Node.js Technical Steering Committee (TSC) Meeting 2020-05-21

Links

* **Recording**:
* **GitHub Issue**: https://github.com/nodejs/TSC/issues/868

Present

* Anatoli Papirovski @apapirovski (TSC)
* Anna Henningsen @addaleax (TSC)
* Сковорода Никита Андреевич @ChALkeR (TSC)
* Gabriel Schulhof @gabrielschulhof (TSC)
* Matheus Marchini @mmarchini (TSC)
* Michael Dawson @mhdawson (TSC)
* Rich Trott @Trott (TSC)
* Ruben Bridgewater @BridgeAR (TSC)
* Sam Roberts @sam-github (TSC)

* Beth Griggs @BethGriggs (TSC)
* Colin Ihrig @cjihrig (TSC)
* Shelley Vohr @codebytere (TSC)
* Daniel Bevenius @danbev (TSC)
* Franziska Hinkelmann @fhinkel (TSC)
* Gireesh Punathil @gireeshpunathil (TSC)
* James Snell @jasnell (TSC)
* Joyee Cheung @joyeecheung (TSC)
* Matteo Collina @mcollina (TSC)
* Myles Borins @MylesBorins (TSC)
* Michaël Zasso @targos (TSC)
* Tobias Nießen @tniessen (TSC)

Agenda

Announcements

CPC and Board Meeting Updates

*Extracted from **tsc-agenda** labeled issues and pull requests from the **nodejs org** prior to the meeting.

nodejs/node

* \[Regression in 14.1.0 - Windows\] `stdout` is sometimes empty
[#33166](https://github.com/nodejs/node/issues/33166)
  * Agreed this could be removed from Agenda.

* src,win: Replacement of unsupported versions of Windows runtime exit.
[#33108](https://github.com/nodejs/node/pull/33108)
* Dispute of "block running on EOL Windows versions #31954" - Alternative Suggestion
[#33034](https://github.com/nodejs/node/issues/33034)
  * Looks like it's progressing in https://github.com/nodejs/node/pull/33176, nothing
    to discuss in the meeting.

* process: Throw exception on --unhandled-rejections=default
[#33021](https://github.com/nodejs/node/pull/33021)
  * Survey is ready, going to be merged after the meeting
  * Agreed 2 weeks as minimum time.
  * Past surveys we had a couple of thousand replies

* bootstrap: include bootstrapped Environment in builtin snapshot
[#32984](https://github.com/nodejs/node/pull/32984)
  * Joyee and Anna had concurrent PRs, Anna closed hers (but not happy
    on how review process went). James may try to resurrect, PR.
  * Lets defer the discussion until James is here.
    * Joyee created detailed design doc.

nodejs/admin

* Approval to enable github pages for node-addon-examples
[#503](https://github.com/nodejs/admin/issues/503)
  * Should have removed this after discussion last week. Nothing to discuss.

* Audit Google account access [#389](https://github.com/nodejs/admin/issues/389)

Strategic Initiatives

* Startup time
- Relanding the blocking v8 patch https://chromium-review.googlesource.com/c/v8/v8/+/2212085
- https://github.com/nodejs/node/pull/32761 was closed though the technical disagreement is
still not settled (for differences between https://github.com/nodejs/node/pull/32761 and
https://github.com/nodejs/node/pull/32984, see the discussions in
https://docs.google.com/document/d/15bu038I36oILq5t4Qju1sS2nKudVB6NSGWz00oD48Q8/e
dit?usp=sharing, would love more input there to break the tie). As @jasnell mentioned in

https://github.com/nodejs/node/pull/32761#issuecomment-628972927 I think this reveals that we have some issues with the current process. Some points that I can think of
  - There should've been a better way to communicate about active efforts to avoid duplication of work and competition. In this case there were occasional updates in the tracking issue https://github.com/nodejs/node/issues/17058, related PRs and meetings, but those were not visible enough and still lead to conflicts
  - We need more reviewers for certain tasks or certain subsystems, ideally more than two people (to break the tie when there are technical disagreements) and from more than one party (to avoid bias). I don't think it's a good idea to always escalate these to TSC, because not all of us are interested in or familiar with the same part of the code base, and so we don't always get the attention we ask for if the disagreement involves substantial amount of reading or context. Maybe a team + CODEOWNERS structure can help
  - When the change involves substantial amount of work, maybe starting a design doc first to seek high-level and more abstract inputs before jumping in to do the actual implementation would be a good idea. I can see this may be a double edged sword but this works relatively well in V8 (though sometimes prototyping is still necessary to keep the technical discussion realistic)

* No updates for:
  * Promises
  * V8 Currency
  * Build resources
  * QUIC
  * Modules

Upcoming Meetings

* **Node.js Foundation Calendar**: https://nodejs.org/calendar

Click `+GoogleCalendar` at the bottom right to add to your own Google calendar.