Visualizing genomes: techniques and challenges

Cydney B Nielsen[1], Michael Cantor[2], Inna Dubchak[2,3], David Gordon[4] & Ting Wang[5]

As our ability to generate sequencing data continues to increase, data analysis is replacing data generation as the rate-limiting step in genomics studies. Here we provide a guide to genomic data visualization tools that facilitate analysis tasks by enabling researchers to explore, interpret and manipulate their data, and in some cases perform on-the-fly computations. We will discuss graphical methods designed for the analysis of *de novo* sequencing assemblies and read alignments, genome browsing, and comparative genomics, highlighting the strengths and limitations of these approaches and the challenges ahead.

The study of genomes has to a large extent become a digital science made possible by the advent of sequencing technology and its power to detect genomic sequence at nucleotide resolution. The emergence of extensive sequence data resources opened new interfaces with computer science, fuelling fields such as bioinformatics, and enabled biological questions to be tackled computationally. The recent innovations in sequencing technology provide an unprecedented capacity for data generation. Now more than ever, we require intuitive and rapid data exploration and analysis capabilities.

Although many genome data analysis tasks can be accomplished with automated processes, some steps continue to require human judgment and are frequently rate limiting. Visualization can augment our ability to reason about complex data, thereby increasing the efficiency of manual analyses. In some cases, the appropriate image makes the solution obvious. Given the importance of human interpretation particularly in the early hypothesis generation stages of biological research, visual tools also provide a valuable complement to automated computational techniques in enabling us to derive scientific insight from large-scale genomic data sets. Visual and automated approaches are particularly powerful when used in combination, such that a user can seamlessly inspect and perform computations on their data, iteratively refining their analyses.

One challenge in designing visual tools is deciding on a graphical representation—essentially, how the data are encoded into colors and shapes or transformed onto different scales. The choice of representation can either help or hinder a user's ability to interpret the data and ideally should be designed to facilitate the analysis task. For example, genomic rearrangements may be more easily viewed as arcs on a circle than on a line. Genomic data are derived from diverse sources using different techniques, each accompanied by its own experimental error. It is important that visual representations capture this technical uncertainty and any resulting inconsistencies. There is also substantial biological variation between individuals, which needs to be distinguished from the technical variation mentioned above. In addition to the challenges of choosing an appropriate visual representation, some types of primary data are unavailable owing to their prohibitive online storage requirements, and enabling real-time interaction with large-scale data sets is nontrivial.

This review highlights examples from three core user tasks: (i) analyzing sequence data, both in the context of *de novo* assembly and of resequencing experiments, (ii) browsing annotations and experimental data mapped to a reference genome and, finally, (iii) comparing sequences from different organisms or individuals.

[1]British Columbia Cancer Agency, Genome Sciences Centre, Vancouver, British Columbia, Canada. [2]Department of Energy Joint Genome Institutes, Walnut Creek, California, USA. [3]Genomics Division, Lawrence Berkeley National Laboratory, Berkeley, California, USA. [4]Department of Genome Sciences, University of Washington, Seattle, Washington, USA. [5]Department of Genetics, Center for Genome Sciences, Washington University School of Medicine, St. Louis, Missouri, USA. Correspondence should be addressed to C.B.N. (cydneyn@bcgsc.ca).

PUBLISHED ONLINE 1 MARCH 2010; DOI:10.1038/NMETH.1422

Visualization methods in these domains are at different stages of maturity, and we will discuss their respective strengths and limitations. One important consideration is that the field of genomics is rapidly evolving. Although we have attempted to provide a guide to the techniques in this area, it is likely that new tools and data formats will emerge in the very near future, and we will discuss some of the associated challenges. We encourage readers to consult online resources, such as SEQanswers (http://seqanswers.com/), for the most recent tool developments.

Visualizing sequencing data

Interpreting the raw data from a sequencing machine begins with automated data processing. Base calling and quality calculations are followed by sequence assembly in the case of *de novo* genome sequencing projects, or by read alignment to a reference in the case of resequencing. Recent innovations in sequencing technology have been accompanied by a growth in new assembly and alignment programs to cope with the shorter read lengths and larger numbers of reads (for reviews, see refs. 1,2), but no standards have been reached. For some downstream analysis tasks, visual inspection is valuable in interpreting and validating automated outputs and can drive both biological insight and algorithmic improvements. For example, automated single nucleotide polymorphism (SNP) detection based on sequencing data remains imperfect, and visual inspection is still used to evaluate individual cases for both biological implications and technical observations that may be used to improve the prediction algorithm. This section highlights current graphical tools for analyzing sequencing reads.

Visualizing alignments. Analysis of assemblies and read alignments often involves examination of the sequencing reads themselves, and all tools listed in **Table 1** provide a view of aligned read bases. Read sequences are typically represented as letter strings oriented horizontally from left to right and stacked vertically. In the case of assemblies, a user can scan down the corresponding column in the read stack (**Fig. 1**) to identify the bases contributing to the consensus at a given position. Base qualities (log-transformed error probabilities) are often indicated with gray scale and bases that disagree with the consensus emphasized with color[3–5]. Some tools minimize the visual clutter in the read stack by highlighting only discrepancies and concealing all consistent base pairs (for example, Integrative Genomics Viewer (IGV), Hawkeye[6], US National Center for Biotechnology Information (NCBI) Assembly Archive Viewer[7], Text Alignment Viewer in SAMtools[8]).

Most tools built before the emergence of next-generation sequencing (NGS) continue to support visualization of the underlying primary data for Sanger reads through a separate 'trace' view. For example, in the popular program Consed[3], the 'trace' window can be launched from the 'aligned reads' window, and cursor movement is synchronized between the two displays (**Fig. 1**). This view allows a user to inspect positions with conflicting bases and uncover the source of ambiguity within the primary traces directly (for example, a base-calling error in one of the reads, a misassembly, or a polymorphism). To a large extent, NGS data has changed how a user evaluates uncertain consensus bases. For example, Consed allows the user to inspect raw Roche 454 sequencing data, but in the case of Illumina and Applied Biosystems' SOLiD data, there are no raw read traces, only image data. (Details of these sequencing technologies are reviewed elsewhere[9,10].) Consed and similar programs do not display primary image data, in part because their large size makes them too expensive to keep in online storage and too slow to display. However, the high read coverage routinely generated by NGS often alleviates the need to inspect any one read. A user can evaluate a suspect base in one read through comparison with the corresponding bases in the other aligned reads at the same location.

Finishing. The output of automated sequence assembly programs is imperfect, and repeat regions, read length and coverage limit contiguity. The next step, 'finishing', involves closing gaps, correcting misassemblies and improving the error probabilities of consensus bases. Specialized finishing software facilitates this process by automating and/or allowing a user to perform the above-mentioned tasks. In some cases, automated finishing is sufficient—for example, as performed by Autofinish[11], which is a program that examines the output of an assembly program and suggests what laboratory data to acquire (for example, specific primers for PCR). However, in other situations manual inspection and editing are needed to complement automation. Gap4 (refs. 12,13), Consed and commercially available products such as Sequencher (Gene Codes Corporation) and Lasergene[14] (DNASTAR) are widely used finishing programs that provide rich editing functionality and history tracking and enable the user to manually break apart and join contigs, which distinguishes them from static alignment viewers that do not allow editing (**Table 1**).

In most sequencing protocols, the size range of genomic fragments is known. The sequencing reads derived from opposite ends of the same source genomic fragment ('mate pairs') therefore have an expected distance ('insert size') and expected orientation (one top strand read and one bottom strand read). Mate pairs that violate these spatial constraints can be used to reveal misassemblies, while consistent mate pairs can be used to join contigs together.

Consed's 'assembly view' depicts mate pairs as color-coded lines spanning contigs, with the contigs represented by horizontally oriented blocks. This display visually separates 'consistent' pairs (those of expected insert size and orientation) from the 'inconsistent' pairs (those with unexpected insert size or orientation) by plotting them above and below the contig boxes, respectively, which helps to reveal misassemblies (**Fig. 1a**). One advantage of this tool is that it allows interactive filtering of the displayed data (contigs, mate pairs, similar sequences and so on). Despite this filtering, one limitation is that the view can quickly become cluttered as the number of mate pairs increases. For example, in Consed it is sometimes desirable to turn off the display of all consistent mate pairs internal to a contig because their number overwhelms the image. Applying biologically meaningful aggregation methods and summary techniques to highlight only the most well-supported connections remains an outstanding challenge.

In addition to mate pair relationships, sequence similarity can be used to identify possible contig joins missed by the assembly program. For example, a user can interactively request an alignment of two selected regions within Consed and inspect the output in the 'compare contigs' window. Similar functionality exists in other finishing software; for example, Gap4 provides a 'contig joining editor'. These sequence-based views are complemented by overview displays. Gap4 uses a dot-plot representation wherein each axis indicates positions along a contig's length and dots demark the locations sharing above-threshold

Table 1 | Tools for visualizing sequencing data

Name	Cost	OS	Description	URL
Stand-alone tools				
ABySS-Explorer[25]	Free	Win, Mac, Linux	Interactive assembly structure visualization tool	http://tinyurl.com/abyss-explorer/
CLC Genomics Workbench	$	Win, Mac, Linux	Integrates NGS data visualization with analysis tools; user friendly	http://www.clcbio.com/
Consed[3]*	Free	Mac, Linux	Widely used; assembly finishing package; NGS compatible	http://www.phrap.org/
DNASTAR Lasergene[14]	$	Win, Mac	Analysis suite with an assembly finishing package; NGS compatible	http://www.dnastar.com/
EagleView[17]	Free	Win, Mac, Linux	Assembly viewer; compatible with single-end NGS	http://tinyurl.com/eagleview/
Gap[12,13]	Free	Linux	Widely used; assembly finishing package; Gap5 is NGS compatible	http://staden.sourceforge.net/
Hawkeye[6]	Free	Win, Mac, Linux (S)	Sanger sequencing assembly viewer	http://amos.sourceforge.net/hawkeye/
Integrative Genomics Viewer (IGV)*	Free	Win, Mac, Linux	Genome browser with alignment view support (**Table 2**); NGS compatible	http://www.broadinstitute.org/igv/
MapView[18]	Free	Win, Linux	Read alignment viewer; custom file format for fast NGS data loading	http://evolution.sysu.edu.cn/mapview/
MaqView	Free	Mac, Linux	Read alignment viewer; fast NGS data loading from Maq alignment files	http://maq.sourceforge.net/
Orchid	Free	Linux (S)	Assembly viewer customized to display paired-end relationships	http://tinyurl.com/orchid-view/
Sequencher	$	Win, Mac	Assembly finishing package	http://www.genecodes.com/
SAMtools tview[8]	Free	Win, Mac, Linux	Simple and fast text alignment viewer; NGS compatible	http://samtools.sourceforge.net/
Web-based tools				
LookSeq[19]	Free		Uses AJAX; *y* axis for insert size; user configures data resources; NGS compatible	http://lookseq.sourceforge.net/
NCBI Assembly Archive Viewer[7]	Free		Graphical interface to contig and trace data in NCBI's Assembly Archive	http://tinyurl.com/assmbrowser/

Free means the tool is free for academic use; $ means there is a cost. OS, operating system: Win, Microsoft Windows; Mac, Macintosh OS X. Tools running on Linux usually also run on other versions of Unix. (S) indicates that compilation from source is required. "Assembly finishing package" enables interactive sequence editing and/or integration with tools for automated assembly improvement.
*Our recommendation

sequence similarity. A user can interactively explore the sequence relationships between different contigs and view the results of search operations such as 'find repeats'. Consed's assembly view can display the output of a sequence comparison utility called 'cross_match', using arcs to connect regions with sequence similarity between user-selected contigs. Different colors distinguish features such as directed repeats from inverted repeats. One advantage of viewing sequence similarity in 'assembly view' is that it can be integrated with a read coverage plot (**Fig. 1a**), which can reveal regions of unexpectedly high coverage often indicative of similar sequences that were erroneously collapsed by the assembler into one. The user can click to examine the sequence similarity at the base level, and click again to examine the underlying reads. There are also standalone tools with related functionality; for example, Miropeats[15], widely used for early genome sequencing projects, is a UNIX C-shell script that generates static images using arc representations to indicate different types of repeats.

Next-generation sequence viewers. As sequencing throughput increases and costs decrease, individual genome sequencing has become feasible and has led to initiatives such as the 1,000 Genomes project (http://www.1000genomes.org/). These data provide an unprecedented opportunity to characterize the landscape of human genotypes, and a new generation of computational methods has emerged as a result[16]. In some cases, visual inspection can facilitate the evaluation and interpretation of read alignment techniques and variation detection outputs. Assembly visualization tools possess most of the necessary functionality, but they were built with Sanger data in mind and initially strained under the substantially higher read volume of NGS technologies. Several of these tools are being retrofitted to tackle larger data sets, including Consed and the updated Gap5, but a new wave of tools is also being designed with this purpose in mind: for example, EagleView[17], MapView[18] and IGV (**Table 1**). Unlike finishing software, these tools are primarily data viewers and do not provide direct editing functionality. Because of their emphasis on browsing, many provide more flexible zooming capabilities and enable a user to freely zoom out to higher-level views. The commercially available CLC Genomics Workbench (CLC bio) is particularly user friendly and includes its own read alignment programs, which can be launched through a GUI.

In the resequencing context, mate pairs provide valuable information about structural variation, such as insertions, deletions and inversions. As discussed in the previous section, mate pairs can also indicate misassemblies, and users performing variation detection on draft assemblies should be aware of these issues. LookSeq[19] and Gap5 use the vertical-axis position to indicate insertion size. This places inconsistent mate pairs at the extremes of the plot and visually separates large insert sizes, which are consistent with deletions, from small insert sizes, which suggest insertion events. When analyzing structural variations, it is important to consider gene annotations—for example, whether a single nucleotide variation leads to a synonymous or nonsynonymous amino acid change. For this reason, several of these visualization

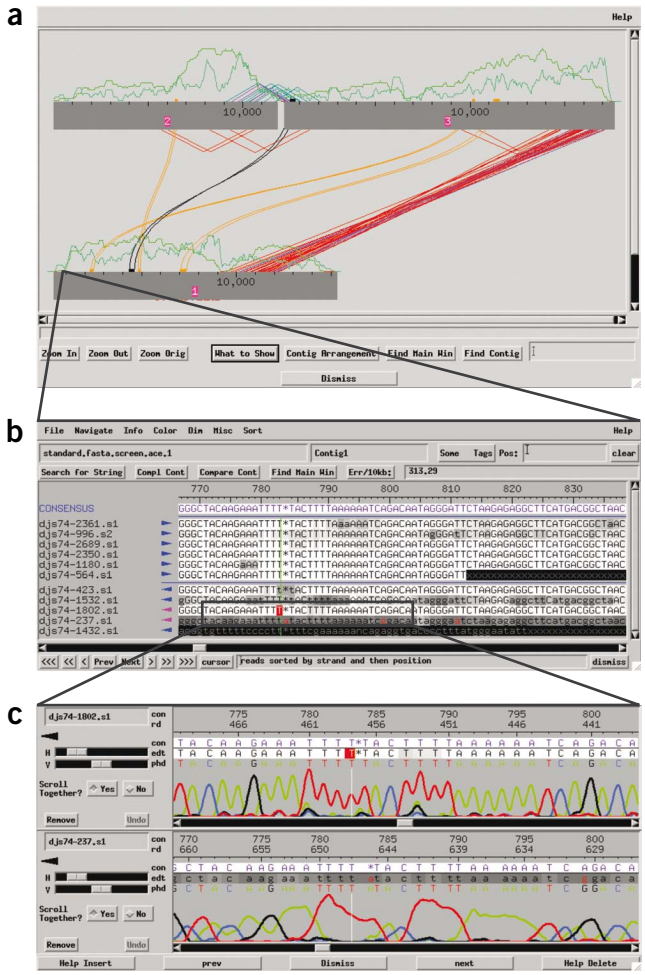

Figure 1 | Screenshots of connected views in Consed. (**a**) Contigs from a human BAC clone assembly are shown in 'assembly view' as gray boxes with a scale of nucleotide positions within the contig. Angled colored lines represent mate pairs (aqua, consistent; red, inconsistent; purple, multiple at same location). Curved lines indicate sequence similarity computed using cross_match (orange, directed; black, inverted orientation). The read coverage is plotted along the contigs in dark green and mate-pair coverage highlighted in light green. (**b**) The 'aligned reads' window displays a vertical stack of read sequences, optionally separated by strand, with forward on top (right arrows) and reverse on bottom (left arrows). The * character in the computed consensus indicates that one or more of the reads contains an insertion at this position that the assembly program deems incorrect. (**c**) By inspecting the read traces in the 'trace' window, the user can evaluate the insertion and override the assembly program's choice of consensus if needed.

tools and some finishing software support the display of annotations. Consed, for instance, optionally displays the amino acid translation of the consensus in all six reading frames and allows the user to annotate genotypes, repeats and user-defined genes.

Challenges. NGS technologies and the high volume of data they produce give rise to both computational and representational challenges. New file formats—for example, the Sequence Alignment/Map (SAM) format[8], adopted for the 1,000 Genomes Project, and the Compact Alignment Format, CALF (http://www.phrap.org/phredphrap/calf.pdf)—provide compact storage of read alignment data. Preindexing—for example, of BAM files (the companion binary representation of SAM)—is being increasingly used to achieve fast random retrieval of alignment data and reduce the memory requirements of interactive alignment viewers.

In addition to these computational hurdles, NGS data pose representational challenges. For example, most read alignment viewers render all available aligned reads using sorting or color-coding by quality to guide the user. However, this representation breaks down when hundreds or thousands of reads map to a single location. Users require summary methods that consider base and alignment qualities in order to obtain a high-level overview, together with interactive access to the underlying data on demand. In addition, recent NGS assembly programs based on de Bruijn graphs produce contig connectivity information that can become complex (for reviews, see refs. 1,2). Assembly graph images[20–24] including an interactive viewer[25] are emerging to enable higher-level assembly structure visualization.

Part of the power of assembly finishing software comes from integrating on-the-fly analysis operations with the visualization. Sequence similarity searches resulting in dynamic alignment visualizations are one example. In addition, user efficiency can be greatly improved by providing recommendations for where to look. For example, a user can jump to the next 'low consensus quality' region using Consed's navigation menu instead of manually evaluating all positions. Achieving this type of integration between visual and computational analyses will be important in tackling our growing data analysis needs.

Browsing genomes

The end product of genome sequencing, assembly and finishing cycles is a highly contiguous sequence in which most contigs have lengths that are orders of magnitude longer than an individual read. How can a researcher navigate this sequence to find regions of interest? The sequence provides a reference coordinate system and a natural platform on which to assemble scientific annotations and genome-mapped data sets from diverse sources. Genome browsers were originally developed to display data on early draft assemblies, such as the *Caenorhabditis elegans* genome[26] (for example, AceDB[27]), and, later, those of other model organisms (for example, GBrowse[28]), and the human genome assembly[29] (for example, the University of California Santa Cruz (UCSC) Genome Browser[30], the Ensembl Genome Browser[31,32] and the NCBI MapViewer[33]). These browsers share much functionality and their main differences have been reviewed elsewhere[34,35]. Today, browsers have become a standard tool for exploring genomes, facilitating analysis of genome-anchored data, and providing a common platform for investigators to share, store and publish scientific discoveries (**Table 2**).

Genome browsers in a nutshell. In general, genome browsers display data and biological annotations from many sources in their genomic context within a graphical interface. These tools support data types including gene expression, genotype variation, cross-species comparisons and many more. Annotations of functionally important regions such as the locations of genes, regions with transcriptional activity or regulatory elements, derive from either experimental results (for example, sequenced transcripts) or from simulation studies (for example, gene model predictions). Both data and annotations are

Table 2 | Genome browsers

Name	Description	URL
Stand-alone browsers		
Argo	Supports manual annotation of whole genomes	http://tinyurl.com/argo-combo
CGView[82]	Circular genome visualization	http://wishart.biology.ualberta.ca/cgview/
Gaggle[83]	Genome browser within an analysis framework; good microarray support	http://gaggle.systemsbiology.net/
Integrative Genomics Viewer (IGV)*	Flexible user interface; can integrate metadata as heat maps	http://www.broadinstitute.org/igv/
Integrated Genome Browser (IGB)[84]	GenoViz project genome browser; reusable visualization components	http://genoviz.sourceforge.net/
NCBI Genome Workbench	Displays sequence data in many views; integrated with BLAST	http://tinyurl.com/gbench/
Web-based browsers		
AnnoJ	Designed for NGS data; uses AJAX; assemble by html configuration	http://www.annoj.org/
Cancer Molecular Analysis Portal	Integrates clinical data; designed for TCGA project	https://cma.nci.nih.gov/cma-tcga/
Ensembl[31,32]*	Comprehensive genome browser and database; strong user support	http://www.ensembl.org/
GBrowse[28]*	GMOD[28]* component; back end of WormBase, FlyBase; v2.0 uses AJAX	http://gmod.org/wiki/Gbrowse
Genome Projector[42]	Offers circular and pathway views; user configures data resources	http://tinyurl.com/gprojector/
JBrowse[39]	Component of GMOD[28]*; AJAX interface; user configures data resources	http://jbrowse.org/
JGI	Supports live annotation; primary portal for JGI genome projects	http://genome.jgi-psf.org/
NCBI Map Viewer[33]	Vertically oriented viewer; integrated with NCBI resources and tools	http://tinyurl.com/mapview1/
UCSC Genome Browser[30]*	Comprehensive genome browser and database; strong user support	http://tinyurl.com/ucscbrowser/
UCSC Cancer Genomics Browser[43]	Integrates clinical data; offers a pathway view; portal for TCGA data	http://genome-cancer.ucsc.edu/
UTGB	Toolkit to construct personalized browser; uses AJAX; user configures data resources	http://utgenome.org/
X:map[41]	Customized to view Affymetrix exon arrays	http://xmap.picr.man.ac.uk/

All listed tools are free for academic use, and all are available for Microsoft Windows, Macintosh OS X and Linux. Tools running on Linux usually also run on other versions of Unix.

usually organized into 'tracks', which can be preloaded into a genome browser or uploaded on demand.

Investigators frequently wish to examine particular regions of interest, and all current genome browsers allow a user to select specific genomic locations to display. Most tools also provide the ability to search for sequences and for specific genome annotations, such as gene names, that reside in the underlying databases. Many genome browsers also permit complex database queries and provide a suite of tools to access annotation lists for specific regions or for the whole genome. For example, Galaxy[36] is a service specifically designed to interface with genome browsers and facilitate data manipulation and analysis.

Part of the value of genome browsers is that they are customizable. For example, a user can decide on the resolution at which information is shown (say, a window of several hundred base pairs versus tens of thousands) and zoom and pan at will. Data tracks can be freely ordered and organized to facilitate comparisons. In most cases, users can also choose among and configure several modes of display to examine the same underlying data. For example, continuously valued data such as that from chromatin immunoprecipitation (ChIP) can be uploaded as 'wiggle tracks' and displayed as heat maps or histograms (**Fig. 2a**). The popularity of the UCSC Genome Browser stems from its flexibility in displaying user-provided data sets and its quick response time. However, the validity of the displayed comparisons requires user evaluation. For example, the user must interpret a colocalization of histone H3 acetylation (H3ac) with Usf1 transcription factor binding as either biologically meaningful or experimental artifact (**Fig. 2a**).

Next-generation genome browsers. Newer and higher throughput genomic technologies, including NGS, have enabled researchers to generate unprecedented amounts of data. International consortia—for example the Encyclopedia of DNA Elements (ENCODE) project[37], the Cancer Genome Atlas (TCGA) project[38], the 1,000 Genome Project and Epigenome Roadmap Project (http://nihroadmap.nih.gov/epigenomics/)—each will produce thousands of genome-wide data sets. Even comparatively small groups of researchers are now able to obtain large volumes of genomic data over a short time period. A new generation of genome browsers and associated databases are emerging to efficiently manage and distribute this high volume of data (**Table 2**).

The traditional web-based genome browsers use a centralized model whereby both data and service are located on the server side. Information flows from data providers to the genome browser server, which renders the requested image and passes it to the end user. When the size of the data set increases to a critical point, the substantial overhead burdens the server and internet connections and ultimately disrupts smooth genome browsing.

Decentralizing the data, the service or a combination of the two can relieve such server load. For example, JBrowse[39] uses Asynchronous JavaScript and XML (AJAX) technology to distribute work between the server and client, thereby incurring substantially less server overhead while also replacing traditional static image loading with smoothly animated genome navigation and track selection. Anno-J[40] (Annotation with JavaScript) provides similar smooth Web 2.0 navigation; however, it achieves its client-side rendering using the 'canvas' HTML element, which only some web browsers support. Several other applications use the technology behind Google Maps API to reduce response time on the server's side and create the effect of panning smoothly when navigating through genomic locations[41,42].

Using another approach, UCSC Genome Browser recently improved its popular custom track function by developing BigBed and BigWig formats to handle very large data sets (hundreds of

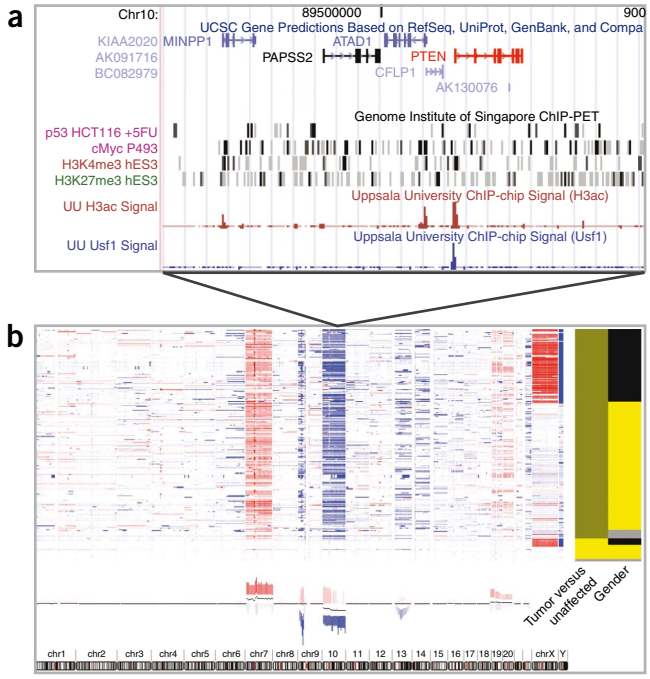

Figure 2 | The UCSC Genome and Cancer Genomics Browsers. (**a**) The UCSC Genome Browser displays diverse data types across the human reference assembly (for example, gene annotations with exons (boxes), introns (thin lines) and untranslated regions (intermediate-height boxes); ChIP data as heat maps or histograms). (**b**) The UCSC Cancer Genomics Browser provides an improved overview and links back to the Genome Browser. Agilent 244A comparative genomic hybridization (CGH) data are taken from randomly selected glioblastoma tumor samples made available through the TCGA consortium, together with a small number of unaffected tissues (blue, deletion; red, insertion). Two publicly available clinical parameters are displayed: tumor (olive) versus unaffected (yellow), and male (yellow) versus female (black); gray, data unavailable.

megabytes to gigabytes of data). Such large data sets are formatted and stored locally on the client computer. Instead of storing the entire data set in the browser's database, the browser only fetches a slice of data around the requested genomic locus. Besides improving efficiency, locally stored data also have the distinct advantage of simplifying the steps necessary to secure sensitive data, such as those from individual human subjects. The University of Tokyo Genome Browser, UTGB, is specifically designed for browsing locally stored data in a customized manner. There are also several standalone tools—in particular, two Java-based packages, the Affymetrix Integrated Genome Browser (IGB, pronounced ig-bee) and the Integrative Genomics Viewer (IGV) developed at the Broad Institute.

In addition to experimental data associated with genomic sequences, other types of data, such as clinical information associated with specimens, are often critical in the interpretation of genomic data. Several recently developed genome browsers are designed to provide a platform to integrate large genomic data sets, especially cancer genomic data. These include the UCSC Cancer Genomics Browser[43], the IGV and the Cancer Molecular Analysis Portal developed at the US National Cancer Institute. The main innovation of these new tools is the simultaneous display of genomics data and clinical data. These browsers display a whole-genome-oriented view of genome-wide experimental measurements for individual samples and sets of samples as heat maps. Clinical features are displayed alongside genomic data in a separate heat map. Investigators interact with the browser to order, filter, aggregate and display data according to clinical features, annotated biological pathways or user-edited collections of genes. Statistical analyses can be applied to defined data sets and displayed graphically on the browser.

The UCSC Cancer Genomics Browser uses a heat map view in which the *x* axis represents genomic coordinates and the *y* axis is an ordered stack of genome-wide measurements, each row representing data of one sample. This display makes it easy to identify common patterns across a sample set. For example, the user can clearly identify where a region of chromosome 10 around the *PTEN* locus appears to be deleted recurrently in available brain tumor samples (**Fig. 2b**). Below the genome heat map is a summary view of the same data, where the characteristic copy number variation profile is apparent. A clinical heat map allows researchers to visually examine the relationship between genomic measurements and selected clinical features available to the user on the basis of their authorized level of data access. Rearrangement of the vertical (clinical sample) order in both the clinical and genomic heat maps can be accomplished by simultaneously sorting on the basis of a numerically encoded clinical feature or combination of features. For example, when glioblastoma data are sorted on 'tumor versus unaffected', there is an obvious difference between the genomic content of these two sample types, with the 'normal' samples showing almost no large-scale copy number abnormalities and the tumors rife with them (**Fig. 2b**).

Constraining the data visualization to sequence-based coordinates can be limiting. This is particularly true when visualizing structural variations or long-range interactions between two genomic loci. In addition, global patterns across genomes are often better appreciated in the context of features that do not map to genome coordinates. One recent example is the UCSC Cancer Genomics Browser, in which genomic data are displayed within the context of biological pathways[43]. By organizing the placement of data into sets of genes according to individual pathways as opposed to chromosomal location, users obtain a more robust and biologically meaningful summary of their genomic data across genes that may act in a concerted manner. Anders and colleagues provide another approach, in which genomic data are organized on a Hilbert curve to provide a global overview[44]. In the future, there is great potential in exploring new ways to better navigate the genomic data landscape.

Challenges. Several key challenges in genomics data analysis have emerged in recent years, including issues of data volume, data type and data representation. Several new genome browsers, as introduced above, are available that tackle some of these topics; however, a consensus has not yet been reached. In addition, it will be important that new genome browsers build on the successes of earlier tools, including easy cross-platform access, data and display customization, and the ability to perform on-the-fly computation within the visualization (for example, the BLAT search functionality in the UCSC Genome Browser).

Genome browsers are beginning to interface with sensitive information, and the community is increasingly aware of the challenge of data security. The personal information encoded in genomic DNA, a person's clinical parameters, and other private information require careful protection. Genome browsers should

take advantage of many security systems developed for electronic information to ensure that only authorized investigators can access these data. In addition, these tools can aim to maximize the utility of sensitive data by presenting them in an anonymized form, such as aggregates or summaries, while preventing the extraction of personal information from such aggregates[45].

Comparing genomes

The recent availability of a large number of completely sequenced and assembled genomes has stimulated active research in the field of comparative genomics. This includes the development of algorithms and tools for pairwise and multiple alignment of very long genomic intervals and complete genomes. Among the goals of this work are (i) the identification of functional elements, such as exons or enhancers (reviewed in refs. 46,47), (ii) the study of large-scale rearrangements and evolution of individual genomes[48] and (iii) the alignment of unfinished and reference genomes in the course of assembly and finishing[49]. Visualization of alignment data is critically important for each of these goals but is challenging because of the difficulty of graphically identifying relationships of interest across multiple chromosomes in multiple genomes and over multiple scales. In this section, we review the variety of techniques that have been developed to help investigators navigate sequence conservation between two or more genomes.

Calculation of whole-genome alignment and synteny. A variety of methods exist for pairwise and multiple whole-genome alignment—for example, BLASTZ[50], MULTIZ[51], Shuffle-LAGAN[52], Mercator and MAVID[53], Mauve[54] and symmetric multiple alignment[55]. All these techniques are unified by the common principle of finding the most similar genomic intervals ('anchors'), extending these regions, chaining alignments to make them contiguous, and analyzing rearrangements. After alignment, the next step is to find conserved signals that may indicate potential functional regions. Methods for calculating short conservation signals in alignments range from a simple window-based approach in PipMaker and VISTA[50,56] to the phylogenetic hidden Markov model Phastcons[57,58] and another statistical model, Gumby[59].

Calculation of conserved synteny, defined as the conservation of chromosomal location of multiple genes[60], is based either on the analysis of DNA alignment or bidirectional comparison of genes on orthologous intervals in two genomes. The evolutionary significance of synteny derives from the assumption that the precise order of genes on a chromosome passes down from a common ancestor[60].

Visualization of alignments has been approached at several levels of resolution, supporting different analytical tasks. Graphical representations of synteny at the level of the whole genome are critical for the exploration of genome evolution. Also critical is the ability to 'drill down' from a global representation of conserved synteny to explore a specific region of conservation between two genomes in the context of annotations. In addition, genome assembly and genome model annotation may be served by comparing the neighborhood of an unknown gene to that of its ortholog in a different organism that has a finished or well annotated genome sequence. Below we describe visualization methods used to depict synteny at both the micro and at the macro level (**Table 3**).

Visualization of whole-genome alignments. A wide variety of strategies have been explored for graphically depicting synteny at the level of whole genomes. Two-dimensional 'dot plots', historically used in the analysis of local alignment, have seen a modern resurgence as a powerful way to visualize increasingly available whole-genome alignments (DAGChainer[61], VISTA-Dot MUMmer[62], GenomeMatcher[63] and MEDEA). The genomes of two organisms are represented along the *x* and *y* axes of the plot, with grid lines indicating chromosome boundaries. Points in the plot indicate some measure of alignment, forming 45° lines in conserved regions. Genome rearrangement and duplication are immediately identifiable as, respectively, off-diagonal lines and identical lines stacked horizontally or vertically. DAGChainer[61], the first publicly available tool for generating dot plots, calculates synteny on the basis of a meta-alignment of genes paired by BLAST matches between two organisms. VISTA-Dot offers a dot-plot viewing mode for the browsing of synteny based on whole-genome DNA alignments (**Supplementary Fig. 1**). This tool has an interactive Google Maps–like interface, allowing users to zoom and pan within the plot, as well as to link out from aligned segments to view them in VISTA or in the JGI Genome Browser. Dot plots are useful not just in analyzing synteny between finished genomes but also in genome assembly and finishing. For instance, the OSLay tool[49] automates the increasingly common technique of using a dot plot to align a collection of contigs from an unfinished assembly against a reference assembly and thereby map the target genome.

Global conservation may also be visualized by representing a reference genome using pill-shaped ideograms of chromosomes, banded to indicate regions of alignment with a compared genome. Bands are color-coded to indicate the chromosome of the aligned region on the compared genome. The ideogram representation of genome alignment is a popular choice for custom-generated figures in the publication of newly sequenced genomes[64–66]. Three options are publicly available for automatically generating variations of this visualization given user-supplied genomic data: Cinteny[67], Apollo[68] and MEDEA. The Sybil 'gradient view' uses an innovative visualization in which genes are displayed along a color gradient in the reference genome, with these colors then used to mark the locations of homologs in a set of aligned genomes. The VISTA Synteny Viewer (VSV) (**Supplementary Fig. 2**) uses an ideogram-based depiction of pairwise genome alignments as a navigational tool to select chromosomes in a reference organism for closer inspection.

In comparison to a dot plot, the ideogram representation of synteny loses information about the physical location of aligned regions on the compared genome. However, the use of color in these diagrams makes it very easy to visualize the way in which the compared genome has been redistributed across the reference genome. Furthermore, colored segments in the reference genome can be linked to specific compared loci by drawing lines to smaller glyphs of compared-organism chromosomes. This approach is taken in Apollo, as well as by the PhIGs website[69], which allows users to generate synteny maps from among 45 sequenced fungi and metazoans.

An alternative and aesthetically pleasing approach to depicting genomic synteny has been introduced by Circos[70]. The Circos tool represents two or more genomes as arcs in a single circle. Tracks of a variety of types can be aligned as inner circles along the genomes. Lines cross the middle of the circle connecting aligned regions.

Table 3 | Tools for comparative genomics visualization

Name	Description	Data	URL
Web-based tools			
Cinteny[67]	Three-scale view of synteny calculated from user-specified markers	H	http://cinteny.cchmc.org/
CoGe SynMap[85]	Dot plots from DAGChainer[61] alignments; histograms of synonymous substitutions	H	http://tinyurl.com/synmap/
GenomeMatcher[63]	A rich, mostly dot plot–based viewer displaying alignments and annotation	F,E,G	http://tinyurl.com/genomematcher/
MEDEA*	A Flash-based suite of linked-track, dot-plot and global-synteny viewing tools	C	http://tinyurl.com/broadmedea/
MultiPipMaker[86]	Vertically arranged display of user-supplied multiple alignments	F	http://pipmaker.bx.psu.edu/pipmaker/
PhIGs[69]	Ideogram-style interactive display of orthologs across >75 genomes	H	
UCSC Genome Browser[72]*	Conservation tracks within popular UCSC genome browser	H,F,G	http://genome.ucsc.edu/cgi-bin/hgGateway/
VISTA[87]*	Conservation tracks connected to a variety of analysis tools	H	http://genome.lbl.gov/vista/index.shtml
VSV, VISTA-Dot*	Three-scale viewer for synteny and dynamic, interactive dot plots for whole-genome DNA alignments	H	http://genome.jgi-psf.org/synteny/
Stand-alone tools			
ACT[76]	Linked-track views; annotation track search; stacking of multiple genomes	E,GF,D	http://www.sanger.ac.uk/Software/ACT/
Circos[70]	Circle-graph presentation of synteny; animations for increased dimensionality	C	http://mkweb.bcgsc.ca/circos
CMap[88]	Stacked vertical depictions of arbitrary relations among DNA segments	D,S	http://gmod.org/wiki/CMap
Combo[77]	Dot-plot and linked-track views; integration of annotation in both views	G,F,C	http://tinyurl.com/argo-combo
GBrowse_syn	GMOD[28]* component; highly customizable linked-track view of synteny	D,S	http://gmod.org/wiki/GBrowse_syn
MizBee[71]	Synteny visualized using circular and linked-track views at multiple scales	C	http://mizbee.org/
Sybil[78]	Local and global views of synteny based on BlastP and protein clustering	D	http://sybil.sourceforge.net/
SynBrowse[75]	GMOD[28] component; local synteny based on gene order, orthology or structure	D	http://www.synbrowse.org/
SynView[79]	GMOD[28] component; synteny at different scales with multiple feature tracks	D	http://gmod.org/wiki/SynView

All tools listed are free and are either web-based or available for all three operating systems. The Data column describes the formats accepted for display within each tool: H, only alignment data hosted at the tool's website; F, FASTA format; E, EMBL/GenBank/DDJB format; G, gff format; C, a custom text-based format; D, designed for use with a user-hosted database; S, requires hosting from a user-supplied web server.
*Our recommendations

This circular arrangement reduces the visual confusion that would result from the equivalent linear representation, in which a spider-web of lines connects distant regions in stacked genomes. The tool also supports animation of the alignment such that connections between individual genomes or chromosomes can be viewed in sequence, further reducing visual complexity. A circular genome viewer is also available in MEDEA and MizBee[71].

The dot-plot, ideogram and circular representations discussed above represent strategies for visually presenting conservation at the whole-genome scale. Tools implementing these representations can be used to identify regions of synteny, duplication and translocation between genomes. Upon identifying such regions, investigators need the means to view them at a higher level of resolution and in visual association with annotation data.

Visualization of local conservation. The most straightforward way to visually associate conservation with annotation data is to represent alignments of compared genomes as 'tracks' within a genome browser. This strategy is best exemplified in the 'conservation tracks' in the UCSC Genome Browser[72] and the VISTA Browser[73] (**Fig. 3**). In both cases, pairwise or multiple alignment is represented as a two-dimensional plot in which the *x* axis shows position along the reference genome and conservation scoring in genome-wide multiple alignments is plotted along the *y* axis. In addition, the UCSC browser has tracks of 'chained alignments', shown as different shades of gray[74]. In the case of VISTA tracks, features such as conserved exons, UTRs and noncoding regions are indicated by color in the areas under the curves. VISTA tracks may also be exported for viewing within their respective reference organisms on other genome browsers such as the JGI Genome Browser and the UCSC Genome Browser.

Alignment tracks provide a valuable means of quickly identifying conservation when browsing within an individual genome. However, this representation of conservation does not allow the investigator to view features within both the reference and compared regions of the alignment simultaneously. For this reason, many tools have been developed with the capability to visualize local synteny[67,75,76,77] (**Table 3**). Generally these tools use a common strategy of stacking track-like representations of a reference and one or more compared genomic regions and drawing lines between them to indicate synteny (a 'linked-tracks' representation). Feature tracks, indicating annotations such as gene models or expressed sequence tag (ESTs), may be overlaid above or below the aligned regions, in a manner analogous to that used by genome browsers. This presentation allows the user to visually browse an alignment while maintaining the context of the genomic annotations that describe the content of all regions under investigation. Links connecting conserved regions may be drawn on the basis of genomic alignment, gene orthology, protein cluster membership[78] or even gene model structure[75,79].

GMOD, the Generic Model Organism Database project (http://gmod.org/), including the popular GBrowse genome browser[28], is perhaps the most widely used framework for developing software tools to support genome analysis and curation. Three synteny browsing tools have been developed within the GMOD framework: SynBrowse[75], SynView[79] and GBrowse_syn. SynBrowse, an extension of the GBrowse family of tools[75], allows users to switch among three modes for displaying links between conserved regions. In 'synteny block' mode, regions are connected according to a user-specified definition of synteny (a certain number of collinear genes within a certain minimum distance). In 'coding gene' and 'coding exon' modes, protein alignments are displayed as lines grouping aligned genes and exons, respectively, across the reference and compared segments. Alignment quality is further indicated by the color of each line.

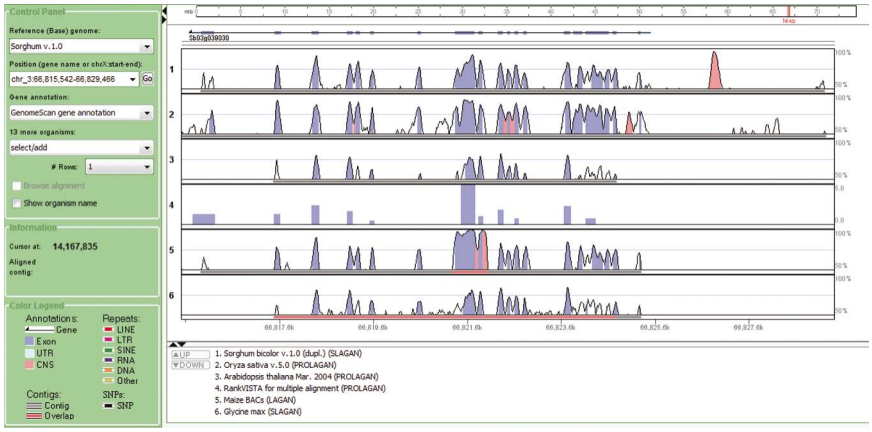

Figure 3 | The VISTA browser. This plot corresponds to a 14-kb interval on the *Sorghum bicolor* v.1.0 assembly (chr. 3, 66815542–66829466). Conserved regions are colored according to the gene annotation displayed above the conservation plot (blue, conservation in exons; light blue, in untranslated regions; pink, in conserved noncoding sequences). Several alignments can be viewed at the same time to assist in analysis. The following VISTA conservation tracks are displayed: (1) duplicated region on *S. bicolor* (chr. 9, 52532014–52544345); (2) *Oryza sativa* in the multiple three-way alignment of *S. bicolor*, *O. sativa* and *Arabidopsis thaliana*; (3) *A. thaliana* in the same three-way alignment; (4) Rank-VISTA plot of the three-way alignment; (5) maize BAC AC198485.2_6; (6) orthologous region in the soybean genome.

Challenges. A variety of representations have been used to visualize synteny at scales ranging from whole-genome alignment to the conservation of intron/exon structure in regions of preserved gene order. A major challenge in the further development of these tools is to provide a means for the investigator to navigate seamlessly across these levels of resolution. Fortunately, the increasing sophistication of web application technology enables ever-greater interactivity and the ability to connect visual elements to informational resources on the internet. The VSV takes advantage of these technologies by presenting a novel interface to unify scales in the display of synteny (**Supplementary Fig. 2**). The VSV depicts synteny in three cross-navigable panels representing different scales of the alignment. Both the Combo[77] tool and GenomeMatcher[63] bridge levels of resolution in the visualization of synteny by connecting an interactive dot plot with a 'linked-track' view of local conservation. MizBee[71], released very recently, provides interactive side-by-side views of the data across the range of scales supporting exploration of all of these relationship types.

Most of the tools described above follow a model of aligning one or more 'compared' genomes against a single 'reference' genome. This model, although seemingly necessitated by visual tractability, is limiting in that the relationships among compared organisms cannot be explored. One approach to address this limitation, taken in both the Artemis Comparison Tool[76] and the CMAP application, is to allow the user to stack genomes so that an arbitrary set of pairwise comparisons can be visualized (although for a given genome it is still possible to compare it to at most two others). Another drawback of the 'reference genome' model for displaying synteny is that the *x* axis for the entire alignment is usually defined by position along the reference genome, potentially obscuring interesting features in the compared sequences. Two tools, Phylo-VISTA[80] and SynPlot[81], implement visualizations of conservation in which position is depicted relative to the length of the overall alignment.

Still another challenge in the visualization of synteny is the graphical representation of insertions and deletions ('indels'), which are critical to tracking genome evolution at the chromosome, gene family and gene structure scales. Although many alignment algorithms are capable of identifying indels, most synteny viewers offer no means of indicating them visually, displaying only correspondence between conserved regions. To our knowledge, only the GBrowse_syn viewer allows for the visualization of indels. When 'grid-lines' are enabled in GBrowse_syn, an indel is represented by grid lines connecting an insertion region on one genome to the single point of deletion on another.

Perspective

Many successful visualization tools are carefully tailored to the specialized analysis demands of their users, and it is unlikely that a universal tool for genomics analysis is feasible or desirable. There is, however, a great need to improve the integration among tools and ease the transition from one analysis to another. Rapid advances in sequencing technologies continue to strain existing software and challenge developers to anticipate future requirements. The paradigms of more mature tools, both in terms of computational approaches and visual representations, struggle to scale to today's data demands. More recent tools address some of the core issues, but they often sacrifice richness of functionality to satisfy the urgent needs for speed and for ease of distribution. It is likely that widespread integration between tools will only be realized once we acquire greater stability in the data generation technologies and file format standards.

We have highlighted several widely used tools to guide a researcher wishing to tackle genome analyses today. However, given the pace at which this relatively young field is evolving, it is very likely that new software tools will emerge and revised file formats will be proposed in the near future. As a consequence of this dynamic nature, the potential for innovation in this domain is great.

To meet future analysis demands, visualization tools first need to successfully integrate diverse data forms, such as clinical information together with genomic data. Second, these tools require visual representations that scale smoothly to comparisons of thousands or millions of elements. For example, the track-based displays used by current genome browsers will not readily support the output of the 1,000 Genomes Project. Third, advances in this domain will require the seamless navigation across relevant levels of resolution, taking advantage of aggregation methods to reveal global trends and interactive interfaces to provide user access to details at lower levels on demand. And fourth, improved integration between automated computation and visualization will be necessary to allow users to interactively refine and iterate their analyses. This type of integration will also enable a broader biology community to perform genome-wide analyses, rather than these studies being limited to computational specialists.

Note: Supplementary information is available on the Nature Methods website.

ACKNOWLEDGMENTS

Thanks to Y. Butterfield, S. Diguistini, P. Gorniak, M. Krzywinski, N. Liao, G. Robertson and G. Taylor for helpful discussions and comments. C.B.N. was funded with US federal funds from the National Cancer Institute, National Institutes of Health (NIH), under contract no. NO1-CO-12400. The contributions of M.C. and I.D. were performed under the auspices of the US Department of Energy's Office of Science, Biological and Environmental Research Program, and by the University of California, Lawrence Livermore National Laboratory under contract no. DE-AC52-07NA27344, Lawrence Berkeley National Laboratory under contract no. DE-AC02-05CH11231, and Los Alamos National Laboratory under contract no. DE-AC02-06NA25396. D.G. was supported by NIH grants R01 HL094976 and 1RC2HL10296-01 and by the Howard Hughes Medical Institute. T.W. was supported by funds from the Helen Hay Whitney Foundation.

COMPETING INTERESTS STATEMENT

The authors declare no competing financial interests.

Published online at http://www.nature.com/naturemethods/.
Reprints and permissions information is available online at http://npg.nature.com/reprintsandpermissions/.

1. Pop, M. Genome assembly reborn: recent computational challenges. *Brief. Bioinform.* **10**, 354–366 (2009).
2. Flicek, P. & Birney, E. Sense from sequence reads: methods for alignment and assembly. *Nat. Methods* **6** (suppl.), S6–S12 (2009).
3. Gordon, D., Abajian, C. & Green, P. Consed: a graphical tool for sequence finishing. *Genome Res.* **8**, 195–202 (1998).
A widely used finishing tool that was the first to use error probabilities as an objective criterion to guide the finishing process.
4. Ewing, B., Hillier, L., Wendl, M.C. & Green, P. Base-calling of automated sequencer traces using phred. I. Accuracy assessment. *Genome Res.* **8**, 175–185 (1998).
5. Ewing, B. & Green, P. Base-calling of automated sequencer traces using phred. II. Error probabilities. *Genome Res.* **8**, 186–194 (1998).
6. Schatz, M.C., Phillippy, A.M., Shneiderman, B. & Salzberg, S.L. Hawkeye: an interactive visual analytics tool for genome assemblies. *Genome Biol.* **8**, R34 (2007).
7. Salzberg, S.L., Church, D., DiCuccio, M., Yaschenko, E. & Ostell, J. The Genome Assembly Archive: a new public resource. *PLoS Biol.* **2**, E285 (2004).
8. Li, H. *et al.* The Sequence Alignment/Map (SAM) format and SAMtools. *Bioinformatics* **25**, 2078–2079 (2009).
9. Mardis, E.R. Next-generation DNA sequencing methods. *Annu. Rev. Genomics Hum. Genet.* **9**, 387–402 (2008).
10. Turner, D.J., Keane, T.M., Sudbery, I. & Adams, D.J. Next-generation sequencing of vertebrate experimental organisms. *Mamm. Genome* **20**, 327–338 (2009).
11. Gordon, D., Desmarais, C. & Green, P. Automated finishing with autofinish. *Genome Res.* **11**, 614–625 (2001).
12. Dear, S. & Staden, R. A sequence assembly and editing program for efficient management of large projects. *Nucleic Acids Res.* **19**, 3907–3911 (1991).
13. Bonfield, J.K., Smith, K.F. & Staden, R. A new DNA sequence assembly program. *Nucleic Acids Res.* **23**, 4992–4999 (1995).
One of the first and a widely used finishing tool with an interactive graphical user interface and sequence editing capabilities. An updated version (Gap5) is designed to handle NGS data.
14. Burland, T.G. DNASTAR's Lasergene sequence analysis software. *Methods Mol. Biol.* **132**, 71–91 (2000).
15. Parsons, J.D. Miropeats: graphical DNA sequence comparisons. *Comput. Appl. Biosci.* **11**, 615–619 (1995).
16. Medvedev, P., Stanciu, M. & Brudno, M. Computational methods for discovering structural variation with next-generation sequencing. *Nat. Methods* **6** (suppl.), S13–S20 (2009).
17. Huang, W. & Marth, G. EagleView: a genome assembly viewer for next-generation sequencing technologies. *Genome Res.* **18**, 1538–1543 (2008).
18. Bao, H. *et al.* MapView: visualization of short reads alignment on a desktop computer. *Bioinformatics* **25**, 1554–1555 (2009).
19. Manske, H. & Kwiatkowski, D. LookSeq: a browser-based viewer for deep sequencing data. *Genome Res.* **19**, 2125–2132 (2009).
20. Kim, P.-G., Cho, H.-G. & Park, K. A scaffold analysis tool using mate-pair information in genome sequencing. *J. Biomed. Biotechnol.* **2008**, 675741 (2008).
21. Zerbino, D.R. & Birney, E. Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res.* **18**, 821–829 (2008).
22. Chaisson, M.J. & Pevzner, P.A. Short read fragment assembly of bacterial genomes. *Genome Res.* **18**, 324–330 (2008).
23. Hernandez, D., François, P., Farinelli, L., Osterås, M. & Schrenzel, J. De novo bacterial genome sequencing: millions of very short reads assembled on a desktop computer. *Genome Res.* **18**, 802–809 (2008).
24. MacCallum, I. *et al.* ALLPATHS 2: small genomes assembled accurately and with high continuity from short paired reads. *Genome Biol.* **10**, R103 (2009).
25. Nielsen, C.B., Jackman, S.D., Birol, I. & Jones, S.J. ABySS-Explorer: visualizing genome sequence assemblies. *IEEE Trans. Vis. Comput. Graph.* **15**, 881–888 (2009).
26. *C. elegans* Sequencing Consortium. Genome sequence of the nematode *C. elegans*: a platform for investigating biology. *Science* **282**, 2012–2018 (1998).
27. Eeckman, F.H. & Durbin, R. ACeDB and macace. *Methods Cell Biol.* **48**, 583–605 (1995).
28. Stein, L.D. *et al.* The generic genome browser: a building block for a model organism system database. *Genome Res.* **12**, 1599–1610 (2002).
The Generic Model Organism Database project is the most widely used framework for developing software tools to support genome analysis and curation. Three synteny-specific tools have been developed within the GMOD framework: SynBrowse, SynView and GBrowseSyn.
29. Lander, E.S. *et al.* Initial sequencing and analysis of the human genome. *Nature* **409**, 860–921 (2001).
30. Kent, W.J. *et al.* The human genome browser at UCSC. *Genome Res.* **12**, 996–1006 (2002).
Widely used genome browser with user-friendly web interface and capability to display third party data.
31. Birney, E., Bateman, A., Clamp, M.E. & Hubbard, T.J. Mining the draft human genome. *Nature* **409**, 827–828 (2001).
32. Stalker, J. *et al.* The Ensembl web site: mechanics of a genome browser. *Genome Res.* **14**, 951–955 (2004).
33. Wheeler, D.L. *et al.* Database resources of the National Center for Biotechnology. *Nucleic Acids Res.* **31**, 28–33 (2003).
34. Cline, M.S. & Kent, W.J. Understanding genome browsing. *Nat. Biotechnol.* **27**, 153–155 (2009).
35. Furey, T.S. Comparison of human (and other) genome browsers. *Hum. Genomics* **2**, 266–270 (2006).
36. Giardine, B. *et al.* Galaxy: a platform for interactive large-scale genome analysis. *Genome Res.* **15**, 1451–1455 (2005).
37. ENCODE Project Consortium. *et al.* Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. *Nature* **447**, 799–816 (2007).
38. Cancer Genome Atlas Research Network Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature* **455**, 1061–1068 (2008).
39. Skinner, M.E., Uzilov, A.V., Stein, L.D., Mungall, C.J. & Holmes, I.H. JBrowse: a next-generation genome browser. *Genome Res.* **19**, 1630–1638 (2009).
40. Lister, R. *et al.* Highly integrated single-base resolution maps of the epigenome in *Arabidopsis*. *Cell* **133**, 523–536 (2008).
41. Yates, T., Okoniewski, M.J. & Miller, C.J. X:Map: annotation and visualization of genome structure for Affymetrix exon array analysis. *Nucleic Acids Res.* **36** Database issue, D780–D786 (2008).
42. Arakawa, K. *et al.* Genome Projector: zoomable genome map with multiple views. *BMC Bioinformatics* **10**, 31 (2009).
43. Zhu, J. *et al.* The UCSC Cancer Genomics Browser. *Nat. Methods* **6**, 239–240 (2009).
44. Anders, S. Visualization of genomic data with the Hilbert curve. *Bioinformatics* **25**, 1231–1235 (2009).
45. Homer, N. *et al.* Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS Genet.* **4**, e1000167 (2008).
46. Ureta-Vidal, A., Ettwiller, L. & Birney, E. Comparative genomics: genome-wide analysis in metazoan eukaryotes. *Nat. Rev. Genet.* **4**, 251–262 (2003).
47. Freeling, M. & Subramaniam, S. Conserved noncoding sequences (CNSs) in higher plants. *Curr. Opin. Plant Biol.* **12**, 126–132 (2009).
48. *Drosophila* 12 Genomes Consortium. *et al.* Evolution of genes and genomes on the *Drosophila* phylogeny. *Nature* **450**, 203–218 (2007).
49. Richter, D.C., Schuster, S.C. & Huson, D.H. OSLay: optimal syntenic layout of unfinished assemblies. *Bioinformatics* **23**, 1573–1579 (2007).
50. Schwartz, S. *et al.* Human-mouse alignments with BLASTZ. *Genome Res.* **13**, 103–107 (2003).
51. Blanchette, M. *et al.* Aligning multiple genomic sequences with the threaded blockset aligner. *Genome Res.* **14**, 708–715 (2004).
52. Brudno, M. *et al.* Glocal alignment: finding rearrangements during alignment. *Bioinformatics* **19** (suppl. 1), i54–i62 (2003).

53. Dewey, C.N. Aligning multiple whole genomes with Mercator and MAVID. *Methods Mol. Biol.* **395**, 221–236 (2007).
54. Darling, A.C.E., Mau, B., Blattner, F.R. & Perna, N.T. Mauve: multiple alignment of conserved genomic sequence with rearrangements. *Genome Res.* **14**, 1394–1403 (2004).
55. Dubchak, I., Poliakov, A., Kislyuk, A. & Brudno, M. Multiple whole-genome alignments without a reference organism. *Genome Res.* **19**, 682–689 (2009).
56. Frazer, K.A., Pachter, L., Poliakov, A., Rubin, E.M. & Dubchak, I. VISTA: computational tools for comparative genomics. *Nucleic Acids Res.* **32** (Web Server issue), W273–W279 (2004).
A comprehensive suite of programs and databases for comparative analysis of genomic sequences. Whole-genome alignments of many species from different taxa (vertebrates to prokaryotes) and tools for custom analysis of user-submitted sequences are provided.
57. Siepel, A. *et al.* Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res.* **15**, 1034–1050 (2005).
58. Karolchik, D. *et al.* Comparative genomic analysis using the UCSC genome browser. *Methods Mol. Biol.* **395**, 17–34 (2007).
59. Prabhakar, S. *et al.* Close sequence comparisons are sufficient to identify human *cis*-regulatory elements. *Genome Res.* **16**, 855–863 (2006).
60. Gregory, S.G. *et al.* A physical map of the mouse genome. *Nature* **418**, 743–750 (2002).
61. Haas, B.J., Delcher, A.L., Wortman, J.R. & Salzberg, S.L. DAGchainer: a tool for mining segmental genome duplications and synteny. *Bioinformatics* **20**, 3643–3646 (2004).
62. Kurtz, S. *et al.* Versatile and open software for comparing large genomes. *Genome Biol.* **5**, R12 (2004).
63. Ohtsubo, Y., Ikeda-Ohtsubo, W., Nagata, Y. & Tsuda, M. GenomeMatcher: a graphical user interface for DNA sequence comparison. *BMC Bioinformatics* **9**, 376 (2008).
64. Mouse Genome Sequencing Consortium. *et al.* Initial sequencing and comparative analysis of the mouse genome. *Nature* **420**, 520–562 (2002).
65. Galagan, J.E. *et al.* Sequencing of *Aspergillus nidulans* and comparative analysis with *A. fumigatus* and *A. oryzae*. *Nature* **438**, 1105–1115 (2005).
66. Putnam, N.H. *et al.* Sea anemone genome reveals ancestral eumetazoan gene repertoire and genomic organization. *Science* **317**, 86–94 (2007).
67. Sinha, A.U. & Meller, J. Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. *BMC Bioinformatics* **8**, 82 (2007).
A flexible web-based tool allowing investigators to view synteny at the level of whole genomes, individual pairs of chromosomes, or regions around markers of interest, which can be uploaded by the user.
68. Lewis, S.E. *et al.* Apollo: a sequence annotation editor. *Genome Biol.* **3**, RESEARCH0082 (2002).
69. Dehal, P.S. & Boore, J.L. A phylogenomic gene cluster resource: the Phylogenetically Inferred Groups (PhIGs) database. *BMC Bioinformatics* **7**, 201 (2006).
70. Krzywinski, M. *et al.* Circos: an information aesthetic for comparative genomics. *Genome Res.* **19**, 1639–1645 (2009).
71. Meyer, M., Munzner, T. & Pfister, H. MizBee: a multiscale synteny browser. *IEEE Trans. Vis. Comput. Graph.* **15**, 897–904 (2009).
72. Miller, W. *et al.* 28-way vertebrate alignment and conservation track in the UCSC Genome Browser. *Genome Res.* **17**, 1797–1808 (2007).
73. Dubchak, I. Comparative analysis and visualization of genomic sequences using VISTA browser and associated computational tools. *Methods Mol. Biol.* **395**, 3–16 (2007).
74. Kent, W.J. *et al.* Evolution's cauldron: duplication, deletion, and rearrangement in the mouse and human genomes. *Proc. Natl. Acad. Sci. USA* **100**, 11484–11489 (2003).
75. Brendel, V., Kurtz, S. & Pan, X. Visualization of syntenic relationships with SynBrowse. *Methods Mol. Biol.* **396**, 153–163 (2007).
76. Carver, T. *et al.* Artemis and ACT: viewing, annotating and comparing sequences stored in a relational database. *Bioinformatics* **24**, 2672–2676 (2008).
77. Engels, R. *et al.* Combo: a whole genome comparative browser. *Bioinformatics* **22**, 1782–1783 (2006).
78. Crabtree, J., Angiuoli, S.V., Wortman, J.R. & White, O.R. Sybil: methods and software for multiple genome comparison and visualization. *Methods Mol. Biol.* **408**, 93–108 (2007).
79. Wang, H., Su, Y., Mackey, A.J., Kraemer, E.T. & Kissinger, J.C. SynView: a GBrowse-compatible approach to visualizing comparative genome data. *Bioinformatics* **22**, 2308–2309 (2006).
80. Shah, N. *et al.* Phylo-VISTA: interactive visualization of multiple DNA sequence alignments. *Bioinformatics* **20**, 636–643 (2004).
81. Göttgens, B. *et al.* Long-range comparison of human and mouse SCL loci: localized regions of sensitivity to restriction endonucleases correspond precisely with peaks of conserved noncoding sequences. *Genome Res.* **11**, 87–97 (2001).
82. Stothard, P. & Wishart, D.S. Circular genome visualization and exploration using CGView. *Bioinformatics* **21**, 537–539 (2005).
83. Shannon, P.T., Reiss, D.J., Bonneau, R. & Baliga, N.S. The Gaggle: an open-source software system for integrating bioinformatics software and data sources. *BMC Bioinformatics* **7**, 176 (2006).
84. Nicol, J.W., Helt, G.A., Blanchard, S.G. Jr., Raja, A. & Loraine, A.E. The Integrated Genome Browser: free software for distribution and exploration of genome-scale data sets. *Bioinformatics* **25**, 2730–2731 (2009).
85. Lyons, E. *et al.* Finding and comparing syntenic regions among *Arabidopsis* and the outgroups papaya, poplar, and grape: CoGe with rosids. *Plant Physiol.* **148**, 1772–1781 (2008).
86. Elnitski, L., Riemer, C., Burhans, R., Hardison, R. & Miller, W. MultiPipMaker: comparative alignment server for multiple DNA sequences. *Curr. Protoc. Bioinformatics* **Ch. 10**, unit 10.14 (2005).
87. Mayor, C. *et al.* VISTA: visualizing global DNA sequence alignments of arbitrary length. *Bioinformatics* **16**, 1046–1047 (2000).
88. Youens-Clark, K., Faga, B., Yap, I.V., Stein, L. & Ware, D. CMap 1.01: a comparative mapping application for the Internet. *Bioinformatics* **25**, 3040–3042 (2009).