Nucleome Browser: An integrative and multimodal data navigation platform for 4D Nucleome

(SUPPLEMENTAL INFORMATION)

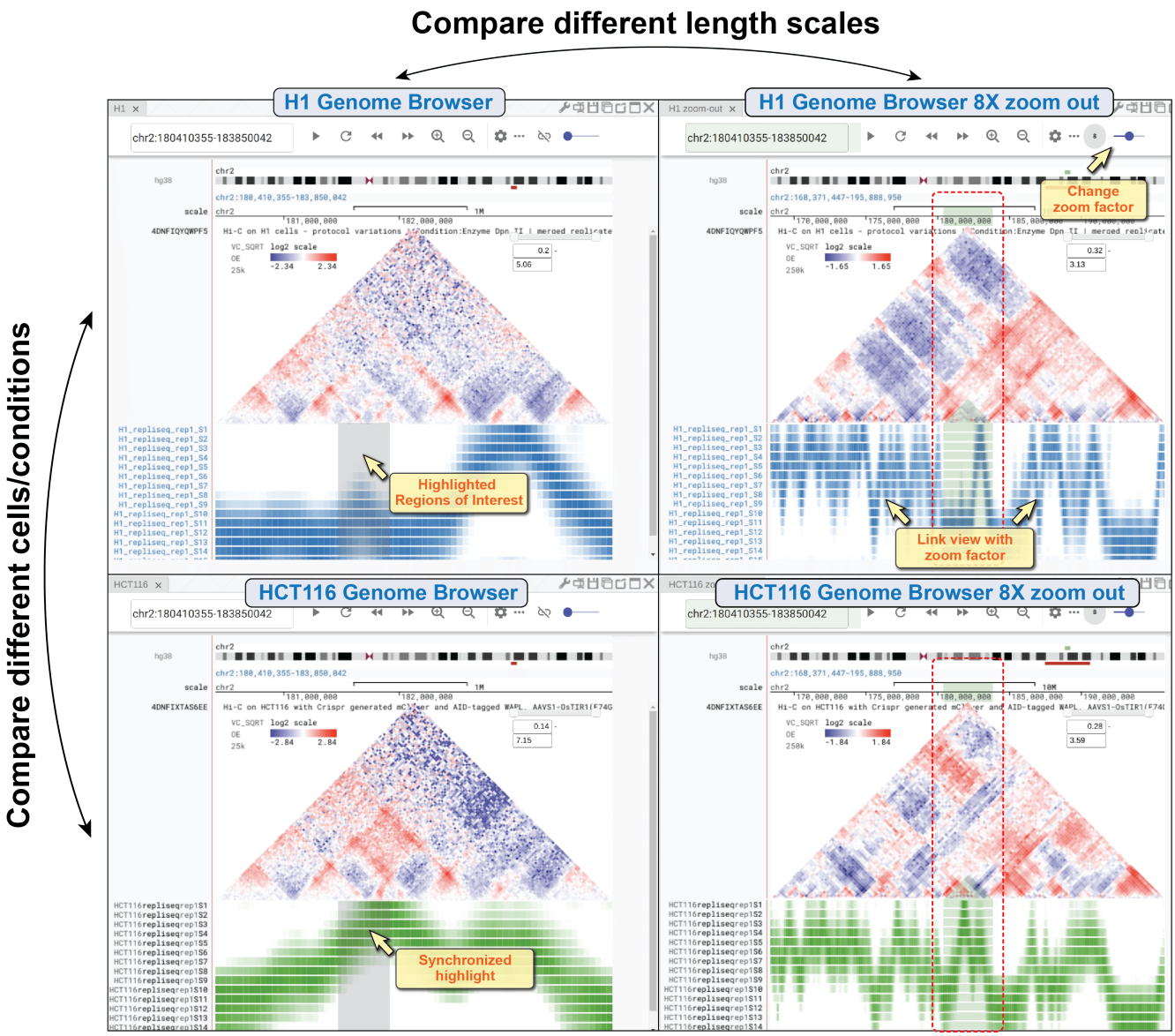

Figure S1: Nucleome Browser allows users to create multiple linked panels (e.g., genome browser panels) for interactive exploration. Users can synchronously explore and compare genomic data as shown by the two panels on the left, where an interactive highlighted region indicates an H1 specific early replicated region as compared to HCT116. The two panels on the right are similar to their counterparts on the left except that these panels are in the context mode. In particular, these two panels on the right show the zoomed-out view (8 times) compared to the panels on the left, enabling users to compare genomic data at different scales. Users can use the zoom factor selection tool to easily change zoom factor (from 1x to 64x).

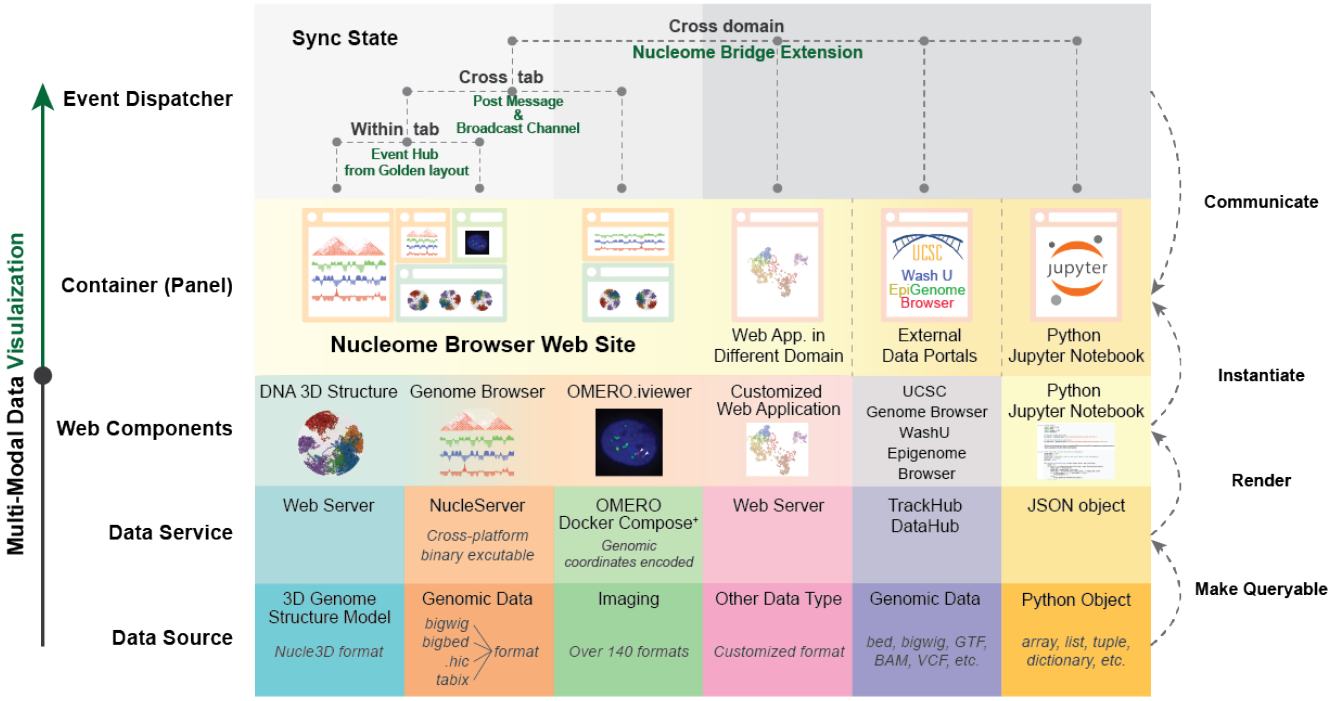

Figure S2: The overall design of Nucleome Browser: from data service to synchronization of web components. This diagram shows the key concepts as well as major functionalities introduced in Nucleome Browser to overcome the major challenges of multimodal data visualization for 4D Nucleome research. Each row of the diagram represents a layer of data processing step from bottom to top. Six columns refer to three main modalities (genomics, imaging, and 3D genome structure model) currently represented in Nucleome Browser, customized data type, external data portal, and Python Jupyter Notebook. From bottom to top, it shows the procedure of using data service to process different data types and various front-end to render and visualize these data types. The top two rows are the major components of the adaptive communication system. Web components are represented by composable web panels. We use different web APIs to allow communication between web components in different configurations. Importantly, we also developed Nucleome Bridge web browser extension to communicate with external web-based data portals.

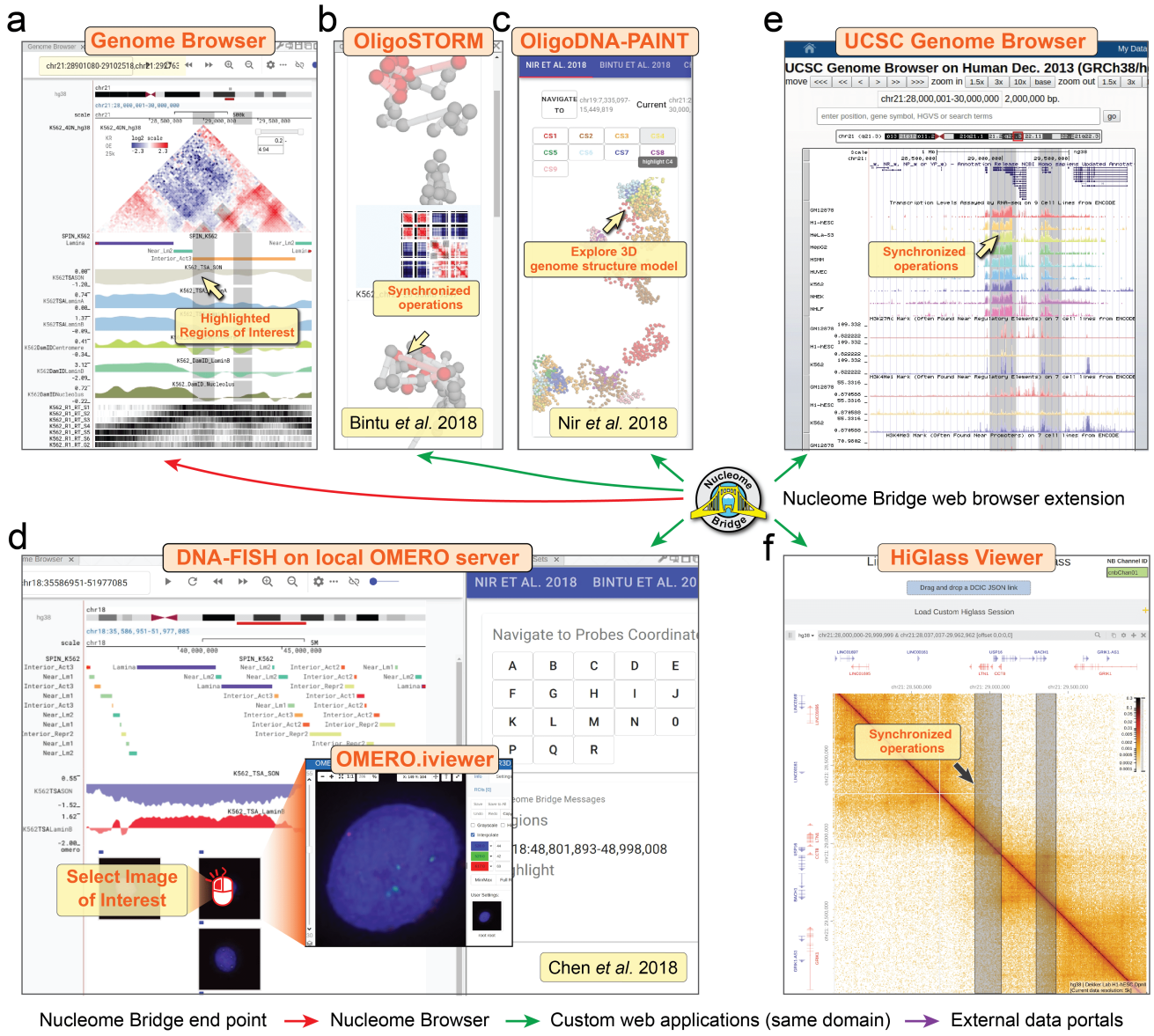

Figure S3: Nucleome Browser allows users to build customized web applications to visualize new data types and analysis results together with datasets available in Nucleome Browser. **(a)** When users navigate the genome browser in Nucleome Browser, customized web applications can update their content synchronously. **(b)** This web application shows the 3D genome structure model and 2D distance matrix derived from hundreds of cells using OligoSTORM data (from [1]). Users can select a cell to visualize its 3D model and 2D distance matrix. Probes highlighted in the genome browser in panel a) are marked in red colors in 3D models and highlighted in the 2D distance matrix as well. **(c)** Users can interactively explore 3D genome structure models (spheres indicate probes) derived from OligoDNA-PAINT data (from [2]) and genomic data in Nucleome Browser. Clicking a sphere can navigate to the genomic location of this probe. **(d)** Users can click the button to jump to the probe location in the genome browser on the left. Clicking a thumbnail on the image track will bring up an OMERO.iviewer to show the image data. **(e-f)** Nucleome Browser supports synchronized exploration with external data portals and custom web applications via the new adaptive communication mechanism. When users navigate to or highlight region(s) on the genome browser panel in Nucleome Browser, the connected external data portals (e.g., UCSC Genome Browser, custom web applications, and HiGlass viewer) will update accordingly.

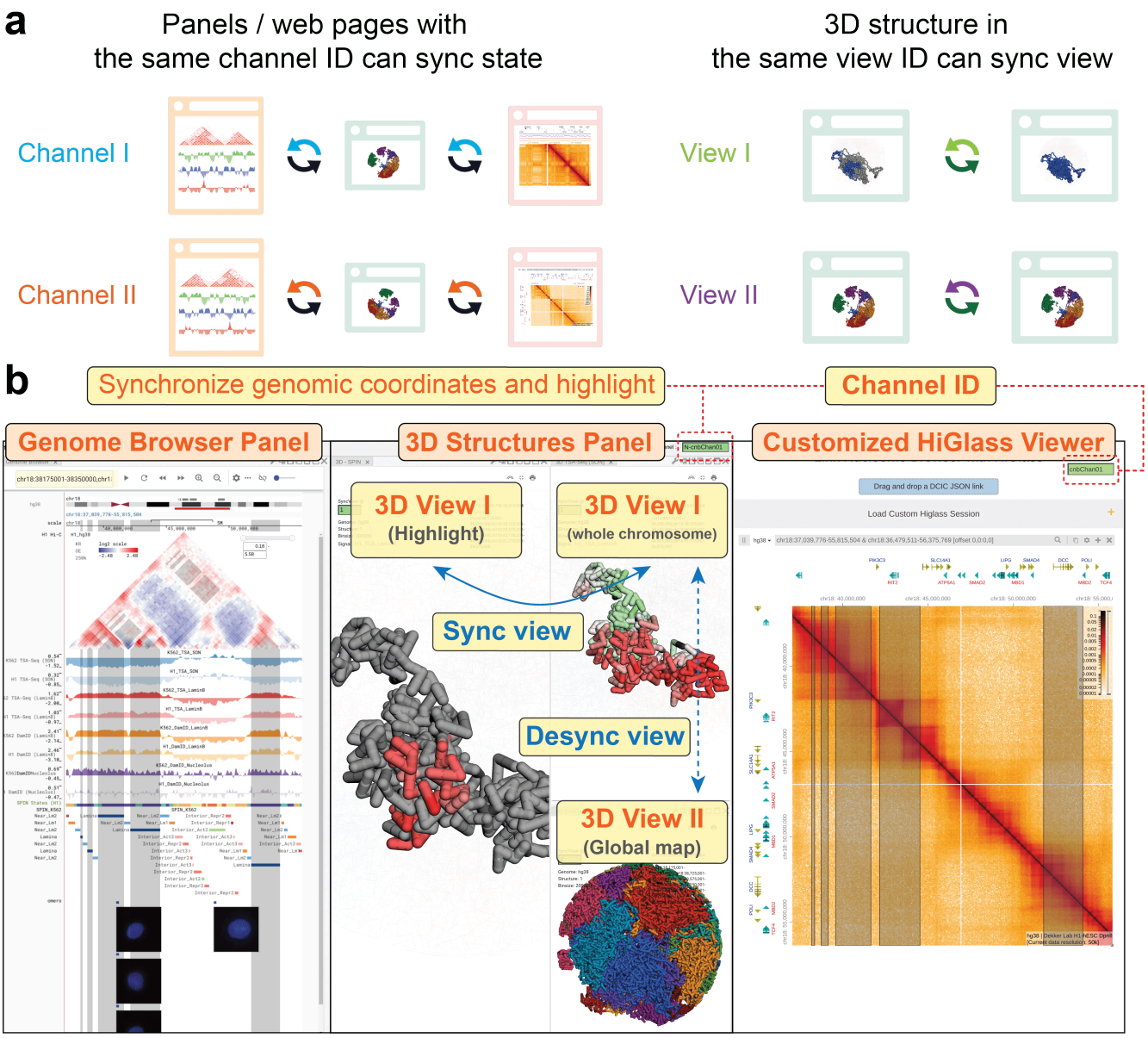

Figure S4: Nucleome Browser supports multi-channel communication. **(a)** An illustration of synchronization across panels and 3D genome structure with multi-channel or multi-view communication. Left: panels can only synchronize their operation (e.g., navigation and highlight) to other panels with the same communication channel ID. Right: For the same 3D genome structure, panels with the same view ID can synchronize the rotation and zooming of the 3D structure. **(b)** A screenshot shows the application of the multi-channel and multi-view communication mechanism. The genome browser panel shares the same channel ID with a custom HiGlass viewer page so that the current region and the highlighted regions are synchronized between them. The bottom panel has a view ID different from two 3D genome structure panels on the top. Users can use the bottom panel to view the global reference map of the nucleus and use the other 3D genome structure panel to visualize details such as the highlighted regions and the current chromosome.

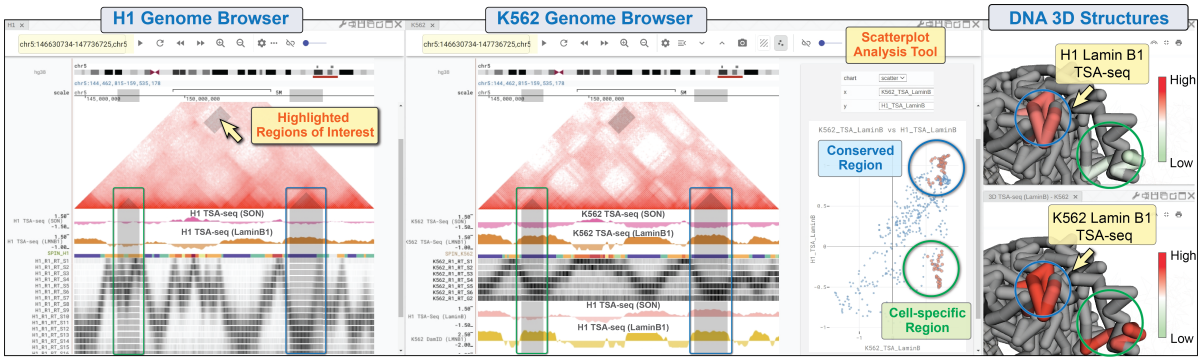

Figure S5: Nucleome Browser facilitates comparisons across multiple conditions. This screenshot shows a comparison between H1 and K562 cells on chromosome 5. Highlighted regions show a cell type-specific LAD/late-replicating (K562) to inter-LAD/middle-replicating (H1) transition (green rectangle on the left) and a conserved LAD (blue rectangle on the right) as indicated with TSA-seq, DamID, and high-resolution replication timing Repli-seq tracks. The scatterplot tool and 3D structures further corroborate with the observations from the genome browser panels.

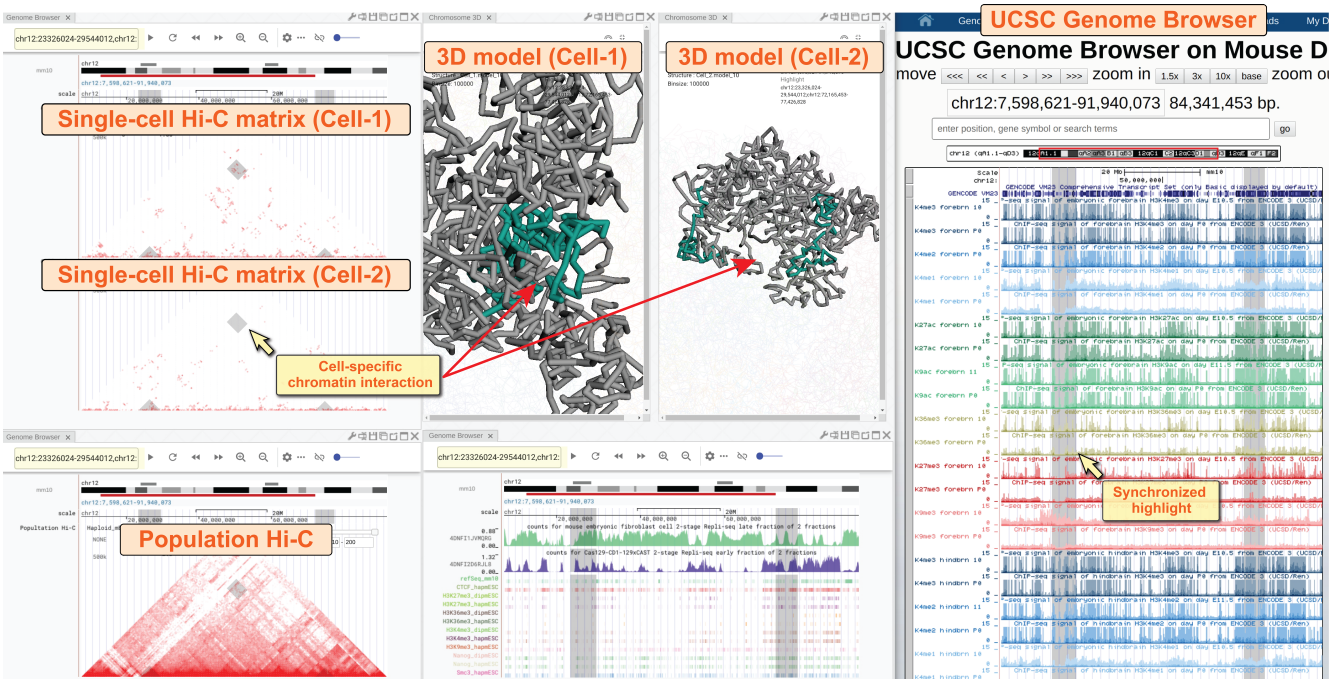

Figure S6: An illustration of exploring mouse single-cell Hi-C datasets together with the inferred 3D genome structure models. Two single-cell Hi-C contact maps are shown on the top-left. Hi-C contact map derived from bulk Hi-C is shown on the bottom-left. The 3D genome structures inferred from Hi-C data in these two cells are shown on the center-top, together with ChIP-seq data tracks (center-bottom). The corresponding UCSC Genome Browser view on the same region is shown on the right. Note that the highlighted regions on all the panels including the external UCSC Genome Browser are synchronized, which greatly facilitates multimodal data navigation and comparison.

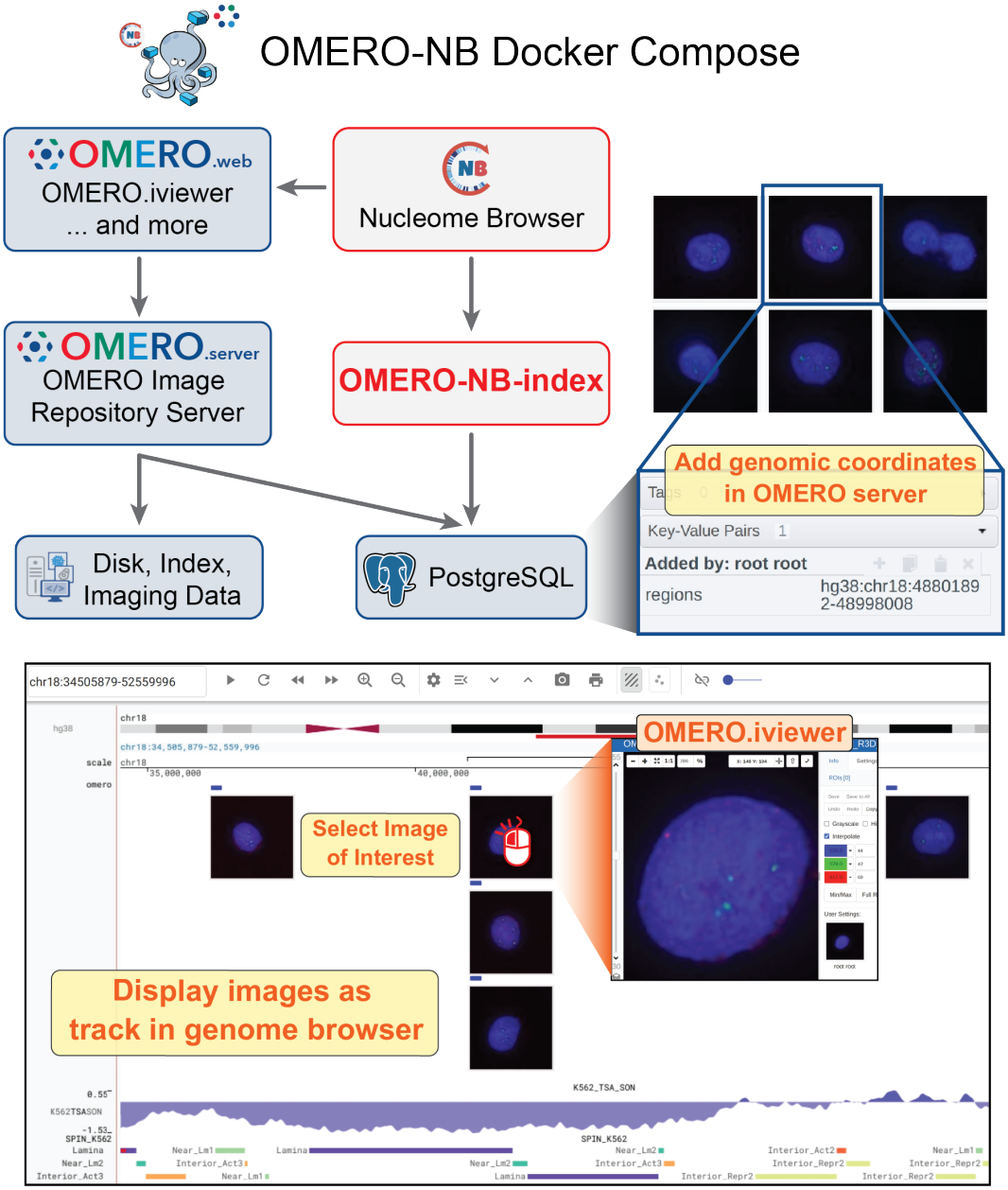

Figure S7: Integration between Nucleome Browser and OMERO image server. Top: We built OMERO-NB-index, a specific plugin of OMERO server to link OMERO server with Nucleome Browser. Users can add genomic coordinates associated with images as key:value annotations in the OMERO server. OMERO-NB-index then starts a web server for efficient query of imaging datasets based on genomic coordinates. Bottom: On the genome browser panel in Nucleome Browser, the thumbnails of images are rendered as tracks with red bars indicating the locations of probes. Users then can click a thumbnail to explore the imaging data using the OMERO.iviewer.

Features	UCSC Genome Browser [3]	Ensembl Genome Browser [4]	WashU Epigenome Browser [5]	IGV & IGV.js [6]	JBrowse [7]	3D Genome Browser [8]	GIVE [9]	Juicebox & Juicebox.js [10]	HiGlass [11]	Nucleome Browser
Basic functions										
Multiscale visualization	***	***	***	***	***	***	***	***	***	***
Gene annotation	***	***	***	***	**	***	*	**	***	***
bigBed/bigWig track	***	***	***	***	***	***	*	***	***	***
4D Nucleome data types										
Intra-chromosomal interaction (Hi-C)	**	-	*	-	-	**	*	***	***	***
Inter-chromosomal interaction (Hi-C)	-	-	-	-	-	*	*	***	***	***
3D genome structure model	-	-	*	-	-	-	-	-	-	***
Integration and interactive exploration of imaging data with other data types										
Integrating imaging data from OMERO server	-	-	-	-	-	-	-	-	-	***
Integrating imaging data from Image Data Resource (IDR)	-	-	-	-	-	-	-	-	-	***, **a**
Interactive exploration										
Chromosome grid	-	-	-	-	-	-	-	***	***	**
Highlight regions on 1D genomic track using mouse	***	-	-	-	-	-	-	-	-	***, **b**
Highlight regions on 2D genomic track using mouse	-	-	-	-	-	-	-	-	*	***
Synchronization between windows (navigation/highlight)	-	-	-	-	-	-	-	**	***	***, **c**
Linked windows with variable zoom factors	-	-	-	-	-	-	-	-	*	***
Synchronization between genome browser and 3D structure	-	-	-	-	-	-	-	-	-	***
Super-impose bigWig and bigBed on 3D structure	-	-	-	-	-	-	-	-	-	***
Interactive scatterplot analysis tool	-	-	-	-	-	-	-	-	-	***, **d**
Interoperability with other data portal & web services & web applications										
Session URL sharing	***	***	***	*	***	-	-	***	***	***
Set up local service	**	-	**	***	**	-	***	***	***	***
Connect local service to main server	-	-	-	-	-	-	-	-	-	***
Interactive data analysis with Jupyter Notebook	-	-	-	-	-	-	-	-	-	***
Interactive exploration with external web-based portals	-	-	-	-	-	-	-	-	-	***

Table S1: Comparing different features between Nucleome Browser with existing genome browsers and 3D genome visualization tools. The support of different tools for each feature is ranked by number of stars from weak to strong. '-' means that this feature is not supported by that tool. **a.** Nucleome Browser can link to IDR to interactively explore data (e.g., the *in situ* genome sequencing (IGS) data) with genome browser. As users highlight regions in the genome browser panels, probes overlapping with highlighted regions are marked on image. WashU Epigenome Browser only supports query image data by gene name based on the description of image datasets in IDR. **b.** Nucleome Browser allows users to highlight genomic regions with the same annotation based on bigBed track, which is not supported on UCSC Genome Browser. **c.** Nucleome Browser allows users to synchronize both navigation and highlight while HiGlass and Juicebox only support navigation. **d.** Nucleome Browser allows users to select data points and highlight corresponding regions on the genome browser. Highlighted regions on the genome browser are also marked in the scatterplot in real time. WashU Epigenome Browser only supports generating static scatterplot without interactive exploration.

Name	Description	Link
NucleServer	A Cross-platform tool to launch genomic/3D data service	https://github.com/nucleome/nucleserver
NucleData	The GUI version of NucleServer	https://github.com/nucleome/nucledata
nb-dispatch	A JavaScript Package of the event-dispatch system	https://github.com/nucleome/nb-dispatch
Nucleome Bridge	A Web browser extension for synchronization with UCSC Genome Browser, WashU Epigenome Browser and other customized apps	https://tinyurl.com/nb-bridge
OMERO-NB-index	A custom add-on to monitor the image data hosted by the OMERO server and launch a data service for images	https://github.com/nucleome/omero2cnb
Nucle3D	The file format used to represent 3D genome structure models in the Nucleome Browser	https://github.com/nucleome/nucle3d
NucleTools	The utility scripts for converting data formats for 3D genome structure web component	https://github.com/nucleome/nucle
HiGlass viewer page	A customized web page to link HiGlass displays with the Nucleome Browser	https://vis.nucleome.org/static/apps/higlass
Nucleome Gallery	The web portal to publish Nucleome Browser sessions	https://gallery.nucleome.org
Video tutorial	The ultimate guide on how to use Nucleome Browser	https://tinyurl.com/nb-video-tutorial
Documentation	Documentation of Nucleome Browser	https://nb-docs.readthedocs.io

Table S2: Summary of tools/packages/websites created in Nucleome Browser

Experiment type	Genome	Processed file count
2-stage Repli-seq	human	74
ATAC-seq	human	18
Capture Hi-C	human	28
ChIP-seq	human	24
DamID	human	44
Dilution Hi-C	human	267
DNase Hi-C	human	12
in situ Hi-C	human	766
Micro-C	human	88
Multi-stage Repli-seq	human	83
pa-DamID	human	216
PLAC-seq	human	11
RNA-seq	human	84
TSA-seq	human	44
2-stage Repli-seq	mouse	74
ATAC-seq	mouse	15
Capture Hi-C	mouse	9
ChIP-seq	mouse	126
DamID	mouse	6
Dilution Hi-C	mouse	28
DNase Hi-C	mouse	7
in situ Hi-C	mouse	224
Micro-C	mouse	8
PLAC-seq	mouse	7
RNA-seq	mouse	29

Table S3: Summary of the public 4DN data hosted in Nucleome Browser (synchronized with the 4DN Data Portal (https://data.4dnucleome.org/). Our latest version was updated in November 2021.

Data set	Data type	Reference or URL
4DN public genomic data	Genomic tracks	https://data.4dnucleome.org
Single-cell Hi-C	Genomic tracks	[12]
4DN public imaging data	Imaging	https://data.4dnucleome.org
in situ genome sequencing (IGS)	Imaging	https://idr.openmicroscopy.org/webclient/?show=project-2054, [13]
DNA-FISH	Imaging	[14]
OligoSTORM	3D genome structure model	[1]
OligoDNA-PAINT	3D genome structure model	[2]
Structure models derived from Hi-C data	3D genome structure model	[15]

Table S4: Summary of datasets used in this manuscript

A Supplementary Note: Comparison to Existing Tools

Here we describe the conceptual advances and technical innovations in Nucleome Browser. Nucleome Browser aims to solve the daunting challenge of building an efficient platform capable of integrating emerging data types into a single web application to facilitate integrative data analysis, hypothesis generation, and data sharing. We highlight the important advances of Nucleome Browser compared to existing tools.

First, Nucleome Browser achieves efficient visualization of multimodal, heterogeneous datasets within a single platform. For studying nuclear organization, integrating genomic and imaging data together with 3D genome structure models is critically important. However, no existing tool can integrate multimodal datasets for efficient exploration. For example, both the widely used genome browsers (e.g., UCSC Genome Browser [3] and WashU Epigenome Browser [5, 16]) and the platforms specifically designed for visualizing 2D contact maps (e.g., 3D Genome Browser [8], Juicebox [17], Juicebox.js [10], and HiGlass [11]) have limited support to interactively explore 1D and 2D genomic signals together with 3D genome structure models. Notably, even for the same data type, linking and communicating between multiple windows with the same or different zoom factors, which is highly useful to explore genomic loci far away from each other at multiple scales, is not fully supported by existing tools. More recently, multiplexed genome imaging methods [1, 2, 13, 18, 19] have shown great potentials to directly measure the spatial position of chromatin in single cells. However, no visualization tool can allow users to view imaging data with results derived from the corresponding genomic mapping assays side-by-side. Nucleome Browser directly addressed such multimodal data visualization challenge in 4D Nucleome research by directly integrating genomic and imaging data as well as 3D genome structures into a unified platform.

Second, Nucleome Browser has a unique adaptive communication mechanism that allows different components to work synergistically for more interactive and integrative visualization. In Nucleome Browser, different data modalities and data sources are represented as distinct web components, with synchronized operations, for efficient data exploration and also for effective development and sharing. Our multi-channel hierarchical communication scheme (in the form of the nb-dispatch library) also allows users to create their own web application with the capability to communicate with the rest of the web components in Nucleome Browser. This design is conceptually different from Juicebox and HiGlass that only allow messages transmission across multiple windows with the same type. Specifically, we designed three features for synchronization across heterogeneous web components. (1) Both navigation and highlight operations can automatically broadcast across different data modalities (genomics, imaging, and 3D genome structure models). (2) Direct comparison between genomic signals and 3D genome structure models is achieved by superimposing bigWig and bigBed data onto the 3D models with a color-coding display. (3) We created Nucleome Bridge, a web browser extension, to allow Nucleome Browser to communicate with the external data sources or customized web applications, e.g., UCSC Genome Browser and WashU Epigenome Browser. As both data portals host abundant genome annotations, users can use Nucleome Browser to control the navigation on these external web-based tools and explore unique resources provided by them (e.g., the ENCODE data at UCSC Genome Browser and the Roadmap Epigenomics data at WashU Epigenome Browser).

Lastly, Nucleome Browser supports data queries from distributed web locations and achieves secure private data visualization together with public resources. As the amount of data generated from an

individual lab grows rapidly, the demand for hosting private servers is common. Several teams have provided their server-less solution to allow users to efficiently visualize data without heavy programming skills [6, 9, 20]. In Nucleome Browser, we developed a cross-platform data service tool to host private data. The only required configuration input is an Excel file or the ID of a publicly-accessible Google Sheet. Once the data service is launched, users can import their private data by adding the address of the data service in Nucleome Browser. Our data service tool can automatically fetch both remote data (e.g., bigBed and bigWig data) stored online and those as local files. Nucleome Browser drastically reduces the learning curve for users to host private data, requiring less computational power on the server end, and more effectively integrating with other datasets.

Overall, Nucleome Browser is an unique solution for visualizing multimodal data and also the emerging new data types for 4D Nucleome research.

B Supplementary Note: Tutorials for using Nucleome Browser

B.1 Tutorial I: Exploring 4DN datasets in Nucleome Browser

Here we show how to use Nucleome Browser to explore the 4DN public data hosted on the main Nucleome Browser portal (at Carnegie Mellon University). First, visit the Nucleome Browser by typing http://vis.nucleome.org in the web browser (we recommend using the Chrome web browser for the best experience). You can then choose to open a saved session or a new genome browser panel after you click the "Start" button on the home page.

Once you open a genome browser panel, you can search and add 4DN data tracks in the configuration page by clicking the panel-configuration button on the top-right corner of the panel as shown in **Supplementary Fig.** S8. We have added 4DN public data to the list of default data services; otherwise, you can add it following **Supplementary Fig.** S8. By default, 4DN datasets are grouped by the experiment type and data format. Once you have selected a category of datasets, you can add tracks by clicking the "selection" button. Note that we have manually curated track names to make them easy to understand. You can click the "exclamation button" to toggle a short description of the track. Clicking the "Read more" will lead you to a web page describing this data on the 4DN DCIC data portal. You can also use the search toolbox to search track by the track name. If one experiment has multiple processed data, we add the experiment ID to the track name. You can copy this experiment ID and use the search box to search all tracks associated with this experiment.

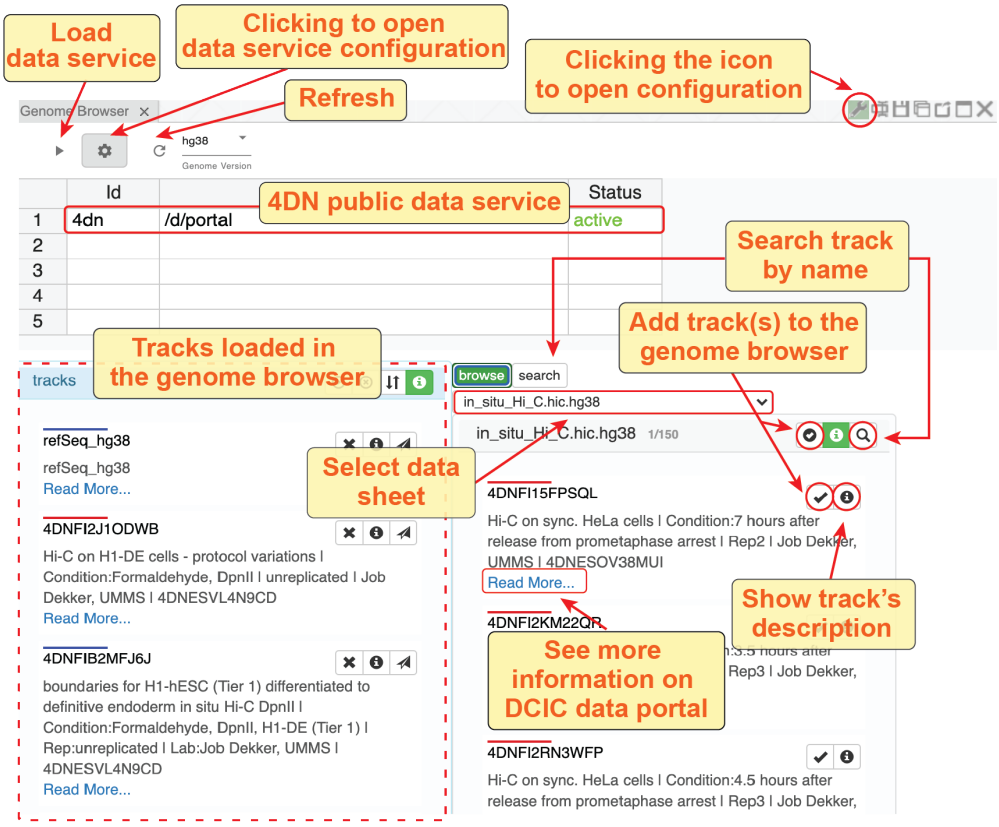

Figure S8: The track configuration interface allows users to load public 4DN datasets in the genome browser panel.

Some microscopy imaging data hosted on the OMERO server in the 4DN DCIC data portal are

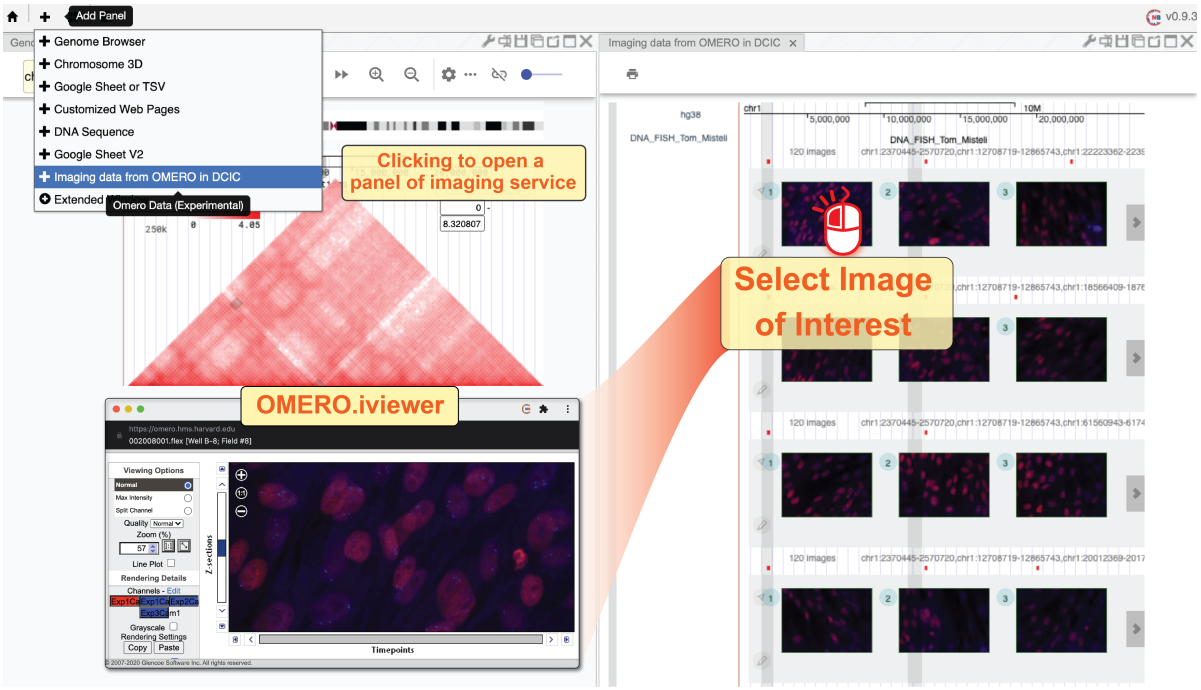

Figure S9: Viewing 4DN imaging data using the DCIC imaging data web component in Nucleome Browser.

associated with genomic coordinates information such as the HIPMap data from [21]. To view these datasets, we have compiled a specific web component called "Imaging data from OMERO in DCIC" (**Supplmentary Fig.** S9). In this panel, each track represents one image dataset and individual images from different cells are shown as thumbnails. You can use the arrow keys on each side to browse the thumbnails. The red bars in the track indicate the targeted genomic regions of each experiment from the FISH probes. You can click the thumbnail of an image and you will see a pop-out window leading to an OMERO.iviewer to further examine the image data.

B.2 Tutorial II: Important features of web components to facilitate integrative exploration

In this tutorial, we introduce important features of different types of web components and how to interactively use them for the visualization of mutlimodal datasets.

B.2.1 Genome Browser web component for genomic data

We developed several useful features to take full advantage of the composable nature of our web components, which represents significant advances over existing genome browsers.

(i) Three visualization modes: In Nucleome Browser, users can open multiple browser panels (web components with the capability to communicate with each other) simultaneously and customize each panel. In particular, we developed three visualization modes to control the synchronization across multiple panels (**Supplemenetary Fig.** S10). (1) Default mode: Both navigation and highlight of region-of-interest synchronize automatically across all genome browser panels. (2) Solo mode: This mode allows users to turn off synchronization in a specific panel, enabling examination of different regions on the same web page. (3) Context mode: This is a variant of the default mode by associating an adjustable zoom factor K. In this mode, the current region-of-interest will zoom out K times centered on the region viewed in other panels under the default mode. Therefore, users can explore genomic signals at multiple scales by arranging multiple components side-by-side, one in the default mode and others in the context mode, which is particularly useful for exploring genomic data at region-of-interest and its local context simultaneously.

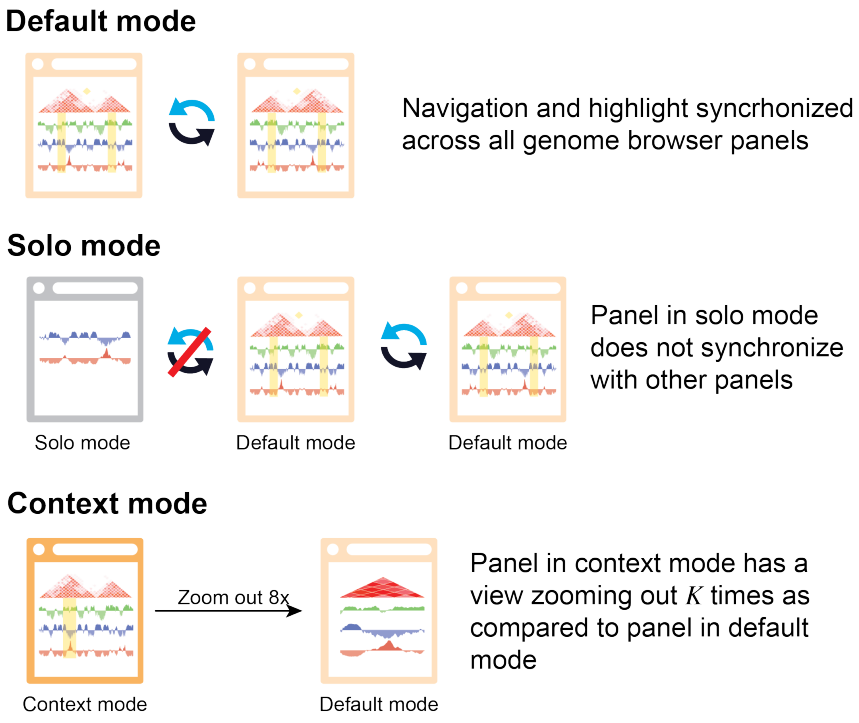

Figure S10: Different visualization modes of the genome browser web component. In the default mode, users can use multiple panels, with synchronized explorations, to compare multiomic datasets from different samples. Sole mode allows users to turn off synchronization in one panel, enabling comparison of the same dataset at different genomic loci. Context mode further assigns a zoom factor to each panel such that users can simultaneously explore genomic signals at different scales.

(ii) Scatterplot analysis tool: We developed a scatterplot analysis tool in the genome browser web com-

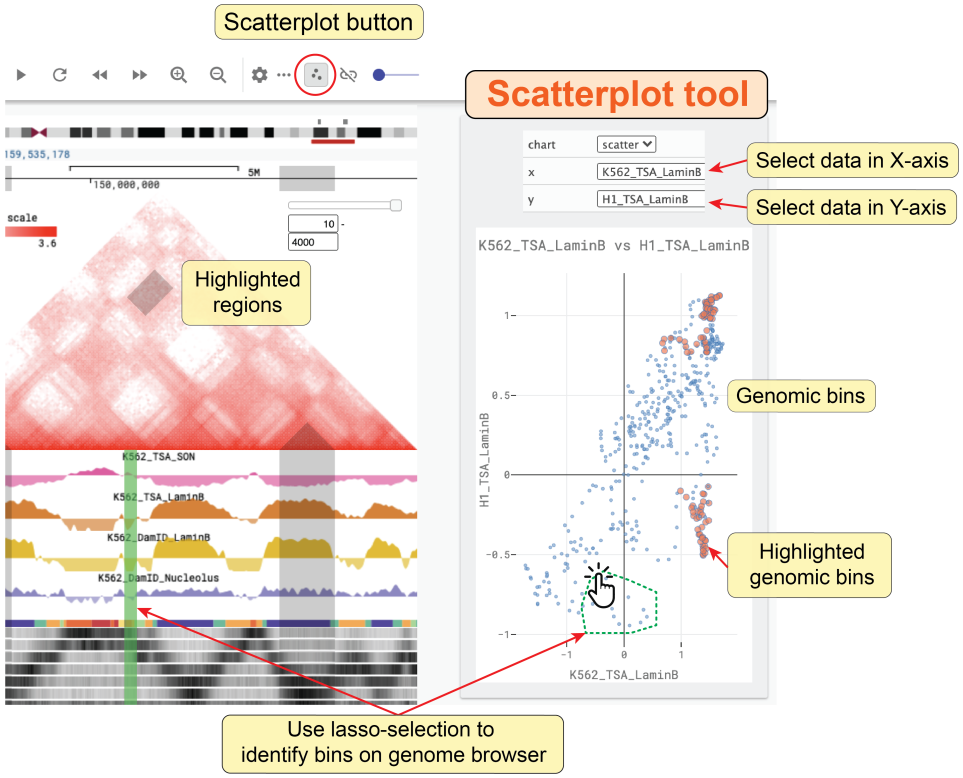

Figure S11: Scatterplot analysis tool in genome browser web component. Users can select tracks to show genomic signals on the X-axis and Y-axis of a scatterplot. Highlighted dots (red color) on the scatterplot will be synchronized with the highlighted regions (grey polygon)on the genomic tracks. Users can also use the lasso selection tool to manually pick data points on the scatterplot to view where these genomic bins locate in the genome browser (green polygon).

ponent allowing users to compare genomic tracks quantitatively (**Supplementary Fig.** S11). In the scatterplot between two genomic tracks, users can mouse-over dots on the scatterplot to view signal values of two genomic tracks. As users navigate along the genome, the dots would update synchronously. Notably, users can use selection tools to select a group of data points on the scatterplot and the corresponding genomic regions will be highlighted on the genome browser tracks. This operation is reciprocal, i.e., users can also highlight regions on data tracks and view them on the scatterplot.

(iii) Use drag-and-drop to transfer genomic data to other web components: We developed a drag-and-drop feature to efficiently share genomic data with other types of web components (**Supplementary Fig.** S12). In the configuration tab of the genome browser panel, users can drag-and-drop a bigBed or bigWig track to a 3D genome structure panel and render the color of 3D genome structure model based on the genomic signals. This new feature offers an effective way to interactively compare genomic signals with 3D genome structures.

Together, these new capabilities in the genome browser web component in Nucleome Browser can greatly improve user experience of exploring multiomic data tracks.

B.2.2 3D genome structure model web component

We developed several unique features to achieve effective exploration of 3D genome structure models.

(i) Modular design: Similar to the genome browser web component, we encapsulate the 3D genome structure web component into composable and communicable panel, allowing users to open multiple panels simultaneously and visualize the same or different structures with customized settings in each panel.

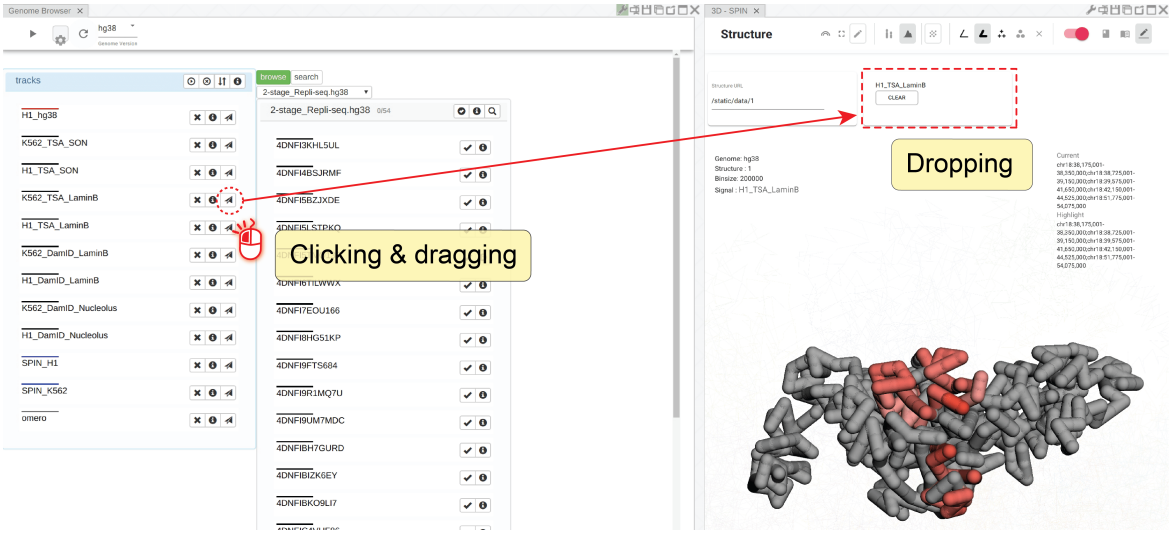

Figure S12: Dragging and dropping bigWig track from genome browser panel to 3D genome structure panel to superimpose its values as a color-coding display on the 3D genome structure.

(ii) Synchronized view with multi-channel communication mechanism: Users can assign different communication channels to each 3D genome structure panel (**Supplementary Fig.** S4). When users visualize the same structure in multiple panels, only structures in panels with the same communication channel ID can synchronize their views including rotation, size, and position of 3D genome structure models. This novel feature is particularly useful when users want to compare different genomic signals superimposed on the same 3D genome structure or examine different spatial contexts of the same genomic region.

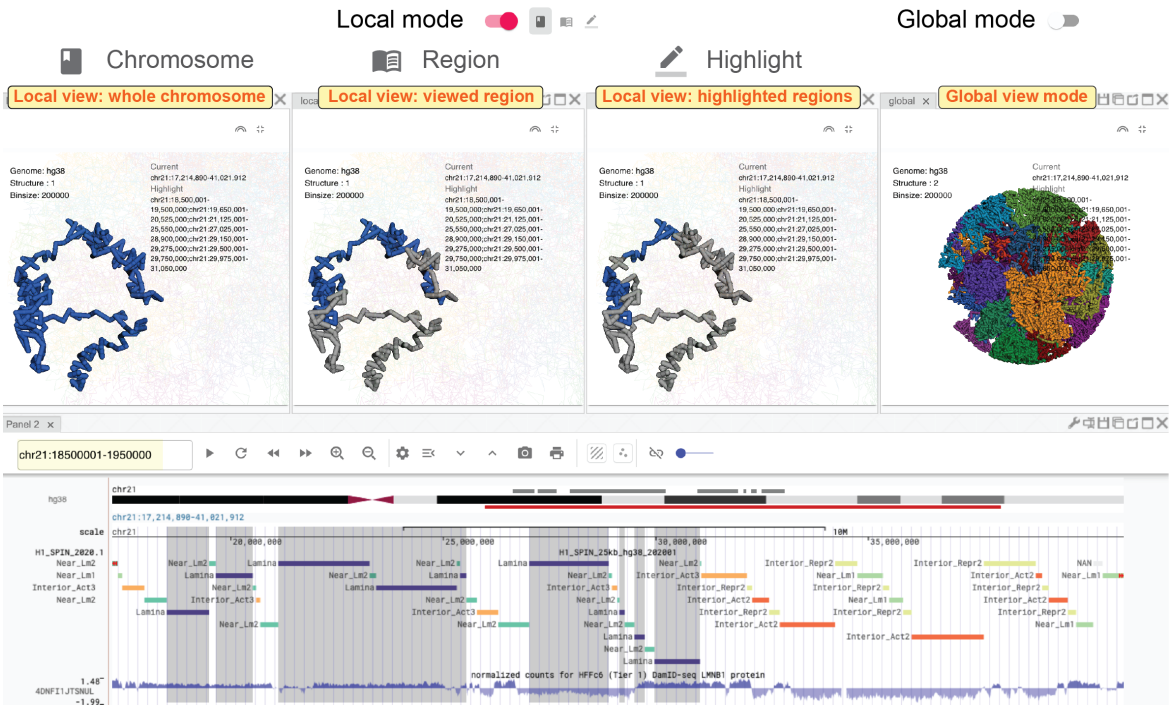

Figure S13: In the local model, users can choose different visualization modes to control how 3D models are highlighted relative to the currently viewed region on the genome browser panel. The global mode displays all chromosomes as a global navigable map.

(iii) Integration with genome browser panel: The 3D genome structure web component can synchronize its content with information from other web components, e.g., the genome browser web component.

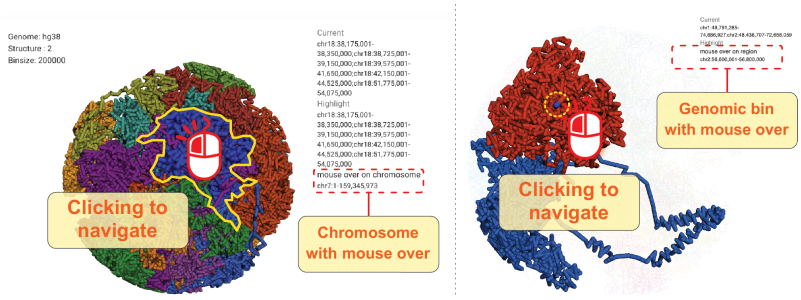

Figure S14: Using the global 3D reference panel to navigate the genome. Users can use mouse to select a chromosome to navigate to that chromosome (left). Alternatively, users can use mouse to select a genomic bin to navigate when 3D genome structure model is shown in local mode (right).

Both navigation and highlighted regions are synchronized between the genome browser web component and the 3D genome structure web components. Users can choose three different types of spatial context relative to the currently viewing genomic region, including currently viewing region, highlighted regions, or chromosomes currently being viewed (**Supplementary Fig.** S13).

Notably, a unique design is that the operation is also reciprocal, meaning that users can click segments on a 3D genome structure model to navigate the corresponding genomic regions in the genome browser panel as well (**Supplementary Fig.** S14).

In summary, the 3D genome structure model web component allows users to not only effectively explore multiple 3D genome structures at different scales but also perform integrative visualization with genomic data, imaging data, and customized web applications.

B.2.3 Region-of-interest web component

The region-of-interest web component works as a convenient tool for users to examine multiple region-of-interest such as ChIP-seq peaks without manually searching for regions (**Supplementary Fig.** S15). Users only need to compile a list of genomic coordinates in a Google Sheet or save them as a tabular text file (tab-delimited) on a web server. This web component then fetches genomic coordinates in the sheet/file, stores them as a local data structure using the binning index algorithm, and renders them as a genomic track. Users can then navigate regions using mouse by clicking each region.

Figure S15: Users can link Nucleome Browser with tabular data to efficiently navigate region-of-interest.

B.3 Tutorial III: Connecting with external data portals and custom web applications

Nucleome Browser supports not only private data hosted locally by users but it also allows users to use their data hosted on external data portals. Here, we demonstrate how to use the Nucleome Bridge web browser extension to connect Nucleome Browser with UCSC Genome Browser and customized web applications.

B.3.1 Connecting Nucleome Browser with UCSC Genome Browser

After installation of the Nucleome Bridge web extension in Google Chrome web browser (download link: https://tinyurl.com/nb-bridge), users can click the button of this extension on the web browser to open a synchronized UCSC Genome Browser window. When users highlight some regions or navigate to different regions in the genome browser panel in Nucleome Browser, the UCSC Genome Browser window will jump to or highlight the same regions, accordingly (**Supplementary Fig.** S16). Note that the operation is reciprocal, i.e., the view of the Nucleome Browser will also synchronize with the view on the UCSC Genome Browser.

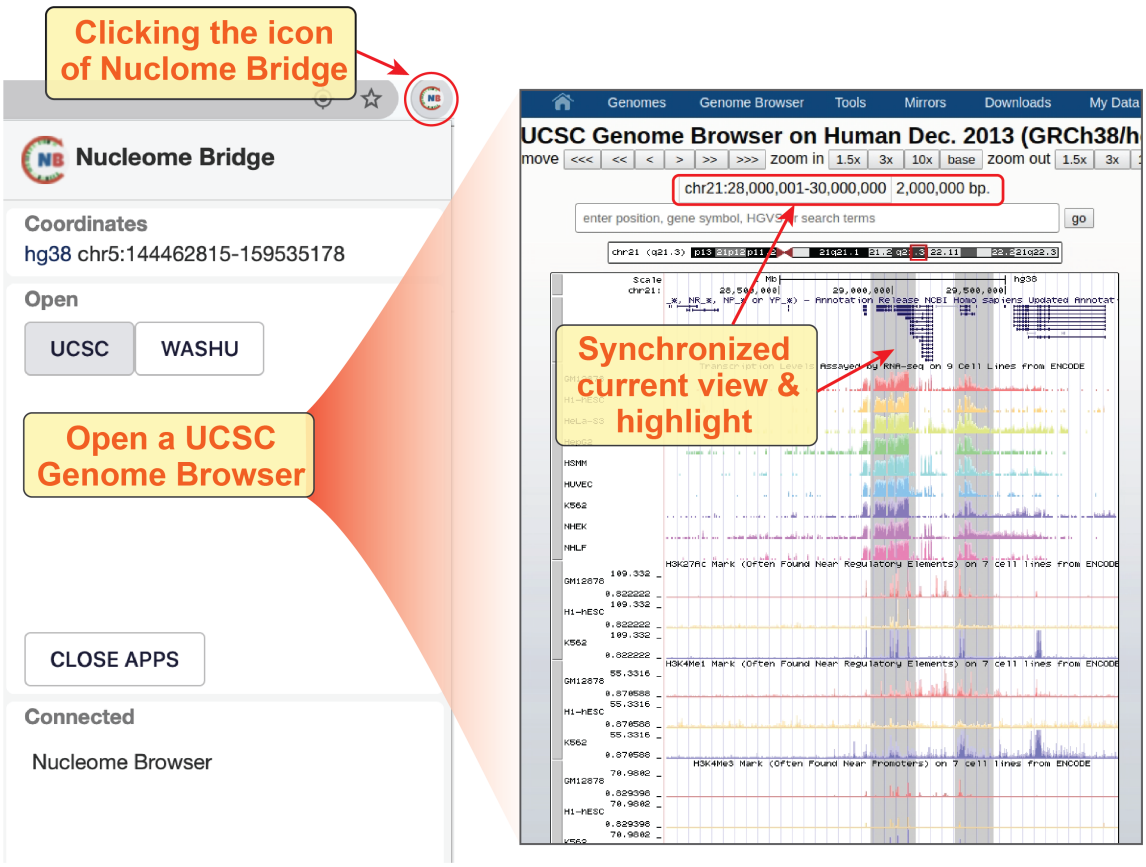

Figure S16: Nucleome Bridge web browser extension allows users to connect Nucleome Browser with UCSC Genome Browser, WashU Epigenome Browser, and other web applications.

B.3.2 Connecting Nucleome Browser with Image Data Resource (IDR)

We built a customized web application to demonstrate that users can interactively explore imaging datasets hosted by IDR together with multimodal datasets in Nucleome Browser as shown in **Fig.** 1d. To interactively explore this example, you can go to http://paper.nucleome.org in a web browser. Click-

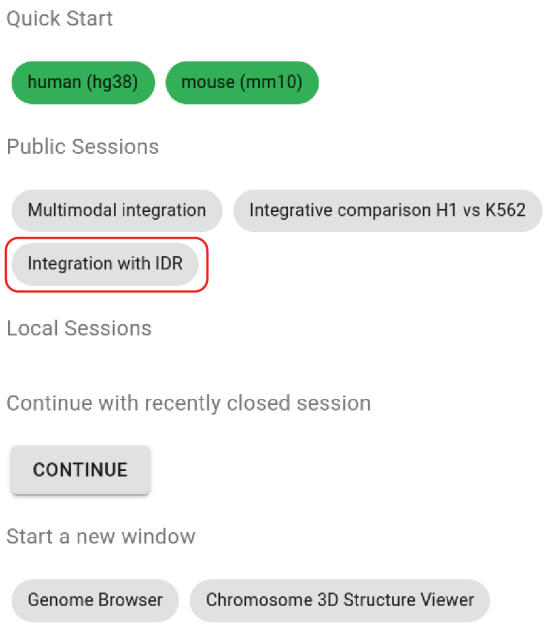

Figure S17: Select "Integration with IDR" on the quick start page

ing the 'Start' button on the home page, you will see the a quick-start page (**Supplementary Fig.** S17). After clicking the "Integration with IDR", you will see a session related to **Fig.** 1d.

Supplementary Fig. S18 shows a screenshot of this session. Here, you can select datasets using the dropdown menu in the web application on the left. After you select a dataset, this tool will automatically fetch images and annotation from the IDR website. When you move mouse over a circle (i.e., probe mark), a tooltip will show you the location of this probe and lead you to an OMERO.iviewer if you click the circle. Note that when you highlight regions on the genome browser on the right, probes overlapping with the highlighted regions will be synchronously marked on the image. This application allows users to directly visualize and compare spatial locations of probes and other multimodal data associated with their 1D genomic coordinates.

B.3.3 Connecting Nucleome Browser with HiGlass

We created a customized web application (http://vis.nucleome.org/static/apps/higlass) to allow users to synchronously view the HiGlass session with Nucleome Browser. This web application allows users to recover the HiGlass session by uploading a configuration file or pasting a session link. Users then can interactively explore datasets on HiGlass together with datasets on Nucleome Browser. For example, when users navigate on the genome browser panel, the HiGlass viewer will jump to the same region (**Supplementary Fig.** S19). Highlighted boxes will synchronously show up on the HiGlass viewer to match the highlighted regions on the Nucleome Browser as well.

We also implemented a drag-and-drop feature to efficiently visualize HiGlass visualization pages on the 4DN DCIC web portal. In any HiGlass display web pages at the 4DN DCIC web portal (e.g., https://data.4dnucleome.org/higlass-view-configs/ae89f1b2-bf71-42db-bee5-e3f7e8b6b590), users can click the 'View JSON' link and drag it on the target box on our web application to directly load the HiGlass session. Our web application will then fetch the HiGlass configuration file automatically from the 4DN DCIC data portal and show the datasets in the HiGlass viewer accordingly.

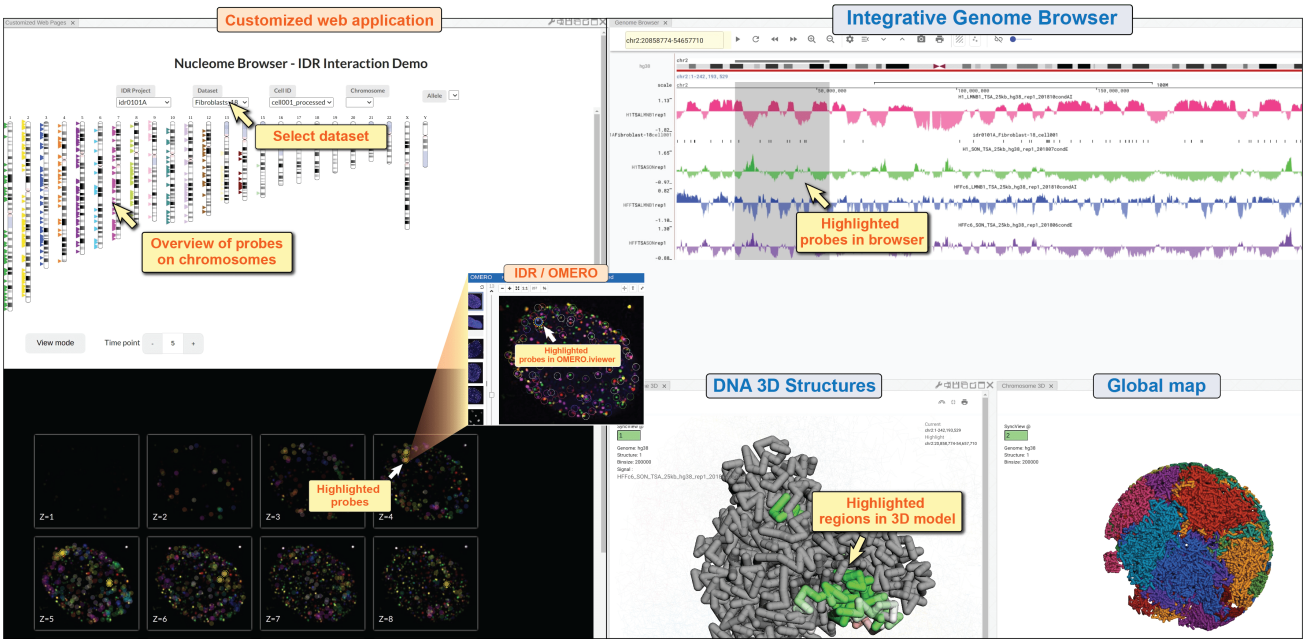

Figure S18: Nucleome Bridge web browser extension allows users to connect Nucleome Browser with Image Data Resource (IDR). This screenshot shows a custom web application that allows users to interactively explore *in situ* genome sequencing (IGS) data [13] from IDR together with the multimodal datasets on Nucleome Browser. Video tutorial can be accessed at https://tinyurl.com/nb-video-tutorial.

B.3.4 Connecting Nucleome Browser with Python Jupyter Notebook

Here, we introduce how to perform interactive data exploration in the Python Jupyter Notebook together with Nucleome Browser. Using the Nucleome Bridge as a relay center, Jupyter Notebook can capture users' operations on Nucleome Browser and update analysis results and plots, accordingly. For example, **Supplementary Fig.** S20 shows that as users change highlighted regions on Nucleome Browser, violin plots update synchronously to display genomic signals under the highlighted regions. Conversely, users can also control which regions to navigate or highlight inside the Jupyter Notebook. This novel feature between the Jupyter Notebook and Nucleome Browser makes data analysis much more interactive and significantly enhances users' productivity. Jupyter notebooks and animations can be found at: https://github.com/nucleome/NB_Jupyter_demo.

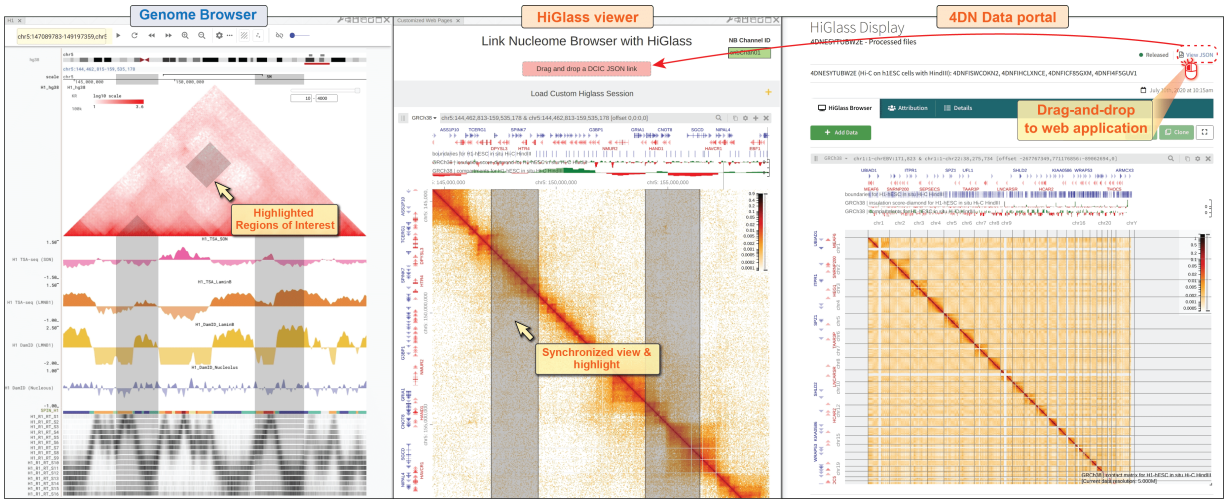

Figure S19: Users can interactively explore HiGlass viewer and Nucleome Browser simultaneously. A drag-and-drop feature allows users to efficiently visualize 4DN data on this web application. When users navigate different regions on the genome browser (left), the HiGlass viewer will update to the same region. Highlighted regions are synchronized from Nucleome Browser to the HiGlass viewer. Video tutorial can be accessed at: https://tinyurl.com/nb-video-tutorial.

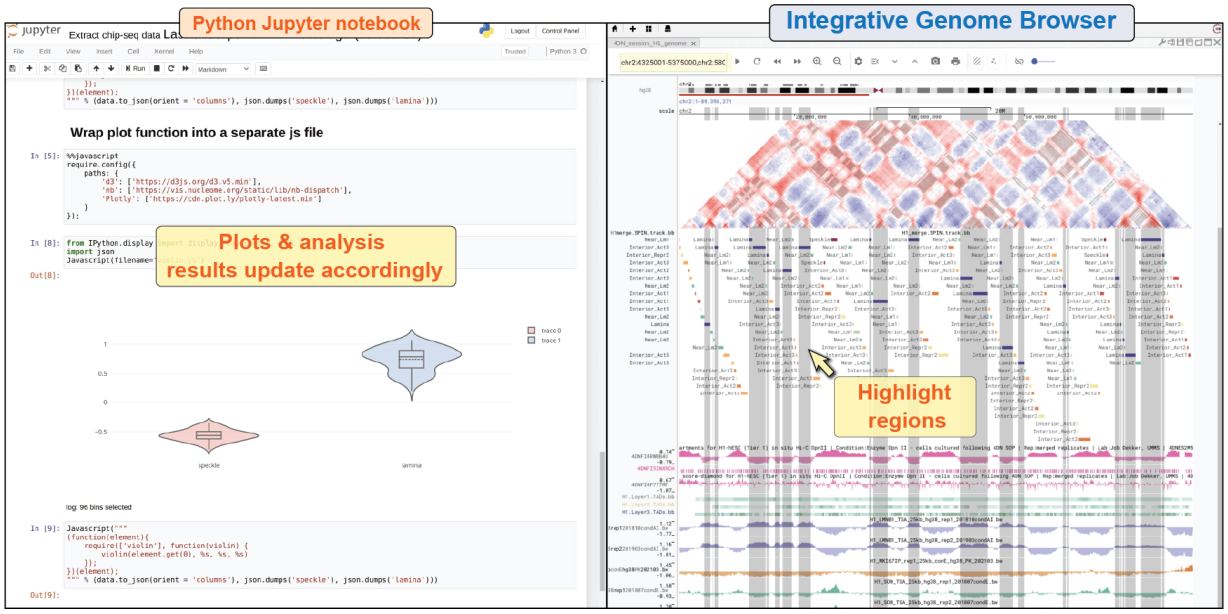

Figure S20: Using the Nucleome Bridge web browser extension, users can use Nucleome Browser to interactively perform data analysis on Python Jupyter Notebook. As users highlight different regions on the genome browser on the right, plots in the Jupyter Notebook automatically update to show the distribution of genomic signals based on the highlighted regions. Video tutorial can be accessed at: https://tinyurl.com/nb-video-tutorial.

B.4 Tutorial IV: Integrative exploration of multimodal datasets

To demonstrate the benefits of integrative data exploration using Nucleome Browser, we showcase its unique capabilities with an example of exploring large-scale genome compartmentalization relative to multiple nuclear bodies (i.e., SPIN states [22]) through interactive data exploration as shown in **Fig.** S21

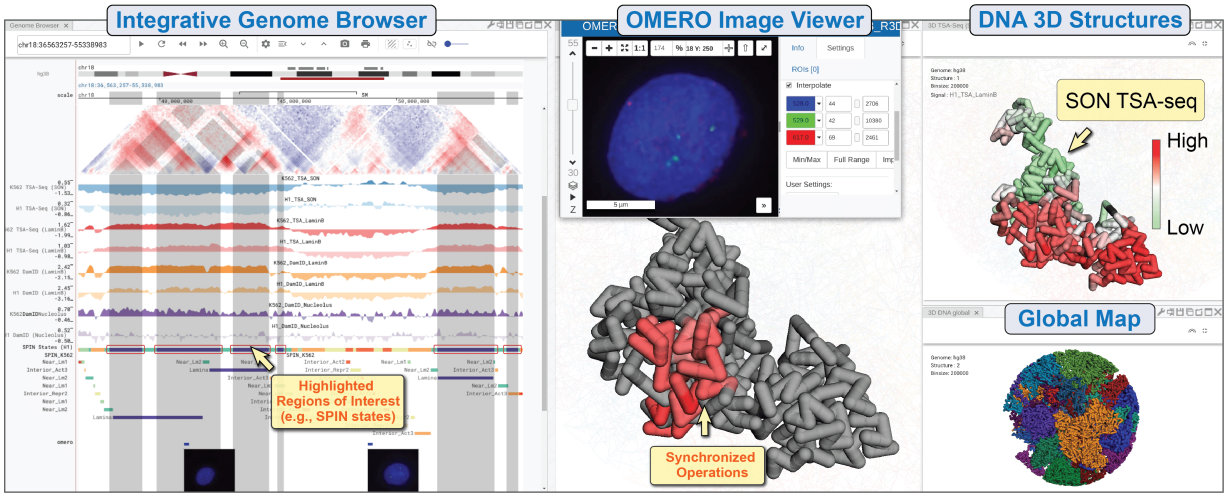

Figure S21: An illustration of multimodal data visualization in Nucleome Browser. Video tutorial can be accessed at https://tinyurl.com/nb-video-tutorial.

To view this example, you can go to: http://paper.nucleome.org in a web browser. Clicking the 'Start' button on the home page, you will see the following quick-start page (**Supplementary Fig.** S22). After clicking the "Multimodal integration", you will see a session related to **Supplementary Fig.** S21.

Specifically, you can select specific SPIN states from a bigBed track and display side-by-side DNA FISH images (from [14]) showing the nuclear localization of these same chromosome regions (**Supplementary Note** C). The image track consists of the thumbnails of the microscopy images hosted on an OMERO server such that the genomic coordinates of the thumbnails represent the targeted genomic loci of the DNA FISH probes. Left-clicking on one of the thumbnails of images you will see a pop-out tab showing the details of this image data using the interface of OMERO.iviewer (**Supplementary Fig.** S23). With such integrative visualization of genomic and imaging data, users can evaluate the SPIN states based on their proximity to nuclear bodies using DNA FISH image data, thereby connecting genomic readout with what we see from microscopy.

You can further integrate 3D genome structure models [15, 23] together with genomic and imaging data. **Supplementary Fig.** S21 displays three different structure model views that integrate features selected in the genome browser panel: (1) Middle panel (bottom): highlighting a chromosome region in the genome browser panel produces highlighting of this same chromosome region in the 3D genome structure model view; (2) Right panel (top): Selecting a SON TSA-seq track in the genome browser panel produces a color-coding display of these SON TSA-seq values superimposed on the 3D genome structure model (SON TSA-seq data from [14]); and (3) Right panel (bottom): a 3D genome global view showing each chromosome in a different color. Importantly, the communication between the genome browser and the 3D structure model panel is bi-directional. You can select a chromosome segment in the 3D genome model view to directly navigate to this region in the genome browser panel.

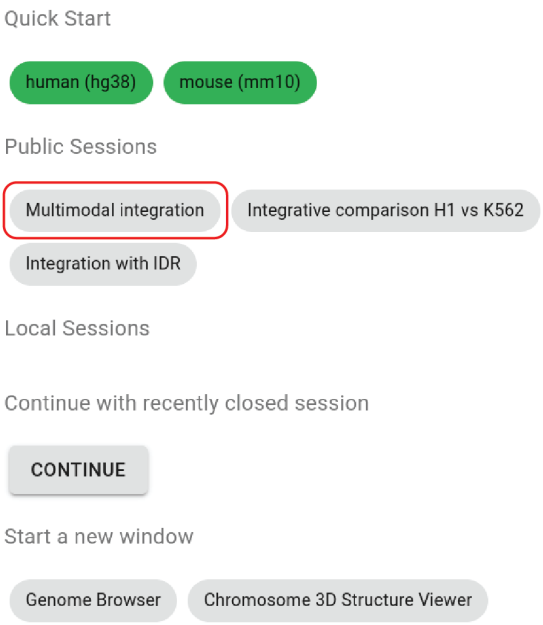

Figure S22: Select 'Multimodal integration' on the quick start page.

Alternatively, you can select a chromosome segment in the genome browser panel to visualize this region in the 3D genome structure model view. In the genome browser panel, left-clicking mouse, holding it, and dragging on the genomic tracks, you will see highlighted regions in transparent grey color (**Supplementary Fig.** S24). After you release mouse, you can move the mouse on top of the highlighted region by left-clicking it and holding it. Note that if you highlight an off-diagonal region on the Hi-C contact map, the two associated regions will be both highlighted, accordingly. Additionally, as you move the highlighted regions, the highlighted region on the 3D structure model panel in the middle will also be updated synchronously.

When you click any region in bed tracks with color annotations, all the regions with the same color will be highlighted simultaneously (**Supplementary Fig.** S25). Here, the colors can be defined with the 9th column of the bed file following the definition of 12-column bed format: https://genome.ucsc.edu/FAQ/FAQformat.html#format1.

Upon right-clicking the highlighted regions, you will be directed to the highlighted regions. If highlighted regions are not consecutive on the genome browser you will see stitched regions connecting multiple 'sliced' highlighted regions (similar to the multi-region used in the UCSC Genome Browser) (**Supplementary Fig.** S25).

By integrating these genomic, imaging, and 3D genome structure model views, Nucleome Browser offers an easy, intuitive navigation experience to interactively explore multimodal 4DN datasets. This enables users to generate testable hypotheses of nuclear structure-function connections that may be obscured by individual data modality.

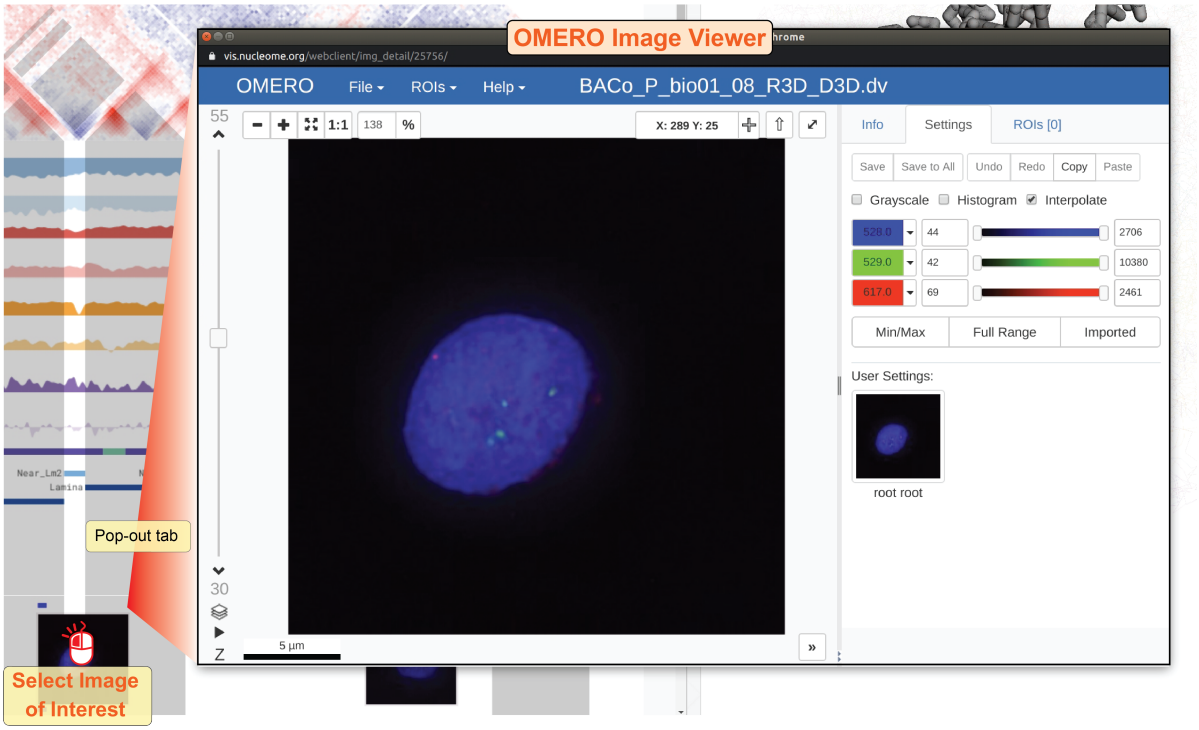

Figure S23: An illustration of exploring imaging data in Nucleome Browser by connecting to a local OMERO server.

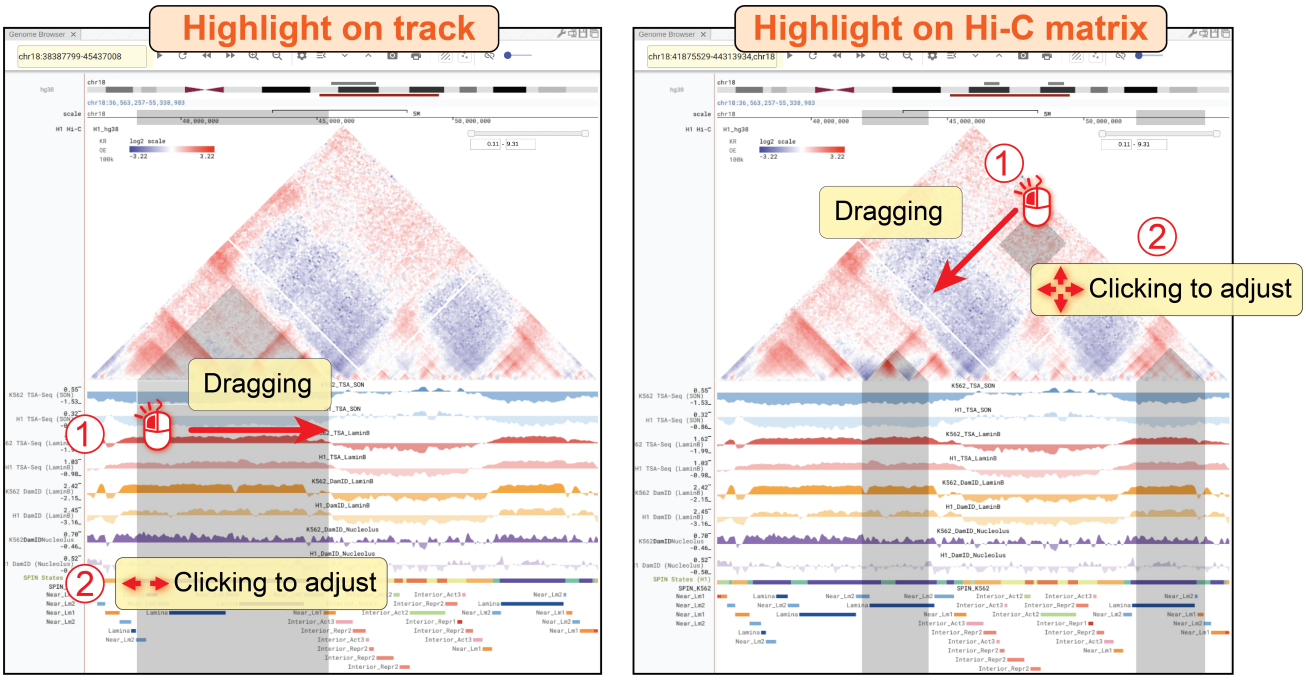

Figure S24: Highlighting on genomic data is straightforward. Users can either highlight a single region on 1D genomic signal track or two genomic loci on 2D track such as Hi-C contact map. Users can click mouse on the highlighted region (grey polygon) and drag to further adjust it.

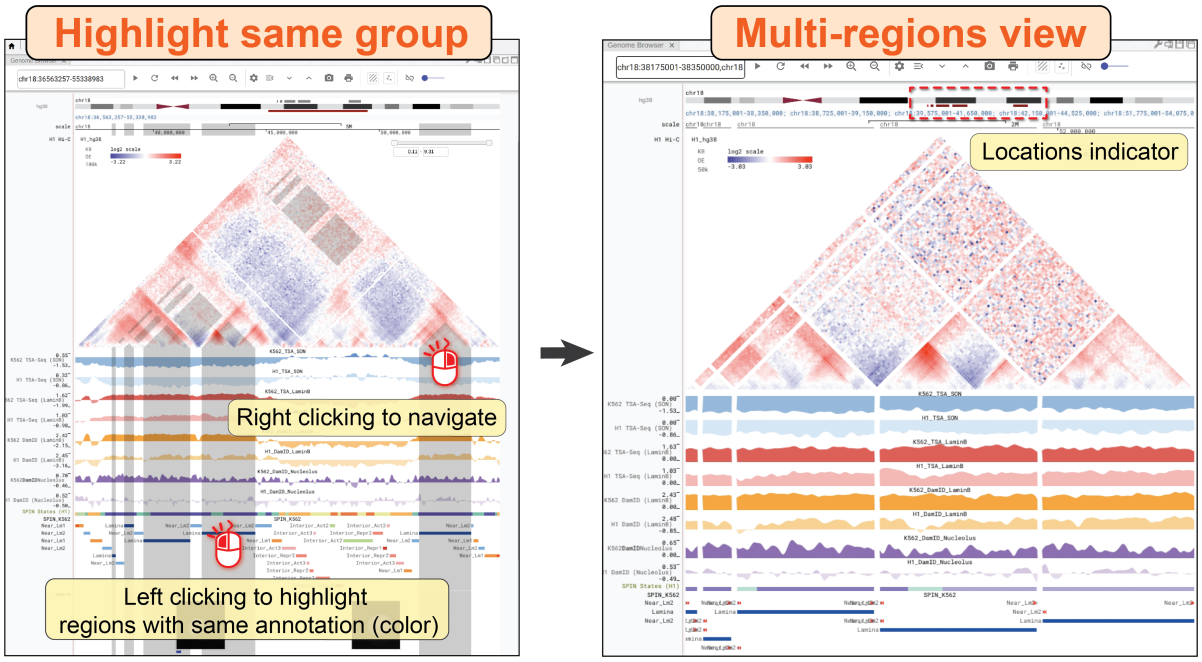

Figure S25: Clicking a region on the bed track, other regions with the same color will be highlighted simultaneously. This feature is useful to visualize annotations of chromatin segmentation such as SPIN [22]. When right-clicking the highlighted regions, you will navigate to the highlighted regions (in multi-regions view) even though they are not consecutive.

B.5 Tutorial V: Integrative comparison of multimodal datasets across multiple cell types

Comparison of 3D genome structures and their roles in nuclear function between different cell types or conditions is highly informative. Nucleome Browser is an ideal tool to achieve this type of comparison because of its modular architecture. For example, users can create multiple panels with each of them representing one condition or cell type. Users then can view or highlight regions in a synchronized manner across multiple panels. Here, we illustrate how to effectively compare multimodal data between two cell types using the session related to **Supplementary Fig.** S5. In the quick-start page on the home page, after clicking the 'Integrative comparison H1 vs K562' button, you will see a session associated with **Supplementary Fig.** S5 showing a comparison of various genomic data between H1 and K562 cells (**Supplementary Fig.** S26).

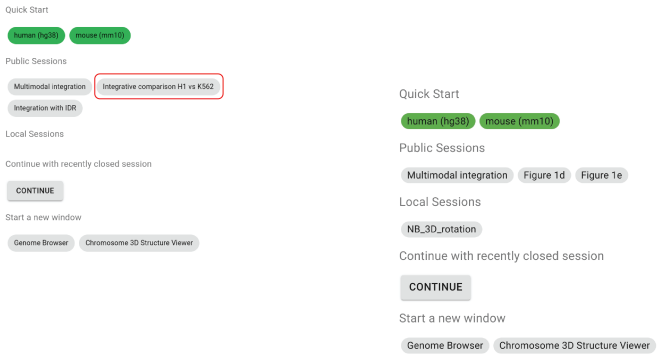

Figure S26: Select "Integrative comparison H1 vs K562" on the quick start page

In this session, two genome browser panels on the left show a comparison of various genomic data between K562 and H1, including Hi-C contact matrices, TSA-seq [14], and DamID tracks mapping to multiple nuclear bodies (e.g., nuclear speckles, nuclear lamina, and nucleolus), and DNA replication timing profiles (**Supplementary Fig.** S27). Two 3D genome structures on the right connect 3D structures with genomic signals by rendering the chromatin according to TSA-seq Lamina B1 signal in H1 and K562, respectively. Importantly, you can highlight any regions on the genomic tracks and compare them to other cell types. For example, the default highlighted regions in this session indicate two dramatically different states: 1) a region showing a conserved lamina-associated profile in H1 and K562 as marked by the blue rectangle; 2) a region showing a cell type-specific pattern between H1 and K562 as marked by the green rectangle. In H1, the cell type-specific region has a relatively lower lamina B1 TSA-seq signal than K562, which is consistent with its earlier replication timing profiles compared with the late replicated profile in K562. You can also use an interactive scatterplot analysis tool to select regions of interest efficiently. As shown by the scatterplot in **Supplementary Fig.** S5, the cell type-specific region we highlighted is one of several outliers deviating from the generally positive correlation between H1 and K562 Lamin B1 TSA-seq scores. Conversely, you can circle genomic bins in the scatterplot and highlight them on the genome browser and 3D genome structure models as well.

Left-clicking any place in the 3D genome structure panel, holding the mouse, and dragging it you will see different views of the 3D genome structure (**Supplementary Fig.** S28). You can use the scroll wheel of the mouse to zoom-in and zoom-out the 3D structure. Holding the control key and left-clicking and holding the mouse you can move the 3D structure inside the panel.

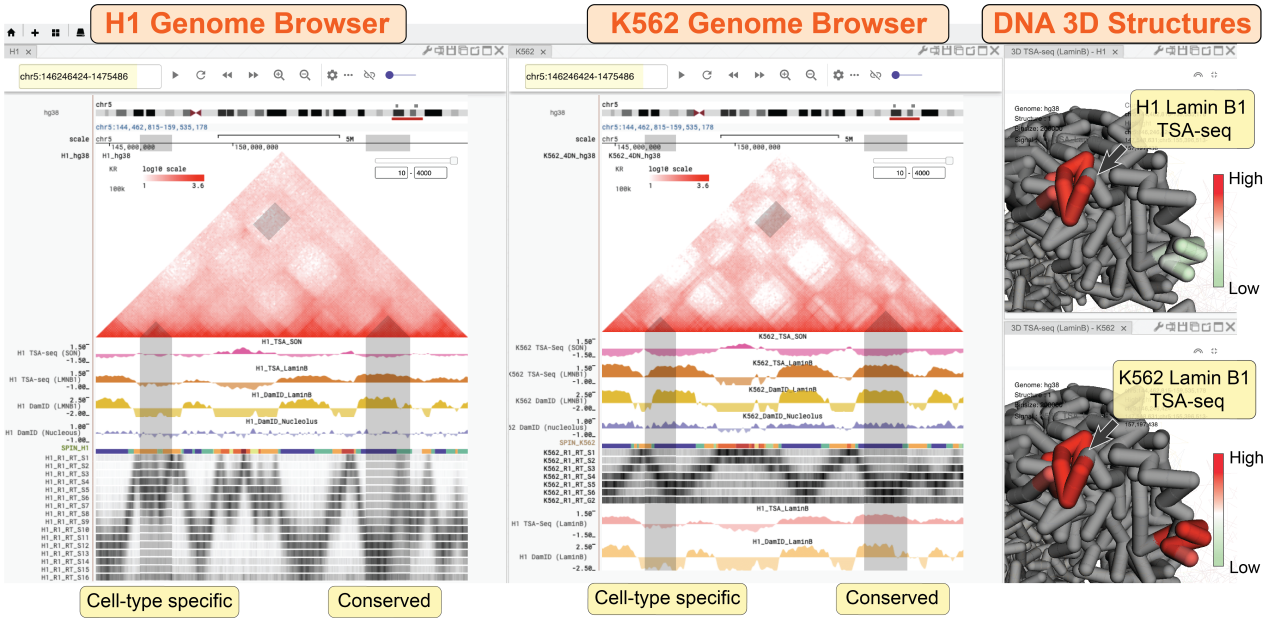

Figure S27: Nucleome Browser facilitates the comparison of nuclear structure-function connection between different conditions and cell types. Video tutorial can be accessed at: https://tinyurl.com/nb-video-tutorial.

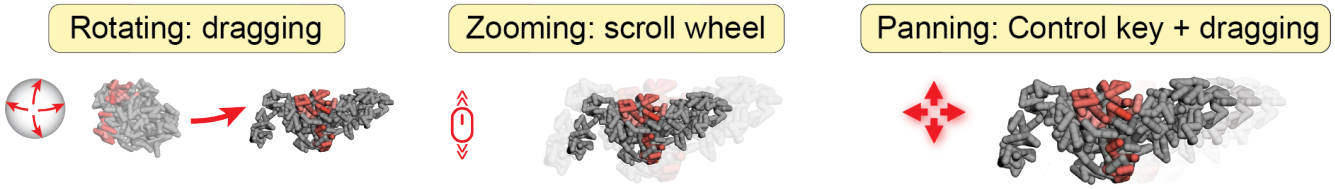

Figure S28: Users can pan, move, and zoom-in/out on the 3D genome structures.

B.6 Tutorial VI: Building a custom data service – An example using single-cell Hi-C data

Here, we show how to set up a custom Nucleome Browser session by launching a local data service using published data (population Hi-C and single-cell Hi-C) [12] downloaded from gene expression omnibus (GEO) under the accession number GSE80280. In [12], the authors generated population Hi-C data from the mouse embryonic stem cells, eight single-cell Hi-C datasets from the same cell type as well as 3D structure models inferred from Hi-C using the Chrom3D tool [24]. The whole script used to process the downloaded data can be accessed at: https://github.com/nucleome/Tutorial-SingeCellHiC. Users can also clone this GitHub repository and use the demo dataset that we have already processed to set up a local data server to view the demo data on the Nucleome Browser. In the following sections, we will introduce the procedures of pre-processing and setting up a custom data service.

Requirements

To start with, you need to prepare a laptop or desktop on Linux or macOS and clone this repo to your machine. You also need to pre-install the following software packages/tools for different operating systems. For Windows machine, you need to install Windows Subsystem for Linux to pre-process the data.

Linux (tested under Ubuntu 20.04/18.04 LTS)

- terminal
- gzip
- tar
- awk (mawk/gawk should also work)
- sed
- Juicer Tools (https://github.com/aidenlab/juicer/wiki/Download). This tutorial is tested under Juicer Tool version 1.22.01.
- fetchChromSizes and bedToBigBed. You can install these binaries built from http://hgdownload.cse.ucsc.edu/admin/exe/. We also provide them in the script folder of the GitHub repo.
- nucleserver (https://github.com/nucleome/nucleserver). This tutorial is tested under nucleserver version 0.2.6.

Most of the tools except for the Juicer Tools, fetchChromSizes, and bedToBigBed should be pre-installed for most Linux distributions. You also need to download the right version of nucleserver based on your machine operating system. You can get that information by typing `dpkg --print-architecture` in the terminal. In the following sections, we assume you have downloaded nucleserver and copy it to the scripts folder under the cloned GitHub repo.

macOS (tested under macOS 12 (Monterey))

- terminal
- gzip
- tar
- awk (mawk/gawk should also work)
- sed
- Juicer Tools (https://github.com/aidenlab/juicer/wiki/Download). This tutorial is tested under

Juicer Tool version 1.22.01.

- fetchChromSizes and bedToBigBed. You can install these binaries built from http://hgdownload.cse.ucsc.edu/admin/exe/. We also provide them in the script folder of the GitHub repo.
- nucleserver (https://github.com/nucleome/nucleserver). This tutorial is tested under nucleserver version 0.2.6

In macOS, you can open the terminal by clicking the Launchpad icon in the Dock, typing terminal in the search field, then click Terminal. You also need to download the right version of nucleserver based on your machine operating system. For macOS, you can get that information by typing set | grep "MACHTYPE" in the terminal. Here, we assume you download nucleserver and put it under the scripts folder.

Getting data from GEO (Linux and macOS)

First, open terminal in Linux or macOS. Then, create a new folder and use wget command to download the whole dataset from GEO to a local computer and named it 'GSE80280_RAW.tar' as shown below:

```
# create a new folder
mkdir single_cell_hic
cd single_cell_hic
# Download data
wget -O GSE80280_RAW.tar "https://www.ncbi.nlm.nih.gov/geo/download/?acc=GSE80280&
    format=file"
```

Note that this will download a large single file of about 20Gb. Next, you need to extract files from the tar file in the current directory:

```
# Extract file
tar -xvf GSE80280_RAW.tar
gzip -d *.gz
```

You should see multiple files in different formats with various suffixes, such as .bed, .pdb, etc. Note that the folder will take about 43Gb after decompression.

Pre-processing of ChIP-seq peaks (Linux and macOS)

In order to launch data service and visualize ChIP-seq peaks in Nucleome Browser, we need to pre-process data and convert them into standard formats supported by Nucleome Browser. First, we use tools from the UCSC Genome Browser and Blat application binaries built (http://hgdownload.cse.ucsc.edu/admin/exe/) to convert ChIP-seq peak files into bigBed format as shown below:

```
# Fetch chrom.sizes information
fetchChromSizes mm10 > mm10.chrom.sizes

# Add additional optional BED fields
# Replace "ChIP-Seq" with sample name
awk '{print $1"\t"$2"\t"$3"\t"$4"\t0\t.\t"$2"\t"$3"\t255,0,0"}' ChIP-Seq_peaks.bed
    > ChIP-Seq_peaks.detail.bed

# Create bigBed file
bedToBigBed ChIP-Seq_peaks.detail.bed mm10.chrom.sizes ChIP-Seq_peaks.bb
```

Note that you need to make fetchChromSizes and bigToBigBed executable using the command chmod (e.g., chmod +x bedToBigBed). You can the tools provided in the GitHub repo.

Pre-processing of population Hi-C and single-cell Hi-C data (Linux and macOS)

The Hi-C contact map must be transformed into .hic format in order to display in Nucleome Browser. Here we use Juicer tools (https://github.com/aidenlab/juicer) to create .hic files from the downloaded contact pair files. The Hi-C contact pair file should be converted into one of the formats as shown in the documentation of Juicer tools (https://github.com/aidenlab/juicer/wiki/Pre).

```
# Medium format (for single cell Hi-C)
<readname> <str1> <chr1> <pos1> <frag1> <str2> <chr2> <pos2> <frag2> <mapq1> <mapq2
    >

# Short with score format (for population Hi-C, which is already binned)
<str1> <chr1> <pos1> <frag1> <str2> <chr2> <pos2> <frag2> <score>
```

We provide customized scripts to convert population/single-cell Hi-C contact pairs to .hic files as shown below. You also need to make these script executable using the chmod command.

```
# Single cell Hi-C
./scripts/create_single_cell_HiC.sh Cell_X_contact_pairs.txt Cell_X /
    path_to_juicer_tools/juicer_tools.jar

# Population Hi-C
./scripts/create_population_HiC.sh GSM2123564_Haploid_mESC_population_hic.txt
    population_hic /path_to_juicer_tools/juicer_tools.jar
```

Note that these two scripts also require an additional file called chr_order, which specifies the list of chromosomes ID. We also provide this file in the GitHub repo.

Pre-processing of 3D genome structures (Linux and macOS)

The 3D genome structures for each single cell are provided in pdb format, and we will need to convert them into nucle3d format.

We provide a customized script named pdb2Nucle3d.sh to do this step as shown below. For each single cell, the pdb file contains 10 structures. This script will generate 10 nucle3d files. Note that this script is only tested under this specific dataset, and it may not work for generic PDB files.

```
# pdb to Nucle3d
./scripts/pdb2Nucle3d.sh Cell_X_genome_structure_model.pdb Cell_X
```

Starting the data service

Now you should have prepared all the data for visualization in Nucleome Browser. Next, you need to prepare a configuration file to start a data service so that Nucleome Browser can display these datasets. You will also need a cross-platform tool called nucleserver (https://github.com/nucleome/nucleserver). Here, we will briefly introduce how to use it.

Suppose you have put all the generated files under a folder named NB_single_cell. First, you need to create an Excel file (we have tested this using the LibreOffice software in Linux and Microsoft Excel Spreadsheet in macOS). Create a new sheet called 'Config' and put the following information inside this sheet as the figure shown in **Supplementary Fig.** S29. The first row is a header of this sheet. The second

row specifies the complete path of the folder. You should replace the path with the real path based on your computer. You can use the pwd command in Linux and macOS to find the complete path of the current folder.

	A	B
1	Var	Value
2	root	/home/yangz6/Project/Nucleome_Browser/Tutorial-SingeCellHiC/NB_single_cell

Figure S29: Screenshot of the Config sheet.

Next, you need to create another sheet called Index and put in the following information (**Supplementary Fig.** S30)

	A	B	C	D	E
1	Genome	Name	Type	Name	Value
2	mm10	Hi_C	track	A	B,C,D
3	mm10	ChIP_seq	track	A	B,C,D

Figure S30: Screenshot of the Index sheet.

Briefly, the first row is the header. The first column indicates the genome assembly. The second column represents the names of data sheets you will create soon. The third column is used for future development, you can just put 'track' here. The fourth column is the column ID for the short label of the track. The fifth column is the column IDs for uri, meta information, and long label.

As the above figure shows, you will need to create two data sheets, one called Hi_C and another called ChIP_seq. All Hi-C tracks will be specified in the Hi_C data sheet. All ChIP-seq tracks will be specified in the ChIP_seq data sheet. In each data sheet, you need to provide four attributes for each track, including short label, uri (relative path to the file from the root path, see the Config sheet), meta link (can be URL or None if it is not available), and long label as shown in **Supplementary Fig.** S31- S32

	A	B	C	D
1	shortLabel	uri	metaLink	longLabel
2	CTCF_hapmESC	chip_seq/GSM2123546_CTCF_hapmESC_peaks.detail.bb	None	CTCF_hapmESC
3	Smc3_hapmESC	chip_seq/GSM2123547_Smc3_hapmESC_peaks.detail.bb	None	Smc3_hapmESC
4	H3K27me3_hapmESC	chip_seq/GSM2123548_H3K27me3_hapmESC_peaks.detail.bb	None	H3K27me3_hapmESC
5	H3K36me3_hapmESC	chip_seq/GSM2123549_H3K36me3_hapmESC_peaks.detail.bb	None	H3K36me3_hapmESC
6	H3K4me3_hapmESC	chip_seq/GSM2123550_H3K4me3_hapmESC_peaks.detail.bb	None	H3K4me3_hapmESC
7	H3K9me3_hapmESC	chip_seq/GSM2123551_H3K9me3_hapmESC_peaks.detail.bb	None	H3K9me3_hapmESC
8	Nanog_hapmESC	chip_seq/GSM2123552_Nanog_hapmESC_peaks.detail.bb	None	Nanog_hapmESC
9	H3K27me3_dipmESC	chip_seq/GSM2123556_H3K27me3_dipmESC_peaks.detail.bb	None	H3K27me3_dipmESC
10	H3K36me3_dipmESC	chip_seq/GSM2123557_H3K36me3_dipmESC_peaks.detail.bb	None	H3K36me3_dipmESC
11	H3K4me3_dipmESC	chip_seq/GSM2123558_H3K4me3_dipmESC_peaks.detail.bb	None	H3K4me3_dipmESC
12	Nanog_dipmESC	chip_seq/GSM2123560_Nanog_dipmESC_peaks.detail.bb	None	Nanog_dipmESC

Figure S31: Screenshot of the data sheet for ChIP-seq peaks.

Finally, you can start a data service using nucleserver with the excel configuration file you just created.

```
# Start data service using nucleserver
./scripts/nucleserver start -i Excel_configuration_file.xlsx
```

You should see the following logging information indicating that data service is successfully created (**Supplementary Fig.** S33).

You can then open a web browser (e.g., Google Chrome or Safari browser), and go to https://vis.nucleome.org. Click the 'START' button on the main page and then click the Genome Browser

	A	B	C	D
1	shortLabel	uri	metaLink	longLabel
2	Haploid_mESC_population_hic	hic/population_hic.hic	None	Haploid_mESC_population_hic
3	Cell_1_hic	hic/Cell_1.hic	None	Cell_1_hic
4	Cell_2_hic	hic/Cell_2.hic	None	Cell_2_hic
5	Cell_3_hic	hic/Cell_3.hic	None	Cell_3_hic
6	Cell_4_hic	hic/Cell_4.hic	None	Cell_4_hic
7	Cell_5_hic	hic/Cell_5.hic	None	Cell_5_hic
8	Cell_6_hic	hic/Cell_6.hic	None	Cell_6_hic
9	Cell_7_hic	hic/Cell_7.hic	None	Cell_7_hic
10	Cell_8_hic	hic/Cell_8.hic	None	Cell_8_hic

Figure S32: Screenshot of the data sheet for Hi-C contact maps.

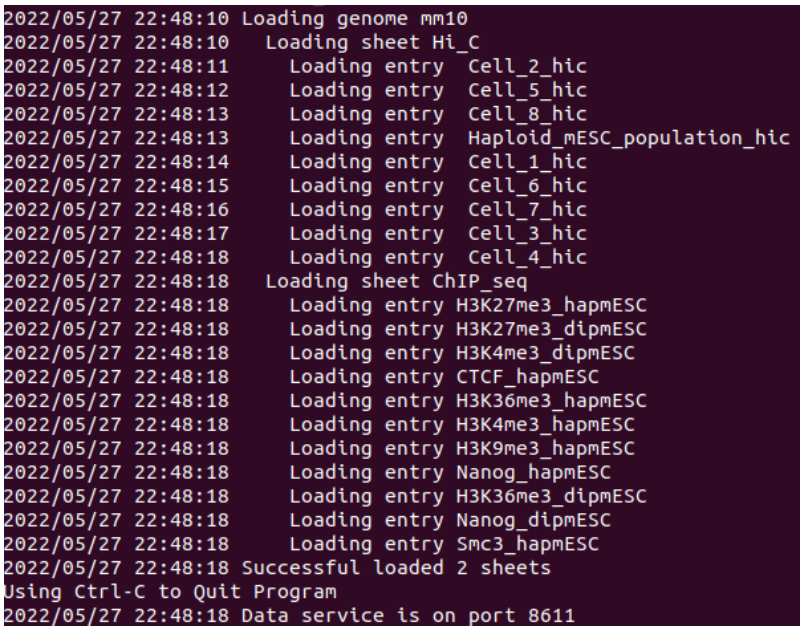

Figure S33: Screenshot of report after successfully running nucleserver.

button at the bottom. You can click the configuration button (step 1) on the top-right of the genome panel and select mm10 as the genome version in the track setting page (**Supplementary Fig.** S34). In the dropdown menu below you should see two data sheets named Hi_C and ChIP_seq which contain data tracks you just created. If these two data tracks do not appear, you can then try to manually add them by clicking the data server setting button (step 2) as shown below and adding http://127.0.0.1:8611 in the URI column (step 3). If you see the 'Status' column showing active after you click the update button (steps 4&5), it means that you have successfully added a local data service to Nucleome Browser.

Note that you must use the HTTPS version of Nucleome Browser (https://vis.nucleome.org) to load a local data service. The HTTP version may not work due to the limitation of cross-origin resource sharing (CORS) restriction.

To add the 3D structure to Nucleome Browser, you also need to start a local data service. You can use any file server to host nucle3D files but we also provide a simple version in nucleserver. You can start a file server to host 3D genome structure models using the following command.

```
# Start a file server to host 3D genome structure models
./script/nucleserver file --root NB_single_cell/3d/ --port 8612
```

This will start a data service showing 3D structure file at http://127.0.0.1:8612/get. You can see all nucle3d files here. Then you can create a new 3D structure panel in the Nucleome Browser and copy the

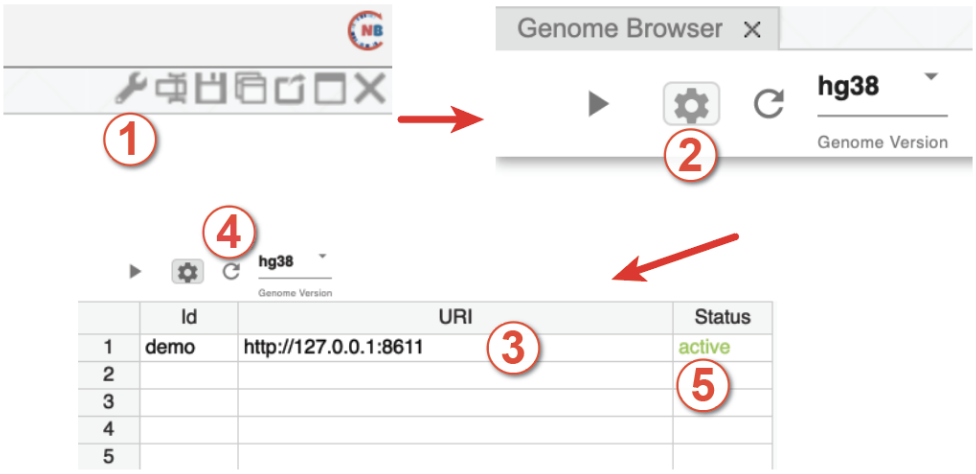

Figure S34: The procedure to manually add a local custom service. You can use a different port depending on the argument you use to run nucleserver.

link of nucle3d file (e.g., http://127.0.0.1:8612/get/Cell_1.model_1.nucle3d) and paste it to the data URL box on the 3D structure panel as shown in **Supplementary Fig.** S35.

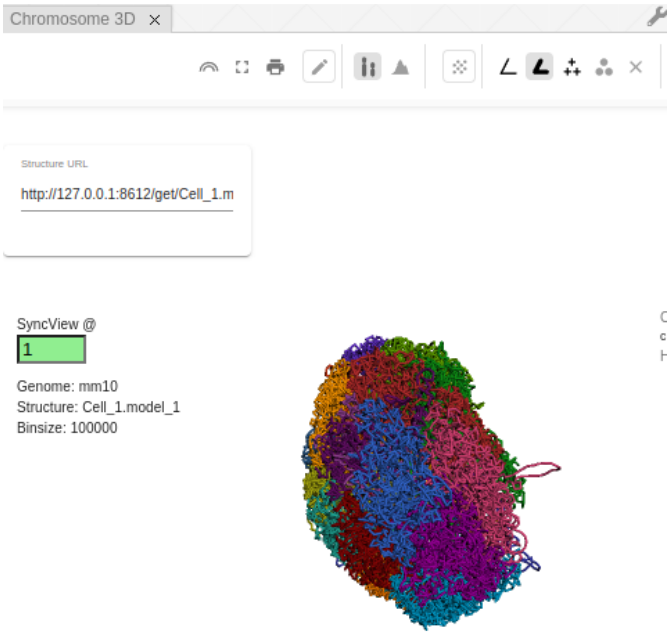

Figure S35: You can copy the URL of the 3D structure file and paste it to the 'Structure URL' box of the 3D structure panel.

We also provide video tutorials for different operating systems to host a custom data service using the demo data provided in the GitHub repo (https://github.com/nucleome/Tutorial-SingeCellHiC#start_demo).

Supplementary Fig. S6 shows a screenshot of visualizing the processed single-cell Hi-C data. In this example, single-cell Hi-C matrices from Cell-1 and Cell-2 together with the population Hi-C contact matrix are shown on the left. The inferred 3D genome structure models for Cell-1 and Cell-2 are shown on the center-top. Other genomic data such as ChIP-seq data is shown on the center-bottom. We highlight a region on the 2D contact map, in which paired reads are only observed in Cell-1 but not in Cell-2. This suggests that the two genomic loci may locate closer to each other in Cell-1 than in Cell-2. In the 3D

model, we indeed observe that these two loci are closer to each other in 3D structures in Cell-1 than in Cell-2.

C Supplementary Note: Methods Description

C.1 Overall design of Nucleome Browser

Nucleome Browser aims to comprehensively address the challenge of multimodal data visualization for 4D Nucleome research. The key conceptual innovation of multimodal data visualization in Nucleome Browser is that each component has its own purpose but collectively they allow users to achieve integrative and interactive navigation of heterogeneous datasets. In Nucleome Browser, we designed composable (flexible arrangement) and configurable (customized setting) web components, which communicate with each other to achieve synchronized operations, to visualize multimodal datasets (**Supplementary Fig.** S2). Importantly, different types of web components are all self-encapsulated into panels to communicate with each other. For example, a panel (web component) designed for displaying 3D genome structure models could automatically update its view to match the views of other web components. This is achieved by our multi-channel hierarchical and adaptive communication mechanism (see text below and **Supplementary Fig.** S2), enabling operations triggered by users (e.g., navigation or highlight) in any web component to broadcast to other connected web components across multiple browser tabs (i.e., multi-tabs). In addition, we further developed a web browser extension named Nucleome Bridge as a relay center for real-time cross-domain communication (i.e., multi-domain) among web applications (**Supplementary Table** S1). Notably, our design allows users to synchronously explore multimodal datasets hosted on Nucleome Browser and external data portals/tools, including UCSC Genome Browser [3], WashU Epigenome Browser [16], HiGlass [11], the Jupyter Notebook, and other customized web applications tailored towards specific purposes. The details of different types of web components and adaptive communication mechanism are described below.

C.2 Web components for the visualization of multimodal datasets

Nucleome Browser consists of a series of web components designed for interactive navigation of multimodal datasets, including genomic, imaging, and 3D genome structures as well as customized web applications.

Genome browser web component for genomic data

The genome browser web component in Nucleome Browser allows users to explore various genomic and epigenomic data in widely used formats, including bigWig, bigBed, Tabix, and .hic format (**Supplementary Fig.** S2). In the back-end, we implemented a cross-platform tool named nucleserver using GO language to host multiscale genomic data as a web service. Nucleserver utilizes Gonetics (https://github.com/pbenner/gonetics) and bix (https://github.com/brentp/bix) for efficient query of bigWig, bigBed, and Tabix data together with custom scripts to handle data query related to .hic data. In the front-end, our genome browser web component allows users to visualize genomic data through navigation and rendering of the queried data into 1D and 2D genomic tracks. In particular, the genome browser not only supports commonly used operations in existing genomic data visualization tools but also many novel features to facilitate integrative exploration across multiple panels, including three different visualization modes, an interactive scatterplot analysis tool embedded in the browser, and a drag-and-drop feature to transfer tracks from a genome browser to a 3D genome structure model panel (see **Supplementary Table** S1 for a comparison with existing tools). It also allows users to add multiple web service

URLs even for private data with password protection. **Supplementary Note** B.2.1 provides a tutorial and more description on the design of the genome browser web component.

3D genome structure model web component

Integrative visualization of 3D genome structures together with genomic and imaging data would provide users with a three-dimensional viewpoint of the multimodal datasets. We developed a 3D genome structure model web component using the 3Dmol JavaScript library [25] to display 3D genome structures derived from either computational modeling approaches from Hi-C [15, 23]) or recently developed multiplexed imaging methods [1, 2, 13, 18, 19]. This component visualizes 3D genome structures with multiple styles, such as line, cartoon, cross-point, and sphere, providing users an integrative solution to consolidate 3D genome structure models with various genomic data. We created a new data format named nucle3d to store spatial information of 3D genome structure models. The nucle3d data format contains a header section with meta-information (e.g., genome and resolution) and a body section with X, Y, Z coordinates of chromatin segments. Users can use a command-line tool named nucle (https://github.com/nucleome/nucle) to convert 3D genome structure models from other formats such as hss and cmm into the nucle3d format. To facilitate the integrative visualization of multiple 3D genome structure models, we developed several features to allow users to compare 3D models and explore genomic signals in the context of 3D genome structure models. A detailed description of these features and a tutorial can be found in **Supplementary Note** B.2.2.

Web applications for a wide variety of customized datasets

In Nucleome Browser, users can create new web components to display customized data and integrate these web components to existing web components using our adaptive communication mechanism (see text below and **Supplementary Fig.** S2). As proof-of-concept, we built four applications to demonstrate the versatility and scalability of Nucleome Browser in integrating new data types and analysis results.

In example I (**Fig.** 1d), we created a customized web application to link Nucleome Browser with *in situ* genome sequencing (IGS) data [13] hosted on IDR [26]. IGS data bridges genomic information of hundreds of probes with their spatial position in single-cell images, which greatly fits the unique capability of Nucleome Browser for integrative visualization of multimodal datasets. We compiled the meta-information of IGS data from human fibroblasts and intact early mouse embryos (IDR project accession #: idr0101) using the IDR API (https://idr.openmicroscopy.org/about/api.html). This web application allows users to select a specific cell, examine *in situ* sequencing images from the IDR platform, and interactively explore the probes' spatial information on images together with their genomic locations and other data in the genome browser web component. For example, when users navigate in the genome browser, the probes overlapping with the highlighted regions in the genome browser will be synchronously marked on the image.

In example II (**Supplementary Fig.** S3a-b), we obtained the reconstructed 3D genome position data from OligoSTORM [1] from https://github.com/BogdanBintu/ChromatinImaging. This dataset contains the 3D positions of hundreds of 30kb chromatin segments at single-cell resolution across six experiments. Users can select a single cell to visualize its 3D genome structure model and the corresponding 2D distance matrix (**Supplementary Fig.** S3b).

In example III (**Supplementary Fig.** S3c), we downloaded the 3D genome structure models generated by OligoDNA-PAINT [2] from http://sgt.cnag.cat/3dg/datasets. This dataset contains the 3D position of an 8.16 Mb region on chromosome 19 in a diploid genome, separately processed into 9 groups

of chromosomal segments (CS1-9) with a resolution of 10kb using the integrative modeling of genomic regions (IMGR) method [2]. Users thus can interactively explore these chromosomal segments together with tracks on the genome browser panel (**Supplementary Fig.** S3a,c).

In example IV (**Supplementary Fig.** S3d), we obtained 3D DNA FISH data corresponding to eight genomic regions (from [14]). We first added these FISH images onto a local OMERO server. We then created a customized image track inside the genome browser web component to display the images as thumbnails based on the genomic locations of FISH probes (**Supplementary Fig.** S7). Users can select any FISH probe to navigate to its genomic location in the genome browser and further click a thumbnail to explore the FISH image data using the OMERO.iviewer interface.

Visualizing imaging data in Nucleome Browser

(i) Visualizing imaging data from 4DN. We developed a special web component to display imaging data hosted on the 4DN data portal (https://data.4dnucleome.org) [27]. We first compiled meta-information associated with imaging data on the 4DN data portal, extracted imaging datasets containing genomic coordinates information, and saved meta-information into a local database using the binning index algorithm for fast query of images for queries. We then built a customized web component to display imaging datasets as tracks of thumbnails of the microscopy images such that the genomic coordinates of the thumbnails represent the targeted genomic loci of the probes. **Supplementary Fig.** S9 shows a screenshot of viewing DNA FISH datasets generated from [21]. If a user clicks a thumbnail, an OMERO.iviewer window will pop out, allowing detailed exploration of the image data.

(ii) Visualizing imaging data from local OMERO server. We also developed a specific plugin of the OMERO server named OMERO-NB-index (https://github.com/nucleome/omero2cnb) to allow users to interactively explore imaging datasets stored on a local OMERO server in a genome browser panel. Once a user adds genomic coordinates meta-information of images as key-value pairs in the OMERO's PostgreSQL database through its user interface, OMERO-NB-index would monitor the database, store the meta-information into a binning index data structure, and create a web service for efficient query imaging datasets based on a genomic coordinate (**Supplementary Fig.** S7). Users can load this web service in the genome browser web component and navigate image datasets as tracks of thumbnails. When users click an image thumbnail, an OMERO.iviewer will pop out to show the image data.

Other web components in Nucleome Browser

Supplementary Note B.1 describes the design of other web components and provides a tutorial.

C.3 Multi-channel cross-tab and cross-domain adaptive communication mechanism

A panel container system to host heterogeneous panels

We developed a panel container system to host multiple panels based on Golden Layout JavaScript library: https://golden-layout.com. This system has three important features. (1) Flexible layout: Users can open multiple panels and flexibly arrange their layout by resizing the boundary of the panel using mouse or drag-and-drop a panel to set a new layout. (2) Panel management: Users can easily manipulate panels, including duplication, pop-out, rename, closing a panel, etc. We created a panel space widget to save and restore panels saved in local storage. Users can also save the layout of panels into a session and share the session to collaborators using a URL. (3) Cross-panel communication: The panel container system provides a local event-dispatch system, which handles the communication of events across panels

in the same domain (as compared with the cross-domain dispatch system supported by the Nucleome Bridge web extension, see below). Notably, this panel container system is flexible and expandable, and can easily incorporate new customized web components to visualize any future data types.

A multi-channel event-dispatch system with a unified communication interface

We developed a multi-channel hierarchical event-dispatch system that allows different panels or customized web applications to effectively communicate with each other. This design has two components: (1) A unified communication protocol embedded in different panels; and (2) A central event-dispatch hub.

The unified communication protocol defines how different types of panels or web applications handle users' operations (event emitter) and respond to operations from other panels (event listener). The event emitter and listener from different panels or web applications share a common set of users' operations (e.g., navigation and highlight), even though they display distinct data types. For example, when users select a region-of-interest in the genome browser panel, the event listener in the 3D genome structure model panel will receive the genomic coordinates and synchronously highlight the corresponding chromatin segments in the 3D genome structure model. Furthermore, both event emitter and event listener have channel ID as attribute, enabling customized synchronization across linked panels in the same communication channel.

The central event-dispatch hub in our event-dispatch framework adaptively handles communication across panels and web applications (**Supplementary Fig.** S2). It automatically recognizes the source of event emitter and dispatches event message to other panels and web applications in the same communication channel. In the simplest case, this hub uses the event hub interface from the Golden Layout library to support communication across panels on the same web page. Additionally, this design supports communication across web browser tabs hosted under the same domain using the PostMessage HTML5 API and the Broadcast channel API. Lastly, we developed a web browser extension named Nucleome Bridge to enable communication across web applications in different domains. Nucleome Bridge works as a bridge to connect Nucleome Browser with web applications hosted on the allowed URLs once users implement event emitter and event listener in their web applications. This feature provides users the opportunity for integrative visualization of datasets stored in different web-based sources effectively. Note that we can easily expand the allowed URLs to support customized web applications built by users.

Together, this multi-channel, cross-domain, and cross-tab communication system drastically improves the scalability and navigation experience of heterogeneous datasets by defining universal communications across panels, web tabs, and web portals in different domains.

To further allow users to visualize new data types in Nucleome Browser or link other web applications with our platform, we allow users to test their code in CodePen (https://codepen.io), jsfiddle (https://jsfiddle.net), or local server. Once the code is wrapped into a web component, it can be added to the panel system of Nucleome Browser, with a simple configuration change, since all communications across panels or web applications use the same interface. This ease of use and integration is the cornerstone of our multimodal data ecosystem, enabling Nucleome Browser to be highly adaptive to new data types that will emerge in the future.

Note that the communication protocol and the event-dispatch hub have been wrapped into the nb-dispatch JavaScript library (https://github.com/nucleome/nb-dispatch or https://www.npmjs.com/package/@nucleome/nb-dispatch). Nucleome Bridge web browser exten-

sion is available at the Chrome Web Store (https://tinyurl.com/nb-bridge).

References

[1] Bintu B, Mateo LJ, Su JH, Sinnott-Armstrong NA, Parker M, Kinrot S, et al. Super-resolution chromatin tracing reveals domains and cooperative interactions in single cells. Science. 2018;362(6413):eaau1783.

[2] Nir G, Farabella I, Estrada CP, Ebeling CG, Beliveau BJ, Sasaki HM, et al. Walking along chromosomes with super-resolution imaging, contact maps, and integrative modeling. PLoS Genetics. 2018;14(12):e1007872.

[3] Kent WJ, Sugnet CW, Furey TS, Roskin KM, Pringle TH, Zahler AM, et al. The human genome browser at UCSC. Genome Research. 2002;12(6):996-1006.

[4] Hubbard T, Barker D, Birney E, Cameron G, Chen Y, Clark L, et al. The Ensembl genome database project. Nucleic Acids Research. 2002;30(1):38-41.

[5] Li D, Hsu S, Purushotham D, Sears RL, Wang T. WashU epigenome browser update 2019. Nucleic Acids Research. 2019;47(W1):W158-65.

[6] Robinson JT, Thorvaldsdóttir H, Winckler W, Guttman M, Lander ES, Getz G, et al. Integrative genomics viewer. Nat Biotechnol. 2011;29(1):24-6.

[7] Skinner ME, Uzilov AV, Stein LD, Mungall CJ, Holmes IH. JBrowse: a next-generation genome browser. Genome Res. 2009 Sep;19(9):1630-8.

[8] Wang Y, Song F, Zhang B, Zhang L, Xu J, Kuang D, et al. The 3D Genome Browser: a web-based browser for visualizing 3D genome organization and long-range chromatin interactions. Genome Biology. 2018;19(1):151.

[9] Cao X, Yan Z, Wu Q, Zheng A, Zhong S. GIVE: portable genome browsers for personal websites. Genome Biology. 2018;19(1):92.

[10] Robinson JT, Turner D, Durand NC, Thorvaldsdottir H, Mesirov JP, Aiden EL. Juicebox. js provides a cloud-based visualization system for Hi-C data. Cell Systems. 2018;6(2):256-8.

[11] Kerpedjiev P, Abdennur N, Lekschas F, McCallum C, Dinkla K, Strobelt H, et al. HiGlass: web-based visual exploration and analysis of genome interaction maps. Genome Biology. 2018;19(1):125.

[12] Stevens TJ, Lando D, Basu S, Atkinson LP, Cao Y, Lee SF, et al. 3D structures of individual mammalian genomes studied by single-cell Hi-C. Nature. 2017;544(7648):59-64.

[13] Payne AC, Chiang ZD, Reginato PL, Mangiameli SM, Murray EM, Yao CC, et al. In situ genome sequencing resolves DNA sequence and structure in intact biological samples. Science. 2021;371(6532):eaay3446.

[14] Chen Y, Zhang Y, Wang Y, Zhang L, Brinkman EK, Adam SA, et al. Mapping 3D genome organization relative to nuclear compartments using TSA-Seq as a cytological ruler. Journal of Cell Biology. 2018;217(11):4025-48.

[15] Yildirim A, Hua N, Boninsegna L, Polles G, Gong K, Hao S, et al. Mapping the nuclear microenvironment of genes at a genome-wide scale. bioRxiv. 2021.

[16] Zhou X, Lowdon RF, Li D, Lawson HA, Madden PA, Costello JF, et al. Exploring long-range genome interactions using the WashU Epigenome Browser. Nature Methods. 2013;10(5):375.

[17] Durand NC, Robinson JT, Shamim MS, Machol I, Mesirov JP, Lander ES, et al. Juicebox provides a visualization system for Hi-C contact maps with unlimited zoom. Cell Systems. 2016;3(1):99-101.

[18] Su JH, Zheng P, Kinrot SS, Bintu B, Zhuang X. Genome-scale imaging of the 3D organization and transcriptional activity of chromatin. Cell. 2020;182(6):1641-59.

[19] Takei Y, Yun J, Zheng S, Ollikainen N, Pierson N, White J, et al. Integrated spatial genomics reveals global architecture of single nuclei. Nature. 2021;590(7845):344-50.

[20] Buels R, Yao E, Diesh CM, Hayes RD, Munoz-Torres M, Helt G, et al. JBrowse: a dynamic web platform for genome visualization and analysis. Genome Biology. 2016;17(1):66.

[21] Finn EH, Pegoraro G, Brandao HB, Valton AL, Oomen ME, Dekker J, et al. Extensive heterogeneity and intrinsic variation in spatial genome organization. Cell. 2019;176(6):1502-15.

[22] Wang Y, Zhang Y, Zhang R, van Schaik T, Zhang L, Sasaki T, et al. SPIN reveals genome-wide landscape of nuclear compartmentalization. Genome Biology. 2021;22(1):36.

[23] Hua N, Tjong H, Shin H, Gong K, Zhou XJ, Alber F. Producing genome structure populations with the dynamic and automated PGS software. Nature Protocols. 2018;13(5):915.

[24] Paulsen J, Sekelja M, Oldenburg AR, Barateau A, Briand N, Delbarre E, et al. Chrom3D: three-dimensional genome modeling from Hi-C and nuclear lamin-genome contacts. Genome Biology. 2017;18(1):21.

[25] Rego N, Koes D. 3Dmol.js: molecular visualization with WebGL. Bioinformatics. 2015;31(8):1322-4.

[26] Williams E, Moore J, Li SW, Rustici G, Tarkowska A, Chessel A, et al. Image Data Resource: a bioimage data integration and publication platform. Nature Methods. 2017;14(8):775-81.

[27] Reiff SB, Schroeder AJ, Kırlı K, Cosolo A, Bakker C, Lee S, et al. The 4D Nucleome Data Portal as a resource for searching and visualizing curated nucleomics data. Nature Communications. 2022;13(1):2365.