

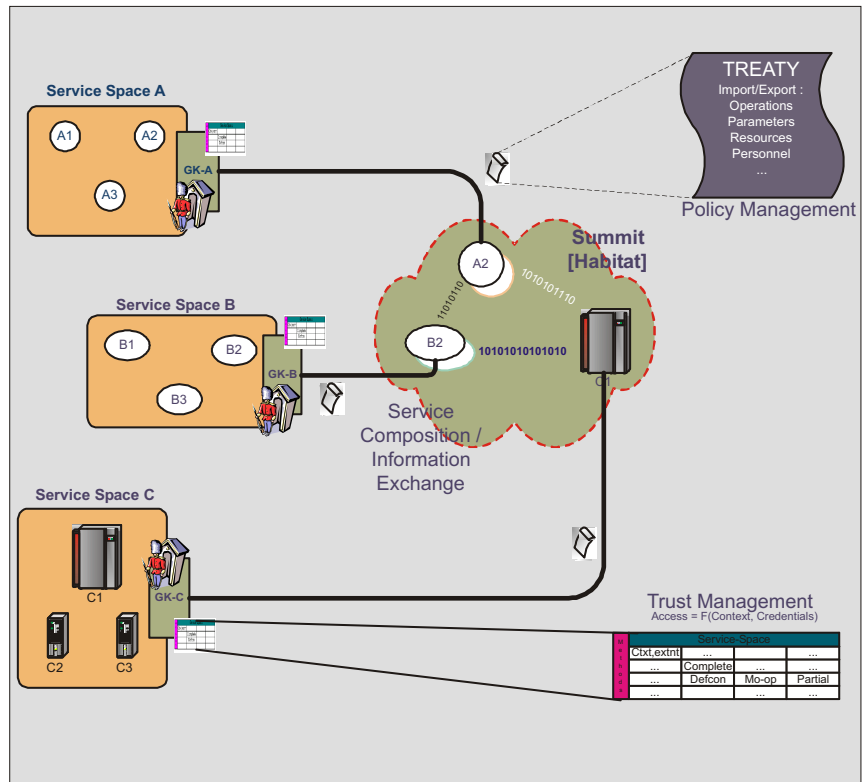
# PSL DISCUS

## Overview

The **DISCUS** (Decentralized Information Spaces for Composition and Unification of Services) project is building Columbia PSL's "proof-of-concept" realization of NICCI Habitats. DISCUS focuses on rapidly forming alliances among existing legacy systems and services that may span organizational boundaries, to deal with a temporary and possibly unique problem. The idea is that this specially-formed and dynamically-integrated suite of tools and data, termed a "**Summit**", will help operational teams to achieve quick understanding and resolution of the task at hand.

## Why DISCUS?

DISCUS Summits are rapidly and automatically assembled systems of systems that know what they are doing and why they are doing it. Summit formation and continuing operation considers the environment (including Service Spaces and other Summits), the task's goals, and its own capabilities, taking advantage of substantial amounts of appropriately represented policies, constraints, workflows, and other knowledge. Summits form cross-organization and multi-level security teams that succeed at missions that individual Service Spaces cannot achieve, and are robust in the face of "surprises".



A summit (habitat) is formed by composing services from different "**Service Spaces**", where each Service Space is a pre-existing collection of services allowing selective access to its services. Since services in a summit interact with outside world services, external to their own service space's security levels and content confidentiality considerations, access to resources is likely to be restricted. The extent of access allowed by a Service Space to external entities is based on their credentials and the nature of the problem, thus employing a dynamic trust model. A "**Treaty**", representing a contract, is dynamically formed (and reformed as needed as the mission progresses) amongst the Service Spaces of these participating services.

The primary enabling technology for DISCUS, to date, has been Web Services. A Web Service is a software application identified by a URI, whose interfaces and binding are capable of being defined, described and discovered by XML - thus making it language independent. Web Services support direct interactions with other software applications using XML-based messages via Internet protocols. While one critical goal of DISCUS is to interoperate with multiple backend component models (e.g., COM+, CCM, EJB), use of Web Services allows us a free choice of platforms for the infrastructure.

A **Service Space** is a cohesive domain in which various services, information, resources, personnel and/or legacy systems are placed under an administrative control. These capabilities need not be collocated and could even span organizational boundaries, as long as there are common policies and security models.

Each **Treaty** is defined pair-wise between any two Service Spaces, which allows each of the local Gatekeepers to form such contracts/agreements in a fully decentralized manner without involving any global authority. Treaty relationships are often asymmetric and never transitive: each Treaty between two Service Spaces must be formed explicitly (except that a Service Space, or a Summit, may aggregate and expose selected facets of internally federated Service Spaces/Summits).

A **Summit** is a virtual gathering of services, resources, etc. for mutual interaction and exchange. The summit is essentially the enactment of the computation and information exchange. Once the capabilities collected within a summit have accomplished their mission, the summit can be dissolved or passivated - although logs may be kept for postmortem analysis, and summit 'ghosts' (templates) may be retained to even more rapidly form future summits.

The **Gatekeeper** is the 'traffic cop' of the DISCUS world. There is one Gatekeeper per Service Space. There cannot be any flow of data, services, etc. among Service Spaces without a Treaty. Once a treaty has been created, any further exchange of data, services, etc. flows through those two Gatekeepers. Each Gatekeeper enforces normative interaction between the 'enlisted' services of a summit by intercepting and verifying every operation according to the relevant treaty. Authorization-related information is asserted by security matrices, allowing both matrix and hierarchical relationships among cooperating organizations and their users and internal services.

## Features

**Quick short-term business affiliations.** Summits enable automatic and transparent integration among COTS, GOTS, open source, etc. applications, particularly emphasizing Web Services or legacy systems wrapped as Web Services. Web services apply web technologies such as HTTP and XML to the concepts of distributed computing technologies such as CORBA and DCOM.

**Information exchange in a truly language and system independent manner.** DISCUS addresses situations where multiple government and non-government organizations with incompatible legacy software and possibly conflicting policies suddenly need to communicate and work together, sharing services and information.

**Dynamic membership and trust models.** Every request is accompanied by the credentials of the requesting Service Space, and in some cases responses may be interrupted while in progress as membership and/or trust level changes due to changing circumstances.

**Dynamic Service Discovery.** Since the service discovery mechanism in DISCUS is based on the UDDI protocol, it is possible to take advantage of other advertised COS software.

## General Specifications

DISCUS is written using C# and Java, and runs on Windows .NET. (Of course, pre-existing legacy facilities may have been written in any language and run on any host.)

## Availability

DISCUS can be downloaded now from <http://www.psl.cs.columbia.edu/software/download>.

### Principal Investigator

**Gail E. Kaiser**  
(kaiser@cs.columbia.edu)  
Programming Systems Lab  
Department of Computer Science  
Columbia University  
500 W. 120th St., 450 CS Bldg  
New York, NY 10027

For more information, visit  
<http://www.psl.cs.columbia.edu>

© Copyright 2002, Trustees of Columbia  
University in the City of New York.  
All Rights Reserved.

The Programming Systems Laboratory is funded in part by Defense Advanced Research Project Agency under DARPA Order K503 monitored by Air Force Research Laboratory F30602-00-2-0611, by National Science Foundation CCR-9970790 and EIA-0071954, by the Office of Naval Research monitored by Naval Research Laboratory N000140110441, by Microsoft Research, and by a NEC Computers, Inc. equipment grant.