

---

---

## Globalization of Software Applications Using UNICODE Based Multilingual Approach

Sonia Dube

Yatrik Patel

T A V Murthy

### Abstract

*Concept of Globalization of Software can be boom to advancement of projects related to Digital libraries and associated software. Much need is felt in the Indian context where support is to be provided for many languages to take care of diversified regional requirements and complexity of INDIC script. In this paper we have presented an approach and Implementation for creating Globalized software using UNICODE based Multilingual approach.*

**Keywords :** Multilingual computing, Unicode, Globalization, Internationalization, Localization.

### 0. Motivation

Developing software acceptable and adaptable to users across the continents and users with different scripts and languages as in case of India is a challenging job. The process of making software acceptable on global basis among user with different languages is carried out using concepts of Globalization, which can be carried out using concepts of Internationalization and Localization, which focuses on technology development to bring users to follow and adapt standards. Adapting the software to meet the specific requirements of customer for interaction with the software is done using concepts of Localization. Scientific advancements in the field of multi-lingual computing and Software standardization such as Unicode has helped in building user friendly software, acceptable by people from different zones and areas. The National Language Support (NLS) supplied by the Microsoft Win32 application programming interface (API) can be used for making Internationalized Software components, whereas modifying the user interface (UI) elements, translating text, and standardizing terminology are localization steps. Success of Globalize Software can only be ensured when these concepts are incorporated in the software from the design phase and issues of Internationalization and Localizations are properly taken care of.

### 1. Unicode

Computers during older days were mainly used for number crunching, but with advancements in the processing power and technology associated with multi-media has made them more users friendly and now are readily being used by people in all possible fields, which can be thought of. Initially, computers just dealt with numbers and store letters and different characters by unique numbers, there was no fixed encoding schemes available to assign numbers to different characters. These encoding schemes were not powerful enough to deal with all available letters and characters. The conflict in the encoding schemes created lot of problems in bringing different applications on a single platform. Unicode was invented to overcome the limitations imposed by old incompatible computer encoding standards ASCII, ISCII, JS etc and has now become standard for character encoding. Unicode apart from standardizing the application has also being considered as a major step towards simplifying multilingual computing.

Unicode[1] is a universal encoded character set that enables information from any language to be stored using a single character set. Unicode provides a unique code value for every character, regardless of the platform, program, or language. The role of unicode in maintaining international standard is to define

---

characters from an international perspective, without preventing any culture from defining its own character sets for internal communications.

There are different techniques to represent each one of the Unicode code points in binary format. Each of the techniques uses a different mapping to represent unique Unicode characters.

The Unicode encoding are classified as:

**UTF-8 :** UTF-8 has been defined by the Unicode Standard to meet the requirements of byte-oriented and ASCII-based systems. Each character is represented in UTF-8 as a sequence of up to 4 bytes, where the first byte indicates the number of bytes to follow in a multi-byte sequence, allowing for efficient string parsing. UTF-8 is highly used in Internet for content exchange and data transfer.

**UTF-16 :** This is the 16-bit encoding form of the Unicode Standard where characters are assigned a unique 16-bit value, with the exception of characters encoded by surrogate pairs, which consist of a pair of 16-bit values. The Unicode 16-bit encoding form is identical to the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) transformation format UTF-16. In UTF-16, any characters that are mapped up to the number 65,535 are encoded as a single 16-bit value; characters mapped above the number 65,535 are encoded as pairs of 16-bit values. As more name space is provided in this format and hence there exist provision for incorporation on more languages in Indian context [2]

**UTF-32 :** Each character is represented as a single 32-bit integer.

## **2. Multilingual Applications**

Multilingual computing [3] is defined as “Use of computer to communicate with people in various languages”. Multilingual computing makes it possible to use the Internet for global communication by supporting more than one language simultaneously and has the ability of handling more than one script of character sets. In multilingual application a single executable can be run in multiple languages and can handle data in multiple languages. At the time of application loading, multilingual enabled application provides the option to the user for language selection. The design of a multilingual application should be such that the language of the application is separate from the language of the data and provision should exist for encoding of data using standards such as Unicode.

In Indian context lot of work is carried out by CDAC for developing Multilingual Technology and Applications such as “Acharya” developed by IITM for providing Multilingual computing environment. Attempts are being made to incorporate Multilingual features in the new version of SOUL.

## **3. Potential Applications for Multilingual Computing are**

- ? Digital Libraries
- ? Regional Web Portals
- ? Education to Home

## **4. Implementation Scenarios**

Unicode based multilingual global applications can be developed using support provided by Visual Studio.Net platform and MS SQL Server, which can be used for backend database storage. Two scenarios can be thought of for Implementation of Multilingual application based on user requirements.

- ? Homogeneous Framework: In this scenario the Application Language and the Data Language is same and the user selects the language during the start of the application. This is a very crude way of implementing Multilingual applications and can be used where applications require handling of few languages.
- ? Heterogeneous Framework: In this approach the system supports for application and data to be of different languages, user selects the language when the application starts and data language can be changed as per the user requirement for providing inputs to the applications, however outputs from the application will automatically be displayed in the language in which data was stored. Although there may be very little requirement of changing the application language during the run-time but such requirements can also be taken care of in this framework. This approach can supports handling of different languages in a single session of application and can help in building true Globalized applications.

#### Comparative Study for different Frameworks

	<b>Homogeneous</b>	<b>Heterogeneous</b>
Web Support	Different Web pages for different Languages	Unicode enabled web-pages and browser with same web pages
Database Archival	Different Database for different Languages	Same Database for Different Languages
Run Time Changing of Applications Language	No	Yes
Run time Changing of Data Language	No	Yes

### 5. Operating System and Database Support for Multilingual Software Development

Windows 2000/NT/XP and above are all completely Unicode enabled. So all provides best support for Unicode and all languages to implement multilingual applications. Windows provides language options to select the required language. Even though the older versions of windows do support but require the specific language edition of the operating system to support.

MS Visual Studio.Net System Globalization namespace can be used for changing the data language in applications being developed on .NET framework.

MS SQL server uses nvarchar and nchar data types for storage of UNICODE data in the database tables.

### 6. Benefits of Globalization

Globalization of software application offers various benefits to developers and users. From developer point of view the benefits are :

- ? Developer has to develop application only once, no need to change source code when new language requirement appears.
- ? Product can compete in global market.
- ? Users can take benefit of using multiple languages in a single product only.
- ? Users can maintain global standards in their data by using same software product.
- ? By creating and using multilingual applications developers and users can save lots of time and money.

## 7. Conclusion

Globalization of software application by using multilingual approach is the best way to share product among users having multilingual requirements. Unicode has evolved as major standard and has simplified the task of multilingual computing and Globalization of the software. The heterogeneous framework proposed above helps in building true Globalized applications.

## 8. References

1. The Unicode Standard, Version 3.0, Addison-Wesley, Massachusetts, 2000
2. Devika Madalli, Unicode for Multilingual Representation in Digital Libraries from Indian Perspective, Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital, 2002, Page 398
3. "Introduction to multilingual computing" <http://www.riverlion.com/jpfaq/mlcintro.pdf>

### About Authors



**Mrs. Sonia Dube** is Project Scientist at INFLIBNET Centre, Ahmedabad, Gujarat and holds MCA. Earlier she was Programmer in Projects at INFLIBNET funded by NISSAT and WHO.

**E-mail** : [sonia@inlibnet.ac.in](mailto:sonia@inlibnet.ac.in)



**Mr. Yatrik Patel** is Scientist - B at INFLIBNET Centre, Ahmedabad, Gujarat and holds B.E. in Computer Science. He is engaged in the software R&D activities, SOUL design, development, installation and troubleshooting and having expertise in server configuration & very large databases. He has published several articles at national and international level and recipient of best paper presentation award by IATLIS and second best article prize by RRLF, Kolkatta on public libraries.

**E-mail** : [yatrik@inlibnet.ac.in](mailto:yatrik@inlibnet.ac.in)



**Dr. T A V Murthy** is Director of INFLIBNET Centre, Ahmedabad, Gujarat and holds B.Sc, M L I Sc, M S L S (USA) and Ph.D. He is President and Fellow of SIS-India, Hon. Director of E.M.R.C., Gujarat University and Member Secretary of ADINET, Ahmedabad. He carries with him a rich experience and expertise of having worked in managerial level at a number of libraries in many prestigious institutions in India including National Library, IGNCA, IARI, Univ of Hyderabad, ASC, CIEFL etc and Catholic Univ and Case western Reserve Univ in USA. He has been associated with number of universities and has guided number of Ph.Ds and actively associated with the national and international professional associations, expert committees and has published good number of research papers. He visited several countries and organized several national and international conferences and programmes.

**E-mail** : [tav@inlibnet.ac.in](mailto:tav@inlibnet.ac.in)