Yoav Weiss - Oct 20, 2022 - Chromium.org

In order to support JS-generated popstate() events in SoftNavigationHeuristics, we need to be able to know that the history.back(), forward() or go() call is the parent task of the eventual popstate event.

When reviewing https://chromium-review.googlesource.com/c/chromium/src/+/3945102/5, altimin@ pointed out that we have we have no guarantee that the history.go() call and the popstate event will happen in the same LocalDOMWindow (e.g. if an iframe is being navigated, or if a past cross-origin document is popped).

As a result the current architecture is wrong, and keeping the parent task ID as state on the LocalDOMWindow is not the right way to go.

Looking into this, I think that the following would work better:

- In History::go(), the call to NavigateBackForward would also pass the current task ID
- We'd pipe that info through LocalFrameClient, and send it to the FrameHost in LocalFrameClientImpl::NavigateBackForward's call to GetToEntryAtOffset
- On the browser side, we'd pick that up at RenderFrameHostImpl::GoToEntryAtOffset
- We'd pass that info to NavigationControllerImpl::GoToOffsetFromRenderer, through GoToIndex, and onto NavigateToExistingPendingEntry
- We'd pass the info to FindFrameToNavigate, and set it on the same document navigation request
    - We may need to limit this to main-frame only, or otherwise restrict this info only for the frame which initiated the back/forward navigation.
    - I believe we can do that by checking in NavigateToExistingPendingEntry if the `root` frame's rfh is equal to the `initiator_rfh`
    - Then we can store the info as part of the CommitNavigationParams, which would mean we don't have to manually pipe it through NavigationRe quest
- After that, NavigationRequest goes through Navigator::Navigate and then NavigationRequest::BeginNavigation is called
- Eventually NavigationRequest calls RenderFrameHostImpl::CommitNavigation which then calls CommitSameDocumentNavigation
- On the renderer side, this calls RenderFrameImpl::CommitSameDocumentNavigation, which calls WebNavigationControl::CommitSameDocumentNavigation
- This calls DocumentLoader::CommitSameDocumentNavigation on the blink side, which calls UpdateForSameDocumentNavigation, which calls LocalDOMWindow::DispatchPopstateEvent

Sub issue - passing an optional parameter through mojom

In the above design, I'm trying to pass an absl::optional<blink::scheduler::TaskAttributionId> through mojom (LocalFrameHost::GoToEntryAtOffset).
Adding a `TaskAttributionId? Task_id` Parameter to that function doesn't do what I thought it would, and I'm getting compile errors on the blink side.
Looking at `/gen/third_party//blink/public/mojom/frame/frame.mojom-blink.h`, the virtual function is defined as

```
virtual void GoToEntryAtOffset(int32_t offset, bool has_user_gesture,
::blink::mojom::blink::TaskAttributionIdPtr task_id) = 0;
```

Whereas in `/gen/third_party//blink/public/mojom/frame/frame.mojom.h` It's defined as

```
virtual void GoToEntryAtOffset(int32_t offset, bool has_user_gesture,
absl::optional<::blink::scheduler::TaskAttributionId> task_id) = 0;
```

Solution

antoniosartori@ solved this!! Turns out my StructTraits definition for TaskAttributionId was defined just for cpp_typemaps and not for blink_cpp_typemaps, and therefore wasn't available from blink. Moving the definition to shared_cpp_typemaps solved the compile issues.