

seaside



seaside 

**A web framework
that doesn't suck!**

seaside 

**Seaside is
different...**

seaside 

...in many ways

- ★ **As much state as possible on the server!**
- ★ **Forget about meaningful URLs!**
- ★ **Skip templates!**
- ★ **Build it all in Smalltalk!**

seaside 

Seaside magic

 **Components**

 **Callbacks**

 **Control flow**

 **...and some candy on top!**

seaside 

**We don't think in
pages...**

seaside 

**...but in stateful
components**

seaside 

A Component

- ★ Inherits from WACComponent
- ★ Can have children components
- ★ Has a life span of the user session
- ★ Draws itself on a given HTML canvas

seaside 

HelloWorld

```
renderContentOn: html  
html render: 'Hello world'
```



ClickList

```
renderContentOn: html
  html orderedList: [
    1 to: 5 do: [:i |
      html listItem: [
        html anchor
          callback: [self inform: 'Item #', i asString];
          with: 'Click item #', i asString]
      ]
    ]
  ]
```

seaside 

SumItUp

```
go
  | a b |
  a := self request: 'a'.
  b := self request: 'b'.
  self inform: 'a + b = ', (a asNumber + b asNumber) asString
```



TodoList

todoList

```
^todoList ifNil: [todoList := OrderedCollection new]
```

renderContentOn: html

```
html heading: 'Todo list'.
```

```
html orderedList: [
```

```
  self todoList do: [:each |
```

```
    html listItem: [
```

```
      html render: each.
```

```
      html space anchor
```

```
        callback: [self todoList remove: each];
```

```
        with: 'remove']]].
```

```
html form: [
```

```
  html textInput
```

```
    callback: [:value | self todoList add: value].
```

```
  html submitButton text: 'Add']
```



Combo

children

```
^children ifNil: [children := OrderedCollection new]
```

renderContentOn: html

```
html unorderedList: [  
  {HelloWorld. ClickList. SumItUp. TodoList} do: [:each |  
    html listItem: [  
      html anchor  
        callback: [self children add: each new];  
        with: 'Add ', each name]].  
    html listItem: [  
      html anchor  
        callback: [children := OrderedCollection new];  
        with: 'Clear']].  
self children do: [:each | html render: each]
```

seaside

Yummy!

- ★ No request parsing!
- ★ No XML files!
- ★ No HTML templates!
- ★ Real independent components!
- ★ XHTML! CSS!
- ★ Secure!

seaside 

What about the candy?

- ★ Fully integrated Scriptaculous! (Ajax)
- ★ Fully integrated Comet! (server push)
- ★ Standard Components included
- ★ Lots of addon components
- ★ Extra layers like Pier or Magritte
- ★ RSS, HTML5 etc etc



Noteworthy

- ★ Strong core of developers
- ★ Very high code quality
- ★ MIT license
- ★ De facto standard in the Smalltalk world
 - ★ Squeak, VisualWorks, Gemstone, Dolphin etc
- ★ Does not try to be a "full stack" solution
- ★ Dabbledb.com! **Watch the 8 minute demo**

seaside

"But once you've looked at Seaside, and you really understand the approach Seaside is taking, Rails just looks old. I never want to marshal data through a URL again if I can help it." -- Ken Treis

www.seaside.st

and... ok then:
continuity.tlt42.org