

# Modelling Shared Knowledge and Shared Knowledge Awareness in CSCL Scenarios through Automated Argumentation Systems

María Paula González<sup>1,2</sup>, Carlos Iván Chesñevar<sup>1,2</sup>, Cesar A. Collazos<sup>3</sup>,  
Guillermo R. Simari<sup>2</sup>

<sup>1</sup> National Council of Scientific and Technical Research (CONICET), Argentina

<sup>2</sup> Department of Computer Science and Engineering – Universidad Nacional del Sur  
Av Alem 1253 – 8000 Bahía Blanca, Argentina  
{mpg,cic,grs}@cs.uns.edu.ar

<sup>3</sup> Department of Systems - Universidad del Cauca  
FIET-Sector Tulcan, Popayán, Colombia  
ccollazo@unicauca.edu.co

**Abstract.** Over the last few years, argumentation systems have been gaining increasing importance in several areas of Artificial Intelligence, mainly as a vehicle for facilitating rationally justifiable decision making when handling incomplete and potentially inconsistent information. Argumentation provides a sound model for dialectical reasoning, which underlies discussions among students when solving tasks collaboratively in a CSCL environment. In this setting, we identify the problem of constructing Shared Knowledge and its related Shared Knowledge Awareness. While Shared Knowledge refers to the common knowledge students acquire when they work in a collaborative activity, Shared Knowledge Awareness is associated with the consciousness on the Shared Knowledge that a particular student has. This paper presents a novel approach to model Shared Knowledge construction and the associated Shared Knowledge Awareness through an automated argumentation system.

## 1 Introduction

*Shared Knowledge* (SK) concerns the common knowledge constructed by a student group when carrying out a collaborative learning activity in a CSCL environment. In this setting, *Shared Knowledge Awareness* (SKA) has been defined as the consciousness on the SK that this student group has when performing a specific collaborative task in a restricted moment of time [1]. Indeed, the construction of SK is strongly related to the acquisition of an appropriate level of SKA, as being aware of any knowledge (in particular SK) implies learning something about it.

Students' acquisition of SKA in CSCL scenarios is not a simple task, and a number of questions that should be considered to reach it have been proposed [1]. How-

ever, it is difficult to ascertain how to provide mechanisms to model the construction of SKA in a real CSCL system. Indeed, this problem is related to different features, in particular with characterizing the students' dialectical reasoning underlying negotiation processes when looking for an agreement or consensus about a given claim. In this context, automated *argumentation systems* [2] provide an interesting formalization tool, as they have matured in the last decade to become a sound setting to formalize commonsense, dialectical reasoning. Indeed, defeasible argumentation has been successfully used in legal reasoning, multiagent platforms and decision making systems among others [2, 3, 4].

This paper presents a novel approach for integrating automated argumentation systems as a support tool for characterizing SK and SKA in CSCL scenarios. Taking as a starting point the individual knowledge constructed by each student when performing a collaborative task, our goal is to provide an argument-based mechanism through which part of the SK can be constructed semi-automatically, explicating as well its related SKA. For achieving this, the individual knowledge coming from every student will be collected in a common knowledge base  $K$  in which part of the knowledge is tentative (or *defeasible*). On the basis of  $K$ , different arguments (possibly in conflict) will be automatically obtained, providing additional support for the dialectical discussion performed by the students. Moreover, conflicts among arguments will be automatically detected by means of an automated argumentation system. The system will also determine which arguments ultimately prevail in a discussion, which will be called *warranted arguments*. In our proposal, such warranted arguments will provide a part of the SK among students, whereas visualization and explanation facilities provided by the argumentation system will help to make explicit the associated SKA.

The rest of this paper is structured as follows: Section 2 briefly describes the concepts of SK and SKA. Section 3 provides an overview of the fundamentals of argumentation systems. Section 4 describes our proposal for modelling SK and SKA construction in CSCL by means of an automated argumentation system. Section 5 discusses the feasibility aspects of our proposal and presents a case study. Section 6 discusses related work. Finally, Section 7 concludes and discusses future work.

## 2. Shared Knowledge and Shared Knowledge Awareness

*Shared Knowledge* (SK) concerns the common knowledge constructed by a group of students when carrying out a collaborative learning activity in a CSCL environment. It refers to the understanding that any student of the group has about several aspects of the collaborative work, including coordination, strategy communications, monitoring, and shared comprehension of the problem [1]. In this context, *Shared Knowledge Awareness* (SKA) has been defined as the consciousness on the shared knowledge that a particular student group has when carrying out a specific collaborative learning activity in a CSCL environment in a restricted moment of time [1].

Students' acquisition of SKA in CSCL scenarios is not a simple task, and it is difficult to ascertain how to provide mechanisms to model the construction of SKA in a real CSCL system. In that respect, a number of questions that should be considered to reach it have been proposed [1]. In addition, some guidelines were outlined to ensure

the existence of appropriate reinforcement elements in a CSCL interface related to SKA [5]. However, those proposals focus on modelling the individual perception of the collaborative task, as it is the student who has to answer the questions mentioned before, or who will perceive a particular CSCL interface. Even when the construction of the SKA relies on individual introspective activities carried out by each student, the outcome of discussions performed within the group is crucial to detect the SK and to understand the group dynamics, i.e. to construct SKA.

In the above scenario, a *dialectical discussion* is normally performed among students when solving a collaborative task to be aware of the SK they have. This discussion includes the exchange of objective information or facts, as well as incomplete (and sometimes contradictory) perceptions of the reality related to the task to be solved, as most human activities have to deal with uncertainty and lack of complete information in the real world. In such discussion the participants may be biased in their opinions (given their own preferences, beliefs, etc.), so that a careful analysis of arguments and counterarguments advanced during the discussion is required to determine whether some particular belief is actually accepted by the group.

Modelling dialectical discussions in CSCL scenarios and their relationship with SK and SKA construction is a challenging issue nowadays. It implies moving the focus from the analysis of the isolated student perception to the study of the exchanged information within the group and its dynamics. In that respect, different alternatives have been recently proposed for identifying common aspects of dialogues in CSCL, such as the study of the students' focus of attention [6], the visualization of discussion and agreement during online discussions [7], or the analysis of the text-based communication in a web-based CSCW system [8,9]. However, to the best of our knowledge, none of those existing approaches incorporate automated argumentation systems for providing support to model conflicting situations in the dialectical communication in CSCL. Besides, most of such approaches make use of a slightly cryptic concept of SKA which is not *explicitly* presented to students. Argumentation systems are advantageous in this context, as they can be used to explore automatically *all* possible arguments associated with a given claim on the basis of the students' knowledge, helping them to clearly identify why their claims hold (or do not hold), minimizing thus the bias present in the learning process. As we will see next, automated argumentation systems can also provide a rich formal framework for computing *explicit* SKA, thus characterizing a crucial part of the SK construction in CSCL scenarios.

### **3. Argumentation Systems: a brief overview**

Argumentation is an important aspect of human decision making. In many situations of every day's life, people when faced with new information need to ponder its consequences, in particular when attempting to understand problems and come to a decision. *Argumentation systems* [2] are increasingly being considered for applications in developing software engineering tools, constituting an important component of multi-agent systems for negotiation, problem solving, and for the fusion of data and knowledge. Such systems implement a dialectical reasoning process by determining

whether a proposition follows from certain assumptions, analyzing whether some of those assumptions can be disproved by other assumptions in our premises. In this way, an argumentation system provides valuable help to analyze which assumptions from our knowledge base were really giving rise the inconsistency and which assumptions were harmless. Argumentation systems typically refer to two kinds of knowledge: *strict* and *defeasible* knowledge. Strict knowledge ( $K_S$ ) corresponds to the knowledge which is certain; typical elements in  $K_S$  are statements or undisputable *facts* about the world, or mathematical truths (e.g. implications of the form  $(\forall x)P(x) \rightarrow Q(x)$ ). The strict knowledge is *consistent*, i.e. no contradictory conclusions can be derived from it. On the other hand, defeasible knowledge ( $K_D$ ) corresponds to that knowledge which is *tentative*, modelled through “rules with exceptions” (*defeasible* rules) of the form “if P then usually Q” (e.g., “if something is a bird, it usually flies”). Such rules model our incomplete knowledge about the world, as they can have exceptions (e.g., a penguin, a dead bird, etc.). Syntactically, a special symbol ( $\Rightarrow$ ) is used to distinguish “defeasible” rules from logical implications.

Argumentation systems allow the user to define a knowledge base  $K = K_S \cup K_D$  involving strict and defeasible knowledge. An *argument*  $A$  for a claim  $c$  is basically some “tentative proof” (formally, a ground instance of a subset of  $K_D$ ) for concluding  $c$  from  $A \cup K_S$  [10]. Arguments must additionally satisfy the requirement of *consistency* (an argument cannot include contradictory propositions) and *minimality* (by not including repeated or unnecessary information). Conflicting arguments may emerge from  $K$ ; intuitively, an argument  $A$  *attacks* another argument  $B$  whenever both of them cannot be accepted at the same time, as that would lead to contradictory conclusions. Let us consider an illustrative example:

Example 1: *Consider some basic commonsense knowledge about spiders and dangerous insects, and a particular situation a kid may be facing when reasoning about what he/she has learnt from experience. The situation will be modelled in terms of facts, and commonsense knowledge will be modelled using defeasible rules.*

*Situation (Facts – Strict Knowledge)*

[a] *The black widow is a spider. A black widow (let us call it “bw”) was found on the floor.*

[b] *The black widow “bw” looks dead.*

[c] *The black widow “bw” moves its legs when touched with a stick*

*Defeasible rules (Commonsense knowledge)*

[d] *If X is a spider, then X is dangerous.*

[e] *If X is a spider and it is dead, then I can assume that X is not dangerous.*

[f] *If X looks dead, then this is a tentative reason to believe that X is dead.*

[g] *If X looks dead but moves by itself when touched, then usually X is not dead.*

*From this knowledge base  $K = \{[a],[b],[c],[d],[e],[f],[g]\}$  different arguments may arise, some of them conflicting with each other. Let us consider some cases.*

*ArgA = {[a], [d]} is an argument for concluding that the black widow “bw” found on the floor is dangerous, as it is a spider [a] and spiders are usually dangerous [d].*

*ArgB = {[a], [b], [e], [f]} is an argument for concluding that the black widow “bw” is not dangerous, as it is a spider [a] which looks dead [b], and if something looks dead, this usually*

means that it is actually dead [f]. And if the spider “bw” is dead, then we can believe that it is not dangerous [e].

Clearly, arguments ArgA and ArgB are in conflict. They contradict each other as they lead to opposite conclusions. Note that another argument can be considered, which attacks Arg:

ArgC = {[b], [c], [g]} is an argument for concluding that the black widow “bw” is not dead, as “bw” looks dead [b] but moves by itself when touched [c], and this usually means that it is not dead [g]. So we can conclude that “bw” is not dead.

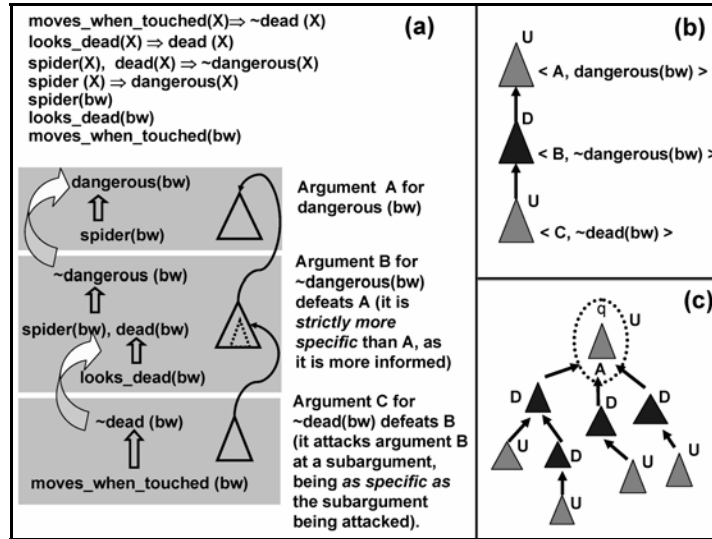
Arguments ArgB and ArgC are also in conflict, but do not have opposite conclusions. Instead, the conclusion of ArgC (“bw is not dead”) is contradicting an inner element in ArgB (“bw is dead”).

The previous example illustrates two kinds of possible “attacks” between arguments in argumentation systems: *symmetric attack* (arguments with opposite conclusions) and *undercutting attack* (an argument attacks some “subargument” in another argument). The notion of *defeat* comes then into play to decide *which argument should be preferred*. An argument A defeats an argument B whenever A attacks B, and besides, A is preferred over the attacked part in B (with respect to some preference criterion). The criterion for defeat can be defined in many ways, being a partial order  $\leq$  among arguments. Thus, for example, arguments can be preferred according to the source (e.g. when having arguments about weather, the argument of a meteorologist should be stronger than the argument of a layman). As a generic criterion, it is also common to prefer those arguments which are *more direct* or *more informed*. This is known as the *specificity principle* (see [10]). For example, if the arguments ArgA and ArgB from Example 1 are considered, both arguments are in conflict, and attack each other. However, argument ArgB is *strictly more specific* than argument ArgA. Therefore, ArgB is preferred over ArgA. In the second case, argument ArgC is *as specific as* argument ArgB, as the attacker and the attacked part are equally specific.

Fig. 1(a) illustrates the notions of knowledge base, argument and conflict between arguments associated with Example 1. <sup>1</sup>Note that arguments are usually abstracted away as triangles with the conclusion on top, and inside the defeasible rules used to reach the conclusion. Note that the notion of defeat among arguments may lead to complex “cascade” situations: an argument A may be defeated by an argument B, which in turn may be defeated by an argument C, and so on. Besides, every argument involved may have on its turn more than one defeater. Argumentation systems allow us to determine when a given argument is considered as *ultimately acceptable* with respect to the knowledge we have available by means of a *dialectical analysis*, which takes the form of a tree-like structure called *dialectical tree*. The root of the tree is a given argument A supporting some claim, and children nodes for the root are those defeaters  $B_1, B_2, \dots, B_k$  for A. The process is repeated recursively on every defeater  $B_i$ , until all possible arguments have been considered. Leaves are arguments without defeaters. Some additional restrictions apply (e.g., the same argument cannot be used twice in a path, as that would be fallacious and would lead to infinite paths).

---

<sup>1</sup> The symbol “ $\sim$ ” stands for negation [13]. Thus,  $\sim p$  accounts for the negation of p.



**Fig. 1:** overview of the main elements in argumentation systems: (a) Commonsense knowledge, arguments and conflicts among arguments (Ex. 1); (b) Dialectical analysis for “bw is dangerous” (Ex. 1); (c) Graphical representation of a dialectical tree involving many arguments

A marking procedure can be then performed for “marking” the nodes in the tree. Leaves will be “undefeated” nodes (or “U” nodes, for short), as they have no defeaters. Then we can propagate the marking from the leaves upward to the root as follows: an inner argument  $A_i$  in the tree will be marked as a “defeated” node (“D” node) if it has *at least* one “undefeated” child. Otherwise, if every child of  $A_i$  is a “D” node, then  $A_i$  will be marked as “U” node. If the root of a dialectical tree (the argument  $Arg$ ) turns out to be marked as “U” node, then it is ultimately undefeated (given the knowledge available), so that the argument  $Arg$  (and its conclusion) is said to be *warranted* (i.e. ultimately accepted). Fig. 1(b) illustrates the dialectical analysis rooted in argument  $ArgA$  with respect to Example 1. In this particular example the dialectical tree involves just one branch, as  $ArgA$  has one defeater  $ArgB$ , which in turn is defeated by  $ArgC$ . Note that the marking procedure makes  $ArgC$  to be considered a “U” node (undefeated), as it has no defeaters.  $ArgB$  is then a “D” node (defeated), as it has at least one undefeated child ( $ArgC$ ). The root of the tree ( $ArgA$ ) is a “U” node, as every of its children (i.e., the argument  $ArgB$ ) is marked as a “D” node. Consequently,  $ArgA$  and its conclusion “*bw is dangerous*” are warranted. Clearly, in complex situations the dialectical tree can actually involve many branches, each of them with several arguments, as shown in Fig. 1(c).<sup>2</sup>

Implemented argumentation systems provide valuable help for users who want to reason with incomplete and potentially inconsistent information stored in a knowledge base. Given a knowledge base  $K$ , such systems *automatically* compute the dia-

<sup>2</sup> For an in-depth treatment of defeasible argumentation (in particular the process for computing warranted arguments) see [2,13].

lectical tree associated with any particular claim (provided by the user as an input). In this context, Defeasible Logic Programming (DeLP)<sup>3</sup> is a general-purpose argumentation system which has been particularly successful in real-world applications (e.g. recommender systems [11] and decision support systems [12]), providing an integrated environment for defining a knowledge base and solving user queries (claims) interactively. For any claim the DeLP engine automatically computes and visualizes the emerging dialectical tree, which acts as an explanation facility for the user, helping him to understand why the given claim is warranted or not. As we have seen in this Section, warranted arguments support *beliefs that are accepted beyond dispute* on the basis of the knowledge available. This notion can be applied in different contexts. In particular, in *multiple-party reasoning*, where a group of several people participate on the basis of some common knowledge (e.g. students in a CSCL scenario), warranted arguments can be seen as supporting beliefs which are part of the *shared knowledge* of the group. In the next section we will analyze how this idea can be integrated in a CSCL framework to characterize part of the emerging SK and its associated SKA by making use of a general-purpose argumentation system like DeLP as a support tool.

#### **4. A Framework for Modelling SK and SKA through Argumentation as a Support Tool**

In what follows we will present a generic framework for integrating an automated argumentation system like DeLP as a support tool for dialectical discussions in a CSCL framework. Our framework will allow to model dialectical analyses carried out by participants in CSCL scenarios, helping them to identify the emerging SK and the explicit specification of its associated SKA. As a starting point we will consider the individual knowledge constructed by different students when performing a collaborative task (probably expressed in natural language and stored in a generic CSCL platform). We depart from the assumption that the knowledge required for solving the collaborative task is complex, so that students should be able to integrate different perspectives and conflicting opinions about the task to be solved.

Our goal is that participating students can make use of the reasoning and visualization capabilities provided by the argumentation system in order to support part of their SK construction as well as making explicit its associated SKA. Since knowledge is normally constructed iteratively by using several alternative tools (e.g. concept maps, virtual blackboards, etc.), in some cases the SK construction derived from our proposal can be later extended by means of these tools.

Fig. 2 shows a schematic view of the framework that characterizes the proposed model. For the sake of simplicity, in Fig. 2 we restrict ourselves to a particular group of only two students S1 and S2, although the proposal can be generalized to more participants. Our proposal is based on extending a conventional CSCL framework by adding the following elements:

---

<sup>3</sup> See [http://lidia.cs.uns.edu.ar/delp\\_client](http://lidia.cs.uns.edu.ar/delp_client).

- A *knowledge base K* which stores the individual knowledge (characterized in terms of facts and defeasible rules) constructed by each student at the beginning of the collaborative learning process.
- An automated *argumentation system* (as described in Section 3) capable of solving students' queries associated with potential claims, providing as well visualization facilities for analyzing results (e.g. visualizing the dialectical tree).
- A *Knowledge Engineer (KE)* with a solid understanding of argumentation in general and of the deployed argumentation system in particular. In many cases, with the appropriate training and support, the teacher involved in the CSCL learning process will be able to assume this role.

At the beginning, every student in the group has to exchange messages with the KE about what he/she has learnt.<sup>4</sup> Note that the KE is not necessarily an expert in the task to be solved collaboratively. Instead, his role will be translating different pieces of knowledge provided by each student  $S_i$  (in natural language) into a set  $K_i$  of facts (strict knowledge) and defeasible rules (representing defeasible knowledge the student has learnt). This set  $K_i$  is then included as part of the knowledge base  $K$ . This exchange involves an active participation of both the student and the KE, until the student is satisfied with what the KE has written. Note that by doing this, each student has to perform a metacognition activity (construction of the *Individual Knowledge* and the *Individual Knowledge Awareness*) on the basis of the disagreements he/she has had with the KE. Indeed, the student can go back and forth in the learning process to complete, reinforce or enhance his/her individual knowledge until he/she is satisfied with the representation adopted by the KE. It must be remarked that the KE is only in charge of translating the student statements into facts or defeasible rules without judging them (i.e. whether they are true or false). In addition, the KE is not contrasting the student's contribution against the information already stored in  $K$  (coming from other participants). This way, possibly inconsistent, incomplete and contradictory rules will be added to  $K$ , which will store the sum of the individual knowledge coming from every student in the group.

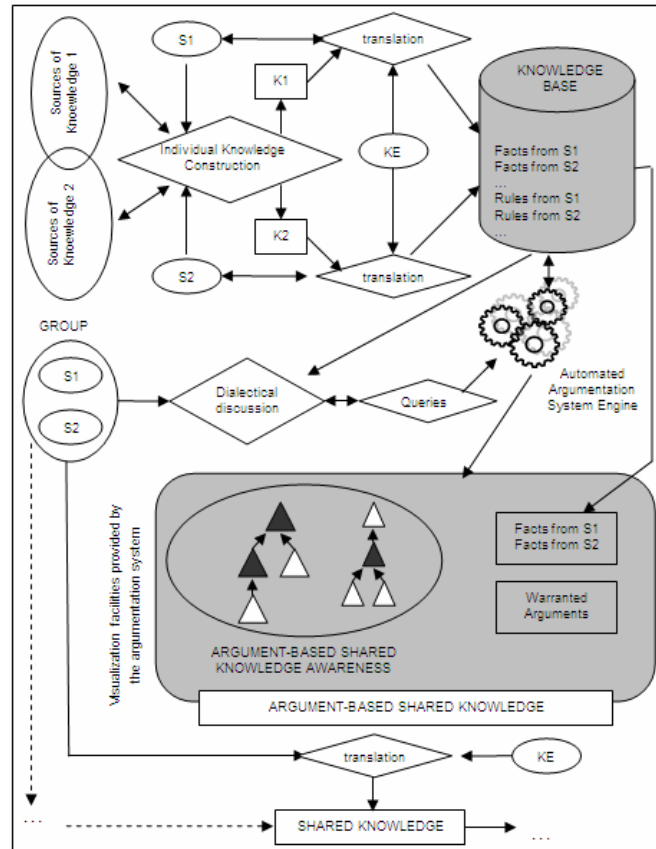
When the students meet again (oval "Group" in Fig. 2) to solve the original task  $T$ , they are allowed to see everyone's contribution in  $K$ , so that their SK about how to solve the task  $T$  is in principle the sum of the knowledge of all participants involved (expressed in terms of facts and rules). Note that the visualization of  $K$  itself provides an explicit element which models part of the current SKA of the group (elements explicating SKA are depicted in grey in Fig. 2). As usual in every CSCL learning process, students will have now to agree about their SK as a necessary step for solving  $T$ . In order to do this a dialectical discussion is performed among students, complemented by other activities (dotted lines in Fig. 2, as e.g. the joint drawing of concept maps). This dialectical discussion is aimed to achieve a consensus about the SK and tends naturally to be biased (given each student's previous knowledge and profile). For minimizing this bias and providing additional support for modelling the

---

<sup>4</sup> It must be remarked that, if necessary, the professor in charge of the group together with the KE can provide a minimal knowledge base  $K$  to help students to start the learning process.



students' dialogues, our proposal incorporates the automatic computation of arguments.



**Fig. 2:** Outline of the Framework. S1 and S2 represent two students, K1 and K2 represent the students' knowledge, and KE stands for Knowledge Engineer support.

Indeed, students will be able to analyze different claims by posing queries using the front-end from an argumentation system environment (as the one provided by DeLP). Based on *K* and by means of the provided argument-based engine, queries will allow to automatically find *warranted arguments* supporting a given claim. This way, potential disagreements among students can be analyzed, discussed and solved on the basis of a formal, objective approach which only relies on the available information stored in *K*, which stands for the sum of individual knowledge of all participants involved. As a result from this process, students will be able to identify what we call *Argument-Based Shared Knowledge* (ArgSK). As depicted in Fig. 2, this ArgSK will be formed by the sum of all individual facts in *K* (contributed by each student), as well as by all those warranted arguments derived from students' queries (as they are ultimately accepted on the basis of the sum of the students' individual knowl-

edge). Besides, the visualization of the dialectical tree associated with every warranted argument provides an additional, explicit element that characterizes what we call *Argument-Based Shared Knowledge Awareness* (ArgSKA): students are aware of how different conflicting pieces of knowledge are related to each other, why some of such pieces should be deemed as warranted (and some others should not), and how their own individual knowledge may be in conflict with other participants' knowledge. At the end of the construction of SK and SKA, ArgSK can be translated again to natural language sentences with the help of the KE, if necessary. This translation will help students to provide an adequate solution to the original task  $T$  to be collaboratively solved. Finally, the translated ArgSK and the corresponding ArgSKA evidence (visualization of dialectical trees) can be added to the global SK of the group, which will use these new rationally justified (and consequently unbiased) knowledge to solve  $T$  by carrying out the next steps of the CSCL process.

## **5. Proof of Concept: a case study**

This section summarizes a Proof of Concept to show the workability and feasibility of the framework proposed in Section 4. First, a feasibility analysis is presented to ensure that our proposal can be deployed in CSCL frameworks in a useful manner within reasonable costs. Second, a case study is presented to illustrate how the framework works in a concrete situation.

### **5.1. Feasibility Analysis**

Our proposal requires an automated argumentation system to be integrated in the CSCL platform, which should include an appropriate front-end for posing queries, and facilities for defining a knowledge base and visualizing results of the computation of the underlying argumentation engine (e.g. dialectical trees). As already mentioned in Section 3, several of such kinds of platforms are freely available nowadays [13, 14], providing appropriate software tools to compute ArgSK and ArgSKA with reasonable costs (including economical resources, time consumption, etc.). Thus, automated argumentation systems can be seen as a first step on the construction of argument-based modules to be completely embedded in a particular CSCL environment.

Clearly, costs associated with the inclusion of a Knowledge Engineer (KE) along the use of the proposed framework must be also considered. In that respect, it must be noted that existing argument-based platforms are not specially oriented towards experts on argumentation, but rather towards general users with a conceptual understanding about the meaning of facts, rules and inference by means of rule chaining. Such concepts are intuitive and suitable for students to learn and use (assuming they have basic abstract thinking abilities). Thus, the coordinator of the CSCL proposal (e.g. the teacher) can be also expected to act in the role of the KE, translating sentences from natural language into defeasible rules and facts (required to construct the knowledge base) while keeping an active dialogue and participation from the students. The existence of graphical front-ends included in some argumentation plat-

forms [14] minimizes the complexity of text input for rules and facts as well as the interpretation of obtained results. Finally, from the CSCL viewpoint, it must be remarked that the integration of Artificial Intelligence techniques and CSCL has proven to be fruitful, resulting in systems such as I-MINDS [15] or SCALE [16].

## 5.2. A case study

Consider the following case study: Computer Science students from three different universities  $U_1$ ,  $U_2$  and  $U_3$  (located in different cities) have to solve an activity collaboratively in a CSCL scenario. The activity is structured using the *JIGSAW* technique [17] and includes the task  $T$  of *detecting good and bad features in different configurations of a personal computer model called "pcu" (acronym for "PC for universities")*, which is the computer model available in the computer labs of the three universities (e.g. the three labs have pcu's with the same configuration, devices, etc). The students are divided into small groups of three people, each of them belonging to a different university. Following the *JIGSAW* technique, each member of the group will be responsible for analyzing a different piece of knowledge when constructing his/her Individual Knowledge. Let us focus on one jigsaw group  $G$  formed by three students, namely  $S_1$ ,  $S_2$  and  $S_3$ . As stated before, we will assume that  $S_1$ ,  $S_2$  and  $S_3$  are using a particular CSCL system to solve  $T$ , as they are located in different cities. For the sake of example the students must learn about different topics related to pcu's as follows: 1)  $S_1$  is assigned the topic "input/output devices"; 2)  $S_2$  is assigned the topic "memory devices"; and 3)  $S_3$  is assigned the topic "processors". Let us assume that students have already constructed their Individual Knowledge and they are coming back to the group  $G$  to solve  $T$ . At this moment,  $S_1$ ,  $S_2$  and  $S_3$  have to present a well-organized report to the others members of  $G$  about the topic each of them has studied. The immediate goal is to construct SK and SKA in order to solve  $T$ . As a part of their SK and SKA, students are offered to construct their ArgSK and ArgSKA. Following our proposal, an automated argumentation platform is integrated with the CSCL scenario. It includes a knowledge base  $K$  (empty at the beginning), an inference engine for computing arguments and a suitable front-end for posing queries and visualizing results. Besides, a Knowledge Engineer (*KE*) will help  $S_1$ ,  $S_2$  and  $S_3$  to translate their individual knowledge into facts and defeasible rules.

First, each student  $S_i$  exchanges (separately) messages with the KE about what he/she knows, and the KE writes down this in terms of rules and facts. Following our example, suppose that student  $S_1$  has acquired knowledge about printers (as they are I/O devices). He/she has learnt the following:<sup>5</sup> *hp1020 and hp1018 are usually abbreviations of laser printers. Laser printers work ok if the computer has a good RAM memory. Inkjet printers work usually ok with any kind of computers.* Besides,  $S_1$  has checked the computer model "pcu" (the object of study) and has seen that there was a printer connected, namely the hp1020. In the same way,  $S_2$  has studied memory devices. He/she has learnt that *a RAM memory of 256 Mb or more is usually good enough for a computer, unless you want to use it with a laser printer, since in such a*

---

<sup>5</sup> Names and values used here are fictitious. They are just considered for the sake of the example and not necessarily according to a real-world situation.

case a RAM of 256 KB has slow access, which is usually not a good feature. In addition,  $S_2$  has checked the computer model “pcu” and has seen that the computer had 256Mb of RAM memory (note that  $S_2$  doesn’t know anything about processors or printers, he just knows that they appeared as related concepts when learning about memory devices). Concerning  $S_3$ , he/she has individual knowledge about processors. He/she has learnt that *if a computer has a processor double-core, then the processor is usually fast. Pentium processors result in slow access time for RAM memory. An exception are Pentiums with the special swap technology, which do not have this problem.* He/she has checked the computer model “pcu” and has seen that it has a Pentium processor with “swap technology”. At the end of all the dialogues between  $S_1$ ,  $S_2$  and  $S_3$  with the KE, the knowledge base  $K$  stores the sum of the three students’ individual knowledge, which could have been written down by the KE as follows:

*Facts about the Object of Study under Analysis (Strict Knowledge)*

*% Facts about the computer in the lab*

- *printer(pcu, hp1020) % fact from student S1*
- *has\_ram (pcu,256) % fact from student S2*
- *processor (pcu,pentium)% fact from student S3*

*Defeasible rules (Commonsense knowledge) %C stands for an arbitrary computer*

*% Knowledge about printers coming from  $S_1$*

- *printer(C, hp1020)  $\Rightarrow$  printer(C, laser)*
- *printer(C, hp1018)  $\Rightarrow$  printer(C, laser)*
- *ram\_memory(C,good) and printer(C, laser)  $\Rightarrow$  printer\_ok(C)*
- *printer(C, inkjet)  $\Rightarrow$  printer\_ok(C).*

*% Knowledge about RAM memories coming from  $S_2$*

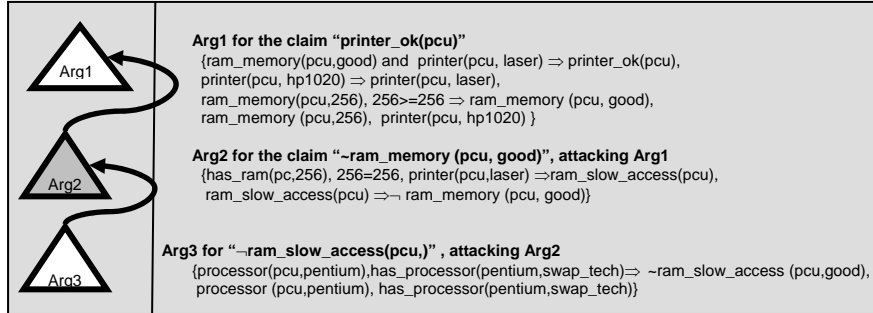
- *has\_ram (C,X),  $X \geq 256 \Rightarrow$  ram\_memory (C, good)*
- *has\_ram (C,X),  $X = 256$ , printer(X,laser)  $\Rightarrow$  ram\_slow\_access(C)*
- *ram\_slow\_access (C)  $\Rightarrow$   $\sim$ ram\_memory (C,good)*

*% Knowledge about processors coming from  $S_3$*

- *processor(C,double\_core)  $\Rightarrow$  processor(C, fast)*
- *processor(C,pentium)  $\Rightarrow$  ram\_slow\_access (C)*
- *processor(C,pentium), has\_processor(pentium,swap\_tech)  $\Rightarrow$   $\sim$ ram\_slow\_access (C)*

Now, consider that as part of task  $T$  to solve (detecting good and bad features in different configurations of a “pcu”),  $S_1$ ,  $S_2$  and  $S_3$  are discussing about the piece of knowledge “*printer\_ok(pc)*”, which stands for the claim “*is it ok to have a printer connected to the computer pcu?*”. By analyzing the individual knowledge provided by each  $S_i$  separately, the members of  $G$  cannot infer anything (except from the facts provided). However, if they jointly consider all the information stored in  $K$  (which accounts for part of their SK) they can rely on the automated argumentation system to automatically compute a dialectical tree rooted in the above claim, which will include all possible combinations of arguments and defeaters related to the claim. This way, they can guarantee that those pieces of knowledge subject to dialectical discussions will be part of the SK only they are warranted on the basis of the joint knowledge of the group, thus avoiding dialectical discussion based on incomplete and biased perceptions of reality. Hence, if the claim results to be supported by a warranted argu-

ment, then the above piece of knowledge can be added to the ArgSK of  $G$  on the basis of rational and justified information.



**Fig. 3:** Outline of the dialectical tree obtained for the claim “*is it ok to have a printer connected to the computer pcu?*”. **Left:** Dialectical tree (white triangles represent warranted arguments, and grey triangle represent defeated arguments). **Right:** the argument contents provided by the argumentation engine.

In this particular example,  $S_1$ ,  $S_2$  and  $S_3$  will obtain a warranted argument supporting the claim “*printer\_ok(pcu)*” (as the warranted argument  $Arg1$  supports it), and they will add the claim to their ArgSK. Note that the claim is deemed as warranted by the underlying argumentation system, based on the dialectical tree shown in Fig. 3. Besides,  $S_1$ ,  $S_2$  and  $S_3$  will visualize the dialectical tree shown in Fig. 3 (left), which explicitates a rational justification of the obtained results. Indeed, it can be seen that there exist an argument  $Arg1$  supporting “*printer\_ok(pcu)*”, which can be obtained by combining knowledge from  $S_1$  and  $S_2$ . The argument is based on knowing that “*pcu*” has enough RAM memory to support hp1020, the laser printer connected to it. However,  $Arg1$  is defeated by  $Arg2$ , which supports the claim “*~ram\_memory (pcu, good)*” (the student who studied memory devices provided a defeasible rule which states that 256 Mb usually do not suffice for a laser printer to run ok). But this argument  $Arg2$  is on its turn defeated by  $Arg3$  standing for “*~ram\_slow\_access(pcu)*” (the student who studied processors provided a defeasible rule which states that computers with Pentium processors with swap technology, as it is the case here, do not have problems with RAM of 256 Mb). This way, the visualization of the tree will be linked to the ArgSKA associated with the claim under consideration, helping  $S_1$ ,  $S_2$  and  $S_3$  to be aware of their own SK. Later on,  $S_1$ ,  $S_2$  and  $S_3$  will be able to use the piece of warranted knowledge (the fact that the printer connected will work ok) when going further on the resolution of  $T$ .

## 6. Related Work

SK and SKA are key concepts when characterizing knowledge in CSCL scenarios. Indeed, different proposals have been presented to model knowledge acquisition in collaborative learning. For example, Kollar et al. introduced some methods to explicit

knowledge acquisition in CSCL by scripting [18]. Another example is shown in [19], where a Knowledge-Building Environment (a software platform intended to support collaborative learning) is proposed. Indeed, the process model of knowledge-building presented in [19] provides a conceptual framework for the design, use and assessment of collaborative systems. However, to the best of our knowledge, there are no other approaches to extend a generic CSCL platform by incorporating automatic computation of SK as presented in this paper.

Concerning knowledge awareness, Collazos et al. have defined the notion of SKA [1]. In addition, Ogata and Yano have proposed a mechanism to model it based on the activities performed by the students [20]. Indeed, in [20] an information filtering process is performed, referring to both finding desired information (filtering in) and eliminating that which is undesirable (filtering out). Another relevant contribution is shown in [21], where graphical representations are suggested to foster externalized cognition and enhance the cognitive processes of managing knowledge and information in resource-based learning and problem solving environments. In this case, knowledge and information visualizations have been shown to be effective for enhancing the cognitive processes of learning [22]. Particularly, digital concept maps seem to have a potential as cognitive tools to enhance individual and group-related cognitive processes in resource-based learning and problem solving [22]. However, to the best of our knowledge, there are no other proposals for automated construction of explicit SKA as presented here.

Finally, it must be remarked that recent research has led to some interesting results to model dialectical discussions and negotiation in CSCL scenarios. For example, [23] discusses the understanding and participation in scientific discourses. Gervasi et al. [24] have proposed a mechanism to manage dialectical discussions when a group of people want to collaboratively define requirements in natural language. However, their proposal is based on constraint satisfaction techniques, and does not consider the use of argumentation. [25] and [26] have considered the use of argumentation and analytic reasoning on CSCL systems. However, these approaches are focused on argumentation as a meta-cognitive activity, not taking into account the automatization of argumentation nor aiming at the explicit construction of SKA, as proposed in this paper. A very interesting research work with some similarities to our approach is presented in [27], where the notion of negotiating about shared knowledge in CSCL is defined. A quantification of shared knowledge (called “degreement”) is provided. Students can negotiate their beliefs about facts with the teacher, characterizing their own “perspective” as a result of this process. Interestingly, [27] mention also “argument exchange” as part of the SKA process. However, their approach does not rely on the use of a formal argumentation system, as in our case.

## **7. Conclusions. Future Work.**

Shared Knowledge (SK) and its related Shared Knowledge Awareness (SKA) are key concepts to ensure an appropriate performance when solving activities collaboratively in a CSCL scenario. In this paper we have presented a novel approach for integrating automated argumentation systems as a support tool for characterizing SK and SKA in

CSCL scenarios. Taking as starting point dialectical discussion of students performed in natural language when constructing SK and SKA, we have shown how argumentation systems can be used to model SK in terms of a general knowledge base which includes the individual knowledge of every participant.

The notion of warranted argument and its automated computation and visualization by means of dialectical trees provides a way of characterizing what we have called ArgSK and ArgSKA. The generic framework presented here has been tested in a prototypical version under different situations (all of them involving the resolution of a given task by collaboration), using DeLP as underlying argumentation system. At the current stage of research the obtained results account for a proof of concept of our proposal, and a full-fledged implementation (integrating CSCL and argumentation) is under development for evaluating more complex situations.

As the notions of SK and SKA are not defined using a logical formalization (as is the case for argumentation), part of our future work will be focused on establishing the scope of this proposal in real CSCL scenarios. In particular, we are interested in those cases involving a considerable volume of information or a large number of students, where the efficiency of the process required to construct the arguments is a critical problem to be solved. Research in this direction is currently being pursued.

*Acknowledgments:* We thank the reviewers for their comments which helped to improve the final version of this paper. This research was partially supported by projects 24/ZN10 (Secretaría de Ciencia y Tecnología, UNS, Argentina), PICT 13096, PICT 15043 and PAV076 (ANPCyT, Argentina); ADACO TIN2004-08000-C03-03 (Spain); IEA TIN2006-15662-C02-01 (Spain); Colciencias 4128-14-18008 (Colombia) and Colciencias 030-2005 (Colombia).

## References

- [1] Collazos, C., Guerrero, L., Pino, J., and Ochoa, S. Introducing Knowledge-Shared Awareness. Procs. of IASTED'02, USA, pp.13-18 (2002).
- [2] Chesñevar, C. I.; Maguitman, A., Loui, R. Logical Models of Argument. ACM Computing Surveys, 32(4) pp. 337-383 (2000).
- [3] Kirschner, P.; Buckingham S.; Carr, C. (Eds.). Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making. Springer-Verlag, London (2003).
- [4] Parsons, S. ; Maudet, N. ; Moraitis, P. ; Rahwan, I (Eds). Argumentation in Multi-Agent Systems. 2<sup>nd</sup> Int. Workshop ArgMAS'05. LNCS Series, Springer (2006).
- [5] González, M. P.; Collazos, C. A.; Granollers, T. Guidelines and usability principles to design and test Shared-Knowledge Awareness for a CSCL interface. LNCS Series, vol. 4154, pp. 102-117. Springer Verlag (2006).
- [6] Maisonnasse, O. Brdiczka, N. Gourier, P. Reignier. Attentional Model for Perceiving Social Context in Intelligent Environments. 3rd IFIP Conference on Artificial Intelligence Applications and Innovations (IAI). Springer, pp.. 171–178 (2006).
- [7] Janssen, J.; Gijssbert, E.; Kanselaar, G. Visualization of agreement and discussion processes during computer-supported collaborative learning. Comput. Hum. Behav, vol 23 (3), pp. 1105-1125. Elsevier Science Publishers B. V (2007)

- [8] Cadiz, J.; Gupta, A.; Grudin, J. Using Web annotations for asynchronous collaboration around documents. Proc. of the ACM CSCW'2000 Conf. on Computer-Supported Cooperative Work. ACM Press, pp.309-318 (2000).
- [9] Severinson Eklundh, K.; Rodriguez, H. Coherence and Interactivity in Text-Based Group Discussions around Web Documents. Proc. 37th Int. Conf. on System Sciences (HICSS'04), IEEE Computer Society, pp. 1530-1605 (2004).
- [10] Simari, G.; Loui, R. A Mathematical Treatment of Defeasible Reasoning and its Implementation. Artificial Intelligence v. 53, pp 125-157 (1992)
- [11] Chesñevar, C.; Maguitman, A.; Simari, G. Argument-Based Critics and Recommenders: A Qualitative Perspective on User Support Systems. Data & Knowledge Engineering, Elsevier, vol.59, Issue 2, pp. 293-319 (2006).
- [12] Brena, R.; Aguirre, J.; Chesñevar, C.; Ramirez, E.; Garrido, L. Knowledge and Information Distribution Leveraged by Intelligent Agents. Knowledge and Information Systems (KAIS), Springer Verlag (in press), (2007).
- [13] García, A.; Simari, G. Defeasible Logic Programming: An Argumentative Approach. Theory and Practice of Logic Programming, 4(1) pp. 95-138 (2004).
- [14] Reed, C.; Rowe, G. Araucaria: Software for Argument Analysis, Diagramming and Representation. Int. J. on Artificial Intelligence Tools, Vol. 13(4), pp. 961-979 (2004)
- [15] Liu, X.; Zhang, X.; Soh, L.-K.; Al-Jaroodi, J.; Jiang, H. A Distributed, Multiagent Infrastructure for Real-Time, Virtual Classrooms, Proc. ICCE'2003, pp. 640-647 (2003)
- [16] Soller, A.; Guizzard, R.; Molani, A.; Perini, A. SCALE: supporting community awareness, learning, and involvement in an organizational learning environment. Proc. of the 6th international conference on Learning sciences, pp 489-496 (2004)
- [17] Aronson, E.; Blaney, N., Stephin, C., Sikes, J., & Snapp, M. The jigsaw classroom. Beverly Hills, CA: Sage Publishing Company. (1978).
- [18] Kollar, I., Fischer, F., & Slotta, J. D. Internal and external collaboration scripts in web-based science learning at schools. In T. Koschmann, D. Suthers, & T. -W. Chan (Eds.), CSCL 2005: The Next 10 Years (pp. 331-340). Mahwah, NJ: Lawrence Erlbaum (2005).
- [19] Stahl, G. A Model of Collaborative Knowledge-Building. In B. Fishman & S. O'Connor (Eds.). 4th Int. Conf. of the Learning Sciences, pp. 70-77. Erlbaum (2000).
- [20] Ogata, H., Yano, Y., Combining knowledge awareness and information filtering in an open-ended collaborative learning environment. Int. Journal of Artificial Intelligence in Education, Vol. 11, pp. 33-46 (2000)
- [21] Cox R. Representation construction, externalised cognition and individual differences. Learning and Instruction, 9, 343-363 (1999).
- [22] Tergan, S.-O., & Keller, T. (Eds.) Knowledge and information visualization: Searching for synergies. LNCS vol. 3426. Springer Verlag, New York (2005).
- [23] Driver, R., Newton, P., & Osborne, J. Establishing the norms of scientific argumentation in classrooms. Science Education, 84(3), 287-313 (2000).
- [24] Gervasi, V.; Zowghi, D. Reasoning about inconsistencies in natural language requirements. ACM Trans. Softw. Eng. Methodol., vol 14 (3), pp 277-330 (2005).
- [25] Yeung, D.; Lowrance, J. Computer-Mediated Collaborative Reasoning and Intelligence Analysis. LNCS vol. 3975. Proc. Intl. Conference on Intelligence and Security Informatics, pp. 1-13, S.Diego, USA, Springer-Verlag (2006).
- [26] Weinberger, A., Fischer, F. A framework to analyze argumentative knowledge construction in computer-supported collaborative learning. Computers&Educ., vol.46, pp.71-95 (2006).
- [27] Pfister, H.; Wessner, M.; Holmer, T.; Steinmetz, R. Negotiating about Shared Knowledge in a Cooperative Learning Environment. In Proc. of the CSCL Conf, USA, pp. 454-457 (1999).