in XES or CSV format, and applying process mining.

15) Data Mining of search engine logs

In [102], a new method is proposed for analyzing web logs, utilizing techniques that seek similarities between queries and identify query transformation sequences. This enables transformation sequences to be represented as network graphs, providing insights into user behavior that sequential descriptions alone cannot offer. Analyzing the behavior of users performing information searches is part of a long list of development design projects. Query transformations occur when searches on web engines are ineffective at identifying requested information. Analyzing these transformations helps better understand the essential generic strategies for improving search engines to make searches more effective. In this article, the authors focused on developing a method for extracting and analyzing types of query transformations using logs obtained from the Excite browser in 2001. The lack of context in such logs complicates their analysis, necessitating the application of analytical methods that consider abundant syntactic changes. The methodology for data analysis was as follows: a) sampling, b) offline data analysis, c) Boolean phrases and terms, d) query comparison, e) behavior transformation codes, f) query transformation results.

16) Data Transformation and Semantic Log Purging for Process Mining

In [103], experimental analysis was conducted to showcase the operational capabilities of the conceptual mining framework in rediscovering workflow process patterns and their enacted proportions from workflow enactment event logs recorded in a standardized XES schema format.

The research proposed the implementation of a workflow mining algorithm solely to discover all process patterns and their enacted proportions. Web-based organizations present new types of requirements and demands regarding intelligence flow and discovery. Improving workflow fidelity relies on the use of execution logs, as it ensures high, consistent, and predictable efficiency.

This article adopted a conceptual approach utilizing the control-flow-oriented process knowledge analysis algorithm, which is effective in discovering disjunctive and conjunctive process patterns by enacting event logs specifically in XES format. As part of the experimentation, this proposal was implemented in a real framework using event log files provided by BPI Challenges of 4TU. The methodology for implementing the proposal included: a) preparation of the workflow enactment event trace, b) extraction of proportional process patterns (extraction of a group of adjacent activity pairs, extraction of a quantitative group of adjacent activities, and extraction of a proportional process pattern graph).

17) Mining console logs for large-scale system problem detection

In [104], the article proposes a method for mining application console logs. Firstly, syntactic analysis of logs and text mining is combined to extract a log structure. Secondly, features are extracted from this structure to detect anomalies in log patterns using principal component analysis. Finally, a session tree is utilized to distill the principal component analysis-based anomaly detection results into a format understandable by domain experts. The complexity of monitoring large-scale services running on extensive server clusters complicates the design, deployment, and maintenance of monitoring systems.

The authors advocate for the utilization of console logs, a common form of tracking in most software systems, which are often disregarded due to their difficulty in comprehension despite being used for generating internal state reports, executing program traces, and real-time execution statistics. Understanding console logs is challenging due to their lack of structured text and ambiguous implementation details. To address this, a console log mining approach is proposed to detect real-time issues in large-scale systems, aiming to present only relevant information from console logs to operators in a more user-friendly format. The proposed method was implemented in a case study using Hadoop logs running on Amazon's EC2. The methodology includes syntactic log analysis and structure extraction, feature vector construction, principal component analysis-based anomaly detection, and visualization of detection using a decision tree.

18) Web logs pre-processing and analysis: a survey

In [105], the article introduces mining techniques for web log files applicable across various domains. The global web serves as a vast data repository, with users interacting significantly with its data. Each user-web interaction is logged in a .txt file, containing raw, incomplete, and unnecessary information. Mining techniques applied to web logs include preprocessing, pattern discovery, and pattern analysis. Various types of logs exist, such as access, referral, error, and user cookies, stored on servers, proxies, and browsers. Each line in these logs represents a user's request to the server, containing IP address, request timestamp, exact user request, URL, file type, etc. The ircache.net repository provides user-friendly datasets in common formats, including Common, Microsoft IIS, and NSCSA. Mining techniques include web usage mining (comprising data preprocessing, pattern discovery, and analysis), web content mining, and web structure mining. Web log preprocessing involves cleaning, user identification, session identification, data transformation, and path completion. Web usage mining identifies patterns within web logs to enhance websites from a user perspective.

19) Mining Web Logs: An automated approach

In [106], an automatic approach is proposed for collecting and mining information from web log files formatted according to W3C standards. The innovation lies in optimizing preprocessing, storage, and mining procedures, employing relevant tools for pattern identification. Web logs are analyzed to create adaptive websites for clients, where usage trends are examined, and information is provided for developers. The authors outline the following methodology for analyzing these logs: a) weblog collection, b) storage, c) preprocessing, d) log storage, e) web log mining, and f) ad-hoc analysis. This proposal aims to reduce the time and resources required for web log analysis when implementing data mining techniques.

20) Enabling Process Mining in Airbus

In [107], the article discusses a practical case study of process mining application within Airbus. Process mining is instrumental in analyzing organizational processes through event logs and discovering process models. However, extracting these logs from complex, massive, distributed, and heterogeneous data systems presents a challenge. For effective process mining, data within organizations must be homogeneous and structured. The authors present a case study conducted at Airbus Space and Defence as part of the Clean Sky 2 project: "A-24 One Step Beyond on Automated Testing Technologies." This scenario involves extracting data from the aerospace industry, where event logs (tests and executions) are generated at various workstations and stored in a non-relational database, MongoDB.

The goal is to obtain event logs in a structured XES format, facilitating user log mining. Tools such as XESame and the PromImport plugin enable the extraction and transformation of information from relational databases into event logs. Data extraction from relational databases has been approached through various contexts, including ontology-based data access and the use of metamodels as intermediaries, as demonstrated by Gupta and Sureka. The proposed methodology involves understanding the data, extracting event logs, analyzing the extracted event logs, and analyzing the effects of actions taken.

21) Process Mining: Extending α-Algorithm to Mine Duplicate Tasks in Process Logs

In [108], the article discusses the importance of workflow management in various information systems, emphasizing the need for effective business process models. However, creating and adapting these models to organizational changes can be time-consuming, requiring employee involvement and process documentation review. Process mining, known for extracting structured processes reflecting real workflow executions, typically involves three stages: preprocessing, processing, and postprocessing. In process mining, event logs are fundamental data. However, primary heuristic mining faces challenges such as automatically identifying duplicated tasks due to the complexity of discerning tasks with the same name but in different contexts. To address this challenge, the authors propose an algorithm for modeling duplicated tasks. They achieve this by using heuristic rules and assigning distinct labels to mark tasks with the same name during the data preprocessing stage. Then, during processing, they apply the classical α-algorithm to discover the workflow model. Finally, in postprocessing, they refine the distilled model to recover the original labels of marked tasks, resulting in a workflow model accurately reflecting duplicated tasks.

22) Process Discovery from Dependence- Complete Event Logs

In [109], a new technique leveraging activity dependencies in traces is proposed. Current processes must adapt to changing requirements and environments, but manual modeling or adjustment of processes consumes time and is error-prone. Process mining, or discovery, automates process model extraction by recording process executions as event logs, considering timestamps. Event logs come in various formats and storage methods. The article considers trace extraction from diverse sources beyond its scope. The proposed technique focuses on activity dependencies instead of activity sequences. These dependencies signify real relationships between activities and require activity control, input/output variables. Unlike the α-algorithm, this approach identifies parallel and sequential paths differently. It involves three phases: 1) preprocessing of traces, 2) constructing a dynamic dependency graph, and 3) deriving the transformation process from the dynamic dependency graph.

23) **Enabling Process Mining in Aircraft Manufacturers: Extracting Event Logs and Discovering Processes from Complex Data**

In [110], the article discusses how process mining aids in analyzing organizational processes through event logs and discovering process models. However, extracting these logs is complex due to the massive, distributed, heterogeneous, and intricate nature of data generated by systems.

To apply process mining effectively, the data within organizations must be homogeneous and structured. The goal is to obtain event logs in a structured format called XES, enabling user log mining due to its homogenous and defined structure. Tools like the XESame and PromImport plugin facilitate the extraction and transformation of information from relational databases into event logs. Data extraction from relational databases has been approached through various contexts, such as ontology-based data access and metamodels as intermediaries, as seen in the OnProm Tool. Gupta and Sureka highlight the need for event log extraction and identify a lack of frameworks to facilitate this task. The proposed methodology includes: 1) understanding the data, 2) extracting event logs, 3) analyzing the extracted event logs, and 4) analyzing the effects of actions taken.

24) Data Transformation and Semantic Log Purging for Process Mining

In [111], the article introduces a method to transform audit trail traces collected through ATNA for process mining purposes. Process mining, closely linked with data mining and business process management, aims to utilize information generated by information systems to identify relevant insights. It serves three main objectives: improvement, compliance, and most commonly, discovery. Preprocessing and conversion steps are tailored to the types of traces generated by various information systems, necessitating an analysis for each new implementation case. Integration of healthcare enterprises (IHE) is a professional and industrial initiative seeking to enhance integration and interoperability of hospital information systems (HIS). Given that process mining relies on record data, standard formats like Extensible Markup Language (XML) and eXtensible Event Stream (XES) ensure record uniformity. The proposed approach in the article focuses on preparing IHE integration for compiling HIS for-process mining analysis, with a specific focus on process discovery. This entails transforming ATNA log data into XES format, considering the feasibility of metaobject standards for the transformation.

2.5.2 Comparison of the relevant publications identified

The exploitation of XES traces forms the basis of process mining, which seeks to utilize machine traces to enhance business processes. Furthermore, the focus of this thesis is on improving project management; consequently, it has been decided that the methodologies, processes, or methods of trace transformation relevant to this research will be considered. Various trace transformation processes were identified in this literature review, and are described and compared in Table 5.

Table 5. Publications related to trace transformation.

Article	Domain	Type of data	Data extraction	Source of data	Process	Type of process	Data transformed format	Useful tool identified for transform traces	Software to exploit traces
Odoo module for extracting and transforming event log traces for Process Mining.	Process mining	Interaction traces	Odoo Module	Database	a) Selection of process instances b) Selection of events c) Selection of additional attributes of the events d) Determination of relationships between entities e) Generation of event log in XES format	Trace transformation	XES	1.- XESame 2.- Sauxe	N/A
Uncovering Object-Centric Data in Classical Event Logs for the Automated Transformation from XES to OCEL	Process mining	XES	N/A	Flat event log XES	1) Extraction of actions and types of objects 2) Correspondence between type and identifier 3) Object instance discovery 4) Assignment of property to type 5) Assignment of instance to event 6) Generation of events in OCEL format	Format transformation	OCEL	N/A	N/A
Component for extraction and transformation of event logs from data-aware web systems	Process mining	Interaction traces	N/A	Data base	1) Define the processes by selecting the process instance, 2) Definition of the process activities, 3) Definition of the additional attributes, 4) Capture the events at the moment of their execution, 5) Generate traces in XES format	Trace transformation	XES	N/A	ProM
Quality Evaluation in transformation of event logs into visual representations	Software engineering	Textual event traces	N/A	Database	1) Import logs file in Excel format, 2) Store in database, 3) Visualize the content 4) Assign entry logs to defined situations on the map 5) Geo-tag each event log entry 6) Map log visualization	Trace transformation	Visual representation in the map	N/A	N/A
LogEvent2vec: LogEvent-to-Vector Based Anomaly Detection for Large-Scale Logs in Internet of Things	IoT	Event logs	N/A	BGL data set	1) Log parsing (event log structure) 2) Feature extraction (event log, event log vector, sequence log vector) 3) Anomaly detection	Trace transformation	Visual representation in graphic	Word2vec	N/A
Extracting Event Logs for Process Mining from Data Stored on the Blockchain	Process mining	Process data from an etherium blockchain	N/A	Public ethereum Blockchain	1) Access to etherscan ethereum explorer 2) Analyzing the list of transactions as the sender 3) Event log extraction 4) Log transformation into XES format	Trace transformation	XES	N/A	ProM
Root Cause Analysis with Enriched Process Logs	Process mining	Raw event logs	N/A	Database	1) Determination of response and predictor variables, 2) Enrichment of raw event logs, 3) Case recording, 4) Classification of ready traces, 5) Data mining analysis,	Trace transformation	XES	N/A	ProM
Process Mining by using Event Logs	Process mining	Event log file	N/A	Event log file	1) String tikenizer, 2) Raw detailed filtering 3) Clustering with ActiTrac algorithm 4) Cluster comparison of mining process 5) Mining model generation 6) Prediction generation of mining process models	Process Minning (1 and 2 steps are transformation traces)	Filtered event log file	N/A	ActiTrac

Article	Domain	Type of data	Data extraction	Source of data	Process	Type of process	Data transformed format	Trace transformation Tool	Software to exploit traces
Discovering Web access patterns and trends by applying OLAP and data mining technology on Web logs	Data Mining	Web Logs	N/A	Web logs	1) Data cleansing 2) Replacement of request sequence with a representative URL 3) Attaching tag to event record 4) Storing transformed event logs in database 5) Construction of structured data cube 6) Data mining on web log data cube 7) Data mining on web log database	Process Mining (1, 2, 3, 4 and 5 steps are transformation traces)	Web event log cube	N/A	OLAP
Mining Moodle Logs for Grade Prediction: A methodology walk-through	Data Mining	Interaction traces	N/A	Interaction traces from Moodle	1) Data collection from Moodle platform, 2) Data preparation 3) Statistical data analysis 4) Exploratory data analysis 5) Data mining model generation 6) Model implementation and application of machine learning algorithm to perform prediction	Data Mining (1 and 2 steps are transformation traces)	Tokenized interaction traces	N/A	N/A
Data Transformation and Semantic Log Purging for Process Mining	Process mining	Interaction traces	N/A	SOA Learning Platform	1) Use of DSL to transform interaction traces 2) Semantic purging of traces 3) Mining process 4) Evaluation	Data Mining (step 1 is the process to transformation traces)	XML, XES, CSV and MXML	DSL (domain-specific language)	N/A
Abstracting Process Mining Event Logs From Process-State Data To Monitor Control-Flow Of Industrial Manufacturing Processes	Process mining	Process- state and product-state	Spark	Influx Database	1) Process-State and Product-State Data as Event Log Input 2) Self-Discovering and Event-Log-Aware Digital Control-Flow Twin	Control-flow discovery and monitoring data mixing of process mining	Not specified	N/A	Weppelin and Shiny
From Low-Level-Event-Logs to High-Level-Business-Process-Model-Activities: An Advanced Framework based on Machine Learning and Flexible BPMN Model Translation	Process mining	Interaction traces	N/A	Not specified	1) Generate BPMN-BP Rules for Translation module 2) Uso of rules to automatic labeling logs 3) Training phase 4) Apply machine learning classifier 5) High level logs	Trace transformation	High Level activity logs	N/A	N/A
The onprom Toolchain for Extracting Business Process Logs using Ontology-based Data Access	Process mining	Interaction traces	N/A	Database	1) Create conceptual data schema 2) Create mappings 3) Choose a perspective 4) Create event data annotations 5) Obtain traces in XES or CSV format 6) Apply data mining tool	Trace transformation	XES	ONPROM	ProM
Data mining of search engine logs	Data Mining	Interaction traces	N/A	Web logs	1) Sampling 2) Offline data analysis 3) Boolean phrases and terms 4) Query comparison 5) Query transformation codes 6) Query transformation result	Query transformation	Not specified	N/A	N/A

Article	Domain	Type of data	Data extraction	Source of data	Process	Type of process	Data transformed format	Trace transformation Tool	Software to exploit traces
An experimental mining and analytics for discovering proportional process patterns from workflow enactment event logs	Workflow process discovery	Event traces	N/A	BPI Challenges of 4TU	1) Preparation of workflow enactment event traces 2) Mining proportional process patterns	Workflow discovery	Not specified	N/A	N/A
Mining Console Logs for Large-Scale System Problem Detection	Software engineering	Log file	N/A	Hadoop running in Amazon EC2	1) Syntactic analysis of traces 2) Data structure extraction 3) Construction of vector characteristics 4) Anomaly detection 5) Visualization of detections with a decision tree	Trace transformation	Not specified	N/A	N/A
Web logs pre-processing and analysis: a survey	Web usage mining	Web logs	N/A	Not specified	N/A	N/A	Not specified	N/A	N/A
Mining web logs: an automated approach	Data Mining	Web Logs	N/A	W3C Web log file	1) Web-log collection 2) Web-log storage 3) Log processing 4) Storage of processed logs 5) Web log mining 6) Ad-hoc analysis	Trace transformation	Not specified	N/A	N/A
Enabling Process Mining in Airbus Manufacturing: Extracting Event Logs and Discovering Processes from Complex Data	Process mining	Event logs	N/A	Mongo DB	1) Data understanding 2) Extraction of event logs 3) Analysis of extracted event logs 4) Analysis of effects of measures taken	Trace transformation	XES	Xsame PromImport Onprom	N/A
Process Mining: Extending α-Algorithm to Mine Duplicate Tasks in Process Logs	Process mining	Event logs	N/A	Text file from WFMS of Staff ware	1) Preprocessing 2) Processing 3) post-processing	Trace transformation	Workflow model	N/A	N/A
Process Discovery from Dependence-Complete Event Logs	Process mining	Event logs	N/A	Foodstuff information system used in ports in China	1) Preprocessing of traces 2) Construction of a dynamic dependency chart 3) Derivation of the transformation process from the dynamic graph of dependencies	Trace transformation	XES	N/A	ProM

In Table 5, different types of processes related to traces or computer traces are described: Data Mining, Control flow, Process Mining, and Workflow Discovery. In this research study, processes for trace transformation, which allow the extraction of event logs, were considered relevant. To compare the stages of these processes, only articles addressing trace transformation and event logs were selected. Table 6 provides a comparison of the transformation process proposed in the literature.

Table 6. Trace transformation process comparaison

Article	Stages 1	2	3	4	5	6
Odoo module for extracting and transforming event log traces for Process Mining.	1) Selection of process instances	2) Selection of events	3) Selection of additional attributes of the events	4) Determination of relationships between entities	5) Generation of event log in XES format	
Uncovering Object-Centric Data in Classical Event Logs for the Automated Transformation from XES to OCEL	1) Extraction of actions and types of objects	2) Correspondence between type and identifier	3) Object instance discovery	4) Assignment of property to type	5) Assignment of instance to event	6) Generation of events in OCEL format
Component for extraction and transformation of event logs from data-aware web systems	1) Define the processes by selecting the process instance	2) Definition of the process activities	3) Definition of the additional attributes	4) Capture the events in their execution	5) Generate traces in XES format	
Quality Evaluation in transformation of event logs into visual representations	1) Import logs file in Excel format	2) Store in database	3) Visualize the content	4) Assign entry logs to defined situations on the map	5) Geo-tag each event log entry	6) Map log visualization
LogEvent2vec: LogEvent-to-Vector Based Anomaly Detection for Large-Scale Logs in Internet of Things	1) Log parsing (event log structure)	2) Feature extraction (event log, event log vector, sequence log vector)	3) Anomaly detection			
Extracting Event Logs for Process Mining from Data Stored on the Blockchain	1) Access to etherscan Ethereum explorer	2) Analyzing the list of transactions as the sender	3) Event log extraction	4) Log transformation in to XES format		
Root Cause Analysis with Enriched Process Logs	1) Determination of response and predictor variables	2) Enrichment of raw event logs	3) Case recording	4) Classification of ready traces	5) Data mining analysis	
Process Mining by using Event Logs	1) String tikenizer	2) Raw detailed filtering				
Discovering Web access patterns and trends by applying OLAP and data mining technology on Web logs	1) Data cleansing	2) Replacement of request sequence with a representative URL	3) Attaching tag to event record	4) Storing transformed event logs in database	5) Construction of structured data cube	
Mining Moodle Logs for Grade Prediction: A methodology walk-through	1) Data collection from Moodle platform	2) Data preparation				
Data Transformation and Semantic Log Purging for Process Mining	1) Use of DSL to transform interaction traces					
Abstracting Process Mining Event Logs From Process-State Data To Monitor Control-Flow Of Industrial Manufacturing Processes	1) Process-State and Product-State Data as Event Log Input	2) Self-Discovering and Event-Log-Aware Digital Control-Flow Twin				
From Low-Level-Event-Logs to High-Level-Business-Process-Model-Activities: An Advanced Framework based on Machine Learning and Flexible BPMN Model Translation	1) Generate BPMN-BP Rules for Translation module	2) Use of rules to automatic labeling logs	3) training phase	4) apply machine learning classifier	5) High-level logs	

From Table 6 of trace transformation processes, it can be inferred that there are various approaches and steps involved in the process of transforming event logs. Each article presents a series of specific stages that address everything from data extraction to generating specific record formats for analysis. Some methodologies focus on raw data extraction and processing, such as cleaning and organizing logs into a database, while others concentrate on creating formats tailored for process mining, such as the XES format. Additionally, it is observed that some methodologies incorporate advanced techniques, such as vector-based anomaly analysis or the use of machine learning for log classification. Overall, this table showcases the diversity of approaches and techniques used in the field of trace transformation, highlighting the importance and complexity of this process in data analysis and process mining.

2.5.3 Summary of the classical literature review of the trace transformations process

The classical literature review has allowed the identification of various methodologies, processes, and methods used in trace transformation with applications in different domains such as process mining, anomaly detection, data mining, and software quality management Based on the analysis of the 24 selected articles.

A diversity of approaches and application domains is observed where trace transformation is employed in various sectors ranging from industrial process mining to web log analysis and software quality assessment The identified transformation processes encompass different types of data such as interaction traces, event logs, blockchain data, web logs and educational platform records. Regarding standardization and trace formats the XES (eXtensible Event Stream) format has been established as a widely used standard in the industry for trace exploitation through process mining Other formats such as OCEL (Object-Centric Event Logs) have also emerged to address the transformation of more complex data with relationships among multiple objects.

The identified trace transformation methodologies include structured approaches that consider key stages such as the selection of process instances, event extraction determination of additional attributes and the generation of records in standardized formats Some methodologies incorporate advanced techniques such as automated log classification through machine learning pattern discovery through clustering and semantic log purging. The identified tools and software highlight popular options such as XESame SAUXE ONPROM and ProM for trace transformation and exploitation Complementary technologies such as data processing in relational and NoSQL databases for example MongoDB are also relevant in the capture and organization of traces.

The challenges in trace transformation lie in the need to understand software architecture user permissions and the type of operating system they run on The main challenges include data cleaning, identification of relationships between events, heterogeneity of source formats and automation of transformation processes. Finally regarding the relevance for project management the applicability of trace transformation methodologies in project management remains a little-explored area The approaches identified in the literature can serve as a basis for developing new specific methods to improve project supervision and control.

In conclusion the reviewed literature demonstrates that trace transformation is an essential process for data analysis in various areas However it is necessary to adapt these methodologies for their effective application in project management considering the specific requirements of this discipline.

2.6 Chapter 2 Summary

This chapter provides a comprehensive review of the literature related to maturity models applied to project management, combined with an analysis of prior research conducted at the ERPI laboratory of the Université de Lorraine and an investigation into the use of computer-generated traces to enhance processes. It begins by describing doctoral studies that have addressed, among other topics, the evaluation of maturity criteria in project management, the formalization of organizational processes through the use of traces, the optimization of innovation capacity in small and medium-sized enterprises, and the development of maturity models geared toward digital transformation. These dissertations highlight the significance of having standardized approaches in place and the need to demonstrate a link between maturity levels and project performance or digital adoption, all aimed at achieving ongoing improvements within organizations.

Regarding the identified maturity models, the discussion includes classical proposals such as CMM and P-CMM—originally intended to evaluate software development processes and talent management—as well as quality-focused frameworks like EFQM, whose scope extends toward business excellence. The chapter also examines project-specific models like OPM3, P3M3, and PMMM, which offer tools for assessing and quantifying maturity in processes, portfolios, and programs. Although these models typically present a progressive path across levels or stages of maturity, along with best-practice recommendations and, sometimes, guidelines for implementation, most rely on subjective evaluations carried out through questionnaires or expert interviews. Consequently, they often fail to provide in-depth details on how to deploy best practices within the company or how to verify, via tangible evidence, the level of compliance at each stage. This lack of objective guidance is even more pronounced for SMEs, which, by nature, face resource constraints and challenges in adopting complex models.

To address this gap and supply more objective solutions, the chapter explores various research initiatives that have leveraged computer-generated traces (often referred to as event logs or interaction records). These traces have been particularly useful in fields such as online education, knowledge management, and process mining. In these contexts, different methodologies have been proposed to capture, transform, and visualize the information flowing through digital tools, aiming to detect patterns, provide recommendations, or even conduct predictive analyses. It becomes apparent that the XES standard has emerged as a key format for the extraction of event logs in process mining environments, and that multiple software applications (XESame, ProM, among others) facilitate the conversion of raw data into suitable structures for subsequent analysis. Despite these substantial technological advances, the literature has yet to explore the use of traces for measuring and analyzing project management maturity, revealing a clear research opportunity.

This review underscores the benefits of integrating maturity models with automatic trace collection for evaluating the level of maturity, thereby reducing the cost of maturity assessments typically characterized by questionnaires. The present doctoral research addresses this approach, which precisely outlines how relevant trace actions are captured, how the data is transformed to align with a given maturity model, and how the resulting analysis helps to measure the maturity level automatically. The trace analysis integration has the potential to enhance the effectiveness and reliability of maturity evaluations, while also laying the groundwork for a more dynamic and automated strategy in project management.

2.7 Chapter 2 Discussion

The discussion section in Chapter 2 are synthesized and critically analyzed, addressing a total of five key issues that guide the evolution of maturity models in project management and the potential use of computational traces for their evaluation. The issues considered after the analysis are described below:

Evolution, Diversity, and Complexity of Maturity Models
It has been identified that maturity models initially designed for software development process evaluation (e.g., CMMI) have been adapted to the field of project management, leading to specific proposals such as OPM3, P3M3, and PMMM. The main issues in this area include variations in the number of maturity levels, diversity in evaluation criteria and best practices, complexity in their application, and the need to tailor these models to heterogeneous organizational contexts, particularly small and medium-sized enterprises (SMEs) with limited human, financial, and knowledge resources. Beyond these challenges, one potential solution emerges: the adoption of new approaches for maturity assessment, such as the use of computational traces and the implementation of conceptual guidelines. These guidelines would help stakeholders understand the foundational knowledge necessary to comprehend the structure and approach of a maturity model, thereby facilitating its adoption.

Subjectivity in Traditional Assessments: Current maturity models heavily rely on expert interviews and questionnaires, leading to evaluations based on personal perceptions and, in some cases, subjective interpretations of best practice implementation. The main issues in this regard include the lack of objective mechanisms to verify compliance with recommended practices and the absence of clear guidelines to support their implementation within organizations. Additionally, traditional questionnaires often function as a test of definitions rather than an assessment of practitioners' knowledge about daily activities within processes. An alternative approach to address this subjectivity would be the use of computational traces, which could provide a more accurate and reliable reflection of process activities without relying on subjective self-assessments.

Understanding Best Practices: Best practices in any domain are typically defined based on a process reference model or a consensus-driven guide developed by experts. However, these references do not always specify what these practices entail, the process required for their implementation, or the criteria for their completion. In other words, many frameworks lack a precise description of the steps needed to systematically execute best practices within an organization. To enhance maturity model assessments, evaluations should focus on analyzing the execution of processes and sub-processes necessary to achieve best practices, rather than merely verifying their existence as an end result. In summary, best practices should be understood as deliverables resulting from a structured set of activities aimed at achieving a specific objective, rather than as abstract concepts without a defined implementation process.

Lack of Explicit Trace Transformation Processes: Although numerous studies discuss the transformation of computational traces, there is a noticeable absence of standardized processes and clearly defined stages for carrying out this transformation. Many studies describe data extraction and final usage but fail to detail the step-by-step procedure, making it difficult to replicate results and compare findings across research efforts. This lack of explicit processes highlights the need for a unified framework that establishes clear stages—from data collection and cleaning to transformation into standardized formats—to optimize the analysis and application of traces in maturity model evaluations across different domains, particularly in project management.

Implications for Project Management in SMEs Regarding Maturity Models: Considering that SMEs constitute the majority of the global business landscape and often lack the resources to implement complex evaluations, it is crucial to develop new approaches to maturity models that incorporate computational trace-based assessments. The absence of such approaches limits the usability of maturity models, potentially hindering the development of SMEs. However, this challenge is not uniform worldwide. In some countries, SMEs receive government support for their development, yet in these contexts, the lack of fundamental knowledge required to implement maturity models presents another significant limitation. Addressing these gaps through computational trace analysis and knowledge-based guidance could enhance the applicability and accessibility of maturity models for SME

Finally, the following section describes three chapters focused on the main contributions of this thesis work, the Cone-based conceptual network, the trace transformation process and a process to analyze the feasibility of tools to assess maturity.

Part 2: Contributions

Introduction to part 2:
The second part of the doctoral thesis addresses the fundamental contributions of three key areas: project management, maturity models, and traceability in computing. Its primary objective is to present the conclusions derived from research focused on integrating traceability as an evaluative factor in project management maturity models. In the third chapter, a conceptual maturity network is presented, which is specifically applied in this context to address the challenges inherent in implementing such models. This approach seeks to clarify the complexities and challenges associated with the practical application of these models. The fourth chapter proposes a detailed process for transforming software traces, aiming to reduce their granularity level and describe user actions in terms of domain activities. This methodology aims to optimize the utility of traces as an evaluative tool. Finally, the fifth chapter focuses on specifying a set of traceable functionalities that enable a comprehensive evaluation of maturity in project management. This step is crucial for ensuring a thorough and accurate evaluation of maturity in this field, using traceability as the main guiding principle.

Chapter 3. Cone-based conceptual network for an maturity assessment mission (Auditor) in engineering activities supported by software traces.

In this chapter, the complexity of implementing maturity models points to resources and knowledge as two relevant limitations for their use within organizations. This complexity has been taken into consideration to generate a useful proposal that allows for the evaluation of maturity, considering the traces generated by domain activity software. Regarding knowledge, a conceptual framework has been developed that enables the scientific community and individuals interested in conducting a maturity assessment to understand the scope to be considered for its use. In terms of resources, the development of a tool that automatically collects usage traces and transforms them into user actions to evaluate best practices and identify a level of maturity has been pursued. This chapter is organized as a follow: In Section 3.1, the Cone-based Conceptual Network (CBCN) is presented with a general approach, describing the traditional approach and the innovative approach proposed involving the use of traces. Section 3.2, delves into the semantic intricacies of the aforementioned general CBCN model. The validation of the general CBCN is detailed in Section 3.3 through its implementation in the NIAM method. Section 3.4 details the application of the proposed CBCN to a general approach to project management. Finally, in Section 3.5, the expansion of CBCN applied to PM is described. Also a complementary sections: 3.6 Chapter 3 Summary and 3.7 Chapter 3 Discussion are addressed in the end of this chapter.

3.1 A general CBCN for an maturity assessment mission definition

This section introduces the cone-based conceptual network with a general approach applicable to engineering activities. The mapping of this model allows for the description of key concepts and their relationships involved in building the cone network. This cone network offers stakeholders a conceptual view for understanding the context when conducting a maturity assessment, along with a novel evaluation approach by leveraging traces from a domain activity management software. The model defined is more restrictive than that of mind or concept mapping, but less restrictive than that of formal ontology [112], even if concepts and associations form the core of the "conceptual structure" [113] modeled.

3.1.1 A Traditional Process Maturity Models Network

There are conceptual requirements for satisfactorily implementing Traditional Process Maturity Models (TP2Ms) [13], [14]. One of them is the existence of different scopes to be considered to extract meaningful information for process maturity assessment. A first collection of scopes can include three items: the organization scope, the domain scope, the reference scope, and the maturity assessment scope. Each scope refers to an issue: in which organization is the process studied? What is the domain of concern of this process? What is the reference, i.e. standardized descriptive and procedural knowledge codifying the habits about this domain? How to assess maturity? Unlike OPM3™, these scopes are not juxtaposed: they are mapped out. That means that their interaction is a key issue in an audit or assessment mission. Furthermore, it is suggested to qualify each scope (node) with a property called granularity. [114] defines the granularity as the level of detail in the description of the system studied. A scope can be then described with a low, medium, or high granularity, explaining why these are called nodes: conceptual cones.

Starting from the lowest granularity, a conceptual boundary is established to outline the scope being described. Medium granularity bounds a specific concept. High granularity indicates a main scope. This type of model is reminiscent of Charles Sanders Peirce's (1839-1914) ''existential graphs''[115] or John Venn's (1834-1923) diagrams [116] depicting inclusion between sets [113]. However, in contrast to the works of these prestigious authors, the nests between granularity levels are only declarative: they do not detail logical inferences between knowledge of different levels of granularity, e.g. instantiation or induction. Lastly, nodes relate to bidirectional relations, e.g., scope A describes scope B, which is similar to scope B described by scope A. NIAM uses this type of label more flexibly than that of formal ontologies or UML structure diagrams.

Having underlined these formal remarks, four scopes have been selected, bearing in mind that this list is not exhaustive. The organization scope contains the identity of the organization of interest, its internal context, and then the business capabilities related to the process studied. This nest enables us to situate a capability: it does not exist in a vacuum and can only be understood in a specific organizational context. This first conceptual cone describes the context-dependent nature of capability. The domain scope is the second conceptual cone packaging the domain process, which is all the elements (process and sub-processes elements) interacting in a certain domain and the specific activities that can be non-standardized tasks or, if they are, that can be compared to best practices. The domain scope encompasses the processes performed by organizations, including specific activities within the domain. The third scope concerns maturity assessment, which requires specific roles, e.g., external, or internal auditors, tools and documents, interviews and information transmitted orally, files, and more generally data as raw materials. The outcome of this assessment activity categorizes the processes assessed into a certain level of maturity. The scope of maturity assessment encompasses the entire assessment process, leading to the assignment of a maturity level. The fourth and last conceptual cone describes the type of knowledge standard describing the specific best practices to implement to achieve organizational goals. However, in some cases, references may not exist, or the best practices of the domain may not be codified or published. Table 7 synthetizes the scopes (conceptual cones) of the T2PMs conceptual network.

Table 7. Scopes of TP2Ms' Conceptual Network

TP2Ms Scope		
Cone's name	Descriptive issue	Content
Organization Scope	What is the contingent organizational situation in which capabilities driving the performance of the process studied are effective?	Low granularity: organization Medium granularity: process context High granularity: capability
Domain Scope	What is the domain of the process studied?	Low granularity: domain Medium granularity: domain process High granularity: domain activities
Maturity Assessment Scope	How can one assess the maturity of the process studied?	Low granularity: maturity assessment Medium granularity: maturity assessment mission High granularity: process maturity level
Referential Scope	What is the reference of the process studied?	Low granularity: referential scope Medium granularity: process referential High granularity: domain best practices

It is now necessary to type de relations between scopes (cones). To this end, it is suggested to build the conceptual network displayed in Figure 15. This network is connected: no scope (cone) is excluded, and all scopes are accessible, even at different levels of granularity. This graph can be interpreted as follows: TP2Ms are focused on capabilities that drive the operational performances of the specific process studied and that can be compared to a collection of best practices. The graph reveals the nodal aspect of the conceptual cone called "domain scope". There is a certain epistemic inconsistency in the graph representing TP2Ms as far as the levels of granularity considered are not the same. Moreover, lack of data or knowledge in any of the identified scope (cone) impedes assessment mission progress.

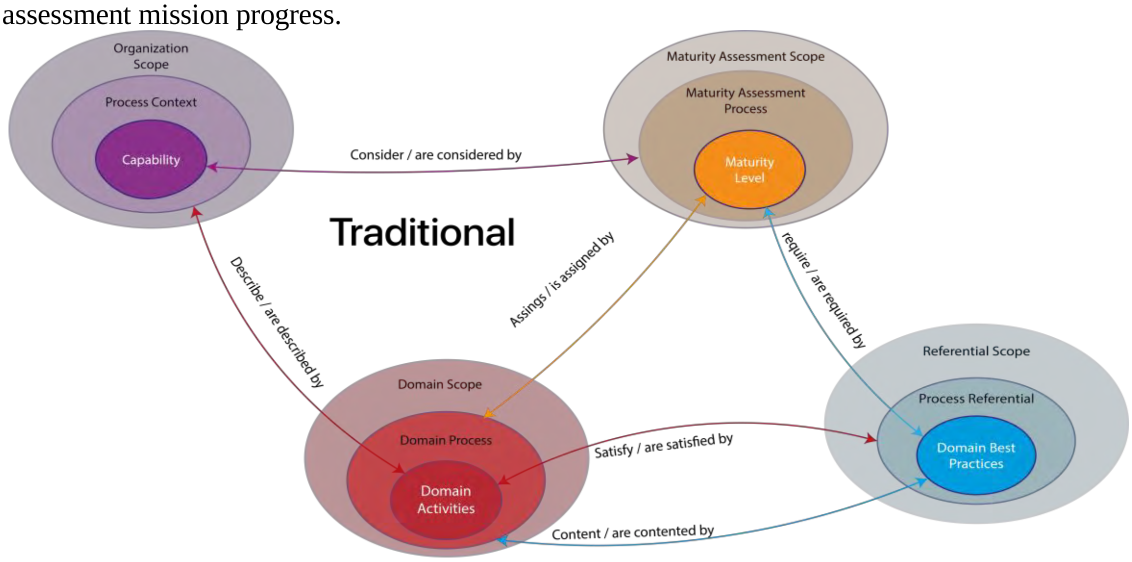


Figure 15. TP2Ms Cones-based Conceptual Network.

Although Figure 15 maps the traditional process of maturity models, it remains complex for some organizations due to the significant resource requirements for implementation. This particularly affects small and medium-sized enterprises, which, lacking these resources, are unable to leverage traditional maturity models. For this reason, an extension of TP2Ms has been developed involving four ITs-focused conceptual cones.

3.1.2 An IT-focused extension of Conceptual Network

The proposed extension adds four IT-focused conceptual cones referring to the domain tool scope, the data processing scope, the knowledge management scope, and the trace-based maturity evaluation scope. This extension enables the creation of the Computer-based Process Maturity Models CP2Ms' CBCN. Since this graph concerns potential TBSs, the first aspect of the new network pertains to data, while the second facet refers to knowledge about the activity of interest.

The domain tool scope is based on a simple hypothesis: in ITs-intensive organizations, software tools integrate functionalities used by actors to perform their activities, and as a result, traces are generated. These traces generated with administrative, or management software allow for the retrieval of users' activities in the IS These traces exhibit a fine granularity, which makes their semantic analysis too difficult, as underlined in the introduction. On the other hand, the data processing scope deals with the abundance of detailed traces that complicate TBSs' functioning. Thus, an algorithm or process is required to transform these details into traces which is based on the knowledge exploited in assessment missions. The transformation aims to decrease the granularity of traces to a point where it is easy to identify the tasks users perform with the software.

CP2Ms require not only data and data processing but also new knowledge programmed or not in software. The alternative to TP2Ms is a support tool allowing the retrieval of the traces transformed by the TBS. In this type of trace, it is possible to visualize users' activities in terms of the domain activities. The scope-called trace-based maturity evaluation enables the determination of maturity based on an assessment element. Under TP2Ms this element is the survey responses, and in CP2Ms it is the transformed traces. The other knowledge-focused is the knowledge management tool scope, and more specifically the formal ontology [117] of the domain of interest. This scope recognizes the relevance of having enough knowledge of the domain to create a domain ontology. Building a specific ontology is a precondition to conducting the transformation of traces. Table 8 synthetizes the scopes (conceptual cones) of the CP2Ms conceptual map.

Table 8. Proposed Scopes of CP2Ms' Conceptual Network

TP2Ms' scopes and CP2Ms' Scopes		
Cone's name	**Descriptive issue**	**Content**
Domain tool scope	Are there sufficient software functionalities to perform the domain activities? Do all functionalities generate traces of each domain activity performed?	Low granularity: domain tool Medium granularity: functionalities support usage High granularity: raw trace
Data processing scope	Does the trace transformation process allow us to visualize users' activities in terms of domains?	Low granularity: data processing Medium granularity: trace transformation process High granularity: transformed trace
Trace-based maturity evaluation tool scope	Is it possible to assess domain maturity based on traces transformed into domain activities?	Low granularity: trace-based maturity evaluation tool Medium granularity: computing evaluation High granularity: evaluation element
Knowledge management scope	Is there sufficient domain knowledge to generate an ontology that describes the domain evaluated?	Low granularity: knowledge management Medium granularity: knowledge management process High granularity: domain ontology

Figure 16 describes the conceptual network of CP2Ms. New connections are defined between the new scopes (conceptual cones) and the mapping with nodes referring to TP2Ms is achieved in several paths. As in the case of the graph in Figure 15, there is a degree of epistemic inconsistency. The new cones are linked to a high granularity, whereas the support tool must provide less granular knowledge.

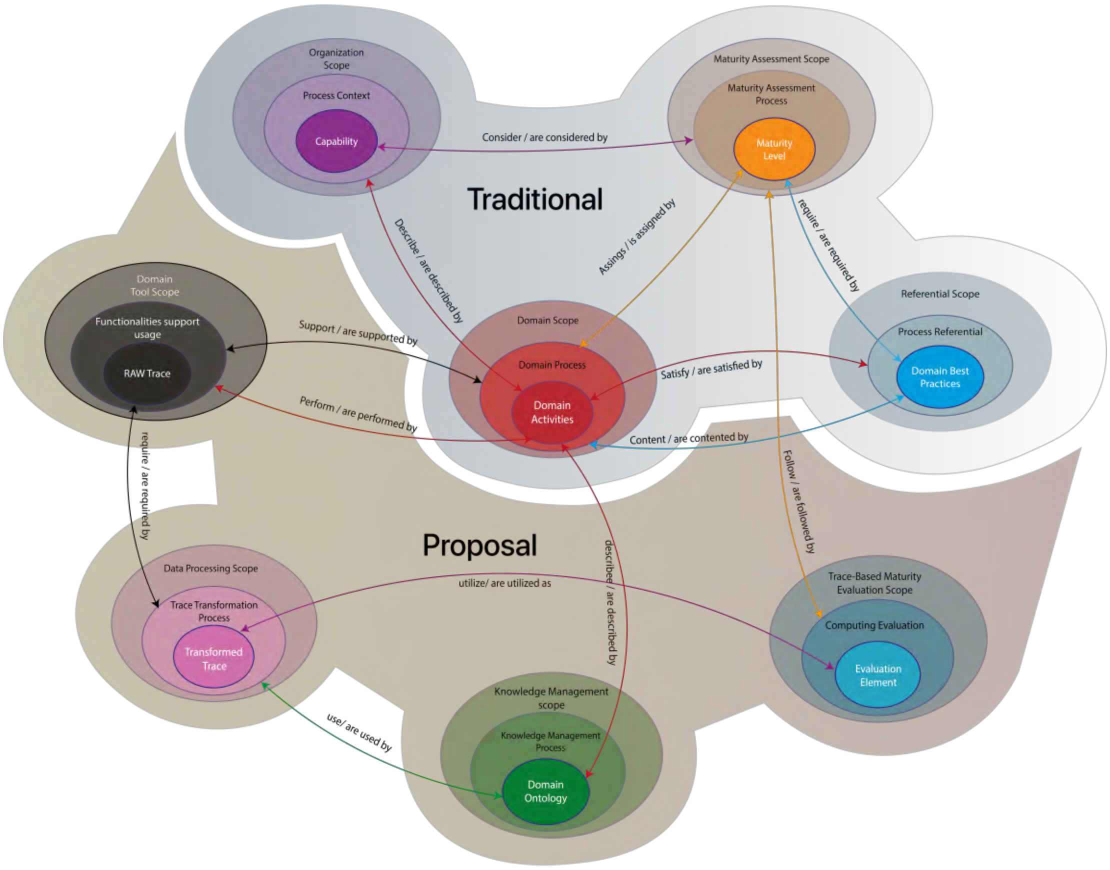


Figure 16. CP2Ms Cones-based Conceptual Network.

Conceptual graphs of all kinds may fascinate, but their consistency must be verified in detail. To achieve this, it is suggested to translate the network shown in Figure 16 into Nijssen's Information Analysis Methodology (NIAM) schema. This translation will facilitate a thorough analysis, allowing for the identification of inconsistencies and ensuring that the relationships within the graph are logical and coherent. By employing a NIAM schema, the clarity and reliability of the conceptual network can be significantly enhanced, leading to a deeper understanding of the represented knowledge.

3.2 The Semantics of a General CBCN for Maturity Assessment Missions

To fully comprehend the semantics of this Conceptual Based Classification Network (CBCN), it is essential to recognize that the connections among the various scopes were developed subsequent to the conceptualization of each individual scope. In the previous section, the elements involved in the model's construction were detailed and graphically represented. Therefore, this section will commence with a thorough exploration of the semantics related to the construction of a scope, addressing the fundamental principles and considerations that guided this process. This examination will provide a foundation for understanding how each scope is defined and how its specific characteristics contribute to the overall structure of the CBCN.

3.2.1 Scopes Semantics

Each area has ovals of different sizes, each with a specific semantic meaning. The largest oval delineates the general area in which knowledge is required. The medium oval delineates the specific process within that scope in which knowledge is required. The smallest oval indicates the key concept of the process, from which, with a basic understanding, it is possible to understand the rest of the ovals. This idea leads to the interpretation of the cones: as the oval gets larger, the knowledge becomes more general and harder to understand by itself, but if there is an understanding of the rest of the ovals, it is possible to better grasp the overall scope. In addition, the connections between the ovals help identify the relationship between the scope, process, and key concept. Figure 17 outlines the questions that should be asked to construct each scope proposed in the cone-based conceptual network.

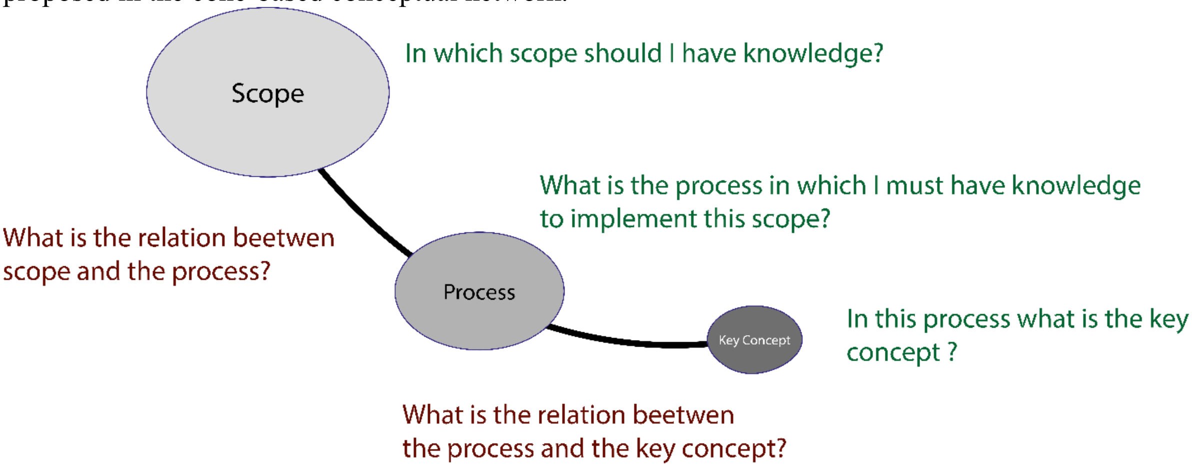


Figure 17. Questions to build a scope from CBCN

3.2.2 Relation Scopes Semantics

In the conceptual model, the different scopes represent basic areas where knowledge is required to implement traditional maturity models and trace-based models. The links identify the relationship and importance of each element within the conceptual network, showing their specific contribution. These links are represented by arrows indicating a cardinal relationship between two elements (process or key concept).

Types of cardinal relationships

The links are generated based on the scopes and their relationship with other elements. In general terms, we can identify that the model describes the following types of cardinal relationships: usability (use), utility (utilized), requirement (required), performance (perform), support (support), description (describe), consideration (consider), assignment (assign), satisfaction (satisfy), and content (content). These types of relationships are used to connect different types of elements: processes to processes, key concepts to processes, and key concepts to key concepts.

Types of elements connected

Processes to Processes: Processes can be interconnected through any of the relationships, depending on how they interact and depend on each other. Key Concepts to Processes: A key concept can be essential for a specific process, whether as a requirement, support, or any other form of cardinal relationship. Key Concepts to Key Concepts: Key concepts can also be related to each other, forming a knowledge network that supports the processes and objectives of the maturity model.

Flexibility of the model

It is important to note that the model does not standardize a specific type of link or cardinal relationship between elements. Instead, it emphasizes that in general, scopes, processes, and key concepts should be linked through a cardinal relationship that facilitates contextual understanding, considering the types of relationships described previously.

3.3 CP2Ms validation with NIAM

In the literature, the validation of conceptual models is recognized through techniques (Object Relational Mapping) such as verbalization and population [118]. Verbalization involves describing the rules or constraints of a model using natural language, allowing technical concepts to be expressed in a way that domain experts can easily understand and validate. Population, in this context, refers to the use of example data or instances to demonstrate how these rules or constraints are applied in practice. This approach involves reviewing the model with users to ensure it accurately reflects the reality in the domain of application. NIAM provides verbalization of objects and their relationships by generating relational statements.

3.3.1 NIAM

The goal of NIAM is to express in an understandable manner complex ideas only through formal expressions or phrases [119]. These expressions contain a subject, verb, and complement. NIAM enables graph modeling associating concepts, their relationships, and whether they have any constraints or mandatory indicators. While there is no maximum limit to the number of concepts to be modeled using NIAM, there is a restriction indicating that a minimum of two linked concepts must be analyzed. From these concepts, a network can be generated, wherein a conceptual link is also mandatory. The phrase container describes the relationship between the concepts. Figure 18 displays NIAM's building block.

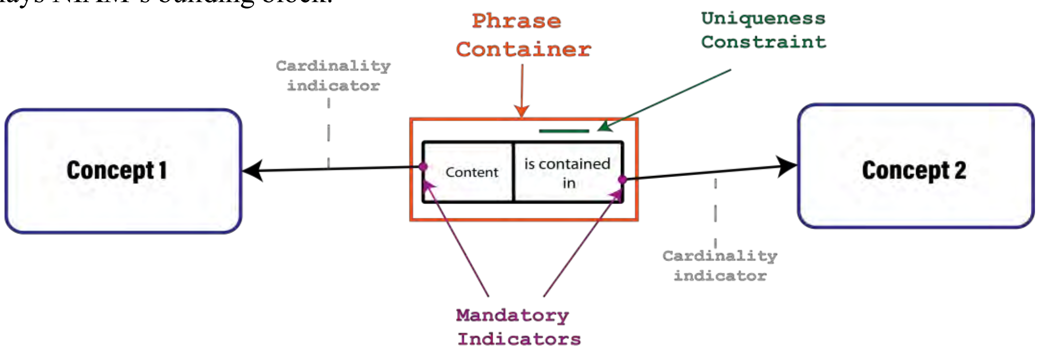


Figure 18. NIAM's Building Block.

The uniqueness constraint allows indicating the type of relationship between concepts, that is, whether the relationship is one-to-many or one-to-one (cardinality). This constraint is graphically represented with a purple-colored line within the phrase container, for example, in a "Customer" entity, the uniqueness constraint on the "Customer Number" attribute would ensure that each customer number is unique in the database. This helps to ensure data integrity and prevent inconsistencies. On the other hand, the mandatory indicator shows that at least one of the concepts must exist in the phrase container. In Figure 18 this is described like a point purple. Once a NIAM graph building block is presented, a literal tool is used to show the explicit relationships between the concepts of interest. A tool is presented in Figure 19. In our case, this instrument was Microsoft Visual Studio Community Version 2022 with the Natural Object-Role Modeling (ORM) Architect module version 2022. The functionality that allowed for the generation of relational statements was the ORM Verbalization Browser.

Concept 1 Content Concept 2.
Each Concept 1 Content **some** Concept 2.
Each Concept 2 is contented in **exactly one** Concept 1.
It is possible that some Concept 1 Content **more than one** Concept 2.

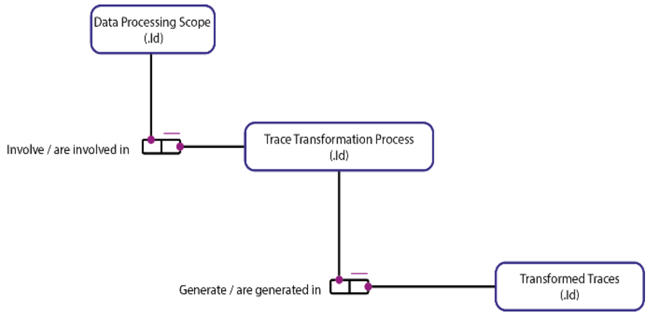
Figure 19. Relational statements of the example

Once we have outlined NIAM, it is necessary to explain how we have used it in the case of C2BMs cone-based conceptual network.

3.3.2 NIAM implementation

Conceptual boundaries validation

The implementation of NIAM was conducted with two analyses, the first to validate the relationship between the conceptual boundaries in each scope (conceptual cone) and the second to validate the relationship between the scopes (conceptual cone). To illustrate the first validation analysis, the data processing domain was selected, which is one of the cones of the CP2Ms conceptual network (Figure 16). The graph and the statements generated in the validation are described in Figure 20.



Data Processing Scope Involve Trace Transformation Process.
- **Each** Data Processing Scope Involve **some** Trace Transformation Process.
- **Each** Trace Transformation Process are involved by **exactly one** Data Processing Scope.
- **It is possible that some** Data Processing Scope Involve **more than one** Trace Transformation Process.

Trace Transformation Process generate Transformed Traces.
- **Each** Trace Transformation Process generate **some** Transformed Traces.
- **Each** Transformed Traces are generated by **exactly one** Trace Transformation Process.
- **It is possible that some** Trace Transformation Process generate **more than one** Transformed Traces.

Figure 20. Example of NIAM Translation of TP2Ms Scopes (Cones).

Figure 20 details the implementation of the data processing scope in the nodes of the CBCN proposed in NIAM. In the red-colored box located on the left, the conceptual relationship, uniqueness constraints, mandatory indicators, and the phrase describing that conceptual relationship in cardinality form are specified. This scope not only explains the concepts contained within it but also their relationships. In summary, this red box indicates that the data processing scope involves a trace transformation process resulting in transformed traces. On the right side the blue-colored box describes the relational statements on the right verbalize the connections depicted in the red box, allowing for a textual explanation of the conceptual connections among the elements modeled in the data processing scope of the CBCN.

The statements generated by this scope indicate that each data processing scope involves the implementation of a trace transformation methodology, indicating a dependency between the two. This relationship describes that each trace transformation methodology is associated with only one data processing scope at a time. However, some data processing scopes may involve multiple trace transformation methodologies, implying variability in their application. Furthermore, each trace transformation methodology produces transformed traces, and each transformed trace is produced by a methodology, indicating a clear mapping between them. Some methodologies may produce multiple transformed traces, implying a potential variety of results. In the example, the explicit articulation of the conceptual relationships and relational statements allows verification that the conceptual relationship boundaries in the data processing area have been correctly generated in NIAM, as no errors have been identified in the system to indicate any inconsistencies.

Scopes relationship validation

This second analysis to validate the relationship between the scopes makes it possible to understand and identify how each of the scopes interacts within the context of the maturity models applied in a domain, considering the traces generated in domain activity management software. While the conceptual relationships between the elements of each scope have been formalized, it is also necessary to formalize the relationships between the scopes. To describe the existing relationships described in the C2PM's cone-based conceptual network (Figure 16), Table 9 was created. This table was used to model the conceptual relationships of the different scopes in the

Visual Studio tool (Figure 21). Once the model was generated, the Verbalization Browser ORM functionality was used to generate relational statements of the concepts (Figure 22).

Table 9. Conceptual interactions between each scope

Scopes Interacted		Interaction elements		Conceptual Interaction
Organization Scope	Maturity Assessment Scope	Capability	Maturity assessment process	Consider
Organization Scope	Domain Scope	Process context	Domain activities	Describe
Domain Scope	Maturity Assessment Scope	Domain process	Maturity level	Assigns
Domain Scope	Referential Scope	Domain activities	Process Referential	Satisfy
Domain Scope	Knowledge Management Scope	Domain activities	Domain ontology	Describe
Domain Scope	Referential Scope	Domain activities	Process referential	Satisfy
Domain Scope	Referential Scope	Domain process	Domain best practices	Content
Domain Tool Scope	Domain Scope	Functionalities Support Usage	Domain activities	Perform
Domain Tool Scope	Data Processing Scope	Raw traces	Trace transformation process	Require
Data Processing Scope	Knowledge Management Scope	Trace transformation process	Domain ontology	Use
Trace-Based Maturity Evaluation Scope	Data Processing Scope	Assessment element	Transformed traces	Utilize
Trace-Based Maturity Evaluation Tool Scope	Maturity Assessment Scope	Computing Evaluation	Maturity assessment process	Follow
Maturity Assessment Scope	Referential Scope	Maturity level	Domain best practices	Require

The complexity of modeling conceptual relationships among different scopes (Cones) is described in Figure 21. Initially, it is possible to observe that the domain scope exhibits multiple relationships with others, becoming a central scope that enables interaction among the rest of the scopes within the network. The domain scope has a total of seven different relationships with the other scopes, making it the most influential in the network.

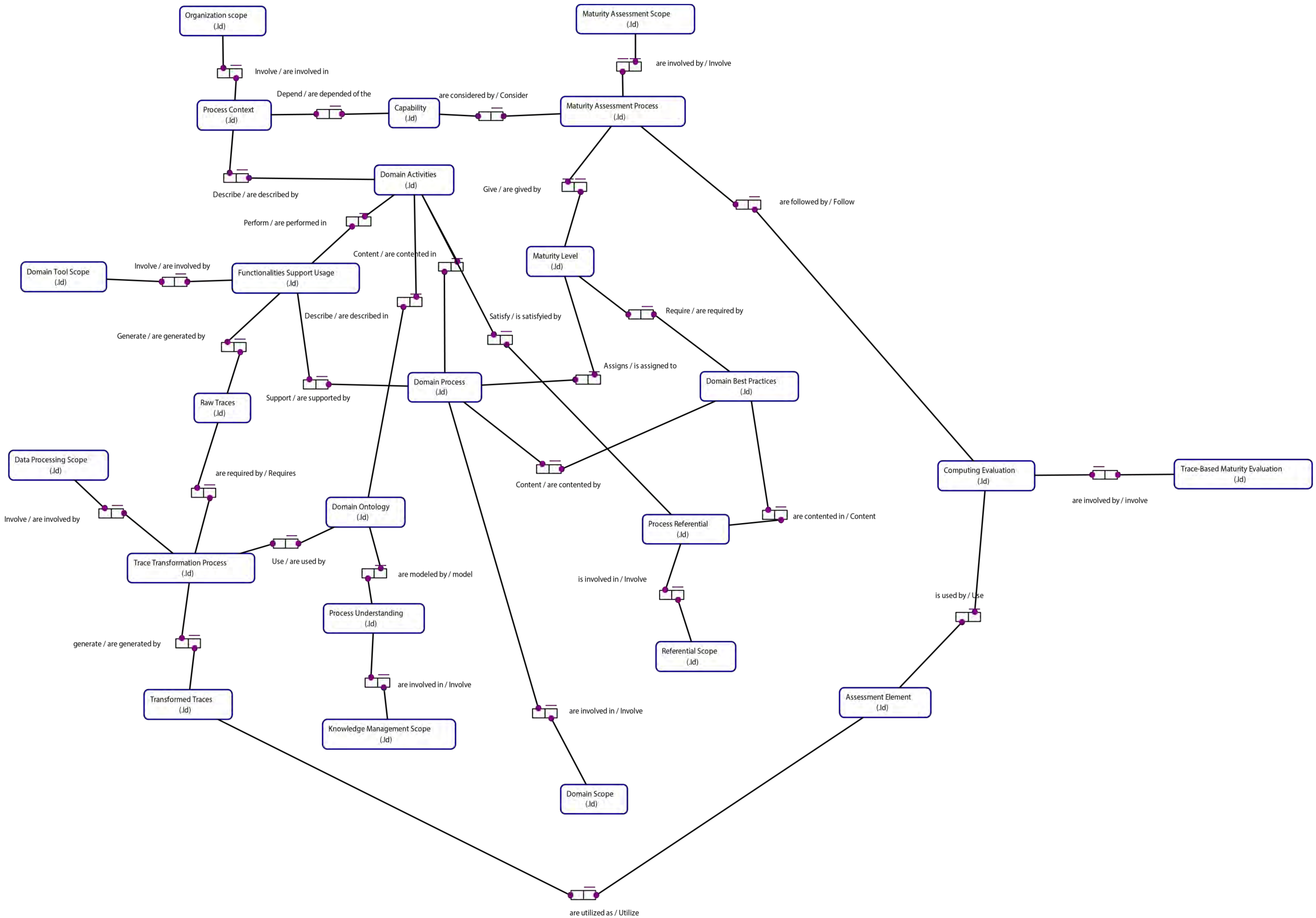


Figure 21. NIAM implementation of the CP2Ms Conceptual Network.

Capability are considered by Maturity Assessment Process.
Each Capability are considered by some Maturity Assessment Process.
Each Maturity Assessment Process Consider exactly one Capability.
It is possible that some Capability are considered by more than one Maturity Assessment Process.
Maturity Assessment Process Give Maturity Level.
Each Maturity Assessment Process Give exactly one Maturity Level.
Each Maturity Level are gived by exactly one Maturity Assessment Process.
Maturity Assessment Process are involved by Maturity Assessment Scope.
Each Maturity Assessment Process are involved by exactly one Maturity Assessment Scope.
Each Maturity Assessment Scope Involve exactly one Maturity Assessment Process.
Capability are depended of the Process Context.
Each Process Context Depend some Capability.
Each Capability are depended of the exactly one Process Context.
It is possible that some Process Context Depend more than one Capability.
Organization scope Involve Process Context.
Each Organization scope Involve some Process Context.
Each Process Context are involved in exactly one Organization scope.
It is possible that some Organization scope Involve more than one Process Context.
Domain Process Content Domain Activities.
Each Domain Process Content some Domain Activities.
Each Domain Activities are contented in exactly one Domain Process.
It is possible that some Domain Process Content more than one Domain Activities.
Domain Process are involved in Domain Scope.
Each Domain Process are involved in some Domain Scope.
Each Domain Scope Involve exactly one Domain Process.
It is possible that some Domain Process are involved in more than one Domain Scope.
Process Context Describe Domain Activities.
Each Process Context Describe some Domain Activities.
Each Domain Activities are described by exactly one Process Context.
It is possible that some Process Context Describe more than one Domain Activities.
Domain Process Assigns Maturity Level.
Each Domain Process Assigns some Maturity Level.
Each Maturity Level is assigned to exactly Trace Transformation Process Use Domain Ontology.
Each Trace Transformation Process Use some Domain Ontology.
Each Domain Ontology are used by exactly one Trace Transformation Process.
It is possible that some Trace Transformation Process Use more than one Domain Ontology.
Domain Ontology Describe Domain Activities.
Each Domain Ontology Describe some Domain Activities.
Each Domain Activities are described in exactly one Domain Ontology.
It is possible that some Domain Ontology Describe more than one Domain Activities.
Assessment Element is used by Computing Evaluation.

one Domain Process.
It is possible that some Domain Process Assigns more than one Maturity Level.
Domain Best Practices are contented in Process Referential.
Each Domain Best Practices are contented in some Process Referential.
Each Process Referential Content exactly one Domain Best Practices.
It is possible that some Domain Best Practices are contented in more than one Process Referential.
Process Referential is involved in Referential Scope.
Each Process Referential is involved in some Referential Scope.
Each Referential Scope Involve exactly one Process Referential.
It is possible that some Process Referential is involved in more than one Referential Scope.
Maturity Level Require Domain Best Practices.
Each Maturity Level Require some Domain Best Practices.
Each Domain Best Practices are required by exactly one Maturity Level.
It is possible that some Maturity Level Require more than one Domain Best Practices. Domain Activities Satisfy Process Referential.
Each Domain Activities Satisfy some Process Referential.
Each Process Referential is satisfyied by exactly one Domain Activities.
It is possible that some Domain Activities Satisfy more than one Process Referential. Domain Process Content Domain Best Practices.
Each Domain Process Content some Domain Best Practices.
Each Domain Best Practices are contented by exactly one Domain Process.
It is possible that some Domain Process Content more than one Domain Best Practices.
Functionalities Support Usage Generate Raw Traces.
Each Functionalities Support Usage Generate some Raw Traces.
Each Raw Traces are generated by exactly one Functionalities Support Usage.
It is possible that some Functionalities Support Usage Generate more than one Raw Traces.
Domain Tool Scope Involve Functionalities Support Usage.
Each Domain Tool Scope Involve some Functionalities Support Usage.
Each Functionalities Support Usage are involved by exactly one Domain Tool Scope.
It is possible that some Domain Tool Scope involve more than one Functionalities Support Usage. Each Assessment Element is used by some Computing Evaluation.
Each Computing Evaluation Use exactly one Assessment Element.
It is possible that some Assessment Element is used by more than one Computing Evaluation.
Transformed Traces are utilized as Assessment Element.
Each Transformed Traces are utilized as some Assessment Element.
Each Assessment Element Utilize exactly one Transformed Traces.
It is possible that some Transformed Traces are utilized as more than one Assessment Element.
Maturity Assessment Process are followed by Computing Evaluation.
Each Maturity Assessment Process are followed by some Computing Evaluation.
Each Computing Evaluation Follow

Functionalities Support Usage Support Domain Process.
Each Functionalities Support Usage Support some Domain Process.
Each Domain Process are supported by exactly one Functionalities Support Usage.
It is possible that some Functionalities Support Usage Support more than one Domain Process.
Functionalities Support Usage Perform Domain Activities.
Each Functionalities Support Usage Perform some Domain Activities.
Each Domain Activities are performed in exactly one Functionalities Support Usage.
It is possible that some Functionalities Support Usage Perform more than one Domain Activities.
Data Processing Scope Involve Trace Transformation Process.
Each Data Processing Scope Involve some Trace Transformation Process.
Each Trace Transformation Process are involved by exactly one Data Processing Scope.
It is possible that some Data Processing Scope Involve more than one Trace Transformation Process. Trace Transformation Process generate Transformed Traces.
Each Trace Transformation Process generate some Transformed Traces.
Each Transformed Traces are generated by exactly one Trace Transformation Process.
It is possible that some Trace Transformation Process generate more than one Transformed Traces. Raw Traces are required by Trace Transformation Process.
Each Raw Traces are required by some Trace Transformation Process.
Each Trace Transformation Process Requires exactly one Raw Traces.
It is possible that some Raw Traces are required by more than one Trace Transformation Process. Knowledge Management Process are modeled by Domain Ontology.
Each Knowledge Management Process are modeled by some Domain Ontology.
Each Domain Ontology model exactly one Knowledge Management Process.
It is possible that some Knowledge Management Process are modeled by more than one Domain Ontology.
Knowledge Management Process are involved in Knowledge Management Scope.
Each Knowledge Management Process are involved in some Knowledge Management Scope.
Each Knowledge Management Scope Involve exactly one Knowledge Management Process.
It is possible that some Knowledge Management Process are involved in more than one Knowledge Management Scope.
exactly one Maturity Assessment Process.
It is possible that some Maturity Assessment Process are followed by more than one Computing Evaluation.
Computing Evaluation are involved by Trace-Based Maturity Evaluation.
Each Computing Evaluation are involved by exactly one Trace-Based Maturity Evaluation.
Each Trace-Based Maturity Evaluation involve some Computing Evaluation.
It is possible that some Trace-Based Maturity Evaluation involve more than one Computing Evaluation.

Figure 22. All relational statements of CP2Ms Conceptual Model.

Considering the statements generated in NIAM and described in Figure 22, they show consistency in the CP2Ms conceptual model. This consistency is the consequence of the explicit articulation of the conceptual relationships and relational statements making it possible to verify that the relationship between the domains (conceptual cone) was correctly generated in NIAM, since no errors were detected in the system that would indicate inconsistencies.

3.4 Application of a general CBCN to Project Management

Given that the objective of this research is to improve the maturity assessment process and considering the lack of conceptual guidelines for project management maturity models identified in the literature, this section presents the implementation of the model proposed in section 3.1.

3.4.1 Implementation of the CBCN to project management

The implementation of the project management model is carried out considering four main stages: 1) Identification of Cones and Levels of Abstraction, 2) Identification of conceptual level of abstraction impacted, 3) Substitution of the conceptual level of abstraction and 4) Graphical Visualization of CBCN Implementation in Project Management.

Stage 1: Identification of Cones and Levels of Abstraction

In this initial stage, the CP2Ms Cones-based Conceptual Network (Figure 18) is examined, analyzing its cones and their respective levels of conceptual abstraction. To this end, the following Table 10 is generated:

Table 10. Scopes and levels of abstraction (CBCN)

Cone's name	Abstraction Level 2	Abstraction Level 1
Organization scope	Process Context	Capability
Domain scope	Domain Process	Domain Activities
Maturity Assessment Scope	Maturity Assessment Process	Maturity Level
Referential Scope	Process Referential	Domain Best Practices
Domain tool scope	Functionalities Support Usage	Raw Traces
Data processing scope	Trace Transformation Process	Transformed Trace
Trace-based Maturity scope	Computing Evaluation	Evaluation Element
Knowledge management scope	Knowledge Management Process	Domain Ontology

Stage 2 : Identification of conceptual level of abstraction impacted

At this stage, the cones and conceptual levels of abstraction that will have to be modified when implementing the CBCN in project management are identified, for this purpose the following $Table$ 11 is generated, in which the conceptual level of abstraction to be modified is marked in blue.

Table 11. Identified impacted scopes and levels of abstraction (CBCN)

Cone's name	Abstraction Level 2	Abstraction Level 1
Organization scope	Process Context	Capability
Domain scope	Domain Process	Domain Activities
Maturity Assessment Scope	Maturity Assessment Process	Maturity Level
Referential Scope	Process Referential	Domain Best Practices
Domain tool scope	Functionalities Support Usage	Raw Traces
Data processing scope	Trace Transformation Process	Transformed Trace
Trace-based Maturity scope	Computing Evaluation	Evaluation Element
Knowledge management scope	Knowledge Management Process	Domain Ontology

Stage 3 : Substitution of the conceptual level of abstraction

At this stage, the word domain in the conceptual levels of abstraction is replaced by the application domain, in this case project management. The result of this substitution is shown in Table 12.

Table 12. Substitution of the impacted conceptual levels of abstraction (CBCN)

Cone's name	Abstraction Level 2	Abstraction Level 1
Organization scope	Process Context	Capability
Domain scope	**Project Management** Process	**Project Management** Activities
Maturity Assessment Scope	Maturity Assessment Process	Maturity Level
Referential Scope	Process Referential	**Project Management** Best Practices
Domain tool scope	Functionalities Support Usage	Raw Traces
Data processing scope	Trace Transformation Process	Transformed Trace
Trace-based Maturity scope	Computing Evaluation	Evaluation Element
Knowledge management scope	Knowledge Management Process	**Project Management** Ontology

Stage 4 : Graphical Visualization of CBCN Implementation in Project Management

At this stage, the application of the CBCN model in the project management domain is graphically visualized. This application allows to the project managers to understand that: each time a project management maturity assessment is implemented, the related activities are evaluated using a reference from the same domain, enabling the assignment of a maturity level. This implementation consumes organizational resources based on its context. Additionally, if the evaluation is based on traces, the raw traces generated in a project activity management system are considered and transformed through a process that uses a project management ontology. This transformation allows the traces to be used as an element for evaluating best practices and determining the organization's maturity level. Figure 23, conceptually illustrates the full application of the model to the project management domain.

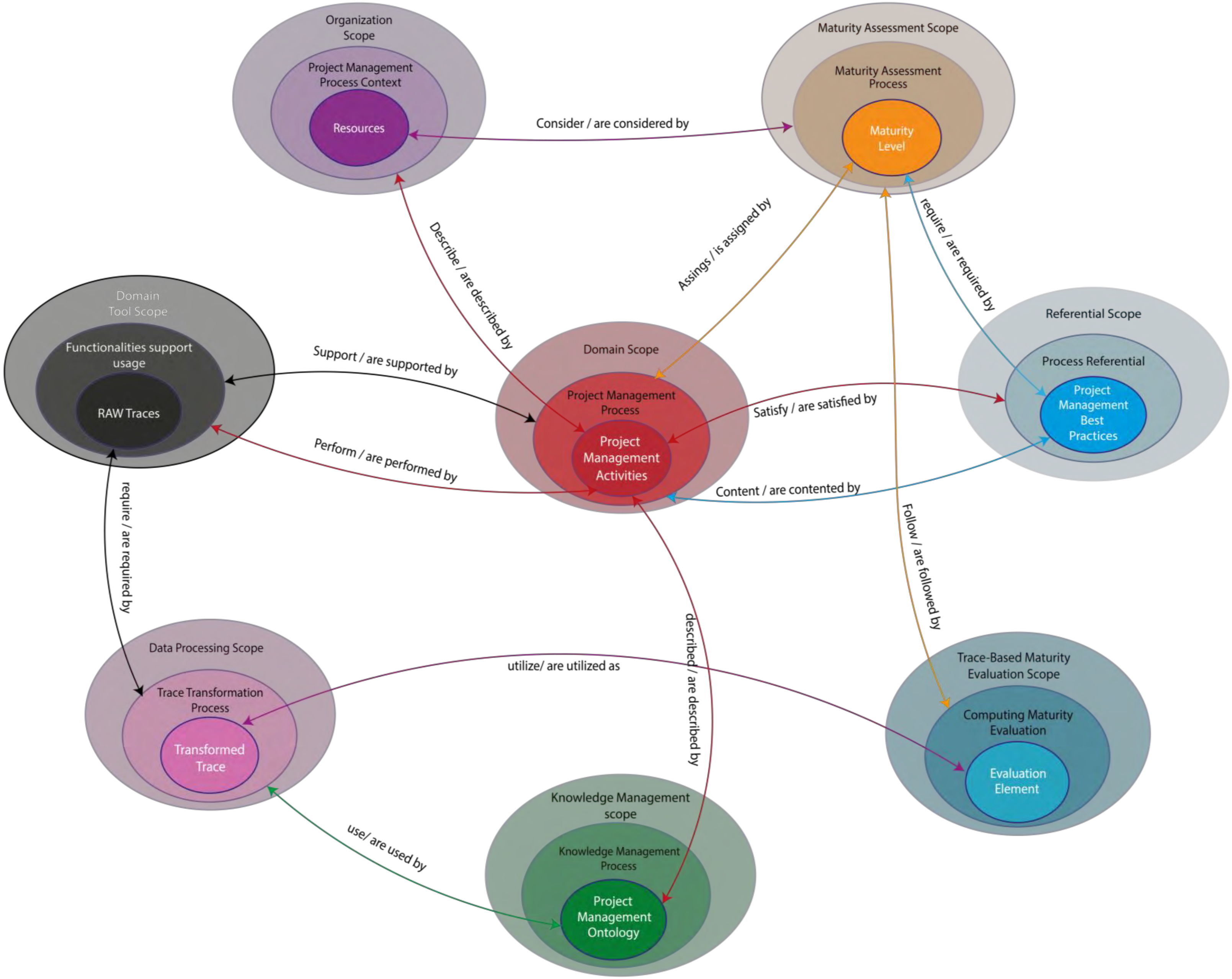


Figure 23. CP2Ms based on CBCN applied to PM applied to PM

3.5 Expansion and Validation of the application of the CP2MS model to project management

After applying each scope described in Section 3.4.1, this section explores an expansion of the application to analyze the concepts identified within each scope and examine their interactions using the NIAM method to verify their consistency.

3.5.1 General scope analysis of the expansion

While the previous section presents the direct application of the CBCN to project management, this section aims to delve deeper into the more specific concepts that can be linked to project management and maturity models during their application. It will further elaborate on the essential knowledge that auditors or stakeholders involved in the implementation must possess to ensure a proper execution.

Domain Scope

The domain of project management describes the main elements that interact with each other to operate within this domain. When applying this framework to a specific domain, it's important to analyze which elements allow for an assessment of maturity. The goal of such an assessment is to identify whether best practices have been implemented in the current domain. Therefore, the elements involved include: 1) project management activities, 2) project phases, 3) project life cycle, 4) project, 5) project management area, 6) project management groups and subgroups of processes, 7) simple and complex links, and 8) best practices. Understanding these elements and their interactions indicates a foundational level of professionalism necessary for implementing project management maturity models.

Referential Scope

As we described in the state-of-the-art review, maturity models are typically based on other process models or frameworks. Generally, these models contain a list of best practices within a domain, and in some cases, there is a brief description of the processes to implement the best practices, though not explicitly. It was identified that the elements interacting in the referential domain are: 1) the reference framework, 2) the best practice process, and finally, 3) best practices themselves. Understanding these elements and their interactions indicates a basic level of professionalization required to comprehend the origins of best practices and their process for implementation within project management.

Maturity Assessment Scope

This scope encompasses the elements that interact with each other to implement a project management maturity model. In general terms, these models include a list of best practices and maturity levels in which the organization is classified based on its assessment results. Therefore, within this scope, the following elements are involved: 1) best practices, 2) maturity level, 3) reference framework, 4) assessment, and 5) maturity assessment tools. Understanding these elements and their interactions indicates a foundational level of professionalism required to determine which maturity model best fits the evaluated process and to facilitate the selection of useful tools for maturity assessment.

Organization Scope

The organization scope encompasses the elements that act as facilitators during a maturity assessment, enabling a better implementation of maturity models. These elements include: 1) resources, 2) organization, 3) project management actors, 4) project management area, 5) quality area, and 6) quality actors. Understanding these elements and their interactions indicates a foundational level of professionalism needed to determine the most suitable maturity model based on available resources and organization type, as well as to identify the areas impacted by the assessment.

Trace-Based Maturity Evaluation tool scope

The scope of a Trace-Based maturity Evaluation tool considers several elements whose interaction allows for the generation of a tool capable of evaluating project management maturity. These elements are as follows: 1) maturity assessment tool, 2) maturity assessment process, 3) assessment element, and 4) traceability. Understanding these elements and their interactions indicates a foundational level of professionalism necessary for using an evaluation tool capable of utilizing transformed traces that describe project management activities, applying an assessment process to them, and obtaining a maturity level.

Data Processing Scope

The data processing domain describes the elements that, when interacting, allow for reducing the granularity level of traces obtained from a project management software using a transformation methodology. These elements include: 1) assessment element, 2) trace, 3) data processing, 4) methodology, 5) transformation rules, and 6) stage. Understanding these elements and their interactions indicates a foundational level of professionalism required to perform the transformation of traces into user activities as an evaluation element.

Domain tool scope

The scope of the domain tool applied to project management considers the most important elements that interact to generate traces within the system and how they relate to each other. These elements are as follows: 1) functionalities, 2) software, 3) API, 4) endpoint, 5) rights, and 6) project management activities. Understanding these elements and their interactions indicates a foundational level of professionalism required to comprehend the operation of a software tool and to identify how traces are generated within the system.

Knowledge Management scope

The knowledge management domain considers the necessary elements to facilitate the transformation of traces from a high level of granularity to a lower and more understandable level. The elements are as follows: 1) transformation rules, 2) process understanding, and 3) ontology. Understanding these elements and their interactions indicates a foundational level of professionalism that enables the generation of transformation rules grounded in a domain ontology, which requires an understanding of the trace generation process.

3.5.2 CP2MS expansion to explore and validate the concepts linked within the scope

Now a detailed description of the conceptual graphical expansion is described, which aims to offer a comprehensive visual representation of each proposed scope within the CBCN model. This approach enables a clearer understanding of the interconnections between various concepts and facilitates a deeper exploration of the domains within project management maturity models. By illustrating these relationships, we seek to enhance the understanding of key components, highlight critical areas for development, and support a structured assessment of maturity models while considering the different aspects of project management.

Domain Scope

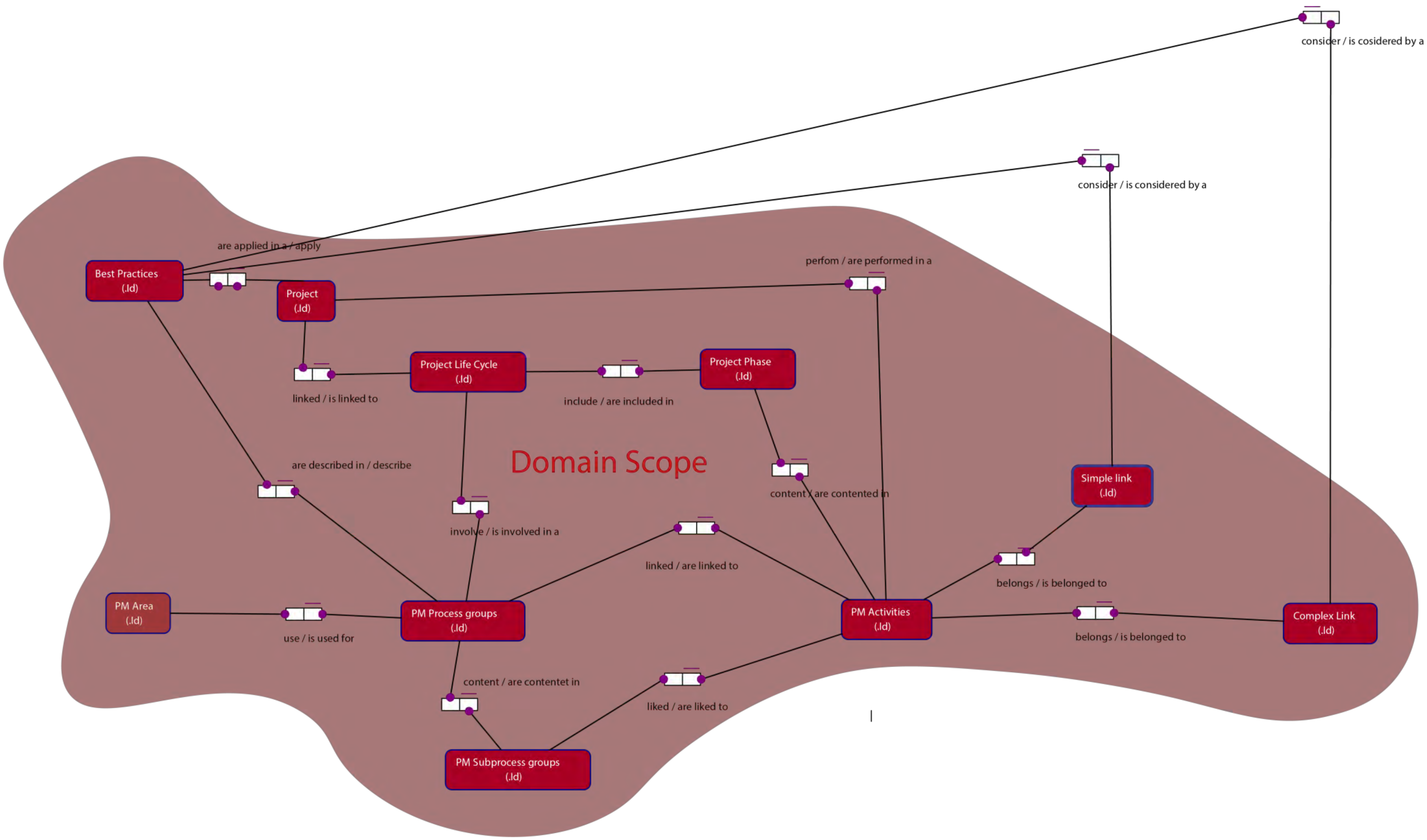


Figure 24. Domain Scope applied to PM

PM Process groups describe PM Best Practices. **Each** PM Best Practices are described in **some** PM Process groups. **Each** PM Process groups describe **exactly one** PM Best Practices. **It is possible that some** PM Best Practices are described in **more than one** PM Process groups. Project Life Cycle include Project Phase. **Each** Project Life Cycle include **some** Project Phase. **Each** Project Phase are included in **exactly one** Project Life Cycle. **It is possible that some** Project Life Cycle include **more than one** Project Phase. Project linked Project Life Cycle. **Each** Project linked **some** Project Life Cycle. **Each** Project Life Cycle is linked to **exactly one** Project. **It is possible that some** Project linked **more than one** Project Life Cycle. Project Life Cycle involve PM Process groups. **Each** Project Life Cycle involve **some** PM Process groups. **Each** PM Process groups is involved in a **exactly one** Project Life Cycle. **It is possible that some** Project Life Cycle involve **more than one** PM Process groups. Project apply PM Best Practices. **Each** PM Best Practices are applied in a **some** Project. **Each** Project apply **exactly one** PM Best Practices. **It is possible that some** PM Best Practices are applied in a **more than one** Project.	PM Process groups linked PM Activities. **Each** PM Process groups linked **some** PM Activities. **Each** PM Activities are linked to **exactly one** PM Process groups. **It is possible that some** PM Process groups linked **more than one** PM Activities. PM Activities belongs Simple link. **Each** PM Activities belongs **some** Simple link. **Each** Simple link is belonged to **exactly one** PM Activities. **It is possible that some** PM Activities belongs **more than one** Simple link. PM Activities belongs Complex Link. **Each** PM Activities belongs **some** Complex Link. **Each** Complex Link is belonged to **exactly one** PM Activities. **It is possible that some** PM Activities belongs **more than one** Complex Link. PM Area use PM Process groups. **Each** PM Area use **some** PM Process groups. **Each** PM Process groups is used for **exactly one** PM Area. **It is possible that some** PM Area use **more than one** PM Process groups. PM Subprocess groups liked PM Activities. **Each** PM Subprocess groups liked **some** PM Activities. **Each** PM Activities are liked to **exactly one** PM Subprocess groups. **It is possible that some** PM Subprocess groups liked **more than one** PM Activities.	PM Best Practices consider Simple link. **Each** PM Best Practices consider **exactly one** Simple link. **Each** Simple link is considered by a **some** PM Best Practices. **It is possible that some** Simple link is considered by a **more than one** PM Best Practices. PM Best Practices consider Complex Link. **Each** PM Best Practices consider **exactly one** Complex Link. **Each** Complex Link is cosidered by a **some** PM Best Practices. **It is possible that some** Complex Link is cosidered by a **more than one** PM Best Practices. Project perfom PM Activities. **Each** Project perfom **some** PM Activities. **Each** PM Activities are performed in a **exactly one** Project. **It is possible that some** Project perfom **more than one** PM Activities. PM Process groups content PM Subprocess groups. **Each** PM Process groups content **some** PM Subprocess groups. **Each** PM Subprocess groups are contentet in **exactly one** PM Process groups. **It is possible that some** PM Process groups content **more than one** PM Subprocess groups. Project Phase content PM Activities. **Each** Project Phase content **some** PM Activities. **Each** PM Activities are contented in **exactly one** Project Phase. **It is possible that some** Project Phase content **more than one** PM Activities.

Figure 25. ORM Verbalization Domain Scope applied to PM

Referential Scope

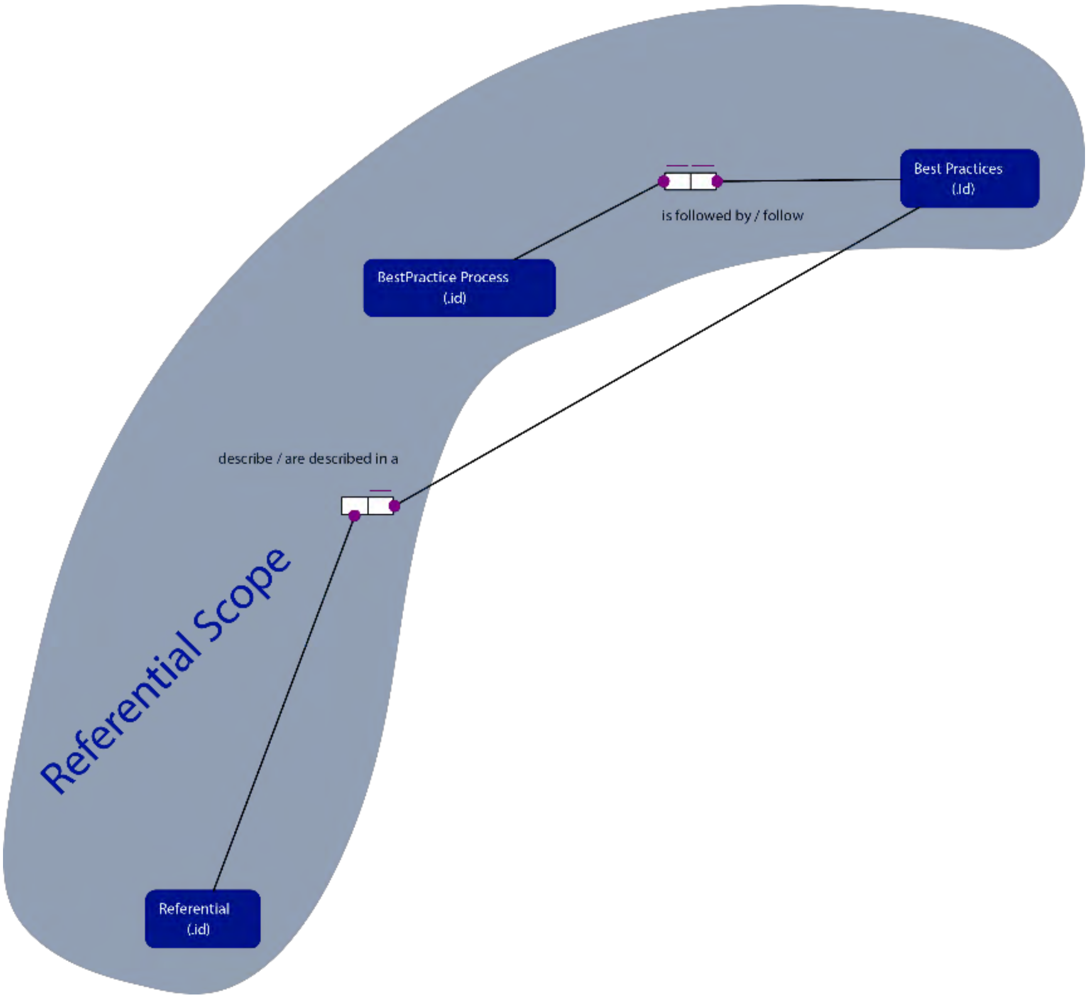


Figure 26. Referential Scope applied to PM

Referential describe PM Best Practices.
Each Referential describe **some** PM Best Practices.
Each PM Best Practices are described in a **exactly one** Referential.
It is possible that some Referential describe **more than one** PM Best Practices.
PM Best Practice Process is followed by PM Best Practices.
Each PM Best Practice Process is followed by **exactly one** PM Best Practices.
Each PM Best Practices follow **exactly one** PM Best Practice Process.

Figure 27. ORM Verbalization Referential Scope applied to PM

Maturity Assessment Scope

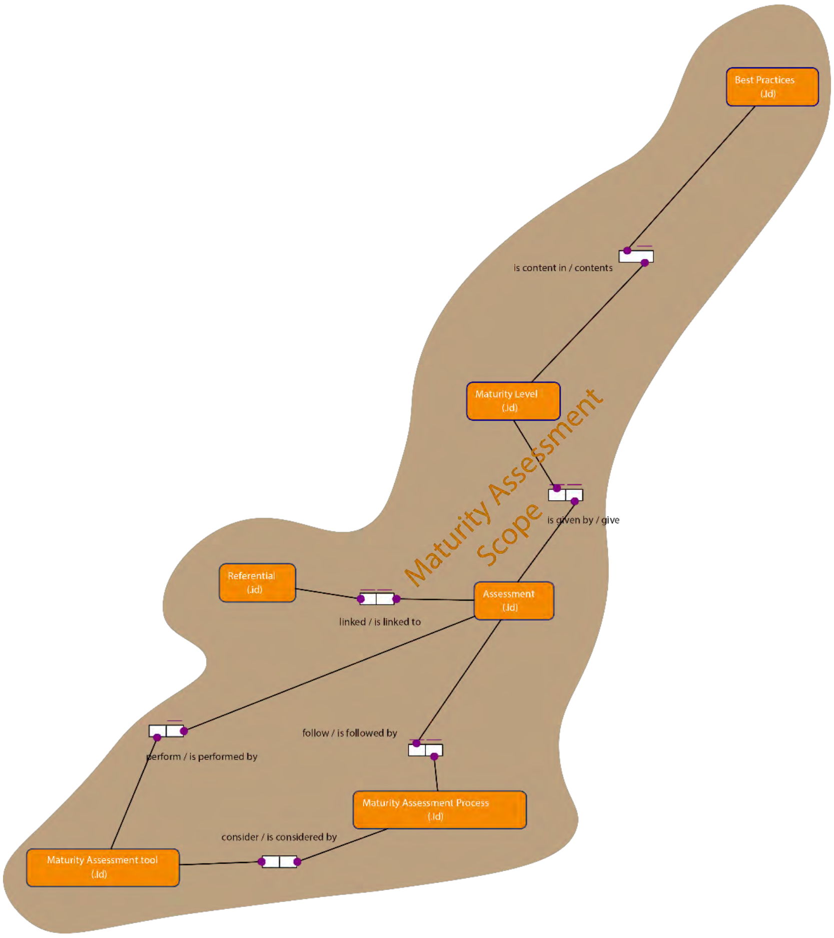


Figure 28. Maturity Assessment Scope applied to PM

Referential describe PM Best Practices. **Each** Referential describe **some** PM Best Practices. **Each** PM Best Practices are described in a **exactly one** Referential. **It is possible that some** Referential describe **more than one** PM Best Practices. Maturity Level contents PM Best Practices. **Each** PM Best Practices is content in **some** Maturity Level. **Each** Maturity Level contents **exactly one** PM Best Practices. **It is possible that some** PM Best Practices is content in **more than one** Maturity Level. Maturity Level is given by Assessment. **Each** Maturity Level is given by **exactly one** Assessment. **Each** Assessment give **exactly one** Maturity Level. Assessment follow Maturity Assessment Process. **Each** Assessment follow **exactly one** Maturity Assessment Process. **Each** Maturity Assessment Process is followed by **exactly one** Assessment.	Maturity Assessment Tool perform Assessment. **Each** Maturity Assessment Tool perform **some** Assessment. **Each** Assessment is performed by **exactly one** Maturity Assessment Tool. **It is possible that some** Maturity Assessment Tool perform **more than one** Assessment. Maturity Assessment Tool consider Maturity Assessment Process. **Each** Maturity Assessment Tool consider **some** Maturity Assessment Process. **Each** Maturity Assessment Process is considered by **exactly one** Maturity Assessment Tool. **It is possible that some** Maturity Assessment Tool consider **more than one** Maturity Assessment Process. Referential linked Assessment. **Each** Referential linked **exactly one** Assessment. **Each** Assessment is linked to **exactly one** Referential.

Figure 29. ORM Verbalization Maturity Assessment Scope applied to PM.

Organization Scope

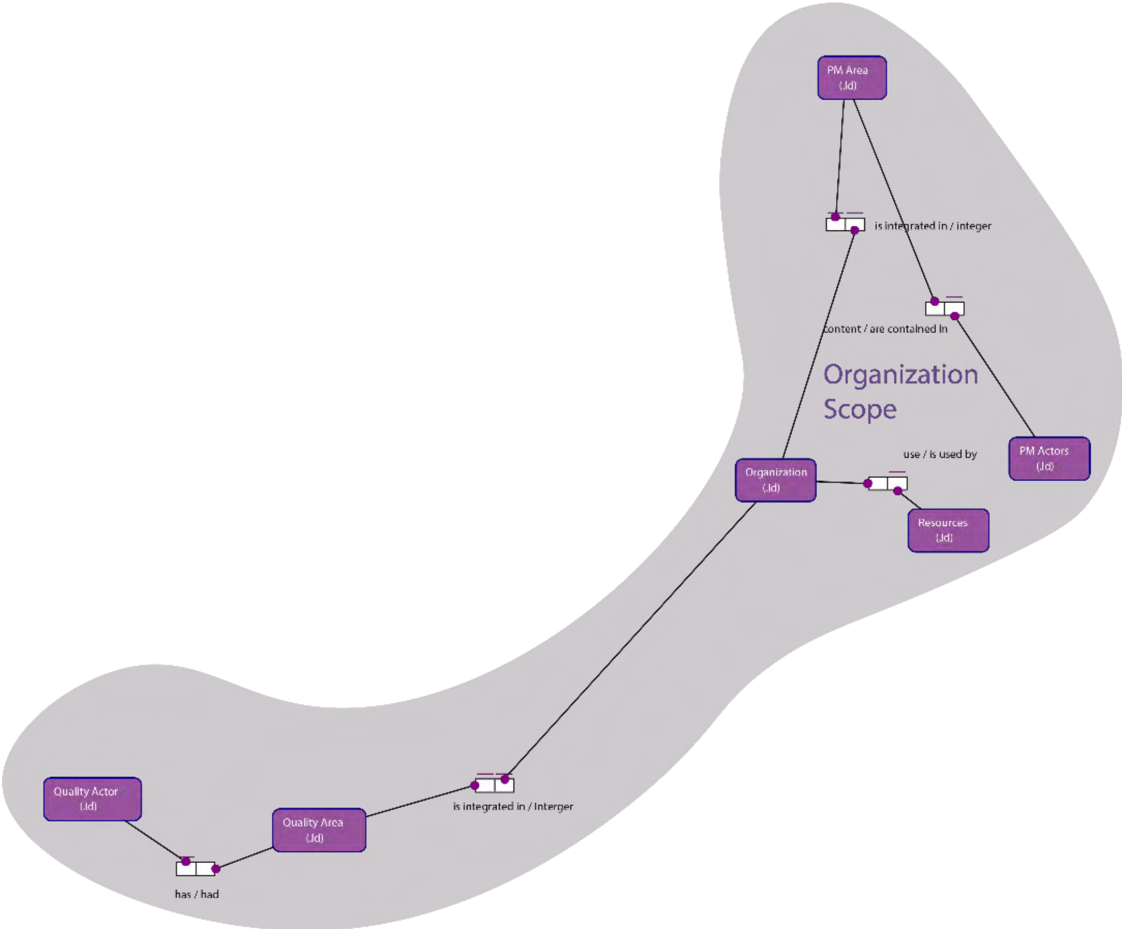


Figure 30. Organization Scope applied to PM

Quality Area is integrated in Organization. **Each** Quality Area is integrated in **exactly one** Organization. **Each** Organization Interger **exactly one** Quality Area. Quality Actor has Quality Area. **Each** Quality Actor has **exactly one** Quality Area. **Each** Quality Area had **some** Quality Actor. **It is possible that some** Quality Area had **more than one** Quality Actor. PM Area is integrated in Organization. **Each** PM Area is integrated in **exactly one** Organization. **Each** Organization integer **exactly one** PM Area.	PM Area content PM Actors. **Each** PM Area content **some** PM Actors. **Each** PM Actors are contained in **exactly one** PM Area. **It is possible that some** PM Area content **more than one** PM Actors. Organization use Resources. **Each** Organization use **some** Resources. **Each** Resources is used by **exactly one** Organization. **It is possible that some** Organization use **more than one** Resources.

Figure 31. ORM Verbalization Organization Scope applied to PM.

Trace-Based Maturity Evaluation Scope

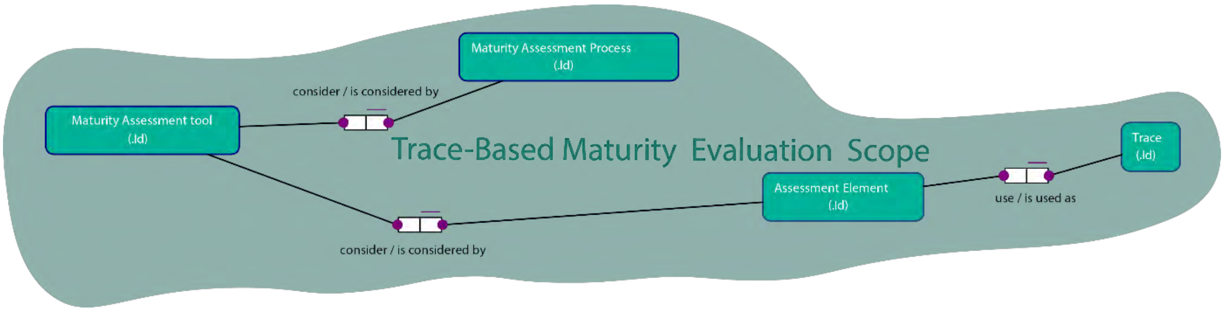


Figure 32. Trace-Based Maturity Evaluation Scope applied to PM

Maturity Assessment Tool consider Maturity Assessment Process. **Each** Maturity Assessment Tool consider **some** Maturity Assessment Process. **Each** Maturity Assessment Process is considered by **exactly one** Maturity Assessment Tool. **It is possible that some** Maturity Assessment Tool consider **more than one** Maturity Assessment Process. **Each** Assessment Element are used by **at most one** Maturity Assessment Tool.	Maturity Assessment Tool use Assessment Element. **Each** Maturity Assessment Tool use **some** Assessment Element. **Each** Assessment Element are used by **exactly one** Maturity Assessment Tool. **It is possible that some** Maturity Assessment Tool use **more than one** Assessment Element. Assessment Element use Trace. **Each** Assessment Element use **some** Trace. **Each** Trace is used as **exactly one** Assessment Element. **It is possible that some** Assessment Element use **more than one** Trace.

Figure 33. ORM Verbalization Trace-Based Maturity Evaluation Scope applied to PM.

Data Processing Scope

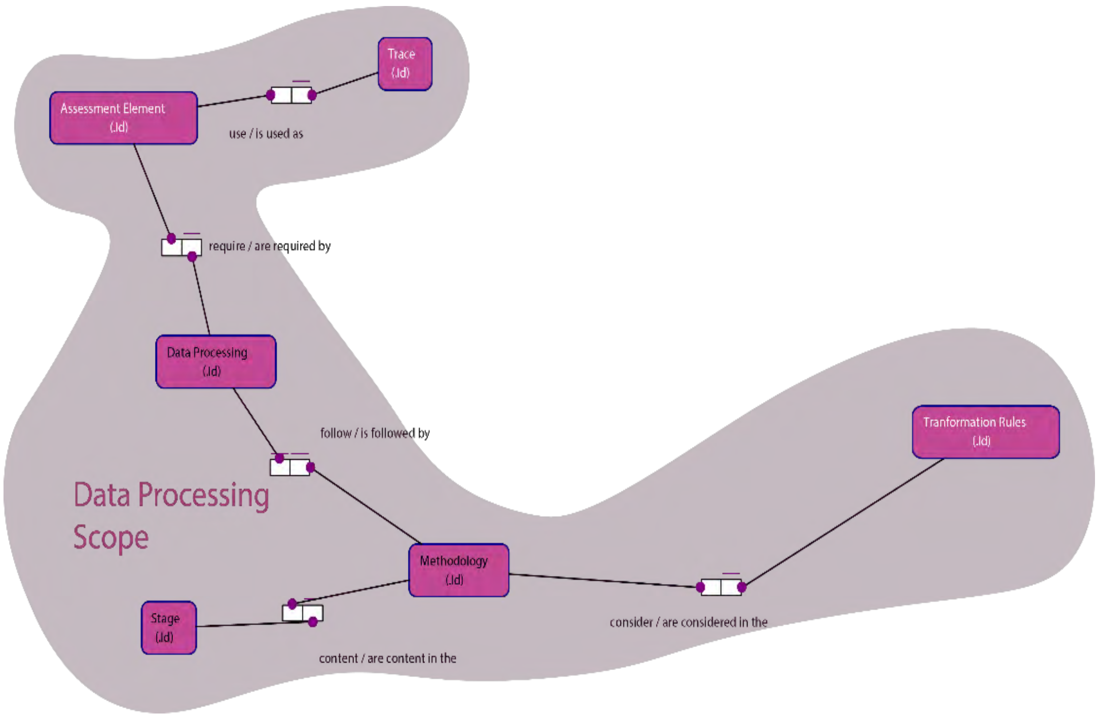


Figure 34. Data Processing Scope applied to PM

Methodology consider Tranformation Rules. **Each** Methodology consider **some** Tranformation Rules. **Each** Tranformation Rules are considered in the **exactly one** Methodology. **It is possible that some** Methodology consider **more than one** Tranformation Rules. Methodology content Stage. **Each** Methodology content **some** Stage. **Each** Stage are content in the **exactly one** Methodology. **It is possible that some** Methodology content **more than one** Stage. Data Processing follow Methodology. **Each** Data Processing follow **exactly one** Methodology. **Each** Methodology is followed by **exactly one** Data Processing.	Assessment Element require Data Processing. **Each** Assessment Element require **some** Data Processing. **Each** Data Processing are required by **exactly one** Assessment Element. **It is possible that some** Assessment Element require **more than one** Data Processing. Assessment Element use Trace. **Each** Assessment Element use **some** Trace. **Each** Trace is used as **exactly one** Assessment Element. **It is possible that some** Assessment Element use **more than one** Trace.

Figure 35. ORM Verbalization Data Processing Scope applied to PM.

Domain tool scope

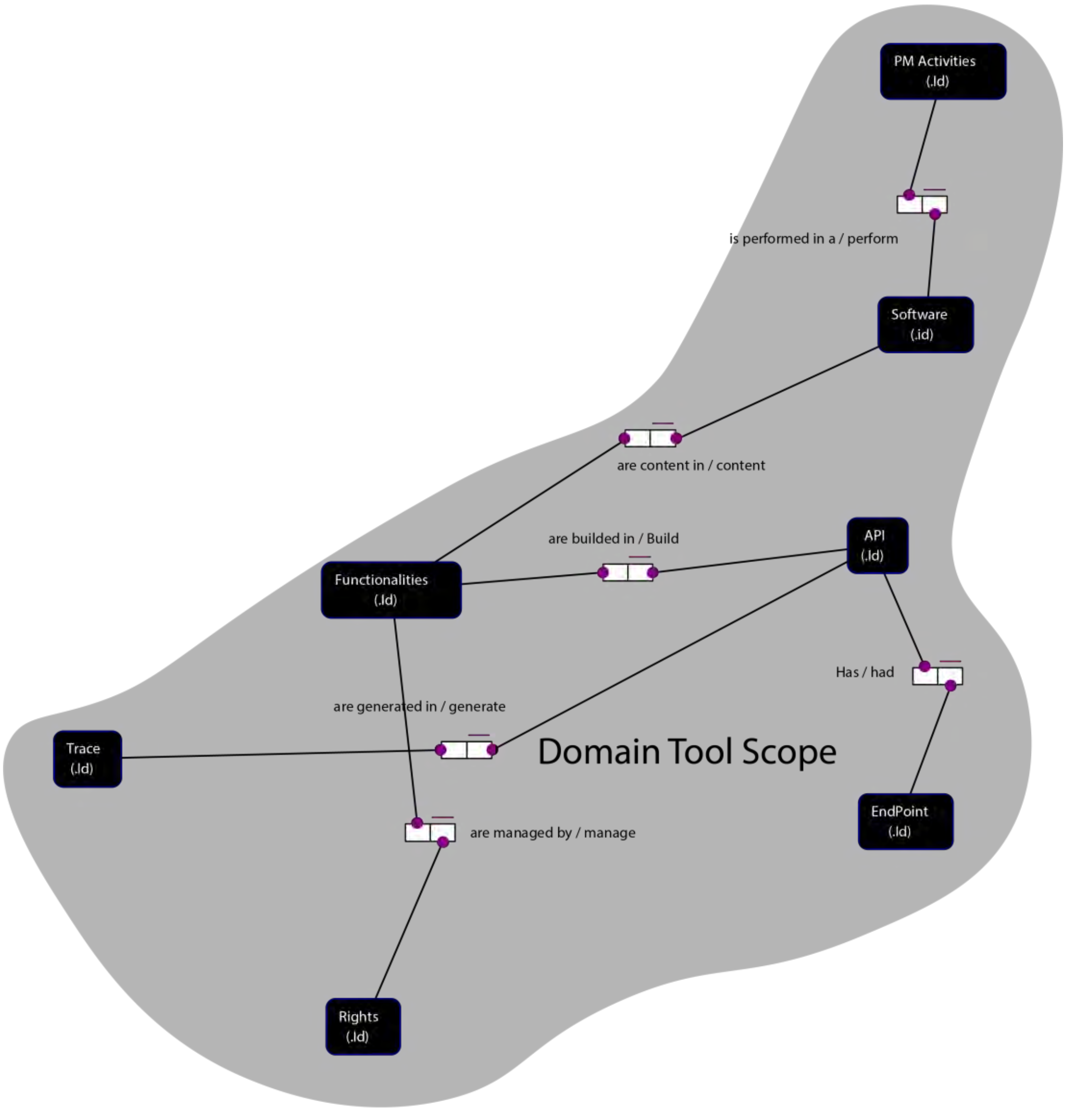


Figure 36. Domain Tool Scope applied to PM

Functionalities are builded in API. **Each** Functionalities are builded in **some** API. **Each** API Build **exactly one** Functionalities. **It is possible that some** Functionalities are builded in **more than one** API. Software content Functionalities. **Each** Functionalities are content in **some** Software. **Each** Software content **exactly one** Functionalities. **It is possible that some** Functionalities are content in **more than one** Software. Functionalities are managed by Rights. **Each** Functionalities are managed by **some** Rights. **Each** Rights manage **exactly one** Functionalities. **It is possible that some** Functionalities are managed by **more than one** Rights.	API generate Trace. **Each** Trace are generated in **some** API. **Each** API generate **exactly one** Trace. **It is possible that some** Trace are generated in **more than one** API. API Has EndPoint. **Each** API Has **some** EndPoint. **Each** EndPoint had **exactly one** API. **It is possible that some** API Has **more than one** EndPoint. PM Activities is performed in aSoftware. **Each** PM Activities is performed in a**some** Software. **Each** Software perform **exactly one** PM Activities. **It is possible that some** PM Activities is performed in a**more than one** Software

Figure 37. ORM Verbalization Domain Tool Scope applied to PM.

Knowledge Management scope

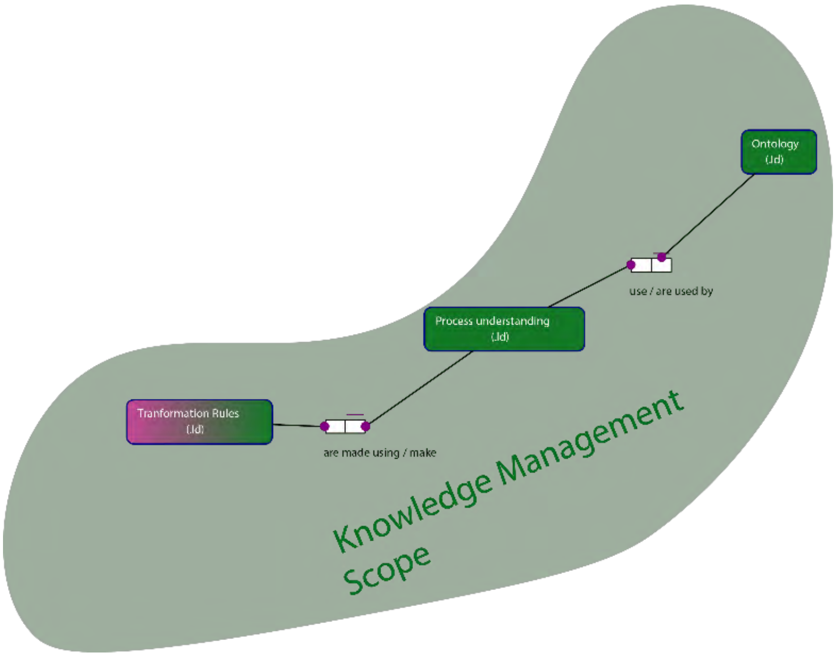


Figure 38. Knowledge Management Scope applied to PM

Process understanding use Ontology. **Each** Process understanding use **some** Ontology. **Each** Ontology are used by **exactly one** Process understanding. **It is possible that some** Process understanding use **more than one** Ontology.	Tranformation Rules are made using Process understanding. **Each** Tranformation Rules are made using **some** Process understanding. **Each** Process understanding make **exactly one** Tranformation Rules. **It is possible that some** Tranformation Rules are made using **more than one** Process understanding.

Figure 39. ORM Verbalization Knowledge Management Scope applied to PM.

3.5.3 Full view of the CP2MS expansion and general validation of the concepts linked.

Full CP2MS Conceptual Framework Model Applied to Project Management

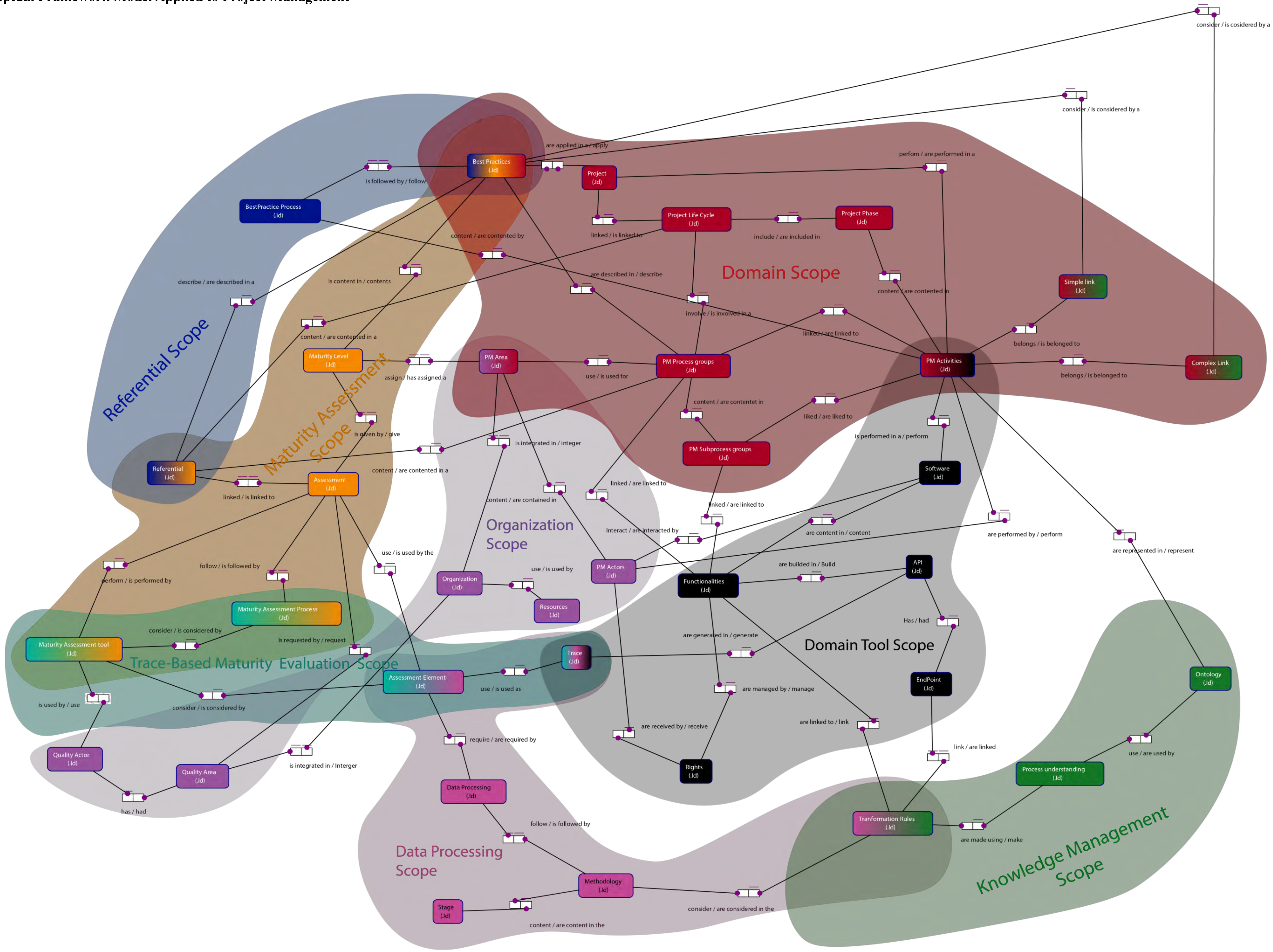


Figure 40. Scopes and interaction applied to Project Management

The Figure 40, describes all the elements that interact when applying the CP2MS conceptual framework to project management. To group the different scopes (Cones), a color was assigned to each one. Additionally, it details the elements that interact between two or more scope (Cones), generating connections in the network. These interactions illustrate how the same element can be present in more than one scope (Cone), which signifies that an element allows for connecting different scopes (Cones). The elements that generate connections between various scopes (Cones) enable the creation of a more realistic model, where not only the main elements are described, but also their interactions. For this reason, we propose that, to apply maturity models in this domain, it is essential to understand the scope (Cones) and their internal and external interactions. External interactions are detailed in Table 13.

Table 13. Interaction elements in Project Management

	Scopes Interacted		**Interaction element**
Domain Scope	Referential Scope	Maturity Assessment Scope	Best Practice
Domain Scope	Organization Scope	N/A	Project Management Area
Domain Scope	Domain Tool Scope	N/A	Project Management Activities
Referential Scope	Maturity Assessment Scope	N/A	Referential
Maturity Assessment Scope	Trace-Based Maturity Evaluation Scope	N/A	Maturity Assessment Process
Maturity Assessment Scope	Trace-Based Maturity Evaluation Scope	N/A	Maturity Assessment Tool
Trace-Based Maturity Evaluation Scope	Data Processing Scope	N/A	Assessment Element
Trace-Based Maturity Evaluation Scope	Data Processing Scope	Domain Tool Scope	Trace
Data Processing Scope	Knowledge Management Scope	N/A	Transformation Rules

Project linked Project Life Cycle. **Each** Project linked **some** Project Life Cycle. **Each** Project Life Cycle is linked to **exactly one** Project. **It is possible that some** Project linked **more than one** Project Life Cycle. Project Life Cycle include Project Phase. **Each** Project Life Cycle include **some** Project Phase. **Each** Project Phase are included in **exactly one** Project Life Cycle. **It is possible that some** Project Life Cycle include **more than one** Project Phase. Project perfom PM Activities. **Each** Project perfom **some** PM Activities. **Each** PM Activities are performed in a **exactly one** Project. **It is possible that some** Project perfom **more than one** PM Activities. PM Best Practices consider Simple link. **Each** PM Best Practices consider **exactly one** Simple link. **Each** Simple link is considered by a **some** PM Best Practices. **It is possible that some** Simple link is considered by a **more than one** PM Best Practices. PM Best Practices consider Complex Link. **Each** PM Best Practices consider **exactly one** Complex Link. **Each** Complex Link is cosidered by a **some** PM Best Practices. **It is possible that some** Complex Link is cosidered by a **more than one** PM Best Practices. Project Life Cycle involve PM Process groups. **Each** Project Life Cycle involve **some** PM Process groups. **Each** PM Process groups is involved in a **exactly one** Project Life Cycle. **It is possible that some** Project Life Cycle involve **more than one** PM Process groups. PM Process groups linked PM Activities. **Each** PM Process groups linked **some** PM Activities. **Each** PM Activities are linked to	**exactly one** PM Process groups. **It is possible that some** PM Process groups linked **more than one** PM Activities. PM Subprocess groups liked PM Activities. **Each** PM Subprocess groups liked **some** PM Activities. **Each** PM Activities are liked to **exactly one** PM Subprocess groups. **It is possible that some** PM Subprocess groups liked **more than one** PM Activities. PM Process groups content PM Subprocess groups. **Each** PM Process groups content **some** PM Subprocess groups. **Each** PM Subprocess groups are contentet in **exactly one** PM Process groups. **It is possible that some** PM Process groups content **more than one** PM Subprocess groups. PM Activities belongs Complex Link. **Each** PM Activities belongs **some** Complex Link. **Each** Complex Link is belonged to **exactly one** PM Activities. **It is possible that some** PM Activities belongs **more than one** Complex Link. PM Activities is performed in aSoftware. **Each** PM Activities is performed in a**some** Software. **Each** Software perform **exactly one** PM Activities. **It is possible that some** PM Activities is performed in a**more than one** Software. Software content Functionalities. **Each** Functionalities are content in **some** Software. **Each** Software content **exactly one** Functionalities. **It is possible that some** Functionalities are content in **more than one** Software. PM Subprocess groups linked Functionalities. **Each** PM Subprocess groups linked **some** Functionalities. **Each** Functionalities are linked to **exactly one** PM Subprocess groups.	**It is possible that some** PM Subprocess groups linked **more than one** Functionalities. PM Process groups linked Functionalities. **Each** PM Process groups linked **some** Functionalities. **Each** Functionalities are linked to **exactly one** PM Process groups. **It is possible that some** PM Process groups linked **more than one** Functionalities. PM Process groups describe PM Best Practices. **Each** PM Best Practices are described in **some** PM Process groups. **Each** PM Process groups describe **exactly one** PM Best Practices. **It is possible that some** PM Best Practices are described in **more than one** PM Process groups. PM Best Practice Process is followed by PM Best Practices. **Each** PM Best Practice Process is followed by **exactly one** PM Best Practices. **Each** PM Best Practices follow **exactly one** PM Best Practice Process. PM Area use PM Process groups. **Each** PM Area use **some** PM Process groups. **Each** PM Process groups is used for **exactly one** PM Area. **It is possible that some** PM Area use **more than one** PM Process groups. PM Area is integrated in Organization. **Each** PM Area is integrated in **exactly one** Organization. **Each** Organization integer **exactly one** PM Area. PM Area content PM Actors. **Each** PM Area content **some** PM Actors. **Each** PM Actors are contained in **exactly one** PM Area. **It is possible that some** PM Area content **more than one** PM Actors. PM Activities are performed by PM Actors. **Each** PM Activities are performed by **some** PM Actors. **Each** PM Actors perform **exactly one** PM Activities.

It is possible that some PM Activities are performed by **more than one** PM Actors. Maturity Level contents PM Best Practices. **Each** PM Best Practices is content in **some** Maturity Level. **Each** Maturity Level contents **exactly one** PM Best Practices. **It is possible that some** PM Best Practices is content in **more than one** Maturity Level. Maturity Level assign PM Area. **Each** Maturity Level assign **exactly one** PM Area. **Each** PM Area has assigned a **exactly one** Maturity Level. Maturity Level is given by Assessment. **Each** Maturity Level is given by **exactly one** Assessment. **Each** Assessment give **exactly one** Maturity Level. Referential linked Assessment. **Each** Referential linked **exactly one** Assessment. **Each** Assessment is linked to **exactly one** Referential. Assessment follow Maturity Assessment Process. **Each** Assessment follow **exactly one** Maturity Assessment Process. **Each** Maturity Assessment Process is followed by **exactly one** Assessment. PM Activities belongs Simple link. **Each** PM Activities belongs **some** Simple link. **Each** Simple link is belonged to **exactly one** PM Activities. **It is possible that some** PM Activities belongs **more than one** Simple link. Maturity Assessment Tool perform Assessment. **Each** Maturity Assessment Tool perform **some** Assessment. **Each** Assessment is performed by **exactly one** Maturity Assessment Tool. **It is possible that some** Maturity Assessment Tool perform **more than one** Assessment.	Maturity Assessment Tool consider Maturity Assessment Process. **Each** Maturity Assessment Tool consider **some** Maturity Assessment Process. **Each** Maturity Assessment Process is considered by **exactly one** Maturity Assessment Tool. **It is possible that some** Maturity Assessment Tool consider **more than one** Maturity Assessment Process. Project Phase content PM Activities. **Each** Project Phase content **some** PM Activities. **Each** PM Activities are contented in **exactly one** Project Phase. **It is possible that some** Project Phase content **more than one** PM Activities. Referential content PM Process groups. **Each** Referential content **some** PM Process groups. **Each** PM Process groups are contented in a **exactly one** Referential. **It is possible that some** Referential content **more than one** PM Process groups. Maturity Assessment Tool use Assessment Element. **Each** Maturity Assessment Tool use **some** Assessment Element. **Each** Assessment Element are used by **exactly one** Maturity Assessment Tool. **It is possible that some** Maturity Assessment Tool use **more than one** Assessment Element. Project apply PM Best Practices. **Each** PM Best Practices are applied in a **some** Project. **Each** Project apply **exactly one** PM Best Practices. **It is possible that some** PM Best Practices are applied in a **more than one** Project. Referential describe PM Best Practices. **Each** Referential describe **some** PM Best Practices.	**Each** PM Best Practices are described in a **exactly one** Referential. **It is possible that some** Referential describe **more than one** PM Best Practices. Functionalities are managed by Rights. **Each** Functionalities are managed by **some** Rights. **Each** Rights manage **exactly one** Functionalities. **It is possible that some** Functionalities are managed by **more than one** Rights. Rights are received by PM Actors. **Each** Rights are received by **some** PM Actors. **Each** PM Actors receive **exactly one** Rights. **It is possible that some** Rights are received by **more than one** PM Actors. Data Processing follow Methodology. **Each** Data Processing follow **exactly one** Methodology. **Each** Methodology is followed by **exactly one** Data Processing. Methodology consider Tranformation Rules. **Each** Methodology consider **some** Tranformation Rules. **Each** Tranformation Rules are considered in the **exactly one** Methodology. **It is possible that some** Methodology consider **more than one** Tranformation Rules. Tranformation Rules are linked to Functionalities. **Each** Tranformation Rules are linked to **some** Functionalities. **Each** Functionalities link **exactly one** Tranformation Rules. **It is possible that some** Tranformation Rules are linked to **more than one** Functionalities. PM Best Practice Process content PM Activities. **Each** PM Best Practice Process content **some** PM Activities.

Each PM Activities are contented by **exactly one** PM Best Practice Process.
It is possible that some PM Best Practice Process content **more than one** PM Activities.
Tranformation Rules are made using Process understanding.
Each Tranformation Rules are made using **some** Process understanding.
Each Process understanding make **exactly one** Tranformation Rules.
It is possible that some Tranformation Rules are made using **more than one** Process understanding.
PM Activities are represented in Ontology.
Each PM Activities are represented in **exactly one** Ontology.
Each Ontology represent **some** PM Activities.
It is possible that some Ontology represent **more than one** PM Activities.
Referential content Project Life Cycle.
Each Referential content **some** Project Life Cycle.
Each Project Life Cycle are contented in a **exactly one** Referential.
It is possible that some Referential content **more than one** Project Life Cycle.
Methodology content Stage.
Each Methodology content **some** Stage.
Each Stage are content in the **exactly one** Methodology.
It is possible that some Methodology content **more than one** Stage.
Process understanding use Ontology.
Each Process understanding use **some** Ontology.
Each Ontology are used by **exactly one** Process understanding.
It is possible that some Process understanding use **more than one** Ontology.
Organization use Resources.
Each Organization use **some** Resources.
Each Resources is used by **exactly one** Organization.
It is possible that some Organization use **more than one** Resources.
Functionalities are builded in API.
Each Functionalities are builded in **some** API.
Each API Build **exactly one** Functionalities.
It is possible that some Functionalities are builded in **more than one** API.
API generate Trace.
Each Trace are generated in **some** API.
Each API generate **exactly one** Trace.
It is possible that some Trace are generated in **more than one** API.
API Has EndPoint.
Each API Has **some** EndPoint.
Each EndPoint had **exactly one** API.
It is possible that some API Has **more than one** EndPoint.
EndPoint link Tranformation Rules.
Each EndPoint link **exactly one** Tranformation Rules.
Each Tranformation Rules are linked **exactly one** EndPoint.
PM Actors Interact Software.
Each PM Actors Interact **some** Software.
Each Software are interacted by **exactly one** PM Actors.
It is possible that some PM Actors Interact **more than one** Software.
Quality Area is integrated in Organization.
Each Quality Area is integrated in **exactly one** Organization.
Each Organization Interger **exactly one** Quality Area.
Maturity Assessment Tool is used by Quality Actor.
Each Maturity Assessment Tool is used by **some** Quality Actor.
Each Quality Actor use **exactly one** Maturity Assessment Tool.
It is possible that some Maturity Assessment Tool is used by **more than one** Quality Actor.
Quality Actor has Quality Area.
Each Quality Actor has **exactly one** Quality Area.
Each Quality Area had **some** Quality Actor.
It is possible that some Quality Area had **more than one** Quality Actor.
Assessmentis requested by Quality Area.
Each Assessmentis requested by **exactly one** Quality Area.
Each Quality Area request **some** Assessment.
It is possible that some Quality Area request **more than one** Assessment.
Assessment Element use Trace.
Each Assessment Element use **some** Trace.
Each Trace is used as **exactly one** Assessment Element.
It is possible that some Assessment Element use **more than one** Trace.
Assessment Element require Data Processing.
Each Assessment Element require **some** Data Processing.
Each Data Processing are required by **exactly one** Assessment Element.
It is possible that some Assessment Element require **more than one** Data Processing.

Figure 41. All relational statements of CP2Ms Conceptual Model applied to PM.

Considering the statements generated in NIAM and described in Figure 41, show consistency in the CP2Ms conceptual model applied to project Management. This consistency is the consequence of the explicit articulation of the conceptual relationships and relational statements making it possible to verify that the relationship between the domains (conceptual cone) was correctly generated in NIAM, since no errors were detected in the system that would indicate inconsistencies.

Although the use of NIAM allows for the explicit representation of the conceptual relationships of the scopes through a model and verbal analysis, another utility has been identified. The expansion of the CBCN model allows for the addition of more concepts to the model generated in NIAM, aiming to better complement and detail each scope. In this sense, during the implementation of CBCN in project management, additional concepts were added, allowing for the expansion of the model and further detailing the key concepts identified in Figure 23 for each scope.

3.5 Chapter 3 Summary

This chapter introduces the cone-based conceptual network as an approach for modeling engineering activities and facilitating process maturity assessment based on traces generated by management software. This network allows for the description of key concepts and their relationships, providing stakeholders with a structured view of the context in which a maturity assessment is conducted. Unlike mind maps or conceptual mapping, the proposed model is more structured without reaching the rigidity of a formal ontology. In this regard, Traditional Process Maturity Models (TP2Ms) require the consideration of multiple interconnected scopes, including the organizational scope, domain scope, maturity assessment scope, and referential scope. Each of these scopes presents different levels of granularity, allowing for the contextualization of the assessment and the alignment of processes with standards and best practices.

Since traditional models can be challenging to implement in organizations with limited resources, an IT-focused extension is proposed, known as Computer-Based Process Maturity Models (CP2Ms). This extension introduces four new conceptual scopes centered on digital tools, data processing, knowledge management, and trace-based maturity evaluation. The incorporation of these elements enables the transformation and analysis of traces generated by software to infer user activities and assess process maturity without relying solely on traditional audits. Implementing this model requires a domain ontology to structure knowledge and facilitate the conversion of data into meaningful information for evaluation.

Finally, the consistency of the conceptual network is validated through NIAM, which allows for the structuring and verbalization of relationships among concepts within each scope. This methodology facilitates the identification of potential inconsistencies in the conceptual network and ensures that the interactions among processes, concepts, and granularity levels remain coherent. The application of CP2Ms to project management demonstrates its usefulness in evaluating organizational maturity based on computer traces, enabling a more objective assessment that is less dependent on human intervention.

3.6 Chapter 3 Discussion

The discussion addresses four key issues: (1) the theoretical quality of CBCN-based modeling, (2) its relevance to AAMs, (3) its value for IT specialists developing functional TBSs, and (4) the theoretical requirements for implementing maturity models.

In the proposed PCS, factors such as the nature and size of the organization—whether it is a manufacturer or a service provider, SME or international group—are not considered. Similarly, the focus of the audited process—exploitation (routine billing) versus exploration (research and disruptive innovations)—and the type of KPIs used—technical versus financial—are left unspecified. This omission leaves unanswered the question of tailoring the PCS to different organizational contexts. Despite this, it is likely that the organization's nature and size significantly influence other nodes within the CBCN, and that KPIs, though different in nature, could be integrated into both the process domain and the best practice domain, as seen in models like SCOR. Furthermore, large organizations possess the resources to implement complex CP2Ms, while SMEs may lack the capacity. Our proposal does not address the accessibility of the PCS or other TBS for different types of organizations.

One of the primary challenges in implementing P2Ms lies in the lack of clarity within AAMs. Typically, P2Ms do not offer clear procedural guidelines for accomplishing their specific missions, which ideally should include preparation, execution, and analysis. However, many organizations tend to bypass the preparation stage. Additionally, after reviewing AAMs, managers often struggle to implement the recommended best practices. In our proposed PCS, procedural knowledge related to AAMs and the subsequent implementation of new practices must be integrated within a specific scope. This integration corresponds to a certain level of absorptive capability in the organization whose processes are being audited.

The implementation of the proposed PCS requires two tools: one for managing domain activities, which generate user traces, and another for transforming raw traces into processed data. Moreover, a trace transformation methodology is needed to move from a high level of granularity to a lower one, where traces are described in terms of domain-specific activities. While the PCS provides an overarching view at a low level of granularity, this is insufficient for developing TBSs. A key challenge is how to rigorously map the proposed graphs onto software engineering and data conceptual models. Our current proposal addresses only the preliminary design phase of the CP2M life cycle.

The PCS proposed operates in tension between exploitation and exploration, between the modeling of an as-is situation—codified routine knowledge in TP2Ms—and a to-be situation, referring to potential CP2Ms based on ill-defined knowledge and software applications. This duality gives the PCS an ambiguous status, where "model" refers both to a stylized representation of what exists and to a sketch of a desired future state. It must therefore strike a balance: being as realistic as possible in representing the current state (TP2M) while also being sufficiently innovative and open-ended to envision the future (CP2M).

The implementation of CBCNs should follow the approach outlined in this article, though it is recognized that the generic model may require adaptations to fit specific contexts. To ensure accuracy in future modifications, NIAM is recommended as a validation tool, as it helps clarify relationships and identify inconsistencies.

Despite its limitations, our proposal provides a foundational framework for transitioning from a literal and insufficiently formalized understanding of processes and AAMs to one that harnesses the potential of modern IT, particularly in knowledge modeling (using NIAM) and trace elaboration (with TBSs). CBCNs serve as a guide for the upstream design of maturity models using TBSs. Ideally, the PCS would facilitate collaborative work between auditors and IT specialists to model the necessary knowledge and data for future computer-based AAMs. NIAM plays a crucial role in expressing the relationships between concepts, generating visual conceptual graphs, verifying network consistency, and validating them through user feedback (from auditors and IT specialists). Only once these collaborative steps are completed can IT specialists move into their domain, utilizing tools and methodologies that may be unfamiliar to auditors.

This contribution focuses on conceptual knowledge and does not delve into the technical or software engineering aspects of implementation. However, it highlights the potential of the proposed PCS for future TBS development. Finally, in the following Chapter 4, the second main contribution of this thesis work, the trace transformation process, is described. This process allows the reduction of the level of granularity (Detail) of the records obtained from the interaction of a user with a computer system. This reduction is focused on facilitating the exploitation of the traces to evaluate the maturity level.

Chapter 4. Process for exploiting computer traces to evaluate an engineering activity maturity

The relevance of the trace transformation process stems from the need to understand the traces generated by modern information systems. Today, organizations worldwide produce vast amounts of data, often exceeding their processing capacities, leading to what is commonly known as "big data." These large datasets serve various purposes, with knowledge extraction being particularly important. Due to the complexity of this data, numerous algorithms, processing tools, and methodologies have emerged to facilitate knowledge extraction.

The importance of extracting knowledge from data has been widely recognized by industries, which increasingly rely on "data-driven decision making" to guide their strategies. One of the most valuable types of data is generated by user interactions with domain-specific software (Traces) . These traces capture the entire path a user follows within a system, generating a massive volume of data each year. However, much of this data remains underutilized due to a lack of knowledge and standardized methodologies to effectively process and exploit it.

These traces are highly complex due to their granular nature. Analyzing them to determine user activities within the system requires a multi-stage process to reduce granularity and accurately identify user actions. This thesis addresses the transformation process required for this task. This chapter is organized as a follow: Section 4.1 outlines the challenges associated with trace transformation, while Section 4.2 presents a general trace transformation process. In Section 4.3, this process is applied to traces from project management software (OpenProject). Finally, Section 4.4 describes the modules of a trace management system designed to validate the proposed transformation process. Also a complementary sections: 4.5 Chapter 4 Summary and 4.6 Chapter 4 Discussion are addressed in the end of this chapter.

4.1 The challenge of trace transformation

The transformation of traces faces different challenges to bring the traces to a level at which it is possible to perceive user actions and cease to observe technical details of software user actions. Below, the challenges faced in carrying out the trace transformation will be listed, according to the literature review described in Section 2.5 and the comparative Table 5 constructed.

4.1.1 Lack of Standardized Methodology or process for Transformation.

The literature reports systems based on traces and some domains that utilize traces, such as: 1) Process mining, which utilizes traces generated in the production line by machines and equipment to increase efficiency and reduce costs. 2) Trace-based systems (TBS), which involve a process of collecting, transforming, and visualizing traces. Despite the existence of this process, implementation is quite complex as it does not explicitly or thoroughly describe each of the stages, leading to the following questions: Where do I find the traces? How are the traces collected? How are the traces transformed? Is there any algorithm? After transforming them, how do I visualize them?

4.1.2 Granularity Challenges

Granularity in traces refers to the detailed technical description of a user's actions within a computer system. This poses a challenge because, during trace transformation, it is essential to identify the text structures to structure the traces and identify the available transformation parameters. Some transformation parameters include time, date, user, path, functionality or API, content, owner, server response to the request in the case of web applications. These elements are not in the same position within the traces, so identification and structuring involve a lengthy process that depends on the granularity level of the traces. Figure 42, describes the levels of granularity in terms of exploitation complexity.

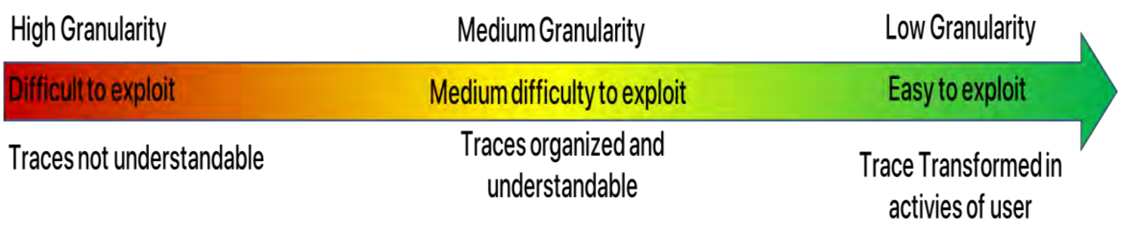


Figure 42. Trace granularity levels

4.1.3 Lack of Tools Documentation

Software or systems capable of generating traces should mention the End-Points of the APIs or in some way provide information on how they were built and what routes are available within them. In the absence of this information, generating transformation rules would require more time and would be done through inferences rather than using a solid foundation offered by trace-generating systems. Traditionally, the type of software that provides this kind of information is referred to as Open Source. However, it is also possible to identify paid systems capable of providing this information.

4.1.4 Traceable vs. Non-Traceable Functionalities

Some project management software (PMS) offers many functionalities that allow managing project activities such as tasks, financial resources, time, human resources, and the projects themselves, however, they do not always generate activity traces of all functionalities. The lack of equivalence between the number of functionalities vs. traceable functionalities makes it impossible to determine which actions were performed directly by the users, thus requiring a transformation based on inference and not on direct use tests.

4.2 A general Trace Transformation Process

The literature mentions processes and methodologies that were not deliberately designed but generated because of seeking to transform computational traces with a particular objective: cybersecurity, improvement of a production process, knowledge management, and recommender systems, among many others. For this research, we considered as a basis the proposal of [23], which describes a general process of trace transformation called the "Trace-Based System" (TBS), which allows the collection, transformation, and visualization of traces described in Figure 43. However, despite the existence of a general trace transformation process, the specific steps to achieve this transformation are not explicitly outlined. This process is one of the most challenging in TBS, as it heavily depends on both the type of trace and the software to which it is applied. In this section, the objective is to address only the transformation process, which in the TBS base proposal does not detail an explicit process to be followed to perform the transformation. It is also important to specify that in other areas of research, transformed traces are used for various purposes, such as anomaly detection and identifying inconsistencies in systems and applications, without necessarily involving the visualization process.

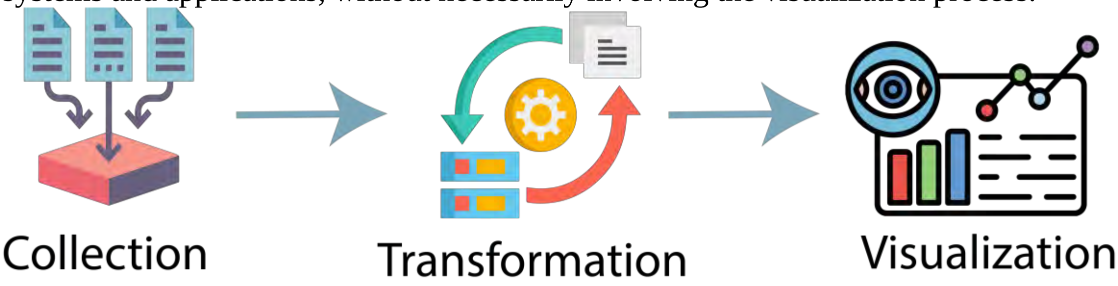


Figure 43. Trace-Based System stages

4.2.1 Building a General Trace Transformation Process

Considering the structure of TBS and the lack of detail in the existing transformation process, an explicit procedure has been developed to facilitate the transformation of traces. For this reason, the focus will be on the transformation phase of the classical TBS. Accordingly, the process will be divided into three major stages, which are illustrated in Figure 44:

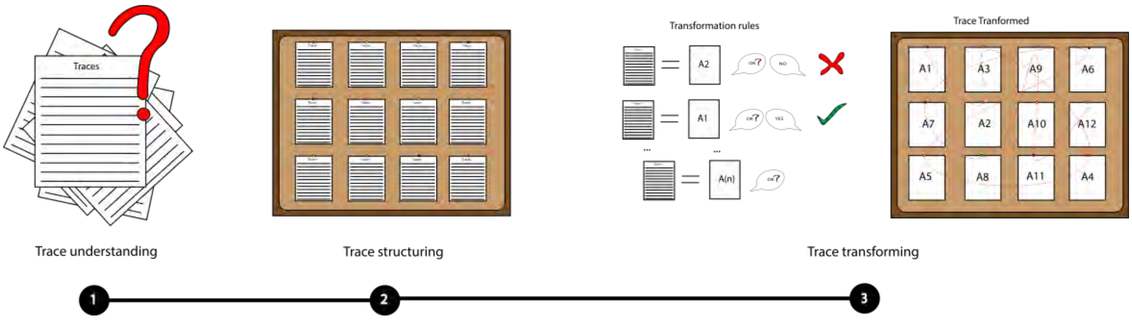


Figure 44. General trace transformation stages

While Figure 44 outlines three main stages: trace understanding, trace structuring, and trace transforming, the transformation process still seems somewhat ambiguous. Therefore, the following sections list the stages along with their descriptions:

Trace understanding

This initial stage of the process involves understanding how traces are generated, the common structure they share, and the selection of common parameters.

- Analyze the structure of the traces.
- Identify specific actions-traces
- Identify and select important parameters

Trace structuring

In this second stage of the process, the use of a programming language is considered to identify trace structures and organize them.

- Apply parsing to structure traces
- Analyze structured traces
- Clean structured traces

Trace transforming

In this third stage, parameters are identified to generate and apply trace transformation rules to obtain user activities.

- Create transformation rules
- Implementation of transformation rules

4.2.2 Modeling a general Trace Transformation Process

In order to model the aforementioned process, which is described in detail in the General Trace Transformation Process (G2TPs) in three distinct stages (trace understanding, trace structuring, and trace transforming), a Business Process Model and Notation (BPMN) model was generated in Figure 45. The model illustrates the progression of activities across the three stages, wherein each stage represents a pivotal aspect of the transformation. It incorporates feedback loops to guarantee precision and accuracy.

The presence of multiple loops in the BPMN model indicates that the process is iterative in nature. Should issues or inconsistencies be identified during the structuring or transforming stages, it is possible to return to a previous step in order to correct the trace data. This serves to reinforce the dynamic and adaptable nature of the process. The model presents an end-to-end workflow, commencing with the initial understanding of the trace and progressing through structured transformations until the process is completed, as indicated by the "End" node. The arrows and loops in the model indicate that alterations or amendments made in one stage can affect subsequent steps. Furthermore, the model employs arrows to illustrate the progression of tasks, with decision points or branches indicated where necessary. This straightforward approach facilitates the visualization of the interconnectivity between stages and the functioning of the feedback mechanisms.

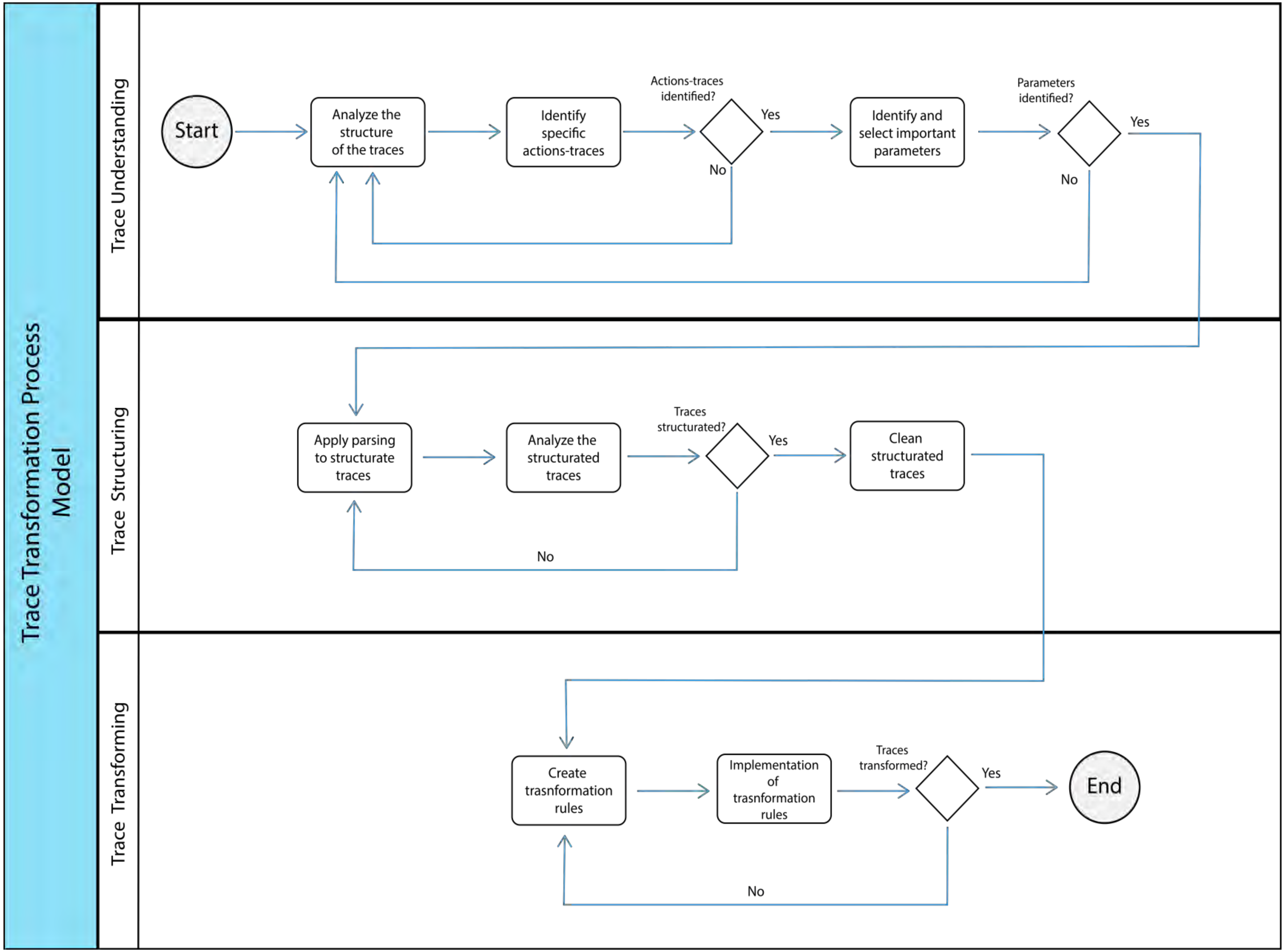


Figure 45. G2TPs modeled in BPMN

4.3 General Trace Transformation process applied to OpenProject

According to the objective of this research, trace transformation is essential to achieve the evaluation of project management maturity based on traces from software within the same domain. For this reason, an open-source software that allows the retrieval of user activity traces was used. OpenProject was selected after a comparison on the website: https://project-management.zone/ , which is regularly updated to add new software, conduct rankings, and enable comparison. To meet this objective, the trace transformation process proposed in this work was applied.

4.3.1 Trace understanding

To obtain traces from OpenProject, a query needs to be made to the web server where the software is installed (Obuntu/Linux) using the command: *sudo openproject logs –n (number of traces)>(Name of the destination file + extension of the file*). This query retrieves the desired number of traces from the OpenProject system and saves them in a file with the .log extension. From this .log file, the transformation process begins with manual analysis of the data structures to identify which user actions generate which traces. Once these actions are characterized, the most important parameters are selected to potentially help understand the actions performed by users within the system. To illustrate this, Figure 46 takes a trace generated as a consequence of a simulation realized in OpenProject software in October 2023 as an example:

```
oct. 31 12:54:40 paco-VirtualBox openproject-web-1.service[1322]: I, [2022-10-31T12:54:40.397853 #1322]  INFO -- : duration=664.29 db=294.31 view=369.98 status=201 method=POST path=/api/v3/projects params={"name"=>"Engineering Project"} User=3
```

Figure 46. Trace extracted from OpenProject

a) Analyze the structure of the traces.

Analysis of the trace described in
reveals information such as date, time, host, status (server response to the request), request duration, view, web service name, method used, path, query parameters and user.

b) Identify specific actions-traces

Identification of traces and user actions considers the elements identified in the trace structure analysis. Using the one obtained from OpenProject as a basis, it was identified that on that date, a project named "Engineering Project" was created within the software, with the simulation performed in October 2023. This allows for generating a link between the user action and the trace.

c) Identify and select important parameters

The identification of important parameters is required to determine automatically and without recapitulation within the software the user actions. Considering the example we can see that some parameters could indicate that a user generated a project. Figure 47 shows some parameters that could be useful to determine user actions. The selected parameters should allow the recognition of a user's action without consulting OpenProject. For this reason, the following parameters were selected: date, time, host, status, method, path, parameters, and user.

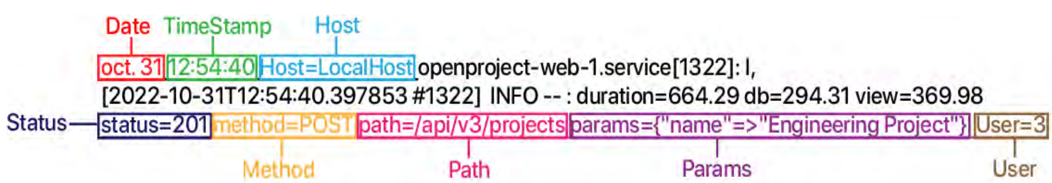


Figure 47. Important parameters

4.3.2 Trace structuring

The structuring of traces requires a natural language processing technique that allows for the search of a text structure (traces) and its extraction into a table. This technique is called regular expression search, which involves using a search algorithm to input the text and obtain a structured table according to specified parameters. In this research, the selected parameters were date, time, host, status, method, path, parameters, and user.

a) Apply parsing to structure traces according to the parameters

Figure 48 shows the regular expressions used to structure the traces considering the selected parameters in Python.

```
def Trace_Analysis(content2):
        data=[]
        for log in content2:
                    date1=re.findall(r"\b20\d{2}-\d{2}-\d{2}T",log)
                    date=date1
                    timestamp=re.findall(r"\s\b(\d{2}:\d{2}:\d{2})",log)
                    status=re.findall(r"\b(status=\d{3})",log)
                    method=re.findall(r"\b(POST|GET|PUT|DELETE|CONNECT|PAT
CH)",log)
                    path=re.findall(r"\b(path=/api/v3/\b(.*?) )",log)
                    params=re.findall(r"\b(params={.*?}})",log)
                    host=re.findall(r"\b(host=\b(.*?) )",log)
                    user=re.findall(r"\b(user=\d(.*?))",log)
                    project=re.findall(r'"project"=>\{"href"=>"[^,]+",
"title"=>"[^"]+"\}\,',log)
                    data.append([set(date),set(timestamp),set(status),meth
od,path,params,host,user,project])
```

Figure 48. Regular expressions considered

After the implementation of the regular expressions, the algorithm goes through all the lines and when it finds a match, it places the ordered information in a table where the attributes are the important parameters selected. To illustrate the example, Table 14 was generated.

Table 14. Example trace structured

Date	Timestamp	Status	Method	Path	Params	Host	Users
jan / 30	14:40:54	**201**	**POST**	**/api/v3/projects**	={"name"=>"Engineering Project", "description"=>{"format"=>"markdown", "raw"=>"", "html"=>""}, "public"=>false, "statusExplanation"=>{"format"=>"markdown", "raw"=>"", "html"=>""}, "_links"=>{"status"=>{"href"=>n>	localhost	3

b) Analyze structured traces

The analysis of the structured traces is relevant to identify gaps that affect the transformation process like empty spaces and to identify if the regular expression was applied correctly. In the analysis of the structured trace used as example it was identified that the mix of parameters: a) status:201, b) method: POST, c) Path:/api/v3/projects, means that a Project was created and in the params we identify the name of the project.

c) Clean structured traces

The process of cleaning structured traces is key after identifying possible blanks. It is also a way to perform a second filtering of traces, since if the trace does not describe a user, this means that it was a trace generated by the software itself at startup or some internal process in which the user is not involved. For this reason, in this thesis work, an algorithm that allows the cleaning of structured traces was also contemplated.

4.3.3 Trace Transforming

The transformation of traces is a mix of domain knowledge expressed using an ontology and a computational algorithm that produces as a result the transformation rules. These transformation rules allows to generate of a link between the generated computer traces and the actions of a user in a specific domain. In this case, the domain to which we applied the research was project management. Also, the Endpoints analysis was considered which is reported in Annex 3.

a) Create transformation rules

With the help of the domain ontology, transformation rules are generated, which are a set of phrases that allow taking relevant parameters of the structured traces and converting them into actions or activities of a domain. In the example taken in this chapter we can infer the following: *If Status=201 & method= POST & Path=/api/v3/projects Then "Project created"*. This indicates that within the OpenProject software when a user creates a new project it generates a trace that contains a *status 201* (successful request), *a POST method* (object creation) and *the path* which describes which API is used in the system

b) Implementation of transformation rules

The implementation of the transformation rules is the heart of the trace transformation because through them it is possible to obtain the users' activities. To automate this process, a capable algorithm was implemented where these transformation rules are inserted in Python. Figure 49, describe an example one transformation rule to identify when a *User Creates a New Project*.

```
comparation75=df[df['Status'].astype(str).str.contains(r'\b201\b') &
df['Method'].str.contains(r'POST') &
df['Path'].str.contains(r'/api/v3/projects\s')]
            comparation75_copy=comparation75.copy()
if len(comparation75_copy)>0:
   comparation75_copy["Anwser"]="Creates a new project."
   comparation75_copy["API"]="Projects"
   comparation75_copy["PM area"]="Project Object Management"
```

Figure 49. Example of transformation rule

After implementing the transformation rules algorithm, a table is obtained in which the date, time, project management activity, and the user who performed this task are indicated. Table 15, describes the example trace transformed.

Table 15. Trace transformed

Date	Timestamp	PM activity	User
janv / 30	14:40:54	“Project created”	3

4.4 General Trace Transformation process applied to OpenProject validation with a Trace-Based Maturity Assessment System

In Section 4.2, a process for transforming computer traces was proposed to reduce granularity and identify user actions within them. This process aligns with the main objective of the research, which is to evaluate project management maturity based on these computer traces. Therefore, the use of a trace-based system encompassing collection, transformation, visualization, and an additional module for project management maturity evaluation is necessary. To validate this implementation, a system was developed that divides tasks into modules for login, collection, transformation, visualization, and maturity evaluation.

4.4.1 Login module
The login module oversees validating if the user that is trying to use the system is the administrator. The user will be prompted to enter the username and password to provide access. Figure 50, shows a preview of the login module.

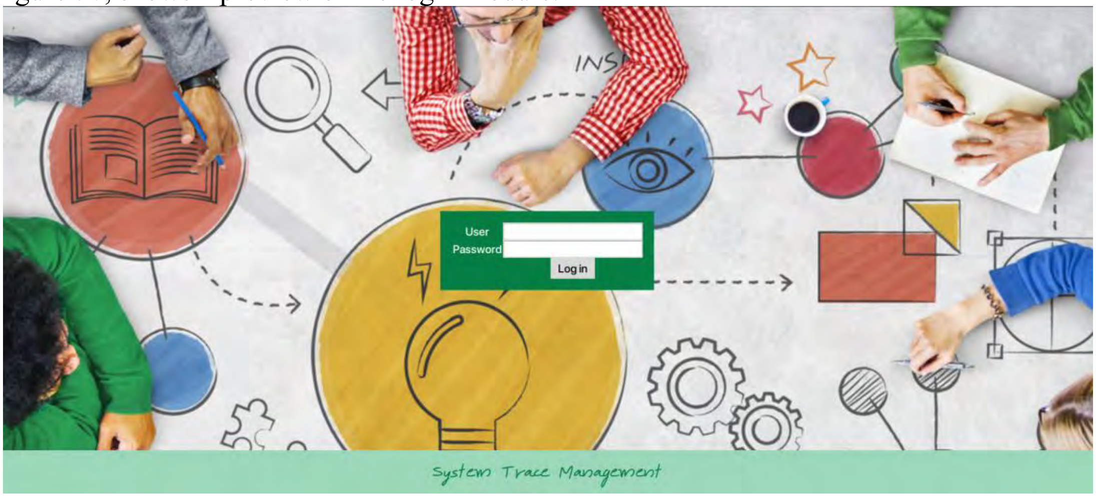


Figure 50. Login module view

4.4.2 Home page module
The home page module allows the selection of functionality within the system, for the purposes of this research the system allows the selection of "Maturity Assessment". More functionalities could be added to the trace management system in the future. A preview of the home page module is shown in Figure 51.

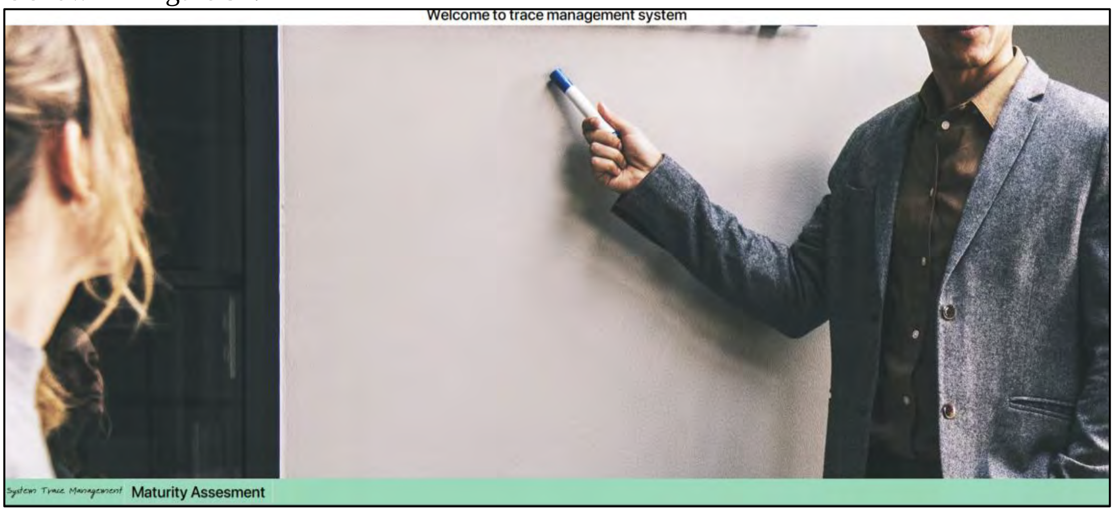


Figure 51. Home page module

4.4.3 Collection module
The collection module allows you to select the root and the file where the logs are stored. The formats supported by the collection module are .log and txt. However, csv files are also supported. A preview of the collection module is shown in Figure 52.

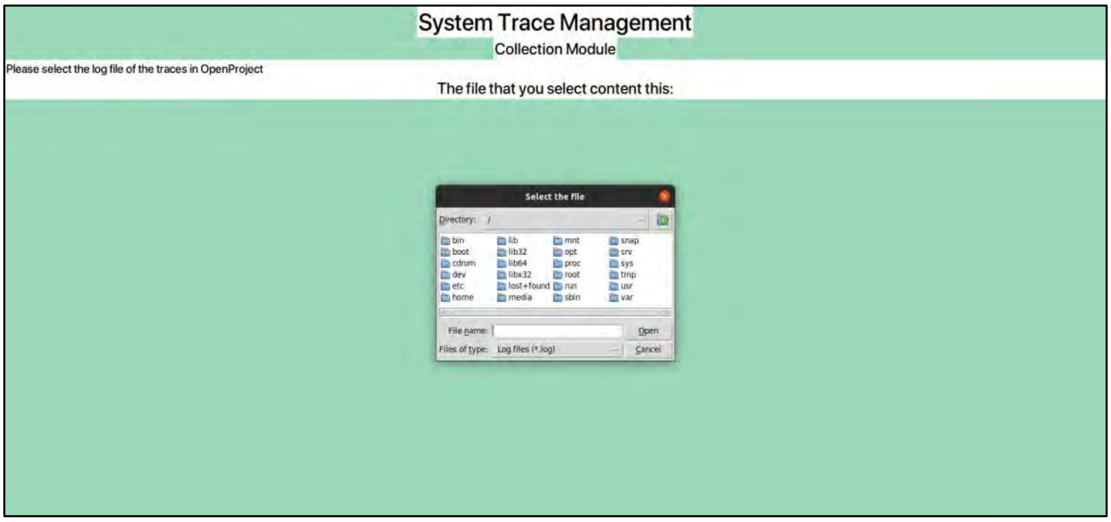


Figure 52. Collection module

4.4.4 Transformation module

The processing module allows the transformation of the traces from a high level of granularity to a low level allowing a better understanding of the data. For this process, the traces collected in the collection module are considered. To perform the transformation, a process was proposed to transform text files into ordered information and to re-label data for transformation of user actions into project management activities. Figure 53 shows a preview of the processing module.

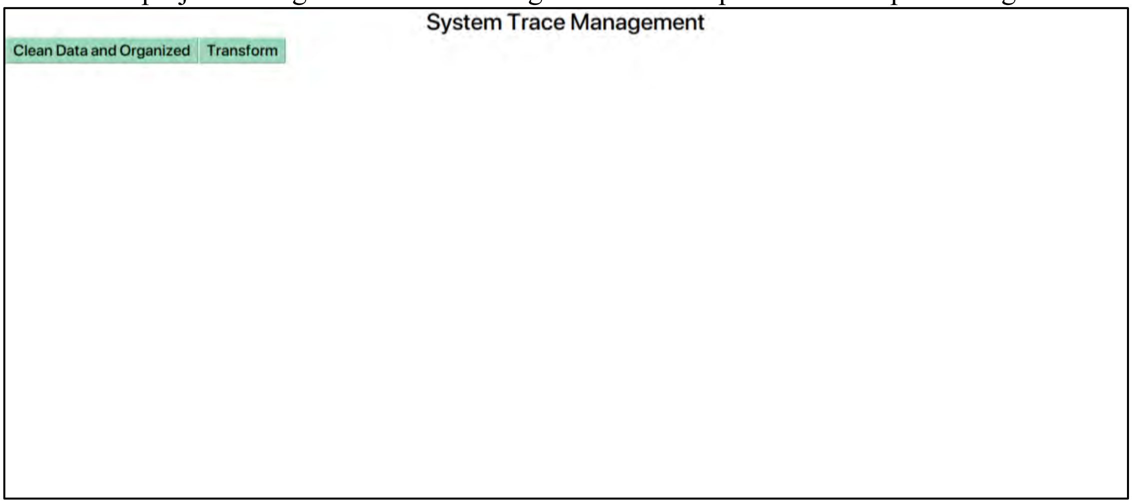


Figure 53. Transformation module

4.4.5 Visualization module

The visualization module oversees displaying the transformed traces allowing the trace management system user to identify which system functionalities were used and which project management activities were performed. Figure 54 shows the visualization module.

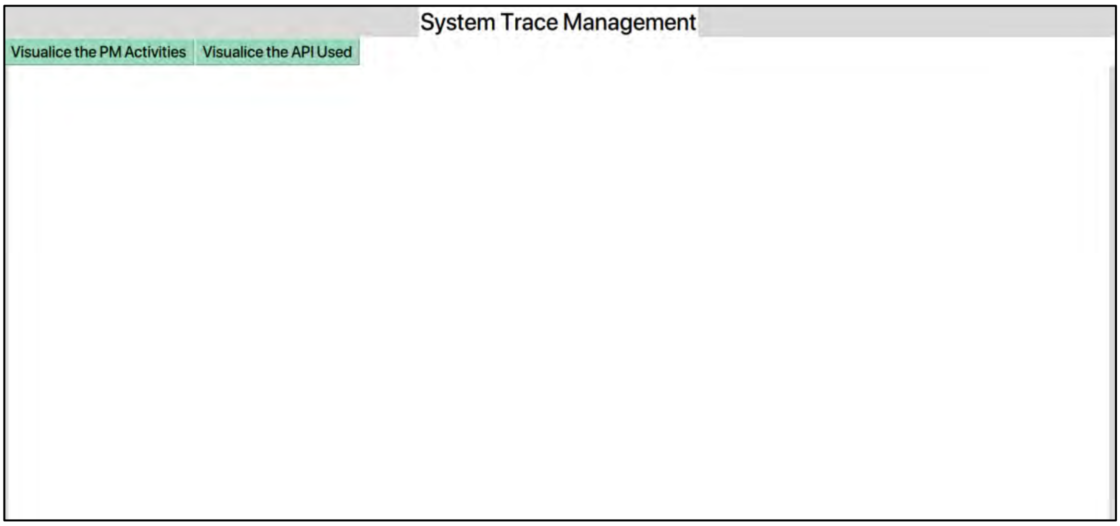


Figure 54. Visualization module

4.4.6 Maturity assessment module

The maturity assessment module oversees measuring and providing a maturity level considering the interaction traces provided. For this module, the utilization rate of OpenProject functionalities that are considered as project management activities was considered. Figure 55 shows a preview of the collection module.

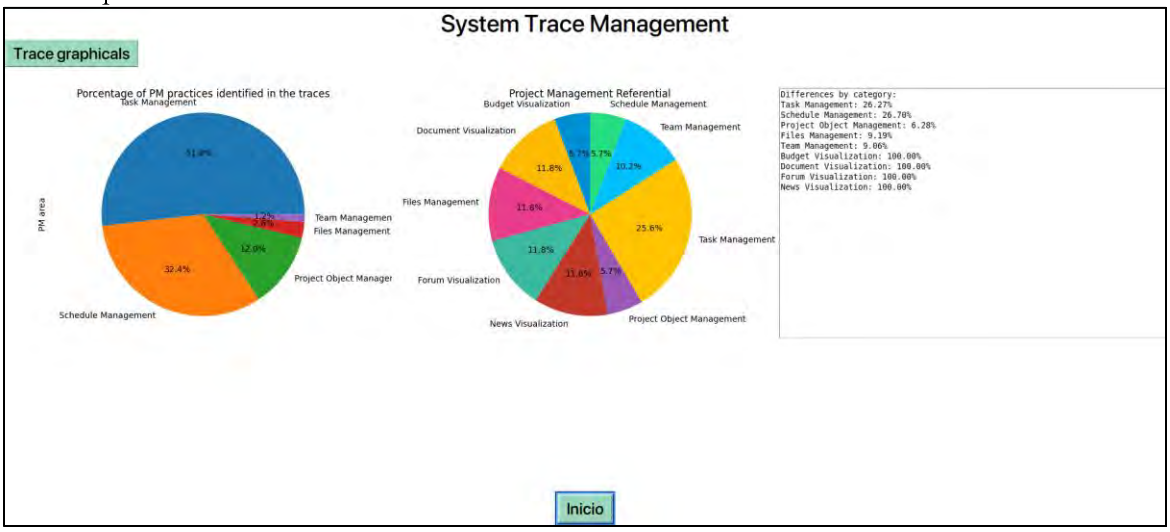


Figure 55. Maturity assessment module

4.5 Chapter 4 Summary

In this chapter, the challenges faced in the transformation of traces when seeking their exploitation for a specific purpose were addressed. In the case of this doctoral thesis, the objective was to evaluate the maturity of project management. Additionally, a process was proposed to perform the transformation of traces using an algorithm that reduces the level of granularity.

The complexity in the transformation of traces begins with the lack of methodologies or general processes. Without these elements, those interested in performing the transformation are unaware of the necessary tools and knowledge required to exploit the traces. Despite the existence of some proposals that describe a process for the exploitation of traces, these do not do so explicitly. For example, TBS mentions a process for the exploitation of traces that involves collection, transformation, and visualization. These elements, which we can call facilitators or transformation guides, allow for the reduction of granularity and the identification of user actions.

On the other hand, the complexity not only lies in the lack of these facilitators. The tools that generate traces must also explicitly state in their documentation how the traces are generated. However, this type of information can only be provided by open-source software providers, who make explicit efforts to help explain how their systems are constructed and how traces are generated, although exceptions may exist. Another important point regarding the challenges in the exploitation of traces is the system functionalities. If these do not generate traces, it is very complex to identify or know if a user has acted. The proposal of this chapter considers that the most complex process in the exploitation of traces is the transformation. Therefore, a process is proposed that involves understanding the traces, structuring them, and finally, transforming them.

4.6 Chater 4 Discussion

The literature review allowed for the identification of current limitations and challenges in the trace transformation process. This section expands on the concept of trace-based systems (TBS), which typically encompass the stages of collection, transformation, and visualization. This general framework, while informative, poses significant challenges for the implementation or programming of systems capable of executing these tasks effectively. Currently, the stages of collection and visualization are widely adopted across various domains for data analysis, data mining, and data science. However, the data transformation stage introduces considerable complexity, as it is contingent upon the specific context, involved elements, systems, and applications.

Trace transformation is particularly intricate due to the granularity of traces generated directly from user interactions with the system. Despite this complexity, a generalized process for trace transformation applicable across different domains is deemed achievable. The proposed process delineates three main stages within the transformation process: trace comprehension, trace structuring, and finally, transformation. This proposed process has been implemented in a system developed from scratch in Python, utilizing Tkinter libraries, which facilitated the creation of modules dedicated to trace transformation.

While the literature review identified several existing processes for trace transformation and contemporary methodologies, it noted that these approaches do not specifically address project management or the application of traces in assessing project management maturity. This discussion highlights how the proposed process aims to address the current limitations in trace transformation

Lack of Standardized Methodology or process for Transformation: The literature describes various methodologies and processes for performing trace transformation. However, none have been identified that pay particular attention to the stage of pure transformation. A clear example is the Trace-Based Systems (TBS) model, which states that for trace exploitation, three stages are necessary: collection, transformation, and visualization. Nonetheless, this model does not specify the appropriate process to follow for trace transformation. Therefore, this chapter's contribution proposes a specific process for the transformation stage that complements what is suggested by TBS. This process consists of three fundamental phases:

1. **Understanding the Traces:** This phase addresses the question: **How were the traces generated?** Understanding the context and conditions under which the traces were produced is crucial to ensure that the transformation is relevant and accurate.
2. **Structuring the Data:** Here, the question is: **How do I organize my data before transforming it?** Establishing a clear framework for handling the information before proceeding with the transformation is vital, as it will facilitate subsequent analysis.
3. **Pure Transformation:** This final stage focuses on answering: **What are the rules that allow me to translate structured data into user activities, and what does this mean?** Defining these rules is essential to ensure that the transformation is effective and that the results are understandable.

By addressing these three phases, a more robust framework for trace transformation is established, thereby contributing to the enhancement of existing methodologies in this field.

Granularity challenges: Although the literature does not explicitly address the issue of granularity in trace transformation, it is a likely recognized challenge in trace processing. Traces with excessive irrelevant information (high granularity) are more difficult to process, making their exploitation complex. On the other hand, low-granularity traces make it easier to identify key elements or parameters for transformation. This chapter proposes a trace transformation process that specifically addresses the issue of granularity by reducing it where necessary. The structuring stage, in particular, plays a crucial role in managing this reduction, making the traces more suitable for transformation by preparing them for the application of transformation rules in the subsequent stage.

Lack of Tool Documentation: Effectively exploiting traces requires a thorough understanding of how they are generated within the system. This involves identifying the elements that interact when a user engages with specific functionalities. While consulting the documentation provided by the software owners or developers is often necessary, this information is not always explicitly available, necessitating a deeper analysis of the tool itself. Despite these challenges, it is still possible to infer the trace generation process through careful analysis.

Open-source tools generally provide more transparency about their internal workings, making it easier to understand how traces are generated. However, proprietary tools that do not disclose their architecture or trace generation mechanisms present a more complex challenge. In such cases, trace exploitation often relies on a trial-and-error approach to map traces to specific activities and establish meaningful links.

The lack of comprehensive documentation significantly impedes effective trace exploitation. In response to this, we propose a transformation process that includes an understanding stage. This stage outlines a series of steps to grasp how traces are generated, even when explicit information is not provided by the tools, ensuring a more complete and reliable trace transformation.

Traceable vs. Non-Traceable Functionalities: Management tools are designed to support users in their processes and enhance the likelihood of achieving organizational goals. To achieve this, these tools digitize tasks that would otherwise be performed manually, thereby improving accessibility and fostering collaboration. These tasks are translated into system functionalities, enabling users to carry out their activities effectively. However, this primary objective does not always guarantee a one-to-one correspondence between system functionalities and trace generation.

As a result, some functionalities are traceable, while others do not generate traces even when they are utilized. This inconsistency presents a significant limitation for trace transformation, as non-traceable functionalities hinder both the transformation and exploitation of traces. When a user performs an action that cannot be traced, it creates a gap in the process, potentially leading to misconceptions that the action was never executed.

In this chapter, we propose a process to better understand how traces are generated and to identify potential errors or missing functionalities within the system or tool. This understanding is crucial for improving traceability and ensuring a more comprehensive approach to the transformation process.

Finally, the following Chapter 5 describes the third main contribution of this thesis work, the feasibility study on a management tool to determine its suitability to generate useful traces for assessing the maturity of a domain. An architecture is also presented as a proposed recommendation for a project management software tool.

Chapter 5. Feasibility study on an activity management software tool to define the ideal architecture for conducting a trace-based maturity

As a result of implementing a trace-based system to assess project management maturity described in Chapter 4, a relationship was identified between the number of system functionalities and the number of best practices. This relationship suggests that software should encompass the necessary functionalities to execute the good practices outlined in the PMBOK or any other project management reference. If users cannot perform such practices, the maturity evaluation becomes limited to considering only the supported best practices within the system. Furthermore, this also strongly correlates with the traces generated by the system; without functionality, there will be no records indicating whether a user performed a specific activity. Additionally, while systems may include all necessary functionalities to execute best practices, not all functionalities generate traces. This chapter addresses the challenge of the lack of traceable functionalities for executing best practices within the system. It proposes a software architecture aimed at facilitating the implementation of best practices as described in the PMBOK. The chapter is organized as a follow: In Section 5.1, the process to perform feasibility study on maturity assessment using traceability in an activity management software tool is described. In Section 5.2, the proposed process to perform feasibility study is applied to OpenProject. In Section 5.3, the proposed process for conducting a feasibility study is applied to best practices reported in the literature.. In Section 5.4, a comparison of the feasibilities is performed between OpenProject and best practices. In Section 5.5, the is outlined. Also a complementary sections: 5.6 Chapter 5 Summary and 5.7 Chapter 5 discussion are addressed in the end of this chapter.

5.1 Feasibility study on maturity assessment based on traces from activity management software tools.

The feasibility analysis in this context aims to determine whether an activity management software tool is capable of generating system functionality traces. Trace generation is a key feature for maturity assessment, as it serves as an evaluation element used to determine if users have followed best practices within the system, enabling an accurate estimation of maturity. The feasibility analysis is divided into two main stages: functionality mapping and the application of graph theory. The functionality mapping process aims to determine whether the system's functionalities are traceable and empirically assess whether they ultimately enable the implementation of best practices within a specific domain. On the other hand, the application of graph theory proposed by [120] allows for the evaluation and measurement of the impact of these functionalities in achieving best practices. To provide a better understanding of the feasibility analysis process, the following components are listed:

A) Functionality Mapping

- Selection of an activity management software tool
- Review of software documentation
- Identification of software architecture
- Analysis of functionalities and users
- Analysis of domain best practices
- Link functionalities vs best practices
- Build a mapping table

B) Application of graph theory (Gephi)

- Categorize nodes by groups (best practices, functionalities)
- Build the table of nodes (Gephi)
- Build the model (Gephi)
- Apply the graph theory to the nodes
- Perform the analysis of graph
- Analysis of results

5.2 Feasibility study on maturity assessment based on traces from activity management software tools, using OpenProject functionalities as the criterion.

5.2.1 Functionality Mapping

1. Selection of an activity management software tool

The OpenProject tool offers various functionalities closely linked to the project lifecycle. These include the ability to create a new project, establish its structure, configure parameters, add members, gain an overview of projects, structure work, plan roadmaps, manage work packages, use boards, conduct meetings, manage news, develop wikis, assign roles, utilize dashboards, set budgets, track time, control costs, generate reports on time and costs, and maintain project files.

2. Review of software documentation

The OpenProject (OP) tool is open-source, which provides access to a wealth of information regarding its construction and development on its official website: https://www.openproject.org/docs/. This site contains detailed documentation, including user guides, technical documents, blog, website, API, and release notes.

3. Identification of software architecture

The OP tool is composed of a Frontend and a Backend. Figure 4 (Chapter 1, Section 1.2 Referential tools) illustrates the architecture of the application. The architecture of OpenProject

(OP) includes a frontend built with HTML, JavaScript, and CSS, and a web server that manages the web application, stores data in a database, and logs user activity. User files containing traces are also stored in the system. OP uses a permission-based system with default profiles: project admin, staff, project manager, reader, and member. The project admin can fully manage projects, while project managers and staff can view, modify, and request information. Readers and members have view-only access. The software also allows for the creation of custom user profiles.

4. Analysis of functionalities and users

OP tool consists of 20 modules and approximately 39 APIs, which log user activities for each action performed within the system. The modules encompass a set of functionalities dedicated to project management, while the APIs are smaller applications that facilitate these activities within the modules. Figure 5 (Chapter 1, Section 1.2 "Referential Tools") details the user permissions for each module. By default, the modules are managed by three main user roles: Project Admin, staff and project manager, and reader/member. The configuration of user rights can be modified, and new roles can be created by the "Project Admin".

5. Analysis of domain best practices

For this analysis the 6th version of the Project Management Book of Knowledge (PMBOK) was considered as a referential of project management best practices.

6. Link functionalities vs best practices

For identifying connections by searching PM activities/functionalities within each PMBOK process was applied. If these connections were not directly identified, one or two intermediaries were explored until the link with the management activity was established. Generally, each PMBOK process is divided into three parts: inputs, which encompass activities or input documents; tools and techniques, which are the methods and approaches used in a process; and outputs, which consist of activities or documents resulting from the process. Figure 56 details the logic employed to discover and establish the connection between project management activities and PMBOK sub-processes.

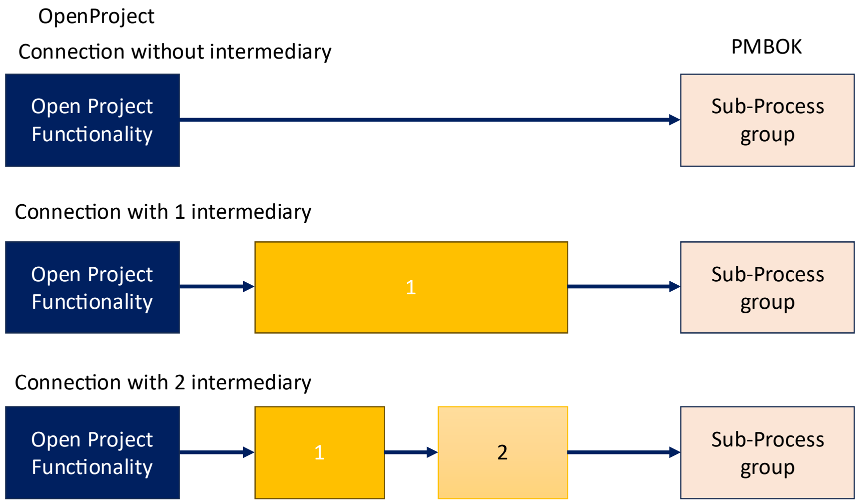


Figure 56. Types of connections (OpenProject Intermediary Vs PMBOK).

7. Build a mapping table

To perform this graph theory analysis, Table 16 was developed, containing all traceable project management activities in OP and the Processes that are impacted. Also is detailed the type of connection between functionalities and PMBOK processes. To build the table the elements. Functionalities of OpenProject, intermediaries 2, intermediaries 1, input, tools & techniques and outputs, were taken into account to analyze the connection between the PM groups of process and functionalities in the software OpenProject. Also the intermediaries that allow to connect the functionality with the groups of processes.

Table 16. Mapping of OpenProject management activities Vs processes (PMBOK)

PM Activities (OpenProject)	Intermediary 2	Intermediary 1	Input	Tools & Technique	Outputs
			Project Management Group of Process		
Team Management	-	Resource Management Plan	-	-	9.1 Plan Resource Management
	-	Project Management Plan	9.4 Develop Team	-	-
	-	Work Performance Reports	9.5 Management Team	-	-
Task Management (WBS, Project Schedule Management)	-	Scope Baseline	-	-	5.4 Create WBS
	Organizational Procedures	Schedule Management Plan	-	6.1 Plan Schedule Management	-
	Organizational Procedures Links	Cost Management Plan	-	-	7.1 Plan Cost Management
	-	Project Management Plan	7.2 Estimate Costs	-	-
	Planned value	Data Analysis	-	7.4 Control Costs	-
	-	Data Representation	-	9.1 Plan Resource Management	-
	Resource Breakdown Structure	Data Representation	-	9.1 Plan Resource Management	-
	-	Risk Categorization	-	11.3 Perform Qualitative Risk Analysis	-
	-	Schedule Management Plan	-	-	6.1 Plan Schedule Management
	Scope Baseline	Project Management Plan	6.2 Define Activities	-	-
	-	Decomposition	-	6.2 Define Activities	-
	-	Rolling Wave Planning	-	6.2 Define Activities	-
	-	Activity List	-	-	6.2 Define Activities
	-	Activity Attributes	-	-	6.2 Define Activities
	-	Milestone List	-	-	6.2 Define Activities
	Schedule Baseline	Project Management Plan Updates	-	-	6.2 Define Activities
	Schedule Management Plan	Project Management Plan	6.3 Sequence Acitvities	-	-
	Scope Baseline	Project Management Plan	6.3 Sequence Acitvities	-	-
		Project Management Information System	-	6.3 Sequence Acitvities	-
	Schedule Management Plan	Project Management Plan	6.4 Estimate Activity Durations	-	-
	Scope Baseline	Project Management Plan	6.4 Estimate Activity Durations	-	-
	Schedule Management Plan	Project Management Plan	6.5 Develop Schedule	-	-
	Scope Baseline	Project Management Plan	6.5 Develop Schedule	-	-
	-	Project Management Information System	-	6.5 Develop Schedule	-
	Bar Charts	Project Schedule	-	-	6.5 Develop Schedule
	Milestone Charts	Project Schedule	-	-	6.5 Develop Schedule
	Schedule Management Plan	Project Management Plan	6.6 Control Schedule	-	-
	Schedule Baseline	Project Management Plan	6.6 Control Schedule	-	-
	-	Project Management Information System	-	6.6 Control Schedule	-
Project Object Management	-	Project Charter	-	-	4.1 Develop Project Charter
Working Days Schedule Management	-	Project Calendars	-	-	6.5 Develop Schedule
	-	Resource Calendars	-	-	9.3 Acquire Resources
Files Management	Organizational Standard polices, processes, and procedures	Organizational Process Assets	4.4 Manage Project Knowledge	-	-
	-	Virtual Teams	-	9.4 Develop Team	-
Budget Visualization	-	Project Management Information System	-	7.2 Estimate Costs	-
	-	Project Management Information System	-	7.4 Control Costs	
Document Visualization	-	Project Management Information System	-	4.3 Direct and Manage Project Work	-
	-	Information Management	-	4.4 Manage Project Knowledge	-
Forum Visualization	Electronic Project Management Tools	Project Management Information System	-	10.2 Manage Communications	-
News Visualization	Social Media Management	Project Management Information System	-	10.2 Manage Communications	-

5.2.2 Application of graph theory (Gephi)

1. Build the model (Gephi)

Based on Table 13, data was integrated into the Gephi tool to construct a node diagram that graphically represents the interactions between OpenProject (OP) functionalities and PMBOK (6th edition) processes. Labels were added to the nodes for classification:

- PP-PMBOK: PMBOK Sub-Processes (49)
- PG-PMBOK: PMBOK Process Groups (10)
- F-PM-AC: Single Intermediary Connector (12)
- GF-M-PM-AC: Complex Intermediary Connector (23)
- PM-AC: Software functionalities (9)

The integration of the table data into Gephi resulted in 103 nodes and 153 edges. Two types of edges were identified: those related to PMBOK processes (Input, Output, Tool, and Technique) and those associated with connector types between functionalities and PMBOK processes (Level 1 and Level 2). A Additionally, the edges need to specify whether they are directed or undirected. In this case, the edges that establish a link between functionalities and PMBOK processes are considered directed. Figure 57 shows an example of the node table, while Figure 58 presents an example of the edges integrated in Gephi.

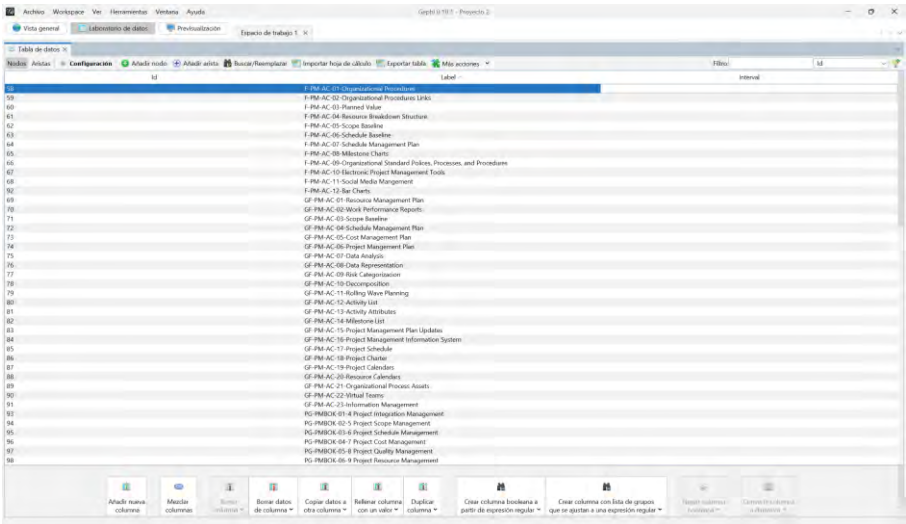


Figure 57. Nodes integrated in Gephi (OpenProject graph theory analysis)

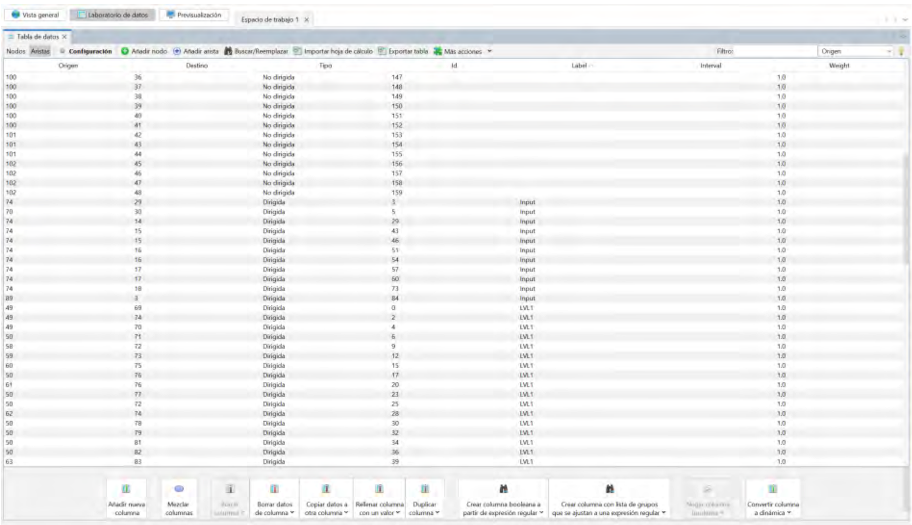


Figure 58. Edges integrated in Gephi (OpenProject graph theory analysis)

The nodes and edges diagram (Figure 59 described below), visually presents the resulting node diagram (data described in Annex 1) from this analysis. This representation reveals that processes related to the project lifecycle, especially in planning, directly impact OpenProject. This suggests that the software is more focused on planning functionalities. However, processes such as risk, quality, and costs show weak or no impact on OpenProject. This observation suggests that the current functionalities of the software may not be sufficient to cover all processes described in the PMBOK. This raises the following question: Is it possible to assess project management maturity with the processes currently impacted by OpenProject from the PMBOK?

To address this question, an analysis of traditional project management maturity models was conducted to determine if in this case, best practices impact all PMBOK processes. These models currently focus on evaluating project management best practices rather than assessing PMBOK processes. This approach presents significant challenges as it describes the practices to be evaluated but does not explicitly specify which part of the project lifecycle or which PMBOK process those best practices are attributed to. As identified in the systematic review, the maturity models analyzed focus on evaluating best practices, typically obtained from references or surveys of project management expert groups.

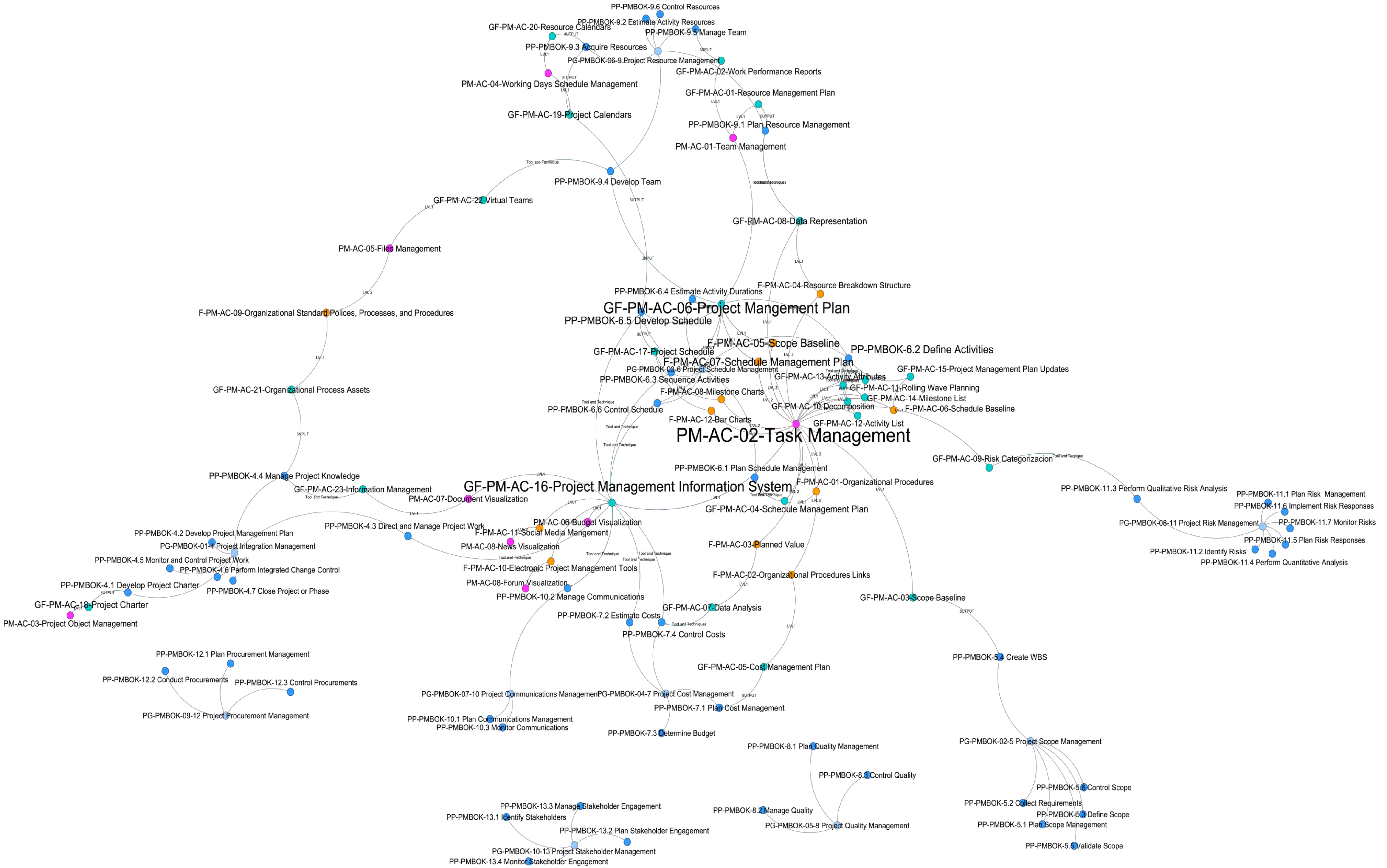


Figure 59. Nodes and edges diagram (OpenProject vs PMBOK).

2. Apply the graph theory to the nodes

The complexity of the graph is 2.971, indicating a high degree of complexity. This complexity is interpreted in the context of graph theory, where complex graphs begin from a value of 1. Furthermore, the graph's complexity also suggests its resilience, being in this case a strong graph. This means that even if a considerable number of edges were lost, the graph would still maintain communication between nodes.

3. Analysis of results

Although graph theory allows for the identification of the complexity of a network and its most influential elements, this analysis also enables the identification of the impact that OpenProject functionalities have on the PMBOK. Figure 60 shows a bar chart was generated to count how many functionalities of OpenProject impact each PMBOK sub-process. In Figure 60, it was identified that 19 PMBOK processes are impacted by the nine traceable PM activities/functionalities that a user can perform in OpenProject.

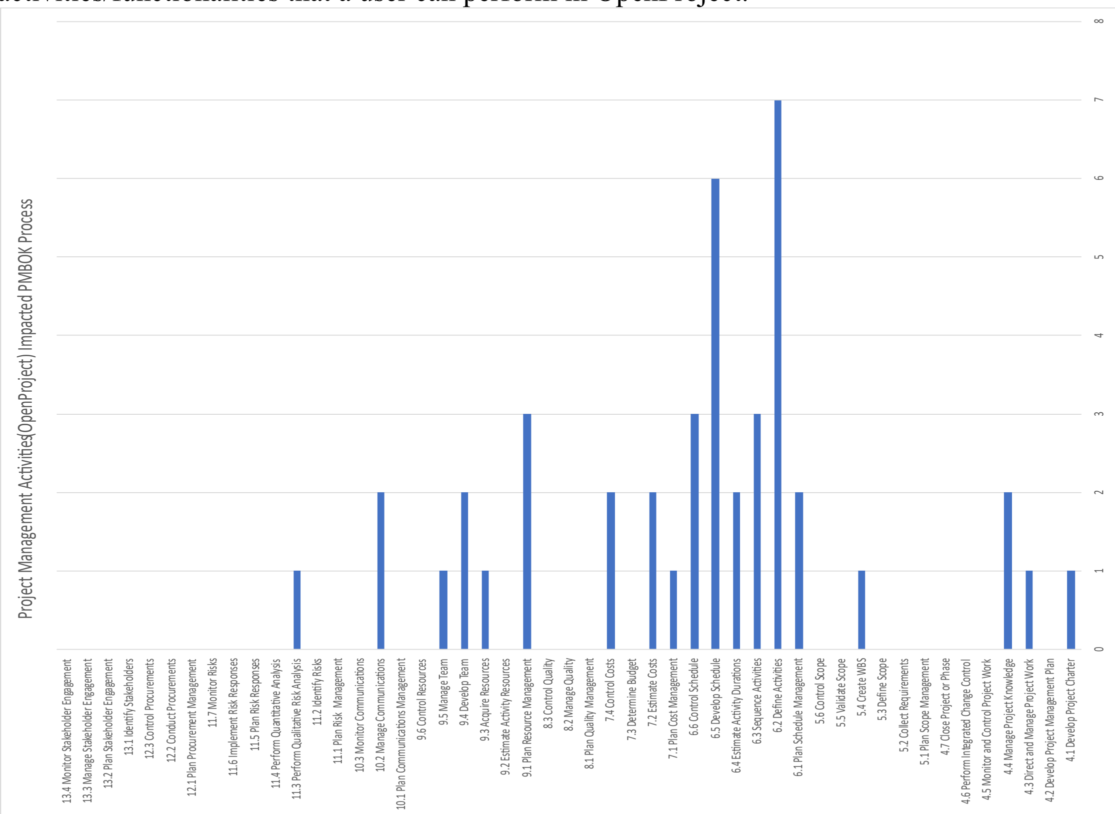


Figure 60. Mapping of impacted processes (PMBOK) by OpenProject.

5.3 Feasibility study on maturity assessment based on traces from activity management software tools using best practices as a criterion

After conducting a feasibility analysis of the OpenProject tool, it was identified that there is a lack of project management functionalities. To address this, the empirical existence of software based on best practices in project management, as reported in the literature, was considered. Recognizing best practices as a key element for evaluating maturity, a mapping of these practices against PMBOK processes was conducted to determine whether maturity models focus on specific aspects or encompass all processes.

5.3.1 Functionality Mapping

Given that this feasibility study aims exclusively to propose the empirical existence of a project management tool that would enable the implementation of the 46 best practices described in the literature, the following steps are omitted:

1) Selection of an activity management software tool,
2) Review of software documentation,
3) Identification of software architecture
4) Analysis of functionalities and users

1. Analysis of domain best practices

For this analysis, the list of 46 best practices used by [32] obtained from [121] were compared with the 49 processes of the PMBOK. Table 17 details the best practices considered in the mapping.

Table 17. Project Management Best Practices

Activity List	Life Cycle Costs (LCC)
Baseline plan / Project Baseline	Monitoring of Cost
Bid documents	Monitoring Schedule
Bid/seller evaluation	Progress Communication Meeting
Bidders conferences	Progress Report
Change Request	Project Charter
Client acceptance form	Project Web Site
Communication Plan	Quality Function Deployment (QFD)
Configuration review	Quality Inspection / Quality Control / Quality Assurance
Contingent Plans	Quality Plan
Control Charts: Gantt chart, S-curve, cause-and-effect diagram	Re-baselining
Cost Estimating, Parametric Estimating	Requirements Analysis
Customer Satisfaction Surveys	Resource Scheduling
Database for Lessons Learned	Resources Leveling
Database of Contractual Commitment Data	Responsibility Assignment Matrix
Database of historical data	Risk Management
Database of Risks	Scope Statement
Earned Value Management	Self-directed Work Teams
Feasibility Study	Stakeholder Analysis
Financial Measurement Tools	Team Building Event
Graphical presentation of risk information	Team Learning Curve
Kick-off Meeting	Team Member Performance Evaluation
Lessons Learned/Post-Mortems	Work Breakdown Structure (WBS)

2. Link functionalities vs best practices

For this analysis, the approach was to identify connections by searching for best practices within each PMBOK process. If these connections were not directly identified, one or two intermediaries were explored until the link with the management activity was established. Generally, each PMBOK process is divided into three parts: inputs, which encompass activities or input documents; tools and techniques, which are the methods and approaches used in a process; and outputs, which consist of activities or documents resulting from the process.

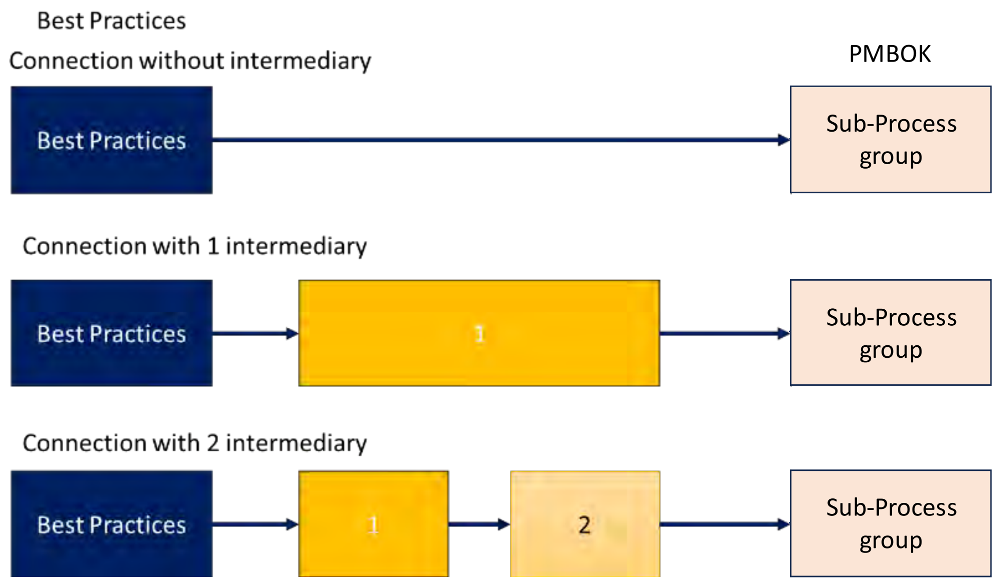


Figure 61. Types of connections (Intermediary Best Practices Vs PMBOK)

3. Build a mapping table

To perform this graph theory analysis, Table 18 was developed, containing 46 best practices reported in literature and the Processes that are impacted. Also is detailed the type of connection between best practices and PMBOK processes.

Table 18. Mapping of Best Practices Vs Processes (PMBOK)

PM Activities (OpenProject)	Intermediary 2	Intermediary 1	Input	Tools & Technique	Outputs
Project Charter	-	-	4.2 Develop Project Management Plan	-	4.1 Develop Project Charter
	-	-	4.7 Close Project or Phase	-	-
	-	-	5.1 Plan Scope Management	-	-
	-	-	5.2 Collect Requirements	-	-
	-	-	5.3 Define Scope	-	-
	-	-	6.1 Plan Schedule Management	-	-
	-	-	7.1 Plan Cost Management	-	-
	-	-	8.1 Plan Quality Management	-	-
	-	-	9.1 Plan Resource Management	-	-
	-	-	10.1 Plan Communication Management	-	-
	-	-	11.1 Plan Risk Management	-	-
	-	-	12.1 Plan Procurement Management	-	-
	-	-	13.1 Identify Stakeholder	-	-
	-	-	13.2 Plan Stakeholder Engagement	-	-
Feasibility Study	-	Business Documents	4.7 Close Project or Phase	-	-
Stakeholder Analysis	-	Data Analysis	-	11.1 Plan Risk Management	-
	-	Data Analysis	-	13.1 Identify Stakeholders	-
	-	Data Gathering	-	13.2 Plan Stakeholder Engagement	-
	-	Data Analysis	-	13.4 Monitor Stakeholder Engagement	-
Requirements Analysis	-	Expert Judgement	-	5.2 Collect Requirements	-
	-	Product Analysis	-	5.3 Define Scope	-
Work Breakdown Structure (WBS)	-	Scope Baseline	-	-	5.4 Create WBS
	Organizational Procedures	Schedule Management Plan	-	6.1 Plan Schedule Management	-
	Organizational Procedures Links	Cost Management Plan	-	-	7.1 Plan Cost Management
	-	Project Management Plan	7.2 Estimate Costs	-	-
	Planned value	Data Analysis	-	7.4 Control Costs	-
	-	Data Representation	-	9.1 Plan Resource Management	-
	Resource Breakdown Structure	Data Representation	-	9.1 Plan Resource Management	-
	-	Risk Categorization	-	11.3 Perform Quqlitqtive Risk Analysis	-
Activity List	-	Project Document Updates	-	-	4.3 Direct and Manage Project Work
	-	Organizational process Assets	6.2 Define Activities	-	-
	-	Decomposition	-	6.2 Define Activities	-
	-	-	-	-	6.2 Define Activities
	-	Project Documents	6.3 Sequence Activities	-	-
	-	Project Documents Updates	-	-	6.3 Sequence Activities
	-	Project Documents	6.4 Estimate Activity Durations	-	-
	-	Project Documents	6.5 Develop Schedule	-	-
	Activity Attributes	Project Documents	9.2 Estimate Activity Resources	-	-
	-	Project Documents	9.2 Estimate Activity Resources	-	-
Scope Statement	Scope Baseline	Project Management Plan	-	-	4.2 Develop Management Plan
	-	Scope Management Plan			5.1 Plan Scope Management
	-	Project Scope Statement	-	-	5.3 Define Scope
	-	Project Documents	5.4 Create WBS	-	-
	-	Scope Baseline	-	-	5.4 Create WBS
	Scope Baseline	Project Management Plan Updates	-	-	5.6 Control Scope
	Scope Baseline	Project Management Plan	-	-	6.5 Develop Schedule
	Scope Baseline	Project Management Plan	7.2 Estimate Costs	-	-
	Project Scope Statement	Project Management Plan	7.2 Estimate Costs	-	-
	Scope Baseline	Project Management Plan	7.3 Determine Budget	-	-
	-	Scope Baseline	8.1 Plan Quality Management	-	-
	-	Deliverables	8.3 Control Quality	-	-
	Scope Baseline	Project Management Plan	12.1 Plan Procurement Management	-	-
Risk Management	-	Organizational process Assets	11.1 Plan Risk Management	-	-
	-	Organizational process Assets	11.2 Identify Risks	-	-
	List of identied risks	Risk Register	-	-	11.2 Identify Risks
Contingent Plans	-	-	-	-	-
Resource Scheduling	-	-	-	-	-
Cost Estimating, Parametric Estimating	-	Organizational process Assets	7.1 Plan Cost Management	-	-
	-	Expert Judgement	-	7.1 Plan Cost Management	-
	-	Organizational process Assets	7.2 Estimate Costs	-	-
	-	Expert Judgement	-	7.2 Estimate Costs	-
	-	Analogous Estimating	-	7.2 Estimate Costs	-
	-	Bottom-Up Estimating	-	7.2 Estimate Costs	-
	-	Project Management Information System	-	7.2 Estimate Costs	-
Quality Plan	-	Testing/Product Evaluations	-	8.3 Control Quality	-
Communication Plan	-	Virtual Teams	-	9.3 Acquire Resources	-
	-	Interpersonal and Team Skills	-	9.5 Manage Teams	-
Kick-off Meeting	-	Meetings	-	4.2 Develop Project Management Plan	-
	-	Meetings	-	11.1 Plan Risk Management	-
Responsibility Assignment Matrix	-	-	-	-	-
Self-directed Work Teams	-	-	-	-	-
Team Building Event	-	-	-	-	-
Progress Communication Meeting	-	-	-	-	-
Quality Inspection / Quality Control / Quality Assurance	Requirements documentation	Project Documents	8.1 Plan Quality Management	-	-
	-	Expert Judgement	-	8.1 Plan Quality Management	-
	-	Quality Management Plan	-	-	8.1 Plan Quality Management
	Quality Control Measurements	Project Documents	8.2 Manage Quality	-	-
	-	Quality Improvements Methods	-	8.2 Manage Quality	-
	-	Project Management Plan	8.3 Control Quality	-	-
	Lessons Learned Register	Project Documents	8.3 Control Quality	-	-
	-	Quality Control Measurements	-	-	8.3 Control Quality
	-	Work Performance Reports	9.5 Manage Team	-	-
	Quality Management Plan	Project Management Plan Updates	-	-	11.5 Plan Risk Responses
Monitoring of Cost	-	-	-	-	-
Resources Leveling	-	-	-	-	-

Table 15 (Continued). Mapping of Best Practices Vs Processes (PMBOK).

PM Activities (OpenProject)	Intermediary 2	Intermediary 1	Input	Tools & Technique	Outputs
Progress Report	-	Work Performance Reports	-	-	4.5 Monitor and Control Project Work
	-	Change Request	-	-	6.6 Control Schedule
	-	Work Performance Reports	10.2 Manage Communications	-	-
	-	Communication skills	-	10.2 Manage Communications	-
	-	Communication skills	-	13.3 Manage Stakeholder Engagement	-
Change Request	-	Project Management Plan	-	-	4.2 Develop Project Management Plan
	Change log	Project Documents	4.3 Direct and Manage Project Work	-	-
	-	Approved Change Requests	4.3 Direct and Manage Project Work	-	-
	-	Work Performance Data	-	-	4.3 Direct and Manage Project Work
	-	Change Requests	-	-	4.3 Direct and Manage Project Work
	-	Project Management Plan Updates	-	-	4.3 Direct and Manage Project Work
	-	Project Management Plan Updates	-	-	4.4 Manage Project Knowledge
	Cost Forecasts	Project Documents	4.5 Monitor and Control Project Work	-	-
	Schedule Forecasts	Project Documents	4.5 Monitor and Control Project Work	-	-
	-	Change Requests	-	-	4.5 Monitor and Control Project Work
	-	Project Management Plan Updates	-	-	4.5 Monitor and Control Project Work
	Risk Report	Project Documents	4.6 Perform Integrated Change Control	-	-
	-	Change Requests	4.6 Perform Integrated Change Control	-	-
	Documents Changes	Change control Tools	-	4.6 Perform Integrated Change Control	-
	Voiting	Decision Making	-	4.6 Perform Integrated Change Control	-
	-	Meetings	-	4.6 Perform Integrated Change Control	-
	-	Approved Change Requests	-	-	4.6 Perform Integrated Change Control
	Change log	Project Documents	4.7 Close Project or Phase	-	-
	-	Change Request	-	-	5.5 Validate Scope
	-	Work Performance Data	5.6 Control Scope	-	-
	-	Change Requests	-	-	5.6 Control Scope
	-	Project Management Plan Updates	-	-	5.6 Control Scope
	Performance Measurement Baseline	Project Management Plan Updates	-	-	5.6 Control Scope
	-	Change Requests	-	-	6.2 Define Activities
	-	Project Management Plan Updates	-	-	6.2 Define Activities
	-	Change Requests	-	-	6.5 Develop Schedule
	-	Project Management Plan Updates	-	-	6.5 Develop Schedule
	-	Change Requests	-	-	6.6 Control Schedule
	-	Control Requests	-	-	7.4 Control Costs
	-	Project Management Plan Updates	-	-	7.4 Control Costs
	Performance Measurement Baseline	Project Management Plan Updates	-	-	7.4 Control Costs
	-	Project Management Plan Updates	-	-	8.1 Plan Quality Management
	-	Change Requests	-	-	8.2 Manage Quality
	-	Project Management Plan Updates	-	-	8.2 Manage Quality
	-	Approved Change Requests	8.3 Control Quality	-	-
	Approved Change Request Review	Meetings	-	8.3 Control Quality	-
	-	Verfied Deliverables	-	-	8.3 Control Quality
	-	Change Requests	-	-	8.3 Control Quality
	-	Project Management Plan Updates	-	-	8.3 Control Quality
	-	Change Requests	-	-	9.3 Acquire Resources
	-	Project Management Plan Updates	-	-	9.3 Acquire Resources
	-	Change Requests	-	-	9.4 Develop Team
	-	Project Management Plan Updates	-	-	9.4 Develop Team
	-	Change Requests	-	-	9.5 Manage Team
	-	Project Management Plan Updates	-	-	9.5 Manage Team
	-	Change Requests	-	-	9.6 Control Resources
	-	Project Management Plan Updates	-	-	9.6 Control Resources
	-	Project Management Plan Updates	-	-	10.1 Plan Communications Management
	Change log	Project Documents	10.2 Manage Communications	-	-
	-	Project Management Plan Updates	-	-	10.2 Manage Communications
	-	Change Requests	-	-	10.3 Monitor Communications
	-	Project Management Plan Updates	-	-	10.3 Monitor Communications
	-	Procurement Documentation	11.2 Identify Risks	-	-
	-	Change Requests	-	-	11.5 Plan Risk Responses
	-	Project Management Plan Updates	-	-	11.5 Plan Risk Responses
	-	Change Requests	-	-	11.6 Implement Risk Responses
		Change Requests	-	-	11.7 Monitor Risks
	-	Project Management Plan Updates	-	-	11.7 Monitor Risks
	-	Change Requests	-	-	12.1 Plan Procurement Management
	-	Agreements	-	12.2 Conduct Procurements	-
	-	Change Requests	-	12.2 Conduct Procurements	-
	-	Project Management Plan Updates	-	12.2 Conduct Procurements	-
	-	Approved Change Requests	12.3 Control Procurements	-	-
	-	Procurement Documentation Updates	-	-	12.3 Control Procurements
	-	Change Requests	-	-	12.3 Control Procurements
	-	Project Management Plan Updates	-	-	12.3 Control Procurements
	-	Change Requests	-	-	13.1 Identify Stakholders
	-	Project Management Plan Updates	-	-	13.1 Identify Stakholders
	Change log	Project Documents	13.2 Plan Stakeholder Engagement	-	-
	Change log	Project Documents	13.3 Manege Stakeholder Engagement	-	-
	-	Change Requests	-	-	13.3 Manege Stakeholder Engagement
	-	Project Management Plan Updates	-	-	13.3 Manege Stakeholder Engagement
	Change log	Project Document Updates	-	-	13.3 Manege Stakeholder Engagement
	-	Change Requests	-	-	13.4 Monitor Stakeholder Engagement
	-	Project Management Plan Updates	-	-	13.4 Monitor Stakeholder Engagement
Control Charts: Gantt chart, S-curve, cause-and-effect diag	Control Charts	Data Representation	-	8.3 Control Quality	-
Monitoring Schedule	Control Thresholds	Schedule Management Plan	-	6.1 Plan Schedule Management	-
Earned Value Management	Rules of Performance Measurement	Schedule Management Plan	-	-	6.1 Plan Schedule Management
	-	Expert Judgement	-	7.1 Plan Cost Management	-
	Rules of Performance Measurement	Cost Management Plan	-	-	7.1 Plan Cost Management
	-	Cost Baseline	-	-	7.3 Determine Budget
Financial Measurement Tools	-	-	-	-	-
Re-baselining	-	-	-	-	-
Customer Satisfaction Surveys	-	-	-	-	-
Team Member Performance Evaluation	-	-	-	-	-
Team Learning Curve	-	-	-	-	-
Lessons Learned/Post-Mortems	Retrospectives	Meetings	-	8.3 Control Quality	-
Database for Lessons Learned	-	-	-	-	-
Database of Contractual Commitment Data	-	-	-	-	-
Database of Risks	-	-	-	-	-
Project Web Site	-	Communications Management Plan	-	-	10.1 Plan Communications Management
	-	Meetings	-	10.1 Plan Communications Management	-
Database of historical data	-	Organizational process Assets	11.5 Plan Risk Responses	-	-
Life Cycle Costs (LCC)	-	Decision Making	-	8.2 Manage Quality	-
Bid/seller evaluation	-	-	-	-	-
Bidders conferences	-	-	-	-	-
Bid documents	-	Bid Documents	-	-	12.1 Plan Procurement Management
Client acceptance form	-	-	-	-	-
Configuration review	-	-	-	-	-
Graphical presentation of risk information	-	-	-	-	-
Quality Function Deployment (QFD)	-	Interpersonal and Team Skills	5.2 Collect Requirements	-	-
Baseline plan / Project Baseline	-	-	-	-	-

5.3.2 Application of graph theory (Gephi)

1. Build the model (Gephi)

Based on Table 18, data was integrated into the Gephi tool to construct a node diagram that graphically represents the interactions between best practices and PMBOK (6th edition) processes. Labels were added to the nodes for classification:

- PP-PMBOK: PMBOK Sub-Processes (49)
- PG-PMBOK: PMBOK Process Groups (10)
- F-PM-AC: Single Intermediary Connector (48)
- GF-M-PM-AC: Complex Intermediary Connector (23)
- BP-XX: Best Practice (46)
- PCL-XX: Project life cycle (5)

The integration of the table data into Gephi resulted in 182 nodes and 463 edges. Two types of edges were identified: those related to PMBOK processes (Input, Output, Tool, and Technique) and those associated with connector types between functionalities and PMBOK processes (Level 1 and Level 2). Additionally, the edges need to specify whether they are directed or undirected. In this case, the edges that establish a link between functionalities and PMBOK processes are considered directed. Figure 62 shows an example of the node table, while Figure 63 presents an example of the edges integrated in Gephi.

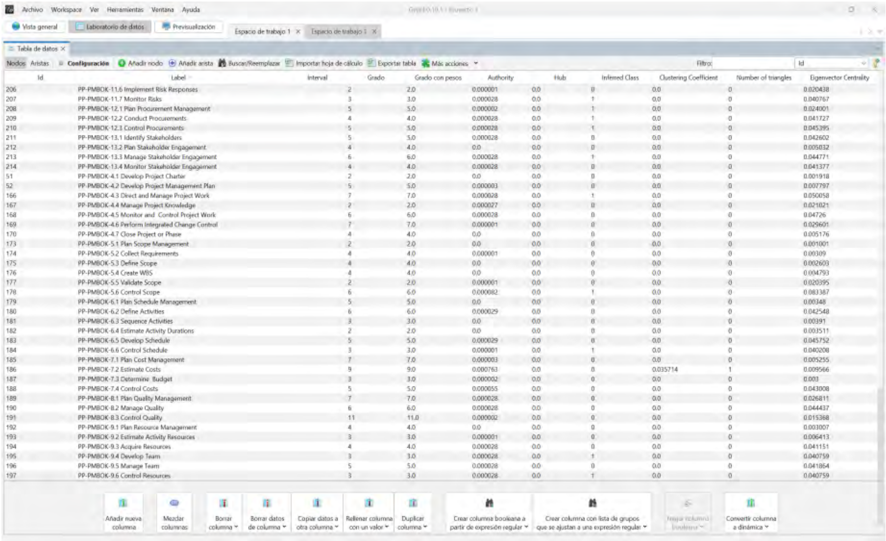


Figure 62. Nodes integrated in Gephi (Best Practices graph theory analysis)

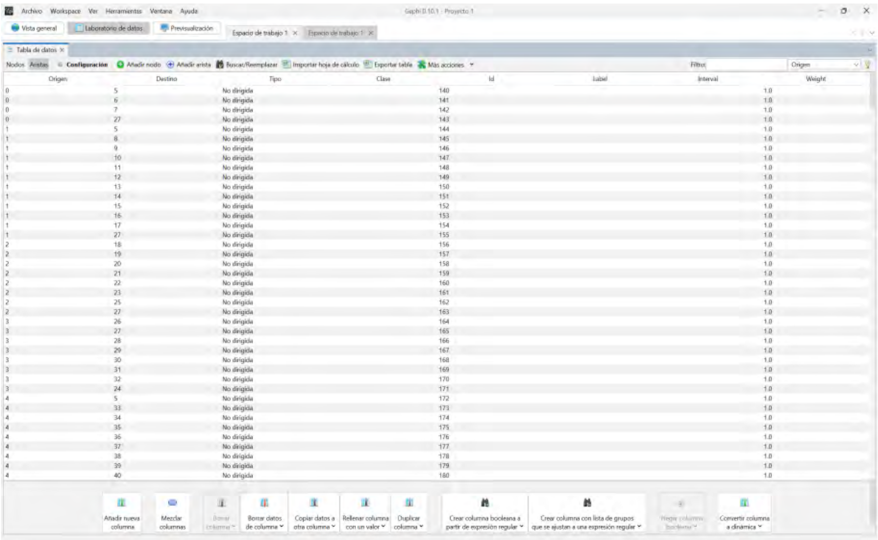


Figure 63. Edges integrated in Gephi (Best Practices graph theory analysis)

The nodes and edges diagram (Figure 64), visually presents the resulting node diagram (data described in Annex 2) from this analysis. This representation shows that PMBOK processes and sub-processes have a strong interaction with best practices described in the literature.

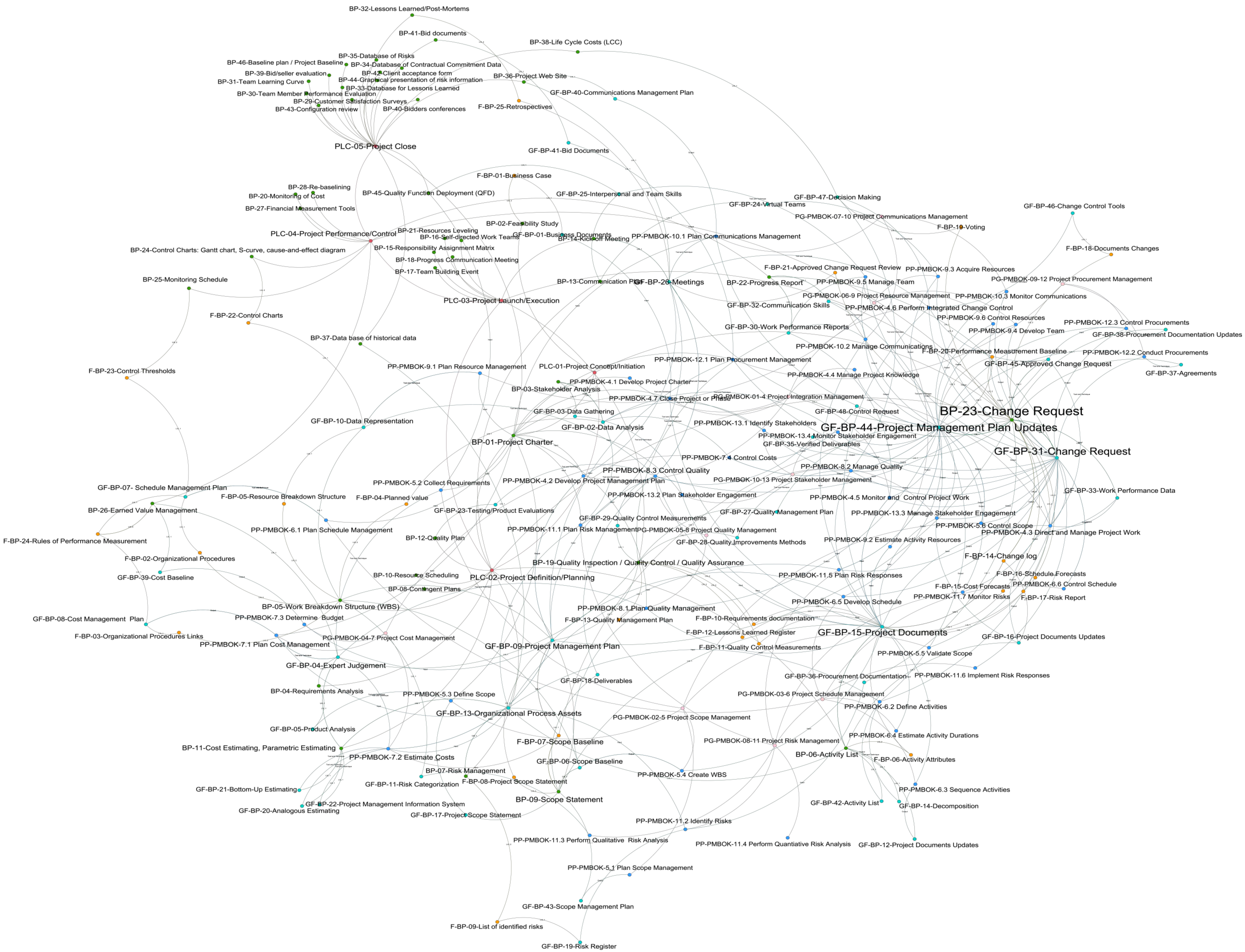


Figure 64. Node diagram (Best Practices vs PMBOK).

2. Apply the graph theory to the nodes

The graph's complexity is 5.008, indicating a high degree of complexity. This complexity is interpreted in the context of graph theory, where complex graphs start from a value of 1. Furthermore, the graph's complexity also suggests its robustness, being in this case a strong graph. This implies that even if a significant number of edges were lost, the graph would still maintain communication between the nodes

3. Analysis of results

The analysis of the impact of best practices on PMBOK processes and sub-processes using node and edge visualization software, such as Gephi, became difficult to interpret graphically due to the large number of connections between best practices and PMBOK processes and sub-processes. Therefore, Figure 65 was developed, which uses a bar chart to count how many best practices impact each PMBOK sub-process.

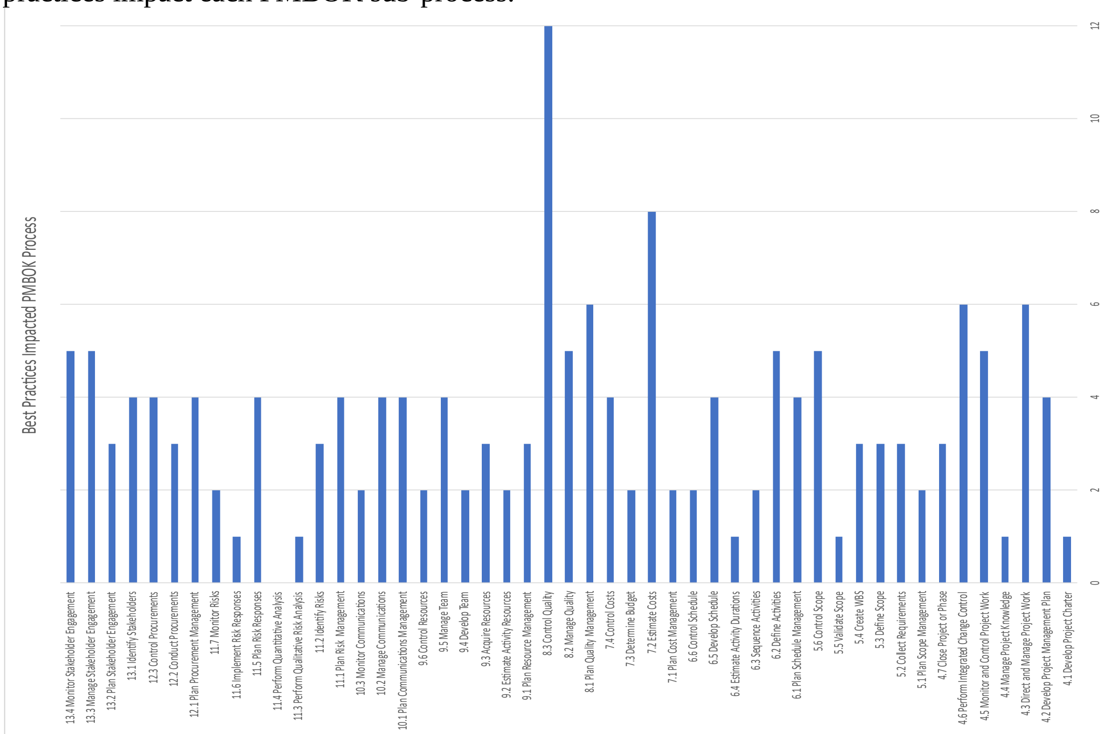


Figure 65. Mapping of impacted processes (PMBOK) by Best Practices

Considering Figure 65, only one of the sub-processes, "11.4 Perform Quantitative Risk Analysis," did not demonstrate an interaction with these practices. This observation suggests that, in general, all best practices aim to address project management processes and sub-processes, ensuring that each has at least one associated best practice.

This raises the following question: Would software that incorporates all the best practices described in the literature allow for a comprehensive evaluation of project management maturity?

To address this question, a proposal was developed outlining the functionalities that an ideal software should include to support the generation of traces for assessing project management maturity. This theoretical proposal, referred to as the software’s API, would enable the implementation and capture of different types of best practices, which we classify as "digital best practices" and "physical best practices."

5.4 Comparison of a feasibility study on maturity assessment using traceability in an activity management software tool applied to OpenProject and literature best practices.

Traditional project management maturity models employ best practices derived from references, expert surveys, or other maturity models to evaluate companies through surveys. These practices are applied to individuals interacting with the project management process, regardless of whether they belong directly to the department or area. The raw material for these surveys is the best practices, which, through their use, increase the success rate of projects and consequently enable organizations to achieve their organizational objectives.

The objective of this thesis work is to enhance the current process of project management maturity evaluation by utilizing computer traces from a software in the same domain. Therefore, a system capable of processing traces from an open-source software (OpenProject) was developed, allowing their transformation and visualization in terms of user actions. However, despite the functionality of the system, the most complex part of this thesis work is the evaluation of maturity based on the transformed traces. The software developed to transform the traces generates a CSV file with traces transformed into user actions. From this file, an analysis of the usage index of project management activities was conducted, determining their usage percentage to create a benchmark that facilitates maturity evaluation in this context.

Based on the results of this evaluation proposal, it was identified that only nine traceable functionalities can be performed in the system. Mapping project management activities from OpenProject to a reference such as PMBOK helps determine which project management processes we can perform and which we cannot. This provides us with information about the traces we can obtain and those we cannot, either because no functionality generates them or because, despite having the functionality, the system does not generate traces from its use. Table 19, summarizes this mapping.

Table 19. Mapping Summary between OpenProject Vs PMBOK

	PMBOK Mapping							Maturity Assessment		
Source	Analysis	Number of PMBOK Process Impacted	Mapping Method	Type of connections in the Graph	Nodes Number	Edges Numbe r	Medium Grade (Complexity)	Type of assessment	Resources	Complexity
Proposal	Open Project Vs PMBO K	19/49	Graph Theory	Intermediary Required	103	153	2.971	Weak	Economic-Low Human-Low Knowledge-Medium Time-Low/Mediu m	Low

The limited number of processes impacted by traceable project management activities in OpenProject indicates a constraint in conducting a comprehensive evaluation of all PMBOK processes. In contrast, when comparing this finding with the mapping of project management best practices using the same reference, it is evident that traditional maturity models employ best practices that impact virtually 99% of PMBOK processes. If traditional models impact all processes, the functionalities of software intended to assess maturity should also impact all or all PMBOK processes. Table 20 provides a summary of the mapping of best practices with the process reference, in this case, PMBOK.

Table 20. Best Practices Vs PMBOK Mapping Summary

		PMBOK Mapping						Maturity Assessment		
Source	Analysis	Number of PMBOK Process Impacted	Mapping Method	Type of connections in the Graph	Nodes Number	Edges Number	Medium Grade (Complexity)	Type of assessment	Resources	Complexity
Traditional Maturity Models	Best Practices of Project Management Vs PMBOK	48/49	Graph Theory	Intermediary Required	182	463	5.008	Complete	Economic-High Human-High Knowledge-High Time-High	High

While a lack of software functionalities enabling project management activities capable of impacting the PMBOK in the same way as best practices has been identified, no other open-source software facilitating trace collection focused on project management has been identified. For example, Project-Open software, which is open-source, does not offer data collection capabilities due to the software's structural limitations. Generally, project management software is more focused on being a tool to assist in task planning and tracking.

This focus becomes more apparent with OpenProject, as the traceable functionalities identified are more centered around task planning, team management, and communication. In contrast, when mapping the best practices that are the raw material of traditional maturity models, we observe that they are generally more focused on integration, scope, scheduling, costs, quality, resources, communication, risks, procurement, and stakeholders.

While these functional limitations are observed with the project management software used for this analysis, it is important to note that a total of 10 project management tools were explored as part of our state-of-the-art review. OpenProject was the only tool identified thus far capable of providing traces of usage and all the basic documentation that clarified its structure. As a conclusion of this section, the generation of artificial traces is proposed, in which functionalities that allow impact on all 49 PMBOK processes are considered. The goal of this is to validate if a software containing the necessary functionalities to impact all PMBOK processes would make maturity evaluation feasible.

5.5 Software Architecture Specification

After analyzing project management processes using the PMBOK theory and best practices with OpenProject, it was identified that despite the software's capability to provide user activity traces, there is a lack of functionalities to effectively perform best practices or impact a significant number of processes as best practices do. Therefore, in this section, we will specify the minimum number of functionalities a software should have to support project management activities and best practices to evaluate its maturity.

5.5.1 Functionalities Vs. Best Practices

The functionalities of the system are those that enable users within a given domain to perform activities. In this thesis, the objective of employing these functionalities is to characterize the activities that a user should be able to perform within the system. It is important to highlight that functionalities are relevant because if the system lacks specific ones, users will not be able to comply with certain best practices, thus hindering the evaluation process. For example, if a user must perform a project best practice called "risk analysis," which will be evaluated to determine project management maturity, and they are unable to do so within the system, the evaluation will be incomplete. It is through this example that we address the need to specify an architecture that allows most best practices to be performed, aiming to impact the most processes of the PMBOK, ensuring a more comprehensive maturity evaluation. qAnother indispensable requirement, in addition to having a complete list of functionalities, is that they must be traceable. Without traceability, even if there are many functionalities, they will not be considered in the maturity evaluation.

5.5.2 Types of Best Practices

In the literature, best practices are classified in various ways. In some cases, they are categorized by domain, such as in the case of the IBM2 maturity model (social domain, contract domain, results domain, and interface domain). However, for this research, a more appropriate classification in terms of functionalities is proposed. There are two types of best practices: those that are performed within software (digital) and those best practices that are performed physically by their nature. Below, we will describe each of them.

Digital Best Practices

This type of practice focuses on activities that can be carried out digitally for project management. In practical terms, they refer to those activities that require the use of digital tools (software) to be performed more efficiently. Digital best practices are activities that generate traces. Table 21 describes the best practices of this digital type:

Table 21. Best Practices (Digital)

Resource scheduling	Probabilistic estimation of duration (PERT)
Responsibility assignment matrix	Top-down / bottom-up estimating
Team building event	Network diagram, critical chain method and analysis
Progress communication meeting	Monte-Carlo analysis
Progress report	Milestone planning
Customer satisfaction surveys	Quality plan
Team member performance evaluation	Monitoring schedule
Requirements analysis	Monitoring of cost
Work Breakdown Structure (WBS)	Quality inspection
Project charter	Resources leveling
Feasibility study	Control charts
PBS: Product Breakdown Structure	Baseline plan
Activity List	Earned value management
Contingent plans	Quality Function Deployment (QFD)
Ranking of risks	Financial measurement tools
Configuration review	Database for cost estimating
Scope statement	Re-baselining
Change request	Stakeholder analysis
Customer acceptance form	Bid/seller evaluation
Graphical presentation of risk information	Bidders conferences
Database of contractual commitment data	Bid documents
Database of risks	Project Web Site
Task scheduling	Database of historical data
Cost/benefit analysis	Life Cycle Costs (LCC)
Critical path method analysis	Lessons learned/Post-mortems
Cost estimating, parametric estimating	Database for lessons learned

Physical Best Practices

This type of practice focuses on activities that cannot be carried out digitally, so they need to be performed in a physical way for project management. In practical terms, they refer to those activities that a digital tool (software) to be performed are not required. Physical best practices are activities that don't generate traces. Table 22 describes the best practices of this physical type:

Table 22. Best Practices (Physical)

Kick-off meeting
Self-directed work teams
Progress communication meeting
Team learning curve

5.5.3 Functionalities specification that impact all PMBOK processes

Considering how functionalities are regarded within OpenProject, we will stipulate that the minimum functionalities required for maturity assessment should be distributed in a microservices-focused architecture. This type of architecture enables the distribution of functionalities across small, isolated applications that communicate via a data bus. These small applications are referred to as APIs, which may group one or more system functionalities and generate traces. The OpenProject software encompasses various APIs, some geared towards system operations and others towards project management activities. In this specification, we will focus on proposing APIs centered on activities. When comparing the APIs proposed by OpenProject, we have a total of 45 APIs, whereas in our proposal, we consider a total of 129 APIs.

Proposed APIs

After conducting a comprehensive analysis of the nodes, which involved detailed mapping of the PMBOK processes and best practices, it has been concluded that best practices have a virtually global impact on all project management processes. Therefore, for the proposal of APIs, the 10 process groups defined in the 6th edition of the PMBOK have been considered. In this context, two types of APIs can be distinguished: those that are linked to documentary processes, i.e., generated to record the execution of documents related to the project, and the latter, which are associated with project management tools or techniques processes, aimed at recording the use of these latter.

A) Documental PM APIs
This type of APIS are those best practices that, by their nature, are carried out within a document. To maturity evaluation, a series of APIs are proposed to register project documents. Table 23 outlines the names of PMBOK processes that will be conceptualized as documentary APIs.

Table 23. Documental PM API

API	
Business documents	Quality Metrics
Agreements	Organization theory
Environmental Factors	Resource breakdown structure
Organizational Process assets	Risk Management Plan
Project Charter	Risk register
Assumption Log	Risk Report
Project Management Plan	Procurement management plan
Change Request	Procurement strategy
Enterprise environmental factors	Procurement statement of work

Deliverables	Source selection criteria
Work Performance data	Stakeholder register
Work Performance reports	Stakeholder engagement plan
Issue log	Ground rules
Knowledge Management	Work Performance Information
Information Management	Scope Management Plan
Lesson learned register	Requirements Management Plan
Final Product, service or result	Quality reports
Final report	Test and Evaluation documents
Project documents	Approved change requests
Requirements documentation	Quality Control Measurements
Requirements traceability Matrix	Verified deliverables
Project Scope statement	Resources requirements
Scope Baseline	Physical resources assignments
Schedule Management Plan	Project Team assignments
Activity List	Resource calendars
Activity Attributes	Resource Management Plan
Milestone List	Resource requirements
Project Schedule Network Diagram	Team performance assessment
Duration Estimates	Communications management plan
Basis of Estimates	Project Communications
Project Calendar	Bid documents
Cost Management Plan	Make-or-buy decisions
Project Documents updates	Independent cost stimation
Historical information review	Selected sellers
Project Management Plan Updates	Closed procurements
Quality Management Plan	

B) Tool or Technique PM API

These APIs are aimed at enabling system users to perform relevant project management techniques and/or utilize tools that can enhance project success rates. Of course, in all cases, these APIs will generate traces indicating their usage. Table 24 describes all APIS considered as a tool or technique.

Table 24. Tool or Technique PM API

API	
Expert Judgment	To-Complete performance index
Data Gathering	Cost Forecast
Interpersonal and team skills	Data Representation
Meetings	Test and Inspection
Change Control	Audits
Project Management Information System	Design for X
Data analysis	Problem Solving
Decision Making (Votation)	Quality Improvements Methods
Context Diagram	Testing / product evaluations
Prototype	Pre-Assignment
Inspection	Virtual Teams
Decomposition	Colocation
Rolling Wave Planning	Communication Technology
Precedence Diagramming Method	Recognition and rewards
Dependency Determination and Integration	Training
Leads and Lags	Individual and team assessments
Analogous Estimating	Communication requirements analysis

Parametric Estimating	Communication models
Tree-point Estimating	Communication methods
Bottom-up Estimating	Communication skills
Schedule Network Analysis	Project reporting
Critical Path Method	Prompt lists
Resource optimization	Risk categorization
Schedule compression	Representation of uncertainty
Agile Release Planning	Strategies for threats
Cost Estimates	Strategies for opportunities
Cost Aggregation	Contingent response strategies
Funding Limit Reconciliation	Strategies for overall project risk
Financing	Source selection analysis

In all cases, the APIs will interact with the server using the communication methods POST, PATCH, GET, and DELETE. In turn, the server will have a series of combinations that, in response to requests, will respond based on the type of request.

5.6 Chapter 5 Summary

This chapter of the thesis presents a detailed analysis of the feasibility of assessing project management maturity through the use of a software tool dedicated to activity management, with a particular focus on the tool's ability to generate traces that evidence adherence to best practices.

The structure of the analysis is divided into several key sections. In the beginning of the chapter, a thorough mapping of the functionalities of the OpenProject tool is conducted, applying graph theory to evaluate the impact of these functionalities on the implementation of best practices. OpenProject is selected as an open-source tool due to its diverse capabilities for managing projects, and a review of its documentation is carried out along with the identification of its architecture. This process enables the establishment of connections between its modules and the best practices defined in the PMBOK.

Next, deficiencies in the functionalities of OpenProject are identified, and a comparison is made with a set of 46 project management best practices. Using a graph theory approach, the interaction between best practices and the 49 PMBOK processes is visualized, revealing that while the majority of best practices impact the processes, there are significant areas that require improvement. Finally, a specification for a software architecture based on microservices is proposed, suggesting a total of 129 APIs that would allow for proper traceability and impact on the PMBOK processes. The proposal emphasizes the importance of having both digital and physical functionalities to facilitate an effective assessment of maturity in project management. Together, this chapter establishes a conceptual and practical framework for maturity evaluation, highlighting the need to integrate functionalities that support best practices and generate useful traces for further analysis.

5.7 Chapter 5 Discussion

The discussion concerns fourth issues: Lack of understanding of functionalities by users, misalignment between functionalities and user needs, insufficient evaluation of tools' ability to implement best practices, and identification of functional gaps.

The first issue pertains to the lack of comprehension of the functionalities by users: Frequently, tools present users with a set of predefined functionalities, yet on numerous occasions, these functionalities are not readily comprehensible or adequately elucidated. This lack of understanding may be attributed to inadequate documentation, intricate interfaces, or an absence of comprehensive user training. Consequently, users may only employ a subset of the tool's capabilities, concentrating on well-known features while neglecting others that could markedly enhance their workflow. This issue underscores the significance of user education and intuitive tool design to guarantee that functionalities are accessible and comprehensible, thereby optimizing the tool's efficiency.

The second issue pertains to the discrepancy between functionalities and user needs: In a number of instances, there is a discernible discrepancy between the functionalities offered by the tool and the actual requirements of the users. This discrepancy frequently arises because tools are designed with a focus on general use cases, rather than being tailored to specific contexts or industries. Consequently, users may encounter difficulties in employing the tool effectively in their distinctive workflows or may be compelled to devise compensatory measures to address the absence of requisite features. To address this issue, it is essential to conduct comprehensive needs assessments prior to tool development or selection. This ensures that the functionalities offered are closely aligned with the real-world requirements of the users.

The third issue pertains to the insufficient project management tools' ability to implement best practices: The capacity of a project management tool should not be based solely on the number of features it offers, as this can be misleading. The presence of numerous functionalities in a tool does not necessarily indicate its suitability for implementing best practices within a specific domain. A more comprehensive evaluation entails an examination of how these functionalities either facilitate or impede the application of established best practices. This necessitates a detailed analysis of the relevance, usability, and efficacy of each functionality in achieving the desired outcome.

The fourth issue pertains to identification of functional gaps: The absence of certain critical functionalities in tools that are necessary for the full execution of best practices can impede the overall success of projects. The identification of these gaps necessitates a comprehensive comparison between the tool's capabilities and the specific requirements of the domain. This analysis not only elucidates which best practices cannot be fully implemented but also provides insight into the potential impact on project outcomes. Furthermore, this issue presents the opportunity for customization or enhancement of the tool, either through additional development or by seeking alternative solutions that better meet user needs. Addressing these functional gaps is essential for ensuring that tools evolve in response to the dynamic needs of their users.

Finally, in the next section, the general conclusions and perspectives around this thesis work are described, considering a general thesis summary, conclusions per chapter and perspectives on maturity models and their evaluations.

General conclusions and perspectives

This thesis puts forth a novel approach to the assessment of project management maturity. The proposed method involves the utilization of interaction traces derived from specialized software instead of the conventional surveys employed by traditional project management maturity models. This approach aims to address some of the significant challenges organizations face when implementing current maturity models, including the need for specialized human resources, financial limitations, and a lack of technical knowledge. Traditional project management maturity models, which rely on surveys, can be costly and time-consuming, and they often depend heavily on the subjectivity of the respondents. In contrast, using software interaction traces offers a more objective and efficient alternative, providing precise and real-time data on how projects are managed. This proposal has the potential to facilitate a more agile and accurate implementation of maturity models, reducing the resource burden and improving the accuracy of evaluations. Furthermore, by integrating this technology into existing project management systems, organizations can obtain a continuous and dynamic view of their maturity level, allowing for more proactive and effective adjustments and improvements.

6.1 Thesis Summary

The activities performed in this doctoral thesis focus on the assessment of project management maturity through the analysis of traces or logs of computer activity of project management software users. First, key concepts related to maturity models, project management, and IT traces were analyzed. Then, a state-of-the-art review was conducted to position the topic within the existing literature. This review revealed three main challenges: the lack of financial and human resources, and the lack of knowledge required for implementation. With these challenges identified, three proposals were formulated to address them.

The first proposal focuses on addressing the lack of knowledge to implement project management maturity models. A conceptual model was developed that identifies the basic knowledge domains and their relationships necessary to implement these models in organizations, applying it as a practical case in project management. This proposal was validated using the NIAM tool, which allows the generation of a conceptual model describing concepts and relationships, in addition to the generation of relational statements to determine the consistency of the model and possible cardinality errors.

The second proposal seeks to address the number of financial resources needed to implement project management maturity models. It proposes the use of computer traces of project management software as an element of maturity assessment. This approach requires a methodology to identify trace management steps and a detailed transformation process to reduce the level of granularity, facilitating the identification of project management domain activities. The validation of this proposal included the development of a tool capable of collecting, transforming, visualizing and assessing project management maturity, using best practices within the system as an evaluation index.

The third proposal arose from the identification of a limitation in the number of functionalities of the selected software, OpenProject, with respect to best practices. An exhaustive analysis was performed using graph theory to map OpenProject functionalities to PMBOK processes, concluding that the current functionalities of the software do not allow a complete assessment of project management maturity. Therefore, an adapted architecture specification for project management maturity assessment was proposed. Two types of APIs are described in this proposal: Documentation APIs and Project

Management Tools or Techniques. These APIs allow the generation of traceable functionalities adapted for maturity assessment.

6.2 Summary of contributions

This doctoral thesis is structured in five chapters of which each one contributes to different scientific domains linked to project management maturity models, project management, and trace transformation. While chapters 3,4,5 are dedicated to the main contributions of this doctoral thesis, chapters 1 and 2 are considered as general contributions to the thesis topic:

Chapter 1: This first chapter of the thesis, the key concepts that form the theoretical framework for understanding this research are described, focusing on maturity models, project management, and trace-based systems. Together, these elements aim to optimize organizational project management through a systematic, data-driven approach. Precision in evaluation, detailed analysis, and best practices enable continuous improvement and efficiency. The thesis selects OpenProject, an open-source software, for its robust project management functionality and detailed trace generation, facilitating pattern identification and performance evaluation. This chapter establishes the foundation for how these concepts and tools can effectively optimize organizational processes and improve project management and overall performance.

Chapter 2: This chapter reviews the literature on maturity models, project management, and the use of computer traces, comparing traditional models that assess maturity through surveys and identifying two main issues: the lack of knowledge and resources for evaluating project management maturity. The absence of conceptual guides leads to ineffective maturity assessments, while resource constraints, especially in SMEs, hinder continuous improvement efforts. The comparison of 12 maturity models shows no existing work on using software traces for maturity assessment. The chapter highlights the need for a methodology to transform and visualize traces, noting that current applications of software traces are primarily in cybersecurity, knowledge valorization, and process improvement. The proposal to use software traces for project management maturity evaluation is innovative and unaddressed in the existing literature.

Chapter 3: This chapter, the P2Ms are highlighted as valuable tools for evaluating organizational performance and departmental effectiveness, yet their implementation is challenging due to the lack of explicit guidance on how to apply them and on best practices. This absence hampers organizational capabilities and model implementation. Beyond understanding basic concepts, conceptual knowledge involves recognizing interactions for maturity assessments and identifying essential elements within specific scopes. Two scope groups are proposed: traditional scopes (organizational, domain, maturity assessment, and referential) and evidence-based scopes (trace-based evaluation, data processing, knowledge, and domain tools). These scopes were used to implement the CP2Ms framework for project management, verifying its effectiveness as a guide for identifying interacting elements within a domain. Using the NIAM method, the model was found complete with no matching errors.

Chapter 4: This chapter, the challenges in transforming traces for specific purposes, such as evaluating the maturity of project management, are addressed. A process is proposed to transform traces using an algorithm that reduces granularity. The complexity arises from the lack of methodologies or general processes, leaving those interested in trace transformation unaware of the necessary tools and knowledge. Although some proposals describe a process for trace exploitation, such as TBS's process of collection, transformation, and visualization, they do not do so explicitly. Additionally, tools generating traces must provide detailed documentation on trace generation, often available only from open-source software providers. Another challenge is system functionalities; if they do not generate traces, identifying user actions becomes difficult. This chapter proposes a process that involves understanding, structuring, and transforming traces to address these complexities.

Chapter 5: This chapter presents the feasibility study proposal, which aims to assess the adaptability of software systems to generate traces and evaluate the maturity level within a specific domain. The feasibility analysis focuses on determining whether an activity management software can produce traceable functionalities to measure compliance with best practices and assess system maturity. The analysis follows two key stages: first, mapping the software's functionalities against recognized best practices; and second, applying graph theory (using Gephi) to evaluate and quantify the impact of these functionalities in achieving those practices. To validate the proposed approach, the project management software *OpenProject* was analyzed. The findings revealed that *OpenProject* does not generate traces for all the necessary functionalities required to evaluate maturity based on the best practices identified in the literature. Two mappings were conducted: one to determine whether best practices impact the majority of PMBOK processes, and another to assess whether *OpenProject* functionalities achieve the same. Through the application of graph theory, it was found that best practices influence nearly 99% of PMBOK processes, whereas *OpenProject* functionalities fall short in this regard. As a result, evaluating project management maturity based on software-generated traces is only feasible if a sufficient number of traceable functionalities exist that impact the majority of PMBOK processes. The minimum required functionalities were identified and classified into two categories: 71 documentary functionalities and 59 tool- or technique-related functionalities, both of which support project management activities.

6.3 Research Perspectives

This section outlines the perspectives on the topic of this doctoral thesis. Despite the effort to assess project management maturity using traces, there are some limitations to its use. However, it is also considered that this research opens the door to new perspectives on evaluating maturity models.

6.3.1 Traditional PM2 Vs. Trace-based PM2

Traditional maturity models are generally implemented by large organizations where limitations in human, financial, and knowledge resources are not as constrained. However, abundant access to resources does not negate the considerable effort required for the process. Conversely, the context for small and medium-sized enterprises (SMEs) is more complex, as they face significant constraints in terms of resources. This compels them to focus more on survival rather than organizational improvement. Nonetheless, there are exceptions, especially for SMEs located in countries that develop support programs for enhancing such enterprises, providing explicit resources for their development.

Though some of the challenges faced by different types of organizations in implementing maturity models have been described, the proposal of trace-based maturity models opens the door to a world where data has begun to become an integral part of organizations' daily operations. Data-driven decisions, as the name suggests, involve decision-making based on data, making a maturity model based on data crucial for making more informed decisions by eliminating the subjectivity of those implementing them.

The advantages of this new type of trace-based model are numerous, with the most prominent being the reduction of financial, human, and knowledge resources required for its implementation. Such a model can help organizations streamline their processes and improve efficiency, which is especially beneficial for SMEs operating with limited resources. Thus, data-based maturity models represent a powerful tool for organizational improvement in an increasingly competitive and data-driven environment. Figure 66, describes the main resources consumed by the different types of project management maturity models.

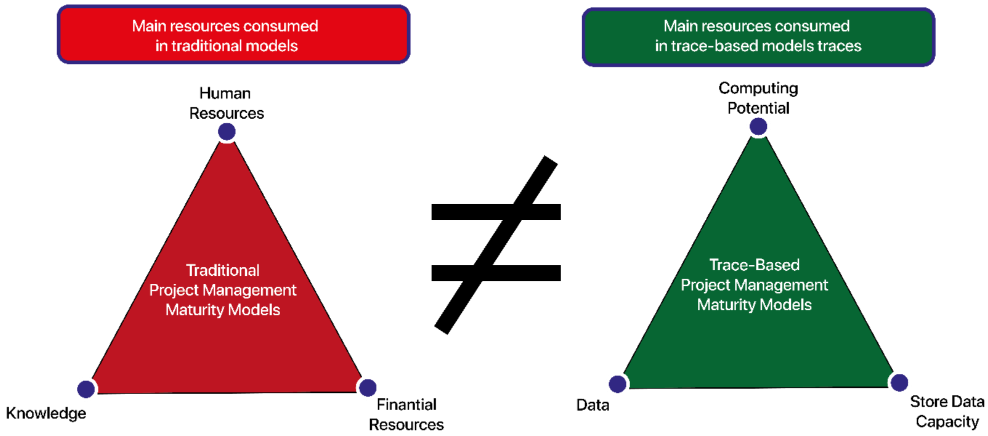


Figure 66. Traditional Vs. Trace-based

6.3.2 Automatic recommender system for trace-based maturity models

It is evident that when a domain starts to be based on data, eventually there are activities that can be automated, allowing for the implementation of artificial intelligence tools to perform tasks that would normally be carried out by humans. This doctoral work presents a first version of software programmed in Python to evaluate maturity in project management, using natural language processing techniques, such as regular expressions, to generate transformation rules.

In this context, the potential for developing a recommendation system is recognized. Such a system, by automatically detecting the level of maturity in project management based on traces, could send information to users to implement best practices in their activities. Traditionally, recommendation systems are used as assistants within software to help users with their tasks. In the educational domain, some authors refer to them as intelligent tutoring systems. The proposal in this section is to create a best practices assistant that guides users based on the guidelines of the PMBOK in project management activities.

6.3.3 Automated extraction of activity traces to extract competencies to manage the company's human capital

This approach is CaTCaP (Capture des Traces opérationnelles des acteurs d'entreprise pour construire le Capital humain et définir les Processus gagnants), which has been proposed as a way to extract competencies (both individual and collective) and, from them, to manage the company's human capital, identify successful organizational practices, and foster continuous learning. These methodologies are grounded in the "knowledge economy," where the objective is to capitalize on the implicit and explicit knowledge that underpins daily operations and consolidate it as a strategic asset for the organization. In these alternative approaches, trace-derived information not only serves to assess process maturity, but also to:

- Locate and formalize key competencies within the company
- Represent these competencies in dynamic maps or cartographies to facilitate decision making
- Analyze success factors or "winning practices" and their impact on project results.
- Identify opportunities for learning and continuous improvement related to the daily activity of teams.

Although the emphasis in this research has been on maturity assessment, it is important to recognize that trace collection and modeling opens the door to multiple applications that transcend the scope of project management. Consequently, leveraging traces to strengthen human capital and guide the evolution of competencies can be regarded as a complementary perspective or future direction for organizations seeking comprehensive process management and the knowledge and skills of their members.

6.4 References

[1] P. R. Alves and A. P. Tereso, "Project management system implementation in SMEs: a case study," in *33rd International Business Information Management Association Conference IBIMA*, Granada, Spain, Apr. 2019, pp. 8322–8332. Accessed: Jan. 24, 2025. [Online]. Available: https://core.ac.uk/outputs/225349467/?utm_source=pdf&utm_medium=banner&utm_campaign=pdf-decoration-v1

[2] S. Marcelino-Sádaba, A. Pérez-Ezcurdia, A. M. Echeverría Lazcano, and P. Villanueva, "Project risk management methodology for small firms," *International Journal of Project Management*, vol. 32, no. 2, pp. 327–340, Feb. 2014, doi: 10.1016/j.ijproman.2013.05.009.

[3] R. Golini, M. Kalchschmidt, and P. Landoni, "Adoption of project management practices: The impact on international development projects of non-governmental organizations," *International Journal of Project Management*, vol. 33, no. 3, pp. 650–663, Apr. 2015, doi: 10.1016/j.ijproman.2014.09.006.

[4] G. Fernandes, S. Ward, and M. Araújo, "Improving and embedding project management practice in organisations — A qualitative study," *International Journal of Project Management*, vol. 33, no. 5, pp. 1052–1067, Jul. 2015, doi: 10.1016/j.ijproman.2015.01.012.

[5] I. R. Luna, "Las pyme en la economía global. Hacia una estrategia de fomento empresarial," *Probl Desarro*, vol. 37, no. 146, pp. 31–50, 2006, [Online]. Available: http://www.jstor.org/stable/43838426

[6] J. Garzás, C. M. Fernández, and M. Piattini, "Una aplicación de la norma ISO/IEC 15504 para la evaluación por niveles de madurez de Pymes y pequeños equipos de desarrollo," *Revista Española de Innovación, Calidad e Ingeniería del Software*, vol. 5, no. 2, pp. 88–98, Jul. 2009.

[7] Á. Cuadros López, J. A. Morales Viveros, and A. B. Rojas Melendez, "Propuesta metodológica para medir el nivel de madurez de la gestión de proyectos en empresas de ingeniería," *Revista EIA*, vol. 14, no. 27, Sep. 2017, doi: 10.24050/reia.v14i27.808.

[8] T.-C. Lin and K. J. Wang, "Project-based maturity assessment model for smart transformation in Taiwanese enterprises," *PLoS One*, vol. 16, no. 7, Jul. 2021, doi: 10.1371/journal.pone.0254522.

[9] Ó. Rubiano Ovalle and A. Cuadros Lopez, "Estado De La Gerencia De Proyectos En Pymes Del Sector Artes Gráficas Y Propuestas De Lineamientos Para Su Mejoramiento (State of Project Management in SMEs of the Printing Sector and Proposed Guidelines for Improvement)," 2012.

[10] S. Sane, "Effect of using project management tools and techniques on SMEs performance in developing country context," *International Journal of Managing Projects in Business*, vol. 13, no. 3, pp. 453–466, May 2020, doi: 10.1108/IJMPB-11-2018-0251.

[11] F. Backlund, D. Chronéer, and E. Sundqvist, "Project Management Maturity Models – A Critical Review," *Procedia Soc Behav Sci*, vol. 119, pp. 837–846, Mar. 2014, doi: 10.1016/j.sbspro.2014.03.094.

[12] M. C. Paulk, B. Curtis, M. B. Chrissis, and C. V. Weber, "Capability maturity model, version 1.1," *IEEE Softw*, vol. 10, no. 4, pp. 18–27, Jul. 1993, doi: 10.1109/52.219617.

[13] A. Keller and J. Moormann, "The challenge to determine a company's process maturity: a case study from the financial services industry," *European Journal of Management Issues*, vol. 25, no. 2, pp. 85–91, Jun. 2017, doi: 10.15421/191712.

[14] J. Guangshe, C. Li, C. Jianguo, Z. Shuisen, and W. Jin, "Application of Organizational Project Management Maturity Model (OPM3) to Construction in China: An Empirical Study," in *2008 International Conference on Information Management, Innovation Management and Industrial Engineering*, IEEE, Dec. 2008, pp. 56–62. doi: 10.1109/ICIII.2008.182.

[15] C. P. Team, “CMMI for Development, version 1.3,” *Massachusetts: Software Engineering Institute*, 2010.
[16] A. Miklosik, “Improving Project Management Performance through Capability Maturity Measurement,” *Procedia Economics and Finance*, vol. 30, pp. 522–530, 2015, doi: 10.1016/S2212-5671(15)01264-2.
[17] W. S. Humphrey, “Characterizing the software process: a maturity framework,” *IEEE Softw*, vol. 5, no. 2, pp. 73–79, Mar. 1988, doi: 10.1109/52.2014.
[18] C. Vergopia, “Project review maturity and project performance: an empirical case study,” University of Central Florida , 2008. Accessed: Jan. 26, 2024. [Online]. Available: https://stars.library.ucf.edu/etd/3653/
[19] T. Jokela, M. Siponen, N. Hirasawa, and J. Earthy, “A survey of usability capability maturity models: implications for practice and research,” *Behaviour & Information Technology*, vol. 25, no. 3, pp. 263–282, May 2006, doi: 10.1080/01449290500168079.
[20] I. Project Management Institute, “Appendix X5 Research and Development for The Standard for Project Management,” in *Guide to the project management body of knowledge (PMBOK® guide) | PMBOK® guide*, 7th ed., Pennsylvania: Project Management Institute, 2021, ch. X5, pp. 245–245.
[21] Inc. Project Management Institute, “Coneceptual framework of project management,” in *Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK®) | Guia del PMBOK* , 3rd ed., Pennsylvania: Project Management Institute, 2004, ch. 1, pp. 3–4.
[22] D. Cram, B. Fuchs, Y. Prié, and A. Mille, “An approach to User-Centric Context-Aware Assistance based on Interaction Traces,” in *MRC 2008, Modeling and Reasoning in Context*, Delft, Netherlands, Jun. 2008. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01584075
[23] P.-L. Barazzutti, A. Cordier, and B. Fuchs, “Transmute : un outil interactif pour assister l’extraction de connaissances à partir de traces,” in *Extraction et Gestion des Connaissances - EGC 2016*, B. Crémilleux and C. de Runz, Eds., in Extraction et Gestion des Connaissances, vol. RNTI-E-30. Reims, France: RNTI, Jan. 2016, pp. 463–468. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01263101
[24] M. L. Roldán, S. Gonnet, and H. Leone, “TracED: A tool for capturing and tracing engineering design processes,” *Advances in Engineering Software*, vol. 41, no. 9, pp. 1087–1109, Sep. 2010, doi: 10.1016/J.ADVENGSOFT.2010.06.006.
[25] B. Fuchs, “Assister l’utilisateur à expliciter un modèle de trace avec l’analyse de concepts formels.,” in *28es Journées francophones d’Ingénierie des Connaissances IC 2017*, C. Roussey, Ed., in Actes IC 2017 28es Journées francophones d’Ingénierie des Connaissances. Caen, France, Jul. 2017, pp. 151–162. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01570274
[26] R. Zarka, A. Cordier, E. Egyed-Zsigmond, L. Lamontagne, and A. Mille, “Trace-based contextual recommendations,” *Expert Syst Appl*, vol. 64, pp. 194–207, Dec. 2016, doi: 10.1016/j.eswa.2016.07.035.
[27] T. Djouad and A. Mille, “Trace Based System in TEL Systems: Theory and Practice,” *IFIP Adv Inf Commun Technol*, vol. 522, pp. 257–266, 2018, doi: 10.1007/978-3-319-89743-1_23/FIGURES/4.
[28] P.-A. Champin, A. Mille, and Y. Prié, “Vers des traces numériques comme objets informatiques de premier niveau : une approche par les traces modélisées,” *Intellectica. Revue de l’Association pour la Recherche Cognitive*, vol. 59, no. 1, pp. 171–204, 2013, doi: 10.3406/intel.2013.1090.
[29] G. Prudhomme, K. Lund, and J.-L. Cassier, “Benefits of a research methodology from organisational sciences for analysing design interactions,” in *IDMME 2010*, Bordeaux, France: Springer, Oct. 2010, p. pp14. [Online]. Available: https://hal.archives-ouvertes.fr/hal-00580357
[30] B. Fuchs, “Focaliser l’extraction d’épisodes séquentiels à partir de traces par le contexte,” in *29es Journées Francophones d’Ingénierie des Connaissances, IC 2018*, S. Ranwez, Ed., in 29es Journées Francophones d’Ingénierie des Connaissances, IC 2018.

Nancy, France, Jul. 2018, pp. 213–227. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01839622
[31] C. Courtin and S. Talbot, "Automatic Analysis Assistant for Studies of Computer-Supported Human Interactions," 2009, pp. 572–583. doi: 10.1007/978-3-642-04636-0_52.
[32] F. Sanchez, "Supporting the transformation of a company's project management by elaborating an invariant-based project management maturity model and a causal predictive model between maturity criteria and project performance," P.H.D Dissertation, Lorraine Universite, Nancy, 2019.
[33] Hind Darwich Akoum, "Approche organisationnelle basée sur le paradigme agent pour la synthèse et la réutilisation des connaissances en ingénierie collaborative," Univiversité de Lorraine, Nancy, 2014.
[34] J. Girodon, "Proposition d'une approche d'amélioration des performances des organisations par le management opérationnel de leurs connaissances et compétences," Université de Lorraine, Nancy, 2015. Accessed: Feb. 28, 2024. [Online]. Available: https://hal.science/tel-01752010
[35] D. G. Manriquez, "Évaluation de la capacité à innover : une approche par auto évaluation et suivi supporté par des analyses multicritères dynamiques," Université de Lorraine, Nancy, 2015. Accessed: Feb. 28, 2024. [Online]. Available: http://www.theses.fr/2015LORR0018
[36] B. Dethine, "Vers une méthode d'aide à la décision pour définir la stratégie numérique des PMEs industrielles," Université de Lorraine, Nancy, 2023. Accessed: Feb. 28, 2024. [Online]. Available: http://www.theses.fr/2023LORR0003
[37] D. A. Doss *et al.*, "The Capability Maturity Model Integrated as a Market Engineering Maturity Model," *International Journal of Service Science, Management, Engineering, and Technology*, vol. 12, no. 3, pp. 175–196, May 2021, doi: 10.4018/IJSSMET.2021050110.
[38] C. V Weber, B. Curtis, and T. Gardiner, "Business process maturity model," in *Object Management Group Inc, USA, June*, 2008.
[39] J. Lee, D. Lee, and S. Kang, "vPMM: A Value Based Process Maturity Model," in *Computer and Information Science 2009*, R. Lee, G. Hu, and H. Miao, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 193–202. doi: 10.1007/978-3-642-01209-9_18.
[40] M. R. Wademan, C. M. Spuches, and P. L. Doughty, "The People Capability Maturity Model," *Performance Improvement Quarterly*, vol. 20, no. 1, pp. 97–123, Oct. 2008, doi: 10.1111/j.1937-8327.2007.tb00434.x.
[41] U. Nabitz, N. Klazinga, and J. Walburg, "The EFQM excellence model: European and Dutch experiences with the EFQM approach in health care," *International Journal for Quality in Health Care*, vol. 12, no. 3, pp. 191–202, Jun. 2000, doi: 10.1093/intqhc/12.3.191.
[42] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," 2007.
[43] A. González, A. Sandoval, and B. Heredia, "Elaboración de planes de entrenamiento cruzado a personal clave en proyectos: el caso de una empresa de ingeniería," *Revista ingeniería de construcción*, vol. 33, no. 3, Dec. 2018, doi: 10.4067/S0718-50732018000300205.
[44] E. Pérez-Mergarejo, I. Pérez-Vergara, and Y. Rodríguez-Ruíz, "Modelos de madurez y su idoneidad para aplicar en pequeñas y medianas empresas," *Ingeniería Industrial*, vol. XXXV, no. 2, pp. 146–158, 2014, [Online]. Available: https://www.redalyc.org/articulo.oa?id=360433597004
[45] F. F. Espinosa and G. E. Salinas, "Evaluación de la Madurez de la Función Mantenimiento para Implementar Innovaciones en su Gestión," *Información tecnológica*, vol. 21, no. 3, 2010, doi: 10.4067/S0718-07642010000300002.
[46] M. V. Jacquez-Hernández and V. G. López Torre, "Modelos de evaluación de la madurez y preparación hacia la Industria 4.0: una revisión de literatura," *Ingeniería Industrial.*

Actualidad y Nuevas Tendencias, vol. VI, no. 20, pp. 61–78, 2018, [Online]. Available: https://www.redalyc.org/articulo.oa?id=215057003004
[47] Prado Palacio and Leidy Johanna, "Grado de madurez en gestión de proyectos de una empresa constructora de vivienda en Santiago de Cali: caso de Buenavista constructora y promotora S.A.S," *Revista Espacios*, vol. 40, no. 43, pp. 20–33, 2019, [Online]. Available: https://bibliotecadigital.univalle.edu.co/handle/10893/15226
[48] J. Shan and R. Lu, "Research on Application of Project Management Maturity Model in Risk Management of Electrical Engineering," *E3S Web of Conferences*, vol. 218, Dec. 2020, doi: 10.1051/e3sconf/202021802040.
[49] D. Yao, R. Zhou, and T. Fan, "Research on maturity of engineering project informatization management—As an example of Anhui Sanjian Engineering Co.,LTD.," *IOP Conf Ser Earth Environ Sci*, vol. 267, p. 042144, 2019, doi: 10.1088/1755-1315/267/4/042144.
[50] R. Morlhon, R. Pellerin, and M. Bourgault, "Building Information Modeling Implementation through Maturity Evaluation and Critical Success Factors Management," *Procedia Technology*, vol. 16, 2014, doi: 10.1016/j.protcy.2014.10.127.
[51] F. Sanchez, E. Bonjour, J.-P. Micaelli, and D. Monticolo, "An Approach Based on Bayesian Network for Improving Project Management Maturity: An Application to Reduce Cost Overrun Risks in Engineering Projects," *Comput Ind*, vol. 119, Aug. 2020, doi: 10.1016/j.compind.2020.103227.
[52] H. Tahri and O. Drissi-Kaitouni, "New Design for Calculating Project Management Maturity (PMM)," *Procedia Soc Behav Sci*, vol. 181, pp. 171–177, May 2015, doi: 10.1016/j.sbspro.2015.04.878.
[53] A. J. Christoph and S. Konrad, "Project Complexity as an Influence Factor on the Balance of Costs and Benefits in Project Management Maturity Modeling," *Procedia Soc Behav Sci*, vol. 119, pp. 162–171, Mar. 2014, doi: 10.1016/j.sbspro.2014.03.020.
[54] Y. H. Kwak and C. W. Ibbs, "Project Management Process Maturity (PM)2 Model," *Journal of Management in Engineering*, vol. 18, no. 3, pp. 150–155, Jul. 2002, doi: 10.1061/(ASCE)0742-597X(2002)18:3(150).
[55] A.-L. Mesquida and A. Mas, "A project management improvement program according to ISO/IEC 29110 and PMBOK ®," *Journal of Software: Evolution and Process*, vol. 26, no. 9, pp. 846–854, Sep. 2014, doi: 10.1002/smr.1665.
[56] M. Gershon, "Choosing Which Process Improvement Methodology to Implement," *The Journal of Applied Business and Economics*, vol. 10, no. 5, p. 61, 2010.
[57] Robert K. Wysocki, *Project Management Process Improvement*. Norwood: ARTECH HOUSE, INC, 2004.
[58] H. K. Mittermaier and H. Steyn, "Project management maturity: An assessment of maturity for developing pilot plants," *South African Journal of Industrial Engineering*, vol. 20, no. 1, pp. 95–107, 2009.
[59] T. F. de Souza and C. F. S. Gomes, "Assessment of Maturity in Project Management: A Bibliometric Study of Main Models," *Procedia Comput Sci*, vol. 55, pp. 92–101, 2015, doi: 10.1016/j.procs.2015.07.012.
[60] J. C. Viana and C. M. de M. Mota, "Enhancing Organizational Project Management Maturity: a framework based on the value focused thinking model," *Production*, vol. 26, no. 2, pp. 313–329, Nov. 2015, doi: 10.1590/0103-6513.169913.
[61] S. Nikkhou, K. Taghizadeh, and S. Hajiyakhchali, "Designing a Portfolio Management Maturity Model (Elena)," *Procedia Soc Behav Sci*, vol. 226, pp. 318–325, Jul. 2016, doi: 10.1016/j.sbspro.2016.06.194.
[62] N. Williams, N. P. Ferdinand, and R. Croft, "Project management maturity in the age of big data," *International Journal of Managing Projects in Business*, vol. 7, no. 2, pp. 311–317, Apr. 2014, doi: 10.1108/IJMPB-01-2014-0001.
[63] F. Y. Hernández, R. I. Laguado, and J. P. Rodriguez, "Maturity analysis in project management in Colombian universities," *J Phys Conf Ser*, vol. 1126, p. 012055, Nov. 2018, doi: 10.1088/1742-6596/1126/1/012055.

[64] D. Seelhofer and C. O. Graf, "National project management maturity: A conceptual framework," *Central European Business Review*, vol. 7, no. 2, p. 1, 2018.
[65] L. De Souza Scotelano, R. Dalvo Pereira da Conceição, U. Da Costa Leonídio, and C. Silva de Jesus, "Project management maturity model: the case in an automotive industry in Brazil," *Brazilian Journal of Operations & Production Management*, vol. 14, no. 4, pp. 500–507, Dec. 2017, doi: 10.14488/BJOPM.2017.v14.n4.a6.
[66] K. Jana and T. Libena, "Proposal and Verification of Project Management Methods and Tools Oriented Maturity Model," *Journal of Management and Projects*, vol. 9, no. 1, 2018.
[67] R. Silva, N. Duarte, T. Barros, and G. Fernandes, "Project Management Maturity: Case study analysis using OPM3® model in manufacturing industry," in *2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, IEEE, Jun. 2019, pp. 1–8. doi: 10.1109/ICE.2019.8792586.
[68] B. Lathoud, "Formalization of the processing of electronic traces," *International Review of Law, Computers & Technology*, vol. 18, no. 2, pp. 185–192, Jul. 2004, doi: 10.1080/1360086042000223490.
[69] A. Mille, "From case-based reasoning to traces-based reasoning," *Annu Rev Control*, vol. 30, no. 2, pp. 223–232, Jan. 2006, doi: 10.1016/J.ARCONTROL.2006.09.003.
[70] R. Zarka, A. Cordier, F. Corvaisier, and A. Mille, "Providing assistance by reusing episodes stored in traces: a case study with SAP Business Objects Explorer," in *18ème Atelier Français de Raisonnement à Partir de Cas*, J. R. Florence Le Ber, Ed., Strasbourg, France, Jun. 2010, pp. 91–103. [Online]. Available: https://hal.science/hal-01381542
[71] O. L. Georgeon, A. Mille, T. Bellet, B. Mathern, and F. E. Ritter, "Supporting activity modelling from activity traces," *Expert Syst*, vol. 29, no. 3, pp. 261–275, Jul. 2012, doi: 10.1111/J.1468-0394.2011.00584.X.
[72] J. Laflaquière, L. S. Settouti, Y. Prié, and A. Mille, "Trace-based framework for experience management and engineering," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4251 LNAI-I, pp. 1171–1178, 2006, doi: 10.1007/11892960_141/COVER.
[73] R. Zarka, A. Cordier, E. Egyed-Zsigmond, and A. Mille, "Rule-Based Impact Propagation for Trace Replay," 2011, pp. 482–495. doi: 10.1007/978-3-642-23291-6_35.
[74] G. Khodabandelou *et al.*, "Cots Products To Trace Method Enactment: Review And Selection," in *21th European Conference on Information Systems*, Utrecht, Netherlands, Jun. 2013, p. [Online]. Available: https://hal-paris1.archives-ouvertes.fr/hal-00803873
[75] R. Zarka, "Trace-based reasoning for user assistance and recommendations," Université de Lyon, Lyon, 2013. Accessed: Feb. 29, 2024. [Online]. Available: https://theses.hal.science/tel-01077945
[76] K. Sehaba and S. M.-V. Metz, "Using Interaction Traces for Evolutionary Design Support - Application on the Virtual Campus VCIel," in *CSEDU 2011 - Proceedings of the 3rd International Conference on Computer Supported Education, Volume 2, Noordwijkerhout, Netherlands, 6-8 May, 2011*, A. Verbraeck, M. Helfert, J. Cordeiro, and B. Shishkov, Eds., SciTePress, 2011, pp. 237–242.
[77] S. Yahiaoui, C. Courtin, P. Maret, and L. Tabourot, "Competences Network Based on Interaction Data for Recommendation and Evaluation Aims," 2018, pp. 989–1001. doi: 10.1007/978-3-319-72150-7_80.
[78] T. Preußer and A. Weiss, "Conclusive On-the-fly Validation of High-Level Functional Tests," Dec. 2020. doi: 10.13140/RG.2.2.31182.89923.
[79] F. Gervais, M. Frappier, and R. Laleau, "Generating relational database transactions from eb 3 attribute definitions," *Softw Syst Model*, vol. 8, no. 3, pp. 423–445, Jul. 2009, doi: 10.1007/s10270-008-0104-1.
[80] P. Pernelle, J. C. Marty, and T. Carron, "From traces to indicators for serious games: case study concerning a new way to introduce product lifecycle management," *International*

Journal of Learning Technology, vol. 8, no. 2, p. 127, 2013, doi: 10.1504/IJLT.2013.055670.
[81] Q. Li, "Modeling and exploitation of the traces of interactions in the collaborative working environment," Université de Technologie de Compiègne, 2013. [Online]. Available: https://theses.hal.science/tel-00935411
[82] O. Champalle, K. Sehaba, and A. Mille, "Facilitate Sharing of Training Experience by Exploring Behavior Discovery in Trainees Traces," 2015, pp. 28–41. doi: 10.1007/978-3-319-24258-3_3.
[83] M. Ollagnier-Beldame, C. Brassac, and A. Mille, "Traces and activity: a case study of a joint writing process mediated by a digital environment," *Behaviour & Information Technology*, vol. 33, no. 9, pp. 954–967, Sep. 2014, doi: 10.1080/0144929X.2013.819528.
[84] A. Cordier and B. Fuchs, "Interprétation Interactive de connaissances à partir de traces," in *IC2016:27es Journées francophones d'Ingénierie des Connaissances*, Montpellier, France, Jun. 2016. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01442736
[85] B. Fuchs and A. Belin, "A Trace-Based Approach for Managing Users Experience," in *Workshop TRUE: "'Traces for Reusing Users' Experience"', ICCBR-2012*, J. A. R.-G. a Luc Lamontagne, Ed., Lyon, France, Sep. 2012, pp. 163–172. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01353065
[86] B. Fuchs, "Extraction d'épisodes séquentiels à partir de Traces : application au jeu Tamagocours," in *ORPHEE RDV 2017 : Orphée Rendez-vous réseau d' e-Education*, Font-Romeu, France, Jan. 2017. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01730413
[87] B. Fuchs, "Théorie de la trace, application à l'analyse musicale," in *séminaire MaMux*, IRCAM, Paris, France, Dec. 2012, pp. 1–23. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01353173
[88] L. A. Mojena Román, "Odoo module for extracting and transforming event log traces for Process Mining.," Universidad de las Ciencias Informáticas, Havana, 2018.
[89] A. Rebmann, J.-R. Rehse, and H. van der Aa, "Uncovering Object-Centric Data in Classical Event Logs for the Automated Transformation from XES to OCEL," 2022, pp. 379–396. doi: 10.1007/978-3-031-16103-2_25.
[90] P. De la Paz and A. Torres, "Component for extraction and transformation of event logs from data-aware web systems," in *UCIENCIA*, Havana, Nov. 2016.
[91] T. Saeedeh Jadid, "Quality Evaluation in transformation of event logs into visual representations," Master theses, Gothenburg University, Gotemburgo, 2010.
[92] J. Wang *et al.*, "LogEvent2vec: LogEvent-to-Vector Based Anomaly Detection for Large-Scale Logs in Internet of Things," *Sensors*, vol. 20, no. 9, p. 2451, Apr. 2020, doi: 10.3390/s20092451.
[93] R. Mühlberger, S. Bachhofner, C. Di Ciccio, L. García-Bañuelos, and O. López-Pintado, "Extracting Event Logs for Process Mining from Data Stored on the Blockchain," 2019, pp. 690–703. doi: 10.1007/978-3-030-37453-2_55.
[94] S. Suriadi, C. Ouyang, W. M. P. van der Aalst, and A. H. M. ter Hofstede, "Root Cause Analysis with Enriched Process Logs," in *Lecture Notes in Business Information Processing*, Berlin: Springer, Sep. 2013, pp. 174–186. doi: 10.1007/978-3-642-36285-9_18.
[95] S. B.Sonawane and R. P. Patki, "Process Mining by using Event Logs," *Int J Comput Appl*, vol. 116, no. 19, pp. 31–35, Apr. 2015, doi: 10.5120/20447-2798.
[96] O. R. Zaiane, Man Xin, and Jiawei Han, "Discovering Web access patterns and trends by applying OLAP and data mining technology on Web logs," in *Proceedings IEEE International Forum on Research and Technology Advances in Digital Libraries - ADL'98-*, IEEE Comput. Soc, Apr. 1998, pp. 19–29. doi: 10.1109/ADL.1998.670376.
[97] Á. Figueira, "Mining Moodle Logs for Grade Prediction," in *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality*, New York, NY, USA: ACM, Oct. 2017, pp. 1–8. doi: 10.1145/3144826.3145394.

[98] L. T. Ly, C. Indiono, J. Mangler, and S. Rinderle-Ma, “Data Transformation and Semantic Log Purging for Process Mining,” in *Advanced Information Systems Engineering*, J. Ralyté, X. Franch, S. Brinkkemper, and S. Wrycza, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 238–253.
[99] M. Mayr, S. Luftensteiner, and G. C. Chasparis, “Abstracting Process Mining Event Logs From Process-State Data To Monitor Control-Flow Of Industrial Manufacturing Processes,” *Procedia Comput Sci*, vol. 200, pp. 1442–1450, 2022, doi: 10.1016/j.procs.2022.01.345.
[100] A. Cuzzocrea, E. Damiani, H. Al-Ali, R. Mizouni, G. Tello, and E. Fadda, “From Low-Level-Event-Logs to High-Level-Business-Process-Model-Activities: An Advanced Framework based on Machine Learning and Flexible BPMN Model Translation,” in *The 29th Italian Symposium on Advanced Database Systems*, Pizzo Calabro, Sep. 2021.
[101] D. Calvanese, T. E. Kalayci, M. Montali, and Santoso, “The onprom Toolchain for Extracting Business Process Logs using Ontology-based Data Access,” in *Proceedings of the BPM Demo Track and BPM Dissertation Award*, Barcelona, Sep. 2017.
[102] M. Whittle, B. Eaglestone, N. Ford, V. J. Gillet, and A. Madden, “Data mining of search engine logs,” *Journal of the American Society for Information Science and Technology*, vol. 58, no. 14, pp. 2382–2400, Dec. 2007, doi: 10.1002/asi.20733.
[103] K. Kim, Y.-K. Lee, H. Ahn, and K. P. Kim, “An experimental mining and analytics for discovering proportional process patterns from workflow enactment event logs,” *Wireless Networks*, vol. 28, no. 3, pp. 1211–1218, Apr. 2022, doi: 10.1007/s11276-018-01899-z.
[104] W. Xu, L. Huang, A. Fox, D. Patterson, and M. Jordan, “Mining console logs for large-scale system problem detection,” in *Proceedings of the Third Conference on Tackling Computer Systems Problems with Machine Learning Techniques*, in SysML’08. USA: USENIX Association, 2008, p. 4.
[105] C. Mugali and P. Dandannavar, “Web logs pre-processing and analysis: a survey,” *International Journal of Emerging Technology in Computer Science & Electronics*, vol. 14, no. 2, Apr. 2015.
[106] G. Sudhamathy, “Mining web logs,” in *Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India*, New York, NY, USA: ACM, Sep. 2010, pp. 1–4. doi: 10.1145/1858378.1858435.
[107] Á. Valencia-Parra, B. Ramos-Gutiérrez, Á. J. Varela-Vaca, M. T. Gómez-López, and A. García Bernal, “Enabling Process Mining in Airbus Manufacturing,” in *Business Process Management Cases Vol. 2*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2021, pp. 125–138. doi: 10.1007/978-3-662-63047-1_10.
[108] J. Li, D. Liu, and B. Yang, “Process Mining: Extending α-Algorithm to Mine Duplicate Tasks in Process Logs,” in *Advances in Web and Network Technologies, and Information Management*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 396–407. doi: 10.1007/978-3-540-72909-9_43.
[109] W. Song, H.-A. Jacobsen, C. Ye, and X. Ma, “Process Discovery from Dependence-Complete Event Logs,” *IEEE Trans Serv Comput*, vol. 9, no. 5, pp. 714–727, Sep. 2016, doi: 10.1109/TSC.2015.2426181.
[110] Á. Valencia Parra, B. Ramos Gutiérrez, Á. J. Varela Vaca, M. T. Gómez López, and A. García Bernal, “Enabling Process Mining in Aircraft Manufactures: Extracting Event Logs and Discovering Processes from Complex Data,” in *17th International Conference on Business Process Management 2019 Industry Forum*, Vienna, 2022, pp. 166–177.
[111] F. Paster and E. Helm, “From IHE Audit Trails to XES Event Logs Facilitating Process Mining,” *Stud Health Technol Inform*, vol. 210, p. 40—44, 2015, [Online]. Available: http://europepmc.org/abstract/MED/25991098
[112] T. R. Gruber, “Toward principles for the design of ontologies used for knowledge sharing?,” *Int J Hum Comput Stud*, vol. 43, no. 5–6, pp. 907–928, Nov. 1995, doi: 10.1006/ijhc.1995.1081.
[113] J. F. Sowa, “Generating language from conceptual graphs,” in *Computational Linguistics*, Elsevier, 1983, pp. 29–43. doi: 10.1016/B978-0-08-030253-9.50008-1.

[114] J. F. Maier, C. M. Eckert, and P. John Clarkson, “Model granularity in engineering design – concepts and framework,” *Design Science*, vol. 3, p. e1, 2017, doi: 10.1017/DSJ.2016.16.
[115] C. S. Peirce, *The New Elements of Mathematics*, vol. IV. Netherlands: Mouton Publishers, The Hague, 1976.
[116] J. Venn, “I. *On the diagrammatic and mechanical representation of propositions and reasonings*,” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 10, no. 59, pp. 1–18, Jul. 1880, doi: 10.1080/14786448008626877.
[117] T. R. Gruber, “Toward principles for the design of ontologies used for knowledge sharing?,” *Int J Hum Comput Stud*, vol. 43, no. 5–6, pp. 907–928, Nov. 1995, doi: 10.1006/IJHC.1995.1081.
[118] Terry Halpin, *Information Modeling and Relational Databases*. Elsevier, 2001. doi: 10.1016/B978-1-55860-672-2.X5000-3.
[119] H. Habrias, “Conception de base de données: la méthode Niam et son symbolisme,” *Bulletin de l’EPI (Enseignement Public et Informatique)*, no. 57, pp. 219–229, 1990.
[120] E. Leonhard, “Solutio problematis ad geometriam situs pertinentis,” 1741, *University of the Pacific Scholarly Commons*. Accessed: Mar. 08, 2024. [Online]. Available: https://scholarlycommons.pacific.edu/euler-works/53/
[121] G. Fernandes, S. Ward, and M. Araújo, “Identifying useful project management practices: A mixed methodology approach,” *International Journal of Information Systems and Project Management*, vol. 1, no. 4, pp. 5–21, Feb. 2022, doi: 10.12821/ijispm010401.

ANNEXES

ANNEX 1: OpenProject VS PMBOK processes Mapping (Nodes)

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
0	PP-PMBOK-4.1 Develop Project Charter	3	3	8	0.189765	0.230792	174
1	PP-PMBOK-4.2 Develop Project Management Plan	1	1	8	0.188161	0.220492	0
2	PP-PMBOK-4.3 Direct and Manage Project Work	2	2	6	0.276398	0.325094	670.396284
3	PP-PMBOK-4.4 Manage Project Knowledge	3	3	8	0.195175	0.24477	134.292568
4	PP-PMBOK-4.5 Monitor and Control Project Work	1	1	8	0.188161	0.220492	0
5	PP-PMBOK-4.6 Perform Integrated Change Control	1	1	8	0.188161	0.220492	0
6	PP-PMBOK-4.7 Close Project or Phase	1	1	8	0.188161	0.220492	0
7	PP-PMBOK-5.1 Plan Scope Management	1	1	10	0.157522	0.186	0
8	PP-PMBOK-5.2 Collect Requirements	1	1	10	0.157522	0.186	0
9	PP-PMBOK-5.3 Define Scope	1	1	10	0.157522	0.186	0
10	PP-PMBOK-5.4 Create WBS	2	2	8	0.223058	0.263911	498
11	PP-PMBOK-5.5 Validate Scope	1	1	10	0.157522	0.186	0
12	PP-PMBOK-5.6 Control Scope	1	1	10	0.157522	0.186	0
13	PP-PMBOK-6.1 Plan Schedule Management	3	3	8	0.228205	0.270827	8.911305
14	PP-PMBOK-6.2 Define Activities	8	8	8	0.252125	0.335781	117.999319
15	PP-PMBOK-6.3 Sequence Activities	4	4	6	0.286174	0.339326	113.926619
16	PP-PMBOK-6.4 Estimate Activity Durations	3	3	8	0.225316	0.271415	6.052275
17	PP-PMBOK-6.5 Develop Schedule	7	7	6	0.297659	0.365169	450.652491
18	PP-PMBOK-6.6 Control Schedule	3	3	6	0.286174	0.339326	67.09694
19	PP-PMBOK-7.1 Plan Cost Management	2	2	8	0.194748	0.229468	6.5
20	PP-PMBOK-7.2 Estimate Costs	2	2	6	0.263314	0.309551	98.192063
21	PP-PMBOK-7.3 Determine Budget	1	1	8	0.177291	0.203117	0
22	PP-PMBOK-7.4 Control Costs	3	3	6	0.265672	0.318914	138.671705
23	PP-PMBOK-8.1 Plan Quality Management	1	1	2	0.6	0.666667	0
24	PP-PMBOK-8.2 Manage Quality	1	1	2	0.6	0.666667	0
25	PP-PMBOK-8.3 Control Quality	1	1	2	0.6	0.666667	0

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
26	PP-PMBOK-9.1 Plan Resource Management	4	4	8	0.241192	0.289901	264.52747
27	PP-PMBOK-9.2 Estimate Activity Resources	1	1	10	0.178	0.213274	0
28	PP-PMBOK-9.3 Acquire Resources	3	3	8	0.206497	0.257063	68.887363
29	PP-PMBOK-9.4 Develop Team	3	3	8	0.236702	0.297258	308.241769
30	PP-PMBOK-9.5 Manage Team	2	2	10	0.179435	0.223573	10.628571
31	PP-PMBOK-9.6 Control Resources	1	1	10	0.178	0.213274	0
32	PP-PMBOK-10.1 Plan Communications Management	1	1	8	0.174168	0.195813	0
33	PP-PMBOK-10.2 Manage Communications	3	3	6	0.262537	0.308614	258
34	PP-PMBOK-10.3 Monitor Communications	1	1	8	0.174168	0.195813	0
35	PP-PMBOK-11.1 Plan Risk Management	1	1	10	0.159213	0.190213	0
36	PP-PMBOK-11.2 Identify Risks	1	1	10	0.159213	0.190213	0
37	PP-PMBOK-11.3 Perform Qualitative Risk Analysis	2	2	8	0.225316	0.267657	574
38	PP-PMBOK-11.4 Perform Quantitative Analysis	1	1	10	0.159213	0.190213	0
39	PP-PMBOK-11.5 Plan Risk Responses	1	1	10	0.159213	0.190213	0
40	PP-PMBOK-11.6 Implement Risk Responses	1	1	10	0.159213	0.190213	0
41	PP-PMBOK-11.7 Monitor Risks	1	1	10	0.159213	0.190213	0
42	PP-PMBOK-12.1 Plan Procurement Management	1	1	2	0.6	0.666667	0
43	PP-PMBOK-12.2 Conduct Procurements	1	1	2	0.6	0.666667	0
44	PP-PMBOK-12.3 Control Procurements	1	1	2	0.6	0.666667	0
45	PP-PMBOK-13.1 Identify Stakeholders	1	1	2	0.571429	0.625	0
46	PP-PMBOK-13.2 Plan Stakeholder Engagement	1	1	2	0.571429	0.625	0
47	PP-PMBOK-13.3 Manage Stakeholder Engagement	1	1	2	0.571429	0.625	0
48	PP-PMBOK-13.4 Monitor Stakeholder Engagement	1	1	2	0.571429	0.625	0
49	PM-AC-01-Team Management	3	3	8	0.227621	0.281715	123.080049
50	PM-AC-02-Task Management	27	27	6	0.35743	0.479963	2352.18103
51	PM-AC-03-Project Object Management	2	2	10	0.138414	0.158708	0
52	PM-AC-04-Working Days Schedule Management	3	3	8	0.194748	0.227127	44.486516

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
53	PM-AC-05-Files Management	2	2	9	0.174853	0.208703	85.207432
54	PM-AC-06-Budget Visualization	2	2	6	0.257971	0.295506	0
55	PM-AC-07-Document Visualization	2	2	6	0.264881	0.308614	148.896284
56	PM-AC-08-Forum Visualization	1	1	7	0.206497	0.230284	0
57	PM-AC-08-News Visualization	1	1	7	0.206497	0.230284	0
58	F-PM-AC-01-Organizational Procedures	2	2	7	0.266467	0.315757	0
59	F-PM-AC-02-Organizational Procedures Links	2	2	7	0.268882	0.319128	129.615873
60	F-PM-AC-03-Planned Value	2	2	7	0.268072	0.318192	49.020358
61	F-PM-AC-04-Resource Breakdown Structure	2	2	7	0.271341	0.319315	0
62	F-PM-AC-05-Scope Baseline	8	8	7	0.289902	0.342242	110.804367
63	F-PM-AC-06-Schedule Baseline	2	2	7	0.267267	0.31763	58.107143
64	F-PM-AC-07-Schedule Management Plan	8	8	7	0.289902	0.342242	110.804367
65	F-PM-AC-08-Milestone Charts	2	2	7	0.270517	0.320064	16.875
66	F-PM-AC-09-Organizational Standard Polices, Processes, and Procedures	2	2	10	0.16036	0.195234	43.428571
67	F-PM-AC-10-Electronic Project Management Tools	2	2	6	0.259475	0.302996	88
68	F-PM-AC-11-Social Media Mangement	2	2	6	0.259475	0.302996	88
69	GF-PM-AC-01-Resource Management Plan	2	2	9	0.204598	0.245162	6.178907
70	GF-PM-AC-02-Work Performance Reports	2	2	9	0.190578	0.230355	17.065593
71	GF-PM-AC-03-Scope Baseline	2	2	7	0.275542	0.32512	574
72	GF-PM-AC-04-Schedule Management Plan	4	4	7	0.268882	0.324184	58.767155
73	GF-PM-AC-05-Cost Management Plan	2	2	8	0.216545	0.252301	50.115873
74	GF-PM-AC-06-Project Mangement Plan	18	18	7	0.285256	0.373515	631.99724
75	GF-PM-AC-07-Data Analysis	2	2	7	0.234828	0.271482	3.5
76	GF-PM-AC-08-Data Representation	4	4	7	0.283439	0.336624	308.936303
77	GF-PM-AC-09-Risk Categorizacion	2	2	7	0.277259	0.326619	648
78	GF-PM-AC-10-Decomposition	2	2	7	0.274691	0.32595	10.779144

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
79	GF-PM-AC-11-Rolling Wave Planning	2	2	7	0.274691	0.32595	10.779144
80	GF-PM-AC-12-Activity List	2	2	7	0.274691	0.32595	10.779144
81	GF-PM-AC-13-Activity Attributes	2	2	7	0.274691	0.32595	10.779144
82	GF-PM-AC-14-Milestone List	2	2	7	0.274691	0.32595	10.779144
83	GF-PM-AC-15-Project Management Plan Updates	2	2	8	0.232376	0.275214	2.252235
84	GF-PM-AC-16-Project Management Information System	16	16	5	0.346304	0.443633	2029.464763
85	GF-PM-AC-17-Project Schedule	4	4	7	0.253561	0.296923	14.29408
86	GF-PM-AC-18-Project Charter	4	4	9	0.16036	0.191582	88
87	GF-PM-AC-19-Project Calendars	4	4	7	0.239892	0.287614	218.62032
88	GF-PM-AC-20-Resource Calendars	2	2	9	0.172147	0.206835	3.02381
89	GF-PM-AC-21-Organizational Process Assets	2	2	9	0.171484	0.208008	59.221139
90	GF-PM-AC-22-Virtual Teams	2	2	8	0.199105	0.238577	147.207432
91	GF-PM-AC-23-Information Management	2	2	7	0.223058	0.258614	82.396284
92	F-PM-AC-12-Bar Charts	2	2	7	0.270517	0.320064	16.875
93	PG-PMBOK-01-4 Project Integration Management	7	7	7	0.231169	0.300375	670.896284
94	PG-PMBOK-02-5 Project Scope Management	6	6	9	0.186583	0.243802	430
95	PG-PMBOK-03-6 Project Schedule Management	6	6	7	0.246537	0.320011	63.301969
96	PG-PMBOK-04-7 Project Cost Management	4	4	7	0.214976	0.263804	129.384127
97	PG-PMBOK-05-8 Project Quality Management	3	3	1	1	1	3
98	PG-PMBOK-06-9 Project Resource Management	6	6	9	0.216019	0.288394	309.122028
99	PG-PMBOK-07-10 Project Communications Management	3	3	7	0.210402	0.248261	175
100	PG-PMBOK-08-11 Project Risk Management	7	7	9	0.18896	0.253433	513
101	PG-PMBOK-09-12 Project Procurement Management	3	3	1	1	1	3
102	PG-PMBOK-10-13 Project Stakeholder Management	4	4	1	1	1	6

ANNEX 2: PM Best Practices VS PMBOK processes Mapping (Nodes)

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
0	PLC-01-Project Concept/Initiation	5	5	4	0.359841	0.388122	504.756508
1	PLC-02-Project Definition/Planning	14	14	4	0.404018	0.448895	2383.349214
2	PLC-03-Project Launch/Execution	10	10	5	0.360558	0.402118	1163.86851
3	PLC-04-Project Performance/Control	10	10	5	0.35214	0.399724	2053.155497
4	PLC-05-Project Close	20	20	4	0.343454	0.410681	2793.545166
5	BP-01-Project Charter	17	17	4	0.398678	0.449816	2762.62348
6	BP-02-Feasibility Study	2	2	5	0.266568	0.282044	114.902964
7	BP-03-Stakeholder Analysis	5	5	5	0.275915	0.296041	65.969743
8	BP-04-Requirements Analysis	3	3	5	0.302676	0.326059	114.272929
9	BP-05-Work Breakdown Structure (WBS)	9	9	5	0.324373	0.365285	868.645084
10	BP-06-Activity List	11	11	5	0.334566	0.376059	641.541824
11	BP-09-Scope Statement	14	14	5	0.338951	0.379006	583.383995
12	BP-07-Risk Management	4	4	5	0.300166	0.324217	216.065202
13	BP-08-Contingent Plans	1	1	5	0.288217	0.303867	0
14	BP-10-Resource Scheduling	1	1	5	0.288217	0.303867	0
15	BP-11-Cost Estimating, Parametric Estimating	8	8	5	0.312608	0.344751	312.213559
16	BP-12-Quality Plan	2	2	5	0.290997	0.309853	46.867223
17	BP-13-Communication Plan	3	3	5	0.293831	0.316298	131.582737
18	BP-14-Kick-off Meeting	3	3	5	0.283699	0.305525	20.566067
19	BP-15-Responsibility Assignment Matrix	1	1	6	0.265396	0.281308	0
20	BP-16-Self-directed Work Teams	1	1	6	0.265396	0.281308	0
21	BP-17-Team Building Event	1	1	6	0.265396	0.281308	0
22	BP-18-Progress Communication Meeting	1	1	6	0.265396	0.281308	0
23	BP-19-Quality Inspection / Quality Control / Quality Assurance	11	11	5	0.33211	0.377624	505.01801
24	BP-20-Monitoring of Cost	1	1	6	0.260807	0.279282	0

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
25	BP-21-Resources Leveling	1	1	6	0.265396	0.281308	0
26	BP-22-Progress Report	6	6	5	0.308348	0.343462	138.764009
27	BP-23-Change Request	76	76	4	0.418981	0.480663	4447.047294
28	BP-24-Control Charts: Gantt chart, S-curve, cause-and-effect diagram	2	2	6	0.268148	0.289134	76.262501
29	BP-25-Monitoring Schedule	2	2	6	0.263848	0.28582	109.403454
30	BP-26-Earned Value Management	5	5	6	0.287302	0.318232	292.551943
31	BP-27-Financial Measurement Tools	1	1	6	0.260807	0.279282	0
32	BP-28-Re-baselining	1	1	6	0.260807	0.279282	0
33	BP-29-Customer Satisfaction Surveys	1	1	5	0.256011	0.27965	0
34	BP-30-Team Member Performance Evaluation	1	1	5	0.256011	0.27965	0
35	BP-31-Team Learning Curve	1	1	5	0.256011	0.27965	0
36	BP-32-Lessons Learned/Post-Mortems	2	2	5	0.260057	0.287201	26.839789
37	BP-33-Database for Lessons Learned	1	1	5	0.256011	0.27965	0
38	BP-34-Database of Contractual Commitment Data	1	1	5	0.256011	0.27965	0
39	BP-35-Database of Risks	1	1	5	0.256011	0.27965	0
40	BP-36-Project Web Site	3	3	5	0.290997	0.321087	132.065536
41	BP-37-Data base of historical data	2	2	5	0.29053	0.317127	215.242114
42	BP-38-Life Cycle Costs (LCC)	2	2	5	0.274242	0.299724	74.66105
43	BP-39-Bid/seller evaluation	1	1	5	0.256011	0.27965	0
44	BP-40-Bidders conferences	1	1	5	0.256011	0.27965	0
45	BP-41-Bid documents	2	2	5	0.265396	0.290884	28.553037
46	BP-42-Client acceptance form	1	1	5	0.256011	0.27965	0
47	BP-43-Configuration review	1	1	5	0.256011	0.27965	0
48	BP-44-Graphical presentation of risk information	1	1	5	0.256011	0.27965	0
49	BP-45-Quality Function Deployment (QFD)	3	3	5	0.279321	0.307919	107.464669
50	BP-46-Baseline plan / Project Baseline	1	1	5	0.256011	0.27965	0
51	PP-PMBOK-4.1 Develop Project Charter	2	2	5	0.293831	0.315378	19.290428

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
52	PP-PMBOK-4.2 Develop Project Management Plan	5	5	5	0.324955	0.353499	118.645233
53	PP-PMBOK-11.3 Perform Qualitative Risk Analysis	2	2	6	0.229987	0.248895	20.163374
54	PP-PMBOK-11.1 Plan Risk Management	5	5	5	0.327306	0.361234	321.091389
55	PP-PMBOK-10.3 Monitor Communications	3	3	5	0.286392	0.321271	38.096263
56	PP-PMBOK-10.2 Manage Communications	5	5	5	0.306261	0.351013	134.629544
57	PP-PMBOK-10.1 Plan Communications Management	5	5	5	0.338318	0.372376	271.13566
58	PP-PMBOK-9.6 Control Resources	3	3	5	0.290064	0.325322	24.190138
59	PP-PMBOK-9.5 Manage Team	5	5	5	0.303691	0.342726	161.347874
60	PP-PMBOK-13.4 Monitor Stakeholder Engagement	4	4	5	0.294788	0.333057	71.857937
61	PP-PMBOK-13.3 Manage Stakeholder Engagement	6	6	5	0.312608	0.360313	162.738292
62	PP-PMBOK-13.2 Plan Stakeholder Engagement	4	4	5	0.323792	0.355801	238.545687
63	PP-PMBOK-13.1 Identify Stakeholders	5	5	5	0.338318	0.372836	184.862879
64	PP-PMBOK-12.3 Control Procurements	5	5	5	0.29288	0.332965	45.824162
65	PP-PMBOK-12.1 Plan Procurement Management	5	5	5	0.348077	0.379558	335.11386
66	PP-PMBOK-11.7 Monitor Risks	3	3	5	0.292407	0.328545	82.84182
67	PP-PMBOK-11.6 Implement Risk Responses	2	2	5	0.275915	0.307274	38.341966
68	PP-PMBOK-11.5 Plan Risk Responses	5	5	5	0.311532	0.348435	202.091028
69	PP-PMBOK-11.4 Perform Quantiative Risk Analysis	1	1	7	0.216249	0.23136	0
70	PP-PMBOK-9.3 Acquire Resources	4	4	5	0.291935	0.330387	126.000222
71	PP-PMBOK-9.2 Estimate Activity Resources	3	3	6	0.270553	0.303867	38.979493
72	PP-PMBOK-9.1 Plan Resource Management	4	4	5	0.302676	0.328729	160.973778
73	PP-PMBOK-8.3 Control Quality	11	11	5	0.324955	0.376059	639.093694
74	PP-PMBOK-8.2 Manage Quality	6	6	5	0.322638	0.367587	263.831934
75	PP-PMBOK-8.1 Plan Quality Management	7	7	5	0.366397	0.403959	503.692801
76	PP-PMBOK-7.4 Control Costs	5	5	5	0.292407	0.328729	136.29661
77	PP-PMBOK-7.3 Determine Budget	3	3	6	0.269747	0.29558	87.630593
78	PP-PMBOK-7.2 Estimate Costs	9	9	5	0.315881	0.353867	376.48689

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
79	PP-PMBOK-9.4 Develop Team	3	3	5	0.290064	0.325322	24.190138
80	PP-PMBOK-6.4 Estimate Activity Durations	2	2	6	0.269345	0.301105	15.204341
81	PP-PMBOK-6.3 Sequence Activities	3	3	6	0.270149	0.304788	35.393485
82	PP-PMBOK-6.2 Define Activities	6	6	5	0.315881	0.357735	253.479636
83	PP-PMBOK-6.1 Plan Schedule Management	5	5	5	0.297697	0.323849	283.514267
84	PP-PMBOK-5.6 Control Scope	6	6	5	0.293355	0.332597	136.816769
85	PP-PMBOK-5.5 Validate Scope	2	2	5	0.278462	0.309669	32.76337
86	PP-PMBOK-5.4 Create WBS	4	4	6	0.278462	0.313812	79.028483
87	PP-PMBOK-5.3 Define Scope	4	4	5	0.295269	0.323573	171.353465
88	PP-PMBOK-5.2 Collect Requirements	4	4	5	0.30678	0.335267	160.241908
89	PP-PMBOK-7.1 Plan Cost Management	7	7	5	0.314783	0.347053	304.600159
90	PP-PMBOK-6.6 Control Schedule	3	3	5	0.275915	0.306446	31.905098
91	PP-PMBOK-6.5 Develop Schedule	5	5	5	0.327899	0.371731	127.192182
92	PP-PMBOK-5.1 Plan Scope Management	2	2	6	0.227102	0.245212	7.114487
93	GF-BP-01-Business Documents	2	2	6	0.246594	0.264549	66.255454
94	GF-BP-02-Data Analysis	8	8	5	0.282813	0.314641	147.288253
95	GF-BP-03-Data Gathering	2	2	6	0.251739	0.270074	11.13481
96	GF-BP-04-Expert Judgement	10	10	5	0.315881	0.356077	372.402827
97	GF-BP-05-Product Analysis	2	2	6	0.251389	0.268048	3.174515
98	GF-BP-06-Scope Baseline	6	6	5	0.295269	0.323573	104.198474
99	GF-BP-07- Schedule Management Plan	6	6	6	0.244265	0.269429	151.157232
100	GF-BP-08-Cost Management Plan	4	4	6	0.252089	0.274494	70.821115
101	GF-BP-09-Project Management Plan	17	17	5	0.358416	0.406262	1001.892538
102	GF-BP-10-Data Representation	6	6	5	0.290064	0.320902	233.595277
103	GF-BP-11-Risk Categorization	2	2	6	0.250693	0.271179	58.676623
104	GF-BP-12-Project Documents Updates	2	2	6	0.254213	0.274586	4.185174
105	GF-BP-13-Organizational Process Assets	12	12	5	0.32554	0.36768	694.664253

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
106	GF-BP-14-Decomposition	2	2	6	0.270958	0.292265	2.905371
107	GF-BP-15-Project Documents	34	34	5	0.357708	0.438858	1987.433955
108	GF-BP-16-Project Documents Updates	4	4	5	0.285489	0.313996	6.736598
109	GF-BP-17-Project Scope Statement	2	2	5	0.27218	0.290792	11.545421
110	GF-BP-18-Deliverables	2	2	6	0.269747	0.293646	11.564902
111	GF-BP-19-Risk Register	2	2	6	0.217287	0.234438	7
112	GF-BP-20-Analogous Estimating	2	2	6	0.251739	0.270442	0.80622
113	GF-BP-21-Bottom-Up Estimating	2	2	6	0.251739	0.270442	0.80622
114	GF-BP-22-Project Management Information System	2	2	6	0.251739	0.270442	0.80622
115	GF-BP-23-Testing/Product Evaluations	2	2	6	0.260057	0.282413	9.79253
116	GF-BP-24-Virtual Teams	2	2	6	0.254571	0.273297	3.355284
117	GF-BP-25-Interpersonal and Team Skills	3	3	5	0.278891	0.299632	72.576614
118	GF-BP-26-Meetings	12	12	5	0.34542	0.387385	674.342687
119	GF-BP-27-Quality Management Plan	2	2	5	0.298188	0.330203	40.46338
120	GF-BP-28-Quality Improvements Methods	2	2	6	0.275076	0.299355	5.054297
121	GF-BP-29-Quality Control Measurements	2	2	6	0.273828	0.299171	9.382057
122	GF-BP-30-Work Performance Reports	6	6	6	0.293355	0.324678	56.081642
123	GF-BP-31-Change Request	48	48	4	0.368635	0.447514	2051.738481
124	GF-BP-32-Communication Skills	4	4	6	0.263464	0.287661	9.269555
125	GF-BP-33-Work Performance Data	4	4	5	0.308874	0.33453	6.040464
126	GF-BP-35-Verified Deliverables	2	2	5	0.309402	0.333333	14.987643
127	GF-BP-36-Procurement Documentation	2	2	5	0.303691	0.326519	137.099182
128	GF-BP-37-Agreements	2	2	5	0.297697	0.320626	1.794545
129	GF-BP-38-Procurement Documentation Updates	2	2	5	0.297697	0.320626	1.561861
130	GF-BP-39-Cost Baseline	2	2	7	0.236601	0.25567	11.19559
131	GF-BP-40-Communications Management Plan	2	2	6	0.263848	0.282781	1.66749
132	GF-BP-41-Bid Documents	2	2	6	0.269345	0.286832	33.102381

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
133	F-BP-01-Business Case	2	2	6	0.219927	0.232781	3.158417
134	F-BP-02-Organizational Procedures	2	2	6	0.25104	0.27256	47.822966
135	F-BP-03-Organizational Procedures Links	2	2	6	0.250346	0.271363	29.14069
136	F-BP-04-Planned value	2	2	5	0.263464	0.284622	36.312057
137	F-BP-05-Resource Breakdown Structure	2	2	5	0.256011	0.276427	0
138	F-BP-06-Activity Attributes	2	2	6	0.283699	0.314088	0
139	F-BP-07-Scope Baseline	13	13	5	0.323792	0.360497	248.73644
140	F-BP-08-Project Scope Statement	2	2	6	0.284591	0.307735	0.6745
141	F-BP-09-List of identified risks	2	2	6	0.23385	0.249448	29.637543
142	F-BP-10-Requirements documentation	2	2	6	0.288676	0.319797	8.539188
143	F-BP-11-Quality Control Measurements	2	2	6	0.288676	0.319797	8.539188
144	F-BP-12-Lessons Learned Register	2	2	6	0.288676	0.319797	8.539188
145	F-BP-13-Quality Management Plan	2	2	6	0.285039	0.306077	3.105322
146	F-BP-14-Change log	12	12	5	0.329091	0.358932	219.201802
147	F-BP-15-Cost Forecasts	2	2	5	0.327899	0.355249	5.156972
148	F-BP-16-Schedule Forecasts	2	2	5	0.327899	0.355249	5.156972
149	F-BP-17-Risk Report	2	2	5	0.327899	0.355249	5.156972
150	F-BP-18-Documents Changes	2	2	5	0.297697	0.320626	65.244719
151	F-BP-19-Voting	2	2	5	0.300166	0.323849	51.558766
152	F-BP-20-Performance Measurement Baseline	4	4	5	0.323214	0.352762	0
153	F-BP-21-Approved Change Request Review	2	2	5	0.307301	0.331768	0
154	F-BP-22-Control Charts	2	2	6	0.245924	0.262615	24.044265
155	F-BP-23-Control Thresholds	2	2	6	0.226817	0.243186	11.372969
156	F-BP-24-Rules of Performance Measurement	4	4	6	0.242953	0.265193	54.909548
157	F-BP-25-Retrospectives	2	2	6	0.265007	0.285912	39.378984
158	PP-PMBOK-4.3 Direct and Manage Project Work	7	7	5	0.315881	0.366759	122.677916
159	PP-PMBOK-4.4 Manage Project Knowledge	2	2	5	0.283699	0.314825	12.681495

Id	Label	degree	weighted degree	Eccentricity	closnesscentrality	harmonicclosnesscentrality	betweenesscentrality
160	PP-PMBOK-4.5 Monitor and Control Project Work	6	6	5	0.314236	0.360313	150.408583
161	PP-PMBOK-4.6 Perform Integrated Change Control	7	7	5	0.324955	0.374309	262.718531
162	PP-PMBOK-4.7 Close Project or Phase	4	4	5	0.324373	0.356262	327.746995
163	PP-PMBOK-12.2 Conduct Procurements	4	4	5	0.287302	0.324954	38.565815
164	PP-PMBOK-11.2 Identify Risks	4	4	6	0.267751	0.296409	186.205694
165	GF-BP-42-Activity List	2	2	6	0.270958	0.292265	2.905371
166	GF-BP-43-Scope Management Plan	2	2	6	0.256738	0.276703	47.56409
167	GF-BP-44-Project Management Plan Updates	55	55	4	0.383475	0.464088	2597.640045
168	GF-BP-45-Approved Change Request	8	8	5	0.317544	0.350829	107.059128
169	GF-BP-46-Change Control Tools	2	2	6	0.265396	0.286372	3.560094
170	GF-BP-47-Decision Making	4	4	5	0.288676	0.314641	112.9693
171	GF-BP-48-Control Request	2	2	5	0.300166	0.323204	2.35517
172	PG-PMBOK-01-4 Project Integration Management	7	7	6	0.286846	0.322099	95.491472
173	PG-PMBOK-02-5 Project Scope Management	6	6	6	0.281493	0.313076	216.083393
174	PG-PMBOK-03-6 Project Schedule Management	6	6	5	0.285039	0.315101	121.521205
175	PG-PMBOK-04-7 Project Cost Management	4	4	5	0.274242	0.301657	47.843086
176	PG-PMBOK-05-8 Project Quality Management	3	3	6	0.290064	0.315562	14.641838
177	PG-PMBOK-06-9 Project Resource Management	6	6	6	0.275495	0.306446	71.727004
178	PG-PMBOK-07-10 Project Communications Management	3	3	6	0.270553	0.293554	8.626838
179	PG-PMBOK-08-11 Project Risk Management	7	7	6	0.275495	0.309576	361.666752
180	PG-PMBOK-09-12 Project Procurement Management	3	3	6	0.266176	0.288029	6.168599
181	PG-PMBOK-10-13 Project Stakeholder Management	4	4	6	0.269747	0.29558	18.452535