A model of navigation history

Connor G. Brewster
Alan Jeffrey

DRAFT of 2016-07-20

Abstact: Navigation has been a core component of the web since its inception: users and scripts can follow hyperlinks, and can go back or forwards through the navigation history. In this paper, we present a formal model aligned with the WHATWG specification of navigation history, and investigate its properties. The fundamental property of navigation history is that traversing the history by δ then by δ' should be the same as traversing by $\delta + \delta'$. In particular, traversing by $+1$ (forward) then by -1 (back) is the same as traversing by 0 (doing nothing). We show that the specification-aligned model does not satisfy this property, by exhibiting a series of counter-examples, which motivate three patches to the model. We present a series of experiments, showing that browsers are inconsistent in their implementation of navigation history, but that their behaviour is closer to the patched model than to the specification-aligned model. We propose patches to the specification to align it with the patched model.

ACM Classification: D.2.1 Requirements/Specifications.

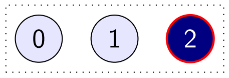
Keywords: Formal model, Navigation, Session history, Specification, Web browsers.

1. Introduction

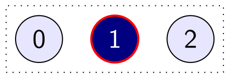
Navigation has been a core component of the web since its inception: users and scripts can follow hyperlinks, and can go back or forwards through the navigation history. Users are exposed to this functionality through following hyperlinks, and by the forward and back buttons. Scripts have many ways of accessing session history, via the navigation API [5, §7.7] and the `element.click()` method.

Prior formalizations of navigation history include [1, 2, 3, 4, 6], which predate and are not aligned with the WHATWG specification [5]. The specification of the navigation API is informal, and has complex dependencies on the rest of the HTML specification [5]. There is little discussion of the goals of the API, and an unclear alignment with browser implementations.

In this paper, we present a formal model of navigation, aligned with the HTML specification, and investigate its properties. The starting point is that there is a total order of *documents*[1], one of which is *active*, for example:

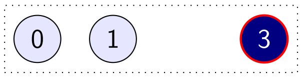


In diagrams, we use left-to-right order to indicate order, and highlight the active document. The user can *traverse* the history which changes the active document, for example pressing the back button:

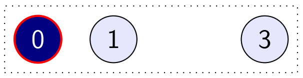


[1]We are eliding some of the technical details of the specification here, in particular we are conflatating a *browsing context* with the document it contains, and we are ignoring issues around document loading and unloading, and around the current entry of the joint session history.

The user can also *navigate*, which replaces any document after the currently active document by a fresh active document:



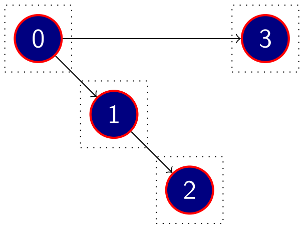
Users can also traverse the history by more than one document at a time, for example by using a pull-down menu from the back or forwards button. This is called *traversing by* δ, for instance we can traverse our running example by -2 to get:



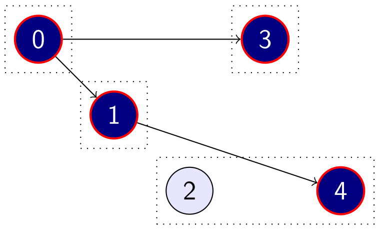
We formalize the notions of traversal and navigation in §2, and show that they satisfy two pleasant properties:

- *traverse-then-traverse*: traversing by δ then by δ' is the same as traversing by $\delta + \delta'$.
- *navigate-then-traverse*: navigating then traversing by -1 has the original active document.

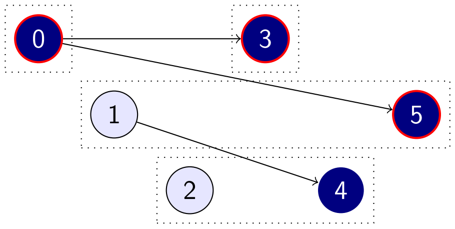
Thus far, the model is refreshingly simple, and corresponds well to the specification and to browser implmentations. Where the problems arise is in the *hierarchical* nature of documents. HTML documents can contain `iframe` elements, which are independent documents in their own right, often used to embed third party content such as advertisements. We can treat each document as a tree, for example:



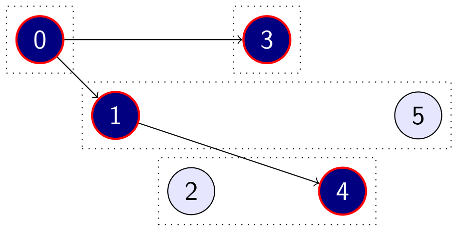
The problem comes from the ability of each document to navigate separately and maintain its own session history, but that traversal is a global operation that operates on the *joint session history*. For example if document 2 in the previous example navigates, the resulting state is:



and then if document 1 navigates, the state is:



Note that node 4 here is in an unusual state: it is active, but has an inactive ancestor. The specification [5, §7.7] distinguishes between *active* documents such as 4, and *fully active* documents such as 0, 3 and 5. Active documents can become fully active by traversals involving their ancestors. For example, after traversing by −1, document 4 is fully active:



As even a simple example like this shows, the combination of features quickly results in a complex mix of session history, ordering, and document hierarchy, which leads to the problems:

- *Formally* there is no simple model, and the model provided by the specification does not satisfy the traverse-then-traverse property.
- *Experimentally* the browsers disagree with each other, and with the HTML specification, about the semantics of navigation.

In this paper, we address these:

- §2 provides a formal model of navigation history, which is intended to align with the specification. We show, through a series of examples, that it does not satisfy the traverse-then-traverse property, and give patches to the model for each example. The final model does satisfy the traverse-then-traverse property.
- §3 investigates how well the patched model aligns with existing browser implementations. We show ways in which the browsers exhibit behaviours which are not aligned with the specification, and discuss how our proposed model matches these behaviours.

Finally, we propose changed wording to the specification, which would bring it in line with our patched model.

2. Model

In this section, we present our formal model of navigation history. §2.1 contains definitions of concepts such as tree and total order, and may be skipped by most readers. The model, together with some examples, is given in §2.2. In §2.3 we define the fundamental property of navigtion history, show that the model does *not* satisfy it, but can be patched to do so.

2.1. Preliminaries

A *directed graph* $G = (V, \to)$ consists of:

- a set V (the *vertices*), and
- a relation $\to \subseteq (V \times V)$ (the *edges*).

The *transitive closure* of $\to$ is defined as $v \to^+ v'$ whenever there exists $v_0, \ldots, v_n$ such that:

$$v = v_0 \to \cdots \to v_n = v'$$

The *reflexive transitive closure* of $\to$ is defined as $v \to^* v'$ whenever $v \to^+ v'$ or $v = v'$. A *forest* is a directed graph where:

- there is no v such that $v \to^+ v$ (*acyclicity*)
- if $v \to v' \leftarrow v''$ then $v = v''$ (*at most one parent*).

A *root vertex* of a forest is a vertex v such that there is no $w \to v$. A *tree* is a forest with a unique root vertex. A *preorder* is a directed graph $(V, \leq)$ such that:

- every v has $v \leq v$ (*reflexivity*), and
- if $v \leq v' \leq v''$ then $v \leq v''$ (*transitivity*).

A *partial order* is a preorder such that:

- for every v and v', if $v \leq v'$ and $v' \leq v$ then $v = v'$ (*antisymmetry*).

A *total order* is a partial order such that:

- for every v and v', either $v \leq v'$ or $v \geq v'$ (*totality*).

A *equivalence* is a preorder $(V, \sim)$ such that:

- if $v \sim v'$ then $v' \sim v$ (*symmetry*).

2.2. Definitions

We can now formalize our model of navigation history, together with the operations of navigation and traversal. This formalizes the navigation history specification [5], and has a pleasant diagramatic presentation, but as we shall see in §2.3, it has unexpected properties.

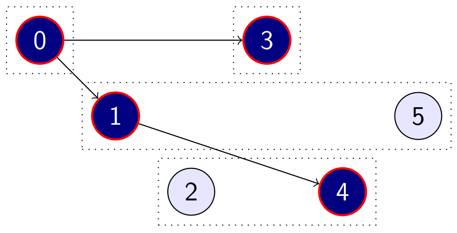
Definition 1 (Navigation history): A *navigation history* $H = (D, A, \to, \leq, \sim)$ consists of:

- a finite set D (the *documents*),
- a subset $A \subseteq D$ (the *active* documents),
- a forest $(D, \to)$ (the *child* relationship),
- a total order $(D, \leq)$ (the *chronological* order), and
- an equivalence relation $(D, \sim)$ (the *same-session* equivalence).

such that:

- for every d there is a unique $d' \in A$ such that $d \sim d'$,
- for every $d \to e \sim e'$ we have $d \to e'$, and
- for every $d \to e$, we have $d \leq e$. $\square$

We present such navigation histories diagramatically, using left-to-right position for chronological order, and grouping documents in the same session. Since documents in the same session must have the same parent, we only draw the document hierarchy for active children. For example the diagram:



represents:

$$\begin{aligned}
&D = \{0, 1, 2, 3, 4, 5\}\\
&A = \{0, 1, 3, 4\}\\
&0 \to 1 \quad 0 \to 3 \quad 0 \to 5 \quad 1 \to 2 \quad 1 \to 4\\
&0 \leq 1 \leq 2 \leq 3 \leq 4 \leq 5\\
&1 \sim 5 \quad 2 \sim 4
\end{aligned}$$

In such a navigation history, we define:

- d_0 is the unique active root document,
- $d \twoheadrightarrow e$ when $d \to e$ and $e \in A$ (the *active child* relationship),
- $FA = \{d \mid d_0 \twoheadrightarrow^* d\}$ (the *fully active* documents),
- $d \lesssim e$ whenever $d \sim e$ and $d < e$,
- the *session future* of d is $\{e \mid d \lesssim e\}$,
- the *session past* of d is $\{e \mid d \gtrsim e\}$,
- the *joint session future* is $\{e \mid \exists d \in FA \,.\, d \lesssim e\}$,
- the *joint session past* is $\{e \mid \exists d \in FA \,.\, d \gtrsim e\}$,

These definitions are intended to align with the specification, for example [5, 7.7.2] has the definition:

> The **joint session history** of a top-level browsing context is the union of all the session histories of all browsing contexts of all the fully active `Document` objects that share that top-level browsing context, with all the entries that are current entries in their respective session histories removed except for the current entry of the joint session history.

which (eliding the concept of "current entry of the joint session history") corresponds to the above definitions of joint session future and past. We now consider how to formalize operations on navigation histories. staring with *navigation*, which is triggered by following hyperlinks, or other actions which trigger document loading.

Definition 2 (Navigation): Define *deleting d from H*, when $d \notin FA$, to be H' where:

- $D' = d \setminus \{e \mid d \to^* e\}$,
- $e \in A'$ whenever $e \in A$,
- $e \leq' f$ whenever $e \leq f$,
- $e \to' f$ whenever $e \to f$, and
- $e \sim' f$ whenever $e \sim f$.

Define *replacing d by d' in H*, where $d \in A$ and $d' \notin D$, to be H' where:

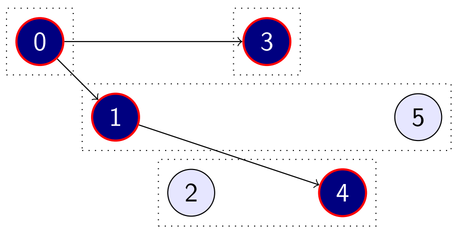
- $D' = D \cup \{d'\}$,
- $e \in A'$ whenever $e \in A$ and $e \neq d$, or $e = d'$,
- $e \leq' f$ whenever $e \leq f$, or $f = d'$,
- $e \to' f$ whenever $e \to f$, or $e \to d$ and $f = d'$, and

- $e \sim' f$ whenever $e \sim f$, or $e = f$, or $e \sim d$ and $f = d'$, or $d \sim f$ and $e = d'$.

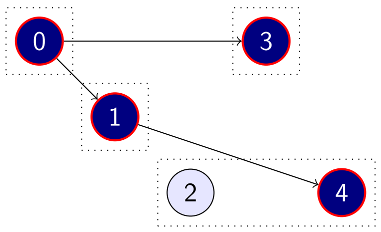
Define *navigating from d to d' in H* to be the result of:

- deleting the session future of d, and
- replacing d by d'. □

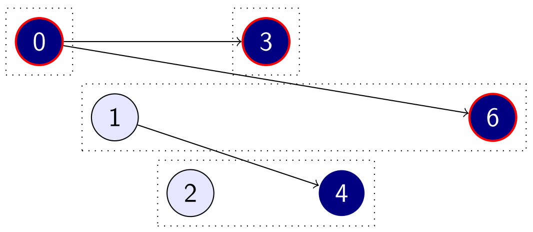
There are two parts to navigation from d to d': deleting the session future of d, followed by replacing d by d'. For example, navigating from 1 to 6 in:



we first delete the session future of 1 (which is $\{5\}$):



then replace 1 by 6:



We also define *traversing the history*, which changes the active documents.

Definition 3 (Traversal): Define *traversing the history to d in H* to be H' where:

- D' is D,
- $e \in A'$ whenever $d \not\sim e \in A$, or $d = e$,
- $e \leq' f$ whenever $e \leq f$,
- $e \to' f$ whenever $e \to f$, and
- $e \sim' f$ whenever $e \sim f$.

Define *H traverses the history by $+\delta$ to H'* when:

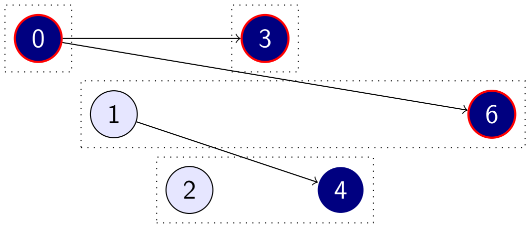
- the joint session future of H is $d_1 > \cdots > d_\delta > \cdots$,
- H traverses the history to d_δ in H'

Define H *traverses the history by* $-\delta$ *to* H' when:

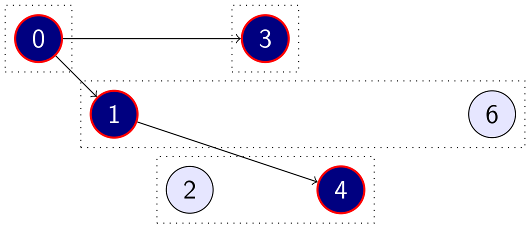
- the joint session past of H is $d_1 < \cdots < d_\delta < \cdots$,
- H traverses the history to d_δ in H'

Define H *traverses the history by* 0 *to* H' when $H = H'$.

For example, to traverse the history by -2 in:



we find the joint session past (which is $1 < 2$) and traverse the history to the second item (which is 2) to arrive at:



These definitions are intended to formally capture the HTML specification, for example [5, 7.7.2] includes:

> To **traverse the history by a delta** δ, the user agent must append a task to this top-level browsing context's session history traversal queue, the task consisting of running the following steps:
>
> 1. If the index of the current entry of the joint session history plus δ is less than zero or greater than or equal to the number of items in the joint session history, then abort these steps.
> 2. Let *specified entry* be the entry in the joint session history whose index is the sum of δ and the index of the current entry of the joint session history.
> 3. Let *specified browsing context* be the browsing context of the specified entry.
> 4. If the specified browsing context's active document's unload a document algorithm is currently running, abort these steps.
> 5. Queue a task that consists of running the following substeps [...]
>
>   3. Traverse the history of the specified browsing context to the specified entry.

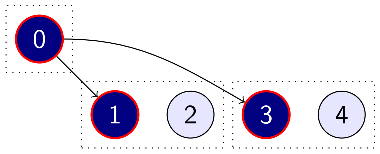
2.3. Properties

We now consider the fundamental property of navigation history:

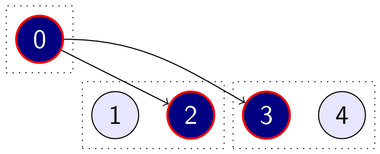
Goal 1 (Traverse-then-traverse): If H traverses the history by δ to H' and H' traverses the history by δ' to H'' then H traverses the history by $\delta + \delta'$ to H''. □

Unfortunately, navigation as specified does not satisfy this property, due to the interplay of session history with the tree structure of documents. In this section, we give a series of counterexamples, and propose patches to the model to address each counterexample.

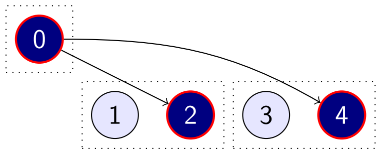
Counterexample 1: Let H be:



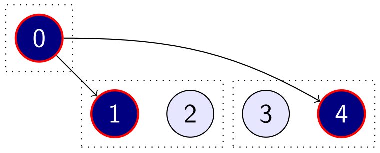
which traverses the history by 1 to:



which traverses the history by 1 to:



but H traverses the history by 2 to:



□

This counterexample is caused by the definition of 'traverses the history by δ' which only traverses one document's session history. Instead, we should traverse the history of all δ documents.

Patch 1 (Traverse intermediaries): Define H *traverses the history by* $+\delta$ *to* H' when:

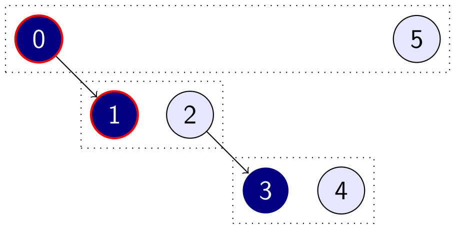
- the joint session future of H is $d_1 < \cdots < d_\delta < \cdots$,
- there is some $H = H_0, \ldots, H_\delta = H'$, such that
- H_{i-1} traverses the history to d_i in H_i for each $1 \leq i \leq \delta$.

Define H *traverses the history by* $-\delta$ *to* H' when:

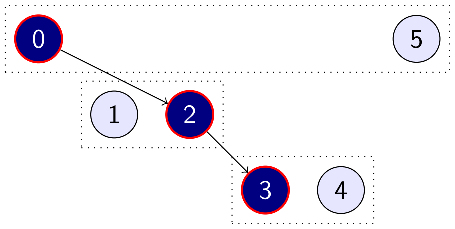
- the joint session past of H is $d_1 > \cdots > d_\delta > \cdots$,
- there is some $H = H_0, \ldots, H_\delta = H'$, such that
- H_{i-1} traverses the history to d_i in H_i for each $1 \leq i \leq \delta$. □

Unfortunately, Goal 1 is not satisfied, even with this patch.

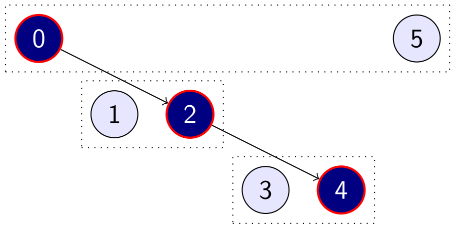
Counterexample 2: Let H be:



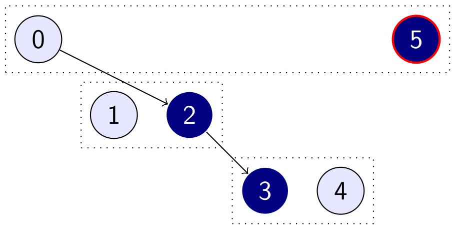
which traverses the history by 1 to:



which in turn traverses the history by 1 to:



but H traverses the history by 2 to:



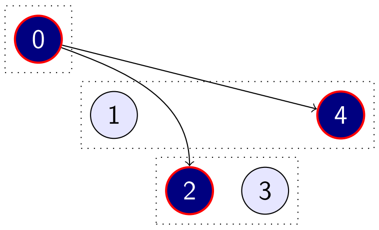
$\square$

The problem this time is that the definition of 'joint session history' only includes the fully active documents, not all active documents.

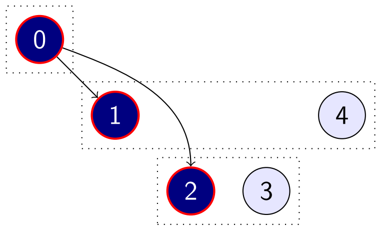
Patch 2 (Active joint session history): Define:

- the *joint session future* is $\{e \mid \exists d \in A \,.\, d \lesssim e\}$, and
- the *joint session past* is $\{e \mid \exists d \in A \,.\, d \gtrsim e\}$. $\square$

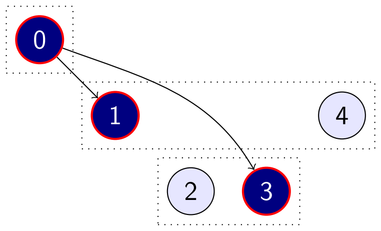
Counterexample 3: Let H be:



which traverses the history by -1 to:



which traverses the history by 1 to:



which is not the same as H. □

This final counterexample is caused by an asymmetry in the definition of traversal: it is defined in terms of navigating *to* a document d, and not navigating *from* a document. We fix this by making the definition symmetric:

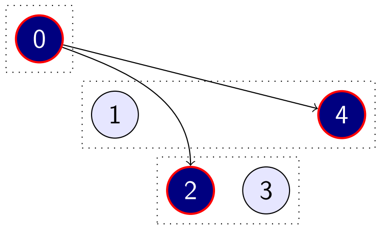
Patch 3 (Symmetric traversal): Define H *traverses the history from* d' when there is some d such that:

- $d \lesssim d'$,
- for any $e \lesssim d'$ we have $e \leq d$, and
- H traverses the history to d.

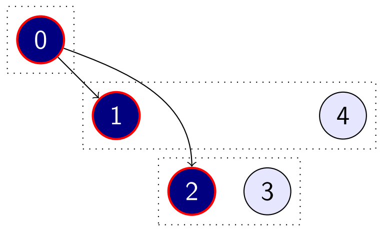
Define H *traverses the history by* $-\delta$ *to* H' when:

- the joint session past and active documents of H is $d_1 > \cdots > d_\delta > \cdots$,
- there is some $H = H_0, \ldots, H_\delta = H'$, such that
- H_{i-1} traverses the history from d_i in H_i for each $1 \leq i \leq \delta$. □

For example, to traverse the history by -1 from:



we find the joint session past and active documents (which is $4 > 2 > 1 > 0$) and traverse the history from the first item (which is 4) which is the same as traversing the history to 1:



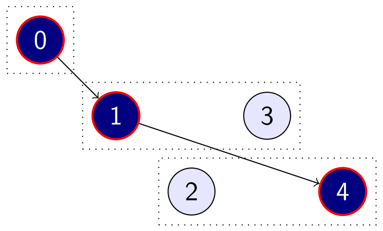
With these patches, we conjecture that our fundamental property is true, and moreover that traversal forms an inverse of navigation.

Conjecture 1 (Traverse-then-traverse): If H traverses the history by δ to H' and H' traverses the history by δ' to H'' then H traverses the history by $\delta + \delta'$ to H''. □

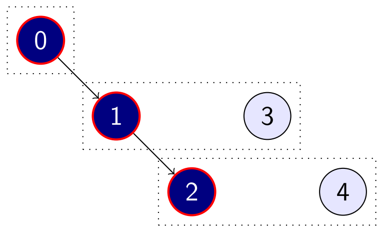
Conjecture 2 (Navigate-then-traverse): If d in H navigates to d' in H', and H' traverses the history by -1 to H'', then $A = A''$. □

[NOT TRUE: oh rats!]

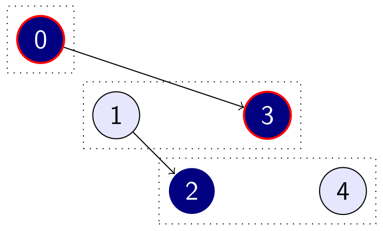
Counterexample 4: Let H be:



which traverses the history by -1 to:



which traverses the history by 1 to:



which is not the same as H. □

3. Experiments

In this section, we summarize our experiments to validate the conformance of browser implementations with respect to the WHATWG specification, to our proposed changes, and to each other. The full details are in Appendix A.

[Present one experiment in detail.]

In summary, our experimental findings are: [summarize the experiments here.]

4. Specification

[Suggested edits to the spec: 1. traverse to each document, not just the selected one, 2. keep all documents in the seession history, not just the fully active ones, 3. change the session history order.]

5. Conclusion

[We did stuff.]

References

[1] Minmin Han and Christine Hofmeister. Modeling and verification of adaptive navigation in web applications. In *Proc. Int. Conf. Web Engineering*, pages 329–336, 2006.

[2] May Haydar. Formal framework for automated analysis and verification of web-based applications. In *Proc. IEEE Int. Conf. Automated Software Engineering*, pages 410–413, 2004.

[3] May Haydar, Alexandre Petrenko, and Houari Sahraoui. Formal verification of web applications modeled by communicating automata. In *Proc. Int. Conf. Formal Techniques for Networked and Distributed Systems*, pages 115–132, 2004.

[4] Karl R. P. H. Leung, Lucas Chi Kwong Hui, S. M. Yiu, and Ricky W. M. Tang. Modeling web navigation by statechart. In *Proc. Int. Computer Software and Applications Conf.*, pages 41–47, 2000.

[5] WHATWG. HTML living standard. https://html.spec.whatwg.org/, 2016.

[6] Marco Winckler and Philippe Palanque. StateWebCharts: A formal description technique dedicated to navigation modelling of web applications. In *Proc. Int. Workshop Interactive Systems. Design, Specification, and Verification*, pages 61–76, 2003.

A. Experiments

Experiment 1: In this experiment Goal 1 is tested.

- *H traverses the history by -4 to H'*
- *H' traverses the history by $+4$ to H''*

By Goal 1, these traversals should be the same thing as H *traversing by* 0 which is a no-op; therefore, $H = H''$.

Firefox:

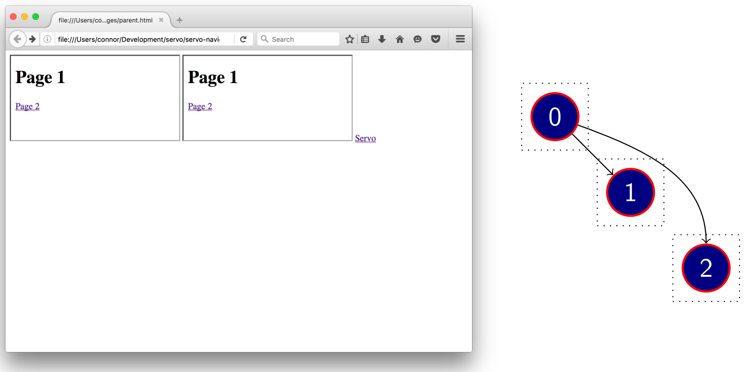

file:///Users/co...ges/parent.html
file:///Users/connor/Development/servo/servo-navi
Search
Page 1
Page 2
Page 1
Page 2
Servo
0
1
2

Figure 1: Initial State

Navigate document 1 to Page 2:

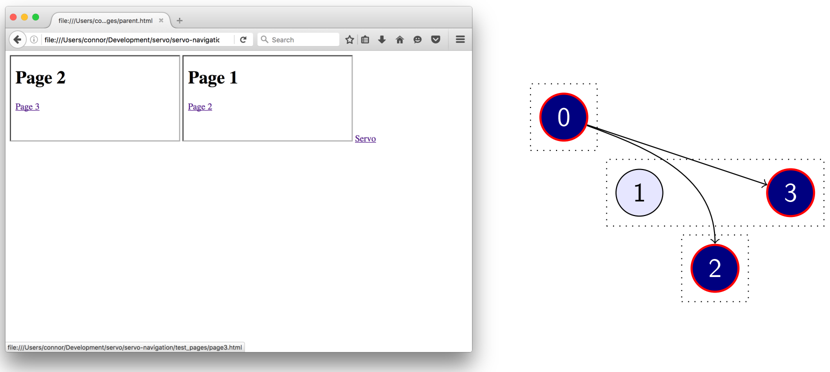

file:///Users/co...ges/parent.html
file:///Users/connor/Development/servo/servo-navigatic
Search
Page 2
Page 3
Page 1
Page 2
Servo
file:///Users/connor/Development/servo/servo-navigation/test_pages/page3.html
0
1
3
2

Figure 2: Navigate document 1 to Page 2.

Navigate document 3 to Page 3:

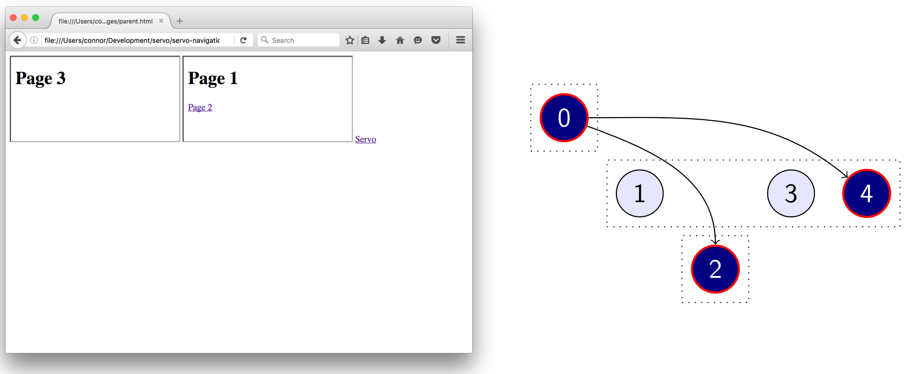

file:///Users/co...ges/parent.html
file:///Users/connor/Development/servo/servo-navigatic
Search
Page 3
Page 1
Page 2
Servo
0
1
3
4
2

Figure 3: Navigate document 3 to Page 3.

Navigate document 2 to Page 2:

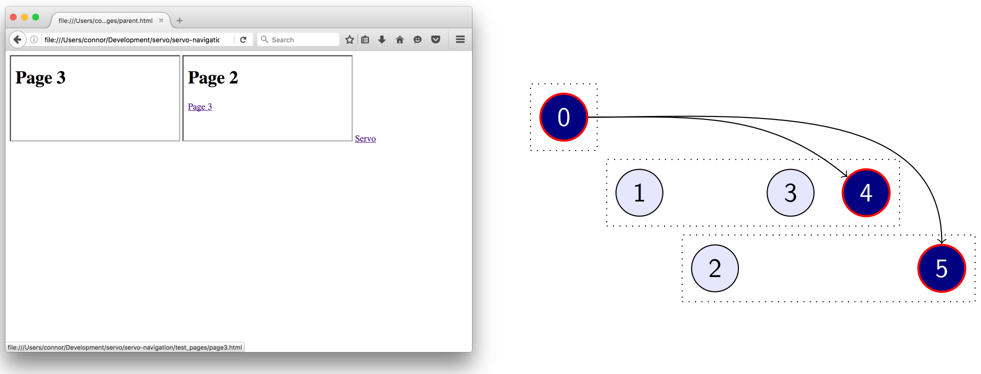

file:///Users/co...ges/parent.html
file:///Users/connor/Development/servo/servo-navigatic
Search
Page 3
Page 2
Page 3
Servo
file:///Users/connor/Development/servo/servo-navigation/test_pages/page3.html
0
1
3
4
2
5

Figure 4: Navigate document 2 to Page 2.

Navigate document 5 to Page 3:

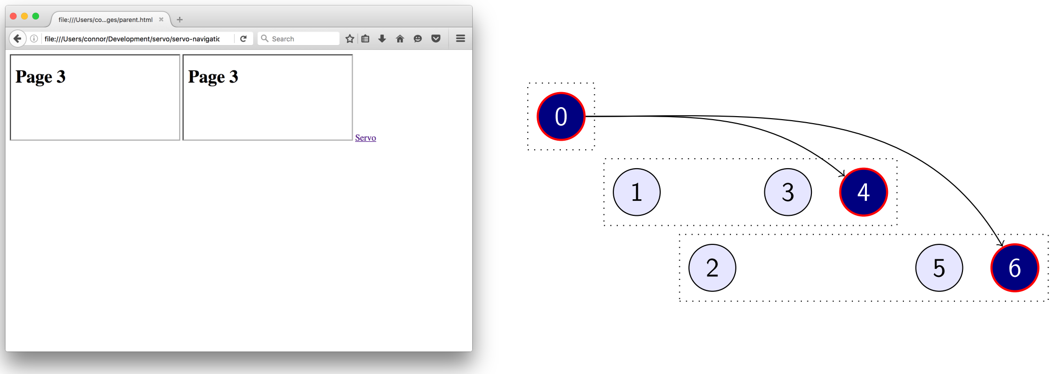

file:///Users/co...ges/parent.html
file:///Users/connor/Development/servo/servo-navigatic
Search
Page 3
Page 3
Servo
0
1
3
4
2
5
6

Figure 5: Navigate document 5 to Page 3.

H traverses the history by -4 to H':

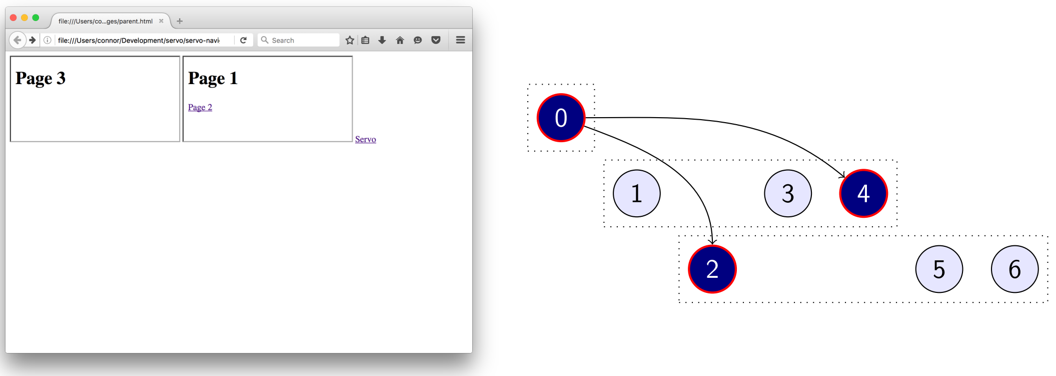

file:///Users/co...ges/parent.html
file:///Users/connor/Development/servo/servo-navi
Search
Page 3
Page 1
Page 2
Servo
0
1
3
4
2
5
6

Figure 6: Traversal by −4.

This state is obviously wrong, as document 4 should have traversed to document 1. This is similar to counterexample 1.

H' traverses the history by 4 to H'':

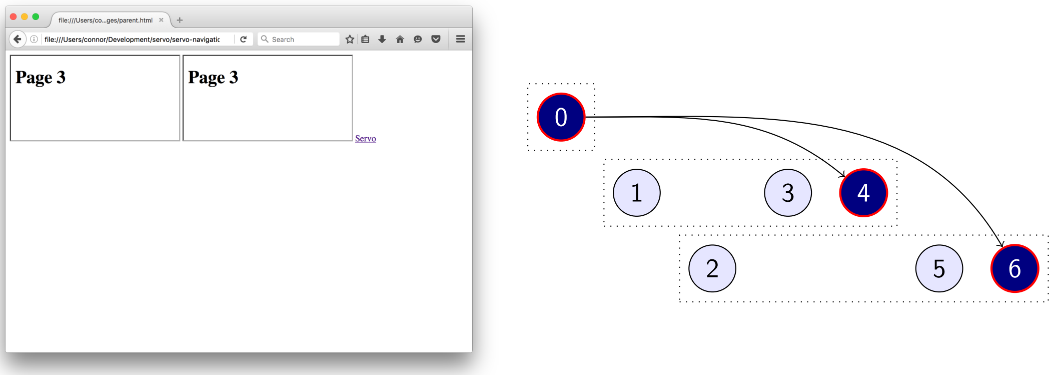

file:///Users/co...ges/parent.html
file:///Users/connor/Development/servo/servo-navigatic
Search
Page 3
Page 3
Servo
0
1
3
4
2
5
6

Figure 7: Traversal by 4.

While this result does satisfy Goal 1, there are still some issues:

- Figure 6 yields an incorrect traversal. [CGB: I believe this actually does break Goal 1 as navigating by −1 four times should yield the correct state.]
- It is impossible to get back to Page 1 on both Frames. [CGB: Looks to be a bug in FF, when holding down the back button, the list of pages to traverse to shows up. Clicking on the oldest item on the list does nothing and does not activate that item.]

Safari:

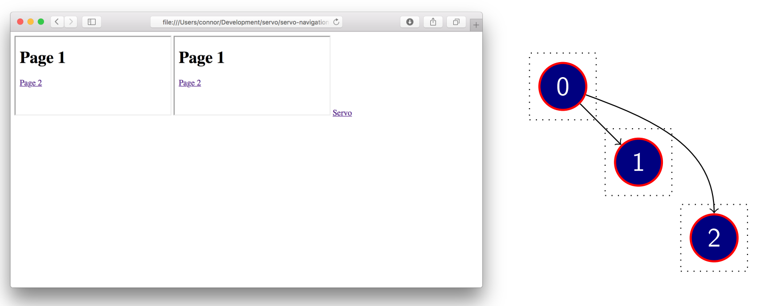


Figure 8: Initial State

Navigate document 1 to Page 2:

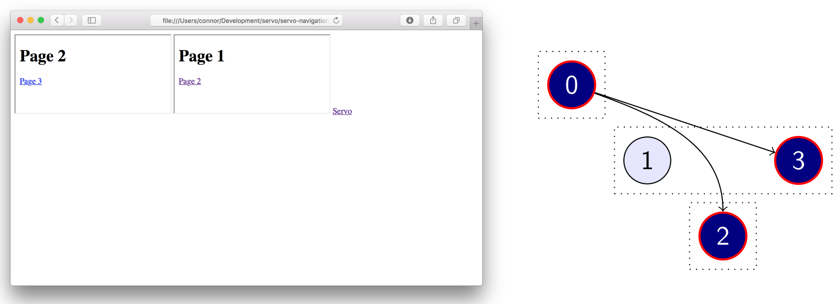


Figure 9: Navigate document 1 to Page 2.

Navigate document 3 to Page 3:

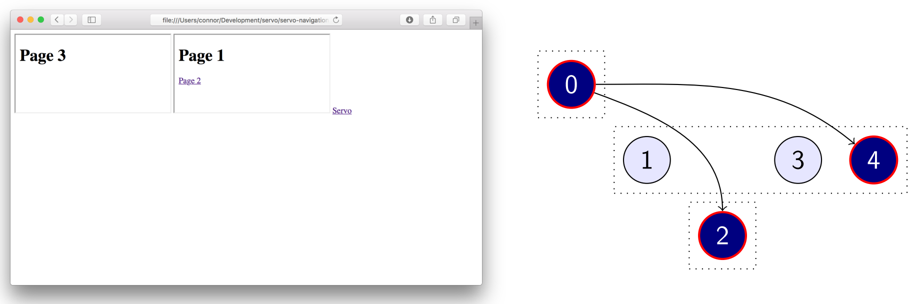


Figure 10: Navigate document 3 to Page 3.

Navigate document 2 to Page 2:

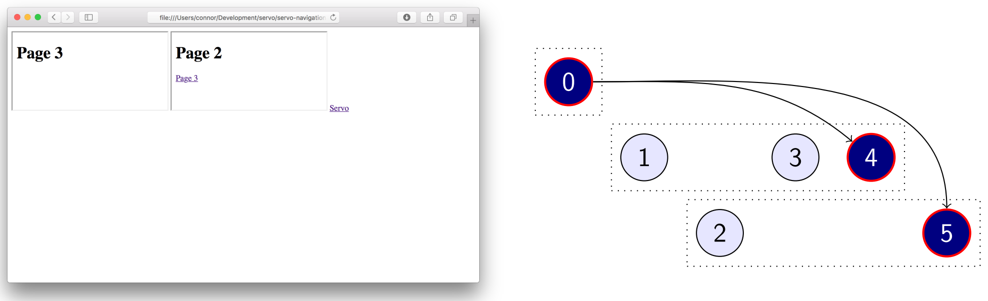


Figure 11: Navigate document 2 to Page 2.

Navigate document 5 to Page 3:

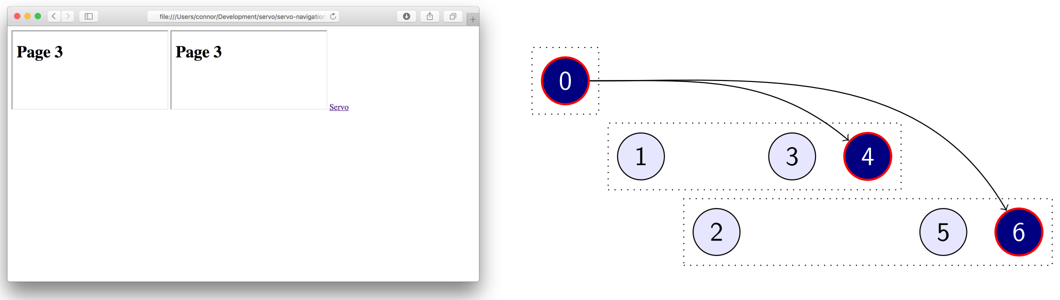


Figure 12: Navigate document 5 to Page 3.

H traverses the history by -4 *to* H':

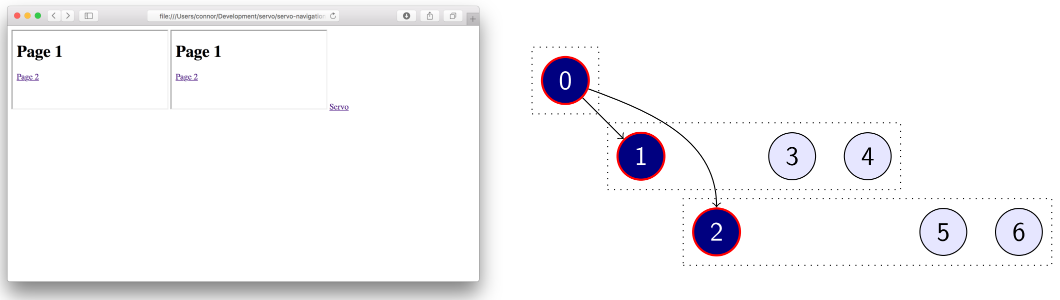


Figure 13: Traversal by -4.

H' *traverses the history by* 4 *to* H'':

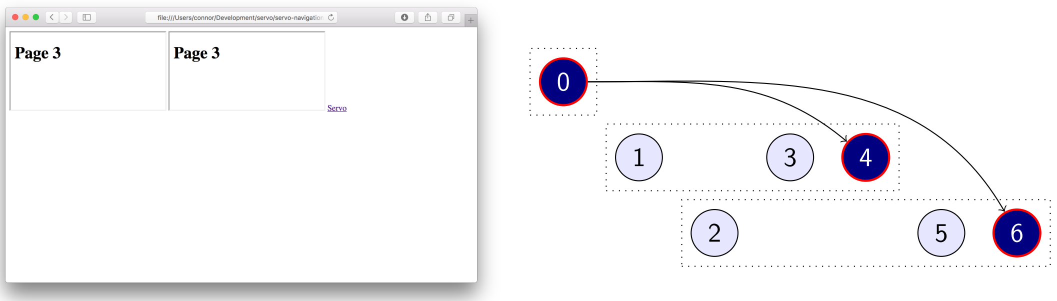


Figure 14: Traversal by 4.

These results in Safari satisfy Goal 1.
Chrome:

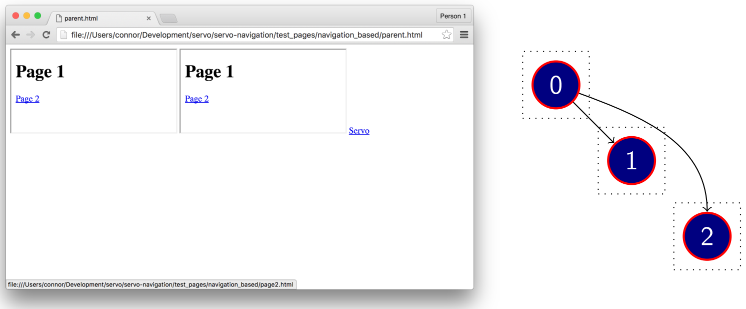

parent.html
Person 1
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/parent.html
Page 1
Page 2
Page 1
Page 2
Servo
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/page2.html
0
1
2

Figure 15: Initial State

Navigate document 1 to Page 2:

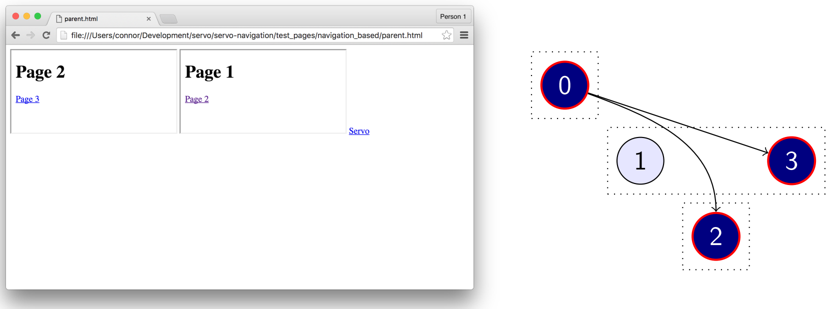

parent.html
Person 1
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/parent.html
Page 2
Page 3
Page 1
Page 2
Servo
0
1
3
2

Figure 16: Navigate document 1 to Page 2.

Navigate document 3 to Page 3:

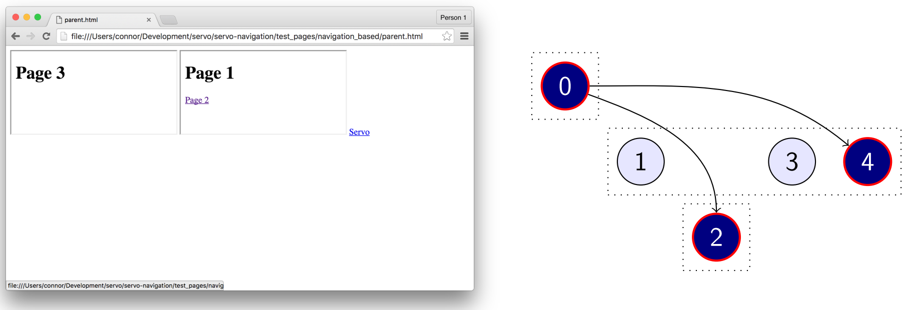

parent.html
Person 1
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/parent.html
Page 3
Page 1
Page 2
Servo
file:///Users/connor/Development/servo/servo-navigation/test_pages/navig
0
1
3
4
2

Figure 17: Navigate document 3 to Page 3.

Navigate document 2 to Page 2:

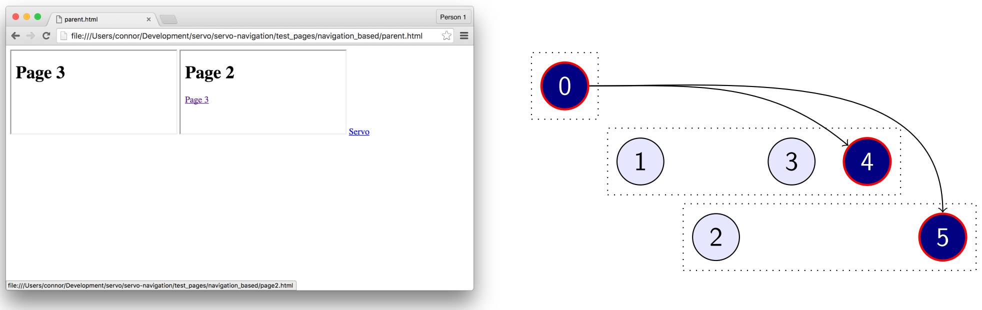

parent.html
Person 1
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/parent.html
Page 3
Page 2
Page 3
Servo
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/page2.html
0
1
3
4
2
5

Figure 18: Navigate document 2 to Page 2.

Navigate document 5 to Page 3:

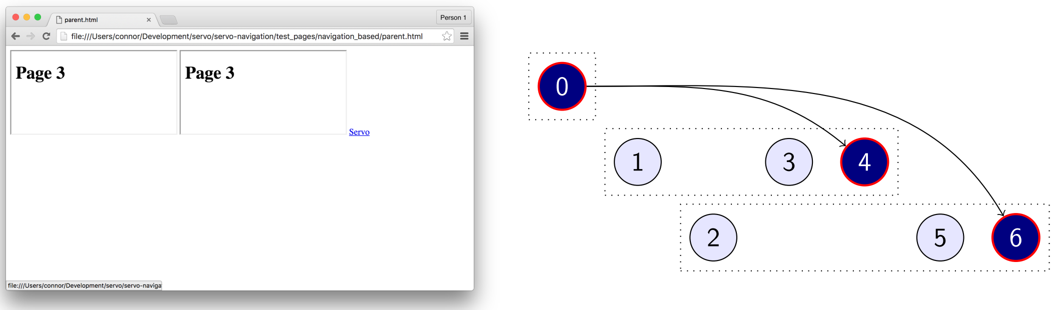
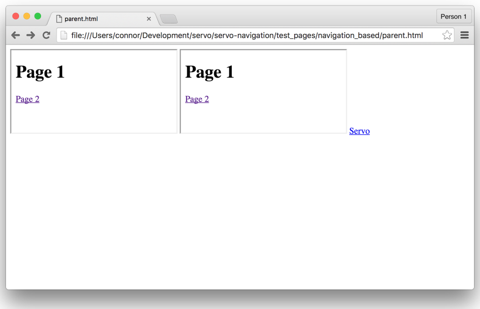

parent.html
Person 1
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/parent.html
Page 3
Page 3
Servo
file:///Users/connor/Development/servo/servo-naviga
0
1
3
4
2
5
6

Figure 19: Navigate document 5 to Page 3.

H traverses the history by -4 to H':


parent.html
Person 1
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/parent.html
Page 1
Page 2
Page 1
Page 2
Servo

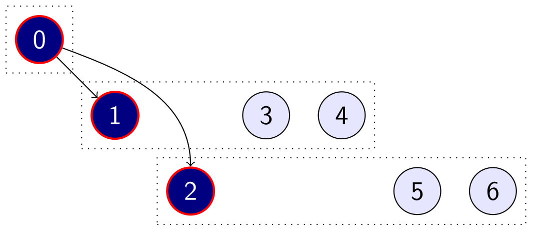
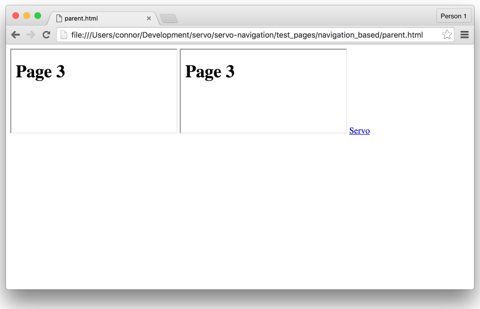


Figure 20: Traversal by -4.

H' traverses the history by 4 to H'':


parent.html
Person 1
file:///Users/connor/Development/servo/servo-navigation/test_pages/navigation_based/parent.html
Page 3
Page 3
Servo

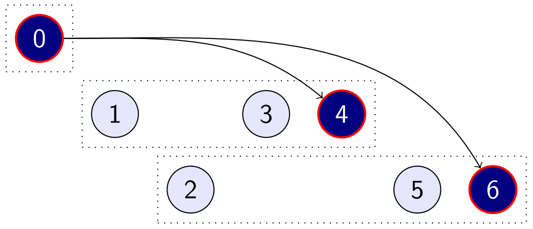
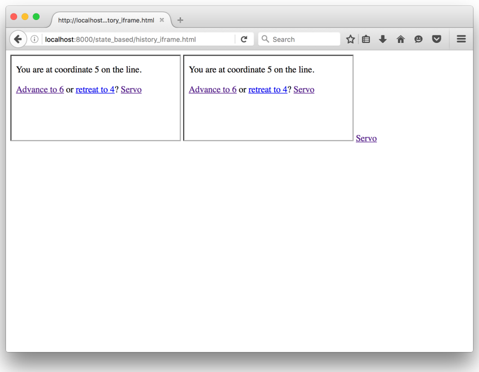


Figure 21: Traversal by 4.

These results in Chrome satisfy Goal 1.

Experiment 2: In this experiment *pushState* and *replaceState* traversals are tested against Goal 1.

Firefox:


http://localhost...tory_iframe.html
localhost:8000/state_based/history_iframe.html
Search

You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

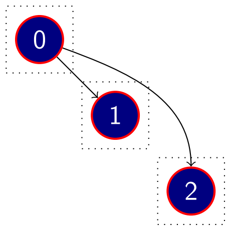
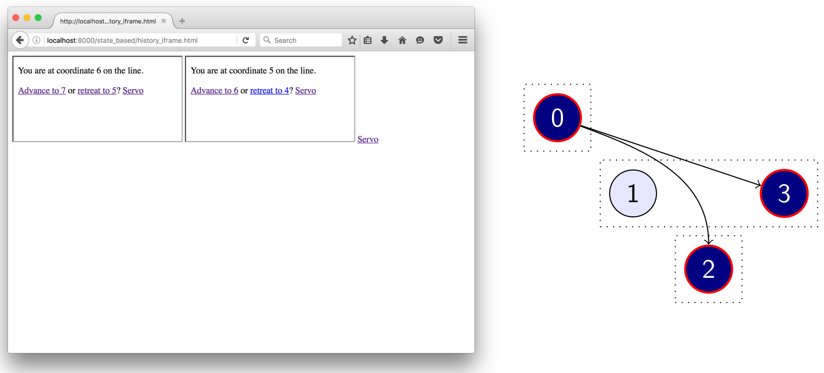


Figure 22: Initial State

Advance document 1 to 6:


http://localhost...tory_iframe.html

localhost:8000/state_based/history_iframe.html

You are at coordinate 6 on the line.

Advance to 7 or retreat to 5? Servo

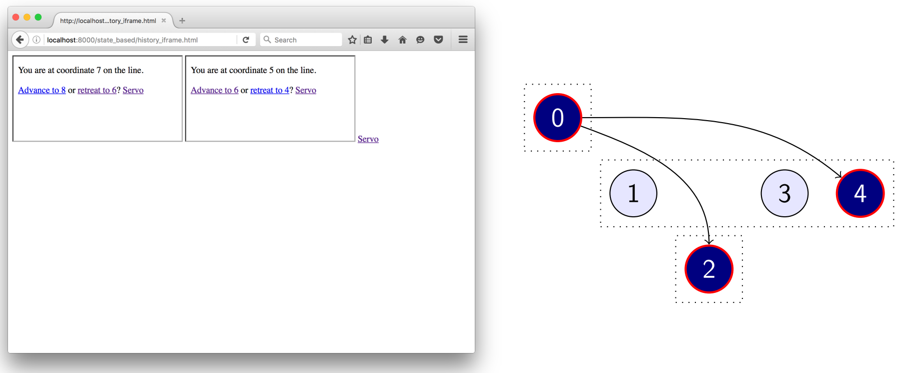
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 23: Advance document 1 to 6

Advance document 3 to 7:


http://localhost...tory_iframe.html

localhost:8000/state_based/history_iframe.html

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

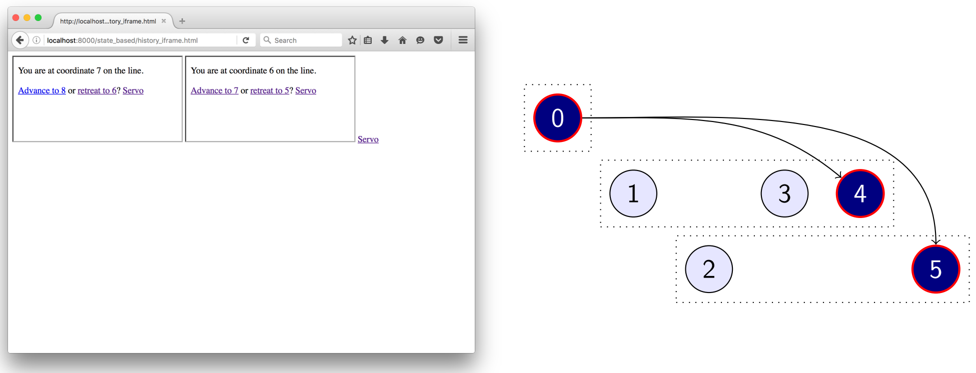
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 24: Advance document 3 to 7

Advance document 2 to 6:


http://localhost...tory_iframe.html

localhost:8000/state_based/history_iframe.html

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

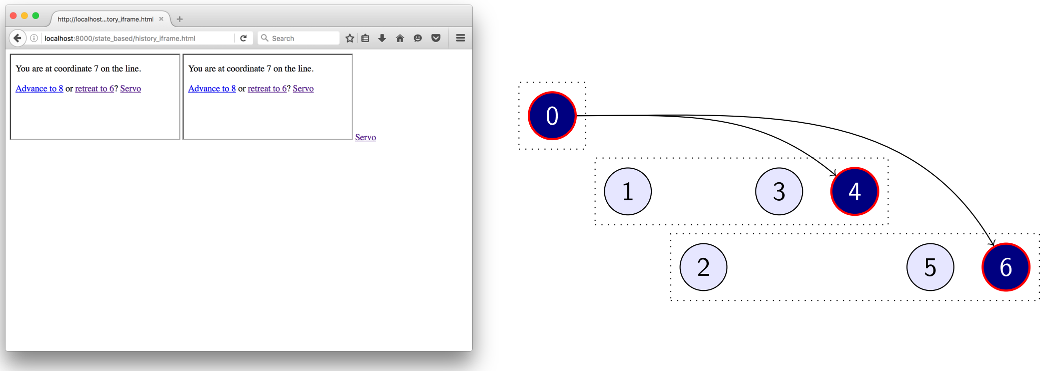
You are at coordinate 6 on the line.

Advance to 7 or retreat to 5? Servo

Servo

Figure 25: Advance document 2 to 6

Advance *document*2 to 7:


http://localhost...tory_iframe.html
localhost:8000/state_based/history_iframe.html
Search
0
1
3
4
2
5
6

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

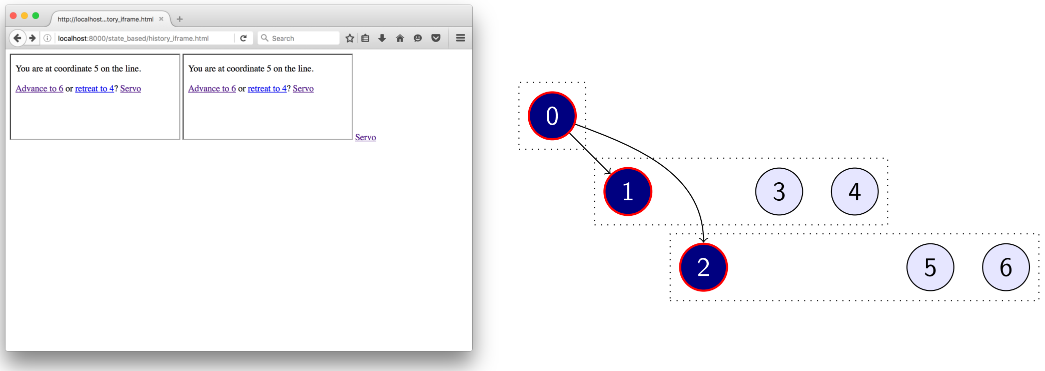
You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

Servo

Figure 26: Advance document 2 to 7

Traverse H by -4:


http://localhost...tory_iframe.html
localhost:8000/state_based/history_iframe.html
Search
0
1
3
4
2
5
6

You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

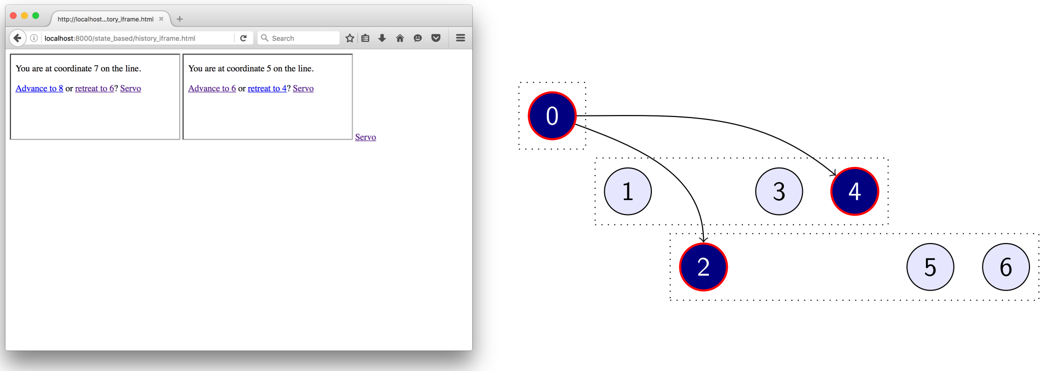
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 27: Traverse H by -4

Traverse H by 4:


http://localhost...tory_iframe.html
localhost:8000/state_based/history_iframe.html
Search
0
1
3
4
2
5
6

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

You are at coordinate 5 on the line.

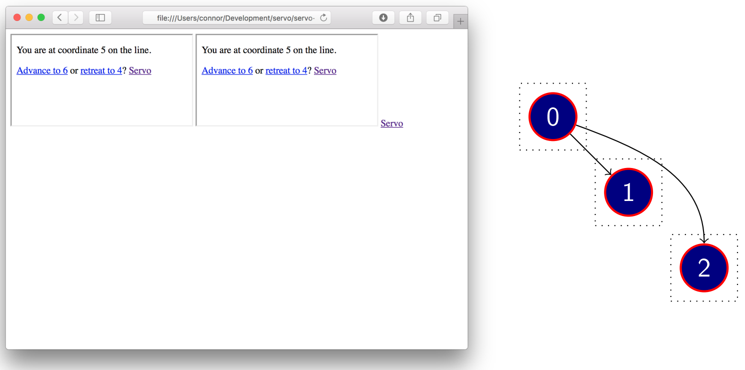
Advance to 6 or retreat to 4? Servo

Servo

Figure 28: Traverse H by 4

The last traversal does not satisfy Goal 1.

Safari:


file:///Users/connor/Development/servo/servo-

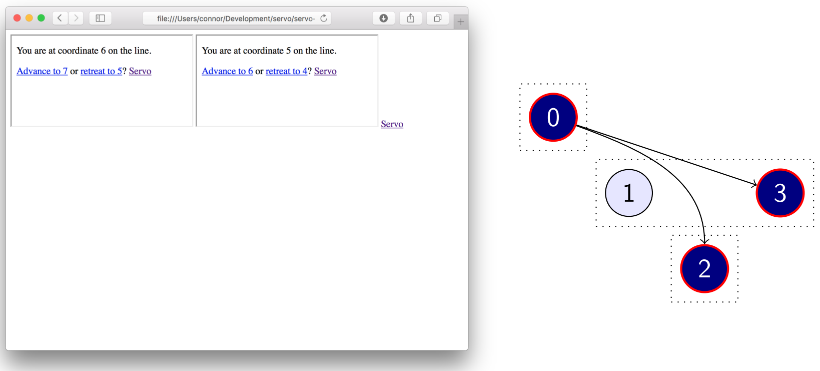
You are at coordinate 5 on the line.
Advance to 6 or retreat to 4? Servo

You are at coordinate 5 on the line.
Advance to 6 or retreat to 4? Servo

Servo

Figure 29: Initial State

Advance document 1 to 6:


file:///Users/connor/Development/servo/servo-

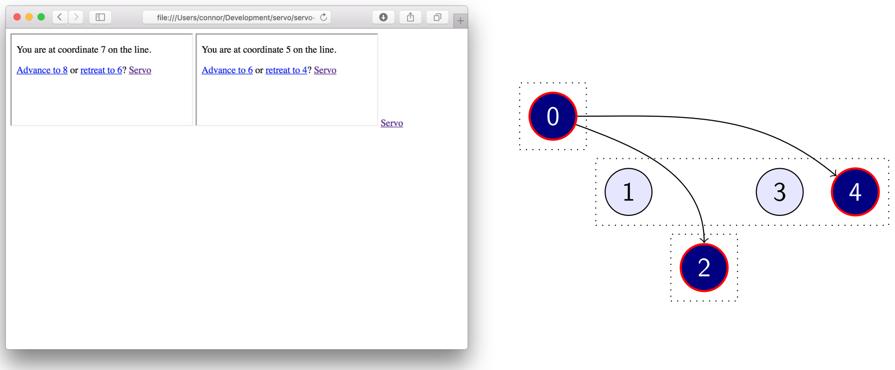
You are at coordinate 6 on the line.
Advance to 7 or retreat to 5? Servo

You are at coordinate 5 on the line.
Advance to 6 or retreat to 4? Servo

Servo

Figure 30: Advance document 1 to 6

Advance document 3 to 7:


file:///Users/connor/Development/servo/servo-

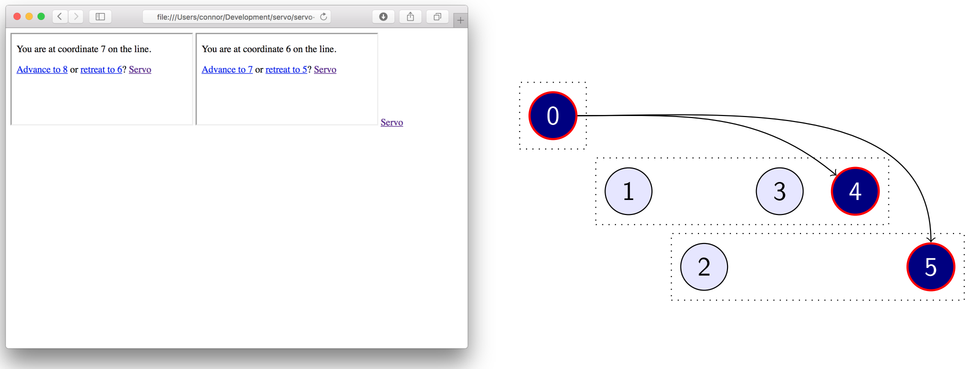
You are at coordinate 7 on the line.
Advance to 8 or retreat to 6? Servo

You are at coordinate 5 on the line.
Advance to 6 or retreat to 4? Servo

Servo

Figure 31: Advance document 3 to 7

Advance document 2 to 6:


file:///Users/connor/Development/servo/servo-

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

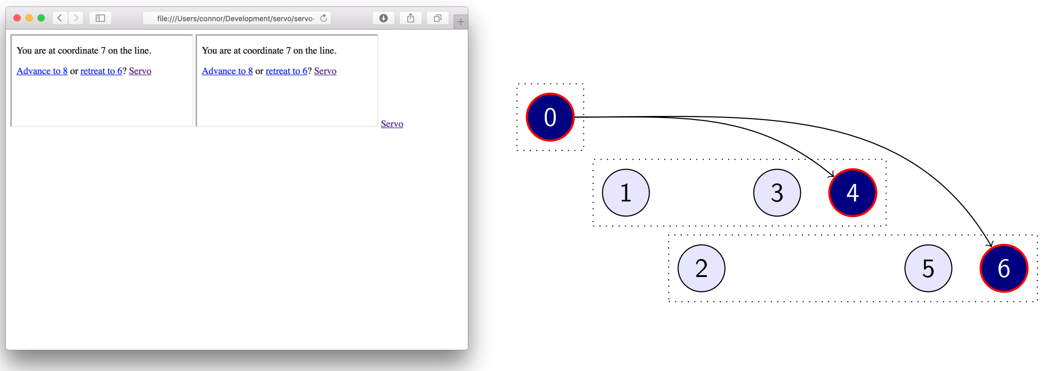
You are at coordinate 6 on the line.

Advance to 7 or retreat to 5? Servo

Servo

Figure 32: Advance document 2 to 6

Advance *document*2 to 7:


file:///Users/connor/Development/servo/servo-

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

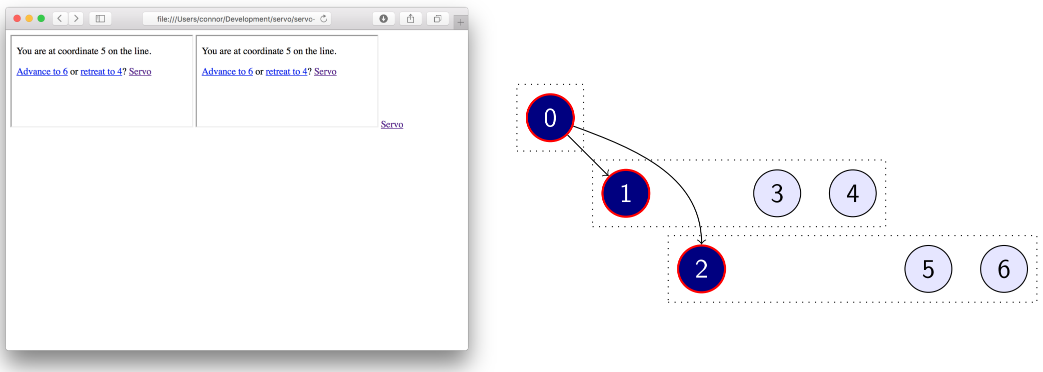
You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

Servo

Figure 33: Advance document 2 to 7

Traverse H by -4:


file:///Users/connor/Development/servo/servo-

You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

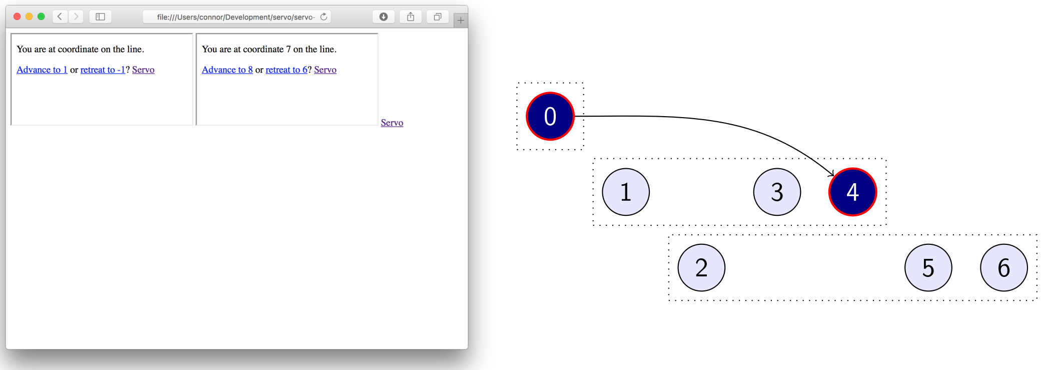
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 34: Traverse H by -4

Traverse H by 4:


file:///Users/connor/Development/servo/servo-

You are at coordinate on the line.

Advance to 1 or retreat to -1? Servo

You are at coordinate 7 on the line.

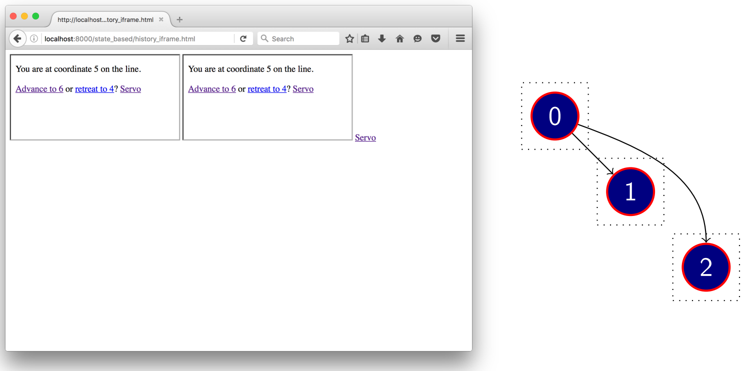
Advance to 8 or retreat to 6? Servo

Servo

Figure 35: Traverse H by 4

In this traversal, we are unable to determine the active entry for first entry as the state is *null*. This traversal does not satisfy Goal 1.

Chrome:


http://localhost...tory_iframe.html

localhost:8000/state_based/history_iframe.html

You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

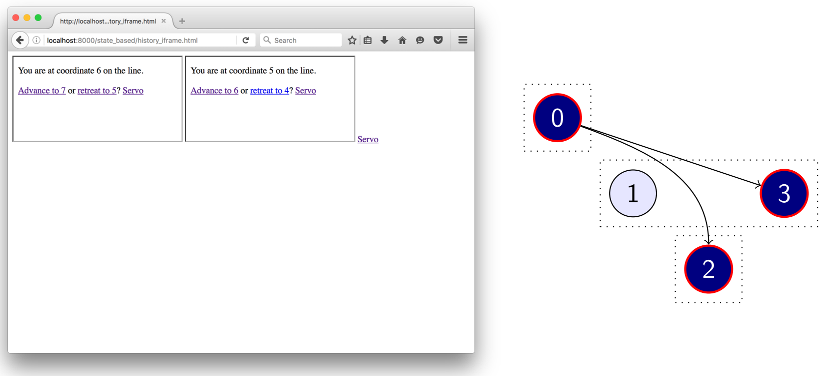
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 36: Initial State

Advance document 1 to 6:


http://localhost...tory_iframe.html

localhost:8000/state_based/history_iframe.html

You are at coordinate 6 on the line.

Advance to 7 or retreat to 5? Servo

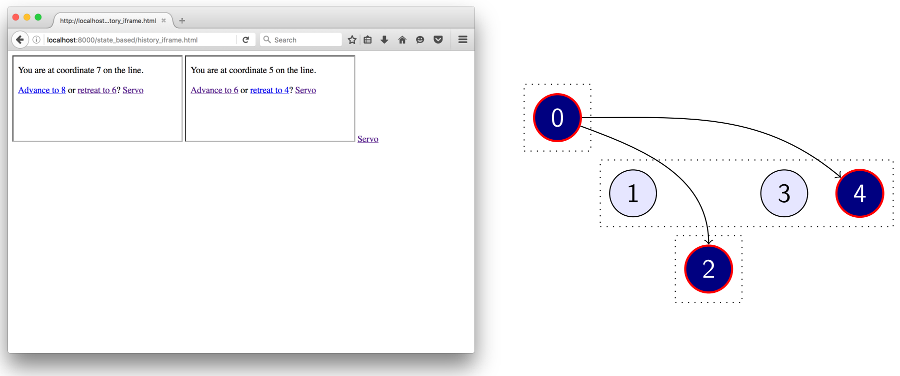
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 37: Advance document 1 to 6

Advance document 3 to 7:


http://localhost...tory_iframe.html

localhost:8000/state_based/history_iframe.html

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

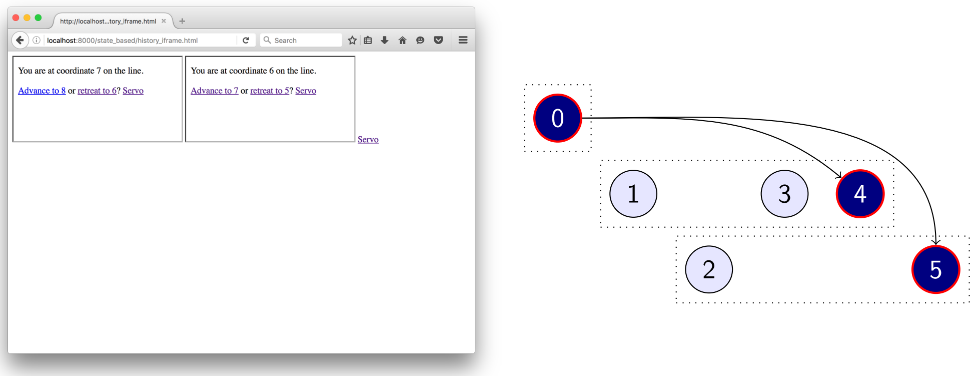
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 38: Advance document 3 to 7

Advance document 2 to 6:


http://localhost...tory_iframe.html

localhost:8000/state_based/history_iframe.html

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

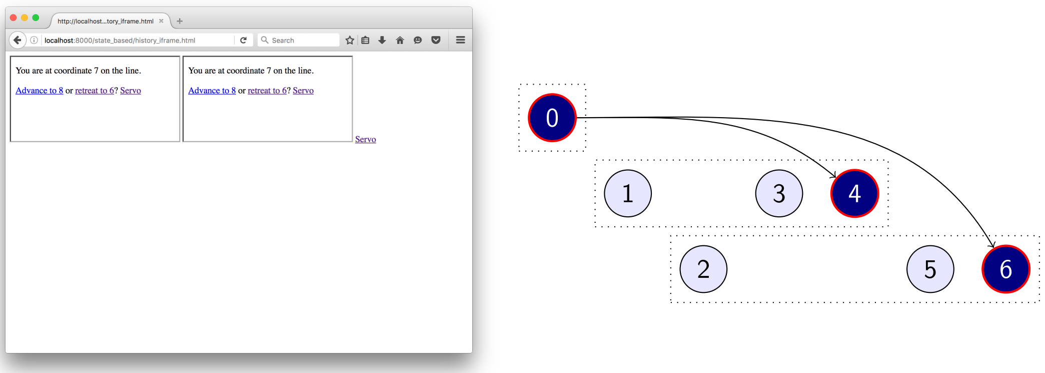
You are at coordinate 6 on the line.

Advance to 7 or retreat to 5? Servo

Servo

Figure 39: Advance document 2 to 6

Advance *document*2 to 7:


http://localhost...tory_iframe.html
localhost:8000/state_based/history_iframe.html
Search
0
1
3
4
2
5
6

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

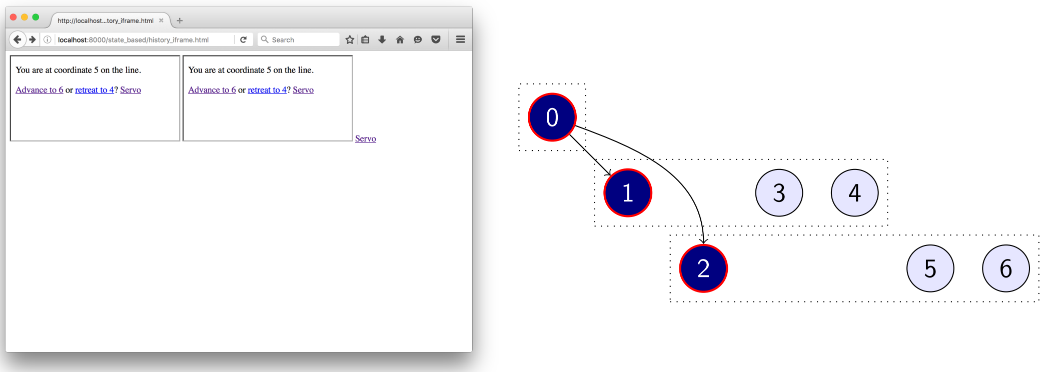
You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

Servo

Figure 40: Advance document 2 to 7

Traverse H by -4:


http://localhost...tory_iframe.html
localhost:8000/state_based/history_iframe.html
Search
0
1
3
4
2
5
6

You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

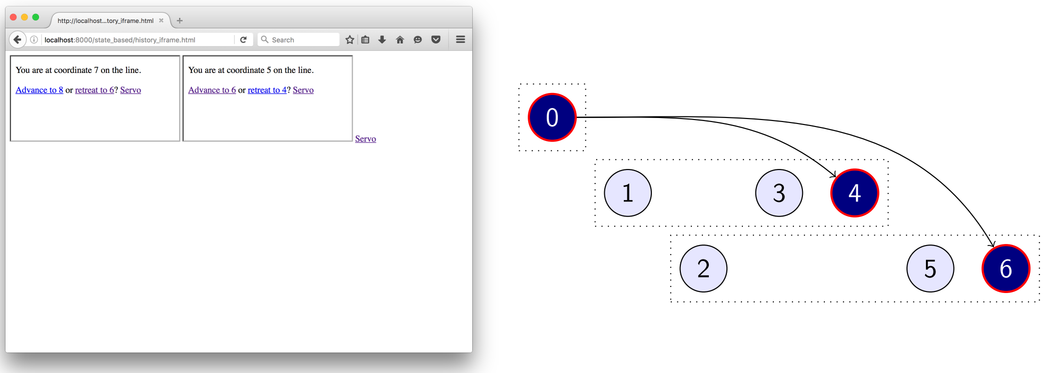
You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 41: Traverse H by -4

Traverse H by 4:


http://localhost...tory_iframe.html
localhost:8000/state_based/history_iframe.html
Search
0
1
3
4
2
5
6

You are at coordinate 7 on the line.

Advance to 8 or retreat to 6? Servo

You are at coordinate 5 on the line.

Advance to 6 or retreat to 4? Servo

Servo

Figure 42: Traverse H by 4

In this experiment, Chrome meets Goal 1.