

Accessibilité et interfaces riches



Renan Michaud,
<renan.michaud@softeam.fr>

Plan

- Qu'est ce que l'accessibilité?

Définition, recommandations, législations et utilisation

- Les interfaces riches

WCAG 2.0, Flex, AJAX et ARIA

- Application pour les projets

Hijax, frameworks d'interfaces

Définition

The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect.

The social value of the Web is that it enables human communication, commerce, and opportunities to share knowledge. One of W3C's primary goals is to make these benefits available to all people, whatever their hardware, software, network infrastructure, native language, culture, geographical location, or physical or mental ability.

Tim Berners-Lee, directeur du W3C et inventeur du WWW

Qui est concerné?

- Les handicaps
- Le vieillissement: polyhandicap visuel, auditif, moteur et cognitif
- La maîtrise de la langue et de l'orthographe, l'illettrisme

Recommandations

Portées par le WAI (Web Accessibility Initiative), lancée en 1997.

- **WCAG (Web Content Accesibility Guidelines)**

Accessibilité des contenus web. WCAG 1.0 en 1999 et 2.0 le 11 décembre 2008.

- **UAAG (User Agent Accessibility Guidelines)**

Accessibilité des agents utilisateur (navigateurs). Version 1.0 publiée le 17 décembre 2002.

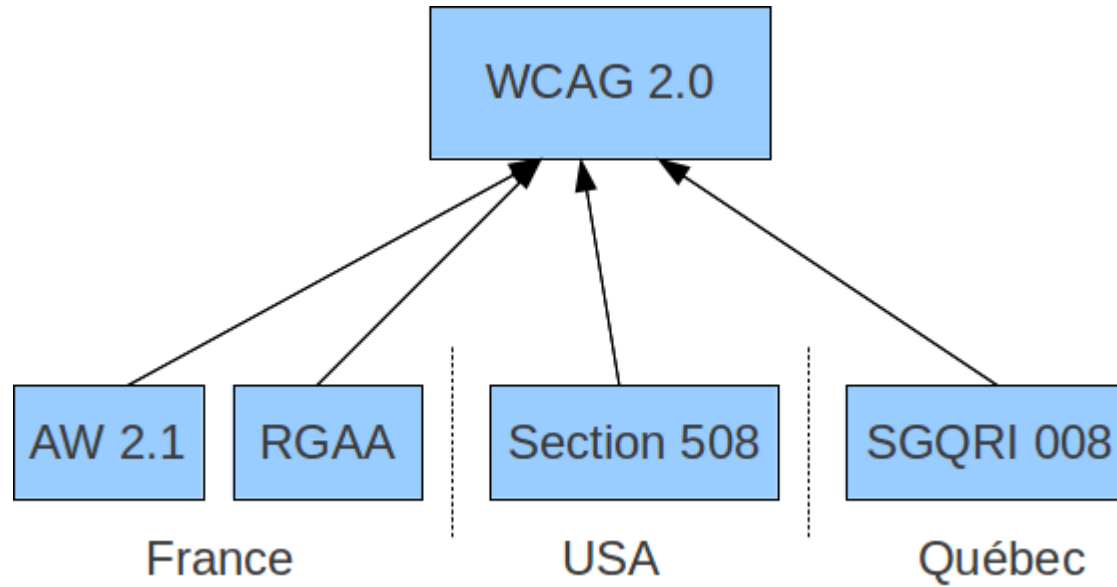
- **ATAG (Authoring Tool Accessibility Guidelines)**

Accessibilité des logiciels de créations de contenus web. Version 1.0 publiée le 3 février 2000.
Certains critères peuvent être appliqués aux CMS.

WAI-ARIA

- Accessible Rich Internet Application
- informations sémantiques
- informations de structure
- comportement pour les différents widgets créé avec AJAX
- Candidate Recommendation depuis le 18 janvier 2011

Applications des recommandations



Législation

Loi n° 2005-102 du 11 février 2005

- Pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées.
- Article 47 : obligation d'accessibilité pour les services de communication publique en ligne.

Décret n° 2009-546 du 14 mai 2009

- Le référentiel précisant les règles d'accessibilité doit être défini par arrêté.
- Délai d'adaptation de trois ans pour les collectivités territoriales et les établissements publics et de deux ans pour les services de l'État.

Arrêté du 21 octobre 2009

- Publication officielle du Référentiel Général d'Accessibilité des Administrations dit RGAA.

WCAG 2.0

- 4 principes
 - *Perceptible*
 - *Utilisable*
 - *Compréhensible*
 - *Robuste*
- Du point de vue WCAG 2.0: pas de technologie spécifique

Règle 1.1 Les équivalents textuels : proposer des équivalents textuels à tout contenu non textuel qui pourra alors être présenté sous d'autres formes selon les besoins de l'utilisateur : grands caractères, braille, synthèse vocale, symboles ou langage simplifié.

- Les référentiels sont des méthodes d'applications des WCAG 2.0

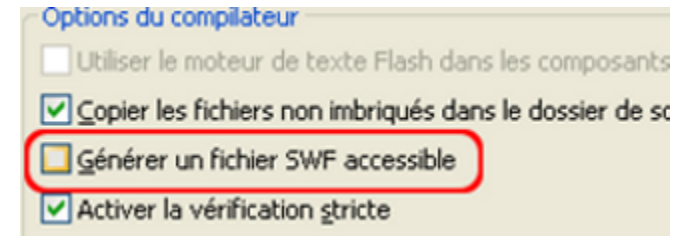
Flex/Flash et l'accessibilité

Possible !

- fournir les alternatives textuelles
- gérer la navigation au clavier (ordre de tabulation et gestion du focus)
- gérer facilement des zoom texte (vectoriel)

Complet?

- Peu de structuration
- Doit être compilé explicitement en mode accessible



Voir [VPAT Adobe](#) pour la conformité à la Section 508

Accessibilité et AJAX: rappels

AJAX

- Mise à jour partielle de la page
- Composants d'interfaces complexes

ARIA

- Supporter les mises à jour partielles de la page
- Ajouter de la sémantique et de la structure
- Étendre la navigation au clavier
- Étendre les possibilités de contrôle de formulaire (association étiquette/champ, champs requis, plages de valeur...)



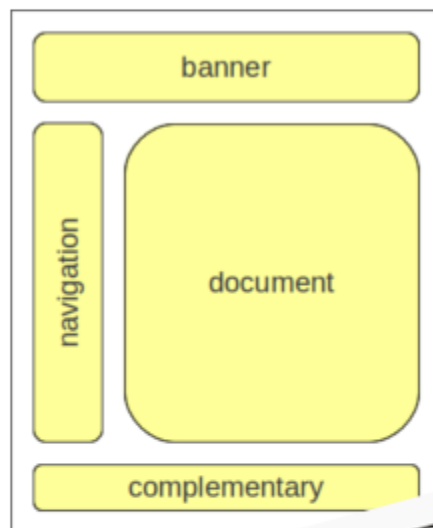
Et ça marche?

■ ARIA en Candidate recommandation

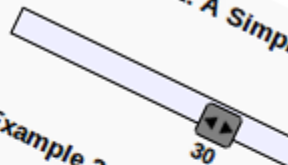
- *Firefox 1.5 and Firefox 2.0 partially supports WAI-ARIA, however it requires the use of namespaces, and doesn't support the use of Liveregions.*
- *Firefox 3+ contains better support for WAI-ARIA, including Liveregions.*
- *IE8 partially supports WAI-ARIA.*
- *JAWS 8 and Window-Eyes 5.5+ partially support WAI-ARIA.*
- *Jaws 10+ supports WAI-ARIA.*
- *FireVox, a self-voicing extension to Firefox, also supports WAI-ARIA via direct DOM access.*
- *NVDA partially supports WAI-ARIA.*

ARIA Techniques for WCAG 2.0

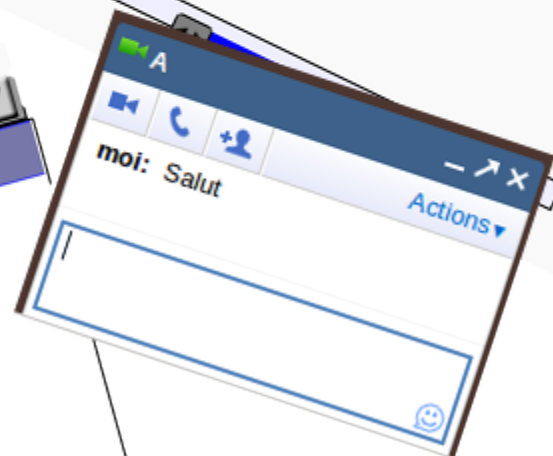
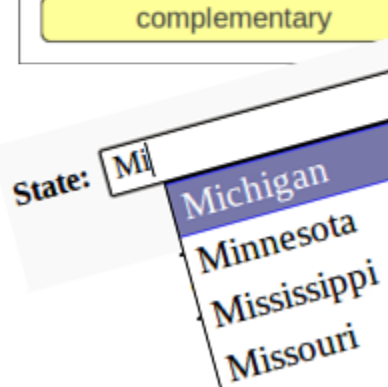
Focus sur ARIA



Slider Example 1: A Simple Slider



Slider Example 2: A Range Slider



Les régions du documents

Possibilité de découper le document par régions/landmark

- **navigation** définit une zone contenant les liens de navigation interne et externe
- **application** définit une zone d'application web (vs **document**)
- **banner** définit la zone d'entête de la page
- **complementary** définit la zone "annexe", type pied de page
- **search** définit la zone "moteur de recherche"
- **form** définit une zone de type formulaire
- ...

Les live-regions

- **aria-live**: [off, polite, assertive]
- **aria-atomic**: décrire toute la zone ou non?
- **aria-relevant**: [additions, removals, text, all]
- ...

```
<div class="example">
  <label id="region2Label" for="liveregion2">Log Text</label>
  <div id="liveregion2"
    class="region"
    role="log"
    aria-labelledby="region2Label"
    aria-live="polite"
    aria-atomic="false"
    aria-relevant="additions">
  </div>
</div>
```

La navigation au clavier

- Composants des widgets ne peuvent prendre le focus au clavier

```
<div id="divEx" tabindex="0" style="border:1px solid red;">  
  <br />  
  <p id="pEx" style="display:none">J'ai le focus</p>  
</div>
```

```
[...]
```

```
<script type="text/javascript" >  
window.onload=function (){  
  if(window.addEventListener){  
    document.getElementById("divEx").addEventListener(  
      "focus", exempleFocus, false);  
    document.getElementById("divEx").addEventListener(  
      "blur",exempleBlur, false);  
  }  
  else{  
    document.getElementById("divEx").attachEvent(  
      "onfocus", exempleFocus);  
    document.getElementById("divEx").attachEvent(  
      "onblur", exempleBlur);  
  }  
};  
  
function exempleFocus(event){  
  document.getElementById("pEx").style.display="block";  
}  
  
function exempleBlur(event){  
  document.getElementById("pEx").style.display="none";  
}  
</script>
```


Les widgets

Existence d'un grand nombre de rôles différents:

- `button`
- `checkbox`
- `treeitem`
- `timer`
- `slider`
- `combobox`
- `radiogroup`
- `progressbar`
- `alert`
- ...

ARIA suffisant?

- Une technique concernant la dynamique
- Désactivation du javascript dans les référentiels
- Les mix de technologies
- Doit-on réaliser deux interfaces?

AJAX dans le cycle de projet

Hijax et Progressive enhancement

1. **First, build an old-fashioned website** that uses hyperlinks and forms to pass information to the server. The server returns whole new pages with each request.
2. **Now, use JavaScript to intercept those links and form submissions** and pass the information via XMLHttpRequest instead. You can then select which parts of the page need to be updated instead of updating the whole page.

- Plan for Ajax from the start.
- Implement Ajax at the end.

Jeremy Keith sur DomScripting

Refaire tous les composants?

Implémenté dans de nombreux frameworks AJAX

- Dojo
- Yahoo UI
- GWT
- extJS
- jARIA pour jQuery
- ...

Étudier les différents outils et leur possibilité avant le démarrage.

Accessibilité et interface riche

- Possible dès aujourd'hui
- Améliore l'ergonomie globale
- Référencement naturel
- Framework d'IHM
- Support des navigateurs
- Support des assistance techniques

Ressources

- [WAI-ARIA](#)
- Open AJAX Alliance - Accessibility
- Introduction to WAI ARIA by Gez Lemon on Dev.Opera
- AccessiWeb
- Legifrance
- CodeTalks
- Adobe [VPAT](#)
- Hixia by Jeremy Keith

Merci de votre attention

