# flowFlowJo

April 20, 2011

---

`collectSummaryFlowInfo`
*Collect Population Statistics on Flow Files*

---

## Description

Creates a data structure containing a comprehensive list of summary statistics and related information on the cell populations of the FCS files referenced in one or more FlowJo workspaces.

## Usage

```
collectSummaryFlowInfo(fj, ...)
```

## Arguments

| | |
|---|---|
| `fj` | a flowJoList object. See readFlowJoList. |
| `...` | Additional arguments detailed below: |

## Details

Additional arguments also include the following:

```
fileNamePatterns - a vector of patterns that will be used to pick only specif
gateNamePattern  - a vector of patterns that will be used to pick only a subs
upperCaseGates   - Logical.  Default = FALSE.  If TRUE, converts all gate nam
keywords         - a vector of fields that you want to retrieve from the text
stripGutters     - Logical.  Currently FALSE and non-functional.  Whether or
method           - what type of measurement to use for MFIs on each populatio
```

## Value

The method collectSummaryFlowInfo returns a moderately complex data structure called an *fcsSummaryList* detailing summary statistics and other information from the cell populations listed in one or more FlowJo workspaces. The main argument for this method is a flowJoList object that was created using the readFlowJoList method. The key issue here is that parsing a set of FlowJo workspaces and converting the referenced gates into flowCore style filters takes only a few seconds or minutes. Reading in each of the referenced FCS files in a very large experiment and gating them

with the filters (which this code does), may take many hours. Thus this method also includes two arguments (*fileNamePatterns* and *gateNamePatterns*) to reduce the number of FCS files examined. This method may, optionally, search the text/header section of each FCS file looking for specific keywords (e.g. $P2R, or $DATE, etc.) and collect the appropriate values. Finally, one of the main operations of this method is to collect the MFIs cell counts for each population in each FCS file. The user can choose either mean, median, or mode for the MFI values. Returned values are in untransformed units (e.g. not log intensities).

## Author(s)

John Gosink

## References

See also FlowJo from TreeStar Inc, at: http://www.flowjo.com/

## Examples

```
library(flowFlowJo);
demoLocation <- system.file("extdata", "DemoWorkspace.wsp", package="flowFlowJo");
actualFCSLoc <- system.file("extdata/fcsFiles", package="flowFlowJo");
testList     <- readFlowJoList(demoLocation, altFileLocation=actualFCSLoc);

# This next statement may take a few moments to run as it is parsing through a FlowJo wor
# and two dozen FCS files...
summaryStatsObj <- collectSummaryFlowInfo(testList);
```

---

`createFlowReport` *Produce a denormalized data frame of summary statistics and other information on a set of FlowJo gated FCS files*

---

## Description

This method produces a data frame that has one row per cell population per FCS file per channel for one or more FCS files as gated using the commercial software, FlowJo.

## Usage

```
createFlowReport(summaryList, ...)
```

## Arguments

| | |
|---|---|
| `summaryList` | a flowFlowJo object that is a list of data structures providing many bits of summary information about each gated population in a gated FCS file. See also *collectSummaryFlowInfo* |
| `...` | Additional factors detailed below. |

### Details

Additional arguments also include the following:

```
factorsFrame    - an additional data frame with various types of meta data r
populations     - a list of cell population names to be reported on.  By def
channels        - a list of scatter and non-scatter (fluorescent) channels t
includeKeywords - a boolean with the default = TRUE.  If FALSE, then none of
joinFactors     - if the user supplies an optional data frame with additiona
dropJoinFactors - a boolean with default = TRUE.  If set to FALSE, then the
```

### Value

The *collectSummaryFlowInfo* method produces a large and complicated data structure that tracks every FCS file location, gate structure, compensation matrix, as well as the summary statistics for every file referenced in a FlowJo workspace. This method, *createFlowReport* distills out a simple data frame of the summary statistics that is convenient and easy to read, plot, and otherwise analyze.

### Author(s)

John Gosink

### Examples

```
library(flowFlowJo);
demoLocation <- system.file("extdata", "DemoWorkspace.wsp", package="flowFlowJo");
actualFCSLoc <- system.file("extdata/fcsFiles", package="flowFlowJo");
testList     <- readFlowJoList(demoLocation, altFileLocation=actualFCSLoc);

# This next statement may take a few moments to run as it is parsing through a FlowJo wor
# and two dozen FCS files...
summaryStatsObj <- collectSummaryFlowInfo(testList);

# And here are some of the summary statistics for this workspace
flowReport      <- createFlowReport(summaryStatsObj);
head(flowReport);
```

---

## flowFlowJo-package *Turn FlowJo workspaces into flowCore compliant objects*

---

### Description

Extract information, gates, FCS file (locations) etc. from a FlowJo workspace

### Details

| | |
|---|---|
| Package: | flowFlowJo |
| Type: | Package |
| Version: | 1.16 |
| Date: | 2008-09-25 |
| License: | GPL 3 or newer |

FlowJo is a commercially available software package used in gating and analyzing data from flow cytometry experiments. Upon saving, FlowJo produces an XML format file that contains all the information necessary to describe the gating structures, compensation, transformation, location of the FCS files, and opened graphs and figures last created by the user. This package (flowFlowJo) is a small set of methods designed to extract the file locations, gates, compensation matrices and other meta-data contained in FlowJo workspaces and return the information in a manner consistent with the BioConductor flowCore packages.

## Note

See a complete discussion with examples via: *vignette("flowFlowJo")*

## Author(s)

John Gosink <gosinkj@amgen.com>

## Examples

```
# A demo set of FCS files and attendant FlowJo workspace are included with
# this package.  This extracts the info from the workspace.

library(flowFlowJo);
demoLocation <- system.file("extdata", "DemoWorkspace.wsp", package="flowFlowJo");
actualFCSLoc <- system.file("extdata/fcsFiles", package="flowFlowJo");
testList     <- readFlowJoList(demoLocation, altFileLocation=actualFCSLoc);
z            <- getFlowJoGates(testList, fileNamePatterns=c("C02"));

# Take a look at z.  It contains the name(s), location(s), compensation matrix, gate/filt
# and gate/filter names for one of the FCS files referenced in the workspace.
```

---

# flowJoCompensate-methods

*Compensate flow cytometry data the same way as FlowJo does*

---

## Description

FlowJo currently implements its compensation/spillover matrices differently than they are implemented in the general flow cytometry community. Currently, in order to obtain similar results (e.g. MFIs and cell counts) between FlowJo and flowCore, it is necessary to apply the compensation matrix to the data in the usual way (ie. via "compensate"), and then to divide all of the observed data by the maximum of the values in the compensation matrix. The flowFlowJo package implements a method, flowJoCompensate, to automatically take care of this issue.

## Methods

**flowData = "flowFrame", compMatrix = "matrix"** flowJoCompensate can take in either a flowFrame or a flowSet as well as a numerical matrix of compensation values.

**flowData = "flowSet", compMatrix = "matrix"** flowJoCompensate can take in either a flowFrame or a flowSet as well as a numerical matrix of compensation values.

getFlowJoGates *getFlowJoGates*

## Description

Given a parsed FlowJo workspace (flowJoObj) or list of FlowJo workspaces (flowJoList), return an indexed list of the gates, as flowCore style filter objects, for each of the FCS files. Also get the associated compensation matrices.

## Usage

```
getFlowJoGates(fj, fileNamePatterns)
```

## Arguments

| | |
|---|---|
| `fj` | a flowJoList or flowJoObj |
| `fileNamePatterns` | a vector of one or more patterns used to pick specific FCS files from the complete list of FCS files (with their paths) listed in the FlowJo workspace(s). The default is "." – ie all of the FCS files will be chosen. |

## Value

The method getFlowJoGates returns a list of ordered lists which includes references to a set of FCS files, their locations, all gating structures related to them (as flowCore filter objects), the names of those gates (filters), and the associated compensation matrices for the FCS files.

## Author(s)

John Gosink

## See Also

See also FlowJo from TreeStar Inc, at: http://www.flowjo.com/

## Examples

```
# Note this may take a moment to process the whole XML file
library(flowFlowJo);
demoLocation <- system.file("extdata", "DemoWorkspace.wsp", package="flowFlowJo");
actualFCSLoc <- system.file("extdata/fcsFiles", package="flowFlowJo");
testList     <- readFlowJoList(demoLocation, altFileLocation=actualFCSLoc);
z            <- getFlowJoGates(testList, fileNamePatterns=c("A01"));
```

`getFlowJoSummary` *Summarize FlowJo Workspaces*

## Description

Makes a table of the number and count of various gates, possibly broken out by FCS file in one or more FlowJo workspaces.

## Usage

```
getFlowJoSummary(x, gatesByFile=TRUE, fileNamePatterns=".", ...)
```

## Arguments

| | |
|---|---|
| `x` | a flowJoList or flowJoObj |
| `gatesByFile` | a boolean. If TRUE (default) then the count of each gate type will be broken out for each referenced FCS file. Otherwise this method will just return a table with the total counts of each named gate listed in the FlowJo workspace(s). |
| `fileNamePatterns` | A list of patterns for picking out specific FCS files. Default = "." (ie. examines all of the FCS files) |
| `...` | Additional arguments: to simplifyGateNames such as stripFileName and removeParentalNames |

## Value

The method getFlowJoSummary returns a table summarizing the number of gates listed in the referenced FlowJo workspaces. This function is useful for ensuring that the cytometrist has used a consistent set of names in their process.

## Author(s)

John Gosink

## References

See also FlowJo from Treestar Inc, at: http://www.flowjo.com/

## Examples

```
demoLocation <- system.file("extdata", "DemoWorkspace.wsp", package="flowFlowJo");
actualFCSLoc <- system.file("extdata/fcsFiles", package="flowFlowJo");
testList     <- readFlowJoList(demoLocation, altFileLocation=actualFCSLoc);

getFlowJoSummary(testList, gatesByFile=FALSE, removeParentalNames=TRUE);
```

`readFlowJoList` *Parse one or more FlowJo workspaces*

## Description

Gather up and organize all of the gates, transformations, and compensation matrices in a FlowJo workspace or list of FlowJo workspaces.

## Usage

```
readFlowJoList(workspaceVec, altFileLocation)
```

## Arguments

| | |
|---|---|
| `workspaceVec` | a list (or vector or single character string) of the path to FlowJo workspace(s) one wants to parse |
| `altFileLocation` | an alternate location to the FCS files may be supplied. See details below |

## Details

FlowJo is a commercial flow cytometry application from the company TreeStar, Inc. The 'save' format for FlowJo is the FlowJo workspace. A FlowJo workspace is an XML document that specifies all of the information necessary to recapitulate the exact user experience in the program at the time of the save. This files includes information on the location of all the relevant FCS files, their grouping, transformation, compensation, gating, and graphical displays last used by the researcher.

In some cases a FlowJo workspace and the associated FCS files need to be moved on the file system. If this is done the paths to the FCS files as embedded in the workspace will be stale (wrong). The *altFileLocation* argument allows the user to supply an alternate path to the FCS files.

The readFlowJoList method parses one or more FlowJo workspaces collecting all of the gate structures designed by the user. These gating structures are converted to flowCore style filter objects. Furthermore, because FlowJo users are often interested in not only the results of the final (i.e. terminal tip) gate, but also the results of many of the intermediate gates, readFlowJoList collects together all possible ordered subgates and chains them together as intersect-filter objects. Thus, if the user specifies the following gating scheme in FlowJo:

```
Gate_A
  |
  ----> Gate_B
          |
          -----> Gate_C
```

Then gatherFlowJoGates will create, as flowCore filter objects:

```
Gate_A
Gate_A  &  Gate_B
```

```
Gate_A  &  Gate_B  & Gate_C
Gate_C
```

To further complicate issues, a single FlowJo workspace often references dozens of FCS files. In addition, each FCS file may be compensated with a different compensation matrix. Finally, the FlowJo user may have implemented many different types of transformations on the data. While currently tracked, it seems that the gating coordinates in the FlowJo workspace are listed in linear space and not as transformed values. Thus, while retrieved, the transformation information is not otherwise used by the code.

Thus, the return value for the readFlowJoList method is a moderately complex data structure (called a flowJoList) that captures the path to all of the referenced FCS files, all of their gates, their compensation matrices, as well as the path to the FlowJo workspace, the version of FlowJo used, and when the file was last updated. This last item can be useful when tracking changes to the gating structures that are made by the cytometrist. The data corresponding to a single FlowJo workspace is called a flowJoObj. One or more flowJoObjs are contained in each flowJoList.

## Value

readFlowJoList returns a *flowJoList*. A flowJoList is essentially a list of lists containing the gating structures, compensation matrices, and other information related to a set of FlowJo workspaces.

## Note

For more details see: The flowFlowJo vignette at `../doc/flowFlowJo.pdf` FlowJo on the web at: `http://www.flowjo.com/`

## Author(s)

John Gosink

## Examples

```
demoLocation <- system.file("extdata", "DemoWorkspace.wsp", package="flowFlowJo");
actualFCSLoc <- system.file("extdata/fcsFiles", package="flowFlowJo");
testList     <- readFlowJoList(demoLocation, altFileLocation=actualFCSLoc);
```

---

# simplifyGateNames *Standardize flow cytometry gate names*

---

## Description

This function helps to standardize typical flow cytometry cell population names. This is useful when trying to combine the results from a large number of flow runs wherein the cytometrist may have used slightly different nomenclature to describe the same set of cells. For example, "Lymphocytes" vs. "lymphoctes".

## Usage

```
simplifyGateNames(x, ...)
```

## Arguments

| | |
|---|---|
| x | Logical. Default=FALSE. A flowCore style filter, or a text string with a gate (filter) name |
| ... | Additional arguments: *upperCaseGates* Logical. Default=FALSE. If TRUE, convert the gate names to all upper case<br>*stripFileName* Logical. Default=TRUE. It is currently the convention that the flowFlowJo code prepends the name of the referenced FCS file within the gate name, followed by a colon, then the rest of the gate information. This setting looks for ".*\.fcs:" and strips it out of the gate name<br>*removeParentalNames* Logical. Default=TRUE. The current *getFlowJoGates* method concatenates a gate's name with the name of all of it's parents. |

## Value

This function returns either a text string or flowCore style filter object.

## Author(s)

John Gosink

## Examples

```
simplifyGateNames(c("LymphoCytes", "lymphocytes", "mOnOcYtEs"));
simplifyGateNames(c("Lymphocytes:CD8+"), removeParentalNames=TRUE);
simplifyGateNames("CD34 positive cells ")
simplifyGateNames("CD34 + ")
```

# Index