

# T-NA083 Development of fluid referent models

B.Dudson<sup>1</sup>, D.Moxey<sup>2</sup>, S.Wright<sup>1</sup>, P.Hill<sup>1</sup>,  
D.Dickinson<sup>1</sup>, E.Higgins<sup>1</sup>

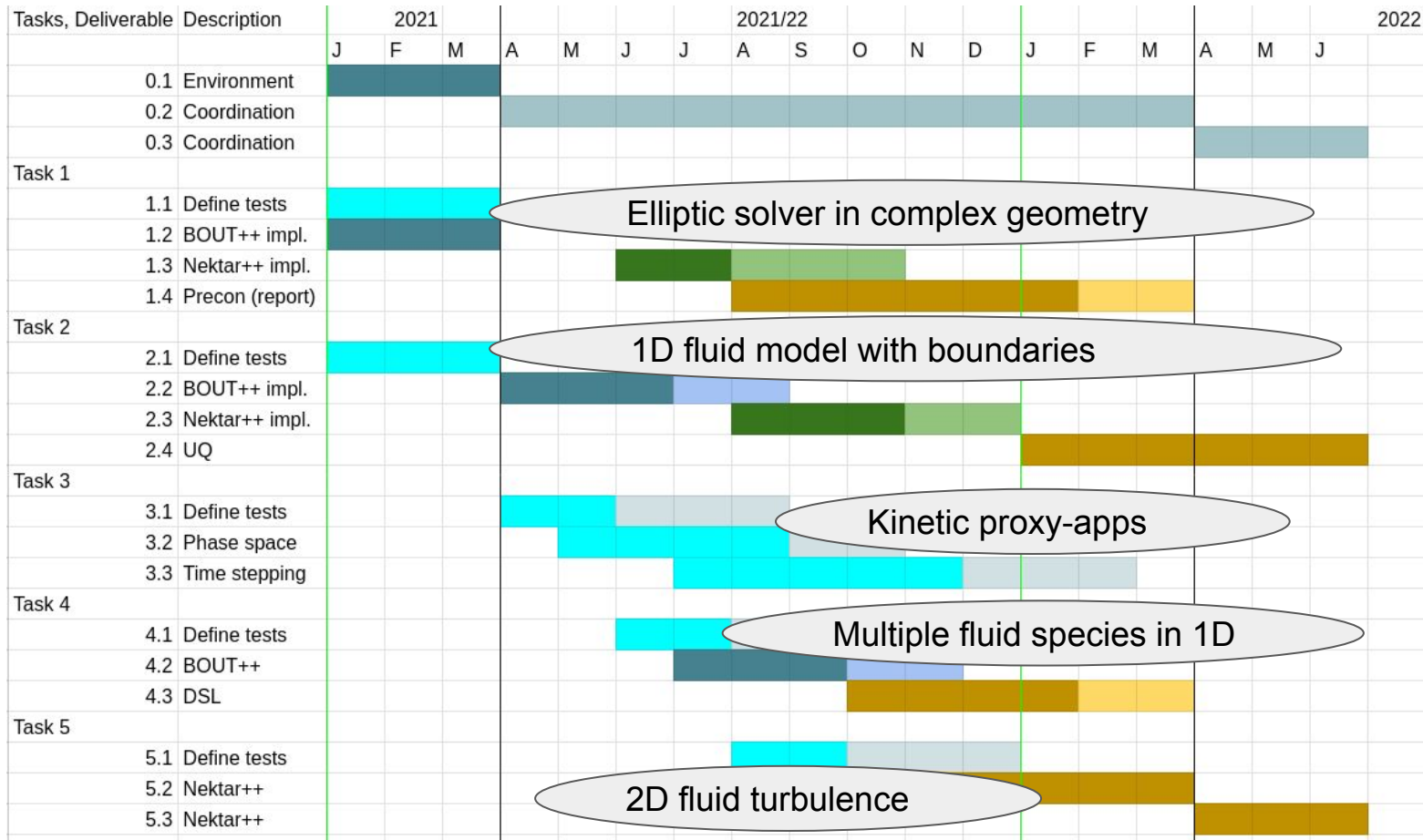
<sup>1</sup> University of York

<sup>2</sup> University of Exeter

# Aims

1. Bring together and integrate the work of other work packages
2. Develop increasingly complicated test cases, with reference and spectral/hp implementations, in steps towards the full NEPTUNE fusion simulation
3. Write exploratory proxy-apps to test algorithms and strategies
4. Coordinate code development practices and training

# Project timeline



# Task 5, system 2.2

Moderately complex plasma model in 2D. Simplified compared to full system!

$$\partial_t n_e + \nabla \cdot (n_e \mathbf{u}_e) = S_{n_e} - \frac{n_e}{\tau_{n_e}}$$

$$\begin{aligned} \partial_t \nabla \cdot \mathbf{E}^+ + \nabla \cdot (\nabla \cdot (\mathbf{u}_i \otimes \mathbf{E}^+)) = & \nabla \cdot \left( n_i (\mathbf{u}_{\nabla B i} + \mathbf{u}_{cx}) - \frac{1}{Z_i} n_e \mathbf{u}_{\nabla B e} \right) \\ & + \frac{1}{Z_i} \frac{n_e}{\tau_{n_e}} - \frac{n_i}{\tau_{n_i}} + \nabla \cdot (\nu \nabla_{\perp} (\nabla \cdot \mathbf{E}^+)) \end{aligned}$$

$$\partial_t \mathcal{E}_e + \nabla \cdot (\mathcal{E}_e \mathbf{u}_e + p_e \mathbf{u}_e) = S_{\mathcal{E}_e} - \frac{\mathcal{E}_e}{\tau_{Ee}} + Q_{ie} + \nabla \cdot (\chi_{\perp e} n_e \nabla_{\perp} T_e)$$

$$\partial_t \mathcal{E}_i + \nabla \cdot (\mathcal{E}_i \mathbf{u}_i + p_i \mathbf{u}_i) = S_{\mathcal{E}_i} - \frac{\mathcal{E}_i}{\tau_{Ei}} - Q_{ie} + \nabla \cdot (\chi_{\perp i} n_i \nabla_{\perp} T_i)$$

$$\partial_t n_n = S_{n_n} + \nabla \cdot (D_n \nabla_{\perp} p_n)$$

# Task 5, system 2.2

Moderately complex plasma model in 2D. Simplified compared to full system!

$$\begin{aligned}
 \partial_t n_e + \nabla \cdot (n_e \mathbf{u}_e) &= S_{n_e} - \frac{n_e}{\tau_{n_e}} \\
 \partial_t \nabla \cdot \mathbf{E}^+ + \nabla \cdot (\nabla \cdot (\mathbf{u}_i \otimes \mathbf{E}^+)) &= \nabla \cdot \left( n_i (\mathbf{u}_{\nabla B i} + \mathbf{u}_{cx}) - \frac{1}{Z_i} n_e \mathbf{u}_{\nabla B e} \right) \\
 &\quad + \frac{1}{Z_i} \frac{n_e}{\tau_{n_e}} - \frac{n_i}{\tau_{n_i}} + \nabla \cdot (\nu \nabla_{\perp} (\nabla \cdot \mathbf{E}^+)) \\
 \partial_t \mathcal{E}_e + \nabla \cdot (\mathcal{E}_e \mathbf{u}_e + p_e \mathbf{u}_e) &= S_{\mathcal{E}_e} - \frac{\mathcal{E}_e}{\tau_{Ee}} + Q_{ie} + \nabla \cdot (\chi_{\perp e} n_e \nabla_{\perp} T_e) \\
 \partial_t \mathcal{E}_i + \nabla \cdot (\mathcal{E}_i \mathbf{u}_i + p_i \mathbf{u}_i) &= S_{\mathcal{E}_i} - \frac{\mathcal{E}_i}{\tau_{Ei}} - Q_{ie} + \nabla \cdot (\chi_{\perp i} n_i \nabla_{\perp} T_i) \\
 \partial_t n_n &= S_{n_n} + \nabla \cdot (D_n \nabla_{\perp} p_n)
 \end{aligned}$$

All coefficients nonlinear,  
time-dependent

# Task 5, system 2.2

Moderately complex plasma model in 2D. Simplified compared to full system!

$$\partial_t n_e + \nabla \cdot (n_e \mathbf{u}_e) = S_{n_e} - \frac{n_e}{\tau_{n_e}}$$

$$\partial_t \nabla \cdot \mathbf{E}^+ + \nabla \cdot (\nabla \cdot (\mathbf{u}_i \otimes \mathbf{E}^+)) = \nabla \cdot \left( n_i (\mathbf{u}_{\nabla B i} + \mathbf{u}_{cx}) - \frac{1}{Z_i} n_e \mathbf{u}_{\nabla B e} \right)$$

Many coefficients are tensors,  
highly anisotropic

FE representation

$$+ \frac{1}{Z_i} \frac{n_e}{\tau_{n_i}} - \frac{n_i}{\tau_{n_i}} + \nabla \cdot (\nu \nabla_{\perp} (\nabla \cdot \mathbf{E}^+))$$

$$\partial_t \mathcal{E}_e + \nabla \cdot (\mathcal{E}_e \mathbf{u}_e + p_e \mathbf{u}_e) = S_{\mathcal{E}_e} - \frac{\mathcal{E}_e}{\tau_{Ee}} + Q_{ie} + \nabla \cdot (\chi_{\perp e} n_e \nabla_{\perp} T_e)$$

$$\partial_t \mathcal{E}_i + \nabla \cdot (\mathcal{E}_i \mathbf{u}_i + p_i \mathbf{u}_i) = S_{\mathcal{E}_i} - \frac{\mathcal{E}_i}{\tau_{Ei}} - Q_{ie} + \nabla \cdot (\chi_{\perp i} n_i \nabla_{\perp} T_i)$$

$$\partial_t n_n = S_{n_n} + \nabla \cdot (D_n \nabla_{\perp} p_n)$$

# Task 5, system 2.2

Moderately complex plasma model in 2D. Simplified compared to full system!

$$\partial_t n_e + \nabla \cdot (n_e \mathbf{u}_e) = S_{n_e} - \frac{n_e}{\tau_{n_e}}$$

$$\partial_t \nabla \cdot \mathbf{E}^+ + \nabla \cdot (\nabla \cdot (\mathbf{u}_i \otimes \mathbf{E}^+)) = \nabla \cdot \left( n_i (\mathbf{u}_{\nabla B i} + \mathbf{u}_{cx}) - \frac{1}{Z_i} n_e \mathbf{u}_{\nabla B e} \right)$$

$$\mathbf{E}^+ = \frac{m_i}{Z_i e B^2} \left( n_i \nabla_{\perp} \Phi + \frac{1}{Z_i e} \nabla_{\perp} p_i \right)$$

Advection velocities  $\mathbf{u}$  depend on  $\nabla \Phi$

Vorticity equation, elliptic inversion (global) with time-varying coefficients an important part of the problem

Preconditioning

$$+ \frac{\nabla \cdot (\nu \nabla_{\perp} (\nabla \cdot \mathbf{E}^+))}{\text{4th derivative of } \Phi}$$

$$+ \nabla \cdot (\chi_{\perp e} n_e \nabla_{\perp} T_e)$$

$$+ \nabla \cdot (\chi_{\perp i} n_i \nabla_{\perp} T_i)$$

# Task 5, system 2.2

Moderately complex plasma model in 2D. Simplified compared to full system!

$$\partial_t n_e + \nabla \cdot (n_e \mathbf{u}_e) = \boxed{S_{n_e}} - \frac{n_e}{\tau_{n_e}}$$

$$\partial_t \nabla \cdot \mathbf{E}^+ + \nabla \cdot (\nabla \cdot (\mathbf{u}_i \otimes \mathbf{E}^+)) = \nabla \cdot \left( n_i (\mathbf{u}_{\nabla B i} + \mathbf{u}_{cx}) - \frac{1}{Z_i} n_e \mathbf{u}_{\nabla B e} \right)$$

Sources from particle model,  
reaction rates with  
uncertainties

Uncertainty Quantification

$$+ \overline{\nu} - \overline{\nu} + \nabla \cdot (\nu \nabla_{\perp} (\nabla \cdot \mathbf{E}^+))$$

Coupling to particles

$$\partial_t \mathcal{E}_e + \nabla \cdot (\mathcal{E}_e \mathbf{u}_e + p_e \mathbf{u}_e) = \boxed{S_{\mathcal{E}_e}} - \frac{\mathcal{E}_e}{\tau_{Ee}} + Q_{ie} + \nabla \cdot (\chi_{\perp e} n_e \nabla_{\perp} T_e)$$

$$\partial_t \mathcal{E}_i + \nabla \cdot (\mathcal{E}_i \mathbf{u}_i + p_i \mathbf{u}_i) = \boxed{S_{\mathcal{E}_i}} - \frac{\mathcal{E}_i}{\tau_{Ei}} - Q_{ie} + \nabla \cdot (\chi_{\perp i} n_i \nabla_{\perp} T_i)$$

$$\partial_t n_n = \boxed{S_{n_n}} + \nabla \cdot (D_n \nabla_{\perp} p_n)$$



# Task 5, system 2.2

Moderately complex plasma model in 2D. Simplified compared to full system!

$$\partial_t n_e + \nabla \cdot (n_e \mathbf{u}_e) = S_{n_e} - \frac{n_e}{\tau_{n_e}}$$

$$\partial_t \nabla \cdot \mathbf{E}^+ + \nabla \cdot (\nabla \cdot (\mathbf{u}_i \otimes \mathbf{E}^+)) = \nabla \cdot \left( n_i (\mathbf{u}_{\nabla B i} + \mathbf{u}_{cx}) - \frac{1}{Z_i} n_e \mathbf{u}_{\nabla B e} \right)$$

This is only one ion species  
NEPTUNE must have many

$$- \frac{1}{Z_i} \frac{n_e}{\tau_{n_e}} - \frac{n_i}{\tau_{n_i}} + \nabla \cdot (\nu \nabla_{\perp} (\nabla \cdot \mathbf{E}^+))$$

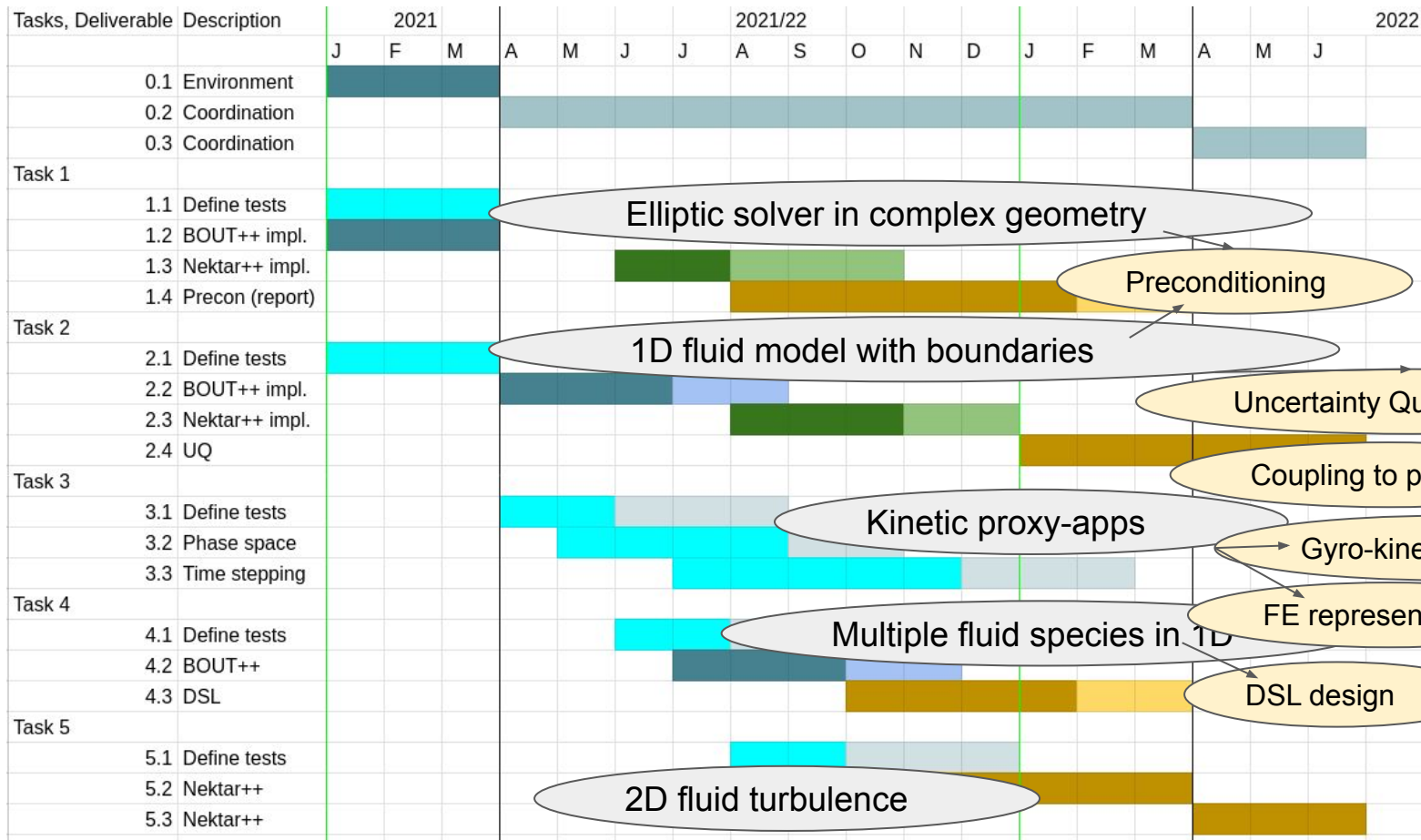
DSL design

$$\partial_t \mathcal{E}_e + \nabla \cdot (\mathcal{E}_e \mathbf{u}_e + p_e \mathbf{u}_e) = S_{\mathcal{E}_e} - \frac{\mathcal{E}_e}{\tau_{Ee}} + Q_{ie} + \nabla \cdot (\chi_{\perp e} n_e \nabla_{\perp} T_e)$$

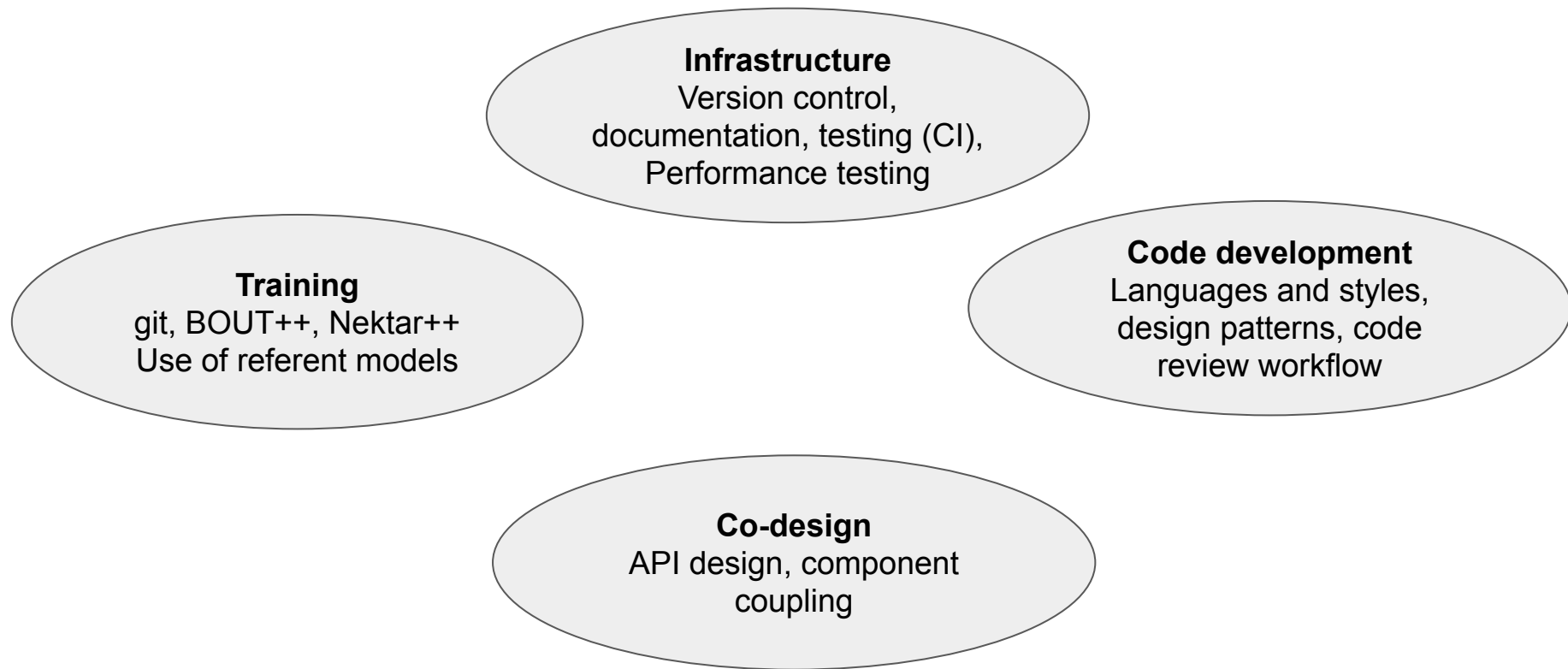
$$\partial_t \mathcal{E}_i + \nabla \cdot (\mathcal{E}_i \mathbf{u}_i + p_i \mathbf{u}_i) = S_{\mathcal{E}_i} - \frac{\mathcal{E}_i}{\tau_{Ei}} - Q_{ie} + \nabla \cdot (\chi_{\perp i} n_i \nabla_{\perp} T_i)$$

$$\partial_t n_n = S_{n_n} + \nabla \cdot (D_n \nabla_{\perp} p_n)$$

# Integration with other work packages



# Community coordination



# Version control: GitHub

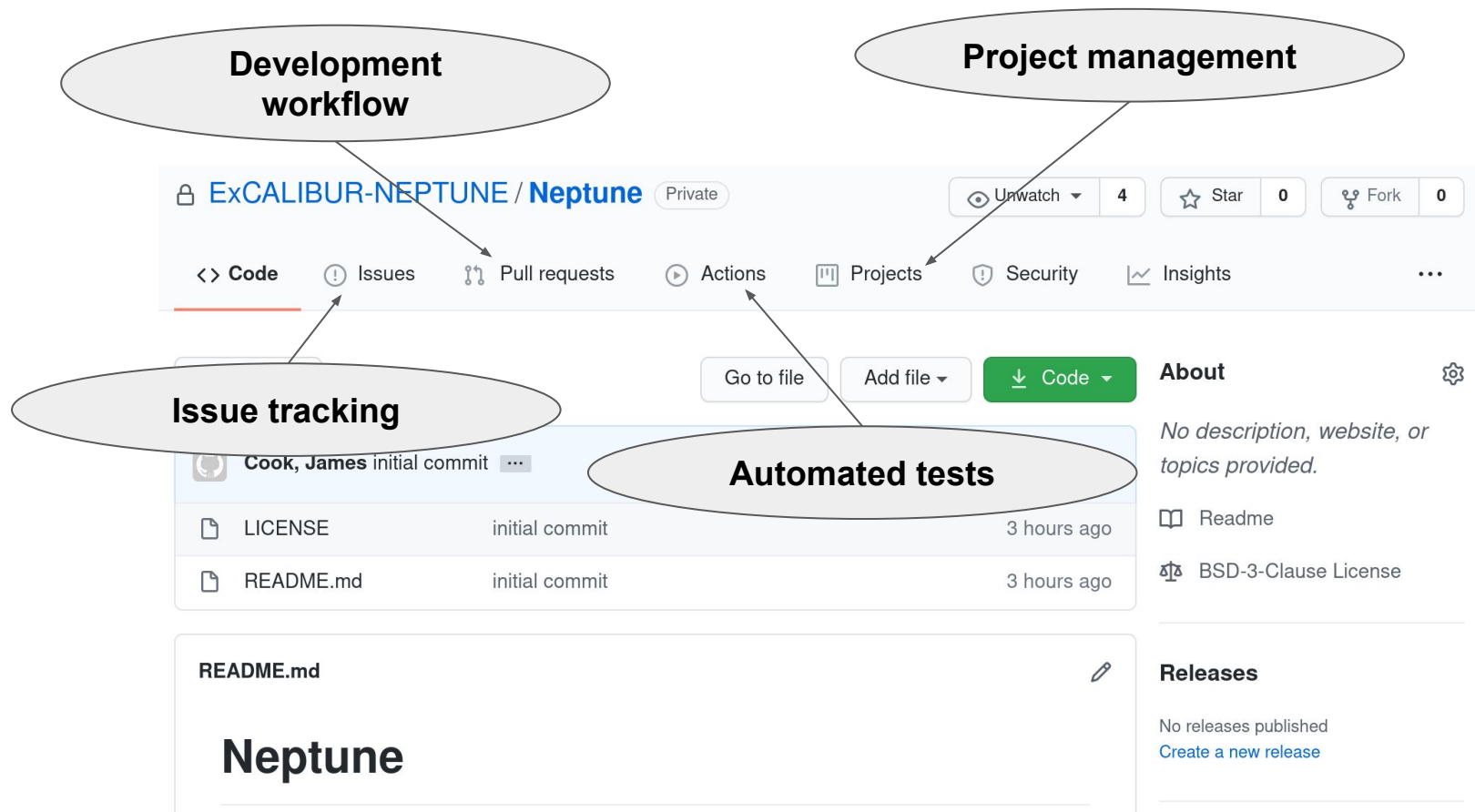
The image shows a screenshot of a GitHub repository page for 'ExCALIBUR-NEPTUNE / Neptune'. The repository is currently private. Annotations highlight key details:

- GitHub organisation**: Points to the repository name 'ExCALIBUR-NEPTUNE / Neptune'.
- Main repository**: Points to the repository name, with the note 'Currently private; will be public soon'.
- BSD license**: Points to the 'LICENSE' file in the file list.
- Copyright (c) 2021, ExCALIBUR-NEPTUNE Authors**: Points to the copyright notice in the README file.

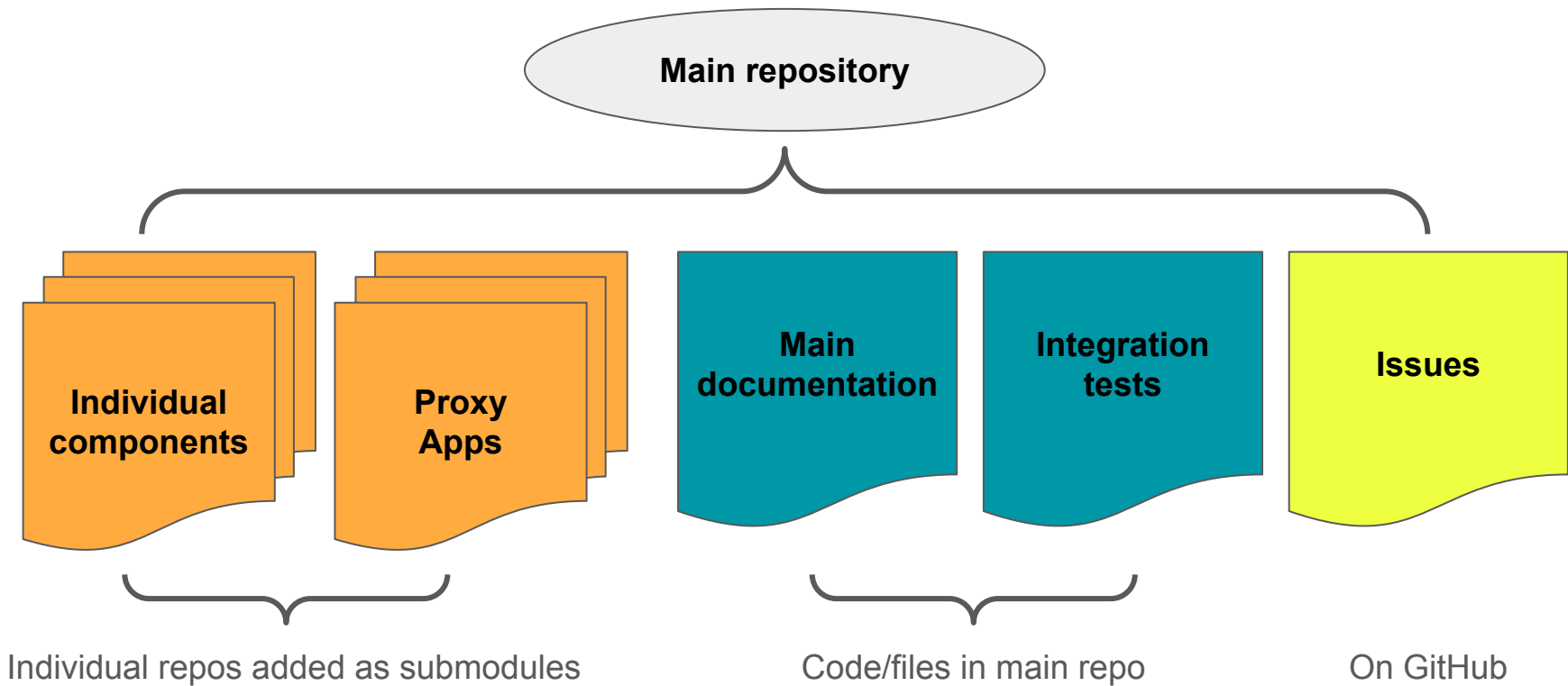
The repository page includes the following elements:

- Repository name: **ExCALIBUR-NEPTUNE / Neptune** (Private)
- Unwatch button, 4 forks, 0 stars, 0 forks.
- Navigation tabs: **<> Code**, Issues, Pull requests, Actions, Projects, Security, Insights.
- Branch selector: **main**.
- Buttons: Go to file, Add file, Code.
- Commit history: **Cook, James** initial commit (3 hours ago, 1 commit).
- File list: **LICENSE** (initial commit, 3 hours ago), **README.md** (initial commit, 3 hours ago).
- About section: No description, website, or topics provided. Includes Readme and BSD-3-Clause License.
- README content: **Neptune**, Copyright (c) 2021, ExCALIBUR-NEPTUNE Authors.

# Version control: GitHub

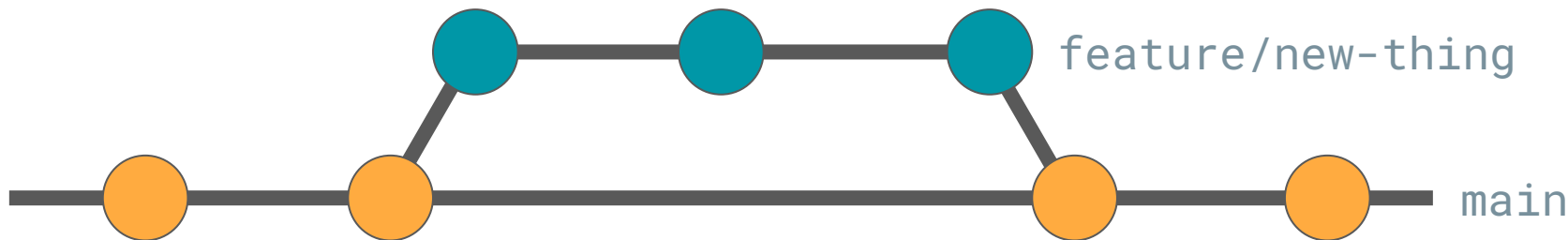


# Version control: Organisation



# Version control: Workflow

- Development to be done via feature branches and pull requests (PRs)
- Allows for code review and discussion — vital for developing cohesive project
  - code review is a responsibility of all developers
- Larger components may have additional `develop` branch
- Automated tests run on PRs — must pass before merging



# Testing

- Individual components contain their own tests
- Write tests as early as possible (even before the code itself!)
- Integration/coupling tests in main repo
  - Will ensure the various components work together
- GitHub Actions to automatically run tests/static analysis on commits/PRs
- Performance testing: likely to be semi-automated, at least for time being
  - Will need to run on variety of hardware/environments
  - Data schema likely to evolve



# Code development

- Development to all be done in public repos
- Common licence: BSD-3
  - “Do what you want, but keep attribution and licence”
- Code style:
  - Use automated tools wherever possible (`clang-format`, `black`)
  - Exact style less important than consistency (but see above)
- Documentation:
  - Use of “doc-strings”, in-line documentation
  - Doxygen + breathe + sphinx + ReadTheDocs == automated building of documentation
  - Markdown/reStructuredText: both allow embedded LaTeX for equations
- Programming languages:
  - (Try to) use latest standards for new code: C++20, Python 3.9, Fortran 2018
    - Limited by availability on machines, but e.g. ARCHER2 has gcc 10.1