

# Software Standards

... following standards, independent developments and productions can eventually be used in cooperation for the benefit of users

# Free and open standards

- How conflicting standards arise
- What makes a standard open: open source perspective
- What makes a standard open: EU definition
- Why insist on open standards?
- Standardization bodies
- Document standards
- Web standards
- Programming standards
- Concerning Java

# How conflicting standards arise

- Political/institutional boundary.
  - Familiar example: power sockets, roads and cars.
  - Software example: character encoding, object code.
- Vendor specific.
  - Familiar example: mobile phone batteries, video tapes.
  - Software example: document format.
- Tenacity of old standards.
  - HTML vs XHTML.
  - IPv4 vs IPv6.

# What makes a standard open: open source perspective

- Available for all to read and implement, preferably with a free reference implementation.
- Allow a wide range of implementations. Thus, end users are not locked to a particular vendor.
- No royalty or fee.
- Low cost non-discriminatory certification or validation.
- Allow uncertified extensions or subsets.
- May require extensions to be similarly open to protect against embrace-and-extend tactics.

# What makes a standard open: EU definition

- Developed openly and democratically by a non-profit organization.
- Specification available at a nominal charge.
- No royalty.
- No constraints on the re-use of the standard.

# Why insist on open standards?

- Allow fair competition based on technical merit.
- Allow interoperability.
- Allow free and open source implementation.
- Prevent forced product selection.
- Prevent vendor lock-in.
- Prevent expensive forced “upgrade”.

# Standardization bodies

- International Organization for Standardization (ISO).
  - Various areas. e.g., ISO 9660 (CD image), OpenDocument
- Internet Engineering Task Force (IETF).
  - Internet. e.g., TCP/IP, HTTP, SSH, POP.
- World Wide Web Consortium (W3C)
  - Web related. e.g., XML, HTTP, XHTML, CSS.
- Organization for the Advancement of Structured Information Standards (OASIS).
  - E-business and web service. e.g., OpenDocument.

# Document standards

- Non-structured: plain text.
- Formatted: DVI, PS, PDF.
- Structured: LATEX, XML, DocBook.
- Graphics: PNG, SVG, EPS.
- Office: OpenDocument.
- Problem: font embedding.



# Web standards

- Contents: HTML (deprecated), XML, XHTML, MathML, SVG.
- Graphics: PNG, SVG.
- Style: CSS, XML-based?
- Client-side script: ECMAScript.
- Offensive notice: Best viewed with xxx.
- Best avoided: objects requiring plug-ins. If used, provide alternative contents.

# Programming standards

- OS interface: POSIX, LSB.
- Graphics API: GTK+, Qt, OpenGL, Swing (?).
- Persistent variables: dot files, reg/conf file.
- Collaborative: CVS.
- Distribution: tarball, deb, rpm, jar, ruby gem.

# Concerning Java

- Language standardization is controlled by Sun.
- Public may participate and contribute.
- Language spec is available free of charge.
- Implementation is available free of charge.
- Source to the “standard” library is available free of charge.
- There is no formal spec for the “standard” library.
- Java is very popular.
- Efforts to free Java: gcj, classpaths.
- Alternatives to Swing: Java-Gnome, SWT.