Essential Skills. Lab 2

Ali Abdulmadzhidov

09 September 2016

1 Debian packagins system

1. How does it work? Debian package system has various different level tools for user. The base of package system is dpkg - Debian package manager. In spite of man page says that the description of steps in installing or removing process is inadequate, i'll try.

   *.deb - Debian package

   Debian package contains three sections:

   Global header Contains version of debian-binary package. Current is 2.0

   Control section Meta information about package. Confiles, preinst, postinst, prerm, postrm scripts, list of dependecies, md5 hashsumm, brief description.

   Data section Includes installable files

   Installation

   dpkg -i

   1 Extract of control files of the package
   2 If another version of package is already installed, execute prerm script of the outdated package
   3 If package has preinst script, execute it.
   4 Run postrm script of old package
   5 Unpack new, and at the same time backup old files.
   6 Configure package

   Configuration

   dpkg -configure

   1 Unpack conffiles, and backup old conffiles.
   2 Run postinit script

   Removing

   dpkg -remove Removes an installed package, but lefts connfiles

   1 Run prerm script
   2 Remove isntalled files
   3 Run posterm script

Purge

dpkg -purge Purges (removes) an installed or removed package with connfiles.

    1 Remove package

    2 Run posterm script

```
wget ftp://ftp.sendmail.org/pub/sendmail/sendmail.8.15.2.tar.gz
```

2. Cheking signature of file

```
wget ftp://ftp.sendmail.org/pub/sendmail/sendmail.8.15.2.tar.gz.sig
wget http://www.sendmail.com/sm/open_source/security/pgp_keys/sendmail2015.asc
gpg --import sendmail2015.asc
gpg --verify sendmail.8.15.2.tar.gz.sig
```

3. Extract souces

```
tar -xzvf sendmail.8.15.2.tar.gz
```

4. Read readme file first... No, really for the first time I just builded it and can't start. Then remove all staff and read readme file.

```
cat README
```

5. Add

```
sudo apt install libapache2-mod-php
```

6. Restarting apache2.

```
sudo service apache2 restart
```

7. Go to http://localhost/dokuwiki/doku.php and add our first page about Federative Wiki

2   XML entities creating

1. XML + DDT

```
<!DOCTYPE root
[
<!ELEMENT root (animal)>
<!ELEMENT animal (name,skin,voice)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT skin (#PCDATA)>
<!ELEMENT voice (#PCDATA)>
```

```
            <!ENTITY  skin "have skin">
            <!ENTITY  fur "have fur">
            <!ENTITY  feathers "have feathers">
            <!ENTITY  bark "bark">
            <!ENTITY  mur "mur">
            <!ENTITY  bee "bee">
            <!ENTITY  brr "brr">
            <!ENTITY  mooo "mooo">
            <!ENTITY  oink "oink">
            <!ENTITY  whistle "whistle">
            <!ENTITY  no_voice "no_voice">
            ]>
            <root>
    <animal>
                <name>cow</name>
                <skin>&skin;</skin>
                <voice>&mooo;</voice>
        </animal>
        <animal>
                <name>sheep</name>
                <skin>&fur;</skin>
                <voice>&bee;</voice>
        </animal>
        <animal>
                <name>horse</name>
                <skin>&skin;</skin>
                <voice>&brr;</voice>
        </animal>
        <animal>
                <name>pig</name>
                <skin>&skin;</skin>
                <voice>&oink;</voice>
        </animal>
        <animal>
                <name>mockingbird</name>
                <skin>&feathers;</skin>
                <voice>&whistle;</voice>
        </animal>
        <animal>
                <name>eal</name>
                <skin>&feathers;</skin>
                <voice>&whistle;</voice>
        </animal>
    </root>
```

2. XS SCHEME

```
    <?xml version="1.0" encoding="utf-8"?>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="root">
        <xs:complexType>
        <xs:sequence>
            <xs:element name="animal" maxOccurs="unbounded">
```

```
                    <xs:complexType>
                    <xs:sequence>
                        <xs:element name="name" type="xs:string"/>
                        <xs:element name="voice" type="xs:string"/>
                        <xs:element name="skin" type="xs:string"/>
                    </xs:sequence>
                    </xs:complexType>
                </xs:element>
            <xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:schema>
```

XSLT and CSS

1. XSLT

```xml
<?xml version='1.0' encoding='UTF-8'?>
<xsl:stylesheet version='1.0' xmlns:xsl='http://www.w3.org/1999/XSL/Transform'>
<xsl:template match='/'>
  <html>
  <body>
  <h2>Animals</h2>
    <table border='1'>
      <tr>
        <th style='text-align:left'>Name</th>
        <th style='text-align:left'>Skin</th>
        <th style='text-align:left'>Voice</th>
      </tr>
      <xsl:for-each select='root/animal'>
      <tr>
        <td class='name-vl'><xsl:value-of select='name'/></td>
        <td class='skin-vl'><xsl:value-of select='skin'/></td>
        <td class='voice-vl'><xsl:value-of select='voice'/></td>
      </tr>
      </xsl:for-each>
    </table>
  </body>
  </html>
</xsl:template>
</xsl:stylesheet>
```

2. CSS

```css
td {
    background: #eeccbb;
}
th {
    background: #ffff00;
}
.name-vl {
```

```
        color: #0f0;
}
.name-vl:hover {
        color: #00f;
}
.voice-vl {
        color: #00f;
}
.voice-vl:hover {
        color: #0f0;
}
.skin-vl {
        color: #f00;
}
.skin-vl:hover {
        color: #f0f;
}
```