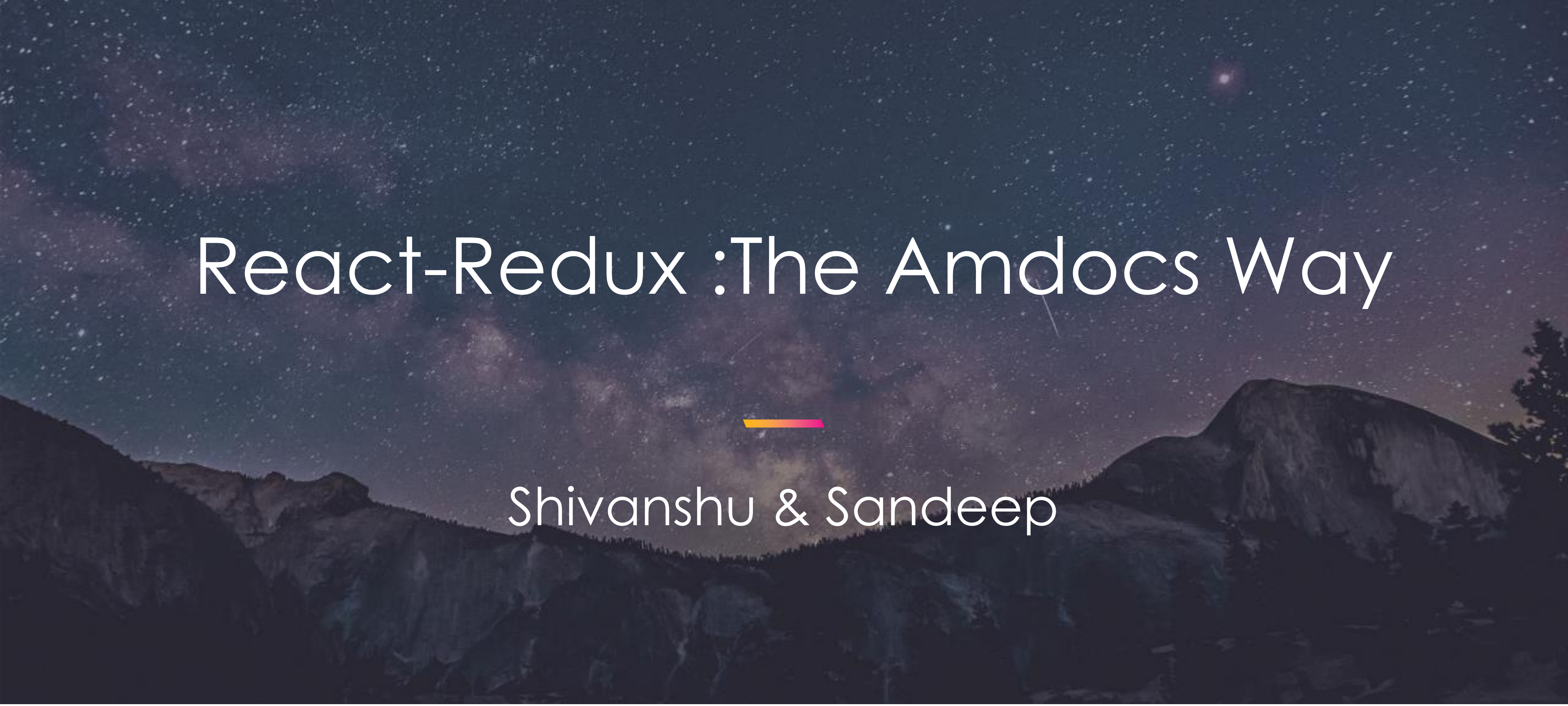


# React-Redux :The Amdocs Way

Shivanshu & Sandeep

# Agenda

- Problems with Redux State Management
- Solution – the Ideation phase
- Implementation  - the prototyping phase
- Impact

# Problems with Redux State Management

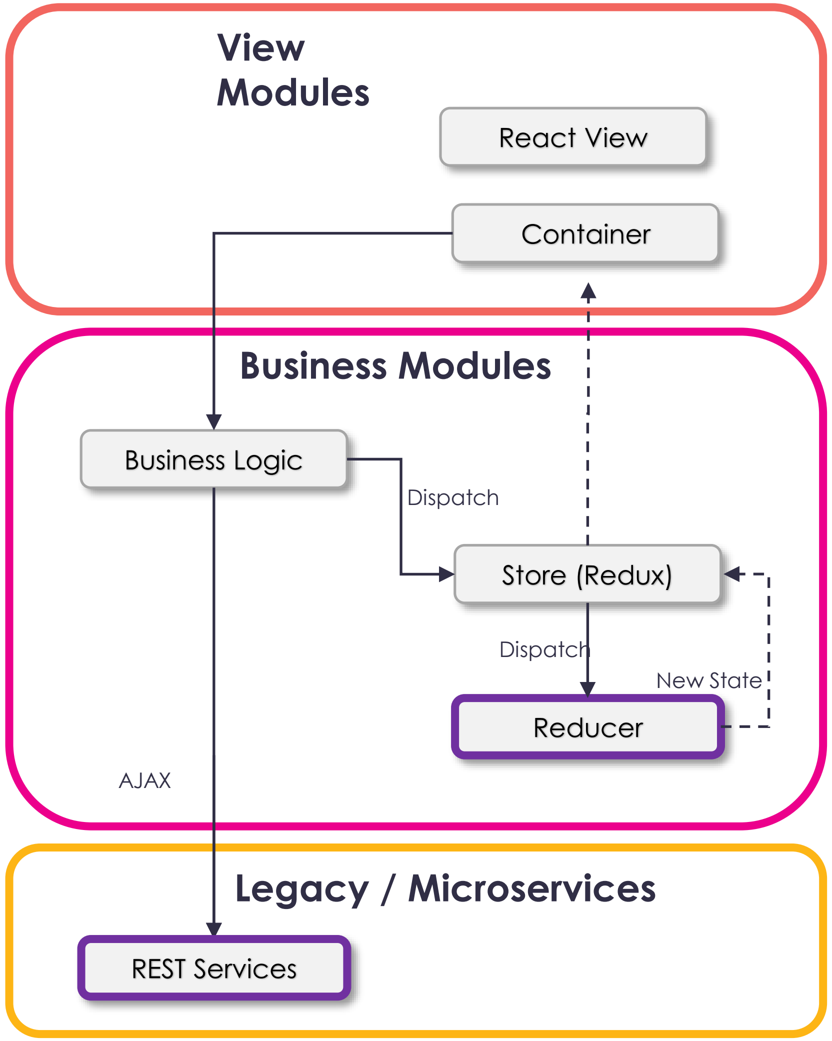
- No Encapsulation over shared state data
- Duplicate copies of business logic
- Component may be with multiple instances but store is singleton
- Confusion whether the state belongs to view or business

# Solution – the Ideation phase

A module which can …

- Encapsulate a part of global state
- Provide Interface to manage this state
- Has Single Responsibility

# Implementation - the prototyping phase



HTML via JSX notations; no business logic **making it lighter, UI focused, easily replaceable.**

**Business Module :: to keep business logic separate.** Encapsulate data and supply services (APIs) to manage it. Dispatches actions to manipulate store or fetch/update backend services.

Application state stored at single place as immutable Simple JS object which becomes single source of truth **making it easy to maintain**

REST full services

# Impact

The impact of this approach is …

- Faster development – allow to make changes to underlying structure of state without impacting the application
- Easy to Develop & Test
- Identification of Code Breaks with enabling the AST based BWC check tool
- Decoupling the business logic from the view, so making views agnostic of UI technologies/ frameworks
- Reusable Business modules

# Impact

| Before | After |
|---|---|
| Presentation was coupled by business logic, so reuse of business logic was minimal.  So across teams for same requirement the development effort was duplicated | Presentation & Business Logic separated, which facilitates the faster development of requirements. And also reuse is now to maximum. |
| Freedom to reuse business logic with different UI framework did not exist | Now Customers have a choice to select the UI framework. |
| State Manipulation was possible from different Views, so code breaks during development was not possible- which resulted in delayed deliverables | State Manipulation is now only possible through Business Modules. Also with AST we have developed a utility to identify the breaks during development phase |
| Now way to only test the business logic without writing the testcases for views | Business logic can have mock test cases ,and also able to test the business flow just by writing simple integration test cases. |

# Thank you