

# RAISING THE BAR

---

React Native Connection 2023

# HI! I'M TOMMY

- Principal Software Engineer at Microsoft
- [@tido64](#) on GitHub
- I've put React Native in Outlook Mobile



The screenshot shows a Mac OS X desktop environment with a dark-themed browser window open to [reactnative.dev](https://reactnative.dev). The browser's address bar shows the URL. The page content is the 'Showcase' section of the React Native documentation.

**Microsoft Showcase:**

- Microsoft Office**: Android • iOS. [Learn more](#)
- Microsoft Outlook**: Android • iOS. [Learn more](#)
- Microsoft Teams**: Android • iOS. [Learn more](#)
- Xbox Game Pass**: Android • iOS. [Learn more](#)
- Skype**: Android • iOS. [Learn more](#)

**Shopify Showcase:**

All new mobile apps at Shopify are React Native and we are actively migrating our flagship merchant admin app Shopify Mobile to React Native as well. You can read more about React Native development at Shopify on our [blog](#).

The screenshot shows the Microsoft Store interface with a dark theme. At the top, there are two large game banners: "Ori and the Will of the Wisps" on the left and "Cyberpunk 2077" on the right. Below the banners is a search bar with the placeholder "Search for games". To the left of the search bar is a user profile icon for "Storm Yeti". A sidebar on the far left contains icons for various categories: Profile, Search (highlighted with a white border), My Library, Cloud Gaming, Community, Store, Installed Games, REDEEM A CODE, DEALS, Games (highlighted), Recently Added, and Settings.

**RECOMMENDED GAMES**

- Halo Infinite**  
The legendary Halo series returns with the most expansive Master Chief campaign yet and a ground-breaking free-to-play multiplayer experience.
- Gears 5**  
From one of gaming's most iconic sagas, Gears 5's celebrated

**RECENTLY ADDED**

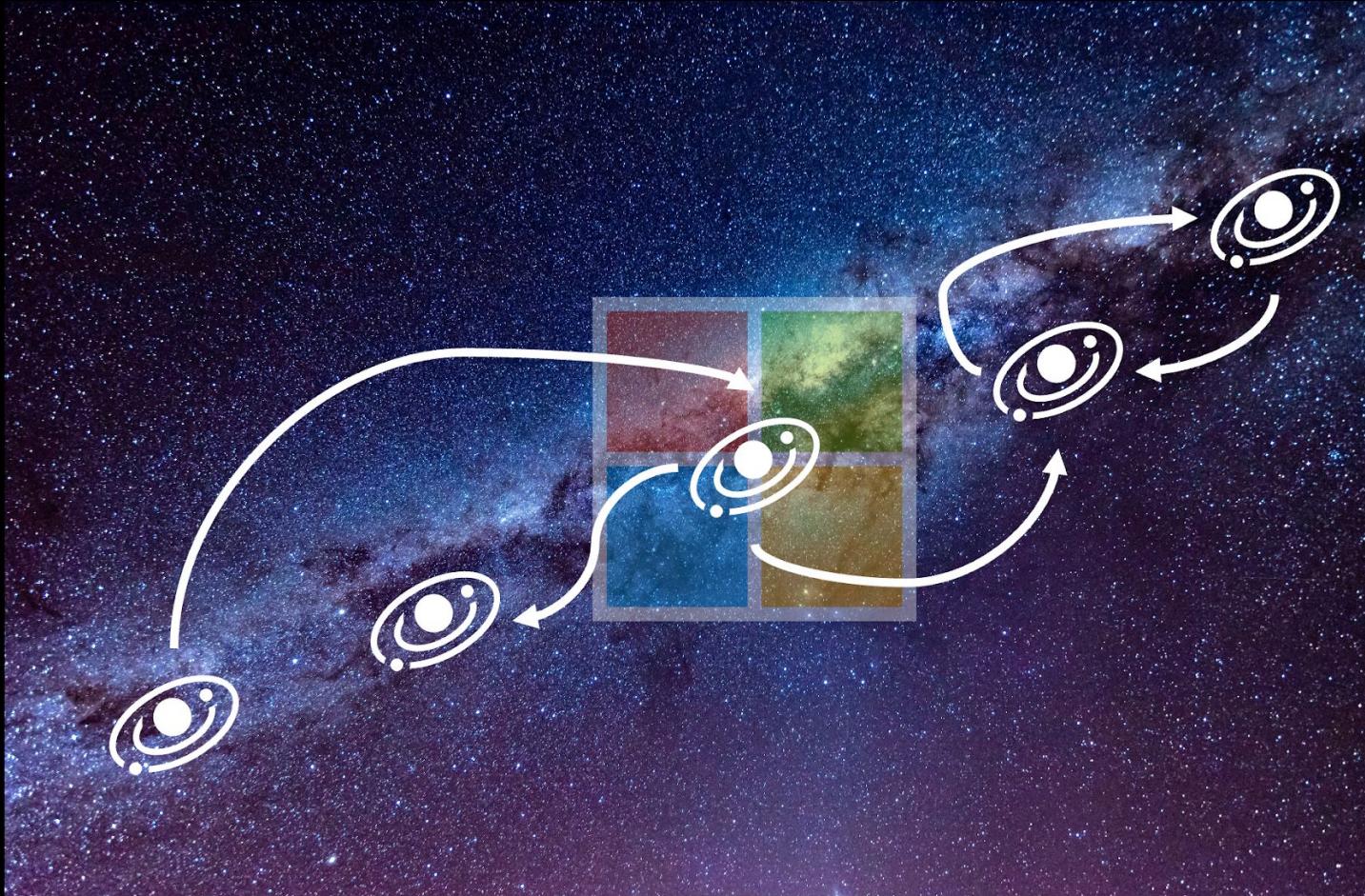
- FORZA HORIZON 5
- AGE OF EMPIRES IV
- PSYCHONAUTS
- Ori and the Will of the Wisps
- MINECRAFT
- STATE OF DECAY 2 JUGGERNAUT EDITION

# **INVISIBLE**

---

# The Microsoft galaxy

- Microsoft has many monorepos
- A monorepo is a place in which many different packages coexist, aware of each other

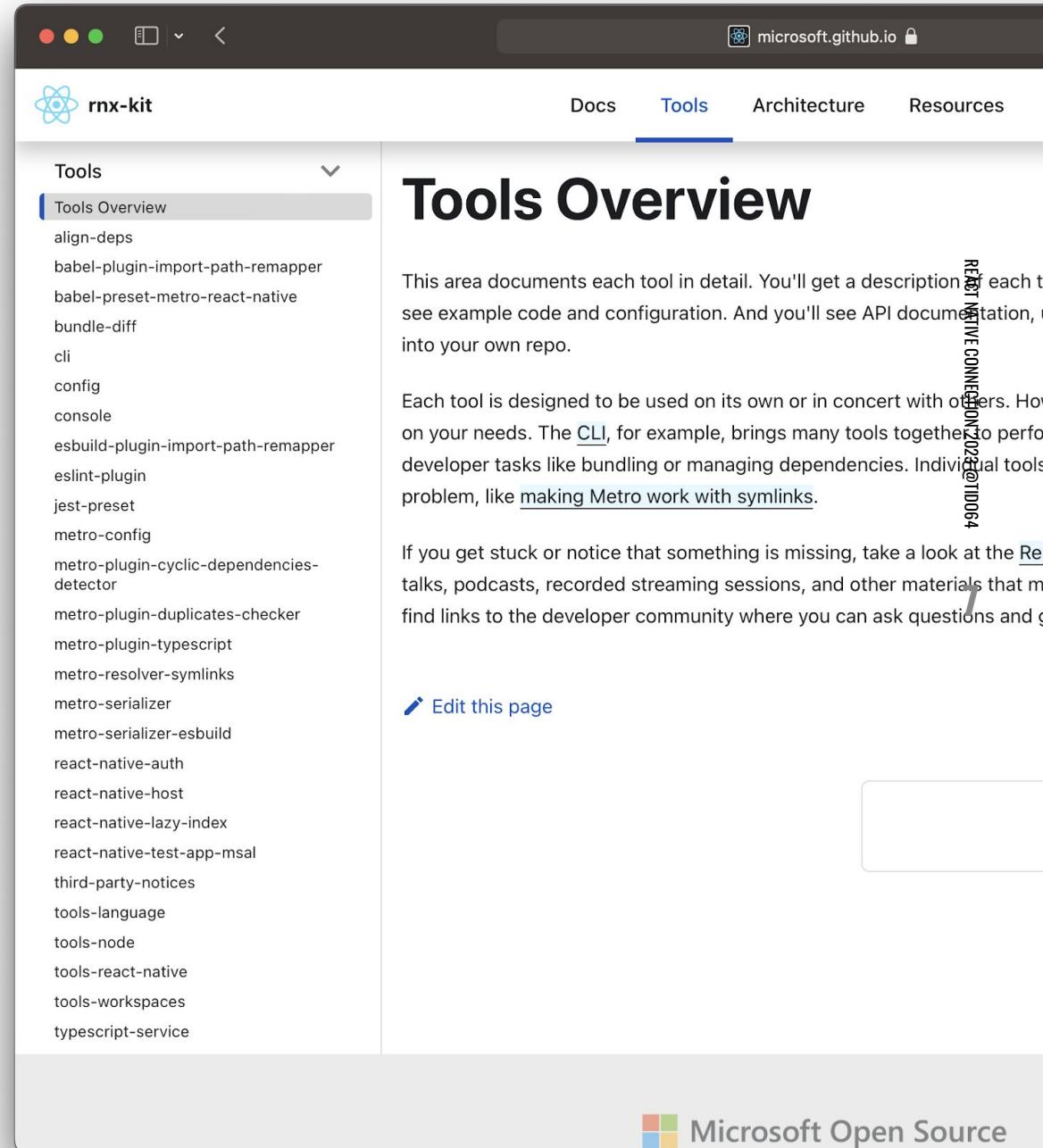


2

A slide from the React Native EU 2021 talk (link at the end)

# RNX-KIT

- Plugins for Metro
- Presets for Babel, ESLint, Jest, Metro
- Drop-in replacement for CLI
  - “All-in-one”



The screenshot shows a web browser window displaying the [rnx-kit](https://microsoft.github.io/rnx-kit/) documentation. The page has a dark header with the Microsoft logo and the URL. Below the header, there are navigation links: Docs, Tools (which is underlined in blue), Architecture, and Resources. The main content area has a title "Tools Overview" with a dropdown arrow. A list of tool names is displayed, starting with "align-deps" and ending with "typescript-service". The right side of the page features a large heading "Tools Overview" and a descriptive paragraph about the tools. There are also sections for "REACT NATIVE CONNECTOR" and "React Native CLI" on the far right.

## Tools Overview

This area documents each tool in detail. You'll get a description of each tool, see example code and configuration. And you'll see API documentation for each tool, so you can start using them in your own repo.

Each tool is designed to be used on its own or in concert with others. How you use them depends on your needs. The [CLI](#), for example, brings many tools together to perform developer tasks like bundling or managing dependencies. Individual tools solve specific problems, like [making Metro work with symlinks](#).

If you get stuck or notice that something is missing, take a look at the [Resources](#) section. It includes talks, podcasts, recorded streaming sessions, and other materials that might help. You can also find links to the developer community where you can ask questions and get answers.

[Edit this page](#)

align-deps  
babel-plugin-import-path-remapper  
babel-preset-metro-react-native  
bundle-diff  
cli  
config  
console  
esbuild-plugin-import-path-remapper  
eslint-plugin  
jest-preset  
metro-config  
metro-plugin-cyclic-dependencies-detector  
metro-plugin-duplicates-checker  
metro-plugin-typescript  
metro-resolver-symlinks  
metro-serializer  
metro-serializer-esbuild  
react-native-auth  
react-native-host  
react-native-lazy-index  
react-native-test-app-msal  
third-party-notices  
tools-language  
tools-node  
tools-react-native  
tools-workspaces  
typescript-service

# TREE SHAKING IN METRO



**renchap** 06/10/2022 16:05

I went from a 18 MB to a 4 MB bundle by switching to your esbuild serializer and removing / fixing a few deps

It really helped the startup time (and the download / on-device size, obviously)



**Janic**

@janicduplessis

...

Got [@th3rdwave](#) JS bundle size down from 7.9mb to 6.6mb using [microsoft.github.io/rnx-kit/docs/g...](https://microsoft.github.io/rnx-kit/docs/g...) Hit a couple issues, but overall works well!

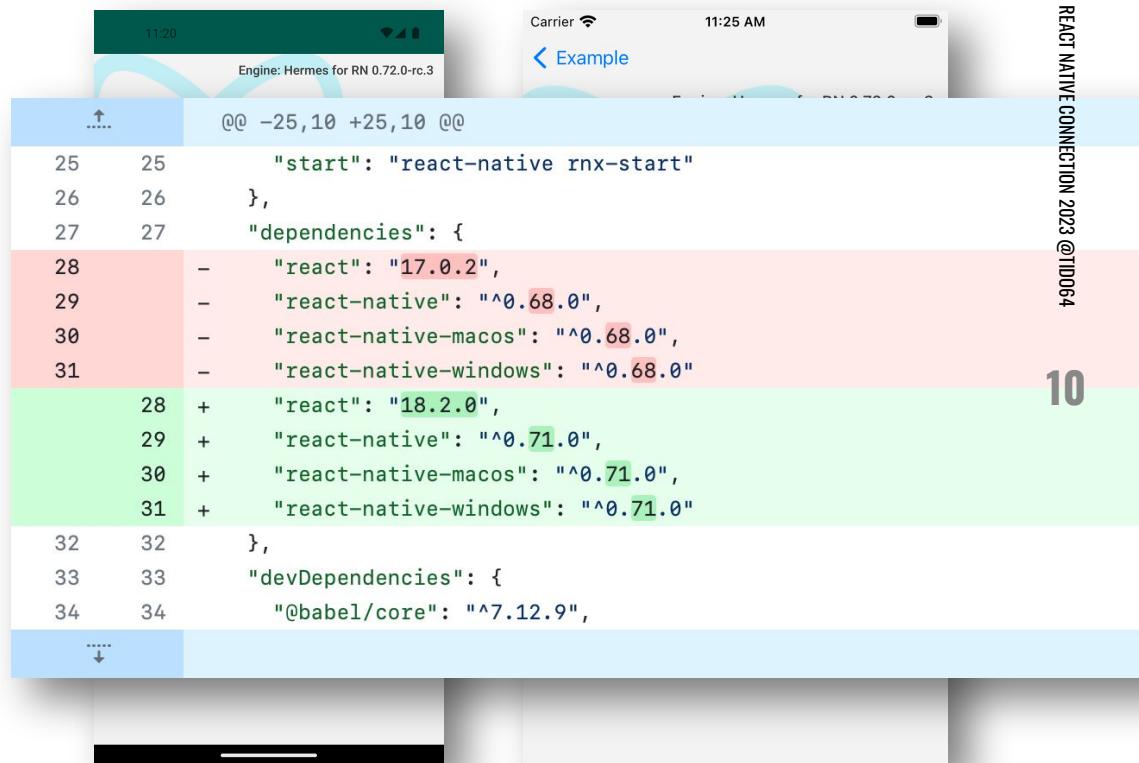
# THE PAIN OF UPGRADING

- Huge diffs
- What other dependencies do I need to bump?

# EFFORTLESS UPGRADES

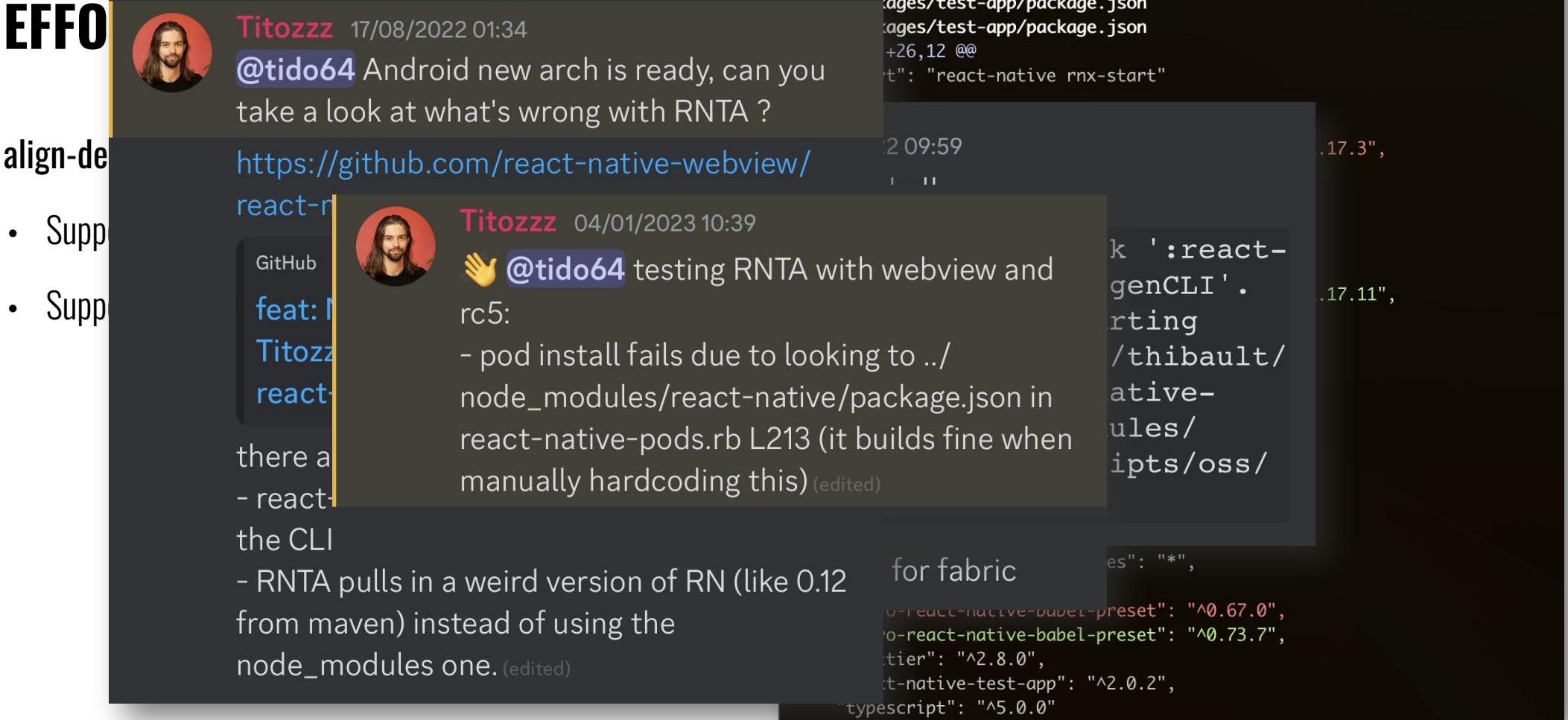
## react-native-test-app

- Supports 0.64 through 0.72
- Supports New Architecture
- Supports [@expo/config-plugins](#)
- Experimental single app mode



The screenshot shows a code diff interface comparing two versions of a React Native project's package.json file. The left side shows the original code (version 0.68.0) and the right side shows the updated code (version 0.71.0). The changes are highlighted with color-coded background overlays: red for removed code, green for added code, and blue for unchanged code.

```
11:20 Engine: Hermes for RN 0.72.0-rc.3  
11:25 AM Example  
... @@ -25,10 +25,10 @@  
25 25 "start": "react-native rnx-start"  
26 26 },  
27 27 "dependencies": {  
28 - "react": "17.0.2",  
29 - "react-native": "^0.68.0",  
30 - "react-native-macos": "^0.68.0",  
31 - "react-native-windows": "0.68.0"  
28 + "react": "18.2.0",  
29 + "react-native": "0.71.0",  
30 + "react-native-macos": "0.71.0",  
31 + "react-native-windows": "0.71.0"  
32 32 },  
33 33 "devDependencies": {  
34 34 "@babel/core": "7.12.9",  
...  
10
```



**INVISIBLE\***

---

microsoft/rnx-kit Public

Code Issues 80 Pull requests 6 Discussions Actions Projects 1 Security Insights

Dependency Dashboard

#1680 opened on Jun 23, 2022 by renovate bot

Open

Use align-deps to help migrate apps to the New Architecture

#1863 opened on Sep 2, 2022 by motiz88

Open 24

is:issue is:closed label:bug

Labels 25 Milestones 0 New issue

Clear current search query, filters, and sorts

6 Open ✓ 76 Closed

Author ▾ Label ▾ Projects ▾ Milestones ▾ Assignee ▾ Sort ▾

# THANK YOU FOR ALL YOUR CONTRIBUTIONS

needs attention

#2419 by Thanaen was closed 2 days ago 4 of 5 tasks

✓ esbuild serializer causes Gradle to crash when the bundle generates a warning bug feature: cli upstream 8

#2416 by renchap was closed 2 days ago 2 of 5 tasks

✓ eslint-plugin: no-export-all misses exports defined with TS "export import =" bug feature: eslint 1

#2375 by ecraig12345 was closed on Apr 18 1 task done

✓ eslint-plugin: incorrect expansion of namespace exports in fixer bug feature: eslint 1

#2373 by ecraig12345 was closed on Apr 18 1 task done

✓ metro-serializer-esbuild: /\*#\_\_PURE\_\_\*/ markers are not respected bug feature: metro 2

#2365 by tido64 was closed on Apr 24 1 of 5 tasks

✗ hermes seems to be a required capability in RN Profile 0.71 bug feature: align-deps 1

#2331 by SparshaSaha was closed on Apr 11 2 of 5 tasks

# WHAT WE'VE BUILT SO FAR

## Ease of use

- Smarter defaults
- Transparent

## Upgradability

- align-deps
- react-native-test-app

## Advanced scenarios

- Hybrid apps
- Monorepos
- Tree shaking
- TypeScript



# DEVELOPER VELOCITY

“I want to put my React Native feature in  
*any* app”

“I want to run *web* code in my React  
Native app”

# WEB/NATIVE CONVERGENCE

```
function GetBatteryLevelButton({ setBatteryLevel }) {
  const getBatteryLevel = useCallback(() => {
    navigator.getBattery().then((battery) => setBatteryLevel(battery.level));
  }, [setBatteryLevel]);
  return <button onClick={getBatteryLevel}>Get Battery Level</button>;
}
```

# WEB/NATIVE CONVERGENCE

```
function GetBatteryLevelButton({ setBatteryLevel }) {
  const getBatteryLevel = useCallback(() => {
    navigator.getBattery().then((battery) => setBatteryLevel(battery.level));
  }, [setBatteryLevel]);
  return <button onClick={getBatteryLevel}>Get Battery Level</button>;
}
```

# WEB/NATIVE CONVERGENCE

```
function GetBatteryLevelButton({ setBatteryLevel }) {
  const getBatteryLevel = useCallback(() => {
    navigator.getBattery().then((battery) => setBatteryLevel(battery.level));
  }, [setBatteryLevel]);
  return <button onClick={getBatteryLevel}>Get Battery Level</button>;
}
```

# USER INTERFACE

- RFC: React DOM for Native
  - [github/react-native-community/discussions-and-proposals/pull/496](https://github.com/react-native-community/discussions-and-proposals/pull/496)
- Bridge web/native via “Strict DOM”

The screenshot shows a GitHub pull request page for the repository `react-native-community / discussions-and-proposals`. The pull request is titled "RFC: React DOM for Native (reduce API fragmentation) #496". It is marked as "Open" and shows 9 commits from the branch `necolas:reduce-f` into the main branch. The pull request has 142 conversations, 9 commits, 0 checks, and 1 file changed. A comment by user `necolas` is visible, dated Aug 5, 2022, with an update in Feb 2023. The comment reproduces the contents of the RFC and links to another RFC: "RFC: DOM traversal and layout APIs in React Native". Below the comment, there are sections for "Summary" and "Motivation".

**RFC: React DOM for Native (reduce API fragmentation) #496**

**Open** necolas wants to merge 9 commits into `react-native-community:main` from `necolas:reduce-f`

Conversation 142 Commits 9 Checks 0 Files changed 1

**necolas** commented on Aug 5, 2022 • edited

Updated: Feb 2023

The contents of the [RFC](#) are reproduced below.

Also see [RFC: DOM traversal and layout APIs in React Native](#)

---

## RFC: React DOM for Native

### Summary

This is a proposal to incrementally reduce the API fragmentation faced by developers using React to target multiple platforms via code shared between native and web. The proposed cross-platform user interface APIs are a subset of web standards for DOM, CSS, and HTML - "Strict DOM". The proposed changes are overwhelmingly additive and require migration of existing React Native UI code (deprecations are optional / follow up work). Incremental progress to reduce the fragmentation between React Native and React DOM components, while the ability to run React DOM (with minor modifications) on native is a longer-term goal.

### Motivation

React Native currently includes many APIs that are modelled on Web APIs but do not conform to the standards. These APIs. React Native also includes many APIs that achieve the same results on Android and iOS but are exposed as props. And React Native includes several APIs that have known performance (network and runtime) drawbacks.

# WEB APIs

```
function GetBatteryLevelButton({ setBatteryLevel }) {
  const getBatteryLevel = useCallback(() => {
    navigator.getBattery().then((battery) => setBatteryLevel(battery.level));
  }, [setBatteryLevel]);
  return <button onClick={getBatteryLevel}>Get Battery Level</button>;
}
```

# WEB APIs

- Well-defined API
- Standard
- For React Native?

The screenshot shows a web browser window displaying the MDN Web Docs website at developer.mozilla.org. The page title is "Web APIs". The URL in the address bar is "developer.mozilla.org". The page content includes a section titled "What are Web APIs?" with a sub-section "Using Web APIs". Below this, there is a section titled "Specifications" which lists APIs starting with 'A', 'B', and 'C'. The "A" section includes links to "Audio Output Devices API" and "Background Fetch API". The "B" section includes links to "Barcode Detection API", "Background Sync", "Background Tasks", "Battery API", and "Beacon". The "C" section includes links to "CSS Counter Styles" and "CSS Typed Object Model". The right edge of the screenshot has a vertical text overlay that reads "REACT NATIVE CONNECTION 2023 @ MIRE64".

What are Web APIs?

Using Web APIs

When writing code for the Web, there are a large number of APIs available. A API may be a list of all the APIs and interfaces (object types) that you can use when developing your Web app or site.

Web APIs are typically used with JavaScript, although this doesn't have to be the case.

## Specifications 22

This is a list of all the APIs that are available.

### A

- [Audio Output Devices API](#)

### B

- [Background Fetch API](#)
- [Barcode Detection API](#)
- [Background Sync](#)
- [Battery API](#)
- [Background Tasks](#)
- [Beacon](#)

### C

- [CSS Counter Styles](#)
- [CSS Typed Object Model](#)



```

JS generate.mjs
1 import * as typescript from 'typescript';
2
3 const compatData = "https://unpkg.com@mdn/browser-compat-data/data.json";
4
5 // https://developer.mozilla.org/en-US/docs/Web/API
6 const ignoredSpecs = [
7   "https://dom.spec.whatwg.org/", // DOM
8   "https://drafts.css-houdini.org/", // CSS Houdini
9   "https://drafts.fxtf.org/", // W3C CSS-SVG effects
10  "https://html.spec.whatwg.org/multipage/browsing-the-web.html", // HTML
11  "https://html.spec.whatwg.org/multipage/canvas.html", // Canvas
12  "https://html.spec.whatwg.org/multipage/common-dom-interfaces.html", // DOM
13  "https://html.spec.whatwg.org/multipage/custom-elements.html", // DOM
14  "https://html.spec.whatwg.org/multipage/dom.html", // DOM
15  "https://html.spec.whatwg.org/multipage/dynamic-markup-insertion.html", // DOM
16  "https://html.spec.whatwg.org/multipage/editors.html", // DOM
17  "https://html.spec.whatwg.org/multipage/embedded-content.html", // DOM
18  "https://html.spec.whatwg.org/multipage/form-control-infrastructure.html", // DOM
19  "https://html.spec.whatwg.org/multipage/form-elements.html", // DOM
20  "https://html.spec.whatwg.org/multipage/forms.html", // DOM
21  "https://html.spec.whatwg.org/multipage/grouping-content.html", // DOM
22  "https://html.spec.whatwg.org/multipage/history.html", // DOM
23  "https://html.spec.whatwg.org/multipage/iframe-embed-object.html", // DOM
24  "https://html.spec.whatwg.org/multipage/image-maps.html", // DOM
25  "https://html.spec.whatwg.org/multipage/imagebitmap-and-animations.html", // DOM
26  "https://html.spec.whatwg.org/multipage/input.html", // DOM
27  "https://html.spec.whatwg.org/multipage/interaction.html", // DOM
28  "https://html.spec.whatwg.org/multipage/interactive-elements.html", // DOM
29  "https://html.spec.whatwg.org/multipage/media.html", // DOM
30  "https://html.spec.whatwg.org/multipage/obsolete.html", // DOM
31  "https://html.spec.whatwg.org/multipage/parsing.html", // DOM
32  "https://html.spec.whatwg.org/multipage/scripting.html", // DOM
33  "https://html.spec.whatwg.org/multipage/sections.html", // DOM
34  "https://html.spec.whatwg.org/multipage/semantics.html", // DOM
35  "https://html.spec.whatwg.org/multipage/tables.html", // DOM
36  "https://html.spec.whatwg.org/multipage/text-level-semantics.html", // DOM
37  "https://html.spec.whatwg.org/multipage/web-messaging.html", // Broadcast
38  "https://html.spec.whatwg.org/multipage/window-object.html#htmldocument", // HTMLDocument
39  "https://html.spec.whatwg.org/multipage/workers.html", // Workers
40  "https://immersive-web.github.io/", // WebXR
41  "https://registry.khronos.org/", // WebGL
42  "https://svgwg.org/", // SVG
43  "https://w3c.github.io/DOM-Parsing/", // XMLSerializer
44  "https://w3c.github.io/IntersectionObserver/", // IntersectionObserver
45  "https://w3c.github.io/ServiceWorker/", // ServiceWorker
46  "https://w3c.github.io/csswg-drafts/", // CSS
47  "https://w3c.github.io/mathml-core/", // MathML
48  "https://w3c.github.io/mediacapture-fromelement", // Canvas
49  "https://w3c.github.io/uievents/", // DOM events
50  "https://webidl.spec.whatwg.org/", // DOMException
51  "https://xhr.spec.whatwg.org/#dom-formdata", // XMLHttpRequest
52 ];
53
54 /**
55  * @typedef {{
56  *   error: typeof console.error;
57  * }
58  */
59
60 
```

TS lib.d.ts U incubator/standard-api/lib.d.ts • Performance

```

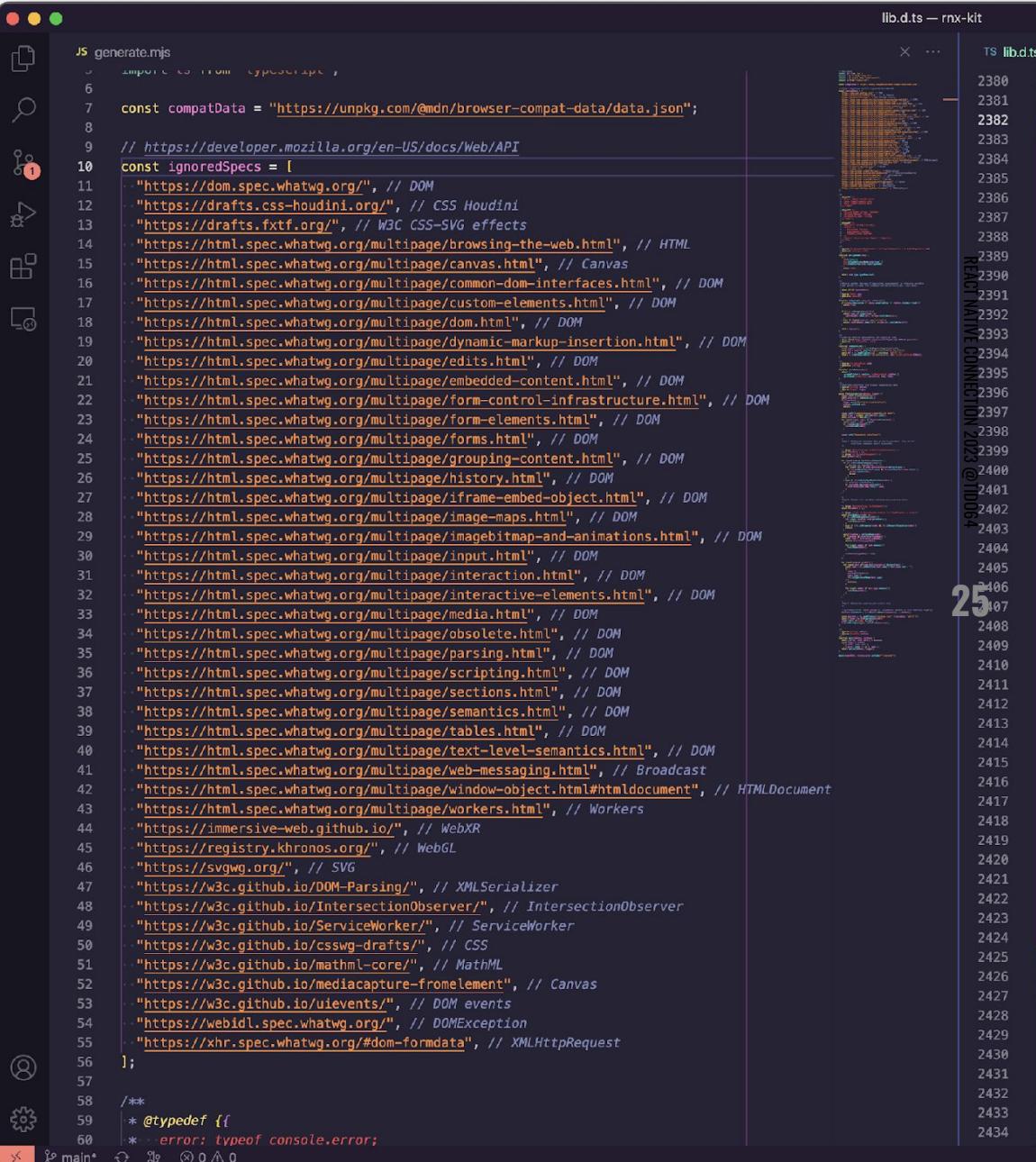
2380  }
2381  /** Provides access to performance-related information for the current page. It's part of the Performance API. */
2382  interface Performance extends EventTarget {
2383    readonly eventCounts: EventCounts;
2384    /* @deprecated */
2385    readonly navigation: PerformanceNavigation;
2386    onresourceTimingbufferfull: ((this: Performance, ev: Event) => any) | null;
2387    readonly timeOrigin: DOMHighResTimeStamp;
2388    /* @deprecated */
2389    readonly timing: PerformanceTiming;
2390    clearMarks(markName: string): void;
2391    clearMeasures(measureName: string): void;
2392    clearResourceTimings(): void;
2393    getEntries(): PerformanceEntryList;
2394    getEntriesByName(name: string, type?: string): PerformanceEntryList;
2395    getEntriesByType(type: string): PerformanceEntryList;
2396    mark(markName: string, markOptions?: PerformanceMarkOptions): PerformanceMark;
2397    measure(
2398      measureName: string,
2399      startOrMeasureOptions?: string | PerformanceMeasureOptions,
2400      endMark?: string
2401    ): PerformanceMeasure;
2402    now(): DOMHighResTimeStamp;
2403    setResourceTimingBufferSize(maxSize: number): void;
2404    toJSON(): any;
2405    addEventListener<K extends keyof PerformanceEventMap>(
2406      type: K,
2407      listener: (this: Performance, ev: PerformanceEventMap[K]) => any,
2408      options?: boolean | AddEventListenerOptions
2409    ): void;
2410    addEventListener(
2411      type: string,
2412      listener: EventListenerOrEventListenerObject,
2413      options?: boolean | AddEventListenerOptions
2414    ): void;
2415    removeEventListener<K extends keyof PerformanceEventMap>(
2416      type: K,
2417      listener: (this: Performance, ev: PerformanceEventMap[K]) => any,
2418      options?: boolean | EventListenerOptions
2419    ): void;
2420    removeEventListener(
2421      type: string,
2422      listener: EventListenerOrEventListenerObject,
2423      options?: boolean | EventListenerOptions
2424    ): void;
2425  }
2426  /** Encapsulates a single performance metric that is part of the performance timeline. A performance entry has a duration, an entry type, a name, and a start time. */
2427  interface PerformanceEntry {
2428    readonly duration: DOMHighResTimeStamp;
2429    readonly entryType: string;
2430    readonly name: string;
2431    readonly startTime: DOMHighResTimeStamp;
2432    toJSON(): any;
2433  }
2434  interface PerformanceEventTiming extends PerformanceEntry {

```

Ln 2382, Col 1 Spaces: 2 UTF-8 LF () TypeScript ⚡ Prettier ⌂ ⌂

# STANDARD APIS

- We're just getting started
  - >1000 interfaces in total
  - ~200 after filtering
- Pay-for-play
- Open source



The screenshot shows a code editor window with a dark theme. The main pane displays a file named "generate.mjs" containing a list of URLs, each preceded by a comment indicating its purpose. The list includes URLs for various Web APIs such as DOM, CSS, HTML, and others. The right side of the editor shows a vertical sidebar with a tree view of the project structure, labeled "lib.d.ts — rnx-kit". The bottom right corner of the sidebar has the number "25".

```
JS generate.mjs
1 import { ... } from 'typescript';
2
3 const compatData = "https://unpkg.com/@mdn/browser-compat-data/data.json";
4
5 // https://developer.mozilla.org/en-US/docs/Web/API
6 const ignoredSpecs = [
7   "https://dom.spec.whatwg.org/", // DOM
8   "https://drafts.css-houdini.org/", // CSS Houdini
9   "https://drafts.fxtf.org/", // W3C CSS-SVG effects
10  "https://html.spec.whatwg.org/multipage/browsing-the-web.html", // HTML
11  "https://html.spec.whatwg.org/multipage/canvas.html", // Canvas
12  "https://html.spec.whatwg.org/multipage/common-dom-interfaces.html", // DOM
13  "https://html.spec.whatwg.org/multipage/custom-elements.html", // DOM
14  "https://html.spec.whatwg.org/multipage/dom.html", // DOM
15  "https://html.spec.whatwg.org/multipage/dynamic-markup-insertion.html", // DOM
16  "https://html.spec.whatwg.org/multipage/edits.html", // DOM
17  "https://html.spec.whatwg.org/multipage/embedded-content.html", // DOM
18  "https://html.spec.whatwg.org/multipage/form-control-infrastructure.html", // DOM
19  "https://html.spec.whatwg.org/multipage/form-elements.html", // DOM
20  "https://html.spec.whatwg.org/multipage/forms.html", // DOM
21  "https://html.spec.whatwg.org/multipage/grouping-content.html", // DOM
22  "https://html.spec.whatwg.org/multipage/history.html", // DOM
23  "https://html.spec.whatwg.org/multipage/iframe-embed-object.html", // DOM
24  "https://html.spec.whatwg.org/multipage/image-maps.html", // DOM
25  "https://html.spec.whatwg.org/multipage/imagebitmap-and-animations.html", // DOM
26  "https://html.spec.whatwg.org/multipage/input.html", // DOM
27  "https://html.spec.whatwg.org/multipage/interaction.html", // DOM
28  "https://html.spec.whatwg.org/multipage/interactive-elements.html", // DOM
29  "https://html.spec.whatwg.org/multipage/media.html", // DOM
30  "https://html.spec.whatwg.org/multipage/obsolete.html", // DOM
31  "https://html.spec.whatwg.org/multipage/parsing.html", // DOM
32  "https://html.spec.whatwg.org/multipage/scripting.html", // DOM
33  "https://html.spec.whatwg.org/multipage/sections.html", // DOM
34  "https://html.spec.whatwg.org/multipage/semantics.html", // DOM
35  "https://html.spec.whatwg.org/multipage/tables.html", // DOM
36  "https://html.spec.whatwg.org/multipage/text-level-semantics.html", // DOM
37  "https://html.spec.whatwg.org/multipage/web-messaging.html", // Broadcast
38  "https://html.spec.whatwg.org/multipage/window-object.html#htmldocument", // HTMLDocument
39  "https://html.spec.whatwg.org/multipage/workers.html", // Workers
40  "https://immersive-web.github.io/", // WebXR
41  "https://registry.khronos.org/", // WebGL
42  "https://svgwg.org/", // SVG
43  "https://w3c.github.io/DOM-Parsing/", // XMLSerializer
44  "https://w3c.github.io/IntersectionObserver/", // IntersectionObserver
45  "https://w3c.github.io/ServiceWorker/", // ServiceWorker
46  "https://w3c.github.io/csswg-drafts/", // CSS
47  "https://w3c.github.io/mathml-core/", // MathML
48  "https://w3c.github.io/mediacapture-fromelement", // Canvas
49  "https://w3c.github.io/uievents/", // DOM events
50  "https://webidl.spec.whatwg.org/", // DOMException
51  "https://xhr.spec.whatwg.org/#dom-formdata", // XMLHttpRequest
52  ];
53
54 /**
55  * @typedef {{
56  *   error: typeof console.error;
57  * }} ErrorType
58  */
59
60 
```



# IN CLOSING

- Improving developer experience
  - At Microsoft, we've increased productivity with React Native and **open-source** tools
  - But it's *not* enough...
- The next level: developer velocity
  - Web/native convergence
  - User Interface → collab with Meta
  - Standard APIs → we're experimenting

**TRULY INVISIBLE**

---

# **THANK YOU**

---

## LEARN MORE

- Blog: <https://devblogs.microsoft.com/react-native/>
- Documentation: <https://aka.ms/rnxkit>
- rnx-kit: <https://github.com/microsoft/rnx-kit>
- react-native-test-app: <https://github.com/microsoft/react-native-test-app>
- RFC: React DOM for Native:  
<https://github.com/react-native-community/discussions-and-proposals/pull/496>

## Q&A

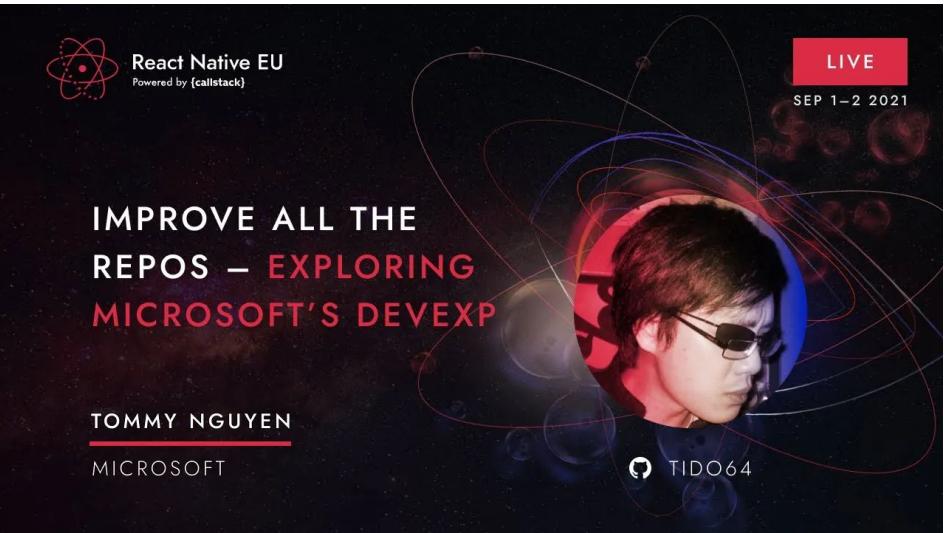
---

@TID064

---

<https://github.com/microsoft/rnx-kit>

## LEARN MORE



<https://youtu.be/DAEnPV78rQc>



<https://youtu.be/zgAjZVcvsv8>