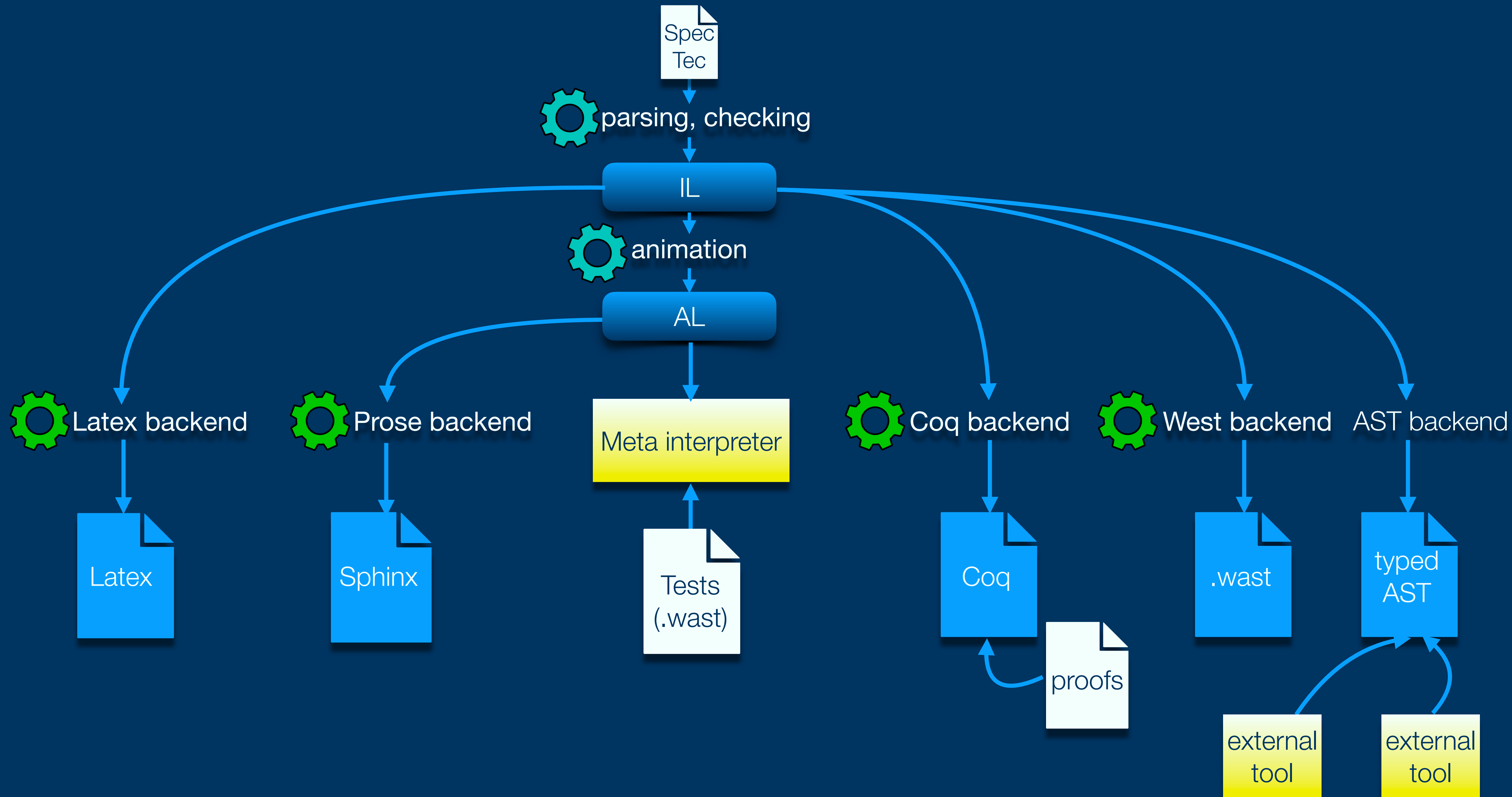


# SpecTec

## Update and Poll

**Andreas Rossberg**

with Dongjun Youn, Wonho Shin, Jaehyun Lee, Suhyeon Ryu, Hyunhee Kang, Sukyoung Ryu,  
Alan Liang, Sam Lindley, Matija Pretnar, Joachim Breitner,  
Rao Xiaojia, Diego Cupello, Philippa Gardner, Conrad Watt



# Status

converted primary chapters of the spec document (syntax, validation, execution, binary)

some remaining polishing of generated prose

meta-interpreter passes 100% of applicable tests

in sync with wasm-3.0 branch: all phase 4/5 proposals (modulo threads)

user documentation

*new:* AST backend for interfacing with external tools

# Value Proposition

Less work and better support for spec & proposal authoring

Higher level of assurance of document correctness

Machine-readable representation of syntax and semantics for tooling

More tractable maintenance of machine-verified correctness proofs

Soon: Automated, semantics-driven test generation

# Limitations

user experience could still be improved

- ...especially wrt assumptions and error reporting in backends

produced prose can be polished further

prose and interpreter hard-code spec choices that may break when spec changes

- ...often inevitable, e.g., must convert from reference interpreter's AST

- ...interpreter requires adjustment for structurally non-trivial spec extensions

- ...hope to get rid of more of these over time, e.g., with hint system, meta-parser

## Poll from last June

Adopt SpecTec as the toolchain for authoring the spec (once it's ready) 

# Poll

Adopt SpecTec as the toolchain for authoring the spec

# Why Now is a Good Time

Sufficient [parity](#) with hand-written document

...superior in correctness, only slightly inferior in prose quality, fewer open todos

All-time low of [outstanding proposals](#) that have already entered the spec'ing phase

...minimises friction from switching tool chains

Getting there was a pain! Need to [maintain](#) two very different variants of the spec

...not viable too much longer, unrealistic to backport all fixes

We have peak momentum and [manpower](#) right now for handling transition and support

Further development should benefit from [user feedback](#)

Prerequisite for automated [test generation](#) with West



# Proposed Setup & Process

Move SpecTec to [main repo](#)

...that way, proposal forks have it as well and can tweak it as needed

Adjust spec build to [invoke SpecTec](#)

Set up CI to also [run SpecTec meta-interpreter](#) on test suite

...but document an easy way to temporarily disable on forks, to catch up later

File SpecTec [issues and feature requests](#) on main repo

...we will triage and address depending on urgency and complexity

...can also help on proposal forks when needed

# Proposed Steps

1. Upstream SpecTec back to main (separate branch at first)
2. Set up CI and GH pages for both versions
3. Finish remaining polishing
4. Eventually, pull trigger to merge into main and replace hand-crafted doc

Could happen in sync with merging 3.0 spec into main

...at this point, there are fewer relevant todos in SpecTec document

# SLA

Continue to [fix](#), [improve](#), and [extend](#) the tool

Continue to incrementally [convert](#) the remaining parts of the spec

Available to [maintain](#) the tool for at least 2 years

- ... react to Github issues within reasonable time frame

- ... assist proposal authors where necessary

- ... knowledge transfer

Invest into building more tooling

- ... [test generation](#) backend

- ... [proof mechanisation](#) backend(s)

# Resources

[KAIST](#) (Sukyoung Ryu) with several students doing PhD around SpecTec

...test generation and SpecTec tooling/applications

[Imperial](#) (Philippa Gardner) with two new students doing PhD/MSc on Wasm mechanisation

...Coq proofs for SpecTec-generated semantics

[NIS](#) (Conrad Watt) with two new PhD student/RA's working on SpecTec backend

...Coq/Isabelle automation and pipeline improvements

Joint [U.K. grant proposal](#) by Imperial & Edinburgh (Sam Lindley) and NIS for two post-docs

...further research on SpecTec

# Longer-Term

On-board members from [community](#)

- ...for spec [authoring](#) and editing (SpecTec [users](#))

- ...for tool [maintenance](#) (SpecTec [developers](#))

Envisioning [gradual transition](#), from simple to more complex contributions

- ...we can provide guidance and support

In parallel, we are [invested](#) in improving and extending SpecTec

# Post-MVP

Replace more hard-coded **assumptions** in backends with customisable hints

More generation for doc: macro definitions, xref anchors, index entries

More **self-contained meta interpreter**: meta decoding, validation, numerics

Another meta interpreter operating on reduction rules directly

**Meta-theory** of SpecTec (SpecTec in SpecTec?)

Other **applications** of SpecTec — broaden the user base

# Poll

Adopt SpecTec as the toolchain for authoring the Wasm spec.