Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks

Janeiro Lopes Da Silva, Jordan

DOI
10.4233/uuid:92bbf688-b5eb-46f3-a844-501e213cb750

Publication date
2018

Document Version
Final published version

Citation (APA)
Janeiro Lopes Da Silva, J. (2018). *Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:92bbf688-b5eb-46f3-a844-501e213cb750

Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. T.H.J.J. van der Hagen,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op
vrijdag 2 februari 2018 om 15:00 uur
door

Jordan JANEIRO LOPES da SILVA

Mestre em Informática, Pontifícia Universidade Católica do Rio de Janeiro,
Brasil
geboren te Rio de Janeiro, Brazil.

This dissertation has been approved by

promotor: Prof. dr. F.M.T. Brazier
copromotor: Dr. S.G. Lukosch

Composition of the doctoral committee:

Rector Magnificus,	Chairman
Prof. dr. F.M.T. Brazier	TU Delft, The Netherlands
Dr. S.G. Lukosch	TU Delft, The Netherlands

Independent members:

Prof. dr. M. Neerincx	TU Delft, The Netherlands
Prof. dr. R.O. Briggs	San Diego State University, USA
Prof. dr. M.R.S. Borges	Universidade Federal do Rio de Janeiro, Brazil
Prof. Dr.-Ing. J.M. Haake	FernUniversität in Hagen, Germany
Prof. Dr. C.M. Jonker	TU Delft, The Netherlands (reserve)

Cover designed by Silvia Ballerini.

SIKS Dissertation Series No. 2018-04
The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

ISBN 978-94-6186-888-6

An electronic version of this dissertation is available at `http://repository.tudelft.nl/`.

... to my family.

Acknowledgements

It is indescribable, the happiness of finally opening the text editor to write this section. The PhD journey is not known to be a walk in the park, and this moment represents for me one of those, in which you stare at the light in the end of the tunnel and you keep going to see finally what is out there.

I have so many things that I am thankful for and I would like to start by saying that I am grateful to God for helping me (not only) in this journey and providing me all necessary talents, skills and means to complete this phase of my life. I would like to thank my promoter Frances Brazier for her support during my research in Delft. Your encouragement and patience made me a better researcher. I learned with you to always observe the "big picture". Your contributions to my research and your critical thinking definitely shaped this thesis. I would like also to thank Stephan Lukosch, my supervisor, for your guidance and to help me keep on track during my PhD research and the Smart Vortex project. I learned a lot, and not only about research, with you through our several meetings, discussions and trips together. I admire your enthusiasm for research and your determination to keep up on the race for the discovery of new research advancements. I thank you both for your efforts to check in on me and support me towards the completion of this thesis. Even while I was living in Switzerland at the end of my PhD, you guided me and kept me motivated to, step-by-step, cross the finish line.

I am a grateful person to have had so many beautiful friends and colleagues in the Netherlands. Shalini, I don't think that I have the words to describe how important you and your lovely husband Luca (Shaluca team) became to us. Thank you for your valuable company and fruitful discussions; they were absolutely joyful. Although living apart, you are the extension of our family. Michele, you are my first friend since my arrival in the Netherlands. Thank you for integrating me into the section and for the many invitations to enjoy off time in Rotterdam. I value a lot all the events in which were together and the talks we had about academic and non-academic life. Yakup, it was always a pleasure to have your company and, with a well-made cappuccino, share so many stories, discussions and reflections. I would like to thank my section colleagues for their support and interesting discussions during my research: Evangelos, Kassidy, Cagri, JP, Selma, Yilin, Martijn, Michel, Thieme and Tanja. Thanks also to Stefan, Massimo and Ruben for the close collaboration during Smart Vortex.

There are countless friends that I made in different countries since I left Brazil. It was a pleasure to have crossed paths and shared so many beautiful moments together. Thank you: Malcher, Laiola, Christian, Josef, Thomas, Daniel, Giovane, Francesco, Anja, Salvatore, Juliana, Bárbara, Luciano, Janaína, Gabriel and Valentina. I would also like to thank my mate in the Netherlands, Rafael. Without your company and the appreciation of a nice selection of samba music, the Netherlands would not have been the same.

I would also like to thank the support of my friends in Switzerland, who watched the progress of this thesis and cheered for its conclusion. Thanks Elizabeth, Tunç, Rossano, Viviana, Fabio, Chico, Guilherme, Riccardo, Stefano, Julie and Beata.

Eu gostaria de agradecer ao apoio incondicional da minha família durante o trajeto do meu doutorado. Obrigado Fátima, Nilva e Renan. Eu aprendi com vocês tantas lições sobre amor, vida e carreira que me acompanharam também durante esse meu período em que me dediquei à pesquisa. Eu nunca estaria nessa posição se não fosse pela sua dedicação na minha educação e formação e a sua confiança inabalável em mim. Eu sou extremamente grato por vocês terem me apoiado em todas as decisões de vida que tomei, principalmente se isso significou me afastar de vocês e vir viver tão longe.

Vorrei anche ringraziare la mia famiglia italiana, Claudio, Clara, Greta e Matisse. Voi siete sempre stati lì per me, sostenendomi con parole e gesti durante le difficoltà del dottorato. Col vostro affetto mi avete sempre fatto sentire che Modena è anche casa. Il mio amore per l'Italia, e in particolare per il cibo italiano, è cresciuto nel cuore della vostra famiglia.

I would like to thank my two "brothers" Bruno e Gustavo for being part of my life since our childhood. Although the distance might make it difficult to keep in touch, I know that we are there for each other. It is a pleasure to have your company because there is always something joyful and new to learn from the swirl that is our combination.

I would like to thank my wife Beatrice, mio amore, for all your support, love and understanding during my PhD. You are my best friend, my pillar, my ferocious critic and my cheerful supporter. This title does not only belong to me but to us; we made it. Thank you for being by my side. It was, at times, the only thing I needed to keep me motivated.

Contents

Chapter 1

Introduction

The use of information and communication technology (ICT) is causing fundamental structural changes in social, political, economical and cultural aspects of our society [Castells et al., 2006]. The increasing availability of infrastructure to access the Internet empowers people to adapt their ways of communication and interaction with each other through a wide range of devices, such as computer, tablets, smart phones and smart TVs [Gubbi et al., 2013]. Such continuous improvement in Internet Infrastructure laid the foundations for the emergence of new inclusive services that span across geographical boundaries and time differences, allowing people to be engaged in new types of participation, such as voting for on-line elections [Alvarez and Hall, 2003], collective compilation of dictionaries [Benjamin, 2015] or community-based decision making [Aichholzer and Westholm, 2009].

Such changes also influence organizations in other ways, unfolding new types of working relationship, in which the composition of teams includes participants spread across different cities, countries and continents [Alliance, 2015]. One of the main motivations for such geographical distribution of teams is to enable access to a variety of distributed resources, such as services, goods and expertise so that they can create products, services and knowledge that were not possible previously (e.g. [Carmel, 1999] and [Mayer and Pinto, 1998]). For example, software development teams frequently use the follow-the-sun model to speed up the release of a software product using teams working across different time zones [Carmel et al., 2010]. Teams hand over the development responsibility at the end of the day to other teams in a different time zone.

Such situations, however, require new types of coordination mechanisms [Redmiles et al., 2007]. One particular challenge related to coordination of ad-hoc distributed teams, is to align the efforts of distributed participants towards the accomplishment of common goals, despite geographical and temporal barriers that keep them apart [Ellis et al., 1991]. This thesis focuses on coordination support for collective efforts of teams, managing the execution of their interdependent individual tasks, pursued within the context of social systems, such as a company, that span across their boundaries.

The distance between team members also introduces additional challenges for teamwork, as distance contributes to the lack of interpersonal spontaneous communication, establishment of shared contextual information, and conflict in work processes [Hinds and McGrath, 2006]. Therefore, workspace environments become a fundamental value for distributed teams to support collaboration enabling participants to share information with each other, communication allowing participants to create common understanding and coordination enabling the individual organization of efforts towards the achievement of a common goal [Ellis et al., 1991]. For example, *Microsoft Skype* [1] enables communication between distributed team members through voice or video calls; *Google Drive* [2] and *Google Docs* [3] together support the collaboration of distributed teams enabling them to share information and context through the exchange of documents and the history of their modifications; and *Balsamiq* [4] enables the coordination of distributed teams to co-design user interface mockups.

1.1 Domain of Coordination: Diagnosis of Data-Centric Engineering Tasks

In other more specific application domains, such as remote diagnosis of heavy machinery, shared workspaces systems have become a fundamental resource for teams of distributed engineers working together in different geographical locations to diagnose machine anomalies [Biancucci et al., 2014a]. Such teams, often use shared workspaces to support their collaboration, coordination and communication in the context of diagnosis of machine anomalies [Muller et al., 2008].

[1] Microsoft Skype, see www.skype.com for details. Last Access in 03.September.2016

[2] Google Drive, see www.google.com/drive for details. Last Access in 03.September.2016

[3] Google Docs, see www.google.com/docs for details. Last Access in 03.September.2016

[4] Balsamiq, see www.balsamiq.com for details. Last Access in 03.September.2016

1.1.1 Research Case

As machines become more complex due to specialization requiring additional functionalities of subsystems, they require a broad range of skills to understand normal operation and to diagnose anomalous behaviour. A team of experts, specialized in different disciplines is often needed to diagnose these types of machines. These teams need to coordinate their efforts to analyse, identify and describe the root-cause of anomalous machine behaviour [Holmberg et al., 2010]. Otherwise, without teamwork coordination, teams might generate diagnosis outcomes inefficiently, e.g. teamwork redundancy or increased conflict of team members in an already time-constrained task [Janeiro et al., 2012c].

In addition to support for interaction between remote engineers of diagnosis teams, shared workspaces must also integrate machine telemetry data, such as in single user machine diagnosis applications, e.g. [Malagoli et al., 2013][Bauleo et al., 2014], to enable analysis and assessment of machine conditions remotely and together. The traditional model of machine diagnosis, in which a single expert is physically close to a machine to read and analyse its telemetry data is changing [Karlsson et al., 2012]. Manufacturers are focusing their efforts in methodologies and technologies that enable remote machine assessment and diagnosis [Holmberg et al., 2010]. These efforts are particularly concentrated on the development of new services to monitor and collect data generated from equipment to obtain better feedback of its use [Karlsson et al., 2012]. For example, engineers often use data-driven methods to identify machine anomalies in real-time, defining software that analyses telemetry data automatically and trigger alarms upon the detection of an anomaly, instead of evaluating large amounts of data themselves to search for anomalies [Alzghoul et al., 2014]. Whenever software identifies an anomaly, engineers analyse in depth data associated to it [Malagoli et al., 2013]. The availability for teams to retrieve and filter real-time telemetry data of remote machine is invaluable for engineers to gain insights about operational conditions and limitations of certain machines and define a diagnosis for them, contributing to reduce machine downtime and therefore, improve their availability [Iung et al., 2009].

1.1.2 Research Objectives

Coordination is challenging for teams, as it is difficult for them to define one single strategy that is suitable to coordinate teamwork [Bernstein, 2000][Buttler et al., 2011].

Most shared workspace systems do not offer flexibility to teams to choose a suitable coordination mechanism. Rather, they are designed based on one coordination mechanism to be used by all team members [Gutwin et al., 2008]. For example, some systems are based on pre-defined processes to coordinate teamwork [Ellis et al., 2005], whereas other systems rely on mutual awareness of team members to coordinate their efforts [Cheng et al., 2003].

Teams might have different working preferences to execute a task together. For example, some teams may prefer to work in such a way that they follow the cycle of problem analysis, solution search or synthesis, and then the execution of a plan [Jablonski and Bussler, 1996]. Whereas other teams may prefer to use plans as resources for action [Suchman, 1987], which are used in conjunction with the environment to articulate and reason about the next action steps.

The prescription of a single coordination strategy is likely to have a negative impact on teamwork. Teams need to be able to choose between different coordination mechanisms and use them according to the type and context of tasks [Van de Ven et al., 1976]. Therefore, the assumption on which this thesis is based is that teams should be self-empowered to choose the coordination mechanism most suitable for their preferences and to the type of the task that they have to accomplish.

The objective of this thesis is to design a shared workspace system that flexibly coordinates teamwork in diagnosis tasks.

This thesis explores whether shared workspaces can be designed to flexibly support the coordination of data-centric diagnosis tasks, such as the diagnosis of machine anomalies.

The main objectives of this thesis are:

- *i)* gain insights on coordination mechanisms that support teams to perform a diagnosis task and
- *ii)* integrate such mechanisms in a shared workspace system for diagnosis tasks.

The following research question targets the required knowledge to achieve the research objectives:

Is it possible to design coordination mechanisms in a shared workspace system to flexibly support collaborative diagnosis in data-centric engineering tasks?

The general research question requires knowledge of coordination mechanisms that support distributed teams in shared workspace systems. The following research question focuses on requirements to support coordination mechanisms in teamwork:

RQ1 Can requirements for the design of coordination mechanisms that flexibly support collaborative diagnosis tasks be identified?

The general research question requires knowledge about diagnosis models and their adaptation to support teams in the execution of collaborative diagnosis tasks. The following research question focuses on requirements to support collaborative diagnosis.

RQ2 What are the requirements to support collaboration of teams in data-centric machine diagnosis tasks?

The general research questions require knowledge about the design of a flexible shared workspace system to support teams with different coordination mechanisms in collaborative diagnosis tasks. The following research question addresses the challenge to design an architecture of a shared workspace system.

RQ3 Can an architecture of a shared workspace system be designed to support flexible coordination of teams involved in data-centric collaborative diagnosis tasks?

The general research question requires knowledge about the capability of a shared workspace system to flexibly coordinate teams involved in collaborative diagnosis tasks through different coordination mechanisms. The following research question assesses such capabilities.

RQ4 Can a shared workspace system flexibly coordinate collaboration in data-centric diagnosis tasks?

The remainder of this thesis pursues knowledge to answer the aforementioned research questions. Systems are designed and implemented for this purpose using the design science research methodology. Chapter 2 focuses on research question RQ1, chapter 3 on RQ2, chapter 4 and 5 on RQ3 and chapter 6 on RQ4.

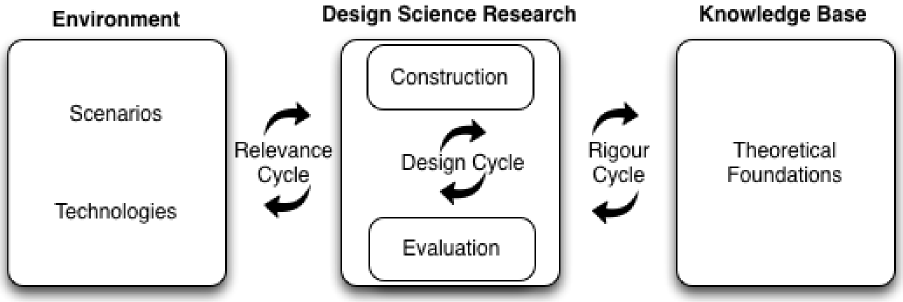


Figure 1.1: Framework of Design Science Methodology [Hevner and Chatterjee, 2010].

1.2 Research Methodology

This thesis follows the guidelines of the design science research methodology [Hevner and Chatterjee, 2010]. This methodology consists of three building blocks, described in Figure 1.1: Environment, Design Science Research and Knowledge Base. In addition, the methodology also consists of three cycles, described in Figure 1.1: Relevance, Design and Rigour Cycle.

1.2.1 Environment

The Environment building block represents the problem space phenomena of interest.

The Environment in this thesis is defined by a project named Smart Vortex[5], an FP7 EU Project that investigated the various uses of telemetry data to enable remote machine diagnosis [Consortium, 2010]. The Smart Vortex project investigated large amounts of data generated during the lifecycle of machines to improve their usage. Four industrial partners participated in project, including three machine manufacturers, Volvo Construction Equipment, Bosch Rexroth Group and Sandvik Coromant, and a major 3D design company, Dassault Systèmes.

The industrial partners indicated that there is an increasing volume of product lifecycle information that is already too excessive for diagnosis engineers to process. In practice, there are often situations in which *the usage of data is either limited or even completely omitted* (challenge *c1*). The

[5] The Smart Vortex project: `www.smartvortex.eu`. Last Access in 03.September.2016

project aimed to investigate different strategies of diagnosis to capture and reuse the various types of data generated in the lifecycle of machines. The focus chosen in the project was to use machine telemetry data to enable remote diagnosis of machines. For the manufacturers, diagnosis is the most complex part of the whole maintenance process of machines, as it requires *specific knowledge about particular components and subsystems* (challenge *c2*), e.g. the regulation mechanism in the hydraulics subsystem that controls the flow of cooling oil, and also global knowledge about the interaction between subsystems to enable machine-specific functionalities, e.g. the relationship of hydraulics, electrical and mechanical subsystems that enable a wheel loader bucket to lift material. Such knowledge is necessary to differentiate between normal and anomalous machine operation. Machine diagnosis requires *collaboration* (challenge *c3*) between experts and the coordination of theirs actions to analyse particular details of machines that are further combined to draw conclusions. Once the diagnosis is formulated and countermeasures are defined, further actions such as replacement and repairing of specific machine parts become straightforward tasks.

1.2.2 Design Science Research

The Design Science Research building block performs continuous refinement of the design of an artefact and its evaluation to meet needs identified in the Environment building block. The Research building block in this thesis is a continuous design and evaluation process of different coordination mechanisms used by distributed teams that support their diverse working preferences in remote machine diagnosis scenarios.

1.2.3 Knowledge Base

The Knowledge Base building block provides foundations, methodologies and formalisms that support the development of research, such as theories, frameworks and diagnosis models as measures and validation criteria from specific literature. The Knowledge Base building block in this thesis uses theories from distributed teamwork, coordination theory, collaboration, shared diagnosis models and validation methodology.

1.2.4 Relevance Cycle

The Relevance Cycle identifies opportunities in actual application environments for the development of new artefacts for collaborative diagnosis. A series of informal interviews and workshops with engineers from industrial machinery manufacturers specialised in diagnosis revealed that remote collaboration is a major challenge for machine diagnosis. The complexity of machines requires experts with different specializations to collaborate in diagnosis tasks to identify and discuss machine anomalies remotely and avoid equipment breakdown. The coordination of teamwork in such scenarios is challenging because it requires teams to consider the working preferences and expertise of all team members and to identify suitable coordination mechanisms, according to the context of a task.

1.2.5 Rigour Cycle

The Rigour Cycle identifies existing research approaches that address the same problem: a shared workspace system designed to support collaboration and coordination of distributed team members involved in diagnosis tasks. The design is based on scientific literature about generic patterns of collaboration in teams, functionalities often involved in remote diagnosis of machinery and the different coordination support types for teams.

1.2.6 Design Cycle

The Design Cycle builds the intended artefact, based on the environmental analysis based on literature knowledge. This thesis focuses on machine diagnosis scenarios, in which a shared workspace system provides a means for team members, to filter telemetry data, facilitating its analysis (*c1*). This thesis also explores the use of different mechanisms to coordinate interaction between team members during diagnosis tasks (*c2*), based on a diagnosis model that supports their collaboration (*c3*). This thesis develops a methodology to assess the usefulness, usability, workspace awareness information and the quality of collaboration of participants using the shared workspace system to perform a machine diagnosis task (*c4*). Finally, this thesis discusses the interaction of these participants with the shared workspace system based on the developed evaluation methodology previously described (*c5*). In this context, the third cycle represents the contribution of this thesis to the identified knowledge base.

1.3 Thesis Outline

The structure of this thesis is as follows:

- *Chapter 2.* This chapter discusses the characteristics of distributed teams and coordination mechanisms used in teamwork. It defines a spectrum of coordination support and properties of its two extremes and discusses requirements for shared workspace systems to support different types of coordination mechanisms. Scenarios defined in the Environment building block are used to derive requirements for different coordination mechanisms that support distributed teams. The Relevance Cycle provides these requirements to the Design Science Research building block, contributing to the construction of coordination mechanisms in the shared workspace system that support distributed teams. The chapter uses theories and models from coordination, collaboration and distributed teams from the Knowledge Base building block to define coordination mechanisms for teams, through the Rigorous Cycle.

- *Chapter 3.* This chapter discusses new technologies to monitor mechanical machines remotely and introduces a new model for collaborative diagnosis. The chapter also defines requirements for shared workspace systems to support such collaborative diagnosis. This chapter uses scenarios defined in the Environment building block to derive requirements to support distributed teams in remote diagnosis tasks. The Relevance Cycle provides these requirements to the Design Science Research building block, contributing to the construction of diagnosis functionalities in the shared workspace system. The chapter uses problem-solving diagnosis models from the Knowledge Base building block through the Rigorous Cycle to define a collaborative diagnosis model.

- *Chapter 4.* This chapter describes the design and architectural decisions for a shared workspace system that supports teams in collaborative diagnosis tasks through different coordination mechanisms. This chapter uses established requirements based on the theories of coordination, collaboration and distributed teams and models from diagnosis to design and build a shared workspace system in the Design Science Research building block.

- *Chapter 5.* This chapter describes the implementation details and the rational of the different tools, technologies and frameworks used to implement the shared workspace system, described in chapter 4. This chapter describes implementation details of the architecture of the shared workspace system in the Design Science Research building block.

- *Chapter 6.* This chapter describes experiments used to evaluate a shared workspace system in diagnosis tasks for two different machine manufacturers. The evaluation is divided in two parts: *i)* the evaluation of diagnosis and coordination functionalities implemented in the shared workspace system and *ii)* the evaluation of the quality of collaboration in the experiments through the use of the shared workspace system. This chapter uses evaluation methods and models from the Knowledge Base through the Rigorous Cycle and evaluates the shared workspace system. The evaluation of the system is an iterative process, described in the Design cycle of this chapter, as it provides information, used to improve coordination support and diagnosis functionalities of the shared workspace system.

- *Chapter 7.* This chapter concludes the thesis with a discussion of the implications and limitations of this research, possible areas for future work and recommendations for stakeholders involved in diagnosis tasks that seek to use shared workspace systems with flexible coordination mechanisms. This chapter provides contributions to the Environment building block, through the Relevance Cycle building block, describing scenarios that use the shared workspace and provides contributions to the Knowledge Base building block through the Rigorous Cycle, describing the results of using different coordination mechanisms in remote diagnosis task for distributed teams.

Chapter 2

Coordination Support for Distributed Teams

Coordination of teamwork is an emergent phenomenon involving the use of strategies and behaviour patterns aimed at integrating and aligning a actions, knowledge, and objectives of interdependent members, with a view to attaining common goals [Espinosa et al., 2002] [Rico et al., 2008]. For example, in collaborative design of artefacts in general, coordination is necessary to enable teams to overcome a set of challenges, such as search and definition of a specific collaborative problem and establishment of work norms among stakeholders to contribute to incremental design of shared artefacts [Piirainen et al., 2012].

Coordination ensures that a team functions as a unified whole [Brannick and Prince, 1997] [Ven et al., 1976]. When a team attains a high level of coordination, all members contribute to the end result, but when coordination is poor, there are negative impacts on outcomes [Steiner, 1972]. Therefore, to mitigate coordination problems, coordination mechanisms are used to achieve a desired goal [Malone and Crowston, 1994].

Explicit coordination mechanisms are used intentionally by team members to manage multiple interdependencies between activities [Malone and Crowston, 1994]. [Espinosa et al., 2002] distinguishes two types of coordination mechanisms, one based on plans (e.g. procedures, schedules, tools and plans) and another based on spontaneous definitions of coordination plans, based on circumstantial information.

Coordination mechanisms based on plans support the coordination of teamwork for the routinised aspects of a task, as the dependencies involved in the task are predictable and therefore are able to be programmed or routinised [March and Simon, 1958]. For example, in software development activity, a team of developers has roles assigned in advance to build specific software components that together contribute to the functionalities of a software system. In this case, a possible plan is to assign specific software developers to work simultaneously on different parts of the code to avoid the interference of the work of other developers or conflicts [Espinosa et al., 2007]. Conversely, coordination mechanisms based on spontaneous coordination are more suitable for situations in which routines change, are no longer applicable for a task, or when a task requires no routine [March and Simon, 1958]. In such situations, team members need to interact to define new strategies to perform a task. For example, in software development activities, it is difficult to predict or anticipate missed deadlines or hardware failures, therefore developers and managers need to often adjust coordination plans to overcome unpredicted situations [Espinosa et al., 2007].

Coordination mechanisms are particularly important for distributed teams, for which miscommunication and misunderstanding, information sharing, feedback exchange, and the establishment and maintenance of shared team identity are known challenges [Hinds and McGrath, 2006].

This chapter discusses the characteristics of teams, in general, and distributed teams, in particular. The chapter highlights the specific challenges of distributed teams. Subsequently, the chapter presents two divergent coordination mechanisms designed to mitigate coordination problems. The chapter concludes with the definition of a theoretical spectrum that integrates opposing coordination mechanisms and serves as an implementation reference for CSCW technology.

2.1 Distributed Teams

This thesis defines a team to be a collection of individuals who are interdependent in their tasks, share responsibility for their outcomes, recognized by others as an intact social entity embedded in one or more larger social systems (e.g., business unit or an organization), and manage many of their relationships across organizational boundaries [Cohen and Bailey, 1997]. For example, in a distributed design team working on the manufacturing of a machine, all members still share the responsibility for the quality of the final produced machine, although they work independently on different machine components. Note that by this definition, people who work on independent tasks are not considered a team. For example, a department of electrical engineers who work on separate projects is not a team. These engineers work independently of each other, do not share responsibility for their outcomes, and are not interdependent.

Teams are different from groups with regard to their level of interdependency and integration among members [Katzenbach, 1993]. Whereas groups are two or more people who work together to achieve a goal, teams extend this concept [Stott and Walker, 1995]. Teams share common and clear goals, are aware of the nature of their independent roles and the complementarity of their respective skills [Fisher et al., 1997].

The continuous development of information and communication technology (ICT) offers new opportunities for organizations to share work across geographic distributed teams [Constant et al., 1996]. Through the opportunities that technology creates, organizations are able to procure talented workers without the limitation of geographical boundaries to any specific distributed teams [Powell et al., 2004]. Distributed teams are teams of individuals who work across time, space and organizational boundaries brought together by information and communication technologies to accomplish one or more tasks [Jarvenpaa and Leidner, 1998]. They are often formed temporarily when collaboration is necessary to provide specific deliverables, or to fulfil specific customer needs [Lipnack, 1997, Chase, 1999].

Distributed teams typically resort to the use of technology not only for communication but also to support their teamwork using computer-supported cooperative work (CSCW) technology [Ellis et al., 1991]. The use of such technology is designed to support distributed teams to move through a task to attain shared goals [Briggs et al., 2003b]. To this purpose, patterns of collaboration used by teams in practice have been identified, analysed and evaluated. [Briggs et al., 2003b] distinguishes six patterns of collaboration.

1. Generate: the goal of this pattern is to support a team to expand the number of shared concepts. Team participants introduce new concepts, moving from a state of having fewer concepts to a state of having more concepts. For example, in requirement engineering, a task that defines a set of requirements involves activities associated to the generate pattern, such as brainstorming with stakeholders from different backgrounds (e.g. users, customers, managers, domain experts, and developers) [Boehm et al., 2001].

2. Reduce: the goal of this pattern is to reduce the cognitive load of a team by reducing the number of concepts. Team participants reduce

the number of concepts, moving from a state of having many concepts to a state of having a focus on the few worthy of further attention. For example, in requirements engineering, a task that defines a set of requirements involves activities associated to the reduce pattern, such as the convergence of a list of requirements [Boehm et al., 2001]. Whenever a list of requirements is prepared, stakeholders involved in the task formulate a list of non-redundant and unambiguous requirements, based on the ones generated previously.

3. Clarify: the goal of this pattern is to further explain concepts generated by a team, moving from having less to having more shared understanding of concepts and of the terms used to express them. For example, in requirement engineering, a task that defines a set of requirements involves activities associated to the clarify pattern, such as the review of existing requirements [Boehm et al., 2001]. Stakeholders refine and together customize proposed requirements, recommending changes or further explanation. This type of activity is necessary to avoid misinterpretation and misunderstanding of specific terms involved in requirement engineering of software, especially in a team of stakeholders with different backgrounds.

4. Evaluate: the goal of this pattern is to assess the relevance of concepts in relation to each other, moving from less to more understanding of the relative value of the concepts under consideration. For example, in risk assessment, a task that assesses financial risks for an institution involves activities associated to the evaluate pattern, such as the evaluation of control mechanisms for risks [Van Grinsven and de Vreede, 2003].

5. Organize: the goal of this pattern is to create and understand relationships among generated concepts, moving from less to more understanding of the relationships among considered concepts. For example, in risk identification, a task that categorizes different risks for an institution involves activities associated to the organize pattern [Van Grinsven and de Vreede, 2003].

6. Build Consensus: the goal of this pattern is to achieve mutual acceptable commitments. For example, heuristic evaluation tasks of a software user interface involves activities associated to the build consensus pattern [Nielsen and Molich, 1990]. After the identification of

usability problems stakeholders must reach consensus with regard to their priorities.

2.2 Structured Coordination Support

Structured mechanisms typically use the concept of a process to plan coordination [Jablonski and Bussler, 1996, Hammer et al., 1977, Zisman, 1978, Mohan et al., 1995]. Processes specify different sequence of activities needed to support teamwork, the conditions in which activities are executed, the flow of data between activities, indicating the team members responsible for the execution of the activities and the tools to be used with each activity [Oberweis, 2005, Jablonski and Bussler, 1996]. For example, a collaboration process used to write a report about the ageing situation of the population of a particular city specifies the human actors involved (e.g. Bob, Jim, Larry and Susan), their expected roles and skills (e.g. one manager and three analysts, respectively), and the necessary activities to generate the report (e.g. request and send report, request and send age information, and request and send writing) [Ellis et al., 2005].

This section first describes the collaboration engineering approach that uses process descriptions and collaboration techniques to guide teams towards the accomplishment of a task. Subsequently, the section provides example of systems based on process descriptions to guide their participants.

2.2.1 Collaboration Engineering

In many situations, professional facilitators play an important role in collaborative processes to improve team productivity [Dickson et al., 1996, Griffith et al., 1998, Niederman et al., 1996]. Facilitators are experts in the design and support of collaborative processes that involve management of relationships among team members, tasks and technology. They structure tasks and contribute to the effective accomplishment of their outcomes [Bostrom et al., 1993].

Collaboration engineering, as a field of research, focuses on the transferability of facilitation skills to teams, sharing responsibility for their own execution of collaboration processes [Briggs et al., 2010]. It aims to structure team interaction, suggesting collaboration techniques that tackle problems, helping teams to achieve the desired goals of tasks [Briggs et al., 2001]. [Briggs et al., 2003b], for example, distinguishes to this purpose patterns of collaboration, known as *thinkLets*, for single process activities, based on the six patterns of collaboration described in 2.1. Processes are designed in anticipation, providing the prescription of a set of activities associated to thinkLets, facilitating the interaction of team members to achieve their goals [Briggs et al., 2006].

2.2.2 Collaboration Process-based Systems

Different software systems implement the concept of a process to coordinate actions of team members. The CACE (Computer Assisted Collaboration Engineering) tool supports the design of facilitation processes using thinkLets as references [Briggs et al., 2010]. Designers of facilitation processes use the CACE tool to search and instantiate thinkLets that are appropriate for process activities; to combine several software components that implement different functionalities described by thinkLets; and to execute the facilitation process for a team.

The Caramba system enables the execution of pre-defined collaboration processes and the extension of the process during its execution [Dustdar, 2004]. Caramba implements a specific coordination model that has several work distribution templates to assign process activities to team members, according to their skills and roles.

The CONTact platform supports automatic facilitation of brainstorming sessions through context-based adaptations [Veiel et al., 2013]. An application introduces structures, triggered by adaptation rules, that aid a group in the session to reduce the information overload caused by a large number of ideas. Groups interrupt the brainstorming session, categorize the ideas according to the proposed structure and then continue with the session.

The GSSOne system extends the concept of thinkLets implemented in the CACE tool with more details, decomposing and describing the set of activities that implement a thinkLet [Knoll et al., 2009]. In addition to the description of a facilitation process, designers also specify the data and events of a collaboration process that flows between activities, e.g. select a discussion topic, create a contribution and list all comments for a contribution.

Overall, the purposes of these systems is to enable teams to coordinate their actions through the prescription of collaboration processes that involve actions, techniques, participants and tools to achieve desired goals. The basic assumption is that through the prescription of collaboration processes for standard, repetitive and well-defined problems, team members may gain efficiency in teamwork [Jerry Fjermestad, 2000, Witte, 2007, Briggs et al., 2010].

2.3 Unstructured Coordination Support

Unstructured mechanisms are based on the approach of spontaneous coordination based on situated actions, in which the course of action depends on the available resources of circumstances, e.g. social circumstances, personal or materials [Suchman, 1987]. This approach aims to understand the use of situated circumstances to achieve intelligent action, rather than abstracting actions from circumstances and represent it as plans.

Team members must mitigate the lack of coordination whenever a process does not describe, partially or entirely, the course of action for a team [Suchman, 1987]. To this end, team members need to understand the circumstances of teamwork (e.g. available resources, current and past actions) and of their pursued task to create and refine continuously coordination plans for their actions to achieve the goals of a task [Suchman, 1987, Rico et al., 2008]. In teamwork, awareness information about team members is necessary for a team to coordinate their actions, especially in distributed teams working with CSCW technology, as it provides common knowledge and shared understanding of their current and past activities [Bellotti and Bly, 1996]. Workspace awareness information, in particular, requires exchange of information between two or more team members about the shared workspace to integrate their respective contributions [Kraut and Streeter, 1990]. For example, in software development projects, teams must be aware of the actions of each team member to coordinate their efforts towards the extension and integration of software artefacts. Such information aids both fine and coarse-grained coordination of actions, as it informs team members about the temporal and spatial boundaries of their actions, and helps them to integrate a next action into a flow of actions [Gutwin and Greenberg, 2002].

Therefore, unstructured coordination mechanisms are mechanisms that rely on the acquisition and provision of workspace awareness information to team members, by which they understand the actions of other members, as part of their circumstances, to coordinate their actions.

2.3.1 Workspace Awareness

Workspace awareness is defined as the up-to-the-moment perception and comprehension of the interaction of a person with the workspace [Gutwin and Greenberg, 2002]. In such situations, workspace awareness information becomes an important requirement for teams to agree on the coordination strategy that involves the division of labour and planning of their collective efforts [Bellotti and Bly, 1996]. During the performance of a task, some teams might reorganize the activities that each member performs, based on the accomplishments of each team member, the planned sub-tasks by each team member and the remaining sub-tasks to be executed. Based on the activity of a colleague, a team member may decide to begin a complementary task, to assist them with an activity, or to move to a different area of the workspace to avoid a conflict [Gutwin and Greenberg, 2002].

Workspace awareness is based on the concept of situation awareness as both types of awareness share similar characteristics. Situation awareness is a state of knowledge in which a person perceives the elements in an environment, with regard to time and space, comprehend their meaning and project their status in a near future [Endsley, 1995]. This definition distinguishes three levels:

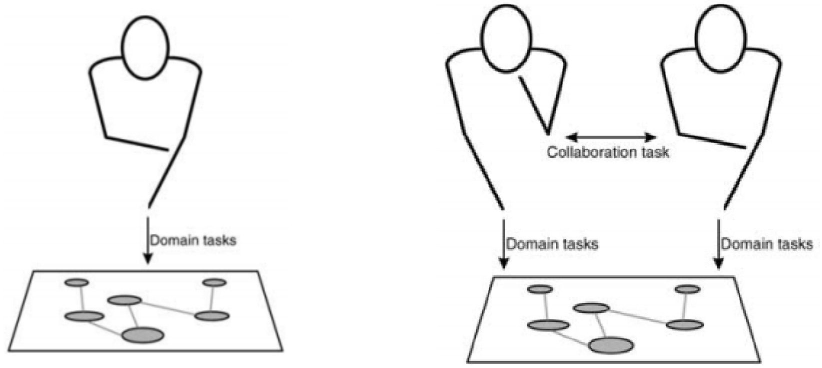
- Level 1: perception of the elements in an environment. At this level, actors perceive the status, attributes, and dynamics of relevant elements for their goals in the environment.

- Level 2: comprehension of a current situation. At this level, actors integrate the disjoint perceived elements from level 1 and create an understanding of their significance with regard to their goals.

- Level 3: projection of future status. At this level, actors anticipate the variation of environment elements, at least in a near future, to predict their status in the future.

Situation awareness is exemplified through different contexts such as the operation of aircrafts, air traffic control, operations of power plants and firefighting. In the case of air traffic control, for example, air traffic controllers manage and project the paths of an ever-increasing number of aircrafts in a airspace area. Such controllers must maintain an up-to-date assessment of the changing locations of aircrafts (perception and comprehension) and their projected locations relative to each other, given the parameters of each aircraft (e.g. destination and speed) to ensure their minimum separation for safety and efficient landings and takeoff [Endsley, 1995].

Conversely, workspace awareness is considered to be a specialization of situation awareness because of the introduction of different actors working in a shared workspace. Workspace awareness is awareness of team members and their interaction with the shared workspace, rather than just awareness of the workspace itself. In addition, workspace awareness is limited to events that happen in the workspace, inside the temporal and physical bounds of



(a) The focus is on the workspace and the domain tasks.

(b) The focus is on the workspace, on the domains tasks and on the collaboration task.

Figure 2.1: The enhancement of complexity in the interaction among actors by the introduction of the collaboration task [Gutwin and Greenberg, 2002].

the task that the team performs. This means that workspace awareness differs from informal awareness of people who are available and searching for an opportunity to collaborate spontaneously [Gutwin et al., 2008], and from awareness of cues and turns in verbal conversation, typically present in co-located teams [Whittaker et al., 1994].

Workspace awareness requires the focus of each team member on *i)* the collaboration task, *ii)* different team members using the shared workspace, in addition to the *iii)* execution of a domain task (e.g. air traffic control) [Gutwin and Greenberg, 2002]. Individuals who work alone in the workspace only have the focus on the use of the workspace and the execution of domain tasks, as illustrated in Figure 2.1a. However, in a collaborative situation, in addition to the focus on the workspace and the domain tasks, actors must undertake the collaboration task, such as illustrated in Figure 2.1b.

2.3.2 Elements of Workspace Awareness

Even though a team member can keep track of many elements in a shared workspace, a basic set of workspace awareness elements was established in a workspace awareness framework, based on their repeated use in different scenarios that involve collaboration [Gutwin and Greenberg, 2002]. The basic set of the elements are those that answer questions such as: *who?*, *what?*, *where*?, *when?* and *how?*.

Category	Element	Specific Question
Who	Presence	Is anyone in the workspace?
	Identity	Who is participating? Who is that?
	Authorship	Who is doing that?
What	Action	What are they doing?
	Intention	What goal is that action part of?
	Artefact	What object are they working on?
Where	Location	Where are they working?
	View	What can they see?
	Reach	Where can they reach?

Table 2.1: Elements of workspace awareness relating to present activities [Gutwin and Greenberg, 2002].

The framework proposed by [Gutwin and Greenberg, 2002] uses these questions and their answers to define the basic set of elements that aggregate workspace awareness. They are all common sense elements that concern the interactions between a person and the environment [Gutwin and Greenberg, 2002]. The elements and the respective questions that they answer are presented in two different tables, Table 2.1 and Table 2.2. Table 2.1 contains elements that relate to the activities performed in the present, whereas Table 2.2 the activities performed in the past.

Table 2.1 presents nine elements: presence, identity, authorship, action, intention, artefact, location, view and reach. Awareness of presence and identity concerns the knowledge that other people interacts with the workspace and who they are. The element authorship describes the relationship between an activity and the person that performs it. Awareness of actions and intentions is the understanding of the activities that other people execute and their purpose, either in detail or at a general level. Awareness of artefact refers to knowledge about the object with which a person is working. Location, gaze, and view relate to where the person is working, where they are looking, and what they can see. Awareness of reach involves understanding the area of the workspace in which a person can change objects.

Table 2.2 presents six elements that are related to past activities, such as the history of: actions ("how" category), artefacts, events, presence, location and action ("what" category). Action and artefact history concern the details of events that have already occurred. Action history describes activities executed by an actor on the workspace and artefact history describes the activities the modified a object of the workspace. Event history describe

Category	**Element**	**Specific Question**
How	Action history	How did that operation happen?
	Artefact history	How did this artefact come to be in this state?
When	Event history	When did that event happen?
Who (past)	Presence history	Who was here, and when?
Where (past)	Location history	Where has a person been?
What (past)	Action history	What has a person been doing?

Table 2.2: Elements of workspace awareness relating to past activities [Gutwin and Greenberg, 2002].

the occurrence time of events on the workspace. The remaining three elements describe the history of events with regard to presence, location, and action.

2.3.3 Systems based on Workspace Awareness Information

Several software systems in the literature of CSCW use and implement the concept of workspace awareness as a means to support a team to coordinate their efforts to accomplish a goal together. Depending on the domain of the system, different workspace awareness information types are exchanged among team members, based on the meta-models of situation awareness [Endsley, 1995] and workspace awareness [Gutwin and Greenberg, 1996, Gutwin and Greenberg, 2002]. The widespread and predictable use of workspace awareness information enabled the definition of a set of patterns of interaction as guidelines to support awareness in software systems [Schümmer and Lukosch, 2013].

For example, in collaborative software development, the Jazz system provides workspace awareness of the development team [Cheng et al., 2003]. The system shows the team members that are connected to the system, their status messages with regard to their current activities, indications of modified software artefacts and the author of the modifications. Similarly, FASTDash (Fostering Awareness for Software Teams Dashboard) provides information about the team members who edit specific software artefacts, modify them concurrently, debug them or visualize their source code [Biehl et al., 2007].

In collaborative user interface sketching, the Gambit system enables the co-creation of user interface sketches [Sangiorgi, 2014]. The system shows an overview of all sketches of a session in a main display and indicates the areas of the sketches in which the members of a team work. By providing such workspace awareness, team members are able to coordinate their efforts to modify sketches, discuss their integration and future steps in the design.

In diagram-based brainstorming processes, the CLSD system provides workspace awareness of shared diagrams [Azevedo et al., 2013]. The system supports participants in building brainstorming diagrams with virtual objects. Whenever a participant modifies a virtual object, the system provides workspace awareness information to the team in real-time through visual cues about the participant that modifies an object and other surrounding objects.

Overall, these systems provide workspace awareness as means for teams to coordinate their actions to better achieve the goals of a task. The basic assumption is that through workspace awareness, teams can understand the way in which their actions contribute to the accomplishment of a task. Therefore, they use workspace awareness information to coordinate their actions spontaneously to improve the efficiency of teamwork, avoiding duplication of work or inconsistency of outcomes by conflict within a team [Ellis et al., 1991, Crowston, 1994].

2.4 Spectrum of Coordination Mechanisms

Traditionally coordination of teamwork is based on a dichotomy between two coordination mechanisms, structured and unstructured [Rico et al., 2008, Espinosa et al., 2002, Schmidt and Simonee, 1996] to manage their multiple interdependencies [Malone and Crowston, 1994, Espinosa et al., 2002]. As indicated above, structured coordination mechanisms refer to pre-defined organizational constructs (e.g., processes, formal structures, methods and plans) that determine actions for teamwork, based on standard solutions for recurring problems [Malone and Crowston, 1990]. On the other hand, unstructured coordination mechanisms refer to coordination mechanisms used to mitigate unpredicted situations spontaneously, in which pre-defined constructs do not describe adequately or entirely plans for action [Selznick, 1948].

Although they are traditionally referenced as the two standard and opposite mechanisms [Espinosa et al., 2002] [Rico et al., 2008], new teamwork situations may require the creation of new coordination mechanisms that are a mix of structured and unstructured coordination mechanisms.

For example, in the context of collaborative work, a new model of collaboration technology proposes to extend the traditional time-space matrix to accommodate the variations between the time axis or space axis[Lee and Paine, 2015]. The original time-space matrix describes a dichotomy in the time axis (synchronous and asynchronous support) and in the space axis (co-located and remote meetings) [Johansen, 1988]. However, with the advancement of information technology, such described differences have become too simplistic [Dix et al., 2003]. Collaboration technology can now, for example, be classified as concurrent synchronized (e.g. video-conference applications), serial (e.g. argumentation applications that record arguments for design decisions), mixed (e.g. co-authoring applications), or unsynchronized (e.g. e-mail exchange). A spectrum of coordination mechanisms is required to denote the numerous and ever-changing possibilities in-between.

Different coordination mechanisms are currently being used in practice. For example, in the domain of collaborative content production, Wikipedia supports three different coordination mechanisms, *i)* to write articles collaboratively, *ii)* through direct communication among collaborators involved in the article, *iii)* through the group structure of collaborators and through specific policies and procedures [Kittur and Kraut, 2010]. Collaborators discuss issues about an article and develop a common strategy to write it together, through direct communication. Communication is used as a coordination mechanism based on mutual adjustment for uncertain situations in which no process is pre-defined. In this thesis, this type of coordination mechanism is classified as an unstructured coordination mechanism, as it relies on a spontaneous definition of a coordination plan, based on communication, to write an article. Group structure in contrast consists of role differentiation, division of labour and formal and informal management. In this hierarchical mechanism, a core team sets the directions of work and collaborators develop a common understanding to turn them into actions. This mechanism is classified as a structured coordination mechanism, as it relies on a hierarchical structure that organizes teamwork. A third coordination method involves the development and use of policies and procedures. Wikipedia has developed various policies to govern areas such as becoming an administrator, the requirements for an encyclopedic article, and the definition of plagiarism. The use of policies and procedures, in this case, represents a combination of communication and structures that allows collaborators to coordinate task assignment.

In the domain of global software development, there are two coordination mechanisms used by teams of software developers: formal and informal [Redmiles et al., 2007]. Formal mechanisms rely on pre-defined processes that define separation of work in multiple and independent tasks, resynchronized periodically. In this thesis, this coordination mechanism is clas-

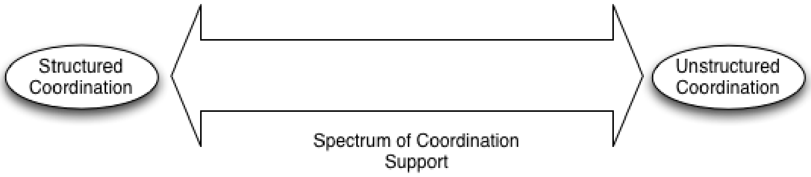


Figure 2.2: Spectrum of coordination support for teamwork.

sified as a structured coordination mechanism, as it relies on a process to coordinate teamwork towards software development. Informal approaches rely on awareness information that allows for the understanding of activities performed by developers. Informal mechanisms rely on the notion of awareness, an informal understanding of the activities of others that provides a context for monitoring and assessing group and individual activity. In this thesis, this type of coordination mechanism is classified as a unstructured coordination mechanism, as it also relies on workspace awareness to define coordination strategies spontaneously to guide teamwork in software development.

In the domain of supply chains, a similar discussion involves the dichotomy between types of processes used in teams, e.g. fixed processes or dynamic processes. [Bernstein, 2000]. There are four types of coordination mechanisms implemented in the system, two representing the extremes of a process specificity, "fixed processes" (e.g. the shipment process of a computer to a customer) and "dynamic processes" (e.g. a shared to-do list software and documents that support users building a process), and two representing a combination of the extremes, "monitoring constraints" (e.g. the definition of events in a shared to-do list software that notifies users when their occurrence is imminent) and "planning options based on constraints" (e.g. a system proposes different approaches to ship a computer to a customer) [Bernstein, 2000].

There is a natural tendency in different domains to implement and provide the support of various coordination mechanisms, such as described in the previous examples. Some systems implement two coordination mechanisms, providing two options of opposing mechanisms [Redmiles et al., 2007]. Depending on the requirements of a situation, other systems extend the concept of a dichotomy and provide more than two options of mechanisms [Kittur and Kraut, 2010, Bernstein, 2000].

As coordination mechanisms are constantly evolving to support the needs of a team in teamwork [Kittur and Kraut, 2010], either because of the evolution of CSCW technology [Ellis et al., 1991] or different team dynamics [Sarma et al., 2010], systems need a common conceptual framework as reference for the definition and implementation of emerging coordination mechanisms [Bernstein, 2000].

Spectra are common abstractions used in different approaches or systems to illustrate the theoretical infinite set of characteristics [Sarma et al., 2010, Lee and Paine, 2015, Geyer et al., 2006, Ellis et al., 1991]. In the case of teamwork coordination, a spectrum of coordination mechanisms safeguards the natural expansion and extensibility of systems that implement new, evolving mechanisms. In the beginning, Wikipedia, started with coordination mechanisms based on communication and over time created hierarchical structures and policies and procedures to coordinate teamwork of collaborators [Kittur and Kraut, 2010]. The spectrum, however, needs to describe and characterize properties of the extremes, to set the implementation boundaries that reflect its coordination mechanisms. In the example of process specificity for supply chains, one extreme defines the use of a process that specifies activities to achieve the goals of a task [Bernstein, 2000]. On the other extreme, teams use a conceptual to-do list to sketch the activities spontaneously to achieve their goals.

The use of distinct coordination mechanisms aims to support a team with different coordination approaches to achieve the goals of a task [Janeiro et al., 2012c]. Structured coordination mechanisms use processes to coordinate teamwork, as they assume that processes are well-defined structures that approach repetitive and predictable tasks [VanGundy, 1988]. Processes represent the description of previous experiences to achieve the goals of a task. In risk identification scenarios, professional facilitators define facilitation processes for teams based on the experience in risk identification tasks and facilitation of teams [Van Grinsven and de Vreede, 2003]. Unstructured coordination mechanisms provide the exchange of awareness information for teams to enable them to create a coordination strategy dynamically, providing circumstantial information to approach non-repetitive and unpredictable tasks [VanGundy, 1988, Suchman, 1987]. Workspace awareness information is the circumstantial resource that enables teams to define their own coordination strategy.

The integration of both coordination mechanisms is necessary to support coordination in different types of tasks, as it is not possible to foresee their type, repetitive and predictable or non-repetitive and unpredictable, and the appropriate coordination mechanism. Therefore, this thesis integrates structured and unstructured coordination mechanisms in a spectrum of coordination mechanisms for teamwork, as illustrated in Figure 2.2. As each coordination mechanism may be more appropriate for a particular task than others, a team is expected to evaluate a task and to choose the appropriate coordination mechanism.

Considering the circumstances of teamwork, the spectrum proposed in this thesis embraces structured and unstructured coordination mechanisms, such as defined in previous sections of this chapter. The structured coordination mechanisms do not only describe a collaboration process to coordinate team members, but also to coordinate and to guide them to achieve their goals through facilitation techniques. The unstructured coordination mechanisms, on the other hand, support the acquisition and provisioning of workspace awareness information, enabling team members to better understand teamwork circumstances to define a coordination strategy spontaneously.

Through the spectrum of coordination support, new coordination mechanisms may also be developed to support teamwork, based on the characteristics of structured and unstructured coordination mechanisms. For example, a recommender coordination mechanism could suggest a process to teams (such as in the prescribed mechanism) and could use workspace awareness information (such as in the ad-hoc mechanism) to provide coordination recommendations. Such a mechanism could analyse the interaction of teams during a diagnosis task, compute coordination recommendations and propose them to teams while they go through a process, e.g. remind team members to switch to following activities after long periods of discussions or recommend an equally balanced distribution of subtasks based on previous activities.

Although the spectrum of coordination supports the possibility to define different coordination mechanisms, this thesis does not describe a criterion for teams to select a mechanism for a task. The selection of a mechanism is delegated to a team working on a task and it is expected that the team reaches consensus on the chosen mechanisms. However, there are important factors that contribute to the selection of a coordination mechanisms, such as: the size of the team and the expertise of team participants. For example, it might be challenging for large teams to agree on a coordination mechanism to be used during diagnosis, as team participants might have conflicting preferences [Wulf, 1995]. Similarly, a team of experts in diagnosis, for example, might not consider the use of a process-based coordination mechanism, as their knowledge transcends reliance on rules and guidelines [Dreyfus and Dreyfus, 1980].

ID	Description
R2.1	A shared workspace system must offer different coordination mechanisms to participants.
R2.2	A shared workspace system implementing structured coordination mechanisms must manage collaboration process descriptions.
R2.3	A shared workspace system that implements structured coordination mechanisms must interpret collaboration process descriptions.
R2.4	A shared workspace system that implements unstructured coordination mechanisms must provide team members with workspace awareness information to support dynamic coordination.
R2.5	A shared workspace system must be extensible to include new, emerging coordination mechanisms.

Table 2.3: Summary of requirements to support coordination in teamwork.

2.5 System Requirements to Support Coordination of Teamwork

The spectrum of coordination mechanisms imposes certain requirements for a shared workspace system to implement coordination mechanisms. Such requirement are summarized in Table 2.3.

A shared workspace system must implement all functionalities needed and different possibilities to structure coordination. A shared workspace system has to implement the two coordination mechanisms, at least, to provide a minimum set of options to its participants (**R2.1**).

A shared workspace system that implements structured coordination mechanisms, as defined in this section, must manage the serialization of collaboration process descriptions (**R2.2**). A system has to provide to participants the functionalities to create, update, retrieve and delete serialized collaboration process descriptions.

In addition to the aforementioned operations, the shared workspace system must also provide the functionality needed to interpret these processes (**R2.3**). The interpretation of the processes coordinates teamwork towards the accomplishment of the goals of a task but also configures the software components for them.

The shared workspace system that implements unstructured coordination mechanisms, as defined in this section, must provide means to capture, distribute and convey workspace awareness information to team members about their actions with the system (**R2.4**). In a shared workspace system, workspace awareness is the information necessary for team members to understand teamwork circumstances, e.g. past and present actions, next activities and task goals

Although the spectrum defines infinite theoretical options of coordination mechanisms, it is not possible to predict all possible coordination mechanisms to implement. Rather, the shared workspace system must be extensible to include possible emerging coordination mechanisms with particular properties that range between both extremes of the spectrum (**R2.5**).

Chapter 3

Diagnosis as a Data-Centric Engineering Task

Manufacturers in the industrial engineering domain focus their efforts on the development of methodologies and technologies to improve the availability and reliability of industrial machines. One approach to improve machine availability is to reduce the downtime of a machine by implementing an immediate response strategy to machine failure.

Currently, efforts are particularly concentrated on the development of new infrastructure and services that collect data generated from machines to monitor current equipment's conditions and usage limitations in real time [Iung et al., 2009][Karlsson et al., 2012].

E-maintenance is a methodology that tackles the fundamental need to monitor equipment usage and degradation to enable a team of experts to intervene and avoid unscheduled downtime, unexpected breakdown or provide a quick fix when necessary [Holmberg et al., 2010]. For example, a machine that issues possible failure signals, triggered by telemetric data, e.g. that its engine is operating over a certain limit, requires experts to be brought together to analyse the problem and take decisions to avoid an abrupt breakdown.

The e-maintenance methodology is based on a generic operational process and covers the assessment of equipment conditions, diagnosis and finally a forecast of future occurrence [Levrat et al., 2008]. [Levrat et al., 2008] distinguishes four sequential phases that run continuously as illustrated in 3.1 : *i) to measure*, *ii) to monitor*, *iii) to diagnose* and *iv) to provide prognosis*.

The goal of the first phase (*to measure*), is to gather sampling signals that indicate real-world physical conditions of machines and to convert the

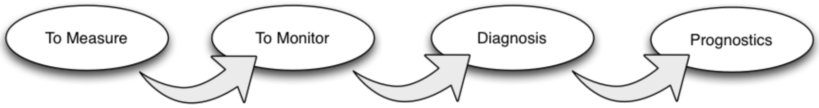


Figure 3.1: A generic machine maintenance process [Levrat et al., 2008].

resulting samples into digital numeric values that can be processed automatically [Heidbrink et al., 2011]. Experts define a set of variables, values and units of measurement that support the definition of machine conditions and install sensors that quantify such measurements, e.g. temperature sensors that quantify oil temperature in Celsius. Data acquisition systems typically convert analog waveforms into digital values for processing.

The goal of the second phase (*to monitor*), is to trace the condition of machines, based on data acquired by sensors [Heidbrink et al., 2011]. Currently, in the industrial engineering domain, machines are deployed with several sensors that generate machine performance data continuously. Such data that grows rapidly over time is referred to as data streams. Data streams are becoming commonly used in the context of diagnosis to enable diagnosis experts to better understand machine's performance and conditions in real-time [Bauleo et al., 2014][Karlsson et al., 2012][Holmberg et al., 2010].

Machine data streams are constantly sent to Data Streams Management Systems (DSMS), to process, monitor, verify and validate machine generated data [Zeitler and Risch, 2011][Risch et al., 2003]. The throughput of generated data streams is becoming so high that it is becoming a challenge for diagnosis experts to constantly analyze data streams continuously. Therefore, DSMSs often implement the continuous query mechanism that enables the data analysis work to be configured once and executed automatically. A continuous query is a query that is issued once and then logically runs continuously over the DSMS until its conditions are satisfied, in contrast to traditional one-time queries which run once to completion over the current data sets [Terry et al., 1992].

In the diagnosis scenario, once a continuous query achieves a threshold, it triggers a warning message directly to diagnosis experts, who can reason about the data and deviations to formulate a machine diagnosis. For example, given a power consumption model that computes the theoretical expected power consumption of a machine at any point in time, a continuous query returns a sensor identification whenever the difference between the actual measured power consumption and the expected power is greater than 100 Watts during one second.

When the data monitored by the query signals an anomaly, experts begin the *diagnosis* phase to further investigate the root-cause [Heidbrink et al., 2011][Johanson et al., 2014]. Continuous queries represent an important source of data for this purpose, as they provide contextualized indications of the deviations among all data collected for the observed machine. However, the challenge to reason over the data, establish co-relations among data streams generated by sensors to identify the location of the problem, and discuss the recognition of root-causes of machine failures remains.

Finally, after evaluating the results of a diagnosis, experts proceed to the *prognosis* phase to forecast possible machine problems, based on failure patterns identified during the diagnosis phase [Heidbrink et al., 2011].

Although all maintenance process phases are important to keep high machine availability and reliability, this thesis focus at the specifics of the diagnosis phase and its related activities involving diagnosis experts, as this phase requires most collaboration between experts to understand machine anomalies [Levrat et al., 2008].

3.1 Diagnosis Team

Advances in telecommunication and information technologies represent alternatives to overcome the challenge of physical separation. They enable distributed teams to diagnose machines remotely and facilitate collaboration when geographic distance is an issue [Garcia et al., 2004][Janeiro et al., 2012b]. In the context of this thesis, a diagnosis team is a team in which experts are distant from each another, e.g. different cities, countries and continents. The team consists of different diagnosis experts (e.g. mechanical, electrical and hydraulic engineers) and a local stakeholder, who is physically close to the observed machine and can interact with it. Most often, the local stakeholder is the operator of a machine.

Figure 3.2 illustrates the setup of a team diagnosing a machine, a wheel loader located in a remote location, e.g. Singapore [Johanson et al., 2014]. The machine sends telemetry data constantly to a data stream management systems which, in turn, processes the streams and issues warning signals upon the occurrence of a failure. The warning is issued to a diagnosis expert who becomes responsible for the diagnosis of a machine, the problem owner. The problem owner, remotely located in Sweden, begins with a diagnosis task reasoning about the telemetry data received by a particular machine

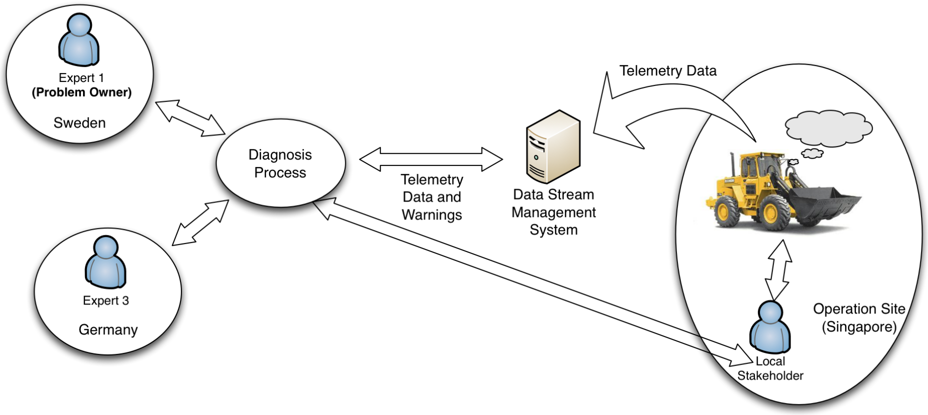


Figure 3.2: A scenario to illustrate the participation of experts in a diagnosis distributed team [Heidbrink et al., 2011].

and the information contained in the warning signal. The problem owner can interact with the local stakeholder, who is at the operation site of the machine, and with other experts who have the proper skills to support the diagnosis task and are possibly in other distant locations (e.g. Germany).

The situation described in the Figure 3.2 represents a typical remote diagnosis reference scenario used in this thesis and in the Smart Vortex project [Consortium, 2010]. However, this thesis does not focus on the influence of participants playing certain roles, e.g. problem owner or other leading roles.

3.2 Problem-Solving Methods for Diagnosis

An understanding of the process of diagnosis is needed to be able to structure coordination between experts. This section describes models (problem-solving methods and generic tasks models) based on extensive analysis of diagnostic process in practice.

In the context of this thesis the problem represents a machine failure, the solution represents the identification of the root causes of a problem, the actions to fix the machine failure and the definition of an solution implementation strategy. A team of experts starts with the diagnosis reasoning once they are aware that a machine failure has occurred and have its telemetry data available.

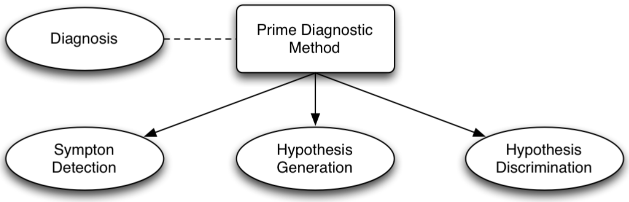


Figure 3.3: The diagnosis task decomposition by the prime diagnostic method [Benjamins, 1993].

3.2.1 Prime Diagnostic Method

The prime diagnostic method [Benjamins, 1993] is a general method to perform diagnosis. It distinguishes three subtasks, symptom detection, hypothesis generation and hypothesis discrimination, as illustrated in Figure 3.3.

The prime diagnostic method reasons over a flat device model or hierarchical device model. A flat device model represents the details of a device's abstraction level, without considering more general or specific parts of the device. In the case of hierarchical diagnosis, the method is recursively applied to different abstraction levels of a device model. Whenever a diagnosis solution is identified for a level but it is not detailed enough, the method is repeated for another device context. Ultimately, a hierarchical device model represents a set of different models of the same device that vary with regard to their level of detail.

The first task, symptom detection, checks whether the initial observations of a device, received as input to a diagnosis method, represent anomalies. The confirmed anomalies of a device are named symptoms and they lead a device to an abnormal behaviour. All other observations that do not represent an anomaly are called normality observations. Note that not every deviant observation is an anomaly. For example, two waveform signals may represent the same amount of energy, although their wave signatures are different. Therefore, the waveforms differ but they do not represent an anomaly with regard to the amount of energy.

The second task, hypothesis generation, generates a set of hypotheses based on the initial normal and abnormal observations of a device. This task can be performed in a simple form, in which an expert generates as many hypotheses as necessary without the support of the device's documentation, just based on experience and previous knowledge of the expert. This task can also be performed in a more refined form, in which an expert generates

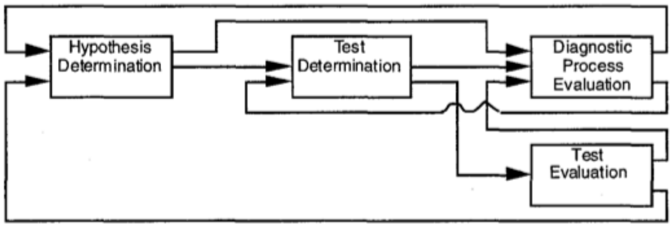


Figure 3.4: SIX: a Generic Task Model for Diagnostic Reasoning [Brazier and Treur, 1994].

hypothesis based on available knowledge of the device. Such knowledge is of two types: static or dynamic. Static knowledge represents knowledge about the causality or connectivity between two parts of the device that are documented beforehand. Dynamic knowledge represents the knowledge that is deduced through an inference process, e.g. fault simulation of hypothesized device parts.

The third task, hypothesis discrimination, aims to reduce the set of hypothesis generated in the hypothesis generation task. Additional observations are obtained from the device and associated to individual hypotheses in the hypothesis set generated. Experts evaluate these hypotheses by associating additional observations or may use a more refined strategy that uses available knowledge, e.g. test costs and probability of test parts. Finally, hypotheses that are inconsistent with regard to the additional observations are excluded from the set and from further consideration.

3.2.2 A Generic Task Model for Diagnostic Reasoning

The generic task model for diagnosis, named SIX, proposes a structured form for diagnostic reasoning based on the notion of shared task models [Brazier et al., 1996][Brazier and Treur, 1994]. Figure 3.4 illustrates the SIX model and its subprocesses and their relationships.

The model consists of four generic phases: *i) hypothesis determination*, *ii) test determination*, *iii) test evaluation* and *iv) diagnostic process evaluation.* In *hypothesis determination* phase, stakeholders reason about the appropriateness of possible hypotheses related to a given state of the diagnostic process and determine which hypotheses should be further investigated. In the *test determination* phase, stakeholders analyze the current state of the diagnostic process and determine a test that is the most appropriate. In *test evaluation* phase, stakeholders perform tests and determine the relation

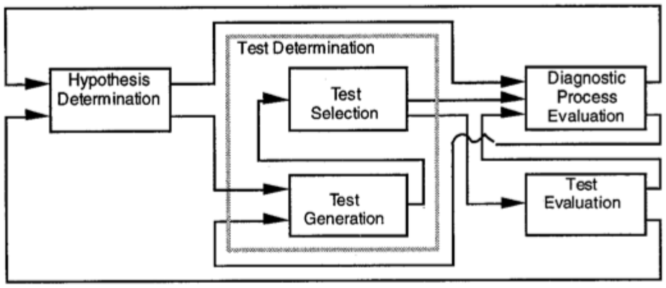


Figure 3.5: A shared task model for diagnostic reasoning based on causal knowledge [Brazier and Treur, 1994].

between the test results and the current hypotheses. Experts reason about the information they expect to acquire and additional investigations that should be performed. Finally, in the *diagnostic process evaluation* phase, stakeholders analyze the implications of the test results for the hypotheses and determine which of them are confirmed and which are rejected. On the basis of an analysis of the state of the diagnostic process, experts decide whether to conclude the diagnostic process. If the diagnostic process is continued, they should start again with determination of hypotheses and tests.

Diagnostic reasoning processes are based on *causal* or *anti-causal* domain knowledge. In the first case, causal domain knowledge derivations about the domain follow the direction of causality: the predicted observable consequences are derived from hypotheses (possible causes), after which some of the predicted observations are verified. This type of reasoning requires causal knowledge to specify the form in which causal consequences of hypotheses are derived. In the second case, domain knowledge is used to derive hypotheses from information on observable symptoms. The direction of derivation is against the direction of causality: it proceeds from observable findings to the causes. This type of reasoning requires causal knowledge to specify the form in which hypotheses are derived from observable findings, which represents anti-causal knowledge.

In both cases, strategic reasoning is required to determine the appropriate hypotheses on which to focus and the appropriate tests that need to be performed, as described in the SIX model. However, depending on the reasoning process, it is necessary to use either the causal of anti-causal reasoning process. Therefore, the SIX generic task model was specialized in two slightly different variations representing causal domain knowledge and

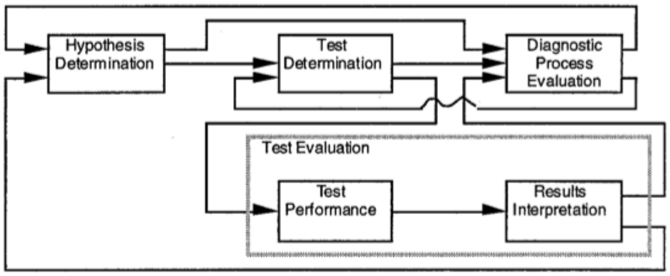


Figure 3.6: A shared task model for diagnostic reasoning based on anti-causal knowledge [Brazier and Treur, 1994].

anti-causal domain knowledge. These variations as illustrated by the figures below.

The specialization for diagnostic reasoning based on causal domain knowledge is obtained by decomposing the test determination task into two subtasks: test generation and test selection, as illustrated in Figure 3.5.

Test generation receives as an input the hypothesis and derives observable causal consequences using causal domain knowledge. These consequences are predictions of the findings that should be observed if the hypothesis is satisfied. The predicted findings, influenced by the assumed hypothesis, become criteria for experts to select tests. Finally, the experts involved in test selection subtask use the criteria and select appropriate tests.

The specialization for diagnostic reasoning based on anti-causal domain knowledge is obtained by decomposing the test evaluation task into two subtasks: test performance and results interpretation, as illustrated in Figure 3.6.

Test performance represents a subtask in which experts are responsible for the execution of the tests selected by test determination. The results of the test may be acquired directly by an expert participant, or automatically by other systems. No further reasoning about the domain is performed in this task. The acquired test information is used to interpret results and described conclusions about the hypotheses.

Although SIX offers two variations for diagnosis, this thesis focuses only on the second variation, anti-causal knowledge, because it reflects the diagnosis approach used in the companies described in the use experiments of this thesis [Heidbrink et al., 2011]. In the case of diagnosis of machinery, as soon as an observable symptom is detected, experts use their domain knowledge to derive hypotheses of machine failures and associated actions to test the hypotheses.

3.2.3 Limitations

The goal of methods for diagnosis in general, and for both of the presented methods in particular, is to identify the causes for an observed symptom. However, the identification of the causes is only part of the strategy to solve machine failures. The definition of the causes (problems) represents only the beginning of a problem-solving process, according to a general problem-solving model [VanGundy, 1988]. Subsequently, experts still need to define actions (solutions), based on information about observed symptoms and their experience, and then implement the actions to verify their effectiveness. Therefore, the scope of the presented methods is limited, given that they do not support activities for definitions of actions and their implementation, nor do they focus on collaboration between team members.

3.3 Rectio: A Team-based Method for Diagnosis

A new method is necessary to support diagnosis processes and support collaboration aspects among experts along all activities [Heidbrink et al., 2013].

This section introduces a stepwise method for solving diagnosis problems, including the planning phase, called *Rectio*. The method is represented as a process, described in Figure 3.7, using the graphical representation of activity diagrams of the Unified Modeling Language (UML)[1].

This method is based on fundamental diagnosis activities, such as formulation of hypothesis, and activities that enable the generation of solutions for failures, identified in diagnosis, and definition of implementation strategies. It distinguishes four general phases: *Data Analysis*, *Hypothesis Definition*, *Actions Generation* and *Solution Evaluation*.

The *Data Analysis* phase begins when the monitored device triggers a failure or an alert event, together with performance data. Team members start acquiring available data to identify possible failures that are represented by sensor readings [Heidbrink, 2013][Leva et al., 2014]. This phase has two activities: acquire data and analyse data. In the first activity (*Acquire Data*), team participants explore available data issued with the failure or alert signals. This is a first attempt to identify whether the available data suffices to diagnose a failure. If not, participants need to acquire additional

[1] Unified Modeling Language (UML) specification, see www.uml.org for details

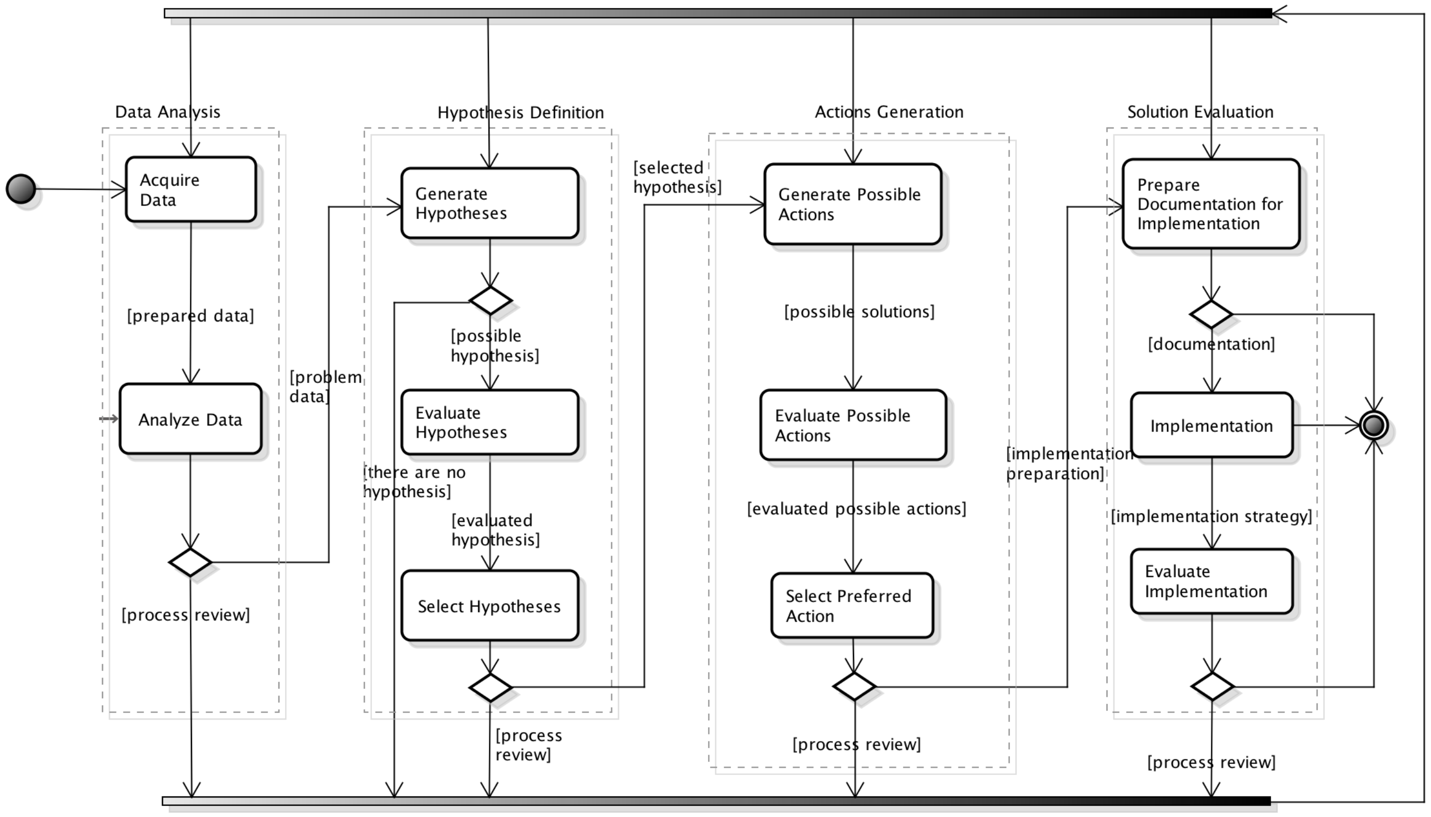


Figure 3.7: Description of the Rectio method for diagnosis.

data, available locally on the device or remotely, via a data stream mechanism. Once all necessary data is acquired and prepared for analysis, the *Analyze Data* activity starts. In this phase, team participants examine acquired data with the purpose of deriving conclusions about it, so that they can start to speculate about causes of possible failures. If there is not enough data, team participants need to review the process and possibly restart the data analysis phase.

The *Hypothesis Definition* phase supports team participants to generate hypothesis of failure causes of a observed machine, based on prepared data [Johanson et al., 2013][Heidbrink et al., 2013]. This phase has three activities: *Generate Hypothesis*, *Evaluate Hypothesis* and *Select Hypothesis.* In the first activity (*Generate Hypothesis*), team participants generate hypotheses that represent possible causes for machine failures based on the results of data analysis. If team participants are not able to generate a hypothesis, they return to the data analysis phase to acquire more data and investigate potential problems.

In the next activity, *Evaluate Hypothesis*, team participants evaluate and compare hypotheses. In addition to acquired data from machines, participants also use other sources of information, e.g. design drawings, manuals or results of simulation models, to evaluate the validity of hypotheses in a particular situation. In this activity, team participants need to determine tests that can further corroborate the validity of one or more hypotheses. Once tests for these hypotheses are determined, team participants need to interact with the machine again to execute tests and therefore confirm or discard hypotheses.

Finally, in the *Select Hypothesis* activity, participants review the set of evaluated hypotheses and the tests used to evaluate them. Based on such information, team participants select hypotheses that, most likely, represent the cause of a failure and discard the hypotheses that are not valid in the particular situation or might not represent the root cause of a failure but secondary causes. Once the team selects a set of hypotheses, they can choose to start the *Actions Generation* phase or review the process, either to acquire more machine data or to generate more hypotheses.

In the *Actions Generation* phase, team participants generate possible corrective actions for each hypothesis [Johanson et al., 2013][Heidbrink et al., 2013]. These actions represent attempts from the participants of the team to solve the causes of failures, described as hypothesis. Also, as described in the hypothesis definition phase, this phase has three activities: *Generate Possible Actions*, *Evaluate Possible Actions* and *Select Preferred Actions.* In the first activity (*Generate Possible Actions*), team participants use selected hypotheses that represent machine failures to generate possible actions that can solve a failure, according to their experience. Actions may vary from simple, e.g. train the operator of a machine on the proper execute a gear shift, to more complex, e.g. replace a gearbox.

Once team participants are satisfied with the set of possible actions, they evaluate each of them to identify their suitability for implementation, in the activity *Evaluate Possible Actions*. Not all actions may be equally appropriate, for example, fixing a degrading gearbox may require a machine to be out of operation for several days. However, if the machine operates a critical service that cannot be stopped, an alternative action for this problem maybe to train its operator on a careful usage of the gearbox to reduce the degradation problem, until the machine can be scheduled for maintenance.

Finally, in the last activity (*Select Preferred Action*), team participants need to choose, among all evaluated actions, the most appropriate action to implement, taking into consideration the constraints of the situation. After this activity, team participants can also review or further analyse acquired data to reinforce the decision about the pursued actions. In addition, they can review the generated hypothesis in relation to the selected action.

The last phase in the method is the *Solution Evaluation* phase. In this phase team participants define an implementation strategy for the selected actions to solve a machine failure [Johanson et al., 2013][Heidbrink et al., 2013]. This phase has three activities: *Prepare Documentation for Implementation*, *Implementation* and *Evaluate Implementation*. In the first activity (*Prepare Documentation for Implementation*) team participants prepare the documentation for operational teams to implement actions that solve machine failures. In this activity, participants use all data that is generated, such as acquired data and hypothesis, to describe the context of a machine failure and then describe actions related to the hypothesis to fix failures.

After the implementation documentation is prepared, the second activity in this phase is *Implementation*. In this activity, an operational team starts to use the implementation documentation to execute the actions recommended by the documentation. The operational team of technicians is often close to a machine physically as some of the actions may require experts to handle the machine, e.g. lubrication or replacement of mechanical components.

Once the operational team executed the actions, they report in the documentation their actions and their effectiveness. Such information is finally used in the activity *Evaluate Implementation* to describe successful and unsuccessful actions used to fix machine failures [Johanson et al., 2013][Heidbrink et al., 2013]. It represents valuable lessons learned that can be used as hints of solutions in future diagnosis cases. After this activity, team participants can finish diagnosis or can review any particular activity of the process.

3.4 System Requirements for Diagnosis Tasks

Although the mere retrieval of telemetry data is a basic resource to enable remote diagnosis, it does not offer additional means that facilitate the work of teams during machine diagnosis tasks, e.g. data visualization or information sharing. The acquisition of telemetry data is a pre-requirement for remote diagnosis tasks but transformation of data, visualization of data and collaboration around the data are some examples of added-value services that a shared workspace system could offer for experts during diagnosis.

During the analysis of machine telemetry data, experts often need to evaluate the behaviour of data over time to assess whether machine components are either functioning normally (expected behaviour) or abnormally [Johanson et al., 2013][Heidbrink et al., 2013]. In engineering applications, experts often visualize such data using line charts, to depict an overview of data trends over the period of time in which it was collected. The use of such charts enables experts, for example, to assess whether machine components degrade over time or show the beginning of a degradation. Therefore, experts require a means to visualize telemetry data as line charts (**R3.1**).

The degradation of machines is, in some situations, due to flaws in the design of certain components. In extreme circumstances, experts identify the limits of a component and have to remanufacture it, restarting the development life-cycle of a component [Johanson et al., 2013][Heidbrink et al., 2013]. Experts often use CAD models to design components and visualize their properties in an iteration process to correct design flaws [Johanson et al., 2013][Heidbrink et al., 2013]. They can modify the physical properties of these models and visualize them immediately. Visualization of these models is an important functionality for experts because it helps them to identify characteristics of components that might influence a certain degradation. Therefore, experts need means in diagnosis tasks to visualize CAD models of the components used in a certain machine (**R3.1**).

During informal discussions, diagnosis experts revealed that certain situations only require visualization of faulty mechanical components or boards of integrated circuits to identify the root-cause of a machine anomaly, instead of the analysis of telemetry data series [Johanson et al., 2013][Heidbrink et al., 2013]. A common practice, mentioned by industrial partners in the context of the Smart Vortex project (see section 1.2.1), is to take pictures of certain parts of a machine and to share them with experts that have complementary diagnosis skills to identify failures. Therefore, experts need means to share these pictures, such as photos of machine components, with each other to formulate a machine diagnosis (**R3.1**).

During diagnosis, machines that are composed of several specialized subsystems require the involvement of experts with knowledge about these respective subsystems [Arnaiz et al., 2010][Bloch and Geitner, 1997]. As described by Rectio, the collaboration of experts is important to define hypotheses about machine anomalies and propose actions. Such collaboration involves not only discussions about machine anomalies but also documentation of rationale exchanged among experts during discussions about topics that may cause machine anomalies and that may overcome them. The documentation of rationale is important as a reference in later phases that follow diagnosis, e.g. maintenance, to implement proposed actions that overcome the anomalies of a diagnosed machine. Maintenance teams, for example, may profit from a diagnosis documentation to implement the proposed actions and evaluate their effectiveness. Experts need a means to document the rationale they share during diagnosis tasks. Therefore the shared workspace system should provide a discussion space through which experts describe, document, share, change and share knowledge [Johanson et al., 2014][Knoll et al., 2014]. In the context of this thesis, the aforementioned knowledge is considered a design rationale for machine diagnosis [Moran and Carroll, 1996].

The patterns of collaboration [Briggs et al., 2003b], described in Chapter 2, indicate general requirements that a diagnosis shared workspace system should provide to support collaboration among teams of experts in diagnosis tasks. In the context of system requirements, the concepts often used in association with the patterns of collaboration are considered diagnosis formulations.

The first two patterns, generate and reduce, together indicate basic operations with contributions. These patterns support team participants to create and delete contributions, respectively [Johanson et al., 2013][Heidbrink et al., 2013]. For example, a team of experts use the support offered by this pattern in the activity "Generate Hypothesis" of Rectio to collaboratively generate a set of hypothesis that represent machine failures. In addition to create and delete operations, team participants also update the contents of a contribution and read contributions that are saved in a storage system. Therefore, as a requirement (**R3.2**), the diagnosis shared workspace system should implement for each diagnosis formulation, described as contribution, a set of four operations, known as CRUD operations (create, read, update and delete) [Martin, 1983].

The clarify pattern describes that team participants should be able to further explain the generated contributions. A team of experts use the support offered by this pattern in the activity "Generate Possible Actions" of Rectio to elucidate a hypothesis that is not well understood by other participants of the team [Johanson et al., 2013][Heidbrink et al., 2013]. For example, an expert who contributes a hypothesis needs to describe the data source that justifies it. One straightforward possibility is to request the creator of the contribution to explain with other terms and words its contents to other team participants that share the same software environment. However, in the diagnosis scenario, contributions are further clarified when associated to data series that explain a certain machine behaviour, CAD drawings or pictures of machine components to illustrate the layout indicating a failure. A diagnosis shared workspace system should implement mechanisms that enables team participants to create associations between contributions and data series, drawings and images, that allow the clarification of contributions. Therefore, the system should enable experts to attach data series, drawings and images to textual diagnosis formulations to enrich its descriptions, as in an analogy to e-mails [Bellotti et al., 2003].

The evaluate pattern describes that team participants are able to evaluate a set of contributions and rank them according to a level of importance [Johanson et al., 2013][Heidbrink et al., 2013]. A natural mechanism to evaluate contributions is to quantify their level of importance, such as a rating, and assign to each contribution a rating. Subsequently, team participants evaluate such a set of contributions by ordering them according to the rating in ascending order (less important first) or descending order (more important first). A team of experts use the support offered by this pattern in the activity "Evaluate Possible Actions" of Rectio to evaluate a set of actions to fix a machine failure. For example, they assign priority ratings to each action and order the actions in a list according to their criticality, most critical actions first, followed by less critical. Therefore, the diagnosis shared workspace system needs to implement a mechanism that enables team participants to associate ratings to contributions and rank them. A shared workspace system should, for example, enable experts to rate diagnosis formulations, showing the average of rates in a total of a five-star symbol, such as in Google Shopping [2].

[2] https://support.google.com/merchants/answer/6059553. Last Access in 03.September.2016

The organize pattern describes that team participants are able to organize contributions in categories [Johanson et al., 2013][Heidbrink et al., 2013]. Categories help team participants to contextualize contributions with regard to a discussion topic. Categories are also relevant to cluster a high number of contributions to a topic, minimizing the amount of cognitive load involved by participants of the software system to find content[3]. Team participants should be able to define categories and associate contributions to categories. A team of experts use the support offered by this pattern in the activity "Select Hypothesis" of Rectio to prepare the several hypotheses for selection by grouping and classifying them. For example, a team of experts creates one category only for failures related to the machine's driving engine subsystem and associate generated hypotheses to this category. Therefore, the diagnosis shared workspace system needs to implement a mechanism that enables participants to organise diagnosis formulations into different categories, by association and disassociation, such as through a drag-and-drop technique.

During informal requirement gathering workshops for data-centric machine diagnosis, industrial partners revealed that they often use different sources of machine telemetry data to formulate diagnosis [Johanson et al., 2013][Heidbrink et al., 2013]. In some scenarios, they need an overview of the several data sources to cross-check and identify machine anomalies, e.g. machine telemetry data is cross-checked with CAD models and photos of integrated circuits of a degrading machine [Johanson et al., 2013][Heidbrink et al., 2013]. These sources of information are often distributed in different specific-purpose applications, making it inconvenient for experts to switch application contexts to cross-reference data [Johanson et al., 2014]. Ideally, such applications should be in the same workspace to provide an overview of the several data sources related to a machine. Therefore, a shared workspace system that supports experts in diagnosis tasks should be designed to present information as a dashboard (**R3.3**).

A characteristic of dashboard-based systems is its extensibility. Such systems often separate functionalities in different independent components, each with a specific purpose to ease extensibility [Marcus, 2006]. During the design of a dashboard, not all functionalities can be foreseen in design time, some are identified later during the use of the system in runtime. Therefore, modularizing the functionalities in components eases the extension of dashboard systems to integrate new functionalities. For example, in

[3] `http://www.nngroup.com/articles/minimize-cognitive-load/`. Last Access in 03.September.2016

ID	Description
R3.1	A shared workspace system that supports diagnosis tasks should offer functionalities that enable experts to share, visualize and analyze machine-related data sources.
R3.2	A shared workspace system that supports diagnosis tasks should offer functionalities that enable experts to share, modify and delete diagnosis formulations.
R3.3	A shared workspace system that supports diagnosis tasks should have its functionalities presented in a dashboard.
R3.4	A shared workspace system that supports diagnosis tasks should be extensible.

Table 3.1: Summary of requirements for a diagnosis shared workspace system to support cooperation among team participants.

the context of software development, a dashboard system divides in different components functionalities such as: information about software build results, review modifications of pending tasks or incoming tasks and their respective approval status [Biehl et al., 2007].

In diagnosis tasks, different functionalities are used, such as telemetry data visualization in a line charts, exchange of ideas that contribute to formulate diagnosis and visualization of internal and external parts of machines [Johanson et al., 2014]. However, the constant emergence of new technologies and approaches that facilitate diagnosis, such as augmented reality [Datcu et al., 2014], requires systems to adapt, quickly integrating new features and making them available to diagnosis teams. For example, the use of augmented reality techniques to diagnose machines remotely might require a system to implement new integration components, not foreseen previously, that trigger an augmented reality session, sending pertinent telemetry data to head-mounted displays. Therefore, a dashboard-based shared workspace system that supports diagnosis tasks should also be designed for extensibility, allowing the continuous extension of its functionalities (**R3.4**).

The result of this chapter is a list of requirements for the implementation of a shared workspace system that support cooperation in diagnosis process. Such a list is the result of the discussion of this section and is summarized in Table 3.1.

Chapter 4

Elastic Coordination Support for Diagnosis Teams

The need to assist coordination of distributed teams and formulation of diagnosis, as described in the chapters 2 and 3, motivated the design of a software shared workspace system that supports the coordination of distributed teams involved in diagnosis tasks, named Elgar. The system targets distributed teams that are dispersed in different geographical locations and cannot work together in a co-located space.

Elgar currently implements two coordination mechanisms: ad hoc and prescribed, representing the extremes of the spectrum of coordination. Each of the coordination mechanisms has specific technological requirements with regard to its logic. Therefore, this chapter discusses the architecture of Elgar and its supporting systems. In addition, the chapter discusses the design decisions guided by the defined requirements.

4.1 Diagnosis Sessions

Elgar is currently based on the two coordination mechanisms, ad hoc and prescribed, that represent the extremes of the spectrum of coordination. The decision to provide these two mechanisms conforms with the requirement **R 2.1**, described in Table 2.3, to enable participants of Elgar to have a minimum set of options regarding possible coordination mechanisms to use during the execution of diagnosis tasks.

Figure 4.1 illustrates the ontology designed to describe relationships between coordination mechanisms, specific concepts of diagnosis tasks and stakeholders involved in these tasks for Elgar. The ontology is represen-

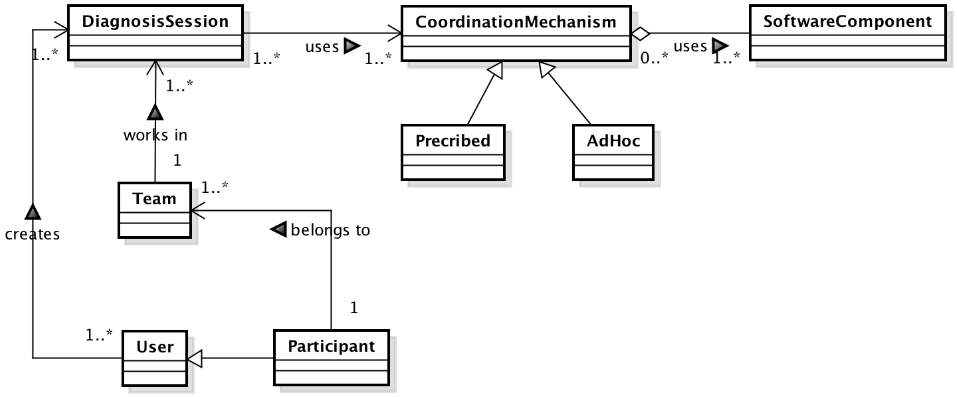


Figure 4.1: The ontology that describes concepts of Elgar in relation to the coordination mechanisms.

ted using the notation of class diagrams of the Unified Modeling Language (UML)[1].

The concept *DiagnosisSession* represents a diagnosis session. The instantiation of a session in Elgar occurs when a machine anomaly is detected by telemetry systems installed in the machines. Elgar supports several diagnosis sessions in parallel. Each instantiation of an Elgar session requires participation of a diagnosis team, represented by the concept *Team*.

A team is composed of several participants, represented by the concept *Participant*. A *Participant* may contribute to several teams. Participants with several skills, required in different sessions, may transit between sessions to provide specific contributions in different diagnosis tasks. Each team allocated to a diagnosis session has specific demands with regard to the required skills. The concept *User* represents a user of Elgar, who can create diagnosis sessions.

For each diagnosis session, there is an associated coordination mechanism, represented by the concept *Coordination Mechanism*. This concept represents the coordination support that diagnosis teams use to execute a diagnosis task. In the context of this thesis, two different types of coordination mechanisms supported by Elgar, the ad-hoc mechanism and the prescribed mechanism, represented by the respective concepts *AdHoc* and *Prescribed*. The ad-hoc mechanism represents the instantiation of a coordination mechanism in the extreme of informal coordination whereas the prescribed coordination mechanism represents a instantiation in the extreme

[1] Unified Modeling Language (UML) specification, see www.uml.org for details

of formal coordination, such as described in the spectrum of coordination. In the current design of Elgar, only one coordination may be defined during creation of a diagnosis session.

Coordination mechanisms, independent of type, use *Software Components* that support teams providing specific diagnosis functionalities and workspace awareness information.

4.2 Elastic Collaboration Components

In Elgar, Software Components used by coordination mechanisms are named Elastic Collaboration Components (ECCs). ECCs represent components of Elgar that offer specific functionalities to participants in the context of diagnosis tasks [Janeiro et al., 2013]. The modularization of functionalities through ECCs is a design decision to satisfy the requirement **R 3.4** of the Table 3.1, for an extensible shared workspace system for diagnosis tasks. The modularization of specific-purpose components facilitates the inclusion of functionalities in Elgar without the need to modify different parts of the system.

ECCs are based on the principle of separation of concerns [Hürsch and Videira Lopes, 1995]. In the context of Elgar, ECCs use the principle of separation of concerns to implement complementary functionalities, based on the model described in Figure 4.2. For example, the same ECC should not implement the functionalities to handle 3D drawings and to handle data series as these functionalities can be isolated and modularized in different components. Instead, two different ECCs are designed and implemented, one for each specific functionality.

All coordination mechanisms use these ECCs, as they define atomic parts of Elgar for specific diagnosis functionalities. In the current design of Elgar both coordination mechanisms (ad hoc and prescribed) make use of ECCs: the ad-hoc coordination mechanism supports the use of several ECCs in a session, selected by team members, whereas the prescribed coordination mechanism uses ECCs prescribed by the collaboration process.

4.3 Support for Diagnosis Functionalities

The ECCs designed for Elgar provide functionalities that support teams in data analysis and diagnosis formulation, such as described respectively by requirements **R 3.1** and **R 3.2** in Table 3.1. The method *Rectio* introduced previously in chapter 3 describes four general phases as guidelines for

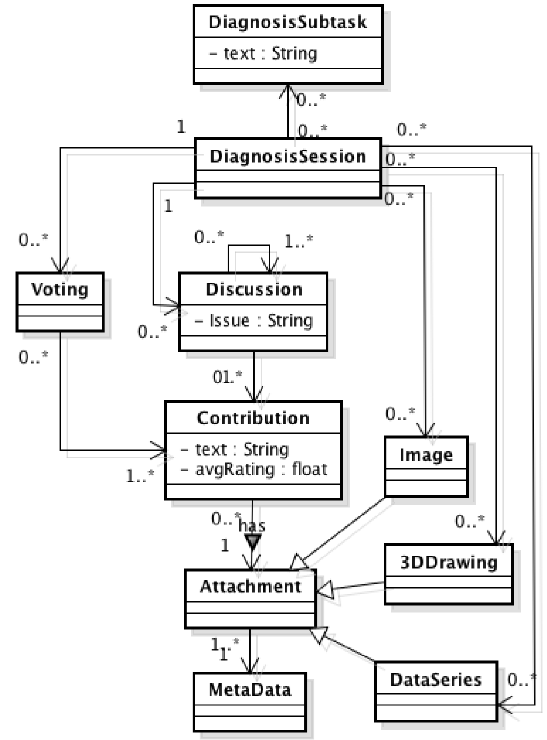


Figure 4.2: The ontology that describes the concepts of diagnosis task in Elgar.

teams in diagnosis. Team members need to analyse data, define hypotheses that identify root-cause failures of malfunctioning machines, describe possible actions to overcome these failures, and generate documentation that helps maintenance teams to fix anomalies detected in a machine. Figure 4.2 describes the concepts of Elgar involved in the functionalities that support teams during diagnosis tasks, implemented by ECCs.

A diagnosis session has three concepts that support data analysis in diagnosis tasks: data series, images and 3D drawings. The first concept *Data Series* represents the telemetry data acquired from a machine for a specific period. The telemetry data of machines has several data series related to specific machine sensors. For example, the data series that represents oil temperature has several readings, presented in the megapascal (MPa) per millisecond (ms). The second concept “Image” represents all pictures and illustrations used by experts in diagnosis tasks to indicate physical anomalies with machines. In some situations, diagnosis experts identify the root-cause failures with machines solely on the basis of the analysis of images of its parts, such as subsystems or components, instead of the analysis of telemetry data. Finally the third concept *3D Drawing* represents three-dimensional computer-aided drafting (CAD) models that diagnosis experts use to analyse the design of certain machine components, rotating the model three-dimensionally for immediate visualization of the different faces of the component.

Besides the data analysis, diagnosis sessions have a set of discussions, represented by the concept *Discussion*. Discussions represent a dialogue among team members, contextualized by a topic. During data analysis, diagnosis teams generate the necessary documentation to report the diagnostics of a machine. Team members use discussions to generate diagnostics, they describe hypotheses for root-cause failures, action plans to overcome failures and implementation plans. Discussions are composed of a set of contributions, represented by the concept *Contribution*. A contribution is a textual description that represents hypotheses of machine failures, actions to overcome them or text provided by team members in a dialogue, in the context of diagnosis task. In addition to the text, contributions have attachments, represented by the *Attachment* concept. Attachments are indications of specific selected data series, images or 3D drawings that are associated with contributions. They are additional resources to illustrate text described in contributions. For example, a team member who generates a hypothesis about a machine failure, associates it with a specific data series that focuses on readings of a delimited time range (e.g. values of two and three megapascal in the time range from two hundred to four hundred ms), to illustrate the hypothesis. Each attachment also has meta information to describe the source location of an image (e.g. the directory and file name of an image in the server); the selected sensors and their range of data series (e.g. oil pressure and oil temperature from two hundred to four hundred ms); and the source location and the state of a 3D drawing (e.g. the directory and file name of an image in the server and degree of the rotated drawing).

A diagnosis session also includes voting, indicated by the concept *Voting*. The goal of this concept is to compare and rank a set of contributions according to their degree of importance, indicated by team members. For example, team members who discuss actions to overcome root-cause failures of machine use voting to compare the actions and identify the most important one. Team members assign a score to each of the actions. The average of the scores is calculated for each action and in the end the voting shows a list of ranked actions.

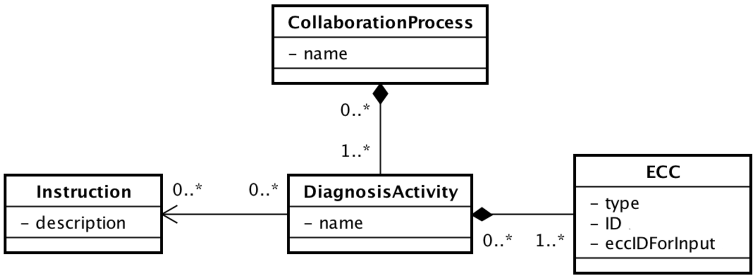


Figure 4.3: The concepts involved in the description of a collaboration process used by the prescribed coordination mechanism in Elgar.

Finally, a diagnosis session that uses the ad-hoc coordination mechanism also have a set of diagnosis subtasks, represented by the concept *DiagnosisSubtask*. This concept describes all of the subtasks that are required for a diagnosis team to accomplish the goals of the diagnosis task in a diagnosis session.

4.4 Prescribed Coordination Mechanism

Elgar enables diagnosis teams to use the prescribed coordination mechanism in a diagnosis session. Through this mechanism, teams follow a prescribed collaboration process to formulate diagnosis for anomalies presented in a machine [Janeiro et al., 2012a]. The Figure 4.3 depicts the concepts of collaboration processes associated to the prescribed coordination mechanism.

A diagnosis session that uses the prescribed coordination mechanism has a *CollaborationProcess* associated to it that represents the process, used by team members of a diagnosis session. Each collaboration process explains its purposes through the property *name*. A *CollaborationProcess* is decomposed in sub-parts, each specified by a *DiagnosisActivity*. The diagnosis activities denote the atomic actions that contribute to the accomplishment of a diagnosis task. *DiagnosisActivity* has a property *name*, for its description, instructions and a set of ECCs. *Instruction* is the concept that has information about the expected actions of team members in a *DiagnosisActivity*, described through the property *description*. Each diagnosis activity has ECCs, each represented by the concept *ECC*, through which team members analyse and share information. An *ECC* has the property *ID*, for its unique identification, the property *type* that describes the type of an ECC and the property *eccIDForInput* that specifies the ID of other ECCs. This property

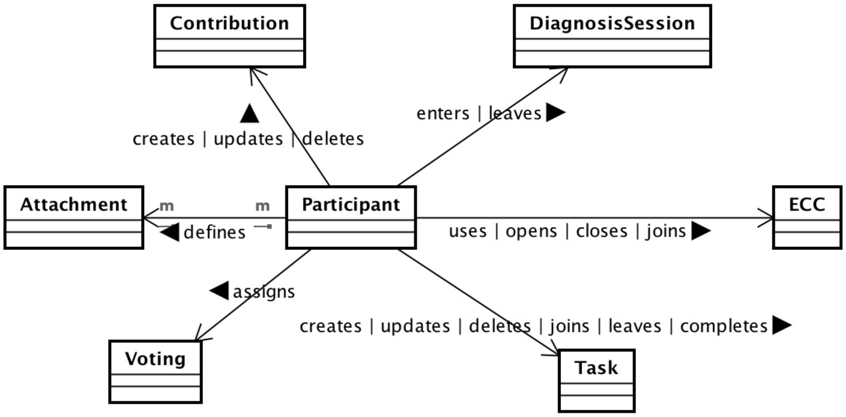

Contribution
DiagnosisSession
creates | updates | deletes
enters | leaves
Attachment
m
m
Participant
ECC
defines
uses | opens | closes | joins
assigns
creates | updates | deletes | joins | leaves | completes
Voting
Task

Figure 4.4: Workspace awareness ontology used by Elgar to monitor and capture the interaction of participants with the shared workspace system.

is used whenever an ECC needs to import data collected by another ECC.

4.5 Ad-Hoc Coordination Mechanism

Elgar provides specific workspace awareness information to teams involved in diagnosis tasks using the ad-hoc coordination mechanism, as described by requirement **R 2.4** of the Table 2.3. In addition to diagnosis functionalities, teams need to have access to workspace awareness information as a means to support the coordination of their actions dynamically to accomplish a task.

The central part of this coordination mechanism is the workspace awareness ontology used by teams during diagnosis tasks. Figure 4.4, describes the ontology used by Elgar to capture, store and distribute workspace awareness information to teams. The ontology is based on the workspace awareness information regarding past (Table 2.2) and current actions of participants with Elgar (Table 2.1).

The core concept in the figure is *Participant*, as workspace awareness information specifies the interaction of team participants with Elgar and its ECCs.

Some of the concepts specified in the ontology refer to both coordination mechanisms, such as the *DiagnosisSession*, *Contribution*, *Attachment* and *Voting*. The workspace awareness ontology indicates when a participant enters or leaves a diagnosis session in Elgar, creates, deletes or updates a contribution, defines an attachment to it or assigns votes to contributions, independent of the coordination mechanism, e.g. participants provide contributions to a diagnosis session that either uses the ad hoc or prescribed mechanisms.

The ontology includes two concepts that are specific to the ad-hoc coordination mechanism: *DiagnosisTask* and *ECC*. The relationship between *DiagnosisTask* and *Participant* in the ontology specifies the subtasks executed by team participants during a diagnosis session, e.g John works on the analysis of the hydraulic oil data of a machine. Similarly, the relationship between *ECC* and *Participant* specifies the ECCs used by team participants during a diagnosis session.

4.6 Presentation of Diagnosis Sessions

Diagnosis session in Elgar are designed to be dashboards, as described by requirement **R 3.3** of the Table 2.3. A Dashboard is a term often used in vehicle manufacturing industry to allow drivers to assess indicators of the status of a vehicle at a glance [Pauwels et al., 2009]. The metaphor of a dashboard is often further used in software systems as means that enable participants to assess data, updated in real-time, facilitating them to take quick informed decisions [Marcus, 2006].

In Elgar, each diagnosis session uses the metaphor of a dashboard to enable team members to have updated information about the status of an observed machine. Using dashboards in diagnosis sessions, team members are able to analyse, at a glance, telemetry data, CAD models and images regarding a machine to better assess its anomalies. It also enables team members to exchange and rate contributions related to diagnosis formulation.

In Elgar, the dashboard used in diagnosis sessions are designed to be flexible and modular, so that team members are able to customize the dashboard space with required information to diagnose a machine. Dashboards are composed of ECCs, as atomic building blocks, that display information and enable information exchange between team members. In the context of dashboards, the ECCs are pluggable snippets of user interface, each focused to provide specific information, e.g. machine telemetry data, or enable interaction between team members to exchange information, e.g. the cause of a specific machine anomaly.

Each dashboard is represented as a theoretical table that has two columns and an unspecified number of rows. Each cell of the theoretical

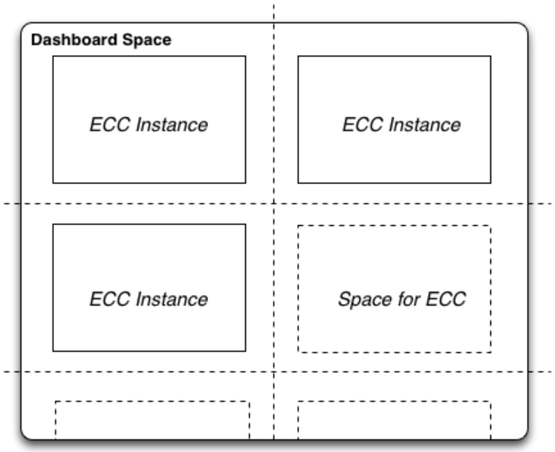


Figure 4.5: The layout of a dashboard space of an Elgar diagnosis session.

table may contain an instantiation of an ECC, as illustrated in Figure 4.5. The layout of two columns and unlimited rows was chosen to provide the maximum number of ECCs to participants of a diagnosis session without the need to scroll through the page. A one column or one row layout would provide a higher width space in the ECC canvas but participants would have to scroll vertically or horizontally through the page to interact with other ECCs.

4.7 System Design

Shared workspace systems, like Elgar, use services to support collaboration of teams: to enable teams to simplify communication about artefacts, coordinate activities and maintain awareness among its members [Gutwin et al., 2008]. Elgar provides these services with the focus on the discussions among team members about machine diagnosis and on the support of different coordination mechanisms, which are chosen according to the preferences of diagnosis teams.

The implementation of these services in Elgar is based on a specific software architecture, described in the Figure 4.6. Elgar is a web application; therefore its logic is deployed and runs in an application server. Clients are web browsers that request access to Elgar through the application server and render the user interface in the browser. Different clients communicate with the server through a dedicated web channel (Web Communication Channel)

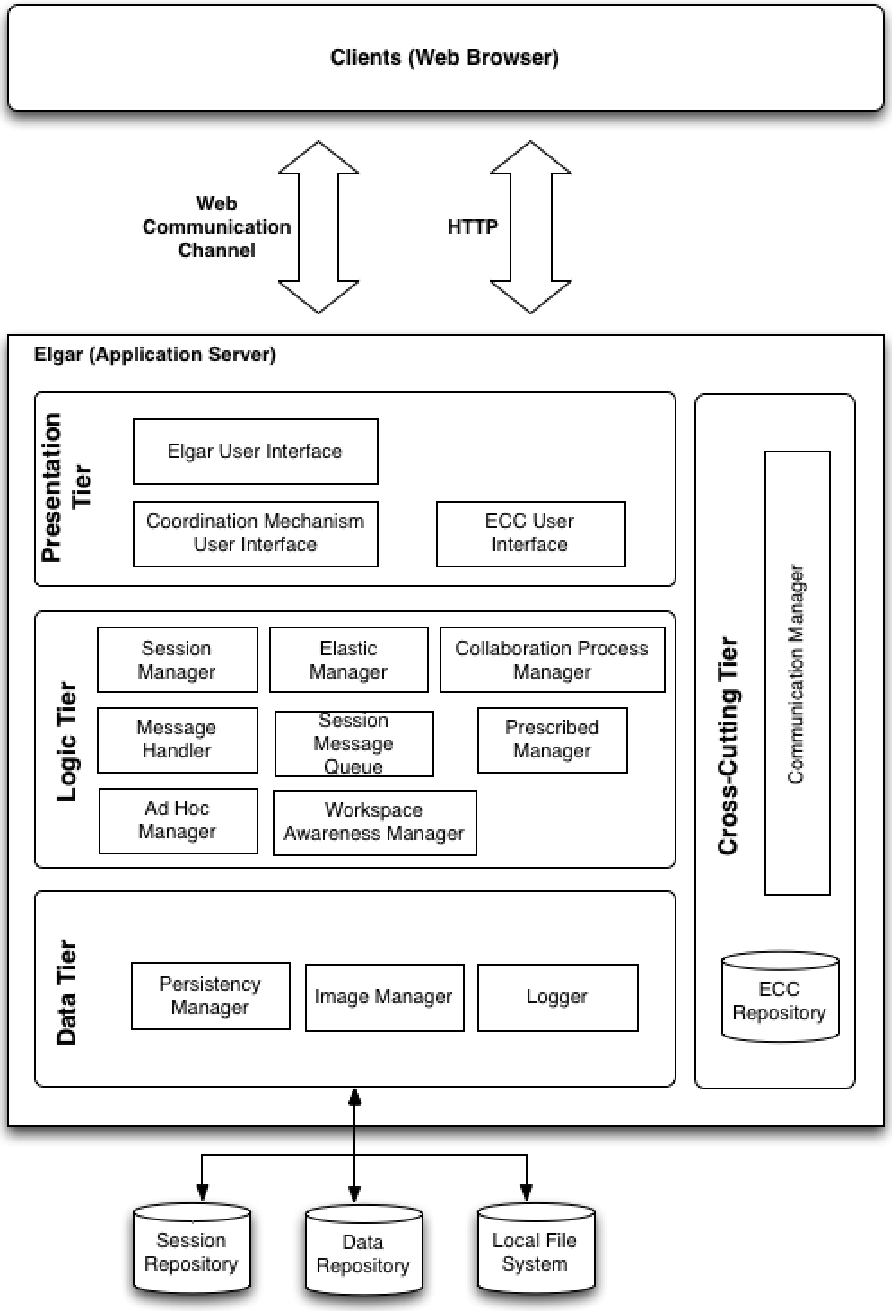


Figure 4.6: Three-tier architecture model of Elgar.

and through the HTTP protocol. Elgar uses persistence systems, which are in this case databases and a local file system.

4.7.1 Communication Channel

Shared workspaces systems are software systems that support multicast communication and are intensive in asynchronous message exchange with low-latency between the several clients and the server [Gutwin et al., 2011]. In these systems, team members are aware of the actions of each other in the workspace, interact simultaneously in the shared workspace system, use the workspace to explore and exchange ideas, and use it as means to review and tune discussions [Greenberg and Marwood, 1994]. Such systems need to keep all involved parts interconnected by the exchange of different types of messages, which in turn requires reliable asynchronous communication mechanisms.

Elgar, as a shared workspace system, is designed with a communication subsystem that enables multicast communication with asynchronous and low-latency message exchange between Elgar hosts (the clients of Elgar and the server). These functionalities are provided by the *Cometd Framework* [Bordet, 2016], which is the basis of the Communication Subsystem of Elgar.

The communication subsystem offers multicast communication through the publish-subscribe communication pattern to ensure that exchanged messages are distributed to connected hosts. All hosts are subscribed to a common communication channel to receive messages distributed through it. Whenever a host needs to communicate with other hosts, it publishes a message in the common communication channel. The communication can be bilateral, involving a client and the server or multilateral, involving the hosts subscribed to the channel.

The communication subsystem of Elgar also offers asynchronous message exchange between hosts to allow the continuous execution of their logic, avoiding blocking themselves while waiting for a response. For example, a client's ECC that requests all exchanged contributions of a diagnosis session, continues its execution (e.g. process other incoming messages or change its user interface) until the response arrives.

Asynchronous communication exchange also has the advantage that it allows a server to initiate communication with a client, without the request of a client. This functionality is achieved through the implementation of a push technology known as Comet [McCarthy and Crane, 2009]. In this technology hosts establish a long-held connection among themselves through which they can send data to the other host. For example, the server notifies the clients of a diagnosis session about a newcomer client that joined the same session. Such a characteristic overcomes the traditional limitation of servers that only respond to requests originated by clients.

Finally, the communication subsystem of Elgar provides low-latency message exchange between the hosts because of the efficient communication services provided by the Cometd Framework. The framework guarantees to deliver messages with a maximum latency of fifty milliseconds[2] [3], when five hundred host are connected through a Amazon EC2 local area network. Most of the exchanged messages in Elgar represent modifications in the user interface, provided by contributions of team members that attend the same diagnosis session. However, the latency of fifty milliseconds is lower than the responsiveness tolerance of two and four seconds of general user interface participants [Galletta et al., 2004], enabling this subsystem of Elgar to offer low-latency communication.

4.7.2 Storage

Elgar uses three different databases: Session Repository, Data Repository and the Local File System. The *Session Repository* stores diagnosis sessions and their state. In the case of sessions that use the prescribed coordination mechanisms, this repository also stores the description of the collaboration process associated with the session, as described by the requirement **R 2.2**. The *Data Repository* stores all data used by the available ECCs in Elgar and the *Local File System* stores all images that are uploaded in Elgar, e.g. pictures of the engine of a failing mechanical machine. The first two repositories are document-oriented databases, whereas the third repository is the local file system of the application server that runs Elgar.

Document-oriented databases are designed to store, retrieve, and manage document-oriented information, often also referred as semi-structured data. A document is the fundamental concept of such databases, which represents a group of data that encodes user-readable information. The major advantage of these types of databases is the absence of a schema that typifies data stored in documents. In practice, two different documents of the same type may contain different schemas and data values. For example, the concept *Contribution*, depicted in Figure 4.2, is represented as a document in these databases. Most of the instances of "Contribution" are expected

[2] `https://webtide.com/cometd-2-throughput-vs-latency-2/`. Last Access in 03.September.2016

[3] `http://cometd.org/documentation/2.x/howtos/loadtesting`. Last Access in 03.September.2016

to have only a textual description. However, some instances might diverge from the pre-defined properties and have additional properties, such as a vote or an attachment.

In the case of Elgar, the use of these databases supports the extensibility of the system.

4.7.3 Architecture of Elgar

Figure 4.6 describes the architecture of Elgar, on the server side, according to the three-tier architecture model [Buschmann et al., 2008]. This model is based on the multi-tier architecture model. It separates the logic of a software system in three tiers or layers, presentation, logic and data. The presentation tier contains components used to design and render the user interface, the logic tier contains components used to control the logic of the system and the data tier contains components used to manage the persistence of data in any type of data storage system. In addition to these three tiers, Figure 4.6 introduces a cross-cutting tier that has components used to manage the communication with clients.

Presentation Tier

The presentation tier has components that render the user interface at the client side. There are three different components for this purpose in this tier, the Elgar User Interface, the Coordination Mechanism User Interface and the ECC User Interface. The *Elgar User Interface* component has the logic that renders the user interface for participants to execute CRUD operations with diagnosis sessions. The *Coordination Mechanism User Interface* has the logic that renders the user interface for a particular coordination mechanism. In the context of this thesis, this component has the logic to render the user interface for the ad-hoc and prescribed coordination mechanisms. The *ECC User Interface* contains the logic that renders the user interface of ECCs at the client side.

Logic Tier

The logic tier has the components of Elgar that support teams in diagnosis tasks and their coordination. The *Session Manager* is a central part of this tier as it manages all diagnosis sessions in Elgar. It enables the creation, deletion, update and read of diagnosis sessions. It enables the association of coordination mechanisms to each session and, in case a session uses the prescribed coordination mechanism, the *Session Manager* also manages the association of the collaboration process to the session. In addition to these operations, this manager tracks the attendance of diagnosis team members in the various diagnosis sessions.

The *Session Manager* also manages the concurrency aspects of simultaneous diagnosis sessions. Elgar stores several diagnosis sessions in parallel with different team members for each. During diagnosis, team members often change the state of a session, e.g. they write contributions, delete, rearrange or update them. Whenever Elgar processes these operations, it identifies the appropriate diagnosis session to change its state accordingly and avoid inconsistencies with other sessions.

The *Ad Hoc Manager* provides services to all diagnosis sessions that use the ad-hoc coordination mechanism. Each of these sessions exchanges control messages with the server that concern the ad-hoc coordination mechanism. The exchanged messages contribute to update specific information with regard to workspace awareness to support team members to coordinate their actions. The *Ad-Hoc Manager* is responsible to process incoming control messages of the ad-hoc mechanism from diagnosis sessions and generate outgoing messages to clients, which keeps workspace awareness information updated, either in the server or in the clients. The exchange of information is triggered upon specific events in sessions that use the ad-hoc coordination mechanism, as described previously by the workspace awareness ontology (Figure 4.4). Whenever one of these events occurs, a client sends a notification message to the server. The server, through the *Ad-Hoc Manager*, updates the workspace awareness model and broadcasts it to all clients, which in turn, update their workspace awareness model locally and the ECC that shows such information.

The *Workspace Awareness Manager* is an auxiliary manager of the *Ad-Hoc Manager*. It holds the workspace awareness information in a model for each of the diagnosis sessions. Whenever the *Ad-Hoc Manager* processes a notification message from one of the clients, it sends updates to the *Workspace Awareness Manager* that executes them internally.

The *Prescribed Manager* is the manager that provides services to all diagnosis sessions that use the prescribed coordination mechanism. It has the logic to read, process and run collaboration processes, assigned to sessions that use the prescribed coordination mechanism, as described by the requirement **R 2.3**. Each of these diagnosis sessions exchange messages with the server to acquire the collaboration process description, instructions and the ECCs of each process activity. The selection of a process activity triggers the exchange of messages between the client and the server. This manager processes these messages and provides the required information to all clients that request it. The manager gathers the process description or the elements of a process activity, serializes it and embeds it in an outgoing message to one specific client. In this case, the *Ad-Hoc Manager* and the *Prescribed Manager* do not prepare a broadcast message. As team members in the same Elgar session do not necessarily work together in the same collaboration process activity, they may request from Elgar information about different process activities.

Whenever, the *Prescribed Manager* requests a process description, or ECCs and instructions for a specific collaboration process activity, it requests this information from the *Collaboration Process Manager* to configure the shared workspace system. In addition, this manager has the logic to serialize collaboration processes and send to clients or deserialize collaboration processes that are persisted in databases.

The *Elastic Manager* recognizes and processes control messages used by both coordination mechanisms of diagnosis sessions. The manager is the controller of coordination mechanisms: the manager delegates the process of an incoming message to one of the auxiliary managers (Ad-Hoc Manager or Prescribed Manager), instead of processing itself. In addition, this manager is also a controller for Elgar, as it controls the flow of execution of Elgar and mediates the results produced by all other managers, such as from communication or data persistence.

A diagnosis session requires Elgar to execute two types of operations: memory operations that access information in the main memory, (e.g. the collaboration process description of a diagnosis session that is already loaded), and persistent operations that read or write information in databases (e.g. read the values of data series for a diagnosis session). Persistence operations are more time consuming due to the overhead involved in processing them. However, Elgar may not be able to process all incoming requests from diagnosis sessions, whenever the throughput of messages is high. Therefore, all incoming requests are temporally placed in an internal buffer, named *Session Message Queue*, until their processing is completed to avoid that certain messages are lost by the system.

The queue prevents data and state corruption of diagnosis sessions, as it preserves requests from clients in a first-in-first-out (FIFO) order. An unpreserved processing order of requests may lead to synchronization problems of the distributed state of clients. For example, a team member requests to update a contribution with a value “x”, right after the team agrees that the correct value is “y” and another team member requests it from the server [Tanenbaum, 2006]. The incorrect processing of these requests leads to an unexpected inconsistent state of the diagnosis session.

The use of a *Session Message Queue* increases the response time of Elgar to process requests from clients because they await for the system to process the several queued requests before they receive their respective responses. However, it is a often used mechanism to avoid request discarding.

In addition, due to the processing overhead of persistence operations, Elgar may become temporarily unresponsive to process memory operations while reading or writing in one of the data repositories of file system. Elgar use the *Message Handler*, a *thread* that executes persistence operations in parallel, to overcome such temporary unresponsiveness. Such parallel execution contributes to increase the responsiveness of Elgar and reduces the waiting time of clients that depend on responses of the system. This component is a dedicated thread that executes persistent operations continuously, concurring with the execution of Elgar. The *Message Handler*, works with the *Session Message Queue* in a producer-consumer situation, in which the clients of Elgar produce requests, stored in the *Session Message Queue*, whereas the *Message Handler* consumes requests stored in the queue. Currently, there is one *Message Handler* for each Elgar diagnosis session, and therefore one thread, per session.

Data Tier

The data tier has components of Elgar that are responsible for data storage and therefore interact with databases and the local file system. This tier has three managers that support persistence: the *Persistency Manager*, the *Image Manager* and the *Logger*.

The *Persistency Manager* connects to the databases used by Elgar and manage its access permissions. The manager interacts directly with the Session Repository and with the *Data Repository.* The interaction of this manager with the *Session Repository* is to store data from diagnosis sessions, such as the descriptions of sessions, the associated coordination mechanisms and, in case of the prescribed coordination mechanism, the prescribed collaboration process description. This manager also deserializes data from the Session Repository and loads the state of a diagnosis session into memory, such as the collaboration process, used ECCs and their respective data.

This manager analyses incoming messages from clients and delegates to the appropriate counterpart of the ECC in the ECC Repository of the server-side to execute a persistence operation with databases. The *Persistency Manager* does not execute operations in the database for ECCs, it only identifies the appropriate ECCs that does it.

The *Image Manager* stores images in the local file system, uploaded by team members in a diagnosis session and retrieves them. The *Data Repository* only stores human-readable documents (described in the JSON format) and not binary objects, which in this case are images. Therefore, the storage of images is decomposed in storage of meta-data which uses the *Data Repository* and the storage of binary objects which uses the local file system.

Whenever this manager receives an image, it extracts meta-data information from the image to store in the *Data Repository*. The meta-data is stored as a document and contains the name of the image, the respective diagnosis session, and the uniform resource identifier (URI) of the image. The *Image Manager* also stores the image in one of the dedicated directories, especially created for the diagnosis session in the local file system. The *Image Manager* only creates a directory for the session in the local file system upon the upload of the first image in the session to avoid allocation of unnecessary resources.

The *Logger* logs all actions of team members in diagnosis sessions for possible analysis of the session, including the information when participants create, update and delete data (e.g. contributions or ratings), add attachments to contributions, join or leave a diagnosis session, instantiate an ECC or delete its instantiation and move through collaboration process activities.

Cross-Cutting Tier

The cross-cutting tier has components that permeate, to a certain degree, the other tiers. This tier has two components, the Communication Manager and the ECC Repository.

The *Communication Manager* manages the communication channels that connect clients and the server. The *Web Communication Channel* aggregates several specific channels that are created by Elgar in the server. Each channel has a unique identification, used by clients and the server to exchange messages. The channels work according to a publish-subscribe mechanism; Elgar instantiates the channels and subscribes to them, subsequently clients also subscribe to channels of their interest. The *Web Communication Channel* has two channels, one for the exchange of control messages, e.g. workspace awareness information and collaboration process descriptions, and another for the exchange of data messages, e.g. CRUD operations for contributions. Another responsibility of the *Communication Manager* is to recognize and queue data messages that involve database operations and are executed by the *Message Handler* in parallel.

The *ECC Repository* is a logical repository that contains the logic of

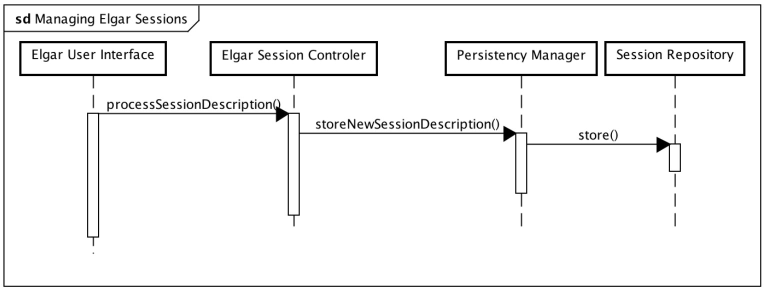


Figure 4.7: Sequence diagram describing Elgar components involved in managing Elgar sessions.

ECCs. Each ECC is divided in two parts: the logic of the server side and the logic of the client side. The logic of the server side executes CRUD operations with the concepts used by the ECC and stored as documents in the *Data Repository*. Whereas the logic of the client side renders the user interface of the ECC in the client and communicates with the server and other clients through the *Web Communication Channel*.

4.8 Use Cases of Elgar

This section describes use cases that represent the interaction between the clients of Elgar and the server side. Each of the following subsections describes the involvement of the components described in the architecture of Elgar to enable the realization of the use cases.

4.8.1 Managing Elgar Sessions

This use case describes the components of Elgar involved in managing the configuration of Elgar sessions, in which participants can create, update and delete Elgar sessions. Such sessions have an identification number, descriptive names and an associated coordination mechanism. Figure 4.7 illustrates the interaction between Elgar components involved in this use case.

Participants who create an Elgar session open the Elgar interface at the client side to describe the information related to the session. Once they provide the described information for the creation of a session, control of Elgar is transferred to the *Elgar Session Controller* to interpret the type of requested operation, e.g. create, delete or update. In case of session creation, Elgar Session Controller extracts the information provided by a

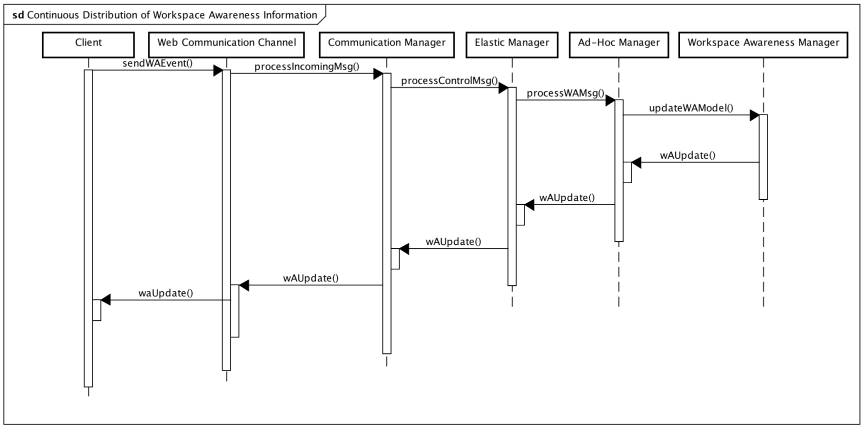


Figure 4.8: Sequence diagram describing Elgar components involved in the processing and continuous distribution of workspace awareness information.

participant and forwards it to the *Persistency Manager* to create an Elgar session in the *Session Repository*.

Once the session is stored, control of Elgar is transferred back to *Elgar Session Controller* that requests the *Persistency Manager* to retrieve an updated list of the existing Elgar sessions in the Session Repository to present to the participant.

4.8.2 Continuous Distribution of Workspace Awareness Information

This use case describes the components of Elgar involved in the processing and continuous distribution of workspace awareness information among the clients of Elgar of a particular session, e.g. the list of participants who are working in the same ECC during a diagnosis task. Figure 4.8 illustrates the interaction between Elgar components involved in this use case.

A participant who executes actions described in the Figure 4.4 triggers the client of Elgar to send an action, e.g. open a CollPad ECC, as a message to the server through the *Web Communication Channel*. Such a message type represents an update in the workspace awareness information of the Elgar session.

Whenever the message is received by the Elgar server, it is processed by the *Communication Manager*. This manager transforms the message in a Map data structure and identifies that it belongs to the category of control messages. The message is then delivered to the *Elastic Manager*, which

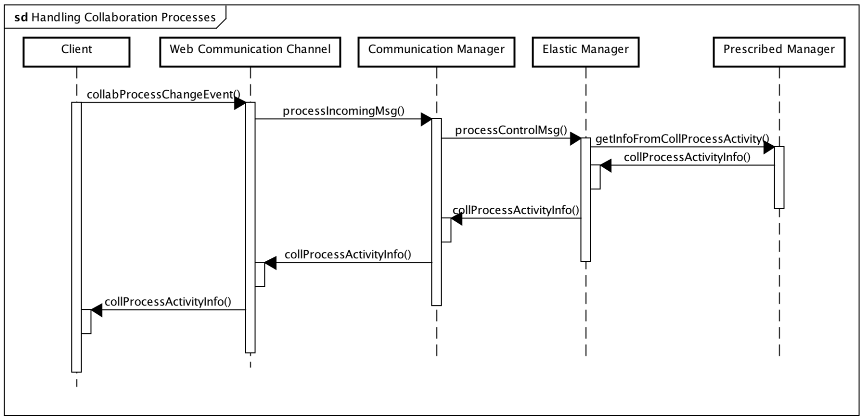

sd Handling Collaboration Processes
Client
Web Communication Channel
Communication Manager
Elastic Manager
Prescribed Manager
collabProcessChangeEvent()
processIncomingMsg()
processControlMsg()
getInfoFromCollProcessActivity()
collProcessActivityInfo()
collProcessActivityInfo()
collProcessActivityInfo()
collProcessActivityInfo()

Figure 4.9: Sequence diagram describing Elgar components involved in handling of collaboration process in Elgar.

identifies it is used in the ad-hoc coordination mechanism. The *Elastic Manager* hands over the message to the *Ad-Hoc Manager* for further processing. The *Ad-Hoc Manager* processes the message and interacts with the *Workspace Awareness Manager* to update the status of the workspace awareness information (stored in the memory of the server) based on the incoming message.

Once the model is updated, the *Ad-Hoc Manager* notifies the *Elastic Manager* with an updated version of the workspace awareness information, which is serialized and forwarded to the *Communication Manager*. Subsequently, this manager distributes to all clients connected to the *Web Communication Channel* so that they can update their local workspace awareness information.

4.8.3 Handling Collaboration Processes

This use case describes the components of Elgar involved in the processing of a collaboration process whenever a participant, using the prescribed coordination mechanism, transits into a collaboration process activity. Figure 4.9 illustrates the interaction between Elgar components involved in this use case.

Whenever a participant moves to a new collaboration process activity, the client of Elgar encapsulates such an event automatically into a message that is dispatched from the client only to the server through the *Web Communication Channel*. Whenever the message arrives at the server, it is

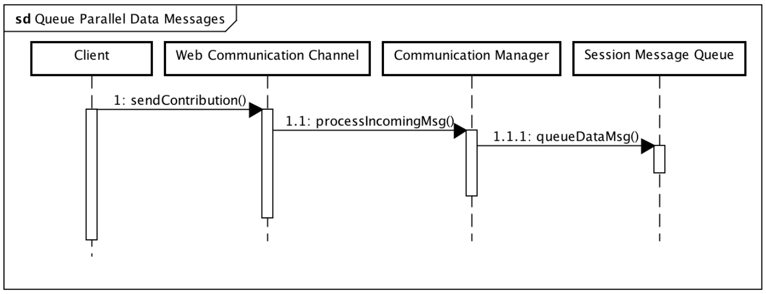


Figure 4.10: Sequence diagram describing Elgar components involved in queueing data messages originated simultaneously in Elgar clients.

transformed to a Map data structure by the *Communication Manager*. This manager identifies the data structure to be a control message and delivers it to the *Elastic Manager* to handle the control message. In this case, the *Elastic Manager* determines that the message is related to the prescribed coordination mechanism and delegates it to the *Prescribed Manager* for further processing. The *Prescribed Manager* interacts with the *Collaboration Process Manager* to acquire information about the requested collaboration process activity, such as the ECCs and the instructions for the activity.

Once the information is obtained, the *Elastic Manager* delivers it to the *Communication Manager* to send it to the client that requested it through the Web Communication Channel.

4.8.4 Queue Parallel Data Messages

This use case describes the components of Elgar involved in queueing data messages originated in different clients simultaneously. Figure 4.10 illustrates the interaction between Elgar components involved in this use case.

All data messages sent by participants from ECCs at the client side of Elgar to the server side, e.g. a hypothesis of a machine diagnosis, are transmitted through the *Web Communication Channel*. Whenever a message reaches the server side it is forwarded directly to the *Communication Manager*, which deserializes and transforms it in a Map data structure and puts it in the *Session Message Queue* for a later processing by the *Message Handler*.

4.8.5 Process Queued Data Messages

This use case describes the components of Elgar involved in the processing queued data messages originated in ECCs at the client side of Elgar. Figure 4.11 illustrates the interaction between Elgar components involved in this use case.

The *Message Handler* is a *thread* that processes continuously in a FIFO (first in, first out) order the messages queued by the *Message Handler* and interacts with the *Persistency Manager* to modify the status of the *Data Repository*. After the storage of data in the repository, the *Persistency Manager* interacts with the *Communication Manager* to broadcast data messages to all Elgar clients connected to the *Web Communication Channel*.

4.8.6 Handling Participant Input

This use case describes the components of Elgar involved in a participant-generated content operation. This particular case describes the generation of hypotheses in a machine diagnosis task within an Elgar session. Figure 4.12 illustrates the interaction between Elgar components involved in this use case.

Whenever a client shares a hypothesis about a machine anomaly in an Elgar session, the hypothesis is sent as a data message from the client only to the server through the *Web Communication Channel*. The message is forwarded directly to the *Communication Manager* that transforms the message in a Map data structure and puts it in the *Session Message Queue* for later processing by the *Message Handler*. Once the *Message Handler* picks the message for processing, it identifies that the request is intended to create

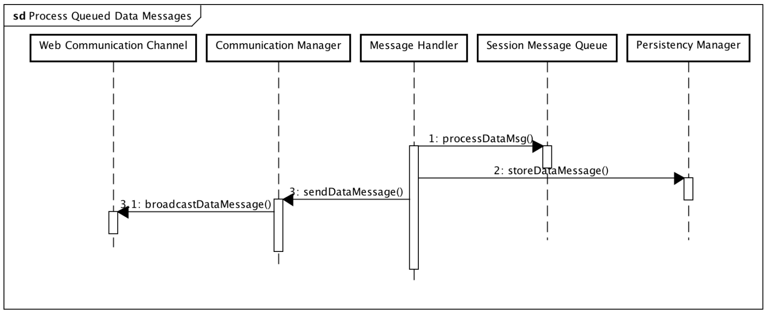


Figure 4.11: Sequence diagram describing Elgar components involved in processing data messages queued in Elgar.

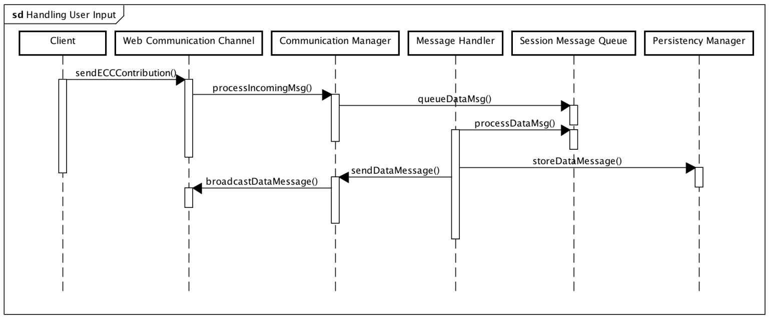


Figure 4.12: Sequence diagram describing Elgar components involved in handling a participant input.

a hypothesis in a diagnosis task for a particular ECC in an Elgar session. The *Message Handler* delivers it to the *Persistence Manager* that identifies the persistence interface of the ECC associated to the message at the server side and delegates to it the writing of the hypothesis in the *Data Repository*.

Finally, after the execution of the writing, the Message Handler notifies the *Elastic Manager* about the accomplishment of the persistence operation. The *Elastic Manager* delivers the message back to the *Communication Manager*, which serializes the message and broadcasts it to all connected clients to the *Web Communication Channel*.

4.9 Extension of Coordination Mechanisms

The architecture of Elgar is designed to enable extensibility to include new coordination mechanisms, as described by requirement **R 2.5** of Table 2.3. There are two extension points in the architecture to include a new coordination mechanism: *i)* the inclusion of a manager in the logic tier and *ii)* the user interface extension of Elgar in the presentation tier.

The extensibility of the coordination mechanisms is based on the spectrum of coordination support (Figure 2.2). New coordination mechanisms can build on functionality currently supported for the two coordination mechanisms - in particular with respect to workspace awareness information provisioning and process support. Therefore, a new coordination mechanism placed in the middle of the spectrum should use characteristics of the structured coordination as processes and the unstructured coordination as awareness information. For example, a recommender coordination mechanism could provide a process to teams (such as in the prescribed mechanism) and could use workspace awareness information (such as in the ad-hoc mechanism) to provide coordination recommendations. Such a mechanism could infer the interaction of teams during a diagnosis task, compute coordination recommendations and propose them to teams while they go through a process, e.g. remind team members to switch to following activities after long periods of discussions or recommend an equally balanced distribution of subtasks based on previous activities. In this scenario Elgar needs a new manager, in addition to the *Prescribed Manager* and *AdHoc Manager*, to interpret coordination of team members in diagnosis sessions and propose coordination recommendations.

In addition to the inclusion of a new manager in the logic tier of Elgar, it is also necessary to adapt the user interface with regard to the new coordination mechanism. For example, a recommender mechanism needs a space in the user interface of diagnosis sessions to present coordination recommendations for teams. Therefore, the *Coordination Mechanism User Interface* of the presentation tier would need extensions to include user interface widgets that present specific recommendation information of a new coordination mechanism.

4.10 Discussion

The ontologies and the architecture of Elgar described in this chapter fulfil the pre-defined set of requirements. The contributions of the architecture are summarized in the following list:

- Extensibility of Coordination Mechanisms.
- Extensibility of the Functionalities of Elgar.
- Management of Collaboration Processes.
- Management of Workspace Awareness Information.
- Asynchronous and Low-Latency Communication.
- Provision of Functionalities to Support Diagnosis Tasks.

The architecture of Elgar supports the extensibility of coordination mechanisms in Elgar. A new coordination mechanism requires the design of a manager to occupy the logic tier, so that the *Elastic Manager* recognizes and delegates specific messages to it for processing. The manager must have the logic to process control messages that are specific for the coordination mechanism. For example, a new coordination mechanism in the middle of the spectrum of coordination could provide suggestions and recommendations of action, based on the interaction of team members, in real-time so that they adjust their course of action accordingly. This type of support satisfies the requirements **R 2.1** and **R 2.5**.

Elgar supports the management of collaboration processes due to the *Session Manager*, *Persistency Manager* and *Collaboration Process Manager*. The first two components manage the creation, update, retrieve and deletion of diagnosis sessions in the Session Repository. The Collaboration Process Manager manages the serialization of processes in the Session Repository and its deserialization from the Session Repository in the memory. This type of support satisfies the requirements **R 2.2** and **R 2.3**.

Elgar also supports the management of workspace awareness information, especially because of the *Workspace Awareness Manager* that supports the ad-hoc coordination mechanism. This manager monitors constantly changes of team members in diagnosis sessions, their use of ECCs and their involvement in the diagnosis task. Such information is collected and distributed constantly to diagnosis sessions, to keep team members aware of their interaction and the interaction of other member of a session with Elgar. This type of support satisfies the requirement **R 2.4**.

Elgar also supports the extension of its functionalities, by modularization of ECCs. Software engineers who need to extend functionalities in Elgar should develop and deploy new ECCs, which have the server part to execute operations in the database and the client part to render the user interface of the ECC at the client. The ECCs implemented by Elgar currently are based on the concepts described in the ontology illustrated in Figure 4.2 that comply with the requirements **R 3.1**, **R 3.2**, **R 3.3** and **R 3.4**.

Chapter 5

Implementation of Elgar

The previous chapter introduced the concepts on which Elgar relies. It describes the main design decisions of Elgar, the models and the concepts used in Elgar for diagnosis tasks and coordination mechanisms and its architecture. The focus of this chapter is on the core of Elgar to support the discussed requirements of previous chapters, the technologies, techniques and programming languages used to realize the concepts involved in Elgar.

This chapter described an initial set of implemented ECCs used to support diagnosis tasks of machines, from data analysis to diagnosis formulation, based on typical machine diagnosis use cases [Heidbrink, 2013][Johanson et al., 2013][Heidbrink et al., 2013]. The chapter also describes the communication technology that is the basis of ECCs to support collaboration among different clients. Subsequently, there is a description of each of the coordination mechanisms implemented currently in Elgar. The description embraces specific workspace awareness information and its presentation to teams, such as in the ad-hoc coordination mechanism, and the specification and serialization of a collaboration process with a sequence of activities and used ECCs, such as in the prescribed coordination mechanism. Finally, the end of this chapter discusses the fulfilment of the requirements defined previously by current the implementation of Elgar.

5.1 Elastic Collaboration Components

Implementation of ECCs in Elgar is divided in two modules: the data management module and the user interface module. The data management module provides access to the data model of the ECC, stored in the Data Repository. The data management module implements a Java interface named *ElgarDataAccess*. This interface defines five methods that ECCs use to access the data model in the Data Repository:

1. Create. The *create* method implements the logic to add data to the data model of the ECC.

2. Read. The *read* method implements the logic to retrieve data from one instance of the data model.

3. ReadAll. The *readAll* implements the logic to retrieve a collection of instances from the data model.

4. Update. The *update* method implements the logic to modify the data model.

5. Delete. The *delete* method implements the logic to remove data from the data model.

The user interface module implements the user interface of an ECC at the client side of Elgar. The implementation of the user interface module is realized through XHTML and JavaScript, supported by the Dojo Toolkit [1].

The implementation in JavaScript is based on the object orientation paradigm, using constructors, destructors and properties to design interfaces for ECCs as objects. Constructors of the user interface implement the code that renders the user interface widgets for the ECC and requests initial data to Elgar (Data Repository) to populate the ECC on the client side with information stored previously. The Destructors implements the code that destroys user interface widgets to release allocated resources. The properties of the user interface of an ECC represent the user interface widgets that are used by the ECC or internal data structures. In Elgar, the properties of an ECC are private, they are only accessed by the ECC internally and other ECC do not have direct access to them.

The user interface module of an ECC is based on the standard layout defined by *portlets* [Java Community Process, 2008], pluggable user interface software components that belong to a page of a web portal. Portlets are designed to be independent components that implement a particular functionality and have a common layout.

Figure 5.1 illustrates the layout of portlets in a web portal. A portlet is contained in a *portlet window*, arranged in a portal page. A portlet has two parts: *i) controls* and *ii) portlet fragment*. The control part implements

[1] Dojo Toolkit: `https://dojotoolkit.org/`. Last Access in 03.September.2016

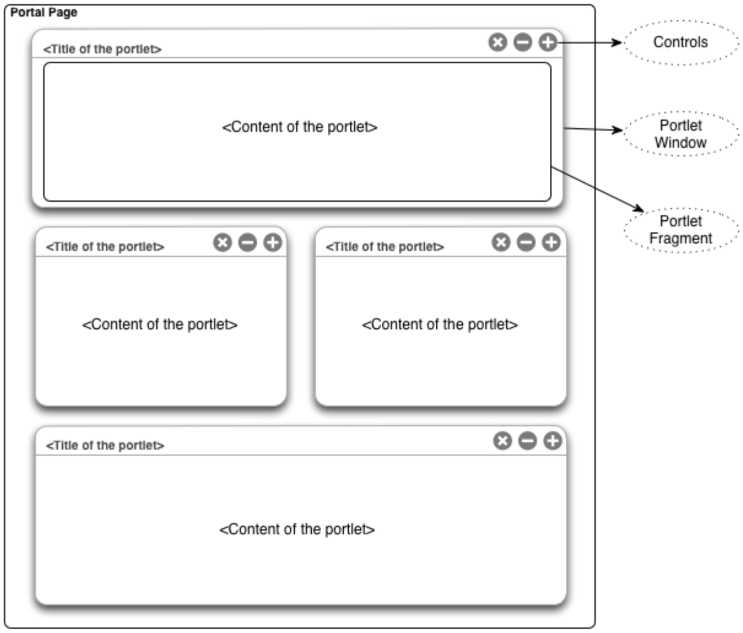


Figure 5.1: Elements of portlets in a web portal page.

the display of the title bar of a portlet and three optional control buttons to minimize, maximize and close a portlet and the title of a portlet. The portlet fragment represents the space for a portlet to implement a functionality. In Elgar, each ECC is enclosed in a portlet window, has a title bar with the three control buttons and the portlet title to fulfil the requirement R3.3 of the Table 3.1.

Elgar implements different ECCs to support teams in diagnosis tasks, as defined by the requirements of the Table 3.1:

- Static Line Chart;
- Visual Query Editor and Visualization Tool;
- VRML Viewer;
- Image Viewer;
- CollPad;
- Rating; and
- Session Summarizer.

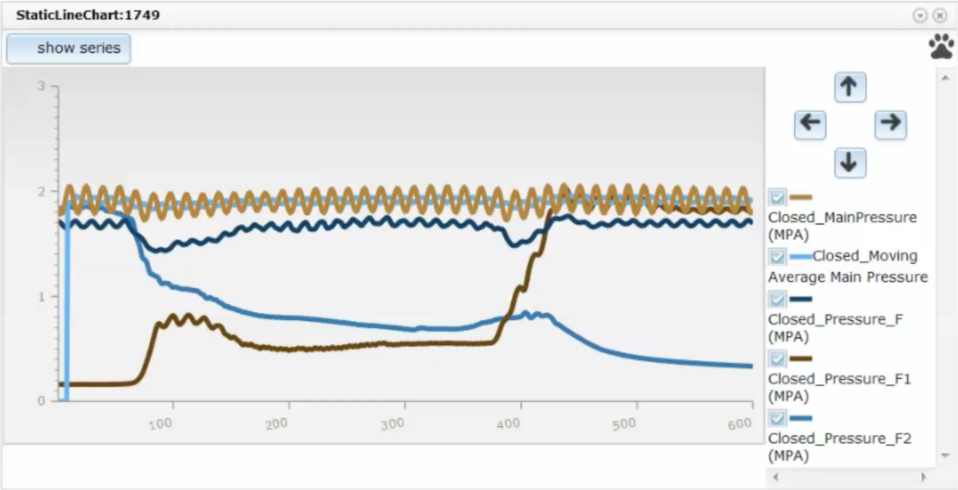


Figure 5.2: Instance of the Static Line Chart ECC showing five sensor readings of the hydraulic oil pressure of a machine.

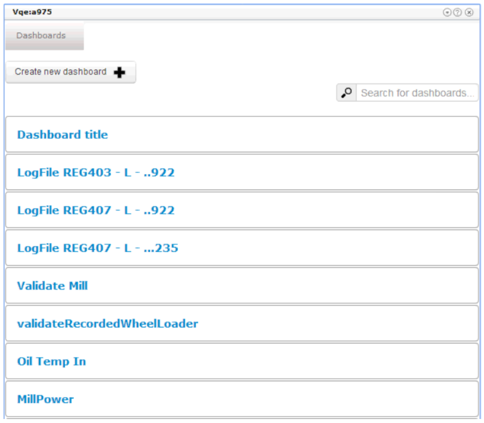
Static Line Chart

The Static Line Chart ECC allows participants to analyse data plotted as line charts during a diagnosis task. Participants can add new data series, rename a specific data series with a label, or delete existing data series. Each label is automatically associated to a *checkbox*, in which its selection determines whether the chart shows the data series associated to a label or hides it.

The Static Line Chart ECC has two axes, the x axis represents time in milliseconds and the y axis represents possible values. In addition, participants can navigate through data series by zooming in, zooming out and moving data to the left or to the right. Zooming in makes the Static Line Chart reveal more details about the data series and reduces the overview of the chart, whereas zooming out hides details of data series and shows an overview of the chart. Participants can also navigate to the left and to the right, putting the focus of the data in a certain time range.

Visual Query Editor and Visualization Tool

The Visual Query Editor (VQE) and Visualization Tool (VT) are two ECCs coupled together that enable participants to pose continuous queries (CQs) visually over machine sensor measurements and visualize their results continuously in a time-oriented graph. These two ECCs make it possible for team participants to gain insights over the occurrence of possible anomalous



Vqe:a975

Dashboards

Create new dashboard

Search for dashboards...

Dashboard title

LogFile REG403 - L - ..922

LogFile REG407 - L - ..922

LogFile REG407 - L - ...235

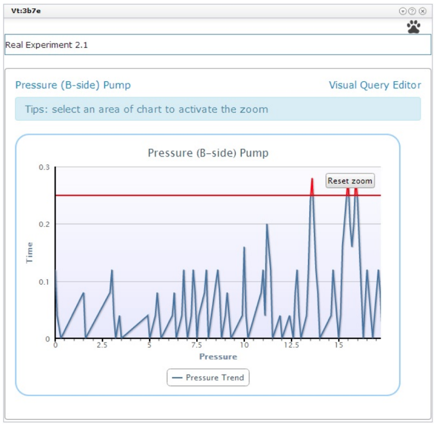
Validate Mill

validateRecordedWheelLoader

Oil Temp In

MillPower

Figure 5.3: An instance of the Visual Query Editor ECC showing examples of stored dashboards with defined visual queries.



Vt:3b7e

Real Experiment 2.1

Pressure (B-side) Pump

Visual Query Editor

Tips: select an area of chart to activate the zoom

Figure 5.4: An instance of the Visualization Tool ECC showing monitored data of a machine sensor and a threshold bar.

equipment behaviour, e.g. the main hydraulic pressure is above a threshold.

In the Visual Query ECC, participants describe a continuous query with visual representations of the sensor data sources of an observed machine and logical operators, e.g. “and”, “or” or “join”. The visual description of the query, named a dashboard, is translated into source code of the query that retrieves the desired information of the machine. Dashboards are stored and are available for reuse in the ECC, as illustrated in Figure 5.3.

The Visualization Tool ECC, is an ECC that enables team participants to visualize results retrieved from the continuous queries of a dashboard. The VT constantly adapts the chart to provide an overview of the retrieved measurements. However, team participants can zoom into a certain part of the chart by selecting the time range of interest. In addition, VT also offers the possibility to display a bar, that describes the threshold defined for a continuous query.

VRML Viewer

The VRML Viewer allows participants to evaluate 3D CAD (computer-aided design) models. This ECC shows all images of a 3D CAD model in a *tree structure*, a directory icon describes the model and document icons describe each file of the model.

In this ECC, a participant can choose an image from a list (a dropdown list named “files” in the Figure 5.5) for analysis. The following actions may

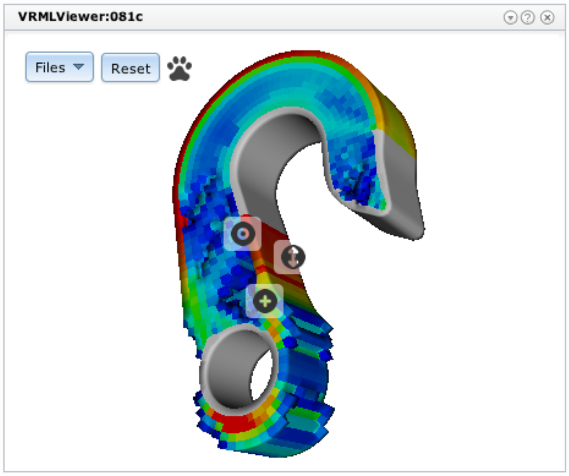


Figure 5.5: Instance of a VRML Viewer ECC showing an intermediary part of the optimization process of CAD models.

be used to evaluate a 3D model: zoom in, zoom out, rotate, translate or scroll a 3D model, as indicated by the transparent handlers shown in Figure 5.5.

Image Viewer

The Image Viewer ECC enables team members to upload and share images with each other to support the diagnosis task, as illustrated in Figure 5.6. The Image Viewer allows its participants to upload any image stored in the device that runs Elgar, e.g. desktops or iPads, assign a label and make it available in a list of photos of the ECC to all participants of a specific Elgar session. This ECC synchronizes the displayed picture automatically to all participants of an specific instance of the Image Viewer, upon selection (mouse click) of a picture by a participant.

CollPad

The Collaborative Notepad (Collpad) is a multi-role ECC that allows its participants to share their ideas and create text structures, based on lists, collaboratively as illustrated in Figure 5.7. This ECC supports participants in documenting hypotheses for machine anomalies and to sketch action plans to overcome anomalies. CollPad allows participants to define the topic of a certain discussion (issue) and provides textual contributions focusing on the topic.

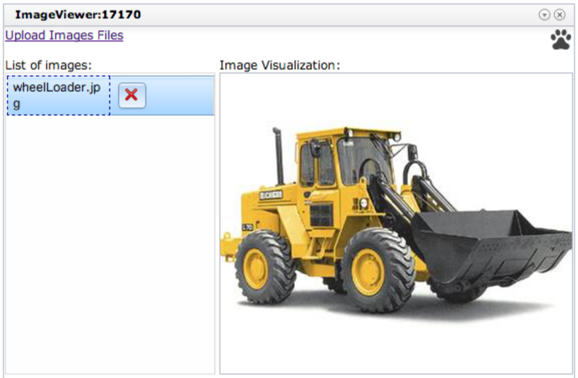


Figure 5.6: Instance of a Image Viewer ECC showing a normal wheel loader.

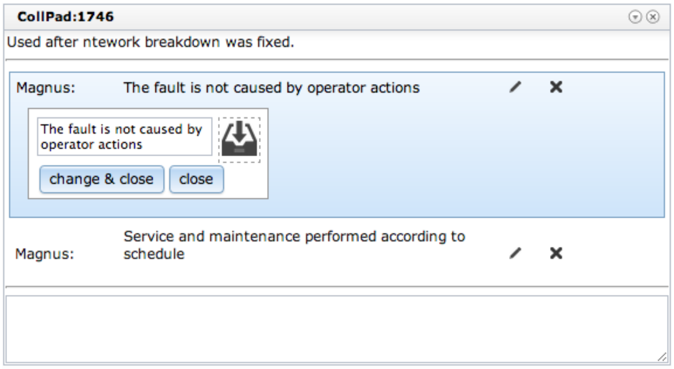
In this ECC, participants generate different contributions that are automatically shared with other participants. Contributions are related to a discussion topic displayed at the top of the ECC. All contributions are shared automatically between team members and stored in CollPad together with the name of the participant who wrote a contribution or the last participant who modified an existing contribution.

Each contribution enlisted in the Collpad is implemented automatically as a draggable object to facilitate their organization in different instances of CollPads or to be used as an input to other ECCs.

In addition to textual descriptions, contributions of a Collpad can be associated to attachments that represent the state of the Image Viewer, VRML Viewer or Static Line Chart ECCs. Such associations between textual description and the state of an ECC support discussion in relation to a specific type of data evaluated in a diagnosis task. For example, the understanding of a textual contribution that describes defect of a 3D drawing may be enhanced by the attachment of the state of a VRMLViewer. Participants can assign the state of a 3D drawing by dragging the *paw* icon associated with the drawing and dropping it on a textual contribution of the CollPad. Through such an attachment, team members can open the VRMLViewer with the associated state that clarifies the textual description.

Rating

The Rating ECC allows participants to rate a set of pre-selected contributions from CollPads. Participants of Elgar drag contributions from a Collpad and drop them in the Rating, creating a dynamic list of contributions collaboratively, as illustrated in Figure 5.8. Participants can also remove a



CollPad:1746

Used after ntework breakdown was fixed.

Magnus: The fault is not caused by operator actions

The fault is not caused by operator actions

Magnus: Service and maintenance performed according to schedule

Figure 5.7: Instance of the Collpad ECC during the discussion of machine in a diagnosis task.

contribution from the Rating ECC by using the *remove* button.

Participants assign ratings to contributions ranging from a scale between one and five to describe their importance, in which one represents low importance whereas five represents high importance. Participants can assign a rating to a contribution by selecting them and defining a rating by choosing the icon of the correspondent *star* icons. Subsequently, the average for each of the ratings is calculated and displayed in Rating ECC.

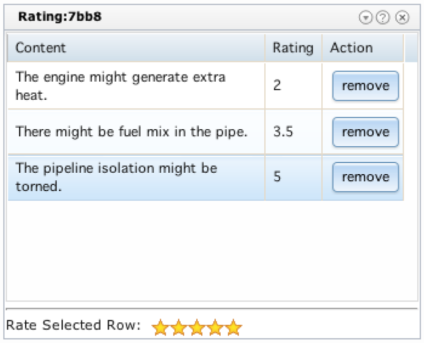
The averages for each of the contributions provides a means to compare relative importance between them. Contributions that have high averages should be considered for further attention of a team, rather than contributions with low averages.

Session Summarizer

The Session Summarizer is an ECC that summarizes results of an Elgar session and generates a report, as illustrated in Figure 5.9.

The Session Summarizer summarizes an Elgar session through a list in which each entry represents an instance of an ECC. The summary of an Elgar session enables participants to review the ECCs used during the session and their content. Elgar includes ECC entries to the list of the Session Summarizer whenever an ECC is instantiated by participants or used in a predefined collaboration process. Participants select a list entry that corresponds to an ECC and open it to review an ECC or modify their content.

This ECC also enables participants to generate a report of the Elgar session. Using the Session Summarizer whenever they leave session and to review the generated results. The generation of the report is presented to

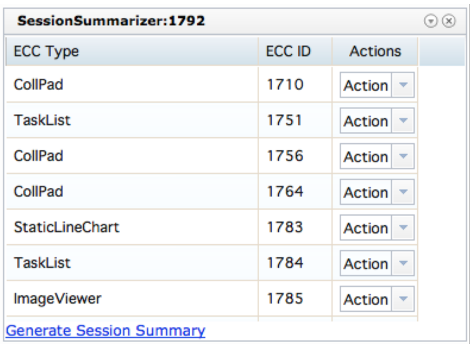


Rating:7bb8

Content	Rating	Action
The engine might generate extra heat.	2	remove
There might be fuel mix in the pipe.	3.5	remove
The pipeline isolation might be torned.	5	remove

Rate Selected Row: ★★★★★

Figure 5.8: Instance of a Rating ECC showing the rating of contributions during a diagnosis task.



SessionSummarizer:1792

ECC Type	ECC ID	Actions
CollPad	1710	Action
TaskList	1751	Action
CollPad	1756	Action
CollPad	1764	Action
StaticLineChart	1783	Action
TaskList	1784	Action
ImageViewer	1785	Action

Generate Session Summary

Figure 5.9: Instance of a Session Summarizer ECC showing all other ECCs used during a diagnosis task.

participants in the format of an HTML page available for download through a web browser if necessary and has the content of all ECCs of a session, listed in the Session Summarizer.

5.2 Data Schema Extensibility of Elastic Collaboration Components

The design of Elgar uses the extensibility software system principle that enables Elgar to accept extensions of its capabilities with minimal efforts to change its basic architecture and to rewrite source code. ECCs are means to distribute functionalities of the system in self-contained components. In this approach ECCs are available to be used in a structured collaboration process (prescribed coordination mechanism) or to be used by team members in their unstructured diagnosis sessions (ad-hoc coordination mechanism).

The extensibility of the functionalities in ECCs also requires the extensibility of database schemas. Traditional relational databases use schemas, which are strongly typed, defined in advance to be used repeatedly as a persistence reference by a software system. However, the extension of such types of strict schemas becomes a problem, as new ECCs cannot change the database schema according to its needs. Therefore, the modification of the schema requires a database expert to change it through a specific language (e.g. SQL) to include the required extensions. In addition, strict schemas require its instances of entities to have the exact same properties.

Flexible schemas represent an alternative to the aforementioned limit-

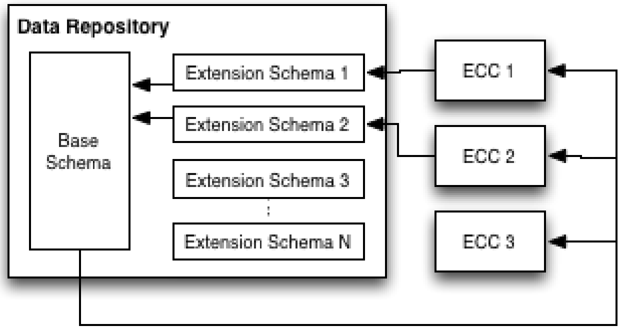


Figure 5.10: A flexible schema of the Data Repository showing its dynamic extension, according to the needs of ECCs.

ations of relational databases as it allows the software to extend schemas dynamically. In Elgar, the use of flexible schemas allows each ECC to extend the data schema automatically, as ECCs include dynamically the data properties required for their use. Figure 5.10 illustrates such a situation. It describes the common schema (base schema) of the Data Repository, used by a set of ECCs. It illustrates that some ECCs only use the base schema (e.g. ECC 3) whereas other ECCs extend the base schema dynamically with properties that are necessary to implement their functionalities.

5.3 Communication Subsystem

The Communication Subsystem is a central part of Elgar, designed and implemented to support collaboration functionalities between the ECCs in different clients. This subsystem of Elgar creates the infrastructure that enables the communication among the clients and the server. The majority of the ECCs implemented in Elgar are designed to support collaboration between participants, requiring the exchange and distribution of messages to synchronize the shared state of ECCs, e.g. CollPad contributions provided by a participant are automatically shared with other clients and stored on the server or an image selected by a participant in the Image Viewer is automatically displayed to the other participants of the same instance of the ECC.

The basis of the communication subsystem of Elgar relies on the *CometD* framework [2], an implementation of the *Bayeux* protocol, that allows peers, in this case clients and Elgar, to establish an asynchronous communication stream over unreliable networks, such a the Internet. The only requirement of the framework is to use the HTTP protocol, implemented by web browsers on the client side and by appropriate libraries of a programming language, e.g. Java, on the server side.

The framework provides a publish-subscribe messaging mechanism by which hosts use the common name of a channel, established by Elgar, to broadcast messages. The subscription mechanism is coupled with *asynchronous callback functions* at the client side of Elgar to avoid *i)* being blocked waiting for a response in a request-response interaction between client and server, and *ii)* being blocked waiting for eventual messages from other connected clients of a diagnosis session. Whenever an ECC at the client side of Elgar is instantiated an ECC subscribes itself to the communication channel and provides a callback function that is executed when a message published in the channel reaches a client. For example, the callback function *processIncomingData* of the *CollPad* is executed after it receives messages published by other CollPad instances in other clients or the server.

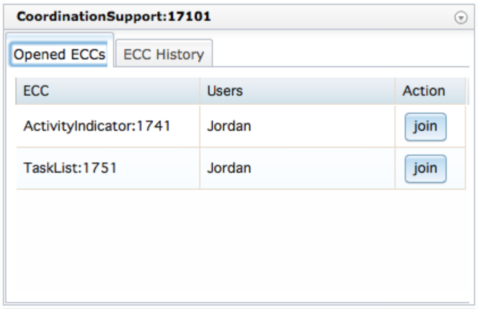
Each ECC includes in the messages their type, the ID of the Elgar session and the ID of the ECC to allow only an specific instance of an ECC type to process incoming messages from the channel. An incoming message addressed to the instance of the CollPad “001” should not be processed by the instance of the CollPad “002” to avoid inconsistencies in the content of each ECC.

The use of CometD made it possible for development to focus on the implementation of the logic of Elgar and its ECCs, rather than on the efforts to develop an additional communication infrastructure over the Web. Therefore, the use of CometD facilitates the implementation of the communication subsystem of Elgar as it provides the necessary means for Elgar to support collaboration among its participants.

5.4 Ad-Hoc Coordination Mechanism

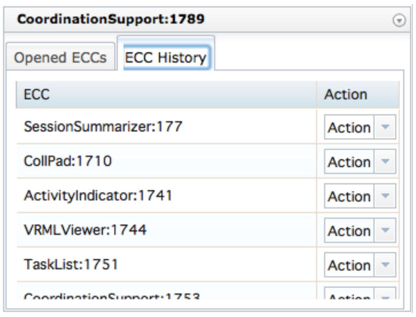
The ad-hoc coordination mechanism is one possible implementation of unstructured coordination support, in which computer-mediated collaboration within a team naturally emerges and is not guided by a predefined process. In the ad-hoc coordination mechanism, participants use workspace aware-

[2] CometD framework, `https://cometd.org/`. Last Access in 03.September.2016



ECC	Users	Action
ActivityIndicator:1741	Jordan	join
TaskList:1751	Jordan	join

Figure 5.11: Instance of the Coordination Support ECC showing the opened ECCs in an Elgar session during a diagnosis task.



ECC	Action
SessionSummarizer:177	Action
CollPad:1710	Action
ActivityIndicator:1741	Action
VRMLViewer:1744	Action
TaskList:1751	Action
CoordinationSupport:1753	Action

Figure 5.12: Instance of the Coordination Support ECC showing the history of all instantiated ECCs in an Elgar session.

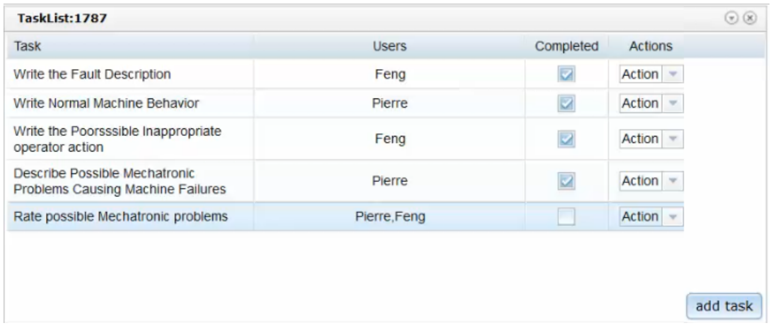
ness information to define their own coordination strategy by the division and synchronization of tasks.

An Elgar session configured with this coordination mechanism provides a list of all available ECCs for instantiation in the user interface of the session, participants choose the most appropriate ECCs for their task. There are two specific ECCs associated automatically to the implementation of this mechanism, the Coordination Support and the Task List, respectively illustrated in the Figure 5.11 and Figure 5.12.

The goal of Coordination Support is to provide, in real time, workspace awareness information about participants and their use of ECCs in Elgar. Figure 5.11 describes the ECCs instantiated in Elgar sessions and the relation of participants of each ECC. Participants may open an ECC described in the list to collaborate with other participants, by clicking the “join” button.

The other part of Coordination Support provides workspace information related to the past, the history of all ECCs used in a specific Elgar session, as illustrated in Figure 5.12.

The goal of the Task List is to support team participants in the division and on the agreement of the necessary tasks to be executed in an Elgar session to accomplish its goals. Participants who participate in an Elgar session can include or remove tasks, and mark tasks as completed. In addition, multiple participants can work at the same tasks simultaneously: participants can describe their involvement in the execution of a task (join a task) or abandon their involvement on the execution of a task (leave a task). Figure 5.13 shows an instance of the Task List.



TaskList:1787

Task	Users	Completed	Actions
Write the Fault Description	Feng	☑	Action ▾
Write Normal Machine Behavior	Pierre	☑	Action ▾
Write the Poorsssible Inappropriate operator action	Feng	☑	Action ▾
Describe Possible Mechatronic Problems Causing Machine Failures	Pierre	☑	Action ▾
Rate possible Mechatronic problems	Pierre,Feng	□	Action ▾

Figure 5.13: Instance of the Task List ECC, showing the distribution of subtasks between two co-workers in machine diagnosis task.

5.5 Prescribed Coordination Mechanism

The prescribed coordination mechanism is one possible implementation of the structured coordination support in which participants plan the sequence of activities to accomplish the goals of a task at design time. In this implementation, participants describe the sequence of activities, their specific instructions and ECCs for each of the activities, such as described in the listing below. Participants can customize:

- The name of the collaboration process to describe its purpose.

- The name and number of collaboration process phases. Collaboration phases are specified by the JSON array named phases. Each array has a minimum of one phase and a variable number of phases.

- The name, instructions of a phase. Each JSON object contained in the phases array and delimited by a pair of brackets (“”), describes the specification of a particular phase. Participants can describe the name of a phase and a text describing the instructions for a particular phase.

- The number, type of ECCs. A phase also has an array of ECCs. This array specifies the number of ECCs that are automatically instantiated for a phase and their defined types. Elgar accepts only types of ECCs that are deployed in the ECC repository of the system. Elgar delivers an error message for unrecognised ECC types that are eventually described by participants.

- The ID and data flows of ECC. Participants can also assign customized IDs to ECCs. Such functionality enables experts to define data flows between ECCs so that the results of an ECC (data source) are reused in another ECC (data target). To create a data flow between ECCs, participants need to define the ID of the data-source ECC and assign it to an optional property of the data-target ECC name *dataInput*.

Listing 5.1: The description of a collaboration process represented in JSON (JavaScript Object Notation).

```
{
  ”name”:”process1”,
  ”phases”:[
    {
      ”name”:”Sensemaking”,
      ”instructions”:”A Wheel Loader shows malfunctioning
          signals. The problem is located on the machine
          clutches. Please, review the chart in this phase
          that shows the readings of oil hydraulic pressure.
          ”,
      ”eccs”:[
        {
          ”id”:”slc1”,
          ”name”:”StaticLineChart”
        }
      ]
    },
    {
      ”name”:”Develop Fault Description”,
      ”instructions”:”In this phase, please, describe
          possible faults related to machine clutches using
          CollPad cp2.”,
      ”eccs”:[
        {
          ”id”:”slc1”,
          ”name”:”StaticLineChart”
        },
        {
          ”id”:”cp2”,
          ”name”:”CollPad”
        }
      ]
    },
    {
```

```
  ”name”:”Describe Possible Mechatronics Problems”,
  ”instructions”:”1) In this phase, please, identify
      possible mechatronics problems that could have
      caused faults in machine clutches and describe
      then in CollPad cp3. /p 2) If there are multiple
      mechatronics problems, use Rating r3 to rank
      problems according to their criticality.”,
  ”eccs”:[
    {
      ”id”:”slc1”,
      ”name”:”StaticLineChart”
    },
    {
      ”id”:”cp3”,
      ”name”:”CollPad”
    },
    {
      ”id”:”r3”,
      ”name”:”Rating”
    }
  ]
},
{
  ”name”:”Describe Possible Downgraded Operation Modes”
      ,
  ”instructions”:”1) In this phase, please, describe
      possible downgraded operation modes for the Wheel
      Loader machine problems using CollPad cp4. /p 2)
      If you identified multiple downgraded operation
      modes, use Rating r4 to rank them prioritising
      less expensive modes for the client.”,
  ”eccs”:[
    {
      ”id”:”slc1”,
      ”name”:”StaticLineChart”
    },
    {
      ”id”:”cp4”,
      ”name”:”CollPad”
    },
    {
      ”id”:”r4”,
      ”name”:”Rating”
    }
  ]
```

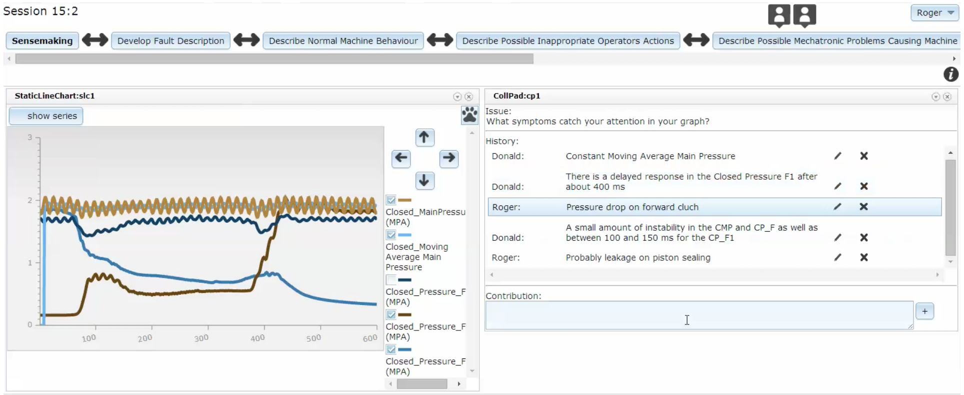
```
    },
    {
      ”name”:”Generate Session Summary”,
      ”instructions”:”Please, generate a session summary
          for this session using SessionSummarizer s1. The
          summary output will have content from all ECCs
          listed in SessionSummarizer s1, used during this
          session.”,
      ”eccs”:[
        {
          ”id”:”s1”,
          ”name”:”SessionSummarizer”
        }
      ]
    }
  ]
}
```

Elgar sessions using the prescribed coordination mechanism require the description of a collaboration process. Elgar interprets a collaboration process and configures the virtual structure of a process in the session as illustrated in Figure 5.14. It illustrates a collaboration process with five collaboration activities. Each activity is represented by a button in which the label is the name of the activity. By clicking the button, Elgar configures the instructions and ECCs related to a collaboration process phase. The label of a button with the font type in bold provides a visual cue, which indicates that team participants work on the activity and have the ECCs and instructions of the activity instantiated. On the top-right side of Figure 5.14, the symbol “*i*” has the current instructions for the selected process activity. The instructions are by default hidden and revealed to participants whenever they hover the mouse pointer on the symbol.

5.6 Discussion

This chapter describes the implementation of the shared workspace system Elgar that supports coordination of distributed teams and execution of collaborative diagnosis tasks.

The design and implementation of ECCs according to the principle of separation of concerns allows continuous development of new functionalities, whenever new requirements are identified. Modularization of ECCs avoids the need to modify source code in other parts of Elgar, as the implementation of each ECC is independent from each other.


Session 15:2
Roger
Sensemaking
Develop Fault Description
Describe Normal Machine Behaviour
Describe Possible Inappropriate Operators Actions
Describe Possible Mechatronic Problems Causing Machine
StaticLineChart:slc1
show series
3
2
1
0
100
200
300
400
500
600
☑ Closed_MainPressu (MPA)
☑ Closed_Moving Average Main Pressure
□ Closed_Pressure_F (MPA)
☑ Closed_Pressure_F (MPA)
☑ Closed_Pressure_F (MPA)
CollPad:cp1
+

Issue:
What symptoms catch your attention in your graph?

History:

Donald:	Constant Moving Average Main Pressure
Donald:	There is a delayed response in the Closed Pressure F1 after about 400 ms
Roger:	Pressure drop on forward cluch
Donald:	A small amount of instability in the CMP and CP_F as well as between 100 and 150 ms for the CP_F1
Roger:	Probably leakage on piston sealing

Contribution:

Figure 5.14: An Elgar session configured with the description of a collaboration process.

Currently, Elgar has a set of ECCs to support collaborative diagnosis tasks, according to the Rectio method. Participants of Elgar are able to analyse machine telemetry data through the Visual Query Editor and the Static Line Chart and to analyse pictures of faulty machine components remotely, through the Image Viewer. Participants describe, as textual contributions through CollPad, hypotheses for potential machine failures and possible actions to tackle the failures. In addition, participants are able to include, in the textual contributions, references of telemetry data or pictures. Finally, participants are also able to rate textual contributions and using the Rating ECC.

Both coordination mechanisms have specific implementation details to support the coordination of distributed teams. The ad-hoc coordination mechanism provides workspace awareness information to its participants through the Coordination Support ECC and the distribution and assignment of task through the Task List. The prescribed coordination mechanism uses collaboration process descriptions to configure the sequence of activities available to the participants and corresponding ECCs and instructions for each activity.

Finally, the communication subsystem, based on the CometD framework [Bordet, 2016], enables ECCs to support collaboration functionalities that involve exchange of data among clients (web browsers) connected to an Elgar session. The communication subsystem provides a common web communication channel by which ECCs , which automatically subscribed to the channel, receive and publish messages to other clients and server asynchronously.

The granularity of the ECCs is a design and implementation decision that enables not only the extensibility of new functionalities for the diagnosis task but also the extensibility of coordination mechanisms. New coordination mechanisms require new implementation efforts, but most of the functionalities offered by ECCs are reusable.

Chapter 6

Experiments

Elgar has been designed to flexibly support different coordination mechanisms. In practice, two coordination mechanisms have been implemented: the prescribed and ad-hoc. The prescribed mechanism provides a pre-defined collaboration process, with pre-configured ECCs and instructions, to structure coordination between participants in a collaborative diagnosis task. The ad-hoc mechanism provides workspace awareness information with which participants are free to structure the coordination they need for their own diagnostic task.

The strengths and weaknesses of the two coordination mechanisms are analysed in this chapter based on their usefulness, usability, workspace awareness information and the quality of collaboration that they support. The chapter presents two different experiments to evaluate the prescribed and ad-hoc mechanisms, illustrated in Figure 6.1.

The machine of the first experiment, manufactured by Hägglunds Drives, is used to grind wood, in this case disposed pallets, into smaller wood chips to reduce the amount of storage space. This type of machine has a high operation time and therefore is expected to have high reliability and availability. Machines have telemetry systems attached to monitor their properties and trigger alarms upon the detection of a degradation.

The machine of the second experiment manufactured by Volvo Construction Equipment is a wheel loader, a type of tractor that has a front-mounted square wide bucket connected to the end of two booms to scoop up loose material from the ground (e.g. dirt, sand or gravel) and move it from one place to another without pushing the material across the ground. These types of machine are often used in construction. They are also deployed with telemetry systems that monitor their properties and trigger alarms upon the

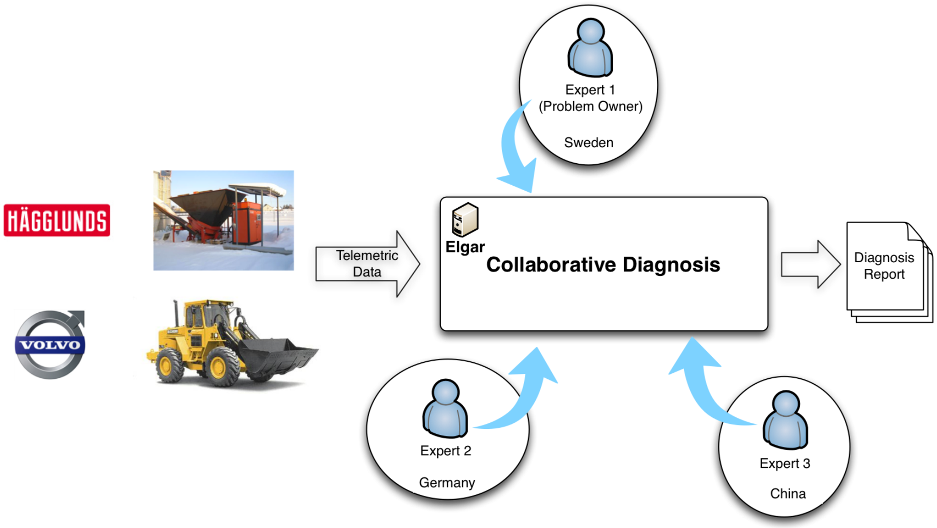


Figure 6.1: Experiments used to evaluate the implemented diagnosis functionalities and coordination mechanisms in Elgar.

detection of a degradation. This experiment, in particular, uses a method to evaluate the quality of collaboration in diagnosis teams.

In both cases, members of diagnosis teams reason about telemetry data using Elgar and one of the two implemented coordination mechanisms to formulate diagnosis about an observed machine, to discuss the data of the machine and, in the end, to generate a diagnosis report describing the root-cause of the failures that occur with a machine, as depicted in Figure 6.1.

6.1 Experiment on Usability - Hägglunds Drives

The goal of this experiment is to evaluate the usability of the functionalities provided by Elgar for collaborative machine diagnosis and coordination. This experiment evaluates the implemented functionalities of Elgar, for the prescribed and ad-hoc coordination mechanisms in a collaborative diagnosis task. This is an experiment to identify potential enhancements of Elgar. Therefore, it uses engineering students as representative users and not diagnosis engineers, who are the intended end-users of Elgar. The results of this experiment are discussed in more detail in [Janeiro et al., 2014].

In this specific experiment, the wood shredder machine, through its micro-controller unit, detects that the temperature of the hydraulic oil that flows through the pumps is increasing, indicating a degradation of the machine as hydraulic oil in high temperatures damages seal components and accelerates oil degradations. The diagnosis team needs to reason about telemetry data of the machine to evaluate its hydraulic oil temperature over time and to report their findings to maintenance teams, to avoid further degradation of the machine or to fix the problem.

The diagnosis task consists specifically of the identification and description of anomalies in the hydraulic oil temperature of pump A and pump B of the hydraulic drive and ranking of their criticality. In this case, a value of a hydraulic oil temperature that exceeds an established threshold is considered to be an anomaly. In this experiment, teams of participants execute a diagnosis task to analyse anomalies and identify and calculate the degree to which a sensor reading exceeds a threshold. In addition, all participants rate the anomalies to establish their relative criticality.

6.1.1 Study Set-up

Teams use the Rectio method to execute this diagnosis task with the ad-hoc and prescribed coordination mechanisms. The adaptation of the method uses some of the activities that are of interest for this experiment, e.g. the use of the functionalities of Elgar and its coordination mechanisms to identify and describe anomalies of the wood shredder machine. The adaptation of the method preserves activities that belong to "data analysis", "hypothesis definition" and "preparation of documentation" phases, and excludes a few activities of the "actions generation" phase, as they are not part of this task. Figure 6.2 illustrates and differentiates the used and not used activities of Rectio for this experiment.

In this experiment, teams use Elgar to execute the diagnosis task with the following ECCs: "Visual Query Editor" (VQE), "CollPad" and "Rating". The VQE has all telemetry data required by participants to execute the diagnosis task of this experiment. Engineers from Hägglunds prepared the data based on an operational machine in a remote site in Sweden and introduced anomalies in the data intentionally to simulate the need of a diagnosis task. The engineers prepared two different types of datasets of hydraulic oil temperature, one to be used with the ad-hoc coordination mechanism and the other to be used with the prescribed coordination mechanism, to avoid learning effects in teams. Through this ECC, participants search for the data of a sensor and visualize it. In CollPad, participants describe and automatically share with each other the anomalies identified in the VQE. For each anomaly, participants may attach information of the source of tele-

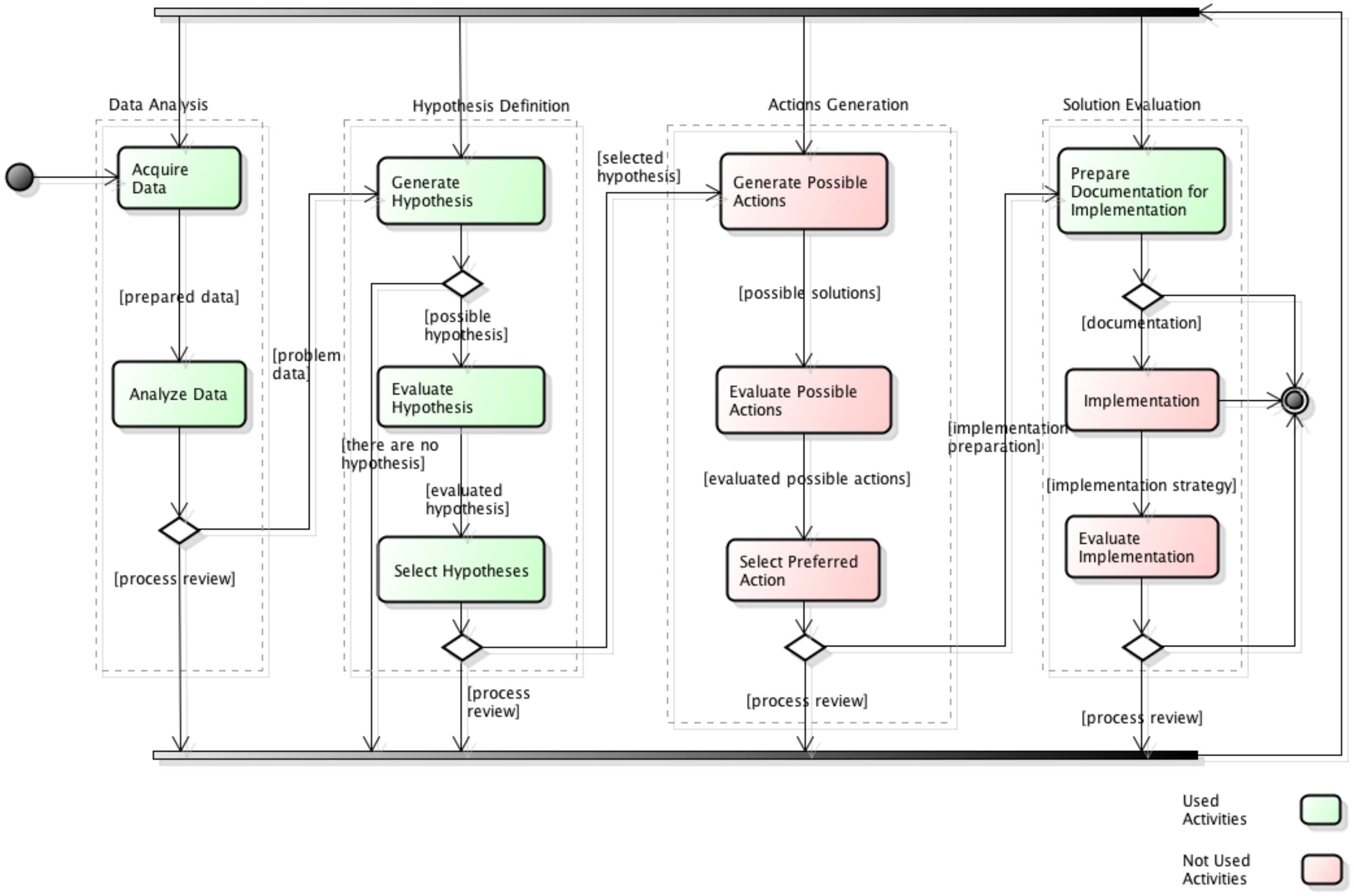


Figure 6.2: Instantiation of the Rectio method for the experiment of Hägglunds.

metry data of the VQE for its contextualization. In the Rating ECC, votes are assigned to each of the anomalies identified in CollPad.

The aforementioned ECCs are used for both coordination mechanisms, the ad-hoc and prescribed, as they support participants in activities that are independent of coordination, e.g. analysis of data and formulation of documentation. In addition to the common ECCs, each coordination mechanism provides specific functionalities during the collaborative diagnosis task. Both coordination mechanisms provide workspace awareness information to participants, the difference between the type of provided information relies on the needs of the two coordination mechanisms to support teams to accomplish the diagnosis task.

In the ad-hoc coordination mechanism, participants have access to more detailed, fine-granular workspace awareness information to support them in the coordination and division of their tasks and on the use of ECCs to accomplish the diagnosis task. In the prescribed coordination mechanism, information is based on the awareness of all activities of the collaboration process that participants follow to conclude the diagnosis task. The prescribed coordination mechanism does not provide the same information offered in the ad-hoc coordination mechanism because the structure and division of collaboration process in activities represents already one possible coordination strategy to approach the diagnosis task, which is absent in the ad-hoc coordination mechanism.

The ad-hoc mechanism does not require previous preparation to support a team in a diagnosis task. However, the prescribed mechanism requires the specification of a collaboration process. The diagnosis activities of the collaboration process for this experiment are based on the Rectio method and on the experience of experts of Hägglunds Drives with wood shredder machines. It describes three process activities with which teams can diagnose the wood shredder machine. The description of the collaboration process, the association to Rectio activities and ECCs is described in Table 6.1.

#	ID	Collaboration Process Activity	Rectio Phases	ECCs
1	CP1	Pressure (B-side) Pump	Data Analysis and Hypotheses Definition	Instruction, VQE, CollPad and Rating

Continued on next page

Continued from previous page

#	ID	Collaboration Process Activity	Rectio Phases	ECCs
2	CP2	Pressure (A-side) Pump	Data Analysis and Hypotheses Definition	Instruction, VQE, CollPad and Rating
3	CP3	Review of Results	Solution Evaluation	Instruction

Table 6.1: Description of the collaboration process designed for the experiment 1 and its association to Rectio phases and used ECCs.

The first two collaboration process activities (CP1 and CP2) use two phases of the Rectio method whereas the last activity (CP3) uses one Rectio method phase. The column "#" of the table describes the order of an activity in the collaboration process. The second column "ID" describes an identification code that represents a collaboration process activity. The column "Collaboration Process Activity" describes the title of an activity. The "Rectio Phases" column describes the phases of the Rectio method used by a collaboration process activity and the column "ECCs" describes the ECCs used by each activity.

In the first activity of the collaboration process (CP1), participants analyse hydraulic oil temperature data that correspond to the pump drain "B", describe anomalies associated to the hydraulic data and rate their criticality. In this process activity, participants execute two phases of Rectio: "data analysis" (RM1) and "hypothesis definition" (RM2). In the RM1 phase, participants have to retrieve the telemetry data that indicates hydraulic oil temperature on pump A and pump B respectively (corresponding to the activity "acquire data" of the Rectio Method) and reason about the data to understand their properties and meaning, e.g. unit of measurement, threshold and peeks (corresponding to the activity "analyse data" of the Rectio Method). In the RM2 phase, participants describe the anomalies identified on the telemetry data (corresponding to the activity "generate hypotheses" of the Rectio Method), evaluate the anomalies that are the most critical (corresponding to the activity "evaluate hypotheses" of the Rectio Method) and eventually select the most critical anomalies (corresponding to the activity "select hypotheses" of the Rectio Method). However, in this phase, for this experiment, teams report anomalies instead of hypotheses. This is necessary because participants selected to participate in this experiment are not professional diagnosis experts, therefore they do not have the knowledge and experience necessary to conjecture hypotheses based on the provided telemetry data of a specific equipment.

The CP1 is configured with one VQE that shows the hydraulic oil temperature data of the pump “A”, one CollPad in which participants describe anomalies that concern this pump drain and one Rating in which participants assign votes to generated anomalies. In addition to the ECCs, each process activity provides instructions that orient participants to achieve the expected goals of the current activity.

In the second activity of the collaboration process (CP2), participants analyse hydraulic oil temperature data that corresponds to the pump drain “A”. This activity uses the same phases of Rectio and the same type and number of ECCs used in the first activity. The difference is the focus of the activity on the pump drain “B” rather than the pump drain “A”.

In the third collaboration process activity (CP3), participants have a last opportunity to review the results of the two previous collaboration process activities, CP1 and CP2. In this process activity, participants execute a part of the last phase of Rectio “solution evaluation”, in which they review and adjust the previous generated findings. Whenever necessary, participants return to a specific activity to adjust the description of anomalies and their ratings. The collaborative diagnosis task is over once the team is satisfied with the results.

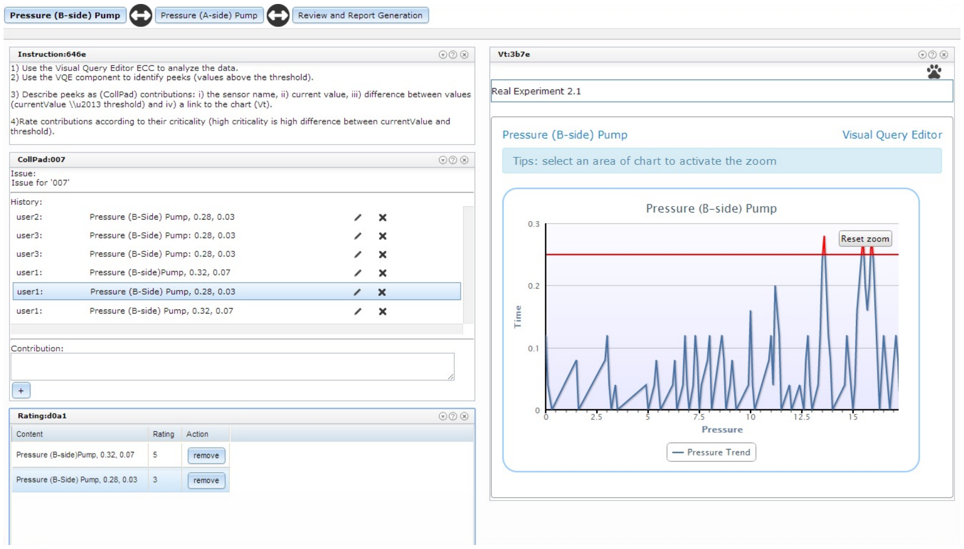
Figure 6.3 illustrates the configuration of Elgar with a collaboration process in the prescribed coordination mechanism.

6.1.2 Experiment Subjects

A total of eighteen participants engaged in this study, fourteen male and four female participants. Fifteen participants have a degree in Computer Science, one in Engineering and Management and two in Automation Engineering. All participants obtained their respective degrees from Sapienza University of Rome. The eighteen participants are divided in six teams of three members each.

6.1.3 Procedure

This experiment follows five phases, as described in the flowchart illustrated by Figure 6.4. The first two phases are training sessions to familiarize teams with Elgar. The last three phases refer to the execution of the diagnosis



Instruction:646e

1) Use the Visual Query Editor ECC to analyze the data.
2) Use the VQE component to identify peeks (values above the threshold).

3) Describe peeks as (CollPad) contributions: i) the sensor name, ii) current value, iii) difference between values (currentValue \\u2013 threshold) and iv) a link to the chart (Vt).

4)Rate contributions according to their criticality (high criticality is high difference between currentValue and threshold).

CollPad:007

Issue:
Issue for '007'

History:

user2:	Pressure (B-Side) Pump, 0.28, 0.03
user3:	Pressure (B-Side) Pump: 0.28, 0.03
user3:	Pressure (B-Side) Pump: 0.28, 0.03
user1:	Pressure (B-side)Pump, 0.32, 0.07
user1:	Pressure (B-Side) Pump, 0.28, 0.03
user1:	Pressure (B-side) Pump, 0.32, 0.07

Contribution:

Rating:d0a1

Content	Rating	Action
Pressure (B-side)Pump, 0.32, 0.07	5	remove
Pressure (B-Side) Pump, 0.28, 0.03	3	remove

Vt:3b7e

Real Experiment 2.1

Figure 6.3: Configuration of Elgar in the prescribed coordination mechanism with a collaboration process. The first collaboration process activity is selected and shows instructions, an instance of the CollPad, Rating and VQE/VT ECCs.

task. There were no time limits for the execution of the training phases of the execution of the diagnosis task.

There are two training sessions per team. In phase 1, team participants use the ad-hoc mechanism and in phase 2, the prescribed mechanism. During the training sessions, participants are co-located and have to describe anomalies for two simulated sensors of a wood shredder machine. An instructor is also present to support them in use of Elgar and the particulars for each coordination mechanism.

In phase 1, the instructor instructs participants on creating and sharing ECCs dynamically, based on workspace awareness information. In phase 2, the instructor instructs participants in following a collaboration process that contains pre-defined instructions and ECCs.

For this experiment, each participant is located in a separate room, equipped with a notebook, connected to an Internet-based voice channel for communication purposes. First, the instructor requests the team to generate a report for two sensors of a wood shredder machine using the ad-hoc coordination mechanism (phase 3). Then, the instructor requests the team to generate another report for other readings of machine sensors using the

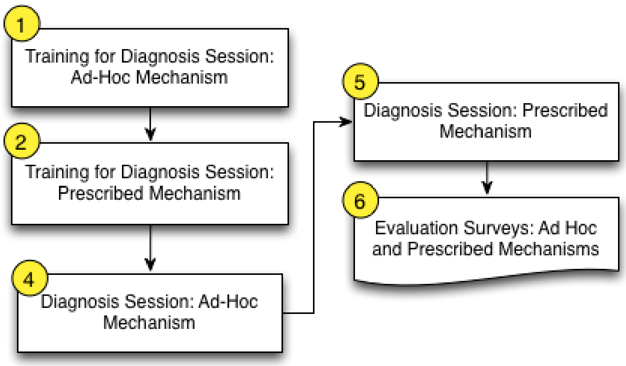

1
Training for Diagnosis Session: Ad-Hoc Mechanism
2
Training for Diagnosis Session: Prescribed Mechanism
4
Diagnosis Session: Ad-Hoc Mechanism
5
Diagnosis Session: Prescribed Mechanism
6
Evaluation Surveys: Ad Hoc and Prescribed Mechanisms

Figure 6.4: A flowchart of the experiment with coordination mechanisms implemented in Elgar.

prescribed coordination mechanism (phase 4). In the end of the experiment, the instructor requests teams to answer two surveys to assess their perception about used coordination mechanisms (phase 5).

6.1.4 Research Method

This experiment uses a questionnaire to measure the usefulness of Elgar. Table 6.2 describes three sections in the questionnaire: *i)* evaluation of collaborative functionalities, *ii)* evaluation of the ad-hoc coordination mechanism and *iii)* evaluation of the prescribed coordination mechanism. The first section evaluates the usefulness of Elgar with regard to the collaborative functionalities implemented by Elgar. The second section of the questionnaire evaluates the ad-hoc coordination mechanism. It is based on the workspace awareness information provided by Elgar to support spontaneous coordination among participant in a diagnosis task. The third section evaluates the prescribed coordination mechanism. It evaluates the use of collaboration processes to guide participants throughout the diagnosis task and to coordinate their actions.

The items are based on an extensive questionnaire to evaluate collaboration technology that embraces the three aspects of collaboration, communication, coordination and cooperation [Lee, 2007]. The questionnaire has two purposes: *(i)* it is either used by software developers of evolving collaboration technology prototypes to collect feedback from participants or *(ii)* it is used as an instrument to evaluate and compare the support of collaboration technologies in collaborative tasks.

#	**Question**
i01	I could express my intentions to other participants.
i02	I could understand the intentions of other participants.
i03	I could interact with other participants to accomplish the task.
i04	I could identify contributions generated by other participants.
i05	I could identify contributions generated by other participants easily.
i06	I could modify contributions created by other participants.
i07	I knew the progress of the task.
i08	I had access to the necessary data to generate contributions with other participants.
	Evaluation of the Ad-hoc Coordination Mechanism
i09	I used workspace awareness information to be aware of if other participants were involved in a task.
i10	I used workspace awareness information to join other participant(s) to work on a task.
i11	I used workspace awareness information to take over on a task in which no other participant was working.
i12	I used workspace awareness information to divide tasks among other participants.
i13	I used workspace awareness information to know the progress of the task.
	Evaluation of the Prescribed Coordination Mechanism
i14	Elgar coordinated all tasks for the participants.
i15	The process helped me to accomplish the task.
i16	I had to coordinate the division of tasks with other participants.
i17	It was important that all participants were working together in one process phase at a time.
i18	It was important that Elgar prescribed the tools that I needed to use in each process phase.
i19	I could follow process phases in Elgar.

Table 6.2: Items of questionnaire to evaluate the usability of collaboration functionalities and coordination mechanisms of Elgar.

The questionnaire outlined in the Table 6.2 describes a set of items that assesses the usefulness of collaboration functionalities implemented in Elgar and on the usability of its coordination mechanisms, in this experiment. This questionnaire has a total of nineteen items: eight items that question participants about collaborative functionalities, five items that question participants about the ad-hoc coordination mechanism and six items that

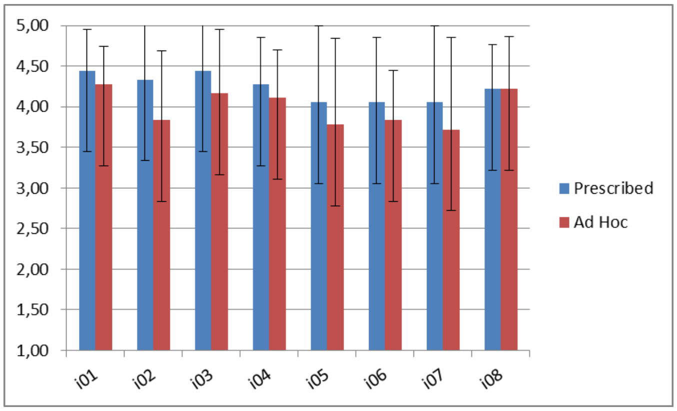


Figure 6.5: Evaluation of the collaboration functionalities implemented in Elgar, independent of the used coordination mechanism.

question about the prescribed coordination mechanism.

All questionnaire items are formulated as positive valence questions. Possible answers for each item are based on a Likert scale and range between: strongly disagree, disagree, neutral, agree and strongly agree. In addition, the questionnaire has two open-ended questions to explore the experience of participants with a particular coordination mechanism, e.g. "*Describe the positive aspects of using the ad-hoc coordination mechanism to accomplish the task.*" and "*Describe the negative aspects of using the ad-hoc coordination mechanism to accomplish the task.*".

6.1.5 Results

The main result of this experiment is to understand the usefulness of the functionalities of Elgar and both coordination mechanisms, during a diagnosis task. All eighteen participants answered the questionnaire, after the execution of the diagnosis task.

Figure 6.5 shows the average of each questionnaire item for the eighteen participants with regard to the collaborative functionalities implemented in Elgar. The overall observation of Figure 6.5 shows the usefulness of implemented functionalities of Elgar. In particular, the item "i01" shows the higher averages among all questionnaire items for the section. The high averages demonstrate the capability of Elgar to allow participants to express their intentions to other participants.

The ad-hoc and prescribed coordination mechanisms are considered to be equivalent for all questionnaire items. The distinction between the co-

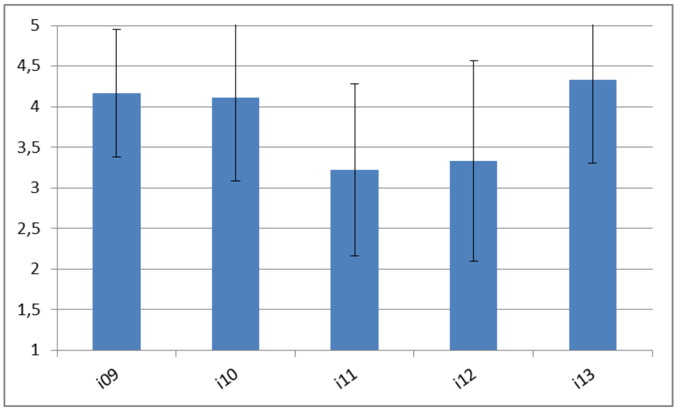


Figure 6.6: Usefulness of the workspace awareness provided by the ad-hoc coordination mechanism in Elgar.

ordination mechanisms for an item is only considered when the difference of their averages is more than one point, on a scale from one to five. Note that the standard deviation for the questionnaire items is high, which demonstrates high dispersion of the answers of participants. Therefore, such a situation makes it difficult to draw conclusions from the analysis of Figure 6.5, as teams members do not agree on convergent answers.

Figure 6.6 shows the results for questionnaire items that concern the adhoc coordination mechanism. As illustrated in Figure 6.6, the items "i09", "i10" and "i13", show the usefulness of Elgar with regard to the workspace awareness information provided by this coordination mechanism. Through the provided workspace awareness, participants are aware of other members that are involved in the diagnosis task ("i09"). Participants also use the information to provide help to other participants in diagnosis subtasks ("i10") whenever required. Finally, participants use the information to follow the progress of a diagnosis task until its completion. Figure 6.6 confirms the usefulness of Elgar, as most of eighteen participants agree with the positive valence questions.

However, Figure 6.6 also shows two questionnaire items that have an average usefulness, the items "i11" and "i12". The item "i11" demonstrates that Elgar does not provide enough workspace awareness information with regard to pending subtasks. The other item that does not capture a high usefulness among participants is "i12". It indicates that the workspace awareness information provided by Elgar is not effective to support delegation of subtasks among the participants.

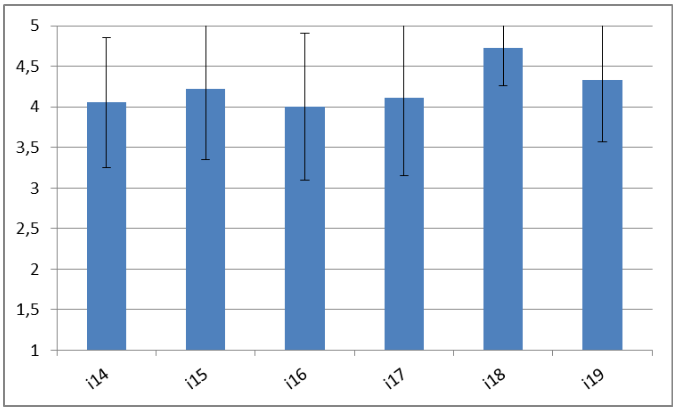


Figure 6.7: Usefulness of the coordination provided by collaboration processes in the prescribed coordination mechanism in Elgar.

Note that the standard deviation for the questionnaire items is high, which demonstrates dispersion of the answers of participants. Therefore, such a situation does not allow to draw conclusions from the analysis of Figure 6.6, as the answers of teams members do not converge.

Figure 6.7 shows the results for questionnaire items that concern the prescribed coordination mechanism. In general, as illustrated in Figure 6.7, the coordination proposed by a prescribed collaboration process is effective to support teams.

The answers of this questionnaire, however show a high standard deviation, meaning a dispersion of the answers of participants. It is difficult to draw conclusions on most of the questionnaire items as they do not converge. One exception concerns the item “i18”, with regard to the usefulness of the prescribed collaboration process, for which the standard deviation is less than one point in the scale. It indicates a convergence point in which all participants acknowledge the importance of the prescription of the collaboration process in the diagnosis task.

6.1.6 Discussion

The overall analysis of the first chart, illustrated in Figure 6.5, shows the usefulness of the collaboration functionalities implemented in Elgar. One particular observation with regard to item “i02”, measures the understanding of the intentions of other participants. In this situation, the understanding of the intention refers to the contributions provided by each participant to identify temperature values above the predefined threshold. Participants that use the prescribed coordination mechanism better understand more of the intentions of others in comparison to participants who used the ad hoc mechanism, as illustrated in Figure 6.5. One possible explanation is that participants' contributions in the prescribed mechanism are contextualised by the structure the prescribed mechanism provides in contrast to the unstructured context, as in the ad-hoc mechanism. Therefore, it becomes easier for participants who use the prescribed mechanism to understand communicated intentions of the contributions of each other.

The analysis of the Figure 6.6 shows the usefulness of workspace awareness information provided by Elgar in the ad-hoc coordination mechanism. However, the two median items "i11" and "i12" demonstrate that the coordination mechanism requires improvements to provide additional awareness information that describes the division of subtasks among participants in a diagnosis session and their involvement in each of the subtasks.

Although the averages of questionnaire items for the prescribed coordination mechanism are only slightly higher than the averages for ad hoc, such a difference indicates a preference. Participants who attended these diagnosis sessions with Elgar are not professional diagnosis experts, they understand the domain but do not have professional experience. The collaboration process in the prescribed coordination mechanism represents a support for participants. It provides a structure that serves as a general guideline for teams to achieve the goals of the diagnosis task, ECCs and instructions. Therefore, the slightest higher preference of the prescribed mechanism in this experiment is influenced by its additional support to tackle a diagnosis task. Such support from Elgar helps participants who are not experienced in diagnosis to accomplish their task.

6.2 Experiment on Coordination Mechanisms - Volvo Construction Equipment (VCE)

The goal of this experiment is to evaluate implemented functionalities using Elgar, and to evaluate the usefulness of the implemented prescribed and ad-hoc coordination mechanisms in a collaborative diagnosis task.

This experiment involves the diagnosis of machine telemetry data that reproduces degrading behaviour. A team of engineers prepared telemetry data based on unusual symptoms observed on operational wheel loaders. This telemetry data describes unexpected machine behaviour on the clutch subsystem (second clutch) by insufficient hydraulic oil pressure and occasional slip warnings to the machine operator, caused by a discrete and nonidentified hydraulic oil leakage. In this experiment, teams of participants execute a diagnosis task to identify unexpected machine behaviour on the clutch subsystem, generate actions to overcome this symptom and prepare the documentation to implement the actions. This experiment intends to gain insights about the collaboration of experts using Elgar in diagnosis tasks. Therefore, the participants in this experiment are diagnosis engineers, who are the intended end-users of Elgar.

6.2.1 Study Set-up

Diagnosis teams use the Rectio method to execute this diagnosis task. The method preserves all activities involved in “hypothesis definition” and “actions generation” phases and most of the activities in “data analysis” and “solution evaluation” phases. The activities “acquire data”, “implementation” and “evaluate implementation” are disregarded from the instantiation of Rectio as they are not required in this experiment. As described above, the data was prepared previously for engineers according to the degrading behaviour of the machine and did not need participants to search or acquire telemetry data, therefore the activity “acquire data” was not used. In addition, the goal of the experiment was to determine the root-causes of the machine and not test approaches, e.g. replacing small electrical components such as fuses, to solve them. Therefore, participants did not discuss approaches, in the activity “implementation”, or evaluate them in the activity “evaluate implementation”. Figure 6.8 illustrates and differentiates the used and not used activities of Rectio for this experiment.

An outcome of the evaluation of the previous experiment indicates that teams, using the ad-hoc coordination mechanism, needed support for the documentation and management of subtasks, providing awareness of the involvement of participants in both. Based on the evaluation of the previous experiment, a new ECC named “Task Manager” has been added to Elgar. This ECC enables participants to describe subtasks of a diagnosis session and the ownership of participants in its execution, e.g. a participant takes responsibility to execute a subtask alone or different participants execute the subtask together.

In this experiment, teams use Elgar to execute the diagnosis task with the following ECCs: “LineChart”, “CollPad”, “Rating”, “TaskManager” and “SessionSummarizer”. The LineChart stores all telemetry data required by participants to execute the diagnosis task of this experiment. Through this ECC, participants visualize different types of sensors and their readings. Telemetry data used in this experiment has readings from five different types of sensors that measure hydraulic oil pressure. In CollPad, participants de-

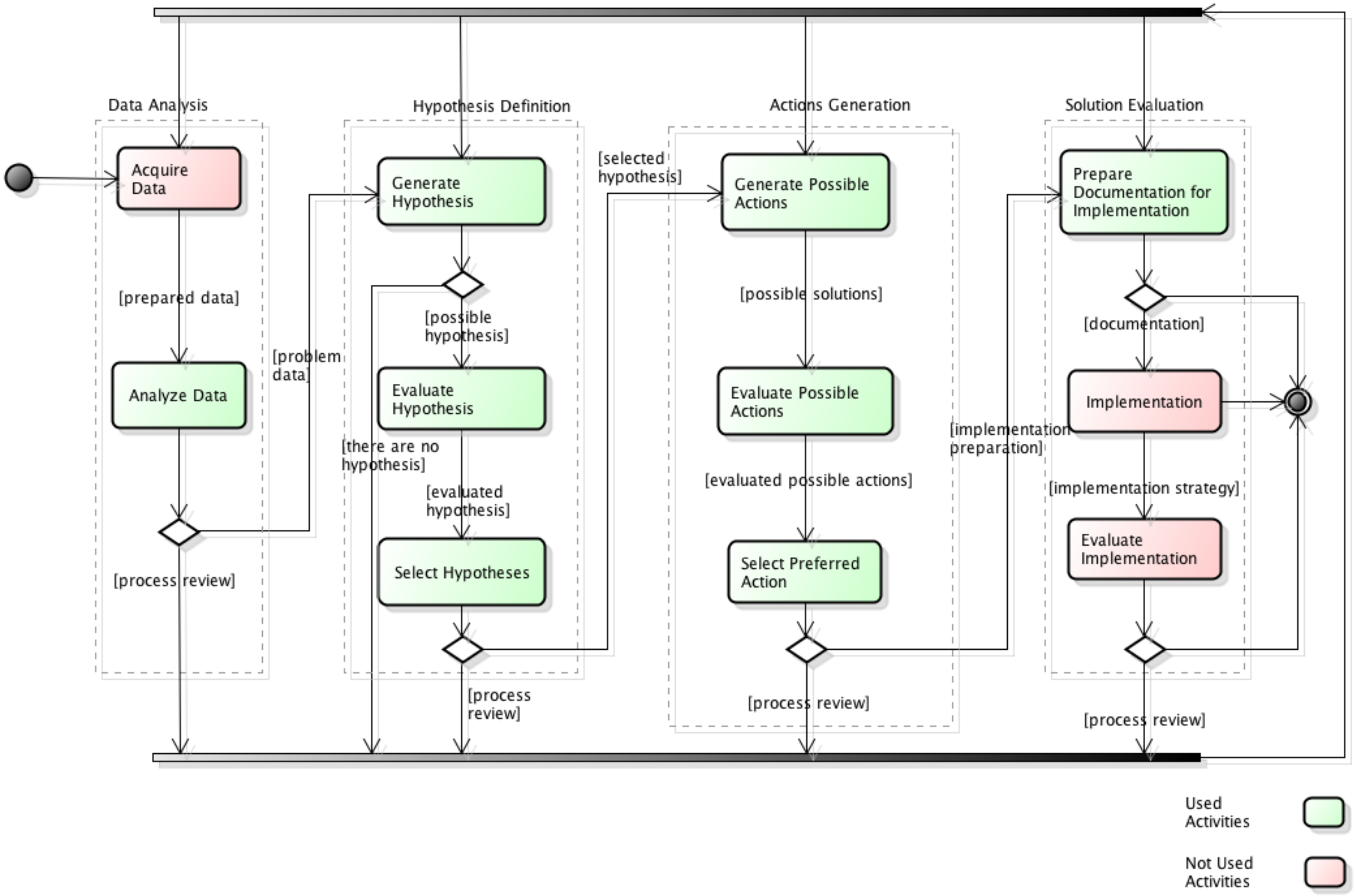
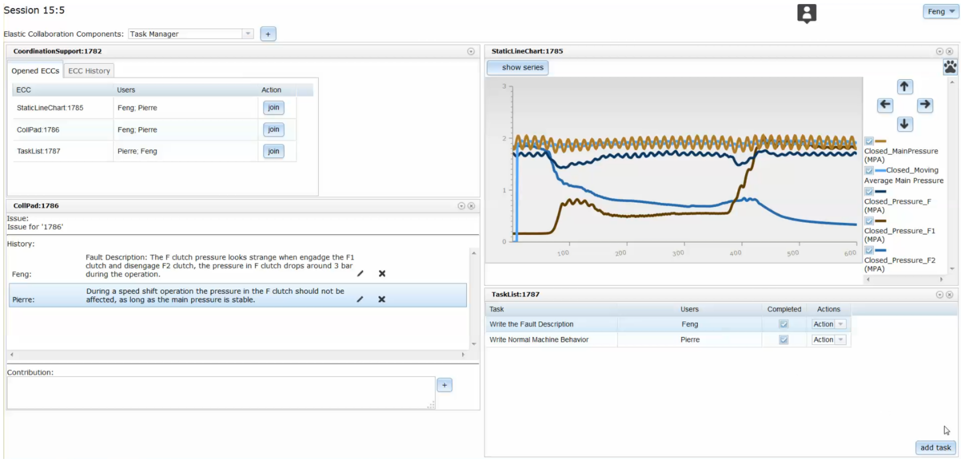


Figure 6.8: Instantiation of the Rectio method for the experiment of Volvo Construction Equipment.



Session 15:5

Elastic Collaboration Components: Task Manager

CoordinationSupport:1782

Opened ECCs | ECC History

ECC	Users	Action
StaticLineChart:1785	Feng; Pierre	join
CollPad:1786	Feng; Pierre	join
TaskList:1787	Pierre; Feng	join

CollPad:1786

Issue:
Issue for '1786'

History:

Feng:	Fault Description: The F clutch pressure looks strange when engadge the F1 clutch and disengage F2 clutch, the pressure in F clutch drops around 3 bar during the operation.
Pierre:	During a speed shift operation the pressure in the F clutch should not be affected, as long as the main pressure is stable.

Contribution:

StaticLineChart:1785

TaskList:1787

Task	Users	Completed	Actions
Write the Fault Description	Feng	☑	Action
Write Normal Machine Behavior	Pierre	☑	Action

Figure 6.9: Configuration of Elgar in the ad-hoc coordination mechanism with instances of the CoordinationSupport, StaticLineChart, CollPad and TaskManager ECCs.

scribe and automatically share with each other contributions that are either hypotheses, identified during the analysis of the LineChart, or actions that tackle these hypotheses. For each contribution, participants may reference an instance of the LineChart and its status to contextualize a contribution. In the Rating ECC, participants reuse all generated contributions from CollPads and assign ratings to each of them. These ECCs are used for both coordination mechanisms, the ad-hoc and prescribed, as they support participants in activities that are independent of coordination, such as analysis of data and formulation of documentation.

In addition to the common ECCs, each coordination mechanism provides specific functionalities during the diagnosis task. The ad-hoc mechanism provides workspace awareness information about all participants connected in an Elgar session and the ECCs they use. In addition, it also provides information about the agreed subtasks related to the main diagnosis task of the session, their distribution among participants and the status of their completion. Figure 6.9 illustrates the configuration of Elgar with the ad-hoc coordination mechanism.

The prescribed mechanism provides a collaboration process used in Elgar for a diagnosis task, in which participants narrow down the root-causes that describe anomalies in the clutch subsystem. The diagnosis activities of the collaboration process are designed for this experiment based on the Rectio method, on the *could-be-should-be* collaboration technique [Briggs and de Vreede, 2009] and on the diagnosis experience of two experts of Volvo Construction Equipment with wheel loader machines.

The collaboration process describes eight activities with which teams can diagnose wheel loaders. The description of the process, the association to Rectio activities and ECCs is described in Table 6.3.

#	ID	Collaboration Process Activity	Rectio Phases	ECCs
1	CP1	Sense Making.	Data Analysis and Hypotheses Definition	LineChart, CollPad and Instruction
2	CP2	Failure Description. Description of faulty components and explanation of the smallest replacement components that fix a symptom.	Data Analysis, Hypotheses Definition and Solution Generation	LineChart, CollPad and Instruction
3	CP3	Normal/Abnormal Machine Behaviour. Description of anomalies represented by a deviation in telemetry data.	Data Analysis and Hypotheses Definition	LineChart, CollPad and Instruction
4	CP4	Possible Inappropriate Operators Actions. Description of actions of an operator that cause a symptom.	Data Analysis and Hypotheses Definition	LineChart, CollPad and Instruction
5	CP5	Possible Mechatronic Problems Causing Machine Failures. Description of symptoms caused by software, mechanical components or their interaction.	Data Analysis and Hypotheses Definition	LineChart, CollPad and Instruction
6	CP6	Possible Downgraded Operation Modes. Description of possible downgraded modes that limit the risk of secondary damages, e.g. increased fuel consumption, lower load, lower torque or limited usage of specific gears.	Solution Generation	LineChart, CollPad and Instruction
7	CP7	Possible Improvements for Failure Detection. Description of improvements for failure detection.	Actions Generation	LineChart, CollPad and Instruction

Continued on next page

Continued from previous page

#	ID	Collaboration Process Activity	Rectio Phases	ECCs
8	CP8	Review of Results and Generation of Report.	Solution Evaluation	SessionSummary and Instruction

Table 6.3: Description of the collaboration process designed for experiment 2 and its association to Rectio phases and used ECCs.

The first seven collaboration process activities (CP1, CP2, CP3, CP4, CP5, CP6, CP7) use two phases of the Rectio method whereas the last activity (CP8) uses one Rectio method phase. The column "#" of the table describes the order of an activity in the collaboration process. The second column "ID" describes an identification code that represents a collaboration process activity. The column "Collaboration Process Activity" describes the title of an activity. The "Rectio Phases" column describes the phases of the Rectio method used by a collaboration process activity and the column "ECCs" describes the ECCs used by each activity.

In the first activity of the process (CP1), participants analyse the telemetry data and describe initial evident anomalies. In CP1, participants execute two phases of Rectio "data analysis" and "hypotheses definition". In "data analysis", participants have to reason about telemetry data to understand their properties and meaning, e.g. the pressure on the first clutch is below average (corresponding to the activity "analyse data" of the Rectio Method). In the "hypotheses definition" phase, participants describe hypotheses identified during the analysis of telemetry data (corresponding to the activity "generate hypotheses" of the Rectio Method), evaluate the generated hypotheses based on current telemetry data (corresponding to the activity "evaluate hypotheses" of the Rectio Method) and select the hypotheses that most likely reflect anomalies indicated in the telemetry data (corresponding to the activity "select hypotheses" of the Rectio Method). The activity is supported by two ECCs: one LineChart and one CollPad. The LineChart ECC shows the readings of hydraulic oil pressure for five sensors and in the CollPad participants describe the most evident anomalies in the telemetry data.

In the second activity of the process (CP2), participants analyse the telemetry data and describe components of the machine that could cause detected symptoms to happen. In CP2, participants execute the phases "data analysis", "hypotheses definition" and "solution generation" of the Rectio method. In “data analysis”, participants search for associations between possible faulty machine components based on the telemetry data (corresponding to the activity “analyse data” of the Rectio Method). In the “hypotheses definition” phase, participants describe possible faulty components (corresponding to the activity “generate hypotheses” of the Rectio Method), evaluate faulty components based on current telemetry data (corresponding to the activity “evaluate hypotheses” of the Rectio Method) and select the hypothesis that most likely corresponds to anomalies indicated in the telemetry data (corresponding to the activity “select hypotheses” of the Rectio Method). In the “solution generation”, participants propose different replacements to fix a faulty component (corresponding to the activity “generate possible actions” of the Rectio Method), evaluate the required resources used to replace a faulty component (corresponding to the activity “evaluate possible actions” of the Rectio Method) and select the replacement that requires the minimum resources to fix a faulty component. The activity is supported by five ECCs: one LineChart and four CollPads. The LineChart ECC shows the readings of hydraulic oil pressure for five sensors. One CollPad in which participants provide hypotheses of faulty components that could contribute to a symptom; one CollPad in which participants provide hypotheses of faulty components that should contribute to a symptom; one CollPad in which participants provide hypotheses for the replacement of the smallest components that could fix the symptom; and one CollPad in which participants provide hypotheses for the replacement of the smallest components that should fix the symptom.

In the third activity of the process (CP3), participants analyse the telemetry data and describe expected anomalies in the telemetry data that contribute to abnormal machine behaviour, e.g. driving a wheel loader in extreme temperatures or rugged terrain may cause an expected symptom to occur. In CP3, participants execute the phases “data analysis” and “hypotheses definition” of the Rectio method. In “data analysis”, participants reason about normal and abnormal machine behaviours based on the telemetry data (corresponding to the activity “analyse data activity” of the Rectio Method). In the “hypotheses definition” phase, participants describe abnormal behaviours of the wheel loader identified during the analysis of telemetry data (corresponding to the activity “generate hypotheses” of the Rectio Method), evaluate the difference between abnormal behaviours and normal behaviours that occur under extreme operational conditions (corresponding to the activity “evaluate hypotheses” of the Rectio Method) and select the abnormal behaviours that correspond to anomalies in the telemetry data (corresponding to the activity “select hypotheses” of the Rectio

Method). The activity is supported by two ECCs: one LineChart that shows the readings of hydraulic oil pressure for five sensors and one CollPad in which participants provide hypotheses of situations that influence the behaviour of a machine.

In the fourth activity of the process (CP4), participants describe according to their experience possible inappropriate behaviour of operators of a machine that cause certain symptoms. In CP4, participants execute the phases "data analysis" and "hypotheses definition" of the Rectio method. In "data analysis", participants reason about inappropriate actions of operators that could damage a wheel loader (corresponding to the activity "analyse data" of the Rectio Method). In the "hypotheses definition" phase, participants describe possible inappropriate actions based on telemetry data (corresponding to the activity "generate hypotheses" of the Rectio Method), evaluate the relationship between the occurrence of an inappropriate action and anomalies in the telemetry data (corresponding to the activity "evaluate hypotheses" of the Rectio Method) and select actions that can potentially damage a machine (corresponding to the activity "select hypotheses" of the Rectio Method). The activity is supported by two ECCs: one LineChart that shows the readings of hydraulic oil pressure for five sensors and one CollPad in which participants provide hypotheses of possible inappropriate actions of a machine operator.

In the fifth activity of the process (CP5), participants describe according to their experience possible symptoms caused by software, mechanical components or their interaction that could cause the machine symptoms reported. In CP5, participants execute the phases "data analysis" and "hypotheses definition" of the Rectio method. In "data analysis", participants reason about mechatronic problems that can cause machine failures (corresponding to the activity "analyse data" of the Rectio Method). In the "hypotheses definition" phase, participants describe possible causes for mechatronic problems identified in telemetry data (corresponding to the activity "generate hypotheses" of the Rectio Method), evaluate the influence of causes in machine failures (corresponding to the activity "evaluate hypotheses" of the Rectio Method) and select a main cause for the mechatronics failures can potentially damage a machine (corresponding to the activity "select hypotheses" of the Rectio Method). The activity is supported by two ECCs: one LineChart that shows the readings of hydraulic oil pressure for five sensors and one CollPad in which participants provide hypotheses of possible mechatronic problems.

In the sixth activity of the process (CP6), participants describe according to their experience possible downgraded modes that would limit the risk of machine degradation. In CP6, participants execute the phase “solution generation” of the Rectio method. In this phase, participants describe possible downgraded machine modes (corresponding to the activity “generate possible actions” of the Rectio Method), evaluate the efficiency of downgraded modes to avoid secondary machine damages (corresponding to the activity “evaluate possible actions” of the Rectio Method) and choose a machine downgraded mode to recommend (corresponding to the activity “select preferred action” of the Rectio Method). The activity is supported by two ECCs: one LineChart that shows the readings of hydraulic oil pressure for five sensors and one CollPad in which participants provide hypotheses of possible downgraded machine modes.

In the seventh activity of the process (CP7), participants describe according to their experience possible improvements for automatic failure detection in the context of this experiment. In CP7, participants execute the phase “solution generation” of the Rectio method. In this phase, participants describe possible improvements in the algorithms of machine failure detection according to their experience (corresponding to the activity “generate possible actions” of the Rectio Method), evaluate the suitability of their implementation based on required resources to change the algorithms in the machine (corresponding to the activity “evaluate possible actions” of the Rectio Method) and choose the most suitable machine improvement (corresponding to the activity “select preferred action” of the Rectio Method). The activity is supported by two ECCs: one LineChart that shows the readings of hydraulic oil pressure for five sensors and one CollPad in which participants provide possible improvements for failure detection.

In the eighth activity of the process (CP8), participants have the possibility to review the generated information and create a report, based on the Volvo Case Report, for the implementation team. The activity is supported by the SessionSummary ECC by which participants open and modify previous generated contributions or ratings, or request directly the creation of the report.

All eight process activities are also configured with contextualized instructions for each activity. The instructions describe the goal and expected results for each activity.

6.2.2 Experiment Subjects

A total of twelve diagnosis experts took part in this experiment. All participants belong to the global diagnosis team of VCE. They are worldwide experts in control and transmission technology.

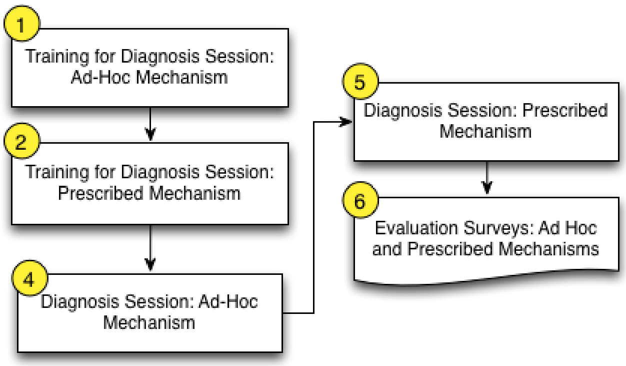

1
Training for Diagnosis Session: Ad-Hoc Mechanism
2
Training for Diagnosis Session: Prescribed Mechanism
4
Diagnosis Session: Ad-Hoc Mechanism
5
Diagnosis Session: Prescribed Mechanism
6
Evaluation Surveys: Ad Hoc and Prescribed Mechanisms

Figure 6.10: A flowchart of the experiment with coordination mechanisms implemented in Elgar.

The twelve participants formed six distributed teams. The identification of each team corresponds to the combination of the initial of their names. The identification of the six teams, the initials of the participants and their expertise are described as follows:

- Team P&F: hydraulic engineer (PT) and mechanical engineer (FL),
- Team J&D, chemical engineer (JA) and mechanical engineer (DB),
- Team J&M, mechanical engineer (JL) and computer science (MN),
- Team R&D, no official degree (RL) and mechanical engineer (DM),
- Team R&L, no official degree (RJ) and mechanical engineer (LA), and
- Team M&O, mechanical engineer (MA) and mechanical engineer (OO)

6.2.3 Procedure

This experiment follows five phases, as described in the flowchart illustrated by Figure 6.10. The first two phases are training sessions to familiarize teams with Elgar. The last three phases refer to the execution of the diagnosis task. Participants are instructed that the duration of training phases is of a maximum of two hours, whereas the execution of the diagnosis task has the maximum duration of one hour.

In phase 1, team participants use the ad-hoc mechanism and in phase 2, the prescribed mechanism. During the training sessions, participants are co-located and have to describe anomalies above a threshold for a predefined chart configured in the “LineChart” ECC. An instructor is also present to support them in the use of the functionalities and coordination mechanisms of Elgar.

In phase 1, the instructor trains participants on the creation of ECCs, in particular the “TaskManager”, added in Elgar after the previous experiment, and the use of shared data, based on workspace awareness information. In phase 2, the instructor trains participants in following a collaboration process that contains pre-defined activities, instructions and ECCs.

For this experiment, each participant is located in a separate room, provided with a notebook, connected to a Internet-based voice channel for communication purposes. First, the instructor requests the team to identify the root-causes of the anomalies of a wheel loader machine using the ad-hoc coordination mechanism (phase 3). Then, the instructor requests the team to to identify the root-causes of the machine anomalies for other readings of the wheel loader using the prescribed coordination mechanism (phase 4). In the end of the experiment, the instructor requests teams to answer two surveys to assess their perception about the use of coordination mechanisms (phase 5).

6.2.4 Research Methods

This experiment uses four different methods to evaluate the use of Elgar in a diagnosis task, based on a scenario designed with Volvo Construction Equipment. An evaluation approach that combines different established evaluation methods was designed to assess collaboration aspects in the interaction among participants belonging to a team. The first method evaluates the usefulness of Elgar in supporting the interaction of participants in their virtual meeting to execute the established diagnosis task. The second method evaluates the usefulness of Elgar in providing workspace awareness information to participants. The third method evaluates the usability of Elgar in the established diagnosis task. The fourth method evaluates the quality of collaboration of teams that use Elgar during the established diagnosis task. These methods combined are designed to evaluate the functionalities that Elgar provides to participants and not to measure their individual motivation to use the system.

The first three methods are based on questionnaires. After the execution of the diagnosis task, a session mediator requests participants to answer the three questionnaires. The fourth evaluation method is based on judging the observation of the screencasts of each team and the transcription of their dialogues during the execution of the diagnosis tasks.

Workspace-based Meeting Assessment Method

This study uses an adaptation of the meeting assessment questionnaire [Briggs et al., 2003a] to evaluate the usefulness of Elgar supporting participants in diagnosis tasks. The meeting assessment questionnaire has originally fifteen questions. It captures the effectiveness of teams working together and the results obtained by the team in the end of the meeting.

#	**Question**
i1	I got (less/more) from the shared workspace system support than I had anticipated.
i2	I benefited (less/more) from the shared workspace system than I expected.
i3	I am (less/more) likely to attain my goals because of this shared workspace system support.
i4	I feel satisfied with the work practices proposed by the shared workspace system.
i5	When the problem analysis session was over, I felt satisfied with the results.
i6	Our accomplishments today give me a feeling of satisfaction.
i7	I feel satisfied with the shared workspace systems support in today's problem analysis session.
i8	I feel satisfied about the way the shared workspace system made it possible for my colleague and I to work together on the assignment.

Table 6.4: Items of an workspace-based meeting assessment questionnaire based on [Briggs et al., 2003a] that evaluate the usefulness of Elgar with regard to the proposed work practices of the meeting.

Table 6.4 describes the adapted version of the meeting assessment questionnaire. This version has eight items that measure the usefulness of Elgar to support participants with regard to the diagnosis session. The other eight questions were not considered in this evaluation because they are not relevant for this experiment. These questionnaire items focus on the frequency of use of the system, Elgar in this case, and on demographic data. However, each team executes only once the diagnosis task and the demographic data focus on aspects that are not assessed in the experiment. Participants answer each item of the questionnaire in a Likert scale that ranges from one (strongly disagree) to seven (strongly agree).

Workspace Awareness Assessment Method

The second method evaluates the usefulness of Elgar in providing workspace awareness information to participants in diagnosis sessions to support their their coordination. The questionnaire uses items that assess the provided workspace awareness related to present actions of team participants [Gutwin and Greenberg, 2002].

#	**Question**
i1	I was aware of the contributions and content generated by my colleague.
i2	I was aware in which tasks my colleague was working and which data he/she was using.
i3	I was aware the ECCs in which my colleague was working.
i4	I was aware which information my colleague was using for that task.

Table 6.5: Items that represent shared situation awareness based on [Gutwin and Greenberg, 2002] to assess the usefulness of provided workspace awareness information provided by Elgar.

Table 6.5 describes four items that evaluate workspace awareness provided by Elgar. Participants answer each item of the questionnaire in a Likert scale that ranges from one (strongly disagree) to five (strongly agree).

Computer System Usability Method

This study uses an adaptation of the Computer System Usability Questionnaire (CSUQ) to evaluate the usability of Elgar in a diagnosis task based on Volvo Construction Equipment information. The CSUQ is a questionnaire that evaluates the overall usability of a software system [Lewis, 1995]. However, some questionnaire items from CSUQ were removed because they assess aspects of the use of Elgar that are not the focus of this experiment, e.g. easy recovery from errors. In addition, some questionnaire items were adapted to state the use of Elgar explicitly in the formulation of items. For example, the questionnaire uses the formulation“*Overall, I am satisfied with this shared system*” instead of “*Overall, I am satisfied with this system*”.

#	**Question**
i1	Overall, I am satisfied with how easy it is to use this shared workspace system.
i2	It was simple to use this shared workspace system.
i3	I can effectively complete my work using this shared workspace system.
i4	I am able to complete my work quickly using this shared workspace system.
i5	I feel comfortable using this shared workspace system.
i6	It was easy to learn to use this shared workspace system.
i7	I believe I became productive using this shared workspace system.
i8	I believe I became productive quickly using this shared workspace system.
i9	The information provided by the system is easy to understand.
i10	The information provided for the shared workspace system is effective in helping me complete the tasks and scenarios.
i11	The organization of information on the shared workspace system screen is clear.
i12	The user interface of this shared workspace system is pleasant.
i13	I like using the user interface of this shared workspace system.
i14	This shared workspace system has all the functions and capabilities I expect it to have.
i15	Overall, I am satisfied with this shared workspace system.
i16	I would use this shared workspace system in Volvo Construction Equipment for collaborative problem analysis situations in the future.

Table 6.6: Items of a usability questionnaire based on the CSUQ [Lewis, 1995] to evaluate the usability of Elgar.

Table 6.6 describes the adapted version of CSUQ. This version has sixteen items that measure the usability of Elgar and the specific functionalities of its coordination mechanisms. Participants answer each item of the questionnaire in a Likert scale that ranges from one (strongly disagree) to seven (strongly agree).

Quality of Collaboration Method

This study uses a method that evaluates the quality of collaboration in teams connected through technology during design situations [Burkhardt et al., 2009]. This method is used in this experiment to assess the quality of collaboration in teams working on diagnosis tasks through Elgar. The method describes quality of collaboration based on seven dimensions:

- Fluidity of Collaboration assesses the management of verbal communication (verbal turns), of actions (tool use) and of attention orientation.
- Sustaining Mutual Understanding assesses grounding processes concerning the design artefact (problem, solutions) and actions of participants.
- Information Exchange assesses design ideas pooling, refinement of design ideas and coherency of ideas.
- Argumentation and Reaching Consensus assesses whether there is argumentation and decision taken on common consensus.
- Task and Time Management assesses planning (e.g. task allocation) and time management.
- Cooperative Orientation assesses the balance of contribution of the actors in design, planning, and in verbal and graphical actions.
- Individual Task Orientation assesses, for each contributor, their motivation (marks of interest in the collaboration), implication (actions) and involvement (attention orientation).

The use of the method requires a referee to watch collaboration recorded in videos and rate them based on the seven aforementioned dimensions. The method defines, for each dimension, a set of indicators that are more specific to clarify and capture its aspects. Each indicator is associated to two questions, one with positive valence and one with negative valence. A referee may answer yes, no or yes/no to any of the questions. Based on these answers, a score is calculated for each indicator and, ultimately, for each dimension. In a positive valence question, the value of an answer 'yes' is 1 and the value of an answer 'no' is 0. Conversely, in a negative valence question, the value of an answer 'yes' is 0 and the value of an answer 'no' is 1. The value of an answer 'yes/no' is 0.5 in any type of question.

Table 6.7 describes the seven dimensions, their respective indicators and the questions used by referees to calculate the scores for each dimension. In this experiment, two referees used the dimensions, indicators and questions described as a coding scheme to estimate the quality of collaboration in all six teams, based on the transcriptions.

Although the original method considers encouragement and motivation to be one indicator in the dimension of individual task orientation, for this experiment the method was adapted and the indicator is split in two, an indicator for encouragement and another one for motivation. In a pre-analysis of the transcriptions, referees faced difficulties to classify excerpts that match simultaneously encouragement and motivation. Therefore, the method was adapted and the indicator divided in two to enable a more accurate of the coding schemes in the transcriptions.

6.2.5 Results

Duration of the Sessions

The duration of the diagnosis session for each team is described in Table 6.8. The average of teams that use Elgar with the ad-hoc coordination mechanism is 57.3 minutes, whereas the average of teams that use Elgar with the prescribed mechanism is 46 minutes.

	P&F	**J&D**	**J&M**	**R&D**	**R&L**	**M&O**
	A			**P**		
Duration (minutes)	43	62	67	29	47	62

Table 6.8: Duration of the diagnosis session for each team.

Therefore, in average, Table 6.8 shows that teams that use the ad-hoc coordination mechanism need less time to accomplish the goals of the diagnosis task for this experiment. Note that in all cases this is the first time a team works with a specific coordination mechanism.

Results of the Workspace-based Meeting Assessment Method

The results of the workspace-based meeting assessment questionnaire indicates the usefulness of each coordination mechanism (ad hoc and prescribed) during the diagnosis task.

After the diagnosis task, all twelve participants answered the meeting assessment questionnaire. Figure 6.11 shows the average of the answers of participants according to the used coordination mechanism.

These results show that the prescribed coordination mechanism is more effective in diagnosis sessions. Almost all answers that report the use of the prescribed coordination mechanism have higher averages (in one point

Indicator	Questions
Fluidity of Collaboration	
Fluidity of Verbal Turns	+ Is the communication among participants smooth? - Are there interruptions in the communication among participants?
Fluidity of the Use of Elgar	+ Do participants accomplish their goals with the available functionalities of Elgar in the first attempt? - Do the functionalities of Elgar interrupt the thinking process of participants?
Coherency of Attention Orientation	+ Are participants focused on the discussion and accomplishment of the diagnosis task? - Do participants get distracted, in some moments, while working on the diagnosis task?
Sustaining Mutual Understanding	
Mutual Understanding of the State of Problem and Solutions	+ Do participants ask questions, give clarifications or complementary information, using verbal channels or Elgar, on the state of the machine? - Are there misunderstandings on the state of the machine during relatively long periods of time?
Mutual Understanding of Current and Next Activities	+ Do participants acknowledge current activities, subsequent activities and their progress to accomplish the diagnosis task? - Do participants ignore current activities, subsequent activities and their progress to accomplish the diagnosis task?
Mutual Understanding of the State of Elgar	+ Do participants know the ECC/process activity that hold a discussion or has the focus of the team? - Do participants analyze or contribute to different ECCs or process activities that are not used by the team or are not in focus?
Information Exchange	
Generation of Ideas	+ Do participants generate ideas that contribute to the diagnosis task? - Do participants omit themselves in the generation of new ideas?
Refinement of Ideas	+ Do participants refine generated ideas that reflect their understanding of a situation? - Do participants leave ideas unchanged as they understand more a situation?
Argumentation and Reaching Consensus	
Criticisms and Argumentation	+ Do participants argue about points of view? - Do participants show disagreement of other points of view?
Common Decision Taking	+ Do participants discuss to take a decision together? - Do participants take a decision alone, ignoring the opinion of others?
Task and Time Management	
Work Planning	+ Do participants plan the order of next activities? - Do participants avoid the planning of next activities?
Task Division	+ Do participants propose the division of activities? - Do participants ignore the discussion about division of activities?
Distribution and Management of Tasks Interdependencies	+ Do participants identify and manage dependencies in the diagnosis task? - Do participants avoid to identify and manage dependencies in the diagnosis task?
Time Management	+ Do participants demonstrate concern about the time spent to execute the diagnosis task? - Do participants waste their time in irrelevant activities for the diagnosis task?
Cooperative Orientation	
Symmetry of Verbal Contributions	+ Do all participants provide verbal contributions equally in the diagnosis task? - Only one participant provides verbal contributions in the diagnosis task?
Symmetry of Use of Elgar	+ Do all participants use equally Elgar functionalities? - Only one participant uses the functionalities of Elgar whereas the other member observes?
Symmetry in Task Management	+ Do all participants execute equally distributed activities? - Only one participant executes for the team distributed or agreed activities?
Individual Task Orientation	
Showing Motivation	+ Does a participant shows motivation to accomplish the diagnosis task? - Does a participant shows indifference to accomplish the diagnosis task?
Encouraging Motivation	+ Does a participant motivates another member to accomplish the diagnosis task? - Does a participant ignores the lack of motivation of another participants with regard to the diagnosis task?
Constancy of Effort Put in the Task	+ Do participants apply their efforts constantly to accomplish a diagnosis task? - Do participants do the minimum to accomplish a diagnosis task?
Attention Orientation in Relation with the Task	+ Do participants pay attention while working on a diagnosis task? - Are participants distracted during the diagnosis tasks?

Table 6.7: Indicators of dimensions to evaluate the quality of collaboration based on [Burkhardt et al., 2009].

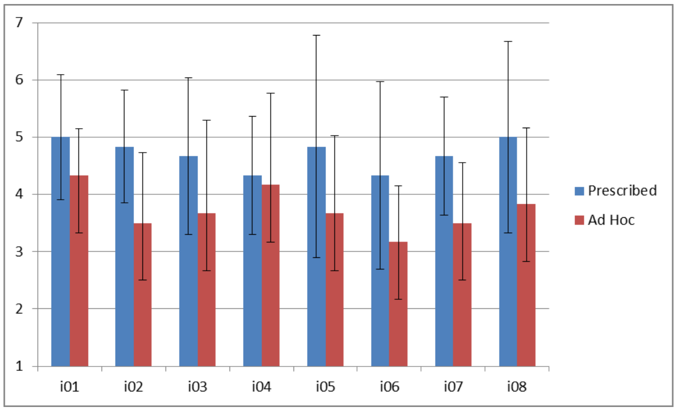


Figure 6.11: Average of answers from the meeting assessment questionnaire from teams that used the prescribed and ad-hoc coordination mechanisms.

in the scale) than the answers that report the use of the ad-hoc coordination mechanism. The exceptions are items “i01” and “i04”. The difference between averages for prescribed and ad-hoc coordination mechanisms is less than one point in the scale, therefore they are considered to be comparable, indicating that participants do not seem to have a preference for the prescribed coordination mechanism over the ad hoc, or vice versa. The high index of both coordination mechanisms in the item “i01” demonstrates that participants had their expectations exceeded with regard to Elgar and the used coordination mechanisms as item “i01” states that they had more support from Elgar than anticipated. The same situation occurs with the item “i04”, the reported average in Figure 6.11 indicates the usefulness of work practices proposed by the coordination mechanisms, which were unknown to participants before this experiments.

Although the chart in Figure 6.11 indicates a general tendency towards a high the usefulness of the prescribed coordination mechanism over the ad-hoc, the tendency lacks validity as the dispersion of the answers in the results, measured by the calculation of standard deviation for each item, is high, indicating that participants have divergent opinions.

Results of the Workspace Awareness Assessment Method

The results of the workspace awareness questionnaire show the usefulness with the provided information as illustrated in the Figure 6.12.

In particular, the item “i01” indicates the usefulness of awareness information, provided by Elgar, to describe content generated by other participants

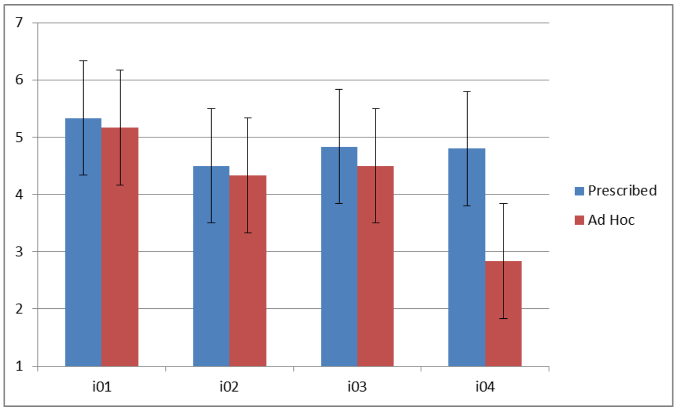


Figure 6.12: Average of answers from the workspace awareness questionnaire from teams that used the prescribed and ad-hoc coordination mechanisms.

during the diagnosis session. The averages show the usefulness of awareness information, independent of the used coordination mechanism. Although the values of error bars have a variation of two points in the scale of possible answers, the minimum values of the error bars are above the medium value.

The items "i02" and "i03" also have similar averages for prescribed and ad-hoc coordination mechanisms. According to both items, participants are aware of the activities on which their peers work and the ECCs used. However, such a result cannot be confirmed because the standard deviations of both items are high.

Another particular case is the item "i04". This item indicates the awareness of a participant about the information used by another participant. The difference between participants who use the prescribed or ad-hoc coordination mechanisms is significant, indicating that participants who use the prescribed coordination mechanism are more aware about the information used by their peers during the diagnosis task, in comparison to participants who uses the ad-hoc coordination mechanism.

Results of the Computer System Usability Method

Analysis of the results of the CSUQ shows a tendency in preference of the prescribed coordination mechanism over the ad hoc. Figure 6.13 shows the averages of the answers of participants in the diagnosis task. For almost all items, participants evaluate the usability of Elgar and the prescribed coordination mechanism higher than ad hoc, except for the item "i14". This item, the average of the ad-hoc coordination mechanism is slightly higher

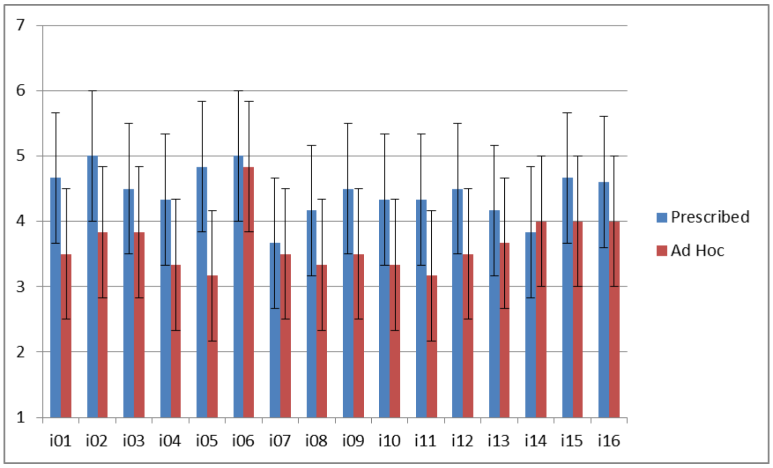


Figure 6.13: Average of answers from the computer system usability questionnaire from teams that used the prescribed and ad-hoc coordination mechanisms.

than the prescribed, indicating that participants believe that Elgar provides all expected functionalities and capabilities for the diagnosis task.

Item “i01” shows that Elgar is easier to use with the prescribed coordination mechanism than with the ad-hoc mechanism. Participants that use the prescribed mechanism tend to agree with this item, whereas participants that use the ad-hoc mechanism tend to disagree with this item. Item “i02” shows that the use of the prescribed coordination mechanism is simpler to use than the ad-hoc mechanism. Item “i05” shows that participants feel more comfortable to use Elgar with the prescribed coordination mechanism than with the ad-hoc. Finally, item “i11” shows that Elgar with the prescribed mechanism provides a clearer organization of information.

Results of the Quality of Collaboration Method

Each coordination mechanism is evaluated to understand its influence in collaboration. Table 6.9 summarizes the scores for each of the teams in this experiment. The score describes an index for the seven dimensions that represent the quality of collaboration, for each of the six teams. The index is represented in a scale from one to five, in which one represents a low score for the dimension whereas five represents a high score. A low score represents a lower quality of collaboration with regard to a dimension and a higher score represents a higher quality of collaboration.

Dimension	**P&F**	**J&D**	**J&M**	**R&D**	**R&L**	**M&O**
	A			**P**		
Fluidity of Collaboration	5	4.6	4.6	2.9	2.9	4.2
Sustaining Mutual Understanding	5	4.2	5	4.6	3.8	5
Information Exchange	4.6	4.6	4.6	5	4.2	5
Argumentation and Reaching Consensus	5	3.75	5	5	5	5
Task Management	4.375	5	4.4	0.6	3.1	3.8
Cooperative Orientation	5	5	3.4	5	2.2	4.4
Individual Task Orientation	4.4	4.4	4.1	4.4	2.5	4.4

Table 6.9: Score of the dimensions that measure the quality of collaboration for each coordination mechanism, normalized in a scale from one to five.

After the transcription of the dialogues of the teams, referees use dimension indicators to tag the transcription individually. Once tagged, referees answer with “yes”, “no” or “yes/no” the positive and negative valence questions, associated to each indicator. Subsequently they compare their individual results and merge it in a common version.

Based on the common version of the tags of the transcription, the score of the dimensions is calculated. The calculation is based on the principle of the cross-multiplication that uses the ratio of the score of a dimension and the representation of its value in a Likert scale of five points, such as indicated by the equation described in the Figure 6.14. Using the cross-multiplication principle in the equation, the value of x is calculated as the result of the multiplication of the score of a dimension (variable *dimensionScore*) by five (the maximum of a Likert scale used in this experiment), divided by the maximum score of the considered dimension (variable *maximumDimensionScore*). For example, a dimension has a score of 4.2 (approximately) with a *dimensionScore* of 5 and a *maximumDimensionScore* is 6, according to the equation.

The score of a dimension (*dimensionScore*) depends on the answers to the questions of dimension indicators. For example the score of the dimension “Fluidity of Collaboration” is six when a referee assign “yes” to all positive valence questions and “no” to all negative valence questions. Whereas the maximum score of a dimension (*maximumDimensionScore*) is calculated based on the number of questions of a dimension. For example the dimension “Fluidity of Collaboration” has six questions, therefore its maximum score is six.

The following two sections describe the results of each dimension and

$$\frac{dimensionScore}{x} = \frac{maximumDimensionScore}{5}$$

Figure 6.14: Equation to calculate the score of a dimension of the quality of collaboration method in a Likert scale of five points.

provides the reasoning for such indexes together with excerpts from the transcriptions of the dialogues of the teams. The first section describes the dimensions from the perspective of the three teams that use the ad-hoc coordination mechanism, the second section describes the dimensions from the perspective of the three teams that use the prescribed coordination mechanism.

Ad-Hoc Coordination Mechanism

Fluidity of Collaboration

Teams that use the ad-hoc coordination mechanism demonstrate more fluidity in collaboration as indicated by the indicators “fluidity of verbal turns”, “fluidity of the use of Elgar” and “coherency of attention orientation”. In observed teams with regard to “fluidity of verbal turns”, participants exchange several questions and answers with each other during their discussion to clarify a certain topic. For example, in the following excerpt of transcription, participants have several verbal turns to create a shared understanding about the interpretation of the telemetry data of a machine.

J: *F1 and F2, is that forward 1 and forward 2?*
D: *I presume that it is for the clutch 1 and f2 is the pressure for the clutch 2.*
J: *Yeah.*
D: *And F should be pressure for the clutch F.*
J: *Yeah, just forward.*
D: *Exactly.*
J: *Yeah, that is right.*
D: *We see the time here for this shifting*
J: *Ahhh, 300 ms?*
D: *300 ms*
J: *Something like 350 maybe?*
D: *and we have problem with the clutch*
J: *Yeah but we have, we have a long pressure on both yeah I mean clutch okay clutch for forward 2 is disengaging in this case, I guess.*
D: *Yes, that is right.*

With regard to “fluidity of the use of Elgar” during the diagnosis task, the observed participants use the functionalities provided by Elgar during the diagnosis task often in the first attempts and without constant help from the session mediator. For example, a team manages to analyse the hydraulic oil pressure in the telemetry data, report it in Elgar through CollPads, and generate a summary of the diagnosis session without need of help. The following excerpt of transcription illustrates some of these aspects.

D: *The main pressure yeah, that is the main pressure. Yeah, that is right.*
J: *Maybe there is a small drop in main pressure as well.*
D: *Yeah, that is right.*
J: *I dont know if we can.*
D: *If it is in the same There is ahm ok, there is a block but see each channel one for channel F and another for channel 1.*
J: *I am gonna, I am gonna write something about the analysis and then we can fill in What do you think?*
D: *Good.*
J: *Just to get somewhere.*
D: *Yeah.*
D: *Ah, maybe we should put it in this CollPad. Dropping the picture*
D: *Maybe we talked about different possibilities. May be each of them we can write in the CollPad.*
J: *Yeah.*
D: *and then we rate them later.*
J: *Yeah, good idea.*
D: *Ah maybe, one possibility there can be a problem that the piston of the clutch becomes ahm the prim of the clutch has got sufficient force to return back the piston*
J: *Ahhhmm okay*

The last indicator for this dimension is "coherency of attention orientation" contributes to the high index of this dimension. Most of the observed participants who used this coordination mechanism do not demonstrate distraction or do not request repetition of previous statements often. For example, none of the teams that used the ad-hoc coordination mechanism demonstrated lack of attention because there was no trace on the transcriptions indicating a request for repetition during the diagnosis task.

Sustaining Mutual Understanding

Teams that use the ad-hoc coordination mechanism have a high index with regard to the dimension "sustaining mutual understanding", on the basis of "mutual understanding of the state of problem and solutions", "mutual understanding of the current and next activities" and "mutual understanding of the state of Elgar". The indicator "mutual understanding of the state of problems" indicates that teams constantly interact to ask questions, clarify issues or provide complementary information. For example, the excerpt of transcribed dialogue below shows the amount of information shared among participants during the diagnosis task.

F: *Then possible inappropriate operators action I think this is nothing to do with operators I guess.*
P: *No.*
F: *Seems to be a failure in the transmission.*
P: *Ahhhhm but normal drivers behaviour is not this service maintenance ja*
F: *Service and maintenance it is possible*
P: *I dont think the operators not abnormal driving behaviours.*
F: *No I don't think so I think it should be some problem with the transmission and not with operator.*
P: *Yeah.*

The indicator "mutual understanding of the activities in progress and next activities" indicates through the observation of the screencasts that teams are aware of all activities that are necessary for the diagnosis task. For example, the following excerpt of transcribed dialogue demonstrates that participants, using the ad-hoc coordination mechanism, are aware of the sequence of activities that provide data to fill the information required in the Volvo Case Report.

J: *Should we like ahhmm, suggest possibly inappropriate operators actions as a task? Ahm normal machine behaviour . What do you think?*
D: *Ah !! Uhm, ahm a fault description.*
J: *Ah ok, sorry. Should we, like, suggest . something to replace then?*
D: *The fault description should be written in a Collpad. (not on a task list).*
J: *Are you talking to me now?*
D: *Yeah.. (laughs). Because we should make this fault description normal, machine behaviour, possible inappropriate operator actions and all those to write in the Collpad, I suppose.*
J: *Yeah, right, I think so.*
D: *What do you think?*
J: *Sounds like a good idea.*

The indicator "mutual understanding of the state of Elgar" indicates that participants using the ad-hoc coordination mechanism not only mention the names of ECCs during their discussion but also are aware of their instantiations in the shared workspace system. For example, the following excerpts of transcribed dialogues in teams illustrate this indicator.

J: *Should I open it (Static Line Chart) David, should I open it? You will see if I open it, I guess. <David opens a Static Line Chart without mentioning.>*
J: *Oh, now we have 2.*
J: *I am closing mine then.*

F: *Ok, you see my descriptions and ahm in the CollPad*
P: *Yes.*
F: *Ok, that is good.*

P: *But ahm ja ok that is to 2 bar approximately.*
F: *Or maybe even further*
P: *It is hard to see, you need ahm zoom in. <Pierre provides a collpad contribution>*
P: *So, I have wrote something as well*
F: *Yeah, I can see it.*
P: *But I havent attached the diagram because ahm it the diagram doesnt describe as it should be. Maybe it should not be attached.*

An exception is the team (J&D). In certain situations participants demonstrate confusion with the use of ECCs, as they do not to know in which ECCs to describe possible hypotheses and subtasks.

Information Exchange
Teams that use the ad-hoc coordination mechanism demonstrate a high index in the dimension "information exchange", during the diagnosis task, indicated by "generation of ideas" and "refinement of ideas". All observed teams generated ideas according to their evaluation of provided data that contributed to the formulation of a diagnosis report, in line with the nature of the task. All participants contributed with hypotheses for root-cause failures of machines. All observed teams also constantly refined their hypotheses as a result of discussions with other participants or influence of their suggestions. For example, the observation of different screencasts demonstrate that participants changed their hypotheses after discussions with other participants or after review of the generated content.

Argumentation and Reaching Consensus
Teams that use the ad-hoc coordination mechanism demonstrate a high index of argumentation and reaching consensus, as indicated by "criticism and argumentation" and "common decision taking". With regard to the indicator "criticism and argumentation" participants often discuss about a machine anomaly described through the provided telemetry data. For example, the following excerpt of transcription shows the interaction among two participants in a discussion to define failure detection procedures.

F: *Yeah, but I mean, for the last one failure detection procedures ahmm on the machine I probably can write ahm that when we do the gear shifting other clutches cannot should not drop if it drops then there is some failure in the clutches.*
P: *Yeah unfortunately we don't have this pressure, I think.*
F: *Ok, so it is not in the ECU so*
P: *No. I think you can detect by the speed sensors that you have slip in the transmission.*
F: *Uhum! Speed sensors for the clutch. Ah no for the (not understandable)*

With regard to the indicator “common decision taking” teams often discuss and seek for consensus before they make a decision. This indicates that participants take common decisions together in the diagnosis task, especially with regard to the description of machine anomalies and recommendations to avoid further degradation of the machine. Participants seem to ignore the opinion of another member or take decisions alone. For example, the following transcribed excerpt shows the discussion of a team about a strategy to describe possible mechatronic problems that cause a machine failure.

P: *I can write I can try to write about the about the possible mechatronic problem causing machine failures.*

F: *Shall we write it now or shall we do it later? Because I think maybe you can write it now otherwise we can we will forget what we discussed.*

P: *Yeah.*

Task Management

Teams that use the ad-hoc coordination mechanism demonstrate a high index in the dimension “task manager” as indicated by “work planning”, “task division”, “distribution and management of tasks interdependencies” and “time management”. Work planning indicates that teams plan their work. All observed teams discuss and plan a strategy to identify anomalies, described by the machine telemetry data, and fill the Volvo Case Report, required to accomplish the diagnosis task. For example, the following excerpt of transcription demonstrates a situation in which a team plans their work.

F: *Yeah but I mean, Jordan proposed that we should follow this Volvo Case Report, for the specific items*

P: *Ah, ok.*

F: *So we write the contributions*

P: *Ok.*

F: *So we write in the CollPad. So if I wrote the fault description and then there are 2 questions component that is detected as faulty and explanation of the smallest replacement part to solve the issue.*

With regard to task division teams divide their tasks to contribute to the formulation of Volvo Case Report. They either take the initiative to execute a task or propose that another participant executes a task. For example, the following excerpt of transcription illustrates such a situation.

J: *So, it is ahm, analyse the measurement. Right?*

D: *Yeah.*

D: *So, we kind say that this is completed now, or?*

J: *I dont know. What do you think?*

D: *If it is maybe not, we can come up with more information. We add maybe shall we add a task to describe .. or, or how to say? ahm, or Conclusions.*
J: *Yeah. I will do something about the analysis. If you write something about the ...*
D: *the task*
J: *conclusion*

With regard to distribution and management of dependencies, teams distribute and manage, among participants, subtasks that are necessary to accomplish the goals of the diagnosis task. The observed teams identify, propose and manage the subtasks that influence in the formulation of the Volvo Case Report. For example, the following excerpt illustrates a participant who checks whether another member finished a subtask to progress in the diagnosis task.

D: *Maybe we can write down that probably there is no it is not inappropriate*
J: *Yeah, it is not operator*
D: *Dependent*
J: *Yeah*
D: *It is the hardware*
J: *Have you finished the conclusion?*
D: *The conclu ahmmm*
D: *No, I didnt. I wrote this as only as a task that the task that you did.*
J: *Ah ok.*
D: *to write the conclusion for analyze ... you wrote this.*
J: *Oh, sorry.*
D: *That is why*

Finally, with regard to time management, participants do not demonstrate concern about the execution of the diagnosis task, respecting the time limit. However, none of the observed teams switch their attention in other activities that are irrelevant for the diagnosis task. One team (J&D) represents the exception with regard to time management. In this team, participants have a discussion to agree on the continuation of the diagnosis task, although their time limit is over. The following excerpt of their transcribed interaction demonstrates the concern of one of the participants with the time limit.

D: *It is good to (participant wants to continue with the discussions about the session) It is time or? Mediator: The session should take one hour.*
D: *So we have more 45 min .*
J: *No, we are finished then! ... Arent we?*
J: *The conference call is 15, so it means it is now (the time in which the conference call should be over)*

Cooperative Orientation

Teams that use the ad-hoc coordination mechanism demonstrate a high index in the dimension “cooperative orientation” due to the positive influence of the indicators of this dimension. The “symmetry of verbal contributions” indicates that participants contribute verbal ideas equally to interpret the telemetry data of the machine.

The indicator “symmetry of use of Elgar” indicates that participants use and explore the functionalities of Elgar, not only providing contributions or ratings but also with regard to instantiation of ECCs and the use of workspace awareness information during the diagnosis task.

With regard to “symmetry of task management” participants execute the divided tasks equally, contributing to the report generated in the end of the diagnosis task.

Individual Task Orientation

Teams that use the ad-hoc coordination mechanism demonstrate a high index in the dimension “individual task orientation”. The indicator “show motivation” indicates that participants demonstrate motivation to execute the diagnosis task. For example, the following excerpt of the transcribed dialogue of a team illustrates the motivation aspect.

J: *read another contribution. This is how to use the machine to avoid the downstream damages. Avoid using the defected gear until ahm cause of too low pressure is fixed, if not then the clutch was (not understandable) clutch c certainly happen. I think this is the higher priority, should be a 4 or 5.*
M: *I think 5.*
J: *5.*
J: *read another contribution.*
M: *I just added that (laughs)*
J: *Should we perhaps add the other one?*
M: *We can skip that one.*
J: *read another contribution.*
J: *As a proposal on how to improve the analysis. 3, 4? What do you think?*
M: *Yes sure, put that as 4, sure.*
J: *Sure.*
J: *More work to the software guys.*
M: *Yeah.*
J: *Yeah.*
J: *Great, we are done.*

The indicator “encourage others motivation” indicates that participants whom use either the ad-hoc or prescribed coordination mechanisms do not demonstrate encouragement of other participants to participate on the diagnosis task nor do they demonstrate lack of motivation.

The indicator "constancy of efforts" indicates that participants put their effort in the diagnosis task. Participants explore the provided machine telemetry data, discuss their impressions and share information based on their knowledge and experience. For example, the following excerpt of the transcribed dialogue of a team shows the initiative of a participant to describe one possible fix for a machine problem.

F: *Then I can write the fault description and you can write normal machine behaviour.*
P: *Yeah.*
P: *So then we go to Collpad and write something.*
F: *Yeah, i think so.*
P: *Or should we make the task lists first? It doesnt matter or?*
F: *Hum I think now we have created the task list so we write in the collpad and then when we are finished we click complete.*
P: *Yeah. But then on the other hand we can plan for more actions first and then do it .. and then I guess but we can do it like this now.*
F: *Aham ! Ok, ja.*

The indicator "attention orientation" indicates that participants demonstrate their focus on the diagnosis task. During the observation of the video recordings, participant demonstrate their focus to work on the diagnosis subtask that contributes to the overall task until its completion and do not leave subtasks incomplete. However, the team J&D demonstrates distraction during the task, as a participant requests re-explanation of a situation from the other member. For example, the following part of transcribed dialogue shows the distraction of a participant during a discussion.

Prescribed Coordination Mechanism

Fluidity of Collaboration

Teams that use the prescribed coordination mechanism demonstrate less fluidity in collaboration compared to the ad-hoc coordination mechanism. With regard to "fluidity of verbal turns" observed teams do not interact often to discuss matters of diagnosis or to reach consensus, therefore the fluidity of verbal turns is low. For example, in the transcription of the dialogue in the team L&R, this situation becomes more evident as participant "L" provides verbal contributions with regard to anomalies in the telemetry data, whereas participant "R" just acknowledges the reasoning without providing any contribution.

L: *You have deep in pressure in the forward clutch. Ja, and that is due to the first deep is where we start to feel the which clutch I can't see the information No, the first clutch is filled up and then you will have a pressure drop.*

R: *Ok.*

L: *Because the we need to have oil to the clutch engagement and also in the second drop will be when we increase the pressure to the forward clutch, so that is, or to the first clutch. So that is not unnormal. And the pressure levels themselves about I miss when I have worked with Elisabeth, I miss the speed information I think we have some speed rotational speed as well.*

R: *Yes.*

L: *I can't see from this picture that it should be an obvious problem.*

R: *Then, let's write that as a summary.*

The "fluidity of the use of Elgar" indicates that although participants accomplish their tasks with the functionalities provided by Elgar, the observation of three teams show that the use of these functionalities, for example the transition of activities of the collaboration process, interrupt the reasoning process in a team during the diagnosis task, which in turn contributes to decrease the general index of the fluidity of collaboration. For example, the video recordings show that after moving forward to the next activity, participants focus their efforts on the new configuration provided by a process activity, such as the understanding of new provided instructions, the new provided ECCs and the required data for the activity.

With regard to "coherency of attention orientation" the team R&L demonstrates low attention. During the whole task, participant "R" demonstrates passive behaviour in the discussion of the anomalies of the telemetry data, not showing interest in their analysis. Whereas "L" demonstrates more enthusiasm and participation during the whole task, analysing data and contributing with knowledge and experience. This situation is demonstrated by the following excerpt of the transcription of their dialogue during the task.

L: *So ... This was a dump from a suspicious shift, indicating some problem but I cant from this information say that or see that. It's a bit tricky. I think we have It would have been good to have these transmissions and signals as well.*

R: *Mh. Ok. Do we move on to the next part in the process, or?*

Sustaining Mutual Understanding

Teams that use the prescribed coordination mechanism demonstrate a high understanding about machine anomaly during the diagnosis task. The indicator "mutual understanding of the state of problems" contributes to lower the index of this dimension because the observed teams do not demonstrate mutual understanding about the state of the problem. According to the screencasts, the teams that use the prescribed coordination mechanism did not interact and discuss to clarify or gather more information about the machine anomaly. They started to work directly with the provided information and did not explore it further. Nonetheless, the video recordings or transcriptions from these teams do not demonstrate an inappropriate understanding about machine anomalies or misunderstandings among participants to agree on problems.

The "mutual understanding of the activities in progress and next activities" indicates the mutual understanding of participants with regard to activities in progress and subsequent activities. All observed teams demonstrate an understanding of the sequence of activities to analyse and generate the required data to fill the Volvo Case Report and complete the diagnosis task. For example, the following excerpt of transcription demonstrates how a team explores the process activities and their sequence to complete the diagnosis task and the need to progress throughout them.

L: *Is there ah ok I can pull this bar up here (the bar of process phases) and go to the right. Or it is more phases coming here. I didnt see them. Should we go on to the next one then?*
R: *Yeah.*
L: *Describe possible downgrade operation modes.*
L: *To avoid a break down ok to avoid a breakdown.*

The "mutual understanding of the state of Elgar" indicates that observed teams do not face problems in including the several functionalities of Elgar in their work practices to analyse and describe root-causes failures with machines, described in telemetry data. Teams that use the prescribed coordination mechanism adapted themselves to follow the prescribed process activities to accomplish the diagnosis task, use information provided by Elgar in each of the activities and use provided ECCs to generate data to the Volvo Case Report.

Information Exchange

Teams that use the prescribed coordination mechanism demonstrate a high exchange of information during the diagnosis task. The observed teams generate and refine several hypotheses for the machine anomalies described in the telemetry data, as guided by the prescribed collaboration process. However, in team R&L, the index is lower for other teams because a participant did not contribute in the generation and refinement of ideas, avoiding exchange of knowledge during the task.

Argumentation and Reaching Consensus

Teams that use the prescribed coordination mechanism demonstrate a high level of argumentation and reach consensus as indicated by “criticisms and argumentation” and “common decision taking”. The “criticism and argumentation” indicates that participants who use the prescribed coordination mechanism demonstrate natural discussions and argumentation to develop their understanding about machine anomalies. In addition, none of the observed teams demonstrate explicit manifestation of disagreement or criticism. For example, the following excerpt of transcription demonstrate the argumentation of a team.

L: *The control that controls the system that the speed sensors measure the systems as it triggers sessions in the software. So this is the 2 mechatronic parts that could be blamed here. But how do we go on from here? Is it another*
R: *Well, when we what you said ... we shouldnt if this was a real case, I wouldnt go on doing anything till we have more information.*
L: *Yeah, and we*
R: *Information about other error codes and speed sensor.*
L: *Yeah, that would be required.*

With regard to “common decision taking” participants that use the prescribed coordination mechanism often make decisions together. The observed teams discuss their actions before taking them or seeking the opinion of other participants. In all teams, participants take decisions together and do not impose a decision on other members without a previous discussion. For example, in a team that uses the prescribed coordination mechanism (R&D), participants move together to other process activities whenever all members finish their work. The following excerpt of transcription shows a recurrent pattern of the team in the diagnosis task, the confirmation to move to another process activity when participants realize that they have achieved the results for a process activity .

R: *Should we go further?*
D: *Yeah.*

Task Management

Teams that use the prescribed coordination mechanism demonstrate low efforts of task management as indicated by “work planning”, “task division”, “distribution and management of tasks interdependencies”. With regard to “work planning”, teams did not plan their work, as the collaboration process already prescribes their work to accomplish the goals of the diagnosis task. With regard to “task division”, teams that use the prescribed coordination mechanism do not propose division of subtasks among other participants. They avoid discussions on explicit task division and contribute directly to the topics prescribed in each of the process activities.

With regard to “distribution and management of dependencies” teams that use the prescribed coordination mechanism identify, distribute and manage subtasks that contribute to the diagnosis task. For example, the following excerpt of transcription shows the interaction of participants. They agree that the generated results for a process activity are sufficient before they progress to the next activity.

L: *For this to change the on-board failure detection procedure that would be to include the the speed signals in that case. Hum? Ok? Next one. Or do you have more here?*
L: *Should should we go to the last one?*
R: *Yeah, I think so.*
L: *Generate Session Summary.*

With regard to “time management” none of the observed teams demonstrate concern about the management of time during the diagnosis task. They do not demonstrate concern about the time spent to execute the task, nor do they switch their attention to other irrelevant activities for the diagnosis scenario.

Cooperative Orientation

Most of the teams that use the prescribed coordination mechanism demonstrate an orientation towards cooperation as indicated by “symmetry of verbal contributions”, “symmetry of use of Elgar” and “symmetry in task management”. With regard to “symmetry of verbal contributions”, participants provide verbal contributions to the identification of machine anomalies, described in the telemetry data.

With regard to “symmetry of use of Elgar” participants use and explore the general functionalities implemented by Elgar and particular functionalities implemented by the prescribed coordination mechanism, e.g. prescription of ECCs for process activities and contextualized instructions.

With regard to “symmetry in task management” most of the observed participants execute assigned subtasks.

One of the observed teams represents an exception with regard to this dimension (R&L). Participants in this team do not demonstrate participation in the diagnosis task. One participant contributes with more information about the anomalies in the telemetry data whereas the other is more passive and does not participate in the discussion actively. The same situation happens in the use of the functionalities provided by Elgar (indicator “symmetry of the use of Elgar”), one participant uses and explores Elgar, whereas the other observes more and uses it minimally. Finally, the team also shows asymmetry with regard to task management (indicator “symmetry in task management”). One participant contributes more actively to the diagnosis tasks and executes necessary subtasks of it, whereas the other participant observes and suggests to move forward from one process activity to the other, when the partial activity results have been generated.

Individual Task Orientation

Most of the teams that use the prescribed coordination mechanism demonstrate high individual task orientation, as indicated by “showing motivation”, “encouraging motivation”, “coherency of effort put in the task” and “attention orientation in relation with the task”. The “show motivation and encourage others motivation” of the observed teams that use the prescribed coordination do not demonstrate motivation or motivate other participants.

With regard to “constancy of efforts”, most of the observed teams persisted in putting their efforts to achieve the desired results for the diagnosis task. For example, the following excerpt of transcribed dialogue shows the efforts of a participant to move forward in the collaboration process.

L: *For this to change the on-board failure detection procedure that would be to include the the speed signals in that case. Hum? Ok? Next one. Or do you have more here?*
L: *Should should we go to the last one?*
R: *Yeah, I think so.*

An exception for this indicator is the team R&L, in which one of the participants does not apply effort to contribute to the diagnosis task. The participant contributes few verbal ideas and a few hypotheses.

With regard to “attention orientation”, participants are proactive in contributing to the diagnosis task and are not distracted during the task. However, in one of the teams (R&L), the same participant demonstrates lack of pro-activity and contributes the minimum effort in the diagnosis task.

6.2.6 Discussion

Although the coordination mechanisms are different with respect to some implementation aspects, both mechanisms enable participants to achieve the desired results without a significant variation in the time spent with the task. In this experiment, all teams identified the root-causes that caused the symptoms of the wheel loader machine within an acceptable and similar time range.

Teams prefer to use the prescribed coordination mechanism rather than the ad hoc during the diagnosis session. Such a preference is indicated by the item “i02”, “i05”, “i07” and “i08”, represented in Figure 6.11. These items indicate that teams that use this coordination mechanism benefited more from Elgar than expected (“i02”), generated results in Elgar effectively (“i05”), had the support of Elgar during the diagnosis task (“i07”), and especially the support for collaboration (“i08”). Teams report that the difference in the prescribed coordination mechanism is in the use of a structured collaboration process. In informal conversations after this study, several participants mention that they had never used a software system based on a collaboration process to guide them in a diagnosis task. They often worked with Microsoft Excel® for the analysis of telemetry data and Microsoft Lync® in situations that require collaboration with another distant expert. The combination of software systems resembles the ad-hoc coordination mechanism, in which participants need to coordinate the work by themselves during a diagnosis task. Therefore, the high index of the prescribed mechanism over the ad hoc, for the aforementioned questionnaire items, demonstrates the usefulness of new work practices and structures introduced by the collaboration process.

Teams that use the prescribed coordination mechanism indicate a clearer organization of information on the shared workspace system screen (item “i11”, Table 6.6) in comparison to teams that use the ad-hoc coordination mechanism. In the prescribed mechanism, participants only visualize the ECCs that are assigned to a specific process activity, whereas the other ECCs are hidden by Elgar. ECCs that have local use in an activity are presented, which avoids cluttering the screen of Elgar with unnecessary ECCs. The situation is different in the use of the ad-hoc coordination mechanisms, because participants are allowed to instantiate as many ECCs as required to support them in the accomplishment of the diagnosis task. Teams that use this coordination mechanism use more ECCs (around five) than the teams that use the prescribed mechanisms (around three). This situation is well captured in the item “i11”, as the item shows a higher index for the prescribed coordination mechanism.

The observed teams that use the prescribed coordination mechanism take less time performing the diagnosis task most likely because the collaboration process prescribes the activities that are necessary for the diagnosis task and their order. The process represents a prescribed plan, formulated in advance by experts in diagnosis, prescribing the activities that are necessary to identify the root-causes of a machine, through the telemetry data, filling in the Volvo Case Report. The prescription of a collaboration process facilitates task management work of teams because it describes a structure that the team should follow for a certain type of situation. Therefore teams that use the prescribed coordination mechanism have shown to spend less time on the diagnosis task, because they do less task management work than teams that use the ad-hoc mechanism.

However, this result can be different if participants had to attend more diagnosis sessions, executing the same collaboration process using the prescribed coordination mechanism. Participants would become more familiar with the process, possibly making the execution of the process more time-consuming and redundant for them, as they would have had assimilated the activities of the process and the order of their execution. In this situation participants might have had preferred to use the ad-hoc coordination mechanism, as they would have more flexibility to define a coordination strategy according to their previous accumulated knowledge and experience about diagnosis.

Observed teams that use the ad-hoc coordination mechanism have shown to be slower because they have to do more coordination work to accomplish the expected results for the diagnosis task. These teams had to plan their work to identify and report the root-cause for a machine anomaly, they had to agree on plans, divide their work among participants and manage the dependencies of the divided work. These teams interact more with each other to agree on task management, therefore they demonstrated a higher rating in task management dimension. It reflects the additional work that is provided by collaboration processes in the prescribed coordination mechanism.

Teams that use the prescribed coordination mechanism and follow collaboration processes experience quick pauses that represent a discontinuation of the reasoning process. Whenever the teams obtain the expected partial results for a process activity, they switch to a next process activity, which reconfigures the workspace in Elgar, it opens new ECCs prescribed for a process activity and it provides new instructions for the particular activity. The observation of video recordings demonstrate that after an activity switch, participants need process again the new instructions to understand the required outcomes for the activity and associate these outcomes with the achievements of the diagnosis task. Although quickly, such a pause interrupts their reasoning process, creating by each process activity an interruption that hinders the fluidity of collaboration. Therefore, the index that measure the fluidity of collaboration is lower for teams that use the prescribed coordination mechanism.

On the contrary, the results indicate that the fluidity of collaboration in teams that use the ad-hoc coordination mechanism is high, most likely, because participants do not need to switch context of their workspace to understand new instructions or use new ECCs. These teams centre their efforts on the identification of the root-causes for a machine anomaly and on filling the Volvo Case Report, adapting the use of the shared workspace system accordingly. They decide the type of ECCs and the number of instances to organize the information they generate during the diagnosis task, which becomes part of the Volvo Case Report.

The analysis of the results obtained with the quality evaluation method also shows that the use of a particular mechanism does not influence argumentation and reach of consensus in teams, as most of teams have the maximum score for this dimension. Both coordination mechanisms do not influence the introduction of disagreements in the discussions of teams or avoid them to take decisions together. The analysis of the video recordings demonstrate that participants naturally discuss about their hypotheses with each other to become sure about them or the task management, and these discussions seldom turn into disagreements. Such a situation provides indications that the implementation of the provided coordination mechanisms does not hinder collaboration aspects that are important in the diagnosis tasks, especially when several experts are required.

The results of the workspace-based meeting and usability questionnaires demonstrate the usefulness of functionalities provided by the prescribed coordination mechanism. The prescription of a collaboration process facilitates their efforts in the diagnosis task, as it anticipates and describes the activities that teams follow to achieve their goals. Although the functionalities provided by the ad-hoc coordination mechanism have a lower index with regard to the assessment of the meeting and usability, the evaluation of the quality of collaboration shows higher scores for teams that use this mechanisms. In particular, the fluidity of collaboration is higher because participants interact more to discuss the situation of a machine but also to coordinate their work to fill the Volvo Case Report.

6.3 Discussion

The results of the experiments indicate that Elgar and its mechanisms can help coordinate collaborative machine diagnosis. Diagnosis teams in both experiments accomplished their tasks successfully and overall, the interpretation of the results indicate the usefulness of the functionalities of Elgar and the coordination mechanisms in diagnosis tasks. The system and the mechanisms are evaluated in different scenarios developed together with two machine manufacturers, Hägglunds Drives and Volvo Construction Equipment. The second experiment uses an evaluation approach to assess the usefulness of Elgar in diagnosis tasks that combines four research methods to evaluate: *i)* the usefulness of Elgar supporting participants in diagnosis sessions, *ii)* the usefulness of the workspace awareness information provided by Elgar, *iii)* the usability of Elgar and *iv)* the quality of collaboration for teams that use Elgar.

The results of the evaluation indicate the usefulness of Elgar to coordinate the efforts of non-experienced as well as experienced participants to accomplish the goals of diagnosis tasks.

The analysis of the questionnaire results shows that in both experiments there was a slightly higher preference for the prescribed coordination mechanisms by participants that do not have experience in professional machine diagnosis and professional participants with experience in machine diagnosis. In both experiments, the prescribed mechanism has a higher preference because of the support prescribing the collaboration process and the ECCs to be used. The processes used in these experiments are designed in advance by experts in diagnosis, based on established collaboration patterns, to support collaboration among team members.

The analysis of the results produced by the evaluation of quality of collaboration reveals the advantages of the ad-hoc coordination mechanism, as professional experts who attended the experiment of Volvo Construction Equipment demonstrate more fluidity of collaboration. The implementation of collaboration processes is based on established patterns that are translated into a set of ECCs, instructions and an overall visualization of the entire process to achieve its desired goals. However, during the analysis of video recordings it became more evident that the transition from one process activity to another pauses the reasoning process of teams. The focus of all participants during the session is on the diagnosis task. They put their efforts to understand the root-causes from the anomalies in the telemetry data and to formulate their conclusions on the report. Whenever, they transit from one activity to another, they pause this reasoning process to understand what is expected from them in a process activity, to understand new instructions and to become familiar with the new provided ECCs. They shift temporarily the focus from the diagnosis task to understanding the new configuration of Elgar.

Such a situation does not occur when participants use Elgar with the ad-hoc coordination mechanism. During their work with this mechanism, participants have constantly the focus on the diagnosis task and they adapt Elgar to match their goals. They have to coordinate themselves to decompose the activities that lead them to achieve the results of diagnosis and formulate the Volvo Case Report. As they know in advance the available ECCs and their purpose, participants customize the user interface of an Elgar session with the appropriate ECCs for their task. For example, some teams prefer to use one CollPad to describe all the information of the Volvo Case Report whereas other teams prefer to use one CollPad to each process activity. The observation of video recordings shows that participants do not interrupt their reasoning process to customize Elgar with ECCs, which reflects the high scores of the ad-hoc mechanism for the dimension "fluidity of collaboration" in comparison to the prescribed mechanism. Therefore, although participants demonstrate their preference with the prescribed mechanism, objectively, their work is more fluid with the ad-hoc mechanism to achieve the results of the diagnosis task.

Despite their differences, both coordination mechanisms have characteristics that support coordination of teams in diagnosis tasks, according to their preferences. The prescribed coordination mechanism provides guidance to teams in their task, which makes participants to execute the diagnosis task quicker and to prefer objectively to use this mechanism. However, the ad-hoc coordination mechanism supports more interaction among participants and makes their collaboration more fluid during the execution of their diagnosis tasks. Therefore, both mechanisms are important for teams to coordinate collaborative diagnosis tasks, the choice of one depends on the characteristics that they value the most, on the one hand more support for the task and on the other more interaction and fluidity of collaboration.

Chapter 7

Discussion and Future Work

Organisations are becoming more globalised, transcending economic, cultural, and social boundaries of countries. The increasing availability of infrastructure to access the Internet and bandwidth capacity of networks empower people and enable them to adapt their ways of communication and interaction with each other through a wide range of devices, such as computer, tablets, smart phones and smart TVs [Gubbi et al., 2013]. Such continuous improvement in ICT laid the foundations for the emergence of new inclusive services that span across geographical boundaries and time differences. Through these technological changes, teams within organisations are shifting from a collocated working paradigm to a geographical distributed paradigm. In such situations coordination becomes important as a process to synchronize actions of teams members to achieve team goals in an efficient way, e.g. to avoid redundant work execution or uneven distribution of tasks [Ellis et al., 1991].

Several coordination mechanisms have been identified in the literature to manage interdependencies in teamwork to achieve results together [Malone and Crowston, 2003] as well as proposals of implementing variations of these mechanisms in software-based shared workspace system to support teamwork [Ellis et al., 1991][Redmiles et al., 2007][Kittur and Kraut, 2010]. However, these mechanisms are currently not provided in systems to support collaborative tasks. Often shared workspace systems choose one coordination mechanism as basis for their implementation and do not offer participants and teams different options with regard to coordination. This thesis takes the position that shared workspace systems that foster collaboration should offer different coordination mechanisms to teams so that users themselves can decide on a mechanism that is most suitable for them to achieve the goals of the task that they are involved into.

This thesis explored the following main research question: *Is it possible to design coordination mechanisms in a shared workspace system to flexibly support collaborative diagnosis in data-centric engineering tasks?*

The main goal of this thesis is to investigate the feasibility to design a software-based shared workspace system that coordinates teamwork in data-centric engineering tasks through different coordination mechanisms, such as diagnosis.

As a response to the research question, this thesis shows that it is possible to design different coordination mechanisms and integrate them in the design of shared workspaces systems to support flexible collaboration. This thesis explored and described the definition of different coordination mechanisms, implemented them in a shared workspace system (Elgar) and evaluated their use in remote diagnosis scenarios that represented data-centric engineering tasks.

Chapter 2 of this thesis explores the following research question: *Can requirements for design coordination mechanisms that flexibly support collaborative diagnosis tasks be identified?*

This chapter identifies and describes a spectrum of coordination ranging between two extremes, unstructured and structured coordination types. Unstructured coordination types handle situations in which teams define their own coordination strategy spontaneously to accomplish the goals of a task. Structured coordination types handle situations in which teams use a structure, such as a collaboration process, to coordinate their efforts to accomplish the goals of a task.

The definition of a coordination spectrum that ranges between these two extremes represents a conceptual framework to define different types of coordination mechanisms and their implementation in a shared workspace system to support teams. In the extreme of unstructured coordination, one possible mechanism (ad-hoc coordination mechanism) provides workspace awareness information to support participants to define coordination strategies or take coordinated actions. In the extreme of structured coordination, another possible coordination mechanism (prescribed coordination mechanism) provides a collaboration process with activities that guide the actions of teams. Additionally, although not defined or implemented in this thesis, a coordination mechanism situated in the middle of the spectrum may use aspects of both extremes. For example, such a coordination mechanism may have a process with major global activities that aid the accomplishment of the goals of a task and use workspace awareness information to provide real-time recommendations about the coordination of their work and different approaches to achieve theirs goals.

Chapter 2 concludes with a list of requirements to design shared workspace systems to be flexible with regard to the available coordination mechanisms that it supports and offer to the collaboration of distributed teams.

Chapter 3 of this thesis explored the following research question: *What are the requirements to support collaboration of teams in data-centric machine diagnosis tasks?*

This chapter discusses remote machine diagnosis and the use of diagnosis models in general. The chapter proposes a new diagnosis model, named Rectio, that extends earlier diagnosis models with generation and documentation of solutions and support of collaboration for team members involved in diagnosis tasks. The model uses elements that are present in traditional diagnosis models designed to support individuals in the analysis of observed anomalies and formulation of diagnosis and extends them to support collaboration of distributed team members.

Rectio decomposes each of the diagnosis activities of traditional diagnosis models and includes further sub-activities that support collaboration. For example, instead of defining one activity (e.g. hypothesis definition) Rectio distinguishes three sub activities that must be executed by team members together: generate hypotheses for a machine anomaly, evaluate them and select the most probable hypothesis that causes an anomaly. Rectio also extends traditional diagnosis models to include activities that enhance diagnosis tasks. Rectio includes an activity in which experts acquire required data to diagnose a machine, and an activity in which experts analyse the acquired data to create an initial assessment of a machine situation. In addition, Rectio includes activities that support team members to define actions that overcome machine anomalies, document them and evaluate their effectiveness.

Chapter 3 concludes with a list of requirements to design shared workspace systems that support collaboration of distributed teams in the execution of diagnosis tasks with particular diagnosis functionalities.

Chapter 4 and 5 of this thesis explored the following research question: *Can an architecture be designed to support flexible coordination of teams involved in data-centric collaborative diagnosis tasks?*

As a response to this research question, chapter 4 describes the architecture of Elgar that supports the co-existence of different coordination mechanisms, chosen by distributed teams in remote machine diagnosis task. The architecture also describes the realization of extensibility of functionality in Elgar to support diagnosis tasks and to integrate additional types of coordination mechanisms.

Elgar uses the concept of ECCs to modularize and divide available functionalities; each component is self-contained and provides specific diagnosis functionalities to teams. ECCs are arranged in diagnosis sessions, as in a dashboard, that centralizes information about a machine situation. Elgar sessions use the concept of dashboards that present the overview of a diagnosis situation through data. The sessions use ECCs to provide specific machine telemetry data or enable information exchange between team members to support them in diagnosis tasks.

Coordination mechanisms benefit from ECCs to implement their respective coordination support. Due to the use of such a modular architecture, coordination mechanisms reuse the same ECCs for different purposes. Whereas the ad-hoc coordination mechanism enables participants to select and instantiate the ECCs of their interest to execute the diagnosis task, the prescribed coordination mechanism provides the ECCs of a collaboration process activity automatically for their convenience. In addition, ECCs use flexible schemas that support a continuous extensibility of a diagnosis data model in Elgar. In such a data model, each ECC defines for itself a schema dynamically and have different and complementary properties and structures.

Chapter 4 concludes describing the approach used by the architecture to satisfy the requirements specified in the chapters 2 and 3. Therefore, it is possible to design an architecture for a shared workspace system that is extensible to integrate different coordination mechanisms and flexible to enable teams to choose the most appropriate one for a remote diagnosis task. Chapter 5 describes implementation details, e.g. technologies, frameworks and techniques, used for the implementation of the architecture of Elgar.

Finally, the chapter 6 of this thesis explored the following research question: *Can a shared workspace system flexibly coordinate collaboration in data-centric diagnosis tasks?*

This chapter evaluates the usefulness of the diagnosis functionalities provided by Elgar and its coordination mechanisms. The evaluation of two experiments indicates that there are no significant differences between the use of unstructured and structured coordination mechanisms for teams that used Elgar in diagnosis tasks. According to the results described in the evaluation chapter, the diagnosis functionalities implemented in Elgar in both coordination mechanisms (ad hoc and prescribed) are effective to support participants in the experiments to accomplish the goals of their diagnosis tasks.

Each of the coordination mechanisms is based on different constructs that present advantages over the other with regard to teamwork. Structured coordination provides guidance to teams through processes to accomplish the goals of the tasks in which they are involved. This type of coordination is based on the notion that processes represent best practices, learnt and enhanced over time that support teams to achieve specific goals [Jablonski and Bussler, 1996]. On the other hand, the unstructured type of coordination supports unrestricted collaboration of teams that should not be constrained by pre-defined processes. Teams should establish themselves, a coordination strategy dynamically to accomplish the goals of a task [Bellotti and Bly, 1996]. In this case, the only necessary support is to provide workspace awareness information for teams, so that they can agree on the coordination strategy based on a continuous update of such information. As coordination mechanisms provide similar outcomes in the context of this thesis, the evaluation of the two experiments indicate that the use of a specific coordination mechanism over the other does not provide more support or improve the outcomes of teamwork. Rather, the use of coordination mechanisms become flexible options of a shared workspace system, chosen by teams based on their teamwork preferences to execute a task. Therefore, Elgar can help coordinate collaboration flexibly, offering different options of coordination mechanisms for distributed teams, that decide on the most appropriate for them to execute diagnosis tasks collaboratively.

This thesis sets the initial steps to design and develop a flexible shared workspace system (Elgar) that supports different coordination mechanisms, used during diagnosis tasks. It uses two experiments to evaluate the use of Elgar and gives initial insights, discussed in this section. These insights support research towards a definitive clarification of the flexible use of coordination mechanisms in tasks involving teamwork. Future research should identify and detail evaluation scenarios that consider: *i)* a higher number of teams, *ii)* more participants in each team, and *iii)* more complex diagnosis tasks that require team collaboration during longer periods of time, such as a few days.

7.1 Contributions

This thesis has the following main contributions: *i)* a conceptual spectrum of coordination types and *ii)* a diagnosis model based on collaboration of experts *iii)* an extensible software shared workspace system to ease the inclusion of diagnosis functionalities and coordination mechanisms, *iv)* an evaluation methodology that combines different methods to assess the usefulness of shared workspace systems in teamwork and *v)* the evaluation of functionalities and coordination mechanisms in real diagnosis scenarios provided by two different manufacturing companies.

The first contribution of this thesis is the definition of a spectrum of coordination support for teamwork. One extreme of the spectrum corresponds to a highly specified support and the other extreme to a highly unspecified support. The spectrum assumes that each coordination type is important and its use depends on the context of the execution of a task. This is a different approach from [Redmiles et al., 2007], that proposes a combination of the characteristics of both extremes as a solution to avoid the dichotomy of structured and unstructured coordination. It is also a different approach from [Bernstein, 2000], that uses the specificity frontier to cover organizational processes through fixed processes described in workflow systems and dynamic processes that emerge from communication support systems.

The second contribution of this thesis is the definition of Rectio, a collaborative diagnosis model. Different models [Benjamins, 1993][Brazier and Treur, 1994][Brazier et al., 1996] focus on the intrinsic activities that occur in a diagnosis model, such symptom detection and hypothesis generation. However, certain complex equipment requires the collaboration of experts, combining their different range of skills to solve a symptom. Rectio, instead, focus on the collaboration aspect of the diagnosis task and extends these models supporting the collaboration of different experts.

The third contribution is the implementation of Elgar, a shared workspace system for diagnosis that provides two coordination mechanisms for teamwork. There are different shared workspace systems that provide diagnosis functionalities to support teams in diagnosis tasks [Garcia et al., 2004][Wang et al., 2004][Liao and Lee, 2010][Sekar et al., 2011]. However, such systems chose during their design phase a particular coordination mechanism for collaboration and lack to provide flexibility with regard to coordination in teamwork. Elgar not only provides diagnosis functionalities but, being based on the spectrum of coordination, also allows teams to decide the coordination mechanism to use in diagnosis tasks.

The fourth contribution of this thesis is to combine different evaluation methods to evaluate Elgar in supporting teams and the quality of collaboration through Elgar. The evaluation approach combines four evaluation methods: *i)* evaluates the usefulness of Elgar supporting participants in diagnosis sessions, *ii)* evaluates the usefulness of the workspace awareness information provided by Elgar, *iii)* the usability of Elgar and *iv)* the quality of collaboration for teams that use Elgar. Often, software systems are evaluated through usability methods and the workspace awareness information they provide. In addition to these methods, this thesis uses a method to measure the usefulness of diagnosis sessions mediated by Elgar and to measure the quality of collaboration of teams in diagnosis. Finally, the last contribution of this thesis is to use Elgar in a real machine diagnosis scenario, co-designed with Volvo Construction Equipment. Often, systems are tested in constrained environments, such as a laboratory, in which participants are not the end-users. However, this thesis describes the use of Elgar in a real diagnosis situation using diagnosis experts that are the end-users of the system.

7.2 Future Work

Future research should focus on the execution of longitudinal tests to confirm that collaboration outcomes of teams that use unstructured or structured coordination mechanisms are similar. The evaluation of the two experiments in this thesis represent an initial indication that supports the aforementioned results, however they are limited with regard to the amount of available time that teams had to become more experienced with the use of Elgar. The limited training time may have not be sufficient to provide the appropriate experience in the use Elgar in diagnosis sessions. For example, teams that used the ad-hoc coordination mechanism took longer to execute their diagnosis task. However, this situation may relate to the fact that this coordination mechanism has more details to be learned and the teams did not have enough time to become experienced enough with the system, increasing therefore the time to execute their diagnosis task.

Another limitation concerns the number of observed teams and their participants. Future work should increase the overall number of evaluated participants to reduce the margin of error and increase the confidence interval of the answers of surveys that measure the usefulness of Elgar in diagnosis sessions, the use of workspace awareness information and usability of Elgar. A higher number of participants also benefits the evaluation of the quality of collaboration within teams that use a particular coordination mechanism, as it provides more variation in the results that show the interaction between participants during diagnosis tasks. However, a high number of participants would also require specific tools that ease the evaluation of the interaction among participants in teams [Biancucci et al., 2014b]. In addition to team size, team composition is of importance. Team composition may influence the use of different coordination mechanisms. For example, the role of participant, e.g. problem owner or team leader, may automatically influence a team using the ad-hoc coordination mechanism in a diagnosis task.

The improvement of future evaluation of Elgar and of teamwork would also require a repository of several diagnosis situations containing simulated machine anomalies. The experience designing the experiments in this thesis shows that this is a challenging task. For example, in the case of Volvo Construction Equipment, experienced engineers had to identify complex machine anomalies, and reproduce their behaviour through telemetry data, that had not been previously experienced by their colleagues who attended evaluation sessions. In addition, the occurrence of complex types of scenarios is seldom and the number of engineers who did not participate in the diagnosis of such scenarios was also low, limiting the number of potential available scenarios that could be used in the repository of diagnosis situations.

Another important aspect for future work is to evaluate teams of different sizes to identify whether specific coordination mechanisms have limitations according to the size of a team. For example, one hypothesis is that the ad-hoc coordination mechanism might not be suitable for larger teams, as the cognitive capacity of team members might decrease handling a higher amount of workspace awareness information to define and apply coordination strategies.

Future research should also make a more detailed analysis about teamwork to empirically derive different variations of coordination mechanisms between the two extremes of the spectrum. In addition to the two implemented coordination mechanisms, this thesis mentions one possible implementation of a coordination mechanism that theoretically exists in the middle of the spectrum of coordination. It is the recommendation coordination that uses aspects of both structured and unstructured coordination types. Such a mechanisms could use process and workspace awareness information to provide recommendations to teams to efficiently achieve the goals of their diagnosis task, using rules that adapt system according to dynamic changes in diagnosis situation [Haake et al., 2010][Knoll et al., 2013]. However, it is necessary to investigate new possibilities of coordination mechanisms that combine characteristics of structured and unstructured coordination mechanisms. Not only the identification of coordination mechanisms for teams is important but also its evaluation in diagnosis tasks that require coordination.

An important aspect to consider as a future work for Elgar, involves the need to enable the transition between different coordination mechanisms in the same diagnosis session, as the results discussed in this thesis indicate the need for both coordination mechanisms. The possibility of a transition between coordination mechanisms may be useful, for example, whenever a collaboration process described for a diagnosis sessions does not effectively to support a team in diagnosis tasks. In specific situations, a prescribed process may well not suffice, mandating improvisation and creation of a unique process during a diagnosis task. Conversely, during a diagnosis task a team may realise that (parts of) the diagnosis task that were thought to be unique, are not.A prescribed process could provide support. A transition from prescribed to the ad-hoc coordination mechanism is a straightforward transition, as software components once associated with the process, are detached from process activities. However, a transition from an ad-hoc to a prescribed coordination mechanism is less straightforward, requiring interpretation of real-time status information to be related to the prescribed coordination task. One particular approach for such situation is to dynamically associate software components to process activities based on their context of use [Haake et al., 2010].

In the context of this thesis, Elgar is a prototype of a shared workspace system used in two different experiments based on real diagnosis scenarios. However, at this stage of design, the system is not yet prepared to be deployed in a variety of diagnosis domains such as, urban transportation vehicles, satellites or medical devices. The research in this thesis indicates a set of requirements to create a shared workspace system for a specific domain, the one of construction machines. As a continuation of the research initiated in this thesis, it is necessary to build shared workspace systems that support teamwork coordination in diagnosis for other domains. Afterwards, it is necessary to extract the knowledge to design a methodology that instantiates workspace system for collaborative diagnosis that support different coordination mechanisms for various domains.

Finally, as a continuation of this research, it is necessary to further investigate the use of Elgar and the different coordination mechanisms in relation to current solutions used for collaborative diagnosis in workplaces. Although this thesis discusses the usefulness of diagnosis functionalities and the difference of using different coordination mechanisms, it is necessary to compare Elgar with systems and tools used by engineers in different companies to approach diagnosis tasks. Such comparison is valuable to gain insights on the most efficient and preferred solution for engineers to successfully diagnose machines.

As the use of coordination mechanisms provides similar outcomes for diagnosis teams, the selection of a specific coordination mechanisms should be based on the work preferences of teams [Greenberg, 1991][Bernstein, 2000]. Based on these outcomes, manufacturing companies that are involved in diagnosis tasks should consider for future shared workspace systems that different options of coordination mechanisms for teams should be supported. Although such systems are constantly improving their functionalities, there are still limited initiatives to investigate and develop such type of software systems that could be used in production environments of real diagnosis situations and not only in controlled experimental environments. The support of coordination by such systems and the possibility for teams to choose the most suitable coordination mechanism for them can only propitiate more satisfaction to a higher number of engineers involved in diagnosis tasks.

Bibliography

[Aichholzer and Westholm, 2009] Aichholzer, G. and Westholm, H. (2009). Evaluating eparticipation projects: practical examples and outline of an evaluation framework. *European Journal of ePractice*, 7(3):1–18.

[Alliance, 2015] Alliance, S. (2015). The 2015 state of scrum report. *Scrum Alliance*, 3.

[Alvarez and Hall, 2003] Alvarez, R. M. and Hall, T. E. (2003). *Point, click, and vote: The future of Internet voting.* Brookings Institution Press.

[Alzghoul et al., 2014] Alzghoul, A., Backe, B., Löfstrand, M., Byström, A., and Liljedahl, B. (2014). Comparing a knowledge-based and a data-driven method in querying data streams for system fault detection: A hydraulic drive system application. *Computers in Industry*, 65(8):1126–1135.

[Arnaiz et al., 2010] Arnaiz, A., Iung, B., Adgar, A., Naks, T., Tohver, A., Tommingas, T., and Levrat, E. (2010). Information and communication technologies within e-maintenance. In Holmberg, K., Adgar, A., Arnaiz, A., Jantunen, E., Mascolo, J., and Mekid, S., editors, *E-maintenance*, pages 39–60. Springer London.

[Azevedo et al., 2013] Azevedo, D., Fonseca, B., Paredes, H., Lukosch, S., Janeiro, J., and Briggs, R. O. (2013). On the development and usability of a diagram-based collaborative brainstorming component. *J. UCS*, 19(7):873–893.

[Bauleo et al., 2014] Bauleo, E., Carnevale, S., Catarci, T., Kimani, S., Leva, M., and Mecella, M. (2014). Design, realization and user evaluation of the smartvortex visual query system for accessing data streams in industrial engineering applications. *Journal of Visual Languages & Computing*, 25(5):577–601.

[Bellotti and Bly, 1996] Bellotti, V. and Bly, S. (1996). Walking away from the desktop computer: Distributed collaboration and mobility in a product design team. In *Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work*, CSCW '96, pages 209–218, New York, NY, USA. ACM.

[Bellotti et al., 2003] Bellotti, V., Ducheneaut, N., Howard, M., and Smith, I. (2003). Taking email to task: the design and evaluation of a task management centered email tool. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 345–352. ACM.

[Benjamin, 2015] Benjamin, M. (2015). Crowdsourcing microdata for cost-effective and reliable lexicography. In *AsiaLex 2015 Conference Proceedings*.

[Benjamins, 1993] Benjamins, R. (1993). *Problem-Solving Methods for Diagnosis*. PhD thesis, University of Amsterdam.

[Bernstein, 2000] Bernstein, A. (2000). How can cooperative work tools support dynamic group process? bridging the specificity frontier. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, pages 279–288. ACM.

[Biancucci et al., 2014a] Biancucci, M., Janeiro, J., Leva, M., Lukosch, S., and Mecella, M. (2014a). Collaboration in virtual enterprises through the smart vortex suite. In *Collaborative Systems for Smart Networked Environments*, pages 426–434. Springer.

[Biancucci et al., 2014b] Biancucci, M., Janeiro, J., Mecella, M., and Lukosch, S. (2014b). Supporting industrial processes by monitoring and visualizing collaborations. In *Innovative Design and Manufacturing (ICIDM), Proceedings of the 2014 International Conference on*, pages 330–335. IEEE.

[Biehl et al., 2007] Biehl, J. T., Czerwinski, M., Smith, G., and Robertson, G. G. (2007). Fastdash: a visual dashboard for fostering awareness in software teams. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1313–1322, New York, NY, USA. ACM.

[Bloch and Geitner, 1997] Bloch, H. P. and Geitner, F. K. (1997). *Practical Machinery Management for Process Plants: Volume 2: Machinery Failure Analysis and Troubleshooting*. Gulf Professional Publishing.

[Boehm et al., 2001] Boehm, B., Grunbacher, P., and Briggs, R. (2001). Developing groupware for requirements negotiation: lessons learned. *Software, IEEE*, 18(3):46–55.

[Bordet, 2016] Bordet, S. (2016). The cometd reference book 3.0.9.

[Bostrom et al., 1993] Bostrom, R. P., Anson, R., and Clawson, V. K. (1993). Group facilitation and group support systems. *Group support systems: New perspectives*, pages 146–168.

[Brannick and Prince, 1997] Brannick, M. T. and Prince, C. (1997). An overview of team performance measurement. *Team performance assessment and measurement. Theory, methods, and applications*, 3.

[Brazier and Treur, 1994] Brazier, F. M. and Treur, J. (1994). User centered knowledge-based system design: a formal modelling approach. In *A Future for Knowledge Acquisition*, pages 283–302. Springer.

[Brazier et al., 1996] Brazier, F. M. T., Treur, J., and Wijngaards, N. J. E. (1996). Modelling interaction with experts: The role of a shared task model. In *12th European Conference on Artificial Intelligence, Budapest, Hungary, August 11-16, 1996, Proceedings*, pages 241–245.

[Briggs et al., 2003a] Briggs, R., de Vreede, G., and Reinig, B. A. (2003a). A theory and measurement of meeting satisfaction. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, pages 8 pp.–.

[Briggs and de Vreede, 2009] Briggs, R. and de Vreede, G.-J. (2009). *ThinkLets: Building Blocks for concerted Collaboration*. Briggs and de Vreede, Nebraska.

[Briggs et al., 2001] Briggs, R., De Vreede, G.-J., Nunamaker, J., and Tobey, D. (2001). ThinkLets: achieving predictable, repeatable patterns of group interaction with group support systems (GSS). In *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, page 9, Washington DC. IEEE Computer Society.

[Briggs et al., 2006] Briggs, R., Kolfschoten, G., Gert-Jan, V., and Douglas, D. (2006). Defining key concepts for collaboration engineering. *AMCIS 2006 Proceedings*, page 17.

[Briggs et al., 2003b] Briggs, R. O., De Vreede, G.-J., and Nunamaker, Jr., J. F. (2003b). Collaboration engineering with thinklets to pursue sustained success with group support systems. *J. Manage. Inf. Syst.*, 19(4):31–64.

[Briggs et al., 2010] Briggs, R. O., Kolfschoten, G. L., de Vreede, G.-J., Albrecht, C. C., and Lukosch, S. G. (2010). Facilitator in a box: Computer assisted collaboration engineering and process support systems for rapid development of collaborative applications for high-value tasks. In *Proceedings of the 2010 43rd Hawaii International Conference on System Sciences*, HICSS '10, pages 1–10, Washington, DC, USA. IEEE Computer Society.

[Burkhardt et al., 2009] Burkhardt, J.-M., Détienne, F., Hébert, A.-M., Perron, L., safin, S., and Leclercq, P. (2009). An approach to assess the quality of collaboration in technology-mediated design situations. In *European Conference on Cognitive Ergonomics: Designing Beyond the Product — Understanding Activity and User Experience in Ubiquitous Environments*, ECCE '09, pages 30:1–30:9, VTT, Finland, Finland. VTT Technical Research Centre of Finland.

[Buschmann et al., 2008] Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., and Stal, M. (2008). *Pattern-Oriented Software Architecture: A System of Patterns*. Wiley India Pvt. Limited.

[Buttler et al., 2011] Buttler, T., Janeiro, J., Lukosch, S., and Briggs, R. O. (2011). Beyond gss: fitting collaboration technology to a given work practice. In *International Conference on Collaboration and Technology*, pages 126–141. Springer.

[Carmel, 1999] Carmel, E. (1999). *Global Software Teams: Collaborating Across Borders and Time Zones*. Prentice Hall PTR, Upper Saddle River, NJ, USA.

[Carmel et al., 2010] Carmel, E., Espinosa, J. A., and Dubinsky, Y. (2010). "follow the sun" workflow in global software development. *Journal of Management Information Systems*, 27(1):17–38.

[Castells et al., 2006] Castells, M., Cardoso, G., et al. (2006). *The Network Society: From Knowledge to Policy*. Johns Hopkins Center for Transatlantic Relations Washington, DC.

[Chase, 1999] Chase, N. (1999). Learning to lead a virtual team. *Quality*, 38(9):76.

[Cheng et al., 2003] Cheng, L.-T., Hupfer, S., Ross, S., and Patterson, J. (2003). Jazzing Up Eclipse with Collaborative Tools. In *Proceedings of the 2003 OOPSLA workshop on eclipse technology eXchange - eclipse '03*, pages 45–49, New York, New York, USA. ACM Press.

[Cohen and Bailey, 1997] Cohen, S. G. and Bailey, D. E. (1997). What makes teams work: Group effectiveness research from the shop floor to the executive suite. *Journal of Management*, 23(3):239 – 290. A Special Issue Of The Journal Of Management.

[Consortium, 2010] Consortium, T. S. V. (2010). Scalable semantic product data stream management for collaboration and decision making in engineering. Technical report.

[Constant et al., 1996] Constant, D., Sproull, L., and Kiesler, S. (1996). The kindness of strangers: The usefulness of electronic weak ties for technical advice. *Organization science*, 7(2):119–135.

[Crowston, 1994] Crowston, K. (1994). A Taxonomy of Organizational Dependencies and Coordination Mechanisms. Working Paper Series 174, MIT Center for Coordination Science.

[Datcu et al., 2014] Datcu, D., Cidota, M., Lukosch, S., Oliveira, D. M., and Wolff, M. (2014). Virtual co-location to support remote assistance for inflight maintenance in ground training for space missions. In *Proceedings of the 15th International Conference on Computer Systems and Technologies*, CompSysTech '14, pages 134–141, New York, NY, USA. ACM.

[Dickson et al., 1996] Dickson, G. W., Lee-Partridge, J. E., Limayem, M., and Desanctis, G. L. (1996). Facilitating computer-supported meetings: A cumulative analysis in a multiple-criteria task environment. *Group Decision and Negotiation*, 5(1):51–72.

[Dix et al., 2003] Dix, A., Finlay, J. E., Abowd, G. D., and Beale, R. (2003). *Human-Computer Interaction (3rd Edition)*. Prentice Hall, 3 edition.

[Dreyfus and Dreyfus, 1980] Dreyfus, S. E. and Dreyfus, H. L. (1980). A Five-Stage Model of the Mental Activities Involved in Directed Skill Acquisition.

[Dustdar, 2004] Dustdar, S. (2004). Caramba - a process-aware collaboration system supporting ad hoc and collaborative processes in virtual teams. *Distributed and Parallel Databases*, 15(1):45–66.

[Ellis et al., 2005] Ellis, C. A., Barthelmess, P., Chen, J., and Wainer, J. (2005). Person-to-person processes: Computer-supported collaborative work. *Process-Aware Information Systems*, page 37.

[Ellis et al., 1991] Ellis, C. A., Gibbs, S. J., and Rein, G. (1991). Groupware: Some issues and experiences. *Commun. ACM*, 34(1):39–58.

[Endsley, 1995] Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):32–64.

[Espinosa et al., 2002] Espinosa, A., Lerch, J., Kraut, R., Salas, E., Fiore, S. M., Cannon-Bowers, J. A., et al. (2002). Explicit vs. implicit coordination mechanisms and task dependencies: One size does not fit all. *Forthcoming in team cognition: Process and performance at the inter-and intra-individual level*.

[Espinosa et al., 2007] Espinosa, J. A., Slaughter, S. A., Kraut, R. E., and Herbsleb, J. D. (2007). Team knowledge and coordination in geographically distributed software development. *Journal of Management Information Systems*, 24(1):135–169.

[Fisher et al., 1997] Fisher, S. G., Hunter, T. A., and Macrosson, W. K. (1997). Team or group? managers perceptions of the differences. *Journal of Managerial Psychology*, 12(4):232–242.

[Galletta et al., 2004] Galletta, D. F., Henry, R., McCoy, S., and Polak, P. (2004). Web site delays: How tolerant are users? *Journal of the Association for Information Systems*, 5(1):1.

[Garcia et al., 2004] Garcia, E., Guyennet, H., Lapayre, J., and Zerhouni, N. (2004). A New Industrial Cooperative Tele-Maintenance platform. *Computers & Industrial Engineering*, 46(4):851–864.

[Geyer et al., 2006] Geyer, W., Muller, M. J., Moore, M. T., Wilcox, E., Cheng, L.-T., Brownholtz, B., Hill, C., and Millen, D. R. (2006). Activity explorer: activity-centric collaboration from research to product. *IBM Systems Journal*, 45(4):713–738.

[Greenberg, 1991] Greenberg, S. (1991). Personalizable groupware: Accommodating individual roles and group differences. In *Proceedings of the Second European Conference on Computer-Supported Cooperative Work ECSCW91*, pages 17–31. Springer.

[Greenberg and Marwood, 1994] Greenberg, S. and Marwood, D. (1994). Real time groupware as a distributed system: Concurrency control and its effect on the interface. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, CSCW '94, pages 207–217, New York, NY, USA. ACM.

[Griffith et al., 1998] Griffith, T. L., Fuller, M. A., and Northcraft, G. B. (1998). Facilitator influence in group support systems: intended and unintended effects. *Information Systems Research*, 9(1):20–36.

[Gubbi et al., 2013] Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645–1660.

[Gutwin and Greenberg, 1996] Gutwin, C. and Greenberg, S. (1996). Workspace awareness for groupware. In *Conference Companion on Human Factors in Computing Systems*, CHI '96, pages 208–209, New York, NY, USA. ACM.

[Gutwin and Greenberg, 2002] Gutwin, C. and Greenberg, S. (2002). A descriptive framework of workspace awareness for real-time groupware. *Computer Supported Cooperative Work (CSCW)*, 11(3-4):411–446.

[Gutwin et al., 2008] Gutwin, C., Greenberg, S., Blum, R., Dyck, J., Tee, K., and McEwan, G. (2008). Supporting informal collaboration in shared-workspace groupware. *J. UCS*, 14(9):1411–1434.

[Gutwin et al., 2011] Gutwin, C. A., Lippold, M., and Graham, T. C. N. (2011). Real-time groupware in the browser: Testing the performance of web-based networking. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work*, CSCW '11, pages 167–176, New York, NY, USA. ACM.

[Haake et al., 2010] Haake, J. M., Hussein, T., Joop, B., Lukosch, S., Veiel, D., and Ziegler, J. (2010). Modeling and exploiting context for adaptive collaboration. *International Journal of Cooperative Information Systems*, 19(01n02):71–120.

[Hammer et al., 1977] Hammer, M., Howe, W. G., Kruskal, V. J., and Wladawsky, I. (1977). A very high level programming language for data processing applications. *Commun. ACM*, 20(11):832–840.

[Heidbrink, 2013] Heidbrink, H.-U. (2013). Deliverable 13.1: Smart data stream enabled decision making and cooperation support in complex design and engineering environments. Technical report, The Smart Vortex Project.

[Heidbrink et al., 2013] Heidbrink, H.-U., Engel, F., Grabarske, J., Risch, T., Lindström, J., Löfstrand, M., Karlsson, L., Johanson, M., Brocks, H., Brunsmann, J., Janeiro, J., Wedlund, D., Larsson, J., Byström, A., and Jens, H. (2013). Deliverable 15.1: Specification of use cases. Technical report, The Smart Vortex Project.

[Heidbrink et al., 2011] Heidbrink, H.-U., Lindström, J., Löfstrand, Magnus Johansson, M., Lukosch, S., Risch, T., Heutelbeck, D., Grabarske, J., Byström, A., Kalhori, V., Helgosson, M., Wedlund, D., Larsson, J., Radomski, S., Mecella, M., Engel, F., Knoll, S., Janeiro, J., and Hundsdörfer, A. (2011). Deliverable 1.1: Survey of users in europe and smart vortex usage scenarios and cooperation support. Technical report, The Smart Vortex Project.

[Hevner and Chatterjee, 2010] Hevner, A. and Chatterjee, S. (2010). *Design Research in Information Systems: Theory and Practice (Integrated Series in Information Systems).* Springer, 1st edition. edition.

[Hinds and McGrath, 2006] Hinds, P. and McGrath, C. (2006). Structures that work: Social structure, work structure and coordination ease in geographically distributed teams. In *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work*, CSCW '06, pages 343–352, New York, NY, USA. ACM.

[Holmberg et al., 2010] Holmberg, K., Adgar, A., Arnaiz, A., Jantunen, E., Mascolo, J., and Mekid, S. (2010). *E-maintenance.* Springer Science & Business Media.

[Hürsch and Videira Lopes, 1995] Hürsch, W. and Videira Lopes, C. (1995). Separation of concerns. Technical Report NU-CCS-95-03, Northeastern University.

[Iung et al., 2009] Iung, B., Levrat, E., Marquez, A. C., and Erbe, H. (2009). Conceptual framework for e-maintenance: Illustration by e-maintenance technologies and platforms. *Annual Reviews in Control*, 33(2):220–229.

[Jablonski and Bussler, 1996] Jablonski, S. and Bussler, C. (1996). *Workflow Management: Modeling Concepts, Architecture and Implementation.* International Thomson Computer Press.

[Janeiro et al., 2012a] Janeiro, J., Knoll, S. W., Lukosch, S., and Kolfschoten, G. (2012a). Adaptive collaboration support systems: Designing collaboration support for dynamic environments. In *CESUN 2012: 3rd International Engineering Systems Symposium, Delft University of Technology, The Netherlands, 18-20 June 2012*.

[Janeiro et al., 2012b] Janeiro, J., Knoll, S. W., Lukosch, S. G., Kolfschoten, G. L., and Brazier, F. M. (2012b). Designing collaboration support for dynamic environments. *Volume II*, page 29.

[Janeiro et al., 2012c] Janeiro, J., Lukosch, S., and Brazier, F. (2012c). Elastic collaboration support: from workflow-based to emergent collaboration. In *Proceedings of the 17th ACM international conference on Supporting group work*, pages 317–320. ACM.

[Janeiro et al., 2014] Janeiro, J., Lukosch, S., Brazier, F. M. T., Leva, M., Mecella, M., and Byström, A. (2014). *Evaluating Coordination Support Mechanisms in an Industrial Engineering Scenario*, pages 215–222. Springer International Publishing, Cham.

[Janeiro et al., 2013] Janeiro, J., Lukosch, S., Radomski, S., Johanson, M., Mecella, M., and Larsson, J. (2013). Supporting elastic collaboration: Integration of collaboration components in dynamic contexts. In *Proceedings of the 5th ACM SIGCHI symposium on Engineering interactive computing systems*, pages 127–132. ACM.

[Jarvenpaa and Leidner, 1998] Jarvenpaa, S. L. and Leidner, D. E. (1998). Communication and trust in global virtual teams. *Journal of Computer-Mediated Communication*, 3(4).

[Java Community Process, 2008] Java Community Process (2008). Java specification requests 286: Portlet specification 2.0.

[Jerry Fjermestad, 2000] Jerry Fjermestad, S. R. H. (2000). Group support systems: A descriptive evaluation of case and field studies. *Journal of Management Information Systems*, 17(3):115–159.

[Johansen, 1988] Johansen, R. (1988). *Groupware: Computer support for business teams*. The Free Press.

[Johanson et al., 2014] Johanson, M., Janeiro, J., Lukosch, S., and Engel, F. (2014). Deliverable 9.1: Intelligent collaboration studio report and specification. Technical report, The Smart Vortex Project.

[Johanson et al., 2013] Johanson, M., Lindström, J., Byström, A., Wedlund, D., Jonas, L., Knoll, S., and Heutelbeck, D. (2013). Deliverable 14.1: Specification of use cases. Technical report, The Smart Vortex Project.

[Karlsson et al., 2012] Karlsson, L., Lindstrm, J., and Lfstrand, M. (2012). Functional products - goodbye to the industrial age. *Ericsson Business Review*, 18(2):21–24.

[Katzenbach, 1993] Katzenbach, J. R. (1993). *The wisdom of teams: Creating the high-performance organization*. Harvard Business Press.

[Kittur and Kraut, 2010] Kittur, A. and Kraut, R. E. (2010). Beyond wikipedia: Coordination and conflict in online production groups. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, CSCW '10, pages 215–224, New York, NY, USA. ACM.

[Knoll et al., 2014] Knoll, S., Johanson, M., Kolfschoten, G., Lukosch, S., Janeiro, J., Mecella, M., Leva, M., and Biancucci, M. (2014). Deliverable 10.1: Semantic model of collaboration metadata and mechanisms. Technical report, The Smart Vortex Project.

[Knoll et al., 2009] Knoll, S. W., Horning, M., and Horton, G. (2009). Applying a thinklet-and thinxel-based group process modeling language: A prototype of a universal group support system. In *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on*, pages 1–10. IEEE.

[Knoll et al., 2013] Knoll, S. W., Janeiro, J., Lukosch, S. G., and Kolfschoten, G. L. (2013). A semantic model for adaptive collaboration support systems. In *Semantic Models for Adaptive Interactive Systems*, pages 59–81. Springer.

[Kraut and Streeter, 1990] Kraut, R. E. and Streeter, L. A. (1990). Coordination in large scale software development. Technical report, DTIC Document.

[Lee and Paine, 2015] Lee, C. P. and Paine, D. (2015). From the matrix to a model of coordinated action (moca): A conceptual framework of and for cscw. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work &#38; Social Computing*, CSCW '15, pages 179–194, New York, NY, USA. ACM.

[Lee, 2007] Lee, M. (2007). *Usability of collaboration technologies.* PhD thesis, University of Amsterdam.

[Leva et al., 2014] Leva, M., Mecella, M., Summa, D., Biancucci, M., and Tole, A. (2014). Deliverable 6.1: Collaborative user interface and visual query language report and specification. Technical report, The Smart Vortex Project.

[Levrat et al., 2008] Levrat, E., Iung, B., and Crespo Marquez, A. (2008). E-maintenance: review and conceptual framework. *Production Planning & Control*, 19(4):408–429.

[Lewis, 1995] Lewis, J. R. (1995). Ibm computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1):57–78.

[Liao and Lee, 2010] Liao, L. and Lee, J. (2010). Design of a reconfigurable prognostics platform for machine tools. *Expert Systems with Applications*, 37(1):240 – 252.

[Lipnack, 1997] Lipnack, J. (1997). *Virtual teams: Reaching across space, time, and organizations with technology.* Jeffrey Stamps.

[Malagoli et al., 2013] Malagoli, A., Leva, M., Kimani, S., Russo, A., Mecella, M., Bergamaschi, S., and Catarci, T. (2013). Visual query specification and interaction with industrial engineering data. In *Advances in Visual Computing - 9th International Symposium, ISVC 2013, Proceedings - Part II*, volume 8034 of *Lecture Notes in Computer Science*, pages 58–67. Springer.

[Malone and Crowston, 1990] Malone, T. W. and Crowston, K. (1990). What is coordination theory and how can it help design cooperative work systems? In *Proceedings of the 1990 ACM conference on Computer-supported cooperative work*, pages 357–370. ACM.

[Malone and Crowston, 1994] Malone, T. W. and Crowston, K. (1994). The interdisciplinary study of coordination. *ACM Comput. Surv.*, 26(1):87–119.

[Malone and Crowston, 2003] Malone, T. W. and Crowston, K. (2003). Toward an interdisciplinary theory of coordination. Working papers 120, Massachusetts Institute of Technology (MIT), Sloan School of Management.

[March and Simon, 1958] March, J. G. and Simon, H. A. (1958). Organizations. *University of Illinois at Urbana-Champaign's Academy for Entrepreneurial Leadership Historical Research Reference in Entrepreneurship.*

[Marcus, 2006] Marcus, A. (2006). Dashboards in your future. *interactions*, 13(1):48–60.

[Martin, 1983] Martin, J. (1983). *Managing the Data Base Environment.* Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition.

[Mayer and Pinto, 1998] Mayer, M. and Pinto, J. K. (1998). *The Virtual Edge: Embracing Technology for Distributed Project Team Success.* Project Management Institute.

[McCarthy and Crane, 2009] McCarthy, D. and Crane, C. (2009). *Comet and reverse ajax: The next-generation ajax 2.0.* Apress.

[Mohan et al., 1995] Mohan, C., Alonso, G., Günthör, R., and Kamath, M. (1995). Exotica: A research perspective on workflow management systems. *Data Engineering*, 51:18.

[Moran and Carroll, 1996] Moran, T. P. and Carroll, J. M. (1996). *Design rationale: Concepts, techniques, and use.* L. Erlbaum Associates Inc.

[Muller et al., 2008] Muller, A., Marquez, A. C., and Iung, B. (2008). On the concept of e-maintenance: Review and current research. *Reliability Engineering & System Safety*, 93(8):1165–1187.

[Niederman et al., 1996] Niederman, F., Beise, C. M., and Beranek, P. M. (1996). Issues and concerns about computer-supported meetings: the facilitator's perspective. *MIS Quarterly*, pages 1–22.

[Nielsen and Molich, 1990] Nielsen, J. and Molich, R. (1990). Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 249–256. ACM.

[Oberweis, 2005] Oberweis, A. (2005). Person-to-application processes: Workflow management. *Process-Aware Information Systems*, pages 21–36.

[Pauwels et al., 2009] Pauwels, K., Ambler, T., Clark, B. H., LaPointe, P., Reibstein, D., Skiera, B., Wierenga, B., and Wiesel, T. (2009). Dashboards as a service: why, what, how, and what research is needed? *Journal of Service Research*.

[Piirainen et al., 2012] Piirainen, K. A., Kolfschoten, G. L., and Lukosch, S. (2012). The joint struggle of complex engineering: A study of the challenges of collaborative design. *International Journal of Information Technology & Decision Making*, 11(06):1087–1125.

[Powell et al., 2004] Powell, A., Piccoli, G., and Ives, B. (2004). Virtual teams: A review of current literature and directions for future research. *SIGMIS Database*, 35(1):6–36.

[Redmiles et al., 2007] Redmiles, D., Van Der Hoek, A., Al-Ani, B., Hildenbrand, T., Quirk, S., Sarma, A., Filho, R., de Souza, C., and Trainer, E. (2007). Continuous coordination-a new paradigm to support globally distributed software development projects. *Wirtschafts Informatik*, 49(1):28.

[Rico et al., 2008] Rico, R., Sánchez-Manzanares, M., Gil, F., and Gibson, C. (2008). Team implicit coordination processes: A team knowledge–based approach. *Academy of Management Review*, 33(1):163–184.

[Risch et al., 2003] Risch, T., Josifovski, V., and Katchaounov, T. (2003). Functional data integration in a distributed mediator system. In *Functional Approach to Data Management-Modeling, Analyzing and Integrating Heterogeneous Data, Springer, ISBN*. Citeseer.

[Sangiorgi, 2014] Sangiorgi, U. B. (2014). Electronic sketching on a multiplatform context: A pilot study with developers. *International Journal of Human-Computer Studies*, 72(1):45 – 52.

[Sarma et al., 2010] Sarma, A., Redmiles, D., and van der Hoek, A. (2010). Categorizing the spectrum of coordination technology. *Computer*, 43(6):61–67.

[Schmidt and Simonee, 1996] Schmidt, K. and Simonee, C. (1996). Coordination mechanisms: Towards a conceptual foundation of cscw systems design. *Computer Supported Cooperative Work (CSCW)*, 5(2-3):155–200.

[Schümmer and Lukosch, 2013] Schümmer, T. and Lukosch, S. (2013). *Patterns for computer-mediated interaction.* John Wiley & Sons.

[Sekar et al., 2011] Sekar, R., Hsieh, S.-J., and Wu, Z. (2011). Remote diagnosis design for a plc-based automated system: 1-implementation of three levels of architectures. *The International Journal of Advanced Manufacturing Technology*, 57(5):683–700.

[Selznick, 1948] Selznick, P. (1948). Foundations of the theory of organization. *American sociological review*, pages 25–35.

[Steiner, 1972] Steiner, I. D. (1972). *Group process and productivity.* Academic Press New York.

[Stott and Walker, 1995] Stott, K. and Walker, A. (1995). *Teams teamwork and teambuilding : the manager's complete guide to teams in organisations.* Prentice Hall New York ; Sydney.

[Suchman, 1987] Suchman, L. A. (1987). *Plans and situated actions: the problem of human-machine communication.* Cambridge university press.

[Tanenbaum, 2006] Tanenbaum, A. S. (2006). *Structured computer organization.* Pearson.

[Terry et al., 1992] Terry, D., Goldberg, D., Nichols, D., and Oki, B. (1992). Continuous queries over append-only databases. *SIGMOD Rec.*, 21(2):321–330.

[Van de Ven et al., 1976] Van de Ven, A. H., Delbecq, A. L., and Koenig Jr, R. (1976). Determinants of coordination modes within organizations. *American sociological review*, pages 322–338.

[Van Grinsven and de Vreede, 2003] Van Grinsven, J. and de Vreede, G. (2003). Addressing productivity concerns in risk management through repeatable distributed collaboration processes. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, pages 10–pp. IEEE.

[VanGundy, 1988] VanGundy, A. B. (1988). *Techniques of structured problem solving.* Van Nostrand Reinhold New York.

[Veiel et al., 2013] Veiel, D., Haake, J. M., Lukosch, S., and Kolfschoten, G. (2013). On the acceptance of automatic facilitation in a context-adaptive group support system. In *System Sciences (HICSS), 2013 46th Hawaii International Conference on*, pages 509–518. IEEE.

[Ven et al., 1976] Ven, A. H. V. D., Delbecq, A. L., and Koenig, Richard, J. (1976). Determinants of coordination modes within organizations. *American Sociological Review*, 41(2):pp. 322–338.

[Wang et al., 2004] Wang, J., Tse, P. W., He, L., and Yeung, R. W. (2004). Remote sensing, diagnosis and collaborative maintenance with Web-enabled virtual instruments and mini-servers. *The International Journal of Advanced Manufacturing Technology*, 24(9-10):764–772.

[Whittaker et al., 1994] Whittaker, S., Frohlich, D., and Daly-Jones, O. (1994). Informal workplace communication: What is it like and how might we support it? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, pages 131–137, New York, NY, USA. ACM.

[Witte, 2007] Witte, E. H. (2007). Toward a group facilitation technique for project teams. *Group Processes & Intergroup Relations*, 10(3):299–309.

[Wulf, 1995] Wulf, V. (1995). Negotiability: a metafunction to tailor access to data in groupware. *Behaviour & Information Technology*, 14(3):143–151.

[Zeitler and Risch, 2011] Zeitler, E. and Risch, T. (2011). Massive scale-out of expensive continuous queries. *Proceedings of the VLDB Endowment*, 4(11).

[Zisman, 1978] Zisman, M. (1978). *Office Automation: Revolution Or Evolution?* Center for Informations Systems Research: Center for Information Systems Research. Center for Information Systems Research, Alfred P. Sloan School of Management, M.I.T.

List of Figures

List of Tables

Summary

The increasing development of information and communication technology is causing fundamental changes to today's society, changing the communication and interaction of people in their daily lives. For example, geographical distributed co-workers experience collaboration using a broad spectrum of shared workspace systems for communication and interaction despite their physical separation to share data, documents and contextual information.

The objective of this thesis is to design Elgar, a shared workspace system, that flexibly coordinates teamwork for such geographically distributed teams. In particular this thesis focuses on the design and evaluation of the system for remote industrial machine diagnosis by a team of distributed engineers.

This thesis identifies and describes a spectrum of coordination mechanisms, to structure and provide flexible coordination support. The two extremes of this spectrum are explored and implemented in Elgar. In the context of diagnosis, this thesis proposes *Rectio*, based on existing models of diagnosis to analyse observed anomalies, explore potential causes, and propose a diagnosis.

Elgar supports distributed teams in diagnosis tasks providing different coordination mechanisms and various functionalities that allow engineers to analyse, discuss and document diagnosis for machines. Elgar is evaluated in two different experiments one of Hägglunds Drives and another of Volvo Construction Equipment. Both experiments evaluate the usefulness of implemented diagnosis functionalities of the system and the prescribed and ad-hoc coordination mechanisms in a collaborative diagnosis task. The first experiment evaluates the use of Elgar with engineering students, whereas the second experiment evaluates Elgar with experienced diagnosis engineers.

In conclusion, this thesis shows that it is possible and necessary to design different coordination mechanisms and integrate them in a shared workspaces system, supporting flexible coordination for collaborative diagnosis tasks. In addition, the evaluation of the two experiments indicates that

even the use of dissimilar coordination mechanisms provides similar teamwork outcomes, making the selection of a specific coordination mechanism a shared decision of team preferences.

Samenvatting

De toenemende ontwikkeling van informatie- en communicatietechnologie veroorzaakt fundamentele veranderingen in de huidige maatschappij, waardoor de communicatie en interactie van mensen in hun dagelijks leven verandert. Geografisch verspreide medewerkers ervaren bijvoorbeeld samenwerking via een breed spectrum van shared-workspacesystemen voor communicatie en interactie, ondanks hun fysieke scheiding om gegevens, documenten en contextuele informatie te delen.

Het doel van dit proefschrift is om Elgar te ontwerpen, een shared-workspacesysteem, dat flexibel teamwerk coördineert voor dergelijke geografisch verspreide teams. Dit proefschrift richt zich in het bijzonder op het ontwerp en de evaluatie van het systeem voor de diagnose van industriële machines op afstand door een team van gedistribueerde ingenieurs.

Dit proefschrift identificeert en beschrijft een spectrum van coördinatiemechanismen om flexibele coördinatiemogelijkheden te structureren en te bieden. De twee uitersten van dit spectrum worden onderzocht en geïmplementeerd in Elgar. In de context van diagnostiek, stelt dit proefschrift Rectio voor, op basis van bestaande diagnosemodellen om geobserveerde afwijkingen te analyseren, mogelijke oorzaken te onderzoeken en een diagnose voor te stellen.

Elgar ondersteunt gedistribueerde teams bij diagnosetaken met verschillende coördinatiemechanismen en verschillende functionaliteiten waarmee ingenieurs de diagnose van machines kunnen analyseren, bespreken en documenteren. Elgar is in twee verschillende experimenten geëvalueerd, een van Hägglunds Drives en een andere van Volvo Construction Equipment. Beide experimenten evalueren de effectiviteit van geïmplementeerde diagnosefunctionaliteiten van het systeem en de voorgeschreven en ad-hoc coördinatiemechanismen in een gezamenlijke diagnosetaak. Het eerste experiment evalueert het gebruik van Elgar met technische studenten, terwijl het tweede experiment Elgar evalueert met ervaren diagnose-ingenieurs.

Concluderend laat dit proefschrift zien dat het mogelijk en noodzakelijk is om verschillende coördinatiemechanismen te ontwerpen en te integreren in een shared-workspacesysteem, ter ondersteuning van flexibele coördinatie voor gezamenlijke diagnosetaken. Bovendien geeft de evaluatie van de twee experimenten aan dat zelfs het gebruik van ongelijke coördinatiemechanismen vergelijkbare teamwerkresultaten oplevert, waardoor de selectie van een specifiek coördinatiemechanisme een gedeelde beslissing van teamvoorkeuren is.

Curriculum Vitae

Jordan Janeiro was born in Rio de Janeiro, Brazil on the 6th of May 1984. He completed his bachelor in Computer Science at PUC-Rio, Rio de Janeiro, Brazil in December 2005. His thesis focused on the management of household goods through the SNMP protocol in smart houses. Subsequently, his master studies in Computer Science at PUC-Rio in May 2008, investigating automatic adaptation of context-aware protocols for communication services in mobile devices.

In 2008, he moved to Dresden, Germany and worked in different research projects concerning the automatic generation of user interface for web services at TU Dresden, in cooperation with SAP Research. In 2011, he started his PhD at TU Delft, in Delft, the Netherlands at the TPM faculty. His research was based on the EU project Smart Vortex. During his research, he focused on the design of shared workspace systems for machine diagnosis that integrate support of coordination mechanisms for teamwork. He worked in collaboration with several universities, the Research Institute for Telecommunication and Cooperation, Group InMark, Volvo Construction Equipment, Hägglunds Drives, Sandvik Coromant and Dassault Systèmes.

Currently, he works as a senior consultant in digitalization projects focusing on the integration of enterprise systems through APIs in the telecommunications domain.

List of Publications

- Biancucci M., Janeiro J., Leva M., Lukosch S., Mecella M. (2014) Collaboration in Virtual Enterprises through the Smart Vortex Suite. In: Camarinha-Matos L.M., Afsarmanesh H. (eds) Collaborative Systems for Smart Networked Environments. PRO-VE 2014. IFIP Advances in Information and Communication Technology, vol 434. Springer, Berlin, Heidelberg.

- Janeiro J., Lukosch S., Brazier F.M.T., Leva M., Mecella M., Byström A. (2014) Evaluating Coordination Support Mechanisms in an Industrial Engineering Scenario. In: Baloian N., Burstein F., Ogata H., Santoro F., Zurita G. (eds) Collaboration and Technology. CRIWG 2014. Lecture Notes in Computer Science, vol 8658. Springer, Cham.

- Biancucci, M., Janeiro, J., Mecella, M., and Lukosch, S. (2014b). Supporting industrial processes by monitoring and visualizing collaborations. In Innovative Design and Manufacturing (ICIDM), Proceedings of the 2014 International Conference on, pages 330335. IEEE.

- Knoll S.W., Janeiro J., Lukosch S.G., Kolfschoten G.L. (2013) A Semantic Model for Adaptive Collaboration Support Systems. In: Hussein T., Paulheim H., Lukosch S., Ziegler J., Calvary G. (eds) Semantic Models for Adaptive Interactive Systems. HumanComputer Interaction Series. Springer, London.

- Janeiro, J., Lukosch, S., Radomski, S., Johanson, M., Mecella, M., and Larsson, J. (2013). Supporting elastic collaboration: Integration of collaboration components in dynamic contexts. In Proceedings of the 5th ACM SIGCHI symposium on Engineering interactive computing systems, pages 127132. ACM.

- Janeiro, J., Knoll, S. W., Lukosch, S., and Kolfschoten, G. (2012a). Adaptive collaboration support systems: Designing collaboration support for dynamic environments. In CESUN 2012: 3rd International Engineering Systems Symposium, Delft University of Technology, The Netherlands, 18-20 June 2012.

- Janeiro, J., Knoll, S. W., Lukosch, S., Kolfschoten, G. and Brazier, F., Designing Collaboration Support for Dynamic Environments, in: Proceedings of the International Conference on Group Decision and Negotiation (GDN), 2012.

- Janeiro, J., Lukosch, S., and Brazier, F. 2012. Elastic collaboration support: from workflow-based to emergent collaboration. In Proceedings of the 17th ACM international conference on Supporting group work (GROUP '12). ACM, New York, NY, USA, 317-320.

- Azevedo, D., Janeiro, J., Lukosch, S., Briggs R. O. and Fonseca, Benjamim (2011): An integrative approach to diagram-based collaborative brainstorming, Proceedings of the ECSCW 2011 Workshop on Collaborative usage and development of models and visualizations, scientific publication.

- Buttler T., Janeiro J., Lukosch S., Briggs R.O. (2011) Beyond GSS: Fitting Collaboration Technology to a Given Work Practice. In: Vivacqua A.S., Gutwin C., Borges M.R.S. (eds) Collaboration and Technology. CRIWG 2011. Lecture Notes in Computer Science, vol 6969. Springer, Berlin, Heidelberg.

SIKS Dissertations Series

2011 01 Botond Cseke (RUN), Variational Algorithms for Bayesian Inference in Latent Gaussian Models
02 Nick Tinnemeier (UU), Organizing Agent Organizations. Syntax and Operational Semantics of an Organization-Oriented Programming Language
03 Jan Martijn van der Werf (TUE), Compositional Design and Verification of Component-Based Information Systems
04 Hado van Hasselt (UU), Insights in Reinforcement Learning; Formal analysis and empirical evaluation of temporal-difference
05 Bas van der Raadt (VU), Enterprise Architecture Coming of Age - Increasing the Performance of an Emerging Discipline.
06 Yiwen Wang (TUE), Semantically-Enhanced Recommendations in Cultural Heritage
07 Yujia Cao (UT), Multimodal Information Presentation for High Load Human Computer Interaction
08 Nieske Vergunst (UU), BDI-based Generation of Robust Task-Oriented Dialogues
09 Tim de Jong (OU), Contextualised Mobile Media for Learning
10 Bart Bogaert (UvT), Cloud Content Contention
11 Dhaval Vyas (UT), Designing for Awareness: An Experience-focused HCI Perspective
12 Carmen Bratosin (TUE), Grid Architecture for Distributed Process Mining
13 Xiaoyu Mao (UvT), Airport under Control. Multiagent Scheduling for Airport Ground Handling
14 Milan Lovric (EUR), Behavioral Finance and Agent-Based Artificial Markets
15 Marijn Koolen (UvA), The Meaning of Structure: the Value of Link Evidence for Information Retrieval
16 Maarten Schadd (UM), Selective Search in Games of Different Complexity
17 Jiyin He (UVA), Exploring Topic Structure: Coherence, Diversity and Relatedness
18 Mark Ponsen (UM), Strategic Decision-Making in complex games
19 Ellen Rusman (OU), The Mind's Eye on Personal Profiles
20 Qing Gu (VU), Guiding service-oriented software engineering - A view-based approach
21 Linda Terlouw (TUD), Modularization and Specification of Service-Oriented Systems
22 Junte Zhang (UVA), System Evaluation of Archival Description and Access
23 Wouter Weerkamp (UVA), Finding People and their Utterances in Social Media
24 Herwin van Welbergen (UT), Behavior Generation for Interpersonal Coordination with Virtual Humans On Specifying, Scheduling and Realizing Multimodal Virtual Human Behavior
25 Syed Waqar ul Qounain Jaffry (VU), Analysis and Validation of Models for Trust Dynamics
26 Matthijs Aart Pontier (VU), Virtual Agents for Human Communication - Emotion Regulation and Involvement-Distance Trade-Offs in Embodied Conversational Agents and Robots
27 Aniel Bhulai (VU), Dynamic website optimization through autonomous management of design patterns
28 Rianne Kaptein (UVA), Effective Focused Retrieval by Exploiting Query Context and Document Structure
29 Faisal Kamiran (TUE), Discrimination-aware Classification
30 Egon van den Broek (UT), Affective Signal Processing (ASP): Unraveling the mystery of emotions
31 Ludo Waltman (EUR), Computational and Game-Theoretic Approaches for Modeling Bounded Rationality
32 Nees-Jan van Eck (EUR), Methodological Advances in Bibliometric Mapping of Science
33 Tom van der Weide (UU), Arguing to Motivate Decisions
34 Paolo Turrini (UU), Strategic Reasoning in Interdependence: Logical and Game-theoretical Investigations
35 Maaike Harbers (UU), Explaining Agent Behavior in Virtual Training
36 Erik van der Spek (UU), Experiments in serious game design: a cognitive approach
37 Adriana Burlutiu (RUN), Machine Learning for Pairwise Data, Applications for Preference Learning and Supervised Network Inference
38 Nyree Lemmens (UM), Bee-inspired Distributed Optimization
39 Joost Westra (UU), Organizing Adaptation using Agents in Serious Games
40 Viktor Clerc (VU), Architectural Knowledge Management in Global Software Development
41 Luan Ibraimi (UT), Cryptographically Enforced Distributed Data Access Control
42 Michal Sindlar (UU), Explaining Behavior through Mental State Attribution
43 Henk van der Schuur (UU), Process Improvement through Software Operation Knowledge
44 Boris Reuderink (UT), Robust Brain-Computer Interfaces
45 Herman Stehouwer (UvT), Statistical Language Models for Alternative Sequence Selection
46 Beibei Hu (TUD), Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work

47 Azizi Bin Ab Aziz (VU), Exploring Computational Models for Intelligent Support of Persons with Depression

48 Mark Ter Maat (UT), Response Selection and Turn-taking for a Sensitive Artificial Listening Agent

49 Andreea Niculescu (UT), Conversational interfaces for task-oriented spoken dialogues: design aspects influencing interaction quality

---

2012 01 Terry Kakeeto (UvT), Relationship Marketing for SMEs in Uganda

02 Muhammad Umair (VU), Adaptivity, emotion, and Rationality in Human and Ambient Agent Models

03 Adam Vanya (VU), Supporting Architecture Evolution by Mining Software Repositories

04 Jurriaan Souer (UU), Development of Content Management System-based Web Applications

05 Marijn Plomp (UU), Maturing Interorganisational Information Systems

06 Wolfgang Reinhardt (OU), Awareness Support for Knowledge Workers in Research Networks

07 Rianne van Lambalgen (VU), When the Going Gets Tough: Exploring Agent-based Models of Human Performance under Demanding Conditions

08 Gerben de Vries (UVA), Kernel Methods for Vessel Trajectories

09 Ricardo Neisse (UT), Trust and Privacy Management Support for Context-Aware Service Platforms

10 David Smits (TUE), Towards a Generic Distributed Adaptive Hypermedia Environment

11 J.C.B. Rantham Prabhakara (TUE), Process Mining in the Large: Preprocessing, Discovery, and Diagnostics

12 Kees van der Sluijs (TUE), Model Driven Design and Data Integration in Semantic Web Information Systems

13 Suleman Shahid (UvT), Fun and Face: Exploring non-verbal expressions of emotion during playful interactions

14 Evgeny Knutov (TUE), Generic Adaptation Framework for Unifying Adaptive Web-based Systems

15 Natalie van der Wal (VU), Social Agents. Agent-Based Modelling of Integrated Internal and Social Dynamics of Cognitive and Affective Processes.

16 Fiemke Both (VU), Helping people by understanding them - Ambient Agents supporting task execution and depression treatment

17 Amal Elgammal (UvT), Towards a Comprehensive Framework for Business Process Compliance

18 Eltjo Poort (VU), Improving Solution Architecting Practices

19 Helen Schonenberg (TUE), What's Next? Operational Support for Business Process Execution

20 Ali Bahramisharif (RUN), Covert Visual Spatial Attention, a Robust Paradigm for Brain-Computer Interfacing

21 Roberto Cornacchia (TUD), Querying Sparse Matrices for Information Retrieval

22 Thijs Vis (UvT), Intelligence, politie en veiligheidsdienst: verenigbare grootheden?

23 Christian Muehl (UT), Toward Affective Brain-Computer Interfaces: Exploring the Neurophysiology of Affect during Human Media Interaction

24 Laurens van der Werff (UT), Evaluation of Noisy Transcripts for Spoken Document Retrieval

25 Silja Eckartz (UT), Managing the Business Case Development in Inter-Organizational IT Projects: A Methodology and its Application

26 Emile de Maat (UVA), Making Sense of Legal Text

27 Hayrettin Gurkok (UT), Mind the Sheep! User Experience Evaluation & Brain-Computer Interface Games

28 Nancy Pascall (UvT), Engendering Technology Empowering Women

29 Almer Tigelaar (UT), Peer-to-Peer Information Retrieval

30 Alina Pommeranz (TUD), Designing Human-Centered Systems for Reflective Decision Making

31 Emily Bagarukayo (RUN), A Learning by Construction Approach for Higher Order Cognitive Skills Improvement, Building Capacity and Infrastructure

32 Wietske Visser (TUD), Qualitative multi-criteria preference representation and reasoning

33 Rory Sie (OUN), Coalitions in Cooperation Networks (COCOON)

34 Pavol Jancura (RUN), Evolutionary analysis in PPI networks and applications

35 Evert Haasdijk (VU), Never Too Old To Learn – On-line Evolution of Controllers in Swarm- and Modular Robotics

36 Denis Ssebugwawo (RUN), Analysis and Evaluation of Collaborative Modeling Processes

37 Agnes Nakakawa (RUN), A Collaboration Process for Enterprise Architecture Creation

38 Selmar Smit (VU), Parameter Tuning and Scientific Testing in Evolutionary Algorithms

39 Hassan Fatemi (UT), Risk-aware design of value and coordination networks

40 Agus Gunawan (UvT), Information Access for SMEs in Indonesia

41 Sebastian Kelle (OU), Game Design Patterns for Learning

42 Dominique Verpoorten (OU), Reflection Amplifiers in self-regulated Learning

43 Withdrawn

44 Anna Tordai (VU), On Combining Alignment Techniques

45 Benedikt Kratz (UvT), A Model and Language for Business-aware Transactions

46 Simon Carter (UVA), Exploration and Exploitation of Multilingual Data for Statistical Machine Translation

47 Manos Tsagkias (UVA), Mining Social Media: Tracking Content and Predicting Behavior

48 Jorn Bakker (TUE), Handling Abrupt Changes in Evolving Time-series Data

49 Michael Kaisers (UM), Learning against Learning - Evolutionary dynamics of reinforcement learning algorithms in strategic interactions

50 Steven van Kervel (TUD), Ontologogy driven Enterprise Information Systems Engineering

51 Jeroen de Jong (TUD), Heuristics in Dynamic Sceduling; a practical framework with a case study in elevator dispatching

---

2013 01 Viorel Milea (EUR), News Analytics for Financial Decision Support

02 Erietta Liarou (CWI), MonetDB/DataCell: Leveraging the Column-store Database Technology for Efficient and Scalable Stream Processing

03 Szymon Klarman (VU), Reasoning with Contexts in Description Logics

04 Chetan Yadati (TUD), Coordinating autonomous planning and scheduling

05 Dulce Pumareja (UT), Groupware Requirements Evolutions Patterns

06 Romulo Goncalves (CWI), The Data Cyclotron: Juggling Data and Queries for a Data Warehouse Audience

07 Giel van Lankveld (UvT), Quantifying Individual Player Differences
08 Robbert-Jan Merk (VU), Making enemies: cognitive modeling for opponent agents in fighter pilot simulators
09 Fabio Gori (RUN), Metagenomic Data Analysis: Computational Methods and Applications
10 Jeewanie Jayasinghe Arachchige (UvT), A Unified Modeling Framework for Service Design.
11 Evangelos Pournaras (TUD), Multi-level Reconfigurable Self-organization in Overlay Services
12 Marian Razavian (VU), Knowledge-driven Migration to Services
13 Mohammad Safiri (UT), Service Tailoring: User-centric creation of integrated IT-based homecare services to support independent living of elderly
14 Jafar Tanha (UVA), Ensemble Approaches to Semi-Supervised Learning Learning
15 Daniel Hennes (UM), Multiagent Learning - Dynamic Games and Applications
16 Eric Kok (UU), Exploring the practical benefits of argumentation in multi-agent deliberation
17 Koen Kok (VU), The PowerMatcher: Smart Coordination for the Smart Electricity Grid
18 Jeroen Janssens (UvT), Outlier Selection and One-Class Classification
19 Renze Steenhuizen (TUD), Coordinated Multi-Agent Planning and Scheduling
20 Katja Hofmann (UvA), Fast and Reliable Online Learning to Rank for Information Retrieval
21 Sander Wubben (UvT), Text-to-text generation by monolingual machine translation
22 Tom Claassen (RUN), Causal Discovery and Logic
23 Patricio de Alencar Silva (UvT), Value Activity Monitoring
24 Haitham Bou Ammar (UM), Automated Transfer in Reinforcement Learning
25 Agnieszka Anna Latoszek-Berendsen (UM), Intention-based Decision Support. A new way of representing and implementing clinical guidelines in a Decision Support System
26 Alireza Zarghami (UT), Architectural Support for Dynamic Homecare Service Provisioning
27 Mohammad Huq (UT), Inference-based Framework Managing Data Provenance
28 Frans van der Sluis (UT), When Complexity becomes Interesting: An Inquiry into the Information eXperience
29 Iwan de Kok (UT), Listening Heads
30 Joyce Nakatumba (TUE), Resource-Aware Business Process Management: Analysis and Support
31 Dinh Khoa Nguyen (UvT), Blueprint Model and Language for Engineering Cloud Applications
32 Kamakshi Rajagopal (OUN), Networking For Learning; The role of Networking in a Lifelong Learner's Professional Development
33 Qi Gao (TUD), User Modeling and Personalization in the Microblogging Sphere
34 Kien Tjin-Kam-Jet (UT), Distributed Deep Web Search
35 Abdallah El Ali (UvA), Minimal Mobile Human Computer Interaction
36 Than Lam Hoang (TUe), Pattern Mining in Data Streams
37 Dirk Börner (OUN), Ambient Learning Displays
38 Eelco den Heijer (VU), Autonomous Evolutionary Art
39 Joop de Jong (TUD), A Method for Enterprise Ontology based Design of Enterprise Information Systems
40 Pim Nijssen (UM), Monte-Carlo Tree Search for Multi-Player Games
41 Jochem Liem (UVA), Supporting the Conceptual Modelling of Dynamic Systems: A Knowledge Engineering Perspective on Qualitative Reasoning
42 Léon Planken (TUD), Algorithms for Simple Temporal Reasoning
43 Marc Bron (UVA), Exploration and Contextualization through Interaction and Concepts

2014 01 Nicola Barile (UU), Studies in Learning Monotone Models from Data
02 Fiona Tuliyano (RUN), Combining System Dynamics with a Domain Modeling Method
03 Sergio Raul Duarte Torres (UT), Information Retrieval for Children: Search Behavior and Solutions
04 Hanna Jochmann-Mannak (UT), Websites for children: search strategies and interface design - Three studies on children's search performance and evaluation
05 Jurriaan van Reijsen (UU), Knowledge Perspectives on Advancing Dynamic Capability
06 Damian Tamburri (VU), Supporting Networked Software Development
07 Arya Adriansyah (TUE), Aligning Observed and Modeled Behavior
08 Samur Araujo (TUD), Data Integration over Distributed and Heterogeneous Data Endpoints
09 Philip Jackson (UvT), Toward Human-Level Artificial Intelligence: Representation and Computation of Meaning in Natural Language
10 Ivan Salvador Razo Zapata (VU), Service Value Networks
11 Janneke van der Zwaan (TUD), An Empathic Virtual Buddy for Social Support
12 Willem van Willigen (VU), Look Ma, No Hands: Aspects of Autonomous Vehicle Control
13 Arlette van Wissen (VU), Agent-Based Support for Behavior Change: Models and Applications in Health and Safety Domains
14 Yangyang Shi (TUD), Language Models With Meta-information
15 Natalya Mogles (VU), Agent-Based Analysis and Support of Human Functioning in Complex Socio-Technical Systems: Applications in Safety and Healthcare
16 Krystyna Milian (VU), Supporting trial recruitment and design by automatically interpreting eligibility criteria
17 Kathrin Dentler (VU), Computing healthcare quality indicators automatically: Secondary Use of Patient Data and Semantic Interoperability
18 Mattijs Ghijsen (UVA), Methods and Models for the Design and Study of Dynamic Agent Organizations
19 Vinicius Ramos (TUE), Adaptive Hypermedia Courses: Qualitative and Quantitative Evaluation and Tool Support
20 Mena Habib (UT), Named Entity Extraction and Disambiguation for Informal Text: The Missing Link
21 Kassidy Clark (TUD), Negotiation and Monitoring in Open Environments
22 Marieke Peeters (UU), Personalized Educational Games - Developing agent-supported scenario-based training
23 Eleftherios Sidirourgos (UvA/CWI), Space Efficient Indexes for the Big Data Era
24 Davide Ceolin (VU), Trusting Semi-structured Web Data

25 Martijn Lappenschaar (RUN), New network models for the analysis of disease interaction
26 Tim Baarslag (TUD), What to Bid and When to Stop
27 Rui Jorge Almeida (EUR), Conditional Density Models Integrating Fuzzy and Probabilistic Representations of Uncertainty
28 Anna Chmielowiec (VU), Decentralized k-Clique Matching
29 Jaap Kabbedijk (UU), Variability in Multi-Tenant Enterprise Software
30 Peter de Cock (UvT), Anticipating Criminal Behaviour
31 Leo van Moergestel (UU), Agent Technology in Agile Multiparallel Manufacturing and Product Support
32 Naser Ayat (UvA), On Entity Resolution in Probabilistic Data
33 Tesfa Tegegne (RUN), Service Discovery in eHealth
34 Christina Manteli (VU), The Effect of Governance in Global Software Development: Analyzing Transactive Memory Systems.
35 Joost van Ooijen (UU), Cognitive Agents in Virtual Worlds: A Middleware Design Approach
36 Joos Buijs (TUE), Flexible Evolutionary Algorithms for Mining Structured Process Models
37 Maral Dadvar (UT), Experts and Machines United Against Cyberbullying
38 Danny Plass-Oude Bos (UT), Making brain-computer interfaces better: improving usability through post-processing.
39 Jasmina Maric (UvT), Web Communities, Immigration, and Social Capital
40 Walter Omona (RUN), A Framework for Knowledge Management Using ICT in Higher Education
41 Frederic Hogenboom (EUR), Automated Detection of Financial Events in News Text
42 Carsten Eijckhof (CWI/TUD), Contextual Multidimensional Relevance Models
43 Kevin Vlaanderen (UU), Supporting Process Improvement using Method Increments
44 Paulien Meesters (UvT), Intelligent Blauw. Met als ondertitel: Intelligence-gestuurde politiezorg in gebiedsgebonden eenheden.
45 Birgit Schmitz (OUN), Mobile Games for Learning: A Pattern-Based Approach
46 Ke Tao (TUD), Social Web Data Analytics: Relevance, Redundancy, Diversity
47 Shangsong Liang (UVA), Fusion and Diversification in Information Retrieval

2015 01 Niels Netten (UvA), Machine Learning for Relevance of Information in Crisis Response
02 Faiza Bukhsh (UvT), Smart auditing: Innovative Compliance Checking in Customs Controls
03 Twan van Laarhoven (RUN), Machine learning for network data
04 Howard Spoelstra (OUN), Collaborations in Open Learning Environments
05 Christoph Bösch (UT), Cryptographically Enforced Search Pattern Hiding
06 Farideh Heidari (TUD), Business Process Quality Computation - Computing Non-Functional Requirements to Improve Business Processes
07 Maria-Hendrike Peetz (UvA), Time-Aware Online Reputation Analysis
08 Jie Jiang (TUD), Organizational Compliance: An agent-based model for designing and evaluating organizational interactions
09 Randy Klaassen (UT), HCI Perspectives on Behavior Change Support Systems
10 Henry Hermans (OUN), OpenU: design of an integrated system to support lifelong learning
11 Yongming Luo (TUE), Designing algorithms for big graph datasets: A study of computing bisimulation and joins
12 Julie M. Birkholz (VU), Modi Operandi of Social Network Dynamics: The Effect of Context on Scientific Collaboration Networks
13 Giuseppe Procaccianti (VU), Energy-Efficient Software
14 Bart van Straalen (UT), A cognitive approach to modeling bad news conversations
15 Klaas Andries de Graaf (VU), Ontology-based Software Architecture Documentation
16 Changyun Wei (UT), Cognitive Coordination for Cooperative Multi-Robot Teamwork
17 André van Cleeff (UT), Physical and Digital Security Mechanisms: Properties, Combinations and Trade-offs
18 Holger Pirk (CWI), Waste Not, Want Not! - Managing Relational Data in Asymmetric Memories
19 Bernardo Tabuenca (OUN), Ubiquitous Technology for Lifelong Learners
20 Lois Vanhée (UU), Using Culture and Values to Support Flexible Coordination
21 Sibren Fetter (OUN), Using Peer-Support to Expand and Stabilize Online Learning
22 Zhemin Zhu (UT), Co-occurrence Rate Networks
23 Luit Gazendam (VU), Cataloguer Support in Cultural Heritage
24 Richard Berendsen (UVA), Finding People, Papers, and Posts: Vertical Search Algorithms and Evaluation
25 Steven Woudenberg (UU), Bayesian Tools for Early Disease Detection
26 Alexander Hogenboom (EUR), Sentiment Analysis of Text Guided by Semantics and Structure
27 Sándor Héman (CWI), Updating compressed colomn stores
28 Janet Bagorogoza (TiU), Knowledge Management and High Performance; The Uganda Financial Institutions Model for HPO
29 Hendrik Baier (UM), Monte-Carlo Tree Search Enhancements for One-Player and Two-Player Domains
30 Kiavash Bahreini (OU), Real-time Multimodal Emotion Recognition in E-Learning
31 Yakup Koç (TUD), On the robustness of Power Grids
32 Jerome Gard (UL), Corporate Venture Management in SMEs
33 Frederik Schadd (TUD), Ontology Mapping with Auxiliary Resources
34 Victor de Graaf (UT), Gesocial Recommender Systems
35 Jungxao Xu (TUD), Affective Body Language of Humanoid Robots: Perception and Effects in Human Robot Interaction

2016 01 Syed Saiden Abbas (RUN), Recognition of Shapes by Humans and Machines
02 Michiel Christiaan Meulendijk (UU), Optimizing medication reviews through decision support: prescribing a better pill to swallow
03 Maya Sappelli (RUN), Knowledge Work in Context: User Centered Knowledge Worker Support
04 Laurens Rietveld (VU), Publishing and Consuming Linked Data

05 Evgeny Sherkhonov (UVA), Expanded Acyclic Queries: Containment and an Application in Explaining Missing Answers

06 Michel Wilson (TUD), Robust scheduling in an uncertain environment

07 Jeroen de Man (VU), Measuring and modeling negative emotions for virtual training

08 Matje van de Camp (TiU), A Link to the Past: Constructing Historical Social Networks from Unstructured Data

09 Archana Nottamkandath (VU), Trusting Crowdsourced Information on Cultural Artefacts

10 George Karafotias (VUA), Parameter Control for Evolutionary Algorithms

11 Anne Schuth (UVA), Search Engines that Learn from Their Users

12 Max Knobbout (UU), Logics for Modelling and Verifying Normative Multi-Agent Systems

13 Nana Baah Gyan (VU), The Web, Speech Technologies and Rural Development in West Africa - An ICT4D Approach

14 Ravi Khadka (UU), Revisiting Legacy Software System Modernization

15 Steffen Michels (RUN), Hybrid Probabilistic Logics - Theoretical Aspects, Algorithms and Experiments

16 Guangliang Li (UVA), Socially Intelligent Autonomous Agents that Learn from Human Reward

17 Berend Weel (VU), Towards Embodied Evolution of Robot Organisms

18 Albert Meroño Peñuela (VU), Refining Statistical Data on the Web

19 Julia Efremova (Tu/e), Mining Social Structures from Genealogical Data

20 Daan Odijk (UVA), Context & Semantics in News & Web Search

21 Alejandro Moreno Célleri (UT), From Traditional to Interactive Playspaces: Automatic Analysis of Player Behavior in the Interactive Tag Playground

22 Grace Lewis (VU), Software Architecture Strategies for Cyber-Foraging Systems

23 Fei Cai (UVA), Query Auto Completion in Information Retrieval

24 Brend Wanders (UT), Repurposing and Probabilistic Integration of Data; An Iterative and data model independent approach

25 Julia Kiseleva (TU/e), Using Contextual Information to Understand Searching and Browsing Behavior

26 Dilhan Thilakarathne (VU), In or Out of Control: Exploring Computational Models to Study the Role of Human Awareness and Control in Behavioural Choices, with Applications in Aviation and Energy Management Domains

27 Wen Li (TUD), Understanding Geo-spatial Information on Social Media

28 Mingxin Zhang (TUD), Large-scale Agent-based Social Simulation - A study on epidemic prediction and control

29 Nicolas Höning (TUD), Peak reduction in decentralised electricity systems - Markets and prices for flexible planning

30 Ruud Mattheij (UvT), The Eyes Have It

31 Mohammad Khelghati (UT), Deep web content monitoring

32 Eelco Vriezekolk (UT), Assessing Telecommunication Service Availability Risks for Crisis Organisations

33 Peter Bloem (UVA), Single Sample Statistics, exercises in learning from just one example

34 Dennis Schunselaar (TUE), Configurable Process Trees: Elicitation, Analysis, and Enactment

35 Zhaochun Ren (UVA), Monitoring Social Media: Summarization, Classification and Recommendation

36 Daphne Karreman (UT), Beyond R2D2: The design of nonverbal interaction behavior optimized for robot-specific morphologies

37 Giovanni Sileno (UvA), Aligning Law and Action - a conceptual and computational inquiry

38 Andrea Minuto (UT), Materials that Matter - Smart Materials meet Art & Interaction Design

39 Merijn Bruijnes (UT), Believable Suspect Agents; Response and Interpersonal Style Selection for an Artificial Suspect

40 Christian Detweiler (TUD), Accounting for Values in Design

41 Thomas King (TUD), Governing Governance: A Formal Framework for Analysing Institutional Design and Enactment Governance

42 Spyros Martzoukos (UVA), Combinatorial and Compositional Aspects of Bilingual Aligned Corpora

43 Saskia Koldijk (RUN), Context-Aware Support for Stress Self-Management: From Theory to Practice

44 Thibault Sellam (UVA), Automatic Assistants for Database Exploration

45 Bram van de Laar (UT), Experiencing Brain-Computer Interface Control

46 Jorge Gallego Perez (UT), Robots to Make you Happy

47 Christina Weber (UL), Real-time foresight - Preparedness for dynamic innovation networks

48 Tanja Buttler (TUD), Collecting Lessons Learned

49 Gleb Polevoy (TUD), Participation and Interaction in Projects. A Game-Theoretic Analysis

50 Yan Wang (UVT), The Bridge of Dreams: Towards a Method for Operational Performance Alignment in IT-enabled Service Supply Chains

---

2017 01 Jan-Jaap Oerlemans (UL), Investigating Cybercrime

02 Sjoerd Timmer (UU), Designing and Understanding Forensic Bayesian Networks using Argumentation

03 Daniël Harold Telgen (UU), Grid Manufacturing; A Cyber-Physical Approach with Autonomous Products and Reconfigurable Manufacturing Machines

04 Mrunal Gawade (CWI), Multi-core Parallelism in a Column-store

05 Mahdieh Shadi (UVA), Collaboration Behavior

06 Damir Vandic (EUR), Intelligent Information Systems for Web Product Search

07 Roel Bertens (UU), Insight in Information: from Abstract to Anomaly

08 Rob Konijn (VU) , Detecting Interesting Differences:Data Mining in Health Insurance Data using Outlier Detection and Subgroup Discovery

09 Dong Nguyen (UT), Text as Social and Cultural Data: A Computational Perspective on Variation in Text

10 Robby van Delden (UT), (Steering) Interactive Play Behavior

11 Florian Kunneman (RUN), Modelling patterns of time and emotion in Twitter #anticipointment

12 Sander Leemans (TUE), Robust Process Mining with Guarantees

13 Gijs Huisman (UT), Social Touch Technology - Extending the reach of social touch through haptic technology

14 Shoshannah Tekofsky (UvT), You Are Who You Play You Are: Modelling Player Traits from Video Game Behavior

15 Peter Berck (RUN), Memory-Based Text Correction

16 Aleksandr Chuklin (UVA), Understanding and Modeling Users of Modern Search Engines

17 Daniel Dimov (UL), Crowdsourced Online Dispute Resolution

18 Ridho Reinanda (UVA), Entity Associations for Search

19 Jeroen Vuurens (UT), Proximity of Terms, Texts and Semantic Vectors in Information Retrieval

20 Mohammadbashir Sedighi (TUD), Fostering Engagement in Knowledge Sharing: The Role of Perceived Benefits, Costs and Visibility

21 Jeroen Linssen (UT), Meta Matters in Interactive Storytelling and Serious Gaming (A Play on Worlds)

22 Sara Magliacane (VU), Logics for causal inference under uncertainty

23 David Graus (UVA), Entities of Interest — Discovery in Digital Traces

24 Chang Wang (TUD), Use of Affordances for Efficient Robot Learning

25 Veruska Zamborlini (VU), Knowledge Representation for Clinical Guidelines, with applications to Multimorbidity Analysis and Literature Search

26 Merel Jung (UT), Socially intelligent robots that understand and respond to human touch

27 Michiel Joosse (UT), Investigating Positioning and Gaze Behaviors of Social Robots: People's Preferences, Perceptions and Behaviors

28 John Klein (VU), Architecture Practices for Complex Contexts

29 Adel Alhuraibi (UvT), From IT-BusinessStrategic Alignment to Performance: A Moderated Mediation Model of Social Innovation, and Enterprise Governance of IT”

30 Wilma Latuny (UvT), The Power of Facial Expressions

31 Ben Ruijl (UL), Advances in computational methods for QFT calculations

32 Thaer Samar (RUN), Access to and Retrievability of Content in Web Archives

33 Brigit van Loggem (OU), Towards a Design Rationale for Software Documentation: A Model of Computer-Mediated Activity

34 Maren Scheffel (OU), The Evaluation Framework for Learning Analytics

35 Martine de Vos (VU), Interpreting natural science spreadsheets

36 Yuanhao Guo (UL), Shape Analysis for Phenotype Characterisation from High-throughput Imaging

37 Alejandro Montes Garcia (TUE), WiBAF: A Within Browser Adaptation Framework that Enables Control over Privacy

38 Alex Kayal (TUD), Normative Social Applications

39 Sara Ahmadi (RUN), Exploiting properties of the human auditory system and compressive sensing methods to increase noise robustness in ASR

40 Altaf Hussain Abro (VUA), Steer your Mind: Computational Exploration of Human Control in Relation to Emotions, Desires and Social Support For applications in human-aware support systems

41 Adnan Manzoor (VUA), Minding a Healthy Lifestyle: An Exploration of Mental Processes and a Smart Environment to Provide Support for a Healthy Lifestyle

42 Elena Sokolova (RUN), Causal discovery from mixed and missing data with applications on ADHD datasets

43 Maaike de Boer (RUN), Semantic Mapping in Video Retrieval

44 Garm Lucassen (UU), Understanding User Stories - Computational Linguistics in Agile Requirements Engineering

45 Bas Testerink (UU), Decentralized Runtime Norm Enforcement

46 Jan Schneider (OU), Sensor-based Learning Support

47 Jie Yang (TUD), Crowd Knowledge Creation Acceleration

48 Angel Suarez (OU), Collaborative inquiry-based learning

2018 01 Han van der Aa (VUA), Comparing and Aligning Process Representations

02 Felix Mannhardt (TUE), Multi-perspective Process Mining

03 Steven Bosems (UT), Causal Models For Well-Being: Knowledge Modeling, Model-Driven Development of Context-Aware Applications, and Behavior Prediction