

# Plan for rest of talk

- Project overview
- Demos & UX discussion
- Creating content with Fluid
  - how it works today
  - how we want it to work!
  - language overview
- Behind the scenes: dynamic dependence graphs
- Call for Collaboration!

# Project overview

Collaboration between UoB and ICCS  
Initial funding from The Alan Turing Institute



## Contributors (past & present)

Joe Bond <sup>1</sup>

Colin Crawford <sup>4</sup>

Cristina David <sup>1</sup>

Harleen Gulati <sup>1</sup>

Hana Iza Kim <sup>2</sup>

Minh Nguyen <sup>1</sup>

Dominic Orchard <sup>5, 2</sup>

Roly Perera <sup>2, 1</sup>

Tomas Petricek <sup>3</sup>

Achintya Rao <sup>6, 2</sup>

Meng Wang <sup>1</sup>

<sup>1</sup>University of Bristol

<sup>2</sup>University of Cambridge

<sup>3</sup>Charles University

<sup>4</sup>University of Edinburgh

<sup>5</sup>University of Kent

<sup>6</sup>University of West of England

<https://f.luid.org>

<https://github.com/explorables/fluid>

# Demo 1

## Ideas introduced

- Notion of transient vs. persistent selection
- “Transparent” brushing-and-linking

## Discussion points

- You could program these “transparency queries” yourself — but not scalable or robust to change
- To be **ubiquitous** needs to be **automatic** (more on this later)
- Extensional information isn’t enough: if an input is used in more than output, its related inputs are ambiguous

## Future directions

- Intensional queries — how was this calculated?

# Demo 2

## Ideas introduced

- Disparate data sources can be related in virtue of contributing to a common output
- Legends and captions are (at best) “comments” that approximate the authors’ intention
- End-users should be able to discover unstated assumptions and other decisions directly through the artifact

## Discussion points

- Visual elements may have multiple independent components (x, y, radius, colour)
- Unused visual channels (e.g. size, colour) to indicate “selection” information may not always be available

## Future directions

- Selection is tricky for some visual attributes (e.g. colour!)
- Probably want a “generic” property pane for visual elements that makes it clear they are just “data”

# Demo 3

## Ideas introduced

- Views of computations can be used to explain and illustrate
- Infrastructure for “explorable explanations” (Bret Victor)

## Discussion points

- Convolution example on distill.pub is a **hand-crafted animation**
- Fluid demo is a transparent view of an **actual implementation** of convolution
- Implementation details start to matter in new ways!

## Future directions

- Intensional queries essential here
- Need to be able to “unpick” the computation and examine intermediate results (e.g. interim 3x3 arrays)
- Converge the medium in which we **do** science with medium in which we **communicate** science

# Creating a Fluid visualisation

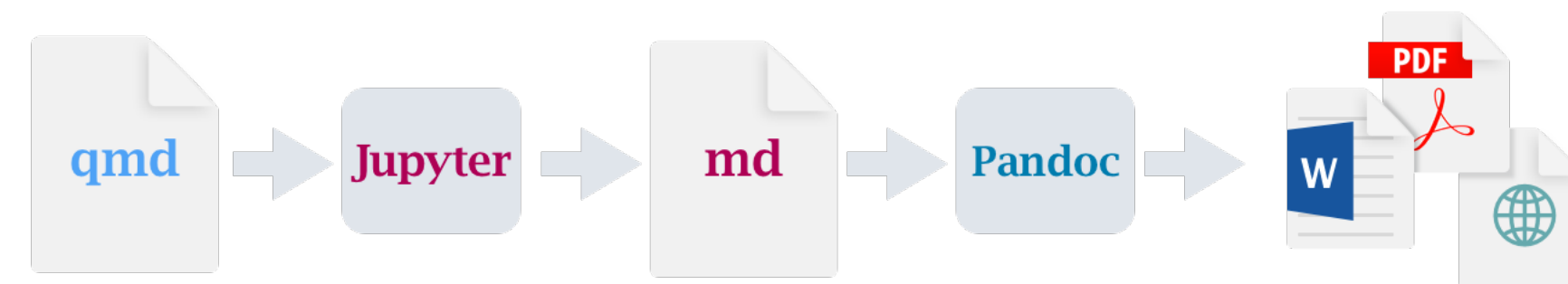
So far project has focused mainly on **end-user experience**  
Developer tools are sorely lacking..

## Current workflow (😬)

- Create `index.html` with `div` to contain Fluid visualisation
- Write Fluid source code (`.fld`) for visualisation
- Write small PureScript program to load visualisation

## What's missing

- Command-line publishing tool (Node.js)
- Fluid kernel for Jupyter — will enable authoring via **Markdown** and **Quarto**



Want to leverage as much existing open source infrastructure as possible  
Three interns will be helping with some of this over the summer!

# Language overview

## Current design

- Purely functional (no side-effects)
- Untyped
- Records, lists, dictionaries, 2D arrays
- Graphics library based on **d3.js**

Implemented in **PureScript**  
(Haskell clone for the web)

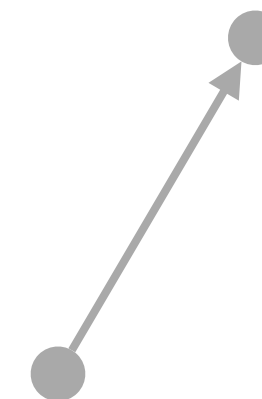
## What's missing

- Modules and imports
- I/O — load from file, db or URL
- Syntax for datatype definitions
- Type system (with units of measure?)

# How does it work?

## Key infrastructure

- Fluid interpreter builds a **directed graph** of data dependencies as program runs
- Treat relationship between inputs and outputs as “metadata” that can be queried
- Related inputs and related outputs are relations of **cognacy** (common ancestry) in the graph



## Further reading

- POPL 2022
- Preprint
- Psallidas and Wu



# Language overview

## Current design

- Purely functional (no side-effects)
- Untyped
- Records, lists, dictionaries, 2D arrays
- Graphics library based on **d3.js**

Implemented in **PureScript**  
(Haskell clone for the web)

## What's missing

- Modules and imports
- I/O — load from file, db or URL
- User-defined datatypes
- Type system (with units of measure?)

# Future directions

How to enable a smooth transition from content consumption to content creation?



## Readers:

- Don't care how it works
- Want responsive, self-explanatory, intuitive UI
- Should be able to transition from passive reading to active engagement
- UI affordances (opportunities for interaction) should present themselves

We wear different hats at different times..

## Authors:

- Proficient in technology
- Invested in specific workflows and skills
- Benefits of new technology need to be obvious
- Barriers to entry need to be low

What are the prospects of doing something like Fluid for R or Python?

# Weaving new stories from existing ones

Example of previous continuum that we would like to enable:



- Reviewer explores claims, data sources and computational methods **in situ**
- Frames queries by interacting with outputs and perhaps making other choices
- Queries/views are **persistent** and versioned and can be shared with authors or other readers
- Interactions are reproducible and can be replayed (cf. Histogram)
- Interesting observations are new knowledge and contribute to the overall science
- Authors' original narrative was just one of many possible narratives

Not a new idea, but definitely an idea whose time has come

# How can we help you tell a story with your data?

We are looking for a VESRI project that we can use to showcase **our infrastructure** and **your research** in the form of an online article with figures transparently linked to data

Timeframe: Aug-Dec 2024

## Call for Case Studies

Do you have:

- an existing open access publication with figures we could reimplement in Fluid to add value?
- a question about prior work that you didn't get a chance to ask and would like to revisit?
- a new idea or question we could help you present in an interesting way?

## Your project may be a good fit if:

- it is not too data-intensive or algorithmically intensive
- has obvious visual elements
- will benefit from being presented in a transparent, explorable way (potentially to a lay audience)