Verification as a Service

COOP ’23 – Paris

Henrik Wachowitz

April 23, 2023

LMU Munich, SoSy-Lab

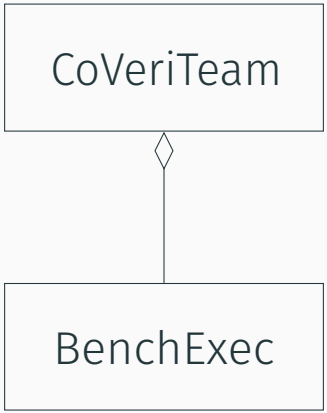
Software Verification is hard.

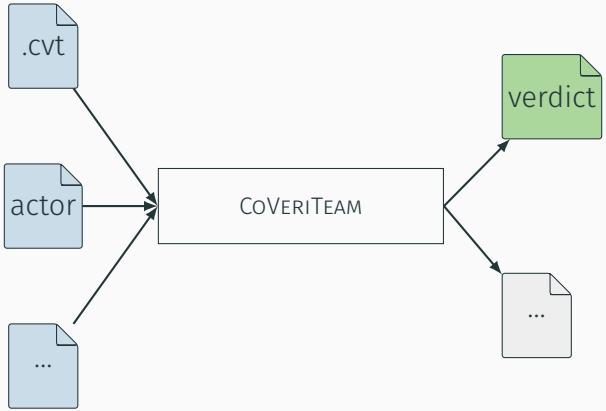
Software Verification is hard.
Using the Tools is harder.

Solutions?

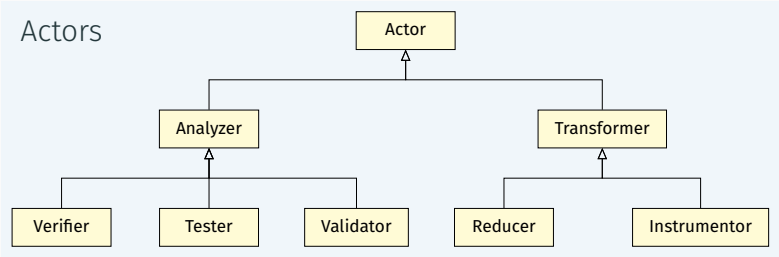
BenchExec

Solutions?



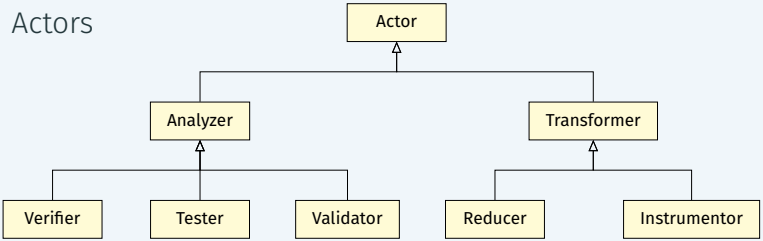

.cvt
actor
...
CoVeriTeam
verdict
...

Domain Model

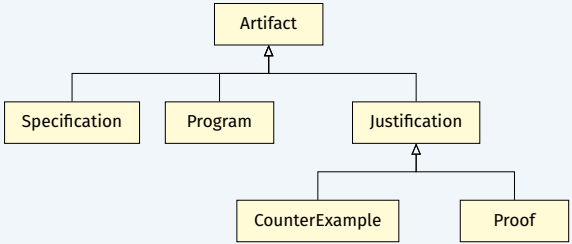


Domain Model

Actors

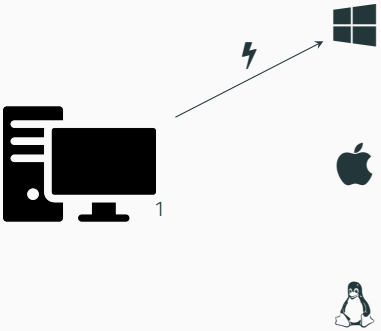


Artifacts

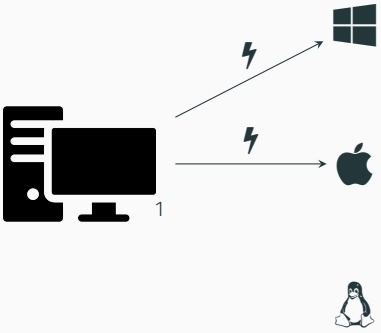


That’s great, but…

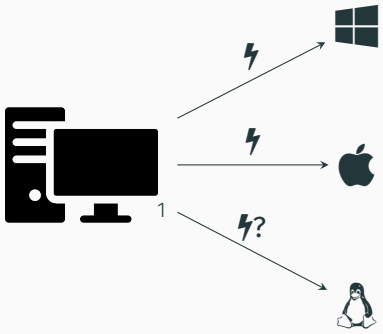
But the System?



But the System?



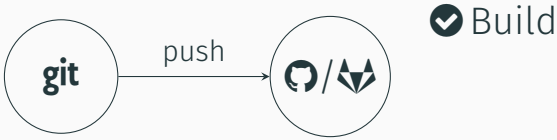
But the System?



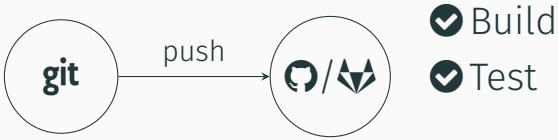
[1]Image: Flaticon.com

Decouple the *System* from the *Tool*.

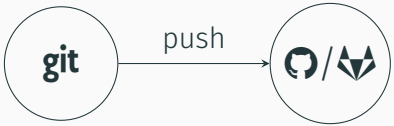
Modern Workflows



Modern Workflows



Modern Workflows

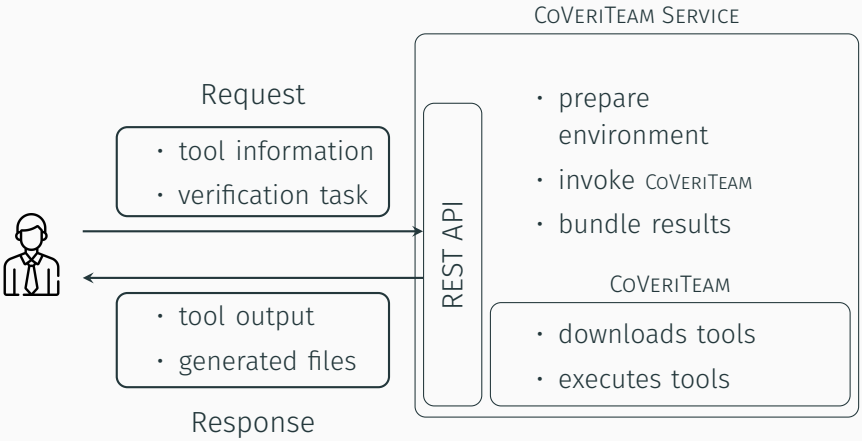


- Build
- Test
- Verify?

Decouple the *System* from the *Tool*.
Make Interaction *Machine Friendly*.

Decouple the *System* from the *Tool*.
Make Interaction *Machine Friendly*.

CoVeriTeam Service


CoVeriTeam Service
Request
• tool information
• verification task
REST API
• prepare environment
• invoke CoVeriTeam
• bundle results
CoVeriTeam
• downloads tools
• executes tools
• tool output
• generated files
Response

Request something

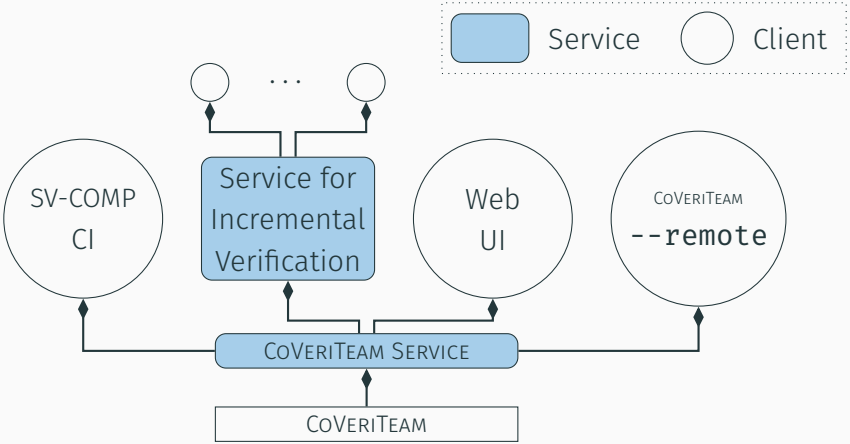
```
curl \
  --form "args=<listing3.json" \
  --form cpachecker.yml=@cpachecker.yml \
  --form test02.c=@test02.c \
  --form verifier.cvt=@verifier.cvt \
  --form unreach-call.prp=@unreach-call.prp \
  --output cvt_remote_output.zip \
  https://coveriteam-service.sosy-lab.org/execute
```

Arguments

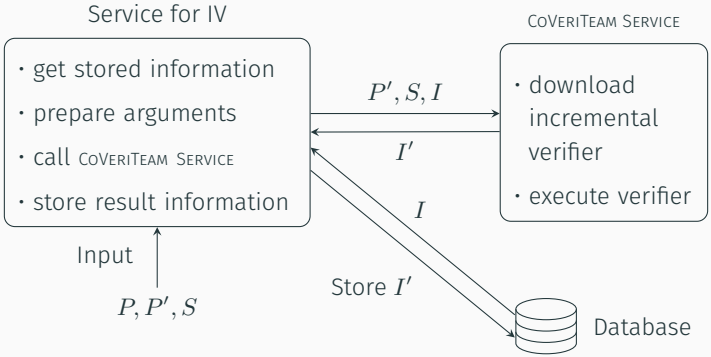
```
{
  "cvt_program": "verifier.cvt",
  "coveriteam_inputs": {
    "verifier_path": "cpachecker.yml",
    "program_path": "test02.c",
    "specification_path": "unreach-call.prp",
  },
  "working_directory": "coveriteam/examples",
}
```

Let’s see it in action!

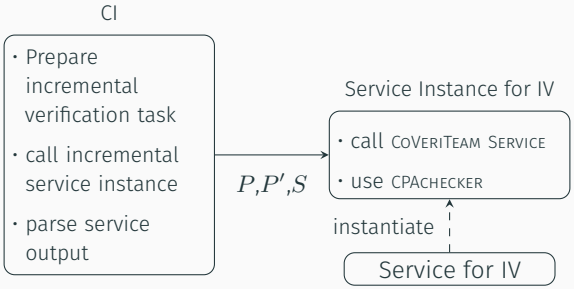
Going Beyond



An Incremental Service



Integrating It



Is this the end?

Is this the end?
Obviously not!

User Stories

AS CoVeriTeam User
I WANT TO specifiy to use ≤ 10 GB Memory and only open source software
SUCH THAT I know that the verification run conforms to my required policies.

User Stories

AS CoVeriTeam User
I WANT TO specifiy to use ≤ 10 GB Memory and only open source software
SUCH THAT I know that the verification run conforms to my required policies.

AS Tool Developer
I WANT TO require 3 GB of memory for my tool and more than 90 s runtime
SUCH THAT my tool can produce meaningful results.

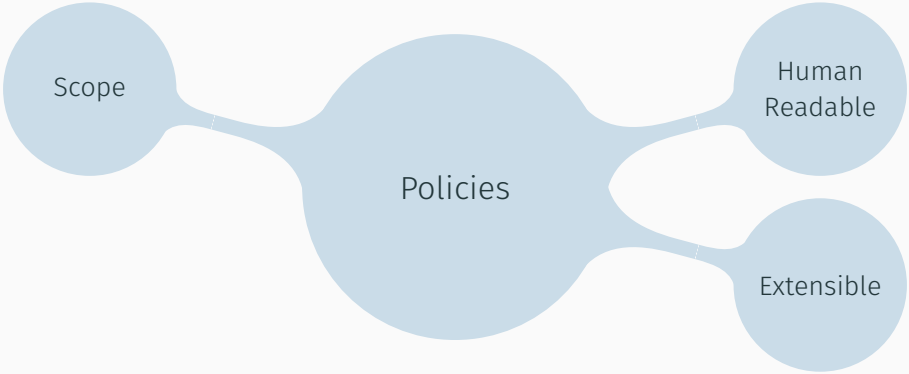
User Stories

AS CoVeriTeam User
I WANT TO specifiy to use ≤ 10 GB Memory and only open source software
SUCH THAT I know that the verification run conforms to my required policies.

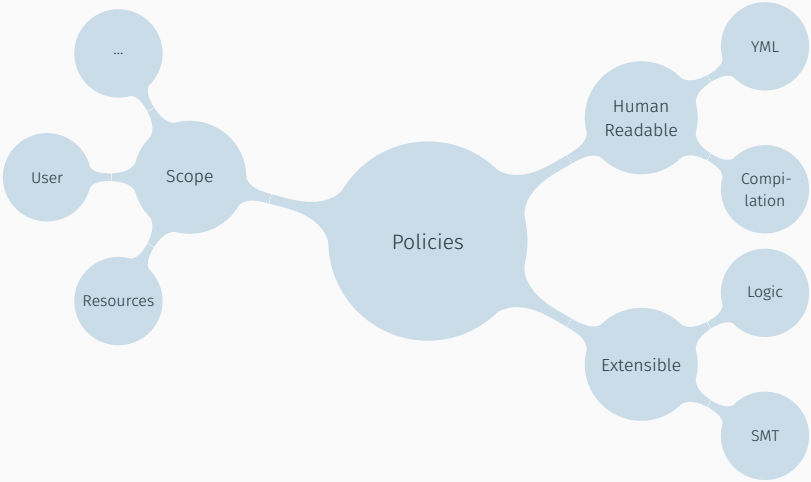
AS Tool Developer
I WANT TO require 3 GB of memory for my tool and more than 90 s runtime
SUCH THAT my tool can produce meaningful results.

AS CoVeriTeam Service Host
I WANT TO rate limit submitted tasks by users
SUCH THAT that the ressources are sufficient for everyone

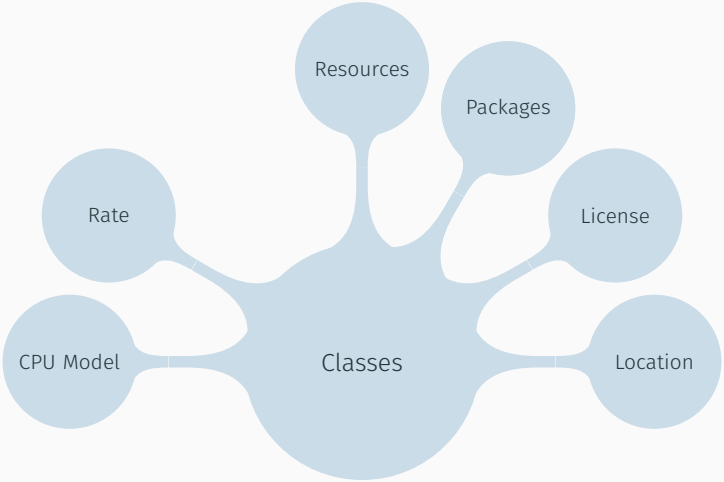
Requirements



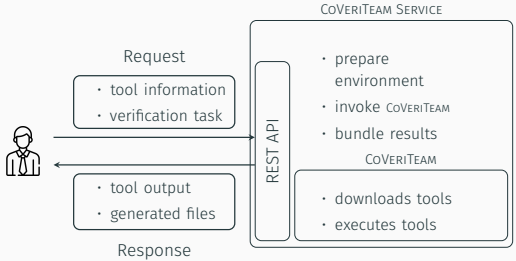
Requirements

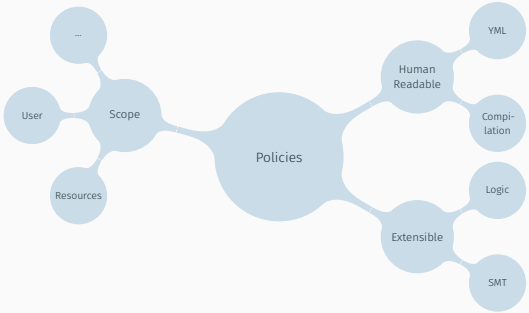


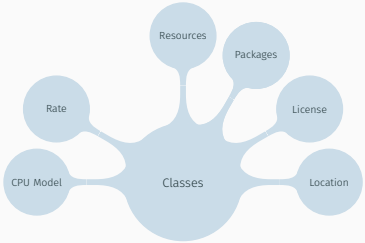
Possible Classes



Conclusion


CoVeriTeam Service
Request
· tool information
· verification task
REST API
· prepare environment
· invoke CoVeriTeam
· bundle results
CoVeriTeam
· downloads tools
· executes tools
· tool output
· generated files
Response


Policies
Scope
...
User
Resources
Human Readable
YML
Compi-lation
Extensible
Logic
SMT


Classes
Resources
Packages
Rate
License
CPU Model
Location