



Choosing a Groove Integration Technology

**Developing Integration Solutions Using Groove Enterprise
Integration Server and Groove Web Services**

December 2002

Executive Summary

One of the unique values of Groove software is that it provides a collaborative extension to existing enterprise systems. Corporate portals and extranets have allowed external users to gain access to customized data, account information, analytical tools and real business function. Groove allows those internal and external parties to interact around that centralized business data in an environment that is secure across firewalls, easy to use , and fosters faster and better decision-making.

This integration of centralized systems with the Groove decentralized collaborative environment is accomplished through a framework that employs a flexible array of techniques. This integration framework is a key infrastructure element for any company that deploys Groove in an enterprise environment, blending solution development, centralized management and administration, and end-user flexibility in a way that supports critical business processes and IT requirements.

As the use of Groove grows, there is increasing demand from organizations for new ways to integrate new systems and devices with Groove content stored in shared spaces. The opportunities for useful integration with Groove are endless, including extending Groove data and functionality to mobile devices integrating with “back office” applications, and integrating with other applications on the user’s local device.

In response to this growing demand, Groove Networks is committed to continually building developer resources and solutions for the Groove platform. This paper offers key considerations when choosing a method of Groove integration. The Groove Enterprise Integration Server, Groove Web Services, and custom connector tool technologies are discussed as potential integration solutions.

Who should read this paper?

This paper is a discussion of the advantages and tradeoffs of using different development methods to integrate with Groove collaboration and data components. The primary audience for this paper is developers who are already familiar with the Groove platform, who are looking to integrate with some portion of the Groove platform, and who are evaluating different methods of integration.

The Integration Challenge

Before starting a discussion about the issues surrounding integration with Groove, we first have to agree on what is meant by the term “integration.” As a platform with a rich set of services, Groove offers multiple approaches to integration. The following are examples of different types of integration challenges:

- **System or Server Integration.** This represents the most common understanding of the term “Integration.” A system integration project may involve pushing or pulling data from a Groove shared space to a wide variety of enterprise systems. An example is an organization that requires their sales related Groove shared spaces to be automatically updated with information from the company’s Customer Relationship Management (CRM) system. The integration challenge lies in building

the software that ties one system's data to the other, and controlling how and when the information is passed back and forth.

- **Local Integration.** Integration may also entail integrating Groove with other applications that are running on the same PC as Groove. An example of integrating Groove components into a local application would be displaying Groove presence information and messaging inside another ISV business process application (BPA). This would make process-oriented applications more valuable by allowing federated users to speed time-to-decision through secure ad-hoc collaboration without ever requiring that the users leave the BPA user interface.
- **Remote Device Integration.** Some organizations will extend Groove data and services to devices not running Groove Workspace. Examples include Macintosh and UNIX machines, as well as mobile PDAs and other handheld devices. Integrating portions of a shared space, including specific data and tools, with applications running on such devices provides a means to extend personal access for Groove Workspace users, as well as to extend access to individuals who are unable to install and run Groove themselves (e.g. Macintosh users).

Each of these approaches to Groove integration poses different challenges for a software developer. The remainder of this paper is intended as a discussion of alternative approaches to Groove integration.

Integrating with Groove

There are multiple methods for connecting Groove to a system, application, or device. Currently the Groove platform offers three ways to integrate with Groove, the Groove Enterprise Integration Server, Groove Web Services, and custom tool development (a.k.a. "connector tool").

Groove Enterprise Integration Server

Sometimes, members of a shared space need cross-firewall access to internal server-based information residing in customer relationship management systems, enterprise resource planning systems, document and knowledge management systems, etc. For example, a consulting team working together in a Groove shared space might require access to customer history, current-pricing schedules, or best practices guidelines. In such a case, it would be inefficient for multiple members to make independent calls to the centralized server: the server would have to process the same request and provide the same result multiple times. Instead, it makes more sense to provide Groove with access to the centralized system through a single point.



Figure 1 Groove Enterprise Integration Server Bots

Groove accomplishes this through the use of *bots*, which are automated agents that perform tasks on the part of members of a shared space. For example, a bot might handle a query to a CRM system, access an inventory record in an ERP system, publish a knowledge asset to a KM repository, or capture a record of customer interaction for audit purposes and store it in a records management system. The bot itself is managed by the Groove Enterprise Integration Server, software that resides on a dedicated machine within a peer network. Bots communicate with the back-end system using communication protocols exposed by that system, such as developer interfaces (COM, C/C++), SOAP requests and database queries via ADO and ODBC.

There are several important advantages to this approach:

- **Single point of administration.** For an organization with many employees interacting in numerous Groove shared spaces, the Groove Enterprise Integration Server limits access to enterprise systems to a single entity. Developers and end users can add bot functionality into a Groove shared space – programmatically or on-the-fly – by simply “inviting” a particular bot into the space itself. In this way, multiple instances of a tool in multiple shared spaces, are replaced by a single bot that participates in multiple shared spaces, much the same way that a single person is a member of multiple shared spaces. Therefore, when a bot is updated or deleted by an administrator, it is updated or deleted in all spaces that use it.
- **Centralized access control.** Access to the bot can be based on access control rules. The administrator or developer can indicate that only individuals with the appropriate access control authority or role may bring a bot into a shared space. Conversely, the bot itself can be programmed to determine the access control rights of each of the members, and to govern how it shares data and functionality accordingly. For example, a bot could allow only certain members of a shared space to use it to make queries against an ERP system.
- **Always on.** Using bots with the Enterprise Integration Server provides an “always on” solution for shared space integration. Designed to be administered by IT personnel for 24x7 availability, bots provide services that would be impossible to provide from a mobile workstation, such as full time or remote access to shared space data. As long as the Enterprise Integration Server is up and running, the bot can provide whatever service it was built for without waiting for members of a shared space to come online.
- **Cross-firewall.** Because the Enterprise Integration Server is based on the Groove platform, data stored in enterprise systems can be securely extended

to shared spaces using the data encryption and integrity protection services built into the platform.

- **Scalable.** The Enterprise Integration Server uses a “Just In Time” activation model, which means that shared spaces and bot components are only loaded into memory when they are needed, and are then automatically unloaded from memory over time. This efficient use of memory allows the Enterprise Integration Server to support many spaces that have a moderate level of activity without using a great deal of memory. In the future, the Enterprise Integration Server will also support clustering to allow the work to be distributed among multiple physical devices.
- **Tightly tied to the Groove platform.** When developing a bot, you are calling Groove platform APIs directly to interact with data. This requires that a developer fully understands the concepts in the Groove platform in order to write the bot, but also provides a tightly woven integration solution.
- **Work closely to where the Groove data is stored.** When using a bot for integration, the developer is able to use interfaces to manipulate Groove data in its native memory space without the need to create extra copies of the data. Not needing to copy the data also means that no additional latency is introduced when communicating between processes or over the network.
- **Supports transaction control.** Bot development atop the Groove platform also allows developers to leverage the Groove transaction model. The developer can then group a series of data updates or transactions together to maintain consistency
- **Has all space data.** Another advantage of integration using a bot and the Enterprise Integration Server is that all data within a Groove shared space is available.
- **Synchronous.** When using a bot for integration, events occur in both a synchronous and asynchronous manner. This means that a bot developer is able to structure code that is able to respond to data-related events as they happen (for data integrity enforcement) or at a later time (for user interface updates).

In addition to the many advantages of integration using bot development and the Groove Enterprise Integration Server, there are some trade-offs and issues to consider before pursuing this path, including:

- **Requirement for data transformation code.** Depending on the type of integration project undertaken, using bots as an integration method may require the effort of developing data transformation code. An example would be integrating data from the Groove Calendar tool into a calendar application from a different vendor. The developer needs to look at all the properties of the Groove calendar entry and write code that maps them to the corresponding fields in the third party calendar tool.
- **Learning curve.** As stated previously, in order to develop bots for use with the Enterprise Integration Server it is necessary for the developer to fully understand the concepts of Groove and to devote time to the effort of learning Groove bot writing.

- **Need for the Enterprise Integration server.** While the Enterprise Integration Server has scalability benefits, there are costs and administrative activities involved in installing and maintaining the server. These costs should be weighed against the size of the integration project.

Groove Web Services

Groove Web Services are a set of open desktop services, and supporting infrastructure, that extend the reach of Groove components and solution data to people, devices, and application or system environments that otherwise cannot gain access to Groove. These services provide programmatic access to Groove data and collaboration components using standard languages, protocols and formats, including XML, SOAP, and WSDL. These services turn Groove Workspace into a loosely coupled open service provider that can be consumed by any application, either locally (on the same device as Groove Workspace) or remotely (on some other network-connected device), regardless of programming language or device platform. The initial release of Groove Web Services provides access for the following Groove components:

- Shared space and tool services
- Account Services
- Contacts and shared space members
- Presence awareness
- Files, Discussion and Calendar tools

All of these services are addressable by a unique URI, and are callable by both local (i.e. same device as the Groove client) and remote programs. Local Groove Web Service access is defined as any application that is installed on and running on the same PC as Groove Workspace, which is the web services provider.

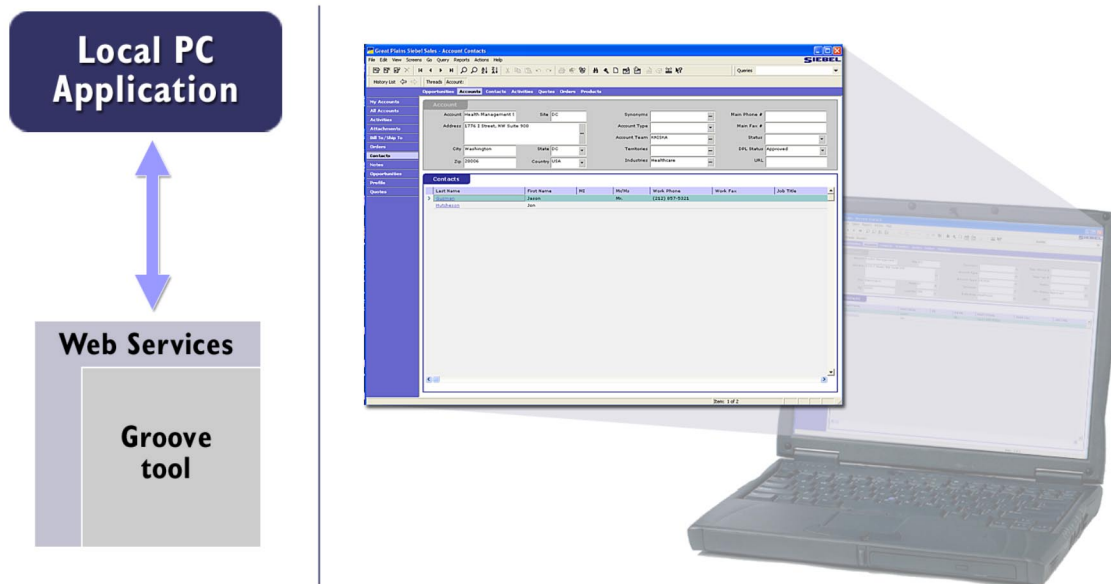


Figure 2 Groove Collaboration extended to a Local PC Application through Groove Web Services

An example of a local application that could take advantage of Groove Web Services would be integrating Groove online presence awareness with another application, such as the Business Process Application (BPA) example cited earlier. Another example would be integration with a corporate enterprise portal, where an end-user's web page is built using "web parts" from multiple applications, including Groove. The portal page could call Groove's local Web Services interface to make the user's Groove contacts and shared spaces viewable and accessible right from that page.

Groove Web Services can also be accessed from remote devices using the Groove Web Service Access Point (available only as a "technology preview" hosted service in v2.5). It includes a SOAP gateway that brokers a connection between Groove services and the SOAP client for remote requests. The Groove Web Service Access Point receives requests from programs on a network and routes them to Groove Web Services offered by the Groove client, which may be mobile, behind firewalls, etc. It routes the Groove Web Service response back to the remote SOAP client.

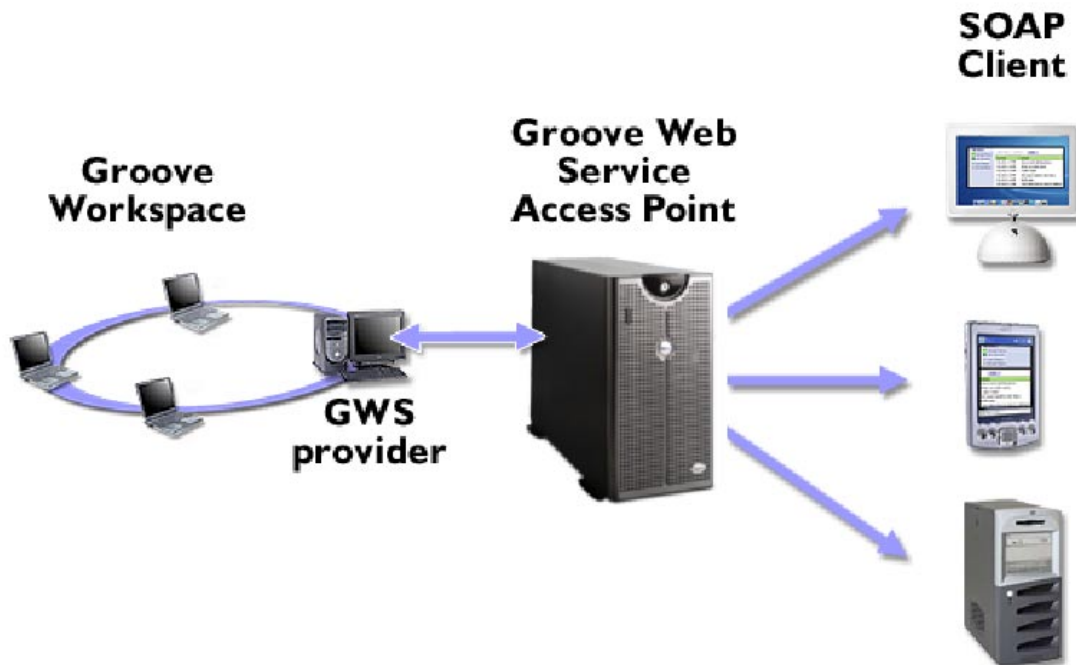


Figure 3 Groove collaboration services can be extended to remote SOAP clients through Groove Web Services

An example of remote access using Groove Web Services and the Web Service Access Point would be a mobile handheld device. A PocketPC device, or other PDA could be integrated to allow access to portions of a Groove shared space while away from the user's PC. Another example would be to integrate Groove data with a web services compliant back-end system.

The advantages of using Groove Web Services technology to solve integration challenges are as follows:

- **Standards based development.** Integrating with standard languages and protocols such as WSDL, XML, and SOAP provide a number of benefits. The learning curve for developers is smaller, as many are already familiar with

Web Services development. There are close to one hundred SOAP toolkits on the market, which focus on providing simple tools for building web services clients.

- **Interoperability.** Software code programmed to the Groove Web Services API can be used in a variety of environments, independent of the underlying hardware or operating system platform. As more and more systems become web services enabled, application integration will no longer be an all-or-nothing, one-to-one, custom development effort.
- **Platform Independence.** Web Services technology including the use of XML and SOAP is being widely adopted to many system platforms. Going forward this means that there is potential to integrate Groove shared spaces with applications on virtually any platform.
- **Extended Groove Access.** With the Groove Web Service Access Point, it becomes possible to extend remote access to Groove from virtually anywhere, including PDAs, Web Browsers, and non-Windows systems.
- **Single API for local or remote integration.** Using Groove Web Services for integration provides a single API to develop to for both local and remote application access. This provides opportunities for more reuse and flexibility of an integration solution.
- **“Bolt on” Web Services support.** The Groove Web Services framework allows third-party developers of custom Groove tools to turn those tools into web services that are consumable elsewhere. Thus, integration projects are not limited to those tools and components that Groove Networks provides as a web service.
- **Ease of data translation.** Using standards based web services provides for a higher degree of ease in data translation. Toolkits for web services, such as XSLT, support a high level of data translation capabilities. This saves you the time of coding the logic for data translation by hand by giving you an easy way to map one XML schema to another.
- **Use only the needed data.** For some integration scenarios, the application being integrated with does not need access to the entire Groove shared space, but only a portion of the shared space. With Groove Web Services, it is easy to provide only the subset of required access.
- **Non-programmer use.** With the Groove Web Service Access Point, it will become possible for end-users to “publish” web services from their Groove PC, for shared spaces and tools.

Tradeoffs and issues to keep in mind when integrating with Groove Web Services include:

Performance considerations. Using web services may have some negative performance effects. Code may run slower because all the data has to be marshaled to XML, which means turning everything into a string, storing a copy of it in memory, and then sending to the receiver, who must convert the strings back to the appropriate data type (integer, for example) and store somewhere in memory.

- **No transaction control.** Unlike integration with EIS, Groove Web Services is unable to provide tightly coupled transaction control services.
- **Asynchronous.** Transactions using Groove Web Services are done in an asynchronous manner; meaning code may need to be developed to take into account events that may have happened after a transaction was initiated. As an example consider the following. A SOAP client is subscribed to events for a Groove tool. The SOAP client gets an event indicating a new item was added, but when the SOAP client queries Groove Web Services for more information about the item, the item may have already been deleted. Receivers of such events must be prepared for such circumstances.

Custom Groove Connector Tools

A custom Groove connector tool is a Groove tool that is built to connect to a back-end server through COM or a publicly available API provided by the enterprise server vendor. An example of a Groove connector tool that was built by Groove Networks is the Mobile Workspace for Microsoft Sharepoint Team Services.

Some of the advantages of integrating using a customized connector tool are similar to those found when building a bot for use with the Enterprise Integration Server. Important advantages to integrating with a connector tool include:

- **Higher performance.** As with developing a bot for use with the Enterprise Integration Server, connector tool development provides the advantages of building a solution that is closely tied to the Groove platform. Working close to the data, the developer is able to use interfaces to manipulate Groove data in its native memory space without the need to create extra copies of the data. Not needing to copy the data also means that no additional latency is introduced when communicating between processes or over the network.
- **Has all space data.** Another advantage of connector tools is that they can access all data within a Groove shared space.
- **Synchronous.** When using a connector tool for integration, transactions occur in both a synchronous and asynchronous manner. This means that the developer is able to structure code that is able to respond to data-related events as they happen (for data integrity enforcement) or at a later time (for UI updates).
- **Authentication.** When a real end user needs to be authenticated for access to a back end system, a tool can prompt them for their user name and password.

Tradeoffs of Groove connector tool development to consider include:

- **Learning curve.** In order to develop a custom connector tool, it is necessary for the developer to fully understand the concepts of Groove and to devote time to the effort of learning Groove tool development.
- **Not "always on."** A Groove connector tool will only be turned on when a user's device is up and running. Once the user shuts down Groove or turns the device off, the service is not available. In the case of the Mobile Workspace for Microsoft Sharepoint Team Services connector, changes to a

Groove shared space cannot be synchronized with Sharepoint until the creator of the space reconnects his or her device to the network.

- **Requirement for data transformation code.** Like bot development using a Groove connector tool for integration requires developing data transformation code.
- **Multiple points of synchronization with external data.** When writing a connector tool, if every device within a shared space were to sync with the same data source, you would end up with multiple copies of the same data. With the connector tool, you need to add the additional logic to work around this.

Using Enterprise Integration Server together with Groove Web Services

In a future release of Groove, it will be possible to use Groove Web Services integration solutions in conjunction with bots and the Enterprise Integration Server. The benefit will be to provide "always on" access to Groove Web Services. This will provide end-users anytime access Groove Web Services, regardless of whether Groove users are online, and logged in or not. Further, from a systems integration perspective, it will allow integrators to write their integration code in their Enterprise Application Integration (EAI) system of choice and have that code talk to a generic bot service at the Enterprise Integration Server that exposes itself as a web service; that is, for some integration projects it can preclude the need to do custom bot development.

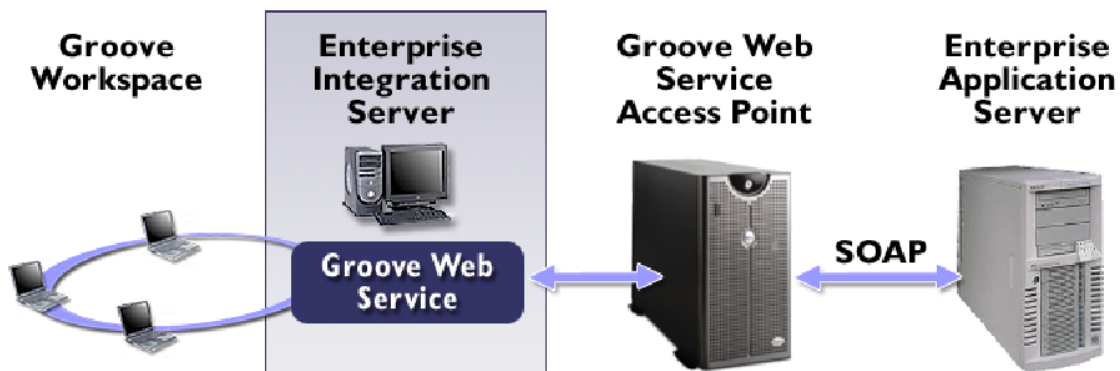


Figure 4 Using Groove Web Services together with Groove Enterprise Integration Server

Availability

Services Available in Groove v2.5

Groove Platform Development Kit

The GDK for Groove is available immediately, and can be downloaded for free from www.groove.net/devzone.

Groove Development Kit for Enterprise Integration Server

The GDK for Enterprise Integration Server is available immediately. It is included with the Enterprise Integration Server product shipment.

Groove Enterprise Integration Server

The Groove Enterprise Integration Server is available immediately. Contact your Groove sales representative for purchase options and pricing information.

Groove Development Kit for Groove Web Services

The GDK for Web Services is available with Groove v2.5. It provides for local programmatic access to Groove services in a secure fashion.

Groove Web Service Access Point Preview

A hosted technology preview of the Groove Web Service Access Point is also available with v2.5. The hosted Web Service Access Point *preview* provides remote programmatic access to Groove services in a non-secure fashion, and is not intended for any production development or deployment.

Future Services

Groove Web Service Access Point

The Groove Web Service Access Point server product will be released in 2003.

Enterprise Integration Server & Web Services Integration

Integration of the Enterprise Integration Server and Groove Web Services is on the Groove product roadmap, but a delivery date for this functionality is not yet publicly available. Contact Groove Networks for further information.

Summary

From a developer's perspective, the various Groove integration options represent a development framework with which to transform various systems, applications, and devices into high-value business solutions. Each method within the framework, including Groove Web Services, Enterprise Integration Server, and customized connector tool possesses both advantages and tradeoffs for specific integration efforts. For companies interested in Groove, both as solution platform and core collaborative infrastructure, the integration framework represents not only a tool, but also a necessary part of that infrastructure.

Other Resources

Find additional white papers, documentation, and resources for Groove development in the Groove DevZone at <http://www.groove.net/devzone>.