Bazel is a build system built with Java.

**But why is Bazel better than Gradle or Maven?** It’s faster if used right. Also, you can have multiple languages within the same project, and Bazel will handle this as well.

**You can also use Bazel in combination with Guice for more lightweight projects.**

Also if working with microservices, Bazel is the right choice. You can limit visibility, have a more fine-grained build structure, and create your own build rules.

Bazel uses [remote caching](https://bazel.build/docs/remote-caching). The outputs can be reused, cached, and make faster builds.

**For some teams, Bazel may not be the right choice.**

The remote cache was a deterrent for the Kubernetes team. [They needed fresh output each time](https://github.com/kubernetes/enhancements/tree/master/keps/sig-testing/2420-reducing-kubernetes-build-maintenance#drawbacks). Even so, for common use cases, this works like a charm.

You can also use Starlark and with meta-programming create a macro to compose build rules. This can help with code reuse but also generate custom build rules.

**Bazel makes builds more versatile, configurable, and transparent.**