Semantic Grid Services for Video Analysis

Gayathri Nadarajan[1] Yun-Heh Chen-Burger[2] James Malone[3]
Artificial Intelligence Applications Institute (AIAI)
Center for Intelligent Systems and their Applications (CISA)
School of Informatics, University of Edinburgh, UK
[1]G.Nadarajan@sms.ed.ac.uk, [2]jessicac@inf.ed.ac.uk, [3]j.malone@ed.ac.uk

Abstract

Employing the power of Semantic Grid services into pervasive problem domains such as video analysis would allow for more effective distributed processing. A vast amount of ecological data from the EcoGrid of varying qualities and features will need to be analysed efficiently. As manual processing by humans can be time and labour intensive, video and image processing tools can go some way to addressing such problems since they are computationally fast. However, most video analyses that utilise a combination of these tools are still done manually. We propose a semantic-based hybrid workflow composition method that strives to provide automation to speed up this process. The main components of this framework are presented, along with the illustration of a scenario where these components act as Semantic Grid services for EcoGrid video analysis.

1. Introduction

1.1. Semantic Grid Services

The Grid [4] is an infrastructure for next-generation e-Science applications aimed at enabling resource sharing and coordinated problem-solving between computers and humans in a distributed and heterogeneous manner. It is complex because it tries to couple distributed and heterogeneous resources such as data, computers, operating systems, database systems, applications and special devices, which may run across multiple virtual organisations. The Semantic Grid [11] is an extension of the current Grid in which information and services are given well-defined and explicitly represented meaning, better enabling this cooperation. Such semantic capabilities are provided by enriched representations such as ontologies and Semantic Web-based languages [8, 6]. Ontologies help with the sharing and reuse of common vocabularies between domains, as well as help to perform reasoning and inferencing about them while Semantic Web-based languages provide a means for representing machine-processable information. This would allow for a high degree of easy-to-use and seamless automation to facilitate flexible collaborations and computations on a global scale as imposed by the Semantic Grid community. Thus the aim of the Semantic Grid is to explore the use of Semantic Web technologies to enrich the Grid with semantics. The provision of Semantic Grid services would prove useful for pervasive problem domains, such as distributed video processing, as outlined in the following section.

1.2. The EcoGrid

The EcoGrid project [1] is aimed at utilising state-of-the-art Grid technologies to establish a cyberinfrastructure for ecological research. This includes the integration of geographically distributed sensors, computing power and storage resources into a uniform and secure platform. Scientists can conduct data acquisition, data analysis and data sharing on this platform. The National Center for High-performance Computing (NCHC), Taiwan and Artificial Intelligence Applications Institute (AIAI), the University of Edinburgh have forged a research collaboration on workflow enactment in the EcoGrid.

Recognising the needs for real-time automatic and continuous information gathering through EcoGrid that offers a unique and immense opportunity for long term ecological monitoring and planning, NCHC has installed and managed Wireless Sensor Nets in several national parks in Taiwan. The information collected is stored in and made available through EcoGrid for access. The information collected includes surveillance videos in the Fu-Shan National Park covering the entire area for observing natural lives and protecting them from potential poachers, audio recording of frogs of rare species, under-sea coral reef and marine life observation stations and more. Due to the continuous and non-intrusive methods deployed, such monitoring and recording efforts have already made ecological discoveries of significant importance that traditional methods otherwise could not have made.

Continuous data collection in the EcoGrid, however, poses a great challenge as how this data may be transformed into useable information for the ecologists and in a timely fashion. For instance, one minute of video clip typically takes 1829 frames and is stored in 3.72 Mbytes. That translates into 223.2 MB per minute, 5356.8MB per day and 1.86 Terabytes per year for one operational camera, and due to the unpredictability of nature, one may not easily skip frames as they may contain vital information. Based on our experience, one minute's clip will on average cost manual processing time of 15 minutes. This means that one year's recording of a camera would cost human experts 15 years' effort just to perform basic analyses. Currently there are three under water cameras in operation and this will cost a human expert 45 years for performing basic processing tasks. This is clearly an impractical situation and more appropriate automation methods must be deployed.

2. Requirements

Given the ecological problem above and the motivation for providing Semantic Grid services, we outline a set of requirements that will correspond to the functionalities of a suitable system in our framework.

Process Automation. A mechanism for automatically processing large amounts of data is required. As we are performing video analysis on real-time data, this task can be broken down into a set of processes applied in sequence to the raw data. For example, the task of video annotation could be decomposed into these high-level steps; keyframe detection, classification by characteristics and annotation. We require this sequence to be run automatically.

Performance-Based Selection. For each of these processes, one or more software tools can be used to achieve the particular function. Depending on the quality of the data, the combination of tools with the *best performing capability* for the overall stated goal should be selected. Thus the mechanism should be able to match the goal of the process with the capabilities (tools) based on a performance measure.

Iterative Processing. Each process in turn, could also be further decomposed into sub-processes. For instance, the process 'classification by characteristics' could be made up of a feature extraction sub-process followed by a recognition sub-process. The recognition sub-process may also backtrack to the feature extraction sub-process. Thus, a mechanism to support iterative processing is required.

Adaptive, Flexible and Generic Architecture. Following from its definition, the Grid is a dynamic environment where availability of resources and their load can change notably from one time point to the next. An ideal system would be one that is extremely sensitive and adaptive to changing environments. It would also possess a generic capability, whereby it should be able to perform a variety of video analysis tasks provided as input. Furthermore, it should be able to generate new sequences of solutions. For this it should include techniques that allow it to incorporate its knowledge of previous experiences in order to generate new solutions as part of its learning. Thus an adaptive, generic and flexible approach is required.

Semantic-Based Compatibility. As well as traditional Grid and video processing requirements, the approach that we take should also be semantic-based, that is to integrate ontologies and Semantic Web-based languages, such as Web Ontology Language (OWL) [8] and more recently, Temporal Resource Description Framework (Temporal RDF) [6] to allow for effective reasoning in compliance with Semantic Grid requirements.

3. Limitations of Current Solutions

One key requirement of the Semantic Grid is facilitating the composition of multiple resources, and mechanisms for creating and enacting these in a distributed manner. This requires means for composing and executing complex workflows, which has attracted considerable effort. Here we provide a brief overview and analysis of several influential Grid workflow composition systems. A comprehensive study on Grid workflow systems can be found in [13].

Pegasus [3] utilises Planning to map abstract workflows to their concrete forms that are executable in the Grid for particle physics experiments. The role of planning is important as it allows for dynamic process selection and composition based on a given set of goals. However, Pegasus does not support looping. Triana [12] is an open-source, graphical environment that allows users to construct and make changes to a workflow by adding, deleting or changing the sequence of execution by drag-and-drop. Additionally, Triana supports looping constructs, which is desirable from a process modelling point of view.

Taverna [10] is used by biologists and bioinformaticians to execute scientific workflows in the Grid. It provides graphical interfaces that allows workflow manipulation and progress invocation easily. It also provides an implicit iteration mechanism and possesses fault tolerance capability. Taverna is particularly suitable for tasks that can handle simultaneous processing as it supports concurrency. Kepler [7] models a workflow as a composition of independent components that communicate through well-defined interfaces. Different execution models can be easily plugged into the workflows without changing any other components within the workflows. Kepler supports looping and also has good reliability as it is able to produce partial results even when an entire workflow fails.

A brief comparison of the four systems indicates that none of them could single-handedly fulfill all the requirements sought. In terms of applicability, Pegasus would best suit a domain with well-defined requirements and where the overall goal could be determined from a given set of rules and constraints. Triana is well-suited for composing complex workflows for Web services and Peer to Peer services. Taverna is also suitable to be used in Web and Grid services contexts, but its use may be limited to composing simple workflows, whereas Kepler works very well for composing workflows for complex tasks but it may not be fully Grid-enhanced since the system it is built upon is primarily aimed at modelling concurrent systems.

Most existing systems also make use of separate tools to define and manipulate ontologies instead of fully integrating them into the system. We wish to integrate the capability to read OWL files and provide results and information in OWL and Temporal RDF within our system. For performing the complex task of video processing, Kepler stands out as a promising system to be considered as an underlying workflow composition mechanism. We wish to take advantage of the most prominent and advantageous features exhibited by existing systems, as well as incorporate new features that will fill the remaining gaps.

4. Proposed Framework - Hybrid Method

Based on the analysis of existing systems provided in the previous section, we propose a semantic-based hybrid workflow composition method within a three-layered framework that distinguishes different levels of abstraction through the design, workflow and processing layers. The architecture diagram for this framework is given in [9].

Design Layer. The design layer contains components that describe the domain, goals, capabilities and processes to be carried out in the system. These are represented using ontologies (section 4.1) and two libraries. A modeller is able to manipulate the components of this layer, for example populate the libraries and modify the ontologies.

Workflow Layer. This layer acts as the main interface between the design and processing layers. It ensures the smooth interaction between the components, access to and from various resources such as raw data, video and image processing toolset, as well as the provision of the final output to the user. A workflow enactor (section 4.1) acts as the interpreter of the events that occur within the system.

Processing Layer. The processing layer consists of a set of video and image processing tools (section 4.1) that will act on the data. The functions of these tools are represented in the capability ontology in the design layer. Once a workflow has been established, these tools may work on the videos directly. The final result is passed back to the workflow layer for output and evaluation.

4.1. Architectural Components

Several essential components to the framework are described here. They consist of ontologies and process manager in the design layer, workflow enactor in the workflow layer and image processing toolset in the processing layer.

Ontologies. We have opted to incorporate three ontologies to keep the goals separate from the capabilities and to provide meaning for the process within a semantically integrated system. Each ontology holds a vocabulary of classes of things that it represents; the goal ontology contains the classes tasks (e.g. "annotation") while the capability ontology contains the classes of video and image processing tools. The domain ontology provides meaning for annotation, for example concepts such as "blur", "clear", "bright" are included. The use of ontologies is beneficial because they provide a formal and explicit means to represent concepts, relationships and properties in a domain. They play an important role in fulfilling semantic interoperability, as highlighted in section 3. A workflow system with full ontological integration has several advantages. It allows for cross-checking between ontologies, addition of new concepts into the workflow system and discovery of new knowledge within the system.

Process Manager. The process manager is responsible for selecting and composing the sequence of processes to be executed in the workflow based on the tools available to perform the task. It has a Planning component, which comprises the goal ontology and the process library, and a Case-Based Reasoning (CBR) component, which is the case library. The process library holds instances of processes that could be executed to perform the tasks of the workflow. Once a set of tools have been identified for achieving the goal, the process library composes the process sequence for execution. The goal ontology and process library, together, constitute the Planning component of the process manager.

However, there could be more than one possible solution or no solution for satisfying a particular task. While the Planning component is responsible for matching the goal with the capabilities (tools) and selecting the procedural steps to be taken, the CBR component is responsible for finding the closest solution from past scenarios for cases where Planning alone doesn't work. The case library keeps track of previous viable solutions and finds a *similar* solution to match the current problem. The heuristic for the closest solution is based on a similarity measure [5].

Workflow Enactor. The workflow enactor plays the important role of choreographing the flow of processing within the system. For the task of annotation, the flow of events could be summarised as follows. The user inputs a request into the system. The workflow enactor fetches data in the form of a video clip from the data storage and delegates the user request to the design layer. Based on the Planning and CBR approaches, a workflow for annotation will be established and passed to the workflow enactor. This is then fed to the processing layer for further action. The processing layer will communicate the annotated video back to the workflow enactor, which will pass this result to a domain expert who will provide feedback. This could be used as a basis for improving the performance of the system. Finally, the case library is updated with this solution.

Image Processing Toolset. A set of image processing tools for key frame detection, segmentation, recognition and annotation would be available. These tools are represented semantically in the capability ontology. It should be noted that there could be more than one tool available for each function. We will demonstrate the use of such tools by incorporating open source libraries for image processing, such as OpenCV [2]. Although these libraries are readily available, they are not Grid-enabled. To get around this, each tool could be wrapped by a service. We also anticipate the use of machine learning techniques to assist with performance measure predictions for the image processing tools.

4.2. Example Scenario

The video clips from EcoGrid are characterised by various features that have to be processed by the system. These features include, for example, various species of fish, varying levels of brightness (e.g. sun light variation), complex and cluttered backgrounds and differing noise levels. The data is also captured from distributed sensors, thus the processing is to be conducted in Edinburgh for data that originated from Taiwan. Furthermore, users could be based in different locations, for instance, marine biologists from Taiwan, or their contacts in other places. Wrapping each tool in a service would enable each of them to be a Grid service. The workflow enactor and ontologies are also wrapped as services. Using Grid services, coupled with Planning and Case Based Reasoning (CBR), would enable a virtual environment to be built on top of a Grid infrastructure. This would allow distributed video analysis to be performed with more success than current technology could afford to perform. For instance, sensor grid nodes in Fu-Shan National Park capture data for a week that is required for our initial study. The user, a domain expert who wishes to classify tiger fish in a video, requests this by supplying the appropriate input parameters. The data is packaged for Grid use and sent to our system where it is determined that tools “X”, “Y” and “Z”, each not necessarily from the same location, should be used on the video clip because the features best match these tools’ capabilities. The analysis is performed and sent back to the user. If the user is a domain expert, s/he could validate the accuracy of the classification performed by providing appropriate feedback to the system.

5. Conclusion

Providing Semantic Grid services for video analysis would prove useful to support distributed image processing within a virtual enviroment. We have proposed a generic framework that enables an adaptive workflow enactment for video analysis in the EcoGrid utilising a hybrid approach of AI Planning and Case-Based Reasoning. Active and notable efforts in the development of Grid workflow systems could benefit from the semantic- and performance-based emphases that our framework proposes.

References

[1] EcoGrid National Center for High Performance Computing, Taiwan, 2006. http://ecogrid.nchc.org.tw/.

[2] Open Computer Vision (OpenCV) Library 1.0, 2006. http://sourceforge.net/projects/opencvlibrary/.

[3] E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G.Mehta, S. Patil, M. Su, K. Vahi, and M.Livny. Pegasus: Mapping Scientific Workflows onto the Grid. In *Across Grids Conference 2004, Nicosia, Cyprus*, 2004.

[4] I. Foster and C. Kesselman, editors. *The Grid: Blueprint for a Future Computing Infrastructure*. Morgan Kaufmann, 2nd edition, 2003.

[5] K. Gupta and A. Montezemi. Empirical Evaluation of Retrieval in Case-Based Reasoning Systems using Modified Cosine Matching Function. In *IEEE Trans. Systems, Man and Cybernetics*, 1997. Vol. 27(5), pp 601-612.

[6] C. Gutierrez, C. Hurtado, and A. Vaisman. Temporal RDF. In *Proceedings of European Conference on the Semantic Web (ECSW05)*, Jan 2005.

[7] B. Ludäscher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E. Lee, J. Tao, and Y. Zhao. Scientific Workflow Management and the Kepler System. *Concurrency and Computation: Practice & Experience*, 2005.

[8] D. McGuinness and F. van Harmelen. *OWL Web Ontology Language*. World Wide Web Consortium (W3C), 2004. http://www.w3.org/TR/owl-features/.

[9] G. Nadarajan, Y.-H. Chen-Burger, and J. Malone. Semantic-Based Workflow Composition for Video Processing in the Grid. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI-2006), Hong Kong*, 2006.

[10] T. Oinn, M. Addis, J. Ferris, D. Marvin, M. Senger, M. Greenwood, T. Carver, K. Glover, M. Pocock, A. Wipat, and P.Li. Taverna: A Tool for the Composition and Enactment of Bioinformatics Workflows. *Bioinformatics*, 2004. Vol. 20, No. 17, pp 3045-3054.

[11] D. D. Roure, N. R. Jennings, and N. R. Shadbolt. The Semantic Grid: Past, Present, and Future. In *Proceedings of the IEEE Volume 93, Issue 3, pp 669-681*, March 2005.

[12] I. Taylor, B. Schutz, B. S. Sathyaprakash, M. Shields, I. Wang, A. Harrison, C. Robinson, E. Al-Shakarchi, I. Kelley, and R. Zhu. The Triana Project, Cardiff University, 2003. http://www/trianacode.org.

[13] J. Yu and R. Buyya. A Taxonomy of Workflow Systems for Grid Computing. *Journal of Grid Computing*, Jan 2006. Volume 3, Numbers 3-4, pp 171-200.