

Transitioning a Large Codebase to BAZEL

Lessons learned after a year of implementation

BENEDIKT WÜRKNER
MAY 22, 2025



Only Three Steps Required

01 Convince

Management Quality Engineering

02 Execute

From Proof of Concept to production

03 Support

Before During After

Convince

Management

- » Create comparison page
 - explain the differences between the current and the Bazel system
- » Less execution and NOT faster execution
- » Reproducibility and reduced workspace complexity
 - Correlate this with lead time for new engineers/projects
 - Multiply number of engineers times hours saved
 - extremely convincing when there are a lot of engineers

Quality

- » With hermeticity
 - Being able to proof what went into a release
- » With reproducibility
 - Being able to proof what was executed and re-do it at any point in time
- » With local test execution
 - Enabling engineers to test complex things routinely locally

Engineering

- » Developers
 - With reduced workspace complexity
 - With removal of differences between CI and local setup
- » Architects
 - With clear definition of dependencies
 - With the powerful query language
 - With the plotting features

Execute

- » Start with a hermetic and air-gapped setup, force buildifier as quality gate and any additional linting from the first day
 - There'll never be a better day than day 0 for this
- » Identify all designs that conflict with BAZEL conceptually (walking directories, config files defining paths to files globally, ...)
 - Create mechanism to mitigate this by writing a rule to update the config files or inject paths for BAZEL via ENV variables
 - Identify central pieces of the architecture and the relevant teams that may need to update something to work with BAZEL
- » Don't let yourself be blocked by teams not being able to deliver BAZEL compatible solutions quickly.
 - Work around them with bazel rules, genrules, anything to achieve your overall goal to get everything into BAZEL
 - Clearly mark why something was done in the workaround way to eventually get rid of it
 - Better to have something in BAZEL in a genrule than not be able to use sandboxing and caching at all
- » Move parts into production gradually, when possible, run in parallel where necessary, let teams take over slowly to build confidence
 - Always remember to be transparent about the progress to not become the scapegoat for other issues
- Perfect is the enemy of good
- Everything working at low efficiency is better than a few things working at 100% efficiency

Support

Before

- » Document plan in clear "how does this affect whom" terms
- » Create a Q&A mechanism that's visible to everybody
- » Document all tests and investigations performed
- » Document alignment meetings with different parties
- » Write instructions how to start from scratch as early as possible and keep them up to date with any changes
 - We checked our own instructions monthly by setting up a workspace from scratch ourselves

During

- » Keep status communicated transparently
- » Keep the Q&A up to date
- » Support any team/person that wants to try out Bazel as well enthusiastically – these are the best multipliers
- » Document additional requests clearly and the appropriate solution strategy where possible
- » Listen and address concerns while documenting them and all responses

After

- » Be responsive to enquiries by teams
- » Be proactive in supporting, don't just send links to documentation.
 - Improve documentation as needed
- » Encourage teams to continue to ask questions, keep updating the Q&A as well

Thank you for the attention

Some Code Examples

- » Enable tools that perform in-place modifications of input by copying the input to the output and then `cd` into that folder
 - `mkdir -p $(RULEDIR)/output && cp -fRL $(location //input:file)/* $(RULEDIR)/output`
- » Zip output of tools that have unspecified numbers of files and pick folders from that afterwards
 - Given a folder “wildcard_output” that contains any number of files from a genrule execution
 - `$(location @bazel_tools//tools/zip:zipper) c $(location wildcard_output.zip) $$ (find wildcard_output -type f)`
- » More examples can be found here: <https://github.com/Ecklebe/bazel-playground/blob/main/BUILD.bazel>