Techniques to Produce Good Web Service Compositions in The Semantic Grid

Eduardo Blanco

Universidad Simón Bolívar,
Departamento de Computación y Tecnología de la Información,
Apartado 89000, Caracas 1080-A, Venezuela
`{eduardo}@ldc.usb.ve`

Advisors: Yudith Cardinale and María Esther Vidal

Abstract. The Grid has emerged as a new distributed computing infrastructure for advanced science and engineering. It aims at enabling resource sharing as means to facilitate a problem-solving approach for dynamic environments that is based on the integration of available resources. Wide agreement that a flexible GRID is not possible without the support of semantic technologies, has lead to the term "SEMANTIC GRID". Transparent service composition has a great potential to facilitate the integration of WEB SERVICEs deployed in the GRID. However, discovering, dynamically composing and optimizing large-scale WEB SERVICEs (e.g., in the rank of 1,000 to 100,000 services) is a challenging problem. We approached the problem of WEB SERVICE compositions in the GRID as an optimization problem, in which for a given request a good plan has to be generated. In this work, we propose the definition of several optimization strategies to identify good orderings of WEB SERVICE compositions.

1   Research Problem

The GRID provides the protocols, services, and software development kits needed to enable flexible, controlled resource sharing on a large scale. Automatic discovery and coordination of services in the GRID is not only desirable, but necessary. Users need tools that assist them on taking full advantage of the possible large number of available services. Any form of automated composition needs an infrastructure where all resources are adequately described in a machine-processable form. SEMANTIC WEB technologies provide the means to incorporate such descriptions to the GRID infrastructure leading to the SEMANTIC GRID. Thus, it is possible to apply SEMANTIC WEB technologies in grid-computing developments.

Numerous efforts have focused on the task of service discovery and WEB SERVICE composition (the `WSC` problem). However, few have addressed the problem of identifying efficient WEB SERVICE compositions in a large-scale scenario, as the one presented in the GRID vision [1]. In such scenarios, the association among services may be extremely complex, and the search space of possible WEB SERVICE compositions could be very large. In consequence, the task of

finding an optimal plan could be not feasible in real time; thus, our objective is to find sub-optimal plans in a reasonable time.

In GRID platforms requests are usually scheduled as batch processes, and real time constraints are relaxed. Thus, it is possible to do a greater exploration of the space of possible WEB SERVICE compositions, while the batch process is waiting to be evaluated.

On the other hand, some GRID resources can only be accessed at certain times. It may be required to incorporate into the space of alternatives, the possibility of executions where intermediate results are stored and retrieved according to time restrictions. In other words, all services used by a plan do not need to be available at the same time.

In this work, we propose the definition of optimization strategies to identify good orderings of WEB SERVICE compositions in large-scale WEB SERVICEenvironments, e.g., in the rank of 1,000 to 100,000 available services. These strategies will follow different meta-heuristics to explore the space of possibilities, while minimizing the estimated evaluation cost. We have already defined two strategies that prune the space of possibilities. Our initial results show that the compositions identified by our algorithms, are close to the optimal solution.

2   Related Work

The problem of identifying a good `WSC` in the GRID is related to problems that have been studied in query optimization and `WSC`. In this section we consider the main related approaches in each of these areas.

In the context of the Web, several strategies have been presented to identify good evaluation plans where sources have limited query processing capabilities; then, the optimizer task is to identify a good ordering of the sub-goals of a query where limited processing capabilities of each considered source, are satisfied [2, 3].

Typically, existing approaches achieve the challenge of identifying good plans by representing query capabilities as binding patterns and using these patterns and meta-heuristics to traverse the space of source plans. Each sub-goal in the query is rewritten in the plan by using sources that define the sub-goal; limited processing capabilities of the sources are satisfied in the plan with query bindings or attributes projected out by previous sources in the plan.

The `WSC` problem has been extensively treated in the literature, and diverse solutions that take advantage of `AI` techniques and Search Meta-Heuristics, have been proposed. First, in the context of `AI`, the `WSC` problem has been represented as a planning problem where actions to be taken by the planner are defined in terms of service preconditions and effects.

The description of a planning domain includes a set of planning operators and methods that establish the way a task can be decomposed into smaller subtasks. The description of a planning problem contains an initial state as in classical planning. Instead of a goal formula, there is a partially-ordered set of tasks to accomplish. Planning proceeds by decomposing tasks recursively into smaller and

smaller subtasks, until primitive tasks, which can be performed directly by using one planning operator, are reached. For non-primitive tasks, the planner chooses an applicable method, instantiates it to decompose the task into subtasks, and then chooses and instantiates other methods to decompose the subtasks even further. If the constraints on the subtasks or the interactions among them prevent the plan from being feasible, the planning system backtracks and tries other methods.

As any planning problem, the approach presented in [4], requires the formalization of the domain-dependent control knowledge in the planner. Thus, a domain expert is needed in order to achieve good performance in real-world domains.

An approach that uses *Answer Set Programming* is presented in [5]. It shows that service descriptions can be expressed in a rule based language that allows to search a repository efficiently and to build solutions that solve a goal with respect to soft and hard constrain. The author reports that the solution performs very well in this rather simple domain. It defines a simple cost function instead of a utility function. Its strength is that it provides means to gain all solutions for a given problem despite the cost of computation time and space. He states that dedicated software employing fast heuristic algorithms could rapidly find a good solution for user requests in much reasonable times.

In the context of Search Meta-Heuristics, the SAM (SERVICE AGGREGATION MATCHMAKING) algorithm is defined [6]. It makes use of an OWL-S ontology, and explicitly returns a sequence of atomic processes that need to be executed in order to achieve the desired result.

SAM follows a greedy approach in which only one sub-plan is generated in each iteration. In each sub-plan, a sub-set of the output attributes is produced considering some of the bindings given in the query. The algorithm ends when all the output attributes are produced. In terms of time, SAM is able to scale up in environments with a moderate number of services (e.g., in the rank of 100 to 200 services). However, since SAM does not consider any cost metric or optimization criteria to compose the services, plans produced by SAM may be costly. To exacerbate this problem, SAM may add processes to the plan that are not needed to produce the output required by the user. Thus, the quality of generated plans may be far from optimal.

SEMANTIC GRID [7] is an infrastructure where it is possible to apply SEMANTIC WEB technologies in grid-computing developments [8, 9]. In the context of SEMANTIC GRID, the project ARGUGRID [10] provides a new model for programming the GRID; it uses argumentative agent technology and semantic descriptions to facilitate the dynamic composition of services.

Finally, in [11], the `WSC` problem is defined as a planning problem where a workflow is generated automatically. It takes as inputs the desired data products, and the planner uses heuristic control rules to identified a high-quality solution. Plan quality is measured with respect to a global utility function that does not represent the dynamic properties of GRID environments.

3   Contributions

We hypothesize that if there is a large number of WEB SERVICEs published in different sites then, services' performance may vary. Thus, it is imperative that solutions to the WSC problem consider an estimate of the service evaluation cost. This cost is used to prune the space of possibly costly plans of services, and to identify the service composition with the lowest estimated cost.

We propose solutions to the WSC problem for large-scale platforms, e.g., GRIDs. These solutions will be adapted to consider the dynamic characteristic presented in such platforms. In particular, we address the problem of selecting and coordinating services that satisfy some of the constraints presented in the GRID, while the evaluation time is minimized.

Plans will be generated considering the time constraints of GRID resources. Thus, we will incorporate into the space of alternatives, the possibility of executions where intermediate results are stored and retrieved according to these constraints.

4   Evaluation

To validate the quality of our techniques, we plan to conduct experimental studies that compare our approach against some of the existing projects. We propose to use at least two available test sets - EEE05 and ICEBE05[1].

We will also analyze the performance of our solutions in real-world scenarios. For this, we will generate a large number of ontological WEB SERVICE descriptions for a given scientist community. WSBen[12] is a tool able of automatically generate sets of WSDL service descriptions. WSBen is inspired by extensive studies on real WEB SERVICEs and therefore, it is designed to support various network topologies and distributions. We will extend this tool to generate ontological service descriptions.

5   Work Plan

We consider that the project can be achieved within the following stages:

1. We propose approaches that adapt some optimization techniques to the WSC problem. We have already defined two extensions to the algorithm SAM. Initial results show that the extensions are effective. In terms of plan quality and optimization time, our generated plans, outperform results produced by SAM [13].
2. We will extend WSBen to generate large sets of ontological WEB SERVICE descriptions.
3. We will propose cost models and test them with the generated sets; we will study the behavior of our approaches using different distributions and WEB SERVICE inter-relations.

---

[1] They are available at *http : //www.comp.hkbu.edu.hk/ ctr/wschallenge/news.html#dataset*

4. We will evaluate existing strategies proposed to discover and compose services in The SEMANTIC GRID.
5. We will enrich the service composition algorithm and the cost models with the experiences on SEMANTIC GRID. In this sense, it is mainly important to consider the dynamic characteristic of grid platforms.

References

1. Foster, I., Kesselman, C., Tuecke, S.: The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications **15**(3) (2001)
2. Levy, A., Rajaraman, A., Ordille, J.: Querying heterogeneous information sources using source descriptions. In: Proceedings of the VLDB Conference. (1996)
3. Papakonstantinou, Y., Gupta, A., Haas, L.: Capabilities-based query rewriting in mediator systems. In: Conf. on Parallel and Distributed Inform. Systems. (1996)
4. Kuter, U., Sirin, E., Nau, D.S., Parsia, B., Hendler, J.A.: Information gathering during planning for web service composition. In: Intl. Semantic Web Conf. (2004) 335–349
5. Rainer, A.: Web service composition using answer set programming. In: PuK 2005. (2006)
6. Brogi, A., Corfini, S., Popescu, R.: Composition-oriented service discovery. In: Proc. of Software Composition'05, LNCS. Volume 3628. (2005) 15–30
7. De Roure, D., Jennings, N., Shadbolt, N.: The semantic grid: A future e-science infrastructure. In Berman, F., Fox, G., Hey, A.J.G., eds.: Grid Computing - Making the Global Infrastructure a Reality. John Wiley and Sons Ltd. (2003) 437–470
8. Goble, C., Roure, D.D.: The grid: an application of the semantic web. SIGMOD Rec. **31**(4) (2002) 65–70
9. Pahl, C.: An ontology-based framework for semantic grid service composition. Grid Services Engineering and Management **3270** (2004) 63–77
10. Programme, E.C.S.F.: ARGUGRID (2006) http://www.argugrid.org/.
11. Deelman, E., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Lazzarini, A., Arbree, A., Cavanaugh, R., Koranda, S.: Mapping abstract complex workflows onto grid environments. Journal of Grid Computing, Vol. 1, No. 1, (2003)
12. Oh, S.C., Kil, H., Lee, D., Kumara, S.R.T.: Wsben: A web services discovery and composition benchmark. In: ICWS '06: Proceedings of the IEEE International Conference on Web Services (ICWS'06), Washington, DC, USA, IEEE Computer Society (2006) 239–248
13. Blanco, E., Cardinale, Y., Vidal, M.E., Graterol, J.: Techniques to produce optimal Web Service Compositions. In: 2008 IEEE Congress on Services 2008 - Part I (SERVICES-1 2008), Honolulu, Hawaii, USA, IEEE Computer Society (2008) To appear.