A cognitive BPM theory for knowledge-intensive processes

João Carlos de Almeida Rodrigues Gonçalves
Free University of Bozen-Bolzano, Bolzano, Italy
Fernanda Araujo Baião
Department of Industrial Engineering, Pontifical Catholic University of Rio de Janeiro, Rio de Janeiro, Brazil
Flavia Maria Santoro
Inteli Institute of Technology and Leadership, São Paulo, Brazil, and
Giancarlo Guizzardi
Free University of Bozen-Bolzano, Bolzano, Italy and Services and Cybersecurity Group, University of Twente, Enschede, The Netherlands

Received 23 November 2021
Revised 14 May 2022
1 October 2022
Accepted 20 December 2022

Abstract

Purpose – A literature review was conducted in order to establish a detailed definition of a knowledge-intensive process (KiP). Moreover, relevant theories from humanities – especially the fields of linguistics and philosophy – were thoroughly researched and adapted for establishing descriptive and analytical foundations for the phenomena involved. Finally, a cognitive business process management (BPM) theory was proposed in order to assess how sufficient are its respective explanatory and predictive powers. This study aims to discuss the aforementioned objective.

Design/methodology/approach – This study proposes a novel theory for KiPs that describe the process flow based on the participants' interactions and their beliefs, desires and intentions as the main drivers of the process enactment and execution.

Findings – The proposal puts forth a comprehensive definition of a KiP, depicting knowledge intensity, participant interaction and decision-making. The inner dynamics of each of these two elements (as well as other associated elements) are described as an information systems (IS) theory that enables the study of KiPs in detail, going beyond the typical techniques of the BPM field and common obstacles.

Originality/value – A theory proposal for KiP that applies concepts from speech act theory and intentional states as the main drivers for understanding the process dynamics are, to the best of the authors' knowledge, not present at the literature. Being an original proposal, the real-world scenario discussed brings up the explanatory and predictive powers of the theory as well as its innovative value for research in the field.

Keywords Knowledge-intensive process, Business process management, Speech act, Common ground, Intentionality, Cognitive BPM

Paper type Research paper

1. Introduction

Business process management (BPM) as a research field has evolved, and many new research questions emerged, many of them involving the dynamics of the knowledge management cycle within an organization. One of these developments is the cognitive BPM paradigm (Hull and Motahari-Nezhad, 2016), which fosters the application of cognitive computing technologies to the BPM ecosystem. Moreover, the evolution of BPM to a process science (Rescher, 1996) requires more than understanding the phenomena of a business process by solely tackling the technological or applied facets (e.g. BPMS systems or process mining) or focusing only on the socio-technical aspects (organizational and business dynamics within or without the organization where the business process occurs).

Fernanda Araujo Baião is partially funded by FAPERJ (grants 200.514/2023 and 211.308/2019) and CNPq (grants 312059/2022-1 and 422810/2021-5).

Business Process Management Journal
Vol. 29 No. 2, 2023
pp. 465-488

1463-7154
DOI 10.1108/BPMJ-11-2021-0746

Many researchers distinguish a specific type of unstructured process as being critical to most business scenarios, a.k.a. knowledge-intensive process (KiP). According to Hagen *et al.* (2005), a business process is knowledge-intensive if its aggregated value can only be reached through the fulfillment of the knowledge requirements of the process participants, while Gronau and Weber (2004) define that KiPs are also characterized by a dynamic and unstable control-flow, and by the execution of complex activities that frequently change over time and even at runtime. All these characteristics demand a precise understanding of all the circumstances that lead to each varying process execution. Examples of KiP span from the strategic level of an enterprise (such as defining next year's strategic plan or specifying a new sale's strategy) to its operational level (such as data exploration and modeling, information computer technology (ICT) troubleshooting, air traffic control and healthy diagnosis or crisis management). A direct link between KiPs and innovation can be traced, as the degree of knowledge intensity is an important factor for measuring how an organization handles knowledge and thus, its knowledge-based processes (Andreeva and Kianto, 2011).

Based on an extensive literature review, Di Ciccio *et al.* (2014) affirmed that KiPs are processes "whose conduct and execution are heavily dependent on knowledge workers performing various interconnected knowledge intensive decision-making tasks". According to those authors, the expertise, experience and decision-making capabilities inherent to knowledge workers essentially determine this kind of process. Furthermore, they derived eight key issues that typically characterize a KiP, namely: knowledge-driven; collaboration-oriented; unpredictable; emergent; goal-oriented; event-driven; constraint-and rule-driven; and non-repeatable. Additionally, Little and Deokar (2016) investigated the relevance of knowledge creation in KiP and argued that the expansion and use of knowledge across organizations rely on social processes, both formal and informal, through effective communication.

However, despite the relevance of such a type of process, as pointed out by Işik *et al.* (2013), the lack of a precise definition of a KiP is still missing. The definitions provided mention informality, collaboration, unpredictability and decision-making driven but they do not explain how the decisions are collaboratively made or how many different instances an "unpredictable" process should have to be considered a KiP, for example. We observe that there are non-structured processes or processes with a reduced number of instances that are characterized as KiPs, as well as well-structured, routine processes with a great number of instances that are also considered as KiPs. Thus, there is a necessity for a thorough characterization, one that precisely describes further details beyond the structure or number of instances.

We position ourselves on the argument that the human factor is the main source of complexity of a KiP, especially due to the hurdle of modeling human behavior in contrast to more structured and less human-centric operational processes. There is a clear gap between data-based approaches such as process mining and approaches that deal with the human factor in BPM. Also, there are limitations regarding how much of human behavior can be extracted from datasets, usually extracted from systems such as BPMS or workflow management systems. Especially in the case of KiP, this gap seems even wider, as human behavior is one of the main sources of unpredictability. This scenario is intensified as digital transformation initiatives and the introduction of non-human agents in the processes increasingly use the knowledge accumulated in large databases to define the steps of a process. Despite this complexity, the cognitive elements which represent the human perspective of a KiP are crucial to understanding its behavior and, without precisely addressing their semantics, it is critical to establish a proper ground to understand and further advance investigation on KiP. In this direction, the present research investigates the following hypothesis H1.

H1. Beliefs, desires and intentions drive KiPs.

(H1 is decomposed into H1.1 and H1.2, as follows.)

H1.1. Beliefs drive the decisions about the courses of action to be / not to be undertaken at a KiP.

H1.2. Desires and intentions depict, respectively, the possible and chosen courses of action of a KiP.

We propose a theory – aligned with the cognitive BPM paradigm – that precisely defines the cognitive aspect of a KiP and describes knowledge-intensity, its essential characteristic, in a way that can be useful for BPM researchers and practitioners. The proposed theory also encompasses the typical business process concepts, such as activities and roles. The theory may be applied as a conceptual basis for a semantically precise understanding of a KiP behavior and for expliciting the rationale behind the decisions made during specific KiP executions, as a core framework for defining new performance metrics and indicators for a KiP and for applying knowledge extraction techniques from execution data, or even as a metamodel for developing a KiP modeling language that addresses its cognitive perspective.

The paper is organized as follows: Section 2 provides the background on the KiP field, including the main difficulties and knowledge gaps that our proposal aims to address; Section 3 briefly describes the theoretical foundations of the proposal, describing the theories and concepts, from other fields of knowledge such as linguistics and psychology, that are going to be used to develop the proposal; Section 4 presents the theory proposal, including an ontology model depicting the theory's constructs and relationships, as well as an overview of information systems theories; Section 5 includes a discussion, using a real case study based on open source software development, about the explanatory and predictive powers of the proposal and Section 6 concludes the paper, with the possible applications, contributions and future work.

2. Knowledge-intensive processes

Traditional approaches for process modeling usually depict a process focusing on the control flow, thus defining it as a composition of well-structured activities or other processes (sub-processes) that an organization performs in order to achieve its goals (Weske, 2007). Each activity of the process is characterized by its composing sub-activities, pre-events and post-events, pre-conditions and post-conditions, input and output artifacts, required resources and the procedures (methods, techniques) to be followed when performing the activity. Particularly in the software domain, a software process can interact with other processes in several ways, e.g. a process can precede the execution of another, two processes can be executed in parallel, or a process can be executed at a specific moment during the execution of another process (Falbo and Bertollo, 2009).

Regarding process structure and the flow of activities, Hagen *et al.* (2005) classify business processes as structured, semi-structured or unstructured. Structured processes are completely pre-defined, easily modeled using a specific language such as business process model and notation (BPMN), and repetitive, having a fixed sequence of activities. Examples of structured processes are attendance orders, deliveries, inventory control and payroll.

Unstructured (or *ad hoc*) processes comprise a kind of process that changes frequently, with its instances being very different from each other, both in terms of activities performed and flow. Its nature brings additional difficulty to model with a traditional method or notation. Finally, a semi-structured process shares unstructured and structured parts, sharing traits of both process types on different parts of its flow.

Among the diverse definitions of KiPs, a concise and brief definition is found on (Vaculin *et al.*, 2011), defining KiP as "processes whose conduction and execution are heavily dependent on knowledge workers performing various interconnected knowledge-intensive, decision-making tasks". KiPs are genuinely knowledge-, information- and data-centric and require substantial flexibility at both design- and run-time.

Expanding on this brief definition, complementary studies (Unger *et al.*, 2015) discuss a set of ten common characteristics for KiPs:

(1) Knowledge-prevalence: Knowledge is of utmost importance for the process. Usually, knowledge from different sources and/or tacit knowledge is necessary for process execution.

(2) Collaboration: KiPs include activities often executed by many different process participants. Intensive information exchange and coordination between them being a vital part of process execution itself;

(3) Predictability: Due to its unstructured nature, the flow of activities of a KiP can vary at each instance, due to situation-specific needs or constraints.

(4) Complexity: The coordination of multiple information sources, the variety of its execution flow, the variety of sub-processes associated with the process itself and the large number of participants makes complexity a key characteristic of a KiP;

(5) Structure: It is only possible to define a workflow that depicts a KiP partially, as unpredictable decisions or tasks guided by creativity are an inherent part of the flow of activities, as well as knowledge flows and knowledge transfers between media and persons being necessary to achieving a successful process completion (Gronau and Weber, 2004);

(6) Goal-orientation: Although the unpredictable nature and complexity of KiPs is an obstacle to achieve a consistent structure, a minimum of structure can be achieved by defining milestones or intermediate goals during process execution;

(7) Event-driven: Internal and external events may affect the quality of information exchanged during a KiP execution or require a participant to react for the successful achievement of the intended KiP process goal;

(8) Repeatability: The exact flow and order of activities, during each instance execution of a KiP, depends on several situational and contextual factors as well as possible external events that affect its participants. KiPs tend to be less repeatable than non-KIP processes, so an exact repetition of an instance, in terms of flow and activities of a previously executed KiP, seems hardly possible, due to the variety of factors affecting each specific execution or instance;

(9) Frequency and time-horizon: KiPs tend to have longer run times than non-KiPs and, due to work changing hands over time and the inherent complexity of the process flow itself, no single individual has a full view of the process instance as a whole.

(10) Context dependency: The information necessary for KiA execution usually comes from the context of the process or activity itself, being, for example, application data, process data, functional data, etc.

Due to the dynamic and people-centric traits of a KiP, Haarmann *et al.* (2021) argue that there are several hindrances to its analysis and modeling. Being complex and human-centric, KiP generates value through the knowledge exchange between participants, usually involving decision-making and problem-solving tasks with diverse alternatives for the same activity and different possibilities for the next step in the process flow, besides a tacit rationale for their choice. Thus, the main challenge to precisely understanding the essence of a KiP is its high variability, since the cognitive elements (e.g. beliefs, desires and intentions) which determine the flow of activities and decision-making rationale of a process instance are often neglected or not made explicit. Hence, process participants and stakeholders alike very

frequently find them difficult to describe completely. An important consequence of this issue is, for example, taking into consideration a point in time, during the process execution, there are many possible "next" activities in the instance flow and the relationship between a particular activity and a process objective may not be apparent (Haarmann *et al.*, 2021).

Going beyond the characterization of a KiP, some authors (Sarnikar and Deokar, 2017; Berniak-Wozny and Szelagowski, 2021) discuss the necessity of investigating precisely the nature of a knowledge-intensive task or activity, and how they differ from a common activity in a process flow. Sarnikar and Deokar (2017) point out a knowledge-intensive task as "requiring high agent innovation, involving multiple decision paths, contingent upon numerous eventualities and being highly dependent on agent actions". Thus, they have a direct impact on the course of each process instance and, more importantly, the value they provide (Berniak-Wozny and Szelagowski, 2021).

Within this perspective, on the activity level, a KiP is a composition of business process activities, among which some of them can be knowledge-intensive (called here knowledge-intensive activity (KiA)). According to Witschel *et al.* (2010), the execution of a KiA is always optional in a process instance, depending on specific information). The information necessary to decide whether to execute or not a KiA usually comes from the context of the process or the activity itself, being, for example, application data (e.g. system files), process data (e.g. workflow system traces) or any other type of data or information about needed resources (Brander *et al.*, 2011). Here the literature mentions another problem frequently found during the execution of this kind of process, the "knowledge hiding", especially knowledge that is related to the activity or the process flow, negatively impacting the team and overall performance of the KiP instance (Chatterjee *et al.*, 2021a).

Barboza *et al.* (2018) further specified a KiP as a composition of prospective activities (events) whose execution contributes to fulfilling a goal and whose control-flow, at the instance level, typically presents a high degree of variability among its past executions. One or more of these activities are classified as a "knowledge-intensive activity" (KiA). Hence, each execution of a KiP (i.e. its realization) is a sequence of activities with at least one occurrence of one KiA (events) that happens toward accomplishing the KiP goal.

In this paper, we argue that the execution of a KiP is causally dependent on the beliefs, desires and intentions that are inhered in the process participants, and historically dependent on the communicative interactions among KiP participants. The following section describes the foundations of our proposal that enables the analysis of these elements in order to tackle the difficulty of modeling and executing a KiP.

3. Theoretical foundations

In order to put our proposal into context, we introduce a brief discussion on what constitutes a theory in Information Systems, as well as how the different theories outside of the fields of BPM and information systems (IS) are used in our proposal.

3.1 Theories in information systems

IS is an applied science research field; however, in spite of the importance that theory development has in IS research, the development of new theories and the refinement of existing theories have been relatively neglected within the information systems discipline (Weber, 2012). In this sense, the IS discipline explores a specific type of phenomenon, going beyond the mere study of either the technological system or the social system, or even the two side-by-side; in fact, IS as a discipline investigates the phenomena that emerge when the two systems interact (Gregor, 2006).

Our research adopts the following definition of a theory: "A theory is an abstract entity that aims to describe, explain and enhance understanding of the world and in some cases to provide

predictions of what will happen in the future and to give a basis for intervention and action" (Gregor, 2006). Central to many understandings of a theory are the goals of explanation and prediction. Explanation is closely linked to human understanding, as an explanation can be provided with the intent of inducing a subjective state of understanding in an individual. Apart from explanations, theories can also aim at making predictions, which allows the theory both to be tested and to be used to guide action. In the same paper, she proposes a method for classifying IS theories, with a focus on four primary goals:

(1) Analysis and description: The theory provides a description of the phenomena of interest, analysis of relationships among those constructs, the degree of generalizability in constructs and relationships and the boundaries within which relationships, and observations hold;

(2) Explanation: The theory provides an explanation of how, why and when things happened, relying on varying views of causality and methods for argumentation. This explanation will usually be intended to promote greater understanding or insights by others into the phenomena of interest;

(3) Prediction: The theory states what will happen in the future if certain preconditions hold. The degree of certainty in the prediction is expected to be only approximate or probabilistic in IS;

(4) Prescription: A special case of prediction exists where the theory provides a description of the method or structure or both for the construction of an artifact (akin to a recipe). The provision of the recipe implies that the recipe, if acted upon, will cause an artifact of a certain type to come into being.

The combinations of the four goals lead to five classifications of IS theory, following Gregor (2006):

(1) Type I (analysis): The theory does not extend beyond analysis and description. No causal relationships among phenomena are specified and no predictions are made.

(2) Type II. (explanation): The theory provides explanations but does not aim to predict with any precision. There are no testable propositions.

(3) Type III. (prediction): The theory provides predictions and has testable propositions but does not have well-developed justificatory causal explanations.

(4) Type IV. (explanation and prediction): The theory provides predictions and has both testable propositions and causal explanations.

(5) Type V. (design and action): The theory gives explicit prescriptions (e.g. methods, techniques, principles of form and function) for constructing an artifact.

From this preliminary taxonomy, Weber (2012) performed an evaluation of the five types and states that a real theory in IS is best aligned with the Type IV, a theory for both explanation and prediction.

An IS theory must have both explanatory and predictive power, in order to be considered a theory *per se*. Although a proposal or model can have these characteristics, it is only considered a theory based on a series of criteria, for example, the relevance of the phenomena being portrayed and described. Another important criterion is the level of detail and how rigorous is the specification of its constructs.

Therefore, we state that a BPM theory that has sufficient expressive power to study the phenomena of KiPs is lacking in the literature and can be detailed by the same criteria as an IS theory. We also argue that our proposal has both explanatory and predictive power within its established boundaries, as demonstrated in the next subsections.

3.2 Background theories

To precisely conceptualize and analyze a KiP, it is necessary to understand the complexity of the cognitive elements involved during its executions, as well as how process participants interact and exchange knowledge. Therefore, our proposal explores the cognitive aspect of the BPM field, focusing on the advances in related domains of research such as philosophy, linguistics and cognitive psychology. The main goal is to propose a descriptive theory of human behavior within a KiP, involving mental states (such as belief, desire and intention) as well as their role in determining human actions.

The proposed theory encompasses concepts from the following works, which provide a theoretically sound basis to study the cognitive perspective. Due to the limitations of a single paper, we will briefly describe its main points of interest:

(1) Searle's theory of intentionality (Searle, 1983) as a theory of the mind that explains individual and collective human behavior and is centered on the concept of the intentional state as a form of describing cognitive elements such as beliefs, desires and intentions, enabling a detailed analysis of them; It enables the depiction of the mental states (beliefs, desires and intentions) involved in a KiP and how they affect the process.

(2) The speech act theory (Searle and Vanderveken, 1985) as a method for analyzing the expression of the intentional states of KiP agents, making it feasible to approach interactions between process participants as the exchange of speech acts expressing their intentional states; This theory reframes sentences spoken or written during an interaction into as actions performed by a speaker, expressing his/her intentional states, allowing for the knowledge exchange to be analyzed as a process itself.

(3) The theory of collective intentionality (Searle, 1995) to deepen the understanding of interactions between KiP participants and shared externalizations of intentional states at the social sphere of the process and the organization, such as the instantiation and description of important BPM concepts such as social roles, social objects, social commitments and claims; It brings up important BPM elements such as process roles and resources and describes their creation in terms of "status-functions", functions imposed upon objects and agents, supported by their intentional states and speech acts.

(4) Stalnaker's theory of common ground (CG) (Stalnaker, 2002) as a baseline to explain the dynamics of interactions between participants, especially the ones that trigger the execution of actions by process participants during process instance execution; This important concept from linguistics defines a "shared set of presupposed beliefs" between participants during an interaction, enabling the theory to study how consensus and different viewpoints are accepted or dismissed by a group and its effects on decision-making activities and even the flow of process instances.

The theory proposed in this paper was deducted from the foundational theories cited in the paper, first by a thorough process of reviewing the concepts of each different theory and how they would be applied to solve an "understanding gap" on the nature of KiPs and then conceptualizing specific relations between the "pure" concepts from each theory and its counterpart in the scope of BPM and KiP (e.g.: A speech act in Linguistics has several ramifications, such as indirect speech acts, illocutionary force indicating devices, etc.) and then defining precisely the boundaries of each concept in our theory.

4. A cognitive BPM theory for knowledge-intensive processes

Our research explores the paradigm "Cognitive BPM" (Hull and Motahari-Nezhad, 2016), which essentially stands for the application of cognitive computing technologies to

the contexts and aspects of the BPM ecosystem. In particular, our goal is to develop an IS theory that provides the foundations of a cognitive paradigm for KiPs. In this direction, we argue that the theories listed in the previous section provide a complementary, adequate and precise conceptual basis for understanding, representing and analyzing the dynamic nature of Interactions and their effects within a KiP.

The first step was to investigate if and how Intentional states drive the actions of agents involved in the execution of a KiP in a social environment, more specifically within a business organization.

In the next subsections, we will describe the concepts of each perspective of the proposed theory, the first five describe a generic business process in terms of the cognitive KiP theory and the last subsection goes into detail on how a KiP can be explained and predicted by cognitive KiP. The first perspective addresses the base BPM concepts such as participants, process activities and goals. The second perspective explores the cognitive aspect of the participants, the role of their mental states, the role of communicative interactions and the exchange of knowledge; the third perspective describes the interactions between process participants and how they can be described using the speech act as a base concept; the fourth perspective points out how collective intentionality and status-functions can be used to depict process roles and social objects; the fifth perspective how CG serves as a model to analyze consensus, shared awareness and the collective acceptance of status-functions; and finally, the sixth and last perspective wraps the theory, describing the specific concepts to be applied to describe a KiP.

4.1 On business process and goals

We begin by defining the basic concepts involved in a business process: event, activity, goal, participant and resource. In this section, cognitive KiP concepts are described using a simple notation as follows: a text box for each concept, with its UFO's stereotype between parenthesis and the full text of the definition below. The concepts present in the figures are defined based on stereotypes from the unified foundational ontology (UFO). Due to space limitations, they are not fully described here, but a thorough description can be found at Guizzardi *et al.* (2008, 2013) and Guizzardi (2005).

We adopt the definition of event from Guizzardi *et al.* (2008), "Events are possible transformations from a portion of reality to another, i.e. they may change reality by changing the state of affairs from one (pre-state) situation to a (post-state) situation. They are ontologically dependent entities in the sense that they existentially depend on their participants in order to exist". Based on this initial definition, we can define an event in a process by two types: unintentional or an activity, the former occurring without the involvement of the intentional action of agents and the latter explicitly caused by the intentional action of an agent within the scope of our process. Both can be optionally composed by other events (their components being either activities or unintentional events).

An activity, being intentional, always has a goal, defined as "an anticipatory internal representation of a state of the world that has the potential for and the function of (eventually) constraining/governing the behavior of an agent towards its realization" (Castelfranchi and Paglieri, 2007). An activity can be further specified as "an intentional Event (i.e. involving agentive function of a Participant) that has the purpose of satisfying a Goal". The activities that compose a higher-level activity also have goals and the goal of the higher-level activity is dependent on the fulfillment of the lower-level goals to be satisfied. This creates a relationship of dependency between goals that extends to the activities involved, as goals, likewise can be conditioned (dependent on the fulfillment of other goals for its own fulfillment) or executive ("atomic" i.e. without the dependency toward any other goals. These initial concepts enable our proposal to describe typical business process notation elements (e.g. BPMN) such as value-added chains, processes and sub-processes.

A participant is defined as an agent that contributes to the process toward the fulfillment of its goals. The process, being a high-level activity, usually has the participant involved in at least one of its composing activities, and contributing to fulfilling one or more of its goals. In our proposal, a participant can be individual (a person) or collective (a group of persons or a social entity such as an enterprise or an institution). The collective participant definition adopts a similar viewpoint as Ludwig (2017), considering large scale groups and institutions as a type of group that might be said genuinely to have a mind of its own, due to their complexity, hierarchical structure and differentiation of roles that may be successively filled by different individuals along the time. Finally, a resource is an object (usually physical, data or digital objects) that is created, used, modified, or terminated by an activity. They are as important as the participants, since each activity depends on the resources needed for its execution. Figure 1 describes the relationships between the participants and activities.

4.2 On participants and intentional states

Each participant has intentional states, defined as "The mental capacity to refer to possible situations of reality". An intentional state always has a propositional content and a psychological mode. Based on its psychological mode, it can be further classified as belief, desire, intention, or feeling. Intentional states are inherent in each participant. Even if two different participants have intentional states of exactly the same psychological modes (for example, a belief) and propositional contents, they are different, being states from different participants, with distinct mental characteristics of their own. For intentional states to be shared, discussed or pondered, it must be externalized in the form of an externalized intentional state. An intentional state can have many forms of externalization (different ways of expressing its propositional content and psychological modes).

The intentional states described by cognitive KiP are specialized into one of the four specific subtypes:

(1) Belief: A mental representation that is used as a plausible substitute for a certain aspect of reality, and that is supposed to be referentially true, i.e. to provide a description that is assumed to correspond, and used as corresponding, to how things actually are.

(2) Desire: A mental representation of a state of affairs that would be desirable to bring to reality, but one is not necessarily committed to act upon reaching it.

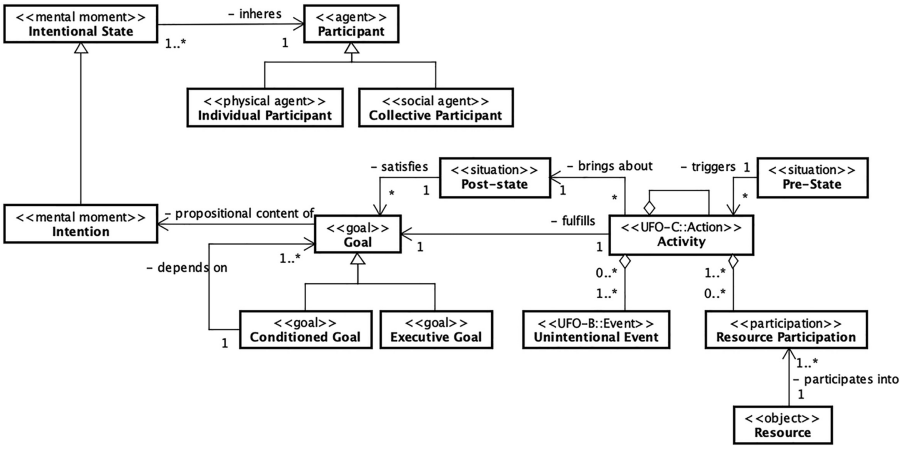

Figure 1.
On business processes and goals

(3) Intention: A mental representation of a self-commitment of a participant toward performing a course of action, in order to bring a desired state of affairs to reality.

(4) Feeling: A mental representation of a sentiment or affect regarding a state-of-affairs or object.

First, we adopt the definition of belief as a "belief state" from Yalcin (2018) as "a set of metaphysically possible worlds, intuitively the worlds 'left open' by what is believed. Thus, propositions are sets of possible worlds, and the propositions an agent believes are those true with respect to all of those worlds the state leaves open". Rather than representing the beliefs of an agent by a single set of possibilities, they are represented by a collection of sets of possibilities. Therefore, agents do not have a single belief state; rather, they have a set of belief states, or "separate systems of belief", the contents of which are each represented by a set of possible worlds. This definition can also be taken as two distinct viewpoints: (1) the map aspect: the state of an agent that represents the world as being a certain way and (2) the steering aspect: the explanation of the agent's actions. Both aspects are critical to the cognitive aspect of a KiP, as they explain not only the perceptions and viewpoints of an agent but the rationale of their actions.

Second, Malle *et al.* (2001) define desires and intentions as representational states, and they both express a pro attitude toward the state of affairs that they represent, frequently propelling the agent to act in such a way as to bring about that state. The main difference between them is that the desire is not related to a following action, while an intention is reached after carefully considering desires, is related to one's own following action and precedes an action.

Finally, there is a close relationship between an action's goals and the action itself, in the sense that the goal is the cause for the action to occur and, sometimes, the action will continue until it satisfies its goal (Dretske, 1988). With both definitions in mind, a relationship between an intention and a goal can also be indirectly stated, since (1) an intention is essentially related to the action it aims to perform, and (2) an action is related to the goal it aims to fulfill.

At this point, we have a clear function for the main three intentional states: beliefs represent the participants' perceptions and the rationale of their action, desires represent possible courses of action still being pondered to be performed and Intentions represent the courses of action that are ready to be acted upon.

For a clear depiction of the dynamics involving intentional states, activities and goals, we adapted the work of Castelfranchi and Paglieri (2007). Usually, a goal exists first as an active goal, a goal that is hypothetical or unable to be realized at the present. After thoughtful consideration and/or the coming of felicitous conditions in reality, the active goal, becomes a chosen goal. A goal is deemed chosen (i.e. pursuable) when there are conditions for its fulfillment by an action and there is a voluntary disposition from one or more participants to fulfill it. Active goals are related to desires and pursuable goals are related to intentions.

The consideration of which goal to be pursued and, consequently, which activity will be performed for its fulfillment is critical for any business process, even more important for a KiP. This can be used to depict AND and XOR gateways, where two separate sets of activities will be either simultaneously performed or exclude the other. The dynamics of the choice of an active goal and its changing into a chosen goal is described by the postulate of cognitive regulation of action (Castelfranchi and Paglieri, 2007): "Each goal of a cognitive agent is necessarily supported and justified by this agent's beliefs (i.e. reasons). Cognitive agents cannot activate, maintain, decide about, prefer, plan for or pursue any goal which is not grounded (implicitly or explicitly) on pertinent beliefs". Two important corollaries can be taken from the postulate: The first, "specificity", points out that for a goal to be chosen for pursuit (i.e. become an intention), it must be supported by specific beliefs that enable its selection.

The second, "dependency", specifies that if a goal remains unpursued (i.e. still a desire and not an intention), there are beliefs, either taking support away from this goal and/or beliefs that previously supported the goal that were invalidated by specific state of affairs.

Taking into consideration the relationship between desires and intentions with goals and activities, as well as beliefs as the rationale for the pursuit of goals and course of actions, we can now explain how a goal is chosen to be pursued and an activity is set for its fulfillment, in three ordered steps:

(1) A participant's desire specifies an active goal to be considered for pursuit by the group.

(2) The active goal is pondered, individually or collectively, and a subset of beliefs either enables the goal to become a pursuable goal OR a subset of beliefs disables the goal, so that it remains an active goal.

(3) The pursuable goal is pondered and discussed again in terms of practical means to achieve it. It can be decomposed in another set of smaller scope goals (sub-goals), conditioned to the broader goal. This cognitive process can iterate until goals that are agreed upon by the group and so close to an action that they become the propositional content of an activity and an intention of the participant that is going to execute the activity.

From the set of results of the decision process, the discarded alternatives remain as active goals and only the chosen alternative becomes the propositional content of the chosen goal. This fact means that the discarded "courses of action" can also be pondered again, at a different time or process instance, for example. It is important to notice here that the dependence relationship between low-level goals and higher-level goals (and their specific associated activity) has interesting consequences. They are related to goals to be achieved and the dynamics is described in terms of beliefs, desires and intentions from stakeholders and process executors, both being either individual or collective participants, as described above. Figure 2 described the relationships between goals, intentional states and participants.

4.3 On interactions and speech acts

The basic mechanism of choosing a goal, activity or process rarely occurs as a decision of a single participant. Given that a KiP is essentially a social concept conducted within an organization by a subset of its agents, it is crucial to extend the discussion on intentionality to address not only its definition from an individual perspective, but also its collective counterpart, in social reality. Thus, intentionality is not only restricted to an individual, but there is also the concept of collective intentionality. In such environments, it is reasonable to assume that there is the need of the organization and commitment by the agents to establish communication among agents who co-participate in the same execution of a KiP. We can reasonably argue that the agents are (or should be) willing to externalize at least a subset of their intentional states, in order to achieve the goals of the activity they execute and more. An intentional state may be externalized by means of several representations, ranging from graphical representations (such as drawings or diagrams) to textual representations (captured from both written texts or spoken language) and in different media formats. A speech act expresses an intentional state, carrying its representation, usually through text or speech. It constitutes the building block for all the mental cognitive concepts we adopt in our proposal, as, unlike the intentional state, it can be performed and shared as text, voice or any other representations. By speaking to an audience or sending a message to a group, for example, process participants can coordinate actions, share knowledge, etc. Also, social networks are a popular infrastructure in which these externalizations take place in a very large number of ways and are frequently used by agents

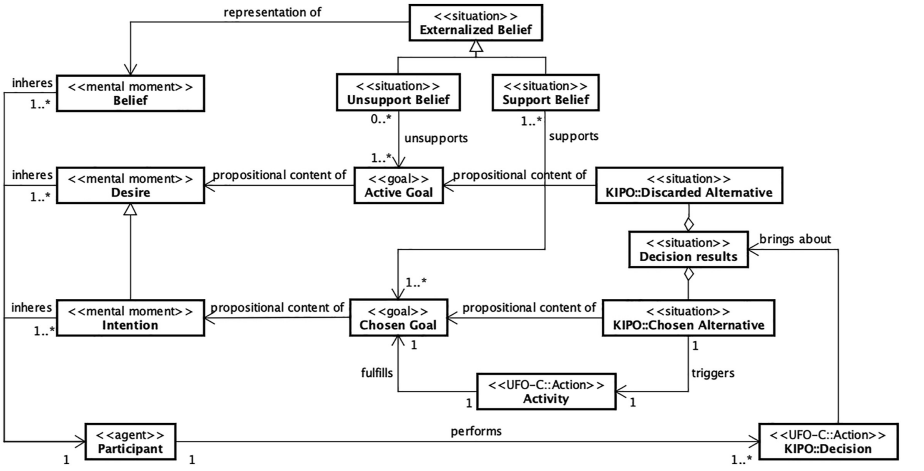

Figure 2.
On participants and intentional states

involved in process executions as a means for exchanging knowledge among them (Chatterjee *et al.*, 2021b), especially between different hierarchical levels of an organization, for example from stakeholders to the lower levels (Chatterjee *et al.*, 2020).

A KiP is directly impacted by the knowledge exchange that occurs during the interactions among participants, in the form of conversations using several platforms. The conversation among participants during the execution of an activity, called here a communicative interaction, may be represented in the form of an exchange of speech acts expressed by the speakers that are involved in the conversation, modifies the intermediate and final results of the tasks performed, because the knowledge exchanged brings about situations that are perceived by the participants. Different kinds of knowledge exchange occur, mainly the shared learning of different viewpoints, that can include complex issues such as power imbalances and trust (Jugdev and Mathur, 2013).

Due to the complexity of the topic, we simply define a conversation as an ordered sequence of speech acts, between different speakers and hearers. Each speech act in the sequence creates a limited set of possible replies, in the sense of limited sets of possible speech acts to be performed as the next step of the sequence of speech acts during the conversation. For example, social networks are a popular infrastructure in which these externalizations take place in a very large number of ways and are frequently used by agents involved in process executions as a means for exchanging knowledge among them.

According to Searle and Vanderveken's taxonomy of speech acts, we can compare each speech act type with typical forms of interactions during a process, as illustrated in Table 1.

The main consequence of the modeling of an interaction using our proposal is that each sentence spoken or written during an interaction becomes an action, performed by the speaker (or sender of the message) to the hearer/audience (or receiver of the message). In a sense, the interaction becomes a process itself, composed of activities in a flow separated by different lanes such as a typical BPMN model. All these concepts are structured as in the conceptual model of Figure 3.

4.4 On process roles and social objects

A business process is not only based on activities. An important part of its logic lies in social concepts that describe the contextual elements and organizational reality within which the

process occurs. Two main categories of social concepts are described at cognitive KiP: The first are process roles, describing the logic of the different roles performed by participants. The second are social objects, that represent how resources take different characteristics that are not intrinsic to them. All of them are created by collective intentionality by status-functions imposed upon them, usually by the performance of declarative speech acts.

At the sphere of a business process, process roles can translate into positions such as manager, analyst, technician, team leader and others, all defined as a series of social commitments (responsibilities and duties) and social claims (institutional power over decisions, people and objects). Participants that take over the specified roles are described as the recipients of the imposition of their respective functions, with responsibilities and powers, described by speech acts that compose work contracts, internal regulations and organizational chart descriptions, for example. Note that neither social commitments nor social claims specify activities, but rather they also constitute a high-level social abstraction that the participants in a business process enact (Singh, 1999), together with the discharge conditions that must hold when a social commitment is fulfilled.

There are interesting traits of the dynamics of social commitments between agents (Dalpiaz *et al.*, 2015): An agent's social commitments typically constrain it to act in accordance with them. A social commitment is discharged when a desired discharge condition is satisfied. The condition can be an event, or a condition relative to one of the agents, typically involving other contextual elements.

A similar operation of status-functions occurs in the form of social objects. Social objects are abstract objects representing the attribution of capacities and functions that were not previously existent to the physical or data object they are imposed upon. For example, a written and signed piece of paper can be imposed as a function of a contract, as a spreadsheet file can become a yearly budget document, or a piece of plastic can become a badge that describes the role of its wearer. The main distinction between process roles and social objects is that the former is only imposed to participants and the latter only to business objects and resources.

Speech Act type	Forms of interaction
Assertive (Belief)	Perceptions, viewpoints, explanations, descriptions, predictions, discussions, replies to questions
Directive (Desire)	Requests for information, requests for action, orders, suggestions, questions, requirements, prohibitions and advices
Commissive (Intention)	Promises, acknowledgments to request, offers
Declarative (Social Sphere)	Declarations, nominations, status updates on activity or request

Table 1.
Speech act types and interaction examples

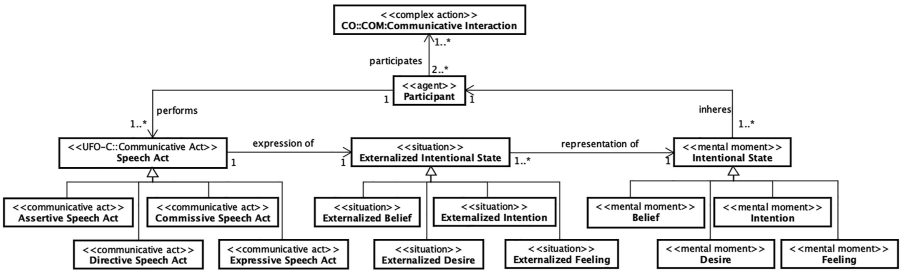

Figure 3.
On interactions and speech acts

In our proposal, process roles and social objects are either (1) pre-existent, already created and recognized by the group, before the execution of the process instance or (2) created in an *ad-hoc* manner during the execution of the process. The pre-existent social concepts are usually described in documents such as contracts, guidelines and rules that form a special kind of social object called a normative description, recognized by all participants. The *ad-hoc* social concepts are created by the exchange of speech acts during the interactions that occur during the process execution.

Our proposal states that the social concepts depict the organizational and social sphere of the process and they have, at its core, the collective acceptance of the status-functions by the groups involved at the process. The dynamics of acceptance and consensus is closely related to the idea of sharing beliefs through speech acts in the form of a CG context. Figure 4 presents a conceptual model for these concepts and corresponding relationships.

4.5 On status-functions and common ground

A key concept for the cognitive perspective of the proposed theory is the CG, specified as the set of externalized beliefs that represent the shared perception, awareness and consensus of the participants during interactions. It is also the key element to describe the collective acceptance of status-functions such as process roles and business objects. We now define the dynamics of CG by reflecting on how the interchange of representations of intentional states between agents and their reactions toward events affect the process. Since each belief is inherent to a specific agent and the CG is the set of presupposed beliefs to be shared by all agents involved at a communicative interactions, we take its externalized representation, the externalized belief, as a definition of the basic building block for the CG.

Being a set of belief representations, its initial state is the set of prearranged presuppositions, what is believed and presupposed as given by all participants. The process roles of the participants of the interactions, the shared awareness of the process, the organizational sphere and the goals of the processes and activities being performed are examples of the initial contents of a CG.

Each speech act performed within an interaction or event perceived by all participants modifies the CG. This is the accommodation process, triggered by a speech act or manifest event. A conversation as a set of exchanged speech acts enables us to apply the concept of CG to every interaction. In a KiP, the CG enables the description of a communicative interaction not only as

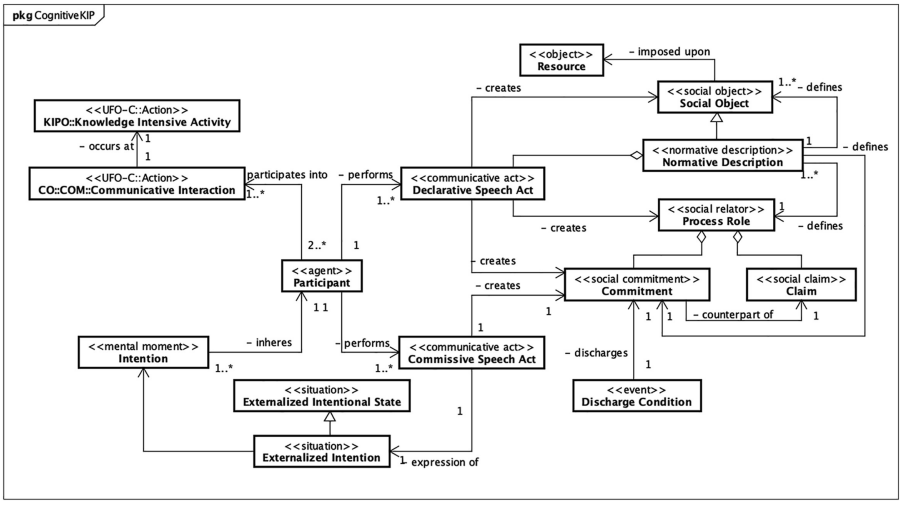

Figure 4.
On process roles and social objects

knowledge (in the form of propositional content) being exchanged, but also as a representation of the shared viewpoints, defined as a set of shared presuppositions composed of externalized beliefs between the participants, that will influence (as described in the previous subsection) the selection and/or rejection of shared goals during a process execution. This specific consequence of the application of Castelfranchi and Paglieri's model , combined with the notion of CG, enables the tackling of a series of issues that form the main hindrances of KiP analysis and modeling. In a KiP scenario, the CG can be used to describe the set of common knowledge presupposed by all participants during the execution of an activity (as well as during the decision-making task of "which activity will be the next step at the process flow"), forming the rationale of the actions involved. It is interesting to point out that the concept of CG enables us to explore, within a KiP, not only the agreed mutual perceptions between participants, but also shared values, strategies and goals, especially with the prevalent knowledge sharing between social media platforms nowadays (Chatterjee *et al.*, 2020). All these concepts are structured as illustrated in the conceptual model depicted in Figure 5.

4.6 On knowledge-intensive processes

The previous subsections addressed the typical business process elements, such as activities, roles and resources. We can now describe the concepts that are unique to a KiP. We adopt the definition of KiP from Vaculin *et al.* (2011): "A process whose conduction and execution are heavily dependent on knowledge workers performing various interconnected knowledge-intensive, decision making tasks". Therefore, we have two main traits, knowledge-intensity and decision making aspect, as parts of one or more activities of this process. This type of activity is called a KiA that is defined as an activity that necessarily has interaction between the participants involved and one or more decision making tasks to be performed as part of its execution.

A clear example would be a KiP that has a goal of defining milestones for next year's strategic planning. It is clearly a KiP as the knowledge-intensity is characterized by the necessary interaction between participants and the decisions to be performed ("Should this year's expected ROI be kept the same as in last year's strategic plan?", "Which department will have a larger project budget?", etc.). Another interesting part of this scenario is that each act of deciding is an activity with its own goals, which are also executive goals that form "smaller steps" of the bigger

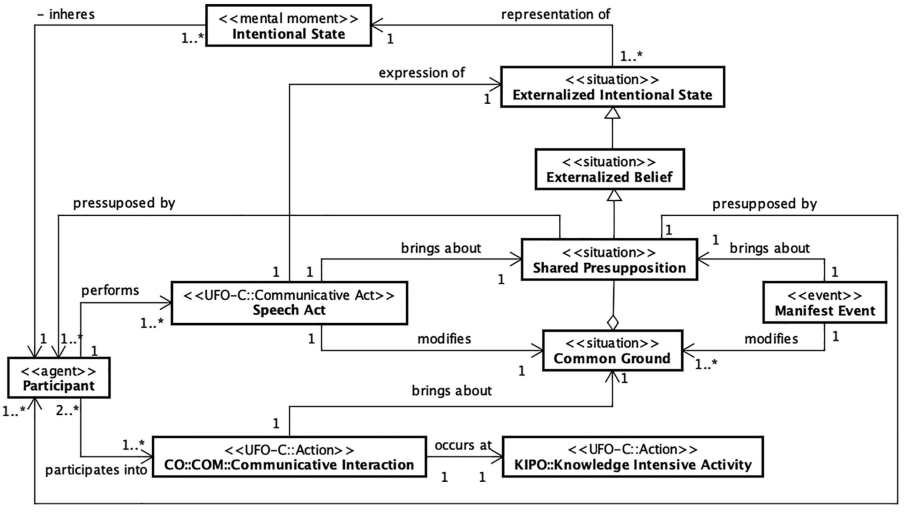

Figure 5.
On status-functions and common ground

goal of the KiP itself. It is also interesting to notice that each decision-making activity can also be a KiA, as it can involve lower-level decisions and interactions as sub-processes.

Being a human-centric endeavor, focused on the knowledge exchange between them, the participants involved in a KiP have two possible roles: An innovation agent is a participant that contributes to a KiA with innovation and creativity, by solely participating in the interactions related to one or more KiAs. In contrast, an impact agent is a participant that intentionally executes a KiA and, optionally, also participates at the interaction that occurs during the KiA he executes. This distinction is critical for a KiP as it creates a new form of participation, by contributing with knowledge instead of executing an activity. The contribution it performs can take two main forms: (1) The participation in the execution of an activity or (2) The participation in an interaction with other participants, sharing knowledge. These forms are not mutually exclusive because it is possible (and often common) that the participant executes an activity and participates in an interaction with the group at the same time. Figure 6 structured these concepts in a conceptual model.

5. Case study discussion

For the discussion of our proposal, we have chosen the open-source software development domain, being a field that necessarily involves different participants exchanging knowledge and a great variability during the execution of each process instance. The level of knowledge exchange, usually between geographically distant process participants as well as the rapidly changing requirements and goals form a promising scenario for the application of cognitive KiP.

One of the most popular open source software (OSS) platforms is the GitHub [1], a web-based, social software development environment that provides source code management, version control, issue tracking and other features. GitHub allows users to set up a public repository that anyone can fork and use for their own code and/or to contribute changes to the code. Three important concepts are critical for its usage: (1) Creating a "Fork", an operation where the user creates a clone or copy of a repository, usually as a starting point to contributing back to the original project through a pull request; (2) a Commit, changes of an individual file or set of files/folders and (3) a Pull request, an operation on which code from one developer (that previously performed a fork, modified the code with different commits and now contains a different version of the software) is offered to be merged with the original software code of the repository, frequently with bug fixes, improvements, new functionalities, etc.

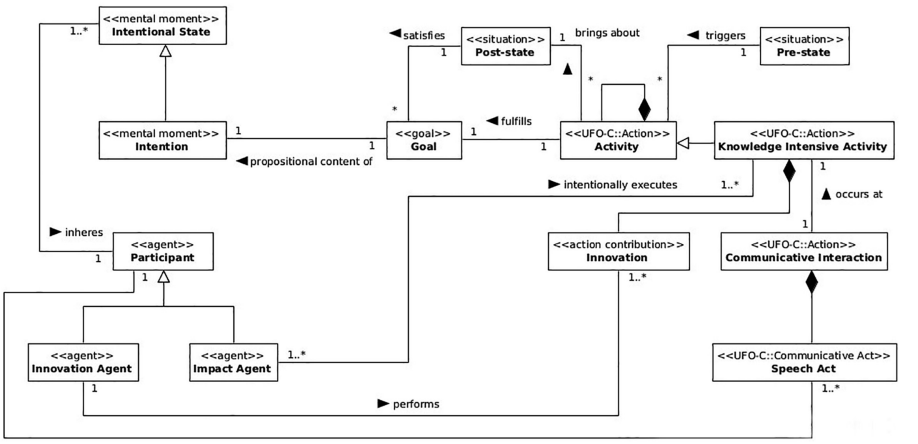

Figure 6.
On knowledge-intensive processes

Among the typical OSS business processes involved in a Github repository, one of the most representative is the "Contribute Code to Repository" process [2], involving the software development tasks, as well as sharing the code, comments, viewpoints, bug reports and other forms of knowledge between developers, users and members of the OSS community. The goal to be reached in this process is to contribute successfully with a code for a specific version of an OSS.

This process can be described by cognitive KiP as the higher-level activity of the model, triggered by the act of creating a new contribution and composed by several other activities such as "Create a branch" and "Add commits". This activity has a conditioned goal "Contribution successfully evaluated" associated with it and is also dependent on all the goals of its lower-level component activities. We argue that the process is characterized as a KiP because there is a clear necessity of communicating the participants' code contributions and receiving feedback (in the form of cognitive aspects such as beliefs, desires and intentions) from contributors and users of the software, through interactions and message exchanges. Besides, there are critical decision-making tasks involved in the process, such as accepting or rejecting a specific functionality to a future version of the software, or deciding whether or not a contribution is sufficiently stable to be merged with the software's repository.

Finally, there are specific business rules, either pre-defined (e.g. the division of roles and responsibilities between developers, especially in mature projects such as the Linux Kernel) or *ad-hoc* or implicit (rules of conduct, code conventions, etc.). All these characteristics enable the description of "Contribute Code to Repository" as a KiP, as depicted in Figure 7.

We begin by depicting the process as a whole as a higher-level KiA (i.e. the KiP in question) that is going to be analyzed. We have depicted our KiP as an activity called "contribute code to project ", being composed by the KiP's sub-processes and activities, all of them modeled as a cognitive KiP activity. Three distinct process roles are involved in the process: (1) Contributor, an independent developer that is willing to improve the code and contribute to the repository; (2) Maintainer, a team member of the repository's project responsible to its version management and code consistency and (3) User, a person with access to the repository that can give opinions, user experience reports and viewpoints about the software being developed. The first two are described by cognitive KiP as impact agents, performing tasks and interacting with other participants, while the users are innovation agents, expressing viewpoints and opinions during the pull request discussion.

The contributor is responsible for creating a new branch (i.e. a parallel version of a repository) and modifying the code, either adding new lines of codes and/or editing or removing other parts (defined as a commit or revision at GitHub, being an individual change to a file or set of files) at his local copy of the repository's code and then submitting his modifications to the repository as

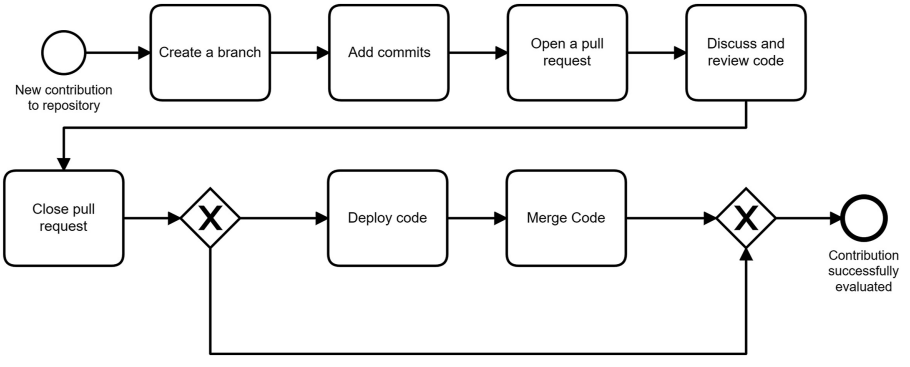

Figure 7. "Contribute code to repository" as a knowledge-intensive process

a pull request. Then, the maintainer will review and discuss the code with the contributor and other users willing to participate, until a decision is made by the maintainer(s) on whether the contributor's code is relevant and ready to be merged with the repository's code or not. A series of new verifications and tests will be executed until the code is merged and deployed as a new version of the code at the remote repository.

We argue that the activity "Discuss and review code" involves knowledge-intensity, in the form of interactions between participants and decision-making tasks being performed, that directly affect each process instance and its flow. Therefore, it can be classified as KiA. Figure 8 depicts the activity as an *ad-hoc* activity, according to the BPMN 2.0 notation and the rest of the process flow in detail.

Four activities compose the KiA, representative of the evaluation, review and discussion of the contribution and associated program code, as well as the critical decision performed by the repository's maintainers, on whether accepting the contribution for the repository or rejecting it. Being an *ad-hoc* activity, each subtask can be executed once, multiple times or not at all during the process execution. Also, there is no specified sequence, in contrast to the rest of the flow, that follows the closing of the pull request and the XOR gateway specific to the results of the maintainers' decision.

In order to illustrate the process in a real case scenario, we have chosen the audacity project Github repository for our study. The audacity project is a free, open source, cross-platform audio software and multi-track audio editor and recorder for Windows, Mac OS X, GNU/Linux and other operating systems and developed by a group of volunteers. We extracted a dataset composed of 231 closed pull requests (KiP instances that were finished by the time of extraction) from March, 2015 until March, 2018, with a total of 503 messages. We have selected a single instance from the data, representative of a typical pull request interaction involving a maintainer, a contributor and two users. The real information about the people involved is omitted and anonymized (see Table 2).

Before looking into the message text, we must perceive the social aspects that exist prior to the interaction. The Github platform already performs most of the status changing of data objects involved (such as code, files, bug issues, user status at a repository) as well as process roles such as maintainer and collaborator. All of these elements form the initial CG of the interaction, presupposed to be known by all participants. *Ad-hoc* process roles and social objects are not forbidden but are discouraged by the clear interface. Due to the same reason, the performing of declarative speech acts is not common during a pull request interaction.

The interaction begins after maintainer A reads the pull request description, analyzes its commits and, in message #1, performs two directive speech acts ("Please explain which build failures you are fixing with each of these commits" and "Please explain more about how the wxFileNameWrapper kludge causes a crash"). The contributor replies in message #2 and #3, first with a commissive speech act ("Ok, I'll add error messages to the commit messages")

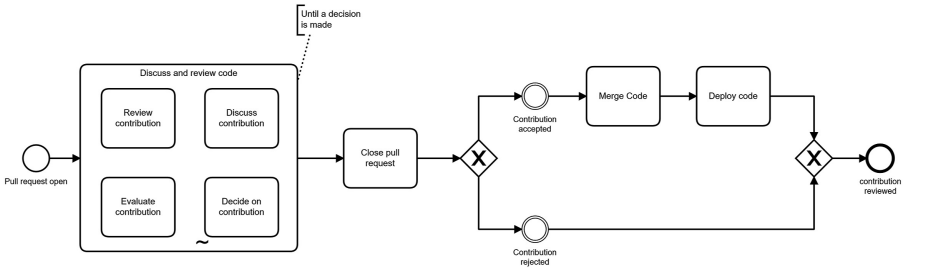

Figure 8.
"Discuss and review code" as a knowledge-intensive activity

Msg ID	User_login	Body
1	Maintainer A	[d]Please explain which build failures you are fixing with **each of these commits**.[d] [d]Please explain more about how the wxFileNameWrapper kludge causes a crash.[d] [d]I would like to keep that kludge for the compilers that let us get away with it[d]
2	Contributor	[c]Ok, I'll add **error messages** to the **commit messages**[c]
3	Contributor	[a]In any case, the wxFileNameWrapper kludge causes severe double-free bugs. Valgrind is screaming loudly, and sometimes, even glibc notices heap corruption.[a] [c]I'll post details[c]
4	Maintainer A	[a]According to the **travis output** linked herein, **commit X** is causing build failures[a]
5	Contributor	[a]Apparently, Travis builds with a very old FFmpeg version which doesn't have the "const" yet.[a] [a]Without **that commit**, build fails here (FFmpeg 3.0 and 2.8.6)[a]
6	Maintainer A	[a]I cherry-picked the fix for **Track.cpp**[a]
7	Contributor	[a]So I'm using my system FFmpeg, because I hate projects which ship (outdated) copies of other libraries.[a] [a]The FFmpeg version in "lib-src/ffmpeg/" is 2.2.2.[a] [d]Would you agree to update those headers again?[d] [a](I'd remove them completely, but that's just my opinion.)[a]
8	Contributor	[a]I have added a valgrind log to the wxFileNameWrapper commit.[a] [d]Do you need to know anything else?[d]
9	Maintainer A	[a]I am not qualified to make the decision about FFmpeg.[a] [d]I suggest you ask the question also at **the audacity list**.[d] [d]Can you figure out conditional compilation that could make **the build** work with either version?[d]
10	Contributor	[a]Ok, wrapped in **preprocessor version checks**[a]
11	Maintainer A	[d]Please see **my commit**,[d] [a]it implements a different cheat for wxFileNameWrapper which might not crash you[a]
12	Contributor	[a]This will fall apart as well any day.[a] [a]I wouldn't do that.[a] [a]I understand your desire for move operations, but in this case I'd wait for WX to support it.[a] [a]Everything else is a kludge that's just waiting to crash (or corrupt data randomly)[a]
13	Maintainer A	[a]Its ugly either way,[a] [d]but humor me and see if it compiles and runs and lets you load and save **a project** without apparent trouble[d]
14	Contributor	[a]Compiles and doesn't crash, no valgrind warning[a]
15	Maintainer A	[a]Good,[a] [a]I enabled the less evil swap function.[a] [a]It should be safe so long as we do not change version of wxWidgets.[a] [a]**This discussion page** is still telling me you did not satisfy **the Travis build for FFmpeg functions**[a]
16	Contributor	[a]Thats because your FFmpeg version numbers are inconsistent![a] [a]In upstream FFmpeg, the "const" was added in commit https://github.com/FFmpeg/FFmpeg/commit/ec4f04da1 and "version.h" said: "#define LIBAVFORMAT_VERSION_MAJOR 55#define LIBAVFORMAT_VERSION_MINOR 20#define LIBAVFORMAT_VERSION_MICRO 0" Now your "version.h" without the "const" says: "#define LIBAVFORMAT_VERSION_MAJOR 55#define LIBAVFORMAT_VERSION_MINOR 33#define LIBAVFORMAT_VERSION_MICRO 100" . . . which is the more recent version number set by **commit** https://github.com/FFmpeg/FFmpeg/commit/db3c9701f46d20fd7e94c3222cf4fd4524a16414[a] [a]The real problem is that FFmpeg applied those two changes in different branches, and your copy is a newer branch but without the "const" change.[a] [a]So now I've changed the minimum version expected for "const" to 55.33.101, one more than **Audacity's FFmpeg copy**.[a] [a]I hope this covers all relevant versions[a]
17	User A	[d]What needs to be done to get **the FFmpeg patch** through?[d]
18	User B	[a]For a start, **the commit** has conflicts.[a] [a]And when that is solved, we have to be sure that the currently recommended FFmpeg 2.2.3 maximum is still supported (so e.g. the recommended Windows/Mac FFmpeg downloads at http://manual.audacityteam.org/man/faq_installation_and_plug_ins.html#ffdown still work).[a] [a]Or upgrade Audacity to support later FFmpeg/libav (what range of versions?) Or migrate to gstreamer and use their FFmpeg support rather than hardcoding our own.[a] [a]The last two are major undertakings but I am not an expert in this.[a] [a]I doubt we will want to make any changes now before **2.1.3 release**[a] [d]but I suggest taking Maintainer A's recommendation to ask on **the -devel list**[d]
19	User A	[a]I was hoping to give Contributor a chance to do this,[a] [c]if he doesn't I will submit a new pull request and this one can be closed.[c] [a]I cherry picked **the FFmpeg commit** only, and it doesn't break **the current master branch**. I don't know about Travis yet.[a] [a]I don't expect any change in behavior (or supported versions) as a result.[a] [a]FWICT this patch is about fixing a build failure since the function prototypes changed in the new headers.[a] [a]But since they are only adding a const where there wasn't one, that won't break anything (const isn't part of the C calling conventions AFAIK).[a] [a]So functionally, nothing has changed, the same symbols are still loaded from the libraries.[a] [a]This can all be verified of course without much effort[a]

(*continued*)

Table 2.
Audacity pull request interaction messages excerpt

Msg ID	User_login	Body
20	Contributor	[a]As of yet, I have no feedback whether my changes will be merged.[a] [a]I don't want to waste time on something that will go to the trash can.[a] [a]When I submitted **this PR**, there was no conflict - **the Audacity project** ignored **my PR** for so long, and **after** that merged changed which conflicted with **this PR.**[a] [c]Yes, I will do you the favor and resolve the conflicts - if you want me to.[c] [a]About the "const": the function pointers are not compatible if the constness of pointer targets is not the same.[a] [a]This has nothing to do with calling conventions - calling conventions describe how calls are made on the machine language level.[a] [a]Thus, calling conventions have little to do with the C language, they're a lower level, and semantic API declarations like constness do not matter on that level[a]
21	Maintainer A	[a]As I said to Contributor before, I do not consider myself competent to decide which version of the library should be used in our **released binaries for Mac and Windows**.[a] [a]However I reexamined **commit 41de5b385b2ca721a71871fcdb00ec3fa4441b5b** and I saw that all changes are conditionally compiled, so that it leaves our source code compatible with use of either version of the library.[a] [a]Therefore I have cherry-picked and commited that.[a] [a]Alone, it was unconflicting.[a] [a]**The other commits** are not relevant to upgrading the library version. Updates of Makefile.in are periodically done by other people, and simply commit the results of an automated tool, which doesn't demand real programming effort.[a] [a]And from discussions earlier with Contributor, I found other changes to wxFileNameWrapper that do not give up the performance improvements intended, while also fixing the complaints from valgrind.[a] [c]Therefore I do not intend to take those commits.[c] [a]**The changes to .gitignore** give developers some convenience while not affecting the build at all.[a] [a]No development really depends on them.[a] [D]Therefore I intend to close **the commit request**, having selected the one most important part[D]
22	Maintainer A	... [a]And the Travis build has succeeded[a]

Note(s): Each speech act has been annotated using a bold superscript letter in the beginning and in the end of each sentence (a = assertive, d = directive, c = commissive, D = declarative) and the social objects that are not so general (such as repository, platform, software library, etc.) since they refer to a specific object within the process' context, are highlighted in bold letters

Table 2.

describing his intention of adding error messages to the commit messages of his code and then, with a series of assertive speech acts to answer the questions of maintainer A.

The interaction proceeds with a number of speech acts being exchanged, but with a higher frequency of assertive speech acts, depicting different viewpoints about the code (e.g.: "For a start, the commit has conflicts" in message #18). Even beliefs about the specific skills necessary for the code's testing are present ("As I said to Contributor before, I do not consider myself competent to decide which version of the library should be used in our released binaries for Mac and Windows" in message #9).

Although the CG is modified by each speech act exchanged, regardless of its type, the assertive speech acts exchanged form the main part of the exchange of viewpoints, status reports, opinions and assessments on whether the pull request will be accepted or refused on the main project's repository.

At messages #17-#19, two users of the repositories participate in the interaction, sharing their viewpoints. Interestingly enough, one of the conditions is the opening of a new pull request, a new process instance of "Contribute Code to a Repository" to the resolution of a problem with the present instance. This highlights the critical nature of knowledge-intensity and interaction during a KiP, not only for a particular instance, but for several instances at the same time.

Finally, the last message involves maintainer a performing a commissive speech.

Act involving the closing of the commit request ("Therefore I intend to close the commit request, having selected the one most important part"). Alternatively, the actor could have used a declarative speech act to do so, but expressed his intention instead, possibly because the Github system already notifies the participants on status changes of the pull request. In fact, additional dataset information shows that the commit code was merged to the repository about 10 min after the last message and the pull request was closed.

Our aim with this case study was to illustrate how the expressive and predictive power of our proposal works in reality. The scenario presents how a typical BPM model is not able to

fully describe the implications of knowledge-intensity, mainly in the effects of interactions and decision-making tasks, on the process execution. In terms of an IS theory, it shows preliminary evidence about the power of the proposal to explain the phenomena involving a KiP, both as explanatory power (the capability of performing an analysis of knowledge-intensity, its consequences for the instance flow as well as the rationale of decisions) and predictive (capability of predicting the activity flow of the tasks composing ad-hoc activities and the effects of the instance on future instances, for example at Msg #17-#19).

6. Discussion and conclusions

We propose a novel theory for KiPs, defining concepts based on cognitive psychology, philosophy and linguistics in order to enable its understanding by process analysts and other people involved in the BPM lifecycle. We argue that the proposed conceptual model, which is well-founded on a top level ontology, enforces the soundness of its concepts and theory's application on a real scenario, with the different cases, described in the previous sections that would not be grasped by traditional BPM methods, as shown in the paper, suggests that our proposal has sufficient explanatory power and predictive power, in accordance with the current definitions of IS theory.

The current proposal contributes to the existing body of knowledge by filling an important gap in the understanding of KiPs, a very important kind of business process that is prevalent at enterprises nowadays. Moreover, the factor of a human decision during the execution of a business process, how participants plan and choose the activities to be executed and the bigger question of the impact of human cognition as a factor driving a process is a major question in the BPM field–our proposal's main contribution to the existing body of literature.

The proposed theory aims to explain concepts that are critical for the modeling and understanding of a KiP, its activities (either knowledge-intensive or not), its events and the dynamics regarding resources and roles. Although not specifically focused on the performance optimization of KiPs, its concepts would be critical for the evaluation and optimization of the performance during the execution of KiP. A clear example of this lies in the Github case study, as it provides practical real-world examples on how the theory can be applied to explain process behavior, ad hoc activities and instance flows deemed "exotic" (i.e. unexpected) that would apparently deviate from the process model. Any of these examples would bring up bottlenecks and behavior that would hinder the performance of one or more instances of a KiP. The theory also addresses the initial hypothesis (H1) as the intentional states, expressed by speech acts, are clearly depicted as the "driving elements" behind the KiP and the knowledge exchange presented at the pull request message flow. The beliefs contained in the several assertive speech acts express the diversity of knowledge, some of them even containing conflicting views and proposals between participants, describing the role of beliefs as the driving force behind decision making tasks (H1.1) on a KiP. Lastly, the directive commissive speech acts, expressions of the desires and intentions of the participants, form a "map" where we can find the different courses of actions being pondered by all of them and chosen by the repository's maintainers.

Our proposal's boundaries and limitations are fourfold: First, the focus on the human participants and their mental states impose a limit for the analysis of processes with a high degree of automation, although this kind of process can arguably be of a low level of knowledge-intensiveness and, thus, outside the scope of the theory; Second, the intentional states that are not expressed as speech acts can be considered as outside the boundary of the theory. The broad definition of a speech act in cognitive KiP encompasses both spoken and written interactions between participants, as well as synchronous and asynchronous interactions. For example, platform companies involved in transportation services usually have process instance flows that are heavily dependent on driver-passenger interaction, making the understanding of its process incomplete without a full analysis of its contents. This includes most forms of communication supported by digital social platforms, such as posts from social networks, chats and emails, as well

as indirect communication that occurs through books, organizational reports and specifications and other forms of written exchanges, including graphic representations. Third, a number of possible intentional states (and their associated expressions as speech acts) such as feeling are outside of the scope of the theory. Finally, our case scenarios were limited in number and in domain (Github pull requests), requiring more case studies and experiments in order to evaluate and enhance the theory as a whole.

The theory's conceptual framework enables a series of practical implications to be explored in the future, such as the application of natural language processing (NLP) techniques for the automatic or semi-automatic extraction of speech acts from interactions, similarly as process mining does, using process logs and possibly as a complementary technique to tackle the human factor and textual interaction logs for traditional process mining techniques and the possible adaptation of modeling techniques, notations and approaches for analysts to better comprehend the KiP and its inner dynamics, as it remains an open problem at the BPM field.

Future work will include the presentation of a formal ontology to represent the conceptualization presented here in an unambiguous form and serve as a meta-model for applications. Moreover, a practical study, using the recent advances in the fields of NLP and data science will be performed in order to assess the possible applications regarding the automatic classification and extraction of speech acts from process textual descriptions.

Notes

1. https://github.com/
2. https://guides.github.com/introduction/flow

References

Andreeva, T. and Kianto, A. (2011), "Knowledge processes, knowledge-intensity and innovation: a moderated mediation analysis", *Journal of Knowledge Management*, Vol. 15 No. 6, pp. 1016-1034.

Barboza, T., Richetti, P., Baião, F., Santoro, F.M., Gonçalves, J.C., Revoredo, K. and Yeshchenko, A. (2018), "Speech acts featuring decisions in knowledge-intensive processes", *OTM Confederated International Conferences: On the Move to Meaningful Internet Systems*, Springer, Cham, pp. 222-237.

Berniak-Woźny, J. and Szelągowski, M. (2021), "Towards the assessment of business process knowledge intensity–a systematic literature review", *Business Process Management Journal*, Vol. 28 No. 1, pp. 40-61, doi: 10.1108/BPMJ-01-2021-0012.

Brander, S., Hinkelmann, K., Hu, B., Martinz, A., Riss, U.V., Thönssen, B. and Witschel, H.F. (2011), "Refining process models through the analysis of informal work practice", *9th International Conference on Business Process Management*, Vol. 6896, pp. 116-131.

Castelfranchi, C. and Paglieri, F. (2007), "The role of beliefs in goal dynamics: prolegomena to a constructive theory of intentions", *Synthese*, Vol. 155 No. 2, pp. 237-263.

Chatterjee, S., Rana, N.P. and Dwivedi, Y.K. (2020), "Social media as a tool of knowledge sharing in academia: an empirical study using valance, instrumentality and expectancy (VIE) approach", *Journal of Knowledge Management*.

Chatterjee, S., Chaudhuri, R., Thrassou, A. and Vrontis, D. (2021a), "Antecedents and consequences of knowledge hiding: the moderating role of knowledge hiders and knowledge seekers in organizations", *Journal of Business Research*, Vol. 128, pp. 303-313.

Chatterjee, S., Chaudhuri, R., Vrontis, D. and Piccolo, R. (2021b), "Enterprise social network for knowledge sharing in MNCs: examining the role of knowledge contributors and knowledge seekers for cross-country collaboration", *Journal of International Management*, Vol. 27 No. 1, 100827.

Dalpiaz, F., Cardoso, E., Canobbio, G., Giorgini, P. and Mylopoulos, J. (2015), "Social specifications of business processes with Azzurra", *Proceedings of the International Conference on Research Challenges in Information Science*, pp. 7-18, 11.

Di Ciccio, C., Marrella, A. and Russo, A. (2014), "Knowledge-intensive processes: characteristics, requirements and analysis of contemporary approaches", *Journal of Data Semantics*.

Dretske, F. (1988), *Explaining Behavior: Reasons in a World of Causes*, MIT Press.

Falbo, R.A. and Bertollo, G. (2009), "A software process ontology as a common vocabulary about software processes", *International Journal of Business Process Integration and Management*, Vol. 4, pp. 239-250.

Gregor, S. (2006), "The nature of theory in information systems", *MIS Quarterly*, Vol. 30 No. 3, pp. 611-642.

Gronau, N. and Weber, E. (2004), "Management of knowledge intensive business processes", *Business Process Management*, Springer, Heidelberg.

Guizzardi, G. (2005), "Ontological foundations for structural conceptual models", PhD thesis, University of Twente.

Guizzardi, G., Falbo, R.A. and Guizzardi, R.S.S. (2008), "Grounding software domain ontologies in the Unified foundational ontology (UFO): the case of the ODE software process ontology", *Proceedings of the XI Iberoamerican Workshop on Requirements Engineering and Software Environments, (IDEAS 2008)*, pp. 244-251.

Guizzardi, G., Wagner, G., Falbo, A., Guizzardi, R.S.S. and Almeida, J.P.A. (2013), "Towards ontological foundations for the conceptual modeling of events", *32th International Conference on Enterprise Modeling (ER 2013)*, pp. 327-341.

Haarmann, S., Seidel, A. and Weske, M. (2021), "Modeling objectives of knowledge workers", *International Conference on Business Process Management 2021*, Springer, Cham, pp. 337-348.

Hagen, C.R., Ratz, D. and Povalej, R. (2005), "Towards self-organizing knowledge intensive processes", *Journal of Universal Knowledge Management*, Vol. 0 No. 2, pp. 148-169.

Hull, R. and Motahari-Nezhad, H.R. (2016), "Rethinking BPM in a cognitive world: transforming how we learn and perform business processes", in La Rosa, M., Loos, P. and Pastor, O. (Eds), *Business Process Management. BPM 2016. LNCS*, Springer, Vol. 9850, pp. 3-19.

Işik, Ö., Mertens, W. and Van den Bergh, J. (2013), "Practices of knowledge intensive process management: quantitative insights", *Business Process Management Journal*, Vol. 19 No. 3, pp. 515-534.

Jugdev, K. and Mathur, G. (2013), "Bridging situated learning theory to the resource-based view of project management", *International Journal of Managing Projects in Business*, Vol. 6 No. 4, pp. 633-653.

Little, T.A. and Deokar, A.V. (2016), "Understanding knowledge creation in the context of knowledge-intensive business processes", *Journal of Knowledge Management*, Vol. 20 No. 5, pp. 858-879.

Ludwig, K. (2017), "Do corporations have minds of their own?", *Philosophical Psychology*, Vol. 29 No. 8, pp. 1-33.

Malle, B.F., Moses, L.J. and Baldwin, D.A. (2001), *Intentions and Intentionality*, MIT Press.

Rescher, N. (1996), *Process Metaphysics: An Introduction to Process Philosophy*, State University of New York Press.

Sarnikar, S. and Deokar, A. (2017), "A design approach for process-based knowledge management systems", *Journal of Knowledge Management*, Vol. 21 No. 4, pp. 693-717.

Searle, J.R. (1983), *Intentionality: An Essay in the Philosophy of the Mind*, Cambridge University Press.

Searle, J.R. (1995), *The Construction of Social Reality*, The Penguin Press, London.

Searle, J.R. and Vanderveken, D. (1985), *Foundations of Illocutionary Logic*, Cambridge University Press.

Singh, M.P. (1999), "An ontology for commitments in multiagent systems", *Artificial Intelligence and Law*, No. 7, pp. 97-113, doi: 10.1023/A:1008319631231.

Stalnaker, R., (2002), "Common ground", *Linguistics and Philosophy*, Vol. 25 Nos 5-6, p. 70121.

Unger, M., Leopold, H. and Mendling, J. (2015), "How much flexibility is good for knowledge intensive business processes: a study of the effects of informal work practices", *HICSS 2015*, pp. 4990-4999.

Vaculin, R., Hull, R., Heath, T., Cochran, C., Nigam, A. and Sukaviriya, P. (2011), "Declarative business artifact centric modeling of decision and knowledge intensive business processes", *The Fifteenth IEEE International Enterprise Computing Conference (EDOC 2011)*, pp. 151-160.

Weber, R. (2012), "Evaluating and developing theories in the information systems discipline", *Journal of the Association for Information Systems*, Vol. 13 No. 1, pp. 1-30.

Weske, M. (2007), *Business Process Management: Concepts, Languages, Architectures*, 1st ed., Springer-Verlag, Berlin, Heidelberg.

Witschel, H.F., Hu, B., Riss, U.V., Thoenssen, B., Brun, R., Martin, A. and Hinkelmann, K. (2010), "A collaborative approach to maturing process-related knowledge", *Business Process Management*, Berlin, Heidelberg, pp. 343-358.

Yalcin, S. (2018), "Belief as question-sensitive", *Philosophy and Phenomenological Research*, Vol. 97 No. 1, pp. 23-47, doi: 10.1111/phpr.12330.

Corresponding author
João Carlos de Almeida Rodrigues Gonçalves can be contacted at: jcargoncalves@protonmail.com