Providing Relevant Knowledge in Disputes: UMCourt Project

Davide Carneiro[1], Ângelo Costa[1], Paulo Novais[1], Francisco Andrade[2], José Neves[1]

[1] Department of Informatics, University of Minho, Braga, Portugal
{dcarneiro, pjon, jneves}@di.uminho.pt
[2] School of Law - University of Minho, Braga, Portugal
fandrade@direito.uminho.pt

Abstract. Parties involved in disputes often lack the information they need to take rational decisions. As a consequence, they frequently enter into agreements that are not as advantageous as they could be. Having the right information in the right time would guide parties into taking more weighted and realistic decisions. Specifically, parties should consider their best, worst and most likely outcomes in litigation as well as all the possibilities in between. In this paper the importance of this information is highlighted and domain-dependent methods for compiling it are presented. Moreover, this work describes three case studies in which these methods are being applied with the objective of informing the parties, empowering their role in the dispute resolution process and helping them achieve more satisfactory outcomes.

Keywords: Online Dispute Resolution, Case-based Reasoning, Rule-based Systems, UMCourt.

1 Introduction

Conflicts arise in the most different scenarios and are present in our day-to-day since our first years of life. Essentially, conflicts are due to the competitive nature of our society, in which each individual wants to maximize his personal gain. A conflict is generally defined as an opposition of interests between two persons. When two persons with opposing interests clash, a dispute arises, that will eventually need to be settled. Each of these persons has ideas and values of its own that will guide and support his actions throughout the dispute resolution process.

Until just a few years ago, these conflicts occurred mainly between two persons that where on the psychical presence of each other. However, given the new information society that we now live on, this is no longer necessarily true. In fact, on-line activities, such as the use of e-commerce sites amazon.com and ebay.com, have led to the development of on-line disputes. We argue that if a transaction occurs online, then disputants are likely to accept online techniques to resolve their disputes. Thus, the development of e-commerce requires new ways of resolving conflicts that avoid courts. Different forms or methods of alternative dispute resolution for electronic environments have been pointed out by legal doctrine. Thus being, we can now speak of Online Dispute Resolution (ODR) as any method of dispute resolution in which wholly or partially an open or closed network is used as a virtual location to solve a dispute [1]

A relevant issue, in a first moment, will be to inquire in what way (and to what point) traditional mechanisms such as negotiation [2], mediation [3] or arbitration [4]can be transplanted or adapted to the new telematic environments, taking advantage of all the resources made available by the newest information and communication technologies. However, technology can be used for much more important tasks such as strategy definition, information retrieval, solution proposal, among others. In order to develop such intelligent and efficient techniques to support Online Dispute Resolution, we must also consider the integration of Artificial Intelligence with Online Dispute Resolution [5]. This knowledge can be considered from two different perspectives: on the one hand, as a tool to help the parties and the decision makers to obtain the best possible results in solving commercial disputes and, on the other hand, considering a new way of autonomous dispute resolution through the use of autonomous and intelligent software, supported by a knowledge base and decision capabilities.

The work presented here develops around these two main ideas. We will therefore analyze to what extent technology can be used to help parties achieve more satisfactory solutions. Specifically, we will look at several methods for efficient and contextualized information retrieval as a way to provide meaningful information that is not available in traditional procedures. Moreover, we will look at three novel ODR prototypes developed with the objective of fastening and making more efficient dispute resolution processes. These prototypes are supported by UMCourt, an agent-based architecture that supports the development of ODR services [6].

Throughout this paper we will be guided by a main idea: there is not one single technological solution that can address all the problems. In that sense, techniques and procedures should be chosen and adapted according to each specific legal domain.

2 UMCourt

On-line dispute resolution methods can provide easy, efficient and fast ways for resolving disputes, contrary to the judicial path which is usually expensive and time consuming. First and second generation ODR [7], with agents performing relevant parts of the agreement procedure can be of inestimable use for the parties. UMCourt is a project being developed at University of Minho in the context of the TIARAC project (Telematics and Artificial Intelligence in Alternative Conflict Resolution) that aims to develop tools to help parties involved in legal disputes. Currently, four domains are object of study: labor law, consumer's law, divorce and heritage's share and conflicts in Virtual Organizations. In that sense, a high level architecture is being developed that can be used in the different legal domains.

This architecture is built on the notion of intelligent agent [8]. Thus, it builds on a group of autonomous software entities that are able to proactively make decisions and cooperate in order to achieve the objectives. Specifically, we are working with Jade platform which provides several interesting agent management and communication services [9]. In order to be able to address different legal domain with as much functionality reuse as possible, a development strategy was followed that organizes the agents of the architecture into two categories: high level agents and low level agents. High level agents perform tasks that do not need explicit domain-dependant information. Low level agents are closer to the legal domain, thus have methods for representing domain-dependant information and procedures. In a general way, high level agents coordinate low level ones, i.e., the first tell the seconds the steps to follow in order to implement a given behaviour. Low level agents then have the knowledge that allows them to decide what to do in each step, according to the domain of the tasks. This knowledge is formalized in ontologies that encode the domain concepts, actions, constraints and rules.

This is possible to do because there are procedures that are independent of domain. Let us take as example a negotiation process. This generally consists on several consecutive rounds in which each agent states an opinion about the proposal currently on the table. And this is independent of the domain of the negotiation. Thus, high level agents guide the process and determine when a new round should start, the turn of the several agents or when the process should finish. Each low level agent then has the autonomy to choose among the several actions in the knowledge base, the one that corresponds to the current domain of the negotiation.

Following this approach increases functionality reuse and allows to have a single architecture supporting services in a wide range of domains. It also simplifies the task of adding a new legal domain. In fact, in order to do so, it is generally only necessary to develop the ontology of the new domain, with all the actions, rules, constraints and specific concepts. This will tell the low level agents how to act when they receive a task from this new domain.

Let us now move to a close description of the architecture. At the moment, it implements two high level functionalities: a case-based reasoning (CBR) [18] algorithm and a negotiation one. The CBR functionality enables a wide range of services used mainly to inform the parties based on other similar cases. The negotiation functionality allows two or more agents to exchange messages in order to iteratively modify a proposal for a solution until an agreement is reached.

All the agents and their roles are depicted in Table 1. These agents were defined after the specific requirements of this project, following an iterative cut-down process of increasing specification. Nevertheless, given the open nature of the architecture, it can be easily extended by adding new agents or ontologies, namely to address other legal domains or to implement new functionalities in a domain.

Table 1. The agents that build up the architecture and their roles.

	Agent	Role
high Level	Learning	Contains the rules for updating the memory structures in order to reflect the results of a learning process (e.g. new case, change in values of case).
high Level	Retriever	Retrieves the cases more similar to a given one. This agent has the autonomy to change the search settings, the similarity parameters and the retrieve algorithms in order to perform a better selection of cases.
	Reuse	When requested by the Coordinator, performs the necessary

		actions to adapt a given case so that it can be used.
	Reviser	Looks at a group of cases in order to select an outcome/solution for a given case. Proposes the outcome to the coordinator as well as a justification and waits for the outcome. If the outcome does not comply with the one suggested provides a list of more probable reasons for the failure.
low level	Case Loader	Agents request case information to this agent. This agent provides the cases by interacting with the Parser agent. Each case that is requested is maintained in memory so that, should they be requested again, they are rapidly available in memory and do not need to be parsed again. This agent also checks constantly for changes in the files of the cases that are in memory in order to maintain them synchronized.
	Roles	Contains a list of agents and the actions that they are authorized to perform.
	Database	Is responsible for implementing services to implement all the interactions with the database. It is also responsible for the security and integrity of the database, making sure that only the correct agents perform the authorized operations.
	FSA	Contains a list of Jade FSM behaviors that describe the guidelines or steps necessary for an agent to perform defined actions.
	Indexer	Indexes new cases in the Database and creates the specific xml files in the correct system folders.
	Parser	Verifies the validity of XML files against the defined schemas. Valid cases are parsed and returned as a Java Object that can be handled by other agent.
	Rules	Embodies rules of type *if condition then action* that provide basic reactive actions for guiding agents in the decision making processes. Can be used to model a rule-based legal domain.
	Selector	Multiple instances of this agent exist that implement different pre-selection algorithms. Some have already been implemented, such as the Template Retrieval while others (e.g. Clustering algorithms) are now being developed.
	Settings	Defines several search and similarity settings according to which retrieve parameters can be changed.
	Similarity	Multiple instances of this agent embody different similarity algorithms.
	Interface Agents	This agent represents a group of agents that have as task to gather information from html forms. Then, they compile that information and forward them according to the defined format to the agents in the backend. As each html page in the frontend generally has, at least, one agent and they perform relatively simple tasks, we will not detail them further here.
	Extension	This agent receives requests from external agents and forwards them to the correct agents inside the platform, provided the external agents are trying to perform an action that they are authorized for.
control	Coordinator	Receives task requests from other agents (e.g. external agents, interface agents) and take the necessary steps (requesting tasks to other agents) in order to perform them. This agent maintains a list of active tasks and has access to a list of finite state

	automata that define the next action for each task, provided by the FSA agent.
Fault Manager	Starts all the necessary agents according to an editable XML configuration file. It then constantly checks the agent registry and, if any agent has crashed, restarts another instance of that agent.
Load Balancer	This agent controls the pending requests to specific secondary agents and starts new instances of agents that have a significant amount of work load.

The organization of this agents is depicted in Figure 1. In this figure, rounded rectangles represent agents and the lines represent the main communication paths. Note the existence of the DF agent (Directory Facilitator), an agent that makes part of the Jade platform and provides support for service registry and lookup.

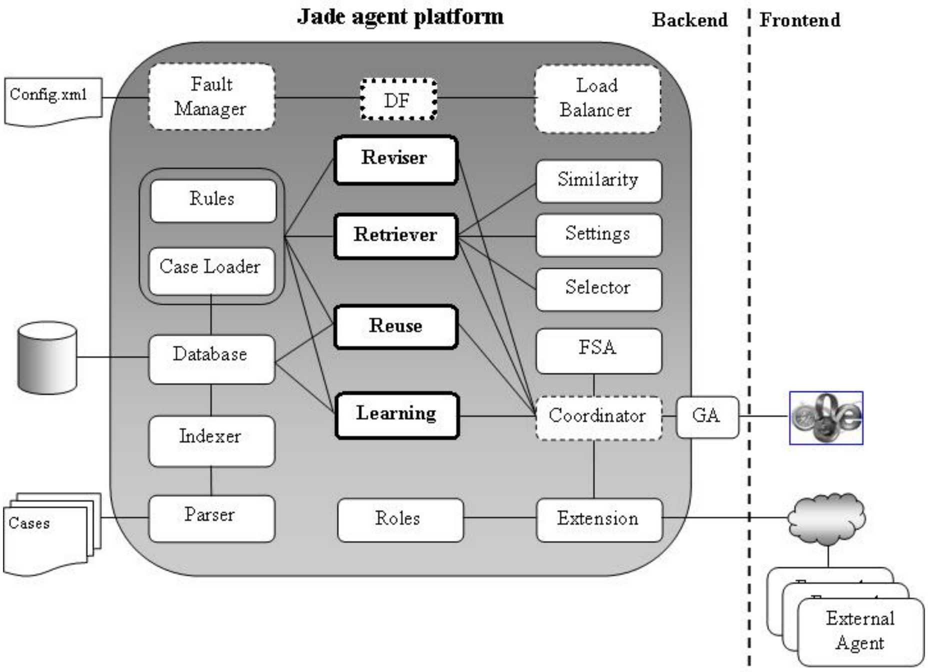

Fig. 1. The organization of the agents that make up the UMCourt architecture.

3 Compiling Relevant Knowledge

The ideal dispute resolution process is one in which the two parties are better at the end than they were at the beginning. Unfortunately, not all disputes have this conclusion. In order to improve this, we believe that it is of ultimate importance to: (1) provide the parties with more knowledge about the dispute and (2) potentiate the role of the parties throughout all the process. In fact, parties that have poor access to important information generally end making bad choices or, at least, they hardly make the best choice. Moreover, parties usually have a reduced role on the resolution process, resulting in suspicion about the outcome, mostly because they do not understand how it was achieved. Thus, in this section we describe the approach followed in the development of UMCourt in order to attack these problems.

3.1 What Knowledge is Significant?

The first step is thus to identify the knowledge that is meaningful for the parties, according to the legal domain of the dispute, and then determine the methodologies suited for compiling that knowledge. On a first instance, it would be interesting for a party to determine to which extent is it reasonable to engage in a dispute resolution process. That is, are there significant advantages against a traditional litigation? This question can be analyzed from several points of view. On the one hand, alternative dispute resolution processes are generally faster, cheaper, more private and personalized [3]. There is however another important factor: the possible outcome reached through each of the processes. That is, will I reach a better outcome using an alternative dispute resolution process instead of litigation?

It would thus be really important for each party to know its BATNA - Best Alternative to a Negotiated Agreement, or the possible best outcome "along a particular path if I try to get my interests satisfied in a way that does not require negotiation with the other party" [10]. A party should then understand the notion of a BATNA and what role it should play in ODR. Doing so will, at least, contribute to the acknowledgement that an agreement may be disadvantageous [11]. In fact, the position of the parties may become much more unclear if they are not foreseeing the possible results in case the negotiation / mediation fails. "If you are unaware of what results you could obtain if the negotiations are unsuccessful, you run the risk of entering into an agreement that you would be better off rejecting or rejecting an agreement that you would be better off entering into" [12]. That is to say, the parties, by determining their BATNA, would on one side become "better protected against agreements that should be rejected" and, on the other side, they would be in a better condition to "reach an agreement that better satisfy their interests" [13]. But, besides that, a BATNA may play additional interesting features for the parties. For instance, it may be used as a "way to put pressure on the other party", especially in dispute resolution procedures allowing the choice of going to court [13].

However, the use of the BATNA alone is not enough to take informed decisions as parties often tend "to develop an overly optimistic view on their chances in disputes" [13]. This may lead parties to calculate unrealistic BATNAs, which will influence later decisions, leading even to either reject generous offers from the other parties, or to stand stubbornly fixed in some positions [13]. It is thus important to also consider the concept of a WATNA, or the Worst Alternative to a Negotiated Agreement [10, 14, 15]. A WATNA intends to estimate the worst possible outcome along a litigation path. It can be quite relevant in the calculation of the real risks that parties will face in a judicially determined litigation, imagining the worst possible outcome for the party.

At this point, a party would be aware of the best and worst scenario if the dispute is to be solved in a court.

However, it could also be interesting to consider the whole space between the BATNA and WATNA as a useful element to be taken into account for making (or accepting) a proposal. If we consider for instance, in the labor law domain, the scenario of a worker being fired, litigation will most likely occur. Under many legal systems, a huge deal of legal parameters have to be considered, including antiquity, supplementary work, just cause for dismissal, among others. For the worker, the amounts involved are not irrelevant: being fired without good indemnities may be seen as a double sacrifice. But he might, on the other side, receive significant financial compensation. In order to clearly see the advantages of a proposed agreement, parties should thus also consider the spectrum between their BATNA and their WATNA. Of course, the less space there is between BATNA and WATNA, the less dangerous it becomes for the party not to accept the agreement (unless, of course, their BATNA is really disadvantageous). A wider space between BATNA and WATNA would usually mean that it can become rather dangerous for the party not to accept the ODR agreement (except in situations when the WATNA is really not inconvenient at all for the party). We can thus argue that knowledge about the space between the BATNA and the WATNA is also very important. This space is evidently related to the Zone of Possible Agreement proposed by Raiffa (1982) [16]. It is the zone where an agreement can be met that is acceptable to both parties.

Moreover, it would be interesting for a party to understand the region of this space in which an outcome is more likely. That is, if the parties are to solve the dispute through litigation, what is the most likely outcome? In fact, sticking only with the BATNA and WATNA may not be realistic as these are usually not the most likely outcomes but merely informative values that establish boundaries. Thus, an informed party should also consider the MLATNA – Most Likely Alternative to a Negotiated Agreement [15]. Using the same arguments, we can also conclude that the existence of metrics that measure the probability of each possible outcome could also be extremely useful for a party.

Concluding, in order for a party that is engaging in an alternative dispute resolution to take informed and rational decisions, he should consider knowledge about: (1) the best possible outcome in litigation; (2) the worst possible outcome in litigation; (3) the space between the two previous values; (4) the most likely outcome in litigation and (5) the probability of each outcome within the zone of possible agreement.

3.2 Domain-dependent Methods for Compiling Knowledge

Having seen the knowledge that a party should have in order to take rational decisions, let us now depict the methods for compiling that knowledge. Considering the BATNA and the WATNA values, we are using a rule-based approach. Rule-based Systems (RbS) are generally the simplest way of implementing intelligent behaviors [17]. Thus, RbS are a way to store, interpret and manipulate knowledge about a given domain (data and procedures) in the form of IF-THEN rules, in which each rule defines a small piece of the knowledge. Considering the legal domain, a parallel can be established as legislations and other legal concepts are built on the concept of rule.

In that sense, one can picture the development of rule-based systems that describe rules of specific legal fields that can then be used to determine which rules apply in a given case. Consequently, it is possible to implement rule-based systems that model specific norms in order to determine the values of the BATNA and the WATNA.

Let us take, as an example, the Portuguese labor law domain, as depicted in Decree of Law (DL) 7/2009 (Portuguese laws), considering a scenario in which a worker wants to end the labor contract claiming that the employer did not pay the last three salaries. According to Article 394th, nr. 2 a), the lack of regular payment of the salary constitutes a just cause for a worker to end the contract. Moreover, Article 394th, nr. 1 when there is a just cause, the worker can immediately end the labor contract. The first question is thus to determine the existence or not of the lack of payment, and thus, of a just cause for ending the contract. Assuming that this has been proved, let us try to determine the best and worst scenarios, from the point of view of the worker. The most important norms are found in Article 396th, numbers 1, 3 and 4. Number 1 states that, if Article 394th is true (there is just cause for ending contract), the worker is entitled to 15 to 45 days of salary plus indemnity for each year of contract. It also states that this value varies according to the degree of wrongfulness of the employer and that the total indemnity paid to the worker should not be inferior to three salaries plus indemnity. However, number 3 states that the indemnity paid can be higher whenever the worker suffered property damage or other damage, of higher value. Finally, number 4 states that, in the cases of a temporary employment contract, the value of the indemnity cannot be smaller than the value of the salaries that would be received until the end of the contract. We can thus formalize the computation of the BATNA and WATNA in the form of IF-THEN rules.

A simplification of the rules that allow the computation of the BATNA and WATNA values according to the Portuguese labour law. This example code considers only the case in which a worker ends the contract with a just cause. M_SALARY denotes the monthly salary; D_SALARY denotes the daily salary; M_REMAINING denotes the months remaining until the end of the temporary contract; +VARIABLE denotes an unknown value, higher than VARIABLE.

```
Def_Rule 396

if RULE_394 then
  WATNA := 3 * (M_SALARY + SENIORITY)
  if TEMPORARY_CONTRACT then
    if WATNA < M_REMAINING *(M_SALARY + SENIORITY) then
      WATNA := M_REMAINING *(M_SALARY + SENIORITY)
  if WATNA < 15 * (D_SALARY + SENIORITY) then
    WATNA := 15 * (D_SALARY + SENIORITY)
  BATNA := 45 * (D_SALARY + SENIORITY)
  if BATNA < DAMAGE then
    BATNA := +DAMAGE
```

There are some interesting advantages in following a rule-based approach. The main is that this is a relatively simple way of implementing legal norms, being also easy to maintain and update. Having defined the values of the BATNA and the WATNA, it is immediately possible to compute the range of the ZOPA: it is given by the distance between these two values. More challenging is the determination of the possible outcomes and its corresponding likeliness.

In this case a purely rule-based approach would not be appropriate as it is necessary to evaluate a group of cases and categorize them according to their likeliness. Moreover, a mechanism to select cases according to its similarity is also obligatory. Therefore, following a case-based approach would be an appropriate choice [18].

In order to determine the possible cases, their likeliness and the MLATNA, UMCourt relies on the previously mentioned CBR algorithm. However, as our focus is merely on compiling information for informing parties rather than suggesting an outcome for the dispute, only the first stage of the algorithm is relevant [6]. In this first stage, the Retrieve one, the algorithm selects a group of cases according to its relevance for the current case, which is given in terms of the similarity. There are several techniques for retrieving cases. Unlike database searches that target a specific value in a record, retrieval of cases from the case base must be equipped with heuristics that perform partial matches, since in general there is no existing case that exactly matches the new case [19]. Moreover, we are not searching for an exact match but for a group of similar cases.

To do it, a hybrid approach is being used that combines a template algorithm with a nearest neighbor one [6]. The template retrieval narrows the search space so that the nearest neighbor algorithm performs quicker. The application of a template retrieval algorithm is possible as it is possible to know a priori which cases have the possibility of being similar and which ones do not (e.g. cases that address different norms cannot be similar). In that sense, template retrieval works much like SQL queries: a set of cases, that match a pre-selection rule, is retrieved from the database. These rules can be changed dynamically by the system whenever the results of the pre-selection don't match the system parameters. For example: one of the pre-selection rules indicates that cases should be selected if they address the same norm. However, the system can consider the norm at several levels: Article, Item or Number. The standard rule is to look at the Article. However, if too many cases are retrieved, the system is able to dynamically change this specific rule and retry the query with a more specific one.

In the next step, the nearest neighbor algorithm must only be applied to the set of pre-selected cases instead of applying it to all the cases in the case memory, a task that could be very time consuming as the nearest neighbor algorithm has linear complexity (equation 1).

$$\frac{\sum_{i=1}^{n} W_i * fsim_i(Arg_i^N, Arg_i^R)}{\sum_{i=1}^{n} W_i} \qquad (1)$$

In this algorithm, the weights are, at this moment, determined by a law expert, based on the importance that, according to his experience, each of the components of the similarity measure has. However, it is our objective that, in the future, the system changes these values dynamically, looking at past iterations, in an attempt to select the most appropriate weights for each case.

The output of this phase of the algorithm is a list of similar cases, ordered according to their value of similarity with the current case. We only need to add the main assumption of CBR: if a case is similar to another one, then its conclusion is also expected to be similar. Based on this, we can look at the solution of the retrieved case with the higher degree of similarity and assume that it is the most likely solution to the current case. Likewise, we can look at the region in which the similarity of the cases is higher and assume that the probable solution lies in that region, i.e., the MLATNA. Following the same approach, for each case retrieved a value of similarity is also provided to the user, which will indicate its likeliness to occur.

At this point, the worker has all the main information that he/she may need in order to make rational decisions throughout the dispute resolution process. It is possible to use a visual tool to represent all this information in a single and intuitive graphic that the party can consult. In Figure 2 a graphic for a fictitious case is presented. Each dot represents a case, with an associated value of similarity and an utility value, which represents the indemnity that the worker will receive. The case with the highest value of similarity is the MLATNA and tells the worker the most likely outcome if he decides to go into a court. The positions of the BATNA and WATNA are also represented, depicting the best and worst possible scenarios. The dashed line is given by a 3rd degree polynomial function and represents an overall view of similarity versus utility. Looking at this line, the worker can conclude that, if he goes into a litigation process, according to the known cases, the indemnity will most likely be between a value of 500 and 650.

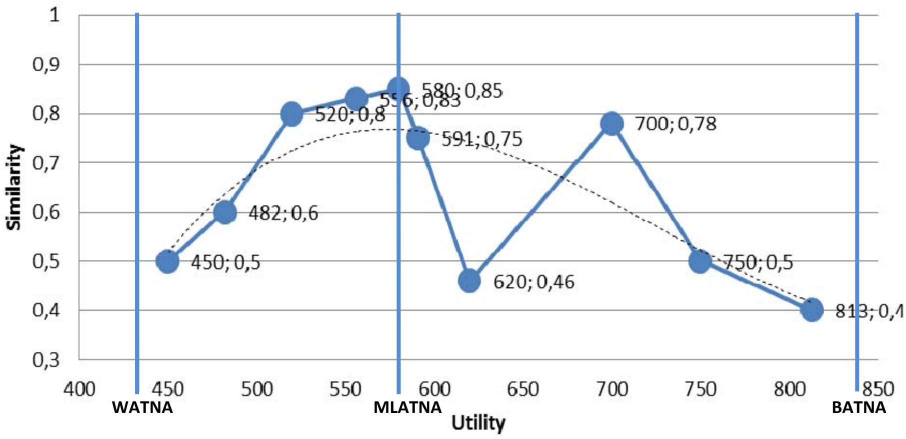

Fig. 2. Graphical representation of all the information compiled by UMCourt.

Combining cases with rules, it was possible to develop a methodology that compiles all the important information at the outset of the dispute resolution process. This will, in a first moment, help the disputant party decide if he/she should advance into litigation or if it would be better to continue with this process. At this point, the disputant party is able to weight the consequences of his possible decisions, assess its chances of success, determine its consequences and thus achieve more satisfactory outcomes.

4 Three Case Studies

Let us now depict three case studies in which the techniques depicted in this paper are being applied, all focusing on the Portuguese legal domain.

4.1 UMCourt Divorce and Heritage's Share

UMCourt Divorce and Heritage's Share [20] addresses property division in two scenarios: divorce and heritage's share. It is based on the Adjusted Winner algorithm (AW) [21] in order to define the share. This algorithm allows the division of *n* items between two parties in conflict. AW uses techniques from the game theory field and deals with the Nash equilibrium concept. It also makes use of a blind attribution of points to the items that are being divided by the parties. Each party must distribute a total of 100 points by the items, designating how much they want it. The points allocated are then submitted to a mathematical manipulation that determines the division of the properties by the parties. This process, as it is, may be considered envy-free because each party receives the items according to the preferences that were assigned, i.e., they receive the items or the most valuable half (according to the subjective evaluation), which makes each party feel satisfied and without greed about what the other party received. Division seems fair because each party receives at least 50% of the intended items. Moreover, each party believes his half of the property is more valuable than the other half (and subjectively it is in fact).

However, the parties may not be completely honest when assigning the points. As an example, if one of the parties makes the allocation of points considering the monetary value of the items, and the other party doesn't have a notion about the prices or simply assigns the points according to the preferences (unaware of the bad intention of the other party), the second party is in disadvantage. In order to address this fair division problem, considering the monetary perspective, this work introduces some changes in the AW algorithm: the Adjusted Winner by Value. Therefore, a component depicting the monetary value of each item, in which the arithmetic manipulation is performed, was added.

Although the division presented by AW by Value may be considered fair according to the preferences and the values of the items, the parties may choose not to accept it as it was proposed. In order to address this issue, a negotiation mechanism that can mediate the process and present other alternatives was developed. This mechanism is supported by the techniques presented in this paper. In that sense, parties can access other similar cases, know their best and worst scenarios, know the most likely scenario and thus cooperate in a more rational and realistic way in the negotiation process.

4.2 UMCourt Commerce

The legal domain of this case study is the Portuguese consumer law. Because this domain is a quite wide one, we restricted it to the problematic of buy and sell of consumer goods and respective warranty contracts. Thus, concrete solutions for the conflicts arising from the supply of defective goods (embodied mobiles or real estate) were modeled. Financial services are also considered, as well as the cases in which there are damages arising out of defective products. Regarding the legal boundaries established, solutions for conflicts are being modeled as they are depicted in Decree of Law (DL) 67/2003, as published by DL 84/2008 (Portuguese laws).

Using this system, an unsatisfied buyer can use a web site[1] or a mobile application (Figure 3) in two different ways: for simulating a defect of a product that he intends to buy of for asking for a solution for a dispute arising out of a defective product already bought [22]. For deciding on an outcome, the system relies on rules that model the necessary legal norms. Thus, in this case, given the clear and relatively simple nature of these norms, the MLATNA is given by the rules, as well as the BATNA and the WATNA. However, the buyer has also access to similar cases, provided as mentioned above, concerning disputes with similar characteristics.

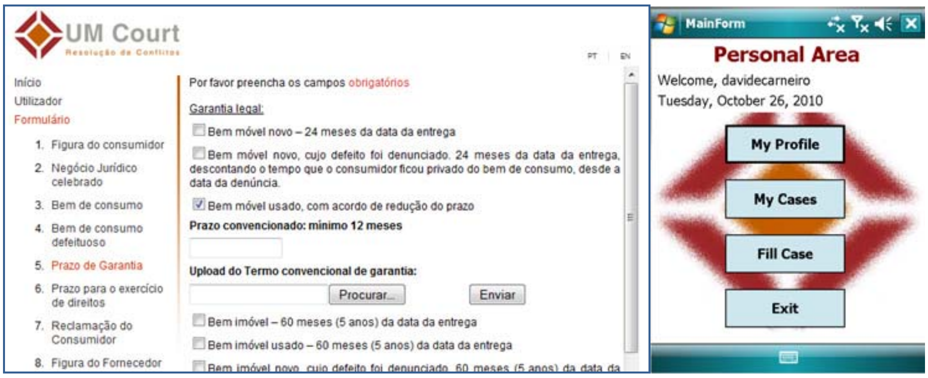

Fig. 3. An excerpt of the web interface (left, in Portuguese) and a screen from the mobile application (right).

4.3 UMCourt Labor

Given the current global crisis, labor disputes are more and more frequent. This case study deals with the issue of an employer being dismissed or wanting to end a work contract. Under legal systems such as that of Portugal, a huge deal of legal parameters need to be considered: (1) the antiquity of the worker in the company, (2) supplementary work, (3) night work, (4) justified or unjustified absence to work, (5) the possibility of a “just cause for dismissal” being declared by Court, (6) the existence (or not) of a valid and legal procedure of dismissal, (7) the possibility of dismissal being accepted without indemnities or (8) of it being accepted but accompanied by indemnities that could range from a very low to a very high amount of money. This, together with a relatively complex legislation on this subject, may make it difficult for workers to take rational decisions.

[1] The UMCourt Commerce site is available at http://tiaracserver.di.uminho.pt/odr

In that sense, the information mentioned above can help workers throughout such processes. Specifically, a worker can consult past similar cases and know the likeliness of their solutions in his dispute, know his BATNA and WATNA and know his MLATNA. Moreover, a multi-party negotiation tool supported by the CBR mechanism is also available (Figures 4 and 5). The main purpose of this tool is to support effective negotiation between two or more parties involved in a labor law dispute. The tool starts by proposing the solution of the MLATNA and the parties engage in a sequence of turns in which, in each turn, all parties can accept, change, ignore or refuse the current proposal for solution. In each round, if there is no consensus, the system will build a solution from the suggestions of the parties (if possible) or will suggest a solution from a similar case. The process goes on until a consensus is achieved or the system runs out of suggestions for solutions. In this case study, not only the party has access to the information described above but can also use a negotiation tool that will improve the efficacy of the alternative dispute resolution process.

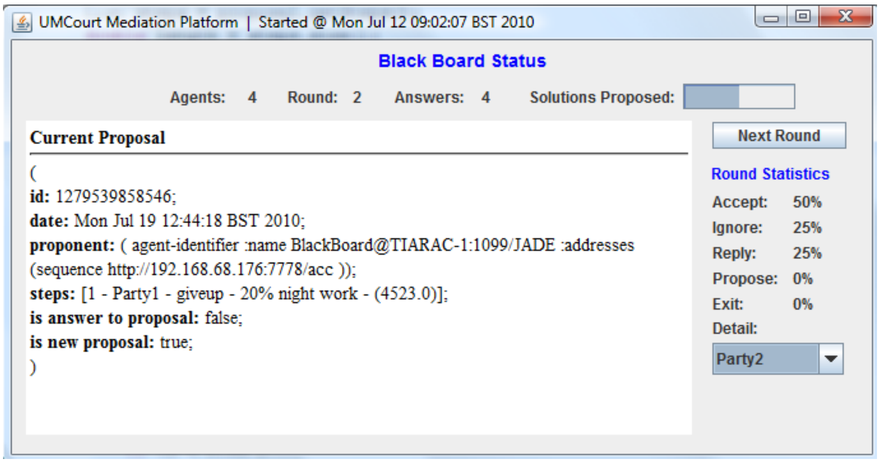

Fig. 4. The administrator interface which shows the state of the negotiation platform, including statistics for the current round and information about the current proposal on the table.

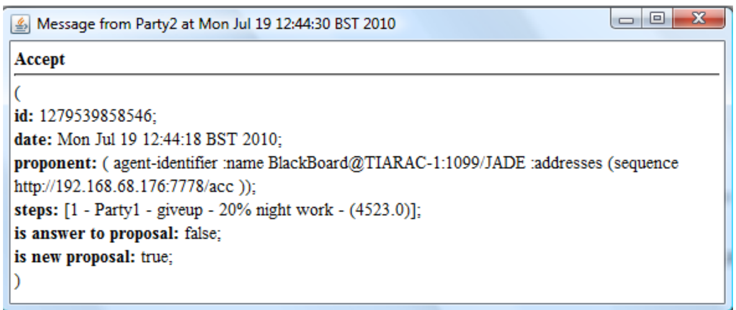

Fig. 5. The interface that depicts a message from a participant of the negotiation.

4 Conclusion

The main idea that is present throughout this paper is that informed parties are in a better position to take weighted and rational decisions. Concretely, a party should consider, in a first moment, the usefulness of entering into an alternative dispute resolution process. In order to take this decision, concepts like the BATNA, WATNA and MLATNA are of utter importance. Moreover, the party should also be aware of all the possible outcomes and their likeliness, in order to have a clear picture of all the possibilities. In this paper we identified the relevant knowledge for taking rational decisions and pointed out to domain-dependent methods for creating that knowledge. In a parallel work, we are developing an abstract architecture that implements these concepts in several legal fields by using ontologies. In this approach, agents are abstract entities that provide services useful for all the domains addressed. However, a service is implemented differently in each domain, according to the specificities of the legislation. In order to implement the services for each specific domain the agents use domain ontologies, which define how each action should be implemented. This approach results in simpler architectures in which functionality reuse is maximized.

Acknowledgments. The work described in this paper is included in TIARAC - Telematics and Artificial Intelligence in Alternative Conflict Resolution Project (PTDC/JUR/71354/2006), which is a research project supported by FCT (Science & Technology Foundation), Portugal.

References

1. Katsch E., Rifkin J.: Online dispute resolution – resolving conflicts in cyberspace. Jossey-Bass Wiley Company, San Francisco. (2001)
2. Raiffa, H.: The Art and Science of Negotiation. Harvard University Press. (2002)
3. Brown, H., Marriott, A.: ADR Principles and Practice. Sweet and Maxwell. (1999)
4. Bennett, S. C. Arbitration: essential concepts. ALM Publishing. (2002)
5. Bellucci, E., Lodder, A. R., Zeleznikow, J.: Integrating Artificial Intelligence, Argumentation and Game Theory to Develop an Online Dispute Resolution Environment. ICTAI 2004: 749-754
6. Carneiro D., Novais P., Andrade F., Zeleznikow J., Neves J.: The Legal Precedent in Online Dispute Resolution. In Legal Knowledge and Information Systems, ed. Guido Governatori. Proceedings of the Jurix 2009 - the 22nd International Conference on Legal Knowledge and Information Systems, Rotterdam, The Netherlands, IOS press, pp 47--52. (2009)
7. Peruginelli, G., Chiti, G.: Artificial Intelligence in alternative dispute resolution. Proceedings of the Workshop on the law of electronic agents – LEA. (2002)
8. Wooldrige, M.: An Introduction to Multiagent Systems, John Wiley & Sons. (2002)
9. Bellifemine, F. L., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE. Wiley. (2007)
10. Notini, J.: Effective Alternatives Analysis In Mediation: "BATNA/WATNA" Analysis Demystified, (http://www.mediate.com/articles/notini1.cfm), 2005. Last accessed October, 2010.
11. Klaming, L., Van Veenen, J., Leenes, R.: I want the opposite of what you want: summary of a study on the reduction of fixed-pie perceptions in online negotiations. "Expanding the

horizons of ODR", Proceedings of the 5th International Workshop on Online Dispute Resolution (ODR Workshop'08), Firenze, Italy pp. 84--94. (2004)
12. Goldberg, S.B., Sander, F.E., Rogers, N., Cole, S.R.: Dispute Resolution: negotiation, mediation and other processes, Aspen Publishers, New York. (2003)
13. De Vries BR., Leenes, R., Zeleznikow, J.: Fundamentals of providing negotiation support online: the need for developing BATNAs. Proceedings of the Second International ODR Workshop, Tilburg, Wolf Legal Publishers, pp. 59--67. (2005)
14. Fisher, R., Ury, W.: Getting To Yes: Negotiating Agreement Without Giving In. Boston: Houghton Mifflin. (1981)
15. Steenbergen, W.: Rationalizing Dispute Resolution: From best alternative to the most likely one, in Proceedings 3rd ODR workshop, Brussels. (2005)
16. Raiffa, H.: The art and science of negotiation: how to resolve conflicts and get the best out of bargaining, Cambridge, The Belknap Press of Harvard University Press. (1982)
17. Hayes-Roth, F.: Rule-based systems. Communications of the ACM archive, Vol. 28, Issue 9, pp. 921 - 932. (1985)
18. Aamodt A., Plaza E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. In: AI Communications, 7(1):39-59. IOS Press. (1994)
19. Watson, I., Marir, F.: Case-based reasoning: A Review. In: Knowledge Engineering Review, vol. 9, pp. 327–354. (1994)
20. Café, A., Carneiro, D., Novais, P., Andrade, F.: Sistema de Resolução Online de Conflitos para Partilhas de bens – Divórcios e Heranças. INFORUM 2010. (2010) (*in portuguese*)
21. Brams, S.J., Taylor, A.D.: Fair Division: From cake cutting to dispute resolution. Cambridge University Press. (1996)
22. Costa, N., Carneiro, D., Novais, P., Andrade, F.: An Agent-based Approach to Consumer´s Law Dispute Resolution. Proceedings of the 12th International Conference on Enterprise Information Systems - ICEIS 2010. (2010)
23. Andrade, F., Novais, P., Carneiro, D., Zeleznikow, J., Neves, J.: Using BATNAs and WATNAs in Online Dispute Resolution. In JURISIN 2009 - Third International Workshop on Juris-informatics, Tokyo, Japan, pp. 15-26. (2009)