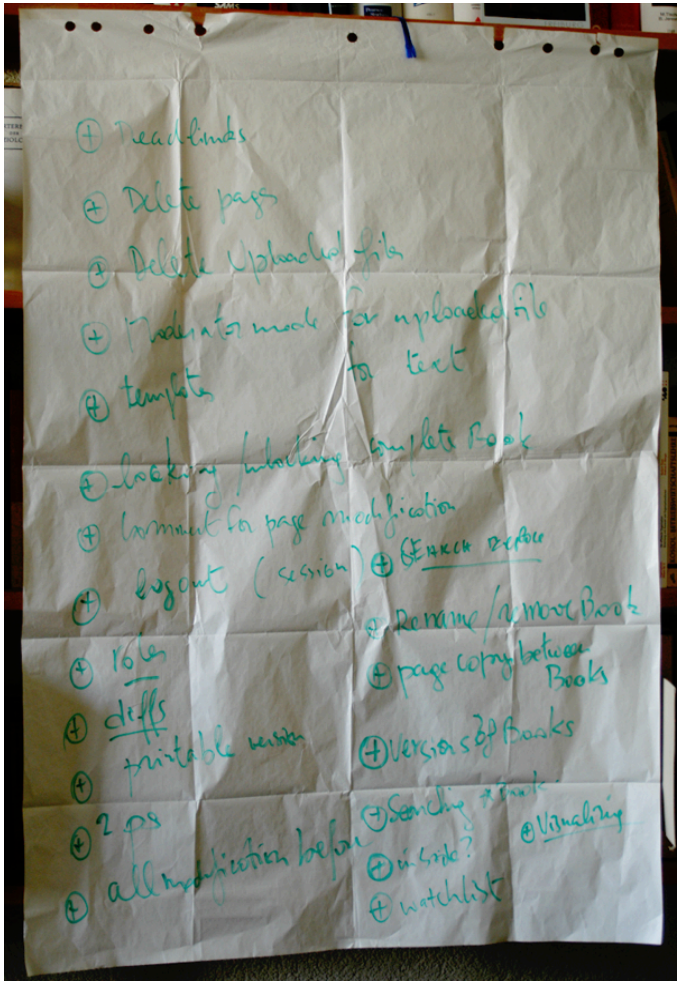
Smalltalk Wiki
towards CMS

Lukas Renggli

Why another Wiki?



- Object Oriented Design
- Extensibility
  - Components
  - Web server
  - Storage
- Open source
- Dialect independent
- Test Suites

Roadmap

- User
  - Editing
  - Navigation
- Administrator
  - Templates
- Developer
  - Testing
  - Implementation
  - Coding Session
- Summary

User

Editing
Navigation

View | Edit | History | Recent Changes | Template

Page Example 1

Paragraph

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas justo pede, feugiat vel, iaculis in, consectetuer ut, ipsum.

Unordered List

- Lorem ipsum dolor sit amet
- Consectetuer adipiscing elit

Links

Page Example 2

Prevoius Parent Next

SmallWiki

SmallWiki / Presentation / Page Example 2

Actions

View
Edit
History
Recent Changes
Template

Session

Login
Logout

References

Page Example 1

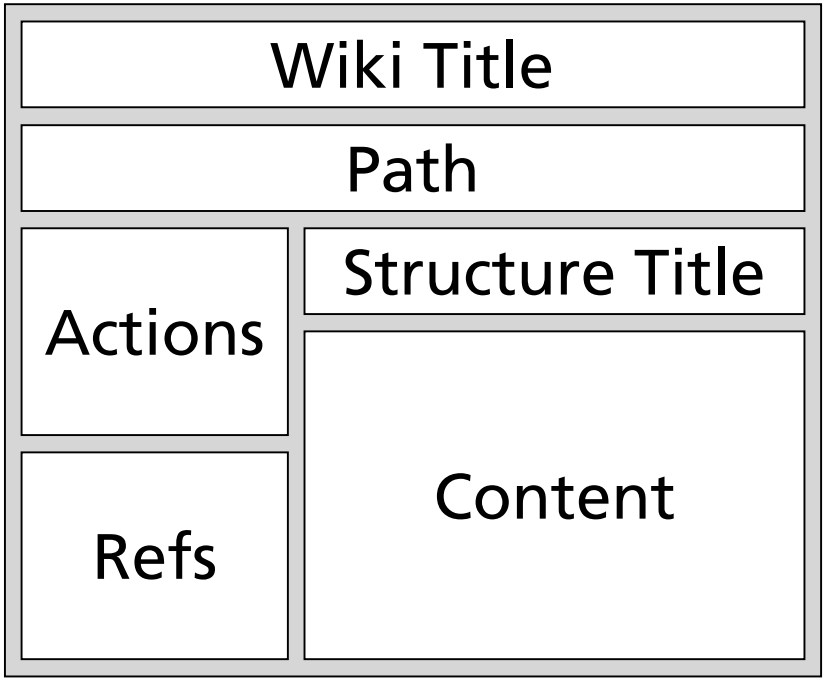
Page Example 2

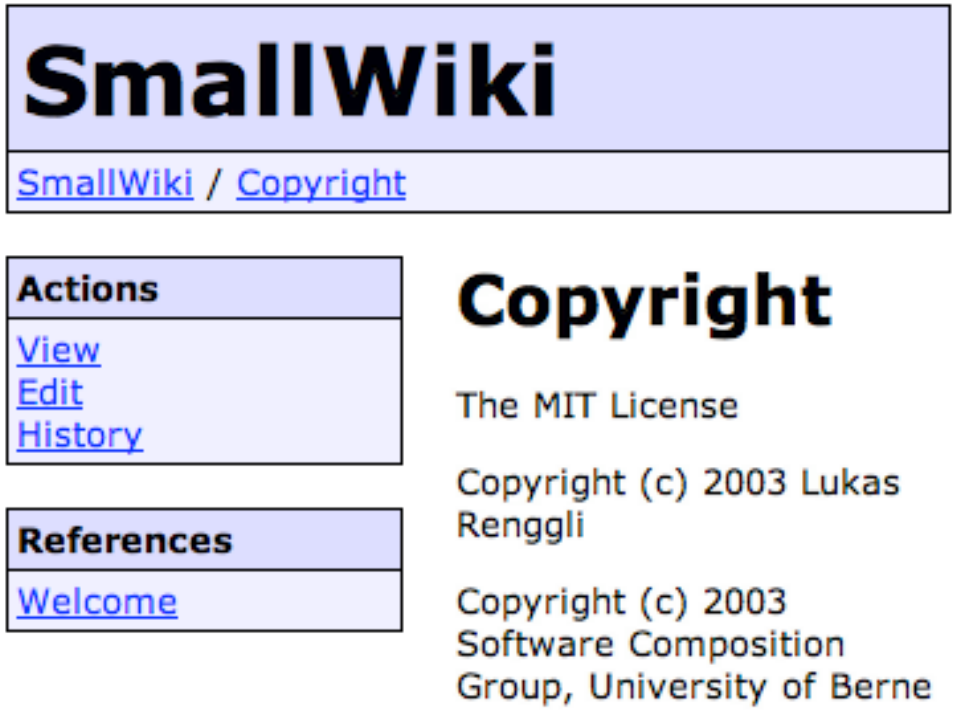
- Renderer
- Actions
- Session
- References
- Path

Administrator

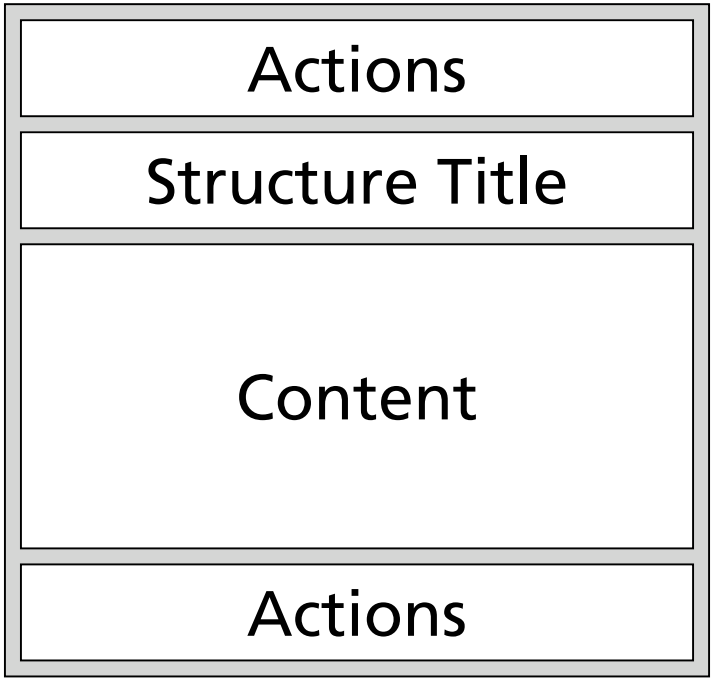
Templates

Template Example I





Template Example II



View Edit History Parent

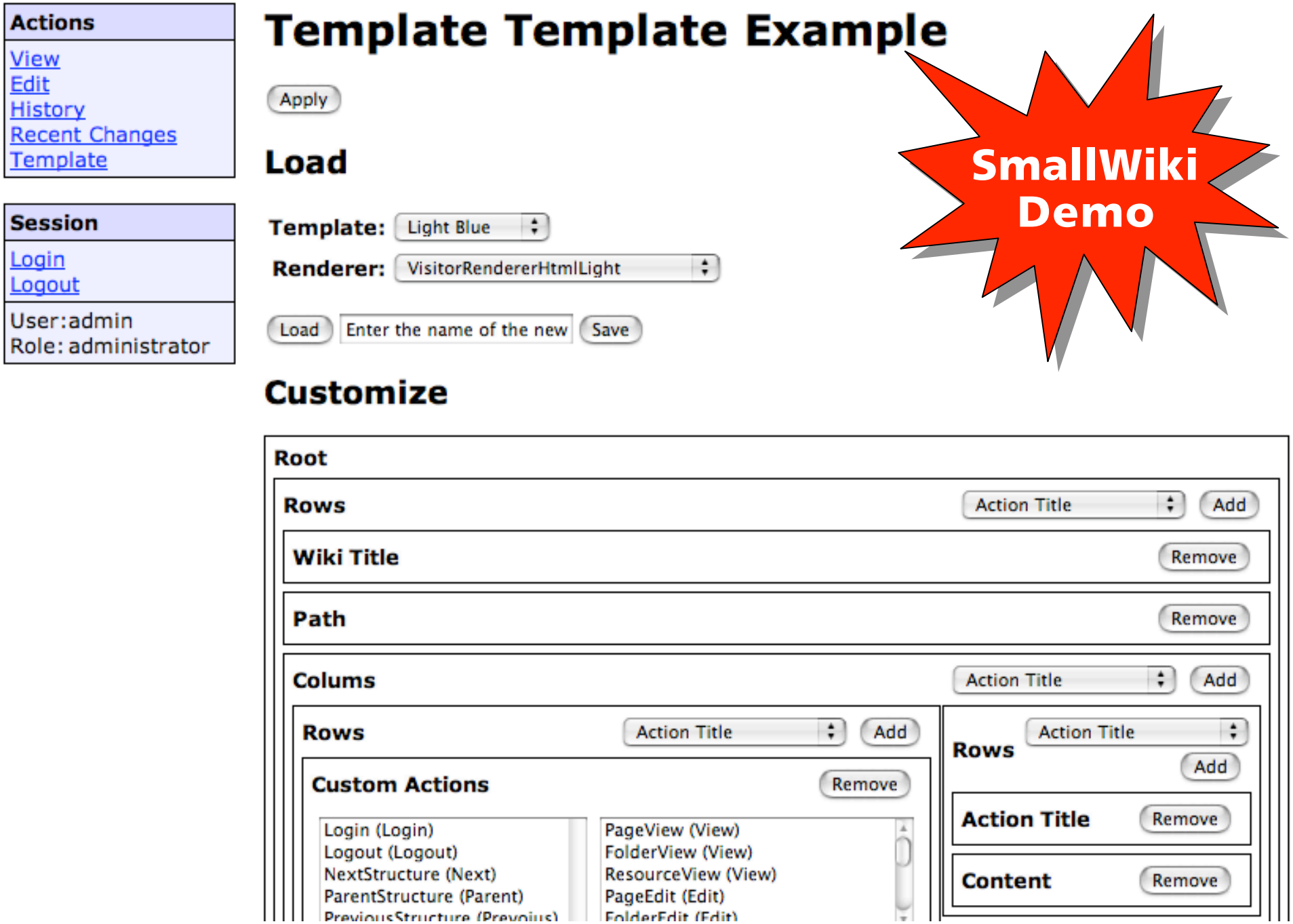
Copyright

The MIT License

Copyright (c) 2003 Lukas Renggli

Copyright (c) 2003 Software Composition Group, University of Berne

View Edit History Parent


Actions
View
Edit
History
Recent Changes
Template
Session
Login
Logout
User:admin
Role: administrator
Template Template Example
Apply
Load
Template:
Light Blue
Renderer:
VisitorRendererHtmlLight
Load
Enter the name of the new
Save
SmallWiki Demo
Customize
Root
Rows
Action Title
Add
Wiki Title
Remove
Path
Remove
Colums
Action Title
Add
Rows
Action Title
Add
Custom Actions
Remove
Login (Login)
Logout (Logout)
NextStructure (Next)
ParentStructure (Parent)
PageView (View)
FolderView (View)
ResourceView (View)
PageEdit (Edit)
Rows
Action Title
Add
Action Title
Remove
Content
Remove

Developer

Testing
Implementation
Coding Session

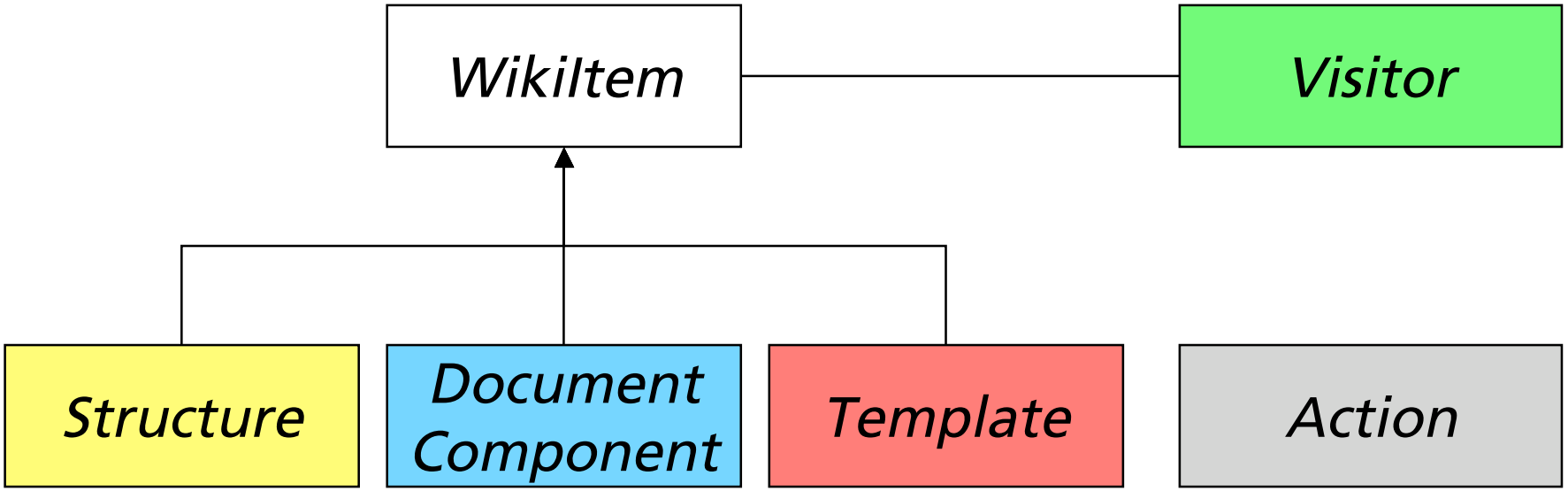
View | Edit | History | Recent Changes | Template

Testing SmallWiki

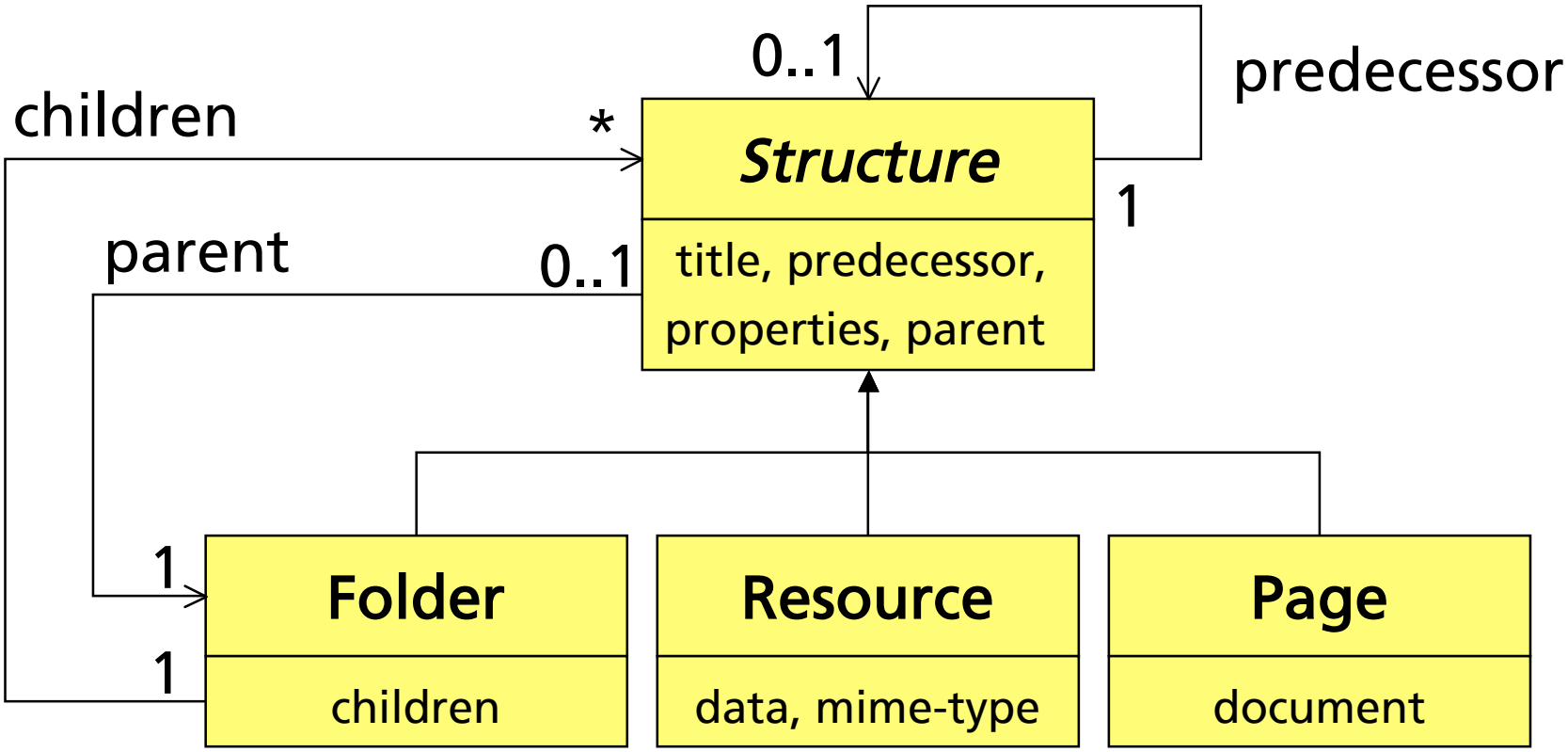
202 run, 202 passed, 0 failed, 0 errors

Prevoius Parent Next

Basic Design

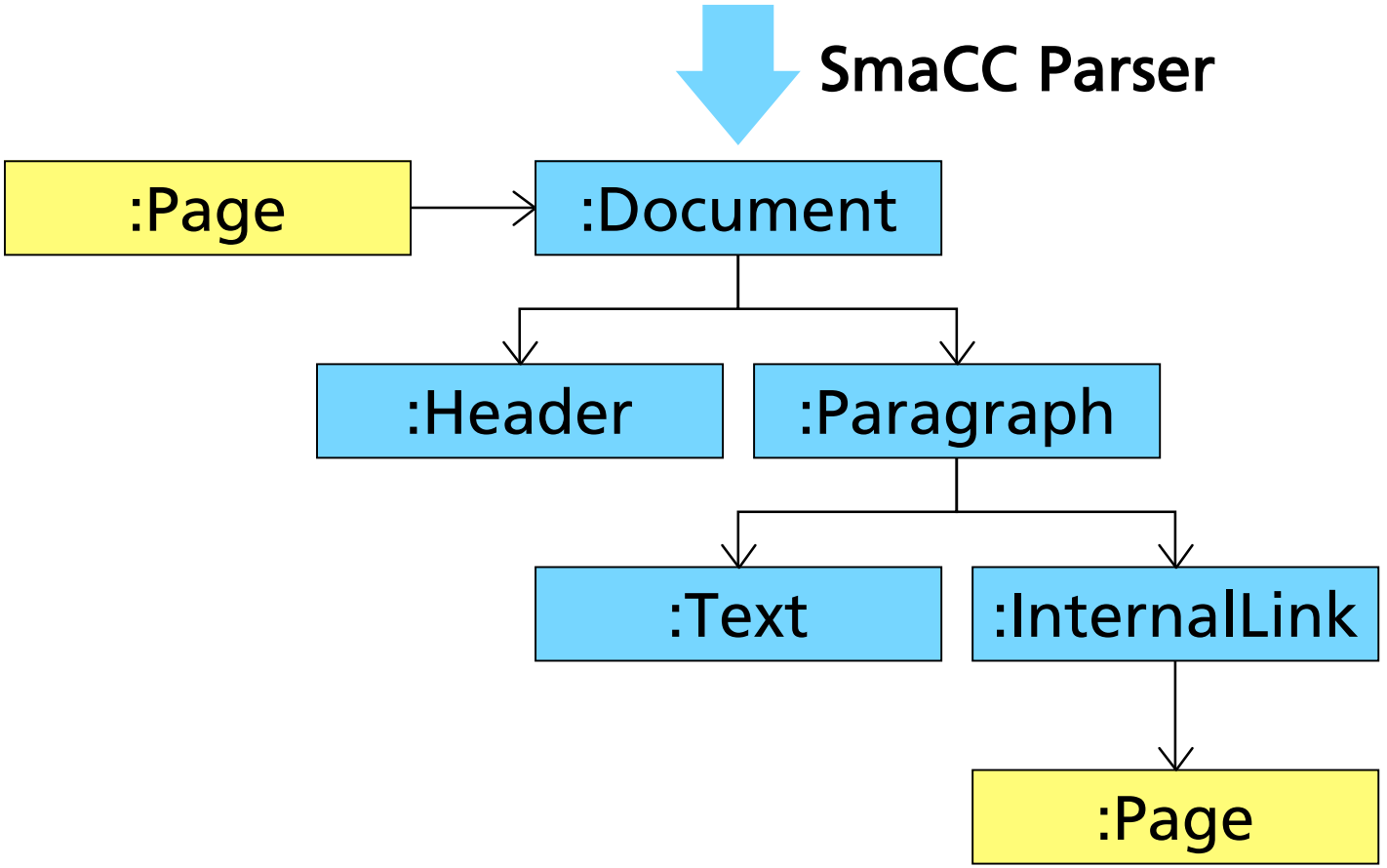


Structure


0..1
predecessor
children
*
Structure
1
parent
0..1
title, predecessor,
properties, parent
1
Folder
Resource
Page
1
children
data, mime-type
document

Wiki Documents

```
!Heading Text
Paragraph containing a *Link*
```



Available Extensions

- Search Engine
- Keyword Index
- Link Collection
- External Link Checker
- Photo Folder
- E-Mail Notification
- Importer

Coding Session

- RSS News Feed
  - BottomFeeder
  - Reuse of Model
- Template Component
  - Visible on all Pages
  - Configuration Interface
  - Automatic Refresh

```
<?xml version="1.0" ?>
<rss version="2.0">
 <channel>
  <item>
   <title>SmallWiki</title>
   <description>A Small ...
   <link>http://www.sm ...
  </item>
  ...
 </channel>
</rss>
```

Summary

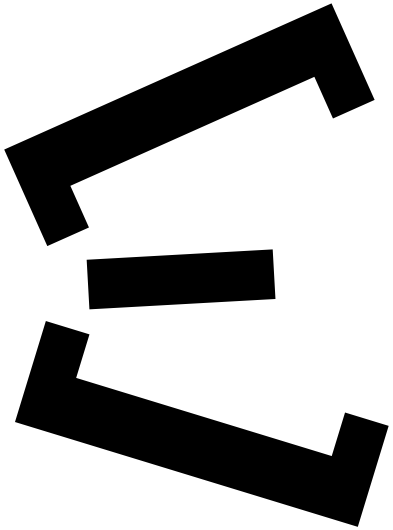
Presented

- Structure
  - Folder
  - Pages
  - Resources
- Documents
  - Parsing
- Templates
- Extensions

Not Seen

- Pluggable
  - Server
  - Storage
- Versioning
- Security
- Actions
- Callbacks

Towards a CMS



- We should not let PHP and Zope win
- SmallCMS: Reusable Components for Content Management
- Join, use, extend, …

References

- Latest Release
  - SCG StORE and Cincom Public StORE
- Running SmallWiki
  - http://kilana.unibe.ch:9090
- Documentation
  - http://c2.com/cgi/wiki?SmallWiki
  - http://scgwiki.iam.unibe.ch:8080/SCG/520
- Related Projects
  - Gardner: CMS using Seaside and SmallWiki
  - Garden: In-Image Wiki Editor for Squeak

General Setup

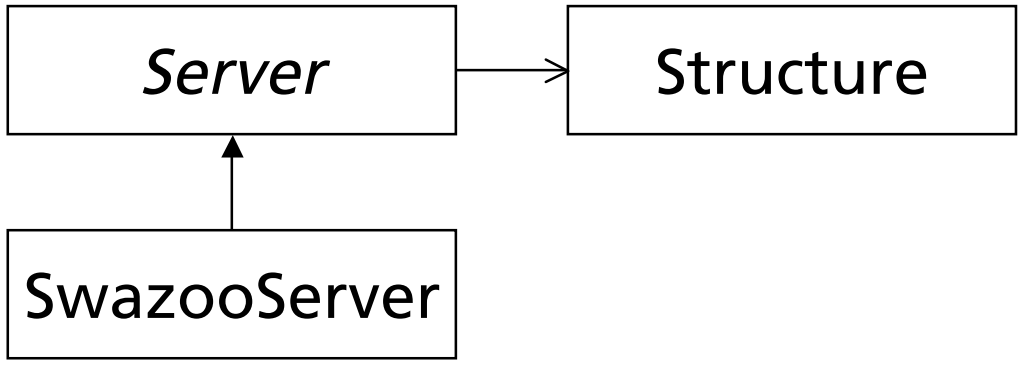
Request

Response

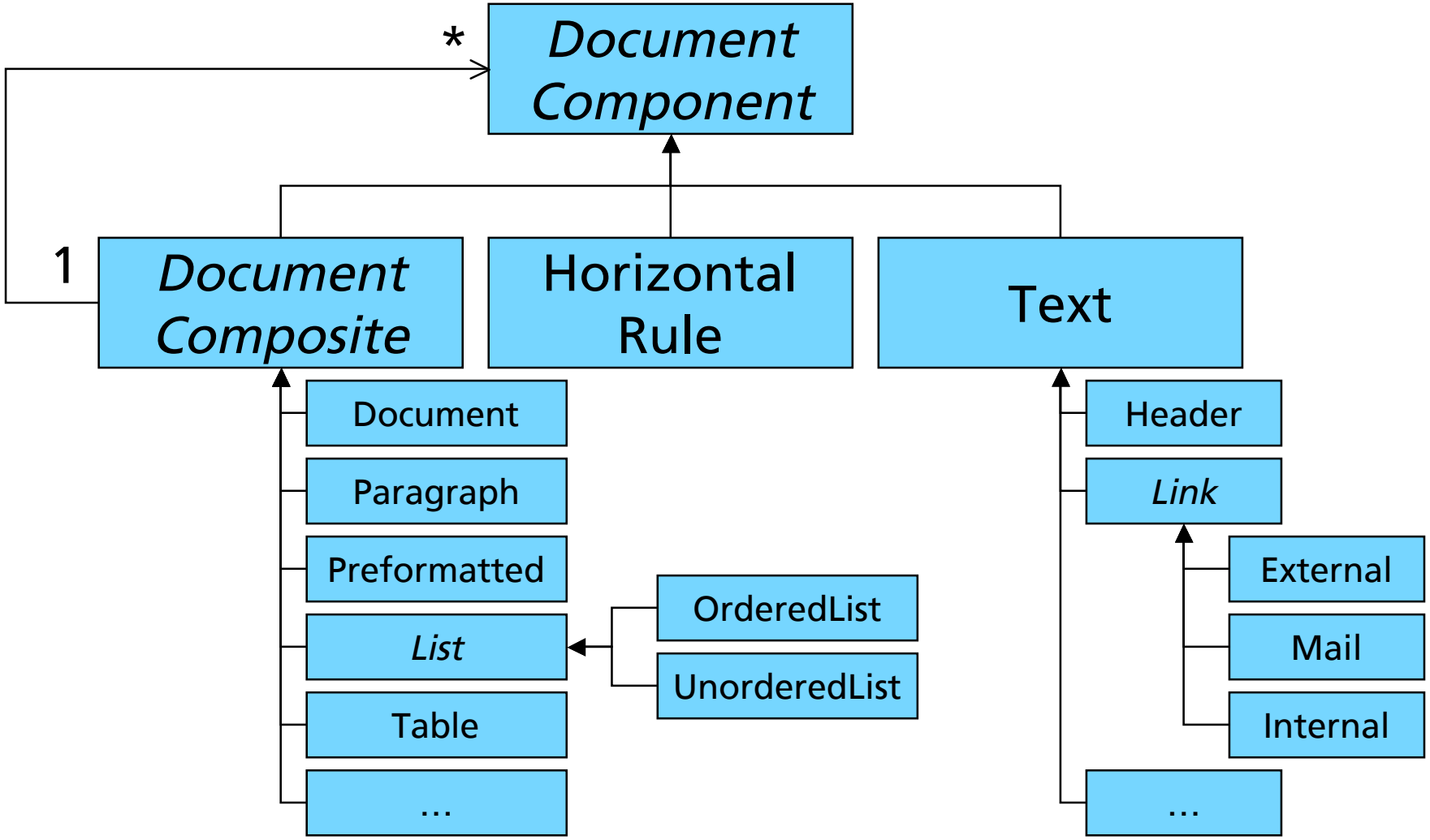
Client

Server

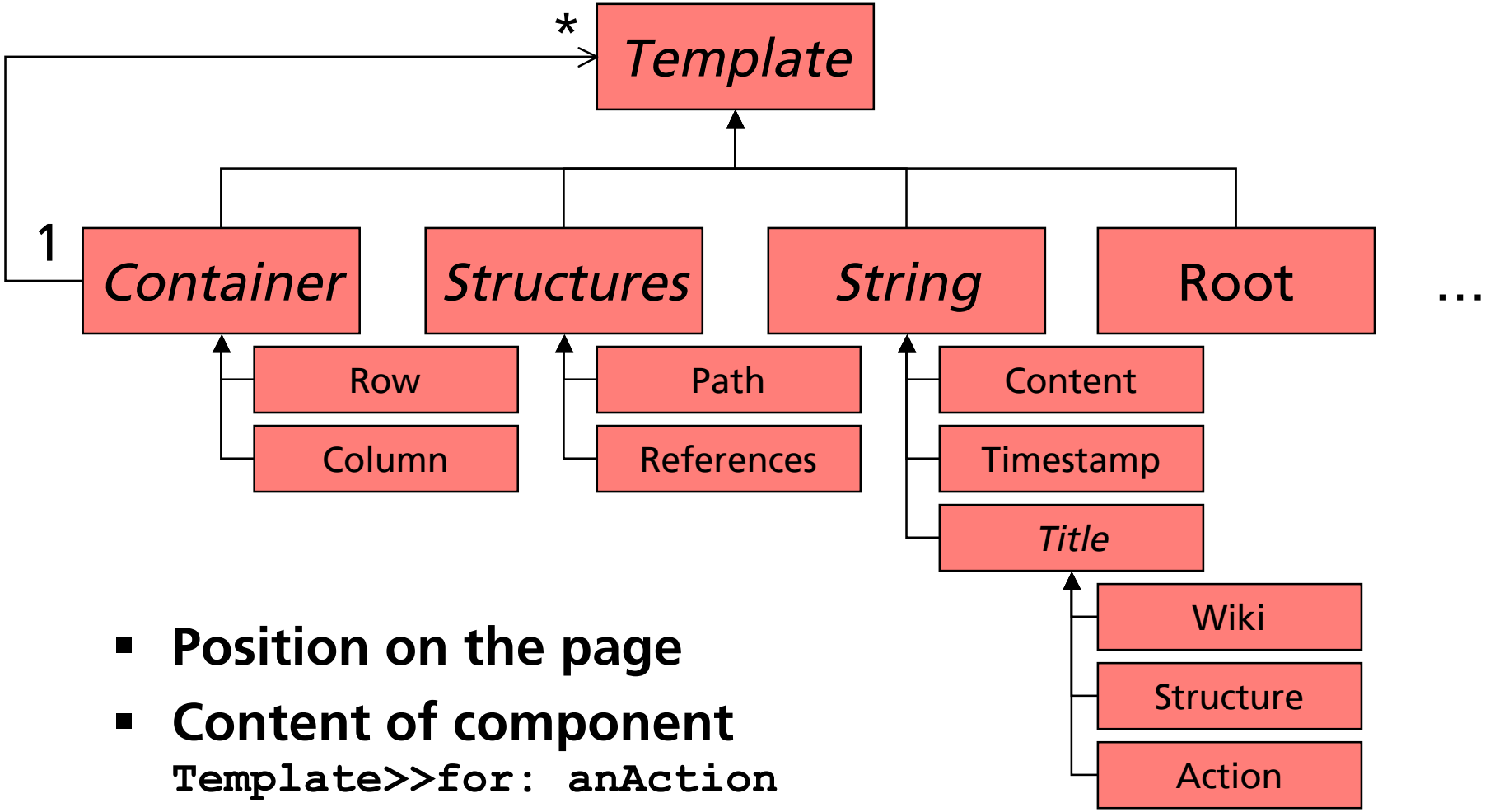
Web Browser



Document

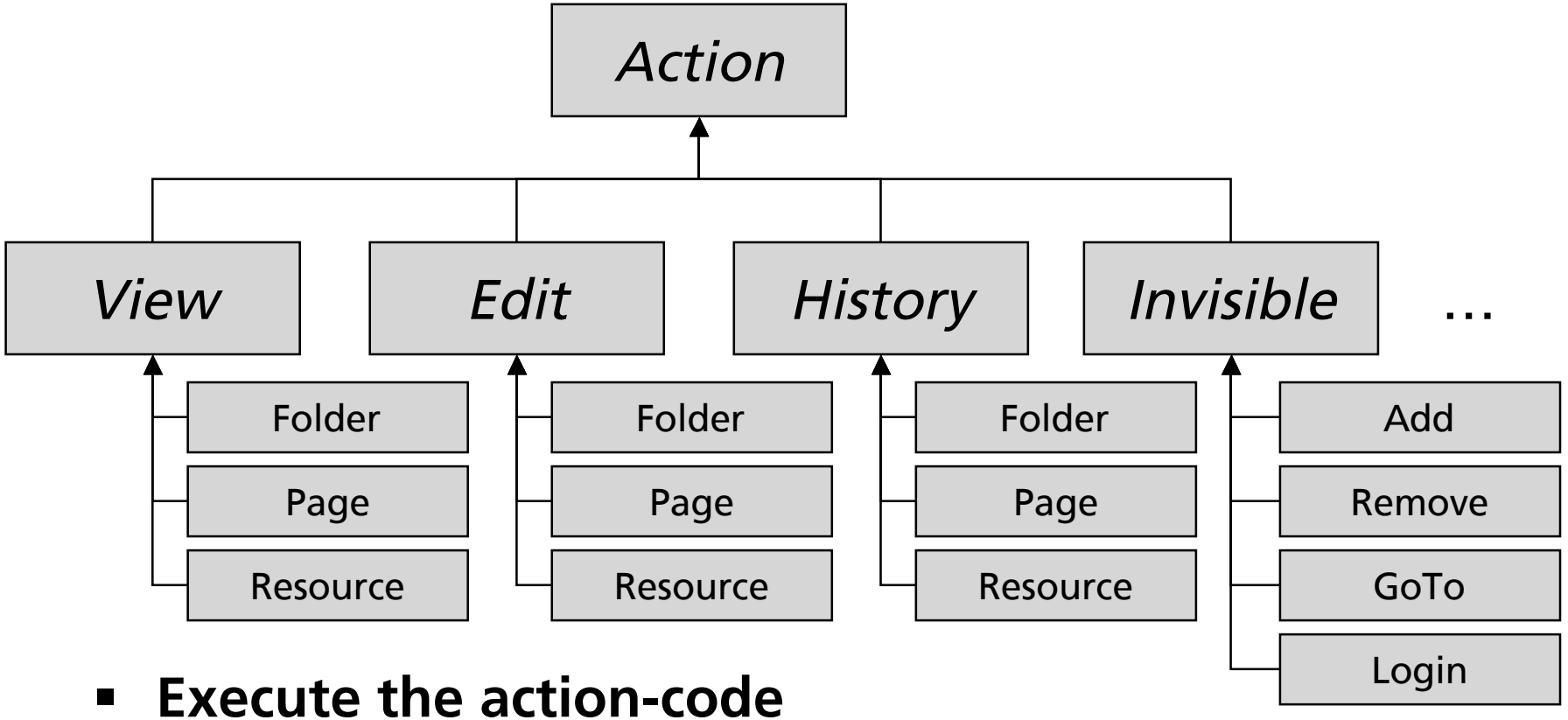

*
Document Component
1
Document Composite
Horizontal Rule
Text
Document
Paragraph
Preformatted
List
Table
...
OrderedList
UnorderedList
Header
Link
External
Mail
Internal
...

Template



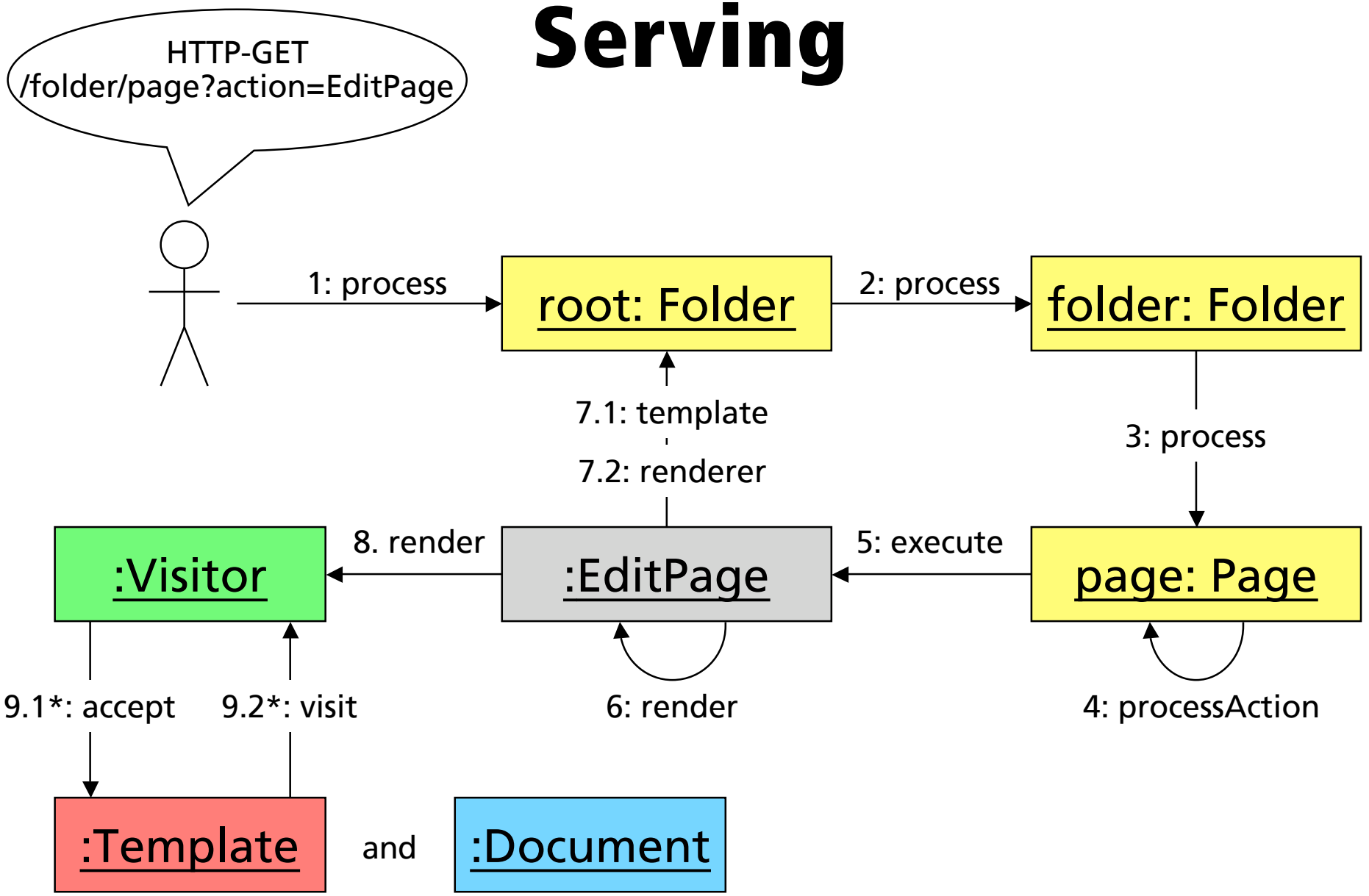
- **Position on the page**
- **Content of component**
  `Template>>for: anAction`

Actions

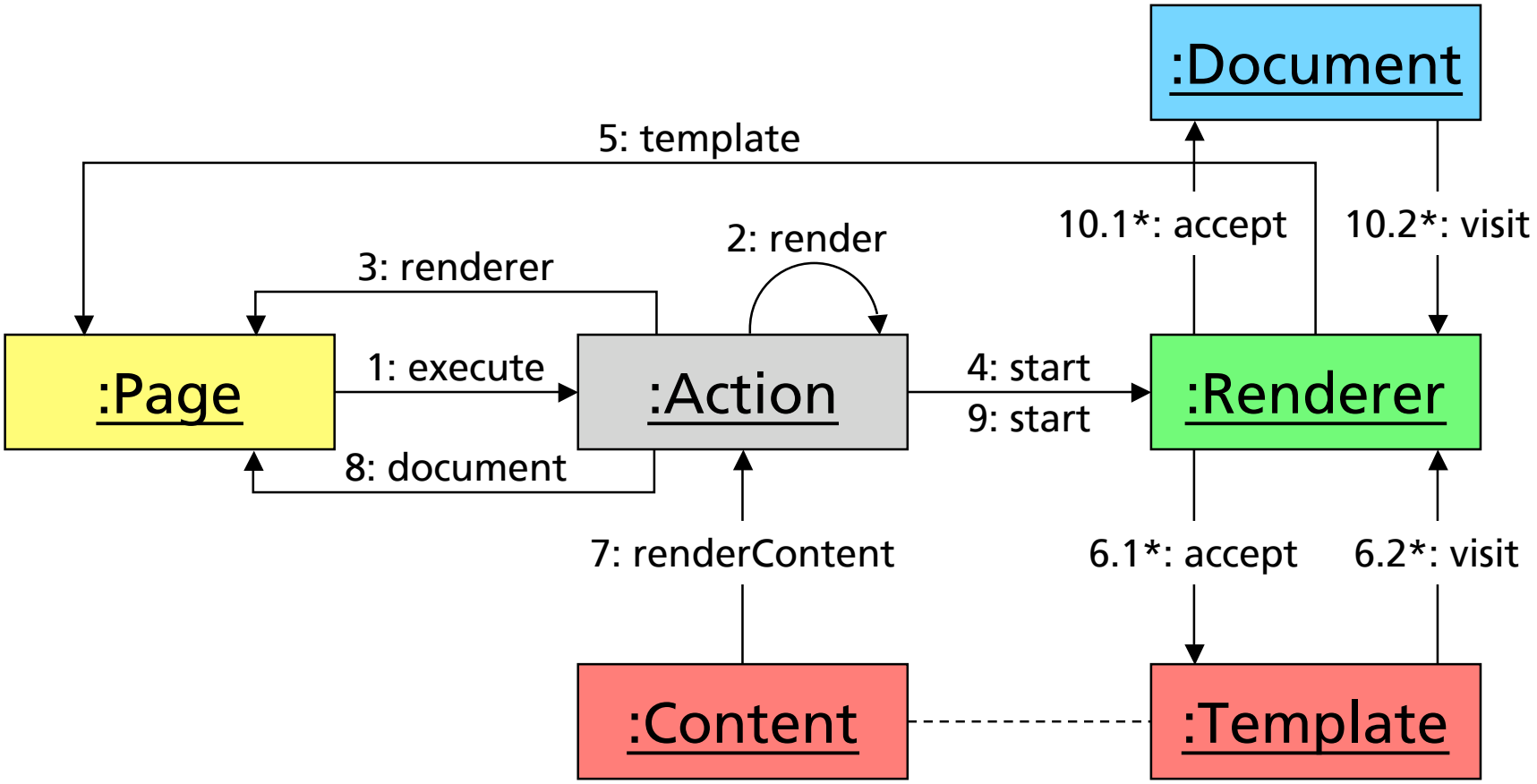


- **Execute the action-code**
  `Action>>execute`
- **Provide an user interface**
  `Action>>renderContent`

Serving


HTTP-GET
/folder/page?action=EditPage
1: process
root: Folder
2: process
folder: Folder
7.1: template
7.2: renderer
3: process
:Visitor
8. render
:EditPage
5: execute
page: Page
9.1*: accept
9.2*: visit
6: render
4: processAction
:Template
and
:Document

Rendering


:Document
5: template
10.1*: accept
10.2*: visit
2: render
3: renderer
:Page
1: execute
:Action
4: start
9: start
:Renderer
8: document
7: renderContent
6.1*: accept
6.2*: visit
:Content
:Template