

Announcing the Samsung ARTIK IDE Powered by Eclipse Che. [Learn More \(artik/\)](#)



(/)

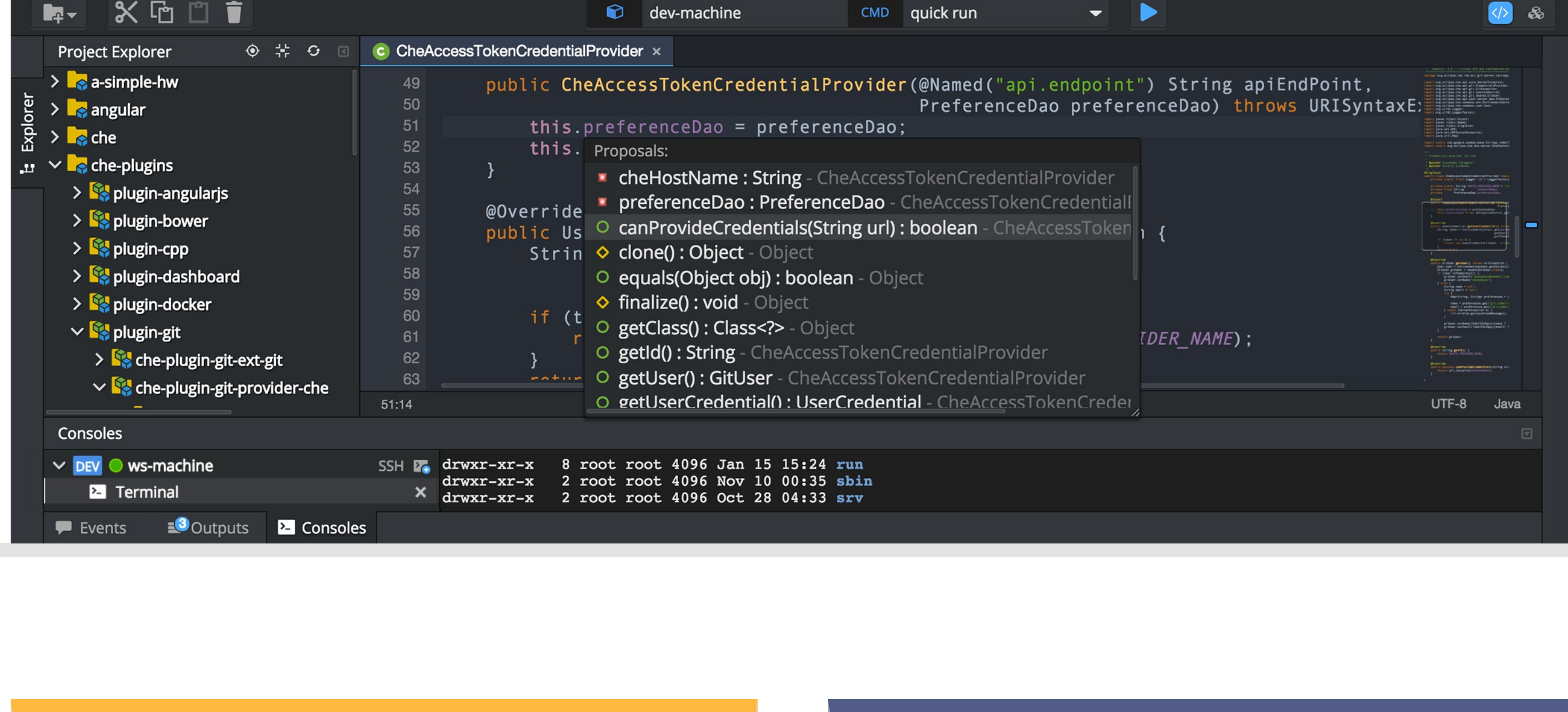
[Features \(features/\)](#) [Downloads \(download/\)](#) [Docs \(https://eclipse-che.readme.io/docs/\)](https://eclipse-che.readme.io/docs/) [Community \(community/\)](#) [Blog \(blog/\)](#)

Star **2,793**

(getting-started/)

# Next-Generation Eclipse IDE

Eclipse Che is a developer workspace server and cloud IDE.



**Try Eclipse Che in the Cloud**

Codenvy hosts Eclipse Che workspaces with free accounts.

[Get Started \(getting-started/\)](#)

**Download Eclipse Che**

Install binaries from an Eclipse mirror or build from source.

[Get Started \(getting-started/\)](#)

## Developers

Use Che as a desktop IDE for any framework or programming language. Take your projects and runtimes anywhere with workspace portability and export to the cloud.

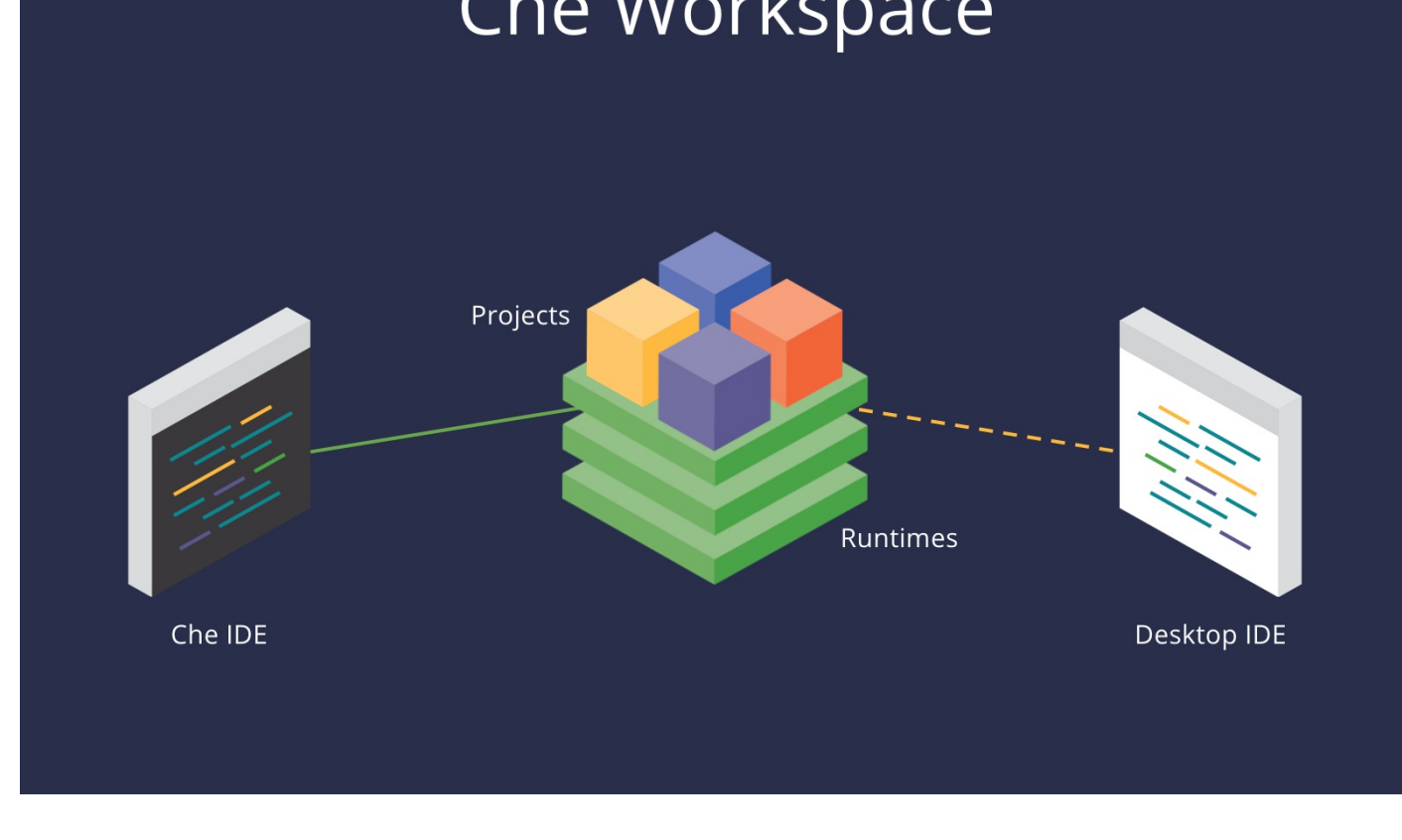
## Product Owners

Use Che to host on-demand workspaces accessible by any IDE or Web app with API/SSH. For example, SAP embeds Che in its development tools for SAP Hana.

## Plug-in Providers

Use Che to create hosted tooling farms for your customers, or package custom assemblies with your plug-ins distributable as executables to customers.

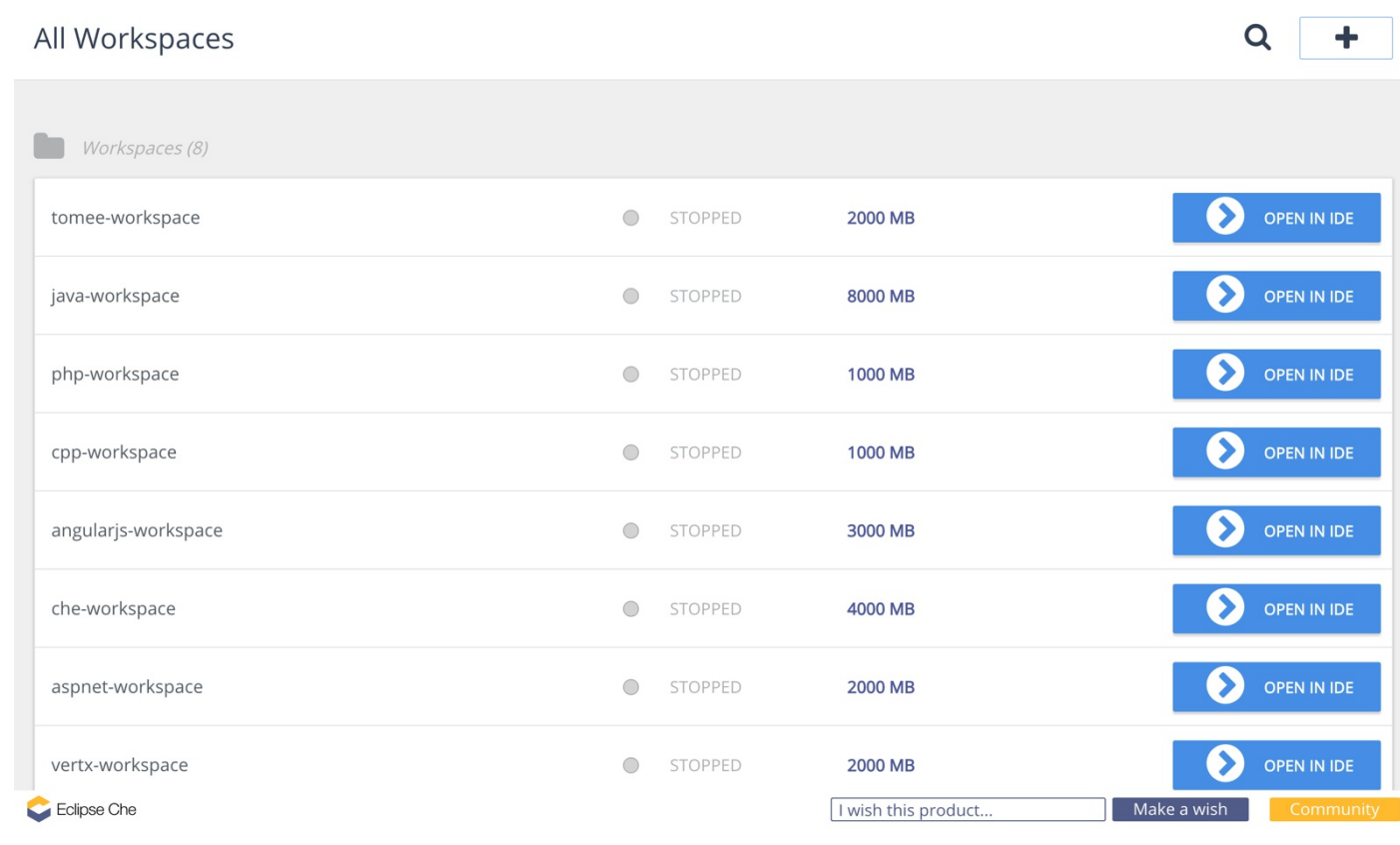
## Our Approach



(images/features/img-features-a-new-kind-of-workspace.png)

### Workspaces with Runtimes

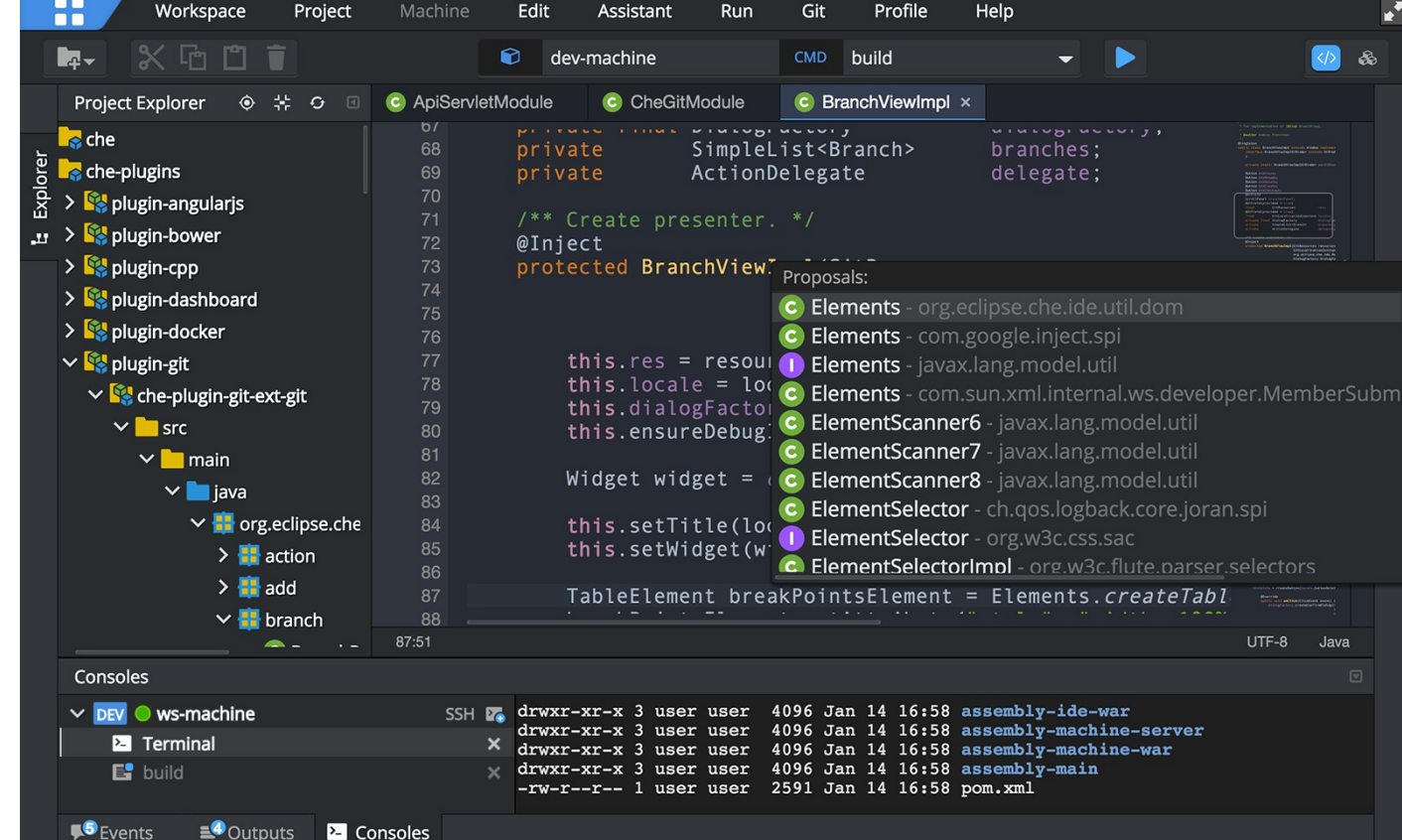
Workspaces are composed of projects and runtimes. Create portable workspace replicas that run anywhere in the cloud or on your desktop ... [Read More \(features/#new-workspace\)](#)



(images/features/img-features-collaborative-workspace-server.png)

### Collaborative Workspace Server

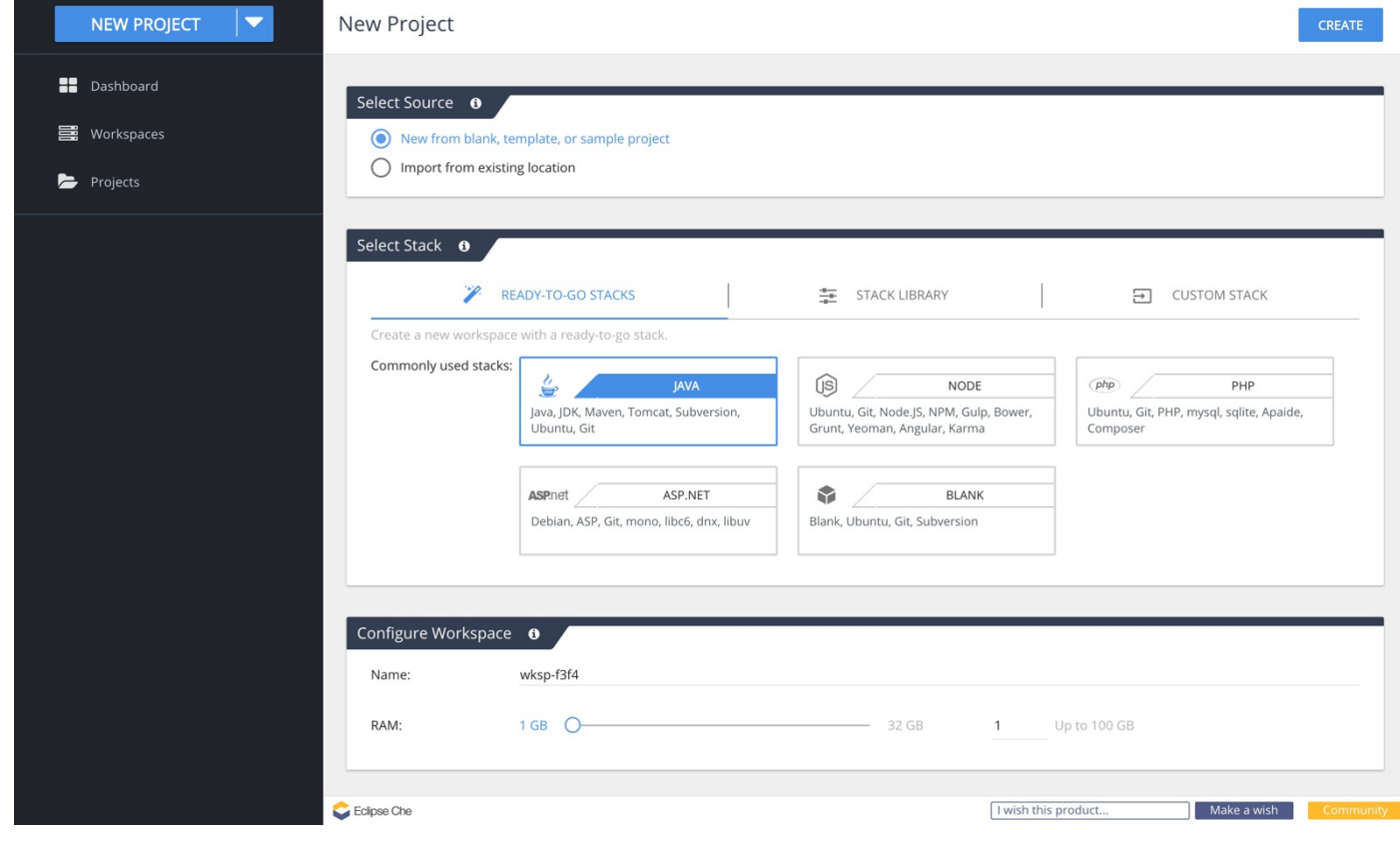
Host Eclipse Che as a workspace server. Share tools, runtime and programming services across workspaces and teams. Control workspaces with REST APIs ... [Read More \(features/#collaborative\)](#)



(images/features/img-features-cloud-ide.png)

### Cloud IDE

A no-installation thin, fast and IOE accessible from any local or remote device. Browser, IDE and beautiful - it's the IDE our own engineers wanted ... [Read More \(features/#cloud-ide\)](#)



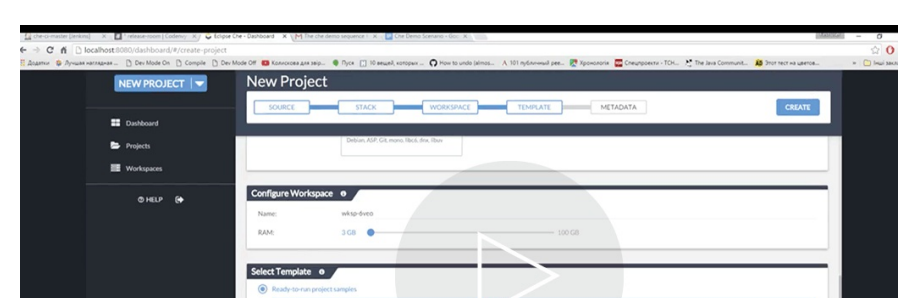
(images/features/img-features-docker-powered.png)

### Docker-Powered Environments

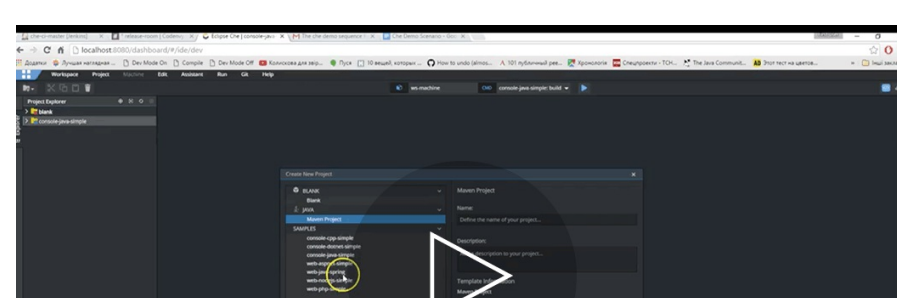
Workspace runtimes are Docker-powered. Use our all-in-one stacks, pull from any registry, or author your own. Snapshot and embed runtimes into ... [Read More \(features/#docker-powered\)](#)

## View All Product Features (features/)

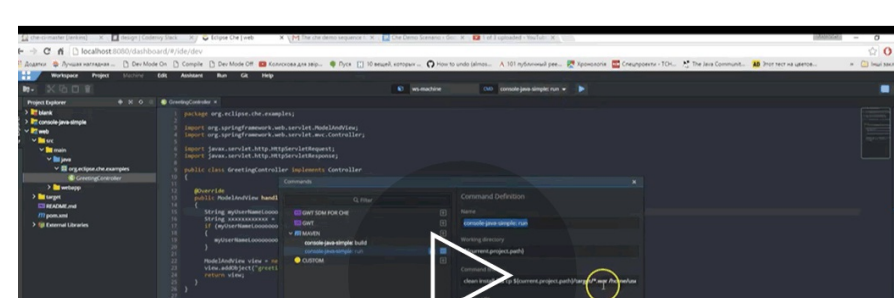
## Silent Screencasts



(https://youtu.be/zvofry3M27k)



(https://youtu.be/AVXyCmDKCGg)



(https://youtu.be/D\_BsLN9LcSA)

### Java IntelliSense

(https://youtu.be/zvofry3M27k)

15 Minutes 1 Second

### Commands & Previews

(https://youtu.be/AVXyCmDKCGg)

13 Minutes 38 Seconds

### Debugger

(https://youtu.be/D\_BsLN9LcSA)

5 Minutes 2 Seconds

## More Videos

### Terminal

(https://www.youtube.com/watch?v=86QmBy\_\_azY) 71 Secs

### Renaming a Java Class

(https://www.youtube.com/watch?v=4kCEimYLw3w) 49 Secs

### Renaming a Java Method

(https://www.youtube.com/watch?v=sKwhv5bB3U) 58 Secs

### Code Auto-Completion

(https://youtu.be/7BtLGqEjFRc) 150 Secs

### Open Declaration

(https://youtu.be/nokYHILL550) 64 Secs

### Parallel Workspaces

(https://www.youtube.com/watch?v=3bRba-2gF3k) 234 Secs

### Swing GUI with VNC

(https://www.youtube.com/watch?v=AjgSp0dkxxU) 182 Secs

### Portable Workspaces

(https://www.youtube.com/watch?v=X6izqLUNEM) 98 Secs

### Find Usage

(https://youtu.be/-GiZe8sksT8) 51 Secs

### Navigating Classes

(https://youtu.be/SDCWEOmgoxA) 55 Secs

### Git Diff

(https://youtu.be/5ItEe5BC4MM) 21 Secs

### Git History

(https://youtu.be/Clu6whdnNx4) 52 Secs

### Quick Create Project

(https://youtu.be/ZumoeSL4MGo) 48 Secs

### RESTful Workspaces

(https://youtu.be/ZumoeSL4MGo) 48 Secs

### Desktop IDE Over SSH

(https://www.youtube.com/watch?v=0WCPczMK\_8) 73 Secs

### OpenShift Plug-In

(https://www.youtube.com/watch?v=T05Y3zwg6Jk) 176 Secs

### Clone From GitHub

(https://youtu.be/\_UVrhcxBz8Q) 58 Secs

### Define Command

(https://youtu.be/Uj4lymKAXZg) 39 Secs

### Build and Run Project

(https://youtu.be/Z15mEmx2-Qg) 41 Secs

### Run Terminal

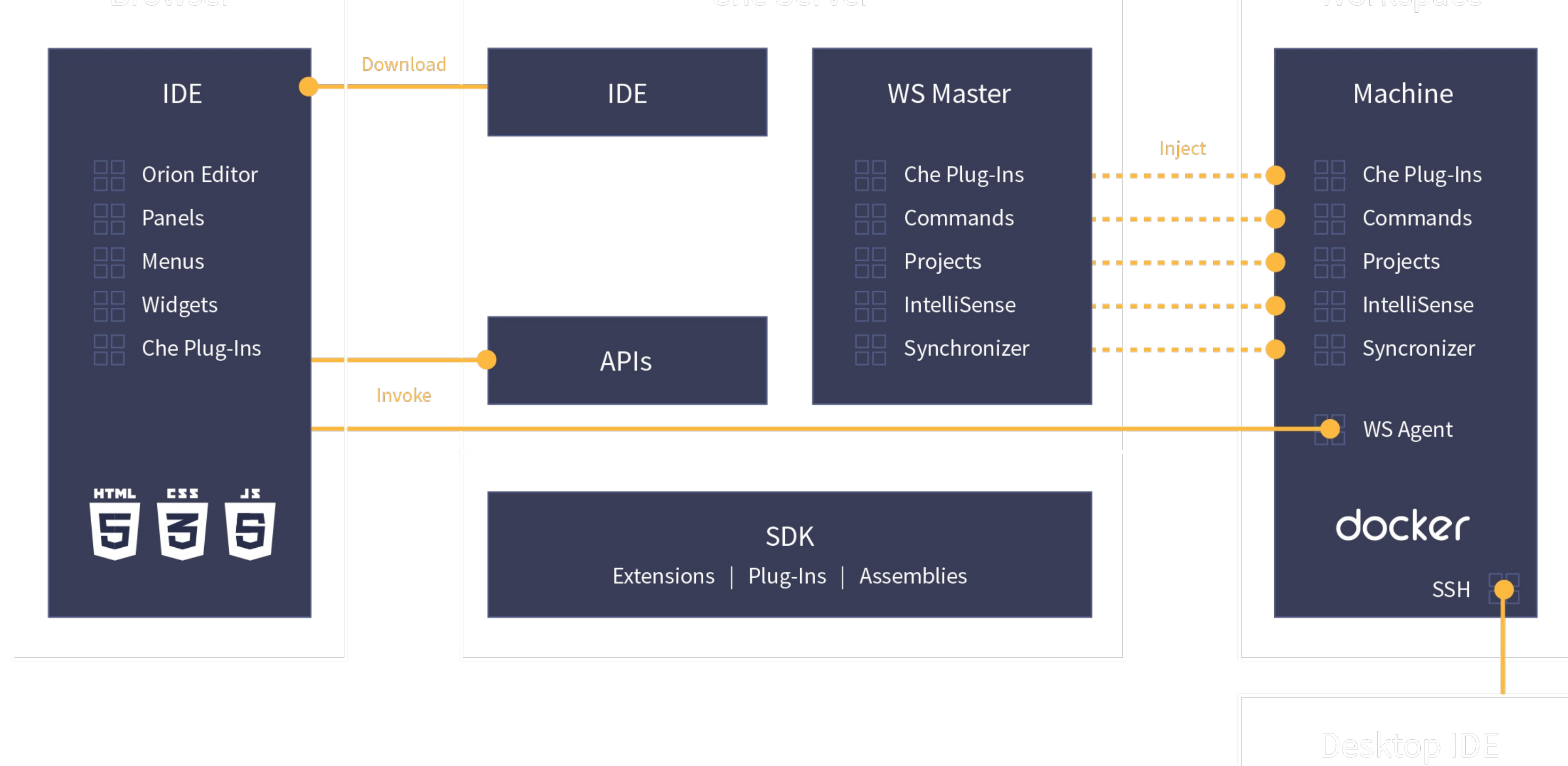
(https://www.youtube.com/watch?v=xKOGJmQPTyg) 305 Secs

### Maven Dynamic Dependencies

(https://www.youtube.com/watch?v=RBU80F2IMkU) 128 Secs

## Technology

We are building a world where anyone can contribute to a project without installing software. Che defines a new type of workspace that is composed of projects and runtimes, making its state distributable, portable and versionable. We use VMs, containers, and Web services to bring repeatability, consistency, and performance to workspaces.



## Cloud IDE

Che's browser IDE is built on an extensible plug-in platform that enables panels, menus, wizards, popups and tools. With an Orion editor performance, server-side RESTful code intellisense, the developer workspaces (the developer, non-blocking), or check out the customization resource center (https://github.com/eclipse/che/blob/master/CUSTOMIZING.md).

## Workspace Server

The Che server controls the lifecycle of workspaces and can be customized with plug-ins. Any client can communicate with the workspace server and any spawned workspaces. Read more in our documentation (https://eclipse-che.readme.io/docs/), or see all the features (features/).

## Workspaces

Workspaces are isolated and personal spaces for developer work. Whether developers use an API, browser, CLI, or SSH to access a workspace, every projects are synchronized and kept consistent. Plug-ins enable service injection into workspaces based upon project type, making their workspace tailored to the projects it manages. Read more in our documentation (https://eclipse-che.readme.io/docs/).

## Plug-Ins

Che includes a growing set of plug-ins (https://eclipse-che.readme.io/docs/plug-ins). You can also create and package (https://eclipse-che.readme.io/docs/developing-extensions) your own. You can also extend Che's runtimes, stacks, commands, look-and-feel. We are starting to build out customization resource center (https://github.com/eclipse/che/blob/master/CUSTOMIZING.md).

### Languages

Syntax highlighting, stacks, templates, and code assistants.

- Java
- C++
- JavaScript
- Python
- PHP
- Ruby
- SQL

### Frameworks

Project types, scaffold tool integration, and syntax analysis.

- OpenShift
- Angular JS
- Docker
- Yeoman

### Builders

Builders to manage unit tests, compilation, linking, and dependencies.

- Ant
- Bower
- Grunt
- Gulp
- Maven
- Npm

### Tools

Extensions that integrate the IDE with other parts of the developer tool chain.

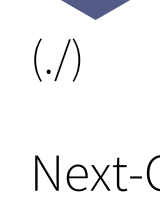
- Git
- Orion
- SSH
- Subversion

## Contributors



## Get Involved With Eclipse Che (community/)

 Was this page helpful? Share it on Twitter (<https://twitter.com/share>)



(,)

Next-Generation Eclipse IDE

Downloads ([download/](#))

Features ([features/](#))

Docs (<https://eclipse-che.readme.io/docs/>)

Community ([community/](#))

Blog ([blog/](#))

Eclipse Che

([getting-started/](#))

Copyright © Eclipse Foundation 2016