

# 1 gitinspector

gitinspector — statistical analysis tool for git repositories

## Synopsis

**gitinspector** [OPTION]... [REPOSITORY]

## DESCRIPTION

Analyze and gather statistics about a git repository. The default analysis shows general statistics per author, which can be complemented with a timeline analysis that shows the workload and activity of each author. Under normal operation, gitinspector filters the results to only show statistics about a number of given extensions and by default only includes source files in the statistical analysis.

Several output formats are supported, including plain text, HTML and XML.

## OPTIONS

List information about the repository in **REPOSITORY**. If no repository is specified, the current directory is used. If multiple repositories are given, information will be fetched from the last repository specified.

Mandatory arguments to long options are mandatory for short options too. Boolean arguments can only be given to long options.

### **-f, --file-types=EXTENSIONS**

A comma separated list of file extensions to include when computing statistics. The default extensions used are: java,c,cc,cpp,h,hh,\*. Specifying a single \* asterisk character includes files with no extension. Specifying two consecutive \*\* asterisk characters includes all files regardless of extension.

### **-F, --format=FORMAT**

Defines in which format output should be generated; the default format is *text* and the available formats are: html,htmlembedded,txt (see [OUTPUT FORMATS](#))

### **--grading[=BOOL]**

Show statistics and information in a way that is formatted for grading of student projects; this is the same as supplying the options **-HlmrTw**

### **-H, --hard[=BOOL]**

Track rows and look for duplicates harder; this can be quite slow with big repositories

### **-l, --list-file-types[=BOOL]**

List all the file extensions available in the current branch of the repository

### **-L, --localize-output[=BOOL]**

Localize the generated output to the selected system language if a translation is available

### **-m, --metrics[=BOOL]**

Include checks for certain metrics during the analysis of commits

### **-r --responsibilities[=BOOL]**

Show which files the different authors seem most responsible for

### **--since=DATE**

Only show statistics for commits more recent than a specific date

### **-T, --timeline[=BOOL]**

Show commit timeline, including author names

**--until=DATE**

Only show statistics for commits older than a specific date

**-w, --weeks[=BOOL]**

Show all statistical information in weeks instead of in months

**-x, --exclude=PATTERN**

An exclusion pattern describing the file paths, revisions, author names or author emails that should be excluded from the statistics; can be specified multiple times (see [FILTERING](#))

**-h, --help**

Display help and exit

**--version**

Output version information and exit

## OUTPUT FORMATS

There are support for multiple output formats in gitinspector. They can be selected using the **-F/--format** flags when running the main gitinspector script.

**text (plain text)**

Plain text with some very simple ANSI formatting, suitable for console output. This is the format chosen by default by gitinspector.

**html**

HTML with external links. The generated HTML page links to some external resources; such as the JavaScript library JQuery. It requires an active internet connection to properly function. This output format will most likely also link to additional external resources in the future.

**htmlembdedded**

HTML with no external links. Similar to the HTML output format, but requires no active internet connection. As a consequence; the generated pages are bigger (as certain scripts have to be embedded into the generated output).

**xml**

XML suitable for machine consumption. If you want to parse the output generated by gitinspector in a script or application of your own; this is the format you should choose.

## FILTERING

gitinspector offers several different ways of filtering out unwanted information from the generated statistics:

- **gitinspector -x myfile**, filter out and exclude statistics from all files (or paths) with the string "myfile"
- **gitinspector -x file:myfile**, filter out and exclude statistics from all files (or paths) with the string "myfile"
- **gitinspector -x author:John**, filter out and exclude statistics from all authors containing the string "John"
- **gitinspector -x email:@gmail.com**, filter out and exclude statistics from all authors with a gmail account
- **gitinspector -x revision:8755fb33**, filter out and exclude statistics from all revisions containing the hash "8755fb33"
- **gitinspector -x message:BUGFIX**, filter out and exclude statistics from all revisions containing "BUGFIX" in the commit message.

The gitinspector command also lets you add multiple filtering rules by simply specifying the **-x** options several times or by separating each filtering rule with a comma;

- **gitinspector -x author:John -x email:@gmail.com**

- `gitinspector -x author:John,email:@gmail.com`

Sometimes, sub-string matching (as described above) is simply not enough. Therefore, gitinspector let's you specify regular expressions as filtering rules. This makes filtering much more flexible:

- `gitinspector -x "author:^(?!John Smith)"`, only show statistics from author "John Smith"
- `gitinspector -x "author:^(?![A-C])"`, only show statistics from authors starting with the letters A/B/C
- `gitinspector -x "email:.com$"`, filter out statistics from all email addresses ending with ".com"

## USING GIT TO CONFIGURE GITINSPECTOR

Options in gitinspector can be set using **git config**. Consequently, it is possible to configure gitinspector behavior globally (in all git repositories) or locally (in a specific git repository). It also means that settings will be permanently stored. All the long options that can be given to gitinspector can also be configure via git config (and take the same arguments).

To configure how gitinspector should behave in all git repositories, execute the following git command:

**git config --global inspector.option setting**

To configure how gitinspector should behave in a specific git repository, execute the following git command (with the current directory standing inside the repository in question):

**git config inspector.option setting**

## AUTHOR

Originally written by Adam Waldenberg.

## REPORTING BUGS

Report gitinspector bugs to [gitinspector@ejwa.se](mailto:gitinspector@ejwa.se)

The gitinspector project page: <https://github.com/ejwa/gitinspector>

If you encounter problems, be sure to read the FAQ first: <https://github.com/ejwa/gitinspector/wiki/FAQ>

There is also an issue tracker at: <https://github.com/ejwa/gitinspector/issues>

## COPYRIGHT

Copyright © 2012-2015 Ejwa Software. All rights reserved. License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>. This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

## SEE ALSO

`git(1)`