



ANGIE 1.1.0: come creare un website

Cesare Gerbino

ANGIE 1.1.0: come creare un website

Sommario

| | |
|--|----|
| Introduzione | 3 |
| Scopo del documento | 3 |
| Campo di applicazione | 3 |
| Riferimenti | 3 |
| Vincoli | 4 |
| ANGIE 1.1.0 con Tree View e senza Multiservice | 5 |
| ANGIE 1.1.0 senza Tree View e con Multiservice | 10 |

INTRODUZIONE

Scopo del documento

Questo documento ha lo scopo di descrivere le modalità con cui creare dei website ArcIMS basati sul framework **ANGIE** (**A**Nother **G**eographic **I**nteractive **E**xplorer), nella release 1.1.0. Le modalità di creazione sono relative a:

- Caso di un website con accesso a dati geografici forniti da altri website ArcIMS (multiservice) ma in cui l'informazione disponibile **NON** è gestita nella Table of Contents utilizzando una struttura "ad albero", ma con la gestione tradizionale offerta da ArcIMS
- Caso di un website senza accesso a dati geografici forniti da altri website ArcIMS (multiservice), ma in cui l'informazione disponibile è gestita nella Table of Contents utilizzando una struttura "ad albero"

Campo di applicazione

Il documento è applicabile, a tutti i website che devono essere realizzati con ArcIMS basandosi sul framework ANGIE release 1.1.0

Riferimenti

Ulteriori e più approfondite informazioni sulle caratteristiche di default di ArcIMS sono reperibili all'indirizzo <http://www.esri.com/>

Vincoli

Il framework ANGIE viene reso disponibile, come molto del software freeware scaricabile dalla rete, "as-is": se si riscontrassero errori o malfunzionamenti è tuttavia gradita la segnalazione all'indirizzo di posta elettronica info@cesaregerbino.it, in modo da permettere le necessarie correzioni.

ANGIE 1.1.0 CON TREE VIEW E SENZA MULTISERVICE

Nel seguito verranno indicati i passi da eseguire per la corretta creazione di un website utilizzando il framework ANGIE: il servizio sarà caratterizzato da una gestione dei livelli informativi organizzati in un tree-view, ma senza la possibilità di usufruire della funzionalità di “multiservice”

1. Copiare la cartella del website di ANGIE-SanFrancisco-1.1.0-SiTreeViewNoMS e cambiargli nome, assegnandogli quello desiderato
2. Modificare ArcIMSParam.js sostituendo il riferimento al mapservice SanFranciscoANGIE con il proprio mapservice e sostituendo il riferimento al mapservice di overview SanFranciscoANGIEOverView con il proprio mapservice di overview. Es.

```
var imsURL = "http://" + hostName + "/servlet/com.esri.esrimap.Esrimap?ServiceName=SanFranciscoANGIE";  
var imsOVURL = "http://" + hostName +  
"/servlet/com.esri.esrimap.Esrimap?ServiceName=SanFranciscoANGIEOverView";
```

3. Modificare ArcIMSParam.js le coordinate minime e massime con quelle di propri dati. Es.

```
//initial map extent  
  
var startLeft = .....;  
  
var startRight = .....;  
  
var startTop = .....;  
  
var startBottom = .....;  
  
//maximum map extent  
  
var limitLeft = .....;  
  
var limitRight = .....;  
  
var limitTop = .....;
```

```
var limitBottom = .....;
```

4. Modificare il titolo del servizio in `viewer.html`. Es.

```
var theTitle = "MyArcIMS Service";
```

5. In `ArcIMSParam.js` rendere visibili tutti campi dell'informazione associata dei diversi layers e non utilizzare alias. Es.

```
selfFieldList [0] = "";  
selfFieldList [1] = "";  
selfFieldList [2] = "";  
selfFieldList [3] = "";  
selfFieldList [4] = "";  
selfFieldList [5] = "";  
selfFieldList [6] = "";
```

```
fieldAliasList [0] = "";  
fieldAliasList [1] = "";  
fieldAliasList [2] = "";  
fieldAliasList [3] = "";  
fieldAliasList [4] = "";  
fieldAliasList [5] = "";  
fieldAliasList [6] = "";
```

6. In `dbgtdata.js` inserire i gruppi (Argomenti), e i layer che li compongono. Es.

```
var grpN = toc.addGroup( new GROUP("Group N",true) );  
grpN.addLayer( new LAYER("Layer 1","", "") );  
grpN.addLayer( new LAYER("Layer 2","", "") );
```

NOTA IMPORTANTE: Occorre fare molta attenzione ai nomi dei layers !!!! I nomi che vengono riportati in `dbgtdata.js` devono essere gli stessi riportati nell'AXL, anche il minimo errore determina un non corretto comportamento del javascript. Inoltre è opportuno, se viene modificato l'AXL, ricostruire il map service, non è sufficiente fare un refresh !!!!

7. Per ogni layer si possono quindi definire in `ArcIMSPParam.js` gli attributi che si intendono visualizzare Es.

```
selfFieldList[0]="#SHAPE# FIELD1 FIELD2";
```

NOTA IMPORTANTE: è necessario fare attenzione all'indice del layer !!! Non è quello riportato nell'AXL, né quello dell'ordine con cui il layer compare nella legenda.

8. Per ogni layer si possono quindi definire in `ArcIMSPParam.js` gli alias per gli attributi visualizzati Es.

```
fieldAliasList[0]="FIELD:Alias campo 1|FIELD2:Alias campo 2";
```

9. Per ogni layer su cui si vuole avere la possibilità di etichettare dinamicamente è necessario impostare l'opportuno valore dell'array `LayerLabelled`. Es.

```
var LayerLabelled = new Array();  
  
LayerLabelled[0] = false; //Imposta il layer con il checkbox per l'etichettura ma non attiva  
LayerLabelled[1] = true; //Imposta il layer con il checkbox per l'etichettura attiva  
LayerLabelled[2] = -1; //Imposta il layer senza il checkbox per l'etichettura
```

Inoltre si possono quindi definire in `ArcIMSPParam.js` le caratteristiche Es.

```
// the name of the layer  
ClassRenderLayer[1] = "Nome del layer";  
  
// the renderer for the shapes of the layer - copy this from the axl file  
ClassRenderString[1] =  
    '<SIMPLERENDERER>'  
    +'<SIMPLEMARKERSYMBOL color="51,102,51" width="6" />'  
    +'</SIMPLERENDERER>';  
  
// the renderer for the shapes of the layer  
ClassLabelString[1] =  
    '<SIMPLELABELRENDERER field="DESCRIZION" labelpriorities="1,2,2,4,5,3,2,4">'  
    +'<TEXTSYMBOL antialiasing="true" font="Verdana" fontstyle="bold" fontsize="8" blockout="255,255,0"  
>'  
    +'</SIMPLELABELRENDERER>';
```

NOTA IMPORTANTE: Anche in questo caso è necessario che i nomi dei layers siano gli stessi riportati nell'AXL: anche il minimo errore determina un non corretto comportamento del javascript. Inoltre è opportuno, se viene modificato l'AXL, ricostruire il map service, non è sufficiente fare un refresh !!!!

8 • ANGIE 1.1.0: come creare un website

La parte di “simbologia” è opportuno che sia la stessa riportata nell’AXL.

10. Settare in `ArcIMSParam.js` il layer che si desidera avere attivo alla partenza del servizio. Es.

```
var ActiveLayerIndex=20;
```

E’ opportuno che sia un layer non all’interno di un gruppo, o nel caso si desideri questa situazione, alla partenza del servizio quel gruppo deve risultare “aperto”

ANGIE 1.1.0 SENZA TREE VIEW E CON MULTISERVICE

Nel seguito verranno indicati i passi da eseguire per la corretta creazione di un website utilizzando il framework ANGIE: il servizio sarà caratterizzato da una gestione dei livelli informativi con modalità standard ArcIMS e quindi **NON** organizzati in un tree-view, ma con la possibilità di usufruire della funzionalità di “multiservice”

1. Copiare la cartella del website di ANGIE-SanFrancisco-1.1.0-NoTreeViewSiMS e cambiargli nome, assegnandogli quello desiderato
2. Modificare ArcIMSParam.js sostituendo il riferimento al mapservice ANGIE-SanFrancisco-1.1.0-NoTreeViewSiMS con il proprio mapservice e sostituendo il riferimento al mapservice di overview Overview con il proprio mapservice di overview. Es.

```
var imsURL = 'http://localhost/servlet/com.esri.esrimap.Esrimap?ServiceName=SFBusiness';  
var imsOVURL = 'http://localhost/servlet/com.esri.esrimap.Esrimap?ServiceName=Overview';
```

3. Modificare aimsMultiServiceParam.js sostituendo il riferimento al mapservice SFBusiness con il proprio mapservice nell'array dei mapservice gestiti con il multiservice. Es.

```
mapServiceList[0] = imsURL; //zero index starts with imsURL  
mapServiceName[0] = "San Francisco Business";
```

4. Modificare `ArcIMSParam.js` le coordinate minime e massime con quelle di propri dati. Es.

```
//initial map extent
var startLeft = .....;
var startRight = .....;
var startTop = .....;
var startBottom = .....;

//maximum map extent
var limitLeft = .....;
var limitRight = .....;
var limitTop = .....;
var limitBottom = .....;
```

5. Modificare il titolo del servizio in `viewer.html`. Es.

```
var theTitle = "MyArcIMS Service";
```

6. In `ArcIMSParam.js` rendere visibili tutti campi dell'informazione associata dei diversi layers e non utilizzare alias. Es.

```
selFieldList[0]="";  
selFieldList[1]="";  
selFieldList[2]="";  
selFieldList[3]="";  
selFieldList[4]="";  
selFieldList[5]="";  
selFieldList[6]="";
```

```
fieldAliasList[0]="";  
fieldAliasList[1]="";  
fieldAliasList[2]="";  
fieldAliasList[3]="";  
fieldAliasList[4]="";  
fieldAliasList[5]="";  
fieldAliasList[6]="";
```

7. Per ogni layer si possono quindi definire in `ArcIMSParam.js` gli attributi che si intendono visualizzare Es.

```
selFieldList[0]="#SHAPE# FIELD1 FIELD2";
```

NOTA IMPORTANTE: è necessario fare attenzione all'indice del layer !!! Non è quello riportato nell'AXL, né quello dell'ordine con cui il layer compare nella legenda.

8. Per ogni layer si possono quindi definire in `ArcIMSParam.js` gli alias per gli attributi visualizzati Es.

```
fieldAliasList[0]="FIELD:Alias campo 1|FIELD2:Alias campo 2";
```

9. Settare in `ArcIMSParam.js` il layer che si desidera avere attivo alla partenza del servizio. Es.

```
var ActiveLayerIndex=20;
```

E' opportuno che sia un layer che risulti visibile all'apertura del servizio