XWiki's WYSIWYG Editor

Overview

XWiki's WYSIWYG Editor

• Written in Java using Google Web Toolkit
• Relies on browser support for rich text editing
• Uses the HTML DOM as its data model
• Uses XWiki's Rendering Engine to convert between HTML and wiki syntax
• Plugin-based

HTML DOM API

- Browsers parse HTML pages into DOM trees
- We can access the DOM of the current page in JavaScript

```
document.body.firstChild
```

- DOM node types common for HTML content are: Document, DocumentFragment, Element, Attribute, Text and Comment
- User actions trigger DOM events

```
element.addEventListener('keydown', keyDownHandler, false)
```

- We can serialize DOM back to HTML with innerHTML property

```
document.body.innerHTML
```

Rich Text Editing

- All modern browsers have built-in support for rich text editing

```
document.designMode = 'on'
element.contentEditable = true
```

- Making a document or an element editable doesn't affect the way we access its content, through DOM API

```
element.contentEditable = true;
element.firstChild.style.backgroundColor = 'red';
```

- Edited content can also be altered using a predefined set of commands

```
document.execCommand('bold', false, null)
document.queryCommandValue('fontsize')
```

- DOM events are still fired

Caret & Selection

- A DOM range models a contiguous HTML source fragment
    - start/end node, start/end offset
    - the start/end point is either inside a text/comment node or between DOM nodes

```
<em>ri|ch</em> tex|t
before|<img/>|after
```

- Caret = a collapsed range
- Selection = a set of ranges
    - most browsers support only one range per selection
- The caret and the selection are managed by the browser but we can fully controll them from JavaScript

DOM Mutations

- Atomic mutations

```
parent.appendChild(child);
parent.insertBefore(newChild, refChild);
parent.removeChild(child);
element.setAttribute(name, value);
element.removeAttribute(name);
textNode.data = 'new text';
```

- Composite (native) mutations

```
parent.replaceChild(insertedNode, replacedNode);
element.innerHTML = 'new inner HTML';
document.execCommand('bold', false, null);
range.deleteContents();
range.insertNode(newNode);
range.surroundContents(newNode);
```

Google Web Toolkit

- A development toolkit for building and optimizing complex browser-based applications
- The code is written in Java and compiled into JavaScript
- Most of the JavaScript API (e.g. DOM API) is made accessible in Java
- Easy to write browser-specific code (by subclassing and overriding the standard behavior)
- Provides a RichTextArea widget

```
iframe.contentWindow.document.desingMode = 'on'
```

- We have extended the GWT-DOM API with support for managing the selection

XWiki's WYSIWYG Editor

- Extends GWT's RichTextArea widget
  - Access the edited document and its selection/caret
  - Execute and query the predefined editing commands
  - Overwrite predefined commands and register new commands

- RichTextArea plugins
  - Full access to the RichTextArea
  - Provide a set of features (e.g. bold, removeformat) and user interface extensions (e.g. tool bar extension)
  - Use GWT-RPC services to communicate with the server side

Commands & Executables

- A command identifies a DOM transformation
  - make the selected text bold
- An executable is a sequence of DOM mutations

```
for each non-empty text node in the current selection
   if computed text style != bold
      split text node if selection starts/ends inside it
      wrap text node with <strong> element
      merge adjacent <strong> elements
```

- Commands are implemented through executables
- CommandManager
  - Map executables to commands
  - Execute/Query commands
  - Notify command listeners

Editor State

- Only the most recent update is executed
- On [MouseUp, KeyUp, Command]
  - Cancel pending update
  - Schedule update

- Context = (content, selection)

- On update
  - Editor enables/disables features based on the current context and the back-end syntax
  - Plugins update the state of their UI extensions (e.g. toggle on/off bold)

Editor Plugins

```
// init(rta:RichTextArea)
CommandManager cm = rta.getCommandManager();
cm.registerCommand(Command.BOLD, new BoldExecutable(rta));
// Register text area event handlers.


// onClick(event:ClickEvent)
cm.execute(Command.BOLD);


// onUpdate()
toggleButton.setDown(cm.isExecuted(Command.BOLD));


// destroy()
// Unregister text area event handlers.
```

XWiki's Rendering Engine

- Converts wiki syntax to HTML: WYSIWYG input
- Converts HTML to wiki syntax: WYSIWYG output

- Annotated XHTML
  - HTML comments used to restore the wiki syntax

```
<!--startimage:Space.Page@attachment.png-->
<img/><!--stopimage-->
```

  - Editor plugins move this meta data in attributes to protect it while editing

```
<img metadata="<!--startimage:Space.[...]" />
```

- Ability to restrict the set of editor features available for a specific wiki syntax