

# Visualizing the Effects of Logically Combined Filters

Thomas Geymayer, **Alexander Lex**, Marc Streit, Dieter Schmalstieg

Graz University of Technology, Austria

# Motivation: Large Data Sets

Trend: ever growing data, e.g.,

Biology

Astronomy

Not possible to show each data item

Performance problems

Not enough pixels, overplotting

Overloaded visualizations, clutter

# Solution Approaches

Increasing data-to-ink ratio

Increasing resolution

Abstraction Methods:

- Aggregation

- Random Sampling

- Segmentation

Really important, but not always the best solution!

Sometimes: good old **FILTERING** preferable

# Why use Filtering?

It is easy!

It scales!

For algorithms and visualization alike

A lot of data that is irrelevant!

Noise

Uncertain data (beyond a threshold)

It can easily be integrated into existing software!

It helps you focus on the interesting data!

# State of the Art in Filtering

Often considered a preprocessing step

Mostly used as a **black box**

A number of filters are applied sequentially, the result is visualized

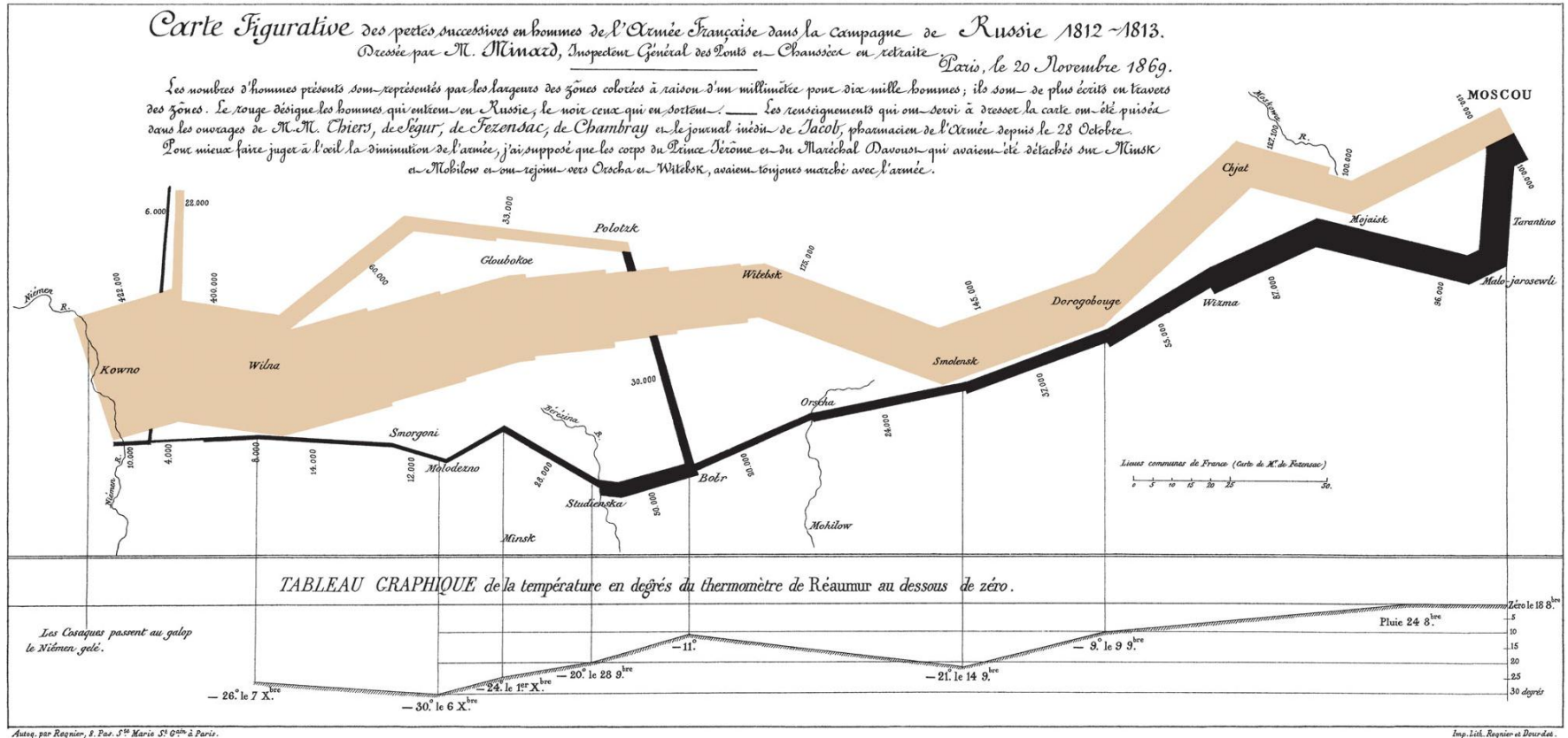
Problems:

Impossible to judge the effects of individual filters

Logical combinations (other than AND) often not supported

# VISUALIZING FILTERS

# Inspiration



[Charles Minard, 1869]



CALEYDO

Alexander Lex

# Main Requirements for Filter Visualization

R1 Show Sequence

R2 Show Consequences

Show how many elements a filter removes.

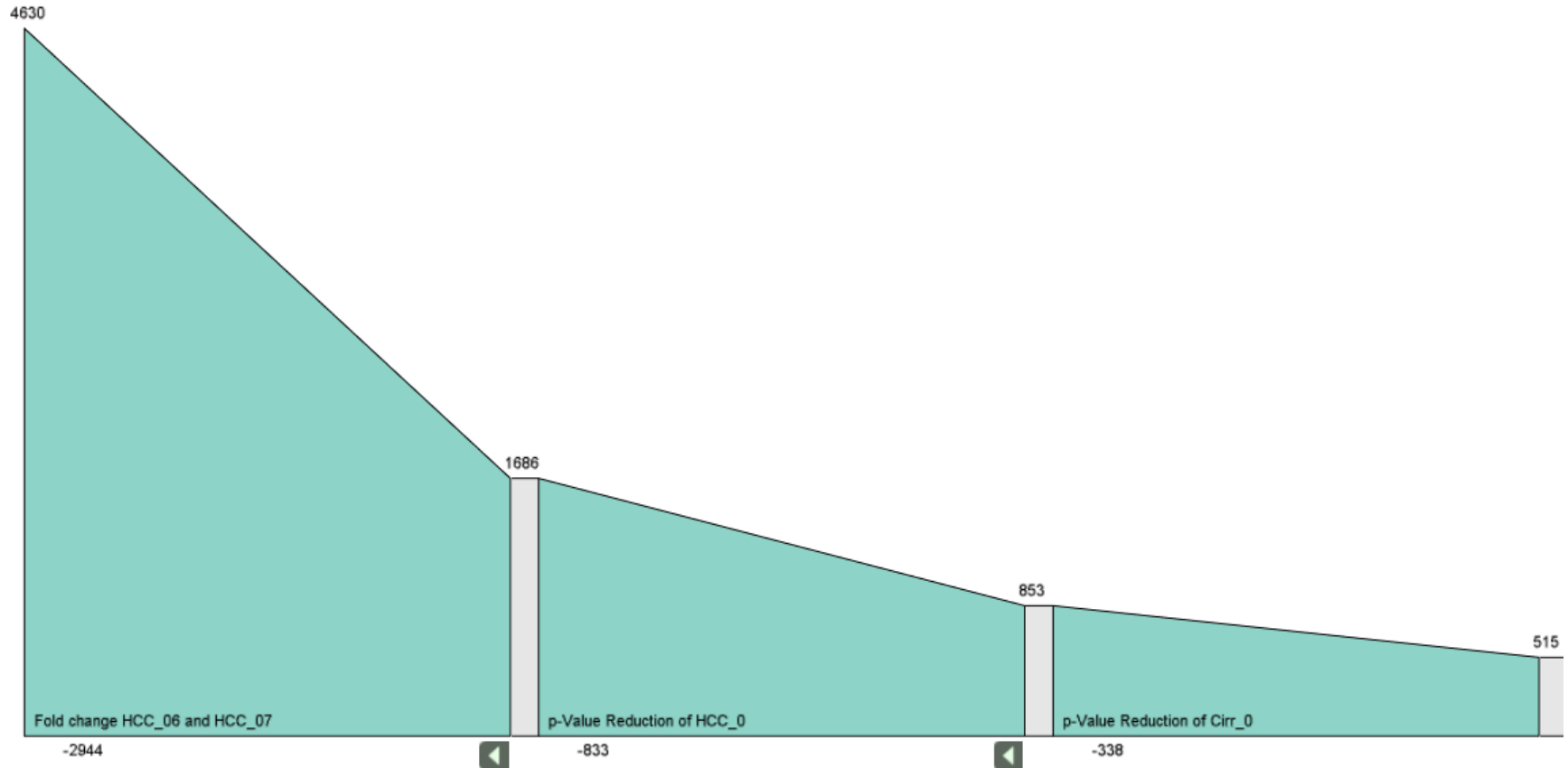
R3 Show and Create Compositions

Sequence of filters is equal to logical AND

Logical operations such as OR and XOR cannot be visualized as easily

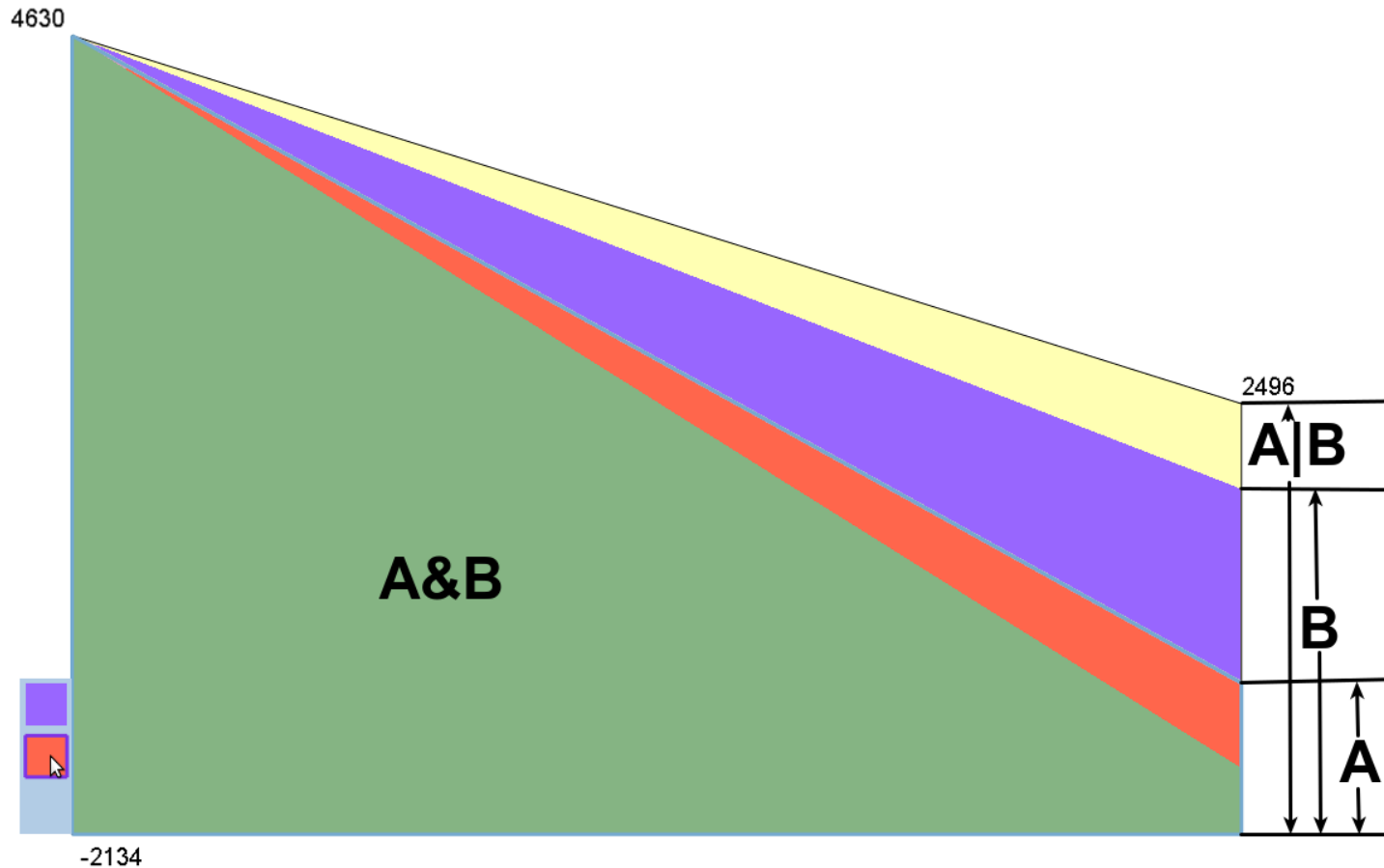


# Filter Pipeline for AND Combinations



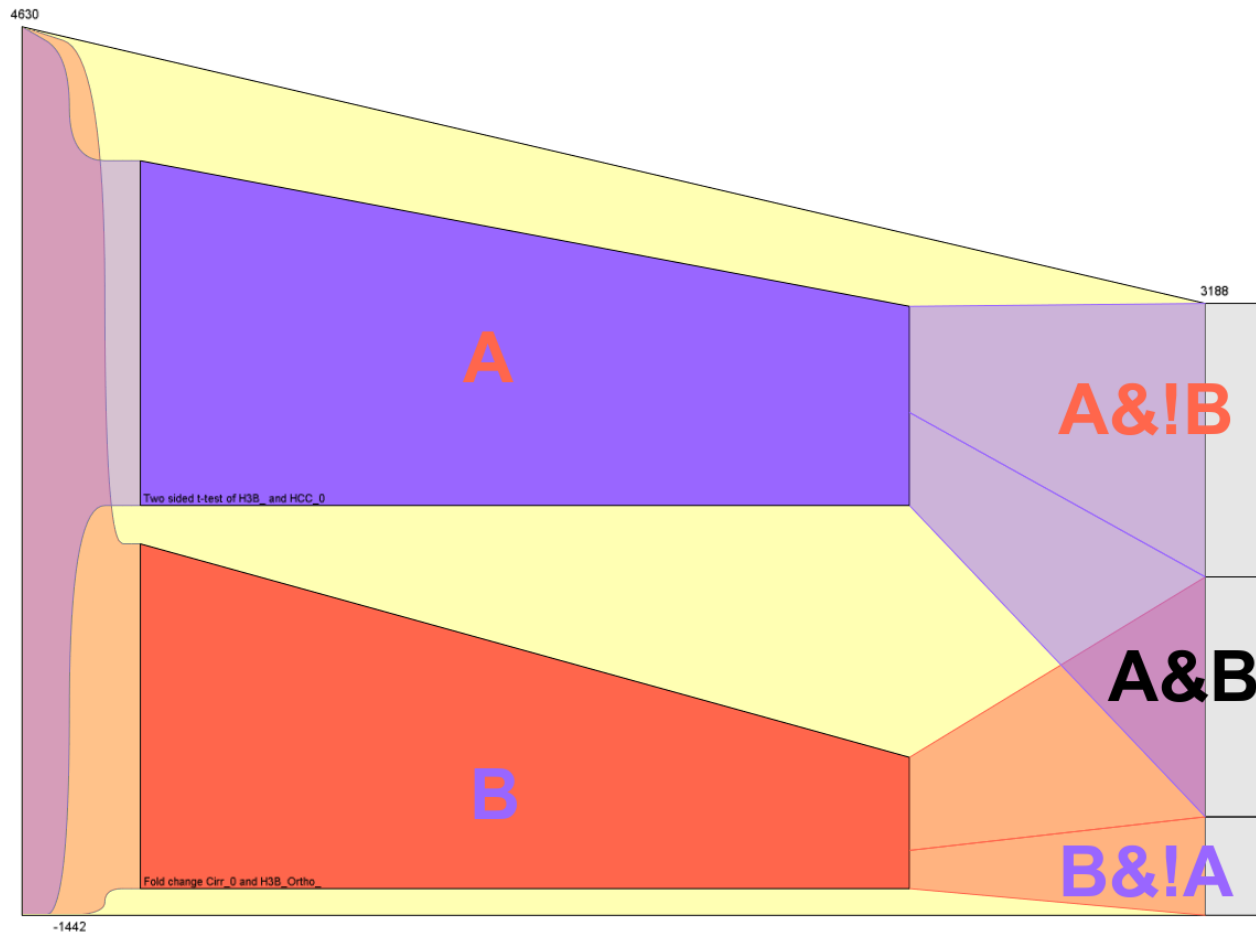
# OR Combinations

## Alternative 1: Filter Fan

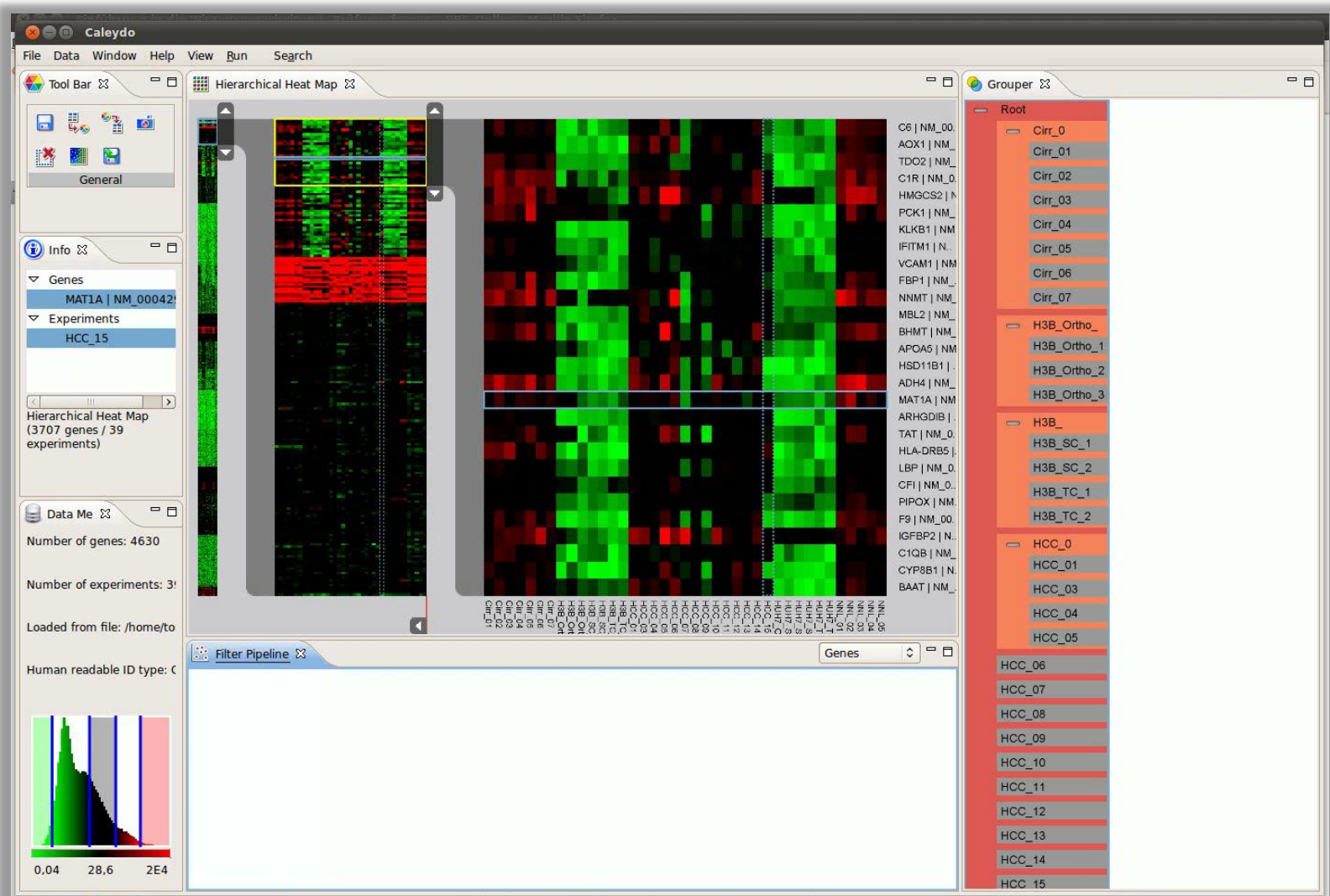


# OR Combinations

## Alternative 2: Parallel Filters



# Interaction



# Recap Main Requirements

R1 Show Sequence



R2 Show Consequences



R3 Show and Create Compositions



# Secondary Requirements

## R4 Modify Filters

Change parameters, remove and move filters

## R5 Hide Filters

To compensate for disproportional filters

## R6 Show Filter Efficiency

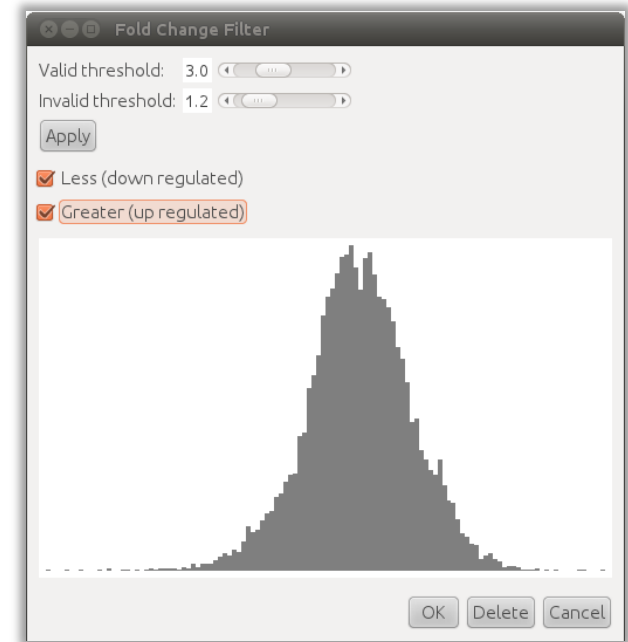
Show effect of every filter on the data set as if it were the only one

# R4: Modify Filters

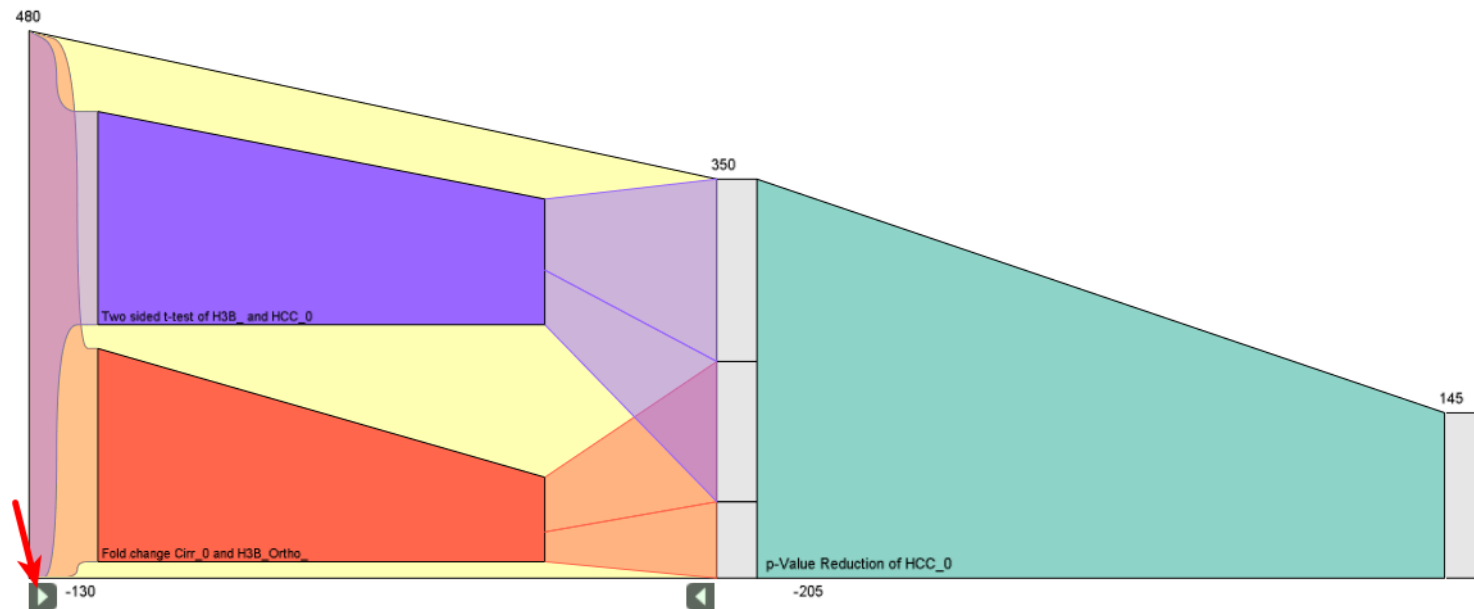
Edit filter parameters by double clicking filter

Move using drag & drop

Remove using context menu

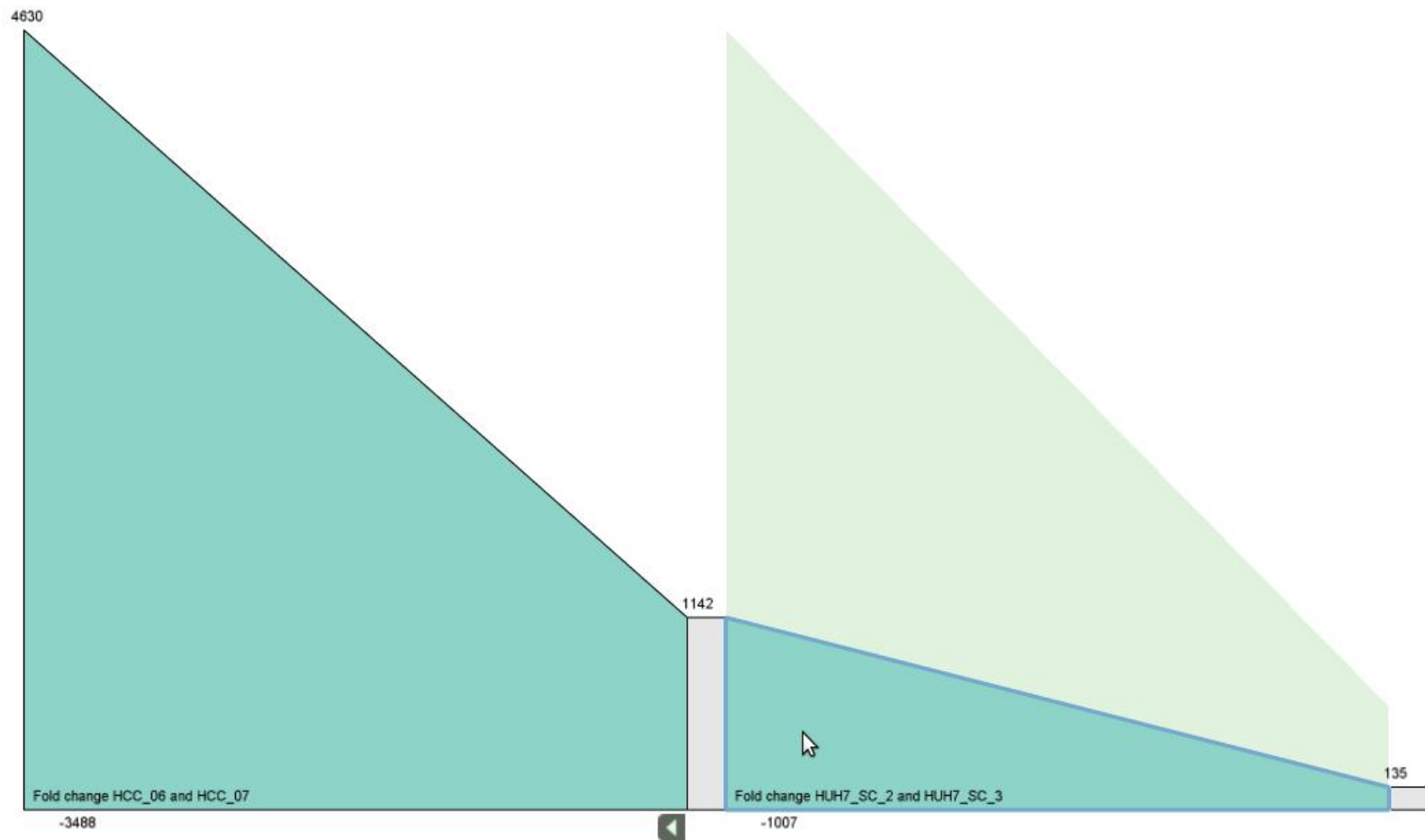


# R5: Hide Filters





# R6: Show Filter Efficiency



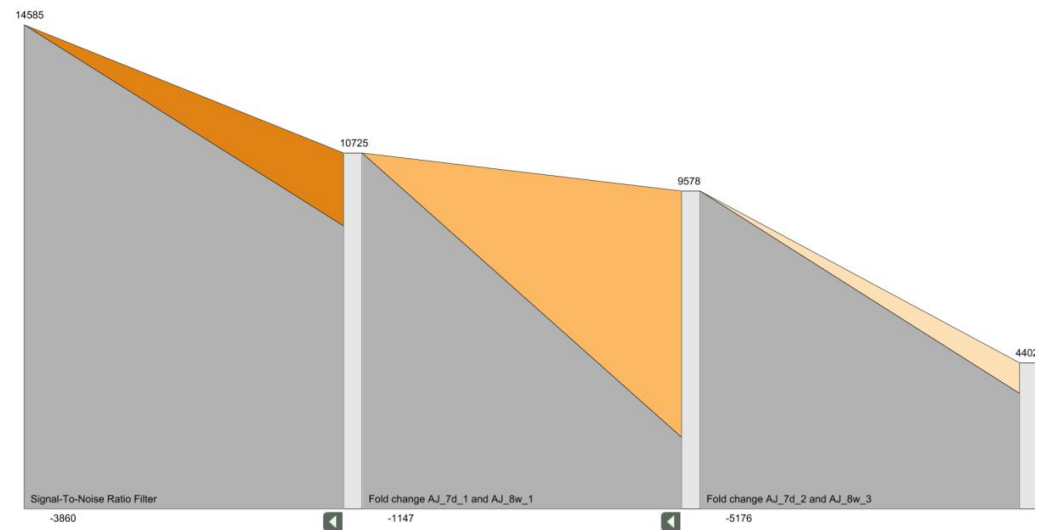
# Future Work, Recent Developments

Non-binary filters (**uncertainty**)

Integrate Not, XOR

Nested Filters

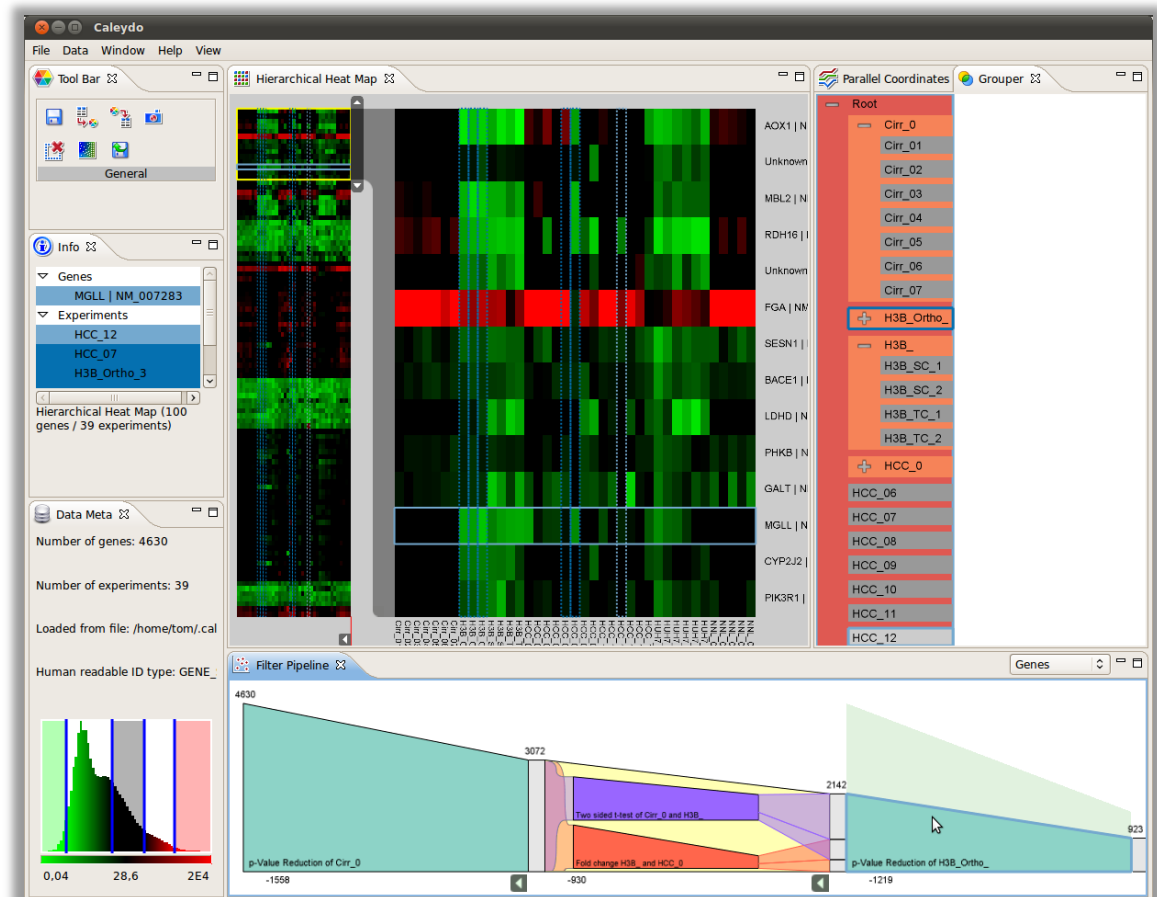
Integrate brushing on subsets



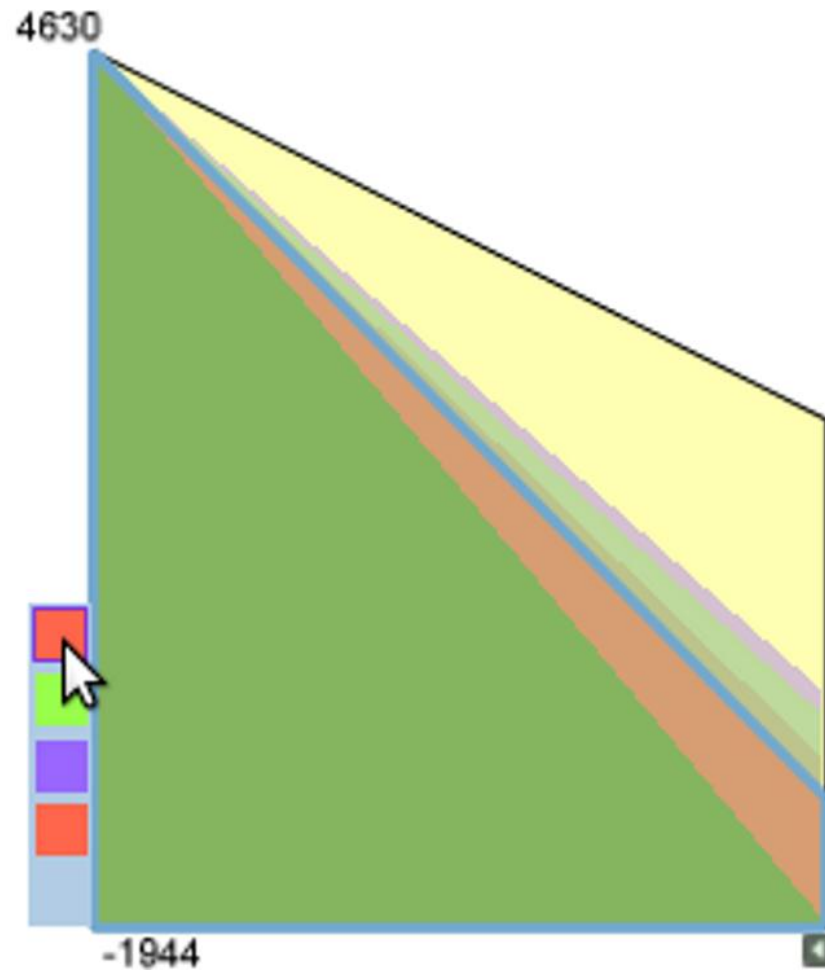
# Thank you for your attention!

## Questions?

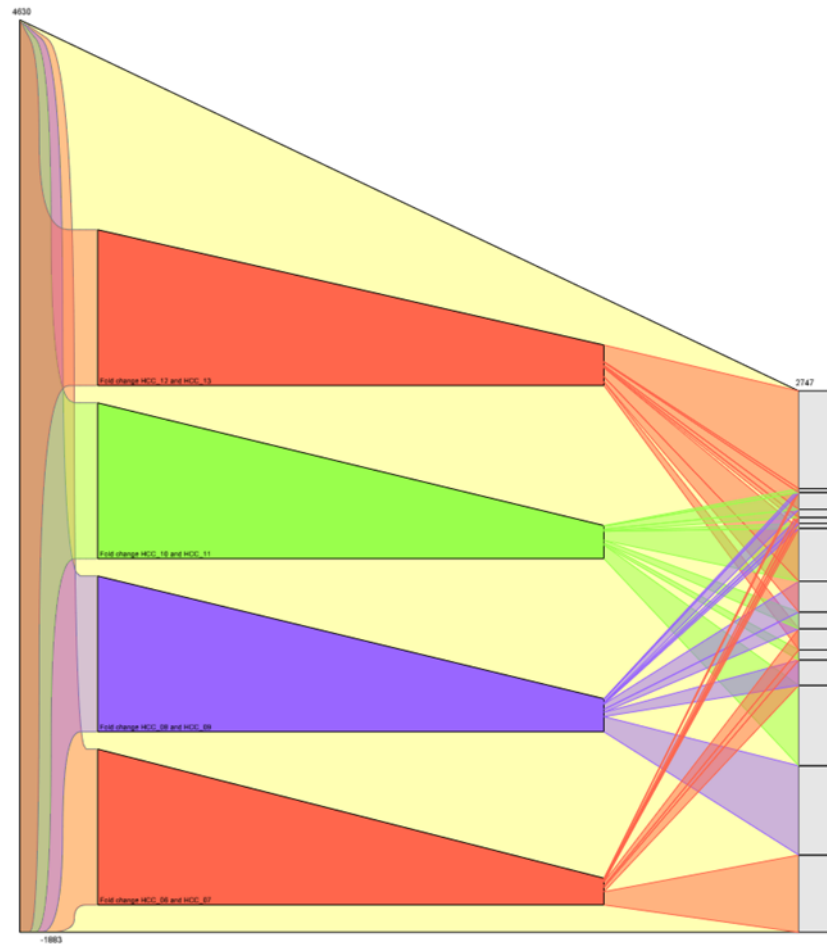
<http://caleydo.org>



# Scalability - FAN



# Scalability - Parallel Filters



# Implementation

Part of Caleydo  
Information  
Visualization  
Framework  
OpenGL, Java, RCP

<http://caleydo.org>

