Analyzing High Energy Physics Experiments

A Multi-Agent Approach

Lito Perez Cruz
School of Computer Science
RMIT
Melbourne, Victoria 3000,
Australia
lcruz@cs.rmit.edu.au

Muthukkaruppan Annamalai
Department of Computer Science and Software Engineering
The University of Melbourne
Melbourne, Victoria 3056,
Australia
mkppan@cs.mu.oz.au

Leon Sterling
Department of Computer Science and Software Engineering
The University of Melbourne
Melbourne, Victoria 3056,
Australia
leon@cs.mu.oz.au

ABSTRACT

In this study, we discuss our experience in applying an open agent system to solve computing problems found in the field of High Energy Physics (HEP). Specifically, through the use of agent technologies, we have taken a grid computing and multi-agent system approach in helping physicists do their work of analysing HEP experiments. HEP analysis is characterised by high demand for computer resources and high demand for the physicists time in deciphering what analysis steps to take. It is our contention that this application domain serves well as a highly suitable model for testing the feasibility and validity of theories found in the field of agent oriented computing. More importantly, we suggest further that this application area will benefit greatly from the advances found in agent and grid computing technologies

Categories and Subject Descriptors

H.4.m [**Information Systems**]: Miscellaneous; D.2 [**Software**]: Software Engineering; D.2.8 [**Software Engineering**]: Metrics—*agent oriented programming, grod computing*

General Terms

agent oriented technology

1. INTRODUCTION

The process of analysing High Energy Physics (HEP) experimental results involves various statistical physics calculations and simulations. These calculation algorithms are quite involved and repetitious requiring a good deal of computer processing power. Ideally also, these algorithms should be shared within a collaboration project. Thus physicists are availing the untapped computing resources in their community through the use of GRID computing concepts[7, 1].

Within the GRID physics community, most research however focus on utilization issues such as task scheduling and resource brokering[3]. They do not address the automating of the analysis performed by the physicists but rather, they heavily concentrate on tackling resource allocation problems. The bulk of the work however, is not only in determining how to maximize computer processing performance but on how to free up the physicists from extensive programming and cause them to channel their energies instead towards experimental analysis(the real task). Up to now this has not been fully addressed.

We believe that by using agent technologies, physicists will decrease their time spent in debugging and programming their programs. Physicists in general would rather focus their attention on analyzing physics phenomena rather than deal with programming minutiae. We believe that an open multi-agent system will help the physicists minimize their programming efforts and maximize their time devoted to qualitative analysis. Hence, HEP will benefit greatly from advances made in agent technologies.

In this paper, we briefly point the considerations that must be made in implementing a distributed open system of agents that are shared and are used to aid physicists analyze HEP experimental data.

2. SOLUTION REQUIREMENTS

Physicists working in the HEP arena share experimental data and software calculation algorithms within their collaborative projects. An example of this would be in the BaBaR[4] and BELLE[9] projects (to name a few). Typically a physicist (also called an analyst), will import the relevant softwares, modify them and run them to test for various hypotheses. These activities are time consuming manually and electronically. In envisioning a scheme whereby their needs are addressed, the physicists we are working with enumerated with us the ideal properties their solution should have.

1. The system should be open such that new calculation algorithms can be utilized.

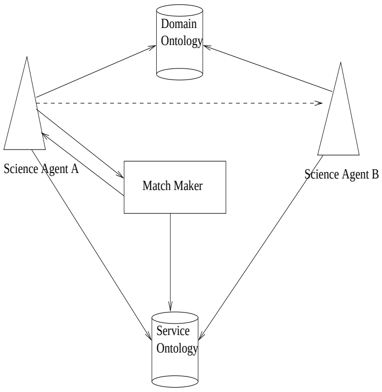

Figure 1: Architecture

2. The system should allow the sharing of calculation algorithms so as to maximize reuse.
3. The calculation algorithms should be discovered dynamically.
4. The system should allow for maximum automation of the analysis process.
5. The system should allow for distributed processing.
6. The system should allow for a high level tool to describe and execute work to be done.

As can be gleaned from the above goals, a GRID and agent approach is the ideal scheme that conveniently addresses these goals[10] (except for goal 6). Goals 1, 2 and 3 clearly allude to the service oriented nature of the problem domain and goals 4 and 5 make agents as suitable programming solutions.

3. THE ARCHITECTURE

From the above, we can envision an agent performing the calculation and setting up steps that is normally performed by the physicist. For this to happen, we are taking as a guide, the Multi-Agent System (MAS) model found in the RETSINA project[11]. The analysis done by the physicists will be delegated broadly to an agent (what we call a *science agent*). Currently there are computational programs in Fortran and C++ that are in place except they require many manual interventions and executions. These routines will be wrapped inside a service provider agent and mapped into an *agent service* that can be shared, offered and consumed by other science agents in the grid network.

We outline now the issues or challenges that we have uncovered in delivering such an agent based solution.

3.1 Agents

The analysis work that is to be performed by the physicist is embodied in a science agent (task agent) as shown in Figure 1. When physicists create calculation algorithms, we then embed this calculation inside an agent such that others may avail of it i.e. turn it into a service. When the physicist desires an analysis to be performed, the analysis steps are described into a science agent through some high level tool. The agent is then launched in the web or network and performs the work in a grid environment.

Issues: How should we generate the science agent? Physicists today do not want an additional programming language, there are already legions of them. Some kind of a graphically based user interface is necessary to make their work a breeze otherwise they will be burdened again with programming concerns. At the moment, there is a need to architect this such that the resulting tool is less abstract and keeps them in their domain of expertise. We are exploring the provision of a UML like diagraming tool for generating agent programs[8].

3.2 Services

As stated earlier, the calculation algorithms developed by fellow physicist will be viewed as agent services that is wrapped in a science agent. This is a major concentration of our work as we see this concept providing distribution, parallel computation and software re-use.

Issues: There are several issues that must be overcomed in constructing services from the routine analysis that are being performed by the physicists. Firstly there is the problem of identification. The analysis being performed by the physicist should be identified clearly. Once identified, they are then described. Once described, they are then analyzed for sub-components. These components may be identified as further services. In other words we have to drive the analyses down to its most primitive or atomic parts. This will maximize reuse. Only when the services have been identified in good precision will the task of automation proceed. This step we feel is necessary before we can encapsulate their calculations into services but more importantly, help should be provided by computer scientists in guiding physicists to think of their analysis steps as a composition of minute building block services. The task is to train physicists in spotting services in their analysis process. Studying how this might be efficiently achieved is itself another subject of research.

There is also the issue of what service model should we adopt? The view that services are actions performed by agents is widely noted. Some view services as a type of agent behavior. We are exploring with the idea that services should be looked at as a type of "work" that the agent performs on behalf of its user (similar to the concept of work as found in physics). Though services are actions, it is far more than that. We realize that others have already extended it beyond the action paradigm. In the physics domain, and particularly in the area of HEP, Monte Carlo simulation is widely used. Often in this regard, the parameters are tweaked depending on some aspect of analysis but in an agent scenario, the number of these parameters is not fixed, is not small and neither is it enumerable at design time. For example, the Fisher Discriminant, a statistical tool for calculating a likelihood that a particle event is of a certain type, combines likelihoods from a number of selection "variables". It may not be practical to write individual services dealing with each of these variables, so we are also asking if there are short cut methods to do this and if there are peculiar service properties that are most specific to HEP physics services. As an example, cost, speed and queue depth are likely properties (to name a few) that will make a service attractive or undesirable.

In our study we have already identified some basic analysis steps which HEP physicists perform and within them we have identified candidate services. Admittedly, these services have been identified from a high level standpoint and need more refinement and work. In Figure 3 we provide a very basic snapshot of a sample service we have identified out of those we have already identified so far. Figure 4 shows the process diagram of one critical but common analysis performed by an analyst, that of searching for a target particle. Those processes designated by a rectangle are candidates for becoming services.

3.3 Broker - Match Maker

In Figure 1 there is a broker or Matchmaker agent whose job is to act as mediator for other agents performing some analysis. The match maker receives queries from science agents to find out if there are services in the grid that can be performed by other agents on its behalf. The broker acts as umpire and intelligently decides between one service to another. Once this has been answered, the science agent contacts the providing agent and negotiates a deal for performing the service.

Issue: A more sophisticated description of services to allow intelligent choices must be made. We are studying the need for a service description language that can be used for representation and reasoning on services. It is expected that this language will be used by the science agents and the matchmaker. What form and characteristics this language should have is under investigation. Additionally we intend to conform to the FIPA specification on directory services [6]

3.4 Ontologies

In an open MAS, it is almost certain that an ontology will be required as these allow heterogenous agents to come and go into the system. It is not realistic to conceive that HEP physicists will use their terms and concepts consistently around the globe. Besides, new theories for explaining particle behavior may require new terms to be introduced to designate such theory. Ontologies provide the MAS a way for agents to talk to each other such that their terminologies may map into some common term.

When a physicist performs an analysis of experiment, it may be that there is a need to review existing analysis results first. Information gained here will be fed into new analyses to be performed and then again, the result here will be used in reporting new findings or theories. The two processes feed back on each other in a spiraling manner as shown in Figure 2. Our view is that there is a need for at least two ontologies. Firstly there is the domain ontology which houses HEP concepts and their relationships. At present, this ontology is initially considered to cater for analysis results. Secondly there is a need for service ontology as well. This is a narrow ontology that houses the concepts pertaining to calculation services and serves the bottom process. Each level of analysis as shown in Figure 2 is served by these ontologies.

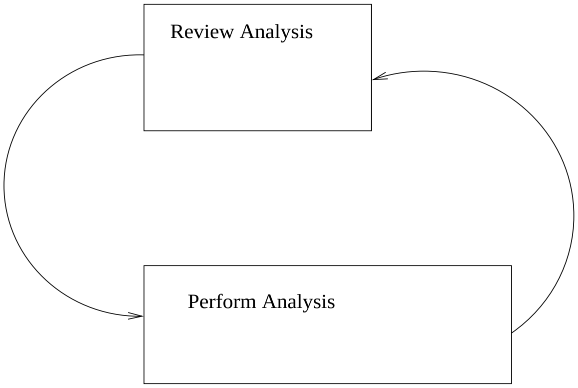

Figure 2: Analysis Levels

3.4.1 *Domain Ontology*

The domain ontology is used by agents whose task is to review the marked up results found in published papers. This is expected as a whole to be a HEP ontology. The construction of this ontology is beset with non trivial problems.

Issues: Firstly there is the issue lack of little common domain expertise between Physicists and Computer Scientists. Physicists are conversant with their terms but are not necessarily skillful in organizing ontologies or agents, while computer scientists though experienced in these may be beginners when it comes to high energy physics. The process of interaction reveals the need to help each other understand each others' concerns in the process of constructing agents and ontologies. There is a need to cross educate each other. Moreover, there are so many terms and concepts present in HEP that needs to be cataloged and analyzed. This is complicated in that the number of concepts and theories grow at an exponential rate. Aside from constructing the domain ontology, we are also pursuing research in ontology reconciliation as a way for physics communities to device their own ontologies and make them compatible.

3.4.2 *Service Ontology*

When it comes to service discovery, one may argue that this could be solved by the use of web services and the Universal Description, Discovery, and Integration scheme(UDDI). However, unlike ontologies, UDDI currently lacks the mature dynamism that is needed in an open heterogeneous environment [5] wherein computing entities may come and go. For this reason, we chose to use agents with ontologies in addressing service discovery.

The service ontology in our architecture is used by the matchmaker and the science agent as a vehicle for mutually discovering services dynamically without a hard pre-registration process. The service provider will announce its service availability and service properties to the matchmaker upon system entry. The provider will also inform the matchmaker when it exits the environment. Information such as these will be kept in the service ontology by the matchmaker.

Issue: As stated in the sections above, the major issue is in the identifying and classifying the phases of work a HEP physicists goes through and precisely pin pointing them to

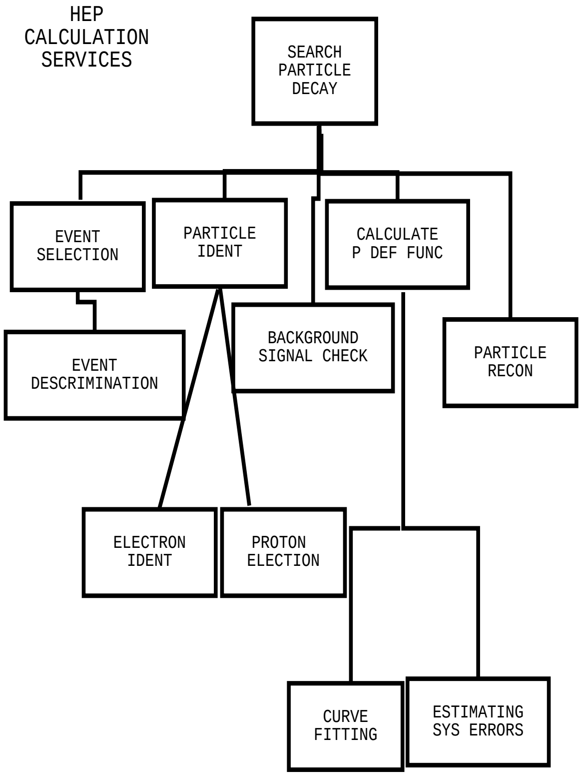

Figure 3: HEP Calculation Services (sample)

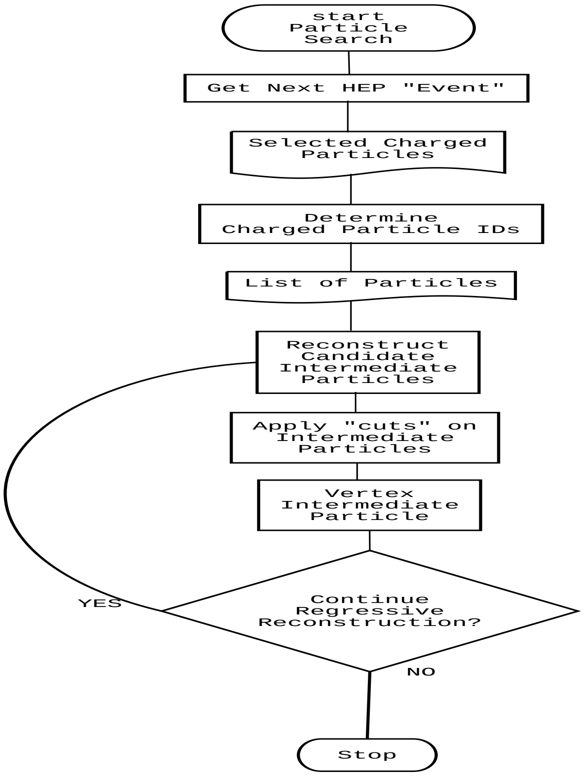

Figure 4: Particle Search Analysis(an example)

be candidates for services. This is input to the construction of the service ontology. However, the differences in knowledge expertise and communication between physicists and computer scientist is a factor that must be overcomed to speed up the progress of the project. In terms of ontology language, we intend to evaluate and record our experience using DAML-S as the ontology language to describe these services[2].

The sample services identified thus far in Figure 3 forms also as a sample of service ontology for the section 'Perform Analysis' phase found in Figure 2 .

4. CONCLUSIONS & FURTHER WORK:

Using grid computing to address calculation intensive processes such as that found in high energy physics experiments is not new. However our study addresses the other need of the physicist, that is to alleviate their programming load. This has not been addressed so far and our approach is the use of grid and agent technologies in a unified infrastructure set up. In this paper,there are several issues that are presented and addressed, they are the subject of on going investigation in our project. The issues we are uncovering are relevant to the practical process of implementing a multi-agent scheme to the domain of high energy physics; they should guide the community as to what tools or techniques may be devised to fully realize the advantages the vision brings. We intend to share further these findings in future research works.

Acknowledgments

We wish to thank Associate Professor Lin Padgham of School of Computer Science at RMIT and Drs. Glenn Moloney and Martin Sevior of the Physics Department, University of Melbourne for their valuable help in this project. This project is part of a grant funded by the Victorian Partners for Advanced Computing(http://www.vpac.org).

5. REFERENCES

[1] G. Allen, W. Benger, T. Dramlitsch, T. Goodale, H.-C. Hege, G. Lanfermann, A. Merzky, T. Radke, E. Seidel, and J. Shalf. Cactus tools for grid applications. *Cluster Computing*, 4(3):179–188, 2001.

[2] A. Ankolekar, M. Burstein, J. R. Hobbs, O. Lassila, D. L. Martin, S. A. McIlraith, S. Narayanan, M. Paolucci, T. R. Payne, K. Sycara, and H. Zeng. Daml-s: Semantic markup for web services. In *International Semantic Web Work-shop (SWWS)*, 2001.

[3] M. Athanas and T. J. et. al. A high-performance data server for high energy physics. In *5th Annual Workshop on I/O in Parallel and Distributed Systems (IOPADS), San Jose.* IOPADS, November 1997.

[4] B. Collaborators. *BaBar Project Home Page.* http://www-public.slac.stanford.edu/babar/.

[5] M. Dumas, J. O'Sullivan, M. Heravizadeh, D. Edmond, and A. Hofstede. Towards a semantic framework for service description. In *Proc. of the 9th Int. Conf. on Database Semantics, Hong-Kong, April 2001. Kluwer Academic Publishers.*, 2001.

[6] FIPA. *FIPA Abstract Architecture Specification*, volume XC00001J. http://www.fipa.org/specs/fipa00001/.

[7] K. Marzullo, M. Ogg, A. Ricciardi, A. Amoroso, F. Calkins, and E. Rothfus. Nile: Wide-area computing for high energy physics. In *7th ACM SIGOPS European Workshop. Connemara, Ireland.* TeX Users Group, Sept 1996.

[8] J. Mylopoulos, M. Kolp, and J. Castro. UML for agent-oriented software development: The tropos proposal. In *The Unified Modeling Language*, pages 422–441, 2001.

[9] E. Prebys. *Belle Physics Prospects.* http://www.hep.princeton.edu/ prebys/misc/slovenia/talk2.pdf.

[10] D. Roure, N. Jennings, and N. Shadbolt. *Research Agenda For the Semantic Grid: A Future E:Science Infrastructure*. Report commissioned for EPSRC/DTI Core e-Science Programme., www.ecs.soton.ac.uk/˜dder/semgrid.pdf, 2001.

[11] K. Sycara, J. Lu, and M. Klusch. *Interoperability among Heterogeneous Software Agents on the Internet.* Carnegie Mellon University, PA (USA),Technical Report CMU-RI-TR-98-22, PA (USA), 1998.