

Zimbra Forum France

Initiation au développement de Zimlets

Présenté par Vincent FRANCOIS



Développement de Zimlets

Zimlets

- Utilisation

- Gestion

- Exemples d'usages

Fonctionnement des Zimlets

- Principaux types

- Architecture : fichiers, emplacements

- « Hello World! »

Fonctions avancées

- Intégrations à l'interface : Tabs, boutons, menus, dans le texte

- « Hello World 2.0 »

- Connections à des services externes

Nouveautés v6

- Zimlet réseaux sociaux

- Formations au développement de Zimlets

Fonction des Zimlets

- **Utilisation**

 - Extension de l'interface utilisateur

- **Gadgets, ajouts à l'interface, outils de productivité**

 - Gestion

 - Interface administrateur

 - Ligne de commande

- **Exemples d'usages**

 - Rappel d'anniversaire, affichage de dates, intégration au SI

Fonctionnement des Zimlets

- Principaux types de zimlets

Service : Connexion à des services extérieurs

Extension : Ajout de fonctionnalités

Contenu, en-ligne : Ajout de liens sur mots-clefs dans le mail, le contact, etc

Fonctionnement des Zimlets

•Architecture

APIs Zimlets

-XML : Permet de créer des Zimlets simples (messages d'info, boites de dialogues simples), mais avec peu de fonctionnalités

-JavaScript+XML : Permet de créer des outils complexes avec facilité, debuggage aisé

Fichiers usuels

.xml : Description de la Zimlet, version, auteur, comportements simples

.js : Fichier principal de fonctionnement

.css : Styles utilisés dans la zimlet

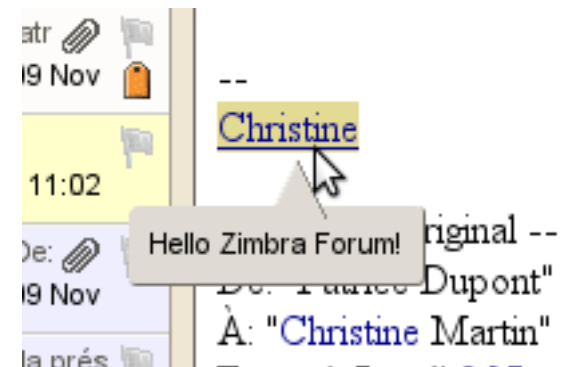
Image : icône pour le panneaux de Zimlet

config_template.xml : Configuration avancée (domaines autorisés,...)

Fonctionnement des Zimlets

•« Hello World! »

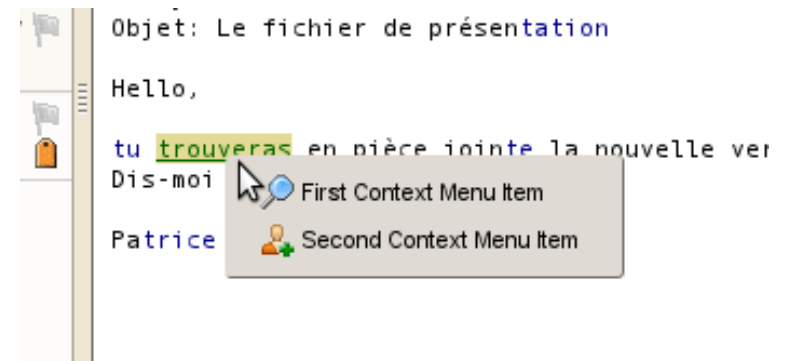
```
<zimlet name="com_zimbra_hello" version="1.0" description="Hello World1">  
  <contentObject>  
    <matchOn>  
      <regex attrs="ig">test|christine|starxpert</regex> <!-- cherche test, christine ou starxpert -->  
    </matchOn>  
    <toolTip>Hello Zimbra Forum</toolTip>  
  </contentObject>  
</zimlet>
```



Fonctionnement des Zimlets

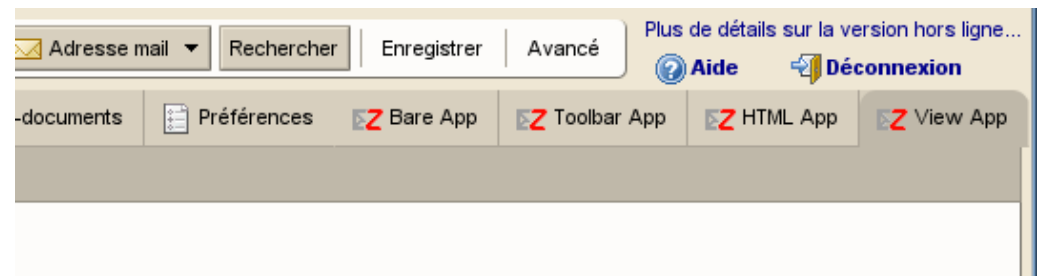
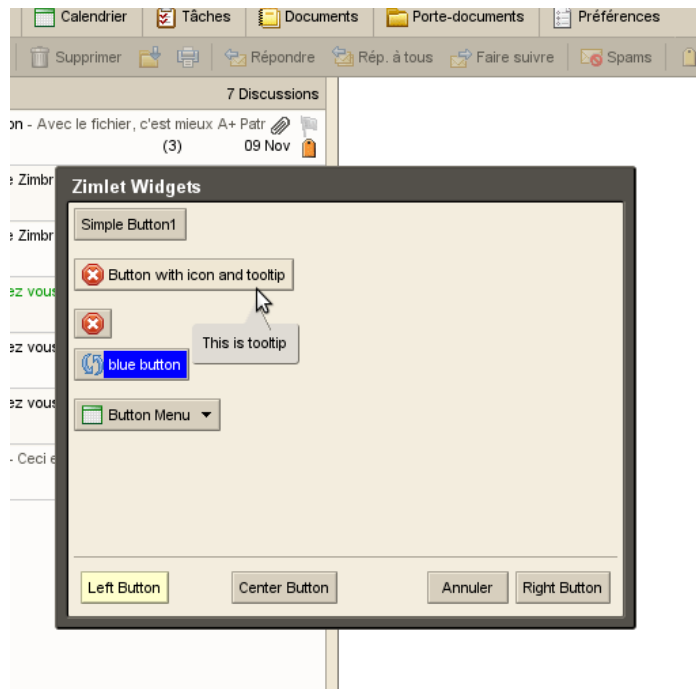
•« Hello World 2 »

```
...  
<matchOn>  
  <regex attrs="ig">t\w+</regex> <!-- Tous les mots avec t dedans -->  
</matchOn>  
<toolTip>Right click for more options</toolTip>  
<contextMenu>  
  <menuItem label="First Context Menu Item" icon="Search" id="SOME_ID1"/>  
  <menuItem label="Second Context Menu Item" icon="NewContact" id="SOME_ID2"/>  
</contextMenu>
```



Fonctions avancées

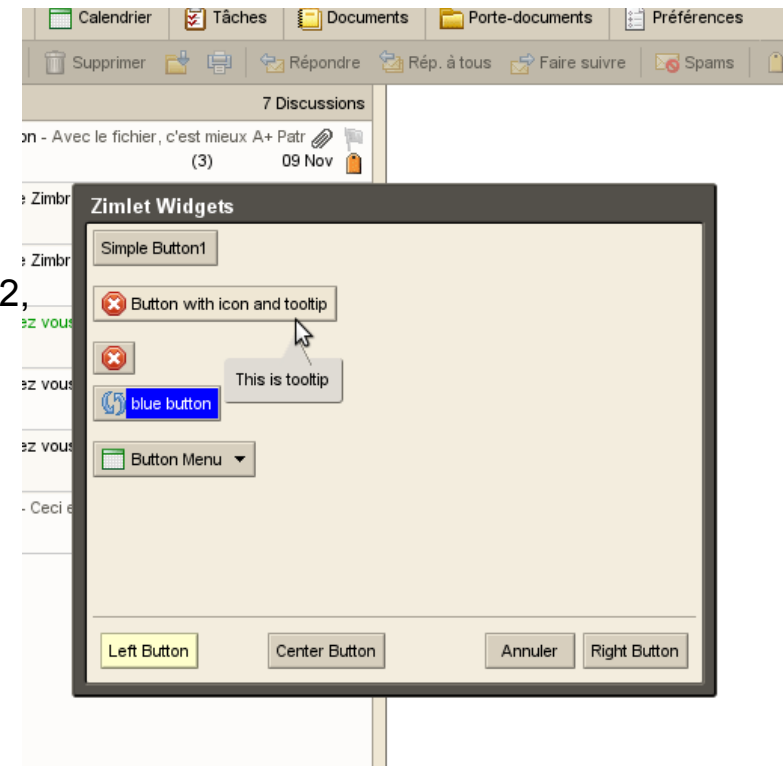
- Intégrations à l'interface
Tabs, boutons, menus, dans le texte...



Fonctions avancées

• Boîtes de dialogues

```
...
var btn1Id = Dwt.getNextId();
var btn1 = new DwtDialog_ButtonDescriptor(btn1Id, "Right Button",
DwtDialog.ALIGN_RIGHT);
...
this.pbDialog = this._createDialog({title:"Zimlet Widgets", view:this.pView,
standardButtons:[DwtDialog.CANCEL_BUTTON], extraButtons:[btn1, btn2,
btn3]});
...
this.pbDialog.setButtonListener(btn1Id, new AjxListener(this,
this._buttonListener, "Right Button"));
...
com_zimbra_hellowidgets1.prototype._buttonListener =
function(buttonName) {
    appCtx.getAppController().setStatusMsg(buttonName + " clicked",
ZmStatusView.LEVEL_INFO);
};
...
```



Fonctions avancées

- Boîtes de dialogues suite

```
var mi = menu.createMenuItem(id, {image:"zimbralcon",  
text:"simple menu item"});
```

...

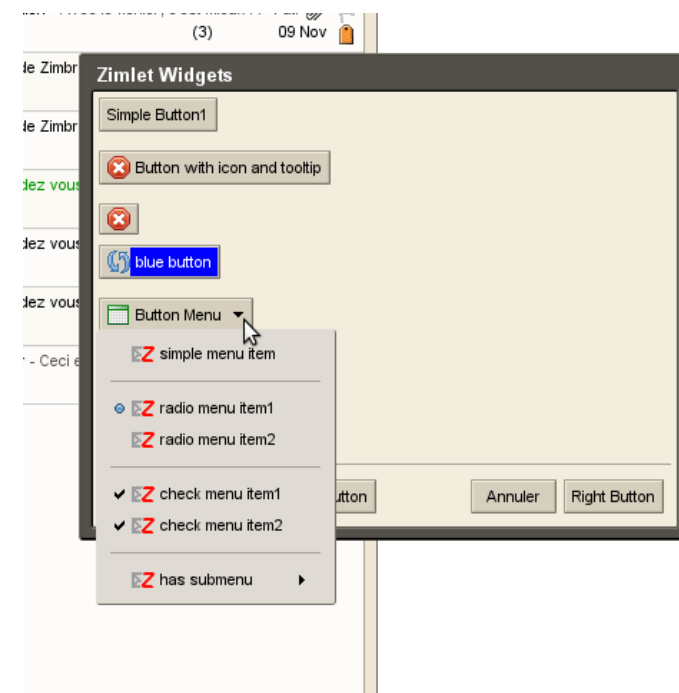
```
var mi = menu.createMenuItem(id, {image:"zimbralcon", text:"radio  
menu item1", style:DwtMenuItem.RADIO_STYLE});
```

...

```
var mi = menu.createMenuItem(id, {image:"zimbralcon", text:"check  
menu item1", style:DwtMenuItem.CHECK_STYLE});
```

...

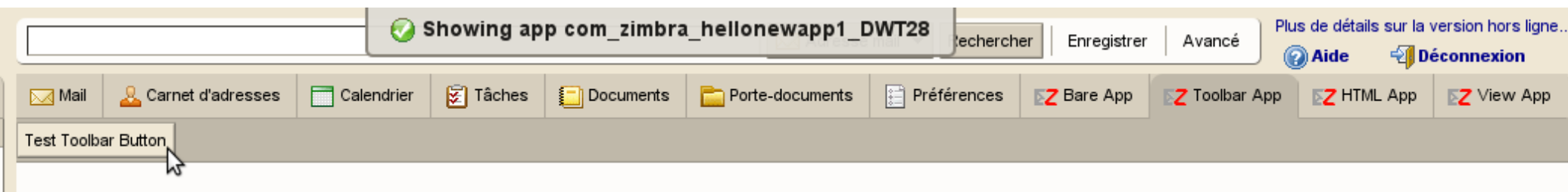
```
mi = menu.createMenuItem(id, {image:"zimbralcon", text:"has  
submenu", style:DwtMenuItem.CASCADE_STYLE});
```



Fonctions avancées

- Tabs

```
com_zimbra_hellonewapp1.prototype.init = function() {
    this._bareAppName = this.createApp("Bare App", "zimbralcon", "Basic app in a new tab");
    ...
com_zimbra_hellonewapp1.prototype.appLaunch = function(appName) {
    var app = appCtx.getApp(appName);//get access to app from appName
    switch (appName) {
        case this._toolbarAppName: {
            var toolbar = app.getToolbar();//returns the toolbar for the current app
            toolbar.createButton("TEST", {text:"Test Toolbar Button"});
            toolbar.addSelectionListener("TEST", new AjxListener(this, this._handleToolbarButton));
            break;
        }
    }
    ...
}
```



Fonctions avancées

- Connexions à des services externes

Requetes HTTP REST : Environnement Stateless, requêtes POST et GET

Affichage direct du résultat dans un cadre HTML, traitement par JS si format autre que htm (xml, texte, autre)

JSON (JavaScript Object Notation : Permet d'envoyer et recevoir des informations sous forme d'object JavaScript, plus facile à traiter

JSP : traitement par le fichier jsp joint

Requete REST:

```
AjxRPC.invoke(requestStr, serverUrl, requestHeaders, new AjxCallback(this, this._httpGETCallback), useGet, timeout)
```

Réponse REST:

```
zimlet.prototype._httpGETCallback = function(response) { ... }
```

Fonctions avancées

- Connexions à des services externes

HTTP SOAP : Environnement Statefull, requête et réponse en XML, fichiers de définition des fonctions

JSON : JavaScript est utilisé pour le traitement des requêtes

Requête SOAP :

```
var jsonObj = {CreateFolderRequest:{_jsns:"urn:zimbraMail"}};  
var folder = jsonObj.CreateFolderRequest.folder = {};
```

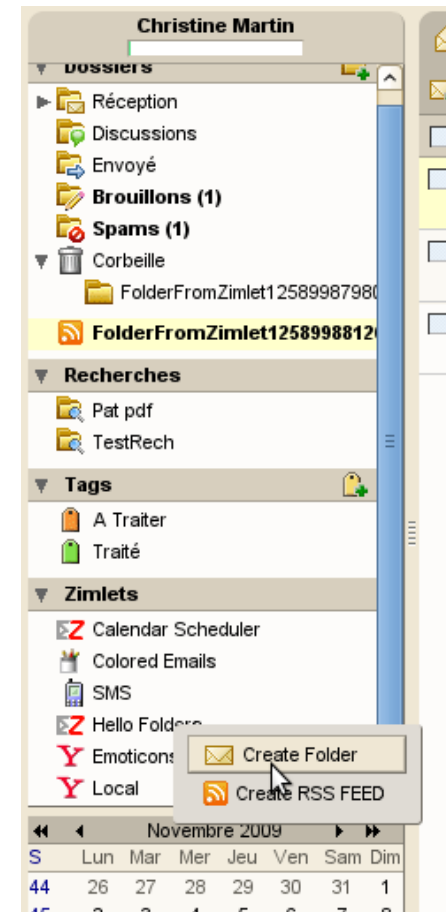
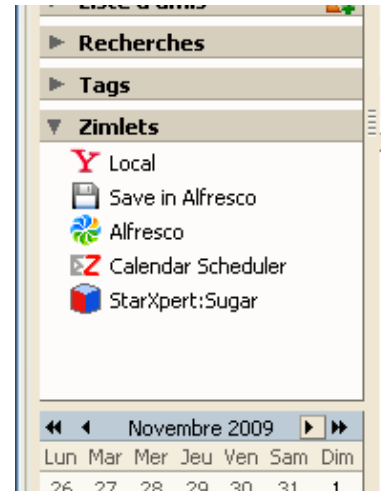
Réponse :

```
return appCtxt.getAppController().sendRequest({jsonObj:jsonObj, asyncMode:true,  
errorCallback:_createFldrErrCallback, callback:_createFldrCallback});
```

Fonctions avancées

- Exemples

- « Nouveau dossier en JSON »
- « SugarCRM » en SOAP
- « Alfresco »



Nouveautés v6

- Zimlet réseaux sociaux

Facebook, twitter, flickr

- Formations au développement de Zimlets

2 jours, profil développeur

Résumé

- Pour résumer
- Prise en main facile
- Profite de l'interface AJAX de Zimbra
- Intégration complète dans l'interface Zimbra
- Intégration d'outils externe