

# Grouper Book - What is Grouper and what can it do?

<a href="#">Wiki Home</a>	<a href="#">Grouper Release Announcements</a>	<a href="#">Grouper Guides</a>	<a href="#">Grouper Deployment Guide</a>	<a href="#">Community Contributions</a>	<a href="#">Internal Developer Resources</a>
---------------------------	---	--------------------------------	--	---	--

## Section contents

[Part 1 - Introductory Information](#)  
[What is Grouper and what can it do?](#)  
[SQL database](#)  
[Grouper API](#)  
[Web Service](#)  
[Web Interface](#)  
[Provisioning](#)

## Part 1 - Introductory Information

### What is Grouper and what can it do?

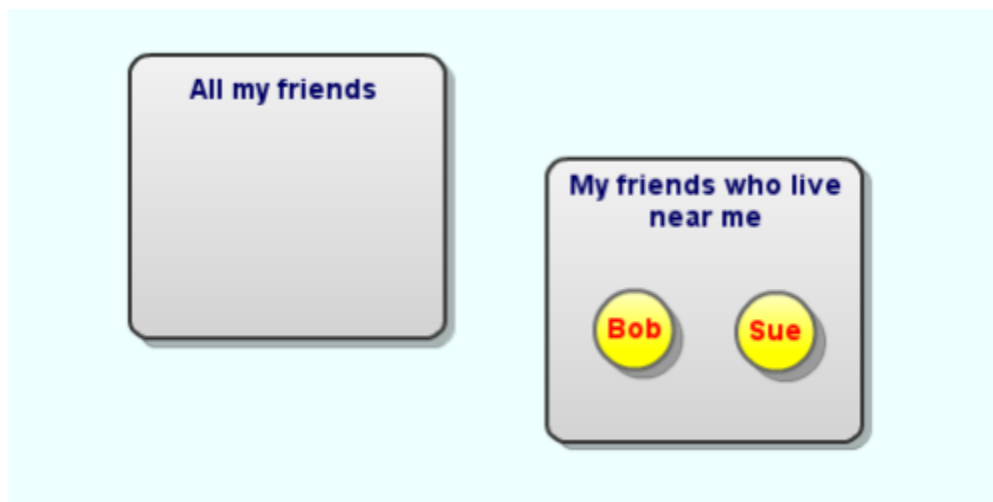
So, let's get started with a few fundamental questions.

Grouper is, at its simplest, a toolkit to put entities into groups. These entities could be people, computers or other assets, but in most implementations the groups made up of people. The groups can be of any size, with memberships from one to tens of thousands or more. Grouping entities together is an essential part of a process of simplification which helps to make sense of complex environments. For example, many people create groups in their personal email clients so that they can quickly send mail to a defined set of individual email addresses without having to worry about mis-sending the email to a wrong person.

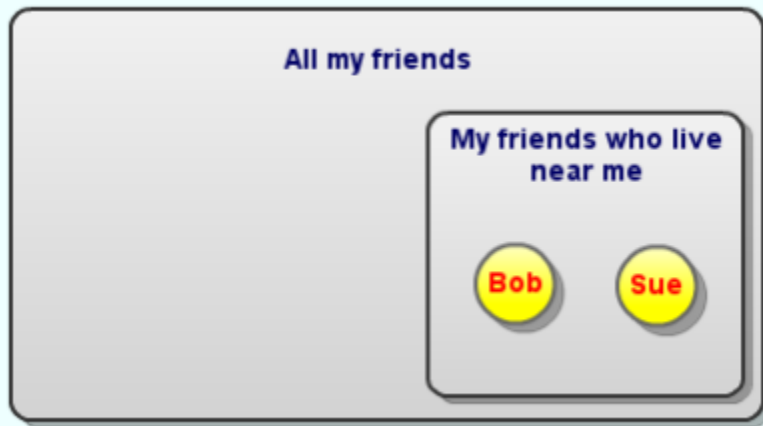
Typically all members of a group share one or more common characteristics, which means it makes sense to group them together. These characteristics can be defined corporately (members of an academic school), or personally (my friends with a good sense of humour). Typically a group that is defined corporately will make sense and be of use to a larger number of people than one defined personally. This is because the characteristics are a product of the environment, and the more common the environment the more widely the group will be of use.

Grouper can be used to manage groups in small, even personal, environments as well as large, common environments. Within Grouper, different groups can be made visible only to people to whom they will make sense and be of use, but all will be contained in central group registry. Having a single registry makes it far easier to manage how the groups interact with other systems than if there were multiple, fragmented registries (such as in many individual mail clients). It also means that disconnected groups in multiple locations no longer need to exist, and membership only needs to be managed in one place.

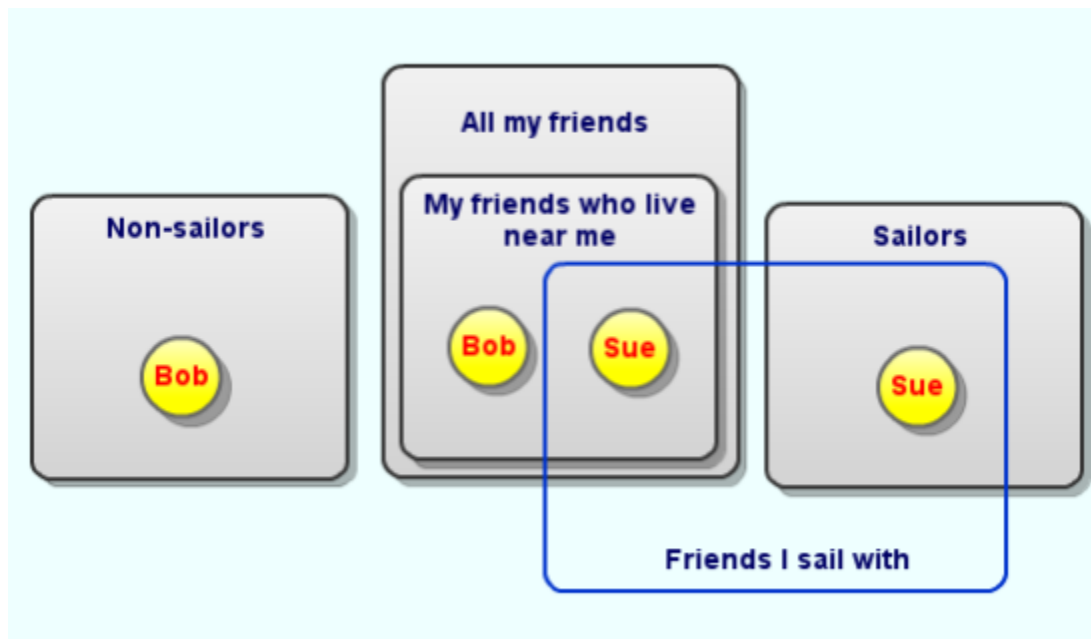
Many applications and directories offer facilities to create and manage groups, and these groups are normally simple groups, where entities have to be directly made members of a group. Some go further than this, and allow groups to be added as members of other groups (known as nesting). For example, imagine that there are 2 groups: "All My Friends" and "My friends who live near me;" Bob and Sue are direct members of "My friends who live near me," "All My Friends" has no direct members.



Now, if I add the group "My friends who live near me" as a member to "All My Friends," Bob and Sue immediately appear as members of "All My Friends," which they have inherited via their membership of "My friends who live near me." "My friends who live near me" is "nested" inside "All My Friends."



Grouper supports "nested" groups and more; using what is termed Group Math, Grouper can define Groups by including or excluding members from any number of other groups. So a "Friends I sail with" group could be defined as including those members of "All my friends" who are also members of "Sailors," but not members of "Non-sailors."



This encourages groups to be managed efficiently and effectively, avoiding the common problem of Group sprawl that occurs when existing groups do not quite fit a requirement, resulting in the creation of a new group and assigning entities direct membership in it. Creating a new self-contained Group will always create a larger administrative and management load than defining a group in terms of other groups that already exist.

Grouper stores data about groups and membership of those groups in an SQL database. An application programming interface (API) is provided to interact with this data in order to add, delete, query or modify the data. A web interface is also provided as an easy way to view and manipulate data. Grouper allows administrators to give users rights to manage certain groups, thus allowing for delegated management and putting the power to manage groups in the hands of the people who are best placed to manage them. Various methods are also provided for provisioning groups from Grouper to other systems, including LDAP directories. Advanced features such as role and permissions management are also included as part of Grouper. In short, Grouper is a richly-featured group management toolkit.

Grouper can get information about the entities who can be members of groups (termed "subjects" in Grouper terminology), either from its own database, or from an existing external database or LDAP directory.

## SQL database

Grouper stores all its data in the SQL database, the following are tested with each new release:

- MySQL (Small installs only)
- Oracle
- PostgreSQL (Recommended)

Other databases supported by hibernate may work, but are not tested by the Grouper development team.

## Grouper API

The API is a set of Java classes that provide functionality to manipulate data within the database. For example, methods exist to create groups and modify their membership. These classes are distributed as a library which can be called from Java code.

## Web Service

Most operations can be achieved via the Web Service (WS) interface. Some high level functionality is not available via the WS, for example configuration changes, Audit logs, and Instrumentation. However, many features can be utilized by constructing web service calls that set attributes on groups. More information on available WS features are available [here](#).

## Web Interface

A full-function administrative web interface is available which enables you to view and modify Grouper data in a browser configure the system, set up data sources and destinations, etc. It uses the API, but no programming is necessary to use it. It is written in Java and runs in an application server, such as Apache Tomcat. .

## Provisioning

Functionality is built into Grouper that makes it relatively easy to provision groups created and managed by Grouper into external data stores and applications. This enables you to keep groups synchronized across different stores and removed the need for all applications to interface directly with Grouper to consume groups.