Bundlers & RSC

A Tale of Two Runtimes

Raghav Khanna & Pranav Chatur — ReactJS Meetup Bangalore 23 August 2025

Production-ready code — 404

Entry — Extra — Elsewhere

It helps if you know:

- JavaScript
- Basics of React Components

You feel right at home with:

- HTTP Node.js Servers
- Developed Web-App before

Not our focus today:

- RSCs from scratch
- Bundler-specific implementations
- Comprehensive Bundler logics

Talk Structure
Presentation — Code — Q&A

1. Bundler - Dependency Resolution Demo (*no deps*)

2. React Server Components from a Node Server (*no deps again!*)

3. Optimise the RSC Implementation

4. How Parcel does it (*learn from Pros*)

5. QnA (*discussion really*)

Bundlers? Let's unpack that.

Bundler Cafe
Cafe serving Cappuccino and Black Coffee

- Menu:
  - Black Coffee
    - Espresso
  - Cappuccino
    - Espresso
    - Milk
    - Sugar

Bundler
Identify blocks + Order them

1. Identify Entry points:

   • i.e. all code present in /src

2. Process Dependency Graphs:

   • For each file, identify the complete list of dependencies (including nested)

3. Generate manifest.json

   • Add the dependency graph in a file to perform look ups later

Cafe Stonks!

Time to go on the web!

React Components on the Server

Resources Finite

Feature to add conditionally display items on the menu

Use Bundlers, not more code!

Parcel Implementation

How the big boys do

References
We stand on the shoulders of Giants

- [1]: Ben Holmes YouTube Video: **React Server Components From Scratch**

- [2]: Dan Abramov Blog: **Why Does RSC Integrate with a Bundler?**

- [3]: Christoph Nakazawa Blog: **Building a JavaScript Bundler**

- [4]: Parcel Documentation: **RSCs using Parcel**

Thank you
for indulging us

The code is available here