

Migrating from Create React App (CRA) to Vite: A Step-by-Step Guide

This guide will walk you through the process of migrating from Create React App (CRA) to Vite, a fast-build tool for web development. Vite offers improved development speed and performance benefits. We'll go through each step to ensure a smooth transition.

Step 1: Install Vite and Plugins

To begin the migration, you'll need to install the necessary packages and plugins:

Command: `yarn add vite @vitejs/plugin-react vite-tsconfig-paths`

OR

Command: `npm install vite @vitejs/plugin-react vite-tsconfig-paths`

- `vite` is the Vite build tool.
- `@vitejs/plugin-react` enables fast refresh, automatic JSX runtime, and custom Babel plugins.
- `vite-tsconfig-paths` resolves TypeScript path mappings for clean imports.

You can also consider other Vite plugins like `vite-plugin-svgr` if your project requires SVG transformations.

Step 2: Create a Vite Config File

In your project's root directory, create a `vite.config.ts` file with the following content:

Typescript:

```
import { defineConfig } from 'vite';

import react from '@vitejs/plugin-react';

import viteTsconfigPaths from 'vite-tsconfig-paths';

export default defineConfig({

  base: '', // Depending on your application, base can also be "/"

  plugins: [react(), viteTsconfigPaths()],

  server: {

    open: true, // Automatically open the browser on server start

    port: 3000, // Set a default port

  },

});
```

This configuration file defines Vite's settings, including plugins and the development server.

Step 3: Create a Vite Types File Reference

Create a `vite-env.d.ts` file in your project's root directory with the following content:

```
/// <reference types="vite/client" />
```

This file is essential for type checks and Intellisense when using Vite.

Step 4: Move the index.html File

Move the `index.html` file from the `public` directory to the root of your project. Vite serves files from the root directory.

Step 5: Update the index.html File

In your `index.html` file, make two updates:

Remove `%PUBLIC_URL%` placeholders and replace them with direct URLs. For example:

Before:

```
<link rel="icon" type="image/svg+xml" href="%PUBLIC_URL%/favicon.svg" />
```

After:

```
<link rel="icon" type="image/svg+xml" href="/favicon.svg" />
```

Add a module script to the bottom of the body tag:

```
<script type="module" src="/src/index.tsx"></script>
```

This script tag is crucial for Vite to resolve module imports correctly.

Step 6: Replace CRA with Vite

To fully transition to Vite, you need to make changes to your project's scripts and configuration:

- Remove CRA by running:

```
"yarn remove react-scripts"
```

- OR

```
"npm uninstall react-scripts"
```

- Update your `package.json` scripts to use Vite. Replace `react-scripts` with `vite` in the `start` and `build` scripts:

```
"scripts": {  
  "start": "vite",  
  "build": "tsc && vite build",  
  "preview": "vite preview"  
}
```

- Update your `tsconfig.json` to match Vite's configuration. Focus on settings like `isolatedModules`, `lib`, `target`, and `types`. Here's an example:

```
{  
  
  "compilerOptions": {  
  
    "lib": ["dom", "dom.iterable", "esnext"],  
  
    "target": "ESNext",  
  
    "types": ["vite/client"],  
  
    "isolatedModules": true  
  
  }  
}
```

• Step 7: Run Your Application

You can now start your application using Vite:

“yarn start”

OR

“npm start”

Congratulations! You've successfully migrated your application from CRA to Vite.