2017

Conjoint utilization of structured and unstructured information for planning interleaving deliberation in supply chains

Naeem K. Janjua
Edith Cowan University, n.janjua@ecu.edu.au

O. K. Hussain

E. Chang

Syed M. S. Islam
Edith Cowan University, syed.islam@ecu.edu.au

10.1145/3106426.3106545
Originally published as:
Janjua, N. K., Hussain, O. K., Chang, E., & Islam, S. M. S. (2017). Conjoint utilization of structured and unstructured information for planning interleaving deliberation in supply chains. In *Proceedings of the International Conference on Web Intelligence* (pp. 411-418). ACM.
Original article available here.

Conjoint utilization of structured and unstructured information for planning interleaving deliberation in supply chains

Naeem Khalid Janjua[1], Omar Khadeer Hussain[2],
Elizabeth Chang[2], Syed Mohammed Shamsul Islam [1]
School of Science, Edith Cowan University[1]
School of Business University of New South Wales, Canberra[2]
n.janjua@ecu.edu.au o.hussain@adfa.edu.au e.chang@adfa.edu.au
syed.islam@ecu.edu.au

Abstract

Effective business planning requires seamless access and intelligent analysis of information in its totality to allow the business planner to gain enhanced critical business insights for decision support. Current business planning tools provide insights from structured business data (i.e. sales forecasts, customers and products data, inventory details) only and fail to take into account unstructured complementary information residing in contracts, reports, user's comments, emails etc. In this article, a planning support system is designed and developed that empower business planners to develop and revise business plans utilizing both structured data and unstructured information conjointly. This planning system activity model comprises of two steps. Firstly, a business planner develops a candidate plan using planning template. Secondly, the candidate plan is put forward to collaborating partners for its revision interleaving deliberation. Planning interleaving deliberation activity in the proposed framework enables collaborating planners to challenge both a decision and the thinking that underpins the decision in the candidate plan. The planning system is modeled using situation calculus and is validated through a prototype development.

Planning, Argumentation, Ontology, Unstructured Information

1 Introduction

Today, business planning systems are characterized by increasing dynamicity, which arises from the trends of the global economy, political situations, distribution of transport services and individual customer demands. To get a competitive advantage, businesses are involved in collaboration and mergers with others on a global scale, competing as Supply Chains (SC) rather than as individuals [31, 12]. The collaboration problem between trading partners consists of two interleaving functions, namely, planning and deliberation. Planning involves generating plans (or revising of candidate plans) whose success is warranted by some evidence coming from either one or more trading partners and deliberation to evaluate the acceptability of these plans by comparing the evidence supporting them against possible objections [2]. In the existing literature, various approaches have been proposed to support collaborative planning among trading partners to optimize business planning in different areas, such as procurement, production and distribution [22, 25]. However, all of them provide critical business insights from structured business data (i.e. sales forecasts, customers and products data, inventory details) only and fail to take into account unstructured complementary information residing in contracts, reports, user's comments, emails etc. Unless collaborative planning systems utilize the unstructured information conjointly with the structured data, planning interleaving deliberation is limited to an increasingly narrow slice of information. This is even more important today as the proportion of structured data to unstructured information is reported around 5% to 95% [4]. Therefore, this research focus on developing a planning system which will empower business planners to conjointly utilize structured data and unstructured information for effective business planning.

The rest of the paper is structured as follows: Section 2 and 3 outline motivation and related work, respectively. Basic action theory and its extension for planning interleaving deliberation are explained in Section 4. Proposed framework is discussed in Section 5. Section 6 outline the prototype development. Conclusion and future directions are discussed in Section 7.

2 Motivation

Data integration to support collaborative planning and deliberation has been considered as one of the core problems in business planning systems [32]. In the last decade, the focus of SC trading partners was on Systems of Records (SoR) i.e. structured data about sales, customer and product information, inventory forecasts and so, and it was used for planning and decision-making [6]. As a result, centralized enterprise information systems such as data warehousing systems exclusively dealt with record-oriented data that was carefully mapped using schema-centric mediation approaches by knowledge experts to support planning decisions. In such information systems, the decision support derived for planning can often answer the questions related to patterns of "What is happening" but provides no information to answer questions related to the pattern "Why it is happening and what is the rationale behind it". As a result, complex SC networks have a tendency to become vulnerable to uncertainties and operational risks [5]. To answer questions related to patterns of "Why is it happening and what is the rationale behind it", it is necessary to utilize SoR with the unstructured information generated as a result of the Systems of Engagement (SoE) with business partners or customers. The SoE is more decentralized, incorporate digital technologies for peer-to-peer interactions, and enable SC members in a network to collaborate and engage across a range of pivotal transactional processes on a global scale. For example, by using SoR, a supplier can predict SC disruptions that may be caused by unexpected demand patterns from other trading partners through predictive analytics, however, they would not be able to obtain information on the nature of complaints or requests made by the trading partner during the negotiation process which is captured and stored as SoE in the repositories' holding emails, contracts, reports or transcribed phone call information records during the planning and decision-making process. Therefore, SoE information complements SoR data with better insight reason and interpretation and this problem of knowledge sharing for effective planning and deliberation involve conjoint utilization of SoR along with SoE.

3 Related Work

The increased business operations outsourcing and the rise of digital technologies lead to widespread adoption of e-business models and businesses are involved in collaboration and mergers with others on a global scale, competing as a SC rather than as individuals [1, 26]. Additionally, the continuous drive towards leaning down the business processes and making more efficient supply chains during recent years have resulted in the SCs becoming more complex and vulnerable to operational risk and disruptions [5]. Therefore, effective planning faces the challenge of aligning the activities of SC trading partners to overcome operational risks. To overcome operational risks, several collaboration practices such as the Vendor Managed Inventory, Just In Time, Efficient Consumer Response, Continuous Replenishment and Accurate Response, Collaborative Planning, Forecasting and Replenishment (CPFR) that have been suggested in the literature [30] focus on better planning through tight processes integration and sharing structured data such as forecasting information, inventory details etc., among the SC trading partners as shown in Figure 1. As a result, the research techniques used for collaboration practices are drawn from applied mathematics, such as optimization, statistics, and decision theory and act as a black box for the business planner. They take in structured information only and generate results but provide no visibility as to the underlying reasoning and justification behind the results and end up with limited adoption in SCs [33].

The use of Semantic Web tools for collaboration activities addressed some of the challenges such as information has meaning attached to it that makes it understandable across enterprise boundaries and facilitates data sharing and integration [14]. Additionally, platforms such as D2RQ enable applications to access an RDF-view of the underlying structured data i.e. a non-RDF database such as SQL databases, through a rule engine API's over the Web via the SPARQL Protocol and as Linked Data. Attempts have been made to represent incomplete and contradictory information in information systems such as Dr-Prolog, Dr-Device, and Situated Courteous Logic [18]. These implementations only represent and handle individual conflicting preferences by defining priorities based on a single criterion between them before engaging in collaboration.

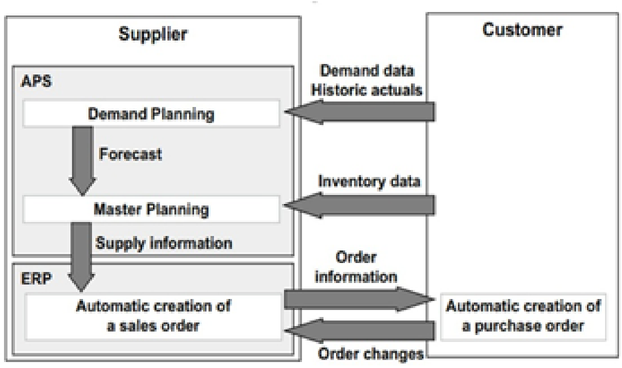

Figure 2: Business Planning System generating insights from structured data repositories

Therefore, these attempts do not provide a solution for collaborative planning that is subject to inconsistencies that derive from multiple data/information sources and multiple users. Furthermore, these techniques have not yet been applied to collaborative planning interleaving deliberation domains. The Collaborative Planning and Acting Model [27] is the first attempt to support planners in managing and planning information and facilitates the planning process with automated reasoning. However, the model lacks the means to represent incomplete and contradictory information and logical relations that define constraints and the axioms of the domain being modeled.

4 Basic action theory for planning interleaving DELIBERATION

We use situation calculus [24] to model the planning system. We describe and employ the extended version of Reiter [29] to formalize our model. In our model, each business planner maintains the representation of the domain as basic action theory and it has the following form:

$$\mathcal{D} = \Sigma \cup \mathcal{D}_{ss} \cup \mathcal{D}_{ap} \cup \mathcal{D}_{una} \cup \mathcal{D}_{S_0} \tag{1}$$

Where

- $\sum$ is a set of fundamental domain-independent axioms describing the basic properties of the situation.
- $\mathcal{D}_{ss}$ is a set of successor state axioms represent relational or functional fluents in the domain. Formally, $Poss(a, s) \supset [T(do(a, s)) \equiv \gamma_T^+(a, s) \vee (T(s) \wedge \neg\gamma_T^-(a, s))]$ where γ_T^+ and γ_T^- represent the add and delete conditions of fluent T.

- $\mathcal{D}_{ap}$ is a set of precondition axioms under which action can be performed. Formally, represented as $\Pi_A(s) \equiv Poss(A, S)$.
- $\mathcal{D}_{una}$ is a set of unique names axioms for actions.
- $\mathcal{D}_{S_0}$ is a set of the first-order sentence that represents the initial state of the world.

A basic action theory for a planning system specifies a plan and the tasks of the domain of concern and the contextual settings in which the business planners operate. A plan in situation calculus is treated as an executable situation that satisfies a goal statement. We assume that the sets Fluents, NonFluents, and Actions are shared among the business planners. Additionally, they share a common goal, knowledge about the fundamental axioms, unique name axioms for actions and the names of the object in the domain. We use the definition of plan and planning problem defined in [2] as follows:

Definition 1 *Given a basic action theory $\mathcal{D}$ and a Goal g with single free variable s, a plan π is a variable-free situation term s_π iff $\mathcal{D} \Vdash\ executable(s_\pi) \wedge g(s_\pi)$ where $executable(s_\pi)\ \ (\forall a, s^*).do(a, s^*) \sqsubseteq s_\Pi \supset Poss(a, s^*)$.*

It is important to note here is that the term s_π represents the history for the execution of the actions of a plan in sequence.

Definition 2 *A planning problem $\mathcal{P}$ is a tuple $< \mathcal{D}, g >$ where $\mathcal{D}$ is a basic action theory denoting the planning domain and g is a fluent sentence specifying the goal.*

As a result of above definition of a plan and the foundational axioms for situations, we can identify that $excutable(do(a, s)) \equiv executable(s) \wedge Poss(a, s)$. This enables the transformation of plan definition as follows:

Definition 3 *A plan $\pi = A_1, A_2; .....; A_n$ is a solution to a planning problem p iif $\mathcal{D} \models\ Poss(A_1, S_0) \wedge do(A_1, S_0) = S_1 \wedge\ Poss(A2, S_1) \wedge do(A_1, S_1) = S_2 \wedge .... \wedge Poss(A_n, S_{n-1}) \wedge do(A_n, S_{n-1}) = S_n \wedge G(S_n)$.*

This definition asserts that the actions in the plan can be performed in sequence eventually performing the final action results in goal sentence $\mathcal{G}$ be true.

A basic action theory is necessary to define a domain for reasoning. However, in classical AI reasoning is performed under certain assumption such as follows:

1. The given problem can be fully addressed with available information (solution to the problem lies within the available situation tree).
2. The domain knowledge is consistent. In other words, they assume that there will be no conflicting events and situations during the collaborative planning process.

3. New information is consistent with the already available information or specifications.

4. New information does not lead to retraction of previous conclusions.

Because of these limitations discussed above, AI failed to provide a solution to many real world scenarios where some of the information or actions in a plan may result in conflicting situations. To overcome this, we employ a model based on an extended version of action theory, $\mathcal{D}_{ext}$, that consider the representation of conflicts and provide support for conflict resolution during the planning process.

Definition 4 *Extended action theory is defined as follows:*
$\mathcal{D}_{ext} = \mathcal{D} \cup \Omega_c \cup \Omega_p$

where

- $\mathcal{D}$ is a basic action theory.
- Ω_c is a set of axioms for representing conflicting information. For example argument(X,p), argument(Y,$\neg p$), counterArgument(Y, X, do(a,s)), underCut(Z,C, do(a,l)) etc.
- Ω_p is a set of axioms used in deliberation module such as speech acts for communication and dialogue movies for establishing a preference between conflicting situations. For example, propose(A), reject(A), Argue(A=>P), Why(P), Support(P) etc.

Definition 5 *Using extend situation calculus A planning problem P_{ext} is a tuple ¡ D_{ext},g,$\{CQ\}$¿ where D_{ext} is a extended action theory denoting the planning domain and g is a fluent sentence specifying the goal and CQ is a set of critical questions to warrant the execution of the plan i.e.* $D_{ext} \Vdash\ executable(s_\pi) \wedge g(s_\pi) \sqsubseteq s\pi \supset\ Poss(\{CQ\}, s^*)$.

$\{$CQ$\}$ is a set of critical questions and can be categorized as a set of exceptions and assumptions against a plan under consideration. Each critical question in a set is represented as $\{Poss(X_i, S_i) \wedge ..... \wedge Poss(X_n, S_n) \vdash G(S_n)\}$. These critical questions are the premises provide reasons for justifying the conclusion only if the assumptions are true and there are no exceptions. If either an assumption is false or an exception is true, unless premises provide reasons for accepting the conclusion, the conclusion would not be valid. Thus, both assumption and exceptions attack the conclusion of the scheme.

Definition 6 *An extended plan $S^{\pi}_{ext} = A_1, A_2; .....; A_n$ is a solution to a extended planning problem p iif* $D_{ext} \models\ Poss(A_1, S_0) \wedge do(A_1, S_0) = S_1 \wedge\ Poss(A2, S_1) \wedge do(A_1, S_1) = S_2 \wedge .... \wedge Poss(A_n, S_{n-1}) \wedge do(A_n, S_{n-1}) = S_n \wedge G(S_n) \wedge Poss(\{CQ\}, S) \vdash true$.

The plan is the solution to the extended problem and all assumption are true and exceptions are false.

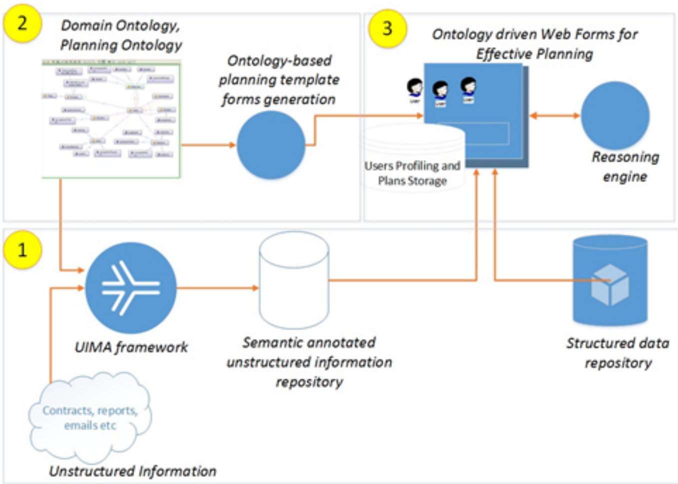

Figure 3: Proposed conceptual framework for effective business planning utilizing knowledge from structured and unstructured information conjointly.

Definition 7 *In extended plan* S^{π}_{ext} *a successor axiom is considered strict if it represents truthful information which contains no ambiguity.*

Consider rule r1 which states that 'if a person is innocent and has no crime history then he is not guilty' and rule r2 which states that 'if someone is not guilty, then he is free'. These rules can be represented as strict production rules thus:

- $[r1] innocent(X) \wedge hasCrimeHistory(X, no) \rightarrow \sim guilty(X)$
- $[r2] not\ guilty(X) \rightarrow do(free(X))$.

Definition 8 *In extended plan* S^{π}_{ext} *a successor axiom is considered defeasible if it represent tentative information which may change in due course.*

Consider rule r3 that states: 'assume that someone is innocent whenever it has not been proven that he is guilty' and rule r4 that states: 'generally, do not cross the railway tracks if it ca not be proven that no train is coming'. These rules can be represented as defeasible production rules as follows:

- $[r3] not\ guilty(X) \dashrightarrow innocent(X)$.
- $[r4] not\ \sim train\_is\_coming \dashrightarrow$ $\sim \_railway\_tracks(X)$.

he set of conditions to a conclusion with a certain doubt and therefore could be refuted by contrary evidence. This type of rule is indicated by words like 'usually', 'presumably', or 'sufficiently' or we could intuitively feel that it is refutable.

Definition 9 *A defeasible successor axiom p conflicts with another defeasible successor axiom ¬p in an extended plan iff ¬p executes action a after p has executed an internal action ¬a.*

$Conflict(a, a) = counterArgument(p, \neg p, do(a, s)) =_{def} p \neq \neg p \wedge Poss(\neg p, a, s) \wedge (\exists \neg a, s^{'})[Poss(p, a^{'}, s) \wedge do(a^{'}, s^{'}) \subset s \wedge Poss(a^{'}, a, >, s)]$

A set of possible conflicts set $\hat{Conflict}$ contains situations that can be used to generate counter-arguments. Note that no situation weights are used in both plan construction and conflict set. Therefore, the attack between arguments is symmetric i.e. they are equally acceptable. Therefore, business planners need to perform deliberation to establish a preference between conflicting arguments among them. Therefore, $Preference(a^{'}, >, a) =_{def} Deliberation(Conflict(a^{'}, a)))$ where assign is a primitive action that triggers deliberation dialogue interleaving planning activity.

The planning interleaving deliberation is a dialogue-based system consists of union of arguments that are constructed by the proponent and the opponents i.e. $\mathcal{A} = \mathcal{A}_{pro} \cup \mathcal{A}_{opps}$ where $\mathcal{A}$ represent an argument consist of preconditions and a successor state. As a result of the dialogue process, arguments lines are constructed and acceptability of arguments is computed to establish priority between conflicting situations in a plan. Once the priority is established, the preference is included into the plan along with justification information. We reuse the syntax and semantics for argumentation system defined in [17] and extend it for dialogue-based system using semantics defined in [21]. We explain the working of dialogue-based system in the next section.

Definition 10 *Given extended action theory $\mathcal{D}_{ext}$, a collaborative plan S_Π for a common Goal G is a variable-free situation term s_Π iff $\mathcal{D}_{ext} \models executable(S_\Pi) \wedge G(S_\pi) \sqsubseteq S\pi \supset Poss(\{CQ\}, S^*)$ where $executable(S_\Pi)$ $(\forall a, s^*).do(a, s^*) \sqsubseteq s_\Pi \supset Poss(a, s^*)$ and if conflict exists, there exists a preference relationship, such that $Preference(a^{'}, a) \equiv Support(a^{'}) \models do(a^{'}, s)$.*

Definition 11 *A collaborative planning solution $\mathcal{CP}_s$ is a tuple $< \mathcal{D}_{ext}, \mathcal{G}, \hat{p} >$ where $\mathcal{D}_{ext}$ is extended action theory for representing the sequence of actions and $\mathcal{G}$ is a fluent sentence specifying the goal and $\hat{p}$ represents the priority relationships over conflicting situations.*

5 Proposed framework

Current planning systems have been reported [9, 27] low penetration in real world application due to the following reasons:

1. Human planners want to use a tool for better visibility of the planning process but they also want to control the decision-making part of the planning phase.

2. High level of automation results in reduced situation awareness and skill degradation.

3. The huge amount of time and manpower needed to enter all information.
4. The difficulty of converting human concepts into tool supported language.

Therefore, this research is an attempt to overcome the limitations of the current planning systems and propose an interactive planning approach where planning is interleaved with deliberation in order to help the business planners in determining which actions are possible at the current stage, helping them in making the best choice by building arguments in favour and against conflicting situations. During the planning phase, the planner will use existing planning template forms to create candidate plan using both structured and semantically annotated unstructured information loaded on the Web forms.

In the domain of uncertainty, a business planner may not be able to identify a plan of action especially if there is not enough information to account for all necessary conditions. Therefore, candidate plan created by a business planner needs to be put forward to collaborating partners for iterative plan development. Therefore, once a candidate plan is ready, it is put forward for collaborative planning. During planning, conflicts may arise, thereby planning interleaving deliberation activity in the proposed framework is a dialogue-based system that allows planners to take into account conflicting situation and resolve them building arguments in favor and against them. Figure 3 depicts the proposed conceptual framework for effective business planning using conjointly the structured data and unstructured information that is stored isolated from each other. In the following subsections, we explain the working of proposed framework in detail.

5.1 Semantic annotation of unstructured information using domain ontology

Ontology is defined as a shared conceptualization of a certain domain [20]. Representing knowledge in the form of ontologies has several advantages such as knowledge sharing, knowledge reuse and it helps in building automated systems using logic-based reasoning [14]. Semantic annotation is a process which makes use of one or several ontologies to tag the unstructured information. During this process, semantic tags are associated with each word or a group of words in a statement. This results in the structure being added to the unstructured information. In our previous work [15], a semi-automated semantic annotation approach was developed where business planners were provided with a web form to load and tag unstructured information using ontologies. In this research, we extend our previous approach using technique proposed by [3] to annotate unstructured information using domain ontology and produce RDF triplets. In the proposed technique UIMA (Unstructured Information Management Architecture) [8] is a fundamental framework and act as a back-end engine that uses both statistical and rule-based annotators for text annotation. As pointed out in the literature [6], this research will take into account some important considerations duration semantic annotation which are: entity extraction; identification of

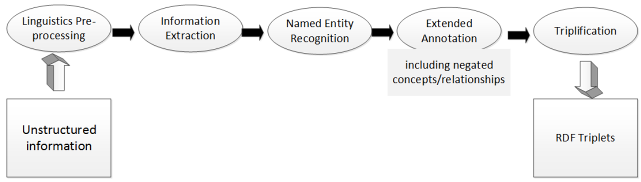

Figure 4: UIMA Pipleline's Moudles

a relationship between two entities; a network of relationships between entities; associations between several entities; and associations between groups of entities such as ones arranged in sections and/or subsections. The UIMA pipeline consists of five functional processing namely; Linguistic Processing, Information Extraction, Named Entity Recognition, Open Annotation, and Triplification. As a result, CAS is transformed to RDF triplets.

UIMA NLP analysis workflow is based on the so-called Aggregate Analysis Engine (AAE) which is composed of many annotators. The analysis results are stored in Common Analysis Structure (CAS) which is in-memory data structure shared between various annotators. Finally, CAS Consume modules handle the results after analysis is complete. The RDF CAS Consumer is responsible for taking a CAS view and write it to a file in a RDF format; this is useful to plug UIMA pipelines with RDF backed systems.

5.2 Embedding an ontology in plan generation form fields

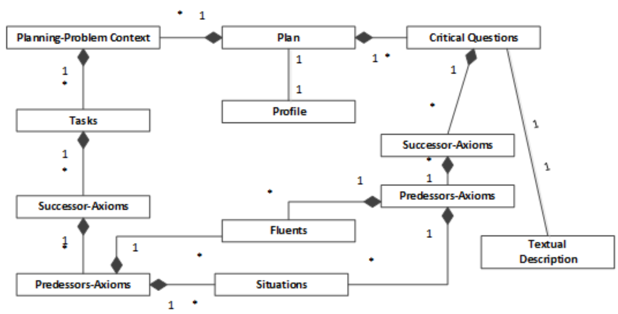

Figure 5: Plan Model Ontology

Data and information available to the business planner for plan generation are usually drawn from traditional databases to the Web forms. Those Web forms are composed of several labels and associated fields. The inherent semantics of the form is encoded in the meaning of the labels, the grouping of fields using fieldsets and order of fields.

In this research, we embedded and associated ontologies illustrated in Figure 5 with Web forms fields to create plan generation templates. These plan generation templates are used by business planners to generate semantically enriched plans that can be put forward to further semi-automated collaborative planning interleaving deliberation activity. The approach we follow for structured semantic data and information acquisition from Web forms using ontology has been discussed by researchers in the past [13, 11].

Web Ontology Langauge (OWL) is preferred language for modeling ontologies. It is based on Open World assumptions and used for class definitions and collection of description logic axioms. However, Template-based knowledge representation system uses Close World Assumption and have local constraints that can be validated easily while in axiom-based system with OWA such local constraint checking is much more problematic [11]. Additionally, in planning domain, different viewpoints of a planner leads to conflicting situations and such conflicts can't be represented using OWL and SWRL [16]. Therefore, Template-based knowledge representation allows a business planner to choose, organise, and revisit his actions and plans [9] before indulging in collaborative planning.

Inspired by the argumentation schemes [28], we have provided syntax and semantic (in section 3) for incorporating critical questions to capture assumptions and exceptions associated with the plan under consideration as depicted in data model ontology shown in the Figure 5. These argumentation schemes have emerged from informal logic (philosophy) and help to categorize the way arguments are built, aiming to fill the gap between logic-based application and human reasoning by providing schemes which capture stereotypical patterns of human reasoning, e.g., arguments from an expert opinion scheme. Formally, an argumentation scheme is composed of a set of premises Ai, a conclusion C, and a set of critical questions CQi with the aim of defeating the derivation of the consequences [28, 23].

The overall semantics of the forms in Figure 6 and 7 are rather complex. The forms are used to generate plan proposal for risk management. The 'Name' label could describe the current company or its dept or even the current process. The semantics of the Name are clear only to the developer of the application or an expert of the system. Therefore, creating a mapping between plan model ontology and form fields helps to overcome such ambiguities. The information from the form is saved in RDF triplet format. For example, information from the Web form depicted in Figure 6 i.e., profile name is saved as Plan(PROFILE,'Discount'). Similarly, every form filled is mapped to data model ontology and form values are saved as RDF triplets.

5.3 Planning interleaving deliberation

In a static environment, SC member may have chosen to flourish specific and efficient process linkages and information sharing/exchange mechanisms with selected partners, but in the dynamic environment, enterprises need to develop more robust and reconfigurable digital linkages that can deal with changes in the business environment [7]. For example, SC works as a network , however, when an issue arises, it's not entire supply chain need to deal with that. Instead, we state that it gives birth to ad-hoc hub and spoke system. Hub being having the

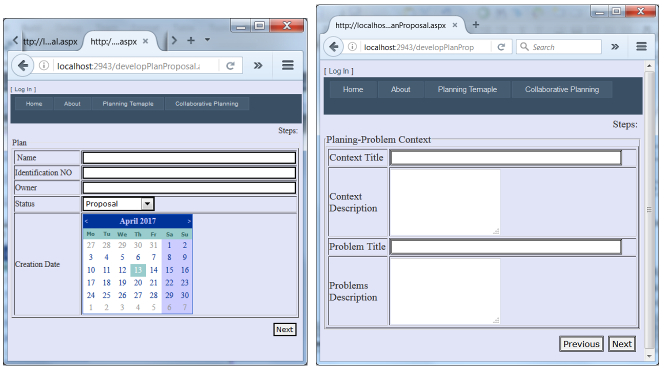

Figure 6: Left (Step1) Right (Step2)

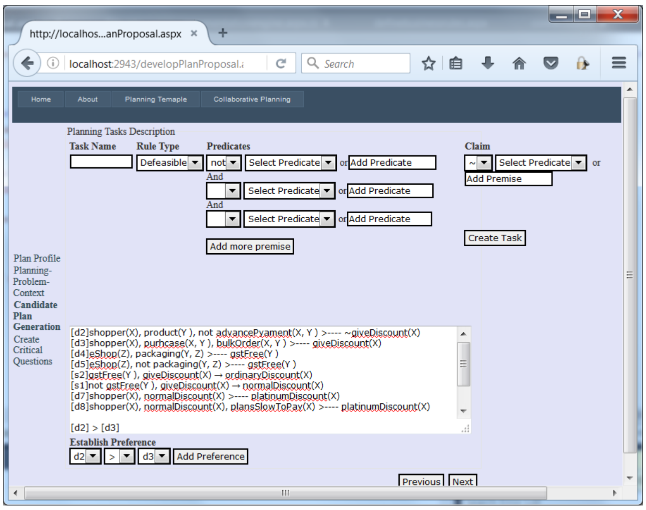

Figure 7: Candidate plan generation

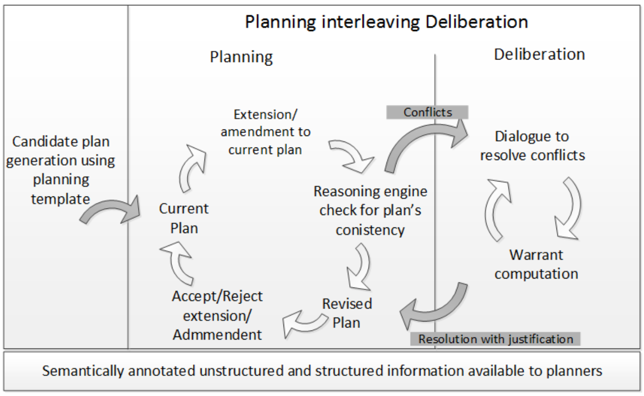

Figure 8: Planning Support System Activity Model

higher impact of risk and spoke being immediate affectee. Figure 8 depicts the planning support system activity model. The business planner located at hub can generate a candidate plan and share it with the planner located at spoke of the network for further extension and revision.

5.3.1 Generation of candidate plan using planning template

Web forms depicted in Figures 6 and 7 are used by a business planner to generate a candidate business plan using semantically annotated unstructured and structured information available to them in the drop down menus. The planner can add new premises to define tasks using text fields. For axiomatising of the planning knowledge, we use Defeasible Logic Programming (DeLP). DeLP is a general-purpose defeasible argumentation formalism based on logic programming, intended to model inconsistent and potentially contradictory knowledge (both strong and weak negation). A defeasible logic program has the form $\psi = (\Pi, \Delta)$, where Π and Δ stand for strict knowledge and defeasible knowledge, respectively. We extended DeLP for knowledge representation and reasoning in semantic web application [14, 18]. We defined syntax and semantics for strict and defeasible rule representation in web-based applications. We reuse our previous work here for defining the planning tasks. In the rule base, a planning task (rule) takes the following form *[rule identifier] [rule body] [type of rule] [head]*. The rule body represents precondition and rule head represent the effects [15]. Using hybrid reasoning engine, individual plans consistency is warranted. In candidate plans, the planner can overcome his conflicting viewpoints by establishing the priority between them. The reasoning engine uses forward chain reasoning to digitize the plan and make them alive for the planners. For more information about forward chain reasoning using rete algorithm, readers are referred to [17]. The backward chain reasoning is used to answer business planner queries.

5.3.2 Iterative plan revision by collaborating business planners

This work is built upon our previous work [19]. Algorithm 1 describes the iterative planning process that involves revision of plans by the partners. Each iteration involves calls to underlying hybrid reasoning engine to find out whether the returned plan is warranted, and the revised theory so that future plans returned from hybrid reasoning engine don't suffer from any contradictions.

Iterative planning interleaving deliberation is a deliberation dialogue-based system that involves participants who share responsibility and collaborate on deciding what action or course of actions should be undertaken in a given situation [34]. In such dialogues, participants don't have fixed positions at the start of the dialogue and the goal and need for action can originate from any of the various participants involved. During the course of action, however, participants may be involved in a persuasion dialogue which may motivate them to model a persuasion dialogue as embedded in a deliberation dialogue.

Data: $(\mathcal{S}^{\pi}_{ext})$
Result: $(\mathcal{C}P_s)$
Array participants []= {p1, p2, p3};
repeat
  foreach *p from participants* **do**
    ArgumentSet = ArgumentSet $\cup$ argumentFrom(a,p)
    if *conflict(a′,a) in ArgumentSet* **then**
      Preference(a′,P,a)= Deliberation(Conflict(a',a));
      ArgumentSet = ArgumentSet $\cup$ Preference(a',P,a);
    end
    $\mathcal{C}P_s = \mathcal{C}P_s \cup$ ArgumentsSet
  end
until $\mathcal{G} \vdash true$;

Algorithm 1: *Planning interleaving deliberation*

A plethora of work exists on building dialogue-based systems for software agents. This research focus on extending the work done by [21] using argumentation schemes [18]. During the process of argumentation, relationships between the arguments are linked with each other in a certain pattern to support the ultimate conclusion. Such linking patterns are called 'Argumentation Schemes' and allow reasoning to be performed using a set of premises and a conclusion. These argumentation schemes have emerged from informal logic [10]. The schemes help to categorize the way that arguments are built. They bridge the gap between logic-based application and human reasoning by capturing stereotypical patterns of human reasoning. An example is an argument from an expert opinion scheme. In this research, arguments are built using argumentation schemes during deliberation. The objective is two-fold, firstly; to enable planners to put forward their arguments that may be incomplete statements and offer them ways of advancing well-formed arguments as well as to reuse arguments that often appear in discussions; secondly, with the help of algorithms, to compute the acceptability of arguments at any stage of the discussion.

The deliberation dialogue system is defined by:

1. Topic Language: DeLP as a logical language.

2. Argumentation Logic: as defined in [17]. The only difference is that in our previous work it was assumed that the system has collated all the relevant information and reasoning engine reasoning over it. In this system, business planners are collaborating and conflict resolution process is a dialogue -driven activity. We reuse the definition of argument, sub-argument, attack, static defeat and dynamic defeat.

3. Communication Language to define set of Locutions $L \epsilon \Omega_p$ and two binary relation $R_a \epsilon \Omega_p$ and $R_s \epsilon \Omega_p$ of attacking and surrendering reply on L. Dialogue moves and termination as defined in [21].

To answer the questions of a decision maker which may help him to understand the reasoning process (that is, to obtain an explanation on the conclusion achieved), planning system provides a querying mechanism to query the knowledge base.

6 Prototype development

The development of the prototype application is carried out with help Microsoft Visual Studio 2015[1], NRuler [2] which is a fast production system library based on the RETE algorithm, written in C sharp. This library is extended for the development of the hybrid reasoning engine. QuickGraph [3] that provides generic directed/undirected graph data structures and algorithms for .NET. It also supports Graphviz [4] to render the graphs. It is used to generate the graphical representation of the reasoning results produced by the prototype and DeLP Server that is is an implementation of defeasible logic programming (DeLP). It is used as a back-end server for the development of the hybrid reasoning engine. MySQL [5] open source relational database for storing Plans profiling information.

7 Conclusion and future work

In this article, a planning support system is designed and developed that empower business planners to develop and revise business plans utilizing both structured data and unstructured information conjointly. In future work, it is intended to enhance this work with the machine learning techniques. In particular, existing machine learning either doesn't consider domain knowledge during

[1]http://www.microsoft.com/visualstudio/en-us
[2]http://nruler.codeplex.com/
[3]http://quikgraph.codeplex.com/
[4]http://www.graphviz.org/
[5]http://www.mysql.com/

classification or if it does, then this knowledge holds for the whole domain. Such approaches ignore any specific information or situation that may apply to some small set of chosen learning examples[6]. Therefore, it is intended to enhance the current generation of planning system with argumentation driven machine learning techniques. As a result, the arguments pertinent to specific situations will be considered during the mining of an planning information and the mining results will be available to business planners for effective business planning.

References

[1] M. Alaranta and S. Henningsson. An approach to analyzing and planning post-merger is integration: Insights from two field studies. *Information Systems Frontiers*, 10:307–319, 2008.

[2] A. Belesiotis, M. Rovatsos, and I. Rahwan. A generative dialogue system for arguing about plans in situation calculus. In P. McBurney, I. Rahwan, S. Parsons, and N. Maudet, editors, *Argumentation in Multi-Agent Systems*, volume 6057 of *Lecture Notes in Computer Science*, pages 23–41. Springer Berlin Heidelberg, 2010.

[3] C. Bretschneider, H. Oberkampf, and S. Zillner. *UIMA2LOD: Integrating UIMA Text Annotations into the Linked Open Data Cloud*, pages 16–31. Springer International Publishing, Cham, 2015.

[4] M. L. Brodie. Computer science 2.0: A new world of data management. In *Proceedings of the 33rd international conference on Very large data bases*, pages 1161–1161. VLDB Endowment, 2007.

[5] Y. Daultani, S. Kumar, O. S. Vaidya, and M. K. Tiwari. A supply chain network equilibrium model for operational and opportunism risk mitigation. *International Journal of Production Research*, 53(18):5685–5715, 2015.

[6] T. S. Dillon, Y.-P. P. Chen, E. Chang, and M. Mohania. Conjoint mining of data and content with applications in business, bio-medicine, transport logistics and electrical power systems. In *Artificial Intelligence Applications and Innovations*, 2014.

[7] J. Dyer. Effective interfirm collaboration: how firms minimize transaction costs and maximize transaction value. 2002.

[8] D. FERRUCCI and A. LALLY. Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10:327–348, 9 2004.

[9] M. Ghallab, D. Nau, and P. Traverso. The actor's view of automated planning and acting: A position paper. *AI*, 208(0):1–17, 2014.

[6] http://www.ailab.si/martin/abml/

[10] D. J. Godden and D. Walton. Advances in the theory of argumentation schemes and critical questions. *Informal Logic*, 27:267–292, 2007.

[11] R. S. Gonçalves, S. W. Tu, C. I. Nyulas, M. J. Tierney, and M. A. Musen. Structured data acquisition with ontology-based web forms. In *ICBO*, 2015.

[12] P. Graça and L. M. Camarinha-Matos. Performance indicators for collaborative business ecosystemsliterature review and trends. *Technological Forecasting and Social Change*, 116:237–255, 2017.

[13] G. Hohmann and M. Fichtner. Embedding an ontology in form fields on the web. *Semantic Web Journal - Interoperability, Usability, Applicability*, 3(4):1–9, 2012.

[14] N. K. Janjua. *A Defeasible Logic Programming-Based Framework to Support Argumentation in Semantic Web Applications*. Springer Theses. Springer International Publishing, 2014.

[15] N. K. Janjua. Process map discovery from business policies: A knowledge representation approach with argumentative reasoning (kr@ pmd). In *A Defeasible Logic Programming-Based Framework to Support Argumentation in Semantic Web Applications*, pages 201–233. Springer, 2014.

[16] N. K. Janjua and F. K. Hussain. Development of a logic layer in the semantic web: Research issues. *Semantics, Knowledge and Grid, International Conference*, pages 367–370, 2010.

[17] N. K. Janjua and F. K. Hussain. Web@ idss–argumentation-enabled web-based idss for reasoning over incomplete and conflicting information. *Knowledge-Based Systems*, 32:9–27, 2012.

[18] N. K. Janjua, F. K. Hussain, and O. K. Hussain. Semantic information and knowledge integration through argumentative reasoning to support intelligent decision making. *Information Systems Frontiers*, 15(2):167–192, 2013.

[19] N. K. Janjua, O. K. Hussain, and E. Chang. *Interleaving Collaborative Planning and Execution Along with Deliberation in Logistics and Supply Chain*, pages 139–148. Springer International Publishing, Cham, 2015.

[20] J. Kim, P. Kim, and H. Chung. Ontology construction using online ontologies based on selection, mapping and merging. *International Journal of Web and Grid Services*, 7:170–189, 2011.

[21] E. M. Kok, J.-J. C. Meyer, H. Prakken, and G. A. Vreeswijk. A formal argumentation framework for deliberation dialogues. In *Argumentation in Multi-Agent Systems*, pages 31–48. Springer, 2011.

[22] M. Kurtuluş. *Collaborative Forecasting in Retail Supply Chains*, pages 39–61. Springer International Publishing, Cham, 2017.

[23] I. Letia and A. Groza. A planning-based approach for enacting world wide argument web. In *Intelligent Distributed Computing, Systems and Applications*, volume 162 of *Studies in Computational Intelligence*, pages 137–146. Springer Berlin / Heidelberg, 2008.

[24] J. McCarthy. Situations, actions, and causal laws. Technical report, DTIC Document, 1963.

[25] M. Nematollahi, S.-M. Hosseini-Motlagh, and J. Heydari. Coordination of social responsibility and order quantity in a two-echelon supply chain: A collaborative decision-making perspective. *International Journal of Production Economics*, 184:107 – 121, 2017.

[26] A. Norta and R. Eshuis. Specification and verification of harmonized business-process collaborations. *Information Systems Frontiers*, 12:457–479, 2010. 10.1007/s10796-009-9164-1.

[27] J. Patel and M. Dorneich. Improving coalition planning by making plans alive. *Intelligent Systems, IEEE*, 28(1):17–25, Jan 2013.

[28] I. Rahwan, F. Zablith, and C. Reed. Towards large scale argumentation support on the Semantic Web. In *AAAI'07: Proceedings of the 22nd national conference on Artificial intelligence*, pages 1446–1451. AAAI Press, 2007.

[29] R. Reiter. On knowledge-based programming with sensing in the situation calculus. *ACM Trans. Comput. Logic*, 2(4):433–457, Oct. 2001.

[30] Y. Sheffi. The value of CPFR. In *Proceedings of the Fourth International Congress on Logistics Research, IMRL, Lisbon, Portugal*, 2002.

[31] R. Spekman and E. W. Davis. The extended enterprise: a decade later. *International Journal of Physical Distribution & Logistics Management*, 46(1):43–61, 2016.

[32] G. C. Stevens and M. Johnson. Integrating the supply chain: 25 years on. *International Journal of Physical Distribution & Logistics Management*, 46(1):19–42, 2016.

[33] E. J. Umble, R. R. Haft, and M. M. Umble. Enterprise resource planning: Implementation procedures and critical success factors. *European journal of Operational Research*, 146(2):241–257, 2003.

[34] D. N. Walton and E. C. Krabbe. Commitment in dialogue: Basic concepts of interpersonal reasoning. *State University of New York Press*, 35, 1995.