

Rapid Prototyping for Coordinated Views of Multi-scale Spatial and Abstract Data: A Grammar-based Approach

P. Harth*, A. Bast, J. Troidl, B. Meulemeester, H. Pfister, J. Beyer, M. Oberlaender, H.-C. Hege, D. Baum

Visual Computing for Biology and Medicine (EG VCBM)

Norrköping, September 20-22, 2023

Team



Jakob Macke



Jan Bölts



Daniel Baum



Hans-Christian Hege



Visual Computing Group



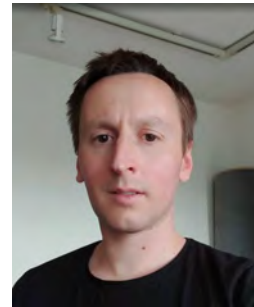
Johanna Beyer



Hanspeter Pfister



Marcel Oberlaender



Philipp Harth



Jakob Troidl



Maria Royo Cana



Arco Bast



Felipe Yanez



Bjorge Meulemeester



Rieke Fruengel

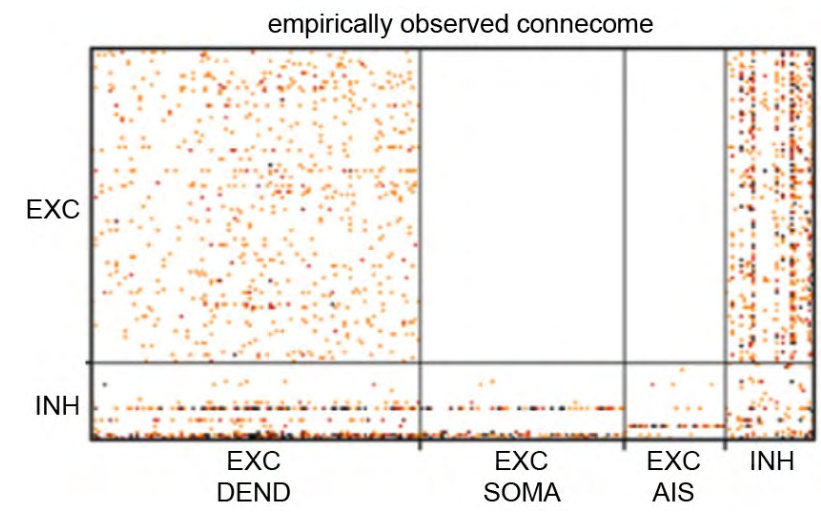
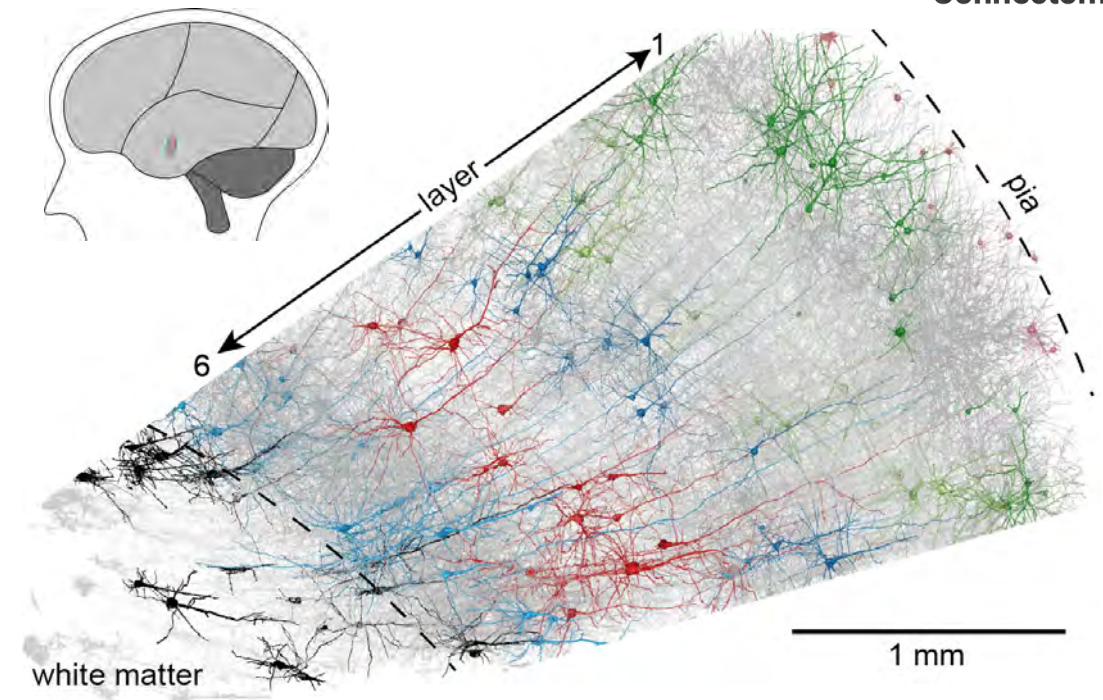
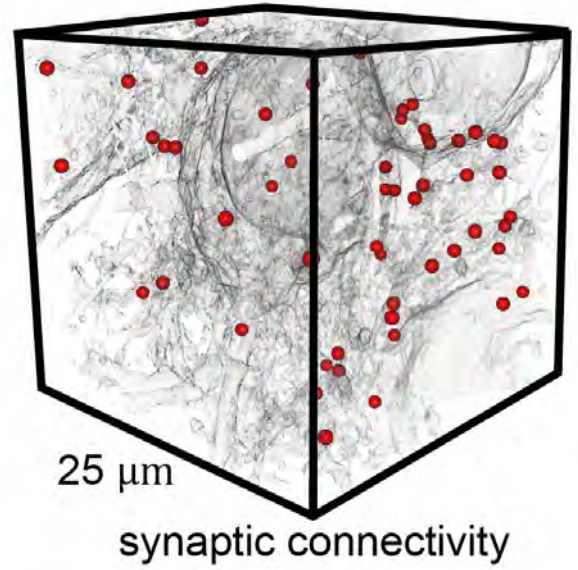
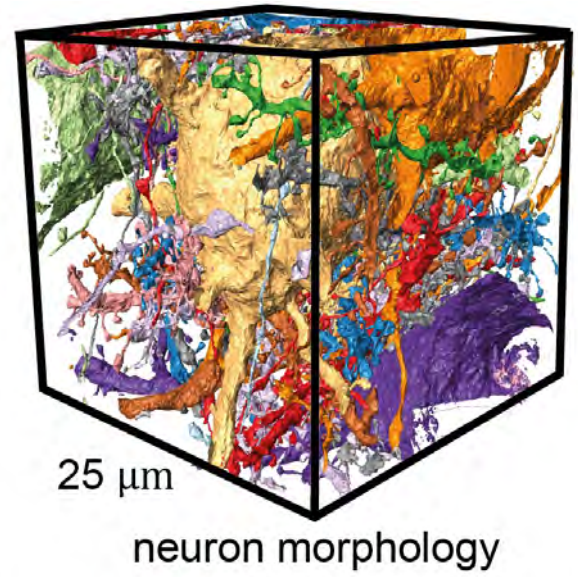
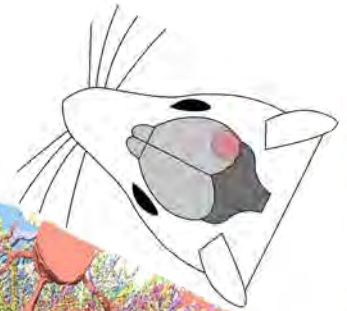
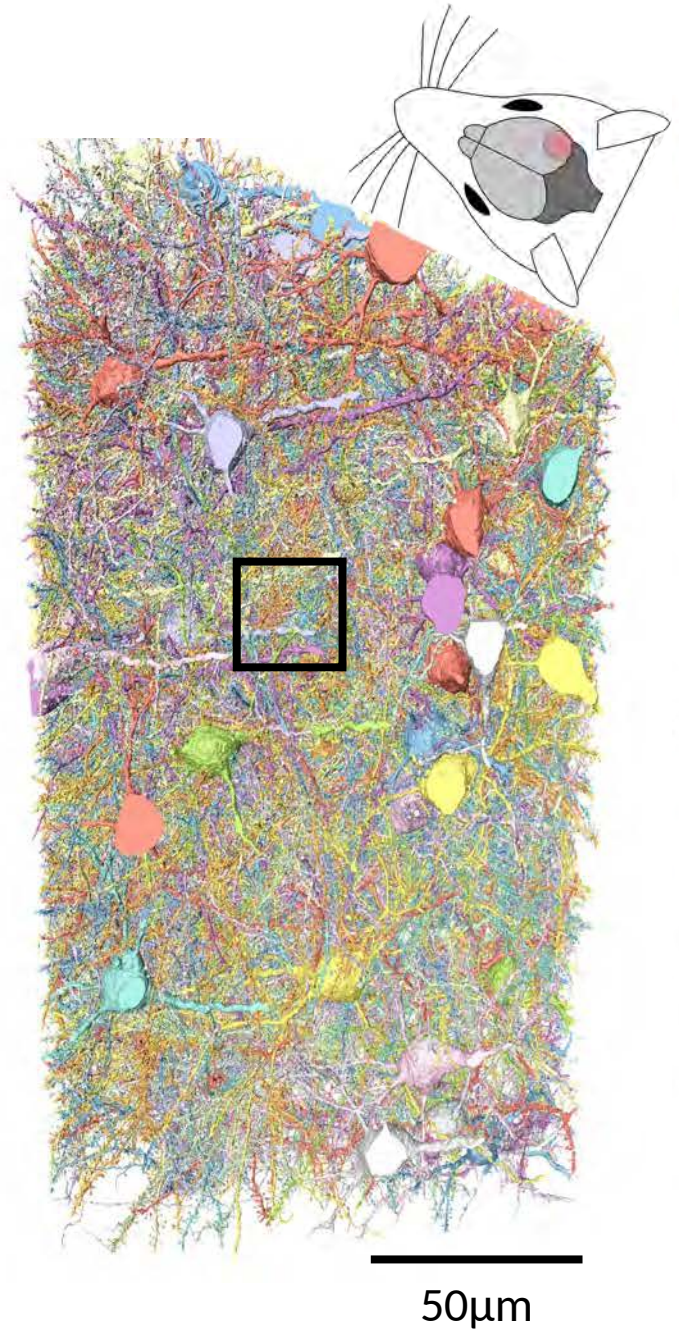


MAX PLANCK INSTITUTE
FOR NEUROBIOLOGY OF BEHAVIOR
- CAESAR



- **Visual computing in computational neuroscience**
- Design considerations and technical realization
- Application of tool and preliminary results
- Summary and outlook

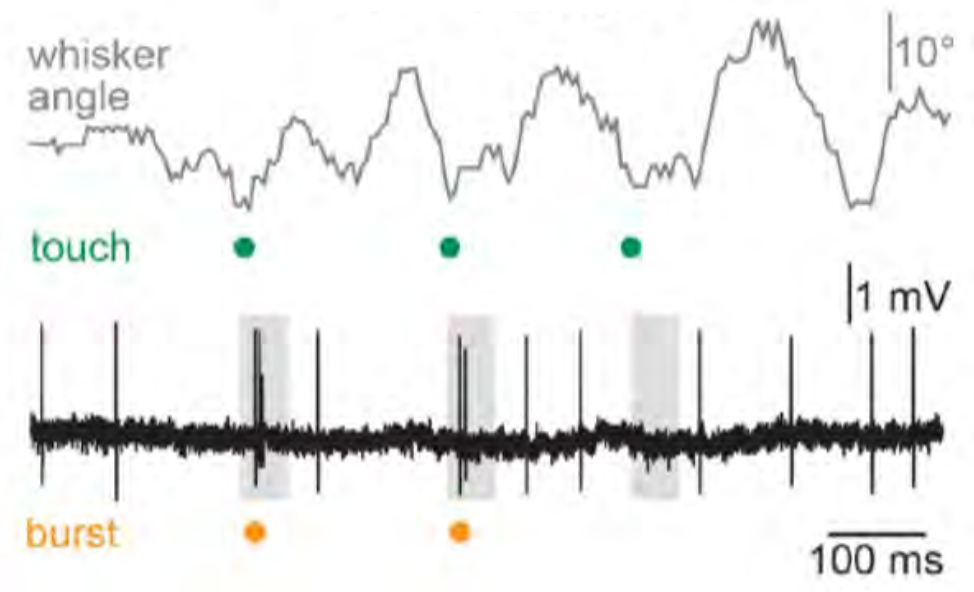
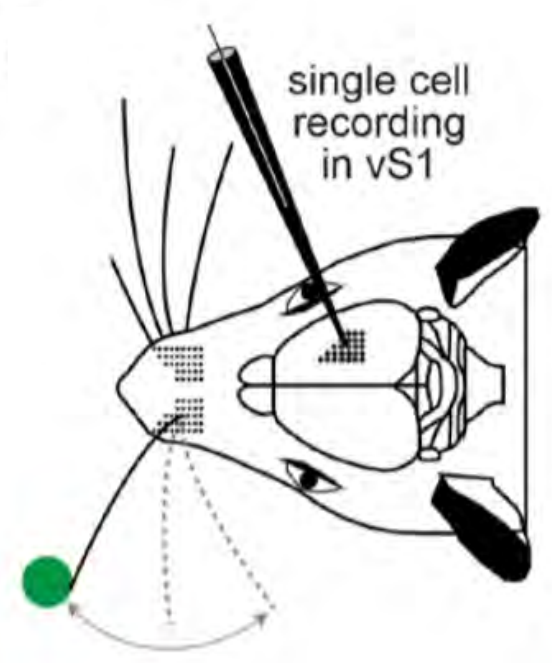
Network structure



mouse dataset
Turner et al., 2022
human dataset
Shapson-Coe et al., 2021

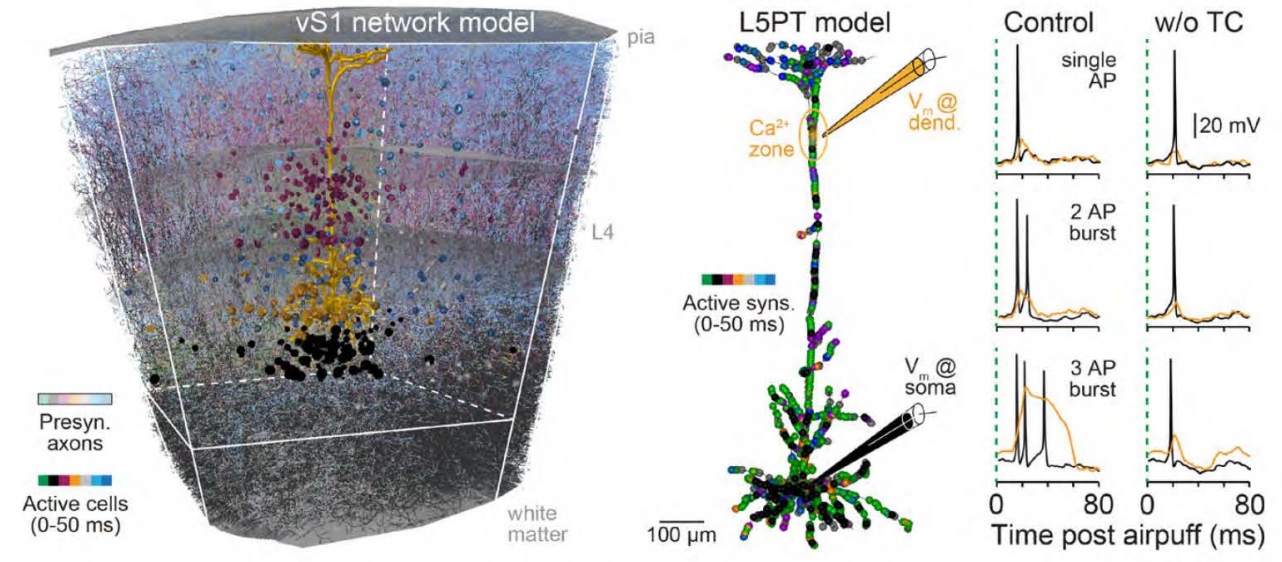
Network function

EXPERIMENT



Bast et al. 2023

MODEL

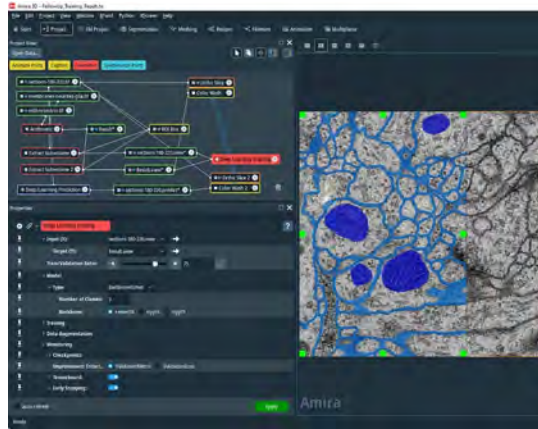
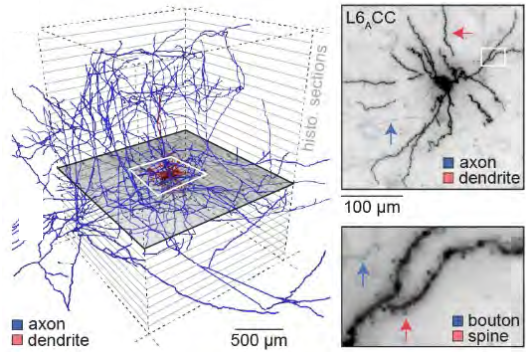


Bast et al. 2023

- High-dimensional parameter/feature space ()
- Spatiotemporal data (voltage traces at different locations)
- million simulation runs

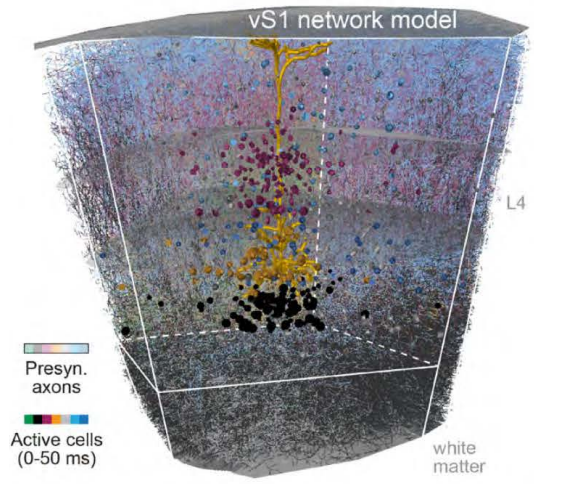
Visual computing in computational neuroscience

model generation



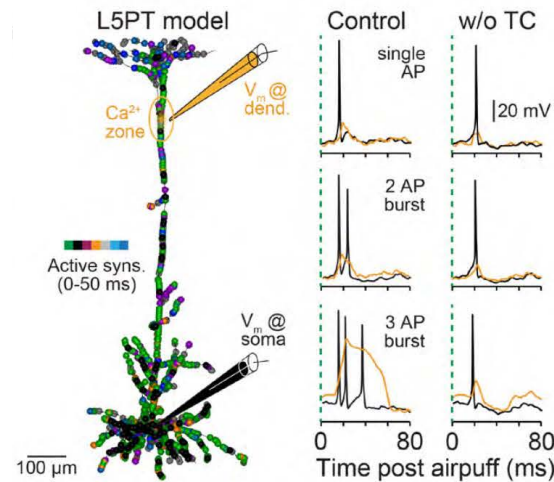
Amira software

structural model

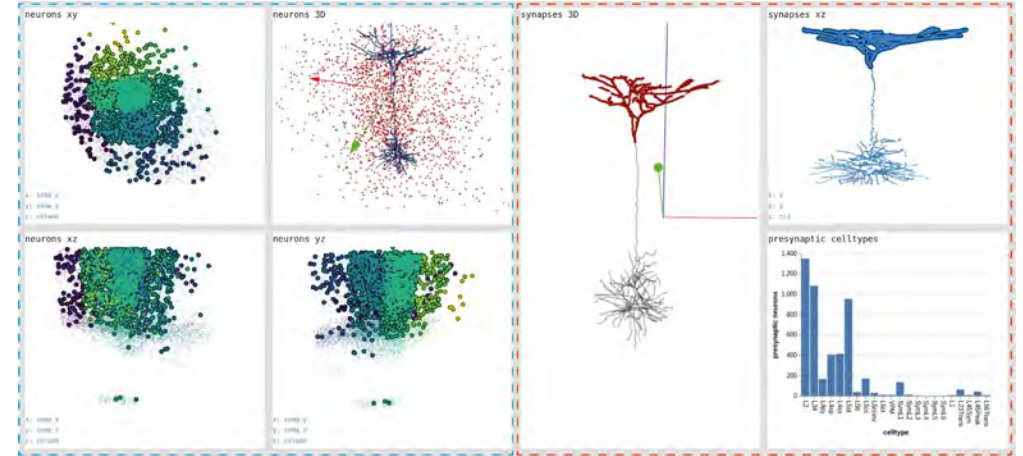


Bast et al. 2023

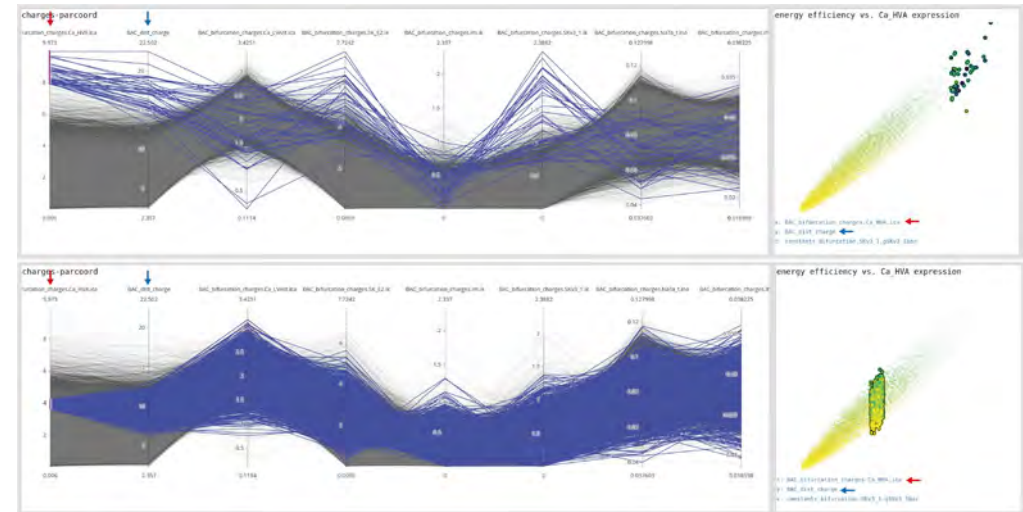
biophysical model



network analysis



model parameter analysis



goal: understand the neural basis of behavior

- Visual computing in computational neuroscience
- **Design considerations and technical realization**
- Application of tool and preliminary results
- Summary and outlook

Design goals

- G1.** Support the combination of different view types.
- G2.** Integrate existing visualization frameworks and libraries to facilitate code reuse and reduce development time.
- G3.** Enable interactive and iterative refinement of views, workspace layout, and interaction logic.
- G4.** Avoid hard-coded or domain-specific ontologies tied to a priori known analysis questions or workflows.
- G5.** Support dynamic transitions between different workspace configurations.

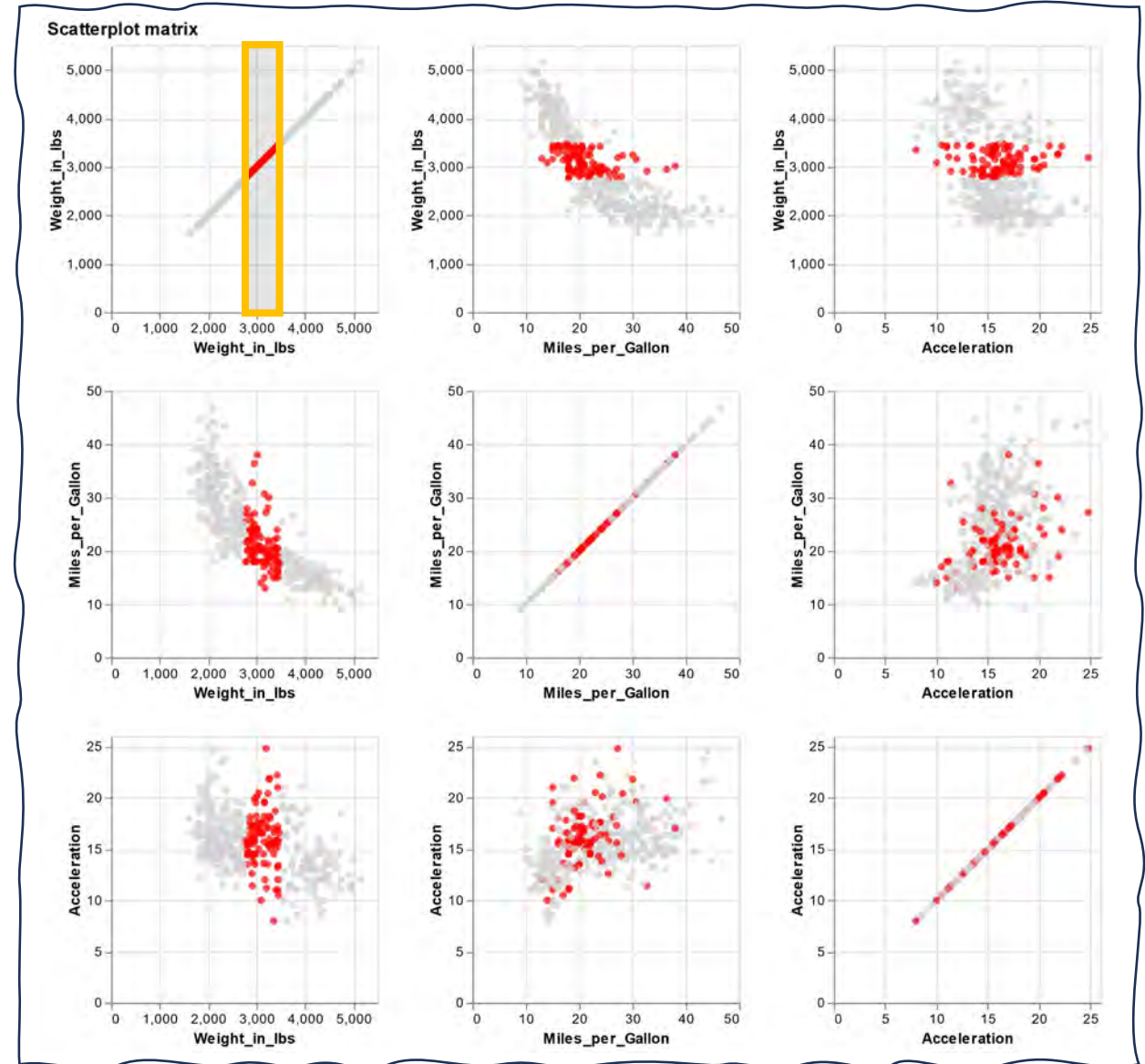
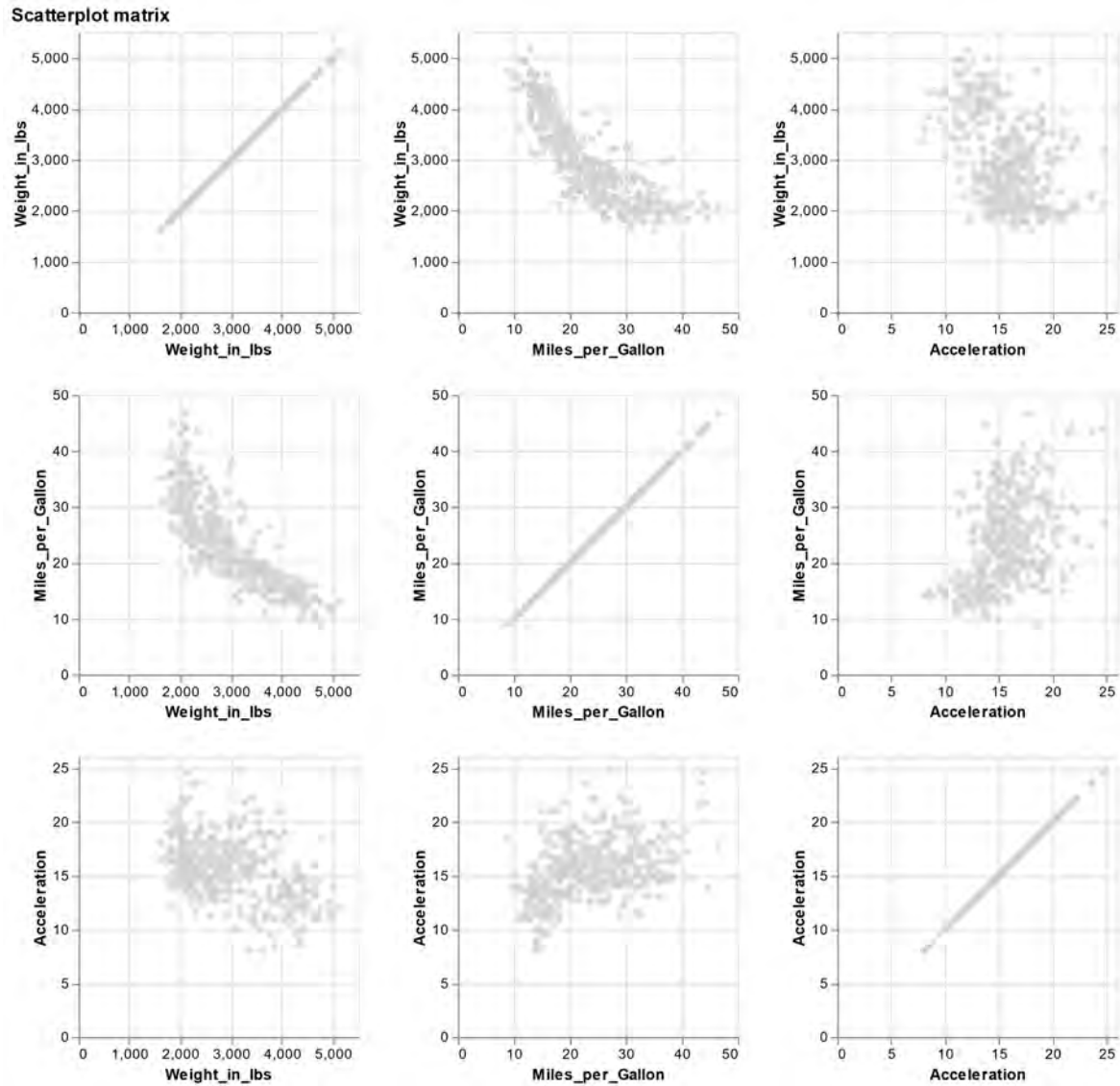
key features of tool

web-based coordinated views

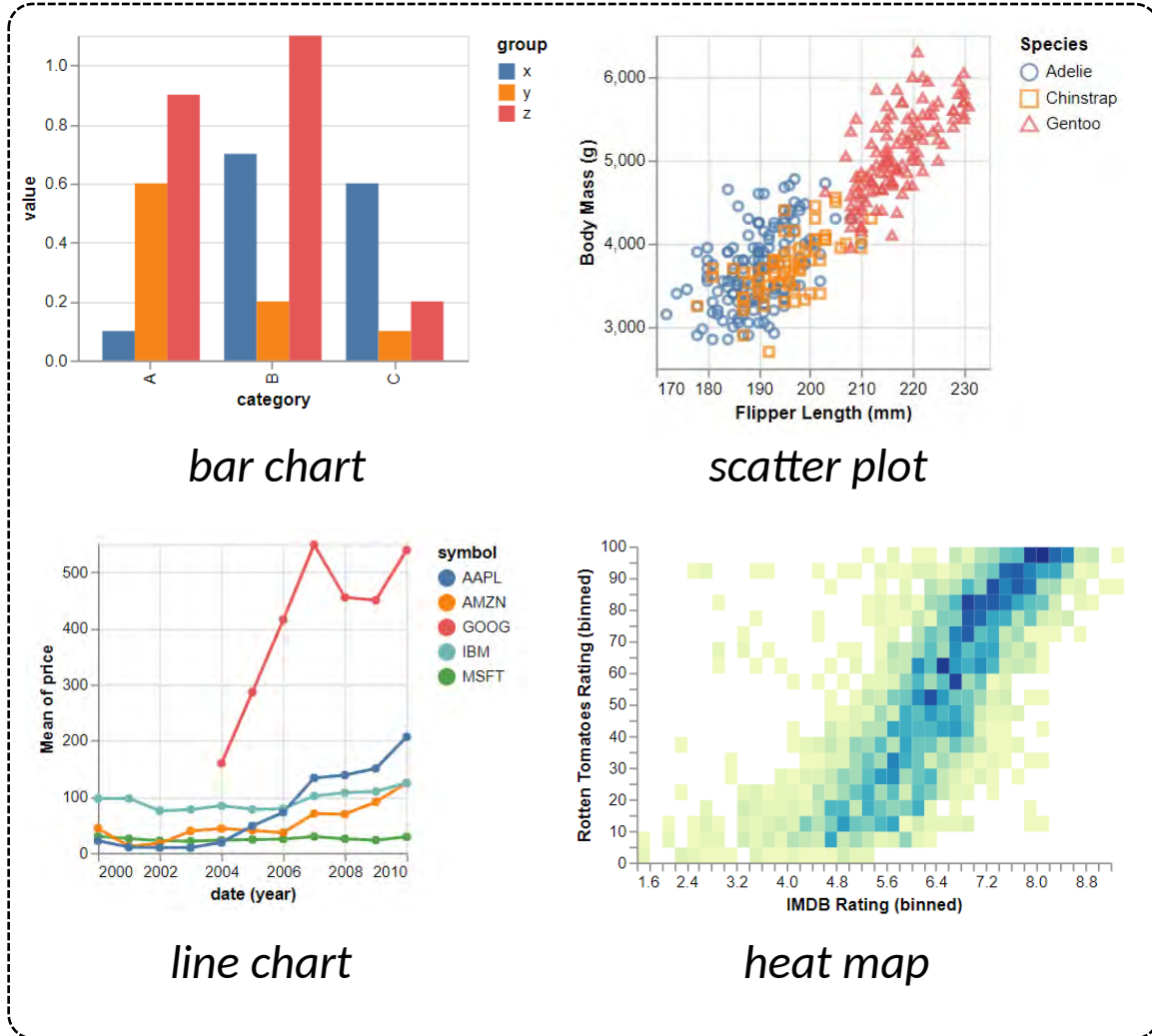
combine InfoVis & SciVis views

grammar-based specification

Coordinated views – *brushing* & *linking*

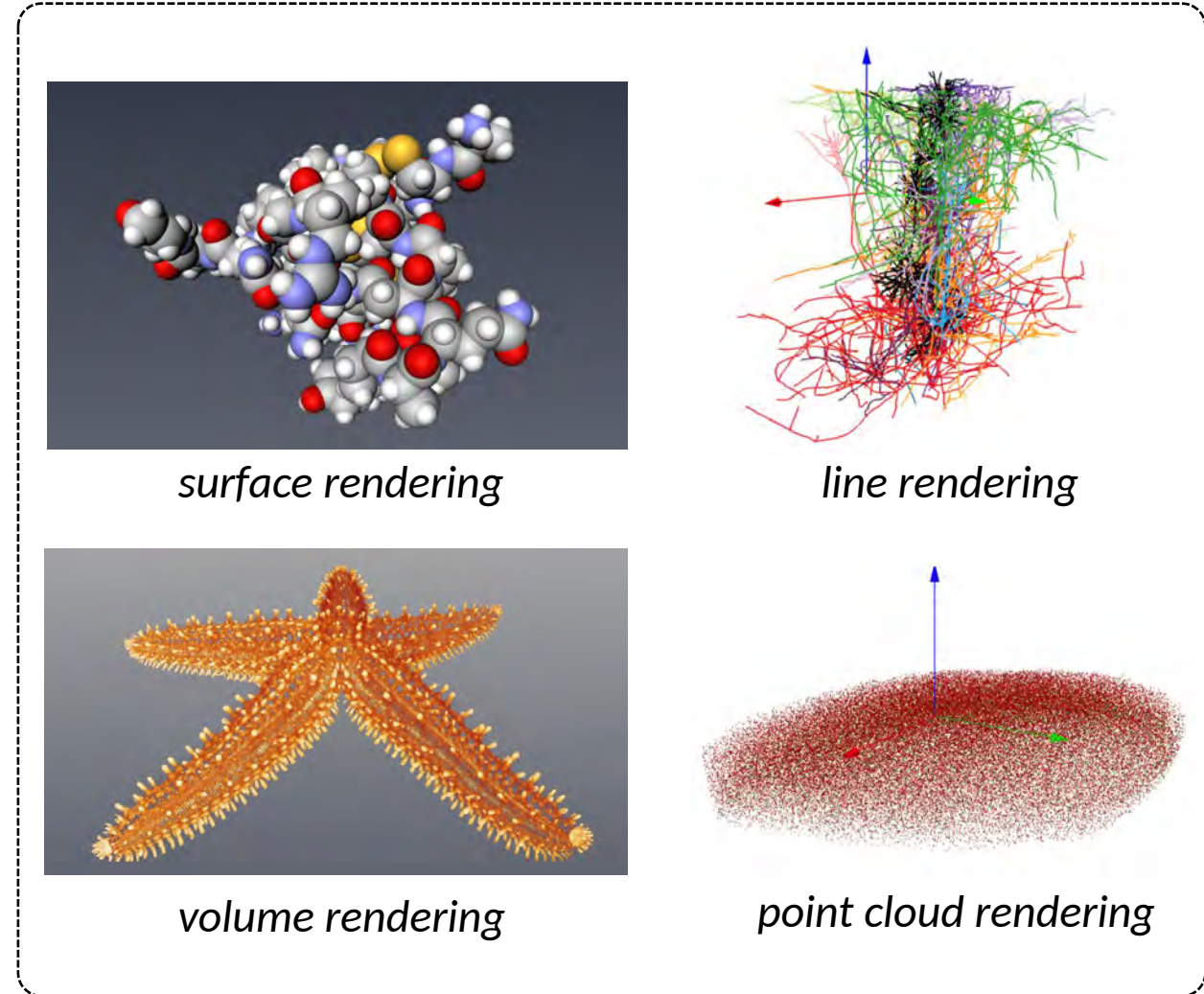


InfoVis: Information Visualization (abstract data)



<https://vega.github.io/vega-lite/examples/>

SciVis: Scientific Visualization (spatial data)

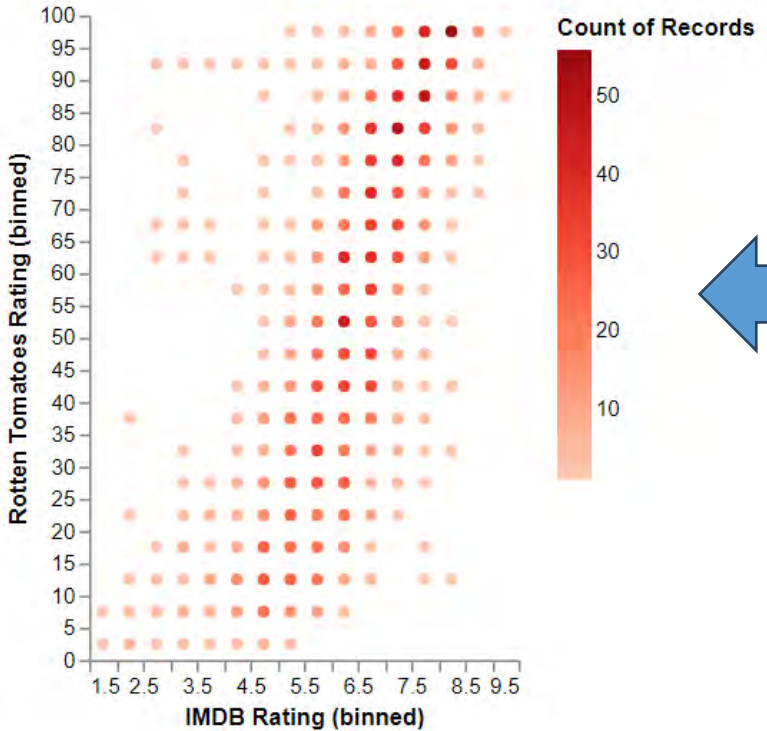
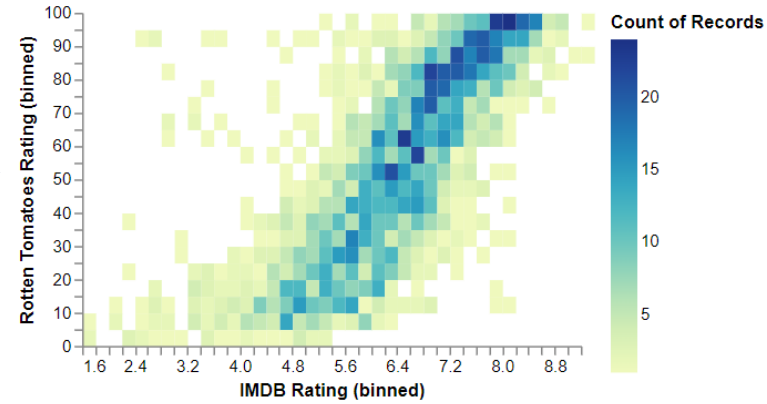


<https://git.zib.de/amira/demos>; cortexinsilico.zib.de

Grammar-based specification – Vega-Lite example

JSON-based specification

```
{
  "$schema": "https://vega.github.io/schema/vega-lite/v5.json",
  "data": {"url": "data/movies.json"},
  "transform": [{
    "filter": {"and": [
      {"field": "IMDB Rating", "valid": true},
      {"field": "Rotten Tomatoes Rating", "valid": true}
    ]}
  ]},
  "mark": "rect",
  "width": 300,
  "height": 200,
  "encoding": {
    "x": {
      "bin": {"maxbins": 60},
      "field": "IMDB Rating",
      "type": "quantitative"
    },
    "y": {
      "bin": {"maxbins": 40},
      "field": "Rotten Tomatoes Rating",
      "type": "quantitative"
    },
    "color": {
      "aggregate": "count",
      "type": "quantitative"
    }
  },
  "config": {
    "view": {
      "stroke": "transparent"
    }
  }
}
```



modified specification

```
{
  "$schema": "https://vega.github.io/schema/vega-lite/v5.json",
  "data": {"url": "data/movies.json"},
  "transform": [{
    "filter": {"and": [
      {"field": "IMDB Rating", "valid": true},
      {"field": "Rotten Tomatoes Rating", "valid": true}
    ]}
  ]},
  "mark": "circle",
  "width": 200,
  "height": 300,
  "encoding": {
    "x": {
      "bin": {"maxbins": 20},
      "field": "IMDB Rating",
      "type": "quantitative"
    },
    "y": {
      "bin": {"maxbins": 40},
      "field": "Rotten Tomatoes Rating",
      "type": "quantitative"
    },
    "color": {
      "aggregate": "count",
      "type": "quantitative",
      "scale": {
        "scheme": "reds"
      }
    }
  },
  "config": {
    "view": {
      "stroke": "transparent"
    }
  }
}
```



<https://vega.github.io/>

Technical realization

JSON-based specification

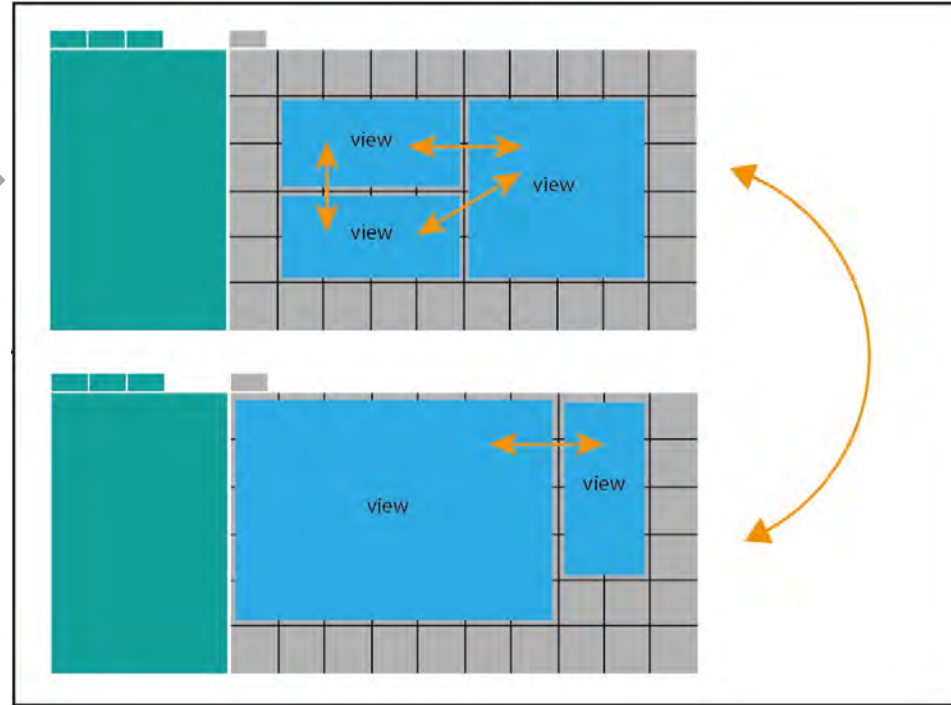
```

{
  "views" : {
    <view_name> : {
      "type" : <view_type>,
      "dataTable" : <table_name>,
      "dataColumn" : [<column_name>, ...],
      "configuration" : {...}
    }
  },
  "grid" : { // partitions workspace
    "cols" : int, "rowHeight" : int, "width" : int
  },
  "layouts" : {
    <layout_name> : {
      "view" : <view_name>,
      "x" : int, "y" : int, "w" : int, "h" : int
    }
  }
  "initialLayout" : <layout_name>
  "interactions" : {
    <source_view_name> : [{ // source of interaction event
      "filter" : {
        "currentLayout" : <layout_name>,
        "interactionType" : <interaction_type>
        "selectedEntityType" : <type_name>
        "numSelectedRange" : int | [int, int],
      }
      "action" : {
        "assignData" : [<target_view_name>, ...],
        "clearData" : [<target_view_name>, ...],
        "assignSelectionAsData" : [<target_view_name>, ...],
        "assignSelection" : [<target_view_name>, ...],
        "intersectSelection" : [<target_view_name>, ...],
        "unionSelection" : [<target_view_name>, ...],
        "clearSelection" : [<target_view_name>, ...],
        "changeLayout" : <layout_name>
      }
    }
  }
}

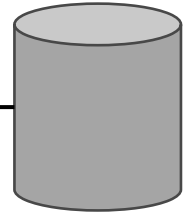
```



web-based frontend



Python backend



view

BarchartView	ParallelCoordsView	AnatomicalView
Vega-Lite	Plotly.js	Babylon.js

JavaScript visualization libraries

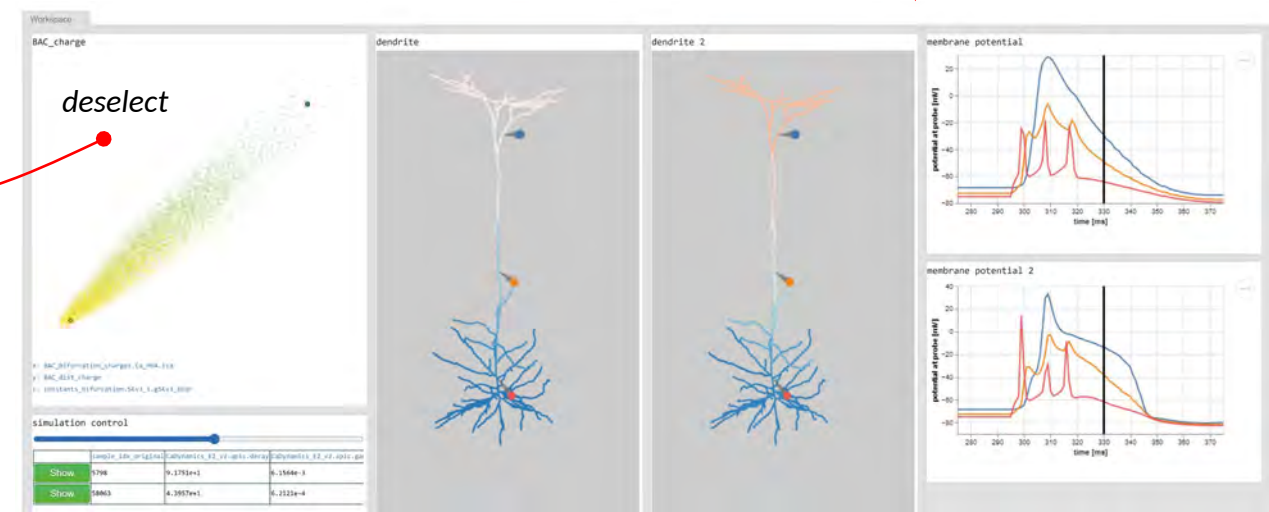
Technical realization



Technical realization



workspace layout L1



workspace layout L2

Simple demo – specifying linked views for point cloud data

The screenshot displays a software interface with a configuration editor on the left and a workspace on the right. The configuration editor shows a JSON object defining a grid and a layout named 'L1' with three views. The workspace contains three empty panes, each labeled 'not specified:'.

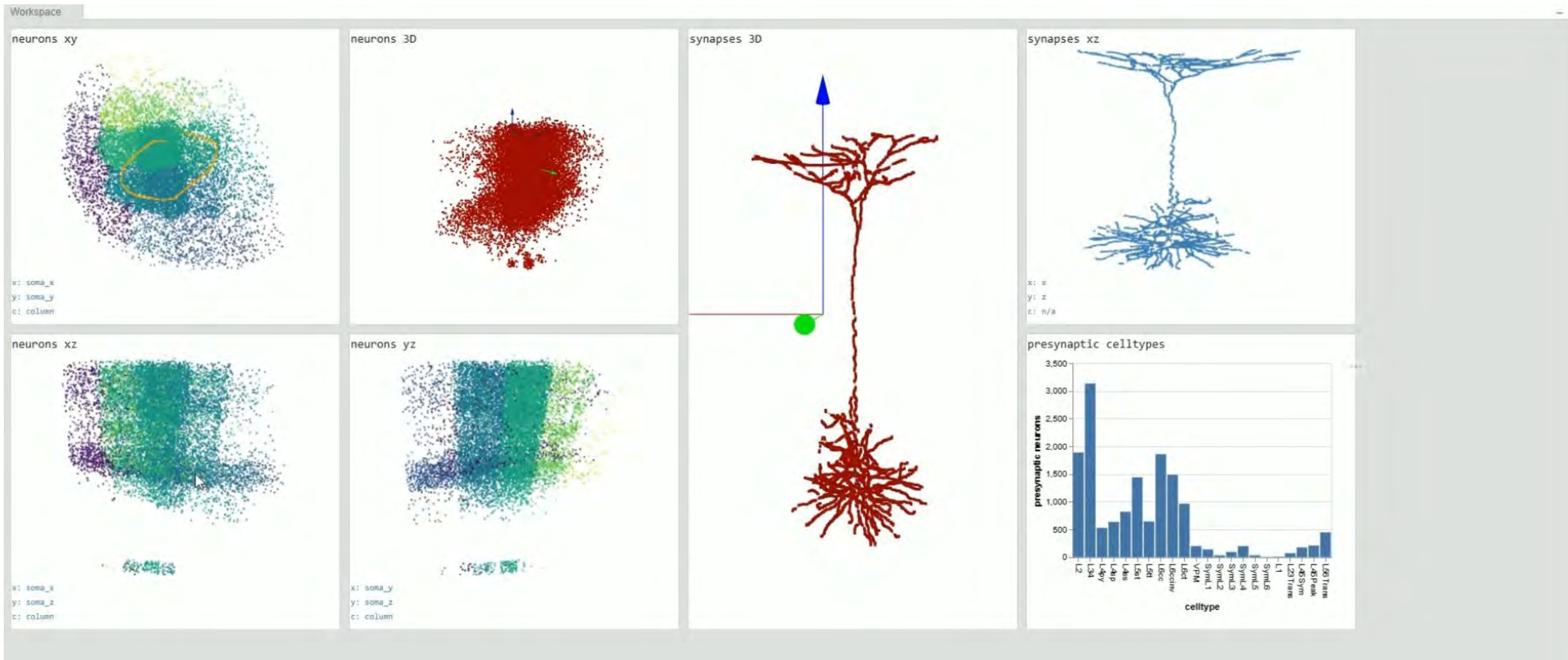
```
{
  "grid": {
    "cols": 18,
    "rowHeight": 80,
    "width": 1800
  },
  "initialLayout": "L1",
  "interactions": {},
  "layouts": {
    "L1": [
      {
        "h": 8,
        "view": "",
        "w": 8,
        "x": 4,
        "y": 0
      },
      {
        "h": 4,
        "view": "",
        "w": 4,
        "x": 0,
        "y": 0
      },
      {
        "h": 4,
        "view": "",
        "w": 4,
        "x": 0,
        "y": 4
      }
    ]
  }
}
```

Workspace panes:

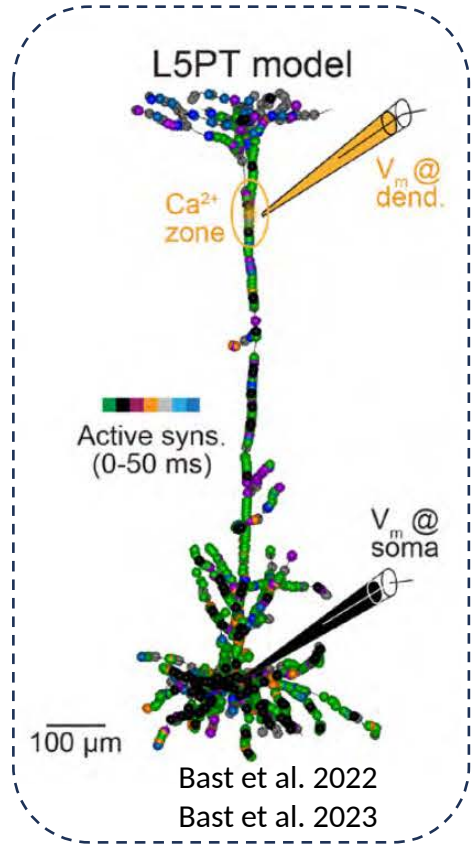
- Top-left: not specified:
- Top-right: not specified:
- Bottom-left: not specified:

- Visual computing in computational neuroscience
- Design considerations and technical realization
- **Application of tool and preliminary results**
- Summary and outlook

Tool in usage with neuroscientific model data



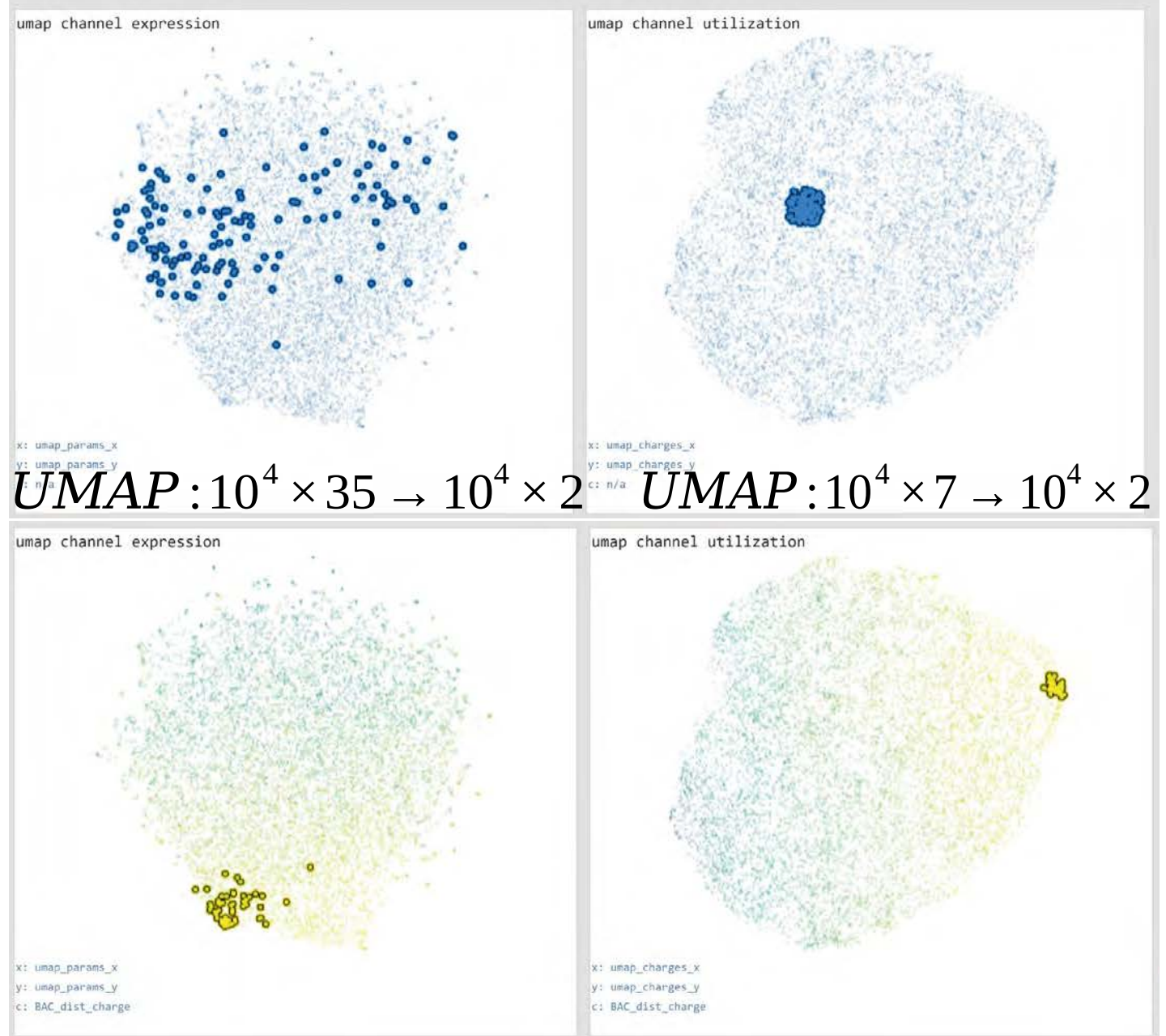
Biophysical model parameter analysis – preliminary results



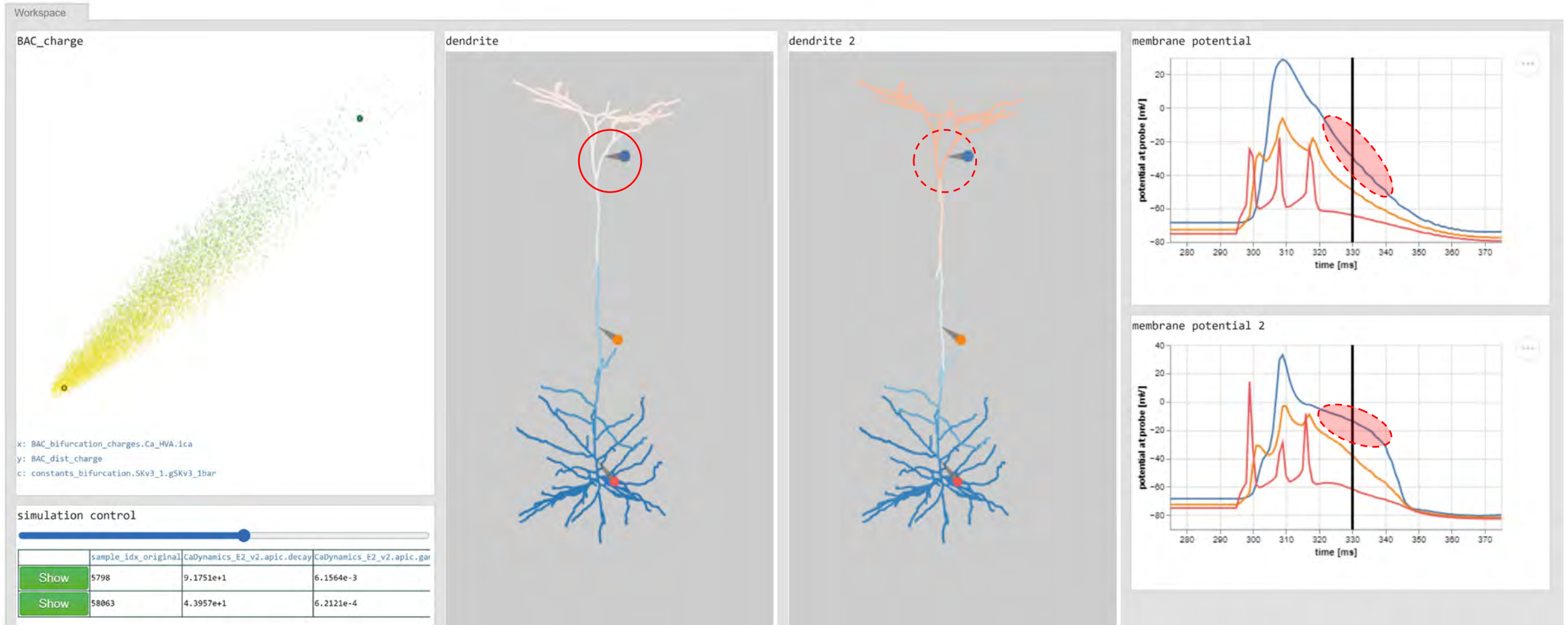
channel utilization features for zone

channel expression parameters

uniformly sampled simulation runs



Biophysical model parameter analysis – preliminary results



comparing membrane potentials on neuron morphology for two selected parameter samples

- Visual computing in computational neuroscience
- Design considerations and technical realization
- Application of tool and preliminary results
- **Summary and outlook**

Key features of tool

- Web-based coordinated views
- Combines InfoVis & SciVis views from different JavaScript libraries
- JSON-based specification of layout and interactions
- Code: <https://github.com/zibneuro/coordinated-views>

Future work

- Integrate vaex (<https://vaex.io/>) to improve performance of Python backend
- Integrate backend server into Jupyter notebook
- Develop expressive syntax for configuration of 3D view
- Demonstrate generalizability with data from other domains

Thank you for your attention!

Acknowledgements:

SPP 2041 Computational Connectomics (DFG)



References

- Bast, A., & Oberlaender, M. (2022). Ion channel distributions in cortical neurons are optimized for energy-efficient active dendritic computations. *BioRxiv*, 2021.12.11.472235. <https://doi.org/10.1101/2021.12.11.472235>
- Bast, A., Guest, J. M., Fruengel, R., Narayanan, R. T., de Kock, C. P. J., & Oberlaender, M. (2023). Thalamus drives active dendritic computations in cortex. *BioRxiv*. <https://doi.org/10.1101/2021.10.21.465325>
- Boelts, J., **Harth**, P., Gao, R., Udvary, D., Yáñez, F., Baum, D., Hege, H.-C., Oberlaender, M., & Macke, J. H. (2023). Simulation-based inference for efficient identification of generative models in computational connectomics. *PLOS Computational Biology*; accepted for publication
- **Harth**, P., Vohra, S., Udvary, D., Oberlaender, M., Hege, H.-C., & Baum, D. (2022). A Stratification Matrix Viewer for Analysis of Neural Network Data. In R. G. Raidou, B. Sommer, T. W. Kuhlen, M. Krone, T. Schultz, & H.-Y. Wu (Eds.), *Eurographics Workshop on Visual Computing for Biology and Medicine (VCBM)*. The Eurographics Association. <https://doi.org/10.2312/vcbm.20221194>
- **Harth**, P., Bast, A., Troidl, J., Meulemeester, B., Pfister, H., Beyer, J., Oberlaender, M., Hege, H.-C., & Baum, D. (2023). Rapid Prototyping for Coordinated Views of Multi-Scale Spatial and Abstract Data: A Grammar-based Approach. *Eurographics Workshop on Visual Computing for Biology and Medicine (VCBM)*; accepted for publication
- Shapson-Coe, A., Januszewski, M., Berger, D. R., Pope, A., Wu, Y., Blakely, T., Schalek, R. L., Li, P. H., Wang, S., Maitin-Shepard, J., Karlupia, N., Dorkenwald, S., Sjostedt, E., Leavitt, L., Lee, D., Bailey, L., Fitzmaurice, A., Kar, R., Field, B., ... Lichtman, J. W. (2021). A connectomic study of a petascale fragment of human cerebral cortex. *BioRxiv*. <https://doi.org/10.1101/2021.05.29.446289>
- Turner, N. L., Macrina, T., Bae, J. A., Yang, R., Wilson, A. M., Schneider-Mizell, C., Lee, K., Lu, R., Wu, J., Bodor, A. L., Bleckert, A. A., Brittain, D., Froudarakis, E., Dorkenwald, S., Collman, F., Kemnitz, N., Ih, D., Silversmith, W. M., Zung, J., ... Seung, H. S. (2022). Reconstruction of neocortex: Organelles, compartments, cells, circuits, and activity. *Cell*, 185(6), 1082-1100.e24. <https://doi.org/https://doi.org/10.1016/j.cell.2022.01.023>
- Udvary, D., **Harth**, P., Macke, J. H., Hege, H.-C., de Kock, C. P. J., Sakmann, B., & Oberlaender, M. (2022). The impact of neuron morphology on cortical network architecture. *Cell Reports*, 39(2). <https://doi.org/10.1016/j.celrep.2022.110677>

Implementation – code snippets



parallel coordinates
Plotly.js

WebGL-enabled scatter plot
[Lekschas, 2023]

```
class CoordinatedView extends React.Component {
  constructor(props) {
    super(props);

    this.name = props.name;
    this.dataTable = props.dataTable;
    this.dataColumn = props.dataColumn;
  }
}
```

```
class ReglScatterPlot extends CoordinatedView {
  constructor(props) {
    super(props);
  }

  handleSelect(eventArgs) {
    this.notifyInteraction(new SelectionEvent(this.name, this.dataTable).setIndices(eventArgs.pointIndices));
  }

  handleDeselect() {
    this.notifyInteraction(new SelectionEvent(this.name, this.dataTable).setDeselect());
  }

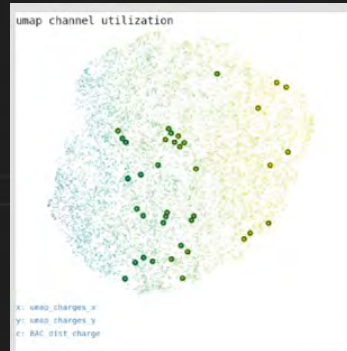
  render() {
    return <div style={styleBackground}><canvas ref={this.myRef} width={this.width} height={this.height}></canvas></div>;
  }

  componentDidMount() {
    const canvas = this.myRef.current;
    canvas.fillStyle = "white";
    if (this.dataColumn.length < 2) {
      return;
    }

    const { width, height } = canvas.getBoundingClientRect();

    this.scatterplot = createScatterplot({

```



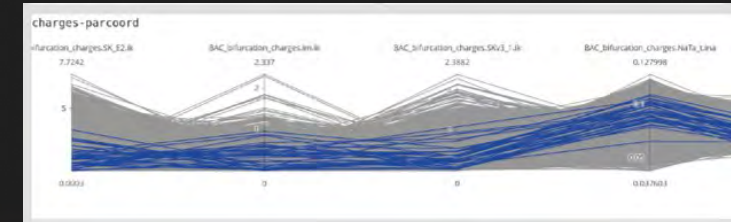
```
class ParallelCoordinates extends CoordinatedView {
  constructor(props) {
    super(props);

    this.state = {
      traces: empty_traces,
    };
    this.constraints = undefined;
  }

  render() {
    this.notifyInteraction(new SelectionEvent(this.name, this.dataTable).setIndices(filtered_indices));
  }

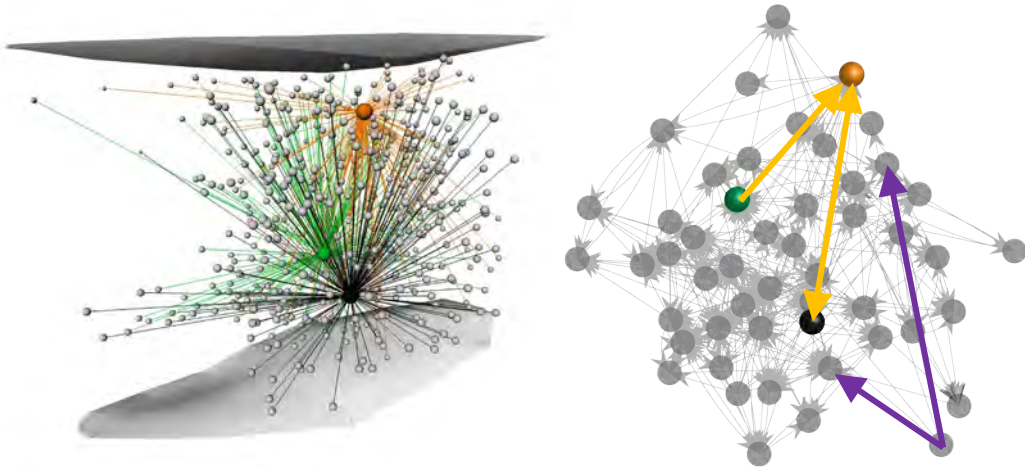
  render() {
    const layout = {
      margin: {
        l: this.marginLeft,
        r: this.marginRight,
        t: this.marginTop,
        b: this.marginBottom,
      },
      width: this.width,
      height: this.height
    };

    return <Plot
      data={this.state.traces}
      layout={layout}
      onUpdate={this.handleEvent.bind(this)}
    />;
  }
}
```



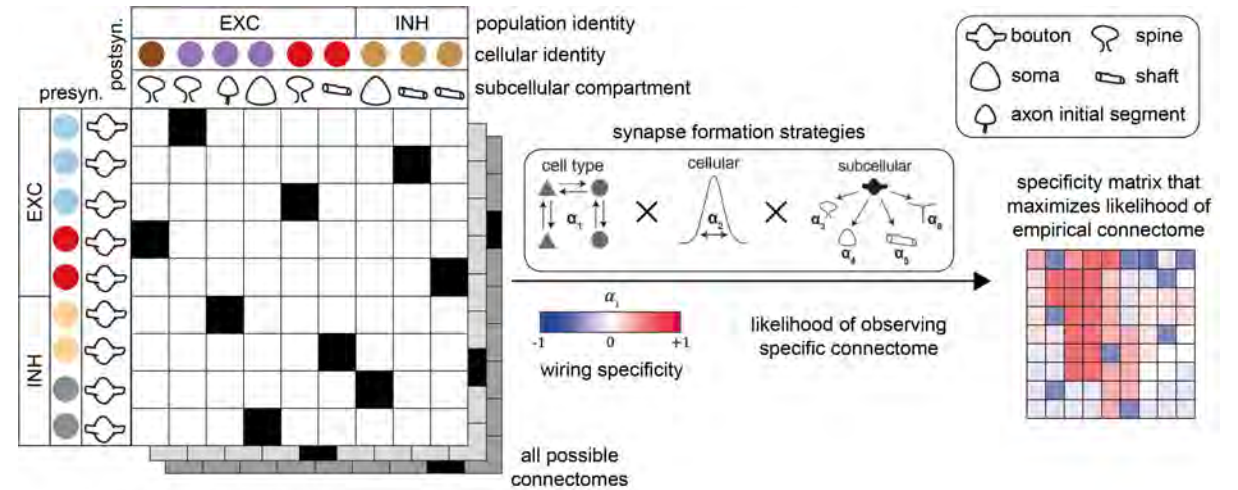
Network analysis

impact of neuron morphology on connectivity



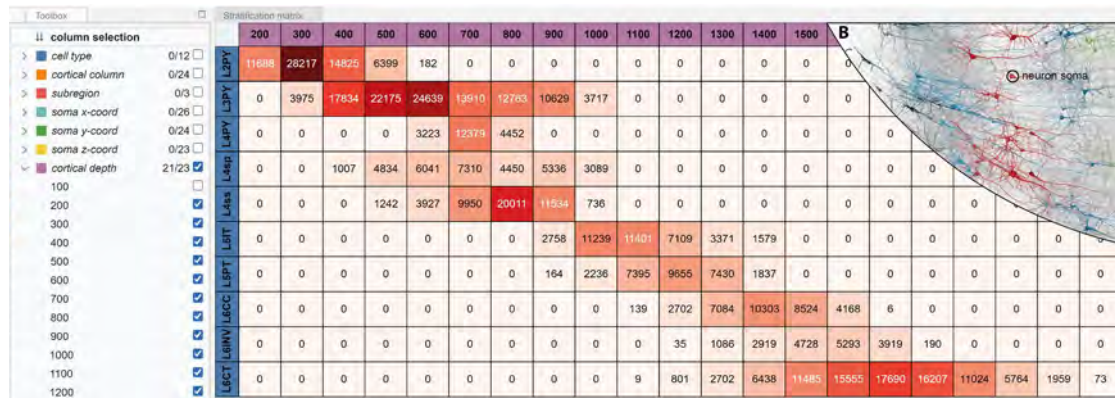
Udvary, Harth et al.; *Cell Reports*, 2022.

feature-rich generative network models



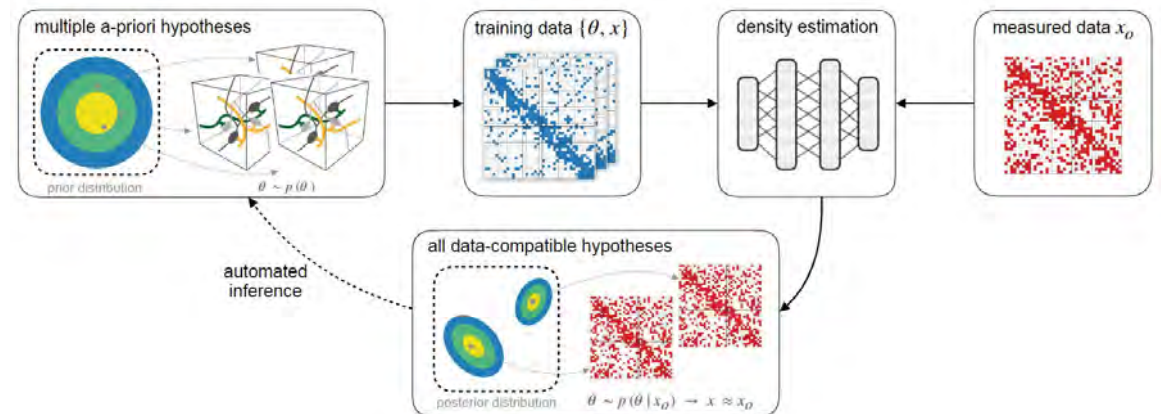
(in preparation)

stratification matrix viewer



Harth, Vohra et al.; *VCBM*, 2022.

simulation-based inference of parameters



Boelts, Harth et al.; *PLOS Comp. Biol.*, 2023. (accepted for publication)