

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-<!--
|
| XSLT REC Compliant Version of IE5 Default Stylesheet
|
| Original version by Jonathan Marsh (jmarsh@microsoft.com)
| http://msdn.microsoft.com/xml/samples/defaultss/defaultss.xsl
|
| Conversion to XSLT 1.0 REC Syntax by Steve Muench (smuench@oracle.com)
|
+
-->
-<xsl:stylesheet version="1.0">
  <xsl:output indent="no" method="html"/>
  <xsl:template match="/">
    <HTML>
      <HEAD>
        <SCRIPT>
          <xsl:comment>
            function f(e){ if (e.className=="ci") { if (e.children(0).innerText.indexOf("\n")>0) fix(e,"cb"); } if
            (e.className=="di") { if (e.children(0).innerText.indexOf("\n")>0) fix(e,"db"); } e.id=""; } function
            fix(e,cl){ e.className=cl; e.style.display="block"; j=e.parentElement.children(0); j.className="c";
            k=j.children(0); k.style.visibility="visible"; k.href="#"; } function ch(e) {
            mark=e.children(0).children(0); if (mark.innerText=="+") { mark.innerText="-"; for (var
            i=1;i<e.children.length;i++) { e.children(i).style.display="block"; } } else if (mark.innerText=="-") {
            mark.innerText="+"; for (var i=1;i<e.children.length;i++) { e.children(i).style.display="none"; } } }
            function ch2(e) { mark=e.children(0).children(0); contents=e.children(1); if (mark.innerText=="+") {
            mark.innerText="-"; if (contents.className=="db"||contents.className=="cb") {
            contents.style.display="block"; } else { contents.style.display="inline"; } } else if
            (mark.innerText=="-") { mark.innerText="+"; contents.style.display="none"; } } function cl() {
            e=window.event.srcElement; if (e.className!="c") { e=e.parentElement; if (e.className!="c") {
            return; } } e=e.parentElement; if (e.className=="e") { ch(e); } if (e.className=="k") { ch2(e); } }
            function ex(){} function h(){window.status=" "; document.onclick=cl;
          </xsl:comment>
        </SCRIPT>
      <STYLE>
        BODY {font:x-small 'Verdana'; margin-right:1.5em} .c {cursor:hand} .b {color:red; font-family:'Courier
        New'; font-weight:bold; text-decoration:none} .e {margin-left:1em; text-indent:-1em; margin-right:1em}
        .k {margin-left:1em; text-indent:-1em; margin-right:1em} .t {color:#990000} .xt {color:#990099} .ns
        {color:red} .dt {color:green} .m {color:blue} .tx {font-weight:bold} .db {text-indent:0px; margin-
        left:1em; margin-top:0px; margin-bottom:0px;padding-left:.3em; border-left:1px solid #CCCCCC;
        font:small Courier} .di {font:small Courier} .d {color:blue} .pi {color:blue} .cb {text-indent:0px; margin-
        left:1em; margin-top:0px; margin-bottom:0px;padding-left:.3em; font:small Courier; color:#888888} .ci
        {font:small Courier; color:#888888} PRE {margin:0px; display:inline}
      </STYLE>
    </HEAD>
    <BODY class="st">
      <xsl:apply-templates/>
    </BODY>
  </HTML>
</xsl:template>
-<xsl:template match="processing-instruction(">
```

```

- <DIV class="e">
- <SPAN class="b">
- <xsl:call-template name="entity-ref">
  <xsl:with-param name="name">nbsp</xsl:with-param>
</xsl:call-template>
</SPAN>
- <SPAN class="m">
  <xsl:text><?</xsl:text>
</SPAN>
- <SPAN class="pi">
  <xsl:value-of select="name(.)"/>
  <xsl:text> </xsl:text>
  <xsl:value-of select="."/>
</SPAN>
- <SPAN class="m">
  <xsl:text>?></xsl:text>
</SPAN>
</DIV>
</xsl:template>
- <xsl:template match="processing-instruction('xml')">
- <DIV class="e">
- <SPAN class="b">
- <xsl:call-template name="entity-ref">
  <xsl:with-param name="name">nbsp</xsl:with-param>
</xsl:call-template>
</SPAN>
- <SPAN class="m">
  <xsl:text><?</xsl:text>
</SPAN>
- <SPAN class="pi">
  <xsl:text>xml </xsl:text>
- <xsl:for-each select="@*">
  <xsl:value-of select="name(.)"/>
  <xsl:text>="</xsl:text>
  <xsl:value-of select="."/>
  <xsl:text>" </xsl:text>
</xsl:for-each>
</SPAN>
- <SPAN class="m">
  <xsl:text>?></xsl:text>
</SPAN>
</DIV>
</xsl:template>
- <xsl:template match="@*">
- <SPAN>
- <xsl:attribute name="class">
- <xsl:if test="xsl:*/@*">
  <xsl:text>x</xsl:text>
</xsl:if>
  <xsl:text>t</xsl:text>
</xsl:attribute>
  <xsl:value-of select="name(.)"/>
</SPAN>
  <SPAN class="m">="</SPAN>
- <B>

```

```

    <xsl:value-of select="."/>
  </B>
  <SPAN class="m"></SPAN>
</xsl:template>
- <xsl:template match="text()">
  - <DIV class="e">
    <SPAN class="b"> </SPAN>
    - <SPAN class="tx">
      <xsl:value-of select="."/>
    </SPAN>
  </DIV>
</xsl:template>
- <xsl:template match="comment()">
  - <DIV class="k">
    - <SPAN>
      <A STYLE="visibility:hidden" class="b" onclick="return false" onfocus="h()">-</A>
    - <SPAN class="m">
      <xsl:text><!--</xsl:text>
    </SPAN>
  </SPAN>
  - <SPAN class="ci" id="clean">
    - <PRE>
      <xsl:value-of select="."/>
    </PRE>
  </SPAN>
  - <SPAN class="b">
    - <xsl:call-template name="entity-ref">
      <xsl:with-param name="name">nbsp</xsl:with-param>
    </xsl:call-template>
  </SPAN>
  - <SPAN class="m">
    <xsl:text>--></xsl:text>
  </SPAN>
  <SCRIPT>f(clean);</SCRIPT>
</DIV>
</xsl:template>
- <xsl:template match="*">
  - <DIV class="e">
    - <DIV STYLE="margin-left:1em;text-indent:-2em">
      - <SPAN class="b">
        - <xsl:call-template name="entity-ref">
          <xsl:with-param name="name">nbsp</xsl:with-param>
        </xsl:call-template>
      </SPAN>
      <SPAN class="m"><</SPAN>
    - <SPAN>
      - <xsl:attribute name="class">
        - <xsl:if test="xsl:*">
          <xsl:text>x</xsl:text>
        </xsl:if>
        <xsl:text>t</xsl:text>
      </xsl:attribute>
      <xsl:value-of select="name(.)"/>
    - <xsl:if test="@*">
      <xsl:text> </xsl:text>

```

```

        </xsl:if>
    </SPAN>
    <xsl:apply-templates select="@*" />
    - <SPAN class="m">
        <xsl:text>/></xsl:text>
    </SPAN>
</DIV>
</DIV>
</xsl:template>
- <xsl:template match="*[node()]">
    - <DIV class="e">
        - <DIV class="c">
            <A class="b" href="#" onclick="return false" onfocus="h()">-</A>
            <SPAN class="m"></SPAN>
        - <SPAN>
            - <xsl:attribute name="class">
                - <xsl:if test="xsl:*">
                    <xsl:text>x</xsl:text>
                </xsl:if>
                <xsl:text>t</xsl:text>
            </xsl:attribute>
            <xsl:value-of select="name(.)" />
        - <xsl:if test="@*">
            <xsl:text> </xsl:text>
        </xsl:if>
    </SPAN>
    <xsl:apply-templates select="@*" />
    - <SPAN class="m">
        <xsl:text>></xsl:text>
    </SPAN>
</DIV>
- <DIV>
    <xsl:apply-templates />
    - <DIV>
        - <SPAN class="b">
            - <xsl:call-template name="entity-ref">
                <xsl:with-param name="name">nbsp</xsl:with-param>
            </xsl:call-template>
        </SPAN>
        - <SPAN class="m">
            <xsl:text></xsl:text>
        </SPAN>
        - <SPAN>
            - <xsl:attribute name="class">
                - <xsl:if test="xsl:*">
                    <xsl:text>x</xsl:text>
                </xsl:if>
                <xsl:text>t</xsl:text>
            </xsl:attribute>
            <xsl:value-of select="name(.)" />
        </SPAN>
        - <SPAN class="m">
            <xsl:text>></xsl:text>
        </SPAN>
    </DIV>

```

```

    </DIV>
  </DIV>
</xsl:template>
- <xsl:template match="*[text() and not (comment() or processing-instruction())]">
  - <DIV class="e">
    - <DIV STYLE="margin-left:1em;text-indent:-2em">
      - <SPAN class="b">
        - <xsl:call-template name="entity-ref">
          <xsl:with-param name="name">nbsp</xsl:with-param>
        </xsl:call-template>
      </SPAN>
      - <SPAN class="m">
        <xsl:text><</xsl:text>
      </SPAN>
      - <SPAN>
        - <xsl:attribute name="class">
          - <xsl:if test="xsl:*">
            <xsl:text>x</xsl:text>
          </xsl:if>
          <xsl:text>t</xsl:text>
        </xsl:attribute>
        <xsl:value-of select="name(.)"/>
      - <xsl:if test="@*">
        <xsl:text> </xsl:text>
      </xsl:if>
    </SPAN>
    <xsl:apply-templates select="@*" />
  - <SPAN class="m">
    <xsl:text>></xsl:text>
  </SPAN>
  - <SPAN class="tx">
    <xsl:value-of select="."/>
  </SPAN>
  <SPAN class="m"></SPAN>
- <SPAN>
  - <xsl:attribute name="class">
    - <xsl:if test="xsl:*">
      <xsl:text>x</xsl:text>
    </xsl:if>
    <xsl:text>t</xsl:text>
  </xsl:attribute>
  <xsl:value-of select="name(.)"/>
</SPAN>
- <SPAN class="m">
  <xsl:text>></xsl:text>
</SPAN>
</DIV>
</DIV>
</xsl:template>
- <xsl:template match="*[*]" priority="20">
  - <DIV class="e">
    - <DIV STYLE="margin-left:1em;text-indent:-2em" class="c">
      <A class="b" href="#" onclick="return false" onfocus="h()">-</A>
      <SPAN class="m"><</SPAN>
    - <SPAN>

```

```

- <xsl:attribute name="class">
- <xsl:if test="xsl:*">
  <xsl:text>x</xsl:text>
</xsl:if>
  <xsl:text>t</xsl:text>
</xsl:attribute>
<xsl:value-of select="name(.)"/>
- <xsl:if test="@*">
  <xsl:text> </xsl:text>
</xsl:if>
</SPAN>
<xsl:apply-templates select="@*" />
- <SPAN class="m">
  <xsl:text>></xsl:text>
</SPAN>
</DIV>
- <DIV>
  <xsl:apply-templates />
- <DIV>
- <SPAN class="b">
  - <xsl:call-template name="entity-ref">
    <xsl:with-param name="name">nbsp</xsl:with-param>
  </xsl:call-template>
</SPAN>
- <SPAN class="m">
  <xsl:text></</xsl:text>
</SPAN>
- <SPAN>
  - <xsl:attribute name="class">
    - <xsl:if test="xsl:*">
      <xsl:text>x</xsl:text>
    </xsl:if>
      <xsl:text>t</xsl:text>
    </xsl:attribute>
    <xsl:value-of select="name(.)"/>
  </SPAN>
- <SPAN class="m">
  <xsl:text>></xsl:text>
</SPAN>
</DIV>
</DIV>
</xsl:template>
- <xsl:template name="entity-ref">
  <xsl:param name="name" />
  <xsl:text disable-output-escaping="yes">&</xsl:text>
  <xsl:value-of select="$name" />
  <xsl:text>;</xsl:text>
</xsl:template>
</xsl:stylesheet>

```