Co-creation with machine learning: Towards a dynamic understanding of knowledge boundaries between developers and end-users

van Krimpen, F.J.; van der Voort, H.G.

DOI
10.1016/j.infoandorg.2025.100574

Publication date
2025

Document Version
Final published version

Published in
Information and Organization

Information and Organization 35 (2025) 100574

Contents lists available at ScienceDirect

Information and Organization

journal homepage: www.elsevier.com/locate/infoandorg

Co-creation with machine learning: Towards a dynamic understanding of knowledge boundaries between developers and end-users

Floris van Krimpen [a,b,*], Haiko van der Voort [a]

[a] Faculty of Technology, Policy and Management, TU Delft, Jaffalaan 5, 2628, BX, Delft, the Netherlands
[b] Department of Management, Economics and Industrial Engineering, Politecnico di Milano, Italy

ARTICLE INFO

Keywords:
Machine learning
Coordination
Knowledge boundaries
Occupational communities
Co-creation
Knowledge sharing
Waterbed effects

ABSTRACT

The impact of machine learning within public organizations relies on coordinated effort over the functional chain from data generation to decision-making. This coordination faces challenges due to the separation between data intelligence departments and operational intelligence. Through theory about knowledge sharing between occupational communities and a case study at a Dutch inspectorate, we explore knowledge boundaries between machine learning developers and end-users and the effects of co-creation. Our analysis reveals that knowledge boundaries are dynamic, with boundaries blurring, persisting, and emerging under the influence of co-creation. Especially the emergence of boundaries is surprising and suggests the presence of a waterbed effect. Furthermore, knowledge boundaries are layered phenomena, with some boundary types more prone to change than others. Understanding knowledge boundaries and their dynamics better can be crucial for improving the intended impact of ML for organizations.

1. Introduction

Machine learning algorithms are increasingly deployed in public sector organizations, including to support critical decision-making processes. For instance, ML algorithms are now being used to inform complex decisions such as evaluating early prison eligibility (Berk, 2017) or the allocation of enforcement resources within regulatory agencies (Yeung, 2018). The use of algorithms to facilitate regulatory tasks is called 'algorithmic regulation.' Algorithms are increasingly used to coordinate the behaviour of inspectees or manage risks within a particular area of inspectees (Ulbricht & Yeung, 2022). These public organizations introduce ML algorithms to enhance decision-making accuracy (Domingos, 2012; Eggers, Schatsky, & Viechnicki, 2017) or streamline data analysis and interpretation (Alexopoulos et al., 2019).

However, ML development presents many challenges, including transparency (Burrell, 2016), accountability (Veale, Van Kleek, & Binns, 2018), and discrimination (Barocas & Selbst, 2016). Recent scholarly attention has increasingly highlighted organizations as the context in which both promises and challenges of ML materialize (Bailey, Faraj, Hinds, von Krogh, & Leonardi, 2019; Faraj, Pachidi, & Sayegh, 2018; von Richthofen, Ogolla, and Send, 2022; Waardenburg & Huysman, 2022). An emerging challenge is specialization. Specialization is an issue when different specialized actors struggle to collaborate effectively in ML development due to divergent types of knowledge. Machine learning development requires specific knowledge and skills, typically beyond the expertise of end-users or

* Corresponding author at: Faculty of Technology, Policy and Management, TU Delft, Jaffalaan 5, 2628, BX, Delft, the Netherlands.
E-mail addresses: f.j.vankrimpen@tudelft.nl (F. van Krimpen), h.g.vandervoort@tudelft.nl (H. van der Voort).

https://doi.org/10.1016/j.infoandorg.2025.100574
Received 3 July 2023; Received in revised form 19 December 2024; Accepted 5 April 2025
Available online 28 April 2025
1471-7727/

decision-makers (von Richthofen, Ogolla, and Send, 2022; Waardenburg, Sergeeva, & Huysman, 2018). Consequently, organizations frequently develop ML-based innovative techniques through dedicated sub-units (van der Voort, Klievink, Arnaboldi, & Meijer, 2019). Lorenz, Van Erp, and Meijer (2022) argue that this separation may affect the effective development and use of ML algorithms.

Similarly, Waardenburg and Huysman (2022) theorize that blurring boundaries between end-users and developers seems needed to create helpful self-learning ML systems. However, these scholars also acknowledge that in practice boundaries often persist, even when interventions such as co-creating ML are taken to blur these boundaries (Waardenburg, Huysman, & Sergeeva, 2022). A key concept is "co-creation". It refers to constant knowledge sharing between developers and end-users to develop ML systems (Lebovitz, Levina, & Lifshitz-Assa, 2021; Lorenz et al., 2022; Waardenburg & Huysman, 2022). Despite a co-creation intervention, boundaries often persist. This persistence necessitates a nuanced and deep-rooted understanding of these boundaries. Particularly, how the boundaries respond to interventions.

Investigating boundaries and their behaviour is important for multiple reasons. First, boundaries may significantly impact ML tool performance in public organizations. Second, boundaries may also lead to important ethical trade-offs falling between the cracks of different occupational communities, like developers and end-users. Lastly, understanding the effects of co-creation on boundaries in ML development is important because not knowing the effects could lead to unintended effects occurring. Ulbricht and Yeung (2022) argue that studying algorithms and their development in their respective contexts is essential for comprehensive insight into their effects.

This paper examines the dynamics of boundaries between machine learning developers and end-users. It focuses on a case where boundaries were intentionally removed to facilitate interaction. Within this case, we explored how boundaries both blur, persist and emerge. This informs us about the boundaries and their behaviour and cooperation between developers and end-users. Our main question is: *How does co-creation influence boundaries between developers and end-users in ML development and how do boundaries behave as a result?*

In our theoretical section, we will use knowledge management theory to explore the boundaries, most notably theories about knowledge boundaries (Carlile, 2004) and knowledge sharing between occupational communities (Bechky, 2003). We further explore the dynamics of boundaries following mostly a prior study by Waardenburg and Huysman (2022). In Section 3 we introduce a case study at the Dutch Human Environment and Transport Inspectorate (ILT). This inspection agency introduced a program management approach to intentionally blur the boundaries between developers and end-users of ML. The findings are presented in Section 4. After that, Section 5 focuses on the discussion of the implications of these findings and links them to the theory. Furthermore, it discusses the limitations of the study and potential future research. Finally, Section 6 concludes.

The paper's theoretical contributions are twofold. Firstly, the paper enriches the organizational literature that discusses knowledge sharing and boundaries by adding new and specific insights about knowledge boundaries in the context of the development and use of ML. Secondly, the paper adds to the organizational literature by exploring the mechanisms through which new ways of co-creation influences the behaviour of boundaries and how they blur, persist, or emerge in the development of public ML algorithms.

2. Knowledge boundaries between machine learning developers and end-users

2.1. Blurring boundaries through the co-creation of ML systems

The increasing presence of ML systems in organizational settings has spurred the attention of organizational scholars who study the consequences for work and organizing (Bailey et al., 2019; van den Broek, Sergeeva, & Huysman Vrije, 2021). More specifically, the study of knowledge boundaries related to ML development has received increasing interest (e.g. Faraj et al., 2018; Waardenburg & Huysman, 2022).

Some studies have focused their attention on how to deal with boundaries. To address boundaries requires collaboration. Mayer, van den Broek, Karacic, Hidalgo, and Huysman (2023) propose managers guidelines to promote collaborative AI development. They include to create a shared vision, to build a collective understanding, and to develop complementary abilities. Programme management (Ferns, 1991; Pellegrinelli, 1997) offers a similar intervention, facilitating cross-departmental collaboration through simultaneous multi-project coordination.

Other scholars studied boundaries and strategies to overcome these even in more detail. Waardenburg et al. (2022) argue that knowledge sharing in ML development occurs through sharing practices. Participating in shared practices would help actors develop a shared understanding (Brown & Duguid, 1991; Lave & Wenger, 1991; Orr, 1996). Hence, Waardenburg and Huysman (2022) advocate for a co-creation perspective in the case of the development, implementation, and use of self-learning ML algorithms. They argue for the sharing of practices or the emergence of new communities of practice. Their work highlights that '*new fields of practice can emerge that allow for knowledge to be shared between communities*' (Levina & Vaast, 2005).

Co-creation suggests that overcoming boundaries requires to move from a perspective of co-existence of stakeholders in the development of ML systems towards a view that sees stakeholders as co-creating the systems (Holmström & Hällgren, 2022; van den Broek et al., 2021; Waardenburg & Huysman, 2022). Co-existence implies hardly any interaction between stakeholders, while co-creation implies the opposite. Such a practice-based perspective on knowledge sharing (Orlikowski, 2002) emphasizes that '*developers and users constantly share technical and work-related knowledge that requires them to constantly cross their boundaries*' (Waardenburg & Huysman, 2022, p.2).

However, empirical evidence suggests that crossing boundaries in ML development remains challenging, because boundaries persist (Waardenburg & Huysman, 2022). Waardenburg et al. (2022) explored interventions, such as deploying intelligence officers as brokers between developers and end-users. The broker role they studied was meant to resolve a semantic boundary between the communities and can be seen as an intervention to blur boundaries. Paradoxically, this intervention to blur boundaries instead had unintended consequences. The algorithmic broker role had the effect of the emergence of novel boundaries.

The example illustrates a critical insight: organizational interventions aimed at blurring boundaries can unexpectedly reinforce or create novel boundaries elsewhere. The example illustrates that despite deliberate efforts boundaries can persist. This raises the question of an in-depth understanding of what these persisting boundaries are. To do this, there is a need to understand more about the literature on knowledge boundaries.

2.2. Knowledge boundaries in ML development

Organization science and knowledge management literature has explored knowledge boundaries and ways to overcome them (Bechky, 2003; Carlile, 2002, 2004; Orlikowski, 2002). Carlile (2004) discusses an integrative framework for managing knowledge across boundaries, delineating three distinct boundary types.

The first is the syntactic boundary, informed by an information-processing perspective on boundaries (Galbraith, 1973; Lawrence & Lorsch, 1967). To overcome a syntactic boundary knowledge can simply be transferred across it. According to Carlile (2004), a second boundary type is semantic. Semantic boundaries arise from the '*differences in meanings, assumptions, and contexts*' (Kellogg, Orlikowski, & Yates, 2006) that create ambiguity between organizational actors. Lastly, Carlile (2004) discusses the pragmatic boundary. The pragmatic boundary reveals the political aspects of knowledge and shows that knowledge is at stake (Kellogg et al., 2006). This perspective acknowledges that organizational stakeholders are invested in their knowledge and tie their measure of worth to it (Carlile, 2004).

Carlile's (2004) typology acknowledges the diversity of knowledge boundaries and delivers a conceptual framework for scholars to respect that diversity. Carlile (2004) also seems to imply that knowledge is manageable and transferable across boundaries, which makes knowledge resemble a tangible resource. Though the typology of boundaries clarifies their diversity, an emphasis on the boundaries may move the attention away from organizational stakeholders being bounded.

Bechky (2003) shifts the focus from the boundaries to the occupational communities these boundaries delineate. An occupational community refers to a community of organizational stakeholders that are part of the same occupation (Van Maanen & Barley, 1984). According to Bechky (2003), knowledge does not exist as a static entity within such a community. Instead, knowledge develops in relation to the activities that people engage in. Knowledge is constructed and situated in the organizational communities in which it is present (Van Maanen & Barley, 1984; Weick, 1979). It is argued that knowledge is embedded in practices (Brown & Duguid, 1991). This means that knowledge cannot be seen as a tangible resource that can be transferred from one community to the other. Instead, knowledge is embedded in daily procedures and routines that occupational communities engage in (Nonaka & von Krogh, 2009).

Given this view, Bechky (2003) argues that overcoming boundaries requires interaction, direct communication, and physical proximity. She highlights that knowledge sharing depends on social interactions, shared experiences and common contexts between professional communities. Bechky (2003) view reveals that knowledge sharing is dynamic. This perspective aligns with Orlikowski's (2002) conception of 'knowing' as an ongoing social accomplishment. As a result, knowledge is shaped by everyday practices and as actors engage the world in practice.

While Bechky (2003) and Orlikowski (2002) highlight the dynamic nature of knowledge and knowledge sharing, existing literature has difficulty to translate these findings to understanding knowledge boundaries. Despite literature on communities emphasizing dynamics, the accounts on boundaries seem more static. Yet from the starting point that knowledge boundaries are diverse, it could easily be theorized that different types of boundaries may also exist at the same time, and that these boundaries may be subject to change. It is our assumption that an account on the dynamics of knowledge boundaries would add to our understanding of the adoption of machine learning in organizations.

This study aims to develop a dynamic account on knowledge boundaries in the context of machine learning, focusing on a conscious attempt to share knowledge across boundaries while developing ML (i.e. co-creation through program management). This study pursues two contributions. First, we want to highlight the complexity of co-creating ML and study the dynamics of the boundaries during this attempt. We answer to the call to do more empirical research in the development and use of ML and the boundaries that might be present (Lorenz et al., 2022; Waardenburg & Huysman, 2022). Second, we aim to contribute to boundaries literature (Bechky, 2003; Carlile, 2004; Orlikowski, 2002) and highlight the connectedness and co-existence of boundaries. In the study we will focus on both the boundaries and practices by occupational communities.

3. Methods

This study employs an exploratory, inductive, single case study to develop existing theory (Siggelkow, 2007). This method enables to further expand on previous theories about overcoming knowledge boundaries and to bring a rich empirical picture to the research of ML technologies in public organizations.

3.1. Case description

The *Inspectie voor Leefomgeving en Transport* (The Human Environment and Transport Inspectorate; ILT from now on) implemented the "Less Greenhouse Gases" programme from begin 2018 to the end of 2022, aimed at reducing emissions of greenhouse gases (F-gases) and ozone-depleting substances (OAS). The ILT started the programme to target the resources of the ILT towards topics where societal risks are the biggest and the greatest social damage can occur.

Programmes are '*a temporary way of working together, aimed at pursuing certain goals, which contribute to achieving the strategy of the organisation(s)*' (Prevaas, 2018). Typically, programmes integrate multiple interdependent projects to realize benefits unattainable through independent project management (Ferns, 1991). Within the program, employees work together on projects that are related to the overarching program, which is headed by a programme manager who oversees the goals and direction of the different sub-projects. The program manager makes sure that the sub-projects are in accordance with the overarching program goal and that programmes have enough resources. A key characteristic of programmes is they circumvent the normal hierarchy existent in organizations and instead facilitate cross-divisional work (Pellegrinelli, 1997).

Within the "Less Greenhouse Gases" program, the ILT brought together employees from different organizational units to co-create novel tools or ways of working. The researched project focused on developing an ML-based inspection tool (ML tool), with a project team of six core members. Throughout the project, a new end-user partially joined the group. The project team comprised members of either the Information, Networking and Programming Directorate or the Supervision and Investigation Service Directorate. Both directorates have different goals within the organization. The Supervision and Investigation Service conducts surveillance and oversight, for example by inspecting objects or companies. The Information, Networking and Programming directorate supports these activities by leveraging information and innovative ways of working.

Two project members enlisted themselves to be project managers within programmes. Before becoming project managers, they were also part of the Supervision and Investigation Service directorate. Throughout the ML tool's development, the project group met periodically. Between these meetings, project team members balanced their regular departmental responsibilities with project-specific tasks. The project managers were accountable to the program manager, with whom they also had various meetings.

The project group developed an ML tool that generates risks scores for potentially fraudulent companies by analysing their online marketplace advertisements. Designed for inspectors, the ML tool supports one of their responsibilities of conducting physical inspections, by identifying high-risk objects or companies. The ML tool was developed through the following process. First, a web scraping pipeline extracts online advertisements for fluorinated greenhouse gases. Then, stakeholders, primarily end-users, label these advertisements. The labelled data trains the ML model to distinguish fraudulent from legitimate advertisements. Then, when presented with new data, the tool predicts advertisement 'riskiness'.

An advertisement with a high predicted risk score indicates potential regulatory violations. For the inspectors, the risk scores serve as proxies for violations by the company that created the advertisement. The scores enable the inspectors to decide for follow-up research or physical company inspections.

3.2. Data collection

We collected data over a period of 12 months, from March 2022 to February 2023. During this period, the first author observed on multiple occasions and took part in some work at the ILT's innovation department, the IDlab. Part of the Information, Networking and Programming directorate, IDlab comprises mostly people with a data-science background dedicated to developing innovative, data-driven solutions that allow the ILT and end-users to make more informed decisions.

The first author was in close contact with the data scientists who were part of the ML tool project. They shared the ML tool and the issues that they encountered during the development and during their interactions with potential end-users. Although the innovation department was the first point of contact for the first author, access to various parts of the organization was possible.

The first author also took part in broader organizational activities. First, he participated in online ID-lab team meetings, in which employees shared various issues and future projects. Furthermore, he attended multiple physical meetings, like the yearly ILT festival, a day of organizational collaboration, team building, and knowledge exchange. Also, he attended a IDlab event connecting innovation-minded stakeholders across departments and directorates. Finally, the author participated in a meeting discussing the ML tools outcomes and the potential for application into new oversight domains. The interactions provided the opportunity for rich contextual observations of developer end-users interactions and ML tool development. Additionally, they gave the opportunity to interact with both stakeholder types.

We triangulated the observations and presence at various on- and offline meetings with secondary data sources such as year reports, (data) strategy documents, frameworks for ML development, and a slide pack about the ML tool and an organization matrix. Triangulation in data collection increases the reliability and credibility of the results (Yin, 2013). The documents provided the necessary context information, and the strategy and organization surrounding the interactions between developers and end-users. Table 1 below gives an overview of the data sources and their use in the analysis.

The first author conducted fourteen semi-structured interviews with fifteen participants. These participants were in the project group (seven), connected to that group, or connected to the program. Fourteen interviews were conducted, with privacy experts interviewed jointly. Table 2 presents an overview of interviewees.

Table 1
Data sources and use in analysis.

Data sources	
Data type	*Use in analysis*
Primary data	
14 semi-structured interviews	Contributed to and enriched our understanding of the worlds of the stakeholders involved and the differences and issues they experienced while working with others.
7 h of presence at online meetings	Contributed to and enriched our understanding of the background and ways of working of different stakeholders.
18 h of presence at physical meetings	Provided broader insight into the workings of the organization, the background of the work of inspectors and data scientists, and the issues that emerge during collaboration.
Secondary data	
4 strategy documents	Provided insight into the goals of programme management and the overall organizational strategy.
5 year-reports	Provided insight into the development of the programme over time.
2 development framework descriptions	Provided insights into the way of working of developers and how they aimed to involve end-users.
1 slide pack about the ML tool	Provided insight into the characteristics of the ML tool.
1 organization matrix	Provided insight into the overall organizational structure.

Table 2
Overview of interviewees.

ID	Interviewee role
I1	Data scientist
I2	Data scientist
I3	Manager
I4	Inspector
I5	Inspector
I6	Privacy expert
I7	Inspector
I8	Manager
I9	Data scientist
I10	Data scientist
I11	Manager
I12	Manager
I13	Project manager
I14	Project manager
I15	Privacy expert

With permission of the interviewees, voice or video recordings were made. The first author transcribed the recordings verbatim afterwards. During the interviews, he took detailed notes, which he re-ordered into a point-wise summary. The participants were selected based on theoretically driven within-case sampling. Hence, we purposefully searched for and contacted people who were potentially able to bring a novel perspective to the case based on their roles and positions within the organization. However, these people still needed to have an adjacent role in the Less Greenhouse Gasses program or the specific ML tool development sub-project. The interview actors included data scientists, inspectors, managers, project managers, and privacy experts. Speaking to persons involved on multiple different sides and with different responsibilities in the project increases the validity of the findings.

The semi-structured interviews had a degree of predetermined order, but also provided flexibility (Longhurst, 2023) and the possibility to go into topics that the interviewee indicated as important. The interview design allowed for exploration and follow-up on responses by the participants. The main themes, loose structure, and questions for the interviews were written down in an interview protocol.

This protocol included questions about the general role of the interviewee, their role in the project, their thoughts on the project, the choices they experienced working on the project, their future aspirations with the ML tool, and the general role of innovative ML tools in their work. Specific questions that were asked to every interviewee were: What is your role? What makes your work complex? However, due to the various and unique backgrounds of the stakeholders, the interviews allowed for exploration tailored to each stakeholders background and experiences.

The data scientists were asked about their development choices and ways for end-user participation. Inspectors provided insight into their experiences with data scientists, how much they felt involved in the development, and the operational implications of novel technologies. The project managers were asked about the interaction dynamics between data scientists and end-users, drawing from their unique position to facilitate and see the interaction. During managerial interviews we asked about strategic choices on various levels in the hierarchy, and explored how these choices could affect boundaries between developers and end-users. Lastly, the privacy experts contributed a valuable perspective, as they could be asked about their experiences with both inspectors and data scientists, and the boundaries that they experienced in working with both within and outside of the ML tool sub-project.

3.3. Data analysis

Our data analysis already started during the data collection. During the data collection, the first author regularly reflected on observations and linked these to related literature. Following the interviews, the coding process started. The first author conducted coding with ATLAS.ti, qualitative data analysis software. The first and second author frequently met up to reflect on the coding and to provide novel input to the codes. The initial phase involved reading the interview transcripts and field notes. While reading the interview transcripts the first author wrote down potential codes and highlighted key statements exemplary for the interviews. This helped to identify important themes. For example, our interest was sparked by the observation that many interviewees discussed the difficulties in collaborating with stakeholders who had divergent roles. This led us in the direction of understanding why these difficulties were present.

Afterwards, we performed open coding on the interview transcripts, which means that the coding followed the data. We performed this coding to identify emergent themes in the data (Williams & Moser, 2019). The data suggested that a reason for collaboration difficulties were the many differences between stakeholders throughout their work together. Informed by literature on knowledge sharing (Bechky, 2003) and managing knowledge across boundaries (Carlile, 2004) we then coded differences between stakeholders while co-creating and initially categorized these in various categories, such as 'language differences', 'expectation differences' and 'knowledge differences'. However, our coding also highlighted that many stakeholders mentioned ways that helped them collaborate more successfully with stakeholders with diverse backgrounds. We then carried out multiple rounds of axial coding, which relates to further refining, aligning, and categorizing the emergent themes (Williams & Moser, 2019). Furthermore, axial coding helps to capture relationships between multiple codes.

Especially in this phase, constant reiteration took place. We noticed that factors inhibited or promoted the successful blurring of boundaries. We realized that the data explained that boundaries persisted but also how boundaries might be blurred. At a later stage, we realized that both categories were linked to co-creation through programme management. Then, the mechanisms emerged through which programmes either allow boundaries to blur, persist, or emerge. Finally, the coding was checked for consistency and overlap. After coding, coded pieces of interview transcript that represent the codes were translated. These quotes are presented as statements from interviewees throughout the findings.

4. Findings

This section discusses the study's findings, categorized into three sections. Section 4.1 discusses how programme management allows boundary blurring. Section 4.2 discusses how programme management allows boundaries to persist. The third section discusses how programme management allows boundaries to emerge.

The findings are discussed as mechanisms that take place. While the mechanisms are described separately for analytical clarity, they are intertwined and form an intricate web that allows for boundaries to be blurred, to persist or to emerge. Fig. 1 synthesizes the different boundary dynamics and mechanisms, and highlights how the different mechanisms impact different boundary types.

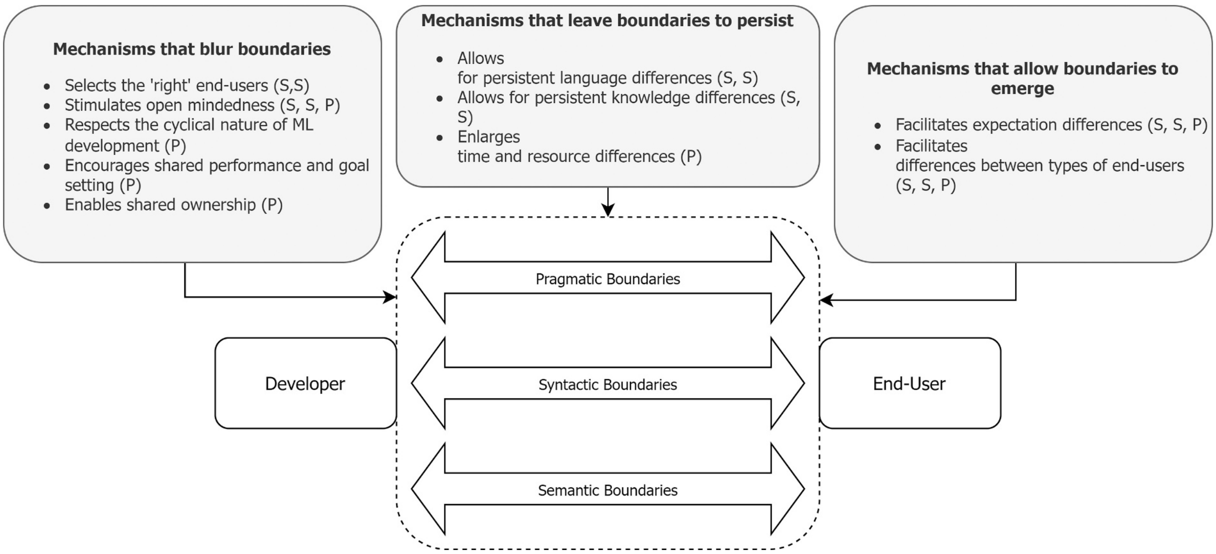

Fig. 1. Overview of boundary dynamics and mechanisms.

4.1. Programme management allows boundaries to blur

The findings indicate the existence of multiple mechanisms through which programme management allows boundaries to blur.

Selects the "right" end-users: Our findings indicate that programme management selects the "right" end-users. Programmes have the reputation of the place where innovations and change happen: "*And for innovation, that's what programmes are for*" *[I11].* This reputation is a reason some end-users are eager to join a programme. Because of its reputation, the programme 'selects' end-users that have a predisposition towards crossing boundaries. These end-users express to their management that they are eager to join the programme. Often such an end-user is someone who has a positive and realistic stance towards technology. They are intrinsically motivated to join the program because they are interested in innovation and change. These end-users are less invested in their domain-specific knowledge and already possess some level of knowledge about and interest in innovation. Hence, the semantic, syntactic and pragmatic boundaries that have to be crossed are relatively small compared to the boundaries between 'regular' end-users and developers. About such end-users it is mentioned that: "*The nicest thing is if you have positive people around it as well because otherwise, you don't get it done" [I7].* The interviewee conveyed the message that for projects that revolve around ML development blurring boundaries is heavily dependent upon the end-users that are involved from the start. This has been emphasized by a data scientist: "*You have to have people who really believe in this in advance" [I2].* These people are end-users who are intrinsically motivated to contribute to innovation and are open to crossing boundaries with developers of such technologies. Within the program, these envisioned end-users are provided space for their willingness to explore and innovate.

Additionally, the right end-users can blur boundaries outside of the programme. These end-users can function as a mediator: *"*I *also ended up being a kind of mediator at some of the consultations, making small presentations about the tool" [I5].* The inspector mentions consultations with other colleagues and future users of the ML system who were not directly involved in the programme. The project team organised consultations with end-users outside of the programme to present the results of the project. The end-users that had been involved mediated between the developers and the more sceptical and less involved potential future end-users. They spanned the boundary between two occupational communities. The right inspector can even function as a kind of 'owner' of the ML system within the group of inspectors, whenever the ML system has been taken up by the team. The inspector might function as *"A coordinator of the outputs from this tool. The inspector ensures that the work for this is done" [I14].* The existence of the programme helps to get the involvement of an end-user with a techno-optimist stance towards ML technology and the abilities and motivations to function as a mediator or coordinator. The right end-user can function as a bridge between both worlds. Without the program, there are fewer opportunities to build bridges.

Stimulates open-mindedness. Our findings indicate that programme management stimulates open-mindedness for stakeholders who are part of a project. Programme management and the projects that occur within the context of programmes are characterized by more freedom and experimentation: "*Incidentally, we have the freedom in a programme to experiment" [I11].* Whenever developers and end-users are part of the program, they can "*Stand away from it, stand above it* "*[I11].* This refers to the developers' and end-users' activities and way of looking at their normal day-to-day work. Their day-to-day work, which happens in the context of their respective teams and departments '*in the line*' is characterized by a distinct perspective. This perspective entails reaching goals that are part of the team's yearly plans, taking common actions, and a way of looking that accompanies those actions and goals. The programme provides the opportunity to take a step back from the line and the way of looking that is stimulated under that banner. The programme allows to temporarily detach from a community of practice and become part of a project and novel community that is not yet set in stone. In some way, both occupational communities start from a somewhat blank slate in terms of practices and the used language. The project gives the opportunity "*to interpret from a broader perspective" [I12].* This also relates to the ambiguousness and complexity of the project. Whereas goals outside of the program are often clear cut and set in stone, goals within the program can be subject to change and the underlying problem to solve as well. Hence, the programme allows the developers and end-users to temporarily free themselves from the mindset that is stimulated in their day-to-day work. Instead, programme management provides the opportunity for developers and end-users to be part of multi-disciplinary project groups, in which these stakeholders can co-create new perspectives in a more open and ambiguous environment.

Respects the cyclical nature of ML development. Our findings indicate that programme management, due to its characteristics, respects the cyclical nature of ML development. It respects a way of working that many developers already practice. The findings are supported by multiple interviewees (I7, I10, I11, I12). The cyclical nature of the program was highlighted by a manager responsible for the program: "*It is important to me to have clear progress reports in the meantime.*" [I11]. These progress reports or progress meetings involve the programme manager and the managers who are responsible for the project. They provide the opportunity to give the project a new direction. In essence, there are multiple moments where decisions regarding the direction of the project can be made. Developers acknowledge that the cyclical nature of ML development is respected. Also, because the development of ML happens within this programmatic context it becomes easier to engage people throughout the different cycles: "*So not only the result but make sure you keep people hooked in the process, maybe get them more motivated*" *[I12].* Programme management closely fits with existing ML development approaches. This is something that developers are used to. However, outside the program context it is hard to respect that process. But, within the program it becomes easier to organize the ML development in a cyclical way, thereby blurring semantic, syntactic and pragmatic boundaries. That programme management respects the cyclical nature also was mentioned by end-users: "*This is something you should do in cycles. Discuss, look at what you have built and be critical of whether you need to go further or not" [I7].* Hence, also the involved end-users highlight that co-creation an ML tool works best in a cyclical way.

Encourages shared performance and goal setting. Our findings indicate that programme management encourages shared performance and goal setting. The program helps to overcome pragmatic boundaries because it encourages to work towards common interest. The mechanism partially comes into play because being in a programme takes time. This has been emphasized by end-users in the project.

An inspector mentioned: "*We have already put our scarce time into this. What if we have zero results*" *[I7]*. The end-users and developers are often part of a project within a programme because they want to be. However, this does not mean that their other obligations are put on hold. Thus, there is a strong wish and incentive to use the time dedicated to the program efficiently. This is also something that one of the project managers noted about the involved developers and end-users: "*Basically they all indicated: I do want to help you, but then I have to be sure that my work leads to results*" *[I13]*. Hence, there is a strong wish present in both for actual performance, or said differently, tangible results. Such a tangible result -. i.e. a 'product' - can serve as a boundary object, bringing together developers and end-users around it. The findings show that shared performance can be encouraged through goal setting. One of the involved end-users mentioned: "*Of course, we said what we hoped to get out of it, but I think in cautious language" [I7]*. The involved developers, end-users and project managers responsible for the project together discussed what the results of the project should be. The group negotiated towards a shared, tangible and clear goal, that they then tried to reach in the brief time available to them. With clarity about the potential result, their time investment became more tangible and boundaries became blurred.

Enables shared ownership: Our findings indicate that programme management enables shared ownership between developers and end-users. Proximity is seen as a condition for shared ownership. A manager stated: "*In my opinion, a programme's strength lies in its ability to bring together all those different roles in this way" [I11]*. What the interviewee refers to is the fact that in the context of programmes different stakeholders in the development of ML technologies can foster close relationships. Such close relationships are less enabled when working outside of the programmes. This is further emphasized by a project manager who mentions: "*And that's sort of what these programmes are for. To pull people from different line departments together" [I14]*. Within a programme, occupations work together on temporary projects. These temporary projects enable shared ownership as developers and end-users work closely together on a goal that they have decided together. In a way, while working in a program, these occupations temporarily can detach from their obligations 'in the line' and come together in close collaboration. Pragmatic boundaries are partially overcome, because developers and end-users are enabled to become shared owners of a project. Both groups feel a responsibility to move the project forward.

The findings also indicate how such shared ownership plays out in practice. The ownership happens because of a proximate collaborative face-to-face relationship between developers and end-users. Both groups of stakeholders meet each other physically and work together on the tool. Both sides provide input and make decisions about steps that need to be undertaken or data or considerations that must be incorporated. One of the interviewees mentioned: "*I personally find the added value of this project that we brought technicians and inspectors together from day one" [I14]*. The interviewee refers to the developers (technicians) and end-users (inspectors) and indicates that bringing both groups together from the beginning has been instrumental. Likewise, another project manager added: "*It has been a co-production*"*[I13]*. The project manager indicates that co-production (i.e. shared ownership) was one of the reasons for the success of the project. The essence of the collaborative and proximate relationship is a non-hierarchical collaboration, in which both occupational communities bring their expertise to the table. This has been acknowledged by inspectors: "*I think we really developed it together*" *[I7]*.

4.2. Programme management allows boundaries to persist

The findings indicate the existence of multiple mechanisms through which programme management allows boundaries to persist. Hence, here we discuss findings that deal with what program management cannot do and why that is the case.

Allows for persistent language differences. The findings indicate that programme management allows for persistent language differences. These language differences are related to the lexicon that developers and end-users have, and the meanings they attach to certain words or terms. Hence, both semantic and syntactic boundaries persisted. This is important, because the co-creation of a useful ML tool demanded the input of end-users. Their input is needed to label the data. If language differences persist, the data quality can suffer, thereby impacting the quality of the ML tool. While the stakeholders met on various occasions and had the opportunity to develop a shared understanding, the temporary nature of the project work and the diverse backgrounds made it hard to create a shared language. As was mentioned: "*So you have to make those translations all the time " [I10]*. This data scientist explained that they constantly had to find the right words that would make it understandable for the end-users, despite both occupational communities working together in a context of co-creation. The data scientists had to make sure that they empathized with the viewpoint and language of the end-users. That different meanings persisted can also be illustrated by a statement made by an end-user: *'Because everyone calls it a tool, except the people who made it and I don't know what word they use for it'"[I5]*. The statement illustrates that despite talking about the same 'thing' (i.e. the ML tool) both occupational communities use different words. In essence, the co-creation provides boundaries around language to partially blur, but some will always remain. The program can only do so much. It is only temporarily, and there are no formalized ways to developed a shared language. Hence, the freedom also has a downside. Furthermore, in times of uncertainty, occupational communities might fall back on what they already know, thereby keeping differences and boundaries intact.

Allows for persistent knowledge differences. The findings indicate that programme management allows for persistent knowledge differences. It was mentioned that *"Data science is like magic for a lot of inspectors." [I2]*. The data scientist indicated that the 'average inspector' has a limited level of knowledge regarding what ML is and how it could help end-users. These knowledge differences seem to persist because the project was focused on tangible results (also see *Encourages shared performance and goal setting)*. This focus on tangible results also strongly ties into the time pressure that especially potential end-users experience. A focus on results and having tangible products along the way might disregard the basis that various stakeholders reason from. Co-creation through programme management cannot change that both groups of stakeholders have a strongly different knowledge basis. Secondly, as noted by one of the interviewees: "*They have no idea what an inspection looks like in practice*" [I4]. This end-user referred to data scientists and the knowledge they have about the activities of inspectors. Programme management brought developers and end-users together. However, they were not present in each other's day-to-day working practices and processes. Instead, they met each other in a context of experimentation and open-mindedness. This helped them to develop a shared basis. However, at the same time, due to the temporary and ad-hoc character of their work together, they were unable to create common knowledge about each other's activities. Likewise, a data scientist mentioned: "*We do not have the substantive knowledge about understanding what is actually happening" [I2]*. The data scientist indicated that it was impossible to have the inspectors' deep base of knowledge based on their experience and obtained tacit knowledge. They are unable to fully understand the reality the end-user sees. Programme management temporarily provides the opportunity for co-creation, but this organizational intervention cannot overcome knowledge boundaries based on years' worth of implicit knowledge obtained through inspections and embedded in the practices of end-users. Also, inspectors and data scientists tend to label each other's knowledge to make it more tangible. An inspector mentioned: "*We always call it knowledge of content and they are the technical knowledge"* [I7]. The statement suggests that data scientists may be viewed solely as technicians, while end users may not possess the same level of technological expertise or interest. The statement emphasizes the different perspectives of developers and end-users. Also, it shows how framing each other's roles might leave boundaries intact, or even reinforce them.

Enlarges time and resource differences. Our findings indicate that programme management enlarges time and resource differences. The context of programmes is one of exploration and space for innovation. Being innovative and working on innovative technologies is one of the main tasks that data scientists are hired for. When these developers are part of projects in the context of programmes, they are given the time and space (by their line managers) to work on the program. It is congruent with their main task in the organization. However, end-users are dealing with a different reality. All interviewed inspectors are experiencing a lack of resources and time pressure. As an inspector noted: *"It's just continuously busy at the moment" [I7]* and another inspector: "*We just don't have the capacity to do something experimental right now" [I5]*. Also, both project managers, who were responsible for the ML tool project, indicated that they observed a strong lack of time and resources from the end-users. They mentioned: "*Ultimately, it is about resources and perhaps a lack of time or a lack of means" [I14]*. The end-users are hired to inspect companies, take the necessary actions based on their findings, and have an overview of their oversight area. However, working on innovative tools in the context of programmes interferes with their daily work and goals. Often, these end-users are assessed based on different measures rather than being innovative. Often, they like to dedicate time towards activities that will help them reach the goals on which they are assessed. However, for developers, programmes are an opportunity to dedicate even more time to innovation and exploration. This creates and enlarges the discrepancy between developers and end-users.

4.3. Programme management allows boundaries to emerge

The findings indicate the existence of multiple mechanisms through which programme management allows boundaries to emerge.

Facilitates expectation differences. Our findings indicate that programme management facilitates expectation differences. This finding does not only relate to involved developers and end-users but also to end-users not directly involved in the programme. Programmes bring together stakeholders from different departments. However, program management does not necessarily solve a discrepancy in expectations. Even more so, it provides the opportunity for involved stakeholders to project their own beliefs and perspectives on what program management can and cannot provide. Some end-users have grand expectations of what programme management and working on innovative tools based on ML will provide them. A developer mentioned: "*They expected there to be golden mountains, but they are not there"[I2]*. The statement illustrates how some end-users had completely different expectations than what the co-creation resulted in. A statement from an end-user adds to this*: "I expected that we would suddenly come across companies that we did not know at all…And that is not what happened*" [I4]. This illustrates the disappointment this end-user experienced about the results of the programme and the ML tool. Despite the existence of the programme, the expectations were not aligned. This might be partially caused by what program management is not. Program management is a temporary way of working together, often cross-divisional. But it also somewhat vague, providing space for differences in expectations. This misalignment in expectations can even be observed between end-users:" They *(other inspectors) had expected that the tool would solve it. And I thought, this makes our job easier" [I5]*. This statement illustrates how programmes, because of the differences in involvement of different end-users can contribute to novel boundaries. Expectation differences can be significant for knowledge boundaries if we assume that overcoming such boundaries requires much effort. Expectation differences among employees create novel boundaries between the willing and unwilling to make this effort.

Facilitates differences between types of end-users. Our findings indicate that programme management facilitates differences between types of end-users. An end-user indicated: "*And there's just really a difficult dividing line and I think you only have that on the user side" [I7]*. Being part of a program means that end-users are temporarily not only working on the targets and tasks from their departments but also working on program-related tasks and goals. Hence, they become part of a novel community and share and build knowledge inside that community with different stakeholders. This also means that the potential end-users that are outside of the program are less involved in the development of the ML tool. In the end, not only end-users involved in the project team but also those outside the program are expected to use ML tools. When end-users in the project become too close with their project team and developers there is a risk that they lose touch with end-users outside of the program. The differences between these groups grow, which makes it harder to eventually transfer a potentially successful ML tool into the organization. Thus, for end-users there is a risk of being too close with their project team: "*I think as a project team we were really close" [I7]*. This can hinder the transfer of an ML As such, co-creation is a double-edged sword tool into the broader group of end-users as they feel even less connected to what their colleagues are doing. This has consequences for the organization. The top management of the organization hopes to gain valuable perspectives and novel ways of working from working in programmes. The idea is that these perspectives and ways of working are continued in 'the line': "*You just want continuity of things that work and are good" [I12]*. However, when differences between types of end-users are facilitated, or even enlarged, they might hinder the eventual use of potentially helpful ML tools. Key to this finding is the quality of programme management. The emerging language on performance and results, as much as the emerging ownership serve as bridges over the boundaries between developers and end-users. At the same time, they lead to differences between end-users who take part in programme management and those that did not participate. The latter did not invest in developing the shared language and are not owners of the results.

5. Discussion

Our study seeks to address the question of how co-creation influences boundaries between developers and end-users in ML development and how boundaries behave as a result. Table 3 highlights the three dynamics that we observed – blurring, persisting, and emerging – and the effects that occur at the different types of boundaries. In presenting these findings as such, we took inspiration from Barrett, Oborn, Orlikowski, and Yates (2012) in their presentation of boundary effects in pharmacy work.

Below, we will elaborate on the main findings in more detail and highlight the contributions to different areas of the literature. First, we discuss how co-creation can have unexpected opposite effects by causing novel boundaries to emerge, referring to this phenomenon as the 'waterbed effect'. Second, we elaborate on how boundaries might persist, while giving the appearance of being blurred. This leads us to highlight the layering of boundaries. Third, we theorize that pragmatic boundaries in co-creating ML are partly the result of differences in goals and incentives coming from the original departments of developers and end-users. These differences lead to a skewed distribution of transaction costs.

5.1. The waterbed effect of co-creating AI

The introduction of programme management brought developers and end-users closer together. This environment facilitated shared ownership, shared goal-setting, and open mindedness. In providing this environment, the developers and end-users were facilitated in overcoming and blurring pragmatic boundaries. The knowledge of both was less 'at stake', because developers and end-users were expected to work towards a common goal. As both were temporarily detached from their normal practices space was offered to form what has been called 'a new joint field' (Levina & Vaast, 2005). As such it seems that co-creation was successful in blurring boundaries between developers and end-users. However, the intervention was only partially 'successful', as new boundaries emerged as well.

Hence, a waterbed effect occurred, that is depicted in Fig. 2. In the context of ML development and boundaries it refers to the undesirable emergence of novel boundaries elsewhere, while trying to blur boundaries in another place. Boundaries were blurred between developers and end-users. At the same time, novel boundaries between different types of end-users emerged. While the program runs, developers and end-users make joint efforts towards a shared goal and create new language within the process. However, when developers and end-users then return to their teams, especially the end-users who were involved experience emergent boundaries. The programme's goals that developers worked on and the language that was formed are far removed from the end-users original practices. Hence, novel boundaries between end-users have emerged.

That novel boundaries can emerge as a consequence of organizational interventions aimed at blurring boundaries has also been highlighted by Waardenburg et al. (2022). Our findings substantiate this claim within the specific context of ML development in public organizations in two ways. First, we show that an intervention with the explicit aim to connect occupational groups may lead to

Table 3
Description of boundary dynamics and boundary effects.

Boundary dynamics	Boundary effects
Blurring	• Blurring of the boundary between occupational groups • Increasing the knowledge base of occupational groups and reducing the distance between knowledge bases
Persisting	• The lack of impact on deep-rooted boundaries leaves boundaries between occupational groups to persist • Persistence of syntactic and semantic boundaries instead of pragmatic boundaries (layering)
Emerging	• Measures to blur boundaries have unexpected opposite effects by causing novel boundaries to emerge (waterbed effect) • Boundaries within occupational groups through divergent expectations and involvement

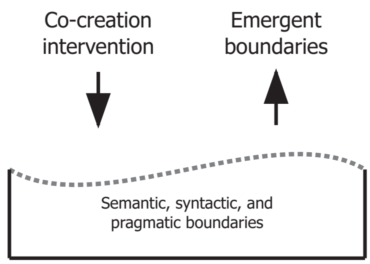

Fig. 2. Illustration of waterbed effect.

emerging boundaries somewhere else. Second, we show that unexpected effects may even occur when developers and end-users are in direct contact with each other, instead of only through a liaison position, as in the work by Waardenburg et al. (2022). These findings challenge the manageability of boundaries. While interventions to connect multiple groups might appear effective on the level of the focal community, a broader systemic perspective reveals the emergence of novel boundaries. More broadly, our finding adds to the literature that calls for the need to blur boundaries, and highlights how interventions aimed to do just that play out in practice (Mayer et al., 2023; Waardenburg & Huysman, 2022).

The findings call for more research on the dynamics of knowledge boundaries. An exploration of causalities between boundary blurring and emerging could be promising. Theorizing may suggest that 'waterbed effects' are likely to occur when key resources – such as time - remain constant. In our case study, a part of the end user's community was able to engage in a joint learning process with developers, accidentally creating a new boundary with the remaining end-users. Empirical research testing these causalities may provide valuable insights into the distribution of novel knowledge. Such an investigation could illustrate how novel knowledge – in this instance, ML expertise disperses across an organization and at which pace.

5.2. The layering of knowledge boundaries

A second area of contribution concerns the understanding of boundaries as they appear in ML development. As noted, through co-creation developers and end-users were brought closer together and worked towards the same goal. On the surface it seemed as if semantic, syntactic, and pragmatic boundaries were overcome. However, as the process carried on it became clear that semantic and syntactic boundaries were more entrenched than expected. The developers had to put effort into translating machine learning concepts in words that the end-users would understand. However, even if this effort seemed successful words proved to have different meanings among the communities. It seems as if semantic and syntactic boundaries were more fine-grained than expected and blurred slowly, while pragmatic boundaries seemed to be overcome quite quickly. As such, co-creation can give the appearance of syntactic and semantic boundaries being blurred, but they may implicitly remain.

These observations imply that different types of boundaries may change in a different pace. As such, we suggest that knowledge boundaries can be seen as layered phenomena. Boundaries are not stand-alone and homogeneous entities, but a combination of multiple boundaries that interact. Some boundaries may persist while others blur. Some boundaries blur, while others persist implicitly.

The interplay between boundaries and the different 'layers' that exist have also been described by Carlile (2004). It is mentioned that the more complex boundaries become - with syntactic boundaries being the least complex and pragmatic the most – that '*the process or capacity at a more complex boundary still requires the capacities of those below it* ' (Carlile, 2004, p.560). This suggests that for blurring pragmatic boundaries, the blurring of syntactic and semantic boundaries is a prerequisite. However, we have not found blurred syntactic and semantic boundaries to be a prerequisite for blurred pragmatic boundaries, as semantic and syntactic boundaries actually persisted, while pragmatic boundaries appeared to be blurred. This could be explained by an implicitness of syntactic and semantic boundaries, as they are grounded in both wording and meanings. The wordings seem measurable, the meanings to a lesser extent. This difference may be significant in the case of ML, a technology that not only introduces a new vocabulary, but also a novel way of thinking about the work processes, in our case risk-based oversight.

The observation that pragmatic boundaries seemed to blur quicker than semantic and syntactic boundaries may be subject to case selection bias. To blur pragmatic boundaries was more or less the programs' aim. Developers and end-users were intentionally united through co-creation goals. However, the case highlights the tenacity of semantic and syntactic boundaries, and the possibility that different boundary types may blur independently from each other. More empirical research could explore this relationship, for instance studying interventions aimed at blurring syntactic, semantic and pragmatic boundaries separately.

5.3. Work processes as source of pragmatic boundaries in ML development

The last area of contribution is the understanding of the source of pragmatic boundaries in ML development. Carlile (2004) notes that a pragmatic boundary arises when knowledge cannot be shared because of different interests among actors, thereby introducing a political dimension to knowledge sharing. Carlile (2004, p.559) states that costs "*are not just the costs of learning about what is new, but also the costs of transforming 'current' knowledge being used (*i.e.*, common and domain-specific knowledge)*". These transaction costs negatively affect the willingness to share knowledge. Carlile (2004) also discusses that we should not assume the actors involved at a boundary occupy politically equal positions in representing their knowledge to each other.

Our research reveals that while shared ownership and common goal-setting initially seemed to blur pragmatic boundaries, looking beyond the surface exposes transaction costs as pointed out by Carlile (2004). We also observed an uneven distribution of transaction costs in ML development. End-users encountered significant obstacles in the form of impenetrable ML development language and unfamiliar working processes. The cyclical nature of programme management aligns more closely with ML development processes and the workflow of developers then of end-users.

This misalignment leads to asymmetry. End-users must adapt to a novel reality that is fundamentally different from what they are used to. Developers, on the other hand, operate within familiar territory aligned with their innovation-focused objectives. The political dimension of pragmatic boundaries in ML development thus emerges from these inherent work process disparities. This is also a key contribution to the literature that highlights the need for interaction (Waardenburg & Huysman, 2022) as it highlights how unequal positions may lead to interaction issues. ML development presents additional challenges beyond standard knowledge transformation. Its data-driven, algorithmic nature represents not only a language boundary, but may also demand fundamental epistemological shift from end-users.

These points prompt critical future research avenues. First, we suggest that studies question the legacies that determine the (un) equality between actors that want to overcome pragmatic boundaries. How is the distribution of transaction costs among these actors determined by their work processes and other legacies? Does the distribution bias apply to all innovation processes, and would the impenetrability of ML to end-users as well as a change of epistemology lead to an amplification of this distribution bias? Empirical work focusing on these questions is promising, since it may help to predict power distribution among ML developers and end-users.

5.4. Limitations and further research perspectives

While our findings offer insight into boundary dynamics due to co-creation, several limitations need mentioning. First, our single case study design may limit external validity of the findings. Second, selection bias may influence our findings, as the studied organization seemed mature in their ML implementation. Hence, this is likely to have contributed to the openness of this organization to research participation. Third, our focus on interactions between developers and end-users excludes other crucial occupational communities needed in ML development, such as legal experts and ICT architects. This is a limitation because ML is a technology of many hands (van Krimpen, de Bruijn, & Arnaboldi, 2023; Zweig, Wenzelburger, & Krafft, 2018).

Future research opportunities are manyfold, as discussed above. First, we suggest studies into causalities between boundaries blurring and emerging. Also, we suggest studies that focus on the (inter)dependencies between different types of boundaries and how interventions interact with different boundary types. Next, we suggest research on the distribution bias of transaction costs. These future studies could reveal new dynamics around waterbed effects, layered boundaries and transaction costs.

6. Conclusion

This study aims to answer the question: *How does co-creation influence boundaries between developers and end-users in ML development and how do boundaries behave as a result?* The study suggests that co-creation has mixed and unpredictable effects on boundaries between ML developers and end-users. While co-creation has various mechanisms to blur boundaries, like selecting the right end-user and enabling shared ownership, our study demonstrates the persistence of boundaries. Indeed, co-creation interventions like programme management may generate a 'waterbed effect', where blurring boundaries in one place, triggers the emergence of boundaries in another place.

Our investigation of boundaries and boundary dynamics showed an interplay of characteristics of boundaries and a layering of boundaries. We found differences in pace between the blurring of distinct types of boundaries. Some mechanisms that potentially blur boundaries emerge quickly, while others tend to persist longer. These mechanisms, like shared ownership, may allow for conditions to cross more persistent boundaries in the long run. Counterintuitively, in ML development co-creation, pragmatic boundaries seem overcome faster than semantic or syntactic boundaries. The findings call for a layered conception of knowledge boundaries, wherein old and new practices co-exist and interact. Finally, our analysis reveals that blurring knowledge boundaries through co-creation is characterized by the presence of transaction costs, stemming from work processes in program management that characterize ML development processes.

This study emphasizes the message by Bechky (2003) once more. She illustrated that '*interactions between members of different communities, while sometimes painful, can lead to enriched understanding, particularly when a tangible object is offered to ground the interaction.*' Also, in the case of the co-creation of ML, there seems much to gain by seeking ways to increase the interactions between developers and end-users in an open, collaborative, and equal manner. However, the 'waterbed effect' tells us that there is no world without boundaries. Given that knowledge boundaries may impede organizational learning, the motivation for co-creation interventions remains clear. However, for these interventions to be effective, we call for more understanding about how co-creation interventions impact boundaries and how boundaries behave and interact.

CRediT authorship contribution statement

Floris van Krimpen: Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization, Writing – review & editing, Writing – original draft. **Haiko van der Voort:** Supervision, Conceptualization, Writing – review & editing, Writing – original draft.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used DeepL and Claude in order to perform proofreading and editing on the written manuscript. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

None.

Acknowledgements

The authors thank handling editor Michael Barrett and the anonymous reviewers for their valuable feedback and guidance during the revision process. Furthermore, they thank the ILT, more specifically Stephanie Wassenburg and Corline Koolhaas, for their supporting role in data collection.

Data availability

Data will be made available on request.

References

Alexopoulos, C., Lachana, Z., Androutsopoulou, A., Diamantopoulou, V., Charalabidis, Y., & Loutsaris, M. A. (2019). How machine learning is changing e-government. In *Proceedings of the 12th international conference on theory and practice of electronic governance, part F1481* (pp. 354–363). https://doi.org/10.1145/3326365.3326412

Bailey, D., Faraj, S., Hinds, P., von Krogh, G., & Leonardi, P. (2019). Special issue of organization science: Emerging technologies and organizing. *Organization Science, 30*(3), 642–646. https://doi.org/10.1287/orsc.2019.1299

Barocas, S., & Selbst, A. D. (2016). Big Data's Disparate Impact. *California Law Review, 104*(671), 671–732. https://doi.org/10.2139/ssrn.2477899

Barrett, M., Oborn, E., Orlikowski, W. J., & Yates, J. A. (2012). Reconfiguring boundary relations: Robotic innovations in pharmacy work. *Organization Science, 23*(5), 1448–1466. https://doi.org/10.1287/orsc.1100.0639

Bechky, B. A. (2003). Sharing meaning across occupational communities: The transformation of understanding on a production floor. *Organization Science, 14*(3). https://doi.org/10.1287/orsc.14.3.312.15162

Berk, R. (2017). An impact assessment of machine learning risk forecasts on parole board decisions and recidivism. *Journal of Experimental Criminology, 13*(2), 193–216. https://doi.org/10.1007/s11292-017-9286-2

van den Broek, E., Sergeeva, A., & Huysman Vrije, M. (2021). When the machine meets the expert: An ethnography of developing AI for hiring. *MIS Quarterly, 45*(3), 1557–1580. https://doi.org/10.25300/MISQ/2021/16559

Brown, J. S., & Duguid, P. (1991). Organizational learning and communities-of-practice: Toward a unified view of working, learning, and innovation. *Organization Science, 2*(1), 40–57.

Burrell, J. (2016). How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data & Society, 3*(1), 1–12. https://doi.org/10.1177/2053951715622512

Carlile, P. R. (2002). A pragmatic view of knowledge and boundaries: Boundary objects in new product development. *Organization Science, 13*(4). https://doi.org/10.1287/orsc.13.4.442.2953

Carlile, P. R. (2004). Transferring, translating, and transforming: An integrative framework for managing knowledge across boundaries. *Organization Science, 15*(5), 555–568. https://doi.org/10.1287/orsc.1040.0094

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM, 55*(10), 78–87. https://doi.org/10.1145/2347736.2347755

Eggers, W. D., Schatsky, D., & Viechnicki, P. (2017). *AI-augmented government. Using cognitive technologies to redesign public sector work* (pp. 1–24). Deloitte Center for Government Insights. https://www2.deloitte.com/content/dam/insights/us/articles/3832_AI-augmented-government/DUP_AI-augmented-government.pdf%0Ahttps://www2.deloitte.com/us/en/insights/focus/cognitive-technologies/artificial-intelligence-government.html.

Faraj, S., Pachidi, S., & Sayegh, K. (2018). Working and organizing in the age of the learning algorithm. *Information and Organization, 28*(1), 62–70. https://doi.org/10.1016/j.infoandorg.2018.02.005

Ferns, D. C. (1991). Developments in programme management. *International Journal of Project Management, 9*(3), 148–156. https://doi.org/10.1016/0263-7863(91)90039-X

Galbraith, J. R. (1973). *Designing complex organizations*. Addison-Wesley Publishing Company.. https://books.google.nl/books?id=FUe3AAAAIAAJ

Holmström, J., & Hällgren, M. (2022). AI management beyond the hype: Exploring the co-constitution of AI and organizational context. *AI & Society, 37*(4), 1575–1585. https://doi.org/10.1007/s00146-021-01249-2

Kellogg, K. C., Orlikowski, W. J., & Yates, J. A. (2006). Life in the trading zone: Structuring coordination across boundaries in postbureaucratic organizations. *Organization Science, 17*(1), 22–44. https://doi.org/10.1287/orsc.1050.0157

van Krimpen, F., de Bruijn, H., & Arnaboldi, M. (2023). Machine learning algorithms and public decision-making: A conceptual overview. *The Routledge Handbook of Public Sector Accounting, 124–138*. https://doi.org/10.4324/9781003295945-12

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation.* Cambridge university press.

Lawrence, P. R., & Lorsch, J. W. (1967). *Organization and environment: Managing differentiation and integration.* Division of Research, Graduate School of Business Administration, Harvard University. https://books.google.nl/books?id=zy1HAAAAMAAJ.

Lebovitz, S., Levina, N., & Lifshitz-Assa, H. (2021). Is AI ground truth really true? The dangers of training and evaluating AI tools based on experts' know-what. *MIS Quarterly, 45*(3), 1501–1526. https://doi.org/10.25300/MISQ/2021/16564

Levina, N., & Vaast, E. (2005). The emergence of boundary spanning competence in practice: Implications for implementation and use of information systems. *MIS Quarterly, 29*(2), 335. https://doi.org/10.2307/25148682

Longhurst, R. (2023). Semi-structured interviews and focus groups. In N. Clifford, M. Cope, & T. Gillespie (Eds.), *Key Methods in Geography* (4th Edition). SAGE Publications. https://books.google.nl/books?id=1qu1EAAAQBAJ.

Lorenz, L., Van Erp, J., & Meijer, A. (2022). Machine-learning algorithms in regulatory practice agencies. *Technology & Regulation, 1–11*. https://doi.org/10.26116/techreg.2022.001

Mayer, A. S., van den Broek, E., Karacic, T., Hidalgo, M., & Huysman, M. (2023). Managing collaborative development of artificial intelligence: Lessons from the field. In *Proceedings of the 56th Hawaii international conference on system sciences, 4* (pp. 6139–6148).

Nonaka, I., & von Krogh, G. (2009). Tacit knowledge and knowledge conversion: Controversy and advancement in organizational knowledge creation theory. *Organization Science, 20*(3), 635–652. http://www.jstor.org/stable/25614679.

Orlikowski, W. J. (2002). Knowing in practice: Enacting a collective capability in distributed organizing. *Organization Science, 13*(3), 249–273. https://doi.org/10.1287/orsc.13.3.249.2776

Orr, J. E. (1996). *Talking about machines: An ethnography of a modern job.* Cornell University Press.. https://doi.org/10.7591/j.ctt1hhfnkz

Pellegrinelli, S. (1997). Programme management: Organising project-based change. *International Journal of Project Management, 15*(3), 141–149. https://doi.org/10.1016/S0263-7863(96)00063-4

Prevaas, B. (2018). Werken aan programma's. https://www.werkenaanprogrammas.nl/.

von Richthofen, G., Ogolla, S., & Send, H. (2022). Adopting AI in the context of knowledge work: Empirical insights from German organizations. *Information (Switzerland), 13*(4), 1–16. https://doi.org/10.3390/info13040199

Siggelkow, N. (2007). Persuasion with case studies. *Academy of Management Journal, 50*(1), 20–24. https://doi.org/10.5465/AMJ.2007.24160882

Ulbricht, L., & Yeung, K. (2022). Algorithmic regulation: A maturing concept for investigating regulation of and through algorithms. *Regulation and Governance, 16*(1), 3–22. https://doi.org/10.1111/rego.12437

Van Maanen, J., & Barley, S. R. (1984). Occupational communities: Culture and control in organizations. In *, Vol. 6. Research in organizational behavior* (pp. 287–365). JAI Press, Inc.

Veale, M., Van Kleek, M., & Binns, R. (2018). Fairness and accountability design needs for algorithmic support in high-stakes public sector decision-making. In *Proceedings of the 2018 CHI conference on human factors in computing systems* (pp. 1–14). https://doi.org/10.1145/3173574.3174014

van der Voort, H. G., Klievink, A. J., Arnaboldi, M., & Meijer, A. J. (2019). Rationality and politics of algorithms. Will the promise of big data survive the dynamics of public decision making? *Government Information Quarterly, 36*(1), 27–38. https://doi.org/10.1016/j.giq.2018.10.011

Waardenburg, L., & Huysman, M. (2022). From coexistence to co-creation: Blurring boundaries in the age of AI. *Information and Organization, 32*(4), Article 100432. https://doi.org/10.1016/j.infoandorg.2022.100432

Waardenburg, L., Huysman, M., & Sergeeva, A. V. (2022). In the land of the blind, the one-eyed man is king: Knowledge brokerage in the age of learning algorithms. *Organization Science, 33*(1), 59–82. https://doi.org/10.1287/orsc.2021.1544

Waardenburg, L., Sergeeva, A., & Huysman, M. (2018). Hotspots and blind spots: A case of predictive policing in practice. In *, Vol. 543. IFIP advances in information and communication technology*. Springer International Publishing. https://doi.org/10.1007/978-3-030-04091-8_8.

Weick, K. E. (1979). *The social psychology of organizing (issue second edition)*. Random House. https://doi.org/10.3917/mana.182.0189

Williams, M., & Moser, T. (2019). The art of coding and thematic exploration in qualitative research. *International Management Review, 15*(1), 45–55.

Yeung, K. (2018). Algorithmic regulation: A critical interrogation. *Regulation and Governance, 12*(4), 505–523. https://doi.org/10.1111/rego.12158

Yin, R. K. (2013). *Case study research: Design and methods* (5th ed.). SAGE Publications Inc.

Zweig, K. A., Wenzelburger, G., & Krafft, T. D. (2018). On chances and risks of security related algorithmic decision making systems. *European Journal for Security Research, 3*(2), 181–203. https://doi.org/10.1007/s41125-018-0031-2