

Agenda con PERL, LATEX y XML

Por Paco Aldarias Raya

Impreso: 9 de septiembre de 2007

Email: pacolinux@pacodebian.es

Web: <http://pagina.de/pacodebian>

Con Linux Debian. En Valencia (España)

El documento tiene version .html, y .pdf, cambiando en el navegador la parte final podrás acceder a ambos.

Este documento es de libre reproducción siempre que se cite su fuente.

Realizado con: **L^AT_EX**

Índice

Índice	1
1. Introducción	1
2. La fuente de datos: agenda.xml	1
3. Generando listado por pantalla con perl: agenda.pl	2
4. Generando pdf con latex y perl	2
5. Generando pdf con xml y fop: agendapdf.sh	4
6. Generando html con xml: agendahtml.sh	7
7. Generando tarjetas vcard con xml y perl: vcard.pl	8
8. Hacer un cuadernillo tipo agenda alfabetica:	8
9. Todo junto: agenda.sh	10

1. Introducción

Vamos a contruir una agenda en xml, y vamos a obtener la agenda en formato pdf, txt, html, tarjetas vcard para mandarlas al movil y un cuadernillo tipo cuartilla para consultarlo en papel.

2. La fuente de datos: agenda.xml

Este fichero tiene dos estructuras básica, agenda y persona, las cuales podremos usar después. Podemos añadir a persona los atributos que querramos, por ejemplo, movil, fechanacimiento, sexo, etc.

Este fichero se puede crear a mano con cualquier editor de html.

```
<?xml version="1.0" encoding="ISO-8859-15"?>
<?xml-stylesheet href="agenda.xsl" type="text/xsl"?>
<agenda>

  <titulo>Agenda</titulo>

  <autor>Paco Aldarias</autor>

  <persona>
    <nombre>SEGUROS.</nombre>

    <apellidos>GENESIS AUTO</apellidos>

    <telefono>902401560</telefono>

    <web>http://www.genesis.es/</web>

    <email>genesisauto@genesis.es</email>
  </persona>

  <persona>
    <nombre>SCL CLIMATIZACION</nombre>

    <apellidos>Jose Vicente - Omar</apellidos>

    <email>scl@gruposcl.com</email>
  </persona>
</agenda>
```

3. Generando listado por pantalla con perl: agenda.pl

Perl permite manipular ficheros xml, lo utilizaremos para generar por pantalla un listado de personas.

```
#!/usr/bin/perl
# http://www.merelo.net/tutoperl/xml.htm c
# http://www.omerique.net/twiki/pub/TIC/LenguajePERL/tutorial-print.html
# Instalar perl: perl -MCPAN -e 'install XML::Simple'
print "*****\n";
use XML::Simple;
my $agenda = XMLin("agenda.xml");
for( @{$agenda->{persona}} ) {
  print "$->{nombre} $->{apellidos} $->{telefono} $->{movil} $->{email} $->{web} $->{direccion} $->{localidad} $->{
}
  print "*****\n";
```

4. Generando pdf con latex y perl

Pasos:

1. Con perl latex.pl: Genermos agenda_datos.tex , que será utilizado por agenda-latex.tex.
2. Con pdflatex agenda-latex.tex: Generamos agenda-latex.pdf

cat latex.pl

```
#!/usr/bin/perl
# http://www.merelo.net/tutoperl/xml.htm
# Instalar perl: perl -MCPAN -e 'install XML::Simple'
use XML::Simple;
open (tex,">agenda_datos.tex");
my $agenda = XMLin("agenda.xml");

print tex "\\begin{itemize} \n";

for( @{$agenda->{persona}} ) {
print tex "\\item \\textbf{$_->{nombre}} $_->{apellidos} $_->{telefono} $_->{movil} \\url{$_->{email}} \\url{$_->{web}}
}

print tex "\\end{itemize} \n";
```

Ahora crearemos el fichero latex que debe contener el fichero agenda-latex.tex generado por latex.pl. Este servirá para crear el fichero pdf agenda-latex.pdf.

cat fichero agenda-latex.tex

```
%Nota: el texto a la derecha de un %
%es un comentario para LaTeX y es ignorado
\documentclass[10pt,a4paper,twocolumn]{article}
\usepackage[spanish, activeacute]{babel} %Españolizacion
\usepackage[latin1]{inputenc} %Letras con acentso, eñes
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{eurosym}
\usepackage{helvet}
\def\rmdefault{\sfdefault}
\usepackage{makeidx} % Indice de materias
\makeindex % Indice de materias
\usepackage{hyperref}
\usepackage{listings}
\usepackage{fancyhdr}
\pagestyle{fancy}
\usepackage{ifpdf}
\ifpdf
  % Margenes normales:
  \usepackage[a4paper,pdftex,top=3cm,bottom=3cm,left=3cm,right=3cm]{geometry}
  \usepackage[pdftex]{graphicx}
\else
  \usepackage[dvips]{geometry}
  \usepackage[dvips]{graphicx}
\fi
\ifpdf
\else
  %% Declaraciones 'utiles para los gr'aficos
  \graphicspath{{figuras/}{fig2/}{fig3/}{fig4/}{fig5/}}
  \DeclareGraphicsExtensions{.eps,.ps,.eps.gz,.ps.gz,.jpg,.gif,.png}
  \DeclareGraphicsRule{.jpg}{eps}{.jpg.bb}{'jgptops #1}
  \DeclareGraphicsRule{.gif}{eps}{.gif.bb}{'giftops #1}
  \DeclareGraphicsRule{.png}{eps}{.png.bb}{'pngtops #1}
\fi
\title{\titulo}
\date{}
\parskip 1ex %Lineas entre prrrafos
\usepackage{lastpage} %Ultima pagina.
\addtolength{\headwidth}{\marginparwidth}
\addtolength{\headwidth}{\marginparsep}
```

```
\fancyheadstyle{plain}
{%
    \fancyhead[l]{\today}
    \fancyhead[r]{\titulo}
    \fancyhead[c]{\today}
    \renewcommand{\headrulewidth}{0.4pt}
    \fancyfoot[l]{Por Paco Aldarias}
    \fancyfoot[c]{\today}
    \fancyfoot[r]{\thepage/\pageref{LastPage}}
    \renewcommand{\footrulewidth}{0.4pt}
}

\lhead{\today}
\chead{}
\rhead{
\titulo}
\renewcommand{\headrulewidth}{0.4pt}
\lfoot{Por Paco Aldarias}
\cfoot{}
\rfoot{\thepage/\pageref{LastPage}}
\renewcommand{\footrulewidth}{0.4pt}
\begin{document}
\newcommand{\titulo}{AGENDA TELEFONICA}
\noindent
\input{agenda_datos}
\end{document}
```

Este script lo hará todo: latex.sh

```
#!/bin/bash
perl latex.pl
#cat agenda_datos.tex | sort > agenda1.tex
#mv agenda1.tex agenda_datos.tex
pdflatex agenda-latex.tex
pdflatex agenda-latex.tex
```

5. Generado pdf con xml y fop: agendapdf.sh

Fichero de estilo: agenda-pdf.xsl
cat agenda-pdf.xsl

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<xsl:stylesheet
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fo="http://www.w3.org/1999/XSL/Format"
  version="1.0">
<xsl:param name="page-number-print-in-footer">true</xsl:param>
<xsl:template match="agenda">

  <fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">

    <fo:layout-master-set>

      <fo:simple-page-master master-name="simple"
        page-height="29.7cm"
        page-width="21cm"
        margin-top="1cm"
        margin-bottom="2cm"
        margin-left="3.5cm"
        margin-right="3.5cm">
        <fo:region-body margin-top="3cm"/>
        <fo:region-before extent="3cm"/>
        <fo:region-after extent="1.5cm"/>
      </fo:simple-page-master>
    </fo:layout-master-set>
  </fo:root>
</xsl:template>
</xsl:stylesheet>
```

```
</fo:simple-page-master>
</fo:layout-master-set>

<fo:page-sequence master-reference="simple">

  <fo:static-content flow-name="xsl-region-after">
    <fo:block text-align="center"><fo:page-number /></fo:block>
  </fo:static-content>

  <fo:flow flow-name="xsl-region-body">
    <xsl:apply-templates/>
  </fo:flow>

</fo:page-sequence>

<!--
<fo:page-sequence>
  <fo:static-content flow-name="xsl-before">
    <fo:block text-align="center" font-size="9pt">
      <fo:page-number/>
    </fo:block>
  </fo:static-content>
</fo:page-sequence>
-->

</fo:root>
</xsl:template>

<xsl:template match="persona">
  <fo:block space-after.optimum="3pt"
    space-before.optimum="4pt">
    <xsl:apply-templates/>
  </fo:block>
</xsl:template>

<xsl:template match="titulo">
  <fo:block font-size="20pt"
    font-family="sans-serif"
    line-height="24pt"
    space-after.optimum="15pt"
    text-align="center"
    padding-top="3pt">
    <xsl:value-of select="."/>
  </fo:block>
</xsl:template>

<xsl:template match="autor">
  <fo:block font-size="12pt"
    font-family="sans-serif"
    line-height="15pt"
    space-after.optimum="3pt"
    text-align="center">
    <xsl:value-of select="."/>
  </fo:block>
</xsl:template>

<xsl:template match="fecha">
  <fo:block font-size="12pt"
    font-family="sans-serif"
    line-height="15pt"
    space-after.optimum="3pt">
```

```
        text-align="center">
    <xsl:value-of select="."/>
</fo:block>
</xsl:template>

<xsl:template match="nombre">
    <fo:block font-size="11pt"
        font-weight="bold"
        font-family="sans-serif"
        line-height="13pt"
        space-after.optimum="2pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

<xsl:template match="apellidos">
    <fo:block font-size="11pt"
        font-weight="bold"
        font-family="sans-serif"
        line-height="13pt"
        space-after.optimum="2pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

<xsl:template match="telefono">
    <fo:block font-size="10pt"
        font-family="sans-serif"
        line-height="15pt"
        space-after.optimum="3pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

<xsl:template match="movil">
    <fo:block font-size="10pt"
        font-family="sans-serif"
        line-height="15pt"
        space-after.optimum="3pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

<xsl:template match="email">
    <fo:block font-size="10pt"
        font-family="sans-serif"
        line-height="15pt"
        space-after.optimum="3pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

<xsl:template match="web">
    <fo:block font-size="10pt"
        font-family="sans-serif"
        line-height="15pt"
        space-after.optimum="3pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

<xsl:template match="direccion">
    <fo:block font-size="10pt"
```

```
        font-family="sans-serif"
        line-height="15pt"
        space-after.optimum="3pt"
        text-align="justify">
    <xsl:value-of select="."/>
</fo:block>
</xsl:template>

<xsl:template match="localidad">
    <fo:block font-size="10pt"
        font-family="sans-serif"
        line-height="15pt"
        space-after.optimum="3pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

<xsl:template match="cp">
    <fo:block font-size="10pt"
        font-family="sans-serif"
        line-height="15pt"
        space-after.optimum="3pt"
        text-align="justify">
        <xsl:value-of select="."/>
    </fo:block>
</xsl:template>

</xsl:stylesheet>
```

Generando el pdf: `agendapdf.sh`

```
fop -xsl agenda-pdf.xsl -xml agenda.xml -pdf agenda.pdf
```

6. Generando html con xml: `agendahtml.sh`

Obtenemos la fecha y la hora del sistema para xml: `time.sh`

```
echo "<?xml version=\"1.0\" encoding=\"ISO-8859-1\" ?>" > time.xml
echo "<time>$(date +%H:%M) $(date +%d-%m-%Y)</time> ">> time.xml
```

Hoja de estilos: `agenda-html.xsl`

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

    <xsl:template match="agenda">
        <html>
            <head>
                <title>Agenda</title>
            </head>
            <body>
                <h1>Agenda</h1>
                <h2>Ordenado por nombre.</h2>

                <!-- http://www.dpawson.co.uk/xsl/sect4/dateTime.html-->
                <xsl:variable name="time" select="document('time.xml')/time"/>

                <xsl:value-of select="$time"/>

                <table border="1" width="100" height="100">
                    <tr>
                        <th>Nombre</th>
                        <th>Apellidos</th>
                        <th>Telefono</th>
                        <th>Movil</th>
                        <th>Email</th>
```

```

        <th>Direccion</th>
        <th>Localidad</th>
        <th>CP</th>
        <th>Web</th>
    </tr>
    <xsl:apply-templates>
    <xsl:sort select="nombre"/>
    </xsl:apply-templates>
</table>
</body>
</html>
</xsl:template>

<xsl:template match="persona">
    <tr>
        <td width="10%" height="19"><font color="blue" size="2"><xsl:apply-templates select="nombre"/></font></td>
        <td width="10%" height="19"><font color="blue" size="2"><xsl:apply-templates select="apellidos"/></font></td>
        <td width="10%" height="19"><font color="red" size="2"><xsl:apply-templates select="telefono"/></font></td>
        <td width="10%" height="19"><font color="red" size="2"><xsl:apply-templates select="movil"/></font></td>
        <td width="10%" height="19"><font color="purple" size="2"> <xsl:apply-templates select="email"/></font></td>
        <td width="10%" height="19"><font color="green" size="1"><xsl:apply-templates select="direccion"/></font></td>
        <td width="10%" height="19"><font color="green" size="1"><xsl:apply-templates select="localidad"/></font></td>
        <td width="10%" height="19"><font color="green" size="1"><xsl:apply-templates select="cp"/></font></td>
        <td width="10%" height="19"><font color="purple" size="1"> <xsl:apply-templates select="web"/> </font></td>
    </tr>
</xsl:template>

</xsl:stylesheet>

```

Generamos el fichero html: agendahtml.sh

```
java -jar /usr/share/java/xalan2.jar -IN agenda.xml -XSL agenda-html.xsl -OUT agenda.html
```

7. Generando tarjetas vcard con xml y perl: vcard.pl

Estas tarjetas se pueden mandar al movil para tener los datos de las personas.
 Generaremos las tarjetas con: vcard.pl

```

#!/usr/bin/perl
# http://www.merelo.net/tutoperl/xml.htm ☺
# http://www.omerique.net/twiki/pub/TIC/LenguajePERL/tutorial-print.html
# Instalar perl: perl -MCPAN -e 'install XML::Simple'
print "*****\n";
use XML::Simple;
my $agenda = XMLin("agenda.xml");
print "Generando Tarjetas";
for( @{$agenda->{persona}} ) {
    print "tarjetas/${_}>{nombre} ${_}>{apellidos}.vcf\n";
    open SALIDA, ">tarjetas/${_}>{nombre} ${_}>{apellidos}.vcf";
    print SALIDA "BEGIN:VCARD\n";
    print SALIDA "VERSION:2.1\n";
    print SALIDA "FN:${_}>{nombre} ${_}>{apellidos}\n";
    print SALIDA "N:${_}>{apellidos};${_}>{nombre}\n";
    if (length($_->{telefono})!=0) {print SALIDA "TEL;VOICE:${_}>{telefono}\n";}
    if (length($_->{movil})!=0) {print SALIDA "TEL;CELL:${_}>{movil}\n";}
    if (length($_->{email}) !=0) {print SALIDA "EMAIL:${_}>{email}\n";}

    if (length($_->{direccion}) !=0) {print SALIDA "ADR;HOME;;;${_}>{direccion}";}
    if (length($_->{localidad}) !=0) {print SALIDA ";;${_}>{localidad}";}
    if (length($_->{cp}) !=0) {print SALIDA ";;;${_}>{cp}";}
    print SALIDA "\n";

    if (length($_->{web}) !=0) {print SALIDA "URL;HOME:${_}>{web}\n";}
    print SALIDA "END:VCARD\n";
    close SALIDA;
}
print "*****\n";

```


8. Hacer un cuadernillo tipo agenda alfabetica:

Mediante perl creamos el fichero: cuadernillo_datos.tex, que construye un fichero en latex, para el librito.

```
#!/usr/bin/perl
use XML::Simple;
my $agenda = XMLin("agenda.xml");

#####
sub linea1{
    $n=1;
    printf "\n";
    while ($n<10) {
        {
            print "-";
            $n=$n+1;
        }
    }
    printf "\n";
}
#####

sub linea{
    print " & \\\\ \n";
    print " & \\\\ \hline \n";
}

#####MAIN #####

for($i="65";$i<"90";$i++)
{
    print "\newpage\n";
    $c=0;
    $char = chr($i);
    print "\marginpar{\begin{Huge}".$char."\end{Huge} } \n \n";
    printf "\begin{tabular}{ p{6cm} p{6cm} } \n";
    printf "\\\\ \hline \n";
    print "\\textbf{NOMBRE} & \\textbf{APELLIDOS} \\\\ \n";
    print "\\textbf{TELEFONO} & \\textbf{MOVIL} \\\\ \hline \n";
    for( @{$agenda->{persona}} ) {
        $imp= ($->{nombre} =~ /^Q$char/) and ( ( -z $->{telefono} ) or ( -z $->{movil} ) );
        if ($imp){
            printf "\n";
            print "$->{nombre} & $->{apellidos} \\\\ \n";
            print "$->{telefono} & $->{movil} \\\\ \hline \n";
            #do linea();
            $c=$c+1
        }
    }

    while ($c < 20 ){
        do linea();
        $c=$c+1;
    }
    printf "\end{tabular}";
}

```

Creamos el fichero plantilla: cuadernillo.tex

```
\documentclass[10pt,a4paper,onecolumn]{article}
\usepackage[spanish, activeacute]{babel} %Españolización
\usepackage[latin1]{inputenc} %Letras con acentso, eñes
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{eurosym}
\usepackage{helvet}
\def\rmdefault{\sfdefault}

```

```

\usepackage{makeidx} % Indice de materias
\makeindex % Indice de materias
\usepackage{hyperref}
\usepackage{listings}
\usepackage{fancyhdr}
\pagestyle{fancy}

\usepackage{ifpdf}
\ifpdf
  % Margenes normales:
  \usepackage[a4paper, pdftex, top=3cm, bottom=3cm, left=3cm, right=3cm]{geometry}
  \usepackage[a4paper, pdftex]{geometry}
  \usepackage[pdftex]{graphicx}
\else
  \usepackage[dvips]{geometry}
  \usepackage[dvips]{graphicx}
\fi

\ifpdf
\else
  %% Declaraciones 'utiles para los gr'aficos
  \graphicspath{{figuras/}{fig2/}{fig3/}{fig4/}{fig5/}}
  \DeclareGraphicsExtensions{.eps,.ps,.eps.gz,.ps.gz,.jpg,.gif,.png}
  \DeclareGraphicsRule{.jpg}{eps}{.jpg.bb}{'jgptops #1}
  \DeclareGraphicsRule{.gif}{eps}{.gif.bb}{'giftops #1}
  \DeclareGraphicsRule{.png}{eps}{.png.bb}{'pngtops #1}
\fi
\title{\titulo}
\date{}
\parskip 1ex %Lineas entre prarrafos
\usepackage{lastpage} %Ultima pagina.
\addtolength{\headwidth}{\marginparwidth}
\addtolength{\headwidth}{\marginparsep}
\fancypagestyle{plain}
{%
  \fancyhead[l]{\today}
  \fancyhead[r]{\titulo}
  \fancyhead[c]{}
  \renewcommand{\headrulewidth}{0.4pt}
  \fancyfoot[l]{Por Paco Aldarias}
  \fancyfoot[c]{}
  \fancyfoot[r]{\thepage/\pageref{LastPage}}
  \renewcommand{\footrulewidth}{0.4pt}
}

\lhead{\today}
\chead{}
\rhead{
\titulo}
\renewcommand{\headrulewidth}{0.4pt}
\lfoot{Por Paco Aldarias}
\cfoot{}
\rfoot{\thepage/\pageref{LastPage}}
\renewcommand{\footrulewidth}{0.4pt}

%*****
\begin{document}
\newcommand{\titulo}{AGENDA TELEFONICA}
\noindent
\input{cuadernillo_datos}
\end{document}

```

Creamos el script que transforma el fichero en un cuadernillo: `cuadernillo.sh`

```
perl cuadernillo.pl > cuadernillo_datos.tex
latex cuadernillo
```

```
dvips -o cuadernillo.ps cuadernillo.dvi
psbook cuadernillo.ps | psnup -2 -pa4 > cuadernillo1.ps
ps2pdf cuadernillo1.ps
```

9. Todo junto: agenda.sh

Aqui tenemos todo reunido para generar todos los archivos: agenda.sh

```
echo agenda en pl
./agenda.pl

echo generando pdf en latex
sh latex.sh

echo generando pdf en xml
sh agendapdf.sh

echo generando html en xml
sh time.sh
sh agendahtml.sh

echo generando tarjetas
rm -r -f vcard/*
rm -r -f tarjetas/*
perl vcard.pl

echo generando cuadernillo
sh cuadernillo.sh
```