



## REST URLs

Lukas Renggli

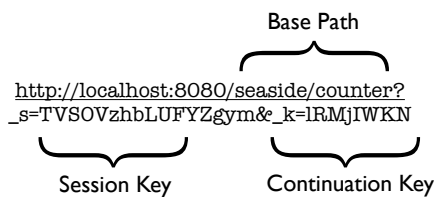
[renggli@iam.unibe.ch](mailto:renggli@iam.unibe.ch), University of Bern  
[renggli@netstyle.ch](mailto:renggli@netstyle.ch), netstyle.ch GmbH

## REST

- ✧ REST stands for *Representational State Transfer*
- ✧ Every URI is a *fixed resource* that can be used with a set of actions: GET, PUT, POST, ...
- ✧ No matter what *server* and *client* is used a given URI represents the same resource.
- ✧ All the state is passed around in URLs, to be able to pick up where one left off by accessing the URL at a later time.

## Seaside URLs

- ✧ Seaside is in conflict with REST because URLs don't define resources.



## Modifying URLs

- ✧ Override `#updateUrl`: in any component:
  - ✧ Add path elements:  
`anUrl addToPath: 'subdir/subsubdir'.`
  - ✧ Add parameters:  
`anUrl addParameter: 'language' value: 'en'.`
  - ✧ Add a fragment:  
`anUrl addFragment: 'bottom'.`

## Parsing URLs

- ✧ Create a subclass of `WARenderLoopMain`.
- ✧ Override the method `#start`: to analyze the request with its URL and to create and setup the *root component* of your application.
- ✧ Register your new subclass in the configuration interface as the *main class* to be used with your application.

## Summary

With Seaside one can build REST applications, but this requires some more work.

## Further Reading

- ✧ Avi Bryant, HREF Considered Harmful
  - ✧ <http://www.cincomsmalltalk.com/userblogs/avi/blogView?entry=3256899497>
- ✧ Lukas Renggli, Seaside Tutorial:
  - ✧ Exercise: 41 – 42