Agent-Based Negotiation Between Partners in Loose Inter-Organizational Workflow

Eric Andonoff, Lotfi Bouzguenda
Laboratoire IRIT / Université Toulouse 1
1 Place Anatole France, 31042 Toulouse Cèdex, France
{andonoff, lotfi.bouzguenda}@univ-tlse1.fr

Abstract

This paper deals with negotiation between partners (organizations) in the context of loose Inter-Organizational Workflow (IOW). By negotiation between partners, we mean all the conversation acts made between a requester partner looking for a workflow service and one or several selected provider partner(s) able to provide the requested service.

The agent technology is at the basis of our proposition since (i) it provides natural abstractions to deal with distribution, heterogeneity and autonomy which are inherent to loose IOW, (ii) it introduces powerful concepts such as groups and roles useful to describe in details the coordination of the different partners involved in the negotiation, and (iii) it has investigated the problem of negotiation between agents.

In this paper, we first answer the following question “What kind of negotiation is suitable in loose IOW?” identifying what kind of agent behavior (competitive or cooperative) and what kind of negotiation protocol (auction, heuristic, argumentation or Contract-net) better suit loose IOW. We also propose an agent-based architecture, compliant with the Workflow Management Coalition reference architecture, to support negotiation between partners, and an organizational model based on the AGR (Agent-Group-Role) meta model, which structures the negotiation in terms of Agents, Groups and Roles.

1. Introduction

The aim of Inter-Organizational Workflow (IOW) is to support the cooperation between distributed and heterogeneous business processes running in different autonomous organizations. The different organizations involved in an IOW need to put resources and skills in common, and coordinate their respective business processes in order to reach a common goal, corresponding to a value-added service. Thus, IOW is a key technology for helping participating organizations to face the emergence of the open and dynamic worldwide economy [1].

A fundamental problem for IOW is the coordination of the different processes. By coordination, we mean all the work needed to put all these processes together in order to provide the global common goal in an efficient manner.

This coordination problem can be investigated in the context of two different scenarios: loose IOW and tight IOW [2]. In this work, we have focused on loose IOW which refers to occasional cooperation between organizations, free of structural constraints, where the organizations (partners) involved and their number are not pre-defined but should be selected at run time in an opportunistic way. Coordination in loose IOW raises several specific sub-problems which are: (i) the finding of partners able to realize a workflow service (i.e. a service implementing a business process), (ii) the negotiation of workflow services between partners, (iii) the contracts specification between partners and (iv) the execution of a workflow service.

Loose IOW coordination problem is insufficiently addressed in the literature [3]. Existing propositions do not provide a solution to all the different coordination sub-problems listed above, are not situated in an engineering perspective that aims at designing, specifying and implementing IOW, and moreover, do not discuss the compliance of their solutions with the Workflow Management Coalition (WfMC) reference architecture [4].

We already have addressed in [5] the “finding of partners” coordination sub-problem. In this paper we have provided a solution to deal with the “negotiation between partners” coordination sub-problem. The negotiation between partners follows up the finding of partners. The finding of partners consists in selecting
one or several provider organizations able to execute a workflow service which is needed for a requester organization. Thus, after the finding of partners, a workflow service requester is connected to different workflow service providers. But connecting is not enough to definitively choose the partner that is going to execute the workflow service: a negotiation step, in terms of due time, price, visibility of the service evolution and way of executing the service (process), is necessary to evaluate and select the best workflow service provider.

The agent technology is at the basis of our proposition for several reasons. First, it provides natural abstractions to deal with distribution, heterogeneity and autonomy which are inherent to loose IOW. Second, it introduces powerful concepts, such as groups and roles [6], useful to describe in details (at a macro level) the coordination of the different partners involved in the negotiation. Third, using this technology, we benefit from all the work around negotiation in multi-agent systems [7].

The contributions of this paper are:
- An answer to the question “What kind of negotiation is suitable in loose IOW?” investigating what kind of agent behavior and what kind of negotiation protocol better suit loose IOW.
- An agent-oriented architecture for negotiation between partners in loose IOW. This architecture models the different processes of the IOW as agents and introduces specific mediators to coordinate them (the processes) and make them cooperate during the negotiation between partners. This architecture is compliant with the WfMC reference architecture.
- An agent-oriented organizational model which highlights, during the negotiation between partners, the coordination of loose IOW processes through the notions of Agent, Group and Role as suggested in [6].

This paper is organized as follows. Section 2 deals with the kind of negotiation in loose IOW. Section 3 presents the agent-oriented architecture. Section 4 describes its corresponding agent-oriented organizational model. Section 5 compares our proposition with related works and concludes the paper.

2. What Kind of Negotiation is suitable in Loose IOW?

By negotiation between partners, we mean all the conversation acts made between a workflow service requester and a set of selected workflow service providers according to criteria such as due time, price, visibility of the service evolution and way of executing the service.

To answer the question “What kind of negotiation is suitable in loose IOW”, we need to know what kind of agent behavior (competitive or cooperative) and what kind of negotiation protocol (auction, heuristic, argumentation or Contract-net) better suit loose IOW.

2.1 What Kind of Agent Behavior is suitable in Loose IOW?

Four criteria are used to define two kinds of agent behaviors:
- The goal of the negotiation which can be individual or common. In a competitive context, each partner wants to maximize its own interest which is not the case in the cooperative context.
- The ending of the negotiation which can be certain or possible. In the second case, the ending of a negotiation depends on the negotiation protocol used. For instance, the gain is anticipated for the auction protocol, while it is not guaranteed for the other negotiation protocols such as the heuristic protocol or the argumentation protocol.
- The universe of the negotiation which can be closed or open. For instance, when the universe is open, a partner which has not enough interest in the negotiation can suddenly refuse to continue with the negotiation. Consequently, the number of partners involved in an open negotiation remains unknown.
- The strategy of the negotiation which can be ambiguous or obvious. Naturally in the competitive context, each partner has its own strategy, which is kept private. For instance, when a proposal provided by X is rejected, Y can provide a counter proposal making some concessions from the last proposal.

<table>
<thead>
<tr>
<th>Goal of the negotiation</th>
<th>Ending of the negotiation</th>
<th>Universe of the negotiation</th>
<th>Strategy of the negotiation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Competitive Behavior</td>
<td>individual</td>
<td>possible</td>
<td>opened</td>
</tr>
<tr>
<td>Cooperative Behavior</td>
<td>common</td>
<td>sure</td>
<td>closed</td>
</tr>
</tbody>
</table>

As shown in table 1 above, the competitive behavior is a behavior where the goal of the negotiation is individual, its ending is possible but not certain, its universe is open and its strategy is ambiguous. This
competitive behavior matches the requirements of loose IOW better than the cooperative one.

2.2 What Kind of Negotiation Protocol is suitable in Loose IOW?

The criteria used to determine the negotiation protocol that better suits loose IOW are (i) the negotiation behavior (competitive or cooperative), the number of partners (one or several) and the fact whether a multi-attribute negotiation is possible or not. The main negotiation protocols proposed in the literature [7] are considered.

Table 2. Negotiation Protocols and Selection Criteria

<table>
<thead>
<tr>
<th>Negotiation behavior</th>
<th>Number of partners</th>
<th>Multi-attribute negotiation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auction protocol</td>
<td>competitive</td>
<td>2 or several</td>
</tr>
<tr>
<td>Heuristic protocol</td>
<td>competitive</td>
<td>1 or several</td>
</tr>
<tr>
<td>Argumentation protocol</td>
<td>competitive</td>
<td>1 or several</td>
</tr>
<tr>
<td>Contract Net protocol</td>
<td>cooperative</td>
<td>1 or several</td>
</tr>
</tbody>
</table>

According to table 2, the heuristic and argumentation protocols fit the loose IOW requirements better since (i) the partners involved in a loose IOW are competitive, (ii) the negotiation concerns several attributes (due time, price, visibility of the service evolution, way of doing it), and (iii) the number of partners is one or several.

2.3 Example of Negotiation between Partners using the Heuristic Protocol

This section gives a brief example of negotiation between partners using the heuristic protocol. The chosen example considers a review process based on volunteer reviewers, as it is the case for the ACM Symposium on Applied Computing, Special Track on Coordination for instance [8]. In this case, the review process is deployed in the context of a loose IOW since it involves actors (PC Chair, reviewers,...) from different organizations (Universities, laboratories, industries), without special agreements between them, and the reviewers involved and their number are not pre-defined but dynamically recruited. Once, a reviewer is found, the PC Chair negotiates the workflow service (i.e. reviewing papers) that the reviewer provides according to different criteria (due time, price, visibility of the service evolution and way of executing the service) in order to select him or not.

Before giving the negotiation example, we have first explained the principle of the heuristic protocol. This protocol improves the effectiveness of the negotiation by providing reactions, which can take the form of a critique or a counter-proposal [7].
- A critique is a comment on the part of the proposal, which the agent accepts or refuses.
- A counter-proposal is an alternative proposal, generated in response to a proposal.

When applying these notions to the reviewing process example, we obtain these two negotiation examples:
- Example of a critique.
  PC Chair: My proposal is that you make the reviewing papers for me during 15 days and I will pay you 100 units.
  Reviewer: I agree to review papers during 15 days but you must pay me more.
- Example of a counter-proposal.
  PC Chair: My proposal is that you provide me with a detailed reviewing for each paper you examine.
  Reviewer: I accept to provide a detailed reviewing of each paper if you provide me with the review sheet.

In these examples, we used a critique to negotiate general criteria such as due time or price, and we used a counter proposal to negotiate the details of workflow service, i.e. visibility of the service evolution or way of executing the service.

3. An Agent-Based Architecture to Support Loose IOW

This section shows how we revisit the WfMC reference architecture using agents and how we deal with the negotiation between partners.

3.1. Revisiting the Reference Architecture with Agents

The reference architecture defined by the WfMC supports syntactic interoperability between WfMSs and communication between workflows. Thus, it is highly appropriate to be conformed with this reference architecture in the context of IOW.

According to this architecture, a WfMS includes a Workflow Enactment Service (WES) that manages the execution of workflow processes (upon the internal structure of which one imposes no constraints) and supports the following interfaces:
- Interface 1 with Process Definition,
- Interface 2 with Workflow Client Applications,
- Interface 3 with Invoked Applications,
First, the Workflow Enactment Service is completed with a Connection Server whose role is to allow the Workflow Agents (providers or requesters) to communicate with the infrastructure of negotiation through a new interface, interface 6. This interface also supports the connection with the infrastructure of the finding of partners as presented in [8].

Second, an infrastructure of negotiation, independent of the Workflow Enactment Service, handles the negotiation between the different Workflow Agents (one requester Workflow Agent and one or several provider(s) Workflow Agents). This infrastructure includes two specific mediators, named Moderator and Conversation Server. There are as many Moderators as the number of conversations initiated by different requester Workflow Agents. The Conversation Server allows requesters Workflow Agents to choose the protocol underlying to the opened conversations via their Connection Server and using interface 6. The protocols are stored in the Negotiation Protocols Database and are managed by the Conversation Server.

### 3.2. Agents for Negotiation Between Partners

This architecture also deals with a negotiation of workflow services between partners adding a new component to the Workflow Enactment Service and defining an infrastructure of negotiation.

![Fig.1: An Agent-Based Architecture for the Workflow Enactment Service](image1)

**Figure 1** explains how we have rethought the WES using agents. The idea is to implement each workflow process instance (stored in the Workflow Processes Database) as a software process, and to encapsulate this process within an (pro-)active agent. Such a Workflow Agent includes a workflow engine that, as and when the workflow process instance progresses, reads the workflow definition and triggers the action(s) to be done according to its current state. This Workflow Agent supports interface 3 with the applications that are to be used to perform pieces of work associated to process’ tasks. The Agent Manager controls and monitors the running of Workflow Agents:

- Upon a request for a new instance of a workflow process, the Agent Manager creates a new instance of the corresponding Workflow Agent type, initializes its parameters according to the context, and launches the running of its workflow engine.
- It ensures the persistency of Workflow Agents that execute long-term business processes extending to a long time in which task performances are interleaved with periods of inactivity.
- It coordinates Workflow Agents in their use of the local shared resources.
- It assumes interfaces 1, 2 and 5 of the WiMS with the environment.

![Fig.2: Overview of the Negotiation between Partners using a Conversation Server and Moderators](image2)

More precisely, as illustrated in Figure 2, the requester Workflow Agent, which is the negotiation initiator, (i) submits its request to its Connection Server for the creation of a moderator implementing the desired negotiation protocol, (ii) the Connection Server forwards the request to the Conversation Server, and this later creates the corresponding moderator (e.g. Mod1). The Conversation Server also stores all the information about each conversation (such as the used protocol, the identity of the Moderator supervising the conversation, the date of its creation and deletion, the requester Connection Server initiating the conversation) in its Conversations Database.

Moreover, the purpose of the Conversation Server is to make the interaction space explicit and public, by
allowing Workflow Agents, via their Connection Server, to get information about the current (and past) conversations and be notified of new conversations [9]. The Conversation Server only publishes, in this interaction space, the information authorized by the providers Workflow Agents. So, the autonomy of each partner, which is an important constraint in the context of loose IOW, is preserved.

The Moderator is an agent which implements the negotiation protocol: it ensures that each conversation act of a conversation is consistent with the corresponding negotiation protocol. To take part in a conversation, the provider Workflow Agents need to be aware of their corresponding Moderator. This information can be obtained by querying the Conversation Server.

The following figure illustrates the negotiation process inside the conversation space through the AUML Sequence Diagram.

This section describes, using an organizational model, how we have structured the negotiation between partners involved in loose IOW through the different agents belonging to the previous architecture.

4.1. An Organizational Perspective

We have adopted an organizational view that provides appropriate notions to describe the macro-level of the coordination among partners during the negotiation stage in terms of externally observable behavior, independent of the internal features of each participating component. These notions (such as roles, groups, teams, interactions or commitment) are conceptual tools that ease the modeling of the structure and behavior of the systems under consideration.

The Agent, Group and Role (AGR) meta-model [6] is one of the frameworks proposed to define the organizational dimension of a multi-agent system, and it is well appropriate to the loose IOW context [5]. According to this meta-model, the organization of a system is defined as a set of related groups, agents and roles.

A group is a set of agents that also determines an interaction space: an agent may communicate with another agent only if they belong to the same group. The cohesion of the whole system is maintained by the fact that agents may belong to any number of groups, so that the communication between two groups may be done by agents that belong to both. Each group also defines a set of roles, and the group manager is the specific role fulfilled by the agent that initiates the group. The membership of an agent in a group requires that the group manager authorizes this agent to play a role, and each role determines how the agents playing it may interact with other agents. So the whole behavior of the system is framed by the structure of the groups that may be created and by the behaviors given to the agents by the roles. On the other hand, the AGR meta-model agrees with any kind of agents since it imposes no constraint on their structure.

4.2. An AGR-Based Organizational Model

The organizational model we propose to describe the coordination between partners is presented in figure 4. It is organized around the following components:

- Three types of groups represented by ellipses. These groups are: RequestNegotiation, Moderators and

4. An Agent-Based Organizational Model to Support Negotiation Between Partners

Fig.3: AUML Sequence Diagram showing Message Exchange during Negotiation in the Conversation Space
Conversations. In this figure, we have two RequestNegotiation Groups.
- Four types of agents represented by candles which are: Connection Server, Workflow Agent, Moderator and Conversation Server. In this figure, we have two Workflow Agents: a requester Workflow Agent and a provider Workflow Agent.
- Eleven roles since each agent plays a specific role within each group.

![Fig. 4: An Agent-Group-Role based Organizational Model](image)

Let us detail now how each group operates. The RequestNegotiation group enables one or several (requester or provider) Workflow Agents to enter in connection with their Connection Server for the negotiation with partners.

The Moderators group enables a Connection Server to submit the creation of a moderator made by its requester Workflow Agents to the Conversation Server. The role of the Conversation Server in this group is to create the requested moderator by instantiating a negotiation protocol and register the general information about the corresponding opened conversation.

Finally, the Conversations group enables one or several provider Connection Servers or one requester Connection Server, to enter in connection with their Moderator to participate in the conversation. The role of the Moderator in this group is to manage and check the conversation acts according to the chosen protocol. The role of the Conversation Server in this group is to make the conversation public, by allowing the different partners, to only get authorized information about the current conversation.

According to the conversation protocol (heuristic or argumentation), the moderator creates one or several Conversations groups. If we consider the argumentation protocol, the Moderator only creates one group gathering all the partners of the conversation. While, in the case of the heuristic protocol, there are as many groups as the number of providers. The Conversations group shown in Figure 4 uses the argumentation protocol.

5. Discussion and Conclusion

Regarding the design and development of workflow systems, the use of agent technology is not new, and several works have been proposed in the literature [10, 11,12]. More precisely, [10] defends the interest of the agent approach in workflow systems, [11] revisits the Workflow Enactment Service with agents and [12] proposes to coordinate Web Services implementing Workflow using agents.

Our work differs from the previous works on three points. First, it addresses the negotiation between partners coordination sub-problem in the context of loose IOW, aspect which has not been addressed in the previous works. Second, the proposed architecture is compliant with the WfMC reference architecture. Third, this architecture fully exploits the organizational dimension of agent technology to abstract and structure the interaction space.

The problem of negotiation between agents has, of course, been addressed in Multi-Agents Systems [7,13] but also in the context of Electronic Business [14] and Information Systems [15].


But, to the best of our knowledge, the problem of negotiation between agents has not been addressed in the context of loose IOW. Thus, we believe that our solution is currently unique in trying to take into account the agent technology to deal with negotiation between partners in loose IOW. The main contributions of our solution are the following:

- An answer to the question “What kind of negotiation is suitable in loose IOW?” investigating what kind of agent behavior and what kind of negotiation protocol better suit with loose IOW.
- An agent-based architecture to support the negotiation between partners by proposing an independent infrastructure of negotiation composed of two original mediators, that is the Moderator and the Conversation Server. The main advantages of this
architecture are: (i) it is compliant with the Workflow Management Coalition reference architecture, (ii) the infrastructure of negotiation is independent of the Workflow Enactment Service (WES) and is dynamically plugged to the WES only when it is necessary, and finally (iii) our architecture imposes only the development of the Connection Server to support interface 6 but does not impose any constraint on the internal architecture of the WES (which can be agent-oriented, as we have defended in the paper, or not) in order to take into account legacy WES.

- An organizational model based on the AGR meta model to structure the negotiation between partners in terms of Agents, Groups and Roles. This organizational perspective makes possible the modeling of the coordination of business process belonging to the different workflow systems under consideration by clearly separating the macro-level (coordination) from the micro-level (agent).

We have implemented a simulator, called MatchFlow [5], whose objective is to make partners participating in an IOW cooperate. To do this implementation, we have used the MadKit platform [6], which fully integrates the AGR meta-model. Currently, MatchFlow supports the finding of partners while the negotiation between partners is in progress. As future work, we plan to continue this implementation and to evaluate our simulator in the ReviewingPapersProcess case study briefly described in section 2.3.

6. References


