On the Perceived Usefulness of the Localization of Compilers in African Indigenous Languages

Momed A. A. Neves and Seraphin D. EyonoObono

Abstract—**There are thousands of languages worldwide but only some of them are flourishing while many face extinction. This leads to a situation where computer graphical interfaces are mostly presented in English even for users from other indigenous languages. This presents a challenge for non-English natives, hence the need for software localization, defined as the presentation of computer graphical user interfaces in non-English languages. The aim of this study was to examine the perceived usefulness of the localization of programming compilers in African indigenous languages. Users' perspectives were examined in this study through a questionnaire based survey of Information and Communication Technology (ICT) students from universities in the KwaZulu-Natal province of South Africa. These perspectives were analysed in the Statistical Package for Social Sciences (SPSS) software, and it was found that localizing compilers in African languages is perceived as very useful by users, irrespective of demographic, cultural, language, and programming proficiency factors. The contribution of this research resides in the provision of new evidence on the usability of localized software.**

Index Terms—**African indigenous languages, compilers, software localization, usability.**

I. Introduction

A computer programming language is used to facilitate the development of computer programs. However, most compilers use mainstream languages such as English, at the expense of other indigenous languages. This results in the scenario that non-English speakers need to learn at least one of these mainstream languages. They also need to learn how to program in that mainstream language, placing a huge cognitive load on the students [1],hence the need to localize compilers' software.Compilers localization can provide assistance to students in their programming activities, especially during debugging, which involves finding, understanding as well as correcting cryptic error messages. Debugging cryptic error messages is a demanding activity that contributes,along with other factors, to the perception that programming courses are difficult. One can mention the example of two of localized compilers: Kids' Programming Language (KPL) and Easy Quick Universal Accessible Language (EQUAL). While KPL covers European languages, it is interesting to note that EQUAL even spanstoAfrican languages such as Shona (Zimbabwe) and Ethiopic (Ethiopia).

II. Aim and Rationale

The objective of this study is to analyse students' perceptions on the usefulness of the localization of compilers. This study aims to examine the perceived usability of compilers localized in an African indigenous language.

This research was motivated by the need to contribute to the preservation of indigenous languages by using them as much as possible including through Information and Communication Technologies (ICTs), not only for end-users but also for developers. This research is also motivated by the need to improve the learning of programming by students.

III. Problem Statement

Programming courses are generally regarded as difficult by students. As a result, these courses are experiencing high failure rates [2] and alarming declines in student enrolments [3]. This may also explain why some computer science courses have high drop-out rates [4]. Moreover, many tertiary education students who complete their first year in programming are not able to write even simple computer programs [5]. These students are so de-motivated by their numerous syntax and semantic errors that they become convinced that compilers' error messages aren't helpful because they are too cryptic [6]. Consequently, they spend much longer than necessary attempting to fix trivial errors because they do not understand compilers' error messages [6]. This is particularly detrimental to students' academic success because the presence of syntax errors in their programs usually results in low exam scores [7].

IV. Theoretical Framework

This study used the Cultured Centred Design (CCD) as its theoretical framework [8]. This is a holistic iterative design approach which takes into account culture, background, language, usability, cognition, and the user interface during all phases of the design. The framework also emphasizes that users' social-cultural experiences as well as their background affects their perceptions and interactions with their user interface. The CCD model is composed of a designer's and a user's filter. The designer's filter uses the designer's knowledge, his/her personal experience and ideas from a specific background during the design of the interface. This is achieved by first collecting cultural data and technical requirements such as usability. Furthermore, the user's filter involves the end-users' understanding of an interface based on cultural issues such as language and taboos. This study

Manuscript received 15 October, 2012; revised December 14, 2012.

The authors are with the Durban University of Technology/Department of Information Technology, Durban, South Africa (email: momedn@email.com, eyonoobonosd@dut.ac.za).

DOI: 10.7763/IJIET.2013.V3.243

also relied on the Quality in Use Integrated Measurement (QUIM) model. This usability model defines usability as the extent of effectiveness, efficiency, and satisfaction provided by a product; and also the extent to which it can be used to achieve users' goals [9].

V. Literature Review

Localization of software in general has been used in software products such as websites [10], educational programs [11], and specifically in the context of this study, in compilers [12]. Factors affecting the usability of localized software are briefly discussed below according to the following QUIM attributes: effectiveness, efficiency, satisfaction, learnability, trustfulness, universality, and usefulness.

Effectiveness and Efficiency:A heuristic evaluation of a US website was used to examine usability problems associated with non-localized user interfaces. This evaluation performed by 3 Asian usability experts and by 21 Taiwanese Management Information Systems students revealed that web-browsing requires more effort and time if the content is not localized [13]. Another study combining different research methods (an experiment involving software localized into an Indian language, a survey of 198 postgraduate Indian students, and a subsequent interview of 40 students from the same sample) also found that users are more efficient when browsing localized websites [14]. Similarly, a literature review by [11] revealed that educational localized software is more effective than non-localized software specifically for children.In addition, questionnaire-based survey of 150 isiZulu speaking South African students from various schools and universities showedthat the majority ofstudents believed that localized user interfaces would allow them to learn new conceptsmore effectively [15].

Satisfaction: A study combining several research methods (questionnaire based survey, interview, and experiment), involving 40 isiZulu speaking small enterprise owners in a South African rural area (Umbumbulu), found that users favoured localized software compared to their English counterparts [16]. Similarly, in a questionnaire based survey of staff members from four universities in the KwaZulu-Natal province of South Africa, users were provided with a description of the African Village Metaphor with localized computer icons representing objects from the African culture. It was found that some older participants like the localized desktop metaphor[17]. Likewise, an experimental study and questionnaire based survey on 28 postgraduate students from various countries (i.e. China, Britain, and Taiwan) found that participants were pleased with a localized desktop metaphor [8].

Learnability: Some of the above described studies [11], [15]-[16] also found that localized software is easier to learn compared to their non-localized counterparts.

Universality: A usability study (using interview, questionnaire based survey, and observation) which presented an e-Commerce software product to 30 South African usersfound that some users recommended the localization of e-commerce websites in other local languages such as isiZulu and isiXhosa [18].

Trustfulness: An experimental and survey-based study of 198 Indian postgraduate students from the Indian Institute of Management found that users trust websites that are in their native language[14]. Similarly, a multi-national survey and an experimental study involving 1156 participants from countries such as Japan and Germany indicated that trust depends on certain cultural dimensions such as uncertainty avoidance [19].

Usefulness: Participants of a study conducted by [16] on the localization of accounting software were of the opinion that localized software contributes to a better understanding of accounting concepts compared to a non-localized interface. In addition, a mixed approach study (usability tests using localized cell-phones interfaces, questionnaires, and interviews) of 88 participants revealed that localized software improves cell-phone users' device competency [20].

VI. Research Design

A questionnaire was used to collect data from 138 students on their perceptions on the usefulness of the localization of compilers. Thirty seven (37) questionnaires were discarded due to incomplete or incorrect entries, resulting in a final sample of 101. The participants were first, second and third year students from three universities from the KwaZulu-Natal province of South Africa. Table I shows the survey's sample which consisted of students from an ethnicity perspective.These students were selected from Computer Science and Information Technology academic departments, and they were requested to complete a questionnaire consisting of five sections, each representing a research variable.

TABLE I: Sampling

Ethnic Group	Frequency	Percentage (%)
African	57	56.4
Indian	30	29.7
White	14	13.9
Total	101	100

A. Research Variables

The dependent variable of this study was the perceived usefulness of localized compilers; and the independent variables were: students' background or demographics, their programming proficiency, their indigenous language proficiency, and their English language proficiency. The demographics consisted of two ordinal items (age and year of study), and eight nominal items (gender, ethnicity, university, home language, programming course in high school, end user computing in high school, qualification, and university). Apart from the demographics, all variables were measured using a 5-point Likert scale. Proficiency in English and indigenous languageswas measured using items such as the ability to read, to speak, and to paraphrase. Programming proficiency was measured using items such as the ability to debug, design, and write program artefacts. The dependent variable measured whether students believed that localization would improve their programming skillsin terms of their ability to design and to optimise program artefacts, and to define client requirements.

TABLE II: RELIABILITY TABLE FOR RESEARCH VARIABLES

Research Variable	Questionnaire Item	Cronbach’salpha(α)
English proficiency	I10, I11, I12, I13, I14, I15, I16, I17, I18, I19, I20	0,946
Indigenous language	I20, I21, I22, I23, I24, I25, I26, I27, I28, I29, I30	0,978
Programming proficiency	I30, I31, I32, I33, I34, I35, I36, I37, I38, I39, I40	0,942
Usefulness of localization of compilers	I40, I41, I42, I43, I44, I45, I46, I47, I48, I49, I50	0,95

B. Data Analysis

TABLE III: VALIDITY TEST RESULTS

	Rotated Component Matrix[a]				
Research Variable		Component			
		1	2	3	4
English proficiency	I11			0,783	
	I12			0,778	
	I13			0,772	
	I14			0,749	
	I15			0,777	
	I16			0,857	
	I17			0,772	
	I18			0,798	
	I19			0,805	
	I20			0,806	
Indigenous language proficiency	I21	0,839			
	I22	0,907			
	I23	0,917			
	I24	0,907			
	I25	0,932			
	I26	0,929			
	I27	0,913			
	I28	0,941			
	I29	0,916			
	I30	0,912			
Programming Proficiency	I31				0,731
	I32				0,796
	I33				0,791
	I34				0,719
	I35				0,841
	I36				0,788
	I37				0,776
	I38				0,724
	I39				0,842
	I40				0,812
Usefulness of localization of compilers	I41		0,793		
	I42		0,886		
	I43		0,803		
	I44		0,881		
	I45		0,876		
	I46		0,836		
	I47		0,828		
	I48		0,853		
	I49		0,676		
	I50		0,850		

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.
a. Rotation converged in 5 iterations.

SPSS software was used to analyse the data collected by the survey. This data was initially tested both for reliability (Cronbach’s alpha coefficient) and validity (factor analysis). Afterwards, descriptive analysis was performed on all attributes and research variables. Thereafter, correlations (Pearson's correlations) were tested between the different Likert scale variables; and regression analysis was performed where correlations were found. Correlations (ANOVA) were also tested between demographics and the dependent variable, with the possibility of joint interactions (ANCOVA) between demographics and the independent variable.

C. Data Reliability and Validity

Reliability tests performed by this study were conclusive for all the questionnaire items (Cronbach’s alpha >0.94), as illustrated by Table II. The survey’s data also fully passed the validity test as all Likert scale items split into exactly four components (Table III). All tests were performed with a confidence level of 95%.

D. Descriptive Analysis

1) Demographics

The descriptive analysis of the demographics of the participants of this research shows that most subjects (45.5%) were male (69.3%) between the age of 18-20. Most of the participants were African (56.4%). The most common home language was English (47.5%). Most participants did not take a programming course in High school (68.3%), nor end user computing (69.3%). However most participants did pure maths (97%). It is also worth noting that most participants were computer science students (63%).

2) Likert scale variables

Descriptive statistics on Likert scaled variables are presented in Table IV. According to these statistics, students perceived their English proficiency as very high (mean of 42.12 out of a maximum of 50). They also rated their programming proficiency (37.1 out of 50), their indigenous language proficiency (33.65 out of 50), and the usefulness of localization of compilers (34 out of 50) as high.

E. Inferential Statistics

TABLE IV: DESCRIPTIVE STATISTICS FOR LIKERT-SCALE BASED RESEARCH VARIABLES

Variable*	N	Min	Max	Mean
EP	101	20.00	50.00	42.1287
IP	101	10.00	50.00	33.6535
ULC	101	10.00	50.00	34.5050
PP	101	16.00	50.00	37.0891

Variables*:
English proficiency (EP),
Indigenous language proficiency (IP)
Usefulness of Localization of Compilers (ULC),
Programming Proficiency (PP)

Table VI presents the ANOVA test results of this study and Table VII presents the Pearson's correlation test results. According to Table VI, students' demographics do not correlate with their perceived usefulness of localized compilers. As can be seen from Table VII, the only correlation is between students’ proficiency in English and their perceived programming proficiency. Perceived usefulness of localized compilers was not found to be associated either with English proficiency, nor with indigenous language proficiency, nor with programming proficiency.However, most participants indicated that localized compilers would be useful for improving the programming ability of students.

TABLE VI: ANOVA RESULTS

Tests of Between-Subjects Effects

Dependent Variable: Usefulness of localized compilers

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	1863.812[a]	20	93,191	1,160	,311
Intercept	5417,355	1	5417,355	67,428	,000
Age	311,087	3	103,696	1,291	,283
Gender	,936	1	,936	,012	,914
Language	867,473	7	123,925	1,542	,165
Year	26,898	2	13,449	,167	,846
Race	66,581	2	33,291	,414	,662
ProgHigh	2,024	1	2,024	,025	,874
EndUSer	,729	1	,729	,009	,924
HighSchoolMaths	71,556	1	71,556	,891	,348
Qualification	0,000	0			
University	218,922	1	218,922	2,725	,103
Error	6427,436	80	80,343		
Total	128541,000	101			
Corrected Total	8291,248	100			

a. R Squared = .225 (Adjusted R Squared = .031)

TABLE VII: PEARSON'S CORRELATION ANALYSIS OF THE RESEARCH VARIABLES

		IP	EP	PP	ULC
IP	Pearson Correlation	1	-0,107	-0,105	0,122
	Sig. (2-tailed)		0,289	0,294	0,224
	N	101	101	101	101
EP	Pearson Correlation	-0,107	1	.507**	0,085
	Sig. (2-tailed)	0,289		0	0,398
	N	101	101	101	101
PP	Pearson Correlation	-0,105	.507**	1	0,055
	Sig. (2-tailed)	0,294	0		0,588
	N	101	101	101	101
ULC	Pearson Correlation	0,122	0,085	0,055	1
	Sig. (2-tailed)	0,224	0,398	0,588	
	N	101	101	101	101

**. Correlation is significant at the 0.01 level (2-tailed).

VII. CONCLUSION

This study highlights the perceived usefulness of compilers' localization. Its targeted languages are African indigenous languages especially in the South African context (isiZulu, isiXhosa, and Afrikaans). Apart from providing empirical evidence in support of software localization, this study can also contribute to the preservation of African indigenous languages. Future work should be done on the full development of localized compilers in a variety of African languages. Such compilers can then be researched in terms of their design, usability, and adoption.

ACKNOWLEDGEMENT

The authors wish to thank the participants and staff members from the University of KwaZulu-Natal, University of Zululand, and Durban University of Technology.

REFERENCES

[1] M. Yousoof, M. Sapiyan, and K. Kamaluddin, "Reducing cognitive load in learning computer programming," in *Proceedings of the World Academy of Science, Engineering and Technology*, Vienna, 2006, pp. 259-262.

[2] J. Bennedsen and M. Caspersen, "Failure rates in introductory programming," *Sigcse Bulletin*, vol. 39, no. 2, pp. 32-36, June 2007.

[3] R. Mason, G. Cooper, and M. de Raadt, "Trends in introductory programming courses in Australian universities languages, environments and pedagogy," in *Proceedings of the Fourteenth Australasian Computing Education Conference)*, Melbourne, 2012, pp. 33–42.

[4] J. M. Cohoon and L. Chen, "Migrating out of computer science," *Computing Research News*, vol. 15, no. 2, pp. 2-3, March 2003.

[5] M. McCracken, T. Wilusz, V. Almstrum, D. Diaz, M. Guzdial, D. Hagan, Y. B. D. Kolikant, C. Laxer, L. Thomas, and I. Utting, "A multi-national, multi-institutional study of assessment of programming skills of first-year CS students," *Sigcse Bulletin*, vol. 33, no. 4, pp. 125-180, December 2001.

[6] S. Kummerfeld and J. Kay, "The neglected battle fields of syntax errors," in *Proc. ACE '03 Proceedings of the Fifth Australasian Conference on Computing*, Darlinghurst, 2003, pp. 105-111.

[7] E. S. Tabanao, M. M. T. Rodrigo, and M. C. Jadud, "Predicting at-risk novice Java programmers through the analysis of online protocols," in *Proceedings of the Seventh International Workshop on Computing Education Research*, New York, 2011, pp. 85-92.

[8] S.-T. Shen, M. Woolley, and S. Prior. (July 2006). Towards culture-centred design. *Interacting with Computers*, [Online]. 18*(4)*. pp. 820–852. Available: http://www.dx.doi.org/10.1016/j.intcom.2005.11.014

[9] Seffah, M. Donyaee, R. B. Kline, and H. K. Padda, (2006). Usability measurement and metrics: A consolidated model. *Software Quality Journal*. [Online]. 14*(2)*. pp. 159–178. Available: http://www.dl.acm.org/citation.cfm?id=1132324.1132342

[10] D. Cyr and H. Trevor-Smith. (July2004). Localization of web design: An empirical comparison of German, Japanese, and United States web site characteristics. *Journal of the American Society for Information Science and Technology*, [Online]. 55*(13)*. pp. 1199–1208. Available: http://www.onlinelibrary.wiley.com/doi/10.1002/asi.20075/full

[11] K. Nikolopoulou. (June 2007). Early childhood educational software: Specific features and issues of localization. *Early Childhood Education Journal*, [Online]. 35*(2)*. pp. 173–179. Available: http://www.springerlink.com/index/10.1007/s10643-007-0168-5

[12] J. Schwartz, J. Stagner, and W. Morrison, "Kid's programming language," in *Proceedings of ACM Siggraph 2006 Educators program*, Boston, 2006.

[13] P.L. P. Rau and S. F. M. Liang. (January 2003). Internationalization and localization: evaluating and testing a Website for Asian users. *Ergonomics*. [Online]. 46*(1)*. pp. 255–270. Available: http://www.dx.doi.org/10.1080/00140130303527

[14] D. Cyr, G. S. Kindra, and S. Dash. (November 2008). Web site design, trust, satisfaction and e-loyalty: The Indian experience. *Online Information Review*, [Online]. 32*(6)*. pp. 773–790. Available: http://www.emeraldinsight.com/journals.htm?issn=1468-4527&volume=32&issue=6&articleid=1753975&show=html

[15] M. Njobe, “Understanding the influence of a second language on the academic performance of learners in Information Technology: A case study of isiZulu-speaking English second language learners in KwaZulu-Natal,” M.Tech Dissertation, Department of Information Technology, Durban University of Technology, Durban, South Africa, 2007.

[16] D. Heukelman, “Can a user centred approach to designing a user interface for rural communities be successful,” in *Proceedings of CHISA*, Cape Town, 2006, pp. 51–58.

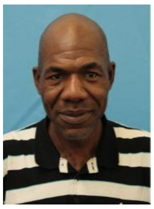
[17] D. Heukelman and S. EyonoObono, “Exploring the African Village metaphor for computer user interface icons,” in *Proceedings of the South African Institute of Computer User Interface Icons*, Cape Town, 2009, pp. 132–140.

[18] Singh, “A guide to improving the e-commerce user interface design,” M.Tech Dissertation, Department of Information Technology, Durban University of Technology, Durban, South Africa, 2005.

[19] D. Cyr, “Website design, trust and culture: An eight country investigation,” in *Proceedings of the Seventh Annual Workshop on HCI Research in MIS*, Paris, 2008, pp. 1–5.

[20] L. D. Krisnawati and Restyandito. (January 2008). Localized user interface for improving cell phone users’ device competency. *International Journal of Information Technology and Web Engineering*. [Online]. 3*(1)*. pp. 38–52. Available: http://www.igi-global.com/article/localized-user-interface-improving-cell/2640

Momed A. A. Neves was born in Maputo, Mozambique in 1985, and completed his primary and secondary education in Maputo. He obtained a national diploma from Tshwane University of Technology, Nelspruit, South Africa, in 2008. He then obtained a Bachelor degree in information technology (software development stream) from the Durban University of Technology, Durban, South Africa, in 2009. His research interests are software development and Human Computer Interaction. He is currently pursuing his Masters degree at the Durban University of Technology.



Seraphin D. EyonoObono was born in Yaounde, Cameroon, in 1967. He completed his primary and secondary education in Yaounde with a “Baccalaureat C” (Mathematics and Physics) obtained in 1986. He was then awarded a bursary by the Cameroonian government to pursue his tertiary education in France where he obtained a BSc degree in Computer Science (Nancy I) in 1990, a BSc honors degree in Computer Science (Nancy I) in 1991, a MSc degree in Computer Science (Rouen) in 1992, and a PhD in Computer science (Rouen) in 1995. He is currently an associate professor in information technology (IT) at Durban University of Technology, Durban, South Africa. His research interests are: pattern matching and ICT for development.