

---

# Open edX Developer Documentation

*Release 1.0.0*

**edX Inc.**

**Aug 25, 2020**



---

## Contents

---

<b>1</b>	<b>Open edX Developer’s Guide</b>	<b>3</b>
1.1	Micro-frontend applications in Open edX . . . . .	3
<b>2</b>	<b>Open edX Named Releases</b>	<b>7</b>
2.1	Latest Open edX Release . . . . .	7
2.2	All Open edX Releases . . . . .	7
2.3	Future Releases . . . . .	17
2.4	Security Updates . . . . .	17
2.5	Feedback . . . . .	17
<b>3</b>	<b>Architecture and Best Practices</b>	<b>19</b>
<b>4</b>	<b>Open edX Development</b>	<b>21</b>



This the landing page for documentation relevant to developers of the Open edX platform. For other documentation resources targeted for edX learners, educators, researchers, and Open edX operators, see [docs.edx.org](https://docs.edx.org).



---

## Open edX Developer's Guide

---

This is the landing page for developer guides created by the edX Engineering team. It contains guides relevant to developers of the Open edX platform. This guide is intended to replace the [\(Legacy\) Open edX Developer's Guide](#).

### 1.1 Micro-frontend applications in Open edX

The purpose of this document is to provide an overview of how micro-frontend applications (MFEs) are developed, configured, and deployed with the Open edX ecosystem. Open edX MFEs are small React applications that can be built and deployed independently.

MFEs are dynamic client side rendered Javascript applications. MFE source code can be found in projects named `frontend-app-*`. These projects leverage `node.js` for their build processes. They are supplied configuration, then run to build HTML, CSS, and Javascript. The static assets are then ready to be deployed and served statically.

To run an MFE locally you must install `node 12` (with `npm` version 6.11 or higher). All MFEs offer the following commands:

```
npm install (install dependencies)
npm start (development server)
npm run lint
npm run test
npm run build (build static output)
```

#### 1.1.1 Configuration with Environment Variables

Open edX MFEs expect a number of process environment variables (Accessible in `node` via `process.env`) in order to properly configure authentication, routing to other parts of the system, etc. Each MFE has `.env`, `.env.development`, and `.env.test` files that are consumed by the MFE. The production `.env` file contains only null values – the build process should supply them via the command line prior to running `npm run build`.

Example `.env.development`:

```
NODE_ENV='development'  
PORT=1995  
ACCESS_TOKEN_COOKIE_NAME='edx-jwt-cookie-header-payload'  
BASE_URL='localhost:1995'  
CREDENTIALS_BASE_URL='http://localhost:18150'  
CSRF_TOKEN_API_PATH='/csrf/api/v1/token'  
ECOMMERCE_BASE_URL='http://localhost:18130'  
LANGUAGE_PREFERENCE_COOKIE_NAME='openedx-language-preference'  
LMS_BASE_URL='http://localhost:18000'  
LOGIN_URL='http://localhost:18000/login'  
LOGOUT_URL='http://localhost:18000/login'  
MARKETING_SITE_BASE_URL='http://localhost:18000'  
ORDER_HISTORY_URL='localhost:1996/orders'  
REFRESH_ACCESS_TOKEN_ENDPOINT='http://localhost:18000/login_refresh'  
SEGMENT_KEY=null  
SITE_NAME='edX'  
USER_INFO_COOKIE_NAME='edx-user-info'
```

### 1.1.2 Overriding Brand Specific Elements

MFEs contain Open edX branded headers, footers and style. To build a MFE to reflect the brand of the particular Open edX instance some npm dependencies are designed to be overridden. **Overriding packages must expose the same interface as the dependencies they are overriding.**

Example: for edX.org the Open edX `@edx/frontend-component-header` is overridden with `@edx/frontend-component-header-edx` using npm aliases (introduced in npm version 6.9.0). Example syntax below:

```
# Syntax: npm install <package-name>@<type>:<branded-package>  
  
# npm package  
npm install @edx/frontend-component-header@npm:@edx/frontend-component-header-  
→edx@latest  
  
# git repository  
npm install @edx/frontend-component-header@git:https://github.com/edx/frontend-  
→component-header-edx.git  
  
# local folder  
npm install @edx/frontend-component-header@file:../path/to/local/module
```

A list overridable packages is being developed. Currently only `@edx/frontend-component-header` and `@edx/frontend-component-footer` are designed to be overridden. See those repositories for the interfaces they expose.

### 1.1.3 Deployment

The basic deployment strategy:

- Run the build script with environment variables on the command line. Example: `NODE_ENV=development BASE_URL=open.edx.org ETC=etc npm run build`
- `dist/` directory is created that contains the deployable artifacts.
- Copy the contents of `dist/` to a web server.
- Configure the platform to point at your MFE. (details on this coming soon)

Note: The following is edX-specific and depends upon automation set up in GoCD. A similar, more manual process will need to be defined for Open edX installations.

edX MFEs are deployed automatically upon updates to the master branch in Github. GoCD collects three materials: the MFE source code, .yml configuration, and pipeline scripts (edX tubular).

.yml configuration contains:

- APP\_CONFIG
- NPM\_OVERRIDES
- S3\_BUCKET\_NAME

Relevant pipeline scripts (edX tubular):

- [https://github.com/edx/tubular/blob/master/tubular/scripts/frontend\\_utils.py](https://github.com/edx/tubular/blob/master/tubular/scripts/frontend_utils.py)
- [https://github.com/edx/tubular/blob/master/tubular/scripts/frontend\\_build.py](https://github.com/edx/tubular/blob/master/tubular/scripts/frontend_build.py)
- [https://github.com/edx/tubular/blob/master/tubular/scripts/frontend\\_deploy.py](https://github.com/edx/tubular/blob/master/tubular/scripts/frontend_deploy.py)

The pipeline scripts perform a routine like below:

- Parse .yml configuration
- Install requirements in the MFE source via `npm install`
- Check for NPM\_OVERRIDES and install them via `npm install @edx/pkg@npm:custom-pkg`
- Collect APP\_CONFIG and transform it into command line form (`NODE_ENV=development`  
`BASE_URL=open.edx.org`)
- Build the static output with the APP\_CONFIG: `NODE_ENV=development` `BASE_URL=open.edx.org`  
`npm run build`
- Deploy static output to s3 using S3\_BUCKET\_NAME



---

## Open edX Named Releases

---

The Open edX community can share knowledge and improvements more easily when most people use the same stable, consistent version of the Open edX codebase. To that end, edX creates “Open edX named releases”, which are distinct from the daily deployments to edx.org and have a longer release cycle (on the order of six months between each release). These releases will be tested both by edX and by the Open edX community.

Open edX releases are named alphabetically with botanical tree names.

### 2.1 Latest Open edX Release

The latest supported release line is *Juniper*, based on code from 2020-05-27.

The next release will be *Koa*.

### 2.2 All Open edX Releases

- *Juniper*
- *Ironwood*
- *Hawthorn*
- *Ginkgo*
- *Ficus*
- *Eucalyptus*
- *Dogwood*
- *Cypress*
- *Birch*

- *Aspen*

Note that the latest release is the only supported release. We currently do not have the ability to support more than one release at a time.

For documentation on the latest release, please visit <http://edx.readthedocs.org/projects/open-edx-release-notes/en/latest/>

Every release line (Ginkgo, Hawthorn, etc) has a branch that accumulates changes destined for that release (`open-release/ginkgo.master`, `open-release/hawthorn.master`, etc). Periodically, we tag that branch (`open-release/hawthorn.1`, `open-release/hawthorn.2`, etc). After a few releases, we might still add important fixes to the branch, but not make a new tagged release. At that point, someone installing that line will want to install from the branch, not the tags.

If an installation of a tag fails, try the corresponding release line master branch, it may have a fix.

### 2.2.1 Juniper

- **Code cut date:** 2020-05-27
- **Status:** supported
- [Release Notes](#)

Release Name	Release Date	Git Tag
Juniper.3	2020-08-25	open-release/juniper.3
Juniper.2	2020-07-13	open-release/juniper.2
Juniper.1	2020-06-09	open-release/juniper.1

### 2.2.2 Ironwood

- **Code cut date:** 2019-01-17
- **Status:** unsupported
- [Release Notes](#)

Release Name	Release Date	Git Tag
Ironwood fixes	2019-06-26+	open-release/ironwood.master
Ironwood.2	2019-06-05	open-release/ironwood.2
Ironwood.1	2019-03-21	open-release/ironwood.1

### 2.2.3 Hawthorn

- **Code cut date:** 2018-07-03
- **Status:** unsupported
- **Note:** Devstack is now based on Docker.
- [Release Notes](#)

Release Name	Release Date	Git Tag
Hawthorn.2	2018-09-26	open-release/hawthorn.2
Hawthorn.1	2018-08-07	open-release/hawthorn.1

## 2.2.4 Ginkgo

A note about Vagrant box files:

- Ginkgo and earlier had Vagrant box files. Hawthorn and beyond do not.
- Not every release needed new box files.
- Box files the same as the previous release are marked with an asterisk \*.
- Hashes are SHA1 hashes of the box file, not git commit hashes.
- **Code cut date:** 2017-07-06
- **Status:** unsupported
- **Latest:** open-release/ginkgo.master
- [Release Notes](#)

Release Name	Release Date	Git Tag	Vagrant Box Files
Ginkgo.2	2017-12-18	open-release/ginkgo.2	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– ginkgo-devstack-2017-07-14*</li> <li>– a7e3fce6d0155cde28e9f3253</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– ginkgo-fullstack-2017-12-14</li> <li>– c05fcd63df5fae452f0c8cb847</li> </ul> </li> </ul>
Ginkgo.1	2017-08-14	open-release/ginkgo.1	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– ginkgo-devstack-2017-07-14</li> <li>– a7e3fce6d0155cde28e9f3253</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– ginkgo-fullstack-2017-08-14</li> <li>– 990d5fdb5bbc7683c158dd99</li> </ul> </li> </ul>

## 2.2.5 Ficus

- **Code cut date:** 2017-01-10
- **Status:** unsupported
- **Latest:** open-release/ficus.master
- [Release Notes](#)

Release Name	Release Date	Git Tag	Vagrant Box Files
Ficus.4	2017-08-10	open-release/ficus.4	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– ficus-devstack-2017-02-07*</li> <li>– a7fb2200ccdb9f847bee7acd9</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– ficus-fullstack-2017-08-10</li> <li>– c9f59b27b39339d12fcf008f7</li> </ul> </li> </ul>
Ficus.3	2017-04-21	open-release/ficus.3	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– ficus-devstack-2017-02-07*</li> <li>– a7fb2200ccdb9f847bee7acd9</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– ficus-fullstack-2017-04-20</li> <li>– 64eb0a247d99454bccf0eed7e</li> </ul> </li> </ul>
Ficus.2	2017-03-29	open-release/ficus.2	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– ficus-devstack-2017-02-07*</li> <li>– a7fb2200ccdb9f847bee7acd9</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– ficus-fullstack-2017-03-28</li> <li>– fc6aa0d3b686c83e38e8c7fa1</li> </ul> </li> </ul>
Ficus.1	2017-02-23	open-release/ficus.1	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– ficus-devstack-2017-02-07</li> <li>– a7fb2200ccdb9f847bee7acd9</li> </ul> </li> <li>• <b>fullstack</b></li> </ul>
<b>2.2. All Open edX Releases</b>			<ul style="list-style-type: none"> <li>– ficus-devstack-2017-02-07</li> <li>– a7fb2200ccdb9f847bee7acd9</li> <li>• <b>fullstack</b></li> </ul>

## 2.2.6 Eucalyptus

- **Code cut date:** 2016-07-13
- **Status:** unsupported
- **Latest:** [open-release/eucalyptus.master](#)
- [Release Notes](#)

Release Name	Release Date	Git Tag	Vagrant Box Files
Eucalyptus.3	2017-01-10	open-release/eucalyptus.3	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– eucalyptus-devstack-2016-09-01*</li> <li>– a26c8fdbb431279863654161</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– eucalyptus-fullstack-2017-01-10</li> <li>– 64fd2a6efd656a7170127cccd</li> </ul> </li> </ul>
Eucalyptus.2	2016-09-02	open-release/eucalyptus.2	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– eucalyptus-devstack-2016-09-01</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– eucalyptus-fullstack-2016-09-01</li> </ul> </li> </ul>
Eucalyptus.1	2016-08-26	open-release/eucalyptus.1	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– eucalyptus-devstack-2016-08-19</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– eucalyptus-fullstack-2016-08-25</li> </ul> </li> </ul>

### 2.2.7 Dogwood

- **Code cut date:** 2015-12-15
- **Status:** unsupported

- **Latest:** [named-release/dogwood.rc](#)
- [Release Notes](#)

Release Name	Release Date	Git Tag	Vagrant Box Files
Dogwood.3	2016-04-25	named-release/dogwood.3	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– dogwood-devstack-2016-03-09*</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– dogwood-fullstack-rc2*</li> </ul> </li> </ul>
Dogwood.2	2016-04-14	named-release/dogwood.2	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– dogwood-devstack-2016-03-09*</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– dogwood-fullstack-rc2*</li> </ul> </li> </ul>
Dogwood.1	2016-03-09	named-release/dogwood.1	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– dogwood-devstack-2016-03-09</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– dogwood-fullstack-rc2*</li> </ul> </li> </ul>
Dogwood	2016-02-11	named-release/dogwood	<ul style="list-style-type: none"> <li>• <b>devstack</b> <ul style="list-style-type: none"> <li>– dogwood-devstack-rc2</li> </ul> </li> <li>• <b>fullstack</b> <ul style="list-style-type: none"> <li>– dogwood-fullstack-rc2</li> </ul> </li> </ul>

## 2.2.8 Cypress

- **Code cut date:** 2015-07-07
- **Status:** unsupported
- **Latest:** [named-release/cypress.rc](#)
- [Release Notes](#)

Release Name	Release Date	Git Tag	Vagrant Box Files
Cypress	2015-08-13	named-release/cypress	<ul style="list-style-type: none"> <li>• devstack</li> <li>• fullstack</li> </ul>

## 2.2.9 Birch

- **Code cut date:** 2015-01-29
- **Status:** unsupported
- **Latest:** [named-release/birch.rc](#)
- [Release Notes](#)

Release Name	Release Date	Git Tag	Vagrant Box Files
Birch.2	2015-08-05	named-release/birch.2	<ul style="list-style-type: none"> <li>• devstack</li> <li>• fullstack</li> </ul>
Birch.1	2015-07-27	named-release/birch.1	<ul style="list-style-type: none"> <li>• devstack</li> <li>• fullstack</li> </ul>
Birch	2015-02-24	named-release/birch	<ul style="list-style-type: none"> <li>• devstack</li> <li>• fullstack</li> </ul>

## 2.2.10 Aspen

- **Code cut date:** 2014-09-05
- **Status:** unsupported
- [Release Notes](#)

Release Name	Release Date	Git Tag	Vagrant Box Files
Aspen	2014-10-28	named-release/aspen	<ul style="list-style-type: none"> <li>• devstack</li> <li>• fullstack</li> </ul>

## 2.3 Future Releases

Upcoming releases have wiki pages for engineers to collect information that will be needed for their release on the [Open edX Release Planning](#) page.

## 2.4 Security Updates

If security vulnerabilities or other serious problems (such as data loss) are discovered in the most recent Open edX release, edX will release a new version of that release that includes the fix. We will not make patches of any releases before the most recent one. We are still working on the details of how often to update Open edX releases. We will publicly announce the security issue, and encourage the Open edX community to update their installations to close the vulnerability. If you have found a security vulnerability in the Open edX codebase, please report it by sending an email to [security@edx.org](mailto:security@edx.org). Please do not post the vulnerability to the public mailing lists.

## 2.5 Feedback

If you find a problem in the release candidate, please report them on the mailing list ([openedx-ops](#) for operations problem, [edx-code](#) for other problems), or even better, create issues in [JIRA](#). We are very interested to hear from the community about what you value about Open edX releases. Please let us know what you would like to see in future releases. Discussion on the [mailing list](#) is always welcome, of course!



---

## Architecture and Best Practices

---

<a href="#">Open edX Technology Radar (coming...)</a>	A bird's eye view of the decisions, tools and technologies that have been adopted or rejected for Open edX development.
<a href="#">Open edX Proposals</a>	Technical decisions made by the Open edX community, in the form of best practices, architecture design, or development processes.
<a href="#">Architecture and Engineering Confluence page</a>	Confluence page for notes, thoughts, and project-related documents on Open edX architecture and engineering.
<a href="#">(New) Open edX Developer's Guide</a>	General guidelines for developing on various parts of the Open edX code base.
<a href="#">(Legacy) Open edX Developer's Guide</a>	General guidelines for developing on various parts of the Open edX code base.



## CHAPTER 4

---

### Open edX Development

---

<a href="#">Open edX Named Releases</a>	Information on each stable Open edX named release.
<a href="#">Open edX Extensions and APIs</a>	Landing page for supported APIs and extensions to the Open edX platform.