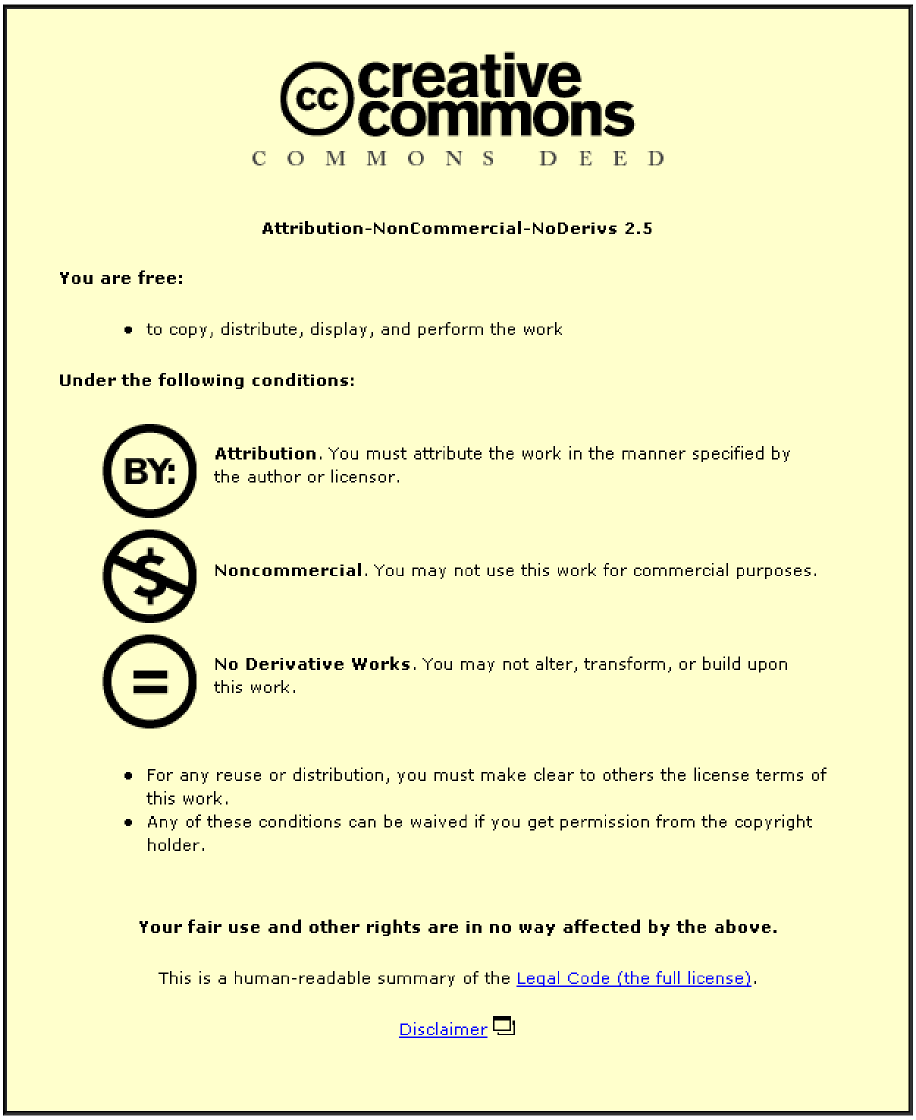


# Cross-Organisational Workflow Enactment via Progressive Linking by Run-Time Agents

Xi Chen[1], Paul Chung[1]

[1] Computer Science Department, Loughborough University
Loughborough, Leicestershire, United Kingdom LE11 3TU
{X.Chen, P.W.H.Chung}@lboro.ac.uk

**Abstract.** Driven by popular adoptions of workflow and requirements from the practice of virtual enterprise (VE), research in workflow interoperability is currently on the increase. Nonetheless, it is still in its early stage compared with the maturity of individual workflow technology. Some attempts have been tried, however results are not satisfactory especially in a VE context, where many of the partnerships are dynamic and temporary. Reasons include the rigidity and high initial coordination cost inherently associated with top-down modelling and enactment approaches. Therefore, this paper proposes a bottom-up and WfMS [1] -independent approach towards cross-organisational workflow enactment, which is via progressive linking enabled by run-time agents. This is expected to pave the way for further cross-organisational workflow needs.

**Keywords:** Multi-Agent Systems, Workflow Interoperability, Virtual Enterprise

## 1 Introduction

Business processes are at the core of productivity for an organisation. They control and describe how business is conducted in terms of “a set of one or more linked procedures or activities which collectively realise a business objective or policy goal, normally within the context of an organisational structure defining functional roles and relationships” [1]. To support mobility and dynamism, individual business processes are vital for a company to react faster and be more flexible in running its daily business in a constantly changing environment [2]. However, the idea of virtual enterprises (VE) blurs the boundaries between organisations and requires cross-organisational interactions, which brings in many challenges.

Workflow was born to tackle the issue of business process automation and is proven, to date, as a mature technology. It is carried out with the support of workflow management system (WfMS) that provides complete design, execution and management services to workflow. The essential strategy of workflow is the separation of business logic from software applications. Although being separated,

[1] WfMS is the acronym for workflow management system [1].

they are still linked in the form of ‘activities’ that represent logical steps within a process [1]. Centred around a set of activities, an activity-based workflow is constructed in contrast to an entity-based workflow [3]. Each activity can be either manual or automatic depending on the task to be carried out, where the automatic ones are, mostly, implemented as applications.

The problem of workflow interoperability has been identified due to adoptions of diverse WfMS products between organisations and the inevitability of interconnections for the purpose of cooperation across organisational boundaries. Three basic interoperation patterns, namely, chained process, nested synchronous sub-process, and event synchronised process [4] should be tackled, among which more concerns are put on the first two [5]. Due to inherent complexity, event synchronisation, although encountered very often in real business, has not received much attention by WfMC itself [6]. A number of research projects have been carried out in the area of workflow interoperability, nevertheless results are not satisfactory when applied in a VE context because of their rigidity and the high initial coordination cost imposed by top-down approaches. This project proposes a more effective approach, which addresses enactment of cross-organisational workflow from a bottom-up view and in the form of a progressive linking mechanism supported by run-time agents. It is expected the success of such an approach will shed light on and facilitate the formalisation and execution of cross-organisational workflow. This paper identifies challenges of workflow enactment in the context of VE, describes the approach, presents a possible way of implementation, discusses its effectiveness and highlights the future work.

## 2 Challenges

Workflow interoperability is commonly examined from a top-down view [7,8,9], which intends to start from the concept of traditional workflow and extend it beyond organisational boundaries in order to keep the control flow manageable. However, as a technology-driven approach, it brings in much initial cost in terms of a detailed and rigid pre-definition that does not reflect the run-time nature of agile interactions within VEs where many of the partnerships are dynamic and temporary. However, if centralised control is removed due to the choice of a bottom-up modelling approach, there seems to be no way of propagating control flow from one workflow to another at run time.

Real-life interoperation always poses a tightly-interwoven control flow structure. Also, many existing business processes are mobile and ever-changing because of their dynamic nature. The dynamism should be dealt with effectively by the cross-organisational workflow to minimise disturbance to cooperation, which implies a loosely-coupled interaction mechanism. Therefore, the realisation of tightly-interwoven processes by means of a loosely-coupled mechanism is identified as a challenge.

According to the Workflow Reference Model [6] initiated by WfMC, Interface 4 is the standard interface dedicated for the purpose of interoperability and has attracted much attention. Although standardisation provides a solution with regard to interoperability, the practical value is discounted in the face of the diversity of standards [10] and the reality towards their acceptance. Therefore, standardisation cannot be fully relied on.

Moreover, approaches that require substantial effort, e.g., dialogue definition [5], workflow view [7,11], Interworkflow [9], agent-based workflow [12], and standardisation [13,14], are unlikely to be adopted widely in the near future.

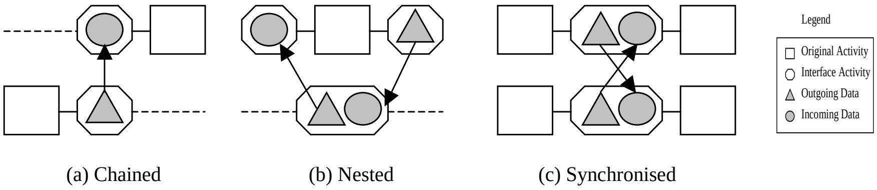
## 3 Progressive Linking Approach

For the purpose of simplicity, interoperation discussed in this section is confined to participation between two organizations but not any particular two. It is assumed that when a workflow is invoked it will always instantiate a new process instance.

### 3.1 Interoperation Modelling

At least, three aspects, namely, control flow, data flow and communication, should be addressed in order for two workflows to interoperate with each other. For control flow, other measures must be taken to route control due to the lack of a centralised architecture. To facilitate the discussion, the concept of interaction point is introduced here. An interaction point can be defined as from (or to) which, a request (or response) is emitted (or targeted). Since workflow engines are state machines, an appropriate sequence of interaction can be ensured at run time [7] as long as interaction points are correctly specified in both participating processes. Synchronisation is achieved by a process sending request and waiting for a response from the other process[5]. Activity-level modelling for interaction point is chosen in order to make the approach adaptive to all interoperability patterns. An interaction point is therefore modelled as an interface activity, which is further implemented in the form of a generic workflow activity. This activity is configured to synchronously (letting the process wait while the application is executing) invoke software agent as an external run-time application, which makes it an agent-enhanced approach [15]. Control token [16] is passed back and forth among WfMSs and agents.

Data flow is managed by the semantics of 'sending' and 'waiting', which are implicitly indicated by the type (incoming or outgoing) of data being exchanged by the two workflow engines through this interface activity. Basic interoperability patterns are all modelled at the activity level by employing the concepts of interface activity, incoming and outgoing data, which is illustrated in Fig. 1.



(a) Chained (b) Nested (c) Synchronised

**Fig. 1.** Interoperation patterns modelling using interface activities

Mediated communication [7] is used due to the loosely-coupled approach is adopted. Semantic service descriptions extracted from individual processes are used for message routing. Thus, agents on the one side pass outgoing data from the interface activity to the mediator, whilst agents on the other side check whether desired incoming data arrives and deliver it to the corresponding interface activity accordingly.

### 3.2 Compatible Workflow

At build time, based on a common agreement, the workflows involved should be modelled and tuned into compatible ones and interface activities are inserted into the processes at desired positions to make both processes ready to go. Semantic service descriptions (in the form of interaction identifiers) are also attached to each pair of outgoing and incoming data belonging to the interface activities on both sides.

### 3.3 Form Filling

At run time, an empty form is created and two compatible workflows begin their interoperation by filling the form jointly in sequence. Using Fig. 1 (a) as an example, when the first interface activity (only containing outgoing data) from Organisation A is executed, Agent A is invoked and puts the source activity ID (A2), interaction identifier (PurchaseOrder), data (PO200511) and attached document (if any) associated with this activity as well as the identifier of the partner process (Process B) into the form. The occurrence of information in the form triggers Agent B who arranges the instantiation of a new process on the side of Organisation B. After being instantiated, Process B reaches its first interface activity (only containing incoming data) who calls Agent B to register its interaction identifier as a mark of interest. Agent B looks at the form and checks whether there is such an identifier in an unfinished entry. If yes, it writes the activity ID (B1) into the entry and transfers the data and the document (if any) to Process B for consumption. Table 1 gives the headings of the form and an example entry.

**Table 1.** Heading of communication form and an example entry

| Seq. | Source | Target | Interaction ID | Message | Doc | Iteration |
|---|---|---|---|---|---|---|
| 1 | A2 | B1 | PurchaseOrder | PO200511 | attached | NIL |
| ... | | | | | | |

By doing so, the form shows the current progress of the interoperation. Apart from making a loosely-coupled structure possible, in case of exception, it can be used to trace and locate the trouble spot. When all interactions finish, the completed form can be saved as a historical record.

The progressively filled form is also able to handle event synchronisation and iterative cases effectively through reasoning based on recorded data and progressive status of the form filling. For example, the appearance of two successive uncompleted entries with a blank Source and Target field in each means a rendezvous point is reached; a completed entry with the Source and Interaction ID fields exactly the same as the ones in a previous entry implies an occurrence of iteration, in which case, the entry needs to be marked in the field of Iteration to draw the attention to the recipient.

## 4 Implementation

Implementation of the approach is underway. A client/server system archetecture is chosen. General characteristics of WfMS are fully exploited in order to achieve a WfMS-independent solution and avoid undue complexity. Both interoperation triggering and acceptance will utilise the workflow application invocation mechanism. To support the mediated communication among software agents, a blackboard system [17] will be adopted. This is due to the structure and functionalities provided by the blackboard system architecture match the proposed approach very well. Knowledge sources (KSs) can be implemented as agents on the client side whilst the blackboard (BB) can be used to hold the form on the server. KS-trigger mechanism can be used as well to bring attention to agents on both sides when something happens to the form.

## 5 Discussion and Future Work

The approach of progressive linking is developed by using artificial intelligence technology based on a comprehensive investigation of workflow model in terms of control flow, data flow, activity model and application invocation mechanism. It addresses the challenges identified in Section 2. Firstly, this approach reflects realistic cooperation between processes. The complexity of cross-organisational control flow is wrapped in the procedure of invoking mediated software agents, which enable interoperation without a centralised control mechanism. Secondly, loosely-coupled interaction mechanism is provided by the run-time progressive interaction. The tightly-interwound process is dealt with by activity-level interaction modelling that provides a general method for all interoperability patterns. Thirdly, interoperability standards are avoided as much as possible in terms of the agent invocation through application invocation interface rather than workflow interoperability interface. Finally, since the substantial work is implemented in the form of external software agents, there is no structural change imposed on involving WfMSs, which brings in a WfMS-independent solution.

However, since this approach relies on compatible workflows, the issue of compatibility has yet to be addressed. Obviously, cross-organisational workflow compatibility cannot be solved by means of cross-organisational workflow enactment alone but the idea of progressive linking paves the way for a possible direction for achieving it by letting the intelligent agents progressively negotiate the flow of interoperation from scratch within an intelligent framework for cross-organisational cooperation. Internal processes are exposed as services, which allows software agents to negotiate and pick up the desired ones on the fly to dynamically construct cross-organisational workflow. These will be addressed in the future work. It is expected that the progressive linking approach enabled by run-time agents will facilitate intelligent interoperation that will benefit B2B e-business among VEs.

## References

1. Workflow Management Coalition: Terminology & Glossary, Technical Report WFMC-TC-1011. Workflow Management Coalition (1999)
2. International Business Machines Corporation: IBM WebSphere MQ Workflow Version 3.6 – Concepts and Architecture. 8th edn. IBM Corp. (2005)
3. Guillaume, F.: Trying to Unify Entity-based and Activity-based Workflows. < http://www.zope.org/Wikis/DevSite/Projects/ComponentArchitecture/TryingToUnifiyWorkflowConcepts>. (accessed 3.7.2005)
4. Workflow Management Coalition: Workflow Standard – Interoperability Abstract Specification, Technical Report WFMC-TC-1012. Workflow Management Coalition (1999)
5. Biegus, L., Branki, C.: InDiA: a Framework for Workflow Interoperability Support by Means of Multi-Agent Systems. Engineering Applications of Artificial Intelligence 17 (7) (2004) 825-839
6. Workflow Management Coalition: The Workflow Reference Model, Technical Report WFMC-TC-1003. Workflow Management Coalition (1995)
7. Schulz, K.A., Orlowska, M.E.: Facilitating Cross-Organisational Workflows with a Workflow View Approach. Data & Knowledge Engineering 51 (1) (2004) 109-147
8. Leymann, F., Roller, D.: Production Worklow – Concepts and Techniques. Prentice Hall, New Jersey (2000)
9. Workflow Management Coalition: Interworkflow Application Model: The Design of Cross-Organizational Workflow Processes and Distributed Operations Management, Technical Report WFMC-TC-2102. Workflow Management Coalition (1997)
10. Bernauer, M., Kappel, G., Kramler, G., Retschitzegger, W.: Specification of Interorganizational Workflows – A Comparison of Approaches. Vienna University of Technology White Paper. Vienna (2002) < http://www.big.tuwien.ac.at/research/publications/2003/0603.pdf>. (accessed 1.11.2005)
11. Chiu, D.K.W., Cheung, S.C., Karlapalem, K., Li, Q., Till, S.: Workflow View Driven Cross-Organizational Interoperability in a Web-service Environment. Information Technology and Management 5 (2004) 221-250
12. Jennings, N., Norman, T., Faratin, P., O’Brien, P., Odgers, B.: Autonomous Agents for Business Process Management. International Journal of Applied Artificial Intelligence 14(2) (2000) 145-189
13. Workflow Management Coalition: Workflow Standard – Interoperability Wf-XML Binding, Technical Report WFMC-TC-1023. Workflow Management Coalition (2001)
14. O'Riordan, D.: Business Process Standards For Web Services. < http://www.webservicesarchitect.com/content/articles/BPSFWSBDO.pdf>. (accessed 6.7.2005)
15. Shepherdson, J., Thompson, S., Odgers, B.: Cross Organisational Workflow Coordinated by Software Agents. Proceedings of WACC Workshop on Cross-Organisational Workflow Management and Co-ordination, San Francisco. (1999)
16. Aalst, W.v.d. and Hee, K.v.: Workflow Management: Models, Methods, and Systems. MIT Press, Cambridge (2002)
17. Corkill, D.: Blackboard Systems. AI Expert 6(9) (1991) 40-47