

# Open Exchange Architecture

New standards for **modular** and  
**composable** scientific content.

<https://oxa.dev>

**CSF**

Continuous Science Foundation  
[continuousfoundation.org](http://continuousfoundation.org)



# Science is trapped in a paper-shaped box

- Content is static, narrative driven
- Data/code/notebooks are relegated to supplementary zip files
- Reviews/credit on container, not components



**CSF**

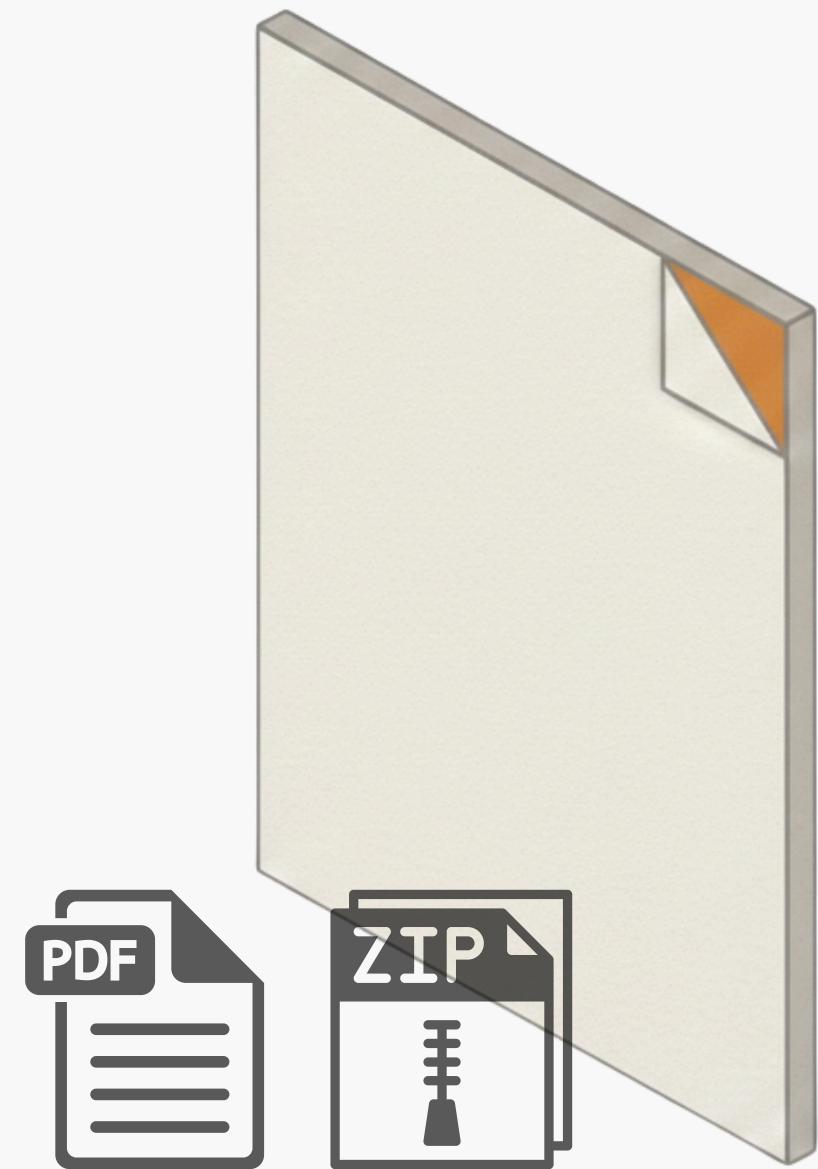
Continuous Science Foundation  
[continuousfoundation.org](http://continuousfoundation.org)

“Science has changed.

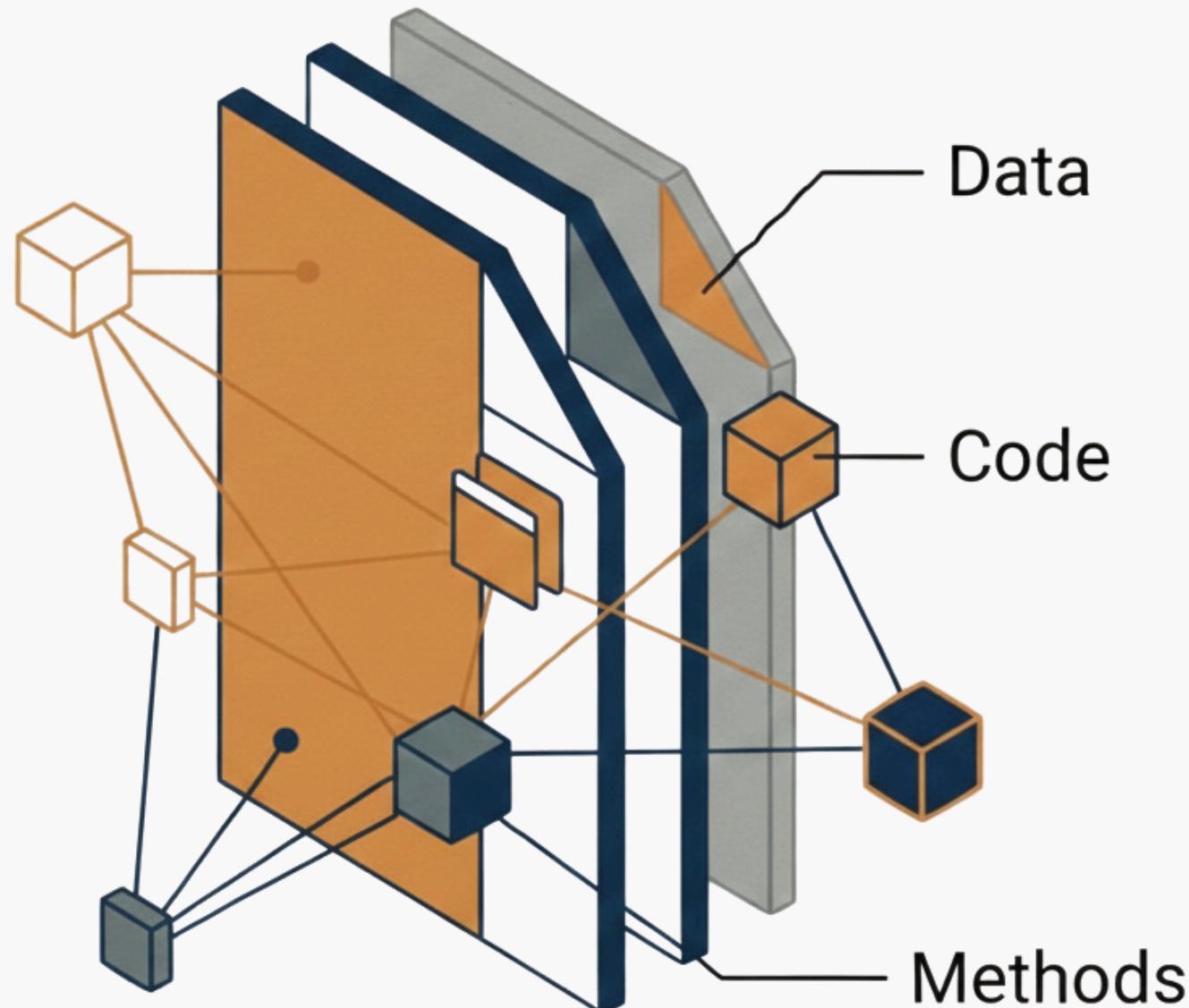
The way we communicate it

has not kept up.”

CSF



Today: Reading & Access Era



Tomorrow: Utility & Reuse Era

# From Tools to Adoption

October, 2025



Carlos Scheidegger and Taylor Campbell

openRxiv CSF

The Navigation Fund

Initial Development & Partners:

 **posit**™

 **eLife**

**Stencila** 

 **creative commons**

 **PRE REVIEW**

 **PLOS**

 **Jisc**

 **Curvenote**

 **NeuroLibre**

 **{MyST}**



Pages

Notebooks

## Supplemental Materials

Count Combinatorial Variants

Count DbSMS Variants

Order DbSMS Library

# Supplemental Materials

make\_dbSMS\_library.py

```
import pandas as pd
from IPython.core.display import display, HTML

def clean():
    #command to get ipynb cells to go to the full width of the browser window
    display(HTML("<style>.container { width:100% !important; }</style>"))

def swap_amino(seq, position, codon_dict, codon_all_dict, codon_aa_nuc, prefix):
    # input: dna sequence, position (0-index) of desired AA swap
    # output: 19 dna sequences with switched codon and barcodes
    if position == 0 or position == len(seq)- 3:
        return False

    left_seq = seq[:position-3]
    right_seq = seq[position+6:]
    left_codon = codon_dict[seq[position-3:position]]
    right_codon = codon_dict[seq[position+3:position+6]]
    seq_wt = seq[position-3:position+6]
    barcode_position = position - 3

    if left_codon == 'Met' or left_codon == 'Trp':
        left_codon = codon_dict[seq[position-6:position-3]] + seq[position-3:position]
        left_seq = seq[:position-6]
        seq_wt = seq[position-6:position+6]
        barcode_position = position - 6
        if left_codon[:3] == 'Met' or left_codon[:3] == 'Trp':
            left_codon = codon_dict[seq[position-9:position-6]] + seq[position-6:position-3]
            left_seq = seq[:position-9]
            seq_wt = seq[position-9:position+6]

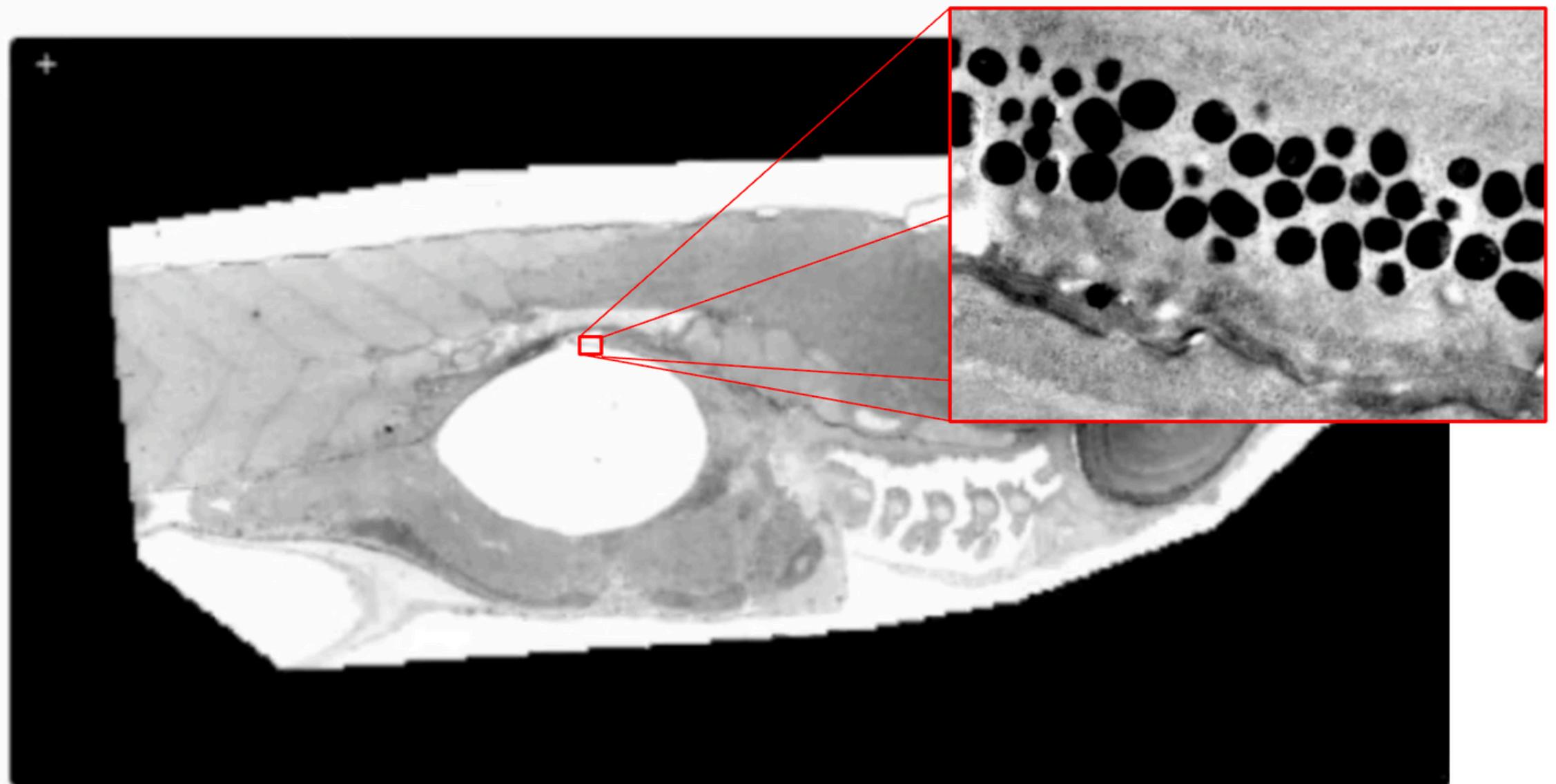
    if right_codon == 'Met' or right_codon == 'Trp':
        right_codon = seq[position+3:position+6] + codon_dict[seq[position+6:position+9]]
        right_seq = right_seq = seq[position+9:]
        seq_wt = seq[position-3:position+9]
        if right_codon[-3:-1] == 'Met' or right_codon[-3:-1] == 'Trp':
```

CSF



[OME-Zarr](#) is a file format for storing large multi-dimensional arrays, such as images. It is based on [Zarr](#), which is storage format for a chunked, compressed, N-dimensional array.

There are several viewers that can display OME-Zarr datasets, including the [Vizarr](#), which is a web-based. Web-based viewers are cool because they can be embedded in web pages, like this one 🎉:



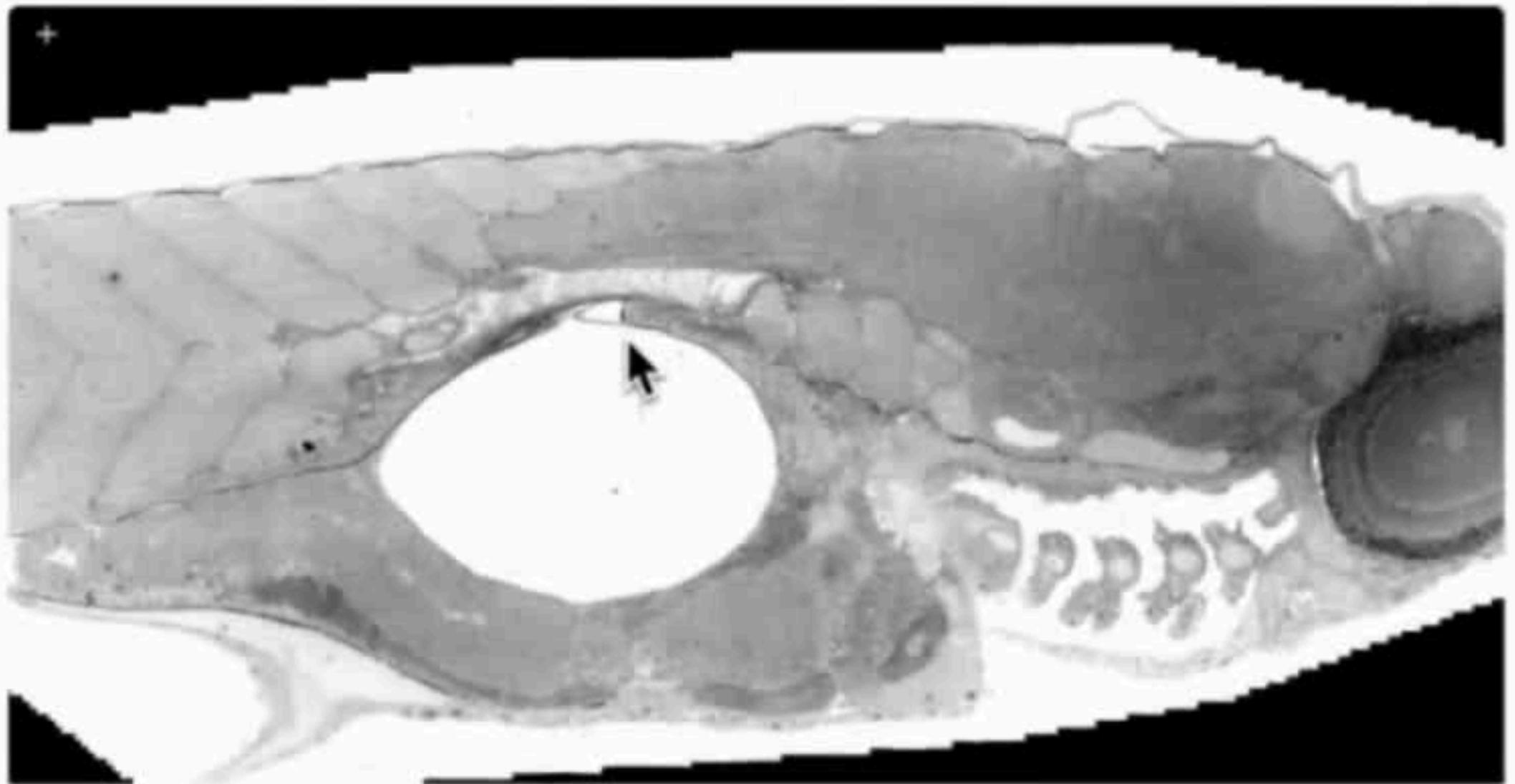
Virtual nanoscopy: generation of ultra-large high resolution electron microscopy maps.

This large electron microscopy dataset is a 1.5TB dataset of a zebrafish embryo, and it is displayed using the Vizarr viewer (Figure 1).

1.5 TB

CSF

There are several viewers that can display OME-Zarr datasets, including the [Vizarr](#), which is a web-based. Web-based viewers are cool because they can be embedded in web pages, like this one 🎉:



Virtual nanoscopy: generation of ultra-large high resolution electron microscopy maps.

This large electron microscopy dataset is a 1.5TB dataset of a zebrafish embryo, and it is displayed using the Vizarr viewer ([Figure 1](#)).

The dataset is stored in an object storage system, and it is accessed using the [HTTP Range Requests](#). To use vizarr, you only need to point to the dataset URL for an OME-Zarr dataset.

1.5 TB

CSF

## 2 Results

### 2.1 Optimization of N49P7-FR

To map the sequence/function landscape of N49P7-FR, we generated double barcode-enhanced saturated mutagenesis scanning (dbSMS) libraries of the variable heavy (VH) and variable light (VL) chains for yeast surface display ([Fig. 1](#)). These libraries incorporated all possible single amino acid substitutions, excluding

king each targeted codon to serve as molecular barcodes for r codes enabled robust differentiation of true mutations from ion of mutation frequencies relative to the parental N49P7-FR r Fab fragments were incubated with sub-saturating ed by fluorescently conjugated antibodies targeting the quantification. Each library was simultaneously screened by panel of six antigenically diverse gp120 monomers, alongside (a) used to establish baseline mutation frequencies FACS enrichment per antigen, antibody-encoding DNA from , generating high-resolution mutational enrichment profiles substitutions across VH and VL domains ([Fig. 2](#) and ed in two ways: (1) as normalized amino acid frequencies at sorted populations relative to the control. These analyses allowing classification of each amino acid substitution as

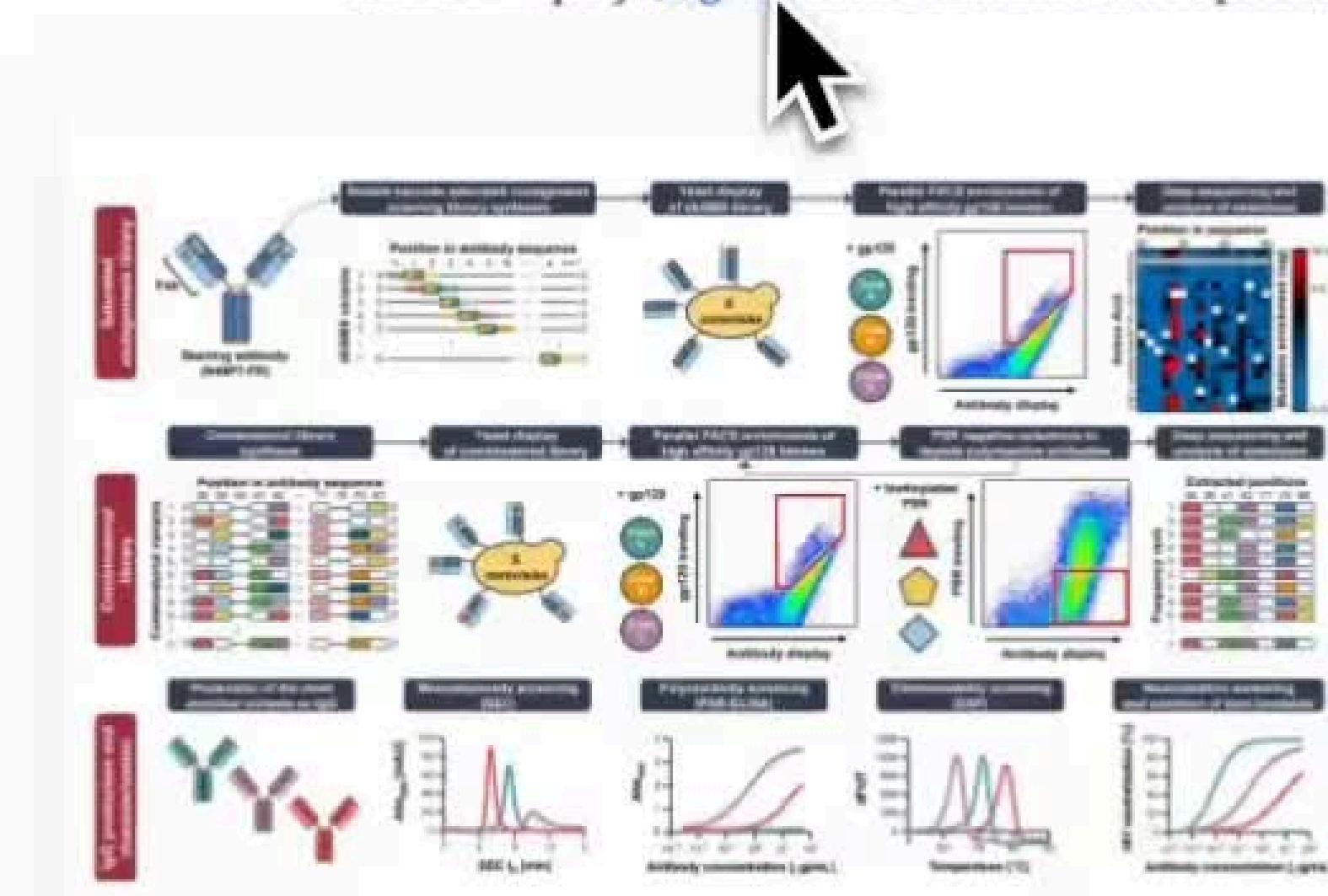


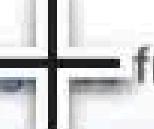
Figure 1: Workflow for *in vitro* bnAb affinity maturation via yeast display

\*Top panel: A dbSMS library was constructed separately for VH and VL regions of

repositories without modification. JATS XML export supports interoperability with discovery engines and indexing services, while also laying the foundation for cross-journal content federation and machine-readable citation networks.

## IN THIS ARTICLE

As more conferences, journals, preprint servers, and MyST-powered documentation sites adopt the structured information and metadata standards, these resources create an interconnected network of scientific knowledge that is browsable, inter-linked, and discoverable. Each article becomes a node in this network, with rich metadata, cross-references, and executable content that can be linked, cited, and built upon from any project. This can allow for reuse from other projects, for example:

1. Referencing  from other publications, (Lujan, 2024);

Source: <https://proceedings.scipy.org/articles/FGCJ9164>

al., 2024);

ck & Elliott, 2024); or

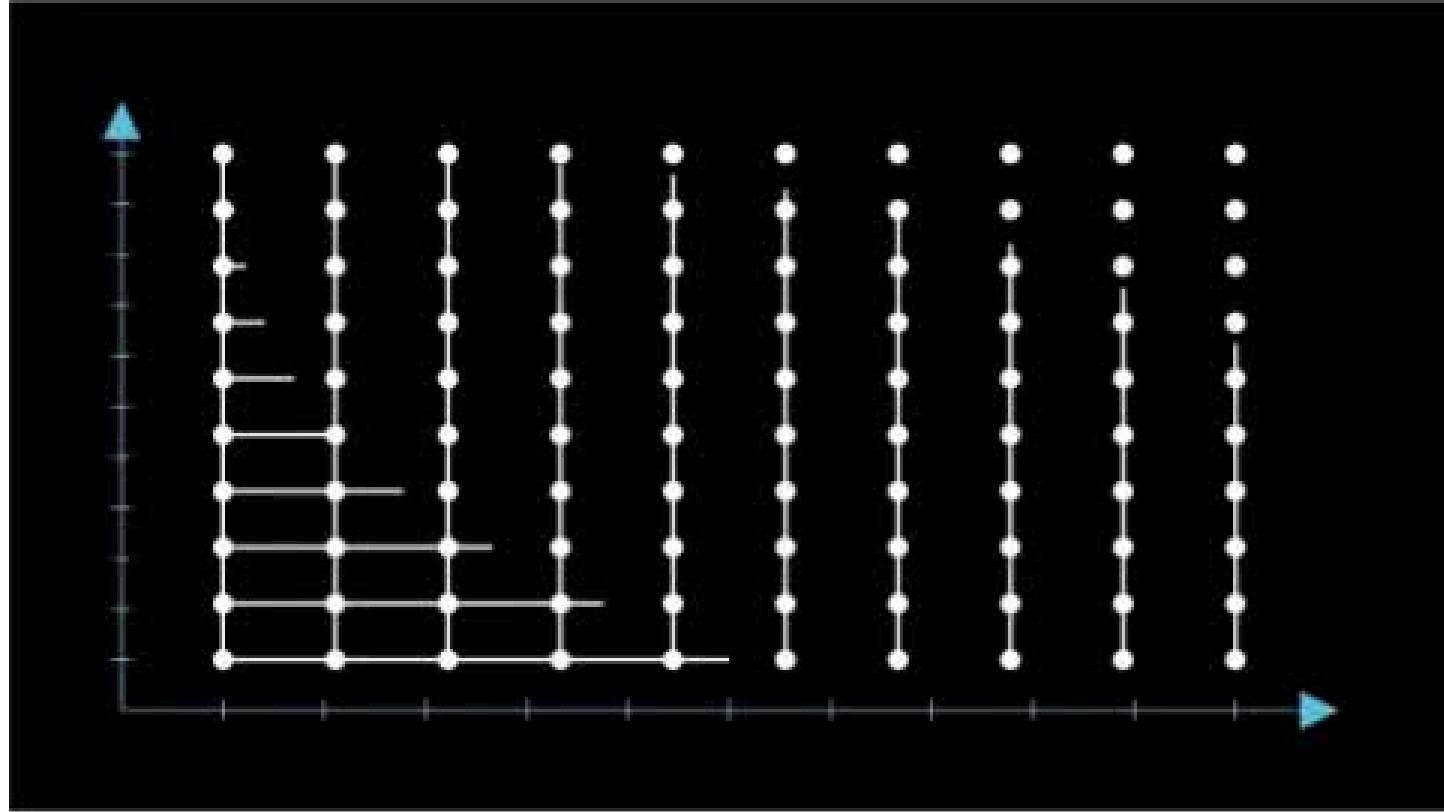


Figure 1: Transformation between non-uniform and uniform rectilinear grids. This process is critical to efficient interpolation on rectilinear grids.

Proceedings (Curvenote Inc., 2024). The fact that high school students can successfully publish scientific content

### Introduction

### A New Model for Scientific Publishing

### Executable Content and Reproducibility

### Enhanced Discoverability and Navigation

### Interactive Exploration and Visualization

### The Authoring Workflow

### Technical Foundation

### Build System and Output Formats

### Metadata and Interoperability

### Demonstrating Accessibility

### Conclusion

CSF



# Principles for OXA

## Modular. Interactive. Composable.



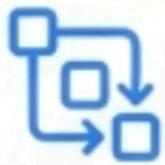
### Open

JSON Schema-based and CC0-licensed.



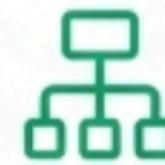
### Extensible

Add new types while preserving validation.



### Composable

Nodes are self-contained and nestable.



### Typed

Everything has a clear type and structured data.



### Interoperable

Compatible with MyST, Stencila, and Quarto.



### Modular

Links across projects and distributed sources.



### Computational

First-class support for Notebooks and code.

CSF



# Granular Linking & Reuse

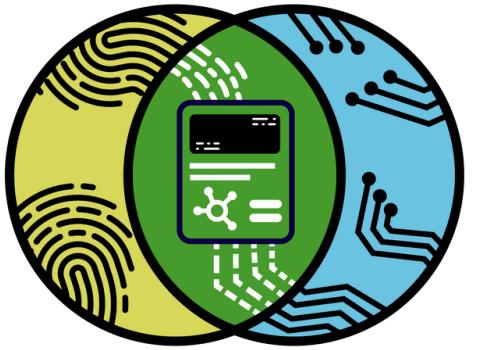
Move from citing 'papers' to citing 'parts'



## Citation

CITATION: [Smith et al., 2024, Fig. 2.3]([link](#))

- Every node (figure, table, equation, output) is individually addressable
- Enables hover-references with attribution
- Enables references in AI tooling
- Integration with existing standards and ability to transform an entire corpus



# Open & Community Driven

## RFC Process, Implementations, Pilots.

- **type** (Required): PascalCase identifier (e.g., Paragraph, CodeBlock).
- **children**: Array of child nodes (The Structure).
- **value**: Scalar payload for leaf nodes (The Content).
- **data**: The extension bucket (The Future).

```
{  
  "type": "CodeBlock",  
  "value": "print('Hello')",  
  "data": {  
    "language": "python",  
    "executable": true  
  }  
}
```



# Into the future



## Moving towards Utility & Reuse.



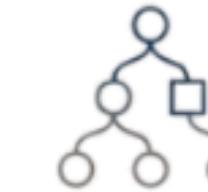
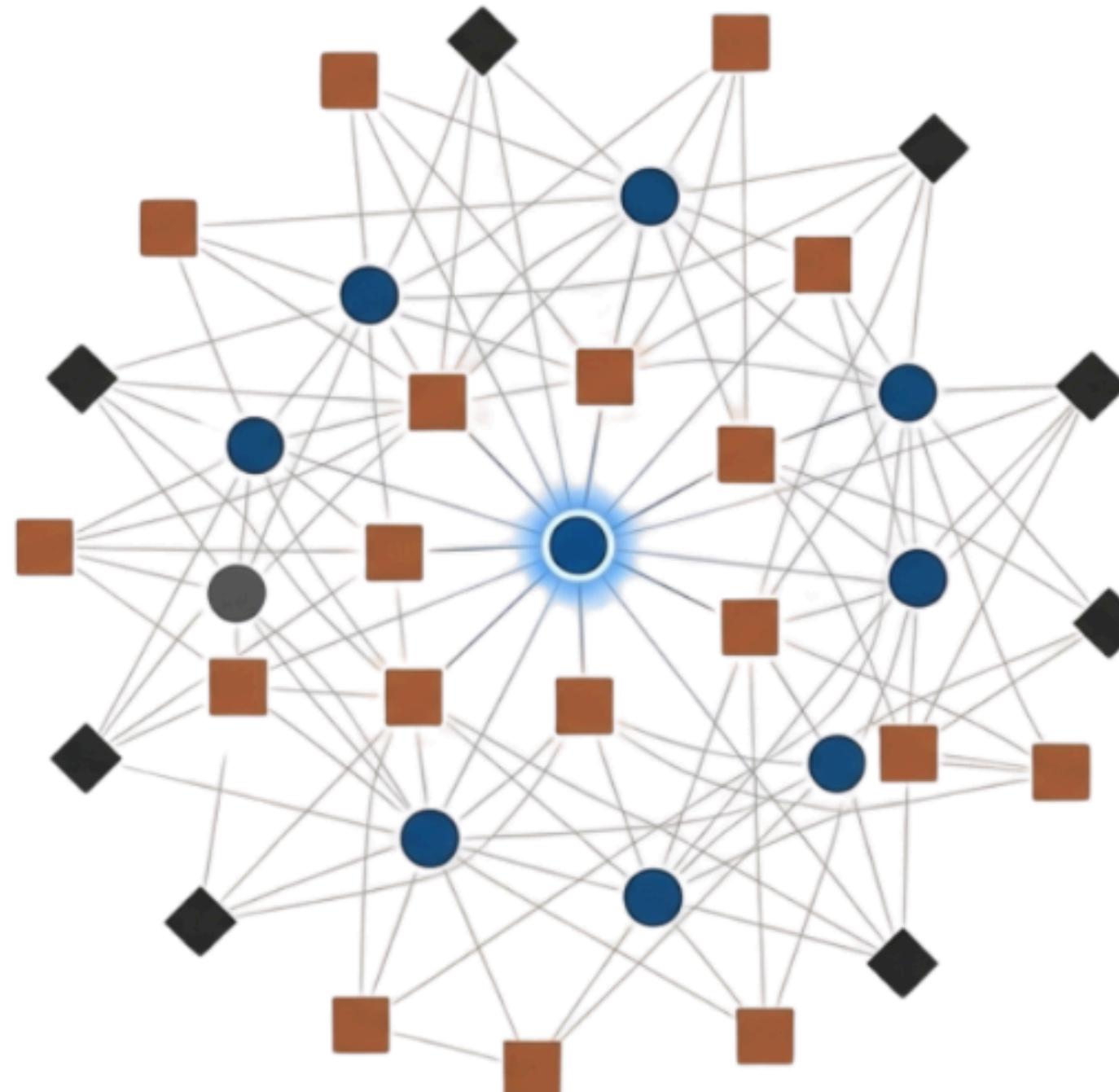
### Claim-level discovery

Search for a specific research question and find the precise figures, datasets, and analyses that support it, wherever they are.



### Method reuse

Find a protocol, see how it has been used or modified, and import it into your workflow with attribution flowing automatically.



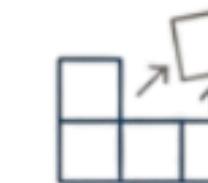
### Living research

Update a dataset or method, and every dependent work can reference the new version transparently.



### Transparent trust

Attach review, licensing, and provenance at the component level, not just to the final publication.



### Composed knowledge

Build curated collections of methods and findings organised around problems, not publication venues.

**CSF**

# Open Exchange Architecture

New standards for **modular** and  
**composable** scientific content.

<https://oxa.dev>

**CSF**

Continuous Science Foundation  
[continuousfoundation.org](http://continuousfoundation.org)

