

Chromium Feature Proposal

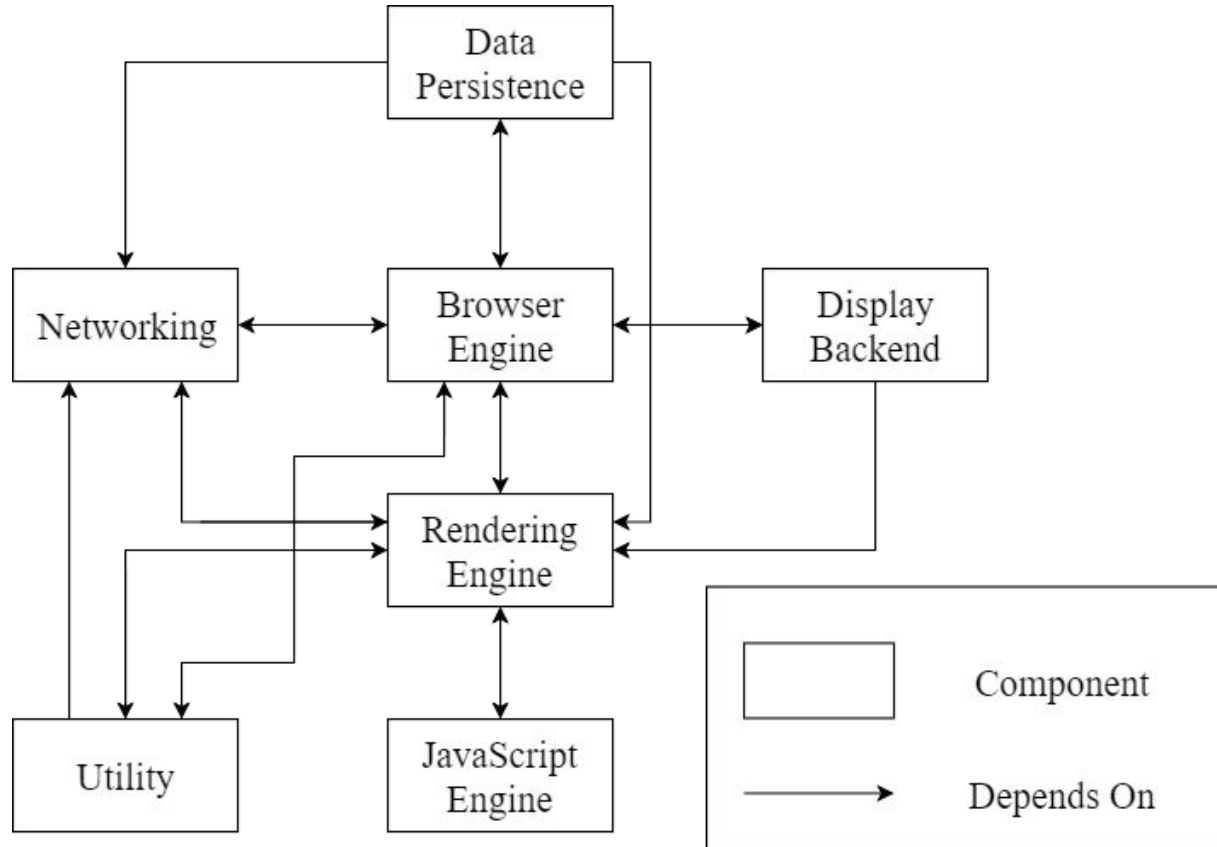
Intro

Cameron Raymond, Chris Molloy, Ross Hill,
Michael Wrana, Brenden Forbes, Brendan Russell

- Better synching for tabs across devices
- Implementation
- Impacted Subsystems
- Test Cases
- Limitations/ Team issues
- Lessons learned



Conceptual Architecture



Alternative Implementation

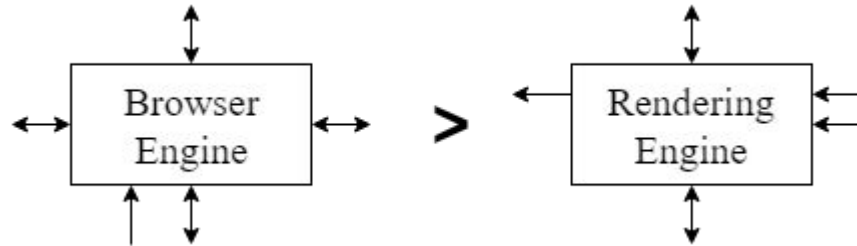
- Have Rendering Engine handle Chrome Sync
 - Direct communication with Javascript Engine
 - Useful for further adaptations where Sync captures more nuance in the webpage's state
- Reduces
 - Performance
 - Reliability
 - New dependencies needed between Rendering Engine and Data Persistence to read in webpage state

Decided Implementation

- Right now, the data that gets synced includes browser history, extensions, bookmarks, and logins
- For our feature, Chromium must also sync **webpage progress**:
 - Scroll distance from top of page and time remaining on videos
- Chrome Sync will therefore need webpage data from **Blink**
- Chrome Sync communicates with Google servers through **Network**
- We keep Chrome Sync in Browser Engine and pass webpage progress data between Blink and Browser Engine
- By keeping Chrome Sync out of Blink we maintain Blink's cohesion

SAAM Analysis

- Stakeholders
 - Users
 - Devs
 - Investors
- SAAM



Impacted Subsystems

Sending updated webpage progress

- Browser Engine
 - Periodically requests page state updates from Blink and pushes the updates to Google sync servers if page state was sufficiently changed
 - The current page state is stored with Data Persistence
- Rendering Engine (Blink)
 - Blink returns page state (scroll distance and video progress) updates to Browser Engine

Impacted Subsystems cont.

Receiving updated webpage progress

- Browser Engine
 - Browser Engine receives an update from Google sync servers with an updated webpage state on a synced tab
 - Updated page state is saved with Data Persistence
 - The updated page state is sent to Blink
- Rendering Engine (Blink)
 - Blink receives an update page state from the Browser Engine and updates the page
- How Browser Engine interacts with Network is largely unchanged
- Browser Engine and Blink remain co-dependent in our architecture diagram

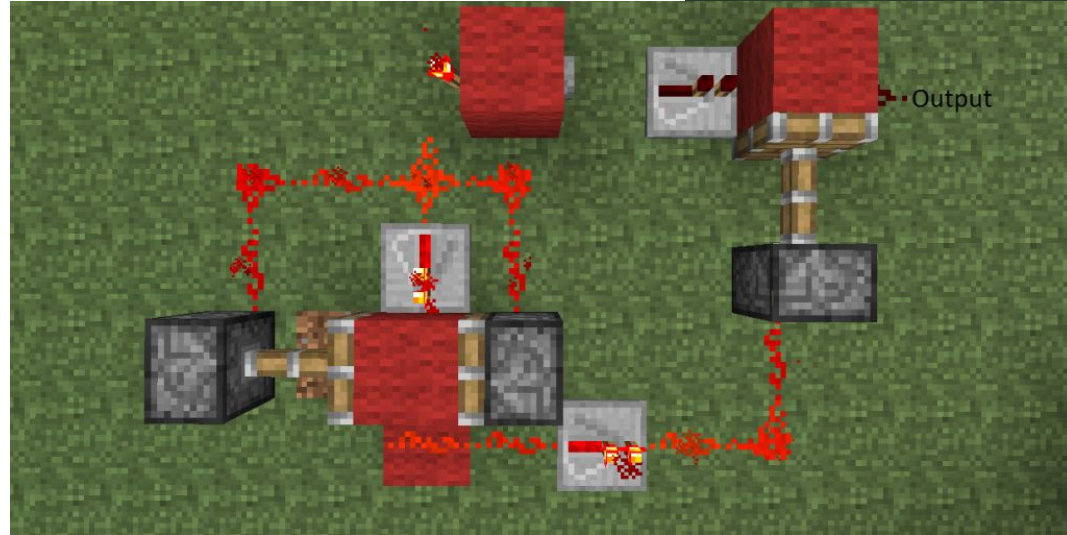
Test Cases

- Load a webpage
- Cross device sync
- Login/Logout
- Tab sync



Concurrency

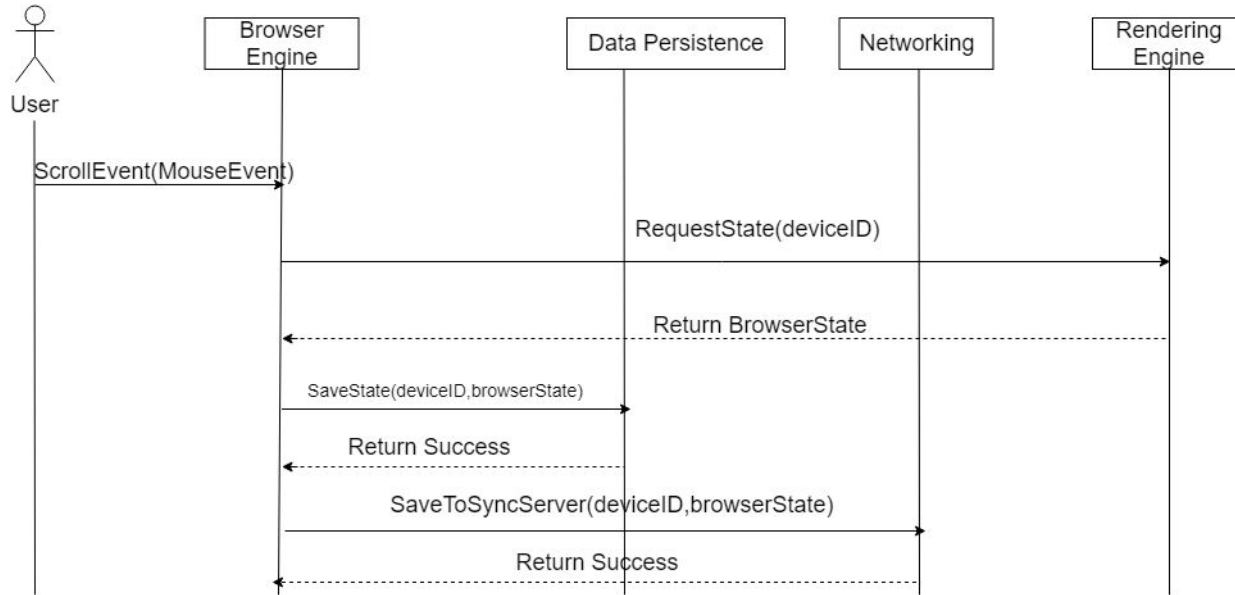
- Separate Renderers
- Implementation Details
- Callback
- Almost no Impact



Sequence Diagram

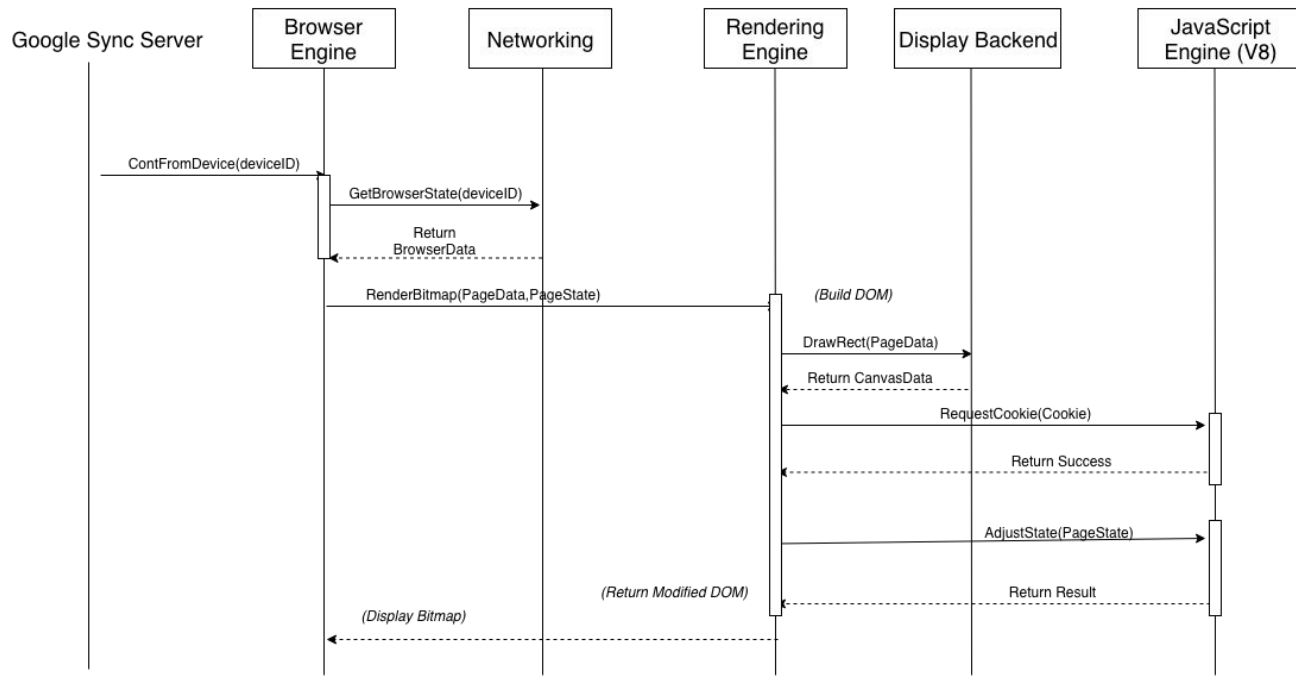
Web Page is scrolled

Legend



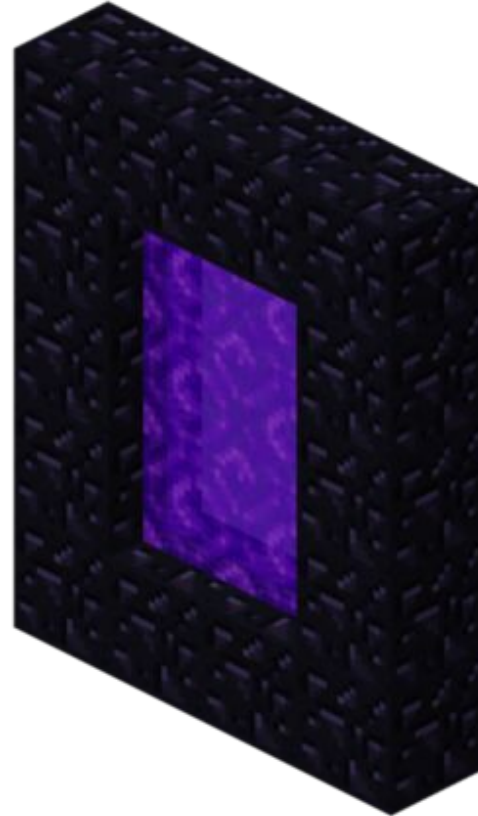
Retrieving State On Seperate Device

Legend



Potential Risks And Limitations

- Risks
 - Change
 - Annoyance
 - Security
- Limitations
 - JavaScript



Team Issues

- Developers in different parts of the world
- Increased coupling through more nuanced Sync
 - Maintenance
- What is the default device to Sync with?



Lessons Learned

- Does not necessarily alter architecture
- Many implementations
- Teamwork



Conclusion

- Better tab synching
- SAAM Analysis
- Test Cases
- Concurrency



— **THANKS FOR LISTENING!!!!**

