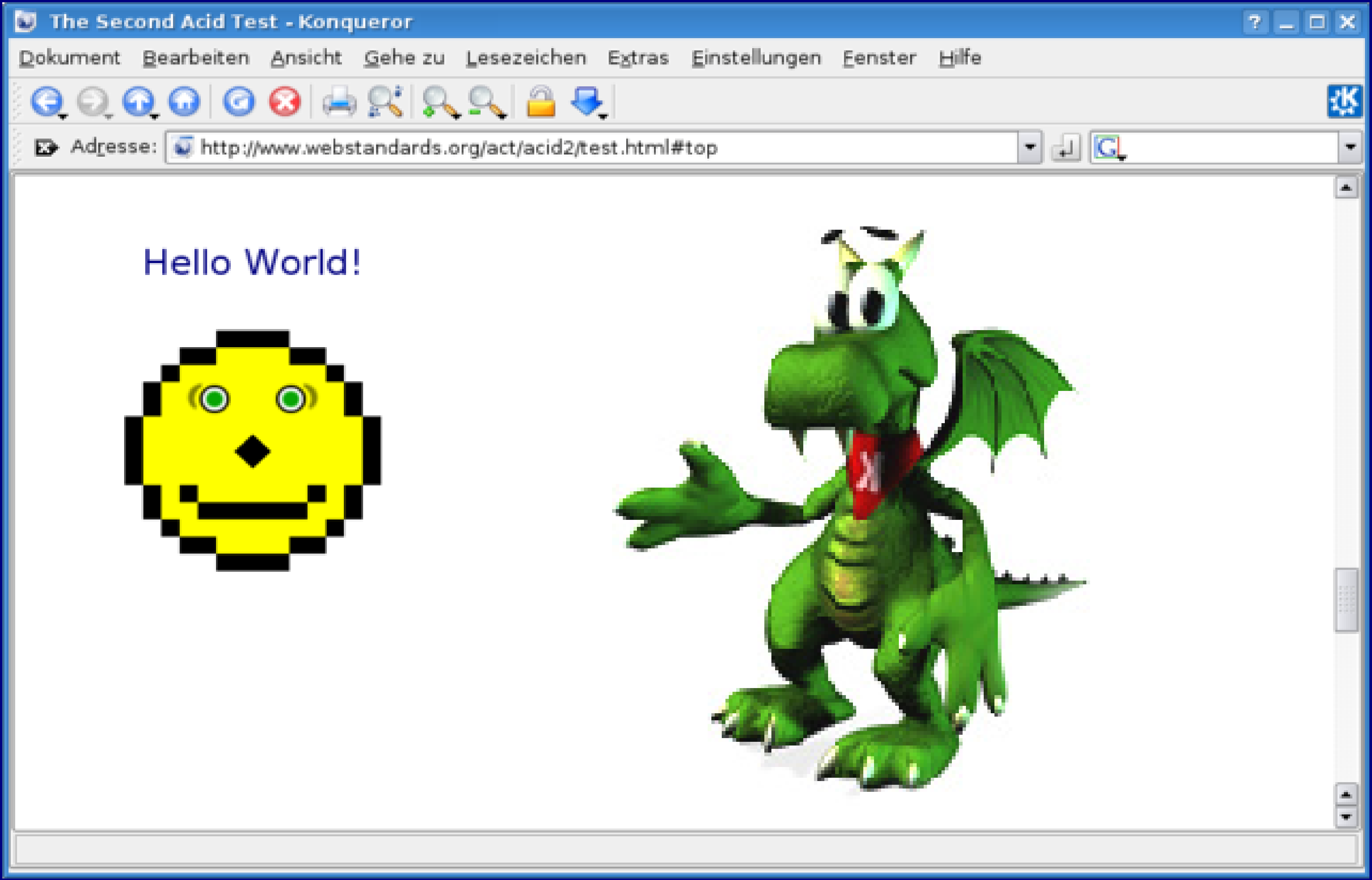
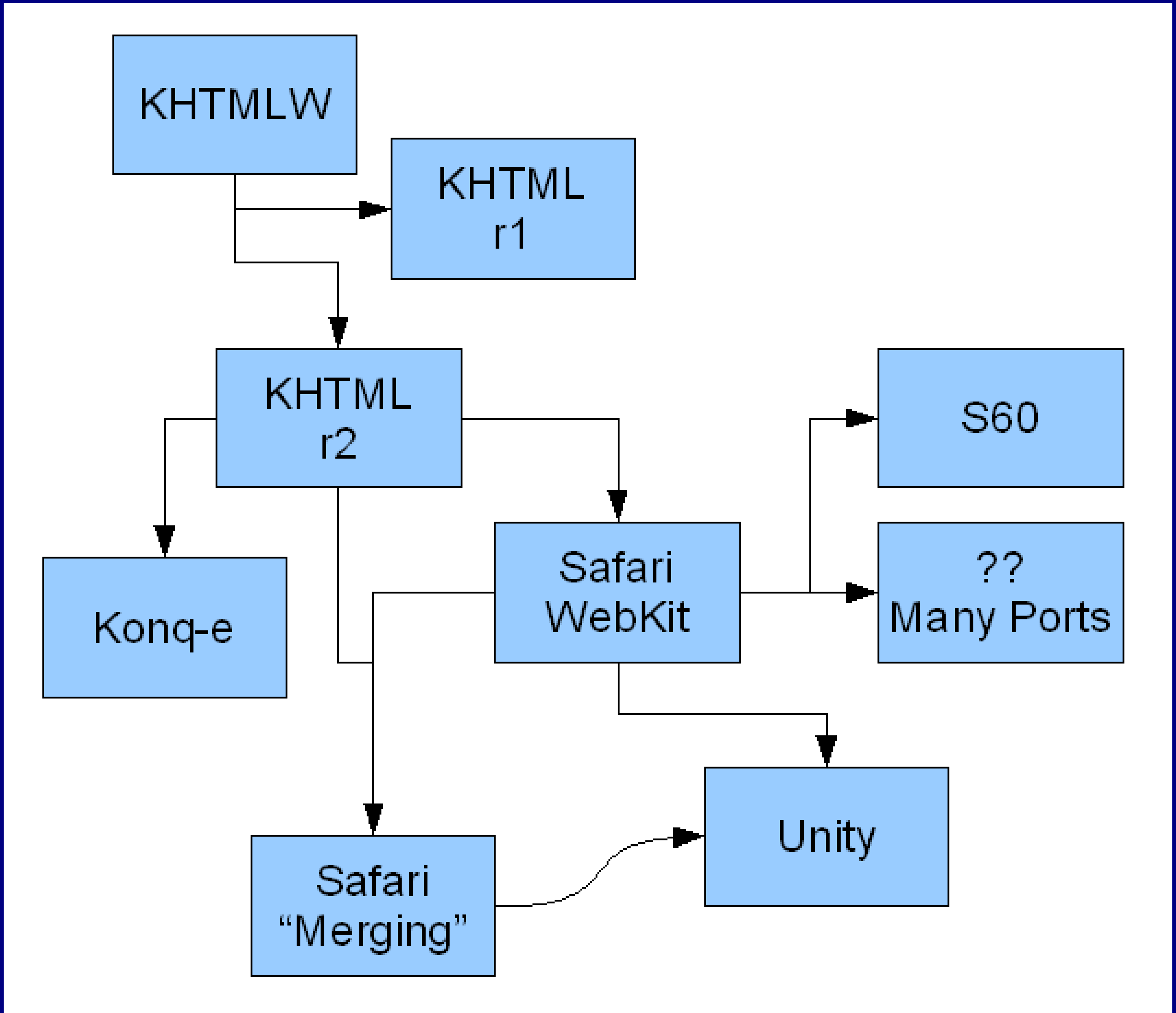
The State of KHTML



The Second Acid Test - Konqueror
Dokument Bearbeiten Ansicht Gehe zu Lesezeichen Extras Einstellungen Fenster Hilfe
Adresse: http://www.webstandards.org/act/acid2/test.html#top
Hello World!

In The Beginning...



Why KHTML Is Important

- KHTML is *critical* to the success of KDE
- Provides a fast, light web browser and component that tightly integrates with KDE
- Gives us higher visibility as a project: “the web” is much larger than the Linux desktop community
- It's a great testcase for our existing infrastructure

What We Have

- CSS 1 / 2 parts of 3 (modular)
- ECMAScript 1.2
- XHTML
- HTML up to 4.x
- “AJAX” support
- NS plugin support
- Java applets
- SSL
- Wallet
- KDE integration
- Basic LiveConnect
- KJSEmbed derived from KJS

What We Don't Have

- Internal SVG support
- Latest NSPlugin API support
- XBL
- Content Editable
- DOM bindings to non-C++/non-JS
- Opacity (Qt4 should help)
- Lightweight widgets

KHTML – From Industry

- Great alternative to Gecko and Opera – small, fast, easy to understand, standards compliant
- In use in many embedded platforms as well as desktop browsers
  - Safari, Nokia, Omni, etc
- Forking is a problem
- Gaining respect from other browser vendors
- Could become a major player with enough 'unity' - >= 10% market share

Can We Complete The “Merge”?

- “Merging” is not really feasible at this point
- Unity – a port of WebKit to Qt4:
  - KPart, KIO development is underway
  - Ca n co-exist with KHTML from KDE
  - Works 'now'
  - Abstraction layer in WebKit makes it relatively easy to port
  - Open questions: How do we avoid re-forking?
How do we merge our own work?

If We Go Unity, What Do We Gain?

- Support for many of the functions we lack as described earlier – XBL, content editable, etc
- Bug-for-bug compatibility with many major browsers
  - This is important for industry acceptance
- More developer resources
- Larger user base for testing and bug reporting
- Easier embedding in Qt-only apps
- Portability – the best Win32 port?

What Do We Lose?

- Some possible trade-offs in bug fixes and performance
  - Esp: Work by Maksim, Germain, Allan
- Some flexibility in development model
  - We need to work as a team with Nokia, Apple, etc
- Complete authority over the codebase
- Some functionality needs rewrite:
  - Form widgets, Java applets, nsplugins, wallet, KDE integration

Working With Working Groups

- W3C – Security
- W3C – General
- WhatWG – HTML5
- ECMA – JS
- CA-Browser Forum
- KHTML/WebKit meetings
- Plugin Futures
- SVG
- Browser testing organization
- JavaScript as a standard programming language

Discussion

- Do we want to pursue WebKit/Unity?
  - If so, how do we approach it?
  - What will the impact be on our community?
- How do we avoid losing our own work?
- How do we ensure that we are equal players in a joint effort with Apple, Nokia and others?
- How can we grow developer-interest without cannibalizing our existing developer base?

George Staikos <staikos@kde.org>

aKademy Sep 23, 2006