

# Spatial Navigation on the Web

TPAC 2018  
Breakout Session

Jihye Hong, LG Electronics



# Overview

❑ Spatial Navigation

❑ Use cases

❑ Spec

- Heuristic processing model
- API

❑ Next Steps

❑ Demo

# Spatial Navigation?

- ❑ Two-dimensional navigation within the pages of web app.
- ❑ Navigate around your pages in an intuitive and robust way in various devices. (TV, PC, IVI, etc)

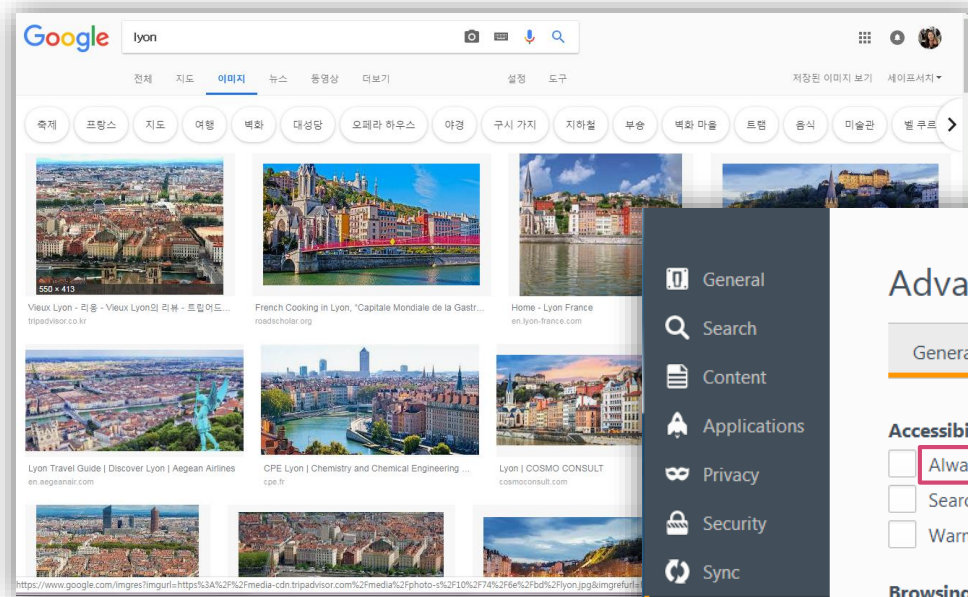


TV Web Application

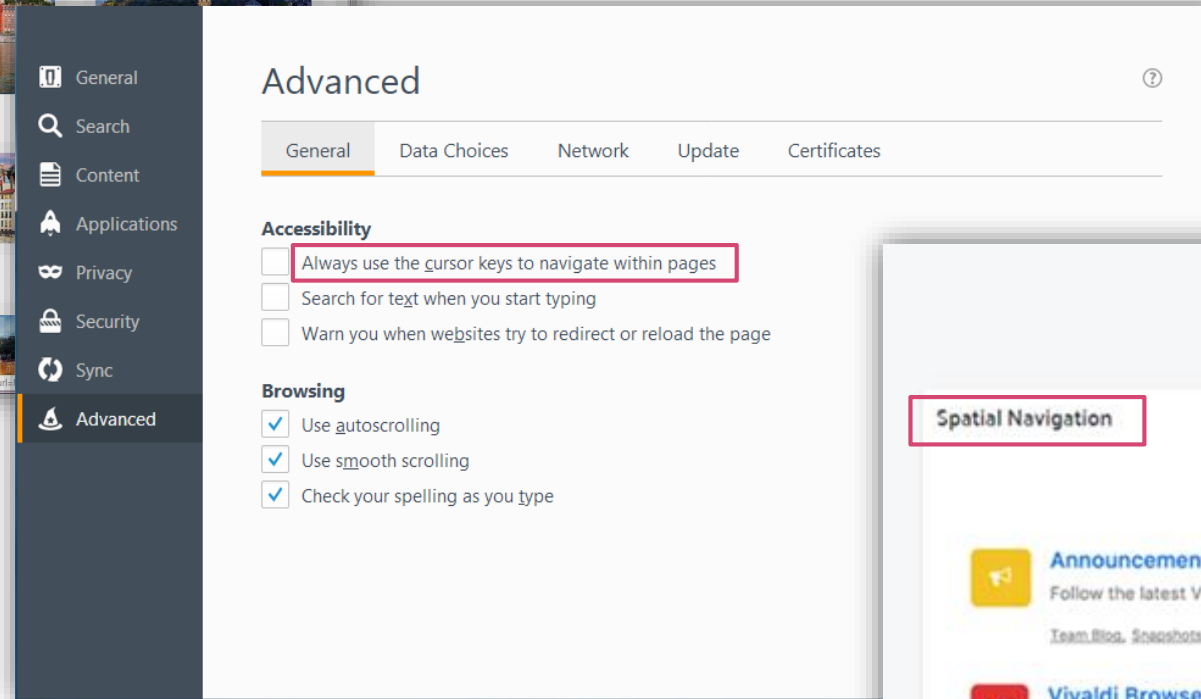


Grid-like Web Application

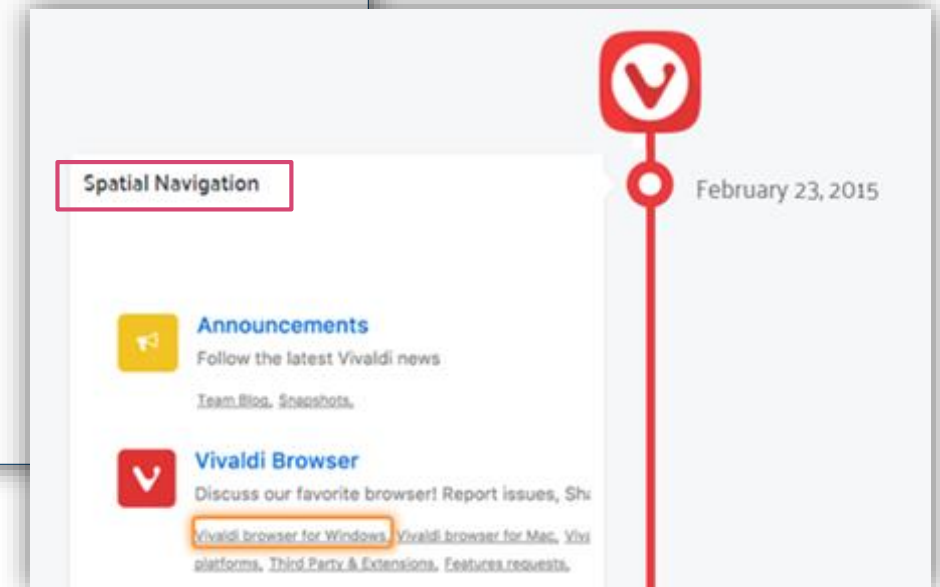
# Use cases



Google's Product  
(Image Search,  
Gmail,  
Google Docs)



Directional Cursor Navigation  
in Firefox



Spatial Navigation Feature  
in Vivaldi

# Use cases

THE NETFLIX  
TECH BLOG

Follow

May 18, 2017 · 7 min read


## Pass the Remote: User Input on TV Devices

by [Andrew Eichacker](#)

The Netflix TV team works with device manufacturers to explore new input methods (like your phone!) and improve the screens we watch our favorite shows on. Beyond that, we're testing the boundaries for content discovery and playback while bringing Netflix to more users around the world.

BBC

HomeGetting StartedOverviewWidgetsOtherJS DocTestingFAQ



## Focus Management

### Introduction

Spatial navigation (i.e. navigating around the UI via 5-point navigation) is handled for you by the framework if you make use of the *VerticalList*, *HorizontalList* and *Grid* widgets (or subclasses of them such as *HorizontalCarousel*).

- VerticalList* consumes up and down key events, preventing bubbling of the event to the list's parent widget if it successfully moved the focus to the next/previous item in the list
- HorizontalList* does the same for left and right key events
- Grid* consumes all directional (up, down, left and right) key events

Events that are not consumed bubble up through the Widget tree (not the DOM), until a higher-level widget can consume them. By nesting both Vertical and Horizontal lists within each other, you can build up a complex UI.

For more details about events, please see the [Events](#) section.

Download ZIP

Download TAR

View on GitHub

Follow @BBCOpenSource

Tweet

buildunknown

npm

spatial navigation

24 packages found

Sort Packages

Optimal

Popularity

Quality

Maintenance

Who's Hiring?

See all 19 companies

spatial-navigation-polyfill

A polyfill for the spatial navigation

spatial-navigation polyfill web

jihyerish published 1.0.0 · a month ago

lrud

Left, Right, Up, Down. A spatial navigation library for devices with input via directional controls

tv navigation tv-apps react react-tv focus focus management spatial navigation smart tv

thunderchild15 published 2.6.0 · a month ago

vue-spatialnavigation

Spatial navigation support for VueJS

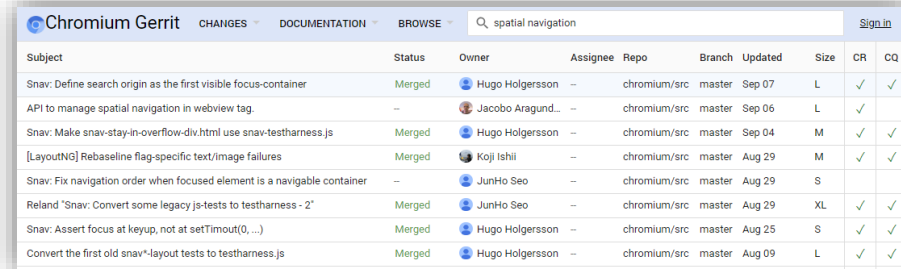
vue spatial navigation navigation tv typescript keyboard vue-spatialnavigation

twcapps published 1.2.1 · 4 months ago

# Need for Standard!

- ❑ Evolving navigation for evolving layout in the web
- ❑ Limitation of JS lib for handling iframe, shadow DOM
- ❑ Less customization for consistent UX

# History



Chromium Gerrit interface showing a list of patches related to spatial navigation. The table includes columns for Subject, Status, Owner, Assignee, Repo, Branch, Updated, Size, CR, and CQ.

Subject	Status	Owner	Assignee	Repo	Branch	Updated	Size	CR	CQ
Snav: Define search origin as the first visible focus-container	Merged	Hugo Holgersson	-	chromium/src	master	Sep 07	L	✓	✓
API to manage spatial navigation in webview tag.	-	Jacobo Aragund...	-	chromium/src	master	Sep 06	L	✓	✓
Snav: Make snav-stay-in-overflow-div.html use snav-testharness.js	Merged	Hugo Holgersson	-	chromium/src	master	Sep 04	M	✓	✓
[LayoutNG] Rebaseline flag-specific text/image failures	Merged	Koji Ishii	-	chromium/src	master	Aug 29	M	✓	✓
Snav: Fix navigation order when focused element is a navigable container	-	JunHo Seo	-	chromium/src	master	Aug 29	S		
Reland "Snav: Convert some legacy js-tests to testharness - 2"	Merged	JunHo Seo	-	chromium/src	master	Aug 29	XL	✓	✓
Snav: Assert focus at keypad, not at setTimeout(0, ...)	Merged	Hugo Holgersson	-	chromium/src	master	Aug 25	S	✓	✓
Convert the first old snav*-layout tests to testharness.js	Merged	Hugo Holgersson	-	chromium/src	master	Aug 09	L	✓	✓

17.11

Introduce the  
Spatial Navigation  
@W3C TPAC 2017

18.04

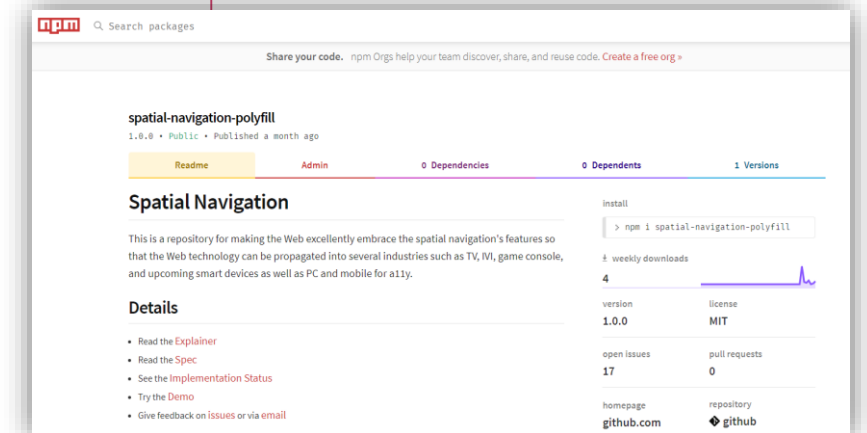
Introduce  
Spatial Navigation  
Spec Draft  
@CSS WG F2F

18.08

Contribute patch (#8)  
to Chromium upstream  
about the Heuristic  
Behavior of  
Spatial Navigation

18.09

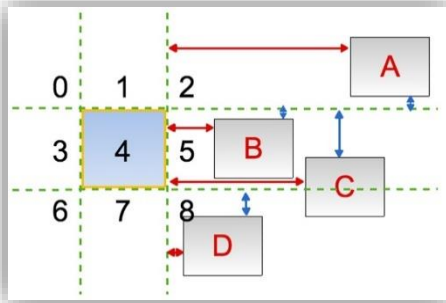
Release polyfill



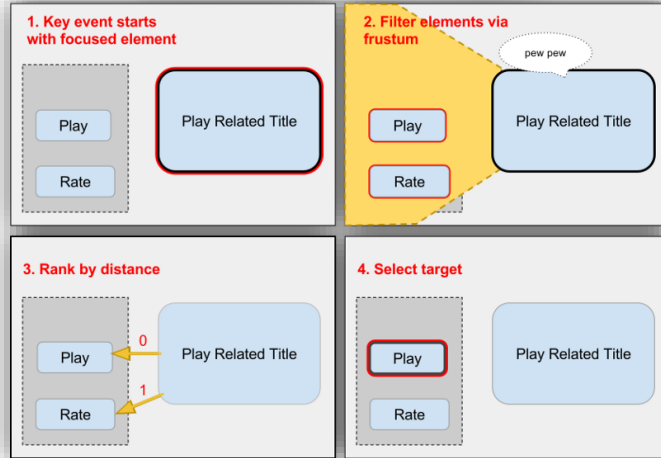
npm package page for spatial-navigation-polyfill. The page shows the package name, version (1.0.0), and a description: "This is a repository for making the Web excellently embrace the spatial navigation's features so that the Web technology can be propagated into several industries such as TV, IVI, game console, and upcoming smart devices as well as PC and mobile for a11y." It also includes a "Details" section with links to the explainer, spec, implementation status, demo, and feedback. The right sidebar shows installation instructions, weekly downloads (4), version (1.0.0), license (MIT), open issues (17), pull requests (0), homepage (github.com), and repository (github).

# Find the Primitive Features

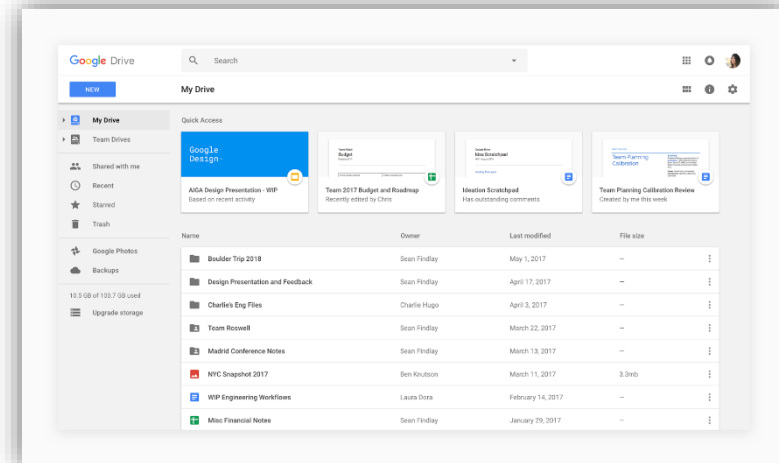
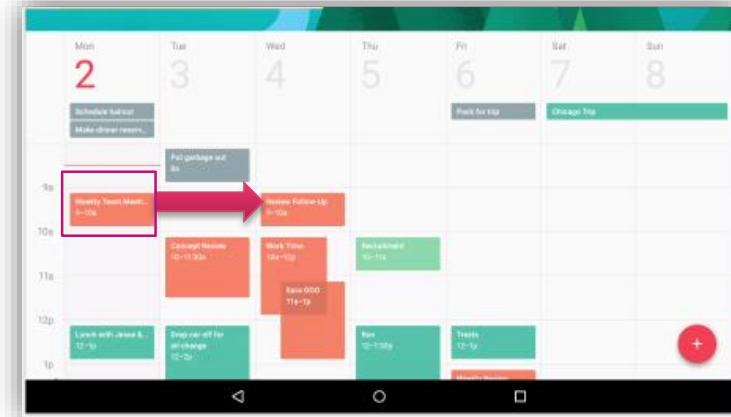
# Processing Model



## Focusing Left via Spatial Navigation



# APIs





# Spatial Navigation in WICG

❑ Processing Model

❑ APIs

TABLE OF CONTENTS	
1	Introduction
2	Module interaction
3	Overview
4	Triggering Spatial Navigation
5	JavaScript API
5.1	Triggering Navigation Programmatically
5.2	Low level APIs
6	Navigation Events
6.1	Interface NavigationEvent
6.2	Navigation Event Types
6.2.1	navbeforefocus
6.2.2	navbeforescroll
6.2.3	navnotarget
7	Processing Model
7.1	Groupings of elements
7.2	Navigation
7.3	Focus Navigation Heuristics

## Spatial Navigation

Editor's Draft, 18 October 2018

**This version:**  
<http://wicg.github.io/spatial-navigation>

**Issue Tracking:**  
[GitHub](#)  
[Inline In Spec](#)

**Editors:**  
[Jihye Hong](#) (LG Electronics)  
[Florian Rivoal](#) (Invited Expert)

[Copyright](#) © 2018 the Contributors to the Spatial Navigation Specification, published by the [Web Platform Incubator Community Group](#) under the [W3C Community Contributor License Agreement \(CLA\)](#). A human-readable [summary](#) is available.

---

### Abstract

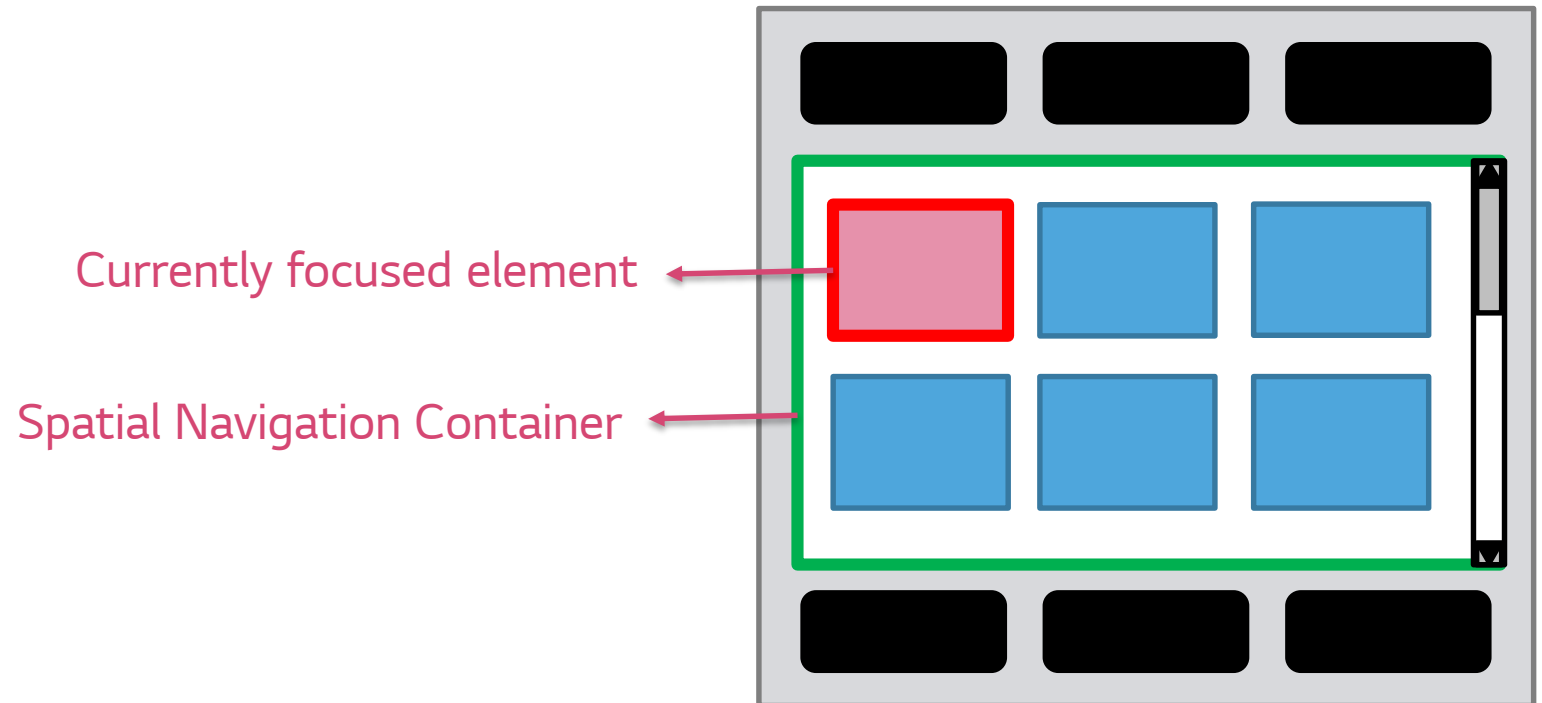
This specification defines a general model for navigating the focus using the arrow keys, as well as related CSS and JavaScript features.

### Status of this document

<https://wicg.github.io/spatial-navigation/>

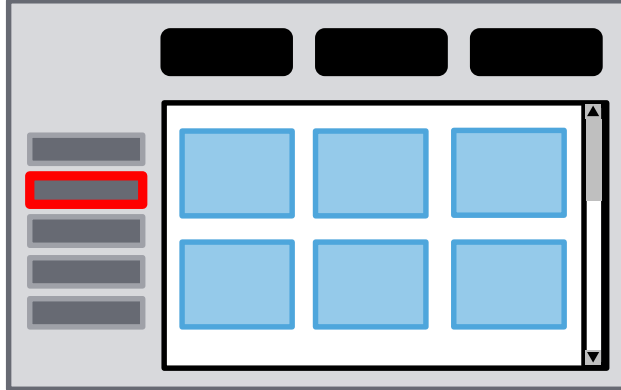
# Spatial Navigation Container

- ❑ Scope for searching the focusable elements from the currently focused element
- ❑ Document, scrollable element (by default)

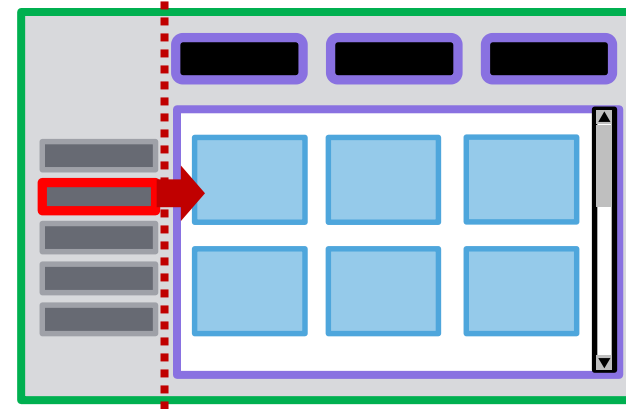


# Basic Spatial Navigation Heuristics

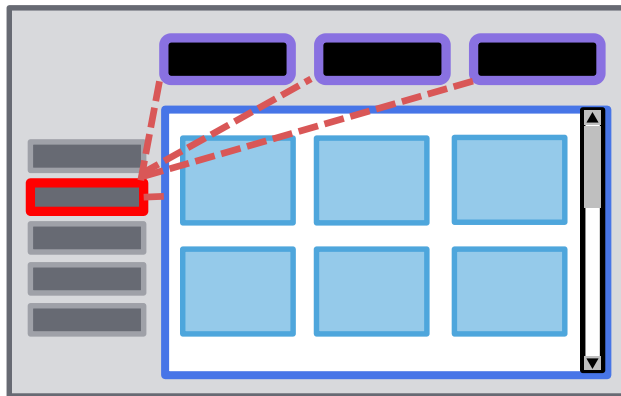
1. Pressing  from an **element**



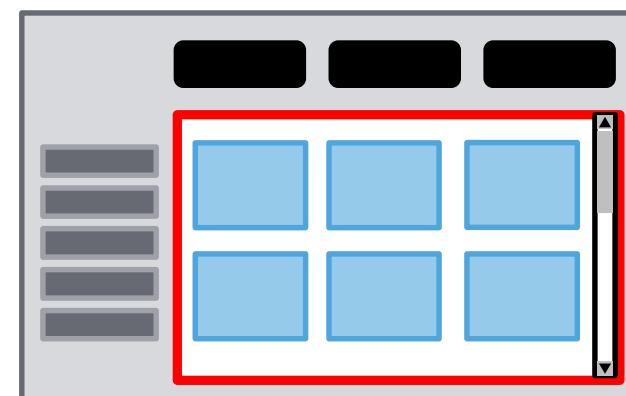
2. Find **candidates** within the **spatial navigation container**



3. Select the **best candidate**



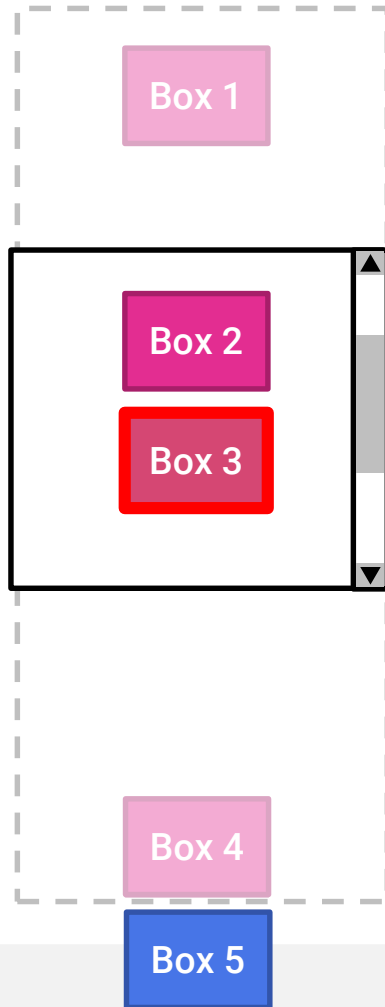
4. **Focus the best candidate**



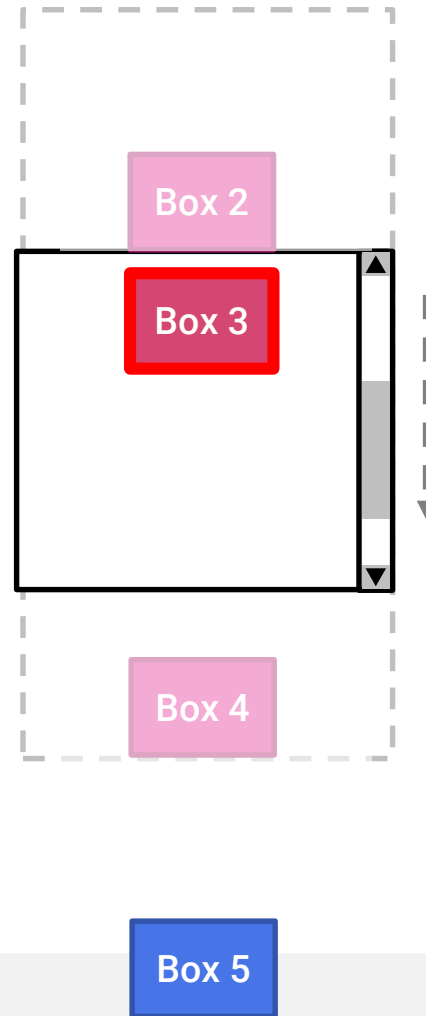
# Spatial Navigation Heuristic Behavior

## ❑ Combining the UA-defined arrow key behaviors

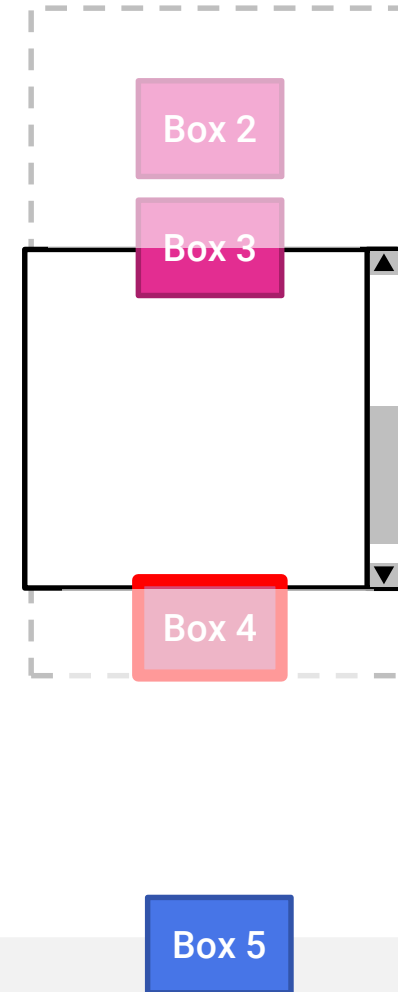
1. Box 3 gains the focus



2. Pressing  triggers the scrolling



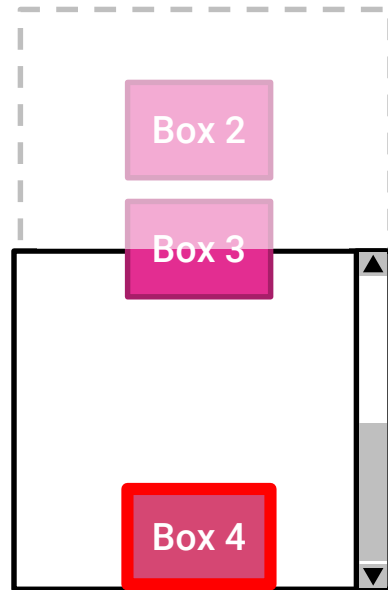
3. Box 4 gains the focus when it comes into the view



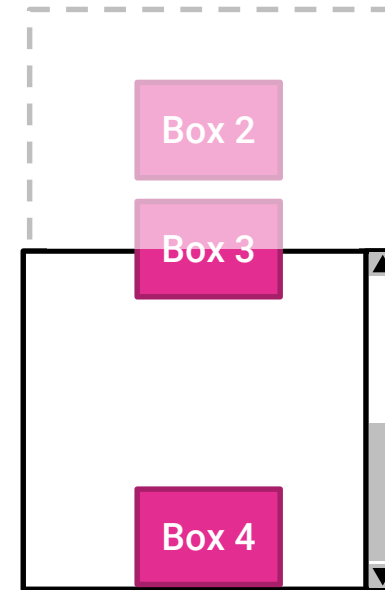
# Spatial Navigation Heuristic Behavior

## ❑ Combining the UA-defined arrow key behaviors

4. Pressing  moves down the scrollbar until the end



5. Pressing  moves the focus out of the scroll area and Box 5 gains the focus



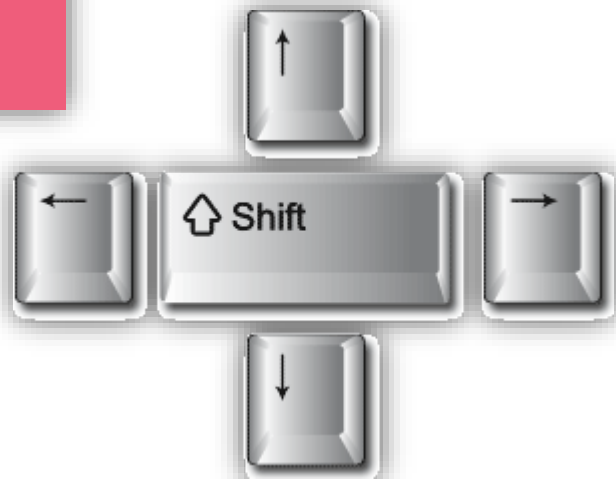
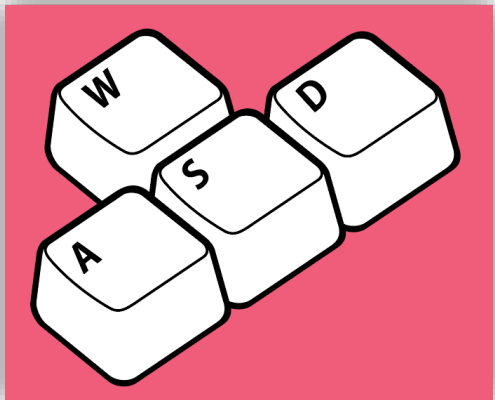
Box 5

Box 5

# APIs

## ❑ window.navigate(direction)

- Move the focus to the best candidate in the corresponding direction
- Mapping other key combinations

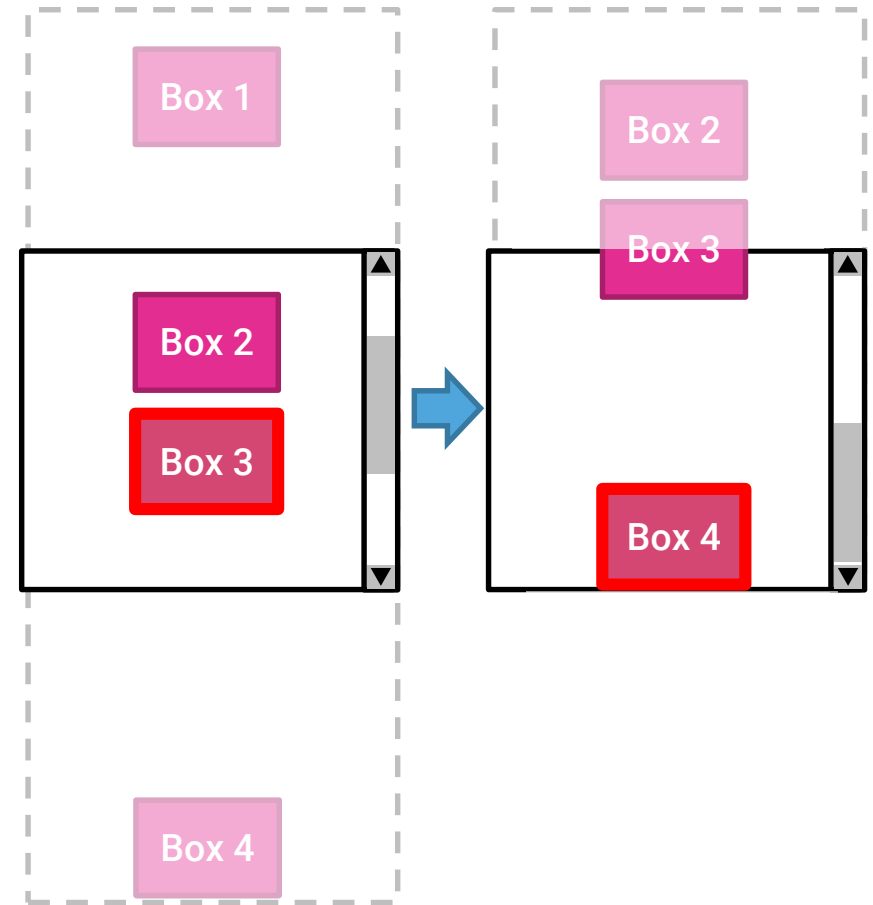


```
const redContainer = document.getElementsByClassName("container c1")[0];
redContainer.addEventListener('keydown', function(e) {
  const dir = WASD_KEY_CODE[e.keyCode];
  if(window.navigate && dir) {
    window.navigate(dir);
    e.preventDefault();
  }
}, true);
```

## ❑ `element.focusables(option)`

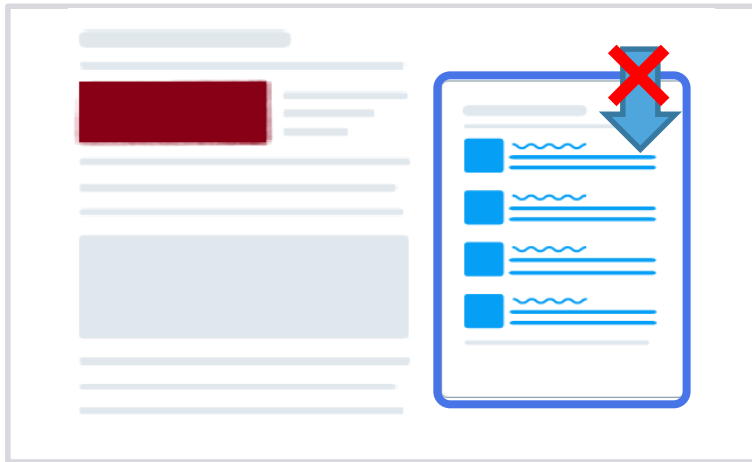
- Finds all the focusable elements inside the container
- Directly move the focus without manually scrolling

```
enum FocusableAreaSearchMode {  
    "visible",  
    "all"  
};  
  
dictionary FocusableAreasOptions {  
    FocusableAreaSearchMode mode;  
};
```

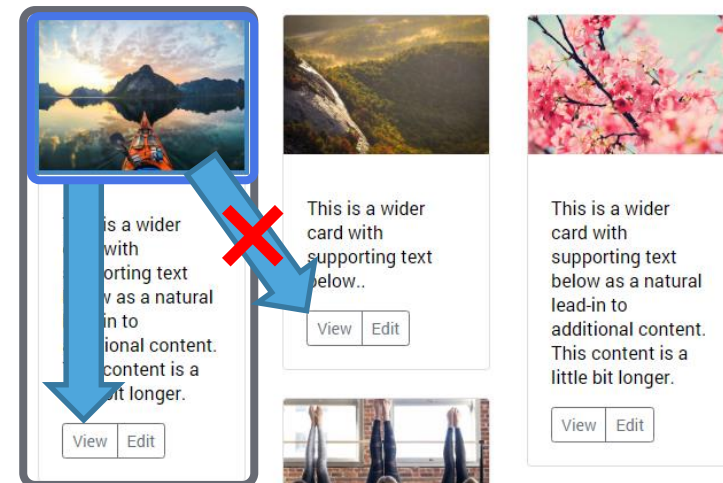


## ❑ Customize finding the best candidate

- [spatialNavigationSearch\(\)](#), [getSpatialNavigationContainer\(\)](#)
- [Navigate Events](#)
- [spatial-navigation-contain](#) CSS property



Infinite scroll



Grouped elements



## ❑ Customize finding the best candidate

- CSS UI nav-\* properties

- Auto

“The user agent determines which element to navigate the focus to in response to directional navigational input.”

Spatial Navigation Heuristics


§ 5.3. Keyboard control

§ 5.3.1. Directional Focus Navigation: the `'nav-up'`, `'nav-right'`, `'nav-down'`, `'nav-left'` properties

<i>Name:</i>	<code>'nav-up'</code> , <code>'nav-right'</code> , <code>'nav-down'</code> , <code>'nav-left'</code>
<i>Value:</i>	auto   <code>&lt;id&gt;</code> [ current   root   <code>&lt;target-name&gt;</code> ]?
<i>Initial:</i>	auto
<i>Applies to:</i>	all enabled elements
<i>Inherited:</i>	no
<i>Percentages:</i>	N/A
<i>Computed value:</i>	as specified
<i>Canonical order:</i>	per grammar
<i>Animation type:</i>	discrete

<https://drafts.csswg.org/css-ui/#nav-dir>

# Try the Spatial Navigation!

 **Spatial Navigation Laboratory**

All demo pages support spatial navigation. Please use arrow keys for moving the focus. ☐

- Introduction
- Processing model
  - Default focus moving
  - Overflow regions
  - Scrollable region
  - <iframe> element
  - <input> elements
  - Spatnav container
- API Function
  - getSpatnavContainer
  - focusableAreas
  - spatNavSearch
  - navigate
- Web applications

This site aims to show several behaviors of spatial navigation using [a polyfill](#).

Please use only arrow keys with several shortcuts to navigate all pages.

Submit bug reports, requests and comments on [github issues](#).

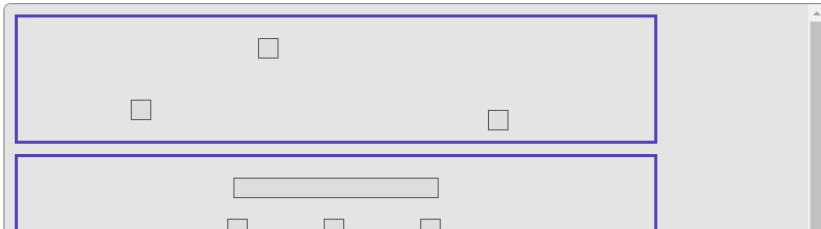
## 1. Why spatial navigation?

[Introduction of spatial navigation](#)

## 2. Processing model

**\* Default focus moving behavior**

You can check the **default spatial navigation behavior using arrow keys**.



<https://wicg.github.io/spatial-navigation/demo/>

<https://www.npmjs.com/package/spatial-navigation-polyfill>

# Next Steps

## ❑ Implementation

- Google
  - Chromium Open Source, Accessibility
- Vewd
  - Chromium Patch and Spec Reviews
- Vivaldi
  - Investigating about applying the Spatial Navigation

# Next Steps

## ❑ Spec Extension

- Handling the various focusable elements
- Improving the distance function
- Managing the focus

### Focusable Elements - Browser Compatibility Table

The following tables show which elements individual browsers consider focusable or tabbable (keyboard focusable). The tables are based on the [focusable test document](#).

Note that touch devices (without a physical keyboard) only show elements as tabbable (keyboard focusable), that can be navigated to through the on-screen keyboard (or "v: keyboard").

#### Table Of Contents

- [Document Elements](#)
- [Form Control Elements](#)
- [Form Element](#)
- [Fieldset Element](#)
- [Label Element](#)
- [Editable Elements](#)
- [TabIndex Attribute](#)
- [Navigation Elements](#)
- [Image Maps](#)
- [Media Elements](#)
- [Shadow DOM](#)
- [Iframe Element](#)
- [Embed Element](#)
- [Object Element](#)
- [SVG Element](#)
- [SVG foreignObject element](#)
- [SVG use element](#)
- [SVG Document in Iframe Element](#)
- [SVG Document in Embed Element](#)
- [SVG Document in Object Element](#)
- [Hidden Attribute](#)
- [CSS Property Visibility](#)
- [Canvas Fallback Content Elements](#)
- [Details Element](#)
- [Scrollable Elements](#)
- [Image Element Ismap Attribute](#)
- [CSS Flexbox Layout](#)
- [Table Elements](#)
- [Questionable Elements](#)
- [Footnotes](#)

<https://allyjs.io/data-tables/focusable.html>

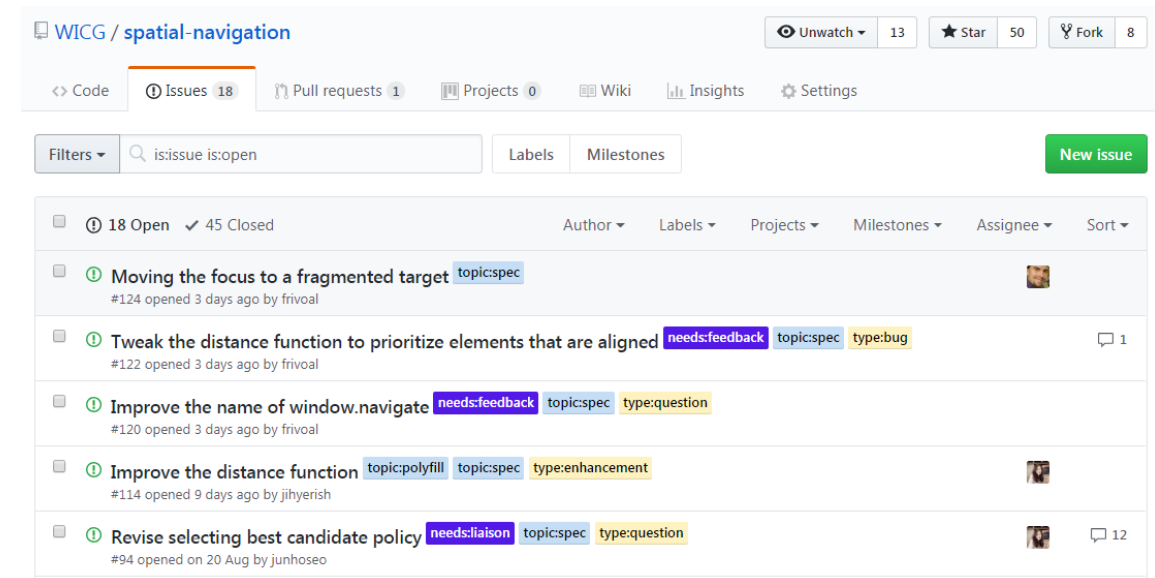
# Discussion

## ❑ Open Questions

- <https://github.com/WICG/spatial-navigation/blob/master/explainer.md#open-questions>

## ❑ Github issues

- <https://github.com/WICG/spatial-navigation/issues>



The screenshot shows the GitHub interface for the WICG / spatial-navigation repository. At the top, there are navigation tabs for Code, Issues (18), Pull requests (1), Projects (0), Wiki, Insights, and Settings. Below the tabs, there is a search bar with the text 'is:issue is:open' and buttons for 'Labels' and 'Milestones'. A 'New issue' button is located on the right. The main content area displays a list of 18 open issues. Each issue entry includes a title, a status icon (green circle with a white 'i'), a topic label (e.g., 'topic:spec'), a type label (e.g., 'type:bug'), and a comment count. The issues listed are:

- #124: Moving the focus to a fragmented target (topic:spec) - opened 3 days ago by frivoal
- #122: Tweak the distance function to prioritize elements that are aligned (needs:feedback, topic:spec, type:bug) - opened 3 days ago by frivoal
- #120: Improve the name of window.navigate (needs:feedback, topic:spec, type:question) - opened 3 days ago by frivoal
- #114: Improve the distance function (topic:polyfill, topic:spec, type:enhancement) - opened 9 days ago by jihyerish
- #94: Revise selecting best candidate policy (needs:laison, topic:spec, type:question) - opened on 20 Aug by junhoseo

**Thank you! : )**

