

# noob: a multiple genome alignment browser

Aneesh Karve

June 6, 2007

## Abstract

We present a multiple genome alignment browser for sequences from numerous multiple alignments. Our goal is to use information visualization to help users think about how sequences have evolved over time. Our application takes as input a series of multiple alignments in MAF format<sup>1</sup> and outputs an interactive visual display of the data.

## 1 Information visualization

Information visualization has informally been described as “using vision to think” [CMS99]. Information visualization concerns itself with abstract data, whereas scientific visualization concerns itself with physical data measured over space and time (e.g. MRI visualization in medical applications). Although some of the data we visualize is inherently spatial—sequence fragments along chromosomes, for instance—no attempt is made to display these fragments as they appear in physical reality. Instead, an abstraction is applied to transform the data into a visual format that is informative to the user. This is, in fact, the core problem of information visualization: to find visual abstractions that efficiently convey data attributes to the user. This involves a variety of perceptual considerations which are covered in [War04] and [Kar07]. The aforementioned papers also provide an overview of information visualization as a whole.

---

<sup>1</sup><http://genome.ucsc.edu/FAQ/FAQformat>

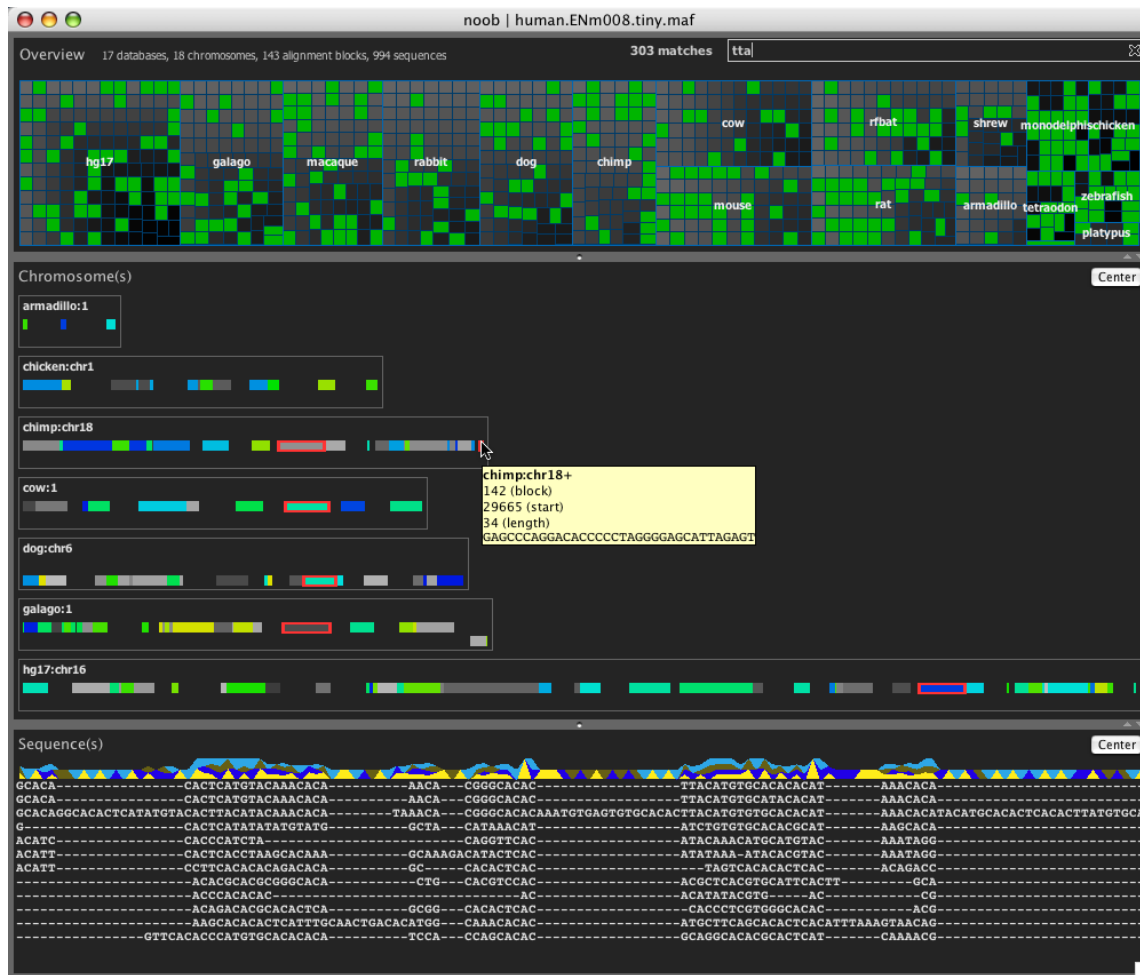


Figure 1: The noob interface.

## 2 noob interface design

One heuristic we have used to design noob is the so-called *visual information seeking mantra*: “Overview first, zoom and filter, then details on demand” [Shn96]. The idea is to start by giving the user a “big picture” feel for the entire dataset and further allow him to explore subsets of interest. The noob application (Fig. 1) is divided into three panels which, in top-to-bottom order, progress from overview to detail. The top panel provides an overview in the form of a treemap. Treemaps are space-filling visualizations of hierarchical data. In our case the hierarchy is *database* > *chromosome* > *sequence*. The advantage of using a treemap visualization is that it organizes the sequences and does so in a non-occlusive way that does not depend on perspective (i.e. all sequences are always visible; compare a browsable tree, which does not necessarily meet these qualifications). Users can explore the details of a sequence by hovering over it with the mouse. The Overview panel also provides filter controls in the form of regular expression searching. The search panel searches a custom data field of the format `[database]:[chromosome][strand]:b[block number]>[sequence]`. The block number indicates which MAF block the sequence came from (the first block in the input file is block 1). The overview treemap uses color for two purposes. First, search hits are highlighted in green. Second, each sequence is given a grayscale color according to its block number. In this way, sequences that appear earlier in the input file will appear darker (black), and sequences that appear later are lighter (gray).

Once the user has filtered the dataset using search, the search results appear in the middle, or “chromosome” panel. This panel reveals how the result sequences are arranged on their parent chromosomes. (A simple way to match all sequences is to type ‘.’ in the search panel.) Each sequence is represented as a rectangle. The color of a rectangle is determined by the sequence’s block number. The color encoding guarantees that all sequences in the same block will have the same color (there may be color collisions across blocks, but we have attempted to minimize them). As above, users can hover over sequences for tooltip details. Users can also select a block, and all constituent sequences, by clicking on any sequence in that block. Selected sequences are outlined in red.

Once a block is selected, it appears in the bottom or “sequence” panel. The sequence panel displays the multiple alignment block along with a sequence-logo-like visualization that reveals the information content of a given nucleotide for each position. Purines are depicted by a pair of yellowish colors; pyrimidines by a pair of bluish colors. In the sequence panel, as in the chromosome panel, users can also zoom and pan the display by dragging and command-dragging, respectively.

To summarize, our design is a tiered visualization in which different levels of

logical zoom are coordinated to allow the user to explore details while maintaining the context of said details.

## 3 Application details

noob is implemented in Java 1.5 with Jeffrey Heer’s `prefuse`<sup>2</sup> information visualization toolkit [HCL05] (`prefuse` beta release 2006.07.15).

### 3.1 Usage notes

- To invoke the application: `java -jar noob.jar foo.maf`. The `noob.jar` file should be in the same directory as the `lib/` folder, which contains the necessary libraries.
- Users can zoom and pan the chromosome and sequence displays by dragging and command-dragging, respectively.
- The “Center” button can be used to reset the view perspective. This is useful if the visual content “disappears” due to panning or zooming.
- If the JVM runs out of heap space during startup (not uncommon for large datasets), try restarting the JVM with heap space flags that request more memory (e.g. `-Xms128m -Xmx128m` specifies the min and max heap sizes as 128MB.)
- The applied version of the `prefuse` library is fairly verbose in logging messages to `System.err`. In general, these are not error messages. Experience indicates that these messages may hamstring performance (at least in the NetBeans IDE).
- On a 1.33 GHz Powerbook G4, noob is passably performant on datasets with 1000 to 5000 sequences. Startup times may be slow. For large datasets, better performance can be obtained by partitioning the dataset and invoking multiple instances of the noob application.

## 4 Evaluation

To benchmark the effectiveness of the noob visualization, user studies are required. These are beyond the scope of this project. That said, the original project proposal

---

<sup>2</sup><http://prefuse.org/>

had the following goal: “A multi-genome browser that shows how homologous genes are distributed across chromosomes and how these genes might have evolved from a common ancestor. The current crop of visualization tools do not fully address this need. On a theoretical level we wish to prototype visualizations that will help users think about how sequences diverged and evolved over time.” The stated goal has been achieved, but there is significant room for improvement (see Section 5).

## 5 Future work

Further development of the noob application would benefit from a comprehensive literature and tool survey to determine what has already been done in this area, precisely which functions are still needed by users, etc.

A phylogenetic tree visualization, revealing how sequences might have diverged from a common ancestor, might be useful. The original project proposal included a fourth panel to display such a tree. Due to time constraints, this feature was cut. The visualization part of the feature could be realized with existing `prefuse` classes, but the parser, be it for Newick or some other format, would require custom code.

A wide variety of enhancements for noob are possible. Perhaps the best way to prioritize these features is through a related works survey and interviews of prospective users. A few simple features that could enhance the sequence view are the following: different colors for different sequences (to facilitate distinction); tooltips or other text to reveal the source database and chromosome for a given sequence.

## References

- [CMS99] Stuart K. Card, Jock Mackinlay, and Ben Shneiderman. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, 1999.
- [HCL05] Jeffrey Heer, Stuart K. Card, and James A. Landay. `prefuse`: a toolkit for interactive information visualization. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430, New York, NY, USA, 2005. ACM Press.
- [Kar07] Aneesh Karve. Glyph-based overviews of large-scale datasets in structural bioinformatics. Master’s thesis, UW-Madison, 2007. <http://www.cs.wisc.edu/~karve/thesis/thesis.pdf>.

- [Shn96] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. IEEE Symp. on Visual Languages*, 1996.
- [War04] Colin Ware. *Information visualization: perception for design*. Morgan Kaufmann, 2nd edition, 2004.