Site Isolation Summit:

Chromium Changes

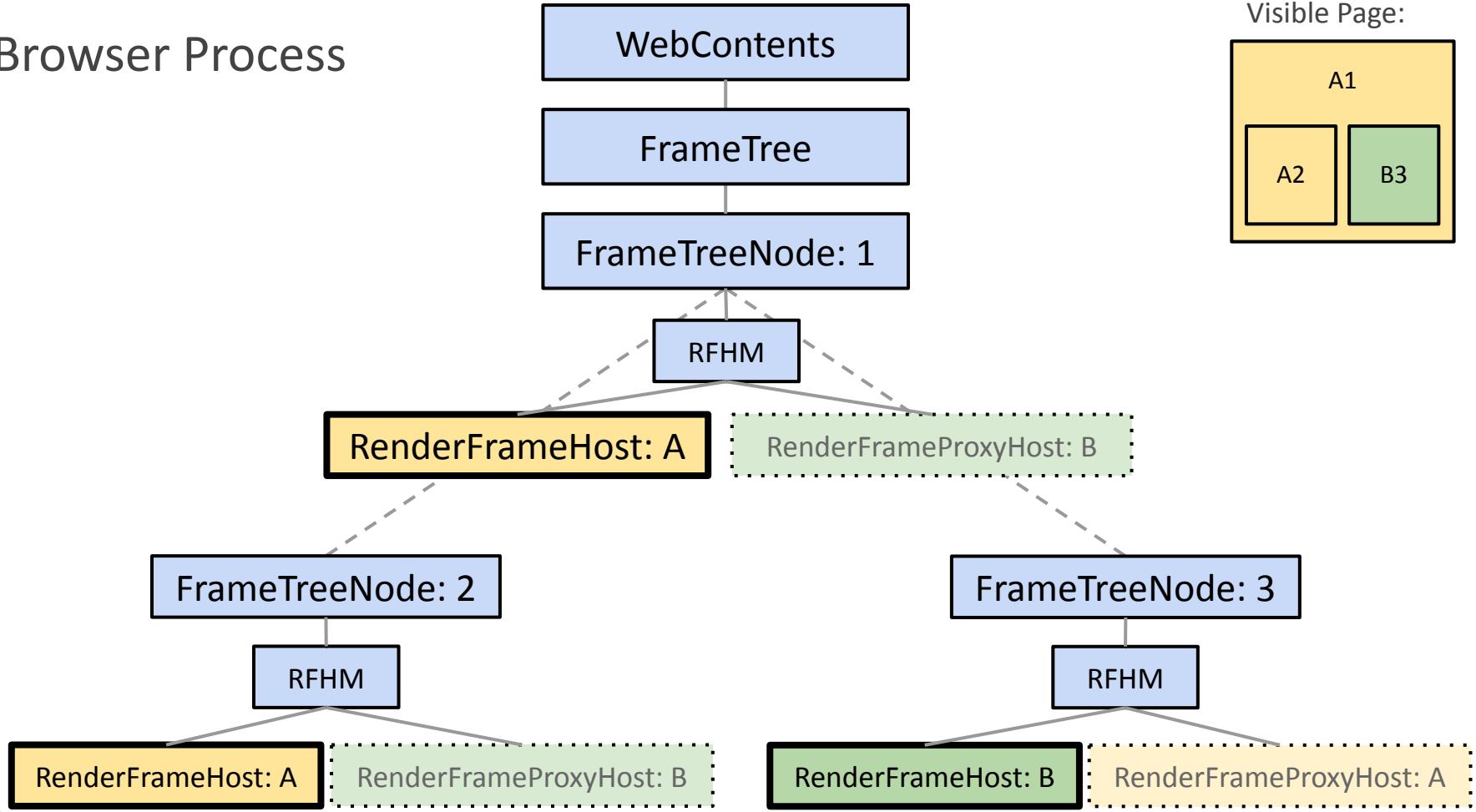
January 2015

What we're changing

Frames are first class objects

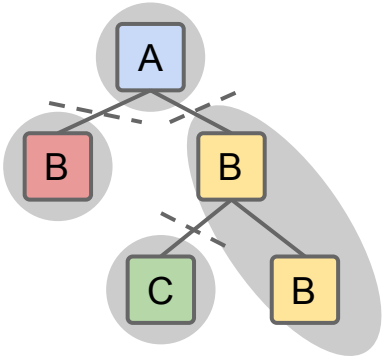
- `FrameTreeNode`s in browser process
- `RenderView*` --> `RenderFrame*`
- Frames tracked by navigation, process model, etc

Browser Process



Rendering and Input events

- `RenderWidgetHost` per "frame group"
  - New RWH when frame and parent are cross-process
  - Compositing, input events routed directly to frame

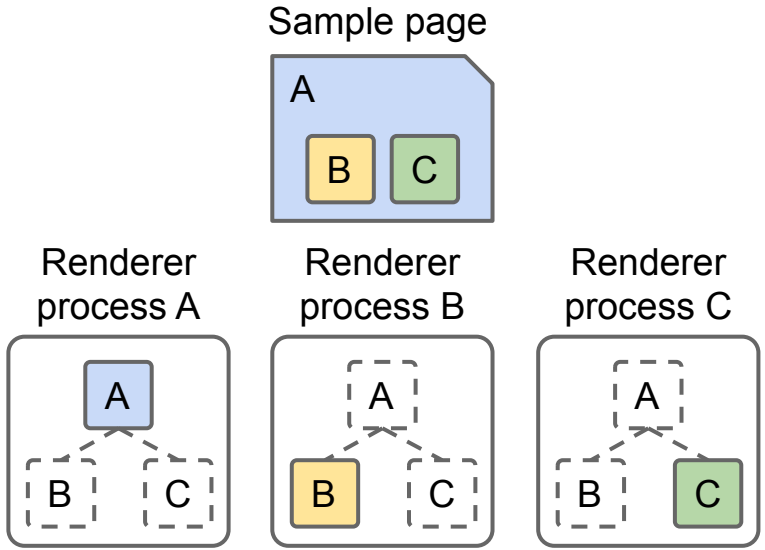


Architecture Cleanup

- Removing `RenderView*`
- swappedout:// RVH --> `RenderFrameProxy`
- Chromium manages page concept
  - Blink is just a frame server

Replicating frame properties

- Frame tree is replicated in each process
  - `RenderFrames` and `RenderFrameProxies`
- `FrameReplicationState`
  - Copied to all “instances” of a frame
  - Currently:
    - origin
    - <iframe sandbox> flags
    - window.name
  - Used to init `blink::SecurityContexts`
  - Must minimize information leaks



What you need to change

Interactions with RenderView{Host}

- Move to `RenderFrame{Host}` or `WebContents`
  - `RenderFrameObserver` in renderer
  - `WebContentsObserver` in browser

1:1 Frame Relationships

- Create frame-specific objects:
  - `WCO::RenderFrame{Created,Deleted}`
- Send and receive messages:
  - `RFH::Send`
  - `WCO::OnMessageReceived` and `IPC_MESSAGE_FORWARD`

Example in content/: MediaPlayerManager CL

1:1 Frame Relationships

MediaPlayerManager Example:

```
bool MediaWebContentsObserver::OnMessageReceived(
    const IPC::Message& msg,
    RenderFrameHost* render_frame_host) {

  BrowserMediaPlayerManager* player_manager =
      GetMediaPlayerManager(render_frame_host);
  DCHECK(player_manager);

  bool handled = true;
  IPC_BEGIN_MESSAGE_MAP(MediaWebContentsObserver, msg)
    IPC_MESSAGE_FORWARD(MediaPlayerHostMsg_EnterFullscreen,
                        player_manager,
                        BrowserMediaPlayerManager::OnEnterFullscreen)
```

Tracking set of current RenderFrameHosts

- A frame's RFH may change after navigation
- Watch for RFH changes or frame deletions:
  - `WCO::RenderFrameHostChanged(old_rfh, new_rfh)`
  - `WCO::FrameDeleted(rfh)`

Example in chrome/: Task Manager CL

Walking the Frame Tree

- Assemble partial results in browser process
- WebContents exposes frames:
  - `ForEachFrame`, `SendToAllFrames`
  - *Note:* `FrameTreeNode` *not (yet) exposed outside content/*

Example in content/: Accessibility CL

Testing OOPIFs

Will my CLs break OOPIF?

- Run tests locally with `--site-per-process`

- Two FYI bots run tests with `--site-per-process`:
  - Linux and Windows
  - All unit_tests and content_unittests
  - Almost all content_browsertests
  - browser_tests with ~200 tests disabled

Are my team's existing tests affected?

- Check if your test is disabled and help us fix it:
  - testing/buildbot/chromium.fyi.json
  - ~200, including WebView, Extensions, Geolocation, etc.

Writing new OOPIF tests

- Use `--site-per-process` and navigate to cross-site iframes
- Use `EmbeddedTestServer`
  - Getting a cross-site URL:
    ```
    embedded_test_server()->GetURL("foo.com", "/bar.html")
    ```
  - Or in HTML:
    ```
    "/cross-site/foo.com/bar.html"
    ```
- Details in Feature Update FAQ
- Examples in `site_per_process_browsertest.cc`

Chromium OOPIF - Q & A