



OS15-Accelerating Shiny Application Development and Validation with Rhino

Phuse US Connect 2025

18th March 2025

André Veríssimo
Vedha Viyash

What does it mean to validate a Shiny app?



The app produces consistent and reproducible behavior

What happens when building a shiny app?



Data

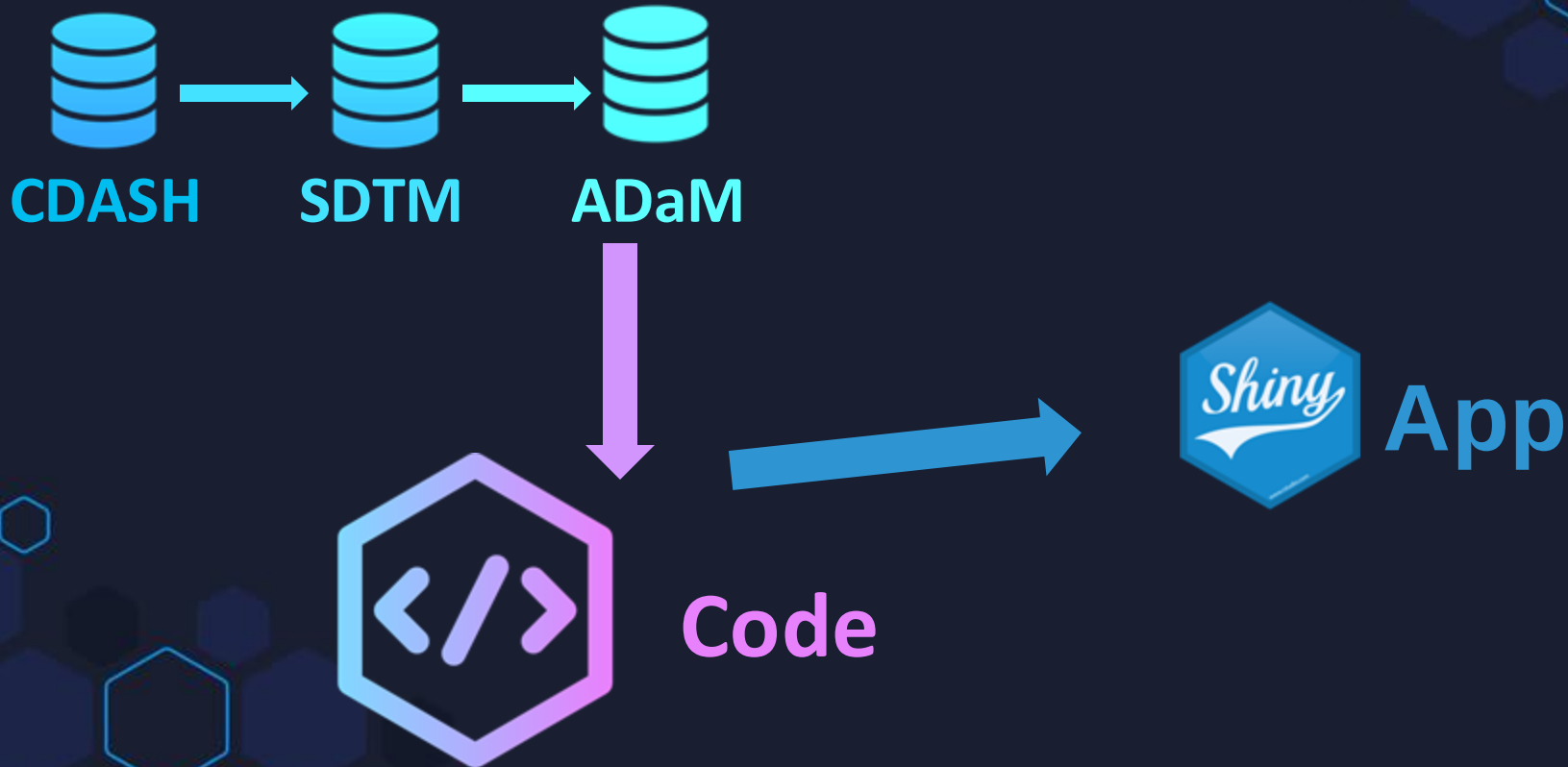


Code

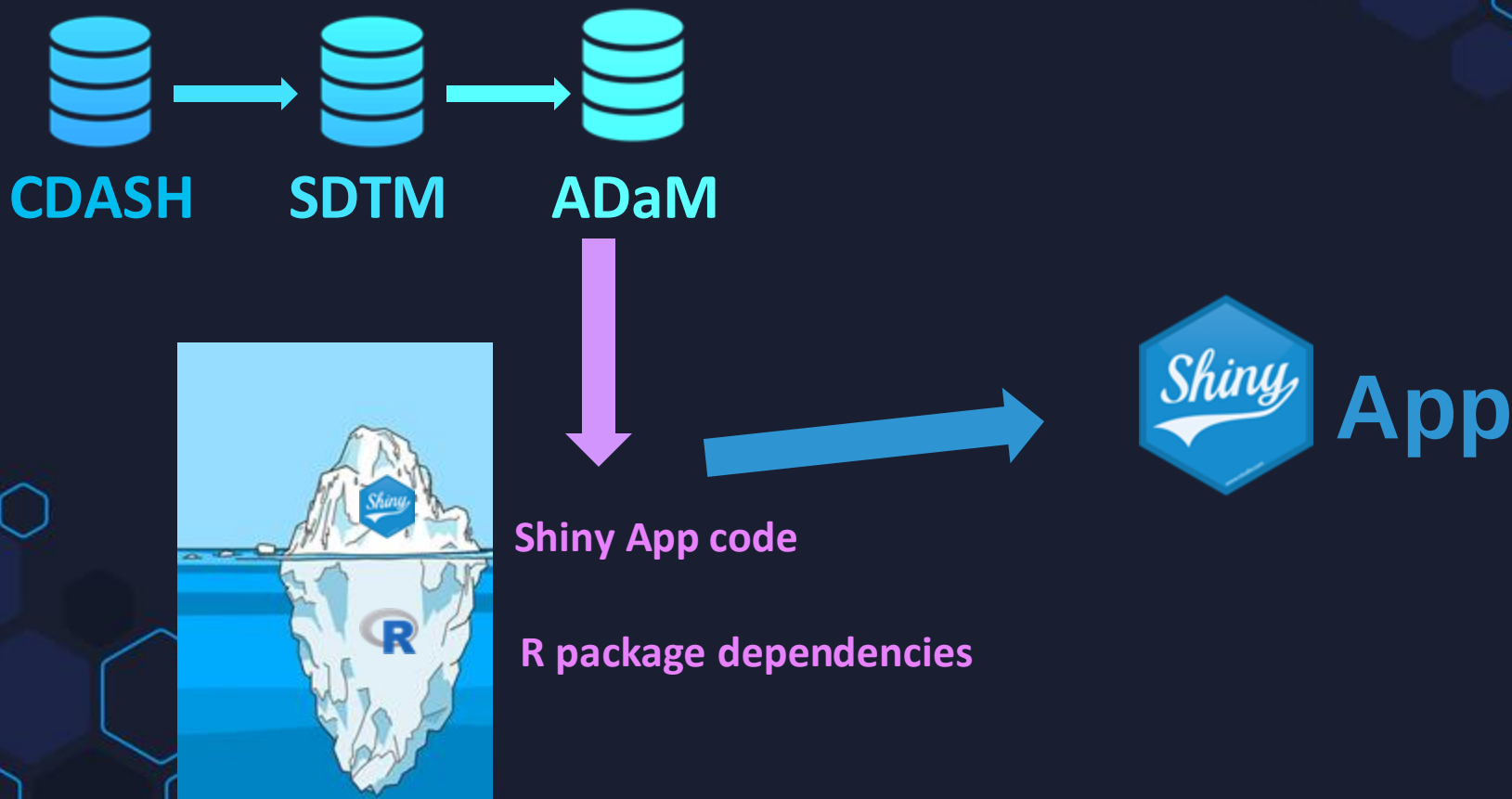


App

What happens when building a shiny app?



What happens when building a shiny app?



Validate in Stages



1. Data Validation



Shiny App code

3. App Validation

R package dependencies

2. Package Validation



Validate in Stages

~~1. Data Validation~~

We get Validated R packages
in a validated environment

~~2. Package Validation~~

3. App Validation

We just have to focus on consuming the data
and our app's logic

R package validation

Leverage best practices for R package development.

Minimize package scope.

Validate as a cohort in a controllable environment.

Source code traceability

Store metadata, test results and generate report

Publish in a validated repo.

R package validation

R Shiny app

Leverage best practices for R ~~package development~~.

Minimize ~~package scope~~.

Shiny app

Validate as a cohort in a controllable environment.

Source code traceability

Store metadata, test results and generate report

Deploy

server

~~Publish in a validated repo.~~

Risk based approach. Package vs Shiny app

Risk	Package	Shiny app
Low	Packages that have nothing to do with data	Function/Module has nothing to do with data
Medium	Automated checks are above certain threshold	Implement unit testing with a good coverage
High	Packages that are superseded. Extensive PQ tests before publication.	Implement extensive testing in functions/modules that manipulate data or update the database.
Critical	Rejected for general publication	Use of a rejected package in the app. Provide extensive PQ tests on the parts of the package that is used.

What are the steps for validation?



Dependency management within the SCE



Extensive testing (Functional, Security, UAT)



Access management and configuration



Easy maintenance throughout the lifecycle



Rhino uses these packages under the hood
“A framework for enterprise Shiny apps”

1. Project structure
2. Development toolbox
3. Enforces the good practices



File structure created by Rhino

```
rhino.yml
config.yml
.Rprofile
```

Config

```
app.R
├── app
│   └── main.R
├── logic
├── view
├── static
├── js
└── styles
```

Sources

```
├── renv
├── renv.lock
└── dependencies.R
```

Deps

```
├── tests
│   ├── cypress
│   ├── cypress.json
│   └── testthat
```

Tests

```
├── .github
├── .lintr
└── example.Rproj
```

Tools

Setup



Setup Project Files

renv::init()

lintr::use_lintr()

install Node.js

Setup CSS infrastructure

Setup JS infrastructure

Setup testing infrastructure

VS.



install Node.js

rhino::init()

Usage



Build SCSS to CSS

Bundle JS code

lintr::lint_dir()

lint CSS / SASS code via node.js

styler::style_dir()

testthat::test_dir()

Run cypress tests

logger::log_*

VS.



rhino::build_sass()

rhino::build_js()

rhino::format_r()

rhino::lint_r()

rhino::lint_js()

rhino::lint_sass()

rhino::test_r()

rhino::test_e2e()

rhino::log\$*



Standards improve efficiency

Standards improve efficiency

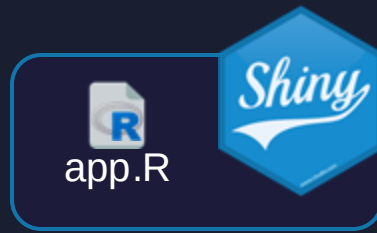
Exhibit A





app.R







app.R



data.R

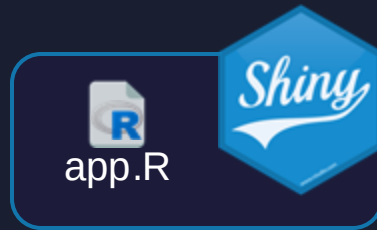
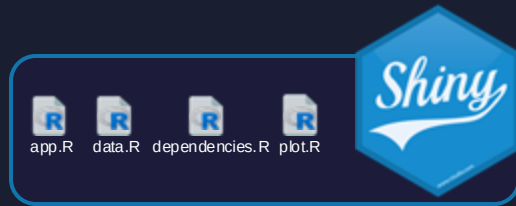


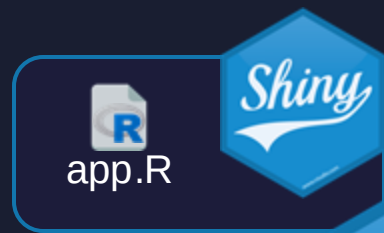
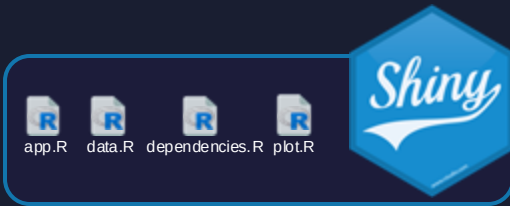
dependencies.R

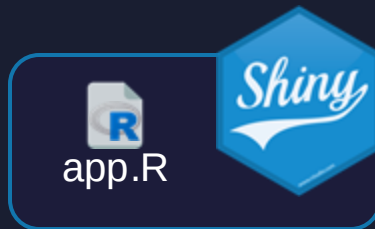
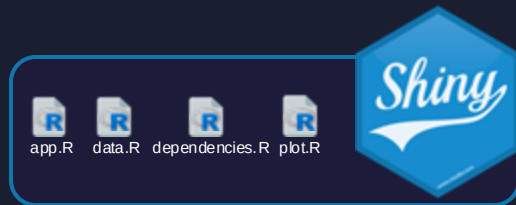


plot.R









app.R data.R dependencies.R plot.R

Shiny

app.R

Shiny

Shiny

app.R
modules data.R plot.R
style.css
testthat test-main.R

golem

DESCRIPTION

NAMESPACE

dev 01_start.R 02_dev.R 03_deploy.R run_dev.R
inst app www favicon.ico style.css
golem-config.yml
man run_app.Rd
R app_config.R app_server.R app_ui.R data.R run_app.R plot.R
tests testthat test-main.R

app.R data.R dependencies.R plot.R

Shiny

app.R

Shiny

golem

DESCRIPTION

NAMESPACE

dev 01_start.R 02_dev.R 03_deploy.R run_dev.R

inst app www favicon.ico style.css

MAN

golem-config.yml

man run_app.Rd

R app_config.R app_server.R app_ui.R data.R run_app.R plot.R

tests testthat test-main.R

R

app.R
modules data.R plot.R
style.css
testthat test-main.R

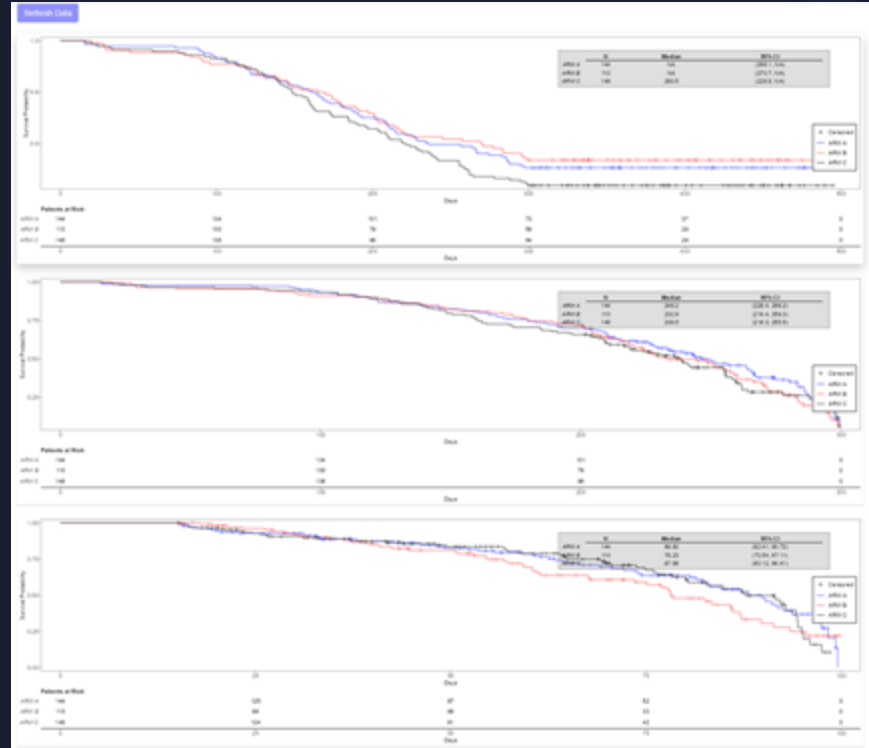
Shiny

app js index.js
main.R
logic __init__.R plot.R data_utils.R
view __init__.R data_refresh.R km_plot.R
styles main.scss
static css app.min.css
favicon.ico

app.R
dependencies.R
config.yml
renv activate.R settings.json
renv.lock
rhino.Rproj
rhino.yml
tests cypress e2e app.cy.js
cypress.config.js
testthat test-main.R



https://github.com/vedhav/shiny_app_structures





André Veríssimo
andre.verissimo@appsilon.com



Vedha Viyash
vedha@appsilon.com

Thank you



Appsilon

