

Now i understood properly , that the virtual dom is created by "Createrooot" and then if there is any component experiences changes then the React will create new virtual dom within the component that experiences change. and the the both the dom will compared to each other to configure the changes by using the Reconciliation of diffing algorithm. and then the changes get updated to the real dom (dom of browser), am i right , Tell me in short .

Yes, you're correct!

Here's the summary:

1. The initial Virtual DOM is created by `createRoot`.
2. When a component experiences changes (due to state, props, or context updates), React creates a new Virtual DOM for that component.
3. React then compares the new Virtual DOM with the previous one using the reconciliation (diffing) algorithm.
4. Finally, only the necessary changes are updated in the real DOM (browser DOM).

You've got it!

not understood 😞 . kindly tell me again about this React fibre in more comprehensive way .

React Fiber is a new way of rendering and updating the UI in React, making React more efficient, flexible, and responsive. Let's break it down step by step:

1. What Problem Does React Fiber Solve?

Before Fiber, React used a process called **Reconciliation** to update the UI. However, this process could become slow and unresponsive for complex applications, especially when there were many elements to update.

React Fiber was introduced to make React more capable of handling large and complex user interfaces efficiently, without making the UI sluggish or unresponsive.

2. How Does React Fiber Work?

React Fiber works by breaking down the rendering work into small units of work, called **fibers**. These fibers can be paused, continued, or discarded depending on the priority of the task.

- **Breaking Down Work:** Instead of rendering everything at once, React Fiber breaks down the work into units and does a bit at a time.
- **Prioritization:** React Fiber assigns priorities to different tasks. For example, updating the UI after a user interaction is more important than background tasks. High-priority tasks are done first.
- **Pausing and Resuming:** If a high-priority task comes up while React is working on something else, React Fiber can pause the current work, handle the high-priority task, and then resume where it left off.

3. Phases of React Fiber

React Fiber operates in two main phases:

- **Reconciliation (Render Phase):** React Fiber goes through the component tree to figure out what changes are needed. During this phase, it can pause and resume work as needed, based on task priority.
- **Commit Phase:** Once React Fiber knows what changes to make, it commits those changes to the DOM in a single step. This phase is fast and cannot be interrupted.

4. Why Is React Fiber Important?

- **Improved Performance:** React Fiber makes updates to the UI smoother, especially in complex applications.
- **Better User Experience:** By prioritizing tasks, React Fiber ensures that the user interface remains responsive, even during heavy updates.
- **Flexibility:** React Fiber allows React to support new features and patterns like Concurrent Mode, which further improves the app's responsiveness by rendering components asynchronously.

5. Real-World Example

Imagine you have a complex app where a user is scrolling through a long list of items. With React Fiber, the scrolling can continue smoothly even if there are background updates happening at the same time, because React Fiber will prioritize the scrolling over less important tasks.

6. Conclusion

React Fiber is like a traffic controller for your app, making sure that the most important tasks are handled first, and that everything

happens smoothly without unnecessary delays. It's a big step forward in making React apps faster, more responsive, and able to handle complex user interfaces efficiently.

If you want more details or have specific questions, feel free to ask!

are you sure that this information is 100% accurate. Tell me in short.

Yes, the information is accurate. Here's a short summary:

React Fiber is the latest architecture of React, introduced to improve the rendering process. It breaks down rendering work into small units, allowing React to pause, prioritize, and resume tasks. This makes animations, transitions, and user interactions smoother, even during heavy updates. Fiber also enables features like concurrent mode for more responsive apps.

what is concurrent mode ??. Tell me in short.

Concurrent Mode in React is an advanced feature that allows React to work on multiple tasks simultaneously, without blocking the user interface. It helps make apps more responsive by prioritizing urgent updates (like user input) over less critical tasks (like data fetching), resulting in smoother user experiences.