Concepts

GenePattern provides access to a broad array of computational methods used to analyze genomic data. Its extendable architecture makes it easy for computational biologists to add analysis and visualization modules, which ensures that GenePattern users have access to new methods on a regular basis.

This **Concepts** guide provides a brief introduction to GenePattern. All other GenePattern documentation assumes that you are familiar with the concepts covered here.

Analysis and Visualization Modules	Analyze data using GenePattern modules.
Pipelines	Combine modules to form analysis pipelines.
Suites	Organize modules and pipelines into suites.
Servers	The GenePattern user interface communicates with a GenePatter server. The server runs the analyses and stores the results.
Jobs	Create a job on the server by running an analysis or pipeline.
Security and Permissions	A GenePattern administrator determines who has what access to the GenePattern server.
Version Numbers	GenePattern ensures reproducible analysis results by uniquely identifying every version of every module and pipeline.
Programming Environments	GenePattern can be accessed directly from Java, MATLAB, or R.

Analysis and Visualization Modules

Analysis and visualization modules are at the heart of GenePattern:

- **Analysis modules** provide computational methods and tools for analysis of genomic data. Analysis modules are run on the GenePattern server.
- **Visualization modules** display analysis results graphically and allow you to manipulate that view interactively. By convention, these modules have "Viewer" in the name. Because visualization modules are interactive, they run on your desktop computer. **Note**: You must have Java 1.5 installed on your local computer to use the visualization modules.

Each module includes its own documentation, which is supplied by the module developer. The Modules page of the GenePattern web site lists the modules available from the Broad Institute with links to their documentation.

Pipelines

Pipelines combine analysis modules, visualization modules, and other pipelines into a single, reusable workflow. Pipelines can be defined to analyze a particular dataset; for example, you might create a pipeline to reproduce published analysis results. Or they can be parameterized, which allows the person running the pipeline to provide datasets and other analysis variables. Often a pipeline runs a progressive series of analyses, where the output from one analysis is used as input for the next.

When you create a pipeline, you select the modules (and pipelines) to be executed by the pipeline. Most modules require one or more parameters. You can specify the parameter values when you create the pipeline, have the pipeline use the output file from one module as the input parameter value for a subsequent module, or prompt the user for parameter values when the pipeline is run.

Pipelines can be used to share analysis methods or to document research. By providing a way to create and distribute an entire computational analysis methodology in a single executable script, pipelines enable a form of *in silico* reproducible research. Colleagues with access to the same GenePattern server can easily share pipelines. Alternatively, a pipeline can be exported from one GenePattern server and imported into another.

The repository maintained by the Broad Institute includes a number of pipelines that document analysis methodologies published by Broad researchers. The Modules page of the GenePattern web site lists the pipelines available from the Broad Institute with links to their documentation.

Suites

Suites group modules and pipelines into convenient packages. For example, if you tend to analyze copy number data, you might find it helpful to create a suite that includes the SNPFileCreator, GISTIC, and other related modules. Suites provide easy access to frequently accessed modules. They also provide a convenient way of collecting a set of modules and pipelines to be shared with other GenePattern users. Colleagues with access to the same GenePattern server can easily share suites. Alternatively, a suite can be exported from one GenePattern server and imported into another.

The repository maintained by the Broad Institute includes a number of suites. The Suites page of the GenePattern web site lists them.

Servers

To use GenePattern, you open a web browser and enter a URL. The URL that you enter is the address of a GenePattern server. The web browser provides the user interface. The server runs the analyses and stores the results.

You can use the GenePattern server hosted at the Broad Institute or download and install the GenePattern software. The server hosted at the Broad Institute is called the *public server* or the *Broad-hosted server*. All other GenePattern servers are known as *local servers*.

Broad-Hosted Server

The Broad Institute hosts a publicly available GenePattern server at http://genepattern.broadinstitute.org/gp/. You can use the Broad-hosted GenePattern server without installing any software.

Using the Broad-hosted server has several benefits:

- The GenePattern team maintains the server for you.
- User accounts are freely available. Just open a web browser and go to http://genepattern.broadinstitute.org/gp/. On the login page, select the *Click to register link*.
- Most of the modules and pipelines available from the Broad Institute (see the Modules page of the GenePattern web site) are available on the Broad-hosted server. Several modules are available only on the Broad-hosted server because they require customized server configuration. A small number of modules are not available on the Broad-hosted server because they run only on the Windows platform; the Broad-hosted server runs under Unix.

Local Server

When you download GenePattern, you install a *local GenePattern server*. You can install a local server on a standalone machine for your personal use or on a networked machine for use by several people or an entire organization. A local GenePattern server shared by several users is sometimes called a *networked GenePattern server*. Instructions for installing a local GenePattern server are provided on the Download GenePattern page.

Using a local server has several benefits:

- You choose which modules and pipelines to install on the server. Modules and pipelines can come from several sources. (1) The Broad Institute hosts a repository that contains more than 100 modules. (2) Other GenePattern users can export their modules and pipelines to ZIP files, which you can then install. (3) GenePattern users create their own modules and pipelines. Only the GenePattern team can create or install modules on the Broad-hosted server.
- You can analyze your data without sending it over the internet. Your analyses are run and the results stored on the local GenePattern server.
- As the server administrator, you control the server configuration. For example, initially a local server does not have password protection. You can easily add password protection by modifying the server configuration.
- You can configure the GenePattern server to be used by researchers across your organization. This makes it easy for lab groups to use GenePattern to automate their analysis pipelines. It also allows researchers to share new analysis methods simply by adding them to GenePattern as new modules or pipelines.

Jobs

When you run a module or pipeline in GenePattern, the web browser sends your request to the GenePattern server. The server starts a job to run the analysis. Job results (analysis result files and execution logs) are stored on the GenePattern server for a period of time (by default, one week) and then deleted. The GenePattern home page displays your most recent jobs and the Job Result Summary page displays all of your jobs.

Every job run on the GenePattern server is owned by person who submitted the job. Owners are identified by their GenePattern usernames. Every job is persistent, which means:

- Once you start a job, it continues to run even if you exit from GenePattern.
- If the GenePattern server is shut down or interrupted while executing a job, when you restart the server the server automatically restarts the interrupted job.

Security and Permissions

GenePattern provides a flexible architecture that allows a user with server administrator privilege to control access to the server in several ways:

- Access filtering defines which computers (identified by IP address) have access to the GenePattern server.
- User authentication defines who can log into the GenePattern server. For example, the server might require a username (the default) or a username and password depending on how it is configured.
- User permissions define which operations the user can perform. For example, a user might have permission to create private pipelines, but

not public pipelines.

- Users can be assigned to groups. Groups serve two functions: (1) permissions are assigned to groups and (2) members of a group can share job result files with other members of the group.

GenePattern servers are generally configured to distinguish between users and administrators. The following table shows the permissions used on the Broad-hosted server and the default permissions for a local server. GenePattern adjusts its user interface based on the permissions assigned to the person logged in; for example, only administrators see the Administration menu. The GenePattern documentation describes all of the GenePattern features. Your permissions determine whether a particular feature is visible.

Server	User Permissions	Administrator Permissions
Broad-hosted server	Run public modules/pipelines Create and run your own pipelines Edit/delete your jobs and pipelines	GenePattern team has all permissions GenePattern team can view/delete all jobs, modules, and pipelines
Local server, standalone	Same as administrator permissions	All users have all permissions All users can view/delete all jobs, modules, and pipelines
Local server, shared*	Run public modules/pipelines Create and run your own pipelines Create and run your own modules Create public pipelines Edit/delete your jobs, modules, and pipelines	All permissions View/delete all jobs, modules, and pipelines

* When several users share a local server, the system administrator typically secures the server by assigning only a few users to the Administrators group. When a local server has designated administrators, users and administrators have the default permissions shown here.

For more information about security and permissions, see Securing the Server in the User Guide.

Version Numbers

GenePattern uses version numbers to uniquely identify modules and pipelines. When you create an object, GenePattern automatically assigns the object a version number of one (1). When you update an object, GenePattern automatically updates the object's version number. By carefully versioning each object, GenePattern ensures that you can accurately reproduce analysis results.

For example, you might create a pipeline that runs two modules: PreprocessDataset version 4 and HierarchicalClustering version 5. If the

HierarchicalClustering module is updated (creating HierarchicalClustering version 6), version 1 of your pipeline still runs HierarchicalClustering version 5; thus, ensuring that the pipeline produces the same results each time it is run. However, depending on why you are using the pipeline, you might prefer to have the pipeline run the latest version of an analysis module rather than a specific version. You make that choice when you create (or edit) the pipeline. For example, you might update the pipeline (creating version 2) to have the pipeline always use the latest version of the HierarchicalClustering module. Now, when you run version 2 of the pipeline it uses HierarchicalClustering version 6. When you run version 1 of the pipeline, it uses HierarchicalClustering version 5.

When you view and edit modules or pipelines, GenePattern shows you their version numbers. Typically, you update the latest version of an object, which increments its version number. For example, editing version 1 creates version 2. At times, you may need to edit an older version, which creates a *point version*. For example, if you have versions 1 and 2, editing version 1 creates version 1.1.

GenePattern implements version numbers using Life Science Identifiers (LSIDs). Thus, object identifers in GenePatterns are sometimes called LSIDs.

Programming Environments

A programmatic interface makes it easy for software programmers to call GenePattern modules from the Java, MATLAB, or R programming environments. For information about the programmatic interface, see the Programmer's Guide.