

Creating Websites in Hindi

Arun Maurya, Sumit Goswami, AK Tyagi,
Ashok Kumar, Mohinder Singh

Abstract

The article is intended to be a veritable encyclopedia of Hindi website development know-how and offers advice on dynamic font creation, embedding, publishing and the related methodology up to their pros and cons. It is one-step guide for the Hindi website development. It provides comprehensive instructions and real world examples, for special coverage of the topics on Hindi website designing and offers advice for developing a truly dynamic Hindi website, on parallel lines to which any multilingual website can be developed. This article is of interest to web developers interested in building dynamic, interactive Hindi websites and for those who are looking for a quick, effective way to build embedded fonts for the web.

1. INTRODUCTION

The Defence Research and Development Organisation (DRDO) has its website on the internet, for the last few years. The website is designed, developed and maintained by Defence Scientific Information & Documentation Centre (DESIDOC). The site is currently publishing its contents in English. Being a widely used site, efforts have been made to create its Hindi version so as to make it accessible to a larger number of interested users. Creation of a website, especially in Hindi, is a very complex task and needs command over language in which website will be designed, various fonts and other related technologies; this article covers the technical specifications and evaluation criteria for the Hindi Website and increases the curiosity to cruise into the technicality of developing the website in Hindi.

The paper is a detailed description of the technicalities involved in the development of a Hindi website so that a person having the basic knowledge of web development even can create the site in Hindi or any other regional language. This article is designed

keeping this in mind that the reader of this paper should not face any problem in coming up with his regional language website due to lack of proper literature and the technicalities at one place. Considering the problems faced by many users in designing of a multilingual website, this article was designed to pave a straight and flawless path for the readers.

The prerequisite for developing a Hindi website, is basic knowledge of web page designing which is not a difficult task. Designing of a web page is as simple as creating a document using Microsoft Word. There are many HTML editors which are used for web page development; some of the frequently & commonly used are Microsoft FrontPage and Macromedia Dream Weaver. Many HTML Editor are freely available on the internet. One can also design a web page using MS Word 2000.

2. NEED FOR MULTILINGUAL WEBSITE

English is not going to serve effectively in every place the world over. Multilingual sites are websites that have country-specific

language content. Paralleling the growth of the public internet is the spread of intranets and extranets throughout the global enterprise. Businesses are enabling their information systems, moving mission-critical business applications to powerful private networks based on web technologies. Systems for order processing, inventory tracking, supply chain management, and invoicing are moving to the web. Enabling systems to tap the power of the web offers a key competitive advantage to global companies who need to conduct business across geographies and time zones. In a global place where customers, suppliers, and co-workers are international and located worldwide, multilingual content is integral to effective and efficient communication.

As internet technologies permeate the enterprise and the web continues to evolve into a global mass medium, the profile of the internet user is rapidly changing. There is a direct correlation between increased international usage and decreased viability of the English-only web. Today users are coming online with all levels of technical and linguistic aptitudes, making multilingual content a key ingredient to a successful global web strategy.

World Wide Web (WWW) is a global medium and websites should also address the non-English-speaking audience, otherwise the efficacy of any web presence is reduced. Web users may first search for content in their native language and will switch to English only if they can't find the information they are seeking. Multilingual content adds a layer of richness and functionality to any site. Creating a multilingual website is one of the best ways to transcend linguistic barriers and 'directly speak' to the target viewer's language. Major search engines such as Yahoo, Lycos, and Excite offer a good case in point. They all allow for multilingual searching and the content within search categories is tailored to the interests of each place.

The basic architecture of site will also influence how one prioritise what to localise and for which place. The site may be database-driven, static, application-rich or

more likely some hybrid. It may contain frequently updated content published at predictable intervals, or may incorporate erratic or constant data streams. Publishing a multilingual site ensures that content will be accessible and acceptable to local visitors, allowing them to utilise the site.

3. TECHNICAL FEATURES

In delivering a multilingual website, apart from translation of images and text, there are also various technical considerations to take into account when providing multilingual content for visitor. One of the most important consideration is to be aware of how the visitor interacts with the site to find the information he needs. If a visitor can fill out a contact form, order items generated by a database, one should make sure that he can find the information that he needs for them, as well as effectively respond to their inquiries.

One does not need to completely localise all the content within a site. For example, a database used for online ordering that stores the size, color and styles of various products would not require localisation. The user interface would have to be translated in to the target place language, but the items the user is selecting and the items that are being referenced in the database would remain in English.

3.1 Browsers and Font

Visitors can see only that text in a web page, which uses those fonts, which are installed on the visitor's computer. For those pages published in English and other European languages, *Arial* and *Times New Roman* fonts are often used because they are standard fonts supplied with Microsoft Windows for PCs and also with Apple Macintosh computers. These fonts also support most European languages and the user can set their browser to their own language. Pages published in other languages such as Japanese or Russian use fonts supplied with the specific language version of Windows or Mac operating system. If a web page is designed using a font that is not installed on the visitor's computer, the browser will substitute another default font in

its place. For English this means that the text will still be meaningful even if the font has changed, but other languages like Hindi will often be displayed as meaningless English characters because the required characters for that language are missing.

3.2 Technologies for Publishing Hindi Website

To deliver a multilingual website, three methods are available. Using which one can publish his website in Hindi. But each of them has some associated disadvantages of which a detailed description is given in following paragraphs. The technologies for publishing any multilingual website are:

- Image Display
- Font Downloading
- Font Embedding

3.2.1 Image Display

This is the simplest solution for presenting Hindi script on a web page. A section or page of typed or hand-written Hindi can be scanned into image-processing software and made into an image file for the web. The most popular image file format for this is GIF (graphical image format) because the number of colours in the image can be reduced to make the image file size smaller. An image of text could be reduced to just two colours, black and white, but because the Hindi script is quite intricate it may be better to use 16 colours so that the outlines of the characters is smooth (a process known as anti-aliasing).

Disadvantages

A large amount of text would require a large image file or number of files, which would make the page slow to load in the user's browser. Any alterations or updation in the text require a completely new revision to be produced, scanned, and made into an image file. Although creating an image map around words which can make links, the image would require more memory if these link words were to be in another colour. Image files of text generally offer poor reproduction in printed format so the user will not be able to make a good print out of the page.

Large images, video clips, and flash movies can take an inordinate amount of time to download for any users who may still be using a 14.4 or 28.8 kbps modem. Thus a slow modem with poor telephone lines and costly internet access fees means that site may suddenly have a very frustrated user. Good web design takes these issues into account while creating a site.

3.2.2 Font Downloading

One way to ensure that a web page is displayed correctly is to allow the users to download the required Hindi font in which the website will be designed. The page publisher can provide the Hindi font file for download or a small program that will install the font directly onto the user's computer. Once installed, the page will be displayed in the correct Hindi font whenever the user revisits the page. With the font installed, the time it takes for the browser to load the text is the same as for any English language page.

Disadvantages

Some Hindi fonts are available in the public domain and have been used by several web publishers, but as there is no common coding standard for Hindi fonts, it is possible that several fonts will be required to view various web pages published using this method. So the users have to install many fonts required for viewing correctly different websites. It can take several minutes for the user to download and install the required font. Different versions of the font are required for both Apple Macintosh and PC platforms. There is currently no Hindi font compatibles on both systems.

3.2.3 Font Embedding

This is the most popular method for publishing Hindi web pages on the internet in which the font is embedded with web pages. A company named Bitstream has developed WebFont Wizard that allows any font to be 'embedded' on a web page. It works by converting the text into a graphical format which can be displayed using a 'font player', now a component of the Netscape browser, but also available for MSIE via a download. Microsoft provides WEFT tool for creating a

dynamic font object to embed any font in web pages but these pages can only be viewed from MSIE. All online newspapers and portals are using font-embedding technique on their sites.

Netscape and Microsoft each has its own technology for embedding fonts. Both browsers MSIE and Netscape Navigator are using different technologies. However, there are some ways to make pages to be displayed correctly in both browsers. The technique is discussed at the end of this article.

3.3 Types of Font

The basic font formats available so far are:

- PostScript Type 1 is the veteran font format from Adobe.
- TrueType was originally developed by Apple and is now built into the Macs and PCs operating systems. It makes better on-screen display.
- OpenType is a newer format created by Microsoft and Adobe that attempts to merge features of PostScript Type 1 and TrueType.

PostScript Type 1, TrueType, and OpenType fonts can not be embedded in its native format. These fonts have to be converted into one of these new formats before using:

- Embedded Open Type (EOT)—from Microsoft
- Portable Font Resource (PFR)—from Netscape and Bitstream

The above two formats are the new font standards, each of them has some limitations: Neither works with all the font formats nor is supported by every web browser. PFR format is supported by Netscape Navigator but to view PFR format fonts in MSIE (Windows only), the user must be willing to install an ActiveX control (provided by Bitstream). While EOT format is supported by MSIE and for this format users don't have to install any ActiveX control. TrueType fonts can be converted into EOT and TrueDoc (PFR) formats. These EOT and PFR formats are called dynamic fonts.

3.3.1 Dynamic fonts

Dynamic font is the special type of font that provides web masters to develop content of their sites using any font they have in their systems. Dynamic fonts are automatically downloaded along with the web pages in which they are embedded, in the user's temporary directory as image files do, instead of relying on the default fonts that users have in their browsers, with the assurance that those pages will be displayed in a browser with their font formatting intact. *Internet fonts* and *Web fonts* are aliases for dynamic fonts.

These fonts are especially useful in case of web pages containing Indian language contents because it is not always possible that the Hindi language fonts used to develop these web pages will be installed on the client's machine. If the Hindi language web page is using a font, which is not present in the client's machine and the web page does not contain any dynamic fonts then the page will not be in a readable format and will show meaningless characters using default fonts. Dynamic Fonts files are browser specific. It means that these files are different for MSIE and Netscape Navigator. Moreover, dynamic fonts files are supported only by MSIE 4.0 & above and Netscape 4.03 & above.

Currently, Dynamic Fonts are available for all Microsoft Windows platforms (Win9x/NT/2000) and Macintosh platforms on the client side. The server side can be running any web server on any operating system. Navigator uses the PFR format, and MSIE uses EOT files.

4. METHODOLOGY

4.1 Using Dynamic Fonts

4.1.1 Requirements

The development of a multilingual web site contains various resources. Hindi web page can be designed using a number of specific requirements. The first requirement is that user should have knowledge of using any HTML editor or MS Word. This article can not cover HTML basics as a complete book will be needed for it and in this little article it is not possible. The second requirement is True

Type Font. This font will be used for web page designing so select the suitable Hindi font. Third is the HTML Editor which best suites the needs. Nowadays many editors are available like Microsoft FrontPage, Macromedia Dreamweaver and even MS word has the facility to save documents as HTML page. Fourth requirement are the tools (WEFT and WebFont Wizard) for converting normal font into dynamic font. Final requirement is the web server where web pages will be uploaded. Web server can be on any operating system.

4.1.2 Step-wise Procedure

(a) To design a web pages in Hindi install a True Type Hindi font in the system. A Hindi font can be downloaded from internet or a new font created using any font creator program.

A sample keyboard layout of Hindi language is shown in fig 1 and fig 2 is an example with shift key

(b) Now type a web page using the selected Hindi font in any HTML editor i.e., Microsoft Frontpage, Macromedia Dream Weaver or Netscape Composer. Also any MS Word document can be converted in HTML format.

(c) Create a dynamic font object (eot file) using WEFT tool and portable font (pfr file) from WebFont Wizard.

Fig 3. is an example of DESIFONT created by DESIDOC. The step-by-step procedure of creating dynamic font using both tools is given in the next paragraph.

(d) Make the following changes in html document:

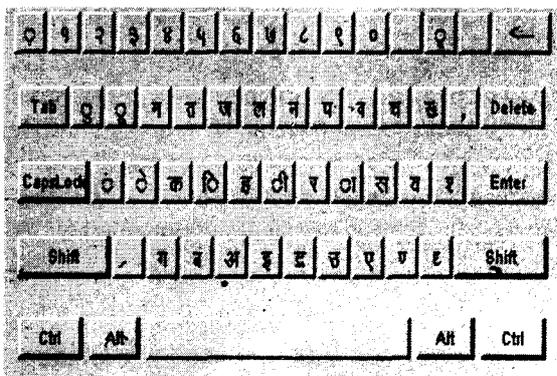


Fig1. Sample keyboard of Hindi language

First change in HTML document is to add link tag to associate the PFR with web page. A link tag tells the browser to associate or 'link' another file with an HTML document. In this case, the link tag tells the browser where to find the font information in the PFR for rendering the fonts on the screen. Place link tags in the document between the <head> and </head> tags. It can be copied from the ReadMe.html file generated each time when a PFR file is created. For example if a dynamic font is created as desifont.pfr then <link> tag looks like this:

```
<link REL="fontdef"
SRC="http://poppy/desifont.pfr">
```

Where poppy is the name of server. Add this link tag in the header of HTML document. The SRC attribute is the only part of the tag that changes. It identifies the name and location of the PFR. Use absolute or relative paths, depending upon how to structure web site.

Add one <link> tag for each PFR to be used, and make sure to add it in each page that uses this font. Although the browser stores the PFR in the cache, include the <link> tags on each page of site to make sure the browser always has the font available.

Next change is to specify the font to be used in the web pages. There are many methods for specifying a font within HTML. Two possible methods include:

- Use of a font tag and specifying a face attribute
- Use of a cascading style sheet (CSS)

In the first method of specifying a font is to specify formatting attributes for the text in the

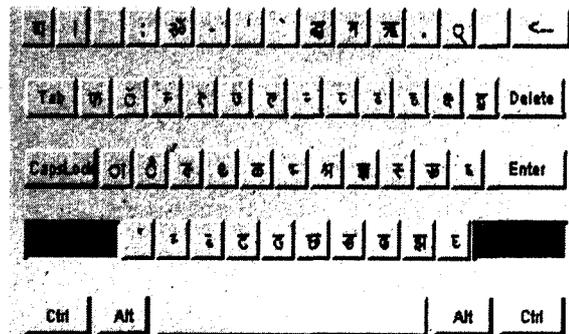


Fig 2. Example with shift key

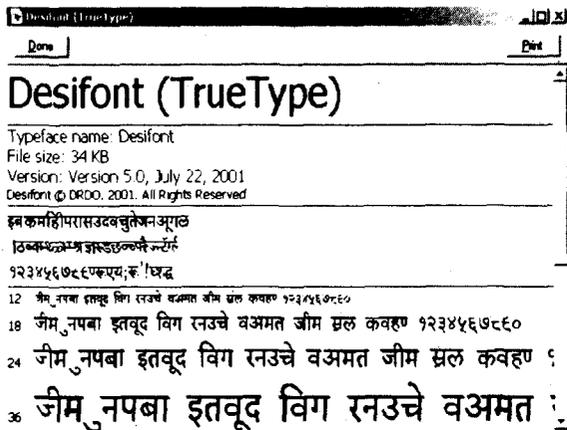


Fig 3. Example of DESIFONT created by DESIDOC

font tag. The face attribute specifies the name of a font. Other font attributes include color and size. Here is an example of a font tag that specifies a face attribute.

```
<font face="DESIFONT" size=2>
insert some text here
</font>
```

The second method of specifying a font is the use of style sheet. Cascading Style Sheet (CSS) can be used to specify fonts in HTML documents. Three types of CSS can be used i.e., inline, embedded, or linked style sheets. There are several ways to specify a font in a style sheet. One method is to use the font-family attribute. Here is an example of a CSS that uses the font-family attribute to format the tags.

```
<STYLE TYPE="text/css">
P {
Font-family: "DESIFONT"
}
</STYLE>
```

If EOT and PFR both type of dynamic fonts are created then following script should be used to associate both files in the web page. To automate this process, add the script in all web pages. In this case there is no need of writing a link tag and style sheet, as already stated, this script will automatically do these things. This script will automatically detect the client's browser and will provide links to .eot and .pfr according to browsers. In the case of Netscape it will provide link tag

to pfr file and if it is MSIE then it will write style sheet for .eot file.

```
<script language="javascript">
akm = parseInt(navigator.appVersion);
if (akm >= 4)
{
if (navigator.appName == "Netscape")
document.write("<link rel='fontdef'
src='http://delhi/web/desifont.pfr'>");
else
{
document.write ('<style type="text/css">');
document.write (' @font-face {');
document.write (' font-family:
DESIFONT'+";");
document.write (' font-style: normal'+";");
document.write (' font-weight: normal'+";");
document.write (' src: url("http://delhi/web/
DESIFON0 EOT")');
document.write ('}');
document.write ('</style>');
}
}
</script>
```

(e) After appropriate changes, HTML pages should be uploaded with PFR and EOT files on web server as usual. Keep in mind that transfer of PFR files should be using binary (not ASCII) transfer mode. If PFR files are transferred using ASCII mode, the browser can not read the PFR file. This embedded font should now be working in both MSIE and Netscape Navigator browsers.

4.1.3 Other Settings

4.1.3.1 MIME Settings for PFR Type Files

In order to host a site that uses PFR web fonts, the web server must be set up to recognize the PFR MIME type. The MIME type is application/font-tdpfr which has file extension pfr. The Netscape Enterprise server already has this MIME type defined. If Windows NT 4 server is being used, and web server is internet information server (IIS) then IIS Server Management Console is used to set the MIME type in the 'File Types' section of the server's 'Properties'. To set the MIME type open IIS Server Management Console. Select server computer in the right window. Select 'Properties' in the left pane. Click 'File Types'. Under *Extension*, type pfr. Under

MIME type, type application/font-tdpfr. Click 'OK' and exit IIS Server Management Console and Restart the server.

4.1.3.2 WebFont Player for MSIE

This section is for those who don't want to create an EOT dynamic font want their pages to be viewed in MSIE. Viewing PFR dynamic fonts in MSIE for Windows requires WebFont player. The player is built into Netscape Communicator for Windows, Macintosh, and UNIX. For site visitors who use MSIE for Windows need to download WebFont player, host this WebFont player on his web site or give it a link to bitstream site. Site visitors only need to download the player only once.

To link to the WebFont Player, insert the following lines of code in the document HEAD (between the <head> and </head> tags). Insert these lines after the LINK tags for PFRs add these lines

```
<SCRIPT LANGUAGE="javascript"
src="http://poppy/tdserver.js">
</SCRIPT>
```

Add these lines to every page that uses dynamic fonts. Where src attribute points to wherever WebFont player files including the file tdserver.js are stored. WebFont player is freely downloadable from Bitstream's website. Download it and install in web server.

A sample page created in Hindi language using DESIFONT is displayed as Fig. 4.

This is the complete procedure of creating a web page in Hindi language. The next section will cover the operations required using both tools.

5. OPERATION

This section will cover each step of creating a dynamic font from a normal font. The step by step procedure of creation of a dynamic font is given below:

- Microsoft WEFT 3
- Bitstream's WFW

Both tools have their importance in creation of a dynamic font because WEFT will create a file with .eot extension and WebFont Wizard will produce a file having .pfr extension. In the previous sections it is

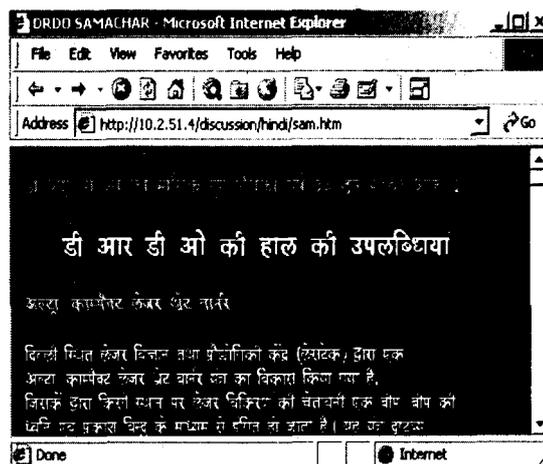


Fig 4. Sample page created in Hindi language using DESIFONT

described that .eot file will be used for MSIE and .pfr file for Netscape Navigator. So this section will cover the process of creation of dynamic fonts using both tools.

5.1 Microsoft WEFT 3

Web Embedded Font Tool (WEFT) is provided by Microsoft and freely available on Microsoft website. This tool is used for creation of dynamic fonts from a normal true type font. As previous sections describes about dynamic fonts this tool is used for creation of EOT font. This font will be used for MSIE. Creation of a dynamic font using WEFT is very simple method. Download WEFT from Microsoft website and install it. After installation, start it, in the first screen it will prompt for name and email address. On the next screen (fig 5.1.1), it will ask to enter location of the web pages to embed fonts. Enter the location of web page or use browse button and click 'Add' to add the pages (fig 5.1.2).

In this process it will automatically include those pages whose link is given in this page. It will also accept URL in the form of http:// else use the browse button to specify any web page from hard disk. When pages will be added the panel will show a tree view to those page.

If there are more pages then add them. If there are pages that show up don't need embedded fonts to appear, select the page

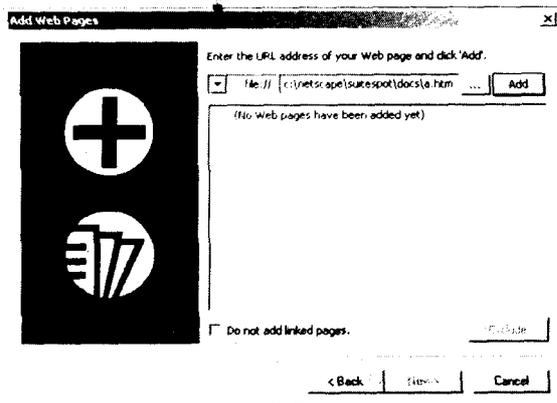


Fig 5.1.1 Add web pages

and click 'Exclude'. After adding all pages click 'Next' to continue.

Now WEFT will analyze all pages and it scans for all the fonts currently used on these pages. After the analysis is complete, a list of all the fonts used on these pages will be displayed.

Next to each font name, there is an icon showing whether the font can be embedded or not. A brief explanation appears for each font in the 'Info' column. The 'Embed' column tells if each font is currently set to be embedded. 'Chars' column displays the number of characters used for each font on those pages. Select the required fonts to embed by eliminating all the others by selecting them and click 'Don't Embed'. If it is required to embed an additional font that isn't shown in the list, click the 'Add' button (fig 5.1.3).

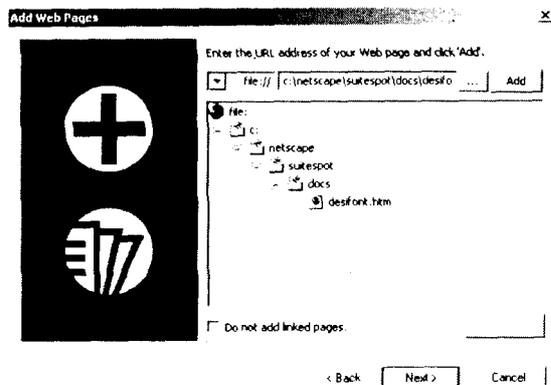


Fig 5.1.2 After addition of desifont.htm

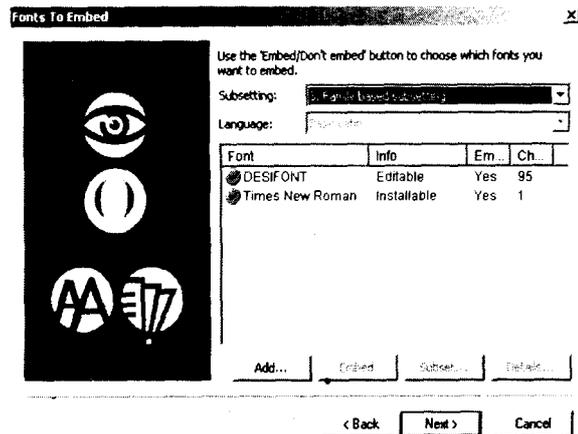


Fig 5.1.3 Select font

Embed only those characters used on web pages, not the entire font, this will keep the generated file very small in size. But it is not always useful such as if anyone wants to generate file and that will be used for every page regardless of character sub setting he should select no sub setting required. Because if anyone selects a sub setting, then each page will require new file because it should have characters included in that page.

After selecting font to be embed click 'Next' button this will show 'Create Font Objects Screen' (fig 5.1.4). Enter the location where dynamic font file to be saved. It also supports FTP, so it can even save the file directly to the server.

In WEFT there is a provision for users to create dynamic fonts for their websites so that web pages using these dynamic fonts can only be viewed from their websites and this

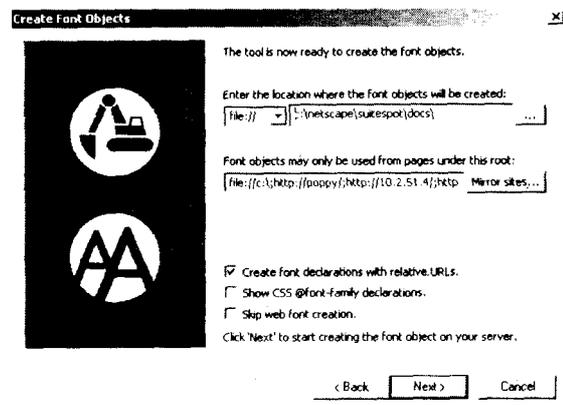


Fig 5.1.4 Specify location of file

will protect others to use their dynamic fonts in his pages. Click 'Mirror Sites' to get the screen shown in fig 5.1.5. Here enter all domain names from where this dynamic font will be served.

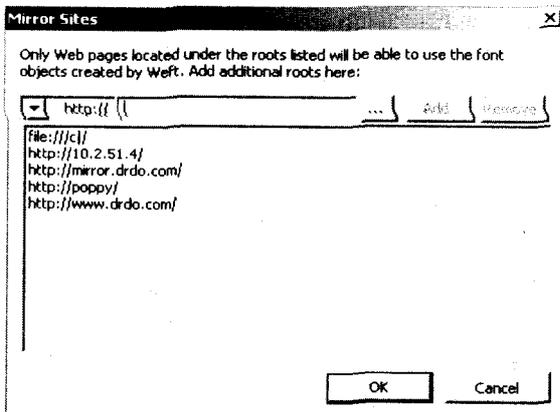


Fig 5.1.5 Binding domain names

For each URL, click 'Add' to add it to the list. Add all domain names where this font will be used. This font can be used from these sites only; no other domain could use this font. When finished, click 'OK'. Check the box labeled 'Show CSS @font-family declarations,' it will display the style sheets code added to all page.

Now click 'OK' to create a font file and .eot font file will be generated in the appropriate directory. Fig 5.1.6 shows DESIFON0.eot is created which is about 10.4kb. That's newborn EOT font file. In case a need arises to redo or revise this particular embedding in future, save all the settings. After clicking 'Finish' on the final Wizard screen, select

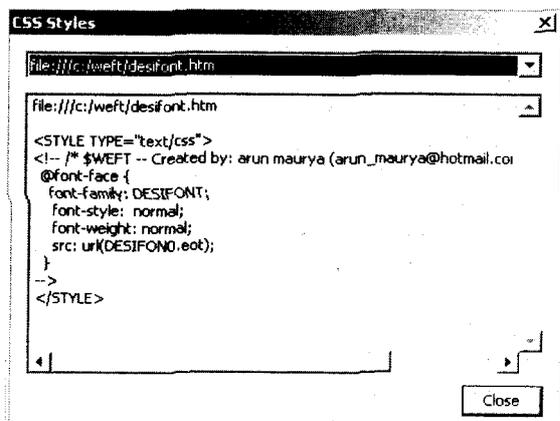


Fig 5.1.6 Generated style sheet

'Save' from the File menu to save this process as a project.

Next section covers the step-wise procedure to create a PFR file using WFW.

5.2 BITSTREAM'S WFW

To create a PFR file bitstream provides WebFont Wizard (WFW) tool. A trial version of this tool can be downloaded or purchased from Bitstream. Now start WebFont Wizard by double-clicking WebFont Wizard icon to start the application. In the first screen (fig 5.2.1) it will show some helpful instructions about how to create a dynamic font. Click 'Next' to begin creation of dynamic font.

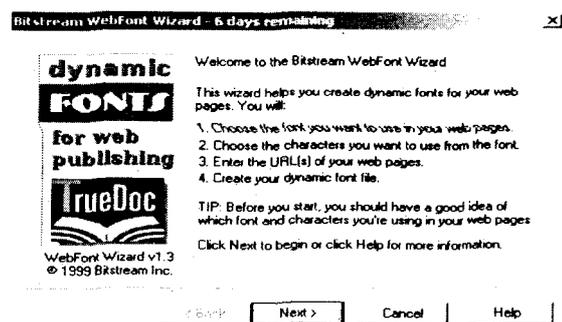


Fig 5.2.1 Welcome screen

In the next screen it will ask to choose the font from which dynamic font will be created. Now select a Hindi font in the 'Font' list that will be used in web pages (fig 5.2.2).

After selecting font choose the characters you want to record in the dynamic font. The fewer the characters, the smaller 'PFR' file will be. Choose an option from this screen (fig 5.2.3).

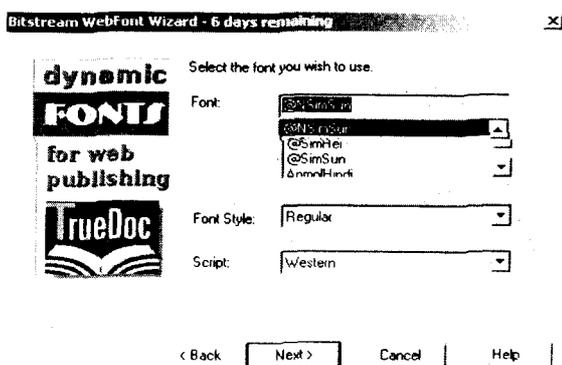


Figure 5.2.2 Choose font

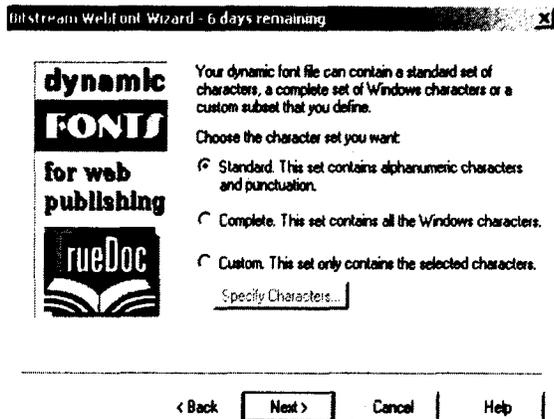


Fig 5.2.3 Select characters

- ❑ **Standard**—Contains the basic upper and lowercase alphabet, numbers, and common punctuation.
- ❑ **Complete**—Contains all characters available in the selected font.
- ❑ **Custom**—Click 'Custom' to define a custom character set. In the 'Custom Character Subset' dialog box, type the characters to be included. Under 'Sets', a set of characters can be selected, or type all characters in the box. It can be done using cut and paste text into the box from another document. Click 'Ok' to close the dialog box.

Click 'Next' to continue. In the next screen enter the URLs of web pages from where .pfr files will be served. Add the URLs of the sites where this .pfr file will be served (fig 5.2.4). This feature prevents other web designers from using this dynamic font on their own sites. The .pfr file created with these URL list can only be used with web pages residing on these URLs only. To add a URL for web

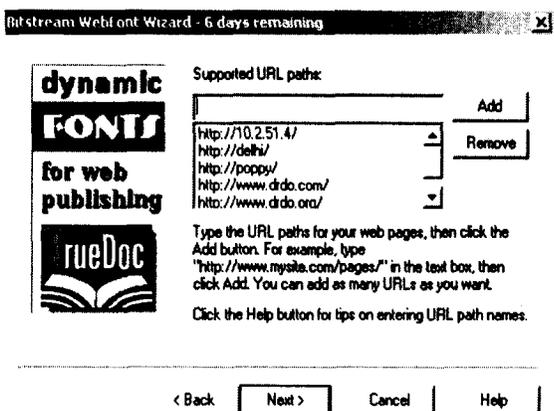


Fig 5.2.4 Entry of URLs

pages, type the address in the box, and click 'Add' button. After addition of all URLs click 'Next' to continue.

Now in the 'Filename' box, type a file name for the PFR file (fig 5.2.5). Click 'Browse' button to select a folder for saving the file and click 'Finish' to create the PFR file in that folder. WebFont Wizard displays a message saying the file was generated successfully.

Click 'Yes' to continue creating PFRs, or click 'No' to exit WebFont Wizard.

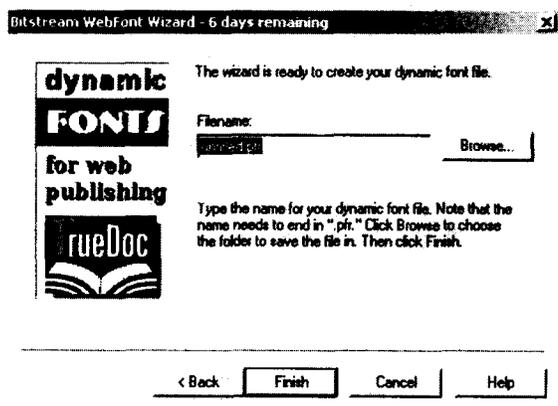


Fig 5.2.5 File name

For working properly in both browsers use .pfr and .eot files together. After using these files the visitors of both browsers can view the pages and he does not require downloading WebFont Player for MSIE. Add this style sheet and <link> tag in header of all pages containing Hindi material as following.

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<Meta name="GENERATOR" content="Microsoft Frontpage 3.0">
<Title>DRDO SAMACHAR </title>
<Link REL="fontdef" SRC="desifont.pfr">
<Style TYPE="text/css">
<!-- / *$ WEFT -- Created by: arun maurya (arun_maurya@hotmail.com) on 7/23/2001 -- * /
@font-face {
font-family: DESIFONT;
font-style: normal;
font-weight: normal;
src: url(DESIFONO.eot) ;!

```

```
}  
-->  
</Style>  
</head>
```

6. INTERACTIVE PAGES

This is an advanced topic for those users who are already in the field of web designing and wants to design their interactive web pages in Hindi/Multilingual language. So before starting to design an interactive web pages in Hindi, users must have some knowledge of any scripting language and their static web page in Hindi should be working properly. Interactive pages are those pages used for taking information from users and render conformant and relevant output using that information. In the process of designing interactive pages there are two main steps:

- (a) To design a form that will accept data from users, and
- (b) Handling that data and giving output using that data.

When users will submit their data using that form the data will be handled by a script on the server. This script will read data entered by users and process it and then give result page. These pages are designed using any scripting languages i.e., ASP, JSP, PHP and PERL. As shown in this article. In static pages embed *.pfr* and *eot* files in headers of all output pages. An example of an interactive web page is given in Appendix A.

7. CONCLUSION

The indigenous or the patriotic federal nature of the web has posthaste the need for well thought-out Hindi web strategy. Traditionally, organisations introduced their websites in English. Now with the web, information products and services are instantaneously available, not only around the

world but even to rural areas. The web provides the campestral agencies with constant and global exposure, heightening the need for careful attention to rural branding and product marketing. The question is no longer when or whether to go ahead with meeting the requirements of our societal community; it's how to manage the process to derive the greatest benefit from our global web presence. If anyone approaches site design with nationalistic mindset, taking structure, artistic, and content into account, he will ensure an architecture that can support any Indian language he wants to go for, and to begin with, Hindi can never be second choice. Web is a global medium, if anyone doesn't address the national Hindi speaking audience, the potency and patronage of his web presence will be diminished. Creating a Hindi website for nationalistic, public-spirited, and zealous target surfers is one of the best ways to transcend linguistic barriers and articulate his language.

Some examples of Hindi websites are:

- www.webdunia.com
- www.jagran.com
- www.naidunia.com
- www.netjaal.com
- www.drdo.org

REFERENCES

1. Microsoft Corp. Website for more information on web embedded font tool <http://www.microsoft.com/typography/web/embedding/>
2. <http://www.truedoc.com/>
3. Bitstream's website <http://www.bitstream.com/>
4. <http://www.hexmac.com/>
5. Font creator program can be downloaded from <http://www.high-logic.com/>

| |
|--|
| Contributors: Dr Mohinder Singh is Director, Defence Scientific Information & Documentation Centre (DESIDOC), Metcalfe House, Delhi. Shri Ashok Kumar is Scientist E at DESIDOC, Metcalfe House, Delhi. Dr AK Tyagi is Scientist E at DESIDOC, Metcalfe House, Delhi. Shri Sumit Goswami is Scientist B at DESIDOC, Metcalfe House, Delhi. Shri Arun Maurya is Sr. Tech. Asstt. 'A' at DESIDOC, Metcalfe House, Delhi. |
|--|

Appendix A

This appendix has some information about designing of interactive web pages using Hindi/Multilingual language. After completing the process of designing static web pages users want to design their interactive web pages. Before designing interactive web pages in Hindi users must ensure that their static pages are working fine then they should proceed. This appendix will demonstrate the step by step procedure of designing an interactive page:

Step 1: Create form using any HTML editor like FrontPage, Macromedia Dream Weaver, etc. (fig 6.)
HTML for the above form is given below.

```
<html>
<head>
<title>TESTING.....</title>
<meta name="GENERATOR" content="Microsoft FrontPage 3.0">
<style TYPE="text/css">
<!--
@font-face {
font-family: DESIFONT;
font-style: normal;
font-weight: normal;
src: url(DESIFON0.eot);
}
-->
</style>
<link REL="fontdef" SRC="desifont.pfr">
</head>
<body>
<form method="POST" action="/akm/test.pl" name="aform">
<table border="0" width="50%">
<tr>
<td width="27%"><font face="DESIFONT" size="5"><strong>uke</strong></font></td>
<td width="73%"><input type="text" name="nam" size="20"
style="font-family: DESIFONT; font-size: 16pt; font-weight: bold"></td>
</tr>
<tr>
<td width="27%"><strong><font face="DESIFONT" size="5">Hkk&quot;kk</font><font size="3"
face="DESIFONT"> </font></strong></td>
<td width="73%"><select name="Lang" size="1"
style="font-family: DESIFONT; font-size: 16pt; font-weight: bold">
<option selected value="vaxzsth">vaxzsth</option>
<option value="fgUnh">fgUnh</option>
</select></td>
</tr>
<tr>
<td width="27%"><strong><font face="DESIFONT" size="5">in </font></strong></td>
<td width="73%"><input type="text" name="pad" value="oSKkfud" size="20"
style="font-family: DESIFONT; font-size: 16pt; font-weight: bold"></td>
</tr>
<tr>
<td width="27%"></td>
<td width="73%"></td>
```

```

</tr>
<tr>
<td width="27%"><input type="submit" value="Hksts" name="B1"
style="font-weight: bold; font-family: DESIFONT; font-size: 16pt"></td>
<td width="73%"></td>
</tr>
</table>
<p>&nbsp;</p>
<p>&nbsp;</p>
</form>

```

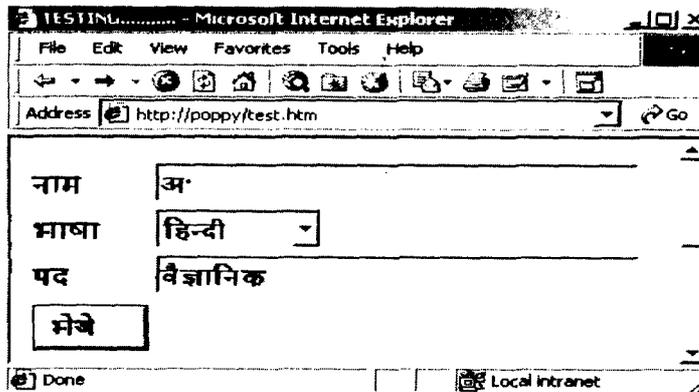


Fig 6. Create form using any HTML editor

```

</body>
</html>

```

Step 2. Now write script for processing this form

Web designer can use any scripting language like ASP, JSP, PHP or PERL. When the user will submit that form the data will be processed by this script and after processing result will be displayed in Hindi. To use these script users must have some knowledge of PERL because this example uses PERL Script.

Test.pl

```

#-----Gets data entered by user-----
&GetQuery;
#-----Print Output Page
print "Content-type: text/html\n\n";
print<EOF_FILE;
<html>
<head>
<title>DRONA - In Hindi</title>
<style TYPE="text/css">
<!-- /* 6/21/2001 -- */
\@font-face {
font-family: DESIFONT;
font-style: normal;
font-weight: normal;
src: url("http://poppy/DESIFON0.eot");
}
-->
</style>
<link REL="fontdef" SRC="desifont.pfr">

```

```

</head>
<body bgcolor="#D88000">
<p align="center">&nbsp;&nbsp;&nbsp;</p>
<p align="center"><font size="4" face="DESIFONT" color="#FFFF00">
<strong>uke $FORM{'nam'}<br>
in $FORM{'pad'} <br>
Hkk"kk $FORM{'Lang'}<br></strong></font></p>
</body>
</html>
EOF_FILE
#-----
#This function gets form data
#-----
sub GetQuery {
local(*in)=@_ if @_;
local ($i,$key,$val);
if ($ENV{'REQUEST_METHOD'} eq "POST") {
read(STDIN,$FORM,$ENV{'CONTENT_LENGTH'});
}
@FORM=split(/&/,$FORM);
foreach $i (0 .. $#FORM) {
$FORM[$i] =~ s/+/ /g;
($key,$val)=split(/=/,$FORM[$i],2);
$key =~ s/%(..)/pack("c",hex($1))/ge;
$val =~ s/%(..)/pack("c",hex($1))/ge;
$FORM{$key} .= "\0" if (defined($FORM{$key}));
$FORM{$key} .= $val;
}
return length($FORM);
}

```

After processing, this script will generate the output as shown in Fig 7. This example shows that user can enter data in Hindi language but he should also know keyboard layout of this font. But if user cannot type in Hindi language, a virtual keyboard layout can be provided on the screen atop his pages, from which he can select specific key by a click of the mouse. But this

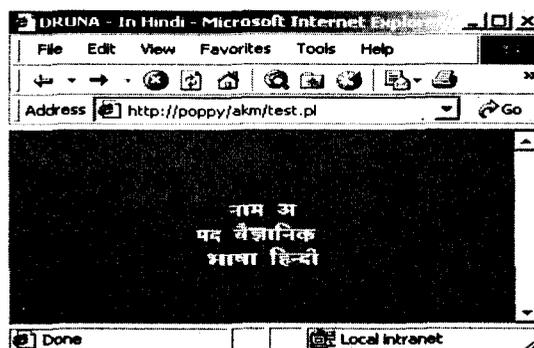


Fig 7. Output after processing the form

can make the input process too slow. Another option available with the site developer is to provide a transliteration software in his web page where the user can type Hindi in phonetic English like 'MERA BHARAT MAHAN' and it automatically gets converted in to the Hindi scripts as "Maora Baart Mahana".