

# High-level picture of PolyScope 6 vs PolyScope 5 development

## Document Changelog

Version	Owner	Change
0.1	THOA	Document created
0.2	JBM	Added Development Environment and Key Changes list
0.3	JBD	Added diagram, tweaked text

## Development Environment

The PolyScope 6 URCap development environment is provided through two artifacts;

- URCapX-dev, a development environment container
- URSim, a containerized PolyScope 6 simulator

The **URCapX-dev container** is a Docker-image containing the URCap SDK, URCap-related software environments (Python, Java, etc.), as well as relevant build-tools.

It is a self-contained package with all the relevant SDK specific tooling you need to develop URCaps independent of your host Operating System.

**URSim** is a simulator of the PolyScope 6 software, packaged in a Docker image including the same software architecture that is present on the real robot. It simulates both the behavior of the frontend (PolyScope) as well as the backend-behavior (ability to install and operator Docker containers). Compared to previous URSim virtual machines, the container is a more light-weight option, and you can use a browser (or VNC) to interact with the PolyScope GUI.

You need two prerequisites on your local computer to get started with the new SDK artifacts:

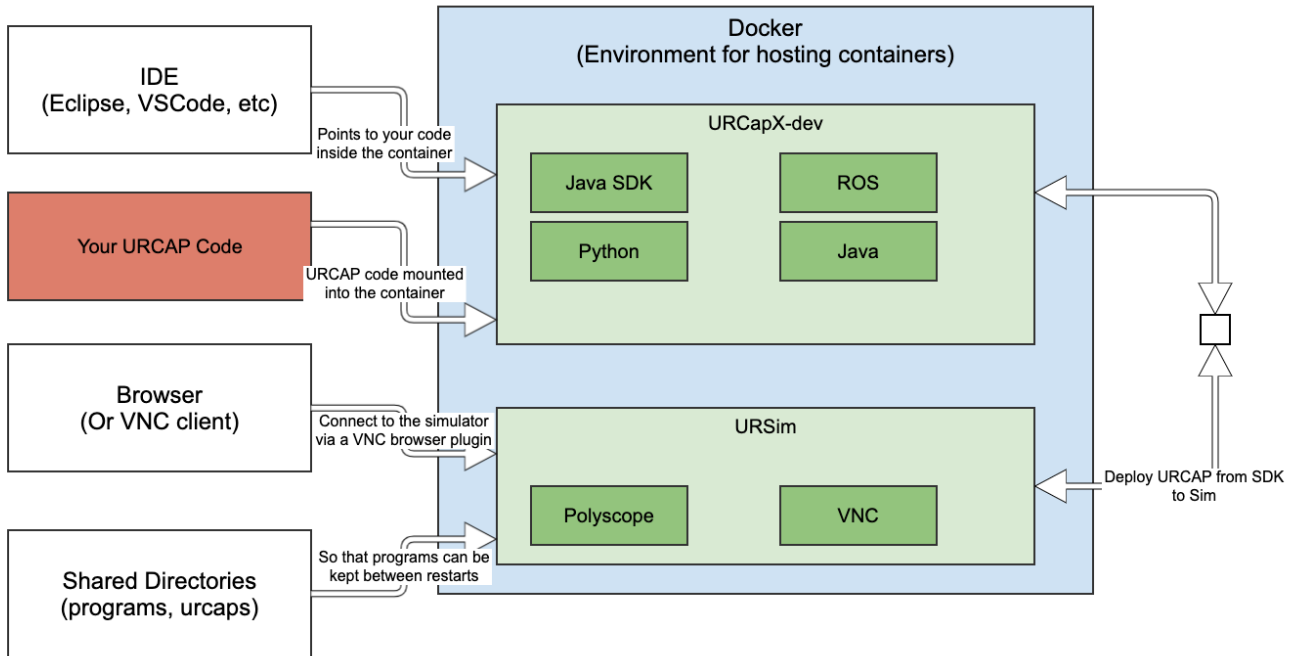
- **Docker**  
This is used to register and spin up the two above containers. Ex. [Docker Desktop](#).
- **Your favorite IDE**, we recommend [VS Code](#)  
To develop your URCap code, and optionally to run build commands through dev-containers in VSCode.

Our setup and getting started guides will instruct you on how to install and spin up the two containers.

Your URCap source code, robot programs, built URCaps, and other user-specific data will be stored in folders on your local machine. They are mapped into the relevant containers as shared directories.

This also makes it easier, when updating to new versions of the URCapX-dev environment or URSim, as your source code and robot programs will remain in the local directories, and simply be mapped into the newly installed version of the containers.

Please refer to below schematic for a conceptual overview:



## Changes in PolyScope 6

Migrating from PolyScope 5-style URCaps to URCaps compatible with PolyScope 6 require two main activities:

- 1. Recompiling the URCap to the new URCapX-format**  
The packaging format of URCaps is changed from the ".urcap" JAR-archive, to a new ".urcapx" ZIP-based architecture.
- 2. Migrating any daemon services into a containerized daemon**  
Access to the core robot Operating System is restricted, and any existing backend services (daemons) must be ported to run inside the newly added Container contribution framework.  
This step is only relevant, if your existing URCap has a daemon.

Key changes in PolyScope 6 (compared to PolyScope 5.13):

- **New URCap format: URCapX.**  
URCaps will be packaged in a new, extensible format: URCapX. This zip-archive format will contain existing Java OSGi-bundles (jars, like in PolyScope 5, but omitting daemons), as well as docker-images for backend services. The format allows extending what URCaps can embed in the future.
- **URCap backend services (daemons) are run in Docker containers.**  
This reduces URCap reliance on the OS, avoids conflicts between multiple URCaps, and gives freedom of technology choice for developers. Recommended base images will be included as part of the SDK
- **Updated OS, 64-bits, Python, C++ compiler, and new Java LTS.**  
The OS is upgraded to 64-bits Debian 11, Java and thus the URCap frontend will follow the latest LTS version (Java 17), and base-images for new Docker containers will use new Python 3 and C++ compiler versions.
- **URScript, UR Programs (URPs), and Installations are compatible.**  
Existing applications remain compatible, yet may require updated URCaps following new URCapX-format.
- **Old URCaps will no longer be compatible.**  
PolyScope 6 will not support the installation of old URCap formats (.urcap). Only URCapX formats.  
The upgraded OS means that existing daemons and backend services are unlikely to work with the upgraded platform. Existing URCaps must be recompiled to use the new URCapX-packaging.



- **New URCap development environment and SDK packaging**

To ease URCap development, the URCap SDK is now packaged as a development-container. This makes it easier to setup your development environment across Windows, Mac, or Linux, and work in your favorite IDE against the Dockerized development-environment. The URSim is also packaged as a Docker image, making it easier to simulate PolyScope from your local machine.

**CONFIDENTIAL**