Guideline 5. Ensure User control of user interface behavior

- Opening and closing of view ports, focus changes and form submissions.

➢**Checkpoint definitions.**

•No automatic content focus change(P2)

✓Allow configuration so that if a view port opens without explicit user request, neither its content focus nor its user interface focus automatically becomes the current focus.

•Keep view ports on top(P2)

✓For GUI, allow configuration so that the view port with the current focus remains on top of all other view ports with which it overlaps

- **Manual viewport open only(P2)**

✓ View ports should only open on user's request

✓ Instead of opening a view port automatically, alert the user and allow the user to open it with an explicit request (e.g. , by confirming a prompt)

✓ Allow user to close view ports (view port → frameset)

- **Selection and focus in view port(P2)**

✓ Changes in viewport's content focus should at least partially be in the viewport after the change

->For example, if users navigating links move to a portion of the document outside a viewport, the viewport should scroll to include the new location of the focus

- **Confirm form submission(P2)**

✓ User should be prompted to confirm any form submission

->Submission when an onmouseover event occurs.

-> In HTML 4 , form submit controls are the "INPUT" element with type="submit".

Guideline 6. Implement interoperable API

This section addresses interoperability between a user agent and other software, in particular assistive technologies.

Before implementation:

- What information must be communicated through an API?
- Which APIs must be used to communicate this information? (conventional or custom API)
- Additional characteristics of these APIs

➢ Checkpoint definitions

- **Programmatic access to HTML/XML infoset(P1)**

✓ Provide read access to XML / HTML content by conforming to the W3C XML infoset

✓ If the user can modify HTML / XML content through the user interface (e.g., by checking a box), provide the same functionality programmatically as is available through the user interface.

- **Programmatic access to non-HTML/XML content (P1)**

✓ For non- HTML/XML, provide programmatic read access to content.

✓ If the user can modify a non- HTML/ XML content through the user interface (e.g., by checking a box), provide the same functionality programmatically as is available through the user interface.

- **Programmatic access to information about rendered content (P1)**

✓ For each rendered text, provide access to font family, font size, foreground and background colors.

-> This provision refers to what is *actually rendered* on the screen

- **Programmatic operation of user agent user interface (P1)**

✓ Provide programmatic read / write access to user agent user interface using standard API’s.

->Standard APIs (e.g., platform-independent APIs such as the W3C DOM; standard APIs defined for a specific operating system; and, plug-ins etc.)

✓ If the user can modify user agent user interface control (e.g., by checking a box), provide the same functionality (read/write access) programmatically as is available through the user interface.

- **Programmatic notification of changes (P1)**

✓ Provide programmatic alert through standard APIs

-> When an action in one frame causes the content of another frame to change

- **Conventional keyboard APIs (P1)**

✓ Implement APIs for the keyboard as follows:

- Respect operating environment conventions

-> An operating environment may define more than one standard API for the keyboard. For instance, for Japanese and Chinese, input may be processed in two stages, with an API for each.

- **API character encodings (P1)**

✓ Support character encodings

-> So that text is not "broken"

Guide line 7. Observe operating environment conventions

➢Checkpoint definitions

- **Respect input configuration conventions (P1)**

✓ Respect operating environment accessibility conventions (e.g., for keyboard accessibility).

->The default input configuration user agent should not include "Alt-F4", "Control-Alt-Delete", or other combinations that have reserved meanings in a given operating environment.

- **Respect operating environment conventions(P2)**

✓ Respect operating environment conventions that benefit accessibility .

-> Conventions include mouse keys, keyboard configuration etc.

- **Provide input configuration indications(P2)**

✓ Indicate the current input configuration

-> e.g., indicate what key strokes will activate a function

-> alert the user if the current input configuration changes
e.g., changing the keyboard bindings

Guide line 8. Implement specifications that benefit accessibility

Implement W3C recommendation when available.

➢ **Checkpoint definitions**

- **Implement accessibility features(P1)**

✓ Implement the accessibility features of specifications

-> e. g , mark-up languages , style sheets languages, etc.

-> Applies to both W3C and non-W3C specifications.

- **Conform to specifications(P2)**

✓ Conform to W3C Recommendations that conforms to Web Content Accessibility Guideline (WCAG)

->For instance, for markup languages, the user agent may conform to HTML 4 *[HTML4]*, XHTML 1.0 *[XHTML10]*, and/or XML 1.0 *[XML]*

Guide line 9. Provide navigation mechanisms

Provide access to content through a variety of navigation mechanisms:

- Sequential navigation (e.g., line scrolling, page scrolling etc)
  Forward and Reverse
- Direct navigation (to a particular link or paragraph)

Faster than sequential navigation, but generally requires familiarity with the content.

Direct navigation may be possible with the keyboard (e.g., keyboard shortcuts).

- Structured navigation

Navigate structural content : headers, tables, forms, etc.

➢ Checkpoints

- **Provide content focus(P1)**

✓ At least one content focus for each viewport

✓ Content Focus of each viewport is the current focus

->When two frames of a frameset contain enabled elements, allow the user to make the content focus of both frames the current focus

- **Provide user interface focus(P1)**

✓ Provide a User Interface Focus

->For users who rely on the user interface focus for interaction e.g., for interaction with user interface through the keyboard

- **Move content focus(P1)**

✓ Move the content focus to any enabled element in the viewport.

✓ Content focus only changes on explicit user request.

✓ If navigation order is not specified, allow at least forward and reverse sequential navigation

- **Restore viewport state history (P1)**

✓ For each state in a viewport's browsing history, maintain information about the point of regard, content focus, and selection

✓ When the user returns to any state in the viewport history (e.g., via the "back button"), restore the saved values for the point of regard, content focus, and selection.

-> ex: In JavaScript, the following may be used to change the Web resource in the viewport, and navigate the history:

```
myWindow.history.back();
myWindow.history.forward();
```

- **No events on focus change (P2)**

✓ Do not automatically activate event handlers of any event type

-> Do not activate any handlers for the onmouseover attributes in HTML docs

- **Show events handlers(P2)**

✓ Show event types for which there are event handlers

->For HTML, event handlers associated with mouse clicks are activated with the left mouse button

- **Move content focus in reverse(P2)**

✓ Allow the same sequential navigation in reverse document order.

- **Provide text search(P2)**

✓ Allow the user to search within rendered text content

✓ When there is a match,

  -matched text content should be within viewport

  -search for the next instance of the text from the location of the match

✓ Alert the user when there is no match or after the last match in content

✓ case-insensitive search option for text

- **Allow structured navigation(P2)**

✓ Allow navigation to important structural elements in rendered content.

->This specification intentionally does not identify which "important elements“ must be navigable as this will vary by specification.

->Important elements in html (APPLET, DIV, FORM, FRAME, H1-H6, IMG

etc.)

Configure important elements(P3)

✓ Allow configuration of important elements

-> ex, allow the user to navigate only paragraphs, or only headings

Guide line 10. Orient the user

Clues to help users understand their location when browsing: where they are, how they got there, where they can go, what's nearby, etc

➢ Checkpoints

- **Associate table cells and headers (P1)**

✓ For each table cell, allow the user to view associated header information.

->A cell may be associated with more than one header.

- **Highlight selection ,content focus ,and visited links (P1)**

✓ Show highlight mechanism for selection ,content focus, and recently visited links in each viewport

Highlight mechanism for

Selection and Content focus : Foreground and background color

Recently visited links : Visited links on a page are highlighted so that users know at a glance where they interacted.

- **Single highlight configuration (P2)**

✓ Allow configuration thru a single setting (ex.: Using setting of user agent)

-> Avoid conflicts between multiple highlight settings (e.g., same insufficient contrast).

- **Provide outline view (P2)**

✓ Outline view of rendered content, composed of labels .

HTML labels include

-The CAPTION element is a label for TABLE

- The H1-H6 elements are labels for sections that follow

- The TITLE element is a label for the document. etc.

- **Provide link information (P3)**

✓ To help the user decide whether to traverse a link in content, make available the following information about it:

-link element content,

-link title,

-Is link internal or external

-If link has been traversed

- Size, and natural language of linked Web resources.

- **Highlight current viewport (P1)**

✓ Highlight the viewport with the current focus

->A window that is the viewport with the current focus is brought to the foreground, or maximized automatically.

- **Indicate viewport position (P3)**

✓ Indicate the viewport's position relative to rendered content

-> (e.g., the proportion of an audio or video clip that has been played, or the proportion of a Web page that has been viewed).

->The user agent may indicate the proportion of content viewed in a number of ways, including as a percentage or as a relative size in bytes

->The proportion should be indicated using a relative value where applicable (e.g., 25%), otherwise as an absolute offset (e.g., 3k) from some recognized landmark.

Guide line 11. Allow configuration and customization

Allow users to configure the user agent so that frequently performed tasks are made convenient, and allow users to save their preferences

➢Checkpoints

- **Current user input configuration (P1)**

✓ Provide information to the user about current user preferences (see keyboard bindings)

- **Current default input configuration (P2)**

✓ Provide a centralized view of the current default input configuration.

->Provide different views for different input modalities (keyboard and pointing device).

->For HTML, provide a view of default keyboard bindings .

- **Allow override of bindings (P2)**

✓ Allow the user to override any binding that is part of the user agent default input configuration.

->The user agent is not required to allow the user to override conventional bindings for the operating environment (e.g., for access to help).

- **Single-key access (P2)**

✓ Override any binding in the user agent default keyboard configuration with a binding to either a key plus modifier keys or to a single key.

-> Offer a single-key mode where, once the user has entered into that mode (e.g., by pressing a single key), most of the keys of the keyboard are configurable for single-key operation of the user agent. Allow the user to exit that mode by pressing a single key as well

- # Default input configuration (P2)

✓ Input configuration should support:

->Search for text, search again for same text

->Increase and decrease the scale of rendered text

->Increase global volume, and decrease global volume

->Stop, continue audio and animations

->Next history state (forward), and previous history state (back);

->Enter a URI for a new resource

->Add a URI to favorites (i.e., bookmarked resources);

->View favorites;

- **User profiles (P2)**

✓ Allow the user to save user preferences in at least one user profile.

✓ Allow the user to choose from available user profiles created by the same user

Tool bar configuration (P3)

✓ Configure tool bars.

✓ Restore the default tool bar configuration.

-> Show and hide user interface controls

-> Choose icons.

Guide line 12.Provide accessible U.A. documentation and help

Ensure that the user can learn about software features from the documentation. Without documentation, users with disabilities may have a very difficult time knowing what the user agent is capable of and how to operate it.

➢Checkpoints

- **Provide accessible documentation (P1)**

✓ Documentation must conform to Web Content Accessibility Guidelines 1.0 *[WCAG10]*.

✓ Keep documentation accessible as the user agent evolves

->Distribute documentation over the Web, on CD-ROM

->Conformance to the Web Content Accessibility Guidelines 1.0 *[WCAG10]*:

-Use screen-shots and flow charts;

-Use search mechanisms;

- **Provide documentation of accessibility features (P1)**

✓ Provide documentation of all user agent features that benefit accessibility.

->A centralized view is sufficient

->ex. Document configurable and default features.

• Provide documentation of default bindings (P1)

✓ Provide documentation of the default user agent input configuration

-> (e.g., the default keyboard bindings).

->Without this documentation, a user may try to use a much less efficient technique to perform a task, such as using a mouse

- **Provide documentation of changes between versions (P2)**

✓ Provide documentation of changes since the previous version of the user agent to features that benefit accessibility.

->In particular, document changes to the user interface.

->Provide a text description of changes (e.g., in a README file).

- **Provide dedicated accessibility section (P2)**

✓ Provide a centralized view of all features of the user agent that benefit accessibility, in a dedicated section of the documentation.

-> Ensure that the section on accessibility features is easy to find.

Bibliography

- http://www.w3.org/TR/UAAG10/
- http://www.w3.org/TR/2002/NOTE-UAAG10-TECHS-20021217/