

Building the Monorepo

“

kchau@microsoft.com

Flywheel Team

”

About Monorepos

- JS codebases have grown, and are maintained by **thousands of devs**
- Modularity is needed: **package as unit of modularity**
- Related packages are updated at the **same commit**
- Monorepo require a **Monorepo Management Stack**

Monorepo Management Stack

1. Workspace-enabled package manager

- **Installs dependencies** for all packages
- **Links internal packages** to satisfy the node resolution algorithm
- Handles **dependency resolution** for all packages
- Optional: **hoisting**, **strictness** enforcement (phantom deps)
- On the market: `yarn` , `pnpm` , `rush` , `lerna + npm`

2. Task scheduler & runner

- **Runs npm scripts** for all packages
- **Optimize** task run speeds at the dev machine and CI
- Optionally in **topological** order or in **parallel**
- On the market: `lerna` , `wsrn` , `rush` , `pnpm`
`recursive` , `lage`

3. Package publish tool

- Automated management of **semver**
 - Change description files or commit messages
- **Validation** of description of changes
- **Synchronize versions** between npm registry and git repository
- Automated **changelog creation**
- On the market: `rush`, `lerna`, `semantic-release`, `beachball`

Our Focus: Task scheduler & runner

Problem statement

Create a task runner that optimizes package tasks in a monorepo for a single machine

Current state

- JS monorepos in the wild run with **all kinds of workspaces**
- The state-of-the-art monorepo task runners are **not optimized**
- **CPU cores sit idle** for topological scripts
- Large monorepos generally have **clustered graph** of related packages

Philosophy

- Distribute work via smaller libraries with multiple owners
- Leverage OSS as much as possible
- Support package.json scripts as the script runner

Requirements of a task runner

- **Open sourced**
 - easily shared, public development demands **polish**
 - **easily contributed** to by many groups
- Works with all workspace implementations
- Easy setup
- Minimize idle CPU cores
- **Sublinear increase** in build time per package

Prior Art

- This should sound familiar because Vincent made a version for Midgard

Lage

“

v. to make (Norwegian); pr. LAH-geh

”

- Open sourced: <https://github.com/microsoft/lage>
- Easy to integrate with existing codebase
- Scales up with pipelining
- Scales out with caching and scoping

Collaboration

- **OneDrive/SharePoint:** `rush` showed us incremental builds
- **Midgard:** `backfill` cache, `task-scheduler`
- **Flywheel:** pipeline config, `workspace-tools`, `lage` tool
- **FluidX:** `p-graph` promise graph that supports priority queuing

What does it look like?

Full Build

```
$ lage build test lint --grouped --verbose --reset-cache
```

```
info codepush/react-template lint completed, took 11.10s
info codepush/react-orchestrator-template lint completed, took 12.09s
info @pinetbl/react-button build completed, took 16.87s
info @pinetbl/react-tabs build completed, took 16.79s
info @pinetbl/storybook lint completed, took 21.18s
info doc-tests build completed, took 20.15s
info @uiFabric/charting build completed, took 44.79s
info @uiFabric/date-time build completed, took 19.43s
info @uiFabric/example-app-base test completed, took 15.42s
info @uiFabric/example-app-base lint completed, took 40.42s
info @uiFabric/experiments build completed, took 16.10s
info @uiFabric/lite build completed, took 46.71s
info @uiFabric/react-cars build completed, took 11.12s
info doc-tests lint completed, took 11.09s
info @uiFabric/charting test completed, took 42.41s
info @uiFabric/charting lint completed, took 50.19s
info @pinetbl/react-button test completed, took 11.86s
info @pinetbl/react-button lint completed, took 16.12s
info @pinetbl/react-fire build completed, took 46.40s
info @pinetbl/react-tabs test completed, took 44.11s
info @pinetbl/react-tabs lint completed, took 19.27s
info @uiFabric/lite test completed, took 18.28s
info @uiFabric/lite lint completed, took 15.48s
info @uiFabric/date-time test completed, took 41.31s
info @uiFabric/date-time lint completed, took 42.84s
info @pinetbl/react-next build completed, took 16.79s
info @uiFabric/experiments test completed, took 46.40s
info @uiFabric/experiments lint completed, took 16.10s
info @pinetbl/react-fis lint completed, took 11.45s
info thesing-designer build completed, took 11.71s
info @uiFabric/api-doc build completed, took 10.23s
info @uiFabric/react-cars test completed, took 10.86s
info @uiFabric/react-docs lint completed, took 18.19s
info @uiFabric/fabric-white-resources build completed, took 28.28s
info @uiFabric/api-docs test completed, took 7.11s
info @uiFabric/api-docs lint completed, took 17.47s
info thesing-designer test completed, took 5.18s
info thesing-designer lint completed, took 16.27s
info @uiFabric/fabric-white build completed, took 20.10s
info @uiFabric/fabric-white-resources lint completed, took 14.89s
info @uiFabric/fabric-white build completed, took 16.06s
info @pinetbl/example build completed, took 15.44s
info @pinetbl/example lint completed, took 9.87s
info @uiFabric/fabric-white test completed, took 2.44s
info @uiFabric/fabric-white lint completed, took 3.81s
info @uiFabric/fabric-white lint completed, took 16.11s
info @uiFabric/fabric-white test completed, took 1.11s
info @pinetbl/docs lint completed, took 18.44s
info @pinetbl/pure lint completed, took 1.44s
info codepush/react-next-template build completed, took 12.61s
info @uiFabric/perf build completed, took 22.17s
info @uiFabric/perf test completed, took 11.18s
info @uiFabric/perf lint completed, took 40.40s
info @pinetbl/react-next test completed, took 16.42s
info @pinetbl/react-next lint completed, took 16.11s
info codepush/react-next-template lint completed, took 18.77s
info test-humans test completed, took 4.41s
info test-humans lint completed, took 19.71s
info @uiFabric/perf test completed, took 4.18s
info @uiFabric/perf lint completed, took 21.81s
info @uiFabric/pr-deploy-lite build completed, took 7.78s
info @uiFabric/pr-deploy-lite test completed, took 4.12s
info @uiFabric/pr-deploy-lite lint completed, took 2.12s
info
info Took a total of 104 12.79s to complete
Done in 143.04s.
/home/USER-00013101/.workspace/pinetbl
```


Cached Build

```
$ lage build test lint --grouped --verbose
```

```

1040 # start #fintest/react:theme-provider lint
1041 # start #fintest/react:theme-provider lint - 235a18b9cd2f6e48b0c19024c846c292
1042 # start #fintest/react:image-build
1043 # start #fintest/react:image-build - 1a8af68e0941c0c759231302241c184802
1044 # start #fintest/react:font-test
1045 # start #fintest/react:font-test - 5e3b57a823854d6d747afad4564a955b407
1046 # start #fintest/react:font-test lint
1047 # start #fintest/react:font-test lint - 5e677a8a19a7670b1163220b8efad26722f4c
1048 # start #fintest/react:bindings-test
1049 # start #fintest/react:bindings-test - 6730512001289774272842b76845749840
1050 # start #fintest/react:multibyte-build
1051 # start #fintest/react:multibyte-build - 822a7891201723777b748824860e01ef5af0508
1052 # start #fintest/react:volatility-build
1053 # start #fintest/react:volatility-build - ba1d092a265181b243c7f318c7d9ba78102b6b8
1054 # start #fintest/react:bindings lint
1055 # start #fintest/react:bindings lint - 23781a684b73ef3a5c1d51028015e4d43141
1056 # start #fintest/react:image lint
1057 # start #fintest/react:image lint - 43646664b70e43117d122e2037777ef49e40b18
1058 # start #fintest/react:image-test
1059 # start #fintest/react:image-test - 0a4f51fe788d77609153a103277e731fa5dadaf
1060 # start #fintest/react:volatility lint
1061 # start #fintest/react:volatility lint - c2e2ae387cf495a12d2058c0eb510e3a35d
1062 # start #fintest/react:multibyte lint
1063 # start #fintest/react:multibyte lint - 4790831e0462461d13299a17e4a2177cf87e3af079
1064 # start #fintest/react:multibyte lint
1065 # start #fintest/react:multibyte lint - 822a2ab9ef4970901684b81d41693baf7a1018
1066 # start #fintest/react:font-test lint
1067 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1068 # start #fintest/react:font-test lint
1069 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1070 # start #fintest/react:font-test lint
1071 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1072 # start #fintest/react:font-test lint
1073 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1074 # start #fintest/react:font-test lint
1075 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1076 # start #fintest/react:font-test lint
1077 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1078 # start #fintest/react:font-test lint
1079 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1080 # start #fintest/react:font-test lint
1081 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1082 # start #fintest/react:font-test lint
1083 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1084 # start #fintest/react:font-test lint
1085 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1086 # start #fintest/react:font-test lint
1087 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1088 # start #fintest/react:font-test lint
1089 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1090 # start #fintest/react:font-test lint
1091 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1092 # start #fintest/react:font-test lint
1093 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1094 # start #fintest/react:font-test lint
1095 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1096 # start #fintest/react:font-test lint
1097 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1098 # start #fintest/react:font-test lint
1099 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1100 # start #fintest/react:font-test lint
1101 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1102 # start #fintest/react:font-test lint
1103 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1104 # start #fintest/react:font-test lint
1105 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1106 # start #fintest/react:font-test lint
1107 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1108 # start #fintest/react:font-test lint
1109 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1110 # start #fintest/react:font-test lint
1111 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1112 # start #fintest/react:font-test lint
1113 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1114 # start #fintest/react:font-test lint
1115 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1116 # start #fintest/react:font-test lint
1117 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1118 # start #fintest/react:font-test lint
1119 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1120 # start #fintest/react:font-test lint
1121 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1122 # start #fintest/react:font-test lint
1123 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1124 # start #fintest/react:font-test lint
1125 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1126 # start #fintest/react:font-test lint
1127 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1128 # start #fintest/react:font-test lint
1129 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1130 # start #fintest/react:font-test lint
1131 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1132 # start #fintest/react:font-test lint
1133 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1134 # start #fintest/react:font-test lint
1135 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1136 # start #fintest/react:font-test lint
1137 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1138 # start #fintest/react:font-test lint
1139 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1140 # start #fintest/react:font-test lint
1141 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1142 # start #fintest/react:font-test lint
1143 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1144 # start #fintest/react:font-test lint
1145 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1146 # start #fintest/react:font-test lint
1147 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1148 # start #fintest/react:font-test lint
1149 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1150 # start #fintest/react:font-test lint
1151 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1152 # start #fintest/react:font-test lint
1153 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1154 # start #fintest/react:font-test lint
1155 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1156 # start #fintest/react:font-test lint
1157 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1158 # start #fintest/react:font-test lint
1159 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff71e0d
1160 # start #fintest/react:font-test lint
1161 # start #fintest/react:font-test lint - 5a3f6e6a1c1465280790128c1bcbff7
```

Scoped Build

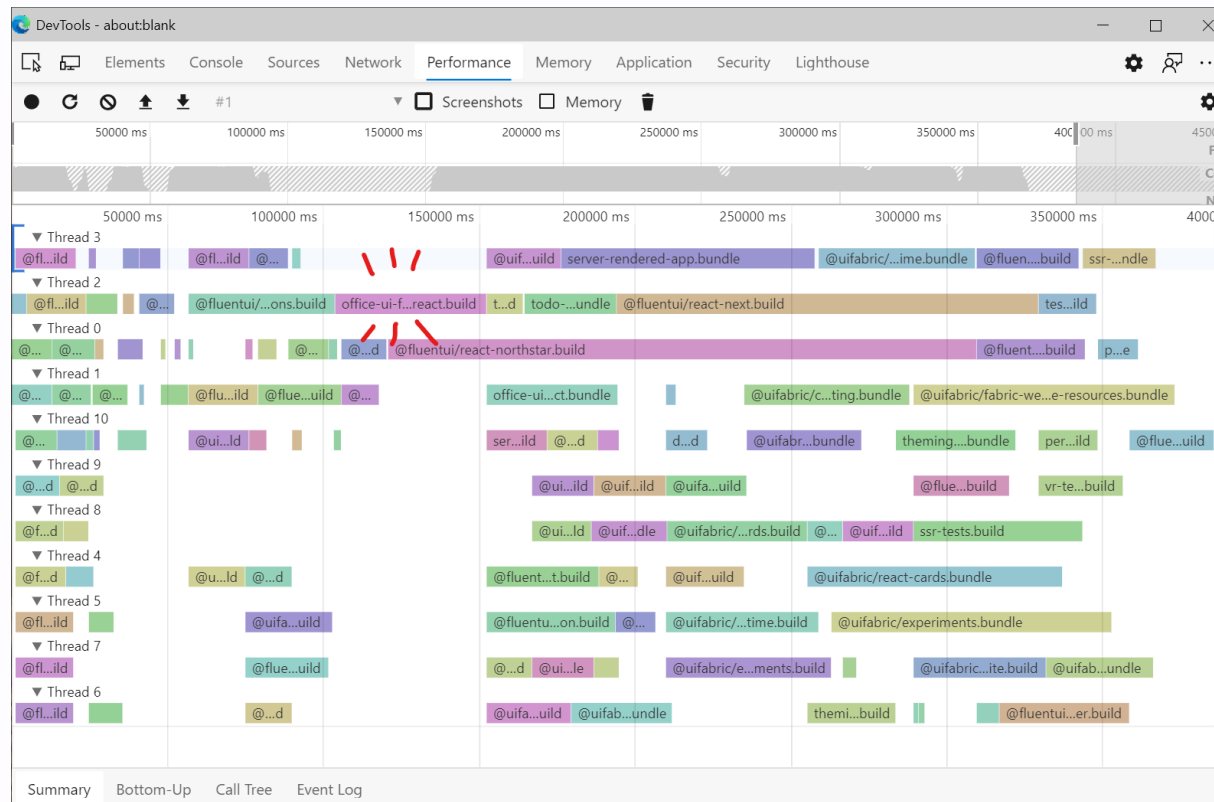
```
$ lage build test lint --grouped --verbose --scope @fluentui/web-components
```

```
~/workspace/fluentui$ npm run build test lint --verbose --no-cache --scope @fluentui/web-components
npm run v1.22.4
$ lage build test lint --verbose --no-cache --scope @fluentui/web-components
info lage test runner - let's make it
verb @fluentui/web-components build start
verb @fluentui/web-components build Running /home/ken/.npm/_nodedir/node/v12.18.1/bin/node run build --
verb @fluentui/web-components build with --script-path The node binary used for scripts is /home/ken/.npm/_nodedir/node/v12.18.1/bin/node itself. Use the --script-path node-path option to include a
verb @fluentui/web-components build path for the node binary you are executing with.
verb @fluentui/web-components build - @fluentui/web-components@0.1.2 build /home/ken/workspace/fluentui/packages/web-components
verb @fluentui/web-components build - & test -p ./tsconfig.json && rollup -c && npm run doc
verb @fluentui/web-components build - npm/index-rollup.js - dist/web-components.js, dist/web-components.doc.js...
```



Profiling

```
$ lage build test lint --profile
```



How does it work?

<https://microsoft.github.io/lage/guide/levels.html>

How to try it at home?

<https://microsoft.github.io/lage/guide/getting-started.html>

1. npm scripts (build, test, lint) are at package level
2. `npx lage init`
 - creates a `lage.config.js` - configure it
 - adds `lage` as a dep
3. `yarn lage build` or `npm run lage build`

Future

- `lage` is a great solution for a **single machine**, not distributed builds
- Microsoft solution is `buildxl`
 - `lage` spits out info for `buildxl` now!

```
lage info build --reporter json
```

- Another popular distributed build solution: `bazel`
 - `lage` can potentially spit out WORKSPACE & BUILD

More info

Github:

<https://github.com/microsoft/lage>

Documentation:

<https://microsoft.github.io/lage/>

Complex Configuration:

<https://github.com/microsoft/fluentui/blob/master/lage.config.js>