

# Calling services via proxy

Two potential setups for prod:

- Services and Front End always on diff servers
  - example: `example.com` and `api.example.com`
  - We won't do this, but is also common
- Services and Front End on same servers in prod
  - Both easier AND harder in dev
  - Can avoid server domains in fetch()
    - Easy to swap between dev and prod
  - You want just paths for fetch()
    - Example: `/api/cats`
    - Not: `https://api.example.com/api/cats`

# **Dev Server doesn't support services**

The CRA dev server doesn't let us write services

- So we would need a domain+port in fetch()
- But those would change in prod

# **Solution: Proxy the service calls**

CRA Dev Server:

- has front end files
- Front end calls services as path
- if a service call comes in
  - proxy it to services server
  - return results

Dev services server:

- Configured for static files
  - But doesn't have any
- Has services

# Moving to Prod

Once a round of development is done:

Prod Server:

- Just like dev services server
- Given static files from CRA `npm run build`
- Service calls made from front end are to paths
- Instead of proxying
  - responds to service calls itself

No changes in code between dev and prod

# **CRA Dev Server**

Build as normal

# Services Server Setup

Serves both static files and services

- Build express server
- Configure `express.static()`
  - Hint: serving `build/` instead of `public/` is convenient
- Write routes for service endpoints
- set the PORT to be different than CRA dev server (3000)

Will be actual server in the end

- Just one way to do it

# Installing packages

This is just one way to do it

- Create your `server.js` in your react project
- run your `npm install` so deps are added to the same package.json CRA created
- you may want to modify the `scripts` section of package.json
  - Example:
    - `npm run build` stays as it is
    - `npm start` starts the services server
    - `npm run dev`
      - runs the old `start` command (using react-scripts)

# Configuring the Proxy

CRA Dev server can proxy service calls in dev

- In prod there will be no proxy
- CRA Dev Server checks the package.json for config

add to package.json:

```
"proxy": "http://localhost:4000",
```

- where 4000 is the port you chose for services server



# Trying it out

Notice dev and prod will use different ports

- Common to set PORT by environment variable
- in services server:
  - `const PORT = process.env.PORT || 4000;`
- to start services server in dev:
  - `PORT=4000 npm run dev`
- to start server in prod:
  - `PORT=80 npm start`
  - (more likely 443 because https)
- `fetch()` using paths will use current origin
  - protocol + domain + port