Emergent Process Interoperability within Virtual Organisations

Martin Carpenter, Nikolay Mehandjiev, and Iain Duncan Stalker

School of Informatics, University of Manchester, PO Box 88, Manchester, M60 1QD, UK
m.carpenter@manchester.ac.uk

Abstract. The current speed of market change means that business opportunities today are increasingly short-lived. To successfully pursue these opportunities, enterprises increasingly establish virtual organisations, drawing upon established networks of partners having complementary skills and expertise. The formation of a virtual organisation traditionally derives from a top-down decomposition of a business goal into a set of activities, followed by a recruitment of members to implement these activities. As this essentially presupposes a closed-world context, it does not foster innovative solutions and will fail if a decomposition cannot be found or the recruited members cannot work with each other. The approach proposed here aims to address these challenges through innovative use of agent technology, allowing process interoperability to emerge as a result of shared interests and complementary expertise of individual agents. Members of the virtual organisation are drawn from a comparatively stable yet open business ecosystem or virtual breeding environment [1] in response to a business opportunity which is "pinned" to a notice board as soon as it appears. We show how this approach can complement top-down decomposition, using a simple case study and a prototype. The prototype is implemented in JADE.

1 Introduction

The dynamic and opportunistic nature of many contemporary markets, together with current trends towards shrinking lead times and volatile demand, mean that business opportunities are increasingly short-lived. To flourish in such an environment and make the most of business opportunities, enterprises face a growing need to share information and to collaborate with others throughout the value chain. This fosters focus on core competencies within a network of complementary expertise and resources. Such collaborations are often result in a *Virtual Organisation (VO)* [2]. Our working definition of a VO is "an aggregation of autonomous and independent organisations connected through a network and brought together in response to a customer need" [3]. This underlines its transient and goal-centred nature.

The speed with which a VO and its business process can be configured is often of a crucial importance in capitalising on a particular business opportunity. Thus, automated software support for VO formation would be a significant advance for business interoperability and process management.

A VO requires recruitment of appropriate members and the creation of a global cross-organisational business process to coordinate the complementary skills of mem-

bers. Conventional approaches to VO formation (cf. Section 2) typically favour a top-down decomposition of the business goal into process activities, followed by a stage of matching organisations to these. Agent technology is often utilised in both stages providing: deliberative goal-driven reasoning over domain knowledge to automate goal decomposition and process refinement; and interaction protocols to structure and manage recruitment.

In this paper we focus on contexts where VOs form within a network of potential partner companies having recurrent dealings with each other, as exemplified by the Upper Austrian Cluster of Automotive Excellence [4]. We appeal to the concept of a *virtual breeding environment*, henceforth VBE, - [1] as a suitable theoretical model for such stable yet open business ecosystems. We examine conventional approaches to virtual organisation formation within a VBE and note several difficulties, most notably an (over-) dependence on a centralised top-down decomposition, see Section 2. The results of this examination motivate our proposing a novel approach, in which members of a congregation form a VO where the global cross-organisational business process *emerges* as a natural corollary (see Section 3); a top-level goal decomposition arises as an adjunct to the natural interactions of the participants.

The aim of the current paper is to expound the underlying principles of the approach. As such, the technical details are kept to a minimum and the examples made deliberately simple. Throughout the paper we indicate potential problems we observe in existing approaches and use these to motivate our proposal. Nevertheless, we would emphasise that we see our approach as complementary to existing approaches and conclude that in many cases, the ideal would be a hybrid, as we discuss in Section 5. The proposed ideas are illustrated using a simple case study (see Section 4): implemented as a multi-agent system built using JADE (jade.tilab.com); ontologies are used to (formally) describe the problem, constraints and dependencies.

2 Preliminaries

In this section we introduce the notion of a virtual breeding environment and examine the constraints on virtual organisation formation within the specific classes of VBE addressed by the work presented. We use these to prepare the ground for our approach (see Section 3).

2.1 Application Domain

Within the current paper we are interested in contexts where VOs form within a network of potential partner companies having recurrent dealings with each other, as exemplified by the Upper Austrian Cluster of Automotive Excellence [4]. While such contexts are encountered in VO literature, see for example, [3], they are usually treated as incidental to the process of virtual organisation formation. One exception to this is the idea of a virtual organisation breeding environment. This idea was originally introduced by Hamideh Afsarmanesh and Luis M. Camarinha-Matos in [1] and later developed into the core application context behind the Ecolead European project.

A concise definition of a VBE was given by the same authors in a later paper,[5] as:

> A VBE is an association of organisations and their related supporting institutions, adhering to a base long term cooperation agreement, and adoption of common operating principles and infrastructures, with the main goal of increasing both their chances and preparedness towards collaboration in potential virtual organisations

A fuller development of this idea with numerous real world examples can be found in [6].

Since this idea describes an automotive cluster quite well we have adopted VBEs as the basic theoretical grounding for the work reported within this paper. The concept however has a very general scope and we modify it.

In particular we do not require the member companies to share a base long term cooperation agreement or common operating principles. Instead we rely on the fact that the companies within the VBE joined it in order to constructively participate in VO formation. As argued by Sen within [7] the social control exhibited as a result of repeated interactions within a stable group often suffices to ensure cooperative behaviour without the use of explicit, formal penalties.

In addition we highlight two important features:

1. *The set of processes available for composition is not fixed.* The VBEs studied within this work are open systems with new members joining and existing members leaving during the lifespan. Moreover, new processes can be made available as these are developed by existing members. Perhaps, the most significant reason derives from the self-interested nature of the participants: the set of processes offered by each is typically in proportion to the perceived return on investment (of effort) of a given opportunity. For instance, a large order might motivate a participant to buy in new machinery; a less appealing proposition is likely to be met with disinterest.
2. *The success of a VO depends upon shared interests of its participants.* Certain companies within the VBE might prefer not to work together. This brings to the fore the need to co-optimise the membership of a VO with the global business process under construction.

2.2 VO Formation within VBEs

In this section we consider the opportunities and challenges attaching to the formation of a VO within a VBE. We identify two phases in the establishment of a VO:

1. *Team formation*, where the members of the VO are chosen from potential candidates.
2. *Detailed process configuration*, where those members (selected) determine how to work together.

These phases are potentially distinct, however, as a VO assembles for a specific purpose, it is important to consider the services offered by each team member during team formation. Notwithstanding, seeking to merge the two phases by combining all processes of a potential team, raises two major difficulties: VBE members might be reluctant to reveal processes in detail to a (central) planning mechanism, compromising any guarantee of a workable solution; and the space of possibilities quickly becomes intractable.

An alternative is to treat the first phase as fully separate from the second. This can lead to the formation of a team where the detailed processes (of the members) do not fit together well. This typically requires a revision of the team or the acceptance of less than optimal solution. We believe that a hybrid solution should yield best results: team formation is guided by partial feedback on how well the processes of potential team members combine and integrate.

We briefly examine some conventional approaches for VO formation, and then explore in detail the top-down techniques to service composition using there. In particular, we consider the applicability of these, noting shortcomings and potential problems. We conclude this section with an examination of an alternative approach.

Conventional Approaches to VO Formation VOs are transient structures assembled to achieve a particular goal or fulfil a given function. Automated approaches to the creation of such goal-driven assemblies typically pursue a (centralised) top-down (functional) decomposition, as exemplified by the software support found within the Service-Oriented Architectures and Web Service communities. Here the aim is to create a composite service to respond to a service request [8]. A service request is received by a central agent which decomposes the request into subtasks. Each of these is matched to a service provider advertising services sufficient to meet the needs of the given subtask. The matching of task requirements to service descriptions (*matchmaking*) is analogous to consulting a *Yellow Pages*. An alternative is to issue a *call for proposal(s) (cfp)*, for example using the *Iterated Contract Net Protocol* (see `www.fipa.org`): we refer to this as the *Contract-Net Approach (CNA)*. Again an agent decomposes a given request into subtasks. The agent then decides to out-source a number of these and in each case makes a cfp, describing the nature of the subtask and soliciting proposals to address this. An agent interested in responding to a cfp submits a proposal, i.e. bids, to supply a necessary service. The (initial) agent chooses from these offers (or rejects these and reissues the cfp, perhaps slightly revised). This is often supplemented by the use of *Library Agents* as a mechanism to manage complexity: the initial agent sends a cfp to the Library Agents, each of which issues a cfp to an appropriate subset of the agents which have registered with it.

Top-Down Techniques for Service Composition Both CNA and the Web Service model distinguish two distinct phases: decomposition and matchmaking. The first of these entail decomposing the original request into a set of sub requests, the second locating appropriate providers for them. Such approaches often consider the interactions among the service providers only once these have been (independently) matched to given subtasks. However, if the selected providers have processes which do not integrate, then revision of the choices is required. Similar difficulties arise if one company is reluctant to work with another, perhaps owing a lack of trust or a previous (bad) experience. Furthermore, certain subgroups of companies may have established close relationships, through previous partnerships and thus, have a good coordination of their actions. Decomposing an initial problem into a set of subproblems presupposes a hierarchy of tasks, or other such semantics, to guide this decomposition. Moreover, an prerequisite to success is that there exist good matches of abstract tasks obtained through such a decomposition to concrete processes available. Hierarchies to support such matches

have been developed and studied in the context of (hierarchical) planning [9, 10]. The utility of these is strongly dependent upon the properties of:

– *Ordered Monotonicity* which requires [9] *"For all abstract plans, all refinements of those plans leave the literals established at the abstract level unchanged."*
– *Downwards Refinement* [10] which guarantees that an abstract plan can be refined into a plan at the next level of abstraction down.

Even when these hold, hierarchies constrain the set of solutions considered [9], which restricts creative solutions. To counter these difficulties, methods for automatically generating efficient hierarchies have been proposed [9]. These techniques require the specification of a set of available operators for a given domain. However within the context of our research a complete or comprehensive set of available processes is difficult to obtain, as discussed above in Subsection 2.1. This makes (convincing) top-down decomposition challenging, if not impossible: consider, incomplete information regarding available processes within a given domain not only hinders matchmaking, it also frustrates the manual creation of appropriate (task) hierarchies.

An interesting development, presented in [11], is to have an agent which *intelligently* coordinates a response to an identified customer need. These agents, however, are used in a conventional top-down fashion: they procure solutions using an auction analogous to a CNA; and the subtask allocation is derived from the top-level goal in a traditional manner. While this an iterative approach and can address multiple criteria, in our opinion, it sacrifices many of the benefits afforded by peer-to-peer negotiation: to which we appeal to in our approach.

A Bottom-Up Approach Our approach, expounded in Section 3, is inspired by the notion of a *Blackboard System (BBS)* [12, 13]. A BBS formalises the metaphor of a group of experts working on a problem and communicating ideas using a blackboard. The blackboard becomes a repository of information which is globally accessible and records the problem specific information available from each expert. The flow of information between the blackboard and each expert is bidirectional: an expert both contributes and extracts information. Ideally, the contributions are revised, extended and restructured to yield a solution to the problem. Traditional implementations make use of a single, central blackboard and a scheduler, the *focus mechanism* or *blackboard control*, which manipulates the flow of data and the order in which the experts are consulted. Multi-agent systems (MAS), however, allow this to be extended [14]. In particular, each expert can have greater autonomy and communicate with others directly in addition to using the blackboard (control) as an intermediary.

3 A Notice Board Approach to Virtual Organisations

We propose a novel agent-based approach which uses notice boards and peer-to-peer negotiation to address the formation of a VO and the coordination of a cross-organisational business process within congregations. Each notice board serves as both a repository of information and as a communication channel. Combined with appropriate negotiation protocols, this promotes an *emergence* of solutions in which the VO and its process

are co-designed in a dynamic and automated manner. We see this as a complementary approach to conventional top-down decomposition.

3.1 Problem Set Up and Consequences

Typically the problem of supporting the formation of virtual enterprises has been considered to be broadly equivalent to that of constructing a composite web service, namely constructing the single best team to solve a specific problem. Within a VBE, such as the Austrian automotive cluster, VO formation typically occurs within the following context:

1. A large company external to the VBE decides to seek an external supplier for a specific part or service and makes an open offer for bids. In the case of the automotive domain this corresponds to an OEM seeking a supplier for a given module.
2. The existence of this opportunity is made visible to members of the VBE and this motivates the formation of one or more potential VOs.
3. The potential VOs plan in sufficient detail to allow them to tender for the proposed contract and proceed to do so.
4. The original external company chooses their preferred VO for the contact.
5. The chosen VO then makes truly detailed plans and moves into actual production. In the case of the automotive domain this step involves several stages of prototype design and typically takes several months before full scale production can commence.

A system aiming to support virtual organisation formation within this context should not try to support the production of a single, 'best' team but rather to support the general formation of virtual organisations. Indeed it is arguable if the concept of the 'best' VO has any meaning during the VO formation phase: the final evaluation criteria are not known and so any evaluation criteria must arise within the VBE.

As there is no entity within the VBE with the authority to impose an evaluation scheme, any evaluation scheme must arise through the combined judgement of the companies within the VBE. The natural self interest of these companies makes it likely that they will differ in their judgement of the worth of particular teams, with a particular preference for teams of which they are a member. A proposal that a supplier should be allowed to join an existing, partially formed, team can consequently only be fairly evaluated by the existing members of that team. Thus a system to support VO formation within a VBE should support the formation of multiple teams while allowing the existing members of a partially formed VO to evaluate any potential further contributions to that VO. The approach proposed within this paper offers support for both of these features.

The context above also affects the nature of the output of any system supporting VO formation. In contrast to many related systems, and especially to those composing web services, there is a considerable period of time between the system identifying a potential VO and that VO moving into production. A set of companies produced as the output of such a system is thus not required to have a detailed, immediately executable

plan to meet the business opportunity as a VO,[1] but rather to believe that they will be able to produce such a plan. Delaying the commitment of companies to specific processes as long as possible allows the process of VO formation to retain maximal flexibility and is thus desirable.

The system proposed within this paper allows companies to propose contributions using a formalism which focuses purely on their intended effects, rather than the process used to achieve those effects. This allows for a full exploration of the space of potential virtual organisations. In consequence the system cannot provide an absolute guarantee that a complete solution produced can later be developed into a virtual organisation. In the current context this corresponds to a team entering negotiations to prepare a bid and failing to agree on what bid to propose. While this is undesirable in principle there are no major costs associated with this in this context. Additionally the complexity of the bid preparation process means that no system could ensure that its complete solution could always be completed. Finally the system contains several features which mitigate the chances of this occurring.

In summary the system the principal goal of the system proposed within this paper is to facilitate the overall formation of virtual organisations within a virtual breeding environment. It achieve this goal by producing sets of companies who are likely to be able to combine to meet the goals of the business opportunity.

3.2 Notice Boards and Partial Solutions

To structure our exposition, it is convenient to distinguish three stages in the establishment of a VO within a congregation: a request is made visible to the agents within the congregation; a team (or set of teams) to fulfil a given request *emerges* as agents volunteer contributions and agree to provide (sub)services; and a solution *emerges* from the (final) coordination of actions among team members. We examine each of these in more detail. We first clarify the notions of partial and complete solutions; and formally define a notice board.

Definition 1. (Partial Solution) *Suppose we have a consistent set of goals. A* partial solution *is any set of tasks (or activities) which satisfies a subset of our set of goals. This is in contrast with a* complete solution *which is a set of tasks (or activities) which satisfies all of our goals.*

Remark 1. A global goal can be represented as a conjunction of one or more subgoals. This applies independently of how such a set of subgoals is achieved: whether through a top-down decomposition or as an adjunct to a bottom-up approach. Accordingly, we overload the term partial solution to apply more informally to any set of activities arising in the pursuit of complete solution to an identified global goal: here the subgoals are left implicit.

Remark 2. A complete solution is necessarily a partial solution. However, a complete solution is (typically) not a unique combination of partial solutions.

---

[1] As would be required in for instance web service composition

Definition 2. (Notice Board) *A notice board is a communication device which contains the following information:*

1. *A precise statement of the top-level goal or service for which the VO is to be assembled and for which a global process is sought. The notice board goal identifies which request the notice board represents.*
2. *Utility information defining the worth of the request. The utility information for the request includes the information required for each agent to decide on the perceived worth of the request. This might include the company issuing the request and the number of items requested.*
3. *A set of partial solutions (cf. Definition 1).*

For convenience, we denote the above categories of information $\mathscr{G}, \mathscr{U}$ and $\mathscr{S}$, respectively; and we denote the corresponding notice board $\mathbb{NB} = \langle \mathscr{G}, \mathscr{U}, \mathscr{S} \rangle$.

In addition to recording the evolution of a set of partial solutions (ideally into a complete solution), a notice board serves in the coordination of information exchange during team formation.

Making a Request Visible Requests are made visible to agents within the congregation through a system of notice boards. A notice board is created for every request which comes into the system.

Team Formation A teams develops incrementally: initially it can offer only a partial solution (to a request); the team grows as additional members join and contribute towards a complete solution. This is driven by the agents' monitoring notice boards within the system, looking for promising partial solutions to extend and proposing extensions where appropriate [2]. A proposed extension triggers negotiation among the existing team members and the proposing agent (*candidate*): this concerns the admissibility of the candidate as judged by the existing team members; a simple model might involve a blacklist reflecting the preferences of the existing members, where if the candidate is not on this, then he is admissible; more sophisticated models would also consider how well processes integrate. To facilitate team formation a notice board includes information on the partially formed team as part of a partial solution. Thus, the noticeboard also enables the team extension negotiations. At this stage each partial solution is owned by a group of companies who are provisionally happy to work together and have each promised to provide a process to the team. The details of the processes promised are left purposefully vague to maintain the maximum possible flexibility. A possible partial solution representation technique is presented in in Subsection 3.4.

---

[2] We do not discuss the technical details of creation of notice boards, nor how an agent is made aware of notice boards in the system. One possibility is that once a agent receives a service request, it arranges for a notice board to be created. This notice board is wrapped by an agent, which registers with the *Directory Facilitator* and makes a broadcast message to notify agents of its creation.

Final Coordination This final stage is triggered once a partial solution on a notice board has been extended into a complete solution. In this case we have a team of agents who expect to solve the problem. Nevertheless, the actual details of how the team will work and interact are yet to be made precise. As our aim in the present paper is to present a mechanism for emergent process interoperability arising from VO formation in congregations, we consider this stage out of scope of the present discussion.

Nevertheless, we note that this stage presents significant challenges and is likely to require much human input. *Inter alia* the following issues will need to considered:

1. A precise product specification. This will include details of raw materials, component interfaces, tolerances, etc. In some cases this will include explorations to arrive at a detailed configuration of part of the final product, which can only be considered once substantial detail is available; and may involve further negotiation [15].
2. A detailed statement of each agent process (cf. local workflow) to a level sufficient to support the composition of a global process (cf. global workflow).
3. The actual global workflow for the combined process.
4. Contractual matters including the division of proceeds; timelines; change request management; recourse upon contractual breaches; etc.
5. If the team is assembled to respond to an invitation to tender, then a price needs to be agreed.

Potentially, these stages are interdependent. Consider, a company might be willing to modify its processes to fit better with the processes within the team only if it is given a greater share of the revenue. Moreover, contractual details and price agreement will generally require human intervention. As such, a full automation of this stage seems unlikely in all but very simple cases. A more realisable objective would be to reduce the onus on the (human) user, by automating the more routine tasks involved.

3.3 Summary

The following set of pictures offers a useful summary guide to the overall operation of the bottom up team formation system. The first depicts a newly created noticeboard, and later pictures depict the stages of solution development. For clarity only the creation of one, simple, complete solution is shown. In the general case multiple solutions of differing forms would be explored.

As an aid when considering these pictures consider the opportunity to require the production of a water tank. Agent A is then a large company which can produce the water tank if someone can supply them with a specifically configured reservoir - this is represented by the resource in the intermediate state. Agent B can supply such a reservoir, and so offers to do so. This triggers negotiations between the two companies which conclude successfully, and the extension request is granted. This then produces a complete solution. The companies within this solution are then notified and move onto a more detailed discussions of how they wish to work together.

3.4 Technical Details

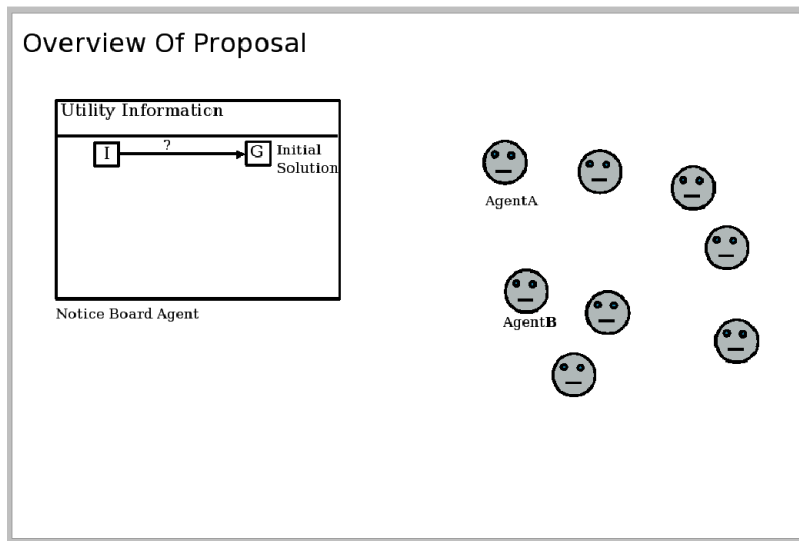We present and discuss some of the more technical aspects of our proposal.

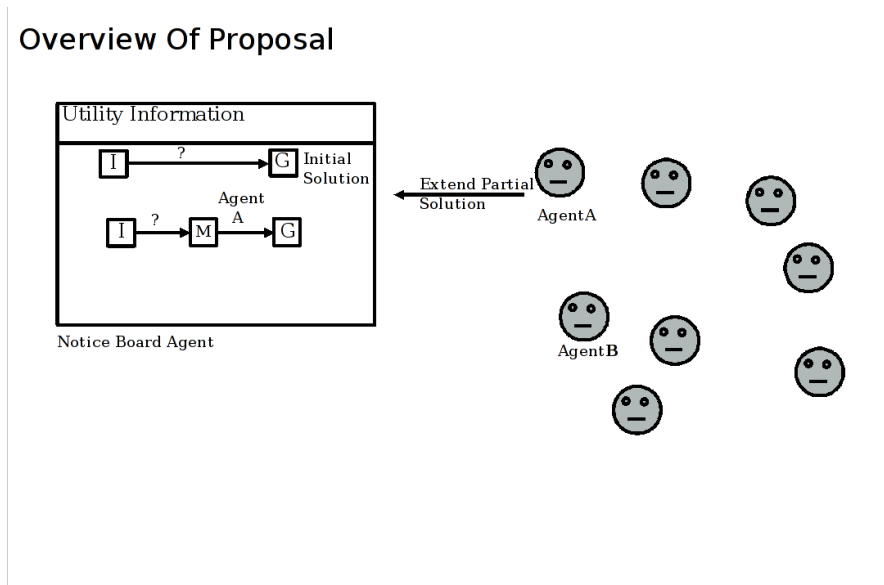**Fig. 1.** The opportunity is noticed and a noticeboard is created

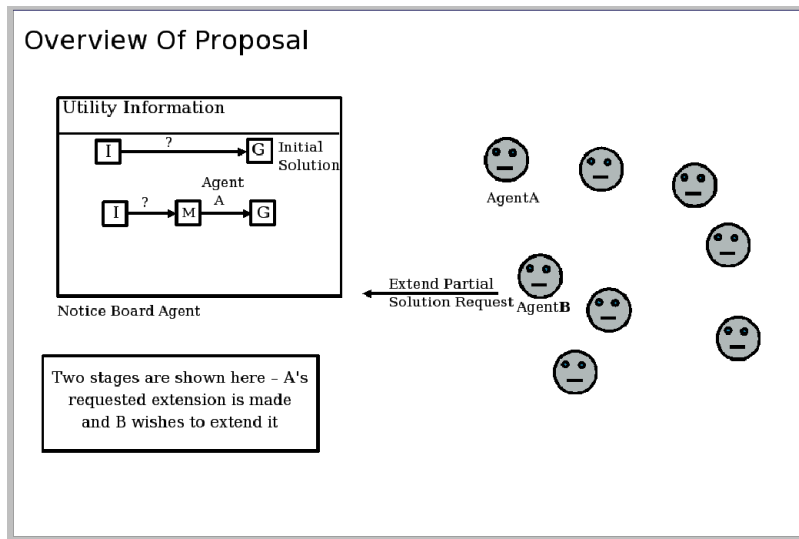**Fig. 2.** Agent A extends the initial solution

Overview Of Proposal

Utility Information

I $\xrightarrow{?}$ G Initial Solution

Agent A

I $\xrightarrow{?}$ M $\rightarrow$ G

Notice Board Agent

Extend Partial Solution Request

AgentA

Agent**B**

Two stages are shown here – A's requested extension is made and B wishes to extend it

Fig. 3. Agent B wishes to further extend the extended solution

Overview Of Proposal

Utility Information

I $\xrightarrow{?}$ G Initial Solution

Agent A

I $\xrightarrow{?}$ M $\rightarrow$ G

Notice Board Agent

Trigger Extension Protocol

AgentA

Negotiation: is extension acceptable?
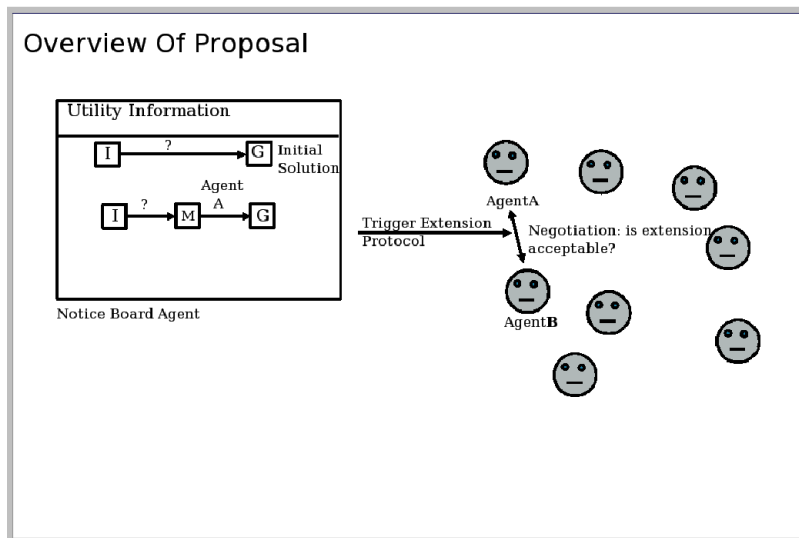
Agent**B**
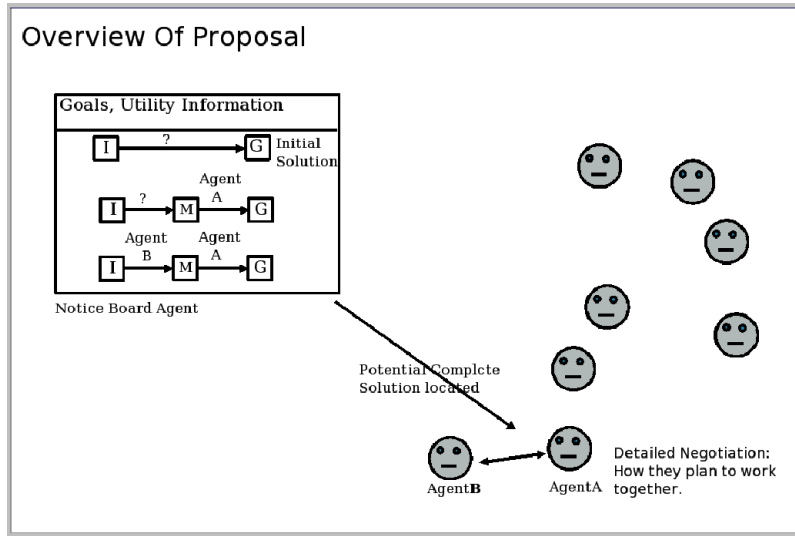
Fig. 4. The proposed extension is verified

Fig. 5. A complete solution has been located

Representation The representation of partial solutions is facilitated through creation of an ontology to realise a vocabulary to describe the objects and states (of interest) which exist within the application domain, together with a constraint language. A partial solution is represented as a set of states connected by transitions. Each transition is labelled with either the name of the agent which has promised to supply it or a question mark ("?") to indicate that it is indeterminate. We refer the latter case as an *unlabelled* transition. Each state is represented by a set of objects from the domain ontology together with constraints on their attributes.

Extending Partial Solutions To extend a partial solution new states are added and connected to existing states using appropriate transitions. There are two basic ways in which to extend an existing partial solution: a *right extension* (cf. forward chaining), where a given state S is transformed into a state S' by a transition T, $S \xrightarrow{T} S'$; and a *left extension* (cf. backward chaining), where a given state S is derived from a state S' by a transition T, $S' \xrightarrow{T} S$; naturally, there is the possibility of an extension achieving both of these simultaneously. We illustrate these, without loss of generality, from a goal process: the goal process of the notice board is represented by a pair of states, the start and the goal state distinguished for the case-in-hand, connected by an unlabelled transition. For instance the following goal process (partial solution) indicates that the goal is to repaint a green ball red. [3]

$$(ball(colour == green)) \xrightarrow{?} (ball(colour == red))$$

---

[3] The system described in the later case study uses a more complex, non linear state representation. A simpler one is used below for clarity of exposition.

Suppose an agent is prepared to strip the paint from that ball as a potential intermediate step, then this is a right extension.

$$(ball(colour == green)) \xrightarrow{PaintStripperAgent} (ball(colour == blank))$$
$$\xrightarrow{?} \quad (ball(colour == red))$$

Alternatively, suppose an agent can paint a blank ball red as a potential intermediate step, then this is a left extension.

$$(ball(colour == green)) \xrightarrow{?} (ball(colour == blank))$$
$$\xrightarrow{PainterAgent} (ball(colour == red))$$

Naturally, if an agent can provide both of these, we have a (simultaneous) left and right extension, which in this case yields a complete solution

$$(ball(colour == green)) \xrightarrow{Paint\&StripAgent} (ball(colour == blank))$$
$$\xrightarrow{Paint\&StripAgent} (ball(colour == red))$$

The idea of using such a resource language to describe a distributed planning problem is taken from [16] where a similar system is used to merge plans in an efficient manner.

Learning Opportunities In the basic system above the notice board serves only to coordinate the communications within the system. Uplifting a noticeboard to an agent status offers an opportunity to extend its role. For example, one possibility would be to enable the reuse of previous partial solutions in a manner analogous to case based planning, see e.g. [17]. The notice board agent could persist beyond the life of the project and respond to related future requests. Alternatively, the information could be used to inform the appropriate ontological structures and problem-solving mechanisms; or passed onto an agent responsible for maintaining, indexing and answering queries concerning previous partial solutions. Such reuse is useful both for speeding the generation of new virtual organisations and for the speedy location of substitute organisations if a problem occurs with one of the partners after the team has formed.

Requirements on Agents The system outlined above devolves significant responsibility upon the process agents and thus demands of them several types of reasoning. The primary requirement on an agent is that it contribute processes to the partially formed virtual organisations. This splits into two parts, namely:

1. Identifying when it wants to contribute a process. This motivates the inclusion of utility information, i.e., data which can be used to inform an agent's preferences. This is peculiar to a given domain.
2. Recognising when it can make a useful contribution. An agent must be cognisant of its own capabilities and be able to query a notice board to determine states which can form (part of) an input to an owned process. Thus, an agent concentrates on expressing *what* it can do, such as paint stripping (a declarative statement) rather specifying *how* it achieves its activities (a procedural or operational statement).

Finally were the process provider agents to contribute one of their processes whenever it was possible for them do so there would be a real risk of an undesirable quantity of partial solutions being produced. Thus the process provider agents should possess some measure of the degree of "usefulness" of its processes to a given partial solution. Within the current context of automotive manufacturing such knowledge is naturally present - a company will know the sorts of product in which their processes are typically used.

4 Case Study

To explore the technical feasibility of the notice board system, a testbed was created using JADE (`jade.tilab.com`). This test bed implemented a technically complete version of the noticeboard system and has been run on many small case studies.

Concepts and Problem-Solving Methods (PSMs). The chosen case study within this paper was inspired by one encountered within the Crosswork European research project. The basic concept within the case study is the creation of a virtual organisation to produce a watertank.

Overall there are five agents within study, including:

1. A watertank constructor agent who can construct the type of watertank required if provided with certain components. (a Grommet, a reservoir, a lid and a pump)
2. A set of agents who each provide one of these required components

4.1 TestBed

The case study demonstrates one of the simple ways in which noticeboard virtual organisation can operate. In this case the scenario flow is:

1. A virtual organisation must be found to produce a watertank
2. A corresponding noticeboard is created
3. The watertank constructor notices that it can construct this kind of watertank and adds an appropriate partial solution to the noticeboard
4. Companies who can provide the individual components notice this and offer to do so, thus eventually forming a complete solution

The initial solution within this case study is simply : $[\phi] \xrightarrow{?} [Watertank]$.

The partial solution generated once the watertank constructor agent has extended this initial solution can be seen in diagram 6. Finally the complete solution can be seen in diagram 7.

This scenario demonstrates how backwards chaining works within a noticeboard system - essentially companies within the system recognise that they could make some form of product if they were supplied with certain other products.

This differs from traditional top down approaches since the knowledge used to decompose the watertank into a set of subcomponents here comes purely from the watertank constructor agent it self.
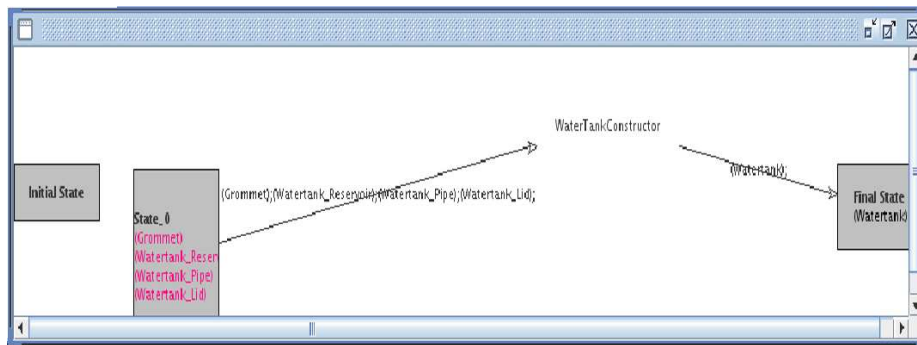
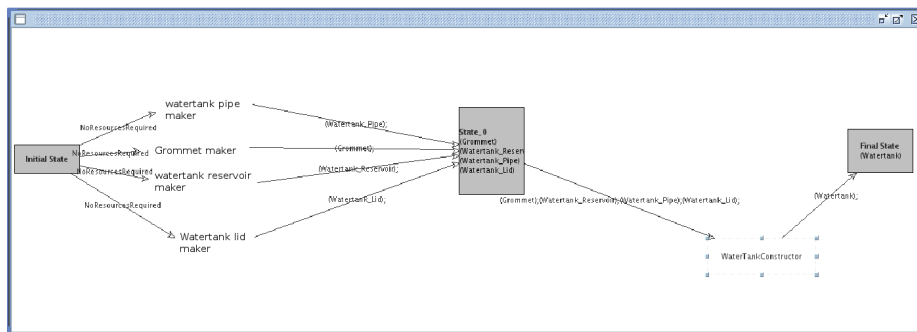**Fig. 6.** The initial partial solution

Fig. 7. The final solution

Indeed the set of components that it needs only includes those components for which it needs an external contractor. A different watertank assembly company, or the same company producing a different form of watertank, might need to be supplied with a very different set of subcomponents.

The noticeboard system also supports forwards chaining - in this case an agent recognises that a process they can provide is likely to be useful for the type of product being produced and offers it. For example here the producer of watertank reservoirs might notice a solution involving the production of watertanks and offer to supply relevant forms of reservoir.

4.2 Implementation and Results

The system was implemented using a (single generic type of) process agent. This agent queries a notice board agent for partial solutions to extend. Since the focus was on core technical feasibility a single notice board agent was created and the tests stopped once a single complete solution had been generated. In addition the process provider agents used contained simplified reasoning - each was responsible for one specific process and were totally cooperative. Thus, while all of the required negotiation stages actually took place, they were always successful. None the less the tests provided a good measure of

the technical progress of the implementation of the system. Indeed these tests emphasised the importance of controlling the numbers of partial solutions generated within the system.

5 Conclusion

Congregations [18] capture the stable yet open business ecosystems in which VOs often form to capitalise upon business opportunities arising in dynamic and complex markets. Conventional approaches to formation of VOs in such congregations typically yield suboptimal solutions. We have proposed a novel notice board based approach to workflow composition which we believe offers a powerful, complementary approach. Through this VO processes and VO structure are co-optimised. The power of our approach derives from a focus on peer-to-peer negotiation. This encourages process interoperability to *emerge* as a result of shared interests of agents (cf. entities, institutions) within a network of complementary expertise. In particular, we have indicated how this circumvents a number of problems which derive from conventional approaches to VO formation, especially within congregations. The simple case study fosters confidence in the core ideas of the approach and highlights areas of further work. Additionally, we plan to investigate situations where this approach works in tandem with conventional top-down decompositions, allowing more flexible search solutions.

References

1. Camarinha-Matos, L.M., Afsarmanesh, H.: Elements of a base ve infrastructure. Computers in Industry **51** (2003) 139–163
2. Mowshowitz, A.: Virtual organization. Commun. ACM **40**(9) (1997) 30–37
3. Faisst, W.: Information technology as an enabler of virtual enterprises: A life -cycle-oriented description. Proceedings of the European Conference on Virtual Enterprises and Netowkred Solutions (1997)
4. Consortium: Intra- and inter-organisational business models. Public Deliverable to work package 1 Crosswork European research project (2004)
5. Afsarmanesh, H., Camarinha-Matos, L.: A framework for management of virtual organization breeding environments. In CamarinhaMatos, L., Afsarmanesh, H., Ortiz, A., eds.: Collaborative Networks and Their Breeding Environments. Volume 186 of International Federation for Information Processing., Springer (2005) 35–48
6. Consortium, E.: D22.1 key components, features and operating principles of the virtual organisation breeding environment (2005)
7. Sen, S.: Believing others: Pros and cons. Artificial Intelligence **142**(2) (Dec 2002) 179–203
8. McIlraith, S., Son, T., Zeng, H.: Semantic web services. IEEE Intelligent Systems and their applications **16**(2) (APR 2001) 46–53
9. Knoblock, C.A.: Automatically generating abstractions for planning. Artificial Intelligence **68**(2) (1994) 243–302
10. Bacchus, F., Yang, Q.: The expected value of hierarichical problem solving. In: National Conference on Artifical Intelligence (AAAI-92). (1992) 364–374
11. Oliveria, E.: Agents' advanced features for negotiation and coordination. Mutli-agents systems and applications (2001) 173–186

12. Carver, N., Lesser, V.: Evolution of blackboard control architectures. expert systems with applications **7**(1) (Mar 1994) 1–30
13. Hayes-Roth, B.: A blackboard architecture for control. Artif. Intell. **26**(3) (1985) 251–321
14. Craig, I.: Blackboard Systems. Ablex, Norwood, NJ (1995)
15. Lander, S.: Distributed search and conflict management among heterogeneous reusable agents. Ph.D. thesis, University of Massachusetts, Amherst (May 1994)
16. de Weerdt; Andre Bos; Hans Tonino;, M., Witteveen, C.: A resource logic for multi-agent plan merging. Annals of Mathematics and Artificial Intelligence; volume 37; Issue 1-2 (2003) 93 – 130
17. Spalzzi, L.: A survey on case-based planning. Artif. Intell. Rev. **16**(1) (2001) 3–36
18. Brooks, C.H., Durfee, E.H.: Congregation formation in multiagent systems. Autonomous Agents and Multi-Agent Systems **7**(1-2) (2003) 145–170