

3.6. Composite Service Executor

When the planner comes up with a valid plan for the given request, the Composite Service Executor converts the plan into an executable process and executes it. Finally, the response is personalized based on the user profile parameters and then returned to Consumer.

4. Related Works

Many proposals of web services composition methods have been presented in recent years [3]. In this section, we present a brief overview of some techniques that deal with automatic web service composition. We consider only techniques that use service dependency information, graph models, and semantics.

Hashemian et al. [4] store I/O dependencies between services in the DG and then build composite services by applying a graph search algorithm. The authors consider only the matching and dependencies between input and output parameters without considering functional semantics, thus they cannot guarantee that the generated composite services provides the requested functionality correctly. Arpinar et al. [5] present an approach which not only use graphs for web service composition, but also use semantic similarity. They consider edges with weights and deploy a shortest-path dynamic programming algorithm based on Bellman-Ford's algorithm. The authors consider the execution time of each service and input/output similarity but they don't take into consideration the nonfunctional attributes of services. Talantikite et al. [6] propose to pre-compute and store a network of services that are linked by their I/O parameters. The link is built by using semantic similarity functions based on ontology. They represent the service network using a graph structure. Their approach utilizes backward chaining and depth-first search algorithms to find sub-

graphs of services to accomplish the requested task. They propose a way to select an optimal plan in case of finding more than one plan. However, they also create the graph at the composition time which incurs substantial overhead.

5. Conclusion

In this paper, we propose a Framework to customize and automate composition of E-government web services. We have described its main components and outlined their interactions. The implementation and evaluation of the proposed Framework in the domain of E-gov is the main focus of our ongoing work.

6. References

- [1] <http://unpan1.un.org/intradoc/groups/public/documents/un-dpadm/unpan048580.pdf>
- [2] H. Elmaghraoui, I. Zaoui, D. Chiadmi and L. Benhlime, "Graph based E-Government web service composition," *IJCSI International Journal of Computer Science* Vol. 8, July 2011
- [3] A. Alamri, M. Eid and A. Elsaddik, "Classification of the state-of-the-art dynamic web services composition," *International Journal of Web and Grid Services*, 2006, Vol. 2, pp. 148-166.
- [4] S. Hashemian and F. Mavaddat, "A graph-based framework for composition of stateless web services," *Proceedings of ECOWS'06*, IEEE Computer Society, Washington, DC, 2006
- [5] I.B. Arpinar, B.A. Meza, R. Zhang and A. Maduko, "Ontology-driven web services composition platform," *Inf. Syst. E-Business Management*, 2005, 3(2):175-199.
- [6] H.N. Talantikite, D. Aissani and N. Boudjlida, "Semantic annotations for web services discovery and composition," *Computer Standards Interfaces*, Elsevier, 2009.