

Advances in TVMLKit

Session 202

Trevor Cortez, Localization Engineer

Parry Panesar, tvOS Engineer

Jeremy Foo, tvOS Engineer

Right-to-left languages

Template optimizations

Development using Web Inspector

Right-to-Left Languages

Trevor Cortez, Localization Engineer

Right-to-Left Languages

Arabic and Hebrew new in tvOS 11

Right-to-Left Languages

Arabic and Hebrew new in tvOS 11

400 million speakers worldwide

Title

Subtitle

Text 1 Text 2 TV-14 HD

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate veli... [MORE](#)



Title 1



Title 2

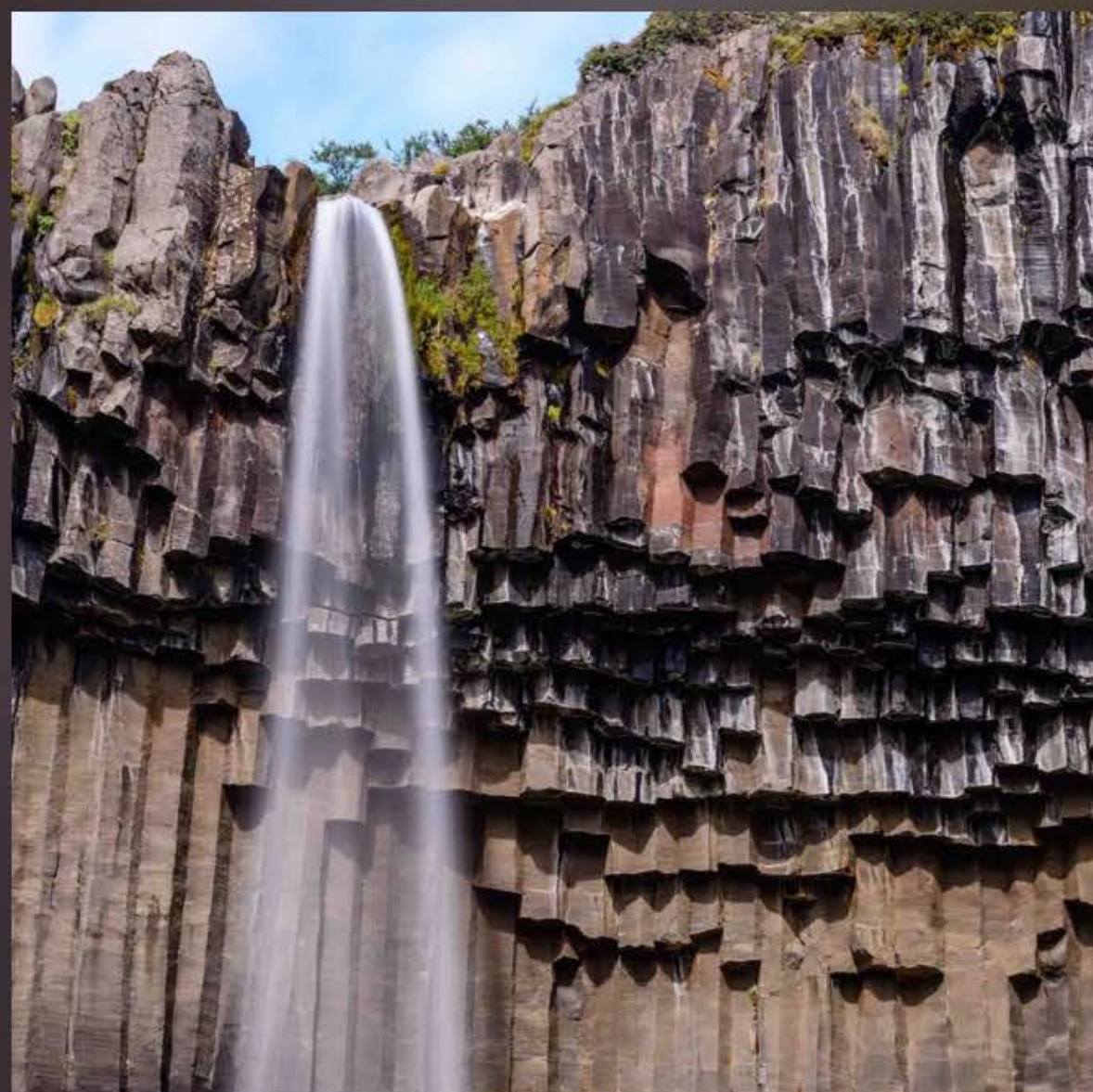


Title 3



Title





כותרת ראשית

כותרת-משנה

גוף א' גוף ב' **HD** **TV-14**

זהו מלל לדוגמא, אשר יוחלף בסופו של דבר במלל סופי, במובן הסופי של המילה, לרבות כל התוכן הסופי, בעיצובו הסופי, באופן סופי בהחלט. זהו מלל לדוגמא, אשר יוחלף בסופו של דבר במלל סופי, במובן הסופי של המילה, לרבות כל התוכן הסופי, בעיצובו הסופי, באופן סופי בהחלט. זהו מלל לדוגמא, אשר יוחלף בסופו של דבר במלל... עוד



כותרת ראשית
ג'



כותרת ראשית
ב'



כותרת ראשית
א'

כותרת ראשית



Title

SECTION HEADER

Title 1 6

SECTION HEADER

Title 2 10

Title 3 6

Loading...



Title 1



כותרת ראשית



כותרת ראשית א'

6

כותרת מקטע
כותרת ראשית א'

10

כותרת מקטע
כותרת ראשית ב'

6

כותרת ראשית ג'



טוען...

Right-to-Left Languages

Arabic and Hebrew new in tvOS 11

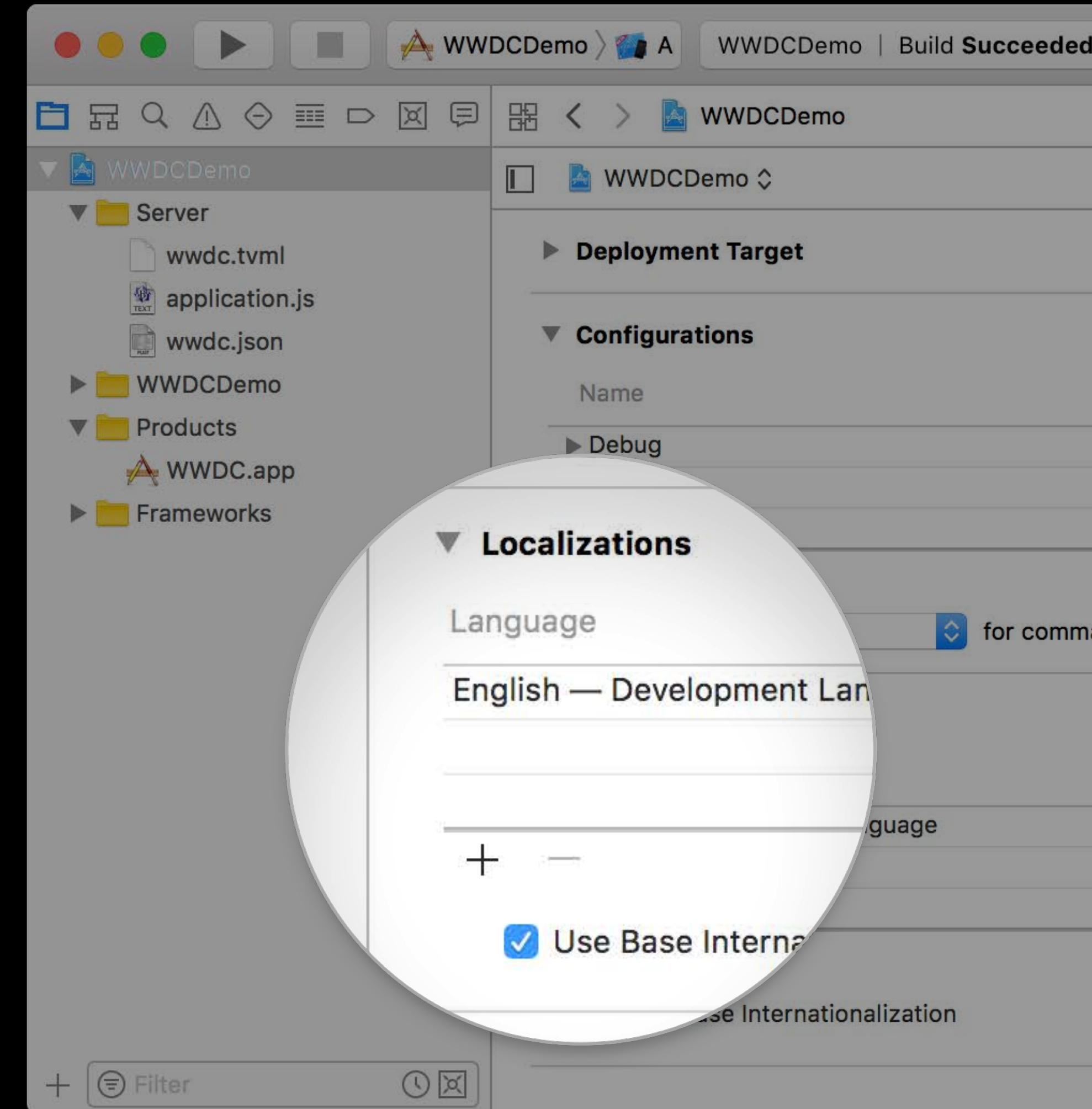
400 million speakers worldwide

Built in support in TVMLKit

Right-to-Left Languages

Adding support

Add Arabic or Hebrew in your Project Settings



Right-to-Left Languages

Layout

Text alignment

Images

Right-to-Left Languages

Layout



NEW

New context aware values for `tv-align` and `tv-position`

`leading`, `trailing` resolve to `right` and `left` at runtime

Right-to-Left Languages

Layout

NEW

New context aware values for `tv-align` and `tv-position`

`leading`, `trailing` resolve to `right` and `left` at runtime

```
.collectionListLockupImage {
    tv-position: right;
}

.sectionHeaderTitle {
    tv-align: left;
}
```

Right-to-Left Languages

Layout

NEW

New context aware values for `tv-align` and `tv-position`

`leading`, `trailing` resolve to `right` and `left` at runtime

```
.collectionListLockupImage {  
    tv-position: right;  
}  
  
.sectionHeaderTitle {  
    tv-align: left;  
}
```



Right-to-Left Languages

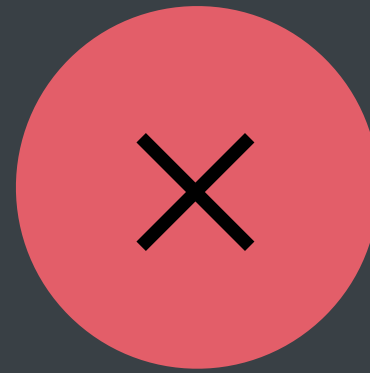
Layout

NEW

New context aware values for `tv-align` and `tv-position`

`leading`, `trailing` resolve to `right` and `left` at runtime

```
.collectionListLockupImage {  
    tv-position: right;  
}  
  
.sectionHeaderTitle {  
    tv-align: left;  
}
```



```
.collectionListLockupImage {  
    tv-position: trailing;  
}  
  
.sectionHeaderTitle {  
    tv-align: leading;  
}
```



Right-to-Left Languages

Layout

NEW

Adjust value inside media query for Layout Direction

- Margin
- Padding

Right-to-Left Languages

Layout

NEW

Adjust value inside media query for Layout Direction

- Margin
- Padding

```
@media (layout-direction: ltr) {  
    .my_interesting_image { margin: 0 12 0 0; }  
}  
  
@media (layout-direction: rtl) {  
    .my_interesting_image { margin: 0 0 0 12; }  
}
```

Right-to-Left Languages

Layout

NEW

Adjust value inside media query for Layout Direction

- Margin
- Padding

```
@media (layout-direction: ltr) {  
    .my_interesting_image { margin: 0 12 0 0; }  
}  
  
@media (layout-direction: rtl) {  
    .my_interesting_image { margin: 0 0 0 12; }  
}
```

Right-to-Left Languages

Layout

NEW

Adjust value inside media query for Layout Direction

- Margin
- Padding

```
@media (layout-direction: ltr) {  
  .my_interesting_image { margin: 0 12 0 0; }  
}  
  
@media (layout-direction: rtl) {  
  .my_interesting_image { margin: 0 0 0 12; }  
}
```

Right-to-Left Languages

Text alignment

Left

Center

Right

Lorem Ipsum	Lorem Ipsum	Lorem Ipsum
أبجد هوز حطي	أبجد هوز حطي	أبجد هوز حطي

Right-to-Left Languages

Text alignment

Left

Center

Right

Lorem Ipsum

Lorem Ipsum

Lorem Ipsum

أبجد هوز حطي

أبجد هوز حطي

أبجد هوز حطي

Right-to-Left Languages

Text alignment

Left

Center

Right

Natural

Lorem Ipsum	Lorem Ipsum	Lorem Ipsum	Lorem Ipsum
أبجد هوز حطي	أبجد هوز حطي	أبجد هوز حطي	أبجد هوز حطي

Right-to-Left Languages

Images

System

INFO

Help >

Legal >

What's New? >

MAINTENANCE

Software Updates >

Reset >

Restart

النظام

المعلومات

< المساعدة

< قانوني

< ما الجديد؟

الصيانة

< تحديثات البرامج

< إعادة تعيين

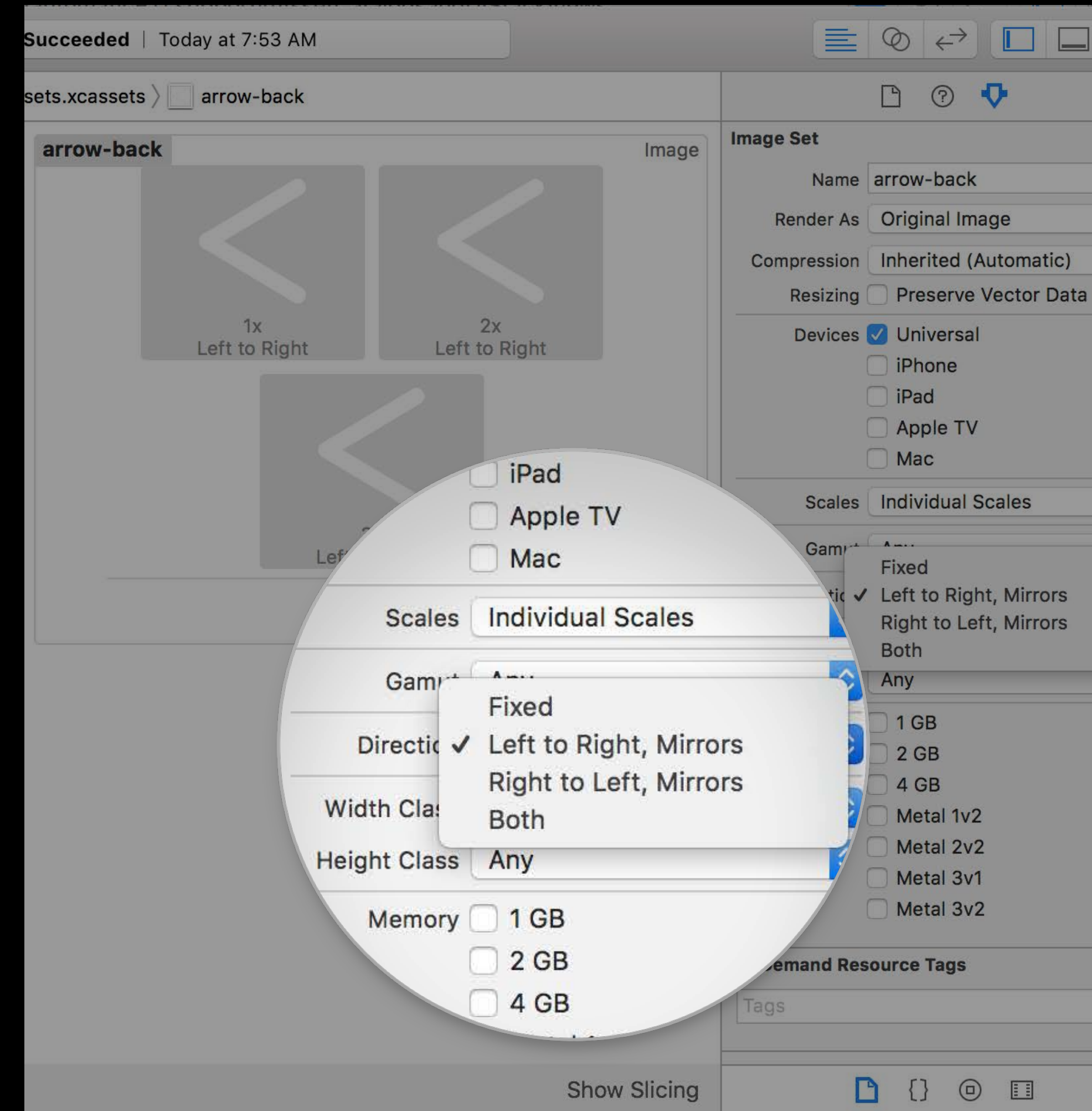
إعادة التشغيل

Right-to-Left Languages

Resource images

For Resource images use asset catalogs

Specify dedicated images or mirror for RTL



Right-to-Left Languages

Remote images

NEW

For server-side images, use layout direction media queries to load direction-specific assets

```
<img srcset="url1 (layout-direction:ltr), url2 (layout-direction:rtl)" ... />
```

Right-to-Left Languages

Remote images

NEW

For server-side images, use layout direction media queries to load direction-specific assets

```
<img srcset="url1 (layout-direction:ltr), url2 (layout-direction:rtl)" ... />
```

Right-to-Left Languages

Remote images

NEW

For server-side images, use layout direction media queries to load direction-specific assets

```
<img srcset="url1 (layout-direction:ltr), url2 (layout-direction:rtl)" ... />
```

Demo

Recap

Leading and trailing values for layout

Media queries for margin and padding

Natural text alignment

Alternate image assets as necessary

Test with RTL pseudolocalization language

Right-to-Left Languages

Handling custom views

Auto Layout with leading/trailing constraints

`effectiveLayoutDirection` when managing frames

Internationalization Best Practices

WWDC 2016

What's New in International User Interfaces

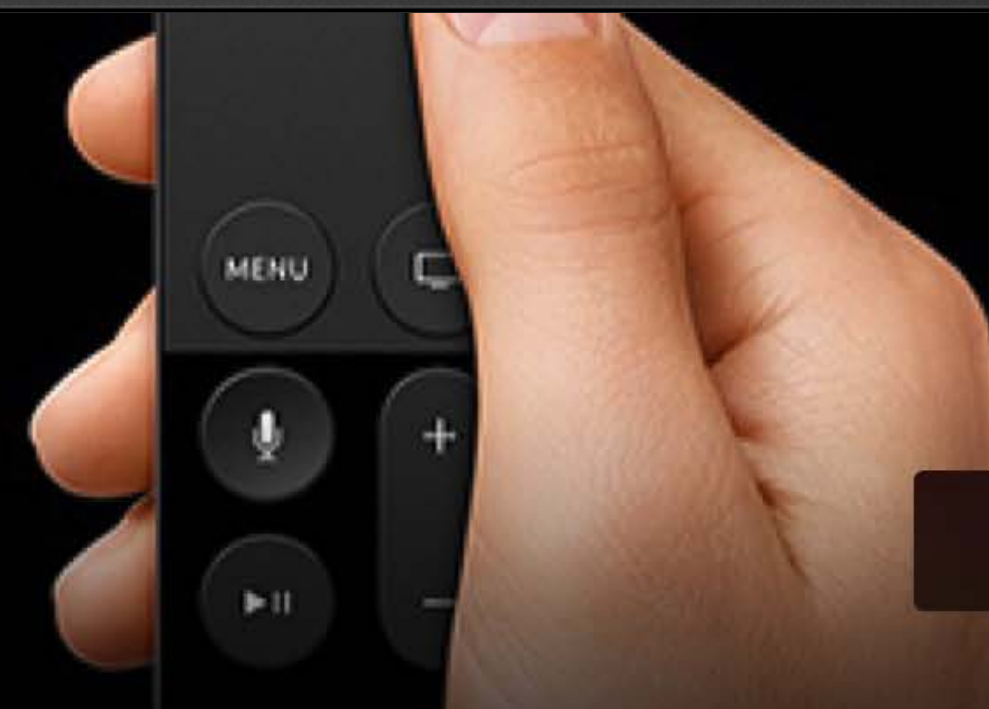
WWDC 2016

Template Optimizations

Parry Panesar, tvOS Engineer

Developing tvOS Apps Using TVMLKit

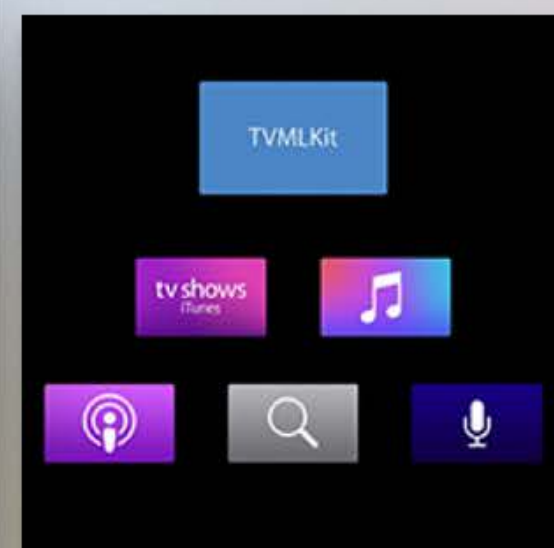
WWDC 2016 • SESSION 229 • APP FRAMEWORKS



Past Sessions



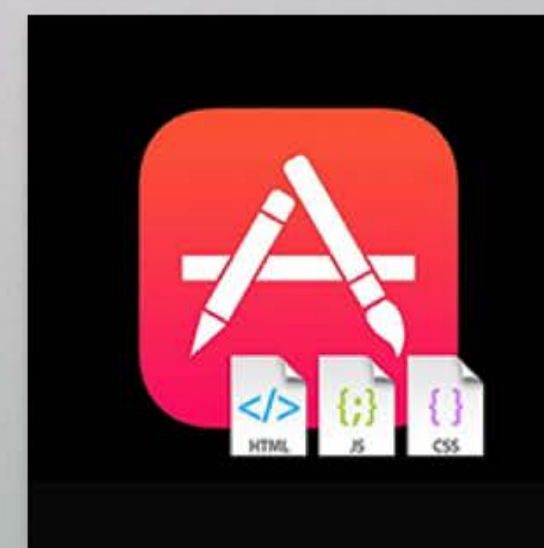
Internationalization Best Practices



Developing tvOS Apps with TVMLKit



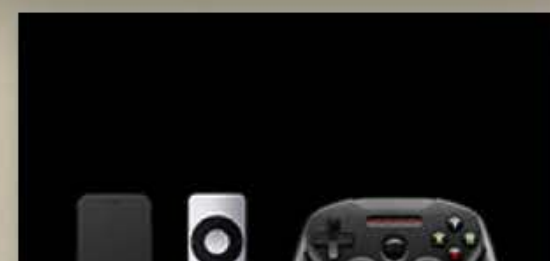
Developing tvOS Apps with AVKit



Optimizing Web Content

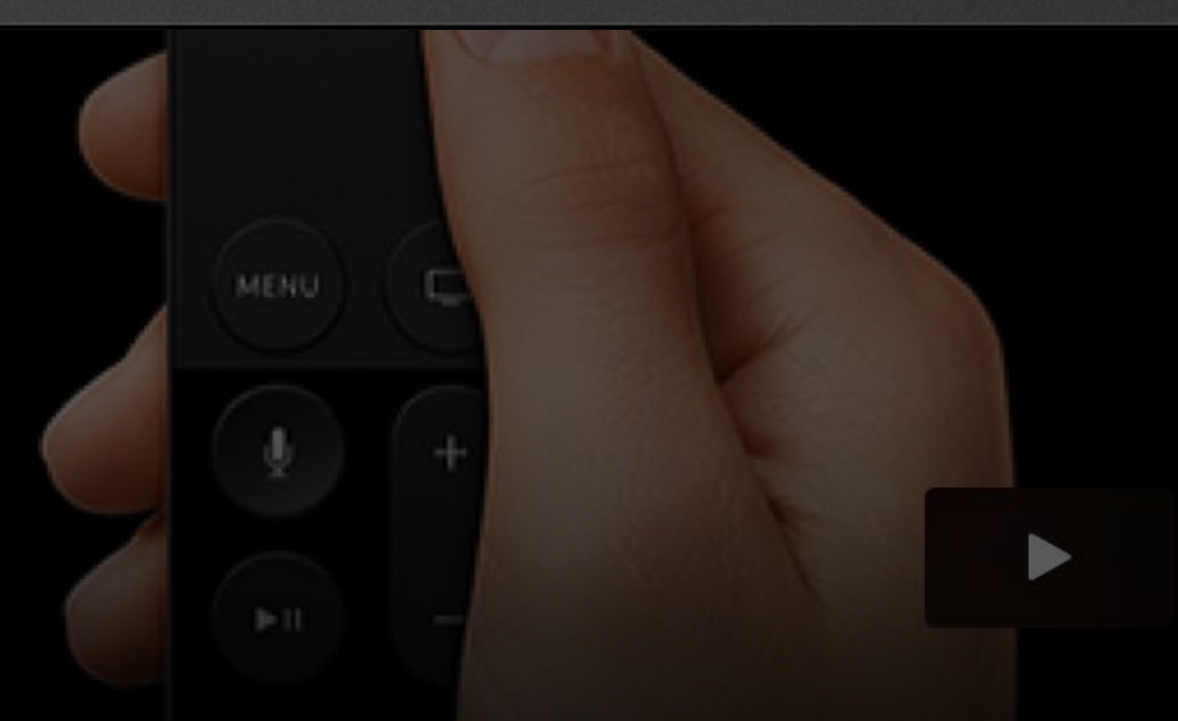


What's New in Internationalization

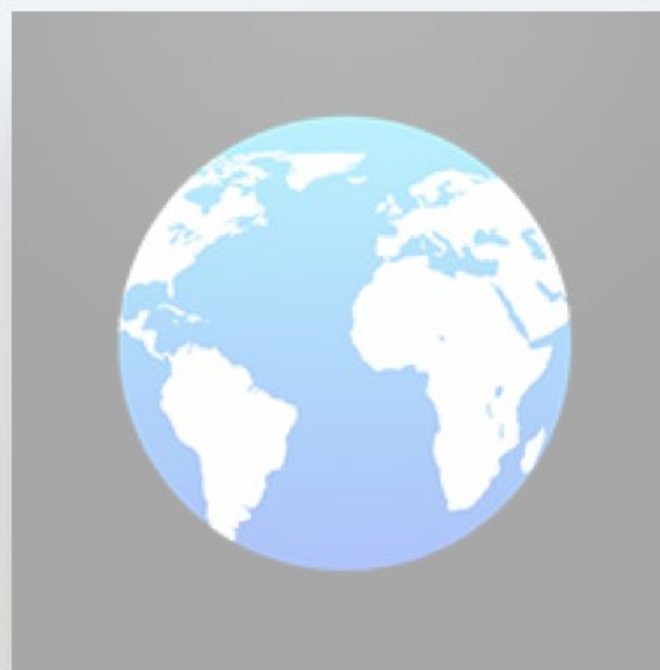


Developing tvOS Apps Using TVMLKit

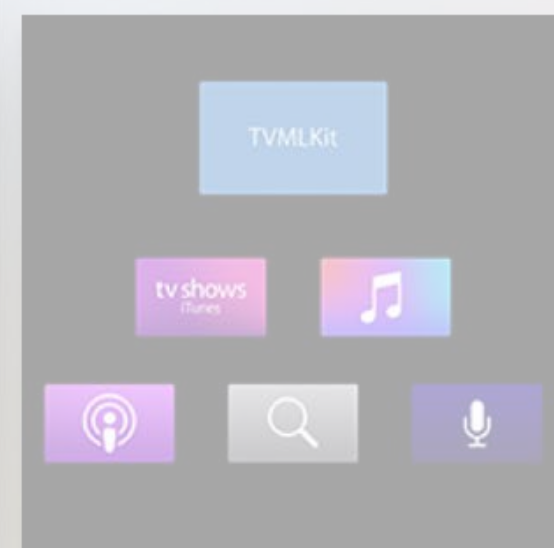
WWDC 2016 • SESSION 229 • APP FRAMEWORKS



Past Sessions



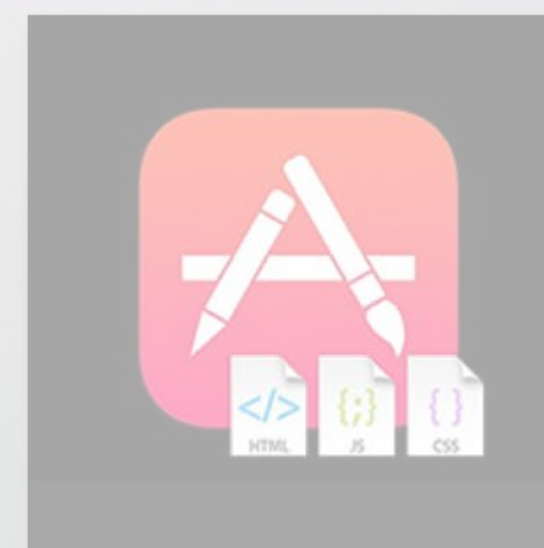
Internationalization Best Practices



Developing tvOS Apps



Developing tvOS Apps



Optimizing Web Content

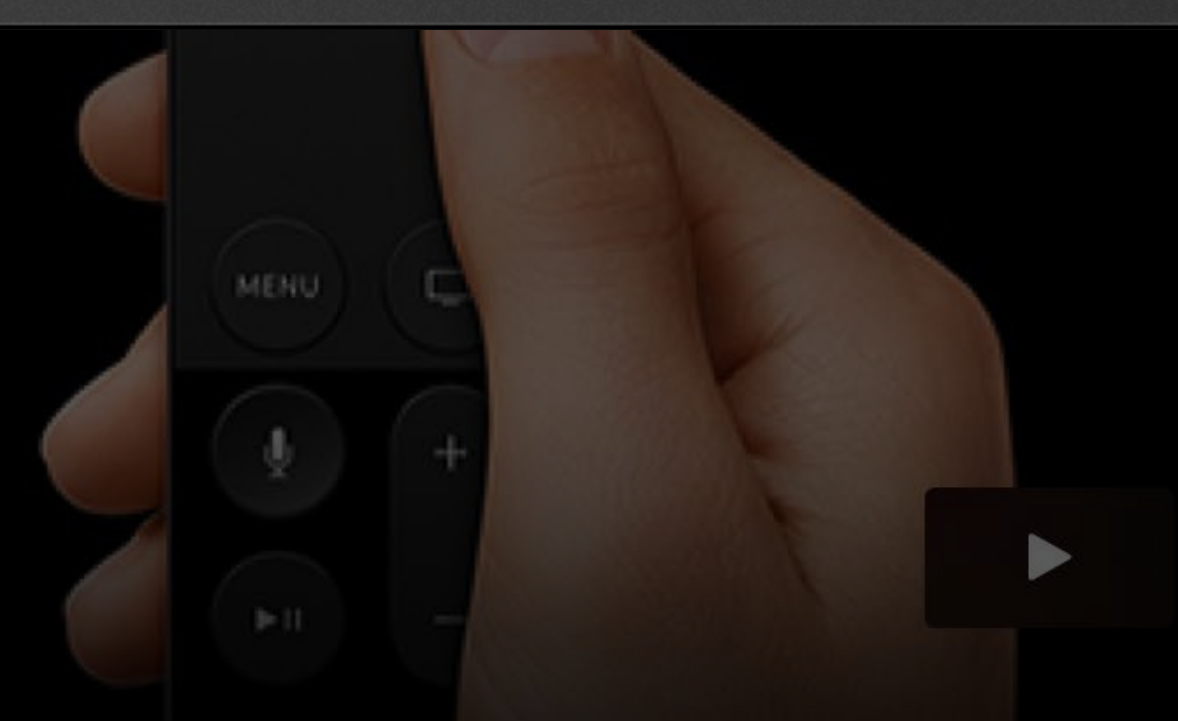


What's New in Internationalization

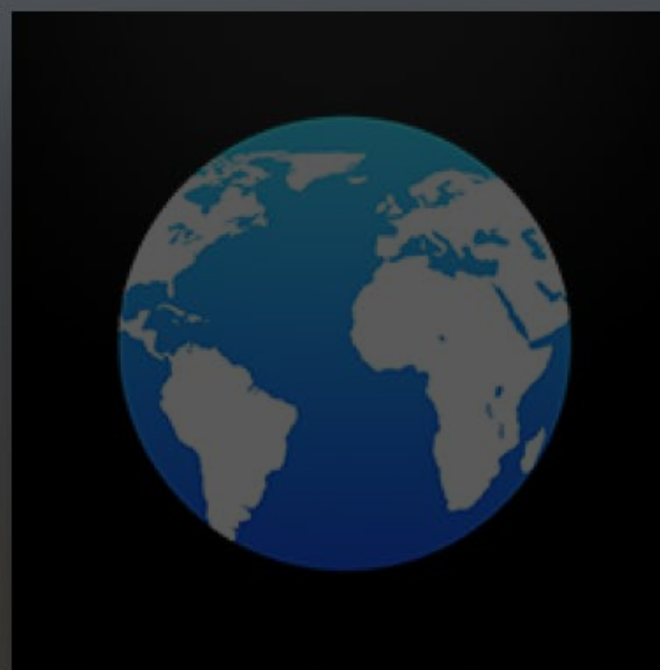


Developing tvOS Apps Using TVMLKit

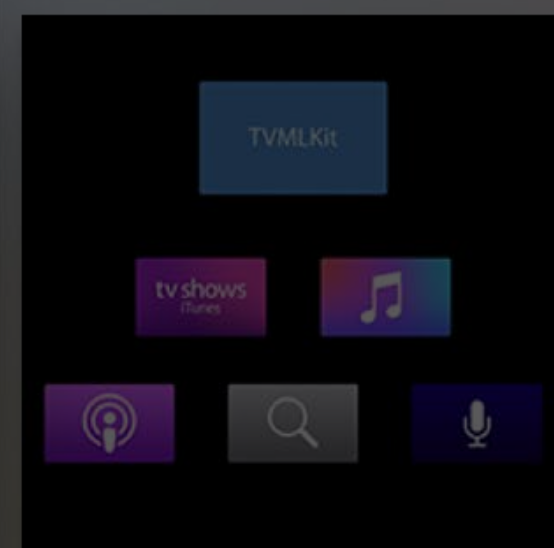
WWDC 2016 • SESSION 229 • APP FRAMEWORKS



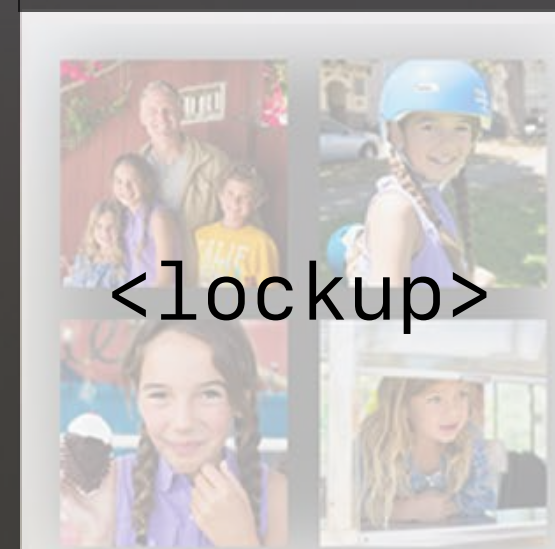
Past Sessions



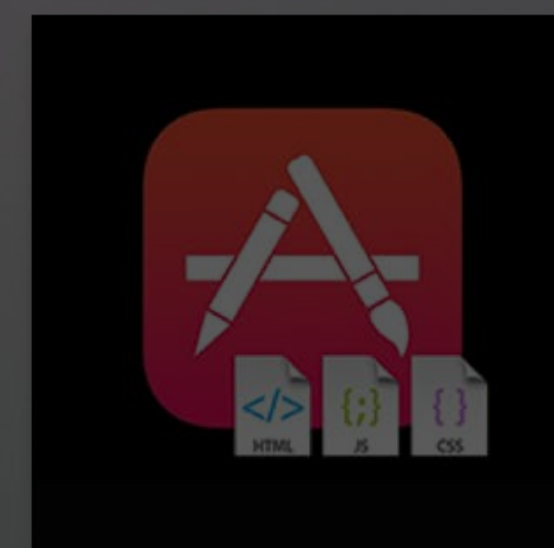
Internationalization Best Practices



Developing tvOS Apps



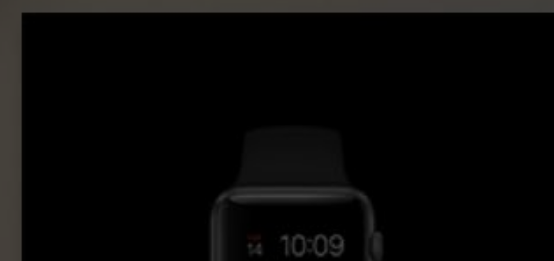
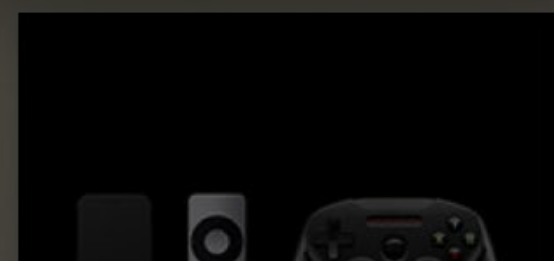
Developing tvOS Apps



Optimizing Web Content

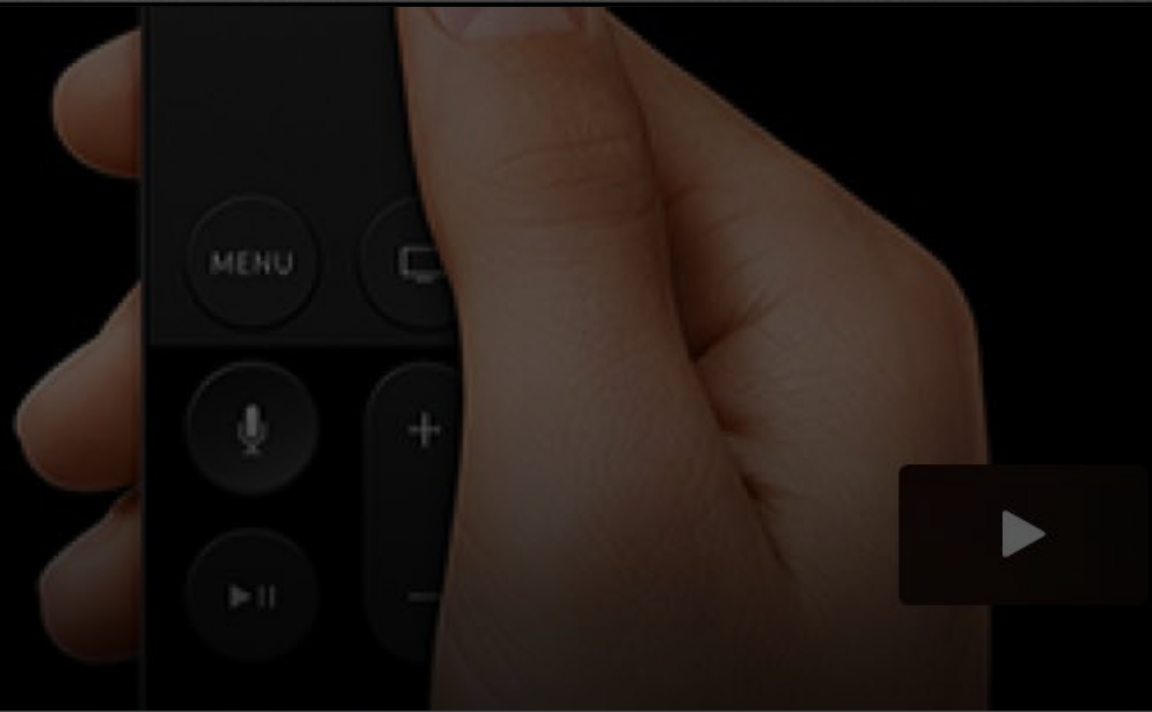


What's New in Internationalization

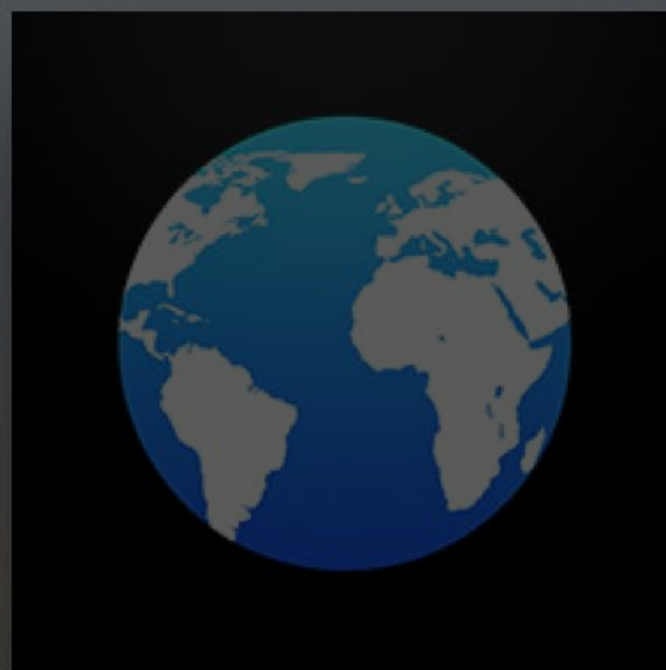


Developing tvOS Apps Using TVMLKit

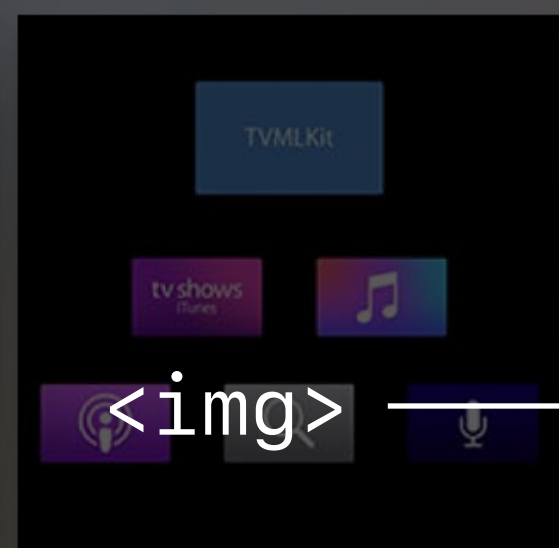
WWDC 2016 • SESSION 229 • APP FRAMEWORKS



Past Sessions



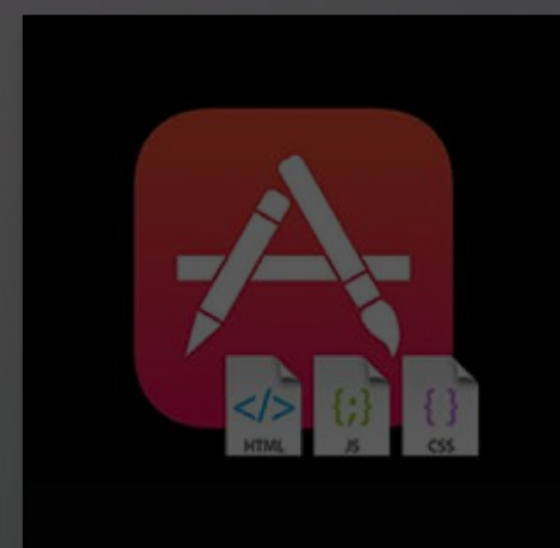
Internationalization Best Pr



 <title>



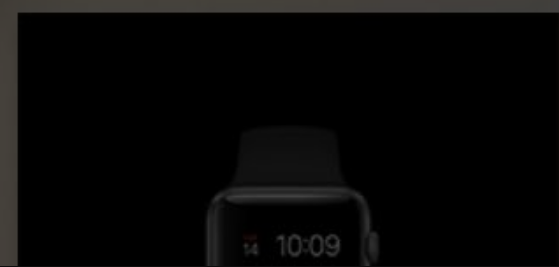
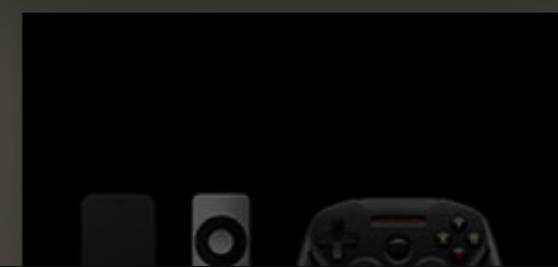
Developing tvOS Apps



Optimizing Web Conte

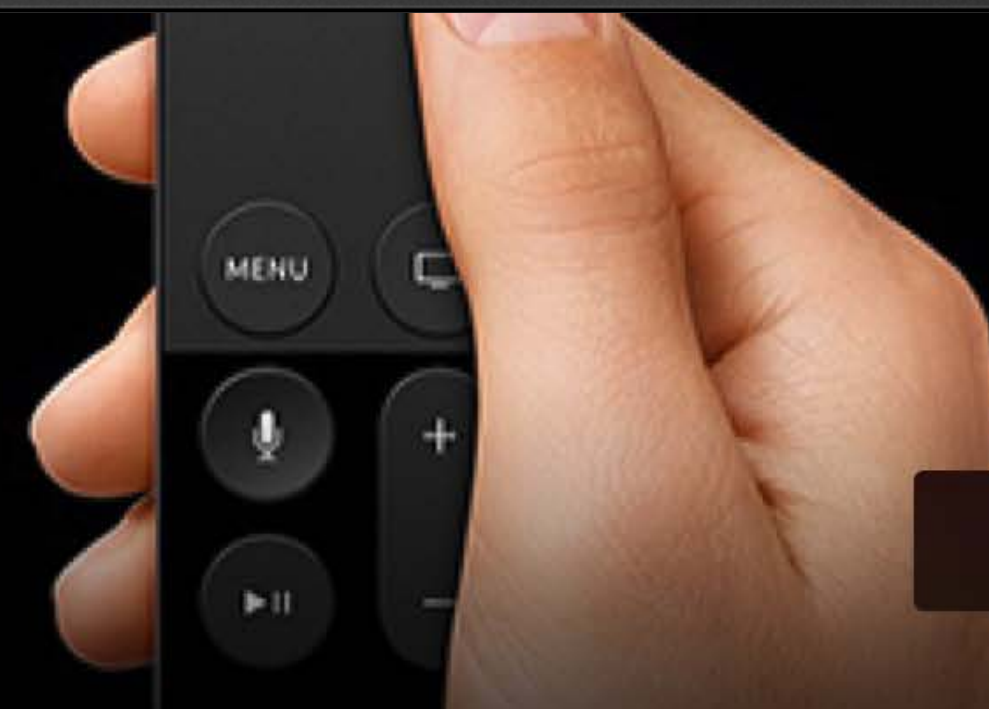


What's New in Interna



Developing tvOS Apps Using TVMLKit

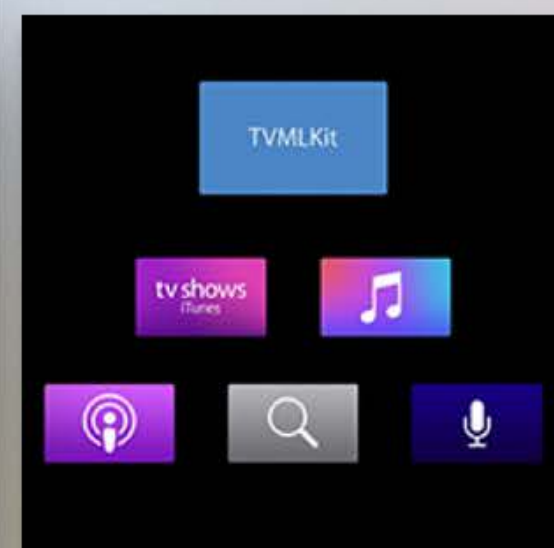
WWDC 2016 • SESSION 229 • APP FRAMEWORKS



Past Sessions



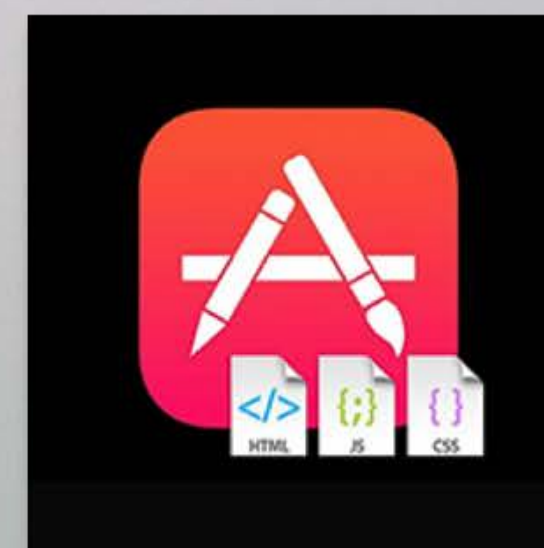
Internationalization Best Practices



Developing tvOS Apps with TVMLKit



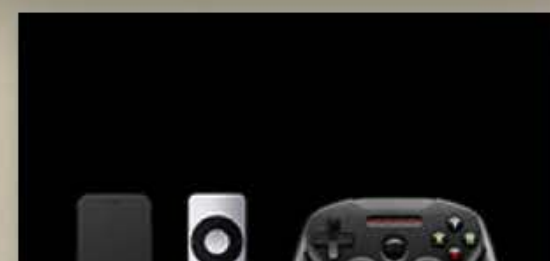
Developing tvOS Apps with SwiftUI



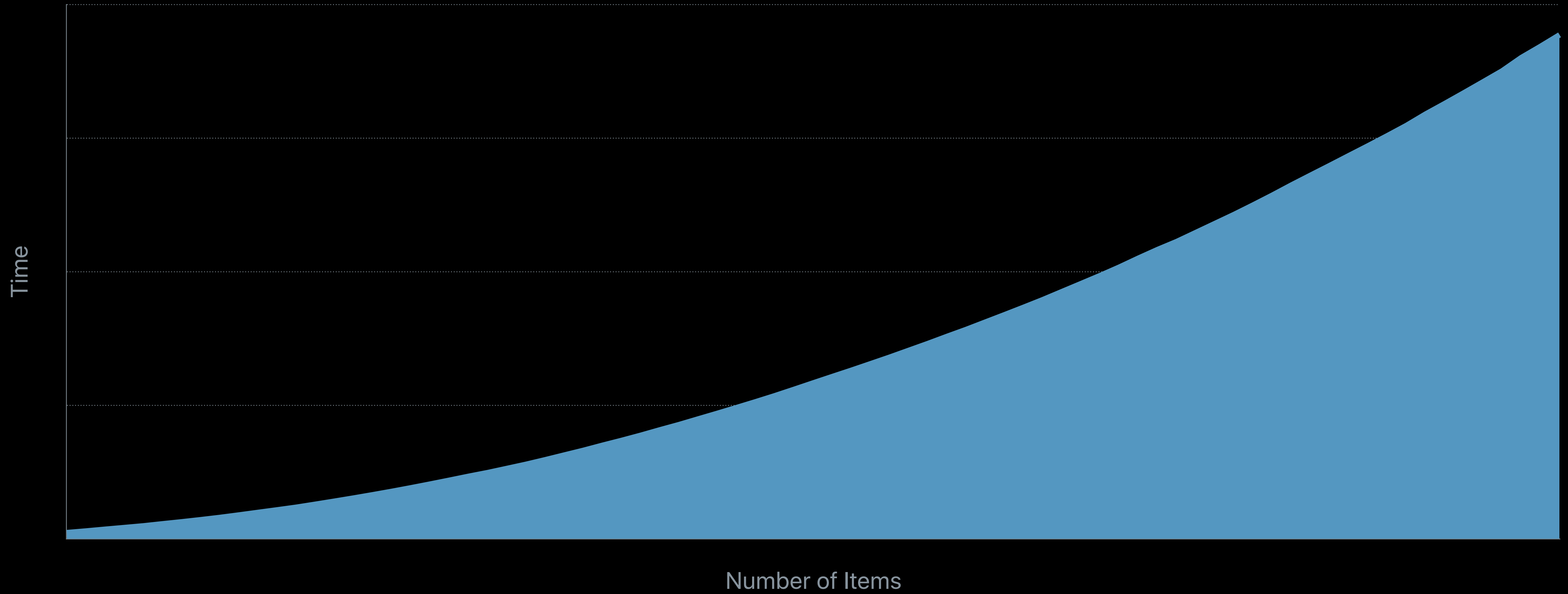
Optimizing Web Content for tvOS



What's New in Internationalization



Scalability Issues



Scalability Issues

Parsing overhead

- Data to DOM
- DOM to views

Scalability Issues

Parsing overhead

- Data to DOM
- DOM to views

Large memory footprint

Template Optimizations



NEW

Prototypes and Data Binding

Pagination

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

Template

```
<grid>
  <section>

  </section>
</grid>
```

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

Template

```
<grid>
  <section>

  </section>
</grid>
```

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

Template

```
<grid>
  <section>
    <lockup>
      
      <title>Title 1</title>
    </lockup>
    <lockup>
      
      <title>Title 2</title>
    </lockup>
    .
    .
    .
    .
  </section>
</grid>
```

Template Optimizations

XML

Template

```
<grid>
  <section>
    <lockup>
      
      <title>Title 1</title>
    </lockup>
    <lockup>
      
      <title>Title 2</title>
    </lockup>
    .
    .
    .
    .
  </section>
</grid>
```

Template Optimizations

XML

Template

```
<grid>
  <section>
    <lockup>
      
      <title>Title 1</title>
    </lockup>
    <lockup>
      
      <title>Title 2</title>
    </lockup>
    .
    .
    .
    .
  </section>
</grid>
```

Template Optimizations

XML

Template

```
<grid>
  <section>
    <lockup>
      <img width="308" height="308" />
      <title></title>
    </lockup>

    <lockup>
      <img width="308" height="308" />
      <title></title>
    </lockup>

    .
    .
    .
    .
  </section>
</grid>
```

Template Optimizations

XML

Template

```
<grid>
  <section>
    <lockup>
      <img width="308" height="308" />
      <title></title>
    </lockup>

    <lockup>
      <img width="308" height="308" />
      <title></title>
    </lockup>

    .
    .
    .
    .

  </section>
</grid>
```


Template Optimizations

XML

Template

```
<grid>
  <section>

    <lockup>
      <img width="308" height="308"/>
      <title></title>
    </lockup>

  </section>
</grid>
```

Template Optimizations

XML

Template

```
<grid>
  <section>
    <prototypes>
      <lockup>
        <img width="308" height="308" />
        <title></title>
      </lockup>
    </prototypes>
  </section>
</grid>
```

Template Optimizations

Prototype

Defines layout

Facilitates partial DOM authoring

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

```
      Template
<grid>
  <section>
    <prototypes>
      <lockup>
        <img width="308" height="308"/>
        <title></title>
      </lockup>
    </prototypes>
  </section>
</grid>
```

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

```
      Template
<grid>
  <section binding="items : {objects};">
    <prototypes>
      <lookup>
        <img binding="@src : {url};" width="308"
          height="308"/>
        <title binding="textContent : {title};"></title>
      </lookup>
    </prototypes>
  </section>
</grid>
```

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

Template

```
<grid>
  <section binding="items : {objects};">
    <prototypes>
      <lookup>
        <img binding="@src : {url};" width="308"
            height="308"/>
        <title binding="textContent : {title};"></title>
      </lookup>
    </prototypes>
  </section>
</grid>
```

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

Template

```
<grid>
  <section binding="items : {objects};">
    <prototypes>
      <lookup>
        <img binding="@src : {url};" width="308"
          height="308"/>
        <title binding="textContent : {title};"></title>
      </lookup>
    </prototypes>
  </section>
</grid>
```

Template Optimizations

JavaScript

```
      Data
{
  "objects" : [
    {
      "url" : "url1",
      "title" : "Title 1"
    },
    {
      "url" : "url2",
      "title" : "Title 2"
    },
    .
    .
    .
    .
  ]
}
```

XML

Template

```
<grid>
  <section binding="items : {objects};">
    <prototypes>
      <lookup>
        <img binding="@src : {url};" width="308"
          height="308"/>
        <title binding="textContent : {title};"></title>
      </lookup>
    </prototypes>
  </section>
</grid>
```


Template Optimizations

Data binding

Associates data to element

Declared through `binding` attribute

- `@<attr>`
- `textContent`
- `items`

Template Optimizations

Data binding

Provide data using dataItem property

```
// Parse JSON text.  
let data = parseFromJSON(json);  
  
// Attach data to section element.  
let section = document.createElement("section");  
section.dataItem = data;
```

Template Optimizations

Data binding

Provide data using dataItem property

```
// Parse JSON text.  
let data = parseFromJSON(json);  
  
// Attach data to section element.  
let section = document.createElement("section");  
section.dataItem = data;
```

Template Optimizations

Data binding

Provide data using dataItem property

```
// Parse JSON text.  
let data = parseFromJSON(json);  
  
// Attach data to section element.  
let section = document.createElement("section");  
section.dataItem = data;
```

Template Optimizations

Data binding

Provide data using dataItem property

```
// Parse JSON text.  
let data = parseFromJSON(json);  
  
// Attach data to section element.  
let section = document.createElement("section");  
section.dataItem = data;
```

Template Optimizations

Data binding

Provide data using dataItem property

```
// Parse JSON text.  
let data = parseFromJSON(json);  
  
// Attach data to section element.  
let section = document.createElement("section");  
section.dataItem = data;
```

Template Optimizations

Pagination

Append using `needsmore`

- `shelf`, `grid`, `list`, `stackTemplate`
- Invoked towards the end

Template Optimizations

Pagination

Append using `needsmore`

- `shelf`, `grid`, `list`, `stackTemplate`
- Invoked towards the end

Requires observable objects


```
// Map item objects into DataItem objects
let dataItems = objects.map((object) => {
    let dataItem = new DataItem("MyObject", object.id);
    dataItem.url = object.url;
    dataItem.title = object.title;
    return dataItem;
});
```

```
// Create a data item for the grid.
let sectionDataItem = new DataItem();
sectionDataItem.objects = dataItems;
```

```
// Provide data through DOM
let sectionElement = document.createElement("section");
sectionElement.dataItem = sectionDataItem;
```

```
// Map item objects into DataItem objects
let dataItems = objects.map((object) => {
    let dataItem = new DataItem("MyObject", object.id);
    dataItem.url = object.url;
    dataItem.title = object.title;
    return dataItem;
});
```

```
// Create a data item for the grid.
let sectionDataItem = new DataItem();
sectionDataItem.objects = dataItems;
```

```
// Provide data through DOM
let sectionElement = document.createElement("section");
sectionElement.dataItem = sectionDataItem;
```

```
// Map item objects into DataItem objects
let dataItems = objects.map((object) => {
  let dataItem = new DataItem("MyObject", object.id);
  dataItem.url = object.url;
  dataItem.title = object.title;
  return dataItem;
});
```

```
// Create a data item for the grid.
let sectionDataItem = new DataItem();
sectionDataItem.objects = dataItems;
```

```
// Provide data through DOM
let sectionElement = document.createElement("section");
sectionElement.dataItem = sectionDataItem;
```

```
// Map item objects into DataItem objects
let dataItems = objects.map((object) => {
    let dataItem = new DataItem("MyObject", object.id);
    dataItem.url = object.url;
    dataItem.title = object.title;
    return dataItem;
});
```

```
// Create a data item for the grid.
let sectionDataItem = new DataItem();
sectionDataItem.objects = dataItems;
```

```
// Provide data through DOM
let sectionElement = document.createElement("section");
sectionElement.dataItem = sectionDataItem;
```

```
// Map item objects into DataItem objects
let dataItems = objects.map((object) => {
  let dataItem = new DataItem("MyObject", object.id);
  dataItem.url = object.url;
  dataItem.title = object.title;
  return dataItem;
});
```

```
// Create a data item for the grid.
let sectionDataItem = new DataItem();
sectionDataItem.objects = dataItems;
```

```
// Provide data through DOM
let sectionElement = document.createElement("section");
sectionElement.dataItem = sectionDataItem;
```

```
// Map item objects into DataItem objects
let dataItems = objects.map((object) => {
    let dataItem = new DataItem("MyObject", object.id);
    dataItem.url = object.url;
    dataItem.title = object.title;
    return dataItem;
});
```

```
// Create a data item for the grid.
let sectionDataItem = new DataItem();
sectionDataItem.objects = dataItems;
```

```
// Provide data through DOM
let sectionElement = document.createElement("section");
sectionElement.dataItem = sectionDataItem;
```

```
// Map item objects into DataItem objects
let dataItems = objects.map((object) => {
  let dataItem = new DataItem("MyObject", object.id);
  dataItem.url = object.url;
  dataItem.title = object.title;
  return dataItem;
});
```

```
// Create a data item for the grid.
let sectionDataItem = new DataItem();
sectionDataItem.objects = dataItems;
```

```
// Provide data through DOM
let sectionElement = document.createElement("section");
sectionElement.dataItem = sectionDataItem;
```

```
// needsmore event handler.
stackTemplate.addEventListener("needsmore", (event) => {

    // Fetch next batch of items.
    fetchNextBatch((objects) => {

        ...

        // Append to existing data items.
        Array.prototype.push.apply(sectionElement.dataItem.objects, dataItems);
        sectionElement.dataItem.touchPropertyPath("objects");
    });
});
}
```



```
// needsmore event handler.  
stackTemplate.addEventListener("needsmore", (event) => {  
  
    // Fetch next batch of items.  
    fetchNextBatch((objects) => {  
  
        ...  
  
        // Append to existing data items.  
        Array.prototype.push.apply(sectionElement.dataItem.objects, dataItems);  
        sectionElement.dataItem.touchPropertyPath("objects");  
    });  
  
}
```

```
// needsmore event handler.
stackTemplate.addEventListener('needsmore', (event) => {

  // Fetch next batch of items.
  fetchNextBatch((objects) => {

    ...

    // Append to existing data items.
    Array.prototype.push.apply(sectionElement.dataItem.objects, dataItems);
    sectionElement.dataItem.touchPropertyPath("objects");
  });
});
}
```

```
// needsmore event handler.
stackTemplate.addEventListener('needsmore', (event) => {

  // Fetch next batch of items.
  fetchNextBatch((objects) => {

    ...

    // Append to existing data items.
    Array.prototype.push.apply(sectionElement.dataItem.objects, dataItems);
    sectionElement.dataItem.touchPropertyPath("objects");
  });
});
}
```

Demo

Recap

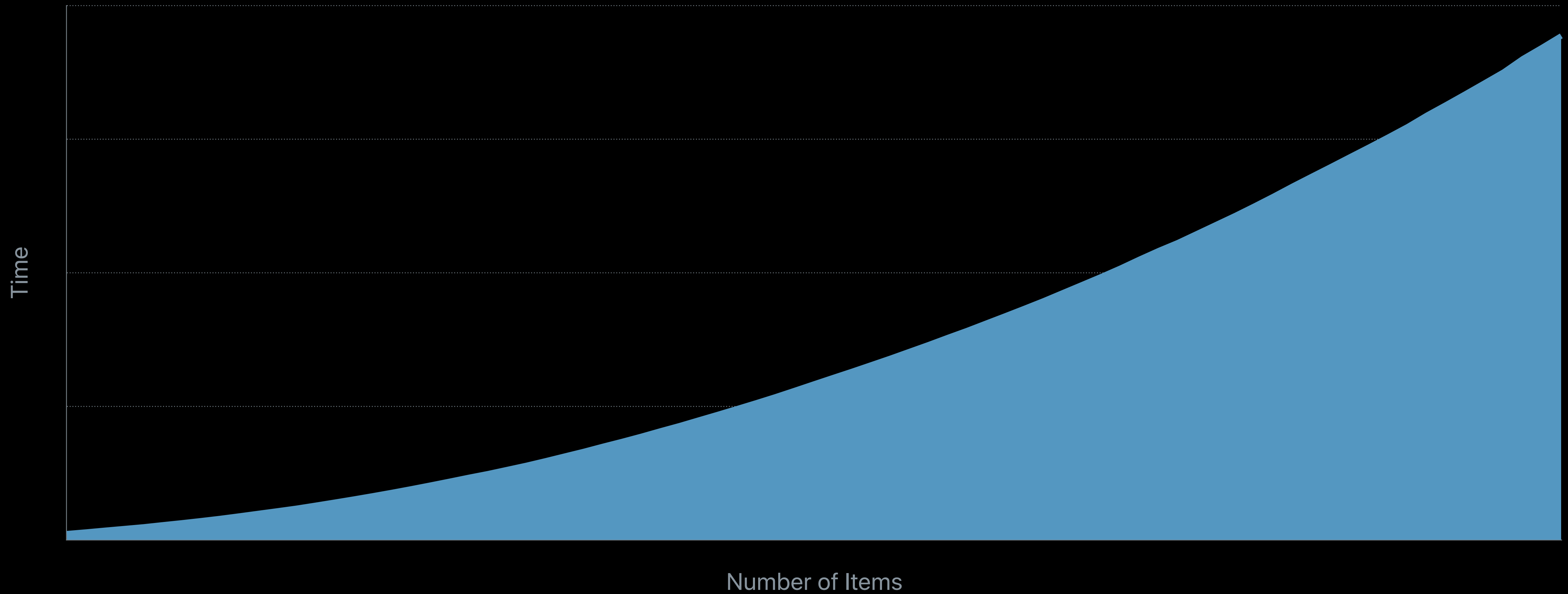
Created template with prototypes

Specified data binding

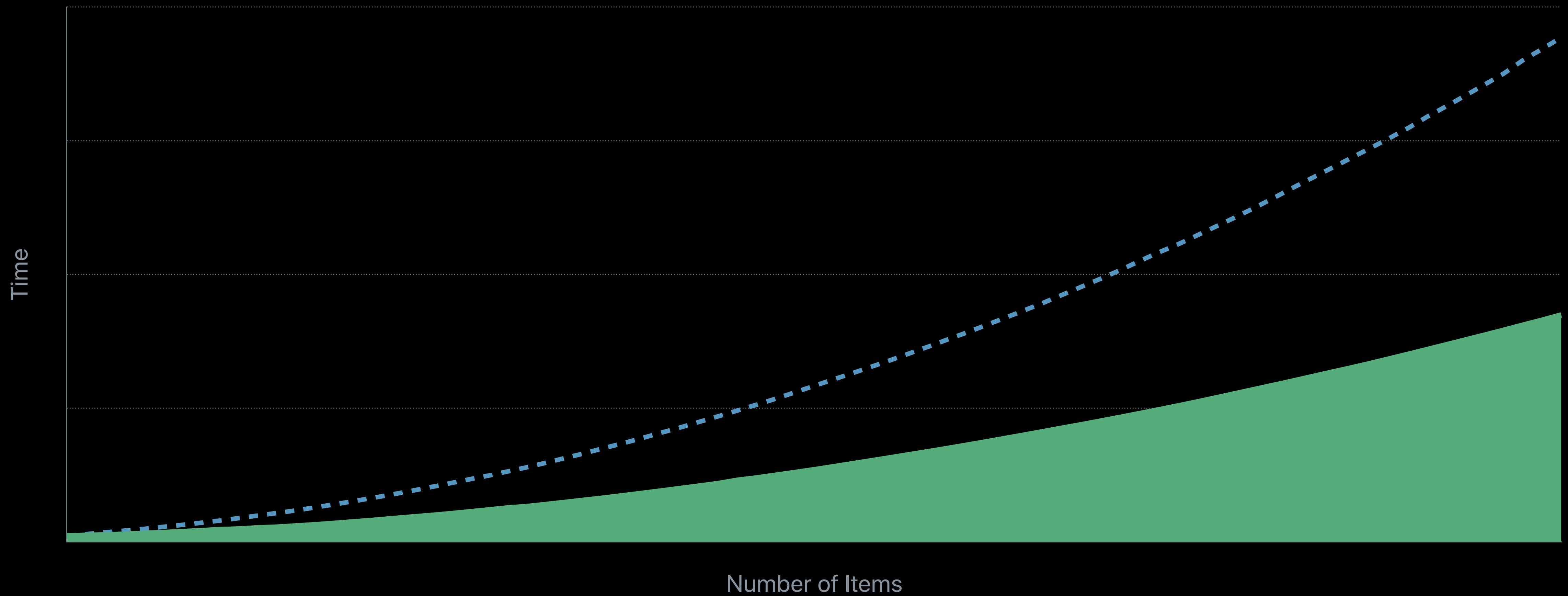
Associated data to DOM

Implemented pagination

Template Optimizations



Template Optimizations

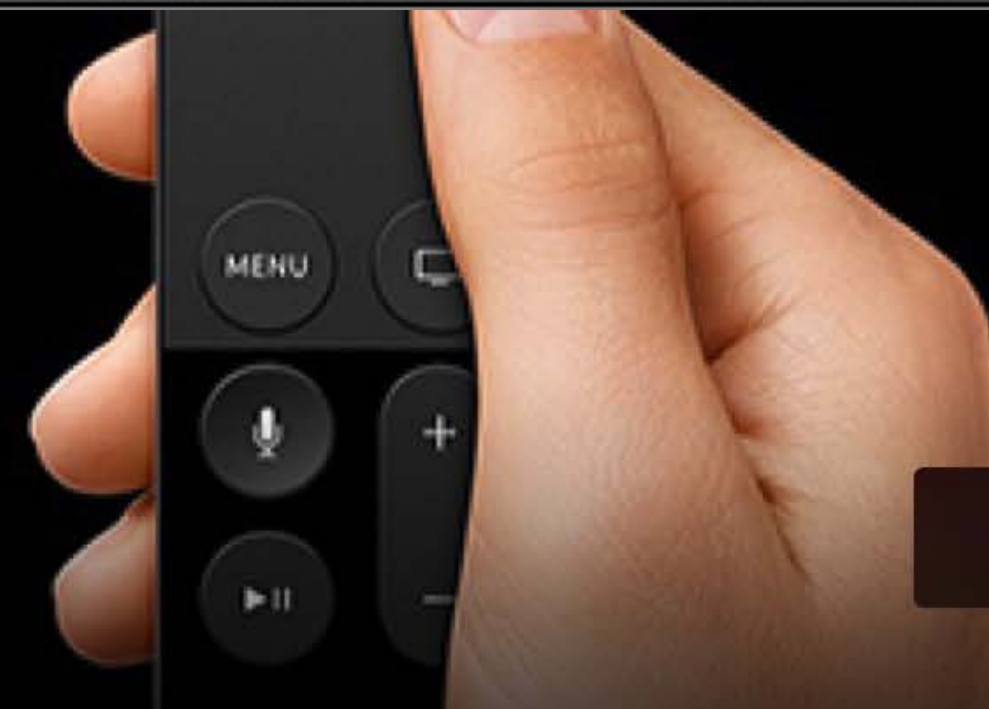


Development Using Web Inspector

Jeremy Foo, tvOS Engineer

Developing tvOS Apps Using TVMLKit

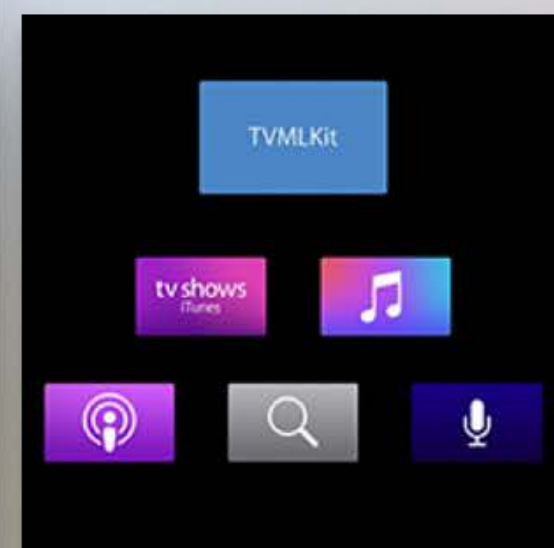
WWDC 2016 • SESSION 229 • APP FRAMEWORKS



Past Sessions



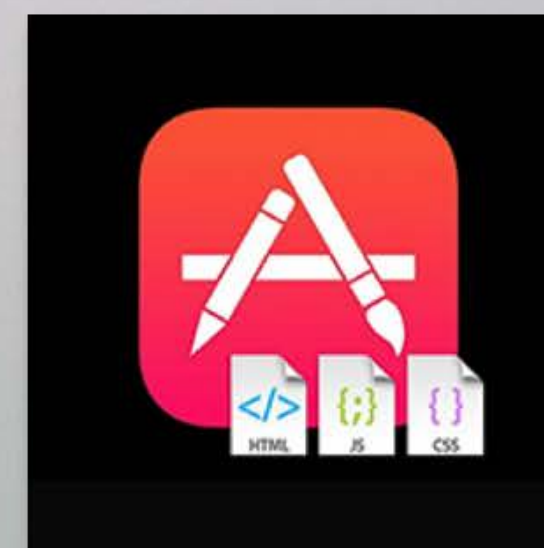
Internationalization Best Practices



Developing tvOS Apps Using TVMLKit



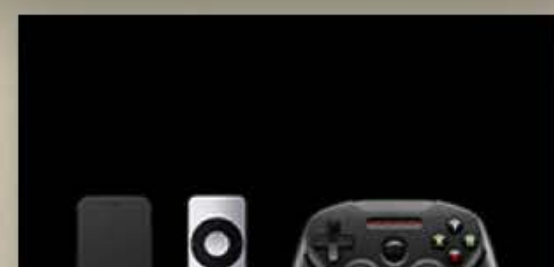
Developing tvOS Apps Using AVKit



Optimizing Web Content for tvOS

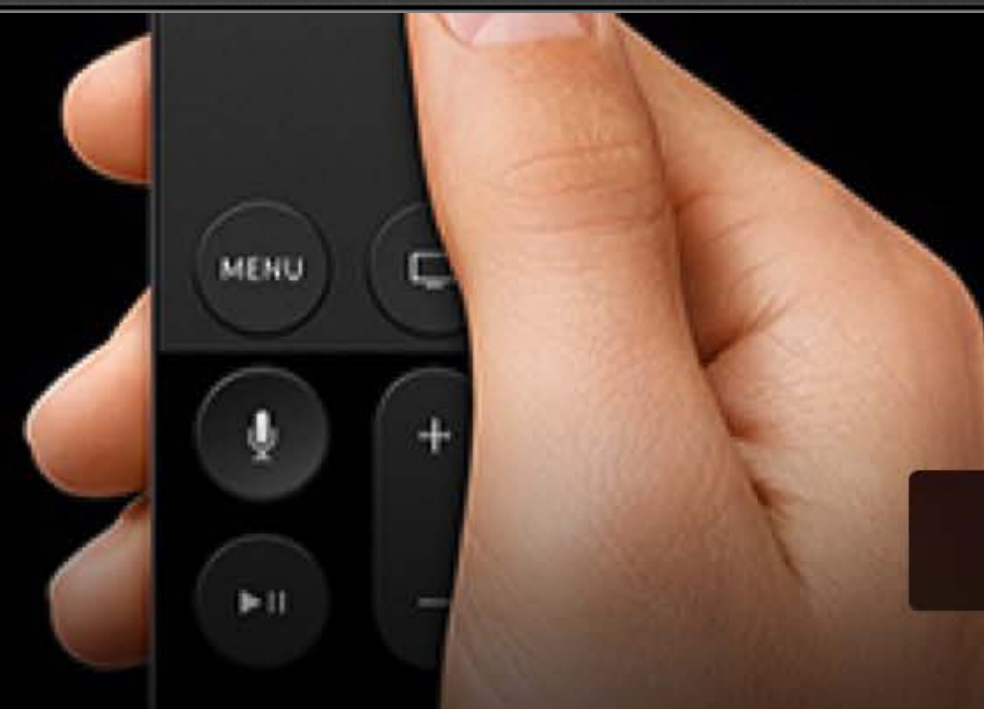


What's New in Internationalization



Developing tvOS Apps Using TVMLKit

WWDC 2016 • SESSION 229 • APP FRAMEWORKS

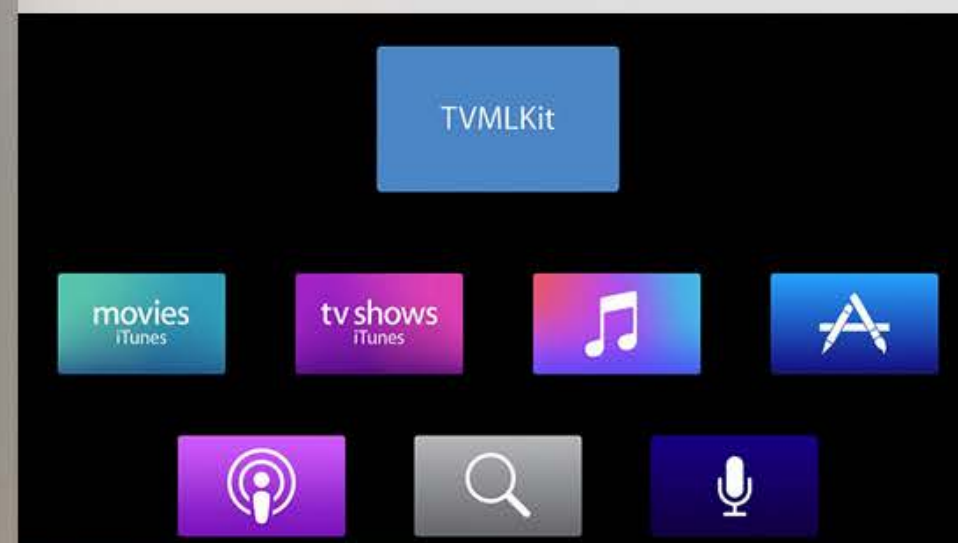


Past Sessions

2016 SESSION 201 • IOS MACOS TVOS...
Internationalization Best Pr...
Mission



2016 SESSION 212 • TVOS
Developing tvOS Apps Usi...
Mission



2016 SESSION 229 • TVOS
Developing tvOS Apps Usi...
Nob Hill



Development Cycle

Speculative fixes

Development Cycle

Speculative fixes

Build-and-run cycle

Development Cycle

Speculative fixes

Build-and-run cycle

Loss of context

Web Inspector

Visual debugging

Network analysis

Local and Session storage

JavaScript debugging

Console logging

Performance analysis



Web Inspector

Visual debugging

Network analysis

Local and Session storage

JavaScript debugging

Console logging

Performance analysis



Web Inspector

NEW

Visual debugging

Network analysis

Local and Session storage

JavaScript debugging

Console logging

Performance analysis



Visual Debugging

NEW

The screenshot shows the Web Inspector interface for a simulator. The title bar reads "Web Inspector — Simulator — WWDC — com.apple.tvml.WWDCDemo". The interface includes a search bar, a refresh button, and three status indicators (3 messages, 0 warnings, 0 errors). The main area is divided into three tabs: "Elements", "Network", and "Storage". The "Elements" tab is active, showing a DOM tree with the following structure:

- <document>
 - <head>...</head>
 - <stackTemplate>...</stackTemplate> = \$0

The selected node, <stackTemplate>, is highlighted in blue. The right-hand pane shows the "Node Inspector" for this element, with the following details:

- Identity**
 - Type: Element
 - Name: stackTemplate
- Attributes**
 - No Attributes
- Properties**
- Event Listeners**
- needsmore**
 - Node: [stackTemplate](#)
 - Function: (anonymous function)
 - Bubbling: Yes

Visual Debugging

Visualize elements



```
Web Inspector — Simulator — WWDC — com.apple.tvml.WWDCDemo

Elements Network

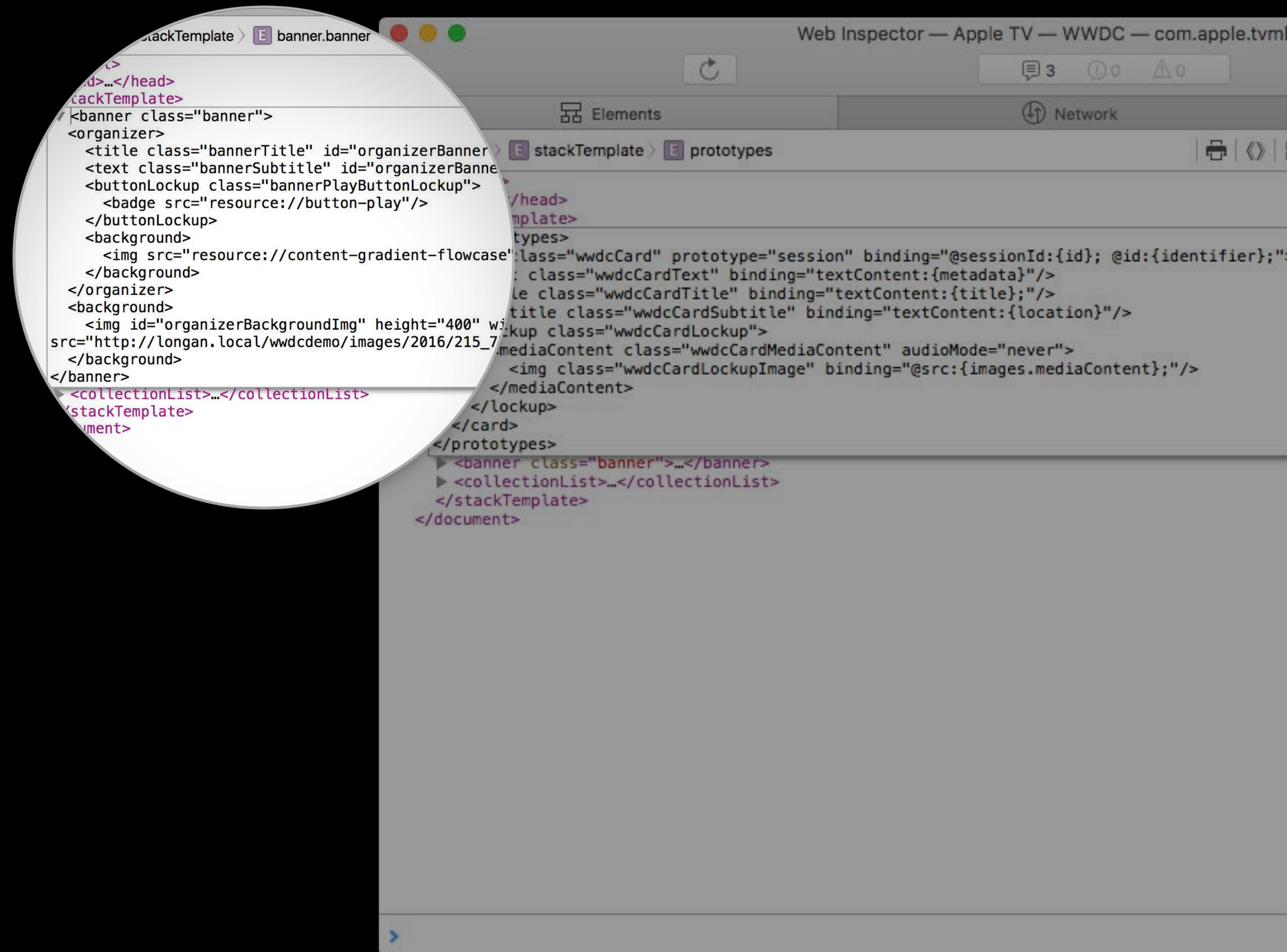
= $0
/head>
nplate>
r class="banner">...</banner>
tionList>
otypes>...</prototypes>
lf>
leader>...</header>
section binding="items: {sessions};">
<card itemID="0D10EBC1-C228-482F-BE8B-A56B2D9AF166" class="wwdcCard">
  <text class="wwdcCardText" binding="textContent:{metadata}">2016 Session 210 • tvOS
  <title class="wwdcCardTitle" binding="textContent:{title};">Mastering UIKit on tvOS
  <subtitle class="wwdcCardSubtitle" binding="textContent:{location}">Presidio</subtitle>
  <lockup class="wwdcCardLockup">...</lockup>
</card>
<card itemID="CE2F91FF-5235-4CC1-9B2C-C6261683D2D1" class="wwdcCard">...</card>
<card itemID="75A53B2A-0DD7-40A0-B0D9-58901A543EAD" class="wwdcCard">...</card>
</section>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</shelf>
</collectionList>
</stackTemplate>
</document>
```

Visual Debugging

Real-time DOM editing

NEW

Edit XML



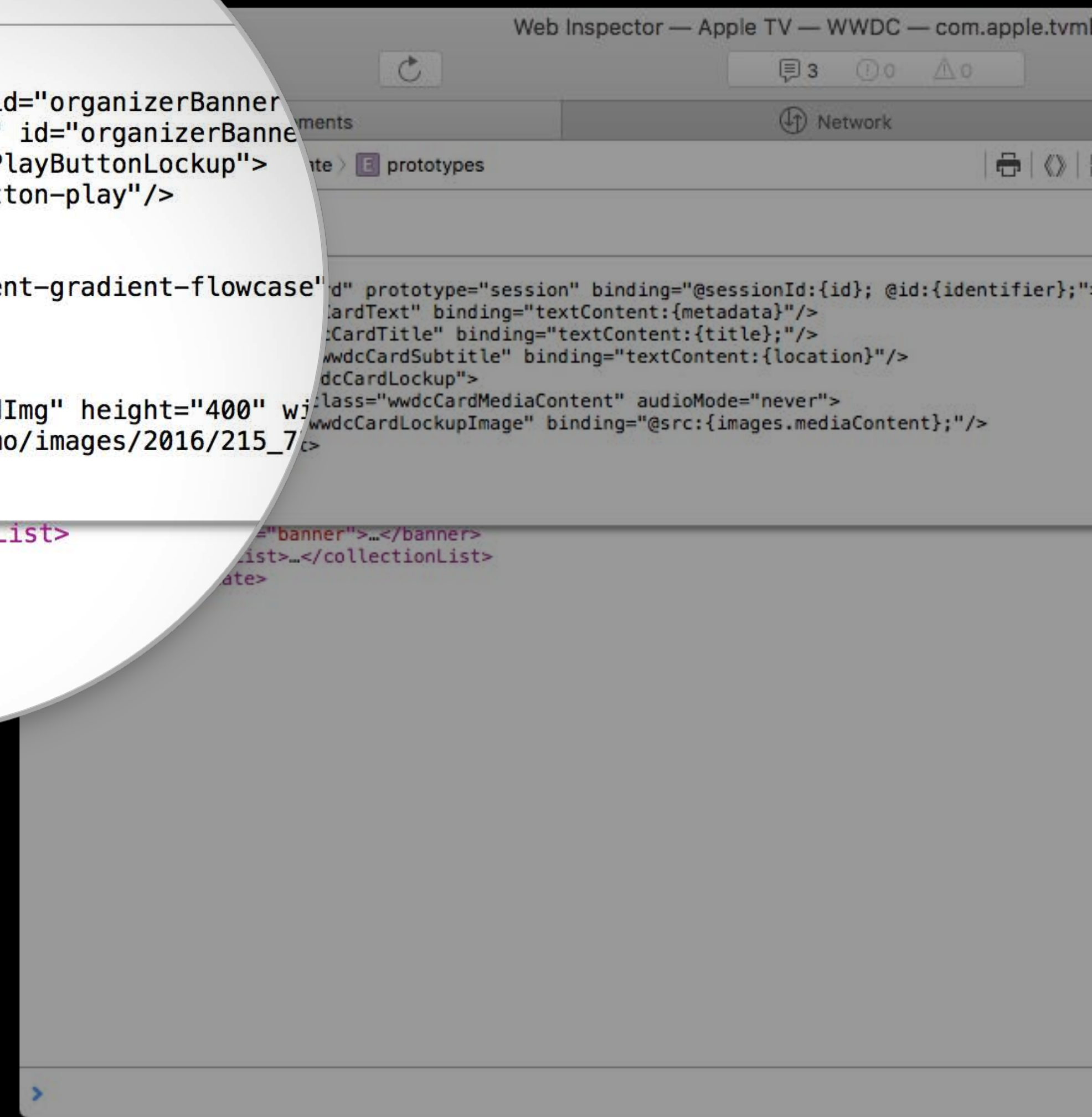
Visual Debugging

Real-time DOM editing

Edit XML

NEW

```
stackTemplate > E banner.banner  
</head>  
</stackTemplate>  
<banner class="banner">  
  <organizer>  
    <title class="bannerTitle" id="organizerBanner">  
    <text class="bannerSubtitle" id="organizerBanner">  
    <buttonLockup class="bannerPlayButtonLockup">  
      <badge src="resource://button-play"/>  
    </buttonLockup>  
    <background>  
        
    </background>  
  </organizer>  
  <background>  
      
  </background>  
</banner>  
</collectionList>  
</stackTemplate>  
</document>
```



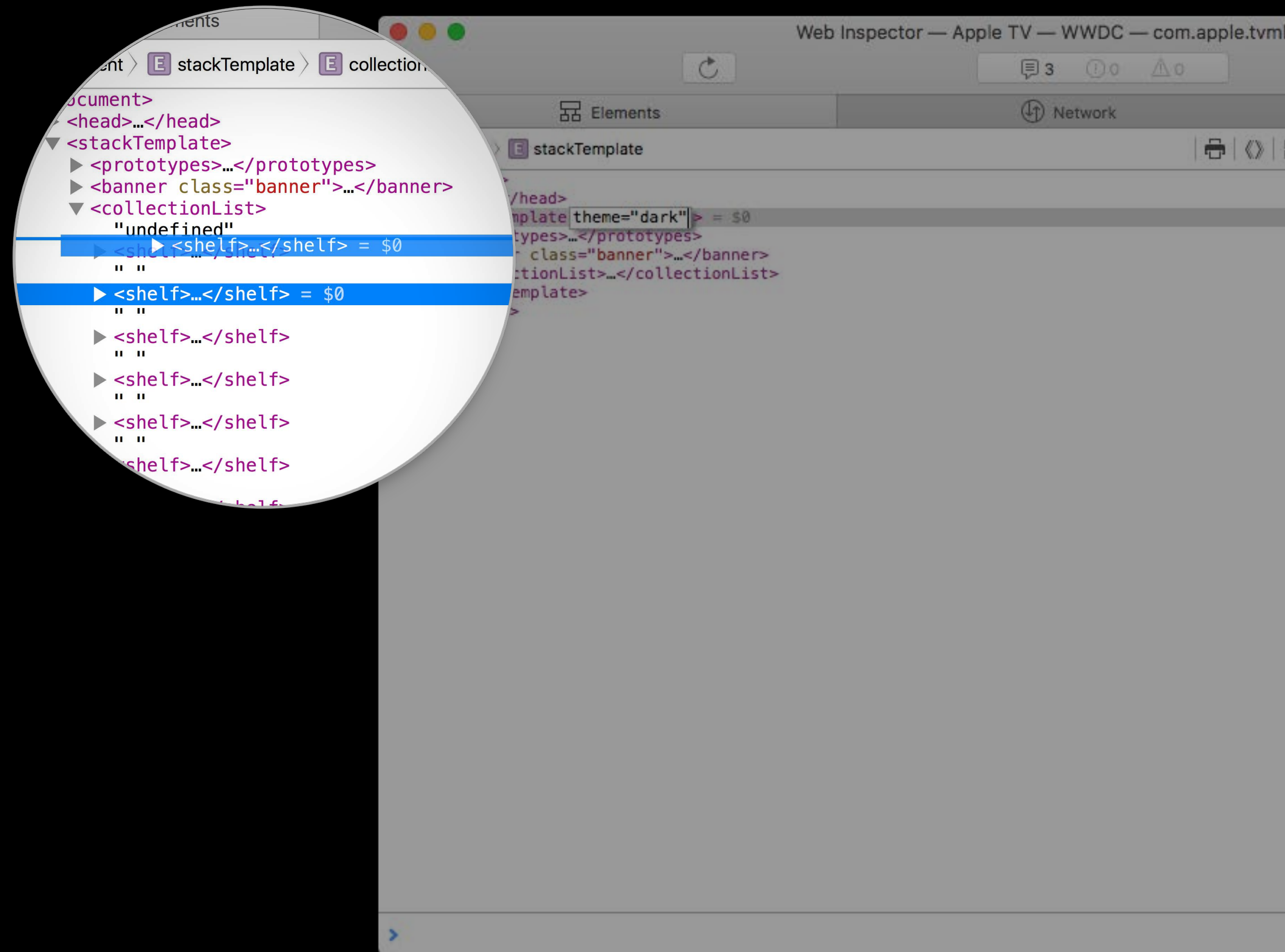
Visual Debugging

Real-time DOM editing

NEW

Edit XML

Reorder/delete nodes



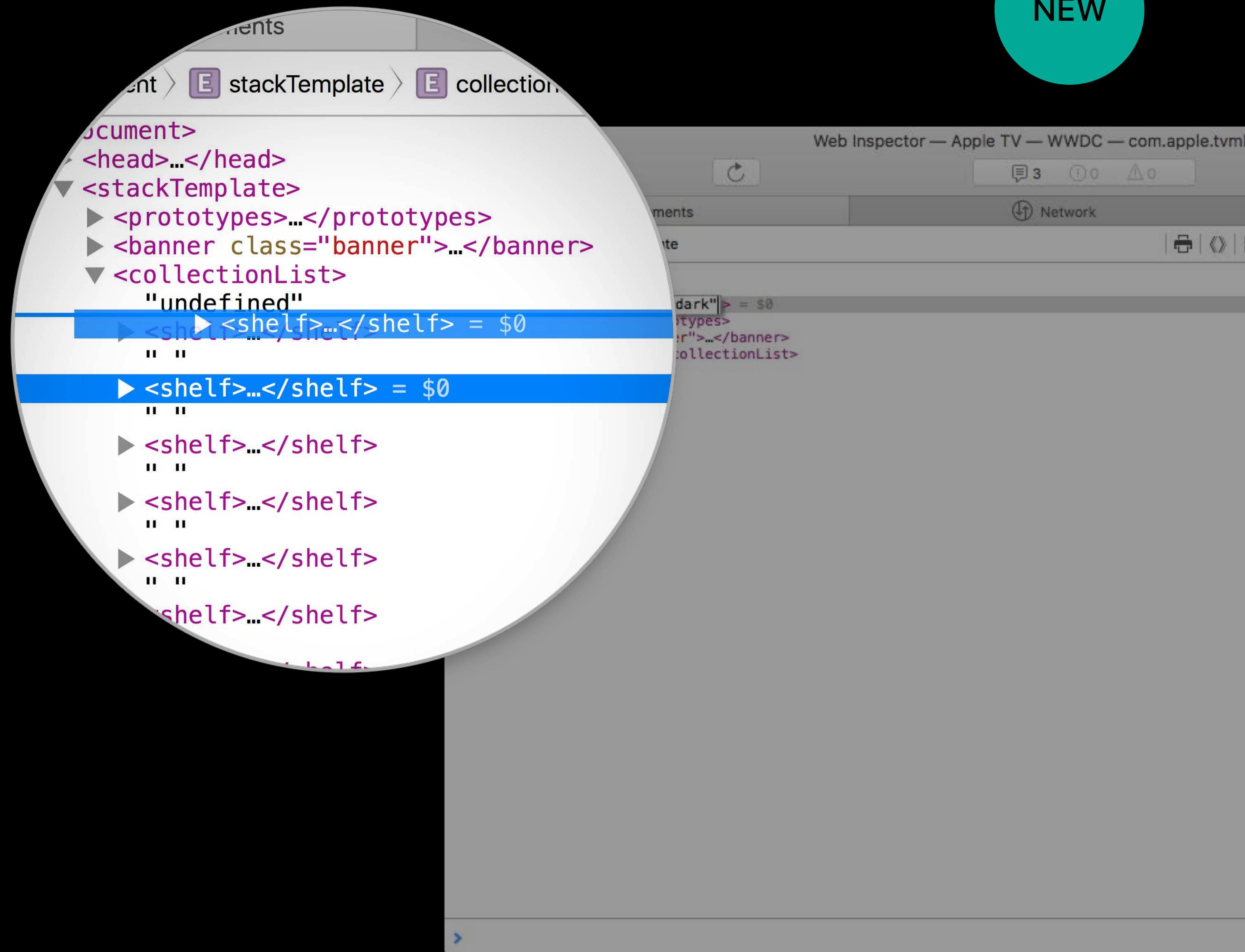
Visual Debugging

Real-time DOM editing

Edit XML

Reorder/delete nodes

NEW



Visual Debugging

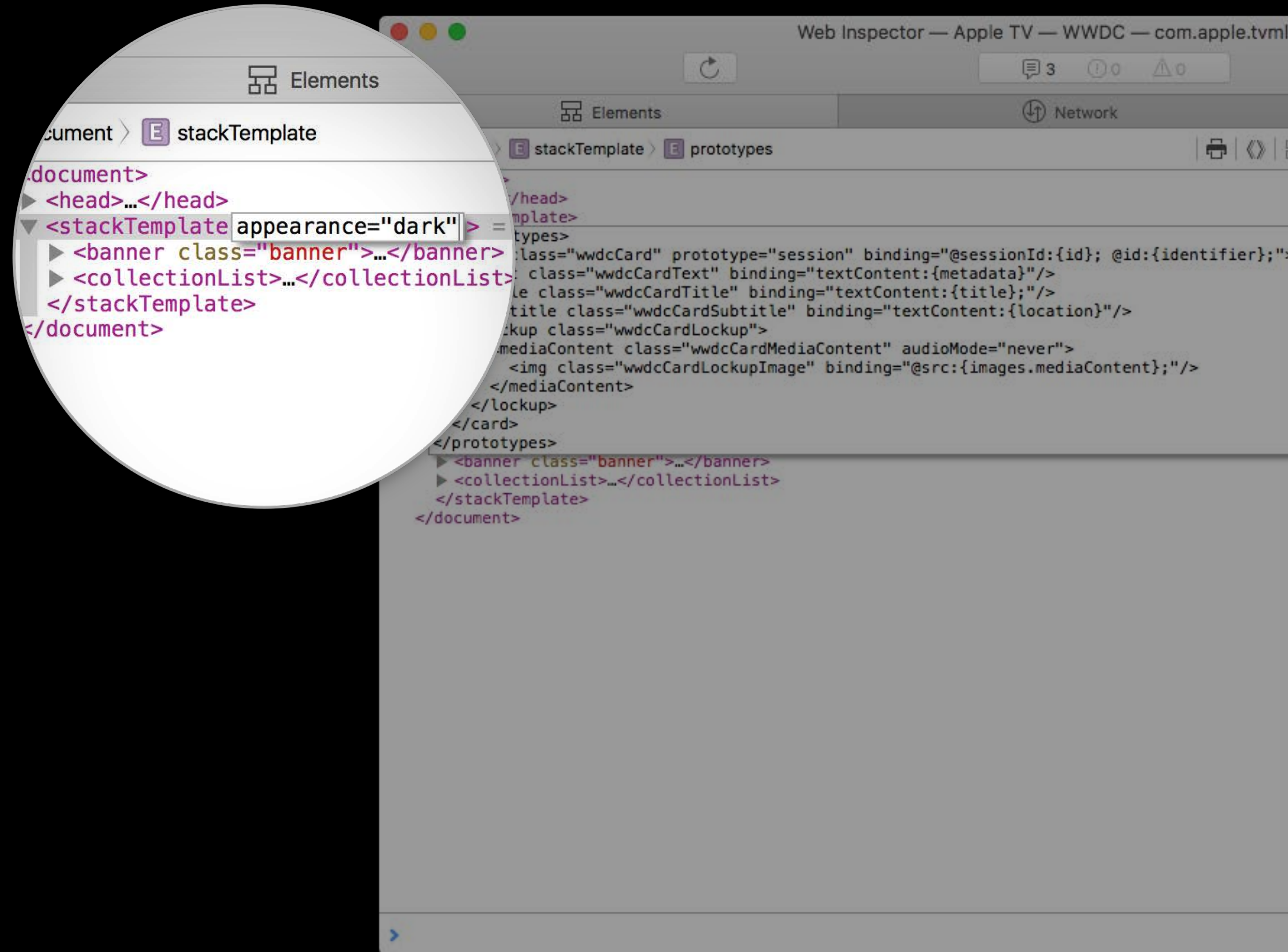
Real-time DOM editing

NEW

Edit XML

Reorder/delete nodes

Modify attributes



Visual Debugging

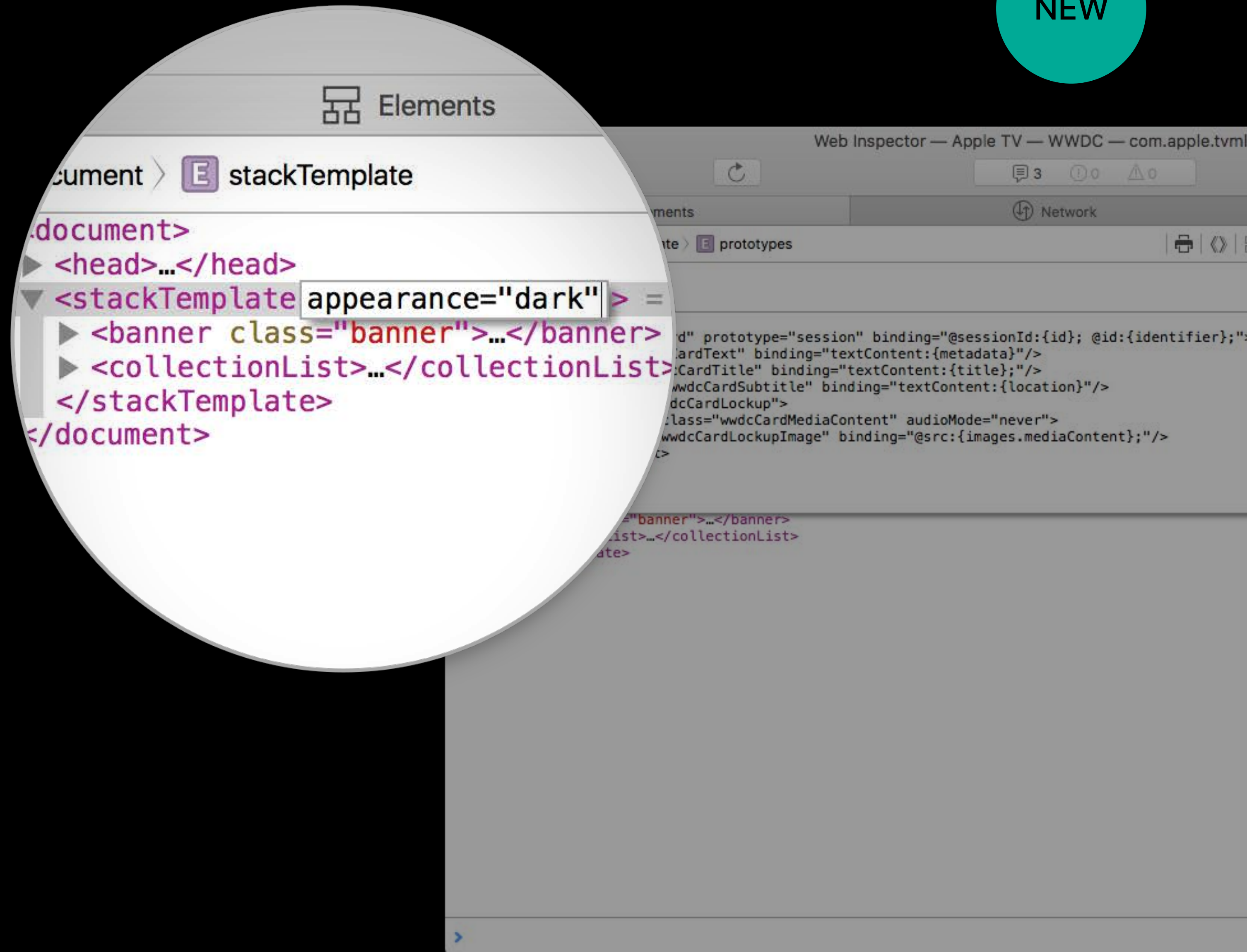
Real-time DOM editing

NEW

Edit XML

Reorder/delete nodes

Modify attributes



Visual Debugging

Real-time DOM editing

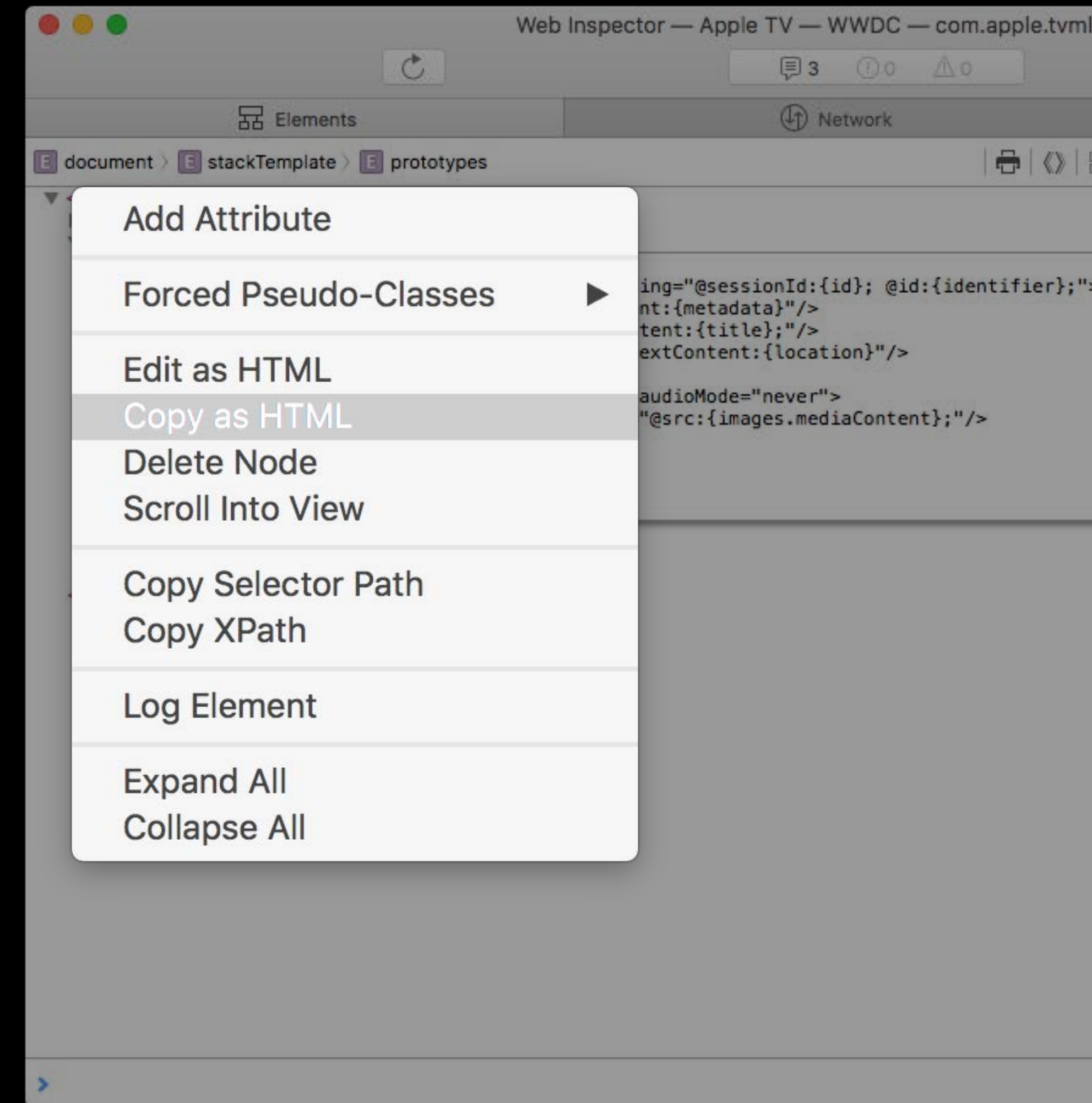
NEW

Edit XML

Reorder/delete nodes

Modify attributes

Copy XML

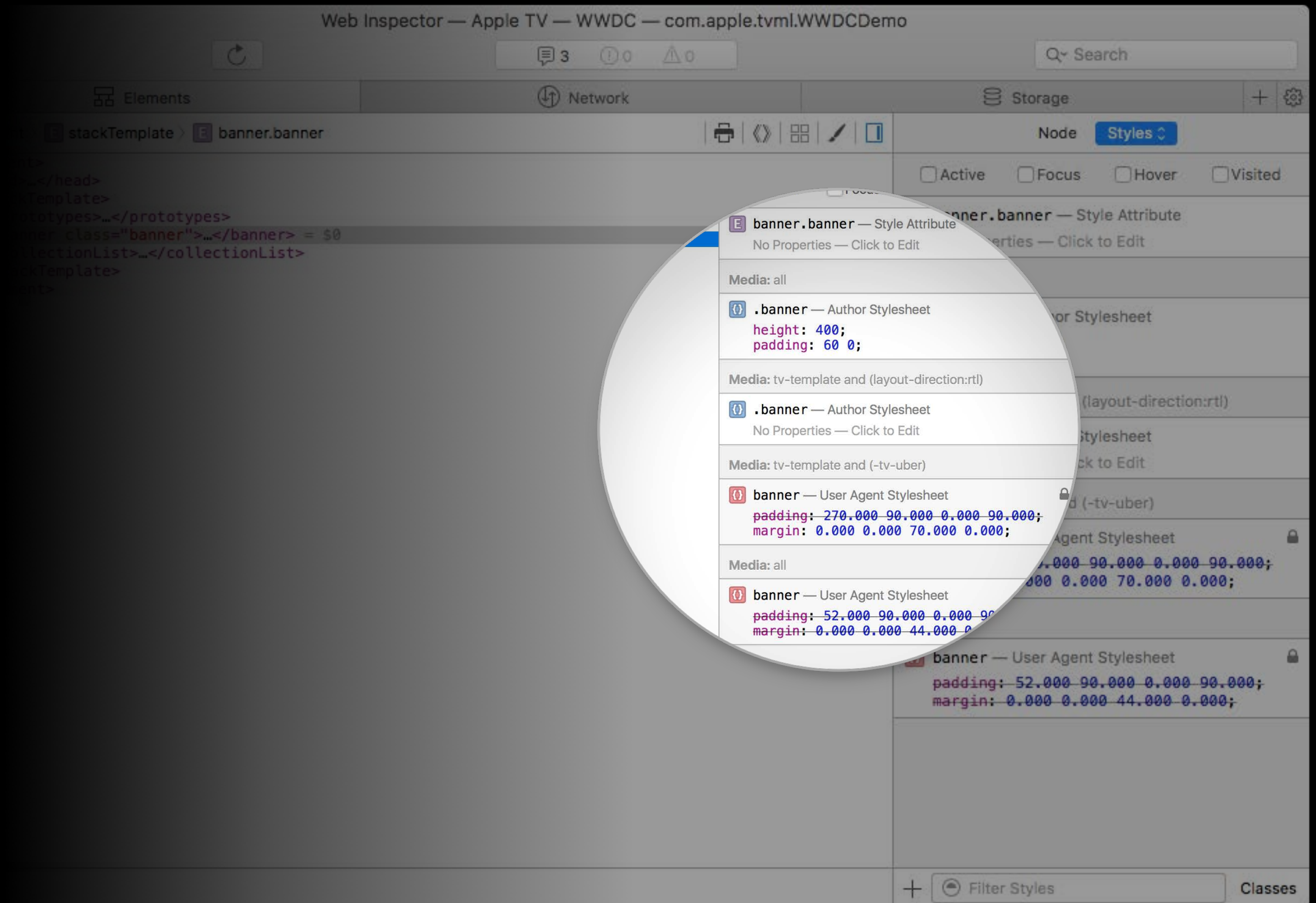


Visual Debugging

Inspect and modify styles

NEW

Per node basis

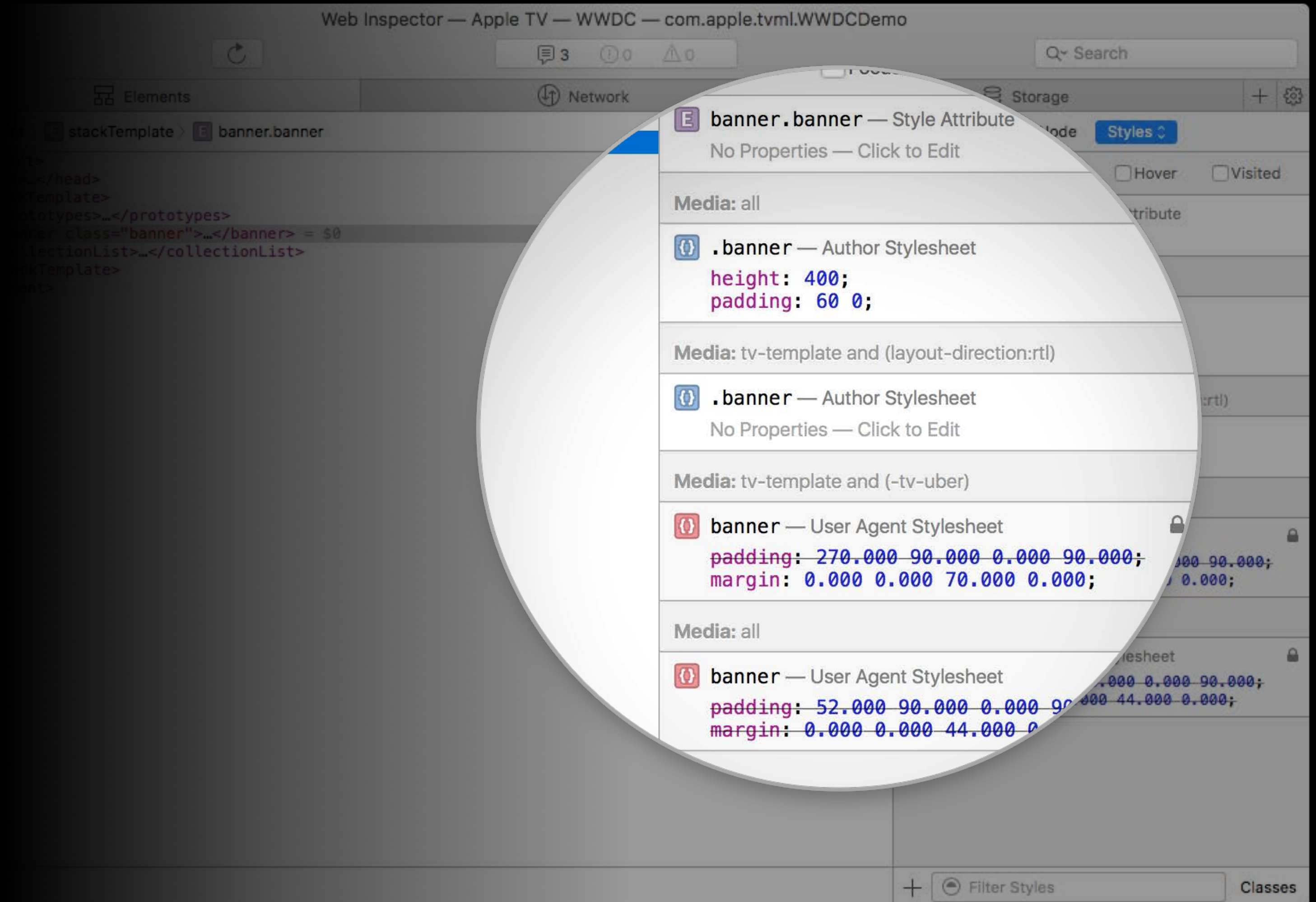


Visual Debugging

Inspect and modify styles

NEW

Per node basis



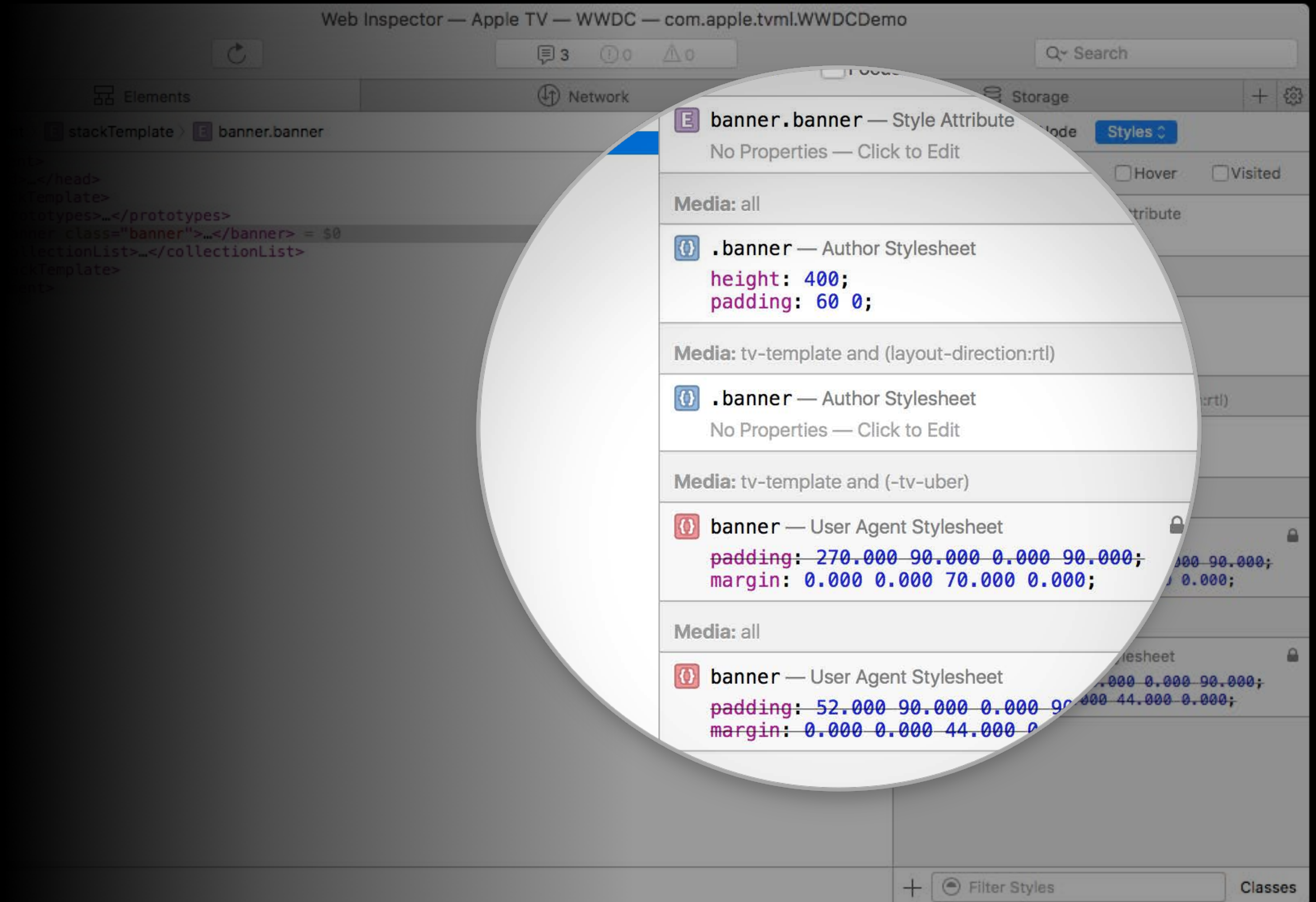
Visual Debugging

Inspect and modify styles

NEW

Per node basis

Cascade ordering



Visual Debugging

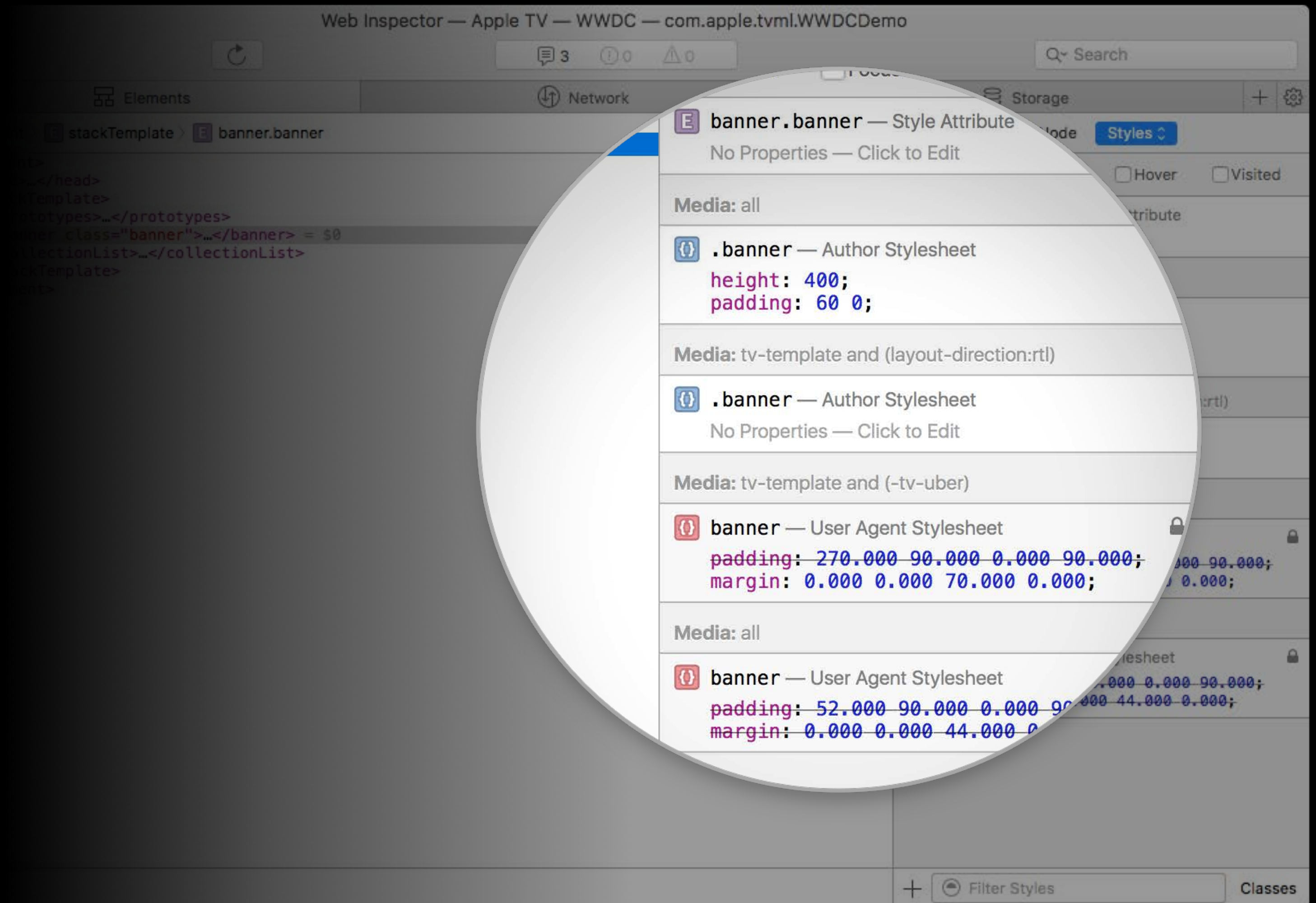
Inspect and modify styles

NEW

Per node basis

Cascade ordering

Media queries



Visual Debugging

Inspect and modify styles

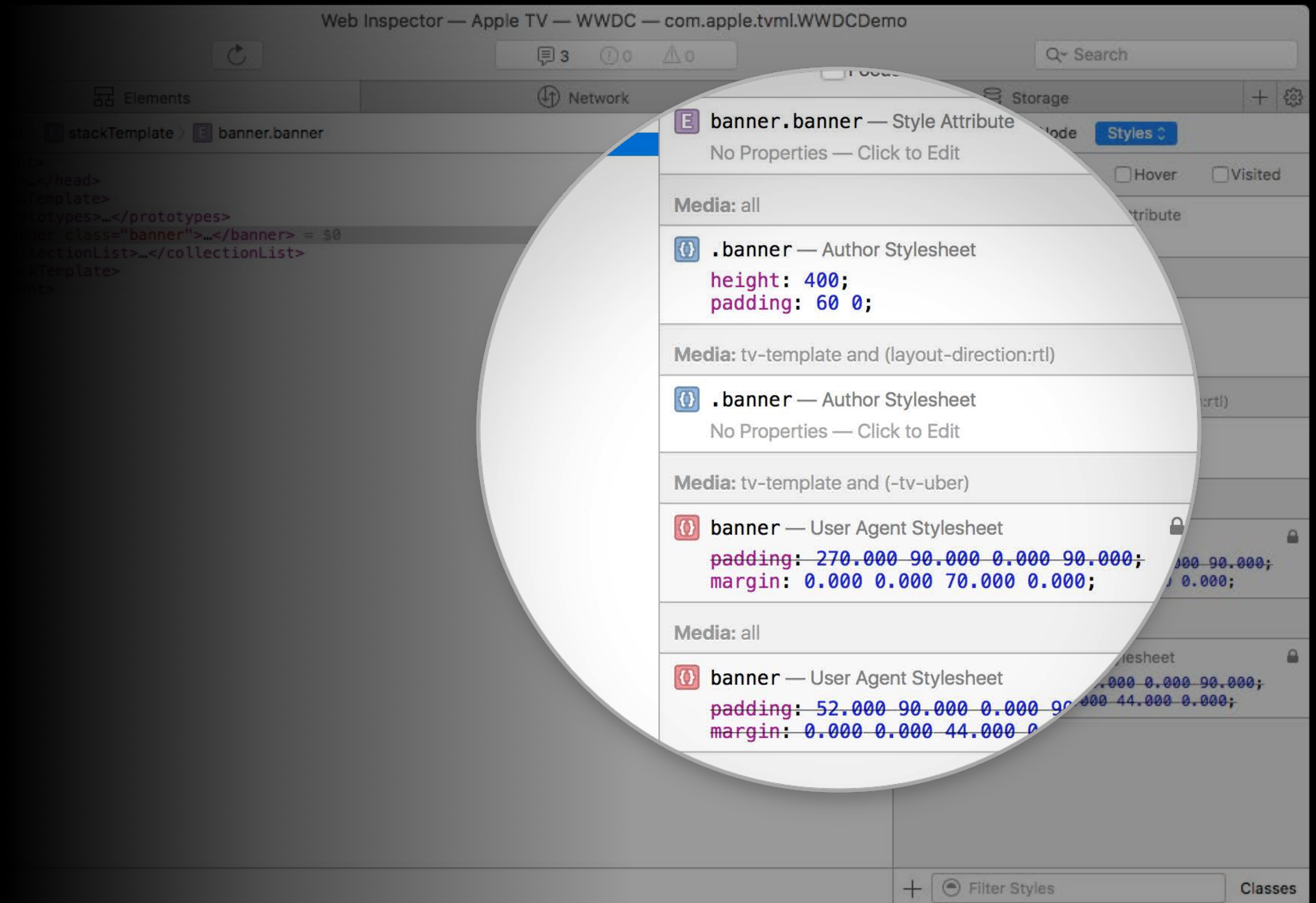
NEW

Per node basis

Cascade ordering

Media queries

Default rules

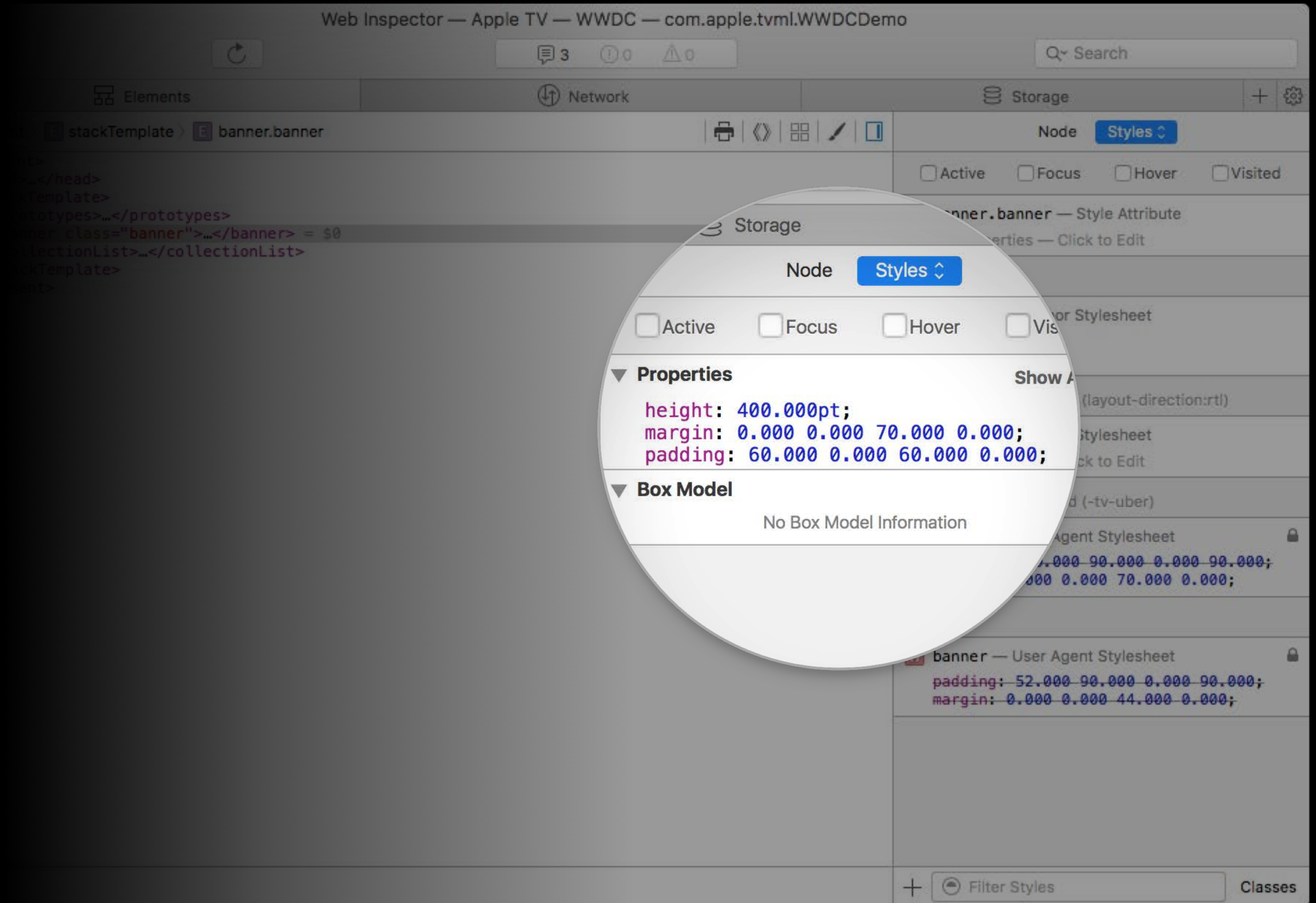


Visual Debugging

Inspect and modify styles

NEW

Computed styles

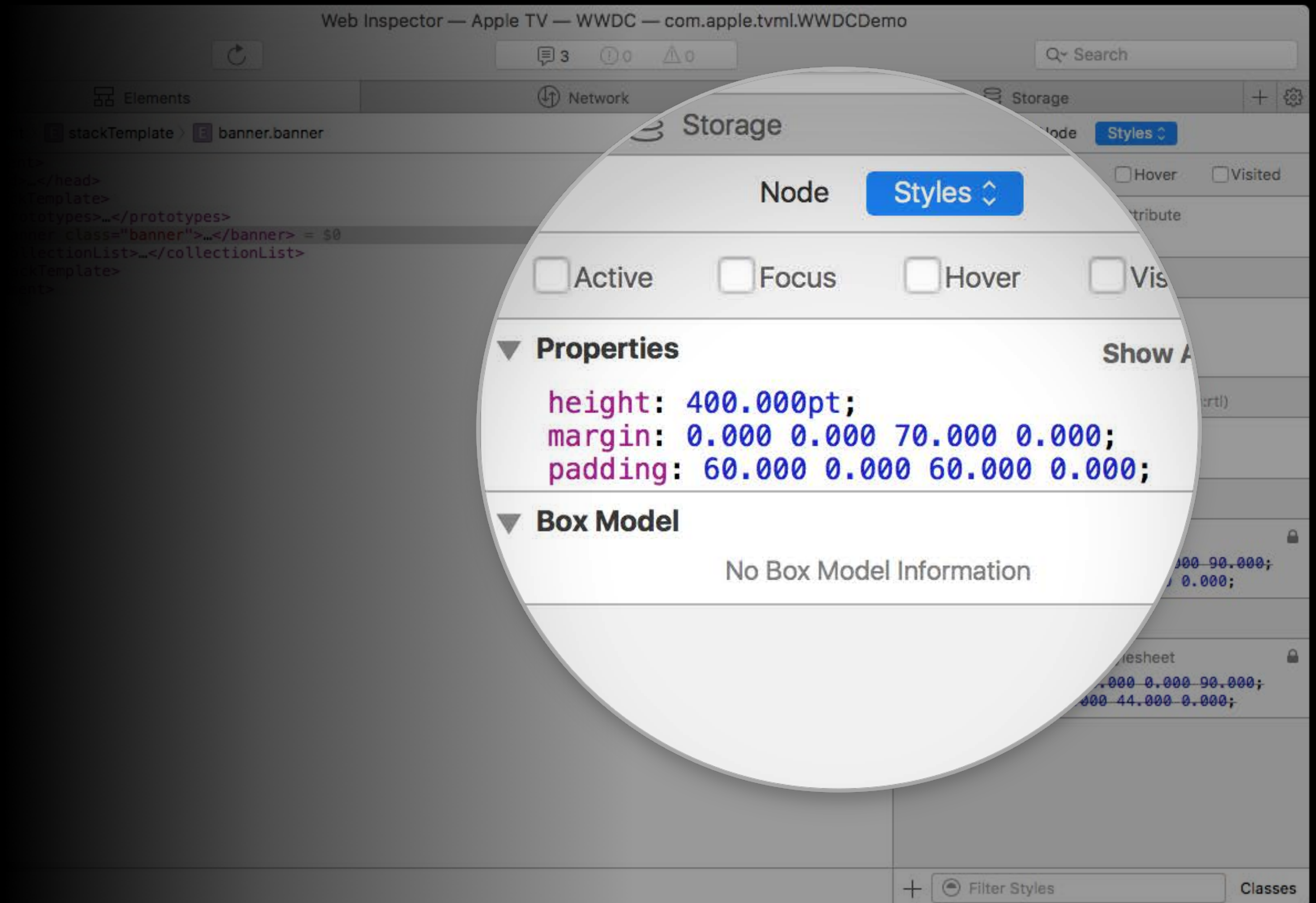


Visual Debugging

Inspect and modify styles

NEW

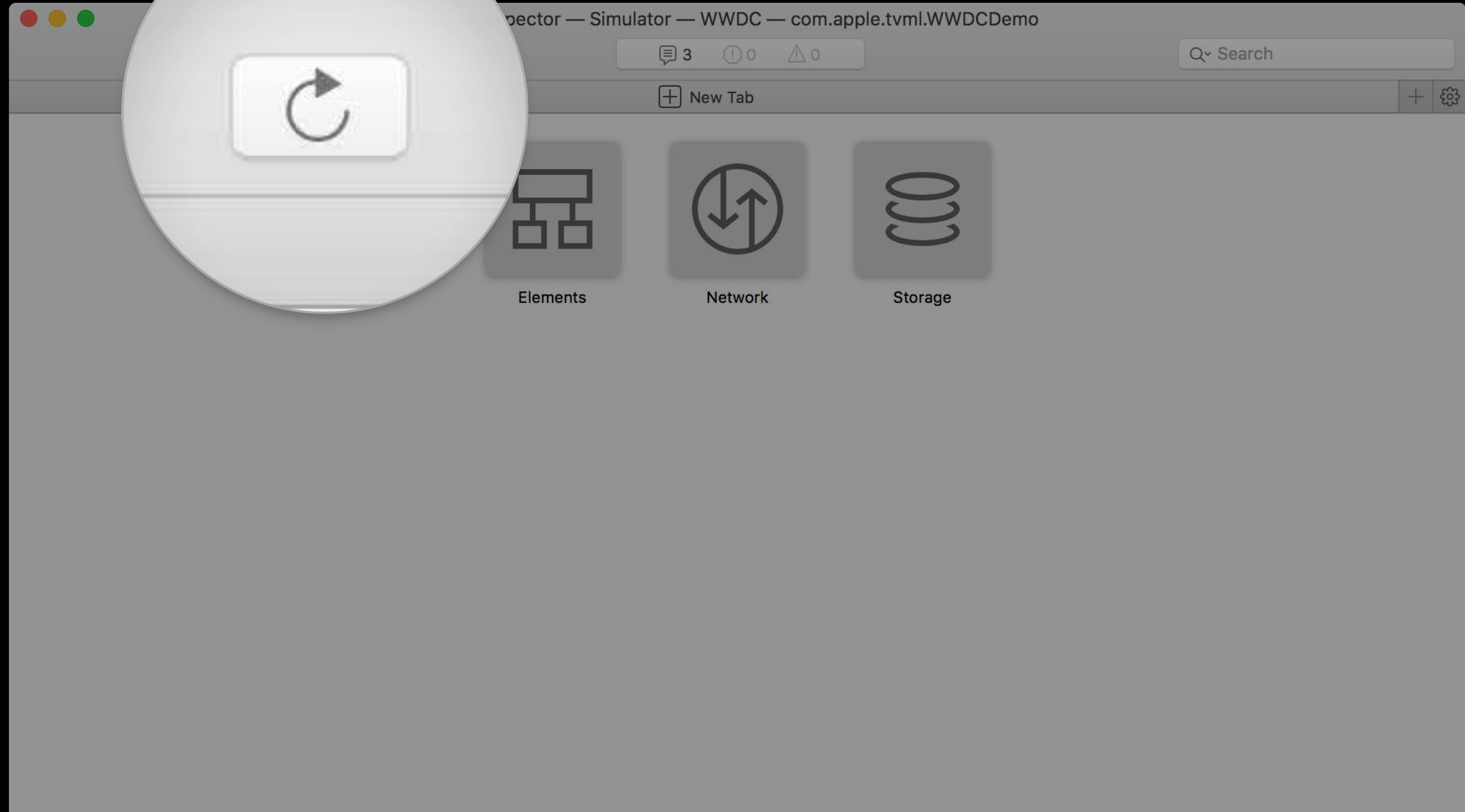
Computed styles



Web Inspector

Reload

NEW



Demo

Recap

Visualize elements

Real time editing

Reload

Network Analysis



Web Inspector — Simulator — WWDC — com.apple.tvml.WWDCDemo

Elements Network Storage

All Resources Documents

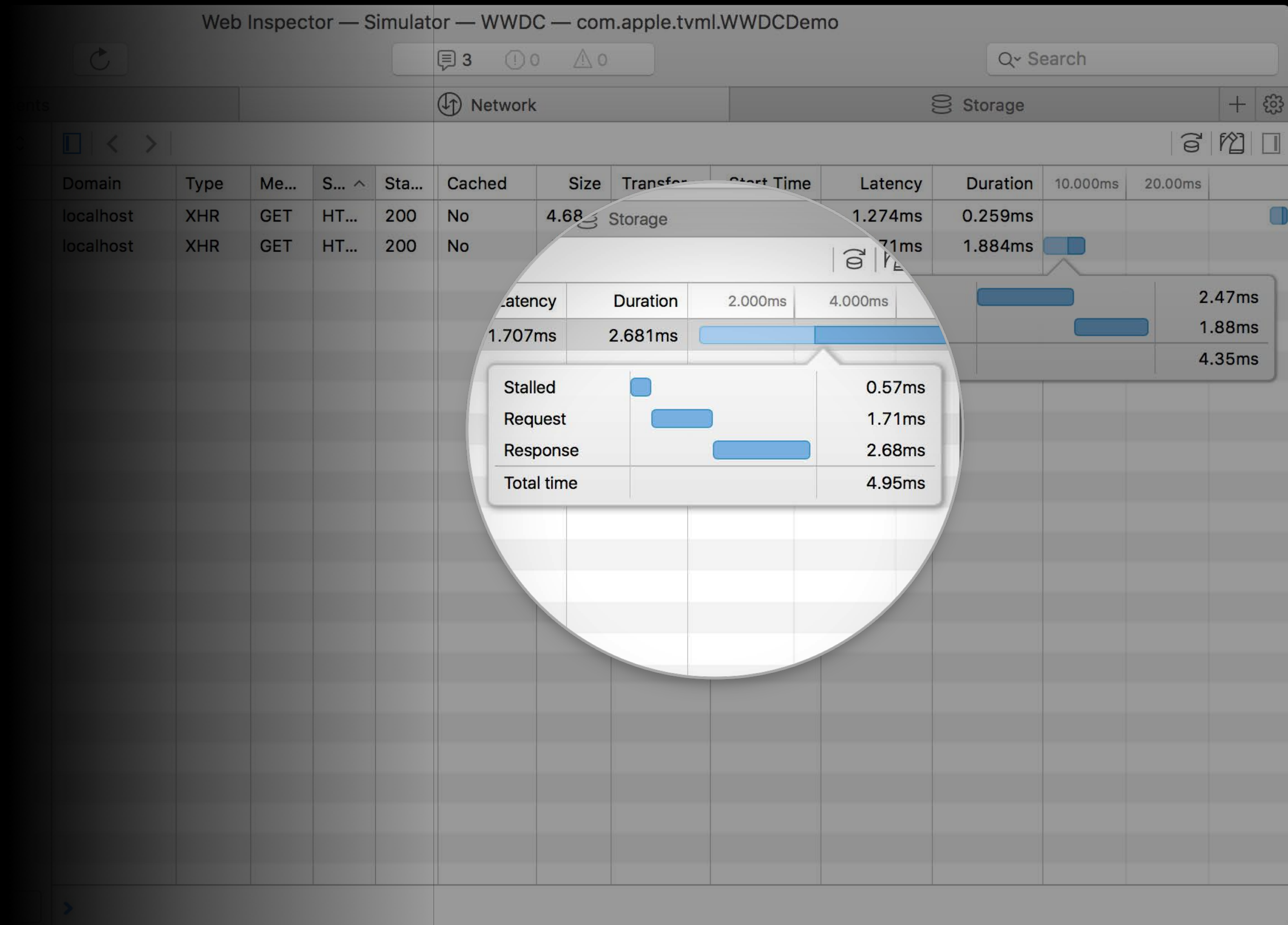
Name	Domain	Type	Me...	S... ^	Sta...	Cached	Size	Transfer...	Start Time	Latency	Duration	10.000ms	20.00ms
wwdc.tvml	localhost	XHR	GET	HT...	200	No	4.68 ...	4.93 KB	23.66ms	1.274ms	0.259ms		
wwdc.json	localhost	XHR	GET	HT...	200	No	762...	762.74 ...	0ms	2.471ms	1.884ms		

Request: 2.47ms
Response: 1.88ms
Total time: 4.35ms

Network Analysis

NEW

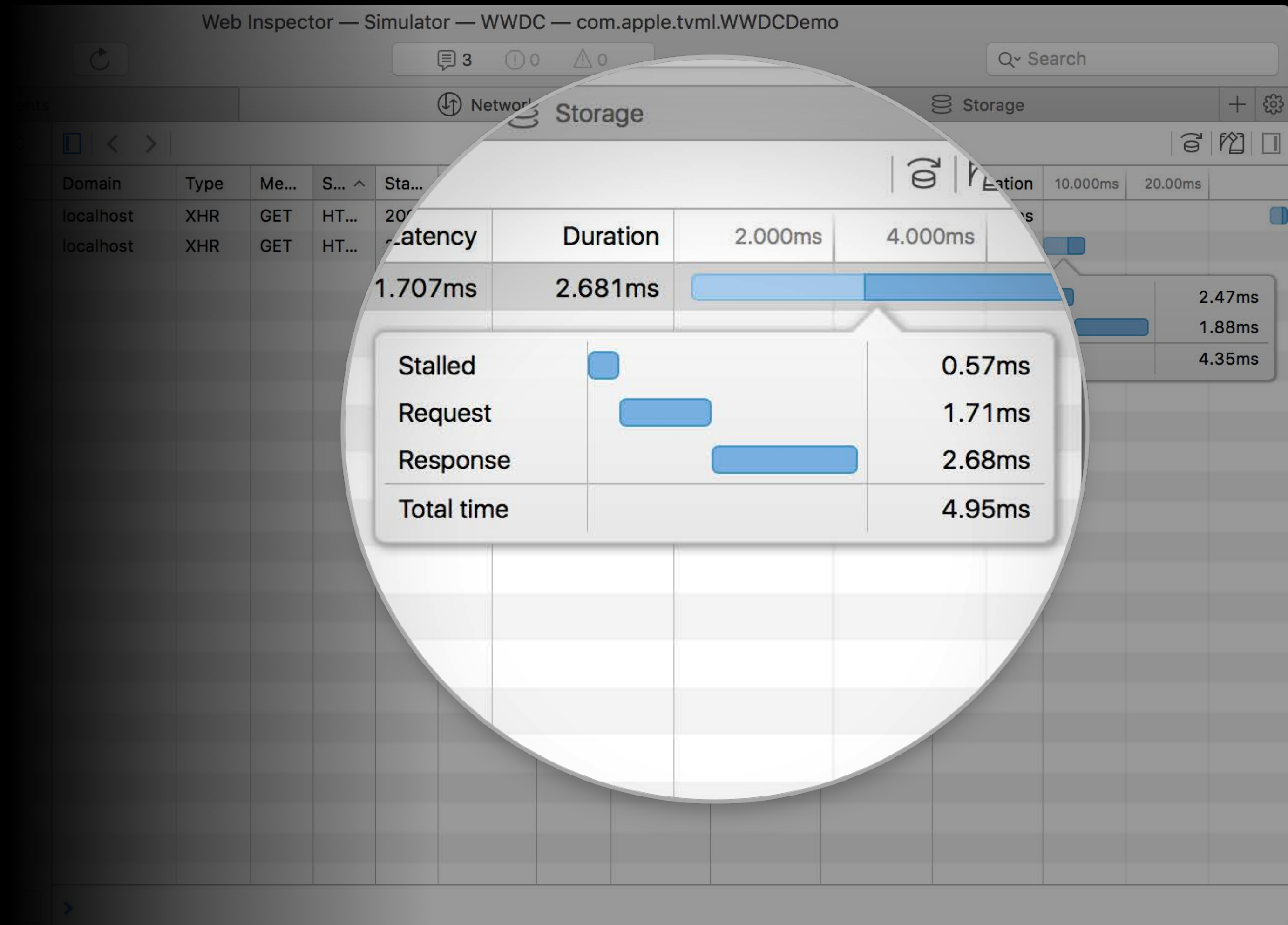
Per request timing information



Network Analysis

NEW

Per request timing information

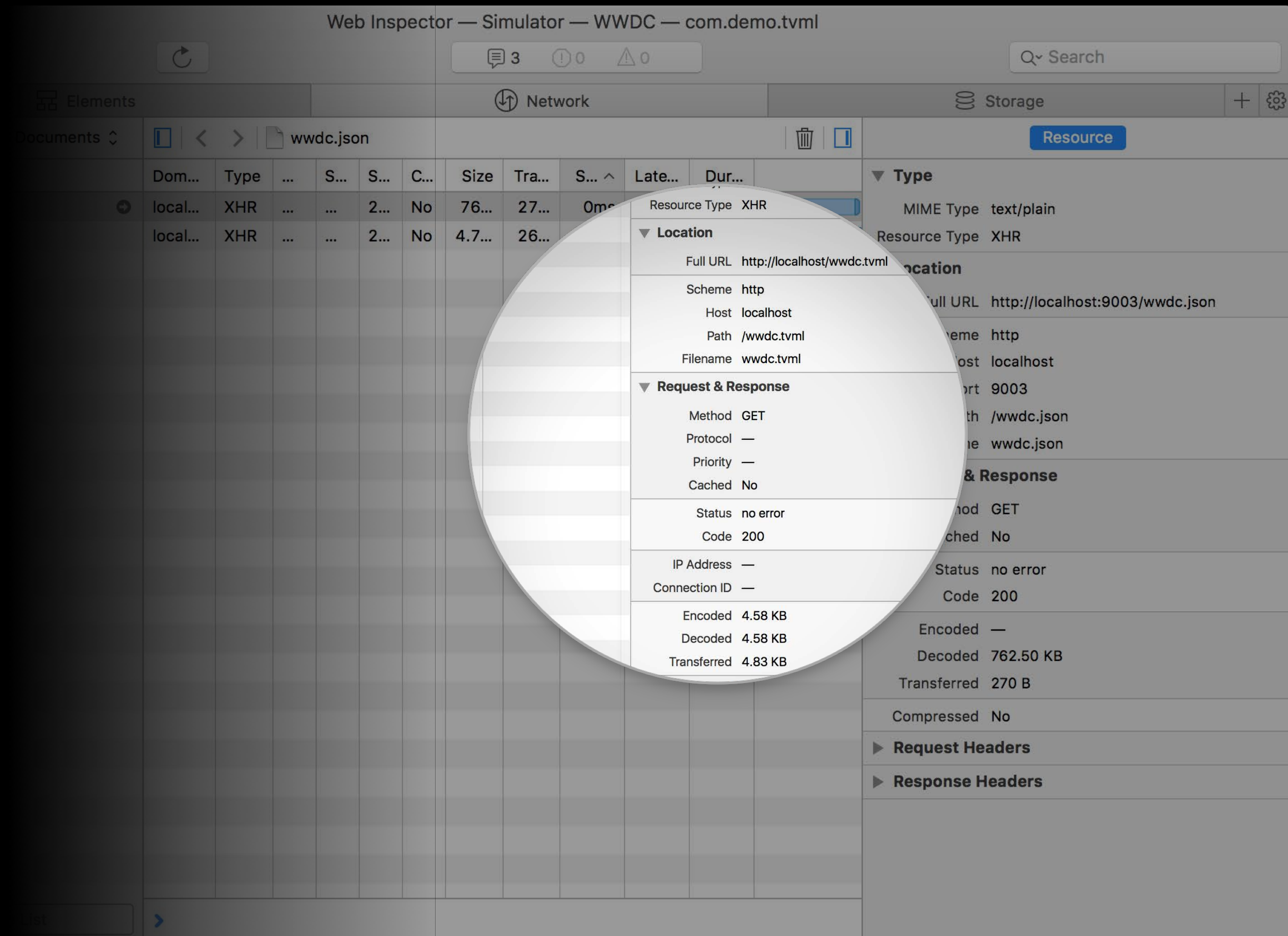


Network Analysis

NEW

Per request timing information

Request properties

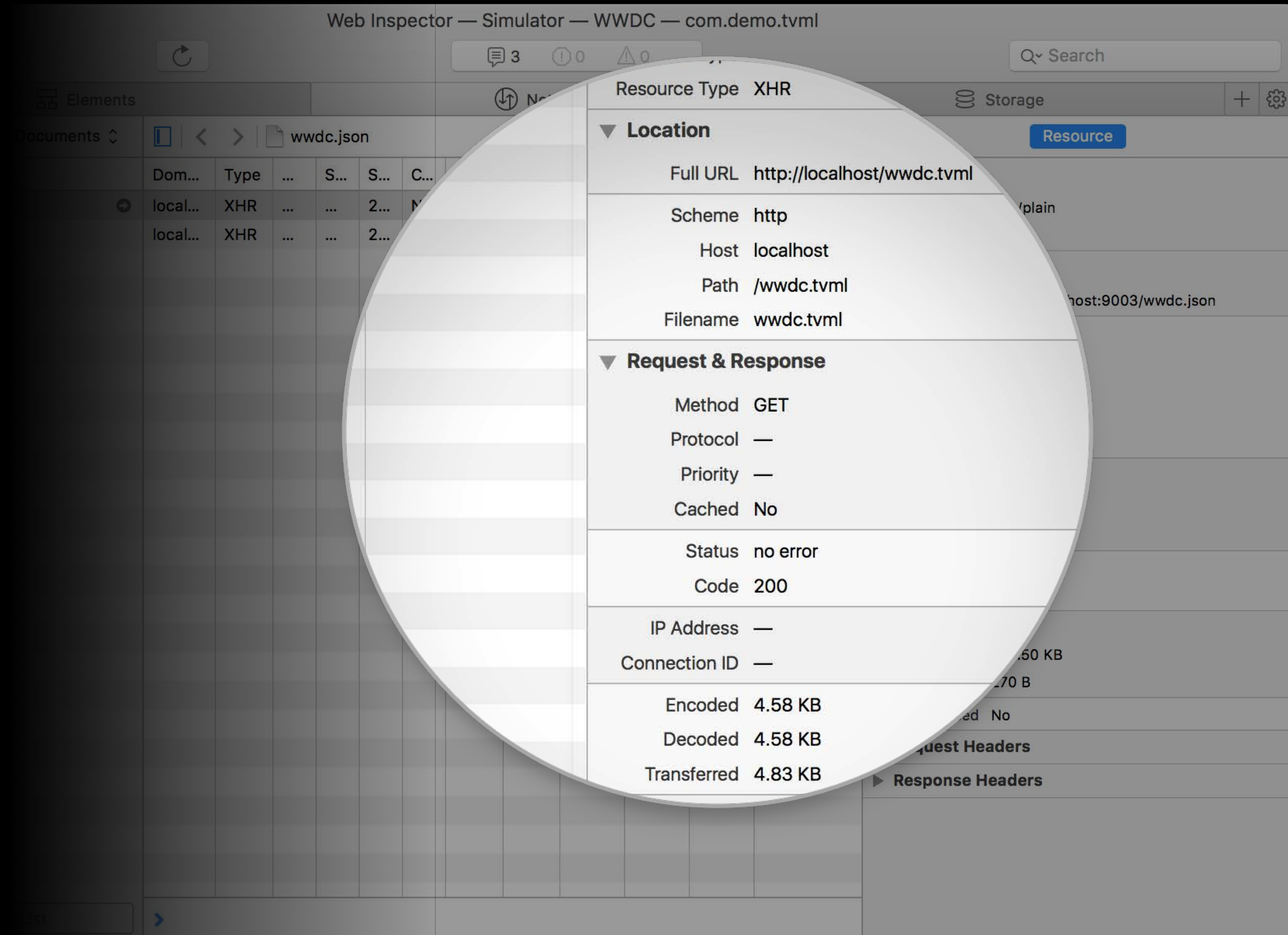


Network Analysis

NEW

Per request timing information

Request properties



Network Analysis

NEW

Per request timing information

Request properties

Headers

The screenshot shows the Web Inspector interface with the Network tab selected. A resource named 'wwdc.json' is highlighted, and its headers are displayed in a circular callout. The callout is divided into 'Request Headers' and 'Response Headers' sections.

Request Headers	
Name	Value
Accept-Encod...	gzip, deflate
Accept	*/*
Accept-Lang...	en-us

Response Headers	
Name	Value
Server	nginx/1.13.0
Content-Type	application/octet-stream
Connection	keep-alive
Last-Modified	Fri, 12 May 2017 21:19:00 GMT
Accept-Ranges	bytes
Date	Tue, 16 May 2017 19:57:13 GMT
Content-Len...	4690
Etag	"591626c4-1252"

Network Analysis

NEW

Per request timing information

Request properties

Headers

The screenshot shows the Web Inspector interface with a circular callout highlighting the 'Request & Response' section. The 'Request Headers' table is as follows:

Name	Value
Accept-Encod...	gzip, deflate
Accept	*/*
Accept-Lang...	en-us

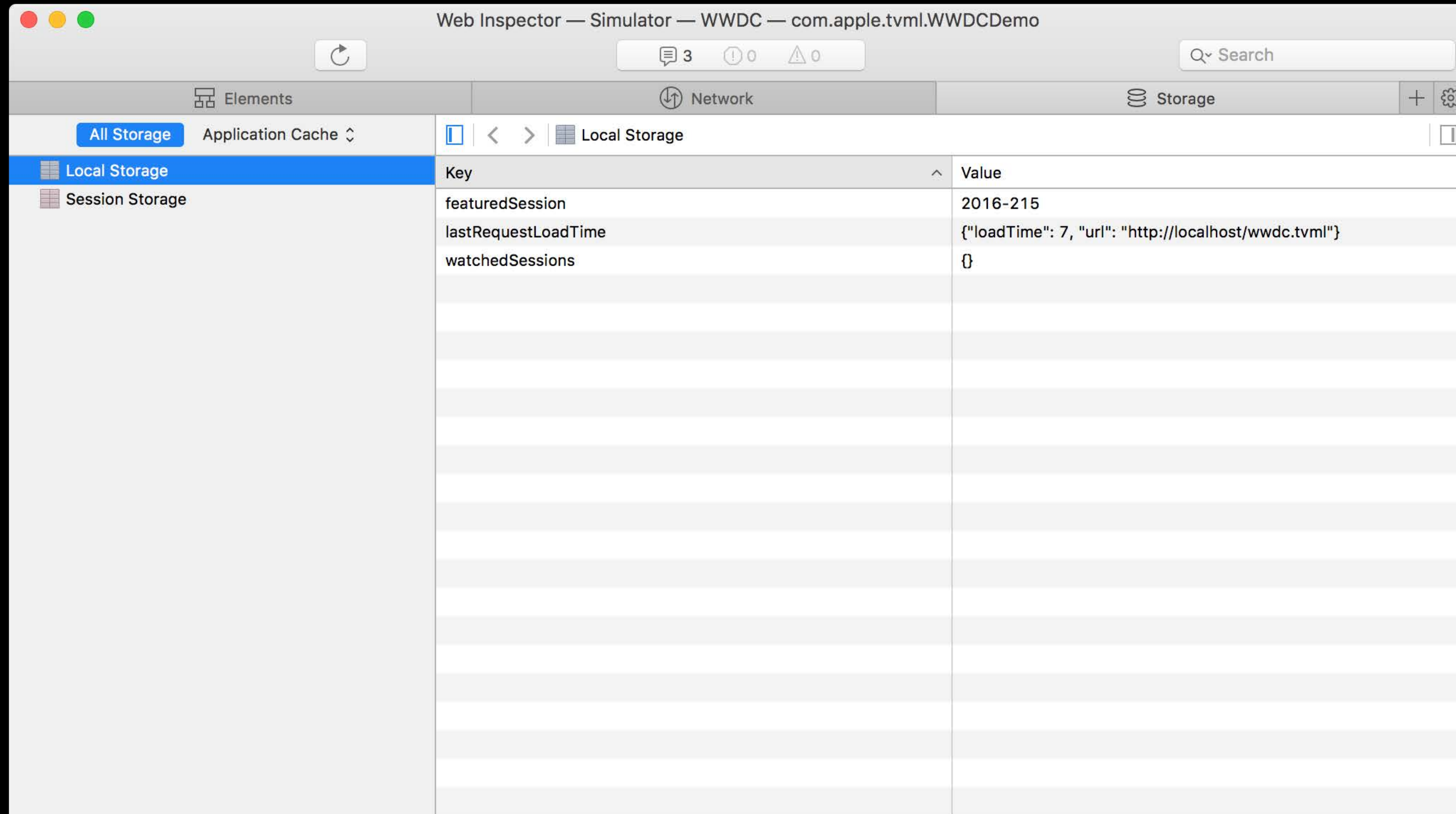
The 'Response Headers' table is as follows:

Name	Value
Server	nginx/1.13.0
Content-Type	application/octet-stream
Connection	keep-alive
Last-Modified	Fri, 12 May 2017 21:19:00 GMT
Accept-Ranges	bytes
Date	Tue, 16 May 2017 19:57:13 GMT
Content-Len...	4690
Etag	"591626c4-1252"

Local and Session Storage

Inspection

NEW



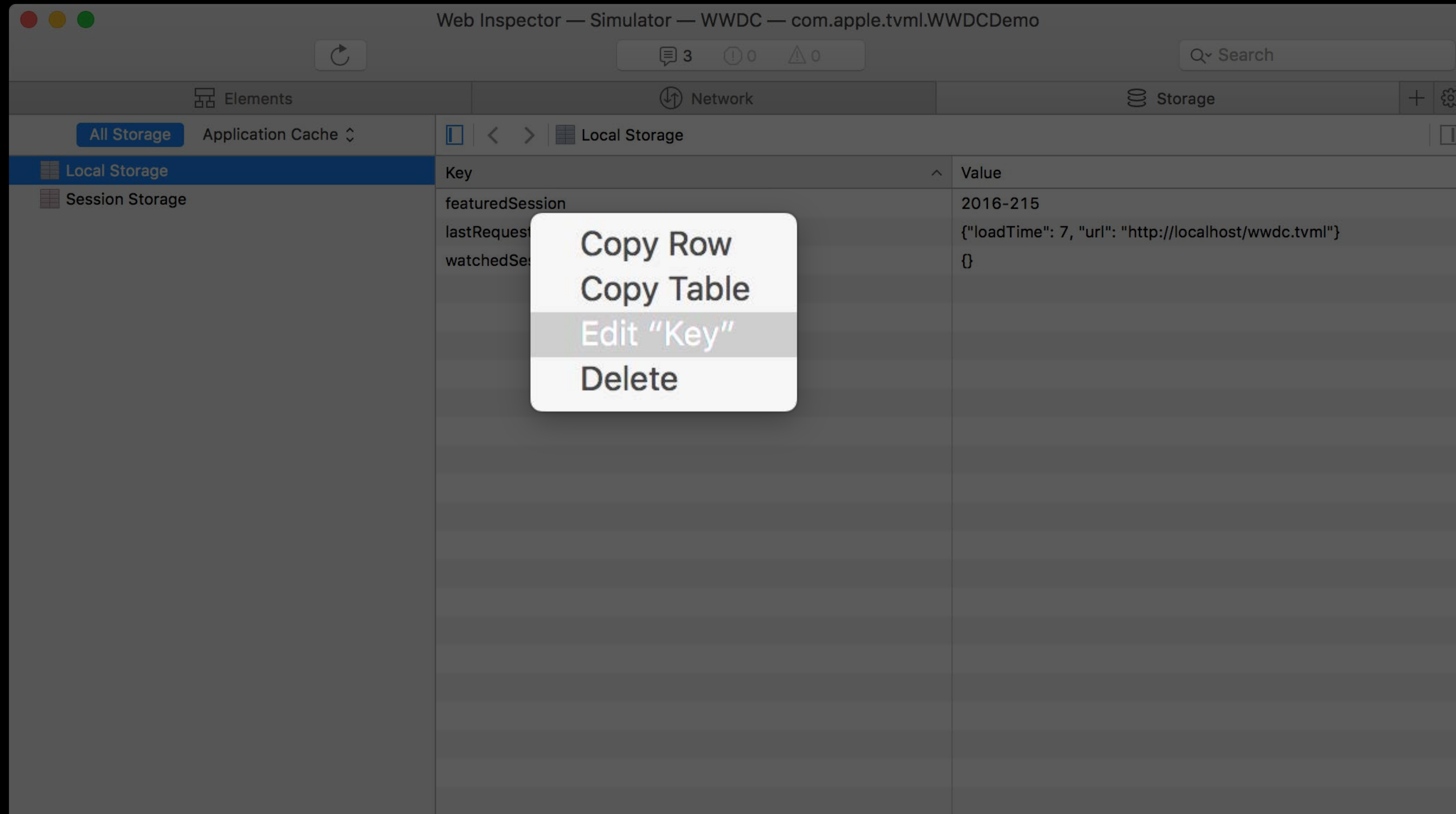
The screenshot shows the Web Inspector interface for a simulator. The title bar reads "Web Inspector — Simulator — WWDC — com.apple.tvml.WWDCDemo". The interface includes a search bar, a refresh button, and three tabs: "Elements", "Network", and "Storage". The "Storage" tab is active, showing a sidebar with "All Storage" (selected), "Application Cache", "Local Storage", and "Session Storage". The main area displays the "Local Storage" data as a table with two columns: "Key" and "Value".

Key	Value
featuredSession	2016-215
lastRequestLoadTime	{"loadTime": 7, "url": "http://localhost/wwdc.tvml"}
watchedSessions	{}

Local and Session Storage

Inspection

NEW



The screenshot shows the Web Inspector interface for a simulator. The 'Storage' tab is active, displaying a table of Local Storage items. A context menu is open over the first row, offering actions: Copy Row, Copy Table, Edit "Key", and Delete.

Key	Value
featuredSession	2016-215
lastRequest	{"loadTime": 7, "url": "http://localhost/wwdc.tvml"}
watchedSe	{}



+

TVMLKit



TVMLKit

Summary

Supports RTL for default templates

Use data binding/prototypes for large data sets

Web Inspector reduces development cycle

More Information

<https://developer.apple.com/wwdc17/202>

Related Sessions

Localizing with Xcode 9

Tuesday 10:20AM

What's new in tvOS

Grand Ballroom B

Wednesday 10:00AM

Focus Interaction in tvOS 11

Grand Ballroom A

Thursday 9:00AM

Developing tvOS Apps Using TVMLKit: Part 1

WWDC2016

Developing tvOS Apps Using TVMLKit: Part 2

WWDC2016

Internationalization Best Practices

WWDC2016

What's New in International User Interfaces

WWDC2016

Labs

tvOS Lab

Technology Lab I

Tuesday 12:00-13:50

tvOS Lab

Technology Lab H

Wednesday 11:00-14:00

tvOS Lab

Technology Lab I

Thursday 11:00-13:00

Internationalization Lab

Technology Lab I

Tuesday 13:50-16:10

Internationalization Lab

Technology Lab I

Friday 9:00-11:00

