

CS 253: Parallel Functional Programming with Java & Android : Overview (Part 3)

Douglas C. Schmidt

d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt



Professor of Computer Science

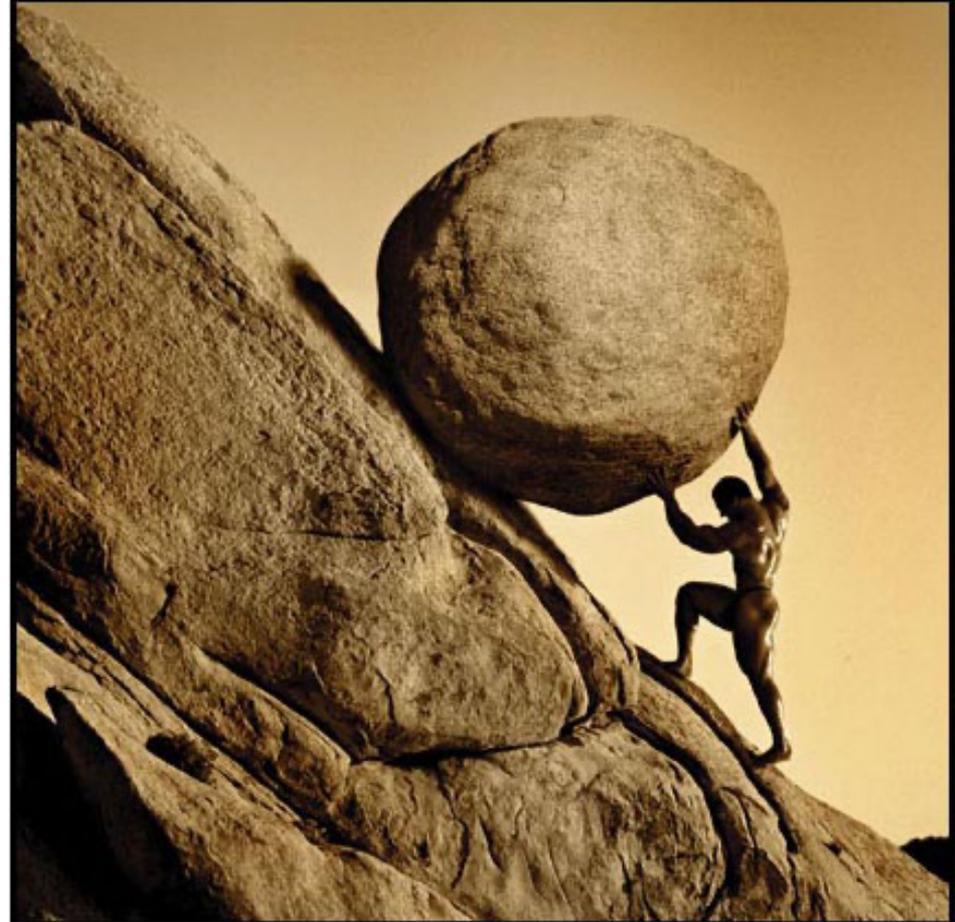
**Institute for Software
Integrated Systems**

**Vanderbilt University
Nashville, Tennessee, USA**



Learning Objectives in this Lesson

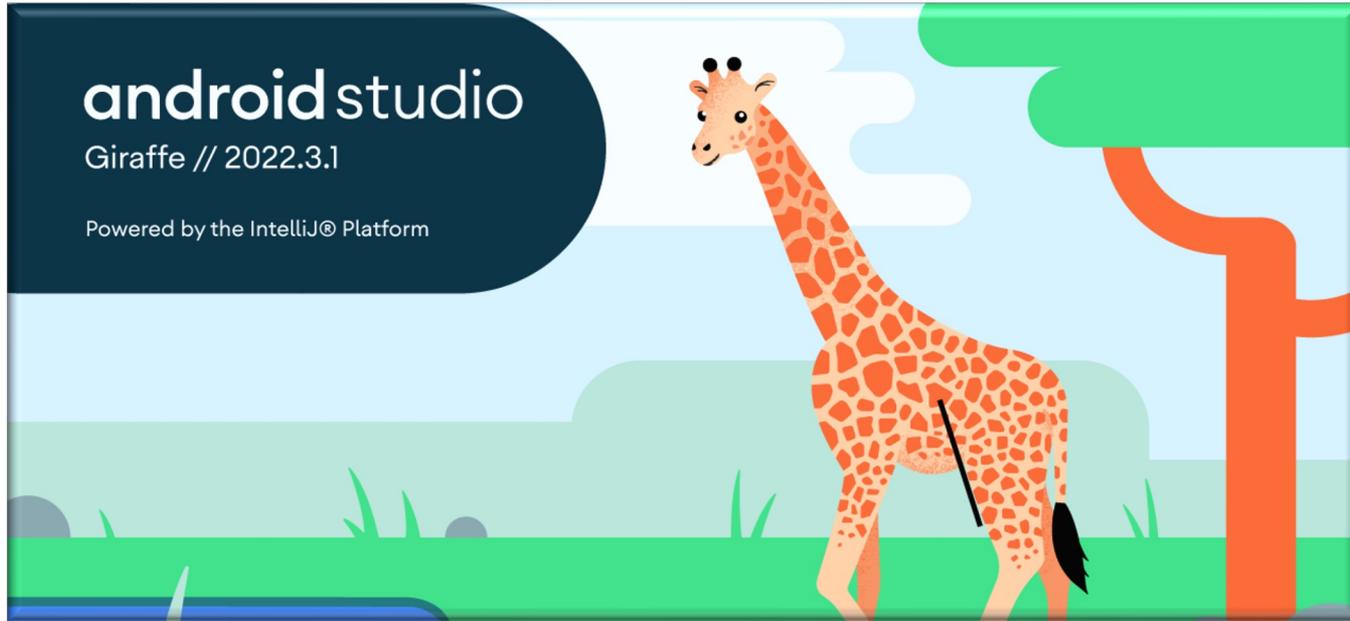
- Understand the course topics & logistics
 - Course philosophy
 - Course contents
 - Structure of the lecture material
 - Overview of the assignments & assessments
- Setting up the Java & Android IDE on Android Studio
- Setting up GitLab et al.
- Accessing Android & Java source code



Setting Up the Android & Java IDE on Android Studio

Installing Java/Android Developer Tools

- To use Android, you need to install the latest release of Android Studio



Select the version of Android Studio that's right for your Mac:

Android Studio Giraffe | 2022.3.1

Mac with Intel chip

Mac with Apple chip

android-studio-2022.3.1.18-mac.dmg

android-studio-2022.3.1.18-mac_arm.dmg

See developer.android.com/studio

Installing Java/Android Developer Tools

- Installation steps



Installing Java/Android Developer Tools

- Installation steps
 - Download & install the latest version of Android Studio

Android Studio Giraffe | 2022.3.1

On this page 

[Android Gradle plugin and Android Studio compatibility](#)

[Older versions](#)

[Maximum supported Android API level](#)

[Use Live Edit to update composables in real time](#)

[New UI preview](#)

...

Android Studio is the official IDE for Android development, and includes everything you need to build Android apps.

This page lists new features and improvements in the latest version in the stable channel, Android Studio Giraffe. You can [download it here](#) or update to it inside Android Studio by clicking **Help > Check for updates (Android Studio > Check for updates** on macOS)

To view the release notes for older versions of Android Studio, see [Past releases](#).

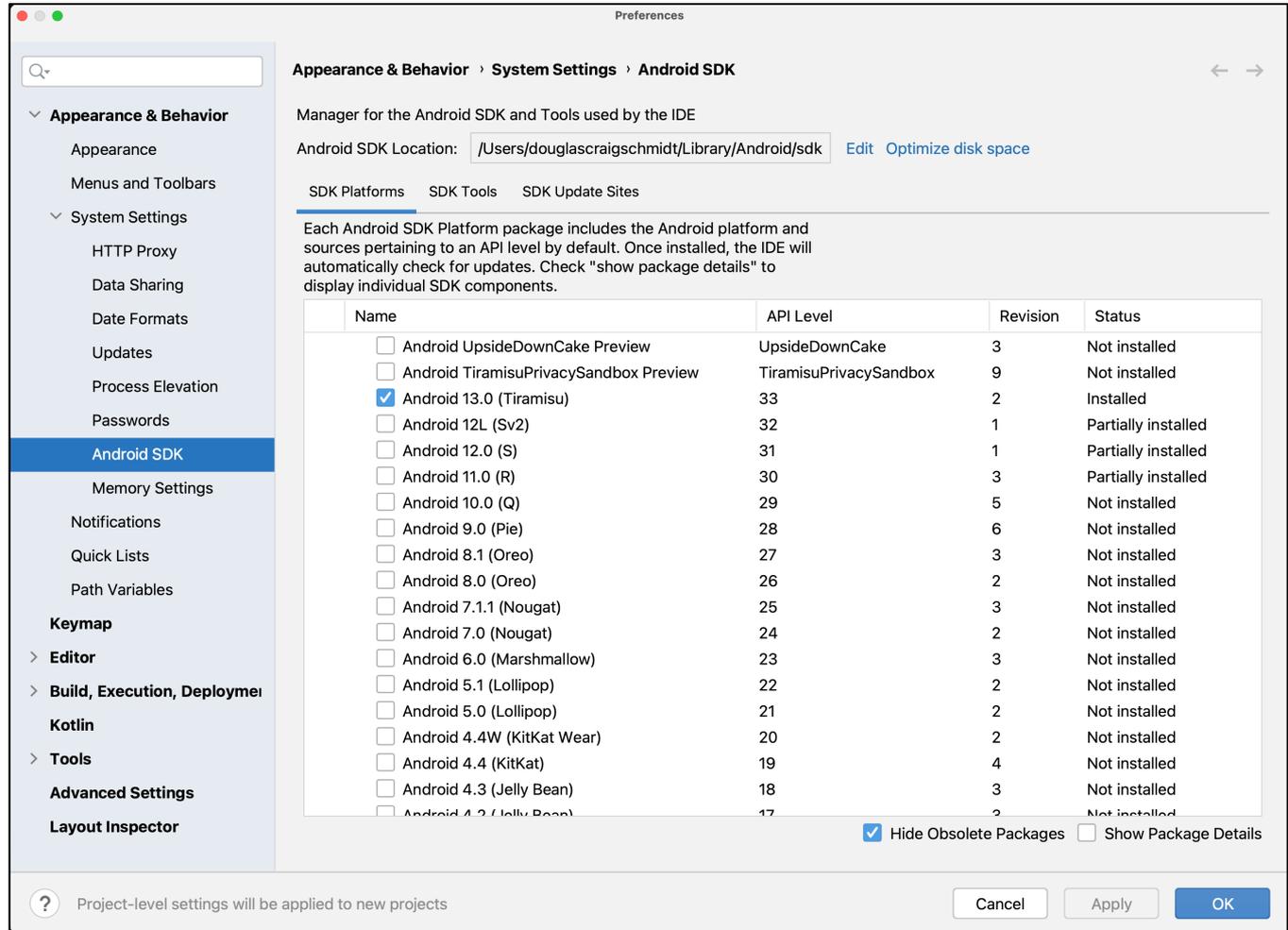
For early access to upcoming features and improvements, see the [Preview builds of Android Studio](#).

If you encounter problems in Android Studio, check the [Known issues](#) or [Troubleshoot](#) page.

See developer.android.com/studio

Add Components to the SDK

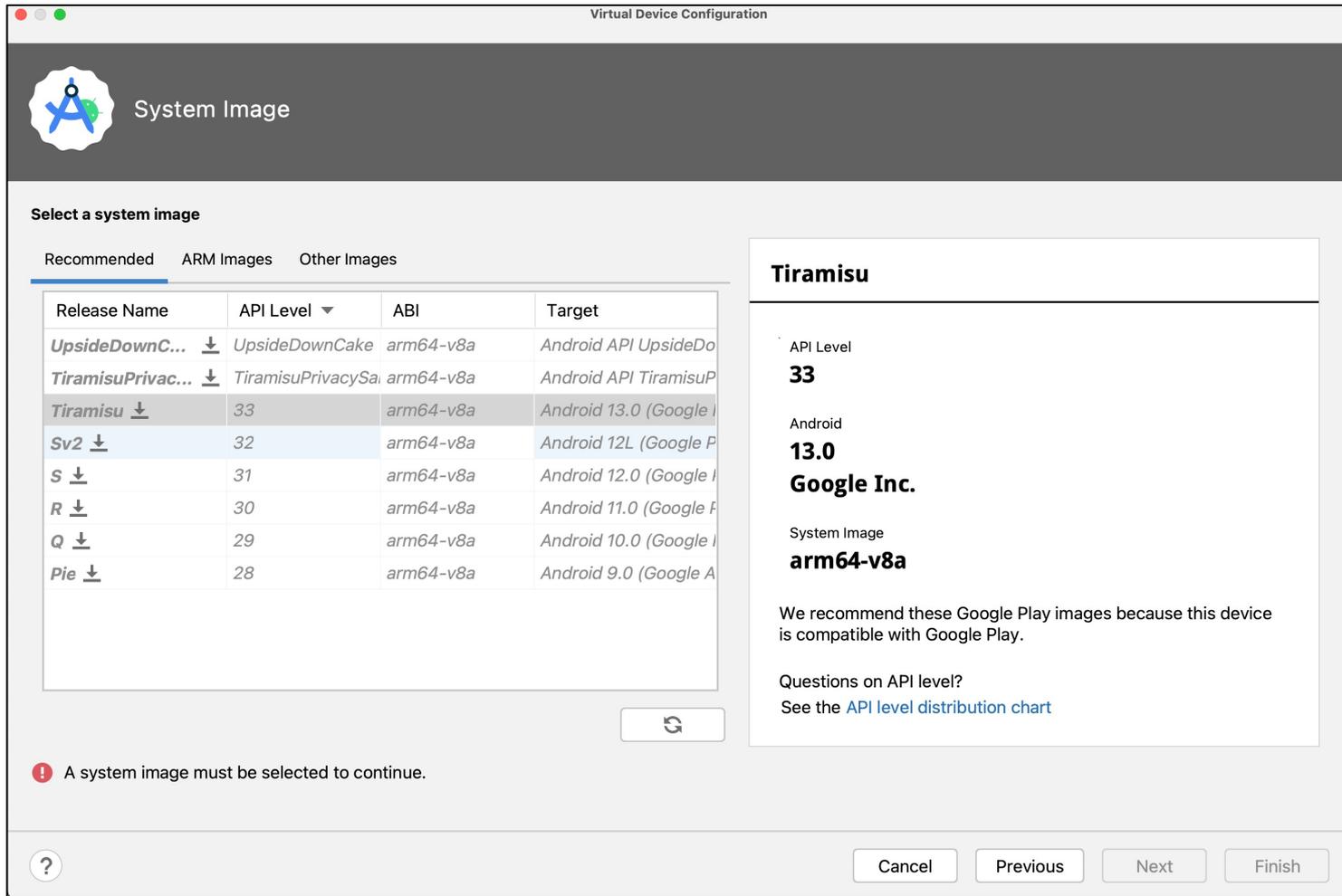
- Launch the Android Studio SDK Manager
- Select Android 13, API level 33



See developer.android.com/studio/intro/update.html

Add Components to the SDK

- Launch the Android Studio Virtual Device Manager
- Create an Android API 33 emulator



developer.android.com/tools/devices/managing-avds.html

Intel HAXM Virtualization Driver

• Requirements

- Intel virtualization extensions (VT, VT-x, vmx)
- AMD virtualization extensions (AMD-v, SVM) [only supported on Linux]
- Download an x86 emulator image

• Windows & Mac OSX

- `<sdk>/extras/intel/Hardware_Accelerated_Execution_Manager/IntelHAXM.exe/dmg`

• Linux

- Install KVM & pass “-enable-kvm” flag to emulator when starting

Configure hardware acceleration for the Android Emulator

On this page 

Configure graphics acceleration

Requirements

Configure graphics acceleration in the AVD Manager

Configure graphics acceleration from the command line

Enable Skia rendering for Android UI

Configuring VM acceleration

General requirements

Restrictions

...

The [Android Emulator](#) can use hardware acceleration features to improve performance, sometimes drastically. This page describes how you can configure graphics and virtual machine (VM) acceleration to get higher performance from the emulator.

developer.android.com/studio/run/emulator-acceleration

Setting Up GitLab et al.

Setting Up GitLab et al.

The screenshot shows the GitLab dashboard homepage. At the top, there is a navigation bar with the GitLab logo, a search bar, and various utility icons. Below the navigation bar, a promotional banner for a "Free Trial of GitLab.com Gold" is visible. The main content area is titled "Welcome to GitLab" and includes the tagline "Code, test, and deploy together". There are four main action cards: "Create a project" (highlighted with a red border), "Create a group", "Explore public projects", and "Learn more about GitLab".

Free Trial of GitLab.com Gold
Try all GitLab has to offer for 30 days. No credit card required. [Start your trial](#)

Welcome to GitLab

Code, test, and deploy together

- Create a project**
Projects are where you store your code, access issues, wiki and other features of GitLab.
- Create a group**
Groups are the best way to manage projects and members.
- Explore public projects**
There are 1,385,678 public projects on this server. Public projects are an easy way to allow everyone to have read-only access.
- Learn more about GitLab**
Take a look at the documentation to discover all of GitLab's capabilities.

Setting Up GitLab et al.

New Project · GitLab

gitlab.com/projects/new

Projects Groups Activity Milestones Snippets

New project

A project is where you house your files (repository), plan your work (issues), and publish your documentation (wiki), among other things.

All features are enabled for blank projects, from templates, or when importing, but you can disable them afterward in the project settings.

To only use CI/CD features for an external repository, choose **CI/CD for external repo**.

Information about additional Pages templates and how to install them can be found in our [Pages getting started guide](#).

Tip: You can also create a project from the command line. [Show command](#)

Blank project Create from template Import project CI/CD for external repo

Project name

Project URL **Project slug**

Want to house several dependent projects under the same namespace? [Create a group](#).

Project description (optional)

Visibility Level

- Private**
Project access must be granted explicitly to each user.
- Public**
The project can be accessed without any authentication.

Initialize repository with a README
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

Setting Up GitLab et al.

The screenshot shows the GitLab web interface in a browser. The address bar shows the URL `gitlab.com/lclfox/CS-253-fall-2023`. The page title is "CS-253-fall-2023". A blue banner at the top of the page content area says "Project 'CS-253-fall-2023' was successfully created." Below this, the project name "CS-253-fall-2023" is displayed with a lock icon and "Project ID: 13944228". There are buttons for "Add license", "New file", "Add README", "Add CHANGELOG", and "Add CONTRIBUTING". The "New file" button is highlighted in green. Below these buttons, there is a section for "Command line instructions" with a code block containing the following commands:

```
git config --global user.name "Lindsey Fox"
git config --global user.email "lindsey.fox@vanderbilt.edu"
```

Underneath, there is a section for "Create a new repository" with a code block containing the following commands:

```
git clone https://gitlab.com/lclfox/cs-891-fall-2019.git
cd cs-891-fall-2019
touch README.md
git add README.md
```

The left sidebar contains navigation links for Project, Details, Activity, Cycle Analytics, Issues (0), Merge Requests (0), CI / CD, Operations, Packages, Wiki, Snippets, and Settings. The top navigation bar includes the GitLab logo, search bar, and various utility icons.

Setting Up GitLab et al.

Members · Lindsey Fox / CS-253- x +

gitlab.com/lclfox/CS-253-fall-2023/-/project_members

GitLab Projects Groups Activity Milestones Snippets

Lindsey Fox > CS-253-fall-2023 > Members

Project members

You can invite a new member to **CS-253-fall-2023** or invite another group.

Invite member Invite group

GitLab member or Email address

Douglas Craig Schmidt

Choose a role permission

Maintainer

Read more about role permissions

Access expiration date

Expiration date

Add to project Import

Existing members and groups

Members of **CS-253-fall-2023** 1

Find existing members by name Sort by Name, ascending

Lindsey Fox @lclfox It's you

Given access 2 minutes ago

Maintainer

Setting Up GitLab et al.

The screenshot shows the GitLab web interface. The browser address bar displays `gitlab.com/lclfox/CS-253-fall-2023/edit`. The page title is "General Settings" for the project "cs-891-fall-2019".

The left sidebar contains a navigation menu with the following items: Project, Issues (0), Merge Requests (0), CI / CD, Operations, Packages, Wiki, Snippets, Settings (selected), General (selected), Members, Integrations, Repository, CI / CD, Operations, Pages, and Audit Events. A "Collapse sidebar" button is at the bottom left.

The main content area is titled "Naming, topics, avatar" and includes a "Collapse" button. Below the title is the instruction: "Update your project name, topics, description and avatar." The form contains the following fields:

- Project name:**
- Project ID:**
- Topics:**
- Project description (optional):**
- Project avatar:** A circular avatar with the letter 'C' and a "Choose file..." button. Below it, it says "No file chosen" and "The maximum file size allowed is 200KB."

A "Save changes" button is located below the avatar section.

The next section is titled "Visibility, project features, permissions" and includes a "Collapse" button. Below the title is the instruction: "Choose visibility level, enable/disable project features (issues, repository, wiki, snippets) and set permissions." The form contains the following field:

- Project visibility:** A dropdown menu currently set to "Private". Below it, it says "The project is accessible only by members of the project. Access must be granted explicitly to each user."

Below the visibility section, the "Issues" section is partially visible.

Setting Up GitLab et al.

2. Clone your GitLab repo

```
git clone git@gitlab.com:your-name/CS-253-fall-2023.git
```

1. Create Your GitLab Repo

www.gitlab.com



**Working Folder
(Student's)**

**GitLab Repo
(Student's)**

```
cd CS-253-fall-2023
```

3. Change Director into Your Working Folder

See docs.gitlab.com/ee/ssh for info on setting up an SSH key for GitLab et al.

Setting Up GitLab et al.

4. Update from Read-Only GitHub Repo

```
git remote add skeletons
```

```
git@github.com:douglasraigschmidt/CS253.git
```

5. Get Current Version

```
git pull skeletons main
```

**Working Folder
(Student's)**

Assignment1a
...

6. Do work!

**Local Repo
(Student's)**

Assignment1a
...

**GitLab Repo
(Student's)**

Assignment1a
...

**GitHub Repo
(Instructor's)**

Assignment1a
Assignment1b
Assignment2a
Assignment2b
...

```
git commit
```

7. Commit Changes

```
git push origin main
```

8. Send Changes to GitLab Repo

See item #13 at github.com/douglasraigschmidt/CS253/wiki/CS-253-FAQ

Accessing Java & Android Source Code

Accessing Java & Android Source Code

- Android source code is available
 - For browsing android.googlesource.com

android Git repositories

To clone one of these repositories, install `git`, and run:

```
| git clone https://android.googlesource.com/name
```

Name

accessories/manifest
device/asus/deb
device/asus/flo
device/asus/flo-kernel
device/asus/grouper
device/asus/tilapia
device/common
device/generic/armv7-a
device/generic/armv7-a-neon
device/generic/art
device/generic/common
device/generic/goldfish
device/generic/mini-emulator-armv7-a-neon
device/generic/mini-emulator-mips
device/generic/mini-emulator-x86

Accessing Java & Android Source Code

- Android source code is available
 - For browsing android.googlesource.com
 - For downloading source.android.com

The Android Source Code

Android is an open-source software stack created for a wide array of devices with different form factors. The primary purposes of Android are to create an open software platform available for carriers, OEMs, and developers to make their innovative ideas a reality and to introduce a successful, real-world product that improves the mobile experience for users. We also wanted to make sure there was no central point of failure, where one industry player could restrict or control the innovations of any other. The result is a full, production-quality consumer product with source code open for customization and porting.

Governance Philosophy

Android was originated by a group of companies known as the Open Handset Alliance, led by Google. Today, many companies – both original members of the OHA and others – have invested heavily in Android. These companies have allocated significant engineering resources to improve Android and bring Android devices to market.

The companies that have invested in Android have done so on its merits because we believe an open platform is necessary. Android is intentionally and explicitly an open-source – as opposed to a free software – effort; a group of organizations with shared needs has pooled resources to collaborate on a single implementation of a shared product. The Android philosophy is pragmatic, first and foremost. The objective is a shared product that each contributor can tailor and customize.

Uncontrolled customization can, of course, lead to incompatible implementations. To prevent this, the Android Open Source Project also maintains the [Android Compatibility Program](#), which spells out what it means to be "Android compatible" and what is required of device builders to achieve that status. Anyone can (and will!) use the Android source code for any purpose, and we welcome all legitimate uses. However, in order to take part in the shared ecosystem of applications we are building around Android, device builders must participate in the Android Compatibility Program.

The Android Open Source Project is led by Google, who maintains and further develops Android. Although Android consists of multiple subprojects, this is strictly a project management technique. We view and manage Android as a single, holistic software product, not a "distribution", specification, or collection of replaceable parts. Our intent is that device builders port Android to a device; they don't implement a specification or curate a distribution.

Accessing Java & Android Source Code

- Java source code is available
 - For browsing zgrepcode.com

JDK Project

The goal of this long-running Project is to produce a series of open-source reference implementations of the Java SE Platform, as specified by JSRs in the Java Community Process. The Project ships a feature release every six months according to a strict, time-based model, *as proposed*.

Releases

- 20 (in development)
- 19 (in development)
- 18 (GA 2022/03/22)
- 17 (GA 2021/09/14)
- 16 (GA 2021/03/16)
- 15 (GA 2020/09/15)
- 14 (GA 2020/03/17)
- 13 (GA 2019/09/17)
- 12 (GA 2019/03/19)
- 11 (GA 2018/09/25)
- 10 (GA 2018/03/20)

Resources

- Development list: *jdk-dev*
- Main-line code repository: <https://github.com/openjdk/jdk/>

Accessing Java & Android Source Code

- Java source code is available
 - For browsing zgrepcode.com
 - For downloading openjdk.org/projects/jdk

JDK Project

The goal of this long-running Project is to produce a series of open-source reference implementations of the Java SE Platform, as specified by JSRs in the Java Community Process. The Project ships a feature release every six months according to a strict, time-based model, *as proposed*.

Releases

- 20 (in development)
- 19 (in development)
- 18 (GA 2022/03/22)
- 17 (GA 2021/09/14)
- 16 (GA 2021/03/16)
- 15 (GA 2020/09/15)
- 14 (GA 2020/03/17)
- 13 (GA 2019/09/17)
- 12 (GA 2019/03/19)
- 11 (GA 2018/09/25)
- 10 (GA 2018/03/20)

Resources

- Development list: *jdk-dev*
- Main-line code repository: <https://github.com/openjdk/jdk/>

CS 253: Parallel Functional Programming with Java & Android: Overview (Part 3)