Engineering Global Software – A System Test Standards Perspective

Pat O' Sullivan & Maria Hyland

There are various definitions of the term Software Engineering. A concise definition can be found in IEEE [1993]: 1.) the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software, and 2.) the study of approaches as in (1).

Global Software and Localisation Testing is a key aspect of IBM's Global Software Engineering division, and is a profession commonly referred to as GVT (Global Verification Testing). At the IBM Dublin Software Lab GVT is a discipline that underpins the success of many IBM products that are released across many global markets and is therefore an important and necessary activity for global success. It is therefore an established and integral part of IBM's end-to-end software development process.

The key motivation for GVT is the common knowledge that it is difficult to sell computer hardware/software which does not support the prevailing language and national customs. Therefore IBM and most software development organisations now produce software that is tuned to the needs of local markets with the notion that early success is essential, and respecting standards is a smooth prerequisite to ensuring customer satisfaction and quality. The IBM GVT process attempts to ensure that standards are uniformly implemented and respected across all of their software applications, as consistency in approach and methodology not only aligns process across development teams, but also delivers a consistent experience to the end users.

Interestingly, one of the first attempts to establish a standard took place in the year 1120 (at the time of the 1st Crusades). The pioneer was King Henry 1st of England. Henry mandated that the ell (the ancient yard) should be the same length as his arm and he insisted that this distance would be the standard unit throughout his lands. The ell (45" in length) was used until recently for measuring cloth. Henry's counterpart across the channel must have been of smaller stature, as the Flemish ell is 27 inches or 3/4 of a yard… Standards, Standards, Standards!!!

Standards in Global Software Engineering and Test

Standards in software engineering and test are critical to repeatability and incremental success. There are many international standards organisations that have helped guide the evolution of International standards and best practices in the global software industry. These organisations have members from a number of large and small organisations, and serve as a "police force" to ensure that the introduction of new technologies, protocols and standards are monitored carefully against global requirements. Everyone is a vested community member of this police force and adherence is key!

The end-to-end process of softwareglobalisation and localisation is complex as Fig.1 shows. Starting with the early stages of development right through the process of release and manufacturing there are many potential pitfalls that can result in mistakes or fundamental errors. Several checkpoints need to exist in the process to allow engineers to identify problems early so as to motivate timely solutions. Standards exist right from code inception (where developers need to design code to be culturally, platform, and linguistically neutral) right through to when the final CD is cut

(where formatting and writing characteristics also need to respect established standards).

To help industry in the difficult task of comprehending and respecting the various global standards we note that there are many international standards organisations who have helped guide the evolution of international standards and best practices in the software industry. These organisations, which have members from a great number of both large and small organisations, serve as our "police force" to ensure that the introduction of new technologies, protocols and standards is monitored carefully against international requirements.

The **International Standards Organization (ISO)** is a consortium which was set up to encourage the interchange of character data between countries which do not share common character sets or alphabets. ISO's main focus is in recommending protocols & GUI rendering mechanisms to preserve and present data in an accurate and persistent way. ISO has made great efforts to evolve legacy 7-bit protocols to 8-bit so that data above 127 decimal can be transported correctly. This effort is most noticeable in the TCP/IP protocol. ISO has also defined the widely accepted 8-bit character set table, allowing for international languages to be included. Examples include ISO-8859-1 for US and Western European countries, IISO-8859-2 for Eastern European countries, ISO-8859-3 for Cyrillic countries, and so on. ISO has also evolved the Unicode standard, and this is a superset code page and character set representation which captures all possible characters for all possible countries and code pages. As we know, ISO's work has influenced many aspects of Global Software Testing.

The **American National Standards Institute (ANSI)** is a national standards body working on standardisation of all types. Noticeable efforts include the development of the ANSI character set and the standardisation of all of the various computer programming languages, particularly C and C++.

The **National Institute of Standards and Technology (NIST)** is a member of ANSI and represents the United States government. NIST advises the US government on standards regarding hardware and software publishing. A good example here is the exporting of encryption technology outside of the US, where it was once illegal to make the 128 bit encryption technology (used in the language applications like Lotus Notes) available to countries outside the U.S.

The **Institute of Electrical and Electronic Engineers (IEEE)** was formerly concerned with hardware standardisation. More recently the organisation has become much more software focused. The organisation has also started to look at more fundamental engineering standards such as interface definition, software integrity and interfacing of systems. Such standards feed various aspects of Design, Development and Global Software Testing.

The **European Computer Manufacturers' Association (ECMA)** is dedicated to standardisation of hardware-oriented facets of computers. It is concerned with the standardisation of paper sizes, page layouts, conversion routines and device compatibility.

Global software testing needs to consider language specific challenges and users' disparate needs. The Unicode project evolved to address this and involves a number of companies

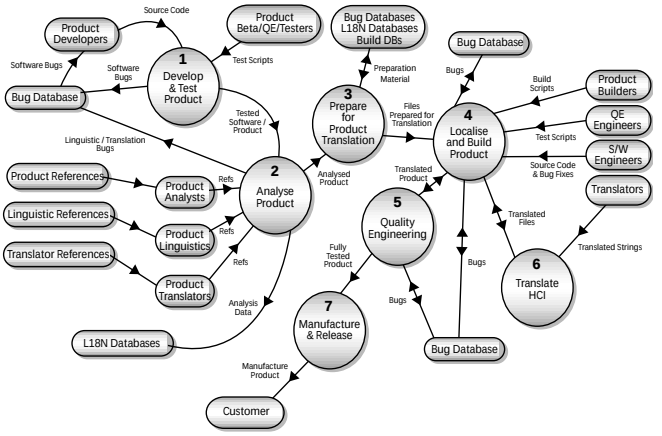

Fig. 1: Software globalisation and localisation process

which include Apple Computers, Xerox, Claris, IBM, Microsoft, Sun Microsystems, and many more. The key objective of the consortium is the delivery of a multi-byte character set to permit many countries to share and exchange data in a way which maintains the persistency of data across the various operating systems and protocols. Other objectives include a) the research, development and standardisation of code points for the characters and character sets of new or emerging countries; b) the implementation of specifications to describe these new character tables in terms of decimal values which can be referenced in software; and c) the influencing of data interchange standards at many levels.

LISA was established to promote standardisation in technical terminology by organising technical seminars, forums, and so on. The goal of the forum is to agree on standard terminology. Today LISA is working to standardise thousands of terms for the information technology industry. It is also attempting to raise the overall recognition and image of the localisation industry. There are several companies on the LISA technical steering board including IBM, Apple, Alpnet, Berlitz, DEC, IBM European Language Services, ICL/UK, INK International, Mendez, Microsoft, NCR, Oracle, Rank Xerox, Sequent, SoftTrans International and many more.

Since its establishment in 1995 at University College Dublin, the **Localisation Research Centre (LRC)** has developed from a funded three year project into a world-recognised research centre with a strong industrial backing manifested in its high-profile Industrial Advisory Board. The centre has representatives from academia and both overseas and indigenous companies, and has become a respected representative body for the industry in Ireland. Its annual conference has become one of the major industry events attracting interest from the globalisation community nationally and internationally. Key areas to the LRC's credit include the coordination of the efforts of the localisation industry, the provision of a forum for information dissemination and sharing, liaising with other relevant interest groups, and the pursuit of excellence in research, development and information dissemination. Since March 1999 the LRC has moved to its new designated laboratories at the University of Limerick in Ireland and is continually expanding its own services and research.

The Localization Institute is an organisation providing quality resources, training, seminars and conferences on localisation, internationalisation and international business development. Their mission is to enable companies to succeed in international markets by providing focused training for professionals and by promoting the sharing of experience and information among them. The Localization Institute is perhaps best known for its annual "Localisation Roundtable" which is a gathering of localisation leaders. The Localization Institute has also conducted a number of seminars such as "Introduction to Localisation" as well as two seminars on "Writing and Designing for Localisation".

The **International Workshop on Internationalisation of Products and Systems (IWIPS)** attempts to bring together internationalisation and localisation specialists from industry, academia and government. It was founded in 1998 by Girish Prabhu and Elisa del Galdo. IWIPS is small by design, to maximise networking and promote a shared experience among delegates.

The standards organisations serve to promote best practice and provide an education forum and reference point for their members. As Scott McNealy, Chief Executive Officer of Sun Microsystems, pointed out, standards arise from common usage and best applicability to the specific problem domain. The standards organisations usually have a very large number of members from worldwide development organisations, and they in turn agree and implement the recommended standards to which smaller companies need to abide to remain compatible and to survive. Without standards, international incompatibility within programming languages, systems, protocols and ultimately geographical regions would hinder progress and best practice.

An IBM Global System Test Perspective

Global verification testing at IBM's Dublin Software Lab includes NLS (National Language Support) and aspects of TVT (Translation Verification Testing). GVT is therefore the process whereby all aspects of Software is verified for functionality and suitability in world markets. IBM's Dublin Software Lab's System Test teams see GVT as a key part of our overall testing paradigm. Under the lab's GVT umbrella IBM plan and execute several classifications of complex tests on English, European and Asian configurations. One aspect to this testing is the need to ensure that all aspects of the software are adhere to recognised industry standards.

Significantly, IBM and most other reputed software development organisations discourage proprietary/break-away standards, and view break-away approaches as bad practice that leads to unforeseen downstream problems. Not having an established and commonly agreed methodology ultimately results in customer problems in aspects of Hardware and Software implementation. During IBM System Test engineering standards are pervasive in cross platform protocol/interoperability testing including some fundamental aspects of LDAP (e.g. to respect the InetOrg Directory Schema), HTTP/HTTPS, SSO, SSL, SOAP, SIP, WSDL, GSM/CDMA/GPRS, and so on.

IBM's need to test on a multitude of platforms (AS400, AIX, Linux SuSe+Redhat, S390, Windows 95/98/2000/XP/ME/2003) and client browsers (Internet Explorer, Netscape, Mozilla) aggregated with the need to support the incremental product versions creates a potentially exponential test matrix. However, ensuring that standards are respected on one operating system platform often permits the ability to scale testing on the other platforms. This is essential for containing testing times and costs, however to reduce the matrix one needs to understand aspects of the underlying standard and its implementation which can oftentimes be difficult.

There are other key dimensions to testing that standards are correctly respected. For example, testing in one, two and three tiered client/server systems will surface unique challenges when compared to testing on simple isolated end user configurations. Also, interface testing of global systems needs to be conducted to ensure seamlessness of data. Cross platform testing also presents unique challenges as each platform has its own nuances. Character encoding decisions may be different, and may generate conversion issues when going across platforms. Also, unique aspects of configuration paramaterised by locale and character set nuances can give rise to new GVT concerns. Implementation specifics and run time behaviours can also give rise to unique issues – for example, client side cellular devices may behave differently and implement standards in an inconsistent way.

Some other less obvious testing concerns where standards apply include the fact that automation needs to be written language neutral, and platform neutral; concerns for Bi-Directionality (RTL/LTR) need to be respected; IMEs (Input Method Editors) need to be tested in the native language and platform; graphics and icons need to be verified as culturally suitable and neutral; currency formats and time formats need to be cognisicant of the local requirements; and so on. Perhaps less obvious but implied standards in global software testing include sacred symbols that need to be respected in software and tested for (where appropriate). Examples here include the Christian Cross in Christianity, the Menorah in Judaism and the Crescent in Islam. Unlucky numbers are also interesting, for example 2 and 514 in traditional Chinese belief; 4 is an unlucky number in Hong Kong, Korea and Taiwan; 4 in China and Japan suggests death. The number 7 is unlucky in East and West Africa while the number 13 is unlucky in most countries. Problematic gestures in Graphics/Icons is also worth noting, for example blinking the eye in Hong Kong and Taiwan; the folded arms in Fiji and Finland; the "Stop" gesture in Greece and Nigeria; and the use of the left hand to point in Islamic cultures.

Industry typically expects conformity to standards from the bigger players, and to serve as role models. However this is not always seen in practice. Hence a significant bias is placed on testing in areas such as sorting and sort sequences, searching, date/time, authentication, data interchange and integration, decimal separators, locales, and DBCS (Double Byte Character Set). In the area of character sets it is worth pointing out that testing for China's GB18030 character set is a mandated requirement from the PRC government if software vendors wish to release to that market.

From a development perspective some obvious concerns can have major impacts on the testing of global software, and can give rise to many questions and testing implications that are underpinned by inherent and established best practice standards:

- Are all executable program files designed so that they do not need to be rebuilt after translation?
- Is all information relating to text strings, dialogues, menus, messages, coordinates, sizes, fonts, etc., completely separated from the code?
- Is all translatable information stored in platform's standard resource file format, so that localisation tools can function without modification?
- Does all user-visible text correctly display all permutations and combinations of foreign characters, both single-byte and multi-byte?
- Is full multi-byte support provided for all user-visible text?
- Is it possible to use accented characters as hit-keys or accelerator keys in dialogs, menus, and lists?
- Are all string comparison algorithms implemented to take into account the requirements of languages other than English?
- Are all date, time, currency, list separator, measurement, number formats, etc., fully resourced for localisation?
- Is all help text separate from the program's core executable?
- Are there any assumptions made concerning the sizes and availability of fonts or font information?
- Are all aspects of the product designed to be culturally neutral?
- Do all resource files contain appropriate comments and documentation?
- Is it verified that any resourced text which is dynamically constructed (such as error messages) can be translated appropriately, and that variable parameters can be rearranged without corrupting the string?
- Does the UI (User Interface) design allow for text strings to expand by up to 30% when they are translated?
- Are all icons, cursors, and bitmaps culturally neutral and designed so that any text is overlaid at runtime?

Conclusion

In recent years developments have materialised to help bridge data interchange problems along established and well debated standards. Examples include XLIFF, CSS, XML, XSL, SRX, OLIF, DXLT and TBX just to name a few. It is also encouraging to see progress in automation tools, from respected software vendors such as Rational and Segue, that allow for automation scripts to be written in a way that allows them to be reused on language versions of software. From a translation perspective encouraging evolution in Translation Memory, Machine Translation and Translation Web Services are bridging fundamental gaps that existed in the past.

IBM have learned that it is one challenge to respect standards, but how the standard is implemented may not be properly standardised. Issues have been observed in the IBM Dublin Software Lab where 3rd party products read and display IBM formatted HTML/HDML/WML correctly, but there have been problems when attempting to render in reverse. Some software vendors embed proprietary tags, and parse for these during run-time execution effectively bypassing agreed standards under the umbrella of proprietary logic. This presents problems for "good citizens" who try to interpret the incoming data based on the established standards. The test harness subsequently becomes "test the standards" + "test the exceptions". Too many times teams have exerted engineering and testing efforts to manage the exceptions and oftentimes containing test plans and test cases is impossible as they spiral out of control.

Critical issues in the rapidly changing global marketplace challenge the formal international standards system. The time to complete an international standard is often out of sync with the marketplace's needs. In the information world web years are measured in months. Development, publication and adoption of international standards must keep pace and this is one key problem. International standards enjoy global acceptance and worldwide agreement on content and acceptance, as opposed to consortia and proprietary international standards. Any attempt to derail these strengths devalue and trivialise the meaning of international standards. ■

Pat O'Sullivan is a Test Architect at IBM's Dublin Software Lab. He completed his BSc in 1991, his MSc in 1997 and his PhD in 2001, and both of his IBM sponsored MSc and PhD HCI research projects won industry awards. He can be reached at patosullivan@ie.ibm.com

Maria Hyland is a Development Manager at IBM and has many years (15) experience in the field of Software Engineering, Software Development and Globalisation. Currently Maria is managing the Dublin and India System Test teams for one of our key projects - Lotus Workplace. She can be reached at maria_hyland@ie.ibm.com