

# Shadow DOM Insertion Points & Events

Cory House  
bitnative.com  
Twitter: @housecor



**pluralsight**   
hardcore developer training

# Agenda

**Content  
Insertion  
Points**

**Shadow  
Insertion  
Points**

**Events**

# Insertion Points

`<content>`

creates insertion point

`<content select="img">`

“insert images here”

`<content>`

“insert anything not already inserted”



Shadow DOM Insertion Points with <content>

Demo

# Insertion Points vs Distributed Nodes

`<content>` creates an **insertion point**.

Elements inserted at this insertion point are called **distributed nodes**.

# Insertion Points Don't Move DOM

`<content>` merely says “Show this content from the Light DOM here”.

The Distributed Node remains part of the Light DOM.



Yep, this is a weird concept.

# Can't Traverse Into Distributed Nodes

<content></content>

Any content that is distributed here can't be traversed  
(with traditional APIs at least)

Traversing Distributed Nodes

Demo



# Supported <content> Selectors

Type selector

```
<content select="p"></content>
```

Class selector

```
<content select=".target"></content>
```

Attribute selector

```
<content select="input type=[text]"></content>
```

ID selector

```
<content select="#target"></content>
```

Negation pseudo-class

```
<content select=":not(.ugly)"></content>
```

Universal selector

```
<content></content>
```



Catch all

Content Selectors

Demo

# Shadow Insertion Points

<shadow>

Youngest tree is the last one added. Previous trees are called older trees.

Only the first shadow insertion point in a shadow tree is recognized.

Shadow Insertion Points with <shadow>

Demo

# getDistributedNodes

What elements are being distributed into this <content> tag?

Returns a list of distributed nodes in a <content> element.

# **getDestinationInsertionPoints**

Where is this DOM element being inserted?

Returns list of insertion points a node is distributed into

**Listing Distributed Nodes and Destination Insertion Points**

**Demo**

**Listing Destination Insertion Points**

**Demo**



# Event Retargeting

Events in the Shadow DOM are retargeted

Events will appear to come from the shadow host

Event Retargeting

Demo

# Distributed Nodes Aren't Retargeted

Events aren't retargeted for elements distributed via `<content>`

Distributed Nodes Aren't Retargeted

Demo

# Some Events Just Stop

**abort**

**error**

**select**

**change**

**load**

**reset**

**resize**

**scroll**

**selectstart**

Some Shadow Events Stop

Demo

# Summary

Create Insertion points via <content>

Create shadow insertion points with <shadow>

Distributed nodes and Insertion points

List distributed nodes with `getDistributedNodes`

List points a node is distributed using `getDestinationInsertionPoints`

Many events are retargeted. A few are stopped.