

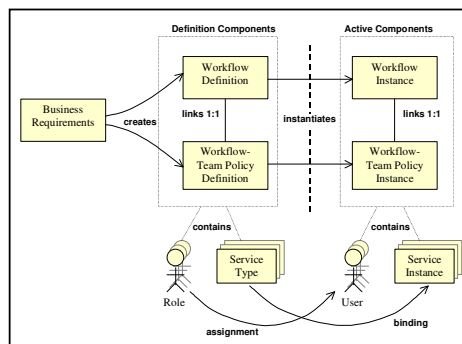
THE WHITE ROSE GRID

e-Science Centre of Excellence

Secure Service-Based Collaborative Workflow

Introduction

The Workflow-Team policy architecture is proposed as a solution for secure collaborative workflows that use services across enterprises. The collaborative workflows involve users sharing knowledge to reach a business goal; an example would be combining experience to diagnose an aircraft engine. The collaborative environment would enable users from different organizations in distributed locations to share services and service instances (for data access or processing services). Access control to collaborative workflows can be simple if all parties are known ahead of time. However, there are situations where they may be numerous users, who are distributed and from different organizations. The access control also becomes more complex when the collaboration of a changeable team of users involves sharing services and service instances that may be supplied by further organizations.



By implementing workflow activities as services, then a service-oriented architecture (SOA) can provide loose coupling between business process definition and workflow implementation. Distributed services can be replaced or modified without the need to change the business process. This allows the workflow implementation to change more often than the business process definition and for services to be outsourced if required. Outsourcing leads to a business model that includes service suppliers and even commodity computing from compute resource suppliers. The combination of these creates the model for grid computing and VOs. In a situation where user and service collaborate across organisations,

secure methods of access control are important.

Collaborative Role-based Workflow

Workflow activities can be automated in computer systems. In a collaborative workflow, the users act in roles given in the workflow definition. The role has a responsibility for completing tasks in the business process. As users enact roles in the workflow, they can combine their skills forming a team to bring the workflow to completion. These role based access restrictions or permissions are defined by combing the business process and role definitions, and stored in the workflow-team policy definition.

Role-based access control can simplify administration of the user permissions. When characterizing each collaboration by the users involved, role-based access control becomes too coarse, and fine-grained access control is necessary. If services create service instances, then each instance is associated with the collaboration of users. Therefore, dynamic recording of the association of users and service instances is required to enforce fine-grained access control.

Workflow-Team Policy Architecture

This work builds on role-based access control and extends other team-based and task-based approaches. The Workflow-Team Policy records services instances as temporal assets created during the workflow and shared between the users. The architecture creates a policy per workflow instances that is used for access control. The policy itself is a fine-grained record of the users and service instances.



The Workflow-Team Policy Instance is dynamic and linked to the active workflow instance. It provides the permissions for users to collaboratively access service instances.

The permissions are defined in the Workflow-Team Policy Definitions. The permissions state the action a role can perform on a service type. When a Workflow Instance is created a Workflow-Team Policy Instance is created. In the policy instance, roles become users and service types become service instances.

The Workflow-Team Policy Instance is dynamic and linked to the active workflow instance. It provides the permissions for users to collaboratively access service instances. To achieve the dynamic policy, the rules are managed by the workflow instance. As the workflow progresses, users join and leave, and service instances are created and destroyed. These actions in the workflow are used to record these new mappings from the role and service type definitions.

- BPEL used to define workflows, which wasn't suitable for GT3 services;
- Shibboleth for user attribute management, with SAML assertions;
- XACML policy descriptions;
- Policy decision engine PERMIS.

The use of PERMIS would be important in judging the feasibility of using decision engines for dynamic, instance based policies.

Grid computing has provided another method of exposing processing and resources to external companies in a service oriented architecture. Service providers and service integrators need methods to control access to their products, maintaining commercial sensitivity, but still offering availability to different organizations. Previous work has built in role-based access control to introduce teams and task-based contexts, or derive policies from business processes. This work builds on those to address how business processing can provide a secure integration platform for grid services, when service instances are used collaboratively across organizations.

This work is published in:

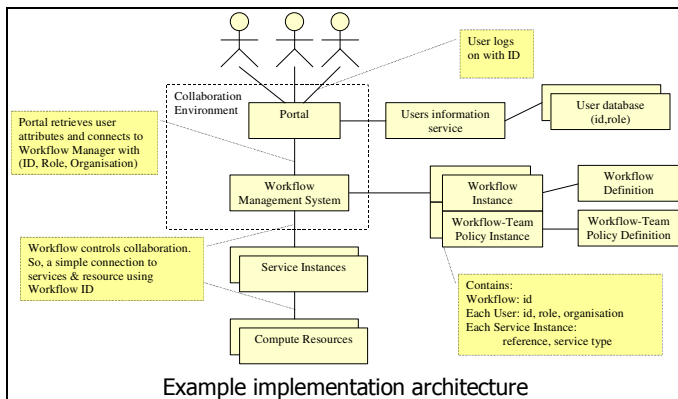
Russell, D., Dew, P. M. and Djemame, K. (2005) Service-Based Collaborative Workflow for DAME. In: *Services Computing, 2005. (SCC 2005). Proceedings. 2005 IEEE International Conference on*, 10-15 July 2005, Orlando, Florida. pp. 139-146

Further Information

Contact:

Duncan Russell
(duncanr@comp.leeds.ac.uk)

Informatics Institute, School of Computing
University of Leeds, LS2 9JT



Future Work

The Workflow-Team Policy architecture has been created by evaluating the implementation of the DAME demonstrator (Distributed Aircraft Maintenance Environment). This system used a portal and workflow management system to access grid services based on Globus Toolkit 3 (GT3). The next generation of the DAME Portal and Workflow Management System presents the opportunity to utilise rising standards along side the Workflow-Team dynamic policy, such as:

- Globus Toolkit 4, using the WS-RF resource model;
- Service instance references become WS-Address end points;