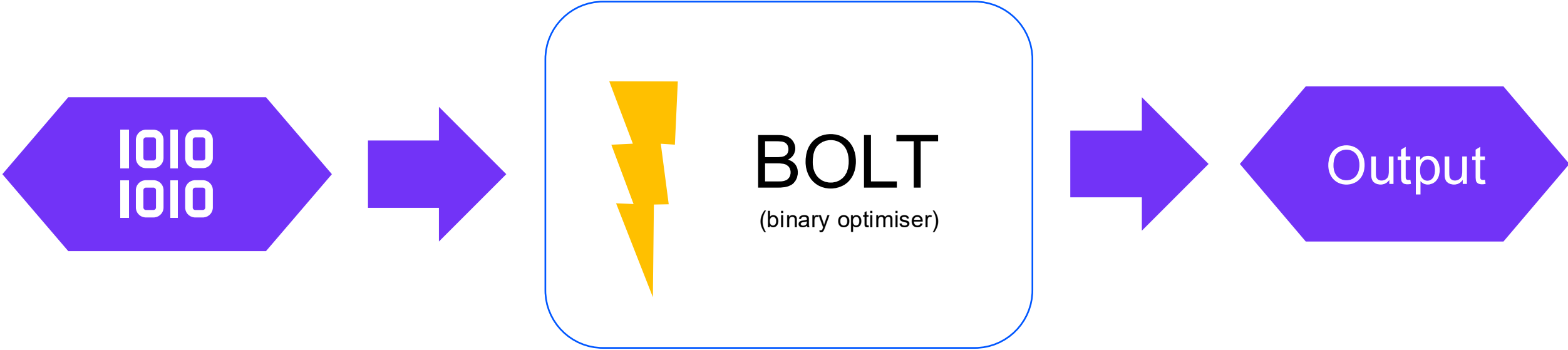


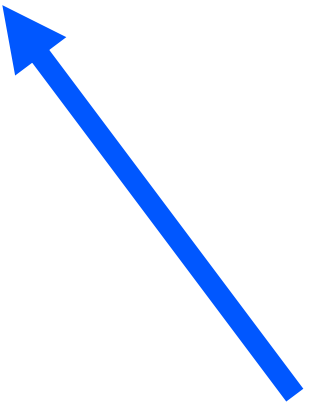
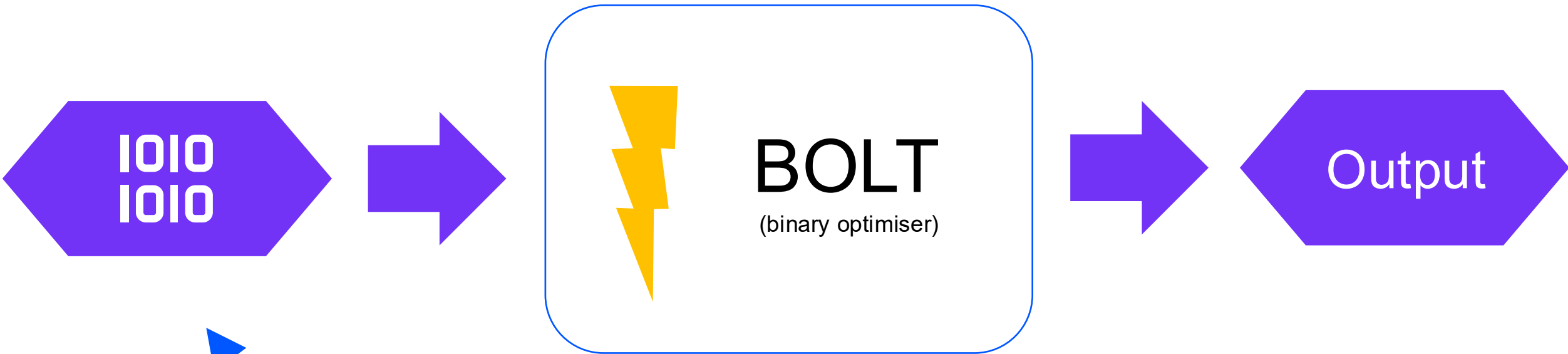


Reproducible Large-Workload Recipes for BOLT with Nixpkgs

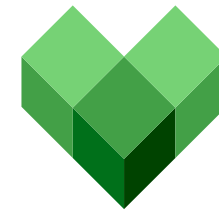
Peter Waller

2026-04-15 | EuroLLVM2026





Tricky to make, share and maintain in the face of change.



Bazel





Defines an execution environment



Defines a build system



Defines a software distribution



- Defined as **build-from-source**.
- **Reproducible build-inputs** via caching.
- Defines a large, **up to date** software distro.
- Enables **diffing** the build input across the whole distro over time.
- **Extendable** with methods that apply to many packages at once.
 - Basic example: “link packages with `--emit-relocs`” for BOLT.



Concretely:

```
nix build nixpkgs/9dcb002ca1690658be4a04645215baea8b95f31d#chromium
```

Contains enough information to define an exact chromium build.
(And to acquire a pre-built in a minute or so!)

With override mechanisms, we can specify details
of the build such as versions or compile and link flags.

arm

Call to action!

- **Roundtable:** “Maintaining Reproducible Test Binaries”
- **14:15 today.**
- What open-source methods can LLVM use to define and maintain definitions of non-trivial binary inputs?

arm

Tack

ಧನ್ಯವಾದಗಳು

Merci

Danke

Gracias

Grazie

谢谢

ありがとう

Asante

Thank you

감사합니다

धन्यवाद

Kiitos

شكرًا

ধন্যবাদ

תודה

ధన్యవాదములు

Köszönöm