



The State of Backstage In 2023

Ben Lambert & Patrik Oldsberg





Ben Lambert

 benjdlambert

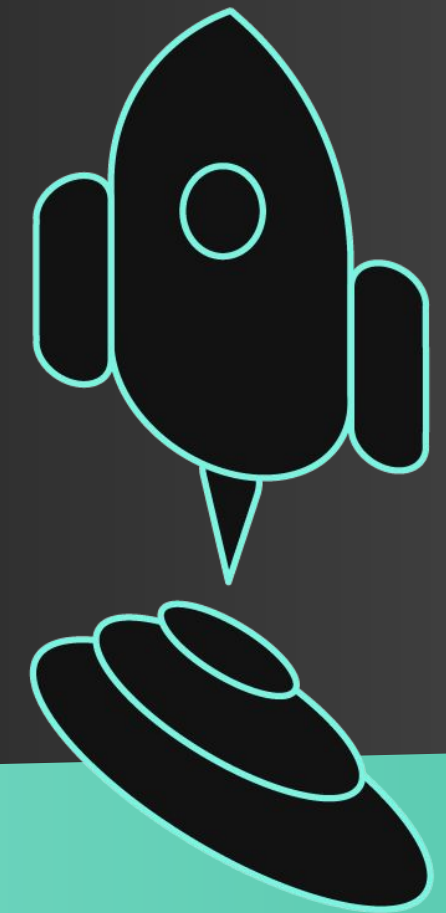
Engineer @ Spotify
Core Maintainer | Backstage



Patrik Oldsberg

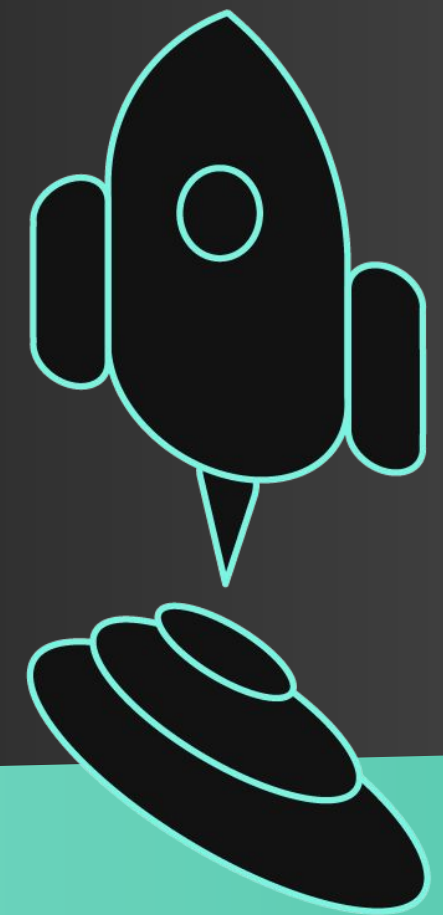
 Rugvip

Senior Engineer @ Spotify
Core Maintainer | Backstage

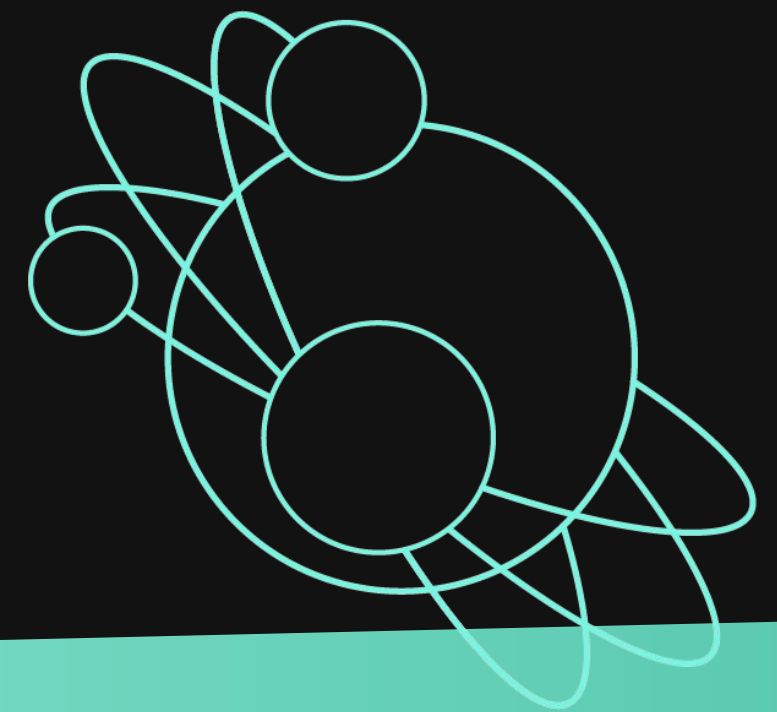


Agenda

- **Project Updates**
- **What's new in Software Templates?**
- **New Backend System**
- **Roadmap**



Project Update





1k+

Adopters

↑ 900%

1k+

Contributors

↑ 230%

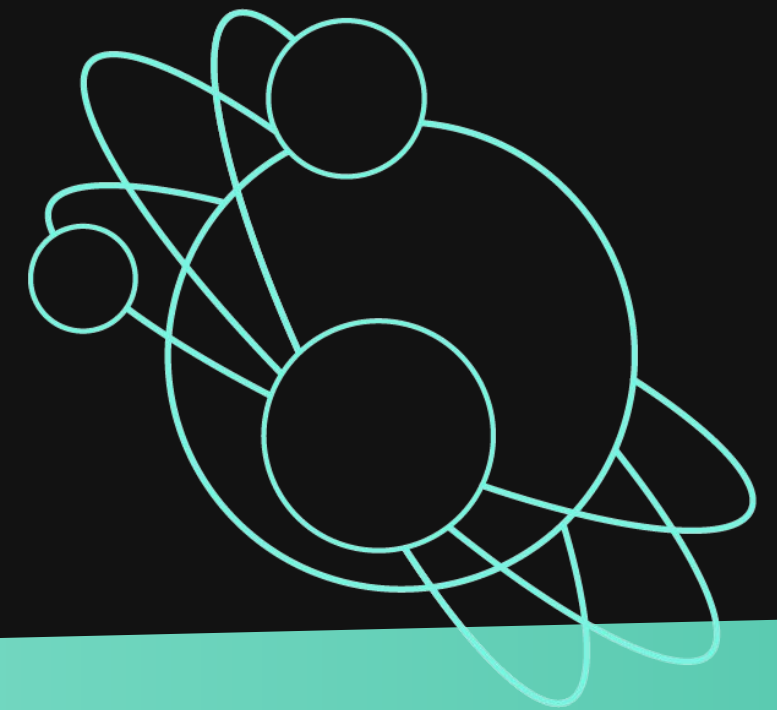
21.5k

Stars

↑ 45%

Project Governance Updates

Project Areas & Contributor Ladder

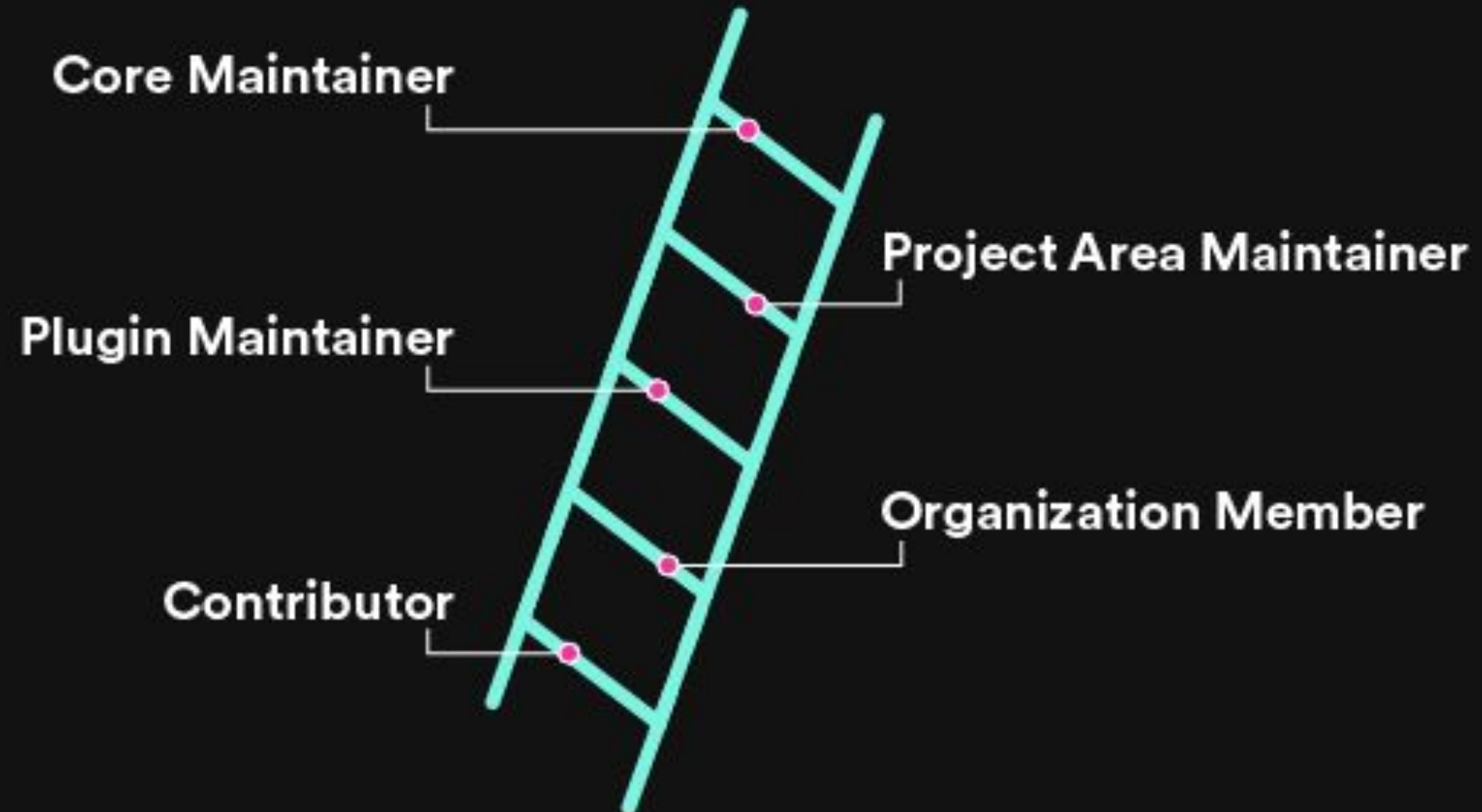


New Governance

- Project Areas
- Contributor Ladder
- Based on CNCF Contributor Ladder Governance Template
 - TAG Contributor Strategy
 - <https://contribute.cncf.io/maintainers/templates/>

<https://github.com/backstage/backstage/blob/master/GOVERNANCE.md>

Contributor Ladder



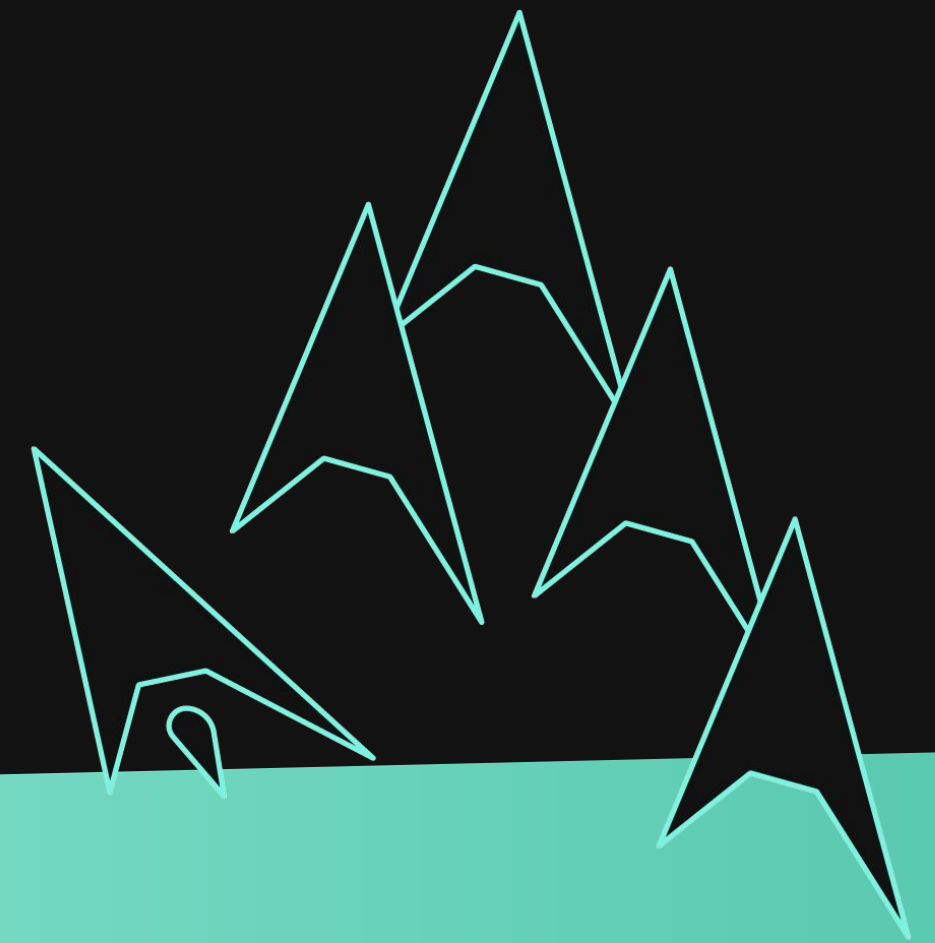
Project Areas

- Catalog
- Discoverability
- TechDocs
- Helm Charts
 - Andrew - RedHat
 - Tom - RedHat
 - Vincenzo - Spotify
- Kubernetes
 - Jamie - VMWare
 - Matthew - Spotify

Moving Forward

- New Project Areas
 - Permissions
 - Software Templates
 - ???
- Become an organization member!
 - <https://github.com/backstage/community/issues/new/choose>

What's new in Software Templates?



Why Software Templates?

- Creation of new Software Components
- Every company is different
- Best practices built in

Create a new component | Template x +

localhost:3000/create?filters%5Bkind%5D=template&filters%5Buser%5D=all

Sprint Board PR Board Blam's PR's Assigned Reviews for Me Catalog Other Bookmarks

Backstage

Search

Home APIs Docs Create... Tech Radar Settings

Create a new component

Create new software components using standard templates in your organization

REGISTER EXISTING COMPONENT SUPPORT

Available Templates

Search X

PERSONAL

★ Starred 0

MY COMPANY

All 1

CATEGORIES

service

Example Node.js Template

An example template for the scaffolder that creates a simple Node.js service

user:guest CHOOSE

▼

react-ssr-template

>

skeleton

template.yaml

| Name | Last commit message | Last commit date |
|--------------------|---|------------------|
| .. | | |
| docs | Move scaffolder templates into their on directory | 2 years ago |
| public/static | Move scaffolder templates into their on directory | 2 years ago |
| src | Move scaffolder templates into their on directory | 2 years ago |
| .editorconfig | Move scaffolder templates into their on directory | 2 years ago |
| .eslintignore | Move scaffolder templates into their on directory | 2 years ago |
| .eslintrc.js | Move scaffolder templates into their on directory | 2 years ago |
| .gitignore | Move scaffolder templates into their on directory | 2 years ago |
| README.md | Move scaffolder templates into their on directory | 2 years ago |
| babel.config.js | Move scaffolder templates into their on directory | 2 years ago |
| catalog-info.yaml | scaffolder-templates: strip whitespace in catalog-info des... | last year |
| jest.config.js | Move scaffolder templates into their on directory | 2 years ago |
| mkdocs.yml | Move scaffolder templates into their on directory | 2 years ago |
| next-env.d.ts | Move scaffolder templates into their on directory | 2 years ago |
| next.config.js | Move scaffolder templates into their on directory | 2 years ago |
| package.json | Move scaffolder templates into their on directory | 2 years ago |
| prettier.config.js | Move scaffolder templates into their on directory | 2 years ago |
| tsconfig.json | Move scaffolder templates into their on directory | 2 years ago |

Parameter Schema

```
# template.yaml
parameters:
- title: Basic information
  required:
  - name
properties:
  name:
    title: Name
    type: string
    description: Unique name of the component
    ui:field: EntityNamePicker
  description:
    title: Description
    type: string
    description: A simple description of the example component
```

The screenshot shows a web browser window with the URL `localhost:3000/create/templates/default/example-nodejs-template`. The page title is "Create a new component" with the subtitle "Create new software components using standard templates in your organization". Below this is a section titled "Example Node.js Template" with the description "An example template for the scaffolder that creates a simple Node.js service". A progress bar at the top of the form shows three steps: "1 Basic information" (active), "2 Choose a location", and "3 Review". The form contains two text input fields: "Name*" with the placeholder text "Unique name of the component" and "Description" with the placeholder text "A simple description of the example component". At the bottom right of the form are two buttons: "BACK" and "NEXT".

Step definitions

```
# template.yaml
steps:
  # Each step executes an action,
  # in this case one templates files into the working directory.
  - id: fetch-base
    name: Fetch Boilerplate and Template
    action: fetch:template
    input:
      url: ./content
      values: ${ parameters }

  # This step publishes the contents of the working directory to GitHub.
  - id: publish
    name: Publish
    action: publish:github
    input:
      allowedHosts: ['github.com']
      description: This is ${ parameters.name }
      repoUrl: ${ parameters.repoUrl }

  # The final step is to register our new component in the catalog.
  - id: register
    name: Register
    action: catalog:register
    input:
      repoContentsUrl: ${ steps['publish'].output.repoContentsUrl }
      catalogInfoPath: '/catalog-info.yaml'
```

Run of example-nodejs-template

Task 97303f7f-0f23-4f00-ba2f-d220d17ba193

Fetch Boilerplate and Template
1 second

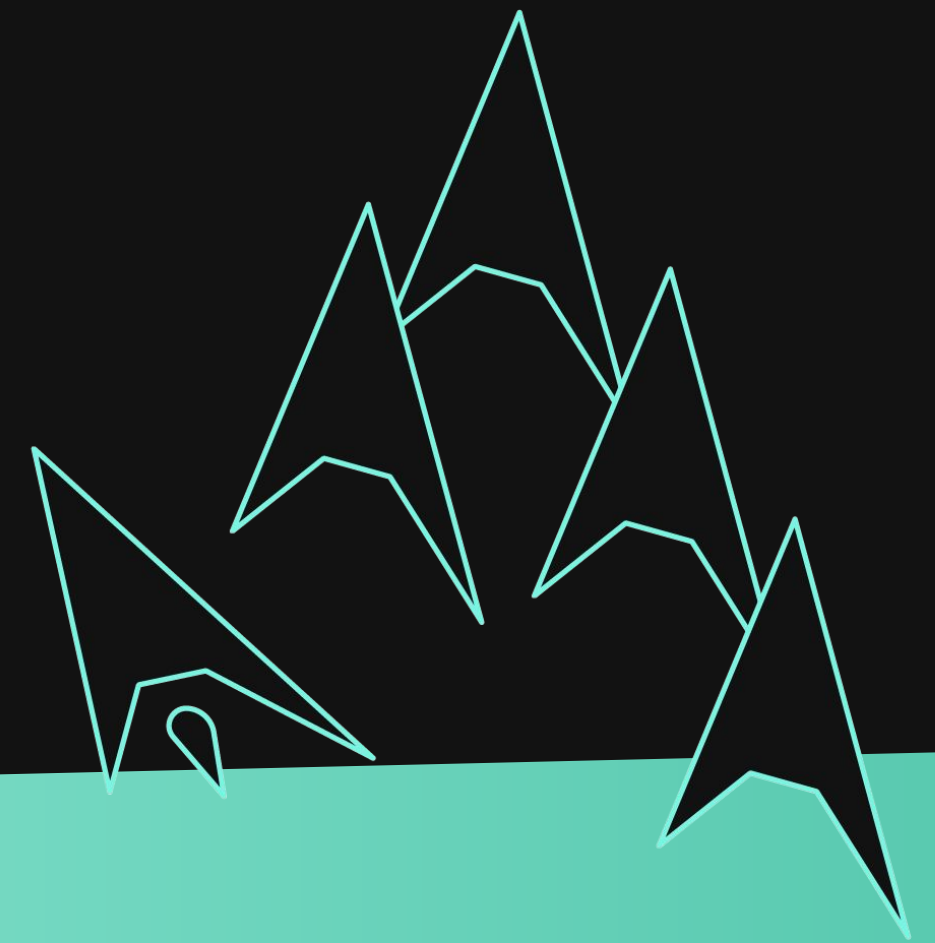
Publish
4 seconds

Register

[REPOSITORY](#) [OPEN IN CATALOG](#)

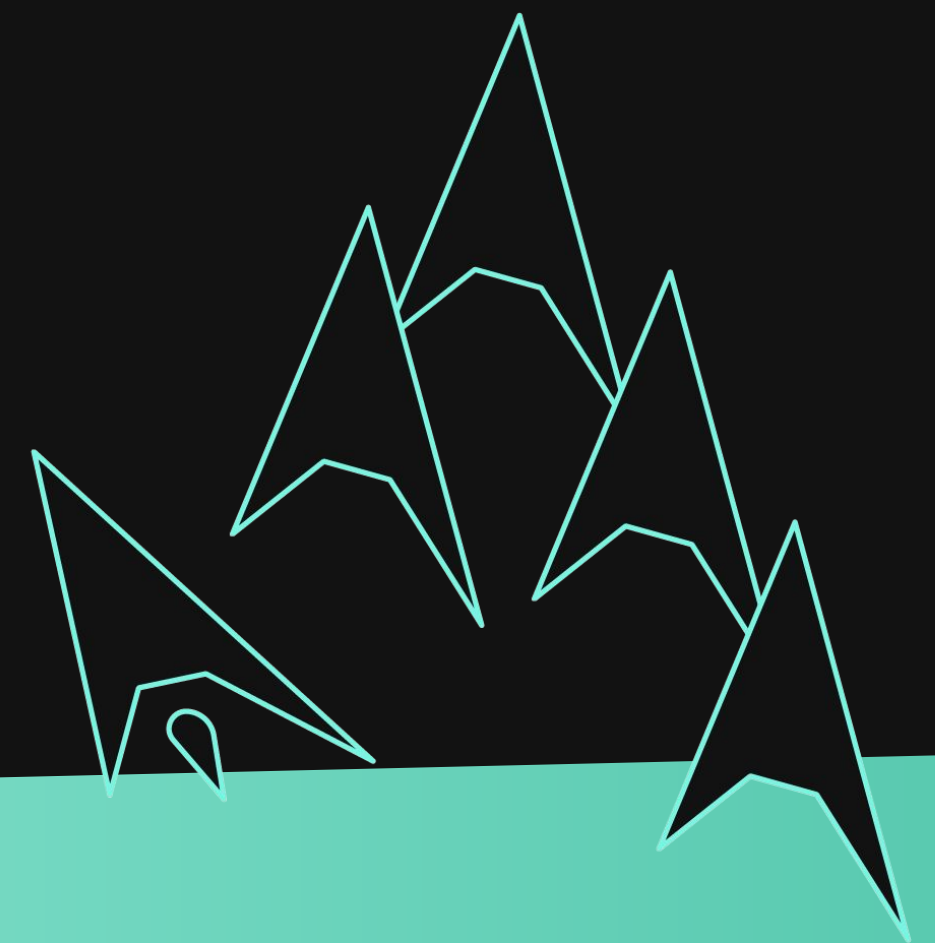
What can you do with them?

- Write your own blueprints for templating
- Create your own Field Extensions
- Create your own Actions

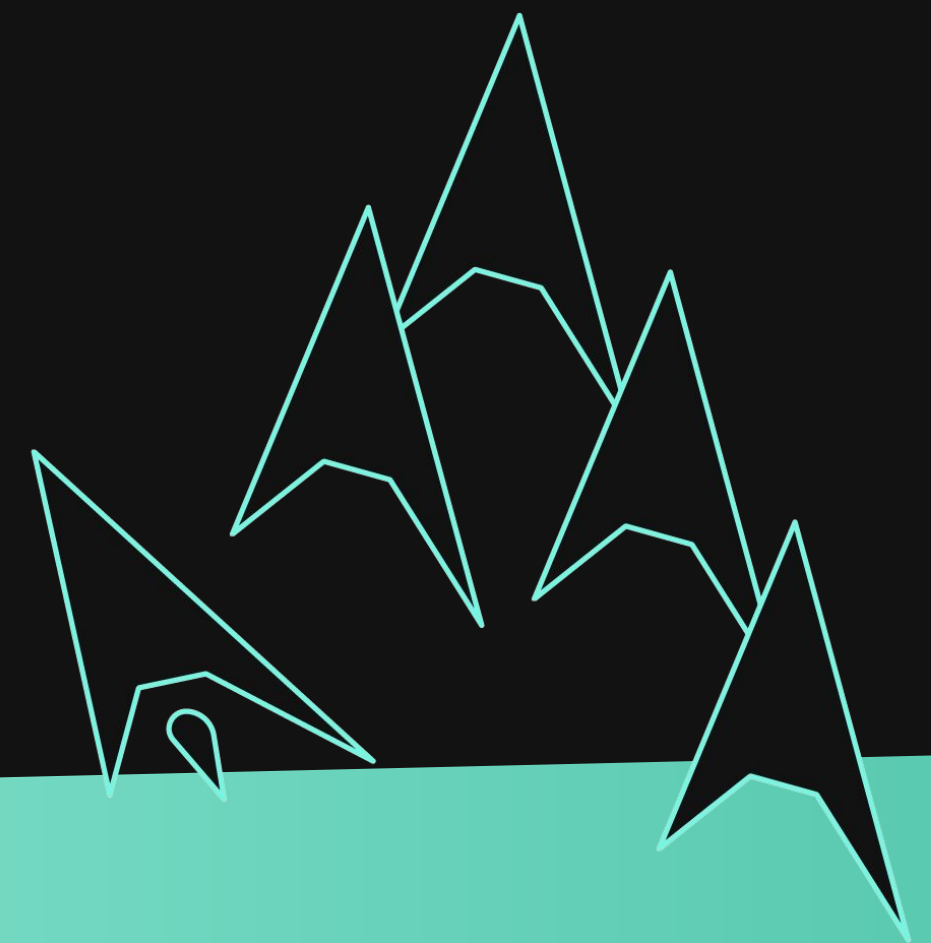


What's new?

- New UI ✨ 🚀
- New react-jsonschema-form
- Async validation for CustomFieldExtensions
- Embedded Workflows

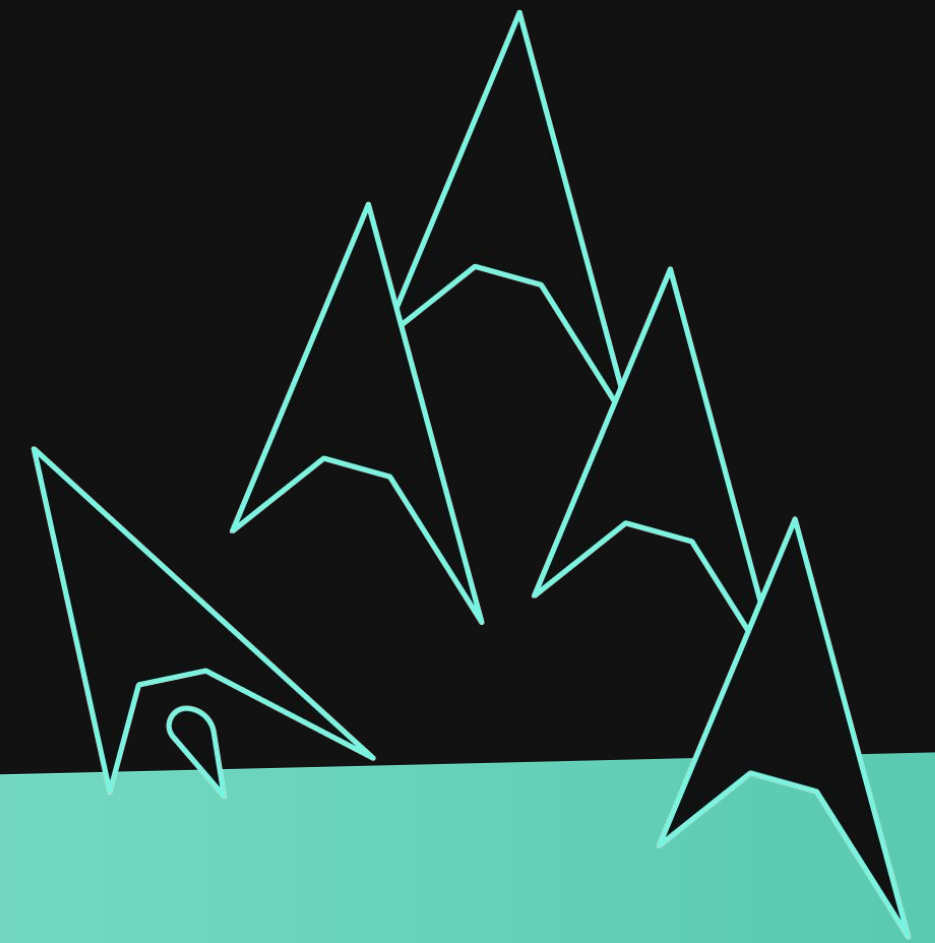


Demo

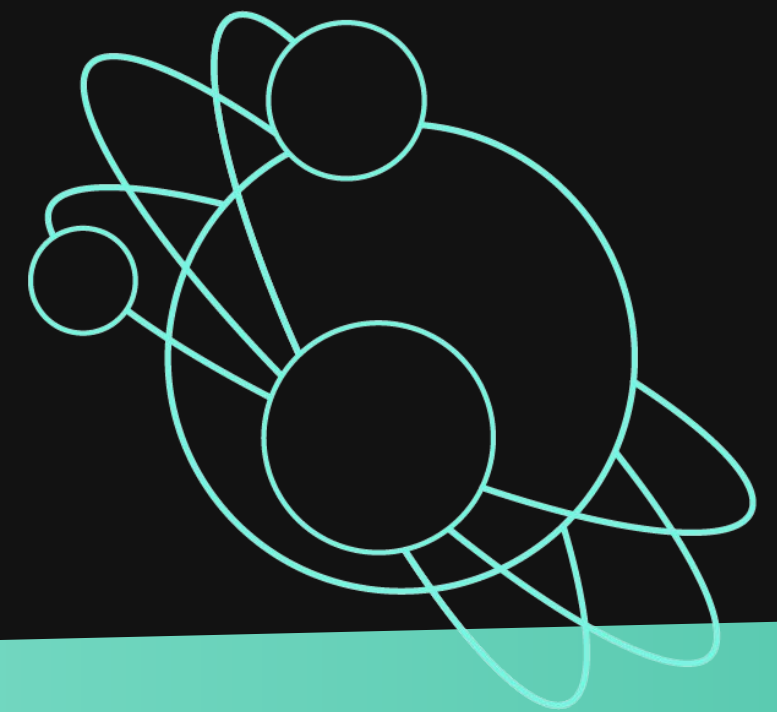


How do I try it out?

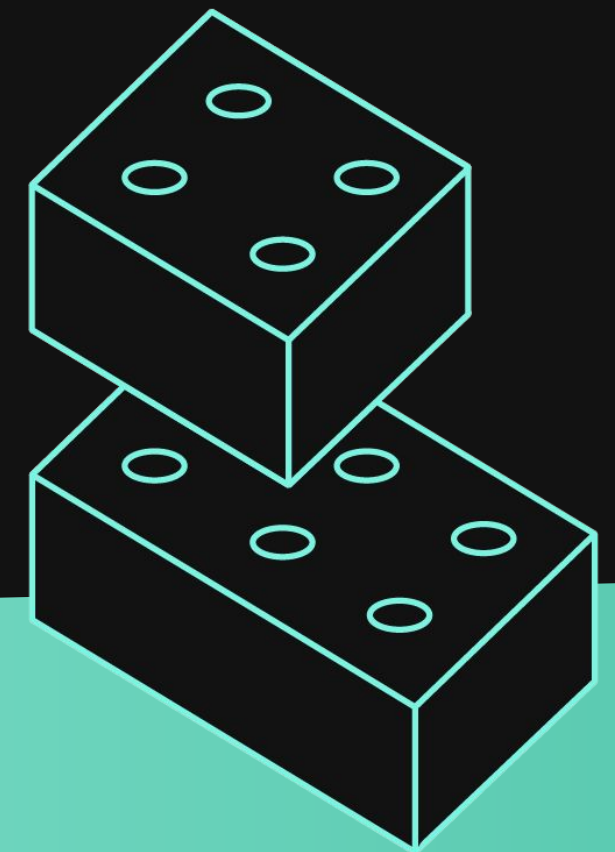
<https://backstage.io/docs/features/software-templates/testing-scaffolder-alpha>



FRAMEWORK UPDATES



Framework Updates



New Backend System

- Presented at BackstageCon/KubeCon NA 2022
 - <https://www.youtube.com/watch?v=ZXF7uxrEBVY>
- Released as alpha in 1.11, February 2023
 - <https://backstage.io/docs/backend-system/>

TL;DW

- Existing backend system grew organically
- Goals for the new system
 - Simpler installation and integration with less code
 - Easy to evolve and extend, without breaking changes
 - Recognizable, encoding existing patterns

▼ 1 example-project/packages/backend/package.json

| | | |
|----|-------------------|---|
| ↑ | @@ -34,6 +34,7 @@ | |
| 34 | 34 | "@backstage/plugin-search-backend-module-pg": "^0.4.0", |
| 35 | 35 | "@backstage/plugin-search-backend-node": "^1.0.2", |
| 36 | 36 | "@backstage/plugin-techdocs-backend": "^1.3.0", |
| 37 | + | "@kubernetes/backstage-client-service": "^1.33.7", |
| 37 | 38 | "better-sqlite3": "^7.5.0", |
| 38 | 39 | "dockerode": "^3.3.1", |
| 39 | 40 | "express": "^4.17.1", |
| ↓ | | |

1 example-project/packages/backend/package.json

↑

@@ -34,6 +34,7 @@

34 34 "@backstage/plugin-search-backend-module-pg": "^0.4.0",

35 35 "@backstage/plugin-search-backend-node": "^1.0.2",

36 36 "@backstage/plugin-techdocs-backend": "^1.3.0",

2 example-project/packages/backend/src/types.ts

↑

@@ -10,6 +10,7 @@ import {

10 10 import { PluginTaskScheduler } from '@backstage/backend-tasks';

11 11 import { PermissionEvaluator } from '@backstage/plugin-permission-common';

12 12 import { IdentityApi } from '@backstage/plugin-auth-node';

13 + import { KubernetesClient } from '@kubernetes/backstage-client-service';

13 14

14 15 export type PluginEnvironment = {

15 16 logger: Logger;

↕

@@ -22,4 +23,5 @@ export type PluginEnvironment = {

22 23 scheduler: PluginTaskScheduler;

23 24 permissions: PermissionEvaluator;

24 25 identity: IdentityApi;

26 + kubernetesClient: KubernetesClient;

25 27 };

example-project/packages/backend/package.json

```
@@ -34,6 +34,7 @@
34 34 "@backstage/plugin-search-backend-module-pg": "^0.4.0",
35 35 "@backstage/plugin-search-backend-node": "^1.0.2",
36 36 "@backstage/plugin-techdocs-backend": "^1.3.0",
```

example-project/packages/backend/src/types.ts

```
@@ -10,6 +10,7 @@ import {
10 10 import { PluginTaskScheduler } from '@backstage/backend-tasks';
11 11 import { PermissionEvaluator } from '@backstage/plugin-permission-common';
12 12 import { IdentityApi } from '@backstage/plugin-auth-node';
13 + import { KubernetesClient } from '@kubernetes/backstage-client-service';
13 14
14 15 export type PluginEnvironment = {
15 16   logger: Logger;
@@ -22,4 +23,5 @@ export type PluginEnvironment = {
22 23   scheduler: PluginTaskScheduler;
23 24   permissions: PermissionEvaluator;
24 25   identity: IdentityApi;
26 +   kubernetesClient: KubernetesClient;
25 27 };
```

example-project/packages/backend/src/index.ts

```
@@ -31,6 +31,7 @@ import search from './plugins/search';
31 31 import { PluginEnvironment } from './types';
32 32 import { ServerPermissionClient } from '@backstage/plugin-permission-node';
33 33 import { DefaultIdentityClient } from '@backstage/plugin-auth-node';
34 + import { KubernetesClientManager } from '@kubernetes/backstage-client-service';
34 35
35 36 function makeCreateEnv(config: Config) {
36 37   const root = getRootLogger();
@@ -40,6 +41,7 @@ function makeCreateEnv(config: Config) {
40 41   const databaseManager = DatabaseManager.fromConfig(config, { logger: root });
41 42   const tokenManager = ServerTokenManager.noop();
42 43   const taskScheduler = TaskScheduler.fromConfig(config);
44 +   const kubernetesClientManager = KubernetesClientManager.fromConfig(config);
43 45
44 46   const identity = DefaultIdentityClient.create({
45 47     discovery,
@@ -56,6 +58,7 @@ function makeCreateEnv(config: Config) {
56 58   const database = databaseManager.forPlugin(plugin);
57 59   const cache = cacheManager.forPlugin(plugin);
58 60   const scheduler = taskScheduler.forPlugin(plugin);
61 +   const kubernetesClient = kubernetesClientManager.forPlugin(plugin);
59 62   return {
60 63     logger,
61 64     database,
@@ -67,6 +70,7 @@ function makeCreateEnv(config: Config) {
67 70     scheduler,
68 71     permissions,
69 72     identity,
73 +     kubernetesClient,
70 74   };
71 75 };
72 76 }
```


example-project/packages/backend/package.json

@@ -34,6 +34,7 @@

34 34 "@backstage/plugin-search-backend-module-pg": "^0.4.0",

35 35 "@backstage/plugin-search-backend-node": "^1.0.2",

36 36 "@backstage/plugin-techdocs-backend": "^1.3.0",

example-project/packages/backend/src/types.ts

@@ -10,6 +10,7 @@ import {

10 10 import { PluginTaskScheduler } from '@backstage/backend-tasks';

11 11 import { PermissionEvaluator } from '@backstage/plugin-permission-common';

12 12 import { IdentityApi } from '@backstage/plugin-auth-node';

13 + import { KubernetesClient } from '@kubernetes/backstage-client-service';

13 14

14 15 export type PluginEnvironment = {

15 16 logger: Logger;

@@ -22,4 +23,5 @@ export type PluginEnvironment = {

22 23 scheduler: PluginTaskScheduler;

23 24 per

24 25 ide

26 + kut

25 27 };

example-project/packages/backend/src/plugins/scaffolder.ts

@@ -17,5 +17,6 @@ export default async function createPlugin({

17 17 reader: env.reader,

18 18 catalogClient,

19 19 identity: env.identity,

20 + kubernetesClient: env.kubernetesClient,

20 21 });

21 22 }

example-project/packages/backend/src/index.ts

@@ -31,6 +31,7 @@ import search from './plugins/search';

31 31 import { PluginEnvironment } from './types';

32 32 import { ServerPermissionClient } from '@backstage/plugin-permission-node';

33 33 import { DefaultIdentityClient } from '@backstage/plugin-auth-node';

34 + import { KubernetesClientManager } from '@kubernetes/backstage-client-service';

34 35

35 36 function makeCreateEnv(config: Config) {

36 37 const root = getRootLogger();

@@ -40,6 +41,7 @@ function makeCreateEnv(config: Config) {

40 41 const databaseManager = DatabaseManager.fromConfig(config, { logger: root });

41 42 const tokenManager = ServerTokenManager.noop();

42 43 const taskScheduler = TaskScheduler.fromConfig(config);

44 + const kubernetesClientManager = KubernetesClientManager.fromConfig(config);

43 45

44 46 const identity = DefaultIdentityClient.create({

45 47 discovery,

@@ -56,6 +58,7 @@ function makeCreateEnv(config: Config) {

56 58 const database = databaseManager.forPlugin(plugin);

57 59 const cache = cacheManager.forPlugin(plugin);

58 60 const scheduler = taskScheduler.forPlugin(plugin);

61 + const kubernetesClient = kubernetesClientManager.forPlugin(plugin);

62 + return {

63 + logger,

64 + database,

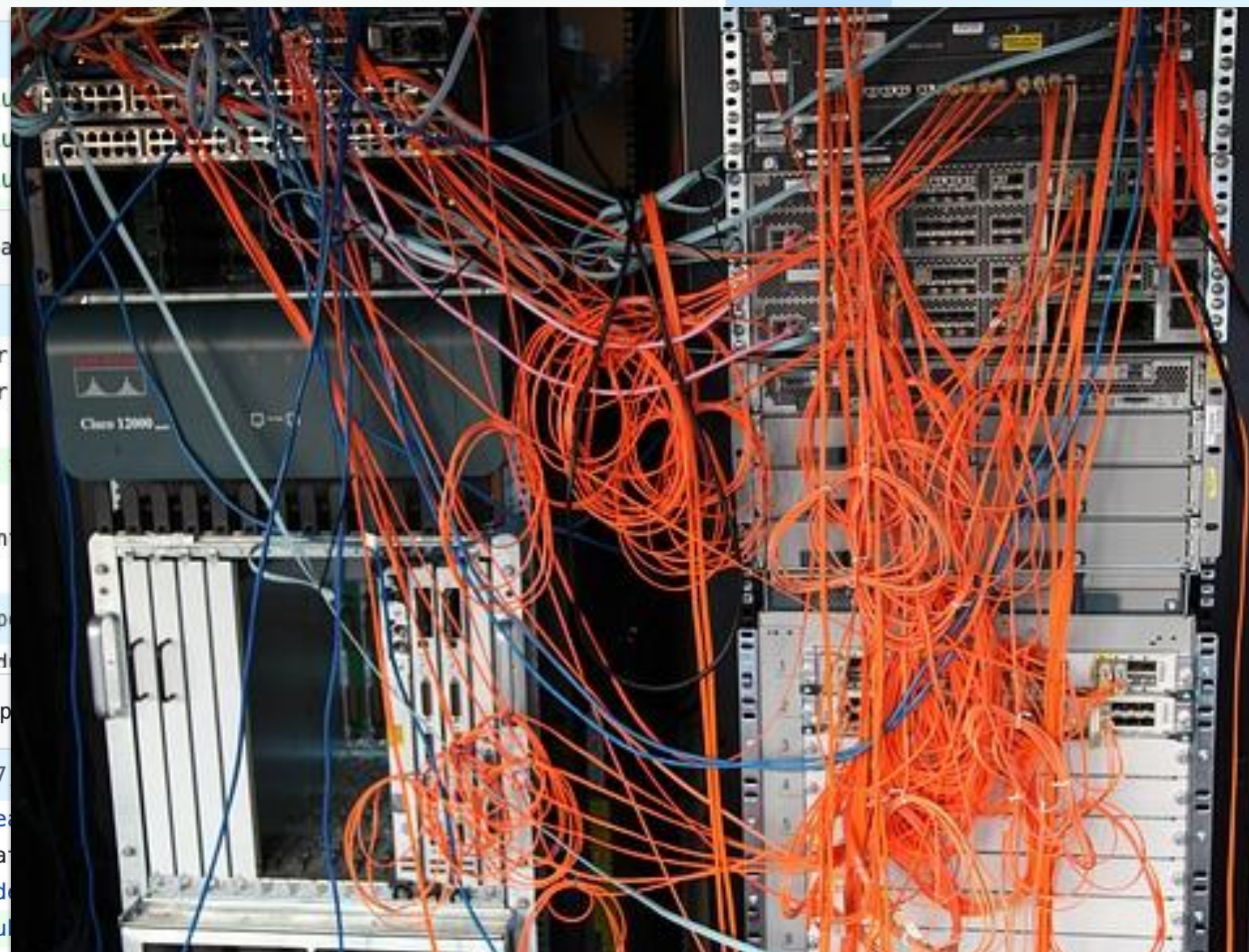
65 + scheduler,

66 + permissions,

67 + identity,

68 + kubernetesClient,

72 76 }



example-project/packages/backend/package.json

| | | @@ -34,6 +34,7 @@ |
|----|----|-------------------|
| 34 | 34 | "@backstage/plu |
| 35 | 35 | "@backstage/plu |
| 36 | 36 | "@backstage/plu |

example-project/packages/ba

| | | @@ -10,6 +10,7 @@ import { |
|----|----|-------------------------------|
| 10 | 10 | import { PluginTaskScheduler |
| 11 | 11 | import { PermissionEvaluator |
| 12 | 12 | import { IdentityApi } from |
| 13 | 13 | + import { KubernetesClient } |
| 13 | 14 | |
| 14 | 15 | export type PluginEnvironmen |
| 15 | 16 | logger: Logger; |

@@ -22,4 +23,5 @@ export typ

| | | |
|----|----|----------------------------|
| 22 | 23 | scheduler: PluginTaskSched |
| 23 | 24 | per |
| 24 | 25 | ide |
| 26 | 26 | + kut |
| 25 | 27 | }); |

example

@@ -17

| | | |
|----|----|------|
| 17 | 17 | re |
| 18 | 18 | ca |
| 19 | 19 | id |
| 20 | 20 | + ku |
| 20 | 21 | }); |
| 21 | 22 | } |

example-project/packages/backend/src/index.ts

@@ -31,6 +31,7 @@ import search from './plugins/search';

```
from './types';
ent } from '@backstage/plugin-permission-node';
nt } from '@backstage/plugin-auth-node';
ager } from '@kubernetes/backstage-client-service';
```

```
g: Config) {
  ();
  takeCreateEnv(config: Config) {
    tabaseManager.fromConfig(config, { logger: root });
    rTokenManager.noop();
    Scheduler.fromConfig(config);
    ger = KubernetesClientManager.fromConfig(config);
```

```
entityClient.create({
```

```
takeCreateEnv(config: Config) {
  eManager.forPlugin(plugin);
  er.forPlugin(plugin);
  heduler.forPlugin(plugin);
  kubernetesClientManager.forPlugin(plugin);
```

```
takeCreateEnv(config: Config) {
```

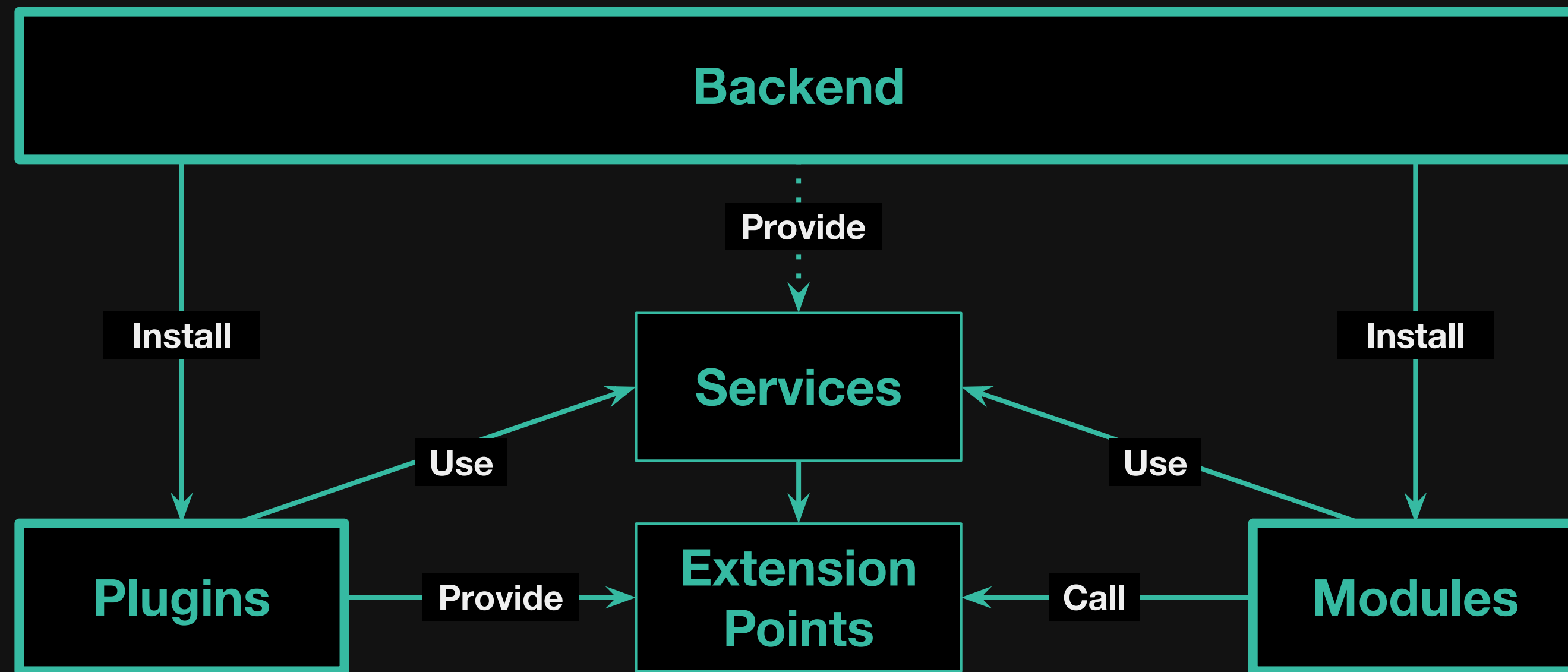
72 76 }

New System



```
const kubernetesClientServiceRef = createServiceRef<KubernetesService>({  
  id: 'kubernetes.client',  
  defaultFactory: ...,  
})
```

Architecture



Example Backend



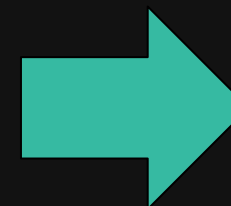
```
const backend = createBackend( )  
  
backend.add(scaffolderPlugin( ));  
backend.add(scaffolderModuleGitlabActions( ));  
backend.add(scaffolderModuleCookiecutterActions( ));  
  
await backend.start( )
```

Lines of code for a typical backend setup



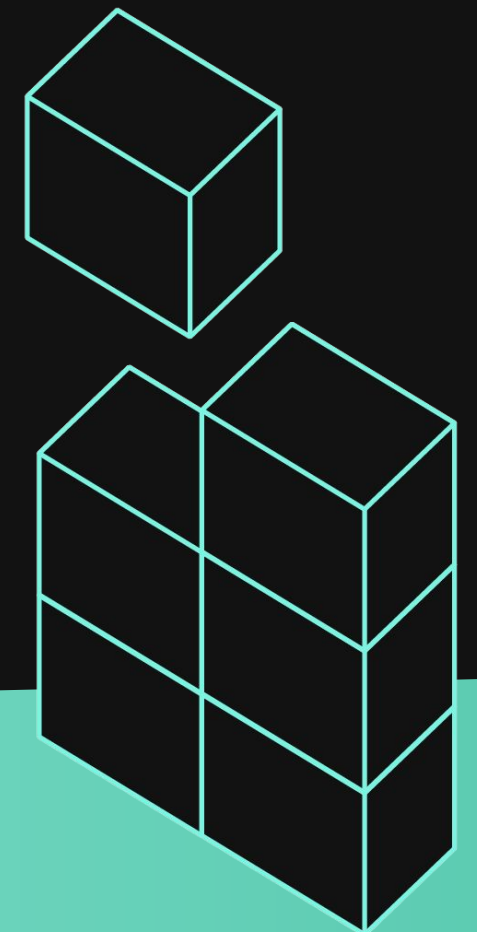
We are shifting focus to a new goal...

26



0

Declarative Integration



Declarative Integration

- Top of the Roadmap
- Scope
 - Plugin installation without TypeScript
 - Frontend and backend
- Still early work

Thank you!

Docs, demos, and updates

backstage.io

Adopting Backstage?

backstage.spotify.com

frontside.com/backstage

roadie.io

thoughtworks.com

