

ESTOS ProCall 3.0 Enterprise/One

Personalizzazione

Modifica layout Bolla di conversazione

09.03.2010, 1.00-ita



URL_RUN_customization		01/03/2010
-----------------------	--	------------

Soggetto ad errori e modifiche

Le specifiche tecniche e le illustrazioni grafiche presenti in questo documento corrispondono alle ultime informazioni disponibili al momento della redazione.

Ci riserviamo il diritto di apportare modifiche.

ESTOS GmbH declina ogni responsabilità per i Danni direttamente e indirettamente risultanti dall'uso di questo documento.

Tutti i marchi e i prodotti sono dei rispettivi proprietary.

Copyright ESTOS GmbH. Tutti I diritti riservati.

I termini e le condizioni d'uso possono essere trovati sul nostro sito:
<http://www.estos.de/agb.htm>

ESTOS GmbH
Petersbrunner Str. 3a
D-82319 Starnberg

info@estos.de
www.estos.de

Change log

Data	Versione	Autore	Descrizione
09/03/2010	1.0	SC	Creazione

Contenuti

Change log	3
1. Introduzione.....	5
1.1 Premesse	5
1.2 Necessità	5
1.3 Breve descrizione della struttura	5
1.4 Obiettivi	5
2. Struttura	6
3. Personalizzazioni.....	10
3.1 Visualizzazione campi personalizzati.....	10
3.2 Link URL	11
3.3 Link URL con immagine	13
3.4 Tabella Reference TAG	14
3.5 Tabella Reference Funzioni	14
4. Tips & Tricks	15
4.1 Modalità Debug	15
4.2 Elenco campi disponibili	16
4.3 HTML	16
5. Bibliografia e riferimenti	17

1. Introduzione

1.1 Premesse

Questa guida si rivolge a chi possiede già delle conoscenze medio/avanzate dell'HTML e delle sue forme e convenzioni. Questo documento non vuole essere un trattato esaustivo sull'XSLT ma solo fornire gli elementi necessari per la personalizzazione del popup di ESTOS ProCall 3.0 Enterprise chiamato Bolla di conversazione. Una bibliografia sulle fonti utilizzate per questa guida e link utili è disponibile al termine della trattazione.

1.2 Necessità

Una delle caratteristiche più importanti della suite ESTOS ProCall 3.0 Enterprise (insieme alla versione single-seat ProCall One) è la possibilità di integrazione spinta verso gli applicativi di produzione aziendale, nello specifico applicativi di CRM e/o ERP. La funzionalità più richiesta è la possibilità di scatenare degli eventi come, ad esempio, l'apertura di un contatto o di una scheda cliente basandosi su quanto succede al nostro sistema telefonico. Idealmente, è molto più facile trattare con un cliente o un fornitore se davanti a sé compare la scheda che lo riguarda, e ciò è ancora più performante se la scheda è apribile ancora prima che la chiamata sia effettivamente accettata.

1.3 Breve descrizione della struttura

Per fare quanto detto la soluzione più pulita e professionale è la modifica della cosiddetta "Bolla di conversazione". Essa è il popup che il client ProCall fa comparire ogni qual volta un evento telefonico accade. E' composta parzialmente da codice statico (Il contorno, I pulsanti per il Remote Call Control, la casella per le Note) e parzialmente da una pagina HTML che viene generata da un foglio XSLT. XSLT (o eXtensible Stylesheet Language Transformations) è il linguaggio di trasformazione XML più diffuso, esso è uno standard W3C e la sua struttura è univocamente definita dal consorzio W3C.

1.4 Obiettivi

Il linguaggio XSLT permette l'estrazione di campi anagrafica e la loro visualizzazione, l'inserimento di condizioni rispetto ai singoli campi e la creazione di indirizzi URL personalizzati contenenti i dati dell'anagrafica stessa. Questo

manuale non volendo e non potendo essere un manuale di XSLT, si occuperà pertanto solamente di quelle sintassi che vanno ad interagire con i dati di ProCall e ci aiutano a creare quelle integrazioni che tanto distinguono la suite ESTOS dal resto delle soluzioni CTI.

2.Struttura

Premessa: Quanto verrà trattato si deve considerare valevole anche nell'utilizzo di ESTOS ProCall One.

Tutti i file su cui andremo a lavorare sono presenti nella directory "templates/" presente all'interno dell'installazione client di ProCall 3 Enterprise. Vi sono differenti file utilizzati in diverse occasioni:

File	Descrizione
common.xslt	Questo file fornisce le direttive grafiche della bolla e alcune funzioni Javascript predefinite. E' fondamentale che questo file sia trasportato dalla directory default alla directory subito superiore insieme agli altri file modificati.
unanswered.xslt	Questo è il template del modello di email che viene spedito in caso di chiamata persa. Ulteriori dettagli sono disponibili nell'help del client ProCall.
contact.xslt	Questo è il template di visualizzazione della linguetta "Veduta d'insieme" presente nella finestra "Dettagli contatto" del client ProCall.
RemoteContact.xslt	E' utilizzato per visualizzare I dettagli di un contatto che chiama/è stato chiamato direttamente.
LocalContact.xslt	E' utilizzato per visualizzare I dettagli del "Contatto chiamato". Funzione disponibile mediante abilitazione dell'opzione "Mostra il contatto chiamato" nelle Impostazioni del client ProCall
HoldContact.xslt	E' utilizzato per visualizzare I dettagli di una chiamata messa in attesa oppure di un interno che ci sta girando una telefonata in attesa.
RedirectedContact.xslt	E' utilizzato per visualizzare I dettagli di una chiamata ridirezionata.

Prima di cominciare l'editing, è necessario trasbordare i file che andremo a modificare dalla directory "templates/default/" ad un livello più in alto ovvero "templates/". Questo passo è necessario poichè in caso di aggiornamento del client la cartella "default/" sarà riportata ai settaggi di default ed eventuali personalizzazioni andrebbero perse. E' importante, inoltre, che anche il file common.xslt sia copiato.

Una volta aperti i file (con qualunque editor di testo ma personalmente sento di consigliarne uno che colori il codice in modo da non dover perdere tempo a cercare quella parentesi mancante, un ottimo e gratuito esempio è PSPad www.pspad.com/it/) troverete del codice HTML mescolato a quell XSLT, il codice

XSLT in realtà verrà convertito in HTML dal renderer incorporato nel client ProCall (Sablotron). Analizziamo in breve la struttura tipo del file RemoteContact.xslt (la scelta ricade su questo in quanto è il file più utilizzato dal traffico telefonico standard).

```
<?xml version="1.0" encoding="utf-8"?>
<!-- This file must be saved in UTF8 encoding with proper UTF8 byte order mark -->
<xslt:stylesheet version="1.0" xmlns:xslt="http://www.w3.org/1999/XSL/Transform">
  <xslt:output method="html" version="1.0" encoding="utf-8"/>
  <!-- Helper functions for generating links, javascript and css declarations -->
  <xslt:include href="common.xslt"/>
  <xslt:template match="/">
    <xslt:text disable-output-escaping="yes">&lt;!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"&gt;</xslt:text>
    <html>
      <head>
        <script type="text/javascript">
          <xslt:call-template name="javascript"/>
        </script>
        <xslt:call-template name="buildCSS"/>
      </head>
      <body minwidth="0" minheight="0" maxwidth="0" maxheight="0" debug="0">
        <table cellpadding="0" cellspacing="0">
          <xslt:call-template name="Company">
            <xslt:with-param name="Company" select="XSLT/Contact/Company"/>
            <xslt:with-param name="Department" select="XSLT/Contact/Department"/>
          </xslt:call-template>
          <xslt:call-template name="RemoteContactInformation">
            <xslt:with-param name="Businessstreet" select="XSLT/Contact/BusinessAddressStreet"/>
            <xslt:with-param name="Businesscity"
select="concat(XSLT/Contact/BusinessAddressPostalCode, ' ', XSLT/Contact/BusinessAddressCity)"/>
            <xslt:with-param name="Businesscountry" select="concat(XSLT/Contact/BusinessAddressState,
' ', XSLT/Contact/BusinessAddressCountry)"/>
            <xslt:with-param name="Privatestreet" select="XSLT/Contact/PrivateAddressStreet"/>
            <xslt:with-param name="Privatecity" select="concat(XSLT/Contact/PrivateAddressPostalCode, '
', XSLT/Contact/PrivateAddressCity)"/>
            <xslt:with-param name="Privatecountry" select="concat(XSLT/Contact/PrivateAddressState, ' ',
XSLT/Contact/PrivateAddressCountry)"/>
          </xslt:call-template>
        </table>
```

Com'è possibile vedere, tralasciando tutta la struttura HTML classica viene creata una tabella che viene popolata richiamando due template, uno chiamato *Company* e uno chiamato *RemoteContactInformation* tramite il tag **<xslt:call-template>** a cui vengono poi passati come parametri alcuni valori presi dall'anagrafica tramite il tag **<xslt:with-param name="Company" select="XSLT/Contact/Company"/>** in cui il campo *Company* viene associato alla variabile XSLT chiamata anch'essa *Company*. I valori così impostati vengono utilizzati nella definizione del template nel modo seguente:

```
<xslt:template name="Company"> Nomina del template
  <xslt:param name="Company"/> Richiamo della variabile
  <xslt:param name="Department"/> Richiamo della variabile
  <xslt:if test="$Company != "">
    <tr><td class="boldfont"><xslt:value-of select="concat($Company, ' ',
$Department)"/></td></tr>
  </xslt:if>
</xslt:template>
```

Qui viene effettuato il primo test tramite il tag **<xslt:if>**, la condizione in questo caso è che il contenuto della variabile *Company* non sia vuoto, soddisfatta la condizione l'interprete proseguirà lungo il codice rimanente, altrimenti nulla apparirà in bolla in quanto non c'è nulla da visualizzare.

```
<xslt:template name="RemoteContactInformation">
  <xslt:param name="Businessstreet"/>
  <xslt:param name="Businesscity"/>
  <xslt:param name="Businesscountry"/>
  <xslt:param name="Privatestreet"/>
  <xslt:param name="Privatecity"/>
  <xslt:param name="Privatecountry"/>

<xslt:choose>
  <xslt:when test="normalize-space($Businessstreet) = '' and normalize-space($Privatestreet) = "">
    <xslt:choose>
      <xslt:when test="normalize-space($Businesscity) != "">
        <tr><td><xslt:value-of select="$Businesscity"/> - <xslt:value-of
select="$Businesscountry"/></td></tr>
      </xslt:when>
      <xslt:when test="normalize-space($Privatecity) != "">
        <tr><td><xslt:value-of select="$Privatecity"/> - <xslt:value-of
select="$Privatecountry"/></td></tr>
      </xslt:when>
    </xslt:choose>
  </xslt:when>
</xslt:choose>
```

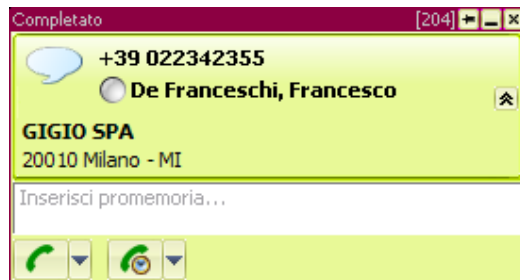

Tramite la tag `<xslt:choose>` invece, abbiamo la possibilità di utilizzare la classica struttura if-then-else comune a tutti i linguaggi di programmazione applicandola ai campi dell'anagrafica e decidendo quindi dinamicamente quali campi mostrare a seconda del loro contenuto.

In questo caso viene effettuato un test sulla variabile `Businesscity` ma non tramite il tag `<xslt:if>` bensì tramite `<xslt:when>` la cui sintassi è identica salvo il fatto che permette l'utilizzo successivo della tag `<xslt:otherwise>` le cui azioni vengono eseguite nel caso in cui le condizioni del tag `<xslt:when>` non siano verificate.

```
<xslt:when test="normalize-space($Businesscity) != "">
  <tr><td><xslt:value-of select="$Businessstreet"/></td></tr>
  <tr><td><xslt:value-of select="$Businesscity"/></td></tr>
  <tr><td><xslt:value-of select="$Businesscountry"/></td></tr>
</xslt:when>
<xslt:when test="normalize-space($Privatecity) != "">
  <tr><td><xslt:value-of select="$Privatestreet"/></td></tr>
  <tr><td><xslt:value-of select="$Privatecity"/></td></tr>
  <tr><td><xslt:value-of select="$Privatecountry"/></td></tr>
</xslt:when>
</xslt:choose>
</xslt:template>
```

Infine ancora un paio di test controllando il contenuto delle variabili `Businesscity` e `Privatecity` e, nel caso non siano vuote, viene creata una riga di tabella contenente i relativi valori (estratti tramite il tag `<xslt:value-of>`).

Il risultato sarà il seguente:



Completato [204]

+39 022342355
De Franceschi, Francesco

GIGIO SPA
20010 Milano - MI

Inserisci promemoria...

Call icons

3. Personalizzazioni

2.1 Visualizzazione campi personalizzati

E' possibile inserire all'interno della Bolla di conversazione qualunque dato presente in anagrafica per una loro visualizzazione a schermo allo scatenamento di un evento telefonico. Un esempio di utilizzo può essere per tutti quegli ambienti in cui è utilizzato un applicativo produttivo su piattaforma UNIX, AS400 o similare in cui il terminale non è interagibile da applicazioni esterne. Potremmo voler far sì che all'interno della Bolla compaia anche il nostro ID cliente.

Supponiamo di aver mappato (all'interno del client ProCall 3.0 o tramite ESTOS MetaDirectory) l'ID cliente del nostro database relazionale al campo *custom0* della struttura LDAP.

Vediamo nel dettaglio le modifiche:

Nella parte di assegnamento delle variabili andiamo ad inserire

```
<xslt:with-param name="Codice" select="XSLT/Contact/Custom0"/>
```

In questo modo andiamo ad assegnare il valore del campo Custom0 alla variabile XSLT chiamata Codice

Poi andremo nella parte di definizione del template e inseriremo:

```
<xslt:param name="Codice"/>
```

per rendere utilizzabile la variabile all'interno del template e

```
<tr><td>Codice: <xslt:value-of select="$Codice"/></td></tr>
```

all'interno della parte di printing delle variabili.

Quindi il codice completo sarà:

[...]

```
<xslt:call-template name="RemoteContactInformation">
  <xslt:with-param name="Businessstreet" select="XSLT/Contact/BusinessAddressStreet"/>
  <xslt:with-param name="Businesscity"
select="concat(XSLT/Contact/BusinessAddressPostalCode, ' ', XSLT/Contact/BusinessAddressCity)"/>
```

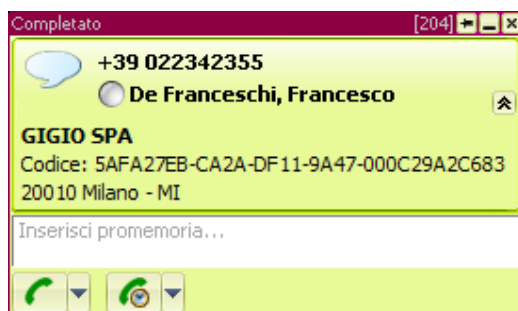
```

<xslt:with-param name="Businesscountry" select="concat(XSLT/Contact/BusinessAddressState,
', XSLT/Contact/BusinessAddressCountry)"/>
<xslt:with-param name="Privatestreet" select="XSLT/Contact/PrivateAddressStreet"/>
<xslt:with-param name="Privatecity" select="concat(XSLT/Contact/PrivateAddressPostalCode, '
', XSLT/Contact/PrivateAddressCity)"/>
<xslt:with-param name="Privatecountry" select="concat(XSLT/Contact/PrivateAddressState, ' ',
XSLT/Contact/PrivateAddressCountry)"/>
<xslt:with-param name="Codice" select="XSLT/Contact/Custom0"/>
</xslt:call-template>
[...]
```

```

<xslt:template name="RemoteContactInformation">
  <xslt:param name="Businessstreet"/>
  <xslt:param name="Businesscity"/>
  <xslt:param name="Businesscountry"/>
  <xslt:param name="Privatestreet"/>
  <xslt:param name="Privatecity"/>
  <xslt:param name="Privatecountry"/>
  <xslt:param name="Codice"/>
  <tr><td>Codice: <xslt:value-of select="$Codice"/></td></tr>
[...]
```

e il risultato sarà:



2.1 Link URL

Seguendo i criteri enunciati fin'ora, è possibile generare un URL clickabile all'interno della Bolla di conversazione e utilizzare i dati dell'anagrafica come parametri. Supponiamo di avere a che fare con un'anagrafica presa da un sistema CRM come Microsoft Dynamics. I requisiti sono i seguenti

- Aver mappato la sorgente dati (via client o tramite ESTOS MetaDirectory2) dell'anagrafica

- Aver mappato l'ID cliente in uno dei campi personalizzati (in questo caso: custom0)

Paradossalmente l'implementazione in questo caso è più semplice, gli step sono due:

```
<xslt:variable name="openlink" select="concat('http://crmsrv/AZIENDAsnc/sfa/conts/edit.aspx?id={',  
normalize-space(XSLT/Contact/CustomerID),'}')" />
```

In questo caso andiamo a creare una variabile di nome "openlink" in cui l'URL contenuto è una concatenazione di stringhe (tramite il comando "concat") che comprende in primis l'url del gestionale da noi utilizzato e in secundis il codice cliente preso direttamente dall'anagrafica.

```
<tr><td><a href="{ $openlink}" target="_blank">Apri Contatto...</a></td></tr>
```

In questo secondo step invece andiamo a inserire il link qui sopra nella parte di definizione del template bolla (similarmente a quanto fatto per la visualizzazione di Codice). Il risultato a quel punto sarà il seguente:



L'URL di "Apri Contatto..." sarà

<http://crmsrv/AZIENDAsnc/sfa/conts/edit.aspx?id={76D8B811-ECBF-DE11-85C6-000C29A2C683}>

Ovviamente quanto detto è possibile anche nella casistica di un URL personalizzato (vedi documento ESTOS INFO05_creazione_url_custom) quindi seguendo l'esempio di "gestionale" la composizione URL sarà (ad esempio):

```
<xslt:variable name="openlink" select="concat('gestionale:', normalize-  
space(XSLT/Contact/CustomerID))" />
```

Il che permetterà il lancio dell'eseguibile legato a "gestionale" passandogli come parametro il CustomerID del contatto direttamente con un click sulla Bolla di conversazione.

2.2 Link URL con immagine

E' possibile rendere più armonioso l'aspetto grafico della Bolla rendendo il link un'icona cliccabile. Sono sufficienti pochi cambiamenti:

```
<xslt:variable name="pic_opencont" select="concat('file:///', normalize-  
space(XSLT/General/TemplatePath), 'opencont.png')"/>
```

Innanzitutto definire una variabile "pic_opencont" il cui contenuto è un puntamento a un'icona di nome "opencont.png" presente nella directory indicata da XSLT/General/TemplatePath (che, di default è la directory "templates/" dell'installazione del client ProCall). Per quanto riguarda le icone consiglio una visita a <http://www.iconarchive.com/> dove troverete una disponibilità enorme e gratuita.

A quel punto creeremo la variabile "openlink" come prima:

```
<xslt:variable name="openlink"  
select="concat('http://crmsrv/PARISSISTEMISnc/sfa/conts/edit.aspx?id={', normalize-  
space(XSLT/Contact/CustomerID),'}')"/>
```

però ci sarà una differenza nella creazione del link vero e proprio

```
<tr><td><a href="{ $openlink}" target="_blank"></a></td></tr>
```

Il risultato sarà il seguente:



2.3 Tabella Reference tag utilizzati nella presente guida:

Tag	Descrizione
<xslt:if>	Permette di porre condizioni al codice, dopo l'apertura del tag si fa seguire l'istruzione "test=". Se la condizione è verificata allora tutto il codice fino alla chiusura del tag sarà trasformato.
<xslt:choose>	Si usa in congiunzione con xslt:when e xslt:otherwise. E' necessario che questi ultimi stiano al suo interno.
<xslt:when>	Uguale, nel merito, ad xslt:if.
<xslt:otherwise>	Il codice all'interno di questo tag verrà eseguito solo nel caso in cui NON siano verificate le condizioni poste dentro xslt:when
<xslt:value-of>	Estrae il valore di una variabile
<xslt:variable>	Definisce il nome di una variabile e le assegna un valore.
<xslt:with-param>	Dichiara l'utilizzo di tale variabile all'interno del template.

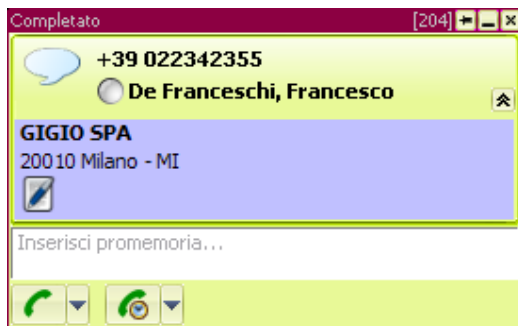
2.4 Tabella funzioni XSLT utilizzate nella presente guida:

Comando	Utilizzo nei Tag	Descrizione
normalize-space	Tutti	Serve ad eliminare spazi in eccesso, nello specifico elimina tutti gli spazi in testa e in coda a una stringa e riduce tutti gli spazi intermedi ad uno spazio solo.
select	<xslt:value-of>	Estrae il valore di una variabile o di un path XSLT (es. select=XSLT/Contact/DisplayedName)
concat	Tutti	Serve a concatenare una serie di stringhe, siano essere stringhe o variabili, al fine di formarne una unica, la sintassi è concat('stringa1','stringa2',stringaN')
name	<xslt:variable>	Definisce il nome di una variabile sia nel caso di creazione che di un suo richiamo.
test	<xslt:if> <xslt:when>	Serve a porre le condizioni in base alle quali vogliamo che il codice seguente sia interpretato.

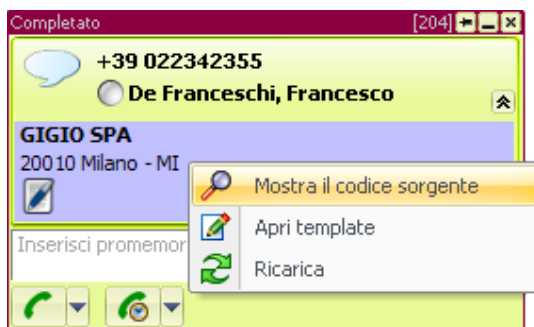
4. Tips & Tricks

4.1 Modalità Debug

E' possibile impostare il file della Bolla in modalità "debug", questa modalità farà sì che, ogni qualvolta la bolla viene utilizzata, lo sfondo della stessa si presenterà in azzurro e sarà possibile fare click di tasto DX sopra e selezionare la voce "Visualizza codice sorgente". Si aprirà il vostro editor di testo predefinito mostrandovi il codice sorgente HTML della bolla DOPO le trasformazioni effettuate dal renderer XSLT. In questo modo, oltre ad avere chiara visione di quello che è stato scritto/trasformato, potrete visualizzare tutti i dati dell'anagrafica del contatto di cui la Bolla può disporre. Esempio:



Tasto DX ->



Per attivare questa modalità è sufficiente pescare la riga della Bolla in cui è presente la voce "debug="0"" e sostituirla con "debug="1"".

Nel caso di RemoteContact.xslt essa è

```
<body minwidth="0" minheight="0" maxwidth="0" maxheight="0" -debug="1">
```

I campi dell'anagrafica appariranno così:

[...]

<XSLT>

<Contact>

<DatabaseName>MSDynamics</DatabaseName>

<EntryIDDB>DB06</EntryIDDB>

<EntryIDStore>MSDynamics</EntryIDStore>

<EntryID>5AFA27EB-CA2A-DF11-9A47-000C29A2C683</EntryID>

<DisplayName>De Franceschi, Francesco</DisplayName>

<FirstName>Francesco</FirstName>

<LastName>De Franceschi</LastName>

<JobTitle>Direttore Tecnico</JobTitle>

<Company>GIGIO SPA</Company>

<CustomerID>76D8B811-ECBF-DE11-85C6-000C29A2C683</CustomerID>

<BusinessAddressPostalCode>20010</BusinessAddressPostalCode>

[...]

4.2 Elenco campi disponibili

L'elenco dei campi anagrafica disponibili e utilizzabili con la sintassi XSLT/Contact/nomecampo all'interno della Bolla è disponibile all'interno del file **contact.xslt**.

4.3 HTML

Ricordo al lettore che la Bolla, nonostante presenti elementi XSLT, è, di fatto, scritta in HTML e quindi accetta tranquillamente l'implementazione di feature standard come iframe o javascript (esempi testati).

5. Bibliografia e riferimenti

Reference XSLT:

Consorzio W3C: <http://www.w3schools.com/XSL/> (inglese)

HTML.it: <http://xml.html.it/guide/leggi/60/guida-xslt/> (italiano)

Utility:

PSPad (editor di testo freeware): <http://www.pspad.com/it/> (italiano e inglese)

Note: Sono ben accetti suggerimenti e critiche riguardante la guida in oggetto. Per ogni richiesta o precisazione potete scrivere a stefano.chittaro@estos.it