

# MAKING YOUR APP ACCESSIBLE







@勾三股四

Vue core team & AlibabaCloud

# AGENDA 议程

- background 背景
- web standards 规范
- strategy 策略
- utils & demo 工具 & 示例

# BACKGROUND 背景

-  /  = 15%
- 2-4% severely so 相对严重
-  图像  行动  声音  认知
- or ...

# BACKGROUND 背景

... normal people in such a situation

... 常人在上述方面的特殊环境下








 /   $\approx 100\%$

## **BACKGROUND 背景**

So it's actually meaningful for EVERYONE.

所以这是对每个人都有意义的事情。

# BACKGROUND 背景

			
			layout 布局 content 内容 details 细节

## **MOTOR 关于行动**

- PC: mouse 鼠标 only?
- Mobile: touch 触控 only?
- TV: remote control 遥控器 only?
- Keyboard  $\approx$  efficiency 效率

## **MOTOR 关于行动**

Creativities 交互创新: Apple Pencil, Voice 语音, Finger ID, Face ID ...

Any backup 有备选方案吗?



# WEB STANDARDS 标准

- WCAG: Web Pages 网页
- UAAG: Browsers 浏览器, Screen Readers
- ATAG: CMS, WYSIWYG 所见即得工具

# WEB STANDARDS 标准

- **WAI-ARIA**: Web App 富应用
- widget, role, props, states

highly recommended to read

强烈推荐深入学习

A simple secondary alert—check it out!

A simple success alert—check it out!

A simple danger alert—check it out!

A simple warning alert—check it out!

A simple info alert—check it out!

A simple light alert—check it out!

A simple dark alert—check it out!

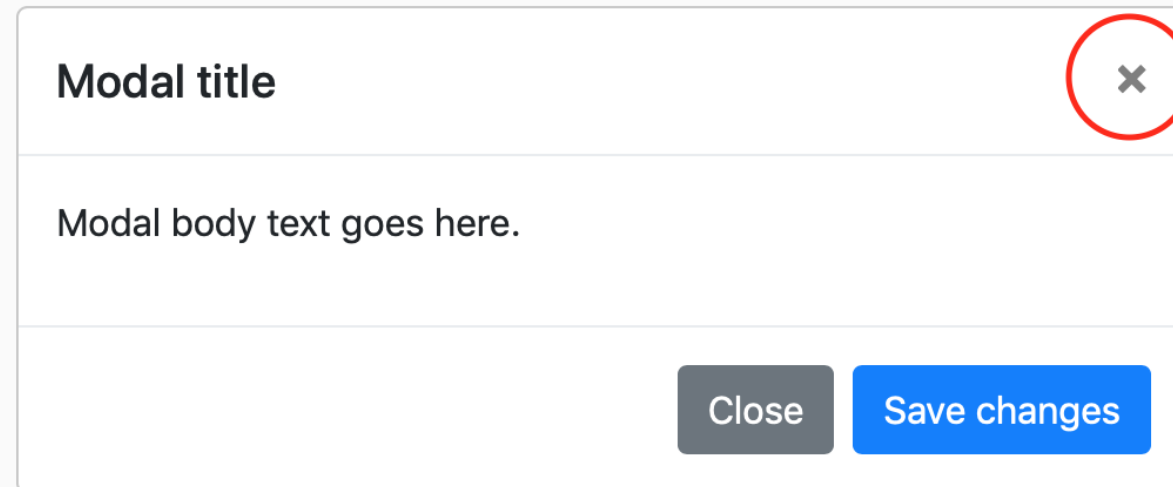
div.alert.alert-dark 497px × 50px  
Role alert

```
> document.querySelectorAll('[role]')
```

```
< ▼ NodeList (21) = $1
```

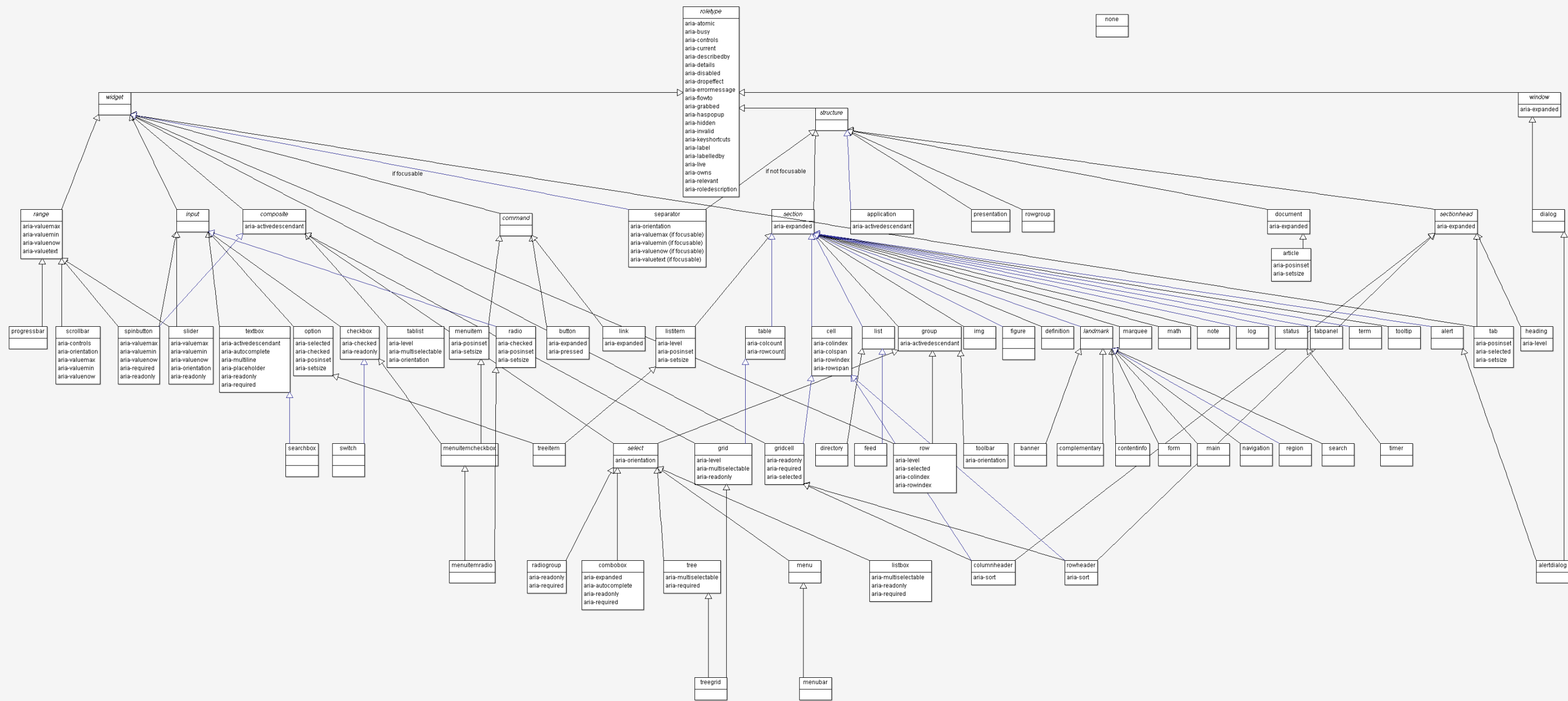
```
0 <input type="search" class="form-control ds-input"
1 ▶ <span class="ds-dropdown-menu" role="listbox" id=
2 ▶ <main class="col-12 col-md-9 col-xl-8 py-md-3 pl-
3 <div class="alert alert-primary" role="alert"> A si
4 <div class="alert alert-secondary" role="alert"> A
5 <div class="alert alert-success" role="alert"> A si
6 <div class="alert alert-danger" role="alert"> A sim
7 <div class="alert alert-warning" role="alert"> A si
8 <div class="alert alert-info" role="alert"> A simpl
9 <div class="alert alert-light" role="alert"> A simp
10 <div class="alert alert-dark" role="alert"> A simpl
11 ▶ <div class="alert alert-primary" role="alert">...</
12 ▶ <div class="alert alert-secondary" role="alert">...
13 ▶ <div class="alert alert-success" role="alert">...</
14 ▶ <div class="alert alert-danger" role="alert">...</d
15 ▶ <div class="alert alert-warning" role="alert">...</
16 ▶ <div class="alert alert-info" role="alert">...</div
17 ▶ <div class="alert alert-light" role="alert">...</di
18 ▶ <div class="alert alert-dark" role="alert">...</div
19 ▶ <div class="alert alert-success" role="alert">...</
20 ▶ <div class="alert alert-warning alert-dismissible
```

via [getbootstrap.com](https://getbootstrap.com)



```
<div class="modal" tabindex="-1" role="dialog">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Modal title</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span></button>
      </div>
      <div class="modal-body">
        <p>Modal body text goes here.</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div>
  </div>
</div>
```

via [getbootstrap.com](https://getbootstrap.com)



Big SVG Picture

Screen Reader & Browser	# of Respondents	% of Respondents
JAWS with Internet Explorer	424	24.7%
NVDA with Firefox	405	23.6%
JAWS with Firefox	260	15.1%
VoiceOver with Safari	172	10.0%
JAWS with Chrome	112	6.5%
NVDA with Chrome	102	5.9%
NVDA with IE	40	2.3%
VoiceOver with Chrome	24	1.4%
Other combinations	180	10.5%

<https://webaim.org/projects/screenreadersurvey7/#browsercombos>

# STRATEGY 实践策略

1. Atoms 原子组件
2. Groups 组件容器
3. Relationships 业务关系描述
4. App Layout 应用框架

灵感源自 *inspired from:* `atomic design`

# STRATEGY 1: ATOMS

- HTMLElement: tag name, attrs
- Component: role, props, states, tabindex

use them properly 把它们的语义用好用对



```
<label for="foo">
  Foo
</label>
<input
  id="foo"
  type="checkbox"
  checked
/>
```

```
<div id="label">
  Foo
</div>
<div
  tabindex="0"
  role="checkbox"
  aria-labelledby="label"
  aria-checked="true"
  class="checkbox"
></div>
```

**BTW YOU WOULD  
USE *ARIA* A LOT FROM NOW ON ...**

接下来你会用到好多 *ARIA*

## **STRATEGY: ATOMS**

Supply text alternatives: icons, images, photos, charts ...

提供必要的文本替代品

# **STRATEGY 1: ATOMS**

Supply long description for some short text

为一些缩写或简短的文字  
配以更详细的描述

# STRATEGY 2: GROUPS

menu / tree / list / table / grid / ...

a `treegrid` example via W3C

*All above could be encapsulated by a component.*

**上述两方面内容均可封装在组件库内**

by component lib authors.

# STRATEGY 3: RELATIONSHIPS

- `aria-flowto`
- `aria-controls`, `aria-owns`
- `aria-activedescendant`
- `aria-errormessage`

interaction logic 描述交互逻辑关系



a `listbox example` via W3C

# STRATEGY 3: RELATIONSHIPS

- `aria-labeledby`
- `aria-describedby`
- `aria-details`

descriptions more than label

更多的文本描述关系

*The relationships should be specified with business logic together.*

**这部分关系描述是需要业务研发团队根据实际情况来完成的**

by business dev team.

## STRATEGY 4: LAYOUT

- landmarks 应用全局布局
- "skip to content" 开门见山
- current 路由描述

# STRATEGY 4: LAYOUT

- `live region(s)` 实时播报

to announce dynamic content changes through assistive technologies

用来通过读屏软件播报动态内容的变化

*It should be supplied in the scaffold.*

**这部分内容应该封装在应用脚手架内**

by scaffold generator/authors.

# STRATEGY: 总结

- Atoms, Groups: component lib 组件库
- Relationships: business team 业务团队
- App Layout: scaffold 脚手架

**SO ...**



# ROUTINES 常规操作

- `role, aria-*` 特性
- ID refs 全局唯一引用
- `tabindex, focus control` 焦点控制
- `modal dialog` 模态对话框
- `keyboard shortcuts` 快捷键
- `live region` 实时播报

**THAT'S WHY I WROTE**

**VUE A11Y UTILS**

**LET'S DEMO NOW**

**EXAMPLE 1:**  
`<VueAria>` **COMPONENT &**  
    `MixinId` **MIXIN**  
    `MixinTravel` **MIXIN**

open

**EXAMPLE 2:**  
`<VueAria>` **COMPONENT &**  
`MixinId` **MIXIN**  
`<VueFocusTrap>` **COMPONENT**

open

# EXAMPLE 3:

MixinShortcuts **MIXIN**

open

# EXAMPLE 4:

## <VueLive> COMPONENT

open

**YOU CAN SEE THE DOCS HERE:**

<https://jinjiang.github.io/vue-a11y-utils/>



**AND MORE EXAMPLES:**

<https://jinjiang.github.io/vue-a11y-examples/>

# CONCLUSIONS 总结

- background 背景
- web standards 规范
- strategy 策略
- utils & demo 工具 & 示例

## AND DON'T FORGET 额外的

- pushing designer for 推动设计团队
  - accessible color pattern 配色方案
  - accessible interactions 交互改进
- testing 测试
- continuous iterations 持续迭代

# REFERENCES

- Vue Accessibility Guide (wip PR)
- Element, VEUI, eslint plugin
- W3C aria & practices

# THANKS

 jinjiang  zhaojinjiang@me.com