Sindh Univ. Res. Jour. (Sci. Ser.) Vol.46 (1) 87-92 (2014)

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Modelling Chemical Plant Construction VO in VOML

NOOR JEHAN RAJPER

Institute of Mathematics and Computer Science, University of Sindh, Jamshoro. email: noor_rajper@hotmail.com

Received 5th January 2014 and Revised 26th March 2014

Abstract: The VOML **(**Virtual Organization Modelling Language) is framework for specifying virtual organizations (VOs) and their breeding environments (VBEs). It consist of the sub-languages VO-S, VO-O and VO-R to model structural, operational and evolution aspects of the VOs respectively. To show that VOML lends itself well to model virtual organizations of different complexity and different nature, it is applied to specify an actual Chemical Plant Development case study conducted as part of the GOLD project. This paper presents the modelling of structural and reconfiguration aspect of the case study using the VO-S and VO-R languages respectively. This work is conducted as part of PhD dissertation.

Keywords: Virtual Organizations, Modelling languages

I. INTRODUCTION

The demand for Virtual organizations (VOs) to successfully achieve targets even in disruptive and hostile environment by evolving and adapting to the changing environment requires a sound understanding of their behaviour, aided by being able to provide unambiguous description of VOs and the capability to reason about core aspects. Hence, there is a need to have a formal modelling language for VOs (Ecolead, http://ecolead.vtt.fi), (Afsarmanesh, *et al.* (1997)), (Bocchi, *et al.* (2009)) VBE defines a base long-term cooperation agreement among a number of participants (individuals or institutions) and characterizes their interoperable infrastructure. As such, a VBE represents the organisational context in which the creation and operation of VOs takes place; VOs are seen as ensembles that are formed dynamically to provide high-level functionalities, or services, by sharing a number of resources in a distributed way. Some efforts have been made at describing formal models of VOs which aim at representing and evaluating different characteristics of VOs abstracting away the business domain. On the other hand there has been some work on VOs which focuses on the functionality offered by the domain. So far there has not been any effort towards developing a richer and more expressible language which could not only express structurally adaptable dimension of VOs, but their functional dimension in parallel as well. VOML framework attempts to fill this gap by developing a modelling language for VOs and VBEs which not only being formal paves ways for different kinds of analysis and evaluation, but also encompasses the structural and functional (business service) dimensions as well. This paper presents languages for formal specification of VOs at a level which not only covers domain concepts abstractly but also captures the functionality (service) offered by the VO; which other specification languages such as (Bryans, *et al.* (2006) fail to capture. The constructs available at the domain level pave the way which makes the virtual organization model easily adapt to the changing circumstances dynamically. Specifically, there are three different modelling languages each capturing a different aspect of VO. The first language named Virtual Organization Structural modelling language (VO-S for short) focuses on structural aspects and many of the characteristics peculiar to VOs such as relationship between two members, etc. The second language permits different reconfigurations on the structure of the VO. These reconfigurations change the core model itself. The third language named VO Operational modelling language (VO-O for short) describes operational models of VOs in more details, out of VO-S model.

2. MATERIAL AND METHODS

This work can be related to many interesting and active fields of research in computing. First of all it is about modelling of virtual organizations as echoed in (Bryans. (2006)). In (Bryans. (2006)) a VO is modelled as dynamic coalition using the Vienna Development Method (VDM) specification language (Fitzgerald, *et al.* (2008)). It defines a VO consisting of choices made in five orthogonal dimensions including membership, information representation, provenance, time and trust. Being a formal model different kinds of analysis and verification are possible. This work differs in the sense that it tries to develop a modelling language with the level of abstraction raised to a point where it is possible

++Correspondence ,author email: noor_rajper@hotmail.com +92 229213177

to directly support notions and concepts that are paramount in the domain of VO such as a VO consisting of different permanent and transient members and resources being utilized, the relationship between them and the workflow that is coordinating different activities.

(Norman *et al.*, (2004)) also attempt to form a resilient Virtual Organization based on Agent Technology. With respect to (Norman *et al.* (2004)) key difference is that the work reported in this paper aims at developing a modelling language rather then providing a platform and technology-specific solution. Another difference is that their reconfiguration is just limited to dynamically replacing one member with another having similar behaviour and capabilities whereas this work allows for dividing or sharing a task between more than one member, hence one member might be replace by more than one member each having different behaviour and capabilities but collectively still equivalent to the behavior of previous member (who was originally performing the task single handedly). This work can also be related to the field of dynamic adaptability in general (Wright *et al.*, (2006).

Chemical Process Development Case Study

To assess the practicality of VOML, a case study which is based on actual chemical process development (CDP), conducted as part of the GOLD project (Wright *et al.*, (2006) (Conlin, *et al.*, (2005), (Bryans, *et al.* (2006) (GOLD Project http://www.neresc.ac.uk/projects/gold/ project description.html) is adapted. The GOLD project is focused on providing a SOA based middleware for VOs (called dynamic coalitions), their access control policies and analysis of the various properties of VOs. Whereas, in the adapted case study only aspects that fall into structural and reconfiguration scope are extracted from the publications and technical reports available. Following case study describes the extracted details and how VOML framework models them in the following way:

Consider a VBE named *ChemicalVBE* which offers different services related to the field of the Chemical industry. One of the VOs currently operating in this VBE is called *ChemicalPD* which serves existing batch processing plants which are approaching the end of their serviceable life. *ChemicalPD* comes into play when the owner(s) of the plant decide to phase out/sell the existing plant and build a new chemical plant in place that produces the same chemical. In doing so *ChemicalPD* also helps in converting the plant from batch to more modern continuous operations if possible. The *ChemicalPD* VO accomplishes this goal in four phases. The basic project plant is divided into following four phases:

_ First phase consists of preliminary laboratory level investigation to determine whether conversion from batch to continuous operation is feasible or not.
_ Based on the results of phase 1, a design for the pilot plant is built during second phase.
_ In third phase a small-scale model pilot plant is build to identify suitable modes of operation, potential problems with start up shut-down, etc.
_ In the final phase, full scale production plant is built.

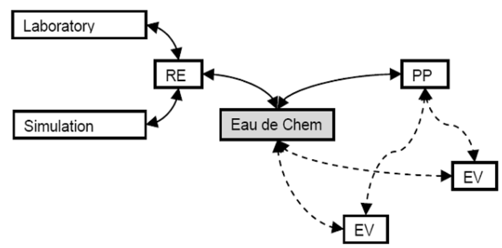


Fig. 1: The structure of the Chemical Development Project at the end of Phase 1

Task Structure of the *ChemicalPD*

The **(Fig. 1)** (courtesy by Wright *et al.*, 2006) shows the structure of the chemical development project (*ChemicalPD* VO in this case) at the end of phase 1. The box labeled RE represents the member performing the laboratory-level preliminary investigation (phase one). The *Laboratory* and *Simulation* labelled boxes show the services that the RE must provide to the VO. The *Eau de cham* box represents the owner and decision maker of the project at each stage. PP represents the member responsible for the design of pilot plant and the two EV boxes represent the equipment suppliers of the VO. Solid lines between boxes represent those members who remain relatively fixed during different phases of the VO whereas, dashed lines represent those members whose membership might change as the project evolve and unanticipated circumstances might demands changes the in the project plan. For example the EV might change from one providing filtration technology products to one providing alternative centrifuge technology.

The detailed and initial VO-S model of *ChemicalPD* VO is available in (Rajper, 2012). However, here a brief overview is provided to show how the ChemicalPD VO is modeled in VO-S. This helps in putting in perspective, different scenarios that are modeled in the papers below.

The *ChemicalPD* VO-S model specification consists of four main tasks; each corresponds to one of the phases mentioned above. In particular, the *PreliminaryAnalysis* task corresponds to phase one of the original case study (performed by RE).

The *Laboratory* and *Simulation* boxes of the figure in turn correspond to VO-S notion of capabilities that the member performing the *PreliminaryAnalysis* task must possess. The *PlantDesign* task corresponds to the phase two, *PilotPlantBuilding&Operation* and *FullScalePlantConstruction* corresponds to phase three and four respectively. The *Eau de Chem* corresponds to the customer of the *ChemicalPD* in this case. Similarly, PP and and two EVs correspond to the tasks *PlantDesign* and *EquipmentProvision* in *ChemicalPD*. The solid arrows in the Figure 1 represent the relationships which remain relatively fixed and the dashed arrow represent the more uncertain ones. The above mentioned tasks depend on a number of other (supporting) tasks, such as providing different off-the-shelf process equipment according to supplied specifications, supplying custom-built equipment not available from the off-the-shelf vendors and providing catalyst currently used by the reaction processes. These tasks are considered supporting tasks by the *ChemicalPD* as their role is limited to supplying the material/equipment. The *CatalystProvision* task provides the catalyst required to the *ChemicalPD* and through *EquipmentProvision* task both off-the-shelf and custom-built equipments are provided. In VO-S the tasks that appear under the umbrella of `Process` description are considered main tasks and those that do not are considered supporting tasks. VO-S does not explicitly tag the tasks with any construct.

However, a main task specify the supporting tasks, it is dependent on through its `supportedBy` attribute. By doing so, it has allowed to keep the vocabulary of the VO-S language as minimal as possible.

The significance of these different tasks is that it is now possible to differentiate between the core tasks carrying out some part of the goal (of VO) and those tasks that do not directly form part of the goal.

Membership Types of *ChemicalPD*

This section elaborates on the membership structure of the *ChemicalPD* VO and will try to justify the rationale behind this structure in particular and types of membership identified in VOML in general using the concepts developed at the beginning of the case study and as depicted in the Fig.1.

1. The members for the main tasks (*PreliminaryAnalysis*, Pilot*PlantDesign*, *BuildPilot Plant* and *BuildFullscalePlant*) have been chosen to be of type `Partner`. The competency requirements for these tasks are mainly the same for different type of chemical plants. The solid lines in the Fig. 1 also indicate to it. Besides, it allows VBE to create multiple (instances of) *ChemicalPD* VO quickly, as many of its aspects (in this case members for all the main tasks) are going to be same for each instantiation of *ChemicalPD*. 2. The catalyst used in the chemical reactions (for the production of the chemical) plays a vital role. The chemistry of the reaction and the separation method both get affected by the properties of the catalyst. However, the goal of *ChemicalPD* itself is not associated with catalyst itself; the catalyst is required by the tasks satisfying the VO goal. Hence the task of catalyst provision (*CatalystProvision*) is included in the ChemicalVO as a supporting task. Since every chemical plant has its own specific catalyst supplier with long term relations and contract and the catalyst also varies as per the chemistry used by the chemical plant, this implies that catalyst and its supplier varies from one chemical plant development to another (different *ChemicalPD* instances) so it can not be the partner of the *ChemicalPD*. However, the catalyst provider is also not eligible to be considered an external entity as external entity is discovered from the outer open universe (of VBE), but the customer of the *ChemicalPD* wants a catalyst supplier with whom the customer already has ties and once the plant is operational, the same catalyst supplier is going to supply the catalyst. These restrictions (by the VBE) and requirements (by the customer) make the member supplying catalyst eligible to be an `Associate` who comes from outside the VBE but not from the open universe, rather on customer's recommendation.

3. The equipment providers (both vendors and manufactures) lend very well to be external entities, as they keep changing from chemical plant to chemical plant and this equipment might not be required once the plant is operational. The above membership types clearly satisfies one of the characteristics highlighted by the original case study that is, the relationships between VO members are of different types, some are relatively fixed and some are more uncertain. The VOML framework provides concrete constructs to represent different types of relationships between the members. The GOLD architecture does highlight these differences but at the architecture level these types of relationship have been implemented with services which in the authors' opinion do not clearly differentiate between them.

Dynamic Nature of *ChemicalPD*

The chemical process development life-cycle is highly dynamic as unanticipated changes may occur at any time with consequence ranging from structure to competencies and membership. It is crucial that *ChemicalPD* remains agile and flexible to face both anticipated and unanticipated changes.

Goal Modification

In VOML framework the changes in capabilities represent the modifications to the goal.

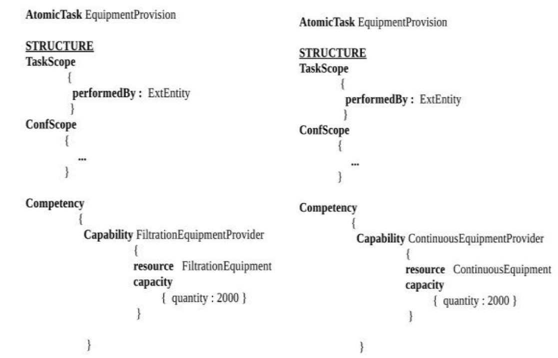
The goal modification does not radically alter the goal rather the path to achieve the (high level) goal has been modified in some way. The Scenario 1 given below describes how the goal a VO is altered in the VOML.

Scenario 1: Downstream processing problems: While upgrading from batch process to continuous process, the new operating conditions unexpectedly affect the downstream recovery of the catalyst. One such problem reported in the original case study occurred during phase one (*PreliminaryAnalysis*) when the chemistry required for continuous operation turned out to be suitable but, the filtration technology proved to be problematic for the downstream separation of the catalyst. A new separation method using centrifuge technology was therefore initiated with a different equipment supplier.

consequence: This involves (1) change in the capabilities of the equipment provider from one having capability to provide filtration technology equipment to one offering centrifuge technology and, (2) a modification to the VO structure through the replacement of a specialist equipment provider by a new one.

The downstream processing problem affects the *EquipmentProvision* task by changing its capability from *FiltrationEquipmentProvider* to *ContinuousEquipmentProvider*. This change is reflected through the *changeSeparationTech* policy. The *relinquishTask* and *changeVOmembership* policies un assigns the current member assigned to the task and then expels it from the *ChemicalPD*, if it is not required elsewhere. The policies are given below:

```
policy changeSeprationTech appliesTo
EquipmentProvision
if isCapabEquivalent
(ContinuousEquipmentProvider,
FiltrationEquipmentProvider )
do ReplaceCapability
(EquipmentProvision,
FiltrationEquipmentProvider,
ContinuousEquipmentProvider)
policy relinquishTask appliesTo
EquipmentProvision
when MemberTaskMismatch (EV1,
EquipmentProvision)
do UnAssignTask (EquipmentProvision,
EV1)
policychangeVOmembershipappliesTo EV1
when MemberWithoutAnyJob (EV1)
do RemoveMember(EV1)
```



(a) *EquipmentProvision* task before capability change (b) *EquipmentProvision* Task after capability change

Fig. 2: *EquipmentProvision* Task before and after capability change.

The policy *change Sepration Tech* replaces the *Filtration Equipment Provider* capability with its equivalent capability, the *Continuous Equipment Provider*, only if the two capabilities are considered equivalent (in the VBE Competency description). This constraint is verified through the `isCapabEquivalent` condition of the VO-R languages. Note that, the task of providing equipment still remains the same, just the kind of equipment has been altered in some way.

This situation however, can lead to a mismatch between the capabilities required by the task from its member and the member currently assigned to it, as the member still possess the the *FiltrationEquipmentProvider* capability. The policy *relinquishTask* takes care of this situation. This policy gets triggered when the event `MemberTaskMismath` gets fired and takes back the responsibility of the *EquipmentProvision* task from the current members (EV1). The policy *changeVOmembership* eventually expels the EV1 from the *ChemicalPD*.

Scenario 2: External Event: The chemistry and reaction process depends heavily on a particular catalyst. As this catalyst is a naturally occurring substance and its properties vary considerably with the geographical location where it is found. The supplier of the catalyst ceased to operate during the course of plant development. The quest for a new catalyst supplier was complicated by the fact that the catalyst supplied by the new supplier has to be compatible to the original one, that is it must have similar properties.

consequence: Many catalyst sources temporarily joined the *ChemicalPD* VO and phase one (*PreliminaryAnalysis*) chemistry (reaction processes) was restarted to ensure that the catalyst was an adequate replacement. This process continued until an appropriate supplier was found that worked well with the selected operations method (continuous method). One instance mentioned in the case study is that a new catalyst supplier's catalyst worked well with the old batch plant but not in continuous operation laboratory trials. Hence, the supplier had to leave the *ChemicalPD* and a new supplier was looked for. Considering this example, the changes that the VO-S model has to undergo to account for the above situation are as follow:

In the VO-S specification the catalyst supplier is an associate because it is recommended by the customer of the VO (*Eau de Chem* in this case), but when new catalyst sources are tried, the customer might not have any ties with them. Hence, in this case the customer might not be able to suggest new supplier sources. This leads to looking for adequate suppliers from the open universe, making them temporary members of the VO, investigating if the catalyst supplied by the new supplier is adequate choice and based on the result, expelling the temporary member or continue working with it. This situation of temporary membership is dealt naturally by the VOML's `ExtEntity` type of membership i.e. they are invited into *ChemicalVO* as external entities rather than associate and once an appropriate supplier is found it's membership is turned from external entities to associate again and continues its support in the next stages of the *ChemicalVO*. In short, provisions need to be made for the following kind of changes:

1. Change of membership type from associate to external entity: This change is covered by the *membersRoleUpdate* policy. This policy gets activated as the result of triggering of the event `MemberLeft`.
2. Invite external entity: This change is achieved through the *findNewCatalystSource* policy, which first searches for the new catalyst source, then adds it to *ChemicalVO* and finally assigns the new member to the *CatalystProvison* task through `SearchMember`, `AddNewMember` and `AssignTask` actions respectively. The `memberFound` is a special keyword referencing the name of the member identity who had been returned by the `searchMember` action.
3. Expel external entity (if not suitable) or, change type from `ExtEntity` to `Associate` (if suitable).

```
policy membersRoleUpdate appliesTo
CatalystProvision when MemberLeft(CP)
do changeRole (CatalystProvision,
ExtEntity)
policy findNewCatalystSource
appliesTo CatalystProvision do
SearchMember (CatalystProvision)
andThen AddMember(memberFound)
andThen AssignTask(CatalystProvision
, memberFound)
policy newCatalystSourceNotSuitable
appliesTo CatalystProvision
do UnAssignTask(CatalystProvision,
tempMem-id) andThen RemoveMember(temp
Mem-id)
policy newCatalystSourceSuitable
appliesTo CatalystProvision
do ChangeRole (CatalystProvision,
Associate)
```

Scenario: The Need for Additional outsourcing of Specialist Services: As the chemical plant development progressed, unanticipated changes kept occurring from time to time, most of which required the previous tasks to be performed again, which in turn lead to the delay in the completion of project. The potential cost for missed opportunity was therefore very great, especially at the end of the project when the full scale plant needed to be built. The solution found for this problem was to increase the labor and expedite the plant construction. However, the contractor did not have the required amount of labor. Hence the need for additional contractors and one more contractor was added to the VO. Collectively the existing and additional contractor could finish the project in the specified time. This kind of situation can also be seen in the Figure 1 where there are two equipment providers (labelled as EV).

Consequence: The provision for sharing a task between more than one member when the members lack the capacity is to replicate the task specification and this is achieved through changing the type of the task to `ReplicalbeTask`. Given below are the policies which account for both of the mentioned changes and the VO-S excerpt for the task *FullScalePlantConstruction* which can now be replicated between more than one member. The *expediatePlantConstruction* policy increases the total labor to 1000 which triggers the event `CapacityDeficit` which in turn is handled by *addMoreLabour* policy by making the task replicable.

```
policy expediatePlantConstuction
appliesTo FullScalePlantConstruction
do ChangeCapacity(FullScalePlant-
Constrution,
FullPlantBuildingCapability.labor,
1000)
policy addMoreLabour appliesTo
FullScalePlantConstruction
when CapacityDeficit(FullScalePlant-
Construction)
```

```
do
MakeTaskReplicable(FullScalePlantCons
truction, 2, cooperation )
```

Scenario: Division of Responsibility: This scenarios covers the case widely quoted in VO literature as one of the reasons and benefits of VO approach. The issue is when different organizations having specific capabilities want to expand their business by forming synergies with other organizations by creating Virtual organizations (or even VOs within VOs). In VOML framework this problem is approached by allowing a task to be subdivided into smaller tasks and then each subtask can be assigned to different members. This situation occurs in the *ChemicalPD* when the member performing the preliminary analysis leaves the VO and no members can be found who satisfy all the capabilities of the *PreliminaryAnalysis* task.

This situation of not being successful in finding a single member with all capabilities is triggered by the event NoMemberWithAllCapabilitiesFound used in the policy *createSubTasks* below. The decision is made to divide the task into two sub tasks, one task performing the laboratory experiments with the capability *LaboratoryService* and other one with the capability *SimulationService*. These are respectively called *LabExpTask* and *SimExpTask* and are represented in the action MakeTaskComposable. Since the criteria parameter is omitted from the action which specifies the relationship between the subtasks, the default relation between the two subtask is of type cooperation. The cooperation relation between subtasks imply that any sub task can be carried out first or both subtasks can be performed in parallel in terms of control flow at the operational model (VO-O) level.

```
policy createSubTasks appliesTo
PreliminaryAnalysis when
NoMemberWithRequiredCapabilityFound
do MakeTaskComposable
(PreliminaryAnalysis, [LabExpTask,
LaboratoryService], [SimExpTask,
SimulationService]])
```

4. CONCLUSION

In this paper an attempt has been made to evaluate the VOML framework and its corresponding modelling languages for a more realistic case study of chemical plant development.

The original case study worked as requirements that the service based middleware must exhibit to cope with dynamic and complex nature of VOs. The different complexities and flexibilities demanded of VOs mentioned explicitly or implicitly were captured and then model it using VOML sub language. This shows that VOML framework is rich enough to model many of complexities VOs in more realistic scenarios.

REFERENCES:

Bocchi, L., J. L. Fiadeiro, N. Rajper, and S. Reiff-Marganiec. (2009) Structure and behaviour of virtual organisation breeding environments. In FAVO, 26-40.

Bryans, J. W., J. S. Fitzgerald, C. B. Jones, and I. Mozolevsky. (2006) Formal modeling of dynamic coalitions, with an application in chemical engineering. In Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation, pages 91–98, Paphos, Cyprus, IEEE.

Conlin, A., N. Cook, H. Hiden, P. Periorellis, and R. Smith. (2005) Gold architecture document. Technical Report CS-TR-923, School of Computing Science, University of Newcastle upon Tyne.

Norman J. T., A. Preece, S. Chalmers. N. R. Jennings, M. Luck, V. D. Dang, T. D. Nguyen, V. Deora, J. Shao, W. A. Gray, N. J. Fiddian (2004) CONOISE: Agent-Based Formation of Virtual Organisations. In Research and Development in Intelligent Systems XX, Coenen F., A. Preece, A. Macintosh, editor, Springer London, 353-366.

Ecolead, http://ecolead.vtt.fi, visited on 1/1/14.

GOLD Project http://www.neresc.ac.uk/ projects/ gold/projectdescription.html visited on: 8/9/2011.

Afsarmanesh. H. and L. M. Camarinha-Matos. (1997) Federated information management for cooperative virtual organizations. In Abdelkader Hameurlain and A Tjoa, editors, Database and Expert Systems Applications, volume 1308 of Lecture Notes in Computer Science, pages 561–572. Springer Berlin/ Heidelberg.

Rajper. N. J. (2012) VOM: Virtual Organization Modelling Languages. PhD Thesis, http://hdl.handle.net/2381/10942, UK.

Wright A., A. Conlin, H. Hiden. (2006) A chemical process development case study as a source of requirements for the GOLD project. Technical Report No. CS-TR-968, School of Computing Science, University of Newcastle upon Tyne, June.

Fitzgerald J. S, P. G. Larsen, and M. Verhoef (2008) Vienna development method. In Benjamin W. Wah, editor, Wiley Encyclopedia of Computer Science and Engineering. John Wiley & Sons, Inc.