WEBRTC API in Chrome

Implementation experience

Harald Alvestrand

Our goal

- Make the web browser the best RTC platform

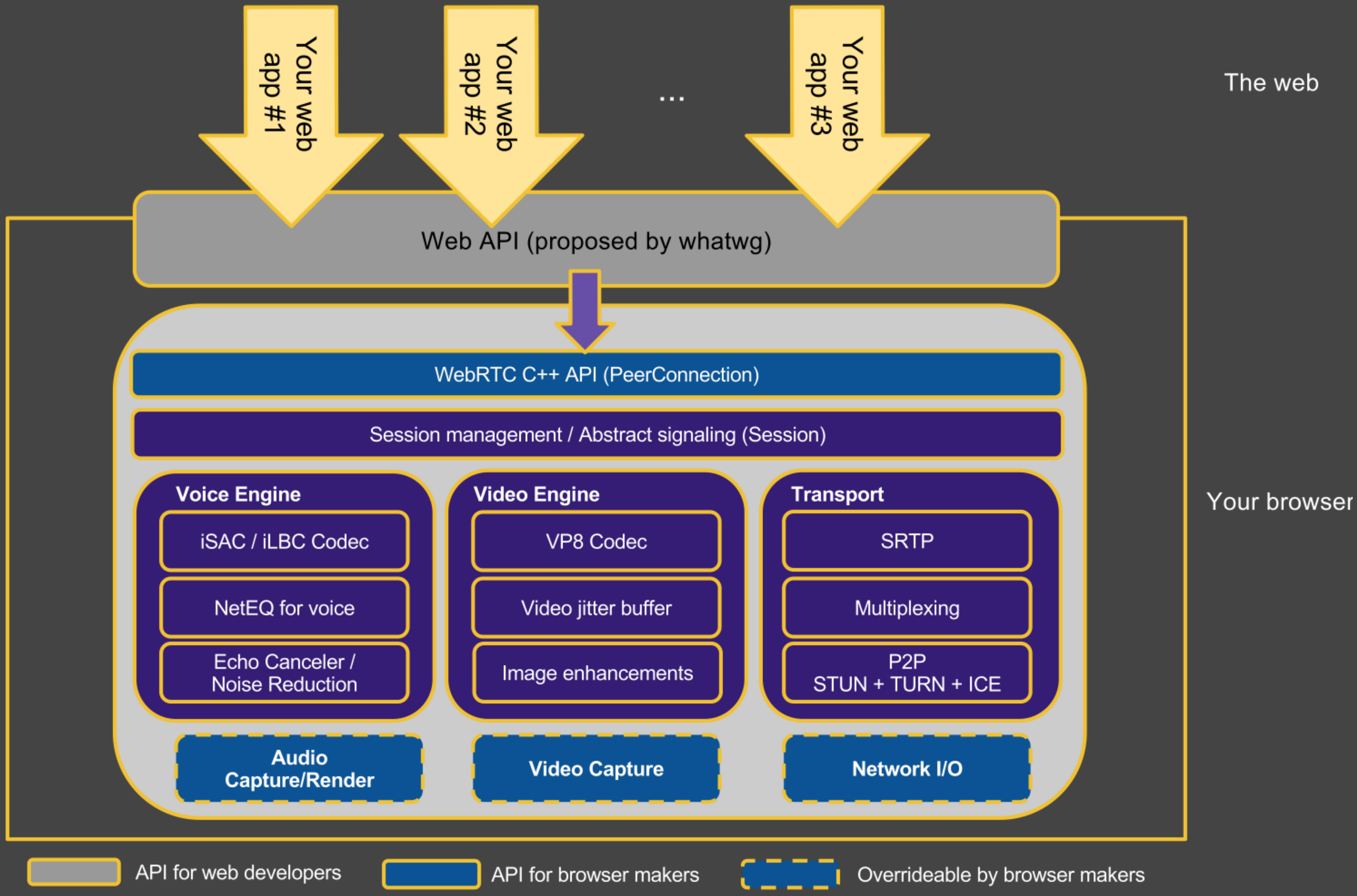
To achieve this:

- Provide a fully functional, production quality implementation of the WEBRTC API and RTCWEB protocol stack
- Use our implementation to provide concrete feedback for APIs and protocols
- Use existing, proven code. No wheel re-inventing.

What we've done so far

- Released key components as open source on code.webrtc.org
- Integrating in Chromium and assisting others
- Improved speed and performance on components

Overview of the webrtc.org package



Implementation strategy

- Build on GIPS VoiceEngine / VideoEngine
  - Production, high quality codebase
  - Used by hundreds of products
- Reuses RTP / ICE / STUN functions from libjingle
  - Tested and proven library
  - Used in Hangouts and Gmail Video Chat
- Define internal C++ API
  - API resembles JS API, but can't be identical
- Integrate with WebKit
  - Open review process
  - Functionality available to other WebKit browsers

Issues encountered

- Integration work is always hard
  - Adapting codebase to Google standards
  - Simultaneous changes to multiple libraries
- Review processes are tricky
  - WebKit is an established community. Style matters.
- API specification is severely unstable
  - Can't track each version
  - Some changes solve implementation problems
  - And other changes introduce implementation problems

Status of implementation

- We have had it working! (roll tape)
- Still rolling changes into other repositories
  - Libjingle
  - WebKit
  - Chromium
- Working in the open - really!
  - You'll have it as soon as we check it in!