PSD3 Publication Plan

Overview

Prepare PSD3 for public release to the PureScript community. This involves cleaning up legacy patterns, splitting into publishable packages, writing documentation, and setting up hosting.

Phase 1: Code & Documentation Cleanup

1.1 Remove/Update Legacy Pattern References

Files requiring updates:

File	Issue	Action
`docs/wizard-guide.md`	Heavy Datum_ pattern documentation	Rewrite or remove - wizard may be obsolete
`docs/LIBRARY_REFACTORING_PLAN.md`	Internal planning doc with Datum_	Move to historical archive or delete
`docs/MODULE_MIGRATION_PLAN.md`	Internal planning doc	Move to historical archive or delete
`demo-website/src/Component/Tutorial/GettingStarted.purs`	"The Datum_ Pattern" section (lines 322-351)	Rewrite to show modern Tree API approach
`demo-website/src/Component/Wizard/Templates.purs`	Generates Datum_ pattern code	Update wizard or mark as legacy/remove

Internal FFI types to keep (these are fine):

- `psd3-selection/src/PSD3/Internal/Types.purs` - `Datum_` type definition
- `psd3-selection/src/PSD3/Internal/FFI.purs` - FFI functions using `Datum_`

These are internal implementation details, properly namespaced with underscore convention.

1.2 Review and Address TODOs

60 TODO/FIXME comments across active source. Need to:

- Triage: fix, remove, or document as known limitations
- Remove any that reference obsolete plans

1.3 Clean Up /docs Directory

Files to evaluate:

- `docs/code-atlas-README.md` - Keep? Part of ce-website?
- `docs/code-atlas-data-model.md` - Keep? Part of ce-website?
- `docs/design/PSD3v2_TRANSITIONS_DESIGN.md` - Internal design doc, archive
- `docs/wizard-guide.md` - Obsolete? Rewrite or remove
- `docs/three-little-circles-v2.html` - Standalone HTML, needed?

1.4 Rewrite README.md

Current: 406 lines (wall of text)

New structure:

```
# PSD3 — Type-Safe D3 Visualizations in PureScript

[One paragraph elevator pitch]

## Quick Example
[10-15 lines showing Tree API]

## Packages
- psd3-selection — Core selection and attribute system
- psd3-simulation — Force simulation engine
- psd3-layout — Hierarchical layouts (tree, treemap, Sankey, etc.)

## Documentation
[Link to website]

## Installation
[spago install commands once published]

## Credits & History
[Link to acknowledgements page]

## License
MIT
```

Phase 2: Add Tests

2.1 Test Strategy

Package	Testable Areas

Package	Testable Areas
psd3-selection	Attribute resolution, scale calculations
psd3-simulation	Force calculations (pure parts)
psd3-layout	Tree layout algorithms, Sankey computations

Focus on pure functions first. FFI integration tests are harder but could verify DOM structure.

2.2 Test Infrastructure

- Add `test/` directory to each package
- Add test dependencies to spago.yaml
- Consider: spec, quickcheck, or assert

Phase 3: Split into Repositories

3.1 New Repositories

Repo	Contents
`afcondon/purescript-psd3-selection`	psd3-selection/
`afcondon/purescript-psd3-simulation`	psd3-simulation/
`afcondon/purescript-psd3-layout`	psd3-layout/
`afcondon/psd3-website`	demo-website/ + docs/
`afcondon/psd3-code-explorer`	ce-website/ + ce-server/ + ce-database/

3.2 Package Configuration

Each package needs:

- `spago.yaml` with publish section:

```
package:
  name: psd3-selection
  publish:
    version: 0.1.0
    license: MIT
    location:
      githubOwner: afcondon
      githubRepo: purescript-psd3-selection
```

- `LICENSE` (MIT)
- `CHANGELOG.md`
- Concise `README.md` pointing to main docs

3.3 Dependency Verification

Test each package builds independently with only declared dependencies.

Phase 4: Documentation & Website

4.1 Website Content Review

Pages to review/rewrite in your voice:

- Home page
- Getting Started (needs major update for published packages)
- Understanding section
- Tour pages
- How-to guides

4.2 AI Collaboration Acknowledgement Page

Create new page documenting:

- Timeline of development (D3 → Ian Ross Attr → your iterations → Claude collaboration)
- What Claude contributed (code volume, refactoring, documentation)
- What you contributed (core architecture, design decisions, domain expertise)
- Philosophy on human-AI collaboration

4.3 Acknowledgements Page Update

Expand existing page to include:

- Ian Ross (Attr typeclass inspiration)
- D3.js / Mike Bostock
- PureScript community
- Claude/Anthropic (with link to detailed page)

Phase 5: Publishing

5.1 Pre-publish Checklist

- [ ] All packages build independently
- [ ] Tests pass
- [ ] READMEs complete
- [ ] LICENSE files present
- [ ] CHANGELOG.md written
- [ ] spago.yaml publish sections configured
- [ ] Getting Started tested with published packages

5.2 Publish to Registry

```
# For each package
spago publish
```

5.3 Update Getting Started

Rewrite to use:

```
spago install psd3-selection psd3-simulation psd3-layout
```

Phase 6: Hosting & Launch

6.1 Website Hosting

- GitHub Pages for main documentation (current setup)
- Custom domain later (post soft-launch)

6.2 Code Explorer Hosting

- Deploy ce-website to Linode or similar
- Set up ce-server with database
- Document deployment process

6.3 Soft Launch

- Announce on PureScript Discourse
- Post in PureScript Discord
- Gather feedback

6.4 Broader Launch (Later)

- Custom domain
- Blog post / write-up
- Share with broader D3/dataviz community

Task Order Summary

1. **Clean up legacy references** (Datum_ pattern docs, wizard)
2. **Triage TODOs**
3. **Clean /docs directory**
4. **Write new README**
5. **Add basic tests**
6. **Split repos** (don't publish yet)
7. **Write package READMEs, LICENSE, CHANGELOG**
8. **Test packages build independently**
9. **Review/rewrite website content**

10. **Create AI collaboration page**
11. **Publish to registry**
12. **Update Getting Started with real package names**
13. **Deploy Code Explorer**
14. **Soft launch**

Notes

- ce-website hosting is parallel work, can happen alongside steps 5-11
- Custom domain is post-launch polish
- Keep `purescript-d3-tagless-II` repo as historical archive