

Natural Web I/O Interface Client

The Natural Web I/O Interface client is used in a remote development environment to display non-GUI information which contains Unicode characters. This is different from the local environment where the output is shown in an output window.

This chapter covers the following topics:

- About the Natural Web I/O Interface Client
- Prerequisites for the Natural Development Server
- Working with the Natural Web I/O Interface Client
- Modifying Color and Field Attributes

See also *Differences between Natural Web I/O Interface Client and Terminal Emulation* in the *Natural Web I/O Interface* documentation.

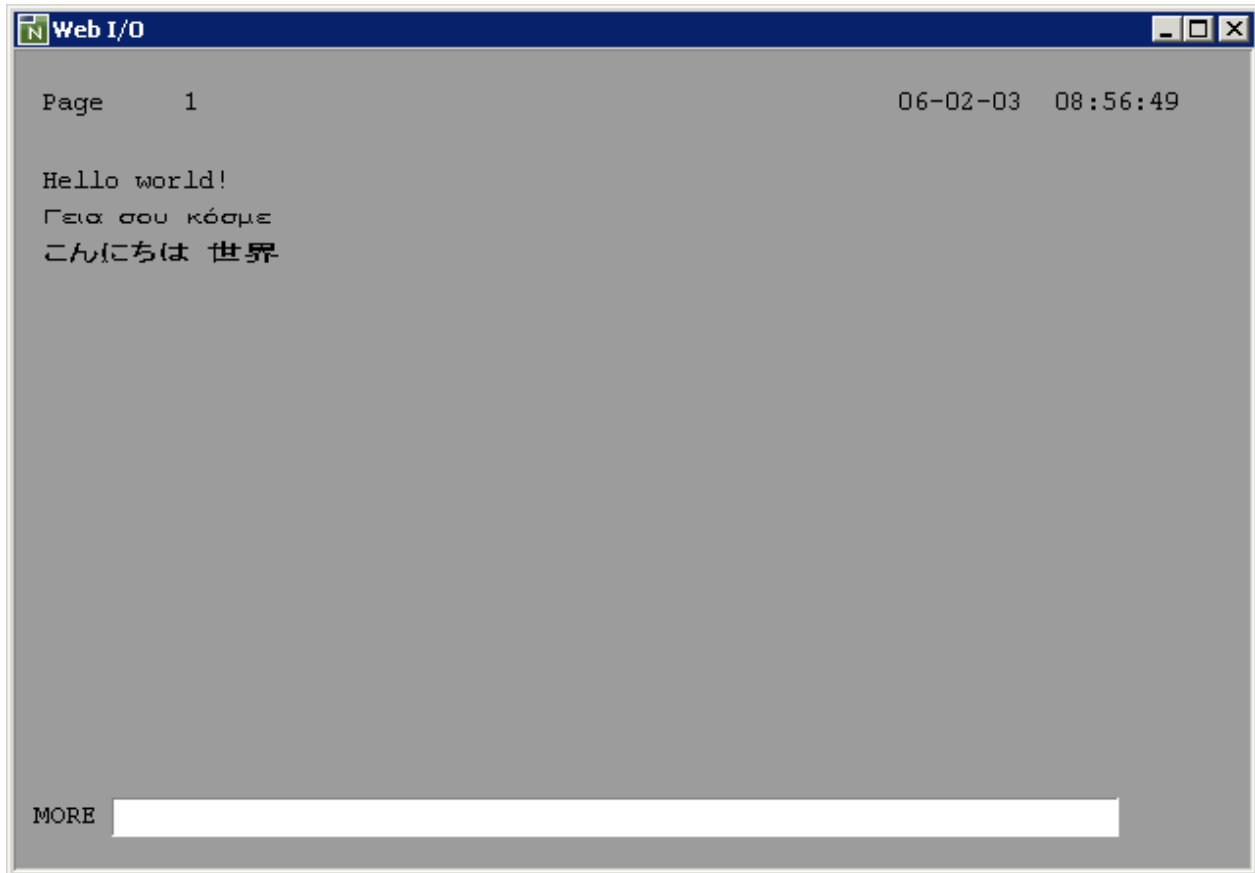
Important:

For running the Natural Web I/O Interface client, the Microsoft Windows Personal Firewall must be deactivated. See *Configuring the Microsoft Windows Personal Firewall to Run Natural* in the *Operations* documentation for further information.

About the Natural Web I/O Interface Client

When specific parameters have been set on the Natural Development Server (see below), the Natural Web I/O Interface client is used instead of the terminal emulation window (which is not Unicode-enabled).

The following example shows a Natural Web I/O Interface client containing the output of a program that has been executed in a remote environment.



The Natural Web I/O Interface client enables you to test a complete application, including all of its character output, without having to leave Natural Studio. A screen output of an executed program automatically causes the Natural Web I/O Interface client to be opened and to be closed again when the program terminates (logically, the development server session reaches the NEXT level).

Utilities for which Natural Studio does not offer a graphical user interface in the remote environment are also made available by the Natural Web I/O Interface client. Entering the corresponding system command in Natural Studio's command line automatically opens a Natural Web I/O Interface client.

When you close the Natural Web I/O Interface client manually, this corresponds to pressing ESC in the application.

Prerequisites for the Natural Development Server

So that the Natural Web I/O Interface client can be invoked, the Natural Development Server (NDV) has to be configured as follows:

- **UNIX and OpenVMS**

If you want to use the Natural Web I/O Interface client in a remote UNIX or OpenVMS environment, the profile parameter `WEBIO` must be set to "ON" on the NDV server. See *Configuration Utility* in the Natural for UNIX and Natural for OpenVMS documentation.

- **Mainframe**

If you want to use the Natural Web I/O Interface client in a remote mainframe environment, the NDV configuration parameter `TERMINAL_EMULATION` must be set to "WEBIO" on the NDV server. See

NDV Configuration Parameters in the Natural Development Server documentation. Moreover, the Web I/O terminal converter module NATWEB must be linked to the Natural nucleus. The Natural profile parameter TMODEL can be used to determine the user screen size.

- **Windows**

In a remote Windows environment, the Natural Web I/O Interface client is always used, regardless of the setting of the profile parameter WEBIO.

Working with the Natural Web I/O Interface Client

When the Natural output is shown in the Natural Web I/O Interface client, any defined PF keys are shown as buttons in the window (instead of the key prompting lines which are normally shown in the native UNIX, OpenVMS or mainframe environment). You can either use these buttons or you can press the corresponding keys on your keyboard (for example, F5 for PF5).

Note:

A refresh function is not executed when you press F5.

You can enter your input as usual. You can also use the usual Windows copy-and-paste functionality.

Modifying Color and Field Attributes

In the Natural Web I/O Interface client, the color attributes for the Natural fields (foreground colors), the background color, and the rendering of the Natural field attributes are controlled using an XSLT file. The name of this XSLT file is *natunistyle.xml*. After the installation of Natural for Windows, this file can be found in the *bin* folder.

The following topics are covered below:

- Setting the Foreground Colors
- Setting the Background Color
- Setting the Reverse Attribute
- Setting the Underline and Blinking Attributes
- Setting the Cursive/Italic Attribute
- Setting the Intensified Attribute

Setting the Foreground Colors

The foreground colors for the font are mapped in the following section of the XSLT file:

```
<!--      Map foreground colors      -->
<xsl:variable name="forecol">
  <xsl:choose>
    <xsl:when test="@forecol='Red'">red</xsl:when>
    <xsl:when test="@forecol='Green'">LimeGreen</xsl:when>
    <xsl:when test="@forecol='Blue'">DodgerBlue</xsl:when>
    <xsl:when test="@forecol='White'">white</xsl:when>
    <xsl:when test="@forecol='Pink'">fuchsia</xsl:when>
```

```

<xsl:when test="@forecol='Turquoise' ">aqua</xsl:when>
<xsl:when test="@forecol='Yellow' ">yellow</xsl:when>
<xsl:when test="@forecol='Black' ">white</xsl:when>
<xsl:when test="@forecol='Transparent' ">
  <xsl:choose>
    <xsl:when test="@intensified='True' ">aqua</xsl:when>
    <xsl:otherwise><xsl:value-of select="$defforecolor"/></xsl:otherwise>
  </xsl:choose>
</xsl:when>
<xsl:otherwise><xsl:value-of select="$defforecolor"/></xsl:otherwise>
</xsl:choose>
</xsl:variable>

```

For example, the color attribute Red of the Natural screen is mapped to the CSS color "red":

```
<xsl:when test="@forecol='Red' ">red</xsl:when>
```

It is also possible to map the color attributes to RGB values. For example:

```
<xsl:when test="@forecol='Red' ">#FF0000</xsl:when>
```

For the color attribute Transparent, it is checked whether the attribute `intensified` is set. If this is true, the color "aqua" is used. If not, the global variable `$defforecolor`, which is defined at the beginning of the XSLT file, is used.

More information on CSS colors can be found in the internet under http://www.w3schools.com/css/css_colors.asp.

Setting the Background Color

The background color is set with the global variable `defbackcolor`, which is defined at the beginning of the XSLT file.

```

<!-- Background color -->
<xsl:variable name="defbackcolor" select="'black' "/>

```

Setting the Reverse Attribute

The field attribute for reverse video is set in the following section of the XSLT file, which occurs directly after the section in which the foreground colors are mapped:

```

<!--      START SET FOREGROUND AND BACKGROUND COLORS FOR REVERSE ATTRIBUTE      -->
<!--      Set foreground colors      -->
<xsl:variable name="ForeCol">
  <xsl:choose>
    <xsl:when test="@reverse='True' ">color:<xsl:value-of select="$deffbackcolor"/></xsl:when>
    <xsl:otherwise>color:<xsl:value-of select="$forecol"/></xsl:otherwise>
  </xsl:choose>
</xsl:variable>

<!--      Set background colors      -->
<xsl:variable name="BackCol">
  <xsl:choose>
    <xsl:when test="@reverse='True' ">background-color:<xsl:value-of select="$forecol"/></xsl:when>
    <xsl:otherwise>background-color:<xsl:value-of select="$deffbackcolor"/></xsl:otherwise>
  </xsl:choose>
</xsl:variable>
<!--      END SET FOREGROUND AND BACKGROUND COLORS FOR REVERSE ATTRIBUTE      -->

```

When the `reverse` attribute is set to true, the foreground color is set to the value of `$defbackcolor` and the background color gets the value of the foreground color.

When the `reverse` attribute is not set to true, the foreground color gets the value of the mapped foreground color and the background color gets the value of `$defbackcolor`.

Setting the Underline and Blinking Attributes

The field attributes for underlined and blinking text are set in the following section of the XSLT file:

```
<!-- Set underline and blinking attributes -->
<xsl:variable name="TextDecoStyle">
  <xsl:choose>
    <xsl:when test="@underline='True'">text-decoration:underline;</xsl:when>
    <xsl:when test="@blinking='True'">text-decoration:blink;</xsl:when>
    <xsl:otherwise>text-decoration:normal;</xsl:otherwise>
  </xsl:choose>
</xsl:variable>
```

Since the Natural Web I/O Interface client uses the Internet Explorer which does not support blinking text, blinking text will not be displayed in the browser window.

Setting the Cursive/Italic Attribute

The field attribute for cursive/italic text is set in the following section of the XSLT file:

```
<!-- Set cursive attribute -->
<xsl:variable name="FontStyle">
  <xsl:choose>
    <xsl:when test="@italic='True'">font-style:italic;</xsl:when>
    <xsl:otherwise>font-style:normal;</xsl:otherwise>
  </xsl:choose>
</xsl:variable>
```

Setting the Intensified Attribute

The field attribute for intensified text is set in the following section of the XSLT file:

```
<!-- Set intensified attribute - draw text as bold -->
<xsl:variable name="FontWeight">
  <xsl:choose>
    <xsl:when test="@intensified='True'">font-weight:bolder</xsl:when>
    <xsl:otherwise>font-weight:normal;</xsl:otherwise>
  </xsl:choose>
</xsl:variable>
```