Redux

Feb 2020 | Kitchener, ON

Manulife

Agenda

- What is Redux?

- Why Redux?

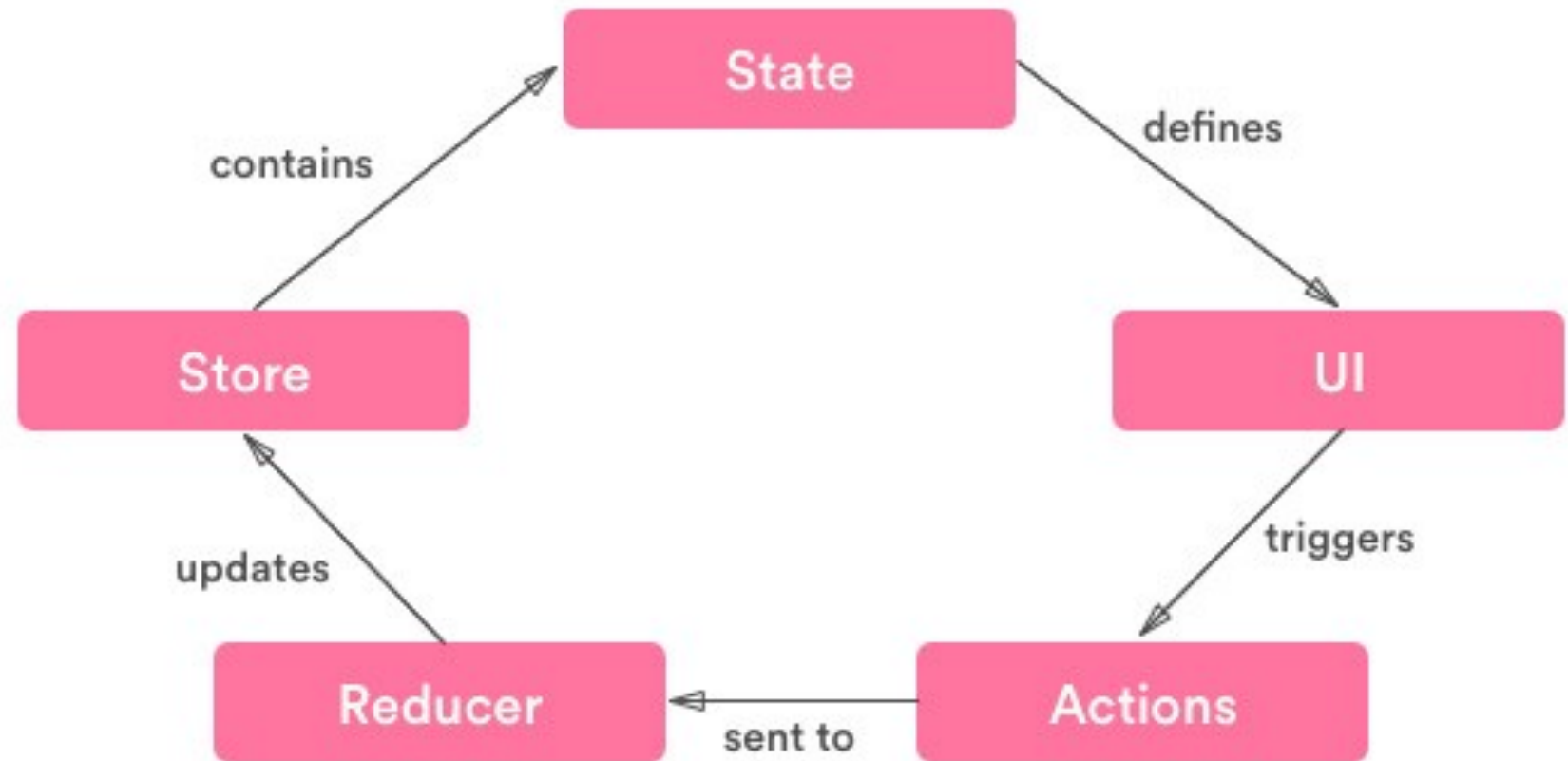- Exercise

Manulife

What is Redux?

Redux is a predictable state container for JavaScript applications.

Data flows in a circle, in one direction between the data store and the view.

The Redux DevTools make it easy to trace **when, where, why, and how** your application's state changed.

Manulife

Redux Data Flow

Manulife

Redux Data Store

- **A single data store** to capture the current state of the application

- The data store is a **single tree of data – a single object.**

- The view uses the data store to display what it needs to.

Redux Action

- Actions are plain JavaScript objects that include a **type** property and an associated **payload** or **data.**
  - {type: 'ADD_DOCTOR', data: doctor }

- To change something in the store, you need to **dispatch** an action.

- Action creators are functions that create actions.

Manulife

Redux Reducer

- A reducer is a pure function.

- It takes in the previous state and an action, then returns the new state

- Reducers typically have a switch statement inside that's checking for the action type, with code for each case to deal with that action

- Reducers should NOT have side effects. Do NOT:
  - mutate its arguments
  - have API calls and routing transitions
  - call non-pure functions (e.g. Date.now() or Math.random())

Redux Containers – Presentational vs Container Components

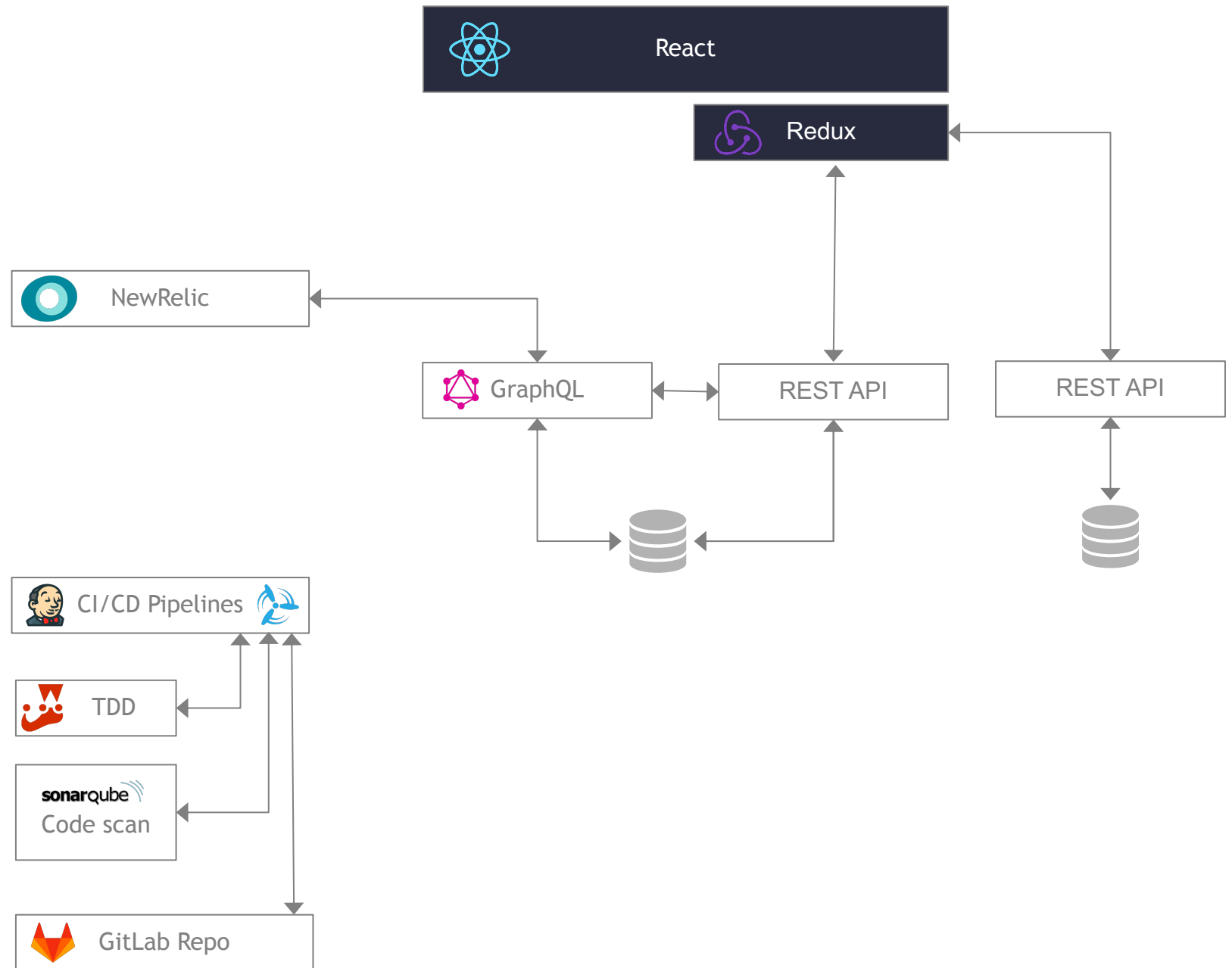|	Presentational Components	Container Components
Purpose	How things look (markup, styles)	How things work (data fetching, state updates)
Aware of Redux	No	Yes
To read data	Read data from props	Subscribe to Redux state
To change data	Invoke callbacks from props	Dispatch Redux actions
Are written	By hand	Usually generated by React Redux

Manulife

When to use Redux?

- When there are reasonable amounts of data changing over time
- A single source of truth of the state is required
- Keeping all the state in a top-level component is no longer sufficient

- Local state (with React) can still be used with Redux

Manulife

App Exercise 10:

Use Redux to connect your React UI to the coach-provided REST API.

React

Redux

NewRelic

GraphQL

REST API

REST API

CI/CD Pipelines

TDD

sonarqube
Code scan

GitLab Repo

Manulife

Exercise

Instructions for this exercise can be found in GitLab

https://git.platform.manulife.io/mu-materials/redux-exercises

Manulife