Standard Operating Procedures: Collaborative Development and Distributed Use

Gerhard Wickler
AIAI, University of Edinburgh, Scotland
g.wickler@ed.ac.uk

Stephen Potter
AIAI, University of Edinburgh, Scotland
s.potter@ed.ac.uk

ABSTRACT

This paper describes a system that supports the distributed development and deployment of Standard Operating Procedures. The system is based on popular, open-source wiki software for the SOP development, and the I-X task-centric agent framework for deployment. A preliminary evaluation using an SOP for virtual collaboration is described and shows the potential of the approach.

Keywords

Artificial Intelligence planning, Standard Operating Procedures, wiki extensions

INTRODUCTION

Emergency situations usually call for quick and appropriate action to minimize loss of life and property. However, knowing what these actions should be is often not obvious. One way to prepare for emergencies of a given type is to develop so-called Standard Operating Procedures (SOPs), manuals describing courses of action that should be followed in a given situation. These SOP manuals represent best-practice knowledge, and are usually written by one or more experts with extensive experience in the field. Such SOP manuals can be used to train emergency managers.

There are many SOP manuals available today, most existing as physical documents and ranging in size from a few pages to several volumes. While these manuals are considered valuable where they exist, there are a number of problems with such documents in practice:

- Access time: While these manuals are useful for teaching the procedures they contain, they are usually not used during an actual emergency. This is simply because there is no time to search for information in large manuals. Emergency managers may have been through the SOPs, but under stressful conditions options may be forgotten or steps may be omitted.
- Structure: SOP manuals are structured, but there is no standard way of structuring these documents. An emergency manager who needs to be familiar with different SOPs deriving from different sources may thus find them confusing to use.
- Updating: SOPs should be updated with lessons learned after every emergency in which they have been applied. This is a cumbersome task to perform with printed documents, and even web-based documents offer limited support for this process.

The OpenVCE project (Hansberger et al., 2010) aims to develop an open virtual collaboration environment that facilitates collaborative work in a virtual space. This environment could, for example, be used to collaborate on the development of SOPs, or it could be used during an actual emergency to manage information and courses of action. In fact, this environment contains a specific piece of software that supports these two functions: the MediaWiki SOP extension described in this paper.

The OpenVCE space consists of two linked environments: a dynamic website and 3D space for meetings (Tate et al., 2009) (see figure 1). This space links aspects of a web-based community portal with virtual world 3D spaces using:

- Apache, MySQL and PHP

Reviewing Statement: This paper represents work in progress, an issue for discussion, a case study, best practice or other matters of interest and has been reviewed for clarity, relevance and significance.

- Drupal and MediaWiki with appropriate extensions:
  - Organic Groups
  - Views module
  - Secondlife Framework, SLUser and SLRegAPI modules
  - Twitter module
  - Calendar, events, messaging and notification modules
  - Images and Videos modules
  - WYSIWYG and TinyMCE Editor

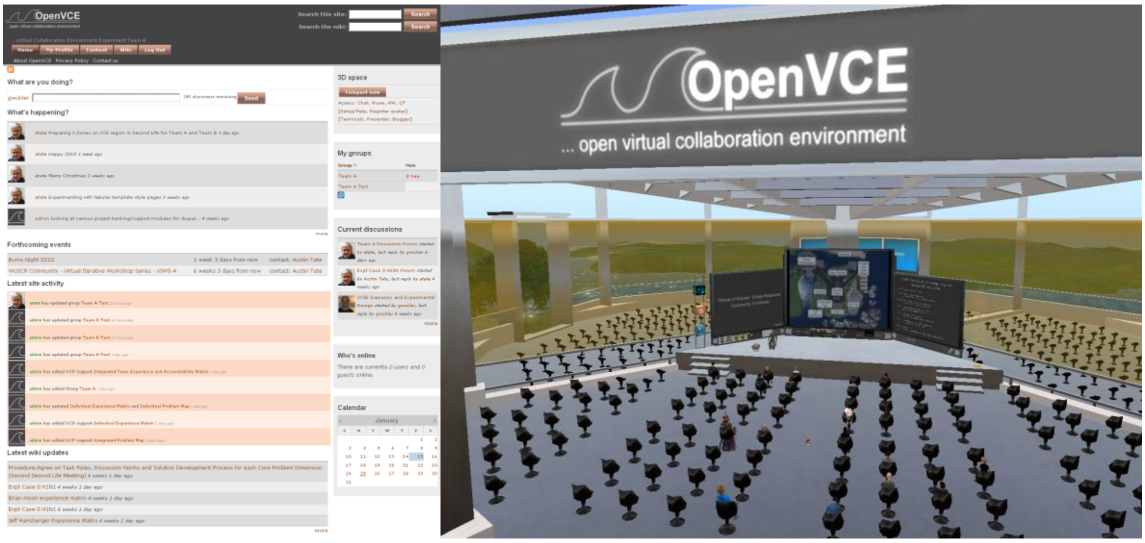
OpenVCE
... open virtual collaboration environment
Search this site:
Search
Search the wiki:
Search
...Virtual Collaboration Environment Experiment Team A
Home
My Profile
Content
Wiki
Log Out
About OpenVCE
Privacy Policy
Contact us
What are you doing?
gwickler
Send
What's happening?
atate Happy 2010 1 week ago
atate Merry Christmas 3 weeks ago
atate Experimenting with tabular template style pages 3 weeks ago
admin looking at various project tracking/support modules for drupal... 4 weeks ago
more
Forthcoming events
Burns Night 2010
1 week 3 days from now
contact: Austin Tate
WoSCR Community - Virtual Iterative Workshop Series - VIWS-4
6 weeks 3 days from now
contact: Austin Tate
Latest site activity
Latest wiki updates
Expt Case 0 H1N1 4 weeks 1 day ago
Brian moon experience matrix 4 weeks 1 day ago
Jeff Hansberger Experience Matrix 4 weeks 1 day ago
more
3D space
Teleport now
My groups
Team A
Team A Test
Current discussions
more
Who's online
There are currently 0 users and 0 guests online.
Calendar
January
OpenVCE
... open virtual collaboration environment

Figure 1. OpenVCE Website and 3D Virtual Meeting Space

To avoid all this technology overwhelming users who are attempting to collaborate in this space, the project has also developed the Virtual Collaboration Protocol, which is itself an SOP that describes how this environment is meant to be used to deal with certain types of emergency. This protocol is itself supported by an extension to the website that guides users who are following the protocol.

COLLABORATIVE SOP DEVELOPMENT USING A WIKI

The collaborative development of SOPs can be supported in a number of ways using dynamic web technology. We have based our collaborative document editing facility that can be used to write SOP manuals on MediaWiki. The reasons for this choice are simple: MediaWiki (Barrett, 2009) is open-source (a project requirement), scalable (it powers Wikipedia), and there is an active community behind it. However, wiki articles are not structured to support SOPs, which is why we have implemented an extension that allows for the structuring of an SOP article according to the principles underlying Hierarchical Task Network (HTN) planning (Ghallab et al., 2004; Tate, 1977), which provides a 'natural' way of decomposing tasks into sub-tasks, and as such is the structure found in many existing SOP manuals.

Hierarchical Task Networks

What are known as Standard Operating Procedures to domain experts are called methods in HTN planning (Ghallab et al., 2004, chapter 11). Methods formally describe how a task can be broken down into sub-tasks. The definition of a method consists of four main parts:

- Name: the name of this method (there may be several methods for performing the same task);
- Objective: an expression describing the task that can be accomplished with this method;
- Constraints: a set of constraints (e.g. on the current state of the world) that must hold for this method to be applicable; and
- Network: a description of the sub-tasks into which this method 'refines' or decomposes the given task.

The name of the method can be used to refer to the method and it usually indicates the way in which the task is to be accomplished. For example, we may define a method "set up camp for ?x people" where ?x is a variable that will get assigned an appropriate value when the method is applied. The objective of a method is used for matching methods to tasks that must be accomplished, e.g. in a to-do list. For example, a task to be accomplished might be to "provide shelter for 100 people" and the objective of the above method could be "provide shelter for ?x people". Clearly, the objective matches the task.

For a method to be applicable to a given task, such a match is necessary. Furthermore for automated planning support, it must be possible to find this match automatically. Further conditions for applicability are represented by the constraints. For example, a constraint associated with the above method might be that "?x must be between 50 and 1000". In this example the constraint would be satisfied and, if this was the only constraint, the method would be applicable. HTN planners often deal with such constraints though constraint managers that can handle many different types of constraints such as temporal and resource constraints associated with a method. However, this requires that all constraints and information about current conditions are expressed in computer-readable formats.

The network is a set of sub-tasks that, when all accomplished, implement the method and achieve the objective of the method. For example, sub-tasks for the above method might be "set up tents for ?x people" "provide food for ?x people" and "provide medical aid for ?x people".

Artificial Intelligence has developed a set of algorithms for building, exploring, managing, and executing a set of HTN plans. The I-X framework is one such toolkit that includes an HTN planner.

The MediaWiki SOP Extension

The problem with HTN planning 'domains', the formal computer- processable expressions of SOPs, is that these are rather difficult to write. Domain experts are usually not capable of producing these formal descriptions. Experts in AI planning on the other hand know the formalism, but do not have the knowledge that needs to be encoded. The approach taken with the SOP extension for MediaWiki was to keep the representation very simple, at least initially, to encourage domain experts to encode their knowledge directly.

The starting point for using this extension is an SOP manual in conventional form, i.e. some kind of document. To get this into the wiki it has to be divided into more manageable 'articles', each corresponding to a single wiki page which in turn describes a single method or refinement, that is, a way of breaking down a given task into smaller subtasks. This division into pages/methods is – as in any wiki, and any conventional encyclopedia for that matter – difficult to prescribe; in some sense we hope that a suitable level of description emerges as the domain experts divide their knowledge into circumscribed sub-tasks.

The next step is to provide some structure; this is done by marking up the individual wiki articles describing a single method with formal tags provided by the extension. For now, only a small number of tags exist to allow for a basic structuring of a set of methods. (These tags are implemented as MediaWiki parser functions.) The first one allows for the explicit specification of an objective:

{{#objective:…}}

This must be used if there are multiple methods that accomplish the same objective. If there is only one method (in the library) then the objective is taken to be the same as the name of the method, which is the title of the wiki article. There can only be one objective per method. The other tag that is provided by the extension allows the explicit specification of subtasks that need to be accomplished by the method:

{{#subtask:…}}

There can be any number of subtask tags added to an SOP article. The order of the tags in the article is taken to be the order in which the subtasks are to be accomplished, i.e. the subtasks are totally ordered: a subtask must be completed before the next is begun.

Only authors of SOP articles will ever see these tags as shown above. When viewing an SOP article, they are transformed by the MediaWiki parser and appear as headings that are hyperlinked. The page they are linked to is a dynamic page that will search the underlying database for all the methods in the wiki that accomplish the given subtask. This enables users to get a quick overview of all the known methods for accomplishing a given subtask. The methods in this list are of course hyperlinked to their SOP articles.

Finally, there is another special page that allows the exporting of all known SOP articles in an XML-based format, in order that tools like I-X can import the representation.

TASK SUPPORT THROUGH I-X

I-X is a framework for writing applications that support distributed task-centered work. Its principal interface is the process panel described in (Tate et al., 2002; Wickler et al., 2006). It is envisaged that SOPs written using the SOP extension described above can be imported into this tool, which effectively acts as an intelligent, distributed to-do list. However, at the time of writing this functionality is still under development.

CASE STUDY: THE VIRTUAL COLLABORATION PROTOCOL

One of the outputs of the OpenVCE project is the Virtual Collaboration Protocol (VCP) by Robert Cross (Cross and Thomas, 2009). The VCP is a reasonably generic procedure for collaborative problem-solving, tailored to the resources available in OpenVCE, namely the collaboration website and 3D meeting space. Since the VCP can be seen as an SOP for collaborating using OpenVCE technology, we have chosen to use it as a test case for the SOP extension.

The Virtual Collaboration Protocol

The VCP provides guidance for collaborative problem solving. It consists of seven phases that correspond to the following main tasks:

- before the first meeting: individuals define problem dimensions
- first team meeting: team agrees consolidated problem dimensions
- before second team meeting: individuals describe experience with respect to problem dimensions
- second team meeting: discuss experience and assign subteams to address each dimension
- before third meeting: develop solutions for each dimension
- third meeting: present and discuss solutions for each dimension
- after third meeting: integrate solutions into coherent solution document

A key concept is the *problem dimension*, which describes an aspect of the problem that needs to be addressed by the team. Each of the phases is described by 1-2 pages of text explaining what needs to be accomplished by the team in that phase (the later phases tend to be more problem-specific and hence less elaborately described). It also contains a number of forms that provide templates for the outcomes of each phase.

Encoding the VCP in the SOP Extension

The first step towards encoding the VCP using the SOP extension was to import the text. This was done by creating seven articles on the wiki for each of the phases described in the original VCP document (which was written using MS Word). In addition, another ‘overview’ article was created from the table of contents. Each of the section titles was marked up as a subtask to enable the extension to find relevant SOPs, namely those corresponding to the respective sections.

The next step was to go through the individual sections and identify subtasks therein that can be marked up as such. In most pages two to three subtasks could easily be identified and were marked up as such. Since for performing a number of these subtasks there was no further advice provided by the document (= no suggested methods), some additional procedures were written that explained how the website and 3D space could be used to accomplish these tasks.

This resulted in a clear structure where the top-level task, to collaborate to solve a problem, was broken down into its seven VCP phases, and each of these was broken down into finer steps that were more closely related to the technology of OpenVCE.

Preliminary Experiences

The first thing to note is that the procedure described in the original VCP document is completely linear, i.e. there are no phases that may be concurrent. This is quite natural since users tend to think of Word documents as linear texts. As a result, the lack of explicit ordering constraints in the SOP extension was not a problem here. We would not expect this to hold in general.

A related point concerns the explicit objectives: there was usually only one way of accomplishing a given objective in the VCP (with one exception). This means the objective tag was (almost) never used during the marking up of the protocol. Again, we would expect this to change once the protocol becomes more complex.

During the marking up it became clear that not all the text of the original VCP was describing procedural knowledge. For example, the first phase is largely a description of what knowledge and experience the process coordinator is supposed to have. While this is relevant to the VCP, it is not clear how to mark up such information using the SOP extension tags. In other words, it is not clear how best to incorporate such supplementary information into procedural information.

Similarly, the original VCP often described what is to be accomplished and how it is to be accomplished in the same paragraph, which makes it difficult to separate out these two aspects as objective and subtask.

Another issue was the fact that the tags are not sufficiently expressive to represent the agent that must accomplish subtasks. For example, the individual problem map is to be completed by every team member; whereas the integrated map is to be consolidated by process coordinator.

Finally, the browsing capabilities provided by the special wiki page are good for editing, but not very good for using SOPs. This really needs export into different tool, which is not yet implemented.

CONCLUSIONS AND FUTURE WORK

This paper describes some preliminary experiences of trying to use an extension to MediaWiki to encode SOPs. The SOP used is the VCP described in the paper.

The result so far is promising in the sense that the envisaged route – writing the SOP using conventional software and then marking it up with special tags – worked reasonably well. The advantage of such an approach is the instant availability of the full SOP, even before any mark-up has been added. However, the mark-up should add value not only to the SOP author, but also to the user, and the latter can only be evaluated once the import into a task-supporting tool has been completed.

ACKNOWLEDGEMENTS

Effort sponsored by USJFCOM-Army Research Labs parent contract DAAD19-01-C-0065, subcontract no. SFP1196749DP (via Alion Science and Technology), task order no. 118, and the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA8655-09-1-3090. The University of Edinburgh and research sponsors are authorized to reproduce and distribute reprints and on-line copies for their purposes notwithstanding any copyright annotation hereon.

REFERENCES

1. Barrett, D.J. MediaWiki (2009) O'Reilley.
2. Cross, R. and Thomas, R. (2009) Driving Results through Social Networks, Jossey-Bass.
3. Ghallab, M., Nau, D. and Traverso P. (2004) Automated Planning: Theory and Practice. Morgan Kaufman.
4. Hansberger, J., Tate, A., Moon, B. and Cross, R., Cognitively Engineering a Virtual Collaboration Environment for Crisis Response, Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Working, Savannah, Georgia, USA, 6-10 February 2010.
5. Tate, A. (1977) Generating Project Networks. . In Proc. of the International Joint Conference on Artificial Intelligence (IJCAI), pp 888-893.
6. Tate, A., Dalton, J. and Stader, J. (2002) I-P2- Intelligent Process Panels to Support Coalition Operations. In Proc. of the Second International Conference on Knowledge Systems for Coalition Operations (KSCO-2002). Toulouse, France, April 2002.
7. Tate, A., Potter, S. and Dalton, J. (2009), I-Room: a Virtual Space for Emergency Response for the Multinational Planning Augmentation Team, Proc. of the Fifth International Conference on Knowledge Systems for Coalition Operations (KSCO-2009) (Lawton, J., Patel, J. and Tate. A. eds.), Chilworth Manor, Southampton, UK, 31 March-1 April 2009.
8. Wickler, G., Potter, S., Tate, A. (2006) Using I-X Process Panels as Intelligent To-Do Lists for Agent Coordination in Emergency Response, International Journal of Intelligent Control and Systems (IJICS), Special Issue on Emergency Management Systems, Vol. 11, No. 4, Dec. 2006.