Rimgaudas LAUCIUS

INTERNATIONALIZATION OF COMPILERS

Summary of Doctoral Dissertation
Technological Sciences, Informatics Engineering (07T)

1395

VILNIUS GEDIMINAS TECHNICAL UNIVERSITY
INSTITUTE OF MATHEMATICS AND INFORMATICS

Rimgaudas LAUCIUS

INTERNATIONALIZATION OF COMPILERS

Summary of Doctoral Dissertation
Technological Sciences, Informatics Engineering (07T)

Vilnius LEIDYKLA TECHNIKA 2007

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS INSTITUTAS

Rimgaudas LAUCIUS

KOMPILIATORIŲ INTERNACIONALIZACIJA

Daktaro disertacijos santrauka
Technologijos mokslai, informatikos inžinerija (07T)

General characteristic of the dissertation

Topicality of the problem. The experience gained when participating in the projects of "OpenOffice.org", "Mozilla", "AbiWord" and other software localization has revealed that even the software developed for international markets is often insufficiently internationalized. Because of that its localization is more difficult and followed by various problems.

By investigating the origin of a low software internationalization level and looking for the solution of this problem, some hypotheses have been made and tested.

Hypotheses

1. The compilers that are in use are not internationalized enough.
2. The internationalization level of software can be increased using an internationalized compiler for its development.
3. The elements of software internationalization can be advanced to a higher level – the internationalization of compiler (Fig 1).
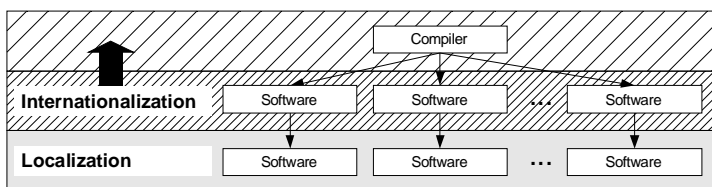
Fig 1. The level of compiler internationalization

Since a single compiler may be used to produce a lot of software, we would achieve great benefit by transferring the elements of software internationalization into the compiler as the internationalization of each software instance developed by that compiler would require less expenditure.

Tasks of the work

1. To analyse scientific and methodical literature, related with software internationalization and discuss the theoretical aspects.
2. To analyse and compare the most frequently used compilers in terms of internationalization.
3. Experimentally internationalize the chosen compiler.

After corroboration of the hypotheses, additional objectives have been made:

4. To analyze aspects of internationalization of compilers and systemize them.
5. To prepare the method of internationalization of compilers.

Aim of the work. To analyze, collect, systemize, and present elements and the method of the internationalization of compilers.

Methodology of research. The information research, systematization, analysis and generalization methods have been used to collect and present the knowledge from the area of compiler internationalization.

The methods of analysis of the cases and the test have been used in the research of the internationalization level of compilers.

The method of comparative information analysis has been used in the analysis of the results on the internationalization level of compilers.

The method of experiment has been used while internationalizing the *Free Pascal* compiler.

Scientific novelty. Compiler internationalization is a new topic and there are only scarce sources of scientific and methodical information, connected with it indirectly. The interest in software internationalization arose in the last decade when it appeared to be needful. However, in the last of this period, no attention has been paid to compiler internationalization, although it is one of the main factors that influence the software internationalization.

A method for evaluating of internationalization level of compilers is presented in this work. The research of internationalization level of compilers based on this method is presented as well.

Most of the publications present the software internationalization as an additional process of software development and try to tackle internationalization problems looking for more effective models and methods of the process itself. That does not solve the problems – the software internationalization remains a complicated and expensive process. This work suggests new ideas claiming that problems can be substantially solved only at the primary step of software production it means, by introducing the methods of internationalization into the software's development tools – and provides a method how to do that.

The method presented is prepared according to the latest information sources, standards, software development methods and technologies. The prevailing practice and tendencies in the field of software internationalization have been taken into the account.

Practical value. The presented method will help to solve the problems of internationalization of compilers and software developed by means of it.

The presented method will allow evaluating the internationalization level of compilers.

Approbation. The results of the dissertation are presented in 8 scientific publications (the total list of publications is given at the end of the summary).

- 2 of the publications are published in the editions included in the ISI lists of scientific publications.
- 6 of the publications are published in the reviewed scientific editions.

The results of the dissertation are presented in 12 conferences.

International conferences:
1. 2nd International Conference Information Technology: Research and Education. London, London Metropolitan University, 2004.
2. LRC – X, The Global Initiative For Local Computing. Limerick, Localization Research Centre, 2005.
3. Teaching mathematics: retrospective and perspectives. Vilnius, Vilnius University, 2005.
4. Informatics in Secondary Schools: Evolution and Perspectives. Vilnius, Seimas of the Republic of Lithuania, 2006.

Local conferences:
5. XLII Lithuanian Mathematical Society Conference. Vilnius, Vilnius University, 2001.
6. Days of Computer Professionals and Users – 2003. Vilnius, Seimas of the Republic of Lithuania, 2003.
7. XLIV Lithuanian Mathematical Society Conference. Vilnius, Vilnius Pedagogical University, 2003.
8. Information technology 2003. Kaunas, Kaunas University of Technology, 2003.
9. XLV Lithuanian Mathematical Society Conference. Kaunas, Lithuanian University of Agriculture, 2004.
10. Days of Computer Professionals and Users – 2005. Klaipėda, Klaipėda University, 2005.
11. XLV Lithuanian Mathematical Society Conference. Vilnius, Vilnius University, 2005.
12. XIII Symposium of the Arts and Sciences. Vilnius, Palace of Science Council of Lithuania, 2005.

The scope of the scientific work. The scientific work consists of the introduction, 4 chapters, conclusions, list of literature, list of publications and addenda. The total scope of the dissertation – 86 pages, 24 pictures, 10 tables and 5 addenda.

1. Introduction

This is an introductory section. It presents the research problem, objectives and motivation, used methods, main results, novelty and relevance and information on the papers in which the main results were published.

2. The theoretical fundamentals of the research

The concept of internationalization and information sources related with the research problem are reviewed and analysed in this section.

Internationalization is the main term, used in this work. Its definitions, given by various authors are slightly different. The definition most used in

scientific publications is provided by LISA[1]: *the internationalization is the product generalization process that it will be able to process different languages and cultural attitudes without redesigning*.

This definition treats subject the internationalization as a particular process. In this work, internationalization is also defined as follos: *the whole of software features that make it easy to adapt to various languages and cultures*.

Internationalization is quite a new subject originated in the last decade. During this period the concept of internationalization has changed drastically. Due to that these changes are briefly reviewed here allowing a better understanding of this concept. The motivation of internationalization is driven by market laws. In order that software to have a success in foreign markets, it has to be localized. However, localization is much more complicated if software has not been internationalized previously. That is why investment into internationalization returns and brings gain depending on how many times software has been localized (Fig 2).
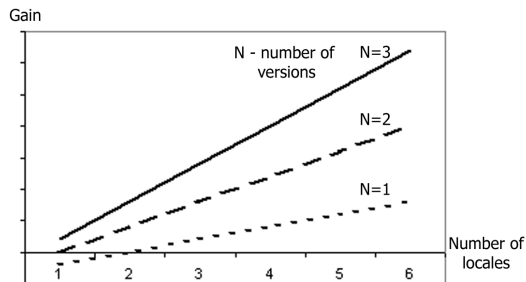
Fig 2. Gain from internationalization

The methods for evaluating the internationalization level, the sources of researches on software internationalization level evaluation, and the reasons of a low level of software internationalization are surveyed and conclusions are drawn in this section as well.

The main conclusions of this section are:

1. Internationalization is a part of the software development process therefore the internationalization of tools, used for software development is very important.

2. The researches done in the area of software internationalization reveal that still there are many problems. The main problems are of technical character, such as: insufficiency of standards and methods meant for the development of internationalized software, problems of text encoding, and insufficient support of internationalization by software development tools, etc.

---

[1] The Localisation Industry Standards Association. http://www.lisa.org/

Because of these problems software internationalization requires more expenditure – it reaches 10–20 % of total software development cost. The expenditure is growing because software developers are obligated to search for individual solutions of software internationalization, and their realization is more complicated.

3. Most of the software internationalization problems may be eliminated by implementing internationalization support into software development tools. Thus the development of internationalized software would become a natural process just like the development of non-internationalized software, and so it would demand less expenditure.

3. The research of compilers internationalization level

Research of the compilers internationalization level and its results are dealt with here.

The three compilers most popular in the world as the object of research has been chosen: Visual C++, Java, Delphi, as well as Free Pascal compiler that has been decided to internationalize. The popularity of compilers has been established by a few additional researches – by finding out which compiler had been used for creating 50 programs that were chosen randomly, as well as by statistics of the popularity of the programming languages and other criteria.

Contemporary compilers usually consist of three parts: translator, runtime library, and linker. For the development of internationalized software the internationalization of runtime library is utmost important, because it consists of code fragments that are inserted into the software produced during its link time (Fig 3). Therefore, if these fragments are not internationalized, then the produced software will not be internationalized as well.
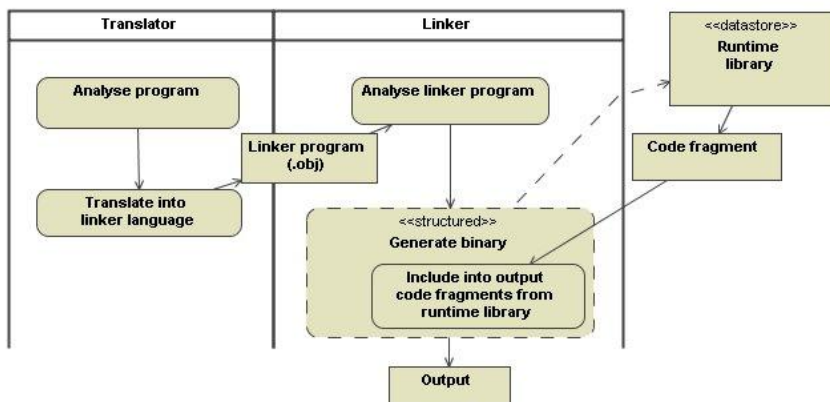

Fig 3. Linking process

The method for evaluating the internationalization level of compilers has been developed, considering similar methods used in the practice of software developers and scientific software internationalization researches. However, it should be pointed out that similar researches of compiler internationalization have not been performed so far.

The method is based on the questionnaire. Its questions are supported by tests. In this way, compiler internationalization can be investigated by analysing their documentation, and in order to support the findings, additionally the compilers themselves can be investigated by using tests. There are 8 groups of questions: 1) data structures; 2) compiler resources; 3) separation of resources; 4) input; 5) output; 6) text processing; 7) cultural elements; 8) platform services.

Table 1. Results of assessment of the internationalization level of compilers

i	Requirements R_i	Assessments Q_i				
		Delphi (Win32)	Delphi (.NET)	Java	Visual C	Free Pascal
	Data structures					
1.1.	Data structures	100	0	0	100	200
	Resources of compiler					
2.1.	Separation of textual resources	0	0	0	0	540
	Service of separation of resources					
3.1.	Separation of textual resources	0	0	0	0	0
3.2.	Combining of grammatical forms of words	400	400	200	400	200
3.3.	Text phrases identification	0	0	0	0	0
3.4.	Text nesting and concatenation	0	0	0	0	0
3.5.	Separation of non-textual resources	0	0	500	0	500
	Input					
4.1.	Encoding	200	200	50	200	300
4.2.	Input methods with third level or additional groups of characters	0	0	0	0	0
4.3.	CJK input methods	500	500	500	500	500
	Output					
5.1.	Encoding	200	100	50	100	300
5.2.	Output of CJK and complex scripts	2500	2500	2500	2500	2500
	Text processing					
6.1.	Encoding	300	300	0	300	300
6.2.	Character attributes	200	200	0	0	200
6.3.	Case conversion	150	150	50	50	150
6.4.	Iteration	800	800	0	800	800
6.5.	Normalization	600	600	50	600	600
6.6.	Bidirectional text transformation	700	700	0	700	700
	Locale					
7.1.	Loading of locale data	800	600	0	300	800
7.2.	Calendar, date and time	400	300	0	300	600
7.3.	Formatting and format analysis	2500	2000	50	2000	2500
7.4.	Collation	300	300	0	150	300
7.5.	Searching	300	300	300	300	300
	Platform services					
8.1.	File system services	0	50	0	50	50
8.2.	Other services	0	250	0	100	250

We will denote set of requirements as *{R_i}*. The inconsistency of a compiler with these requirements is assessed by experts by means of lines of the source code needed to solve the shortcomings during the development of internationalized software. The set of assessments *{Q_i}* is obtained in this way (Table 1) The total internationalization level of a compiler is denoted by

$$Q = \sum_{\forall i} Q_i \; .$$

The generalized results (Fig 4 and 5) reveal rather a low internationalization level of compilers.
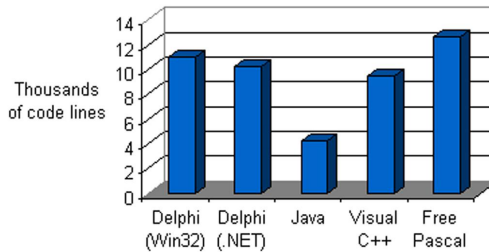
Fig 4. Internationalization's level of compilers
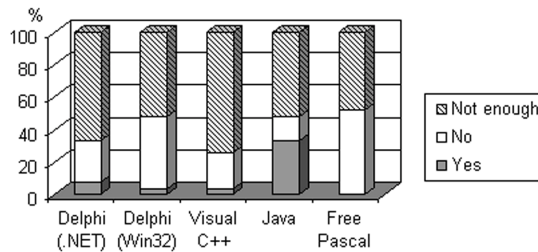
Fig 5. Meeting the requirements

Figure 5 presents how many requirements are met, not met or insufficiently met.

The main conclusions of the research are:

1. The internationalization level of all the compilers is low and internationalization of software created using them requires great additional expenditures.

2. Most of the problems of internationalization of compilers are related to text processing and locale services.

4. The method of internationalization of compilers

Cultural and language factors that influence the internationalization of software are considered at the beginning of the section. The main attention is paid to the realization of support for various writing systems. The analysis is presented employing Daniels and Bright[2] classification of writing systems. It well reflects different characteristics of scripts that determine different realization of their support in a computer. The language factors of software internationalization are discussed in short as well. That helps to understand presented internationalization method better.

The recommended method for internationalizing of compilers is based on the internationalization framework (Fig 6).
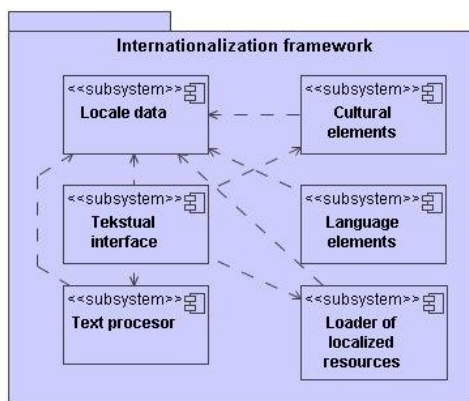
Fig 6. Structure of internationalization framework

Most of the algorithms and data structures used in software internationalization may be reused repeatedly. Because of that it is worth to collect and aggregate those into single structure, the framework of witch is presented.

The generalized use cases of internationalization framework are presented in the following scheme (Fig 7). Internationalization framework is used by compiler and by output binaries generated using that compiler (generally that are applications). These use cases corresponds main functionalities of components of internationalization framework.

---

[2] Daniels P. T. ir Bright W. *The World's Writing Systems.* Oxford University Press 1996.
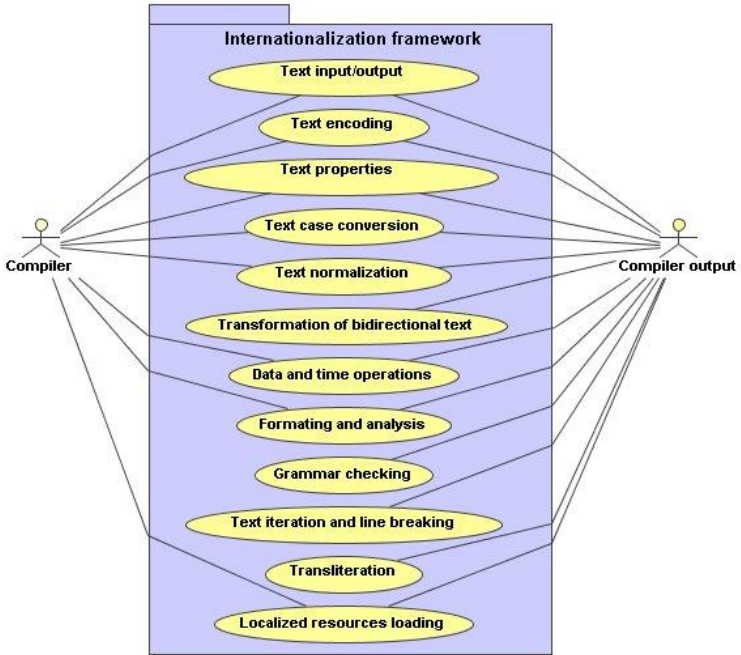
Fig 7. Use cases of internationalization framework

Contemporary software runs in cooperation with a platform. The platform controls its operation while software uses the services provided by platform: file system, memory management services, etc. Services of the internationalization framework complement the services of platform. Both of them play intermediary role between application and low level services. Interaction between application, internationalization framework and platforms is presented in Fig 8. The scheme has three alternatives:

1. Application uses already internationalized services of the platform through internationalization framework API. Application does not use internationalized services of platform directly, what makes application more portable with regard to different platforms.

2. Services of platform are not internationalized or limitedly internationalized therefore their functionality has to be internationalized inside the internationalization framework.

3. Platform does not implement services. Thus these services should be internationalized within the internationalization framework.
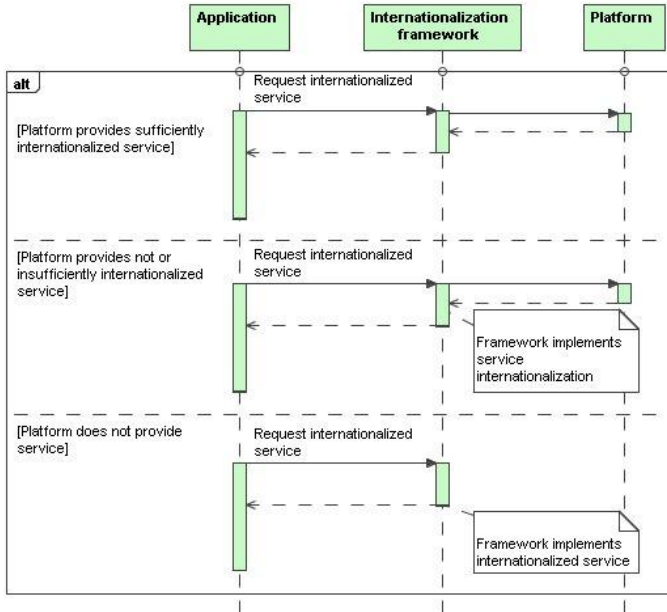
Fig 8. Internationalization framework interaction with platform

The services of internationalization framework may be accessed using its API. It should be pointed out that considering contemporary technologies of software creation possibly the best way to implement this API is object oriented programming. Therefore it may be achieved independence of internationalization framework from particular programming language by using component programming technologies. Hence, different software developers may be interested and involved into its development.

The components of internationalization framework are discussed more thoroughly in the dissertation. Here we will briefly review them.

The main concept of locale data component implementation is locale. Locale is the definition of the subset of a user's information technology environment that depends on language, territory, or other cultural customs (ISO/IEC 15897). There are different locale models to describe locale data. Some of them are described by international standards: POSIX (ISO/IEC 9945, IEEE Standard 1003), C++ (ISO/IEC 14882), FDCC-set (ISO/IEC 14652), the others by de-facto standards: Windows, Java locale models. We propose use of locale model conforming to *UTS 35 Locale Data Markup Language (Common Locale Data Repository)*. Technique proposed in RFC 3066 or XPG4 documents should be used for identification of locales.

14

One of the most important and frequent mistakes appearing in the software development is choice of an 8-bit data encoding method. This method is not usable to encode a multilingual text and has many other shortcomings, so instead of it one of the Unicode encoding methods has to be chosen. Earlier computers did not support Unicode enough because of their limited capabilities but today these problems are not relevant anymore and there are no more obstacles for the use of Unicode.

Text processing has to be realized by implementing the Unicode standard, its appendices and technical standards: *UTS 22 CharMapML, UTR 23 Character Property Model, UAX 24 Text Boundries, UAX 21 Case Mappings, UTR 17 Character Encoding Model, UAX 15 Unicode Normalization, UAX 9 The Bidirectional Algorithm*, etc. However the realization does not limit itself to application of these documents, because it also depends on locale data and other factors.

There are a few of methods used for resource separation: *.RC, .RESX*, GetText, ResourceBundles. All of them have their merits and demerits. The main principles of realization of the service of resource separation are proposed:

- Separated resources can be safely outsourced to a vendor of localization services.
- Localization of resources cannot break the software functionality.
- Tools of visual localizations can be used.

The localization of programming language lexis is not a common phenomenon yet. There are very few compilers that allow that in the world. But this phenomenon is inevitable in future because internationalization and localization of programming language lexis gives it an advantage and improves usability. The localized lexis is simpler, easier to acquire, understand and memorize. Elements of the programming language written in vernacular are more natural and easier to comprehend. That is especially important if a programming language is used for learning purposes when one is seeking deeper objectives not only improvement of practical programming skills. To this end the method for separation of lexical elements from the source code that helps to solve possible problems related with compatibility of localized compilers is proposed.

5. Experimental internationalization of compiler

An experiment of Free Pascal compiler internationalization is presented in this section.

The main objectives of this experiment are:

1. To collect additional knowledge related with the internationalization of compilers.

2. To demonstrate the practical application of the collected knowledge.

3. To internationalize the Free Pascal compiler. This compiler is used in the schools of Lithuania, so its internationalization will improve it and will make it more suitable for pedagogical application.

The Free Pascal compiler has been chosen because it is one of the most popular compilers of the Pascal language nowadays: it is used for teaching in many countries (it is used for teaching in Lithuania as well); it is used in the international Olympiads in informatics. Moreover, its adaptation (including localization) to the programming course in Lithuanian schools has revealed that it still has many internationalization shortcomings that make its localization rather complex.

The Free Pascal compiler was not only internationalized but also its integrated development environment – FPS was created. The primary purpose of it is application to teaching programming. Therefore it has been developed with regard to the defined criteria of simplicity, consistency, functionality, stability, user friendliness, and internationality. The internationality criterion has been satisfied by implementing the results of the dissertation. This environment helps successfully utilize the advantages offered by the compiler internationalization.

The defects of Free Pascal compiler internationalization were arranged according to by their importance to the school course of informatics. Their elimination tasks have been divided into these stages: 1) data encoding; 2) input and output; 3) elements of lexis; 4) elements of locale; 5) separation of resources; 6) services of platform. The projected tasks have been successfully completed and the results have been presented.

The main conclusions of the experiment are:

1. The colleted knowledge has been successfully applied and thus the Free Pascal compiler has been successfully internationalized without major difficulties.

2. Not only Free Pascal compiler has been internationalized, but the internationalized programming environment – FPS has been created. It helps to utilize the advantages of the compiler internationalization.

General conclusions and results

1. The method for evaluating compiler internationalization level has been proposed. The internationalization level of compilers has been studied by means of it. Study revealed that the level of internationalization of compilers is very low. In average the compilers fully satisfied only 10 % of requirements, partly satisfied 58 % and do not satisfied 32 % of requirements. Because of that internationalization of software requires additional expenditure that may reach 10–20 % of total software development cost.

2. It was established that many elements of software internationalization can be advanced to the upper level i.e. the internationalization of compilers,

thus increasing internationalization level of software that is developed by using that compiler and decreasing the expenditures required for its internationalization.

3. The method for internationalization of compilers has been developed and presented. It is supported by knowledge collected during theoretical analysis, study of compilers internationalization level and experimental compiler internationalization.

4. The Free Pascal compiler has been experimentally internationalized. This compiler is used in the schools of Lithuania for the teaching purpose and internationalization gives an advantage as compared to other non-internationalized compilers. The graphical development environment for it compiler has been created.

List of published works on the topic of the dissertation

1. LAUCIUS, R. Issues of Selecting a Programming Environment for a Programming Curriculum in General Education. In R. T. Mittermeir (Ed.). *Lecture Notes in Computer Science*, vol. 4226. *Informatics Education - The Bridge between Using and Understanding Computers*, 2006, p. 169–178. (ISI Master Journal List)

2. DAGIENĖ, V; LAUCIUS, R. Internationalization of open source software: framework and some issues. In T. Boyle, P. Oriogun, A. Pakstas (Ed.). *2nd International Conference Information Technology: Research and Education*, London, London Metropolitan University, 2004, p. 204–207. (ISI Proceedings List)

3. DAGIENĖ, V; LAUCIUS, R. „Free Pascal" panaudojimas informatikos kursui ("Free Pascal" application for informatics course). *Lietuvos matematikos rinkinys*, 2001, 41 (spec. nr.), p. 267–271.

4. LAUCIUS, R; DAGIENĖ, V. Raštinės programinės įrangos „OpenOffice.org" adaptavimas lokalės normoms (Adapting of office software "OpenOffice.org" to locale conventions). *Informacijos mokslai*, 26, 2003, p. 240–245.

5. LAUCIUS, R. Hiperteksto rašyklių palyginimas lokalizavimo galimybių požiūriu (Comparison of hypertext editors by localization possibilities). *Lietuvos matem. rink.*, 43 (spec. nr.), 2003, p. 254–25.

6. LAUCIUS, R. Programinė įrangos vertimo specifika ir dalinis automatizavimas (Specifics and partial automation of software translation). *Lietuvos matem. rink.*, 44 (spec. nr.), 2004, p. 319–326.

7. LAUCIUS, R. Free Pascal kompiliatoriaus internacionalizavimas (Internationalization of Free Pascal compiler). *Informacijos mokslai*, 34, 2005, p. 302–306.

8. LAUCIUS, R; DAGIENĖ, V. Lokalizavimo kurso projektavimas (Projection of localization course). *Lietuvos matem. rink.*, 45, 2005, p. 213–218.

Short description of the author

Rimgaudas Laucius has finished the Kavarskas secondary school in 1995; Vilnius Pedagogical University in 2001 acquiring a master's degree in informatics. Since 2002 he has been a PhD student at the Institute of Mathematics and Informatics. His area of interests includes programming methodology, software localization, and internationalization methodology. He is a member of the Lithuanian Computer Society and Lithuanian Mathematical Society.

KOMPILIATORIŲ INTERNACIONALIZACIJA

Mokslo problemos aktualumas. „OpenOffice.org", „Mozilla", „AbiWord" ir kt. programų lokalizavimo (lietuvinimo) patirtis parodė, kad net tarptautinei rinkai skirta programinė įranga dažnai nepakankamai internacionalizuota, todėl reikia nemažai sąnaudų ją lokalizuojant. Nagrinėjant žemo programinės įrangos (PĮ) internacionalizuotumo lygio priežastis ir ieškant šios problemos sprendimo būdų buvo iškeltos kelios hipotezės.

Hipotezės
1. Šiuo metu naudojami kompiliatoriai nėra pakankamai internacionalizuoti.
2. Programinės įrangos internacionalizavimo sąnaudas galima sumažinti jos gamybai naudojant internacionalizuotą kompiliatorių.
3. Dalį programų internacionalizavimo elementų galima perkelti į aukštesnį lygį – kompiliatorių internacionalizavimą.

Darbo uždaviniai
1. Išanalizuoti mokslinę ir metodinę literatūrą programinės įrangos internacionalizavimo klausimais, išgvildenti ir susisteminti teorinius internacionalizavimo aspektus.
2. Internacionalizacijos aspektu aptarti ir palyginti kelis daugiausia naudojamus kompiliatorius.
3. Nustatyti kompiliatorių internacionalizavimo problemas, identifikuoti svarbiausias sritis (vietas) ir aspektus, juos susisteminti.
4. Parengti kompiliatorių internacionalizavimo metodą.
5. Eksperimentiniu būdu dalinai internacionalizuoti pasirinktą kompiliatorių.

Darbo tikslas. Išanalizuoti kompiliatorių internacionalizacijos elementus, sukaupti ir susisteminti kompiliatorių internacionalizacijos žinias ir pateikti metodą, kuriuo remiantis būtų kuriama internacionalizuota kompiliatorių programinė įranga.

Tyrimų metodai. Darbe naudoti šie metodai: 1) informacijos paieška ir sisteminimas, analizė ir apibendrinimas, 2) atvejų analizė, 3) testas, 4) lyginamoji duomenų analizė, 5) eksperimentas.

Mokslinis naujumas. Ši disertacija yra pirmasis darbas, kuris išsamiai nagrinėja kompiliatorių internacionalizavimo problemas. Kompiliatorių internacionalizacijos tema apskirtai nauja – egzistuoja tik netiesiogiai su ja susiję mokslinės ir metodinės informacijos šaltiniai.

Dauguma šaltinių pateikia PĮ internacionalizavimą kaip procesą papildantį PĮ gamybos procesą ir bando spręsti internacionalizavimo problemas ieškant efektyvesnių, kokybiškesnių šio proceso modelių. Tačiau tai problemų iš esmės neišsprendžia – PĮ internacionalizacija išlieka sudėtingu, daug investicijų reikalaujančiu procesu. Šiame darbe siūloma problemas spręsti nuo pradinės PĮ gamybos pakopos, t. y. įdiegiant PĮ internacionalizavimo metodus PĮ gamybai naudojamose priemonėse.

Šioms idėjos įgyvendinti darbe pateiktas metodas kompiliatorių internacionalizuotumo lygiui įvertinti. Juo remiantis išnagrinėti kompiliatorių internacionalizuotumo lygiai ir nustatyta įtaka jais kuriamos programinės įrangos internacionalizuotumui. Pateikta kompiliatorių internacionalizavimo metodika. Visa tai paremta naujausiais informacijos šaltiniais, standartais, PĮ gamybos metodais, technologijomis. Taip pat bus atsižvelgta į vyraujančią praktiką ir tendencijas PĮ internacionalizavimo srityje.

Praktinė vertė. Disertacijos aktualumas remiasi praktiniais poreikiais. Pateikta metodika ir sukauptos žinios galės būti naudojamos kompiliatorių internationalizavimui ir padės spręsti PĮ internacionalizavimo problemas. Taip pat joje pateikiamas naujas metodas skirtas kompiliatorių internacionalizuotumo lygio įvertinimui.

Aprobavimas. Disertacijos rezultatai pateikti 8 mokslinėse publikacijose ir pristatyti 12 konferencijų:

- • 2 publikacijos išspausdintos leidiniuose įtrauktuose į ISI mokslinių leidinių sąrašus.
- • 6 publikacijos išspausdintos recenzuojamuose moksliniuose leidiniuose.

Darbo turinys

1 skyrius („Įvadas"). Tai įvadinis skyrius, kuriame pateikiama tyrimo tikslai ir uždaviniai, tyrimo aktualumas ir naujumas, tyrime naudoti metodai, gauti rezultatai, paskelbtos publikacijos.

2 skyrius („Teorinės tyrimo prielaidos"). Šiame skyriuje analizuojami ir apžvelgiami su problemine sritimi susiję informacijos šaltiniai. Pateikiama internacionalizacijos samprata, trumpai apžvelgiama šios sampratos raida. Apžvelgiami PĮ internacionalizuotumo vertinimo metodai, nagrinėjamos jų ypatybės, apžvelgiami šaltiniai, pateikiantys PĮ internacionalizuotumo vertinimo tyrimus, nagrinėjamos žemo internacionalizuotumo lygio priežastys ir pateikiamos išvados.

3 skyrius („Kompiliatorių internacionalizuotumo lygio tyrimas"). Šiame skyriuje aprašomas kompiliatorių internacionalizuotumo įvertinimo metodas ir atliktas kompilatorių internacionalizuotumo lygio tyrimas. Apžvelgiami

kriterijai, lėmę konkrečių kompiliatorių kaip tyrimo objekto pasirinkimą. Tyrimo metu nustatytas gana žemas internationalizuotumo lygis, todėl išsamiau nagrinėjami pastebėti kompiliatorių internacionalizuotumo trūkumai bei jų priežastys ir pateikiamos išvados.

4 skyrius („Kompiliatorių internacionalizavimo metodas"). Šiame skyriuje pateikiamas kompiliatorių internacionalizavimo metodas. Skyriaus pradžioje apžvelgiami kultūriniai ir kalbiniai faktoriai įtakojantys PĮ internacionalizaciją. Pagrindinis dėmesys skiriamas įvairių raštų realizacijos klausimams. Trumpai apžvelgiami kalbiniai PĮ internacionalizacijos elementai. Tai leidžia geriau suprasti pateiktą kompiliatorių internacionalizavimo metodą. Yra nagrinėjama duomenų kodavimo, kultūrinių elementų realizacijos, lokalizuotinų išteklių atskyrimo klausimai. Daugumą veiksmų ir duomenų naudojamų internacionalizuojant PĮ galima panaudoti pakartotinai, todėl tikslinga juos apjungti į nuosekliai realizuotą struktūrą tam, kad ją būtų lengviau panaudoti. Pateiktas tokios struktūros karkasas. Išnagrinėta ir aprašyta šį karkasą sudarančių komponentų realizacija. Taip pat nagrinėjamas lokalizuotinų leksikos elementų internacionalizavimas. Pateikiamos rekomendacijos leksikos elementų atskyrimui, leidžiančios išspręsti kylančias lokalizuotų kompiliatorių tarpusavio suderinamumo problemas.

5 skyrius („Eksperimentinis kompiliatoriaus internacionalizavimas"). Šiame skyriuje pristatomas eksperimentas atliktas internacionalizuojant *Free Pascal* kompiliatorių. Apžvelgti eksperimento tikslai ir eksperimento objektas. Darbo metu ne tik internazionalizuotas *Free Pascal* kompiliatorius bet ir sukurta jam skirta programavimo terpė FPS. Ji sukurta remiantis darbo rezultatais ir leidžia geriau išnaudoti kompiliatorius internacionalizacijos teikiamus pranašumus. *Free Pascal* kompiliatoriaus internacionalizacijos išplėtimai suskirstyti į keletą jų realizavimo etapų ir sistemingai apžvelgti. Pateikiama eksperimento metu sukaupta patirtis ir išvados.

Bendrosios išvados ir rezultatai

1. Sukurtas ir pateiktas kompiliatorių internacionalizuotumo lygio vertinimo metodas. Juo remiantis atliktas kompiliatorių internacionalizuotumo tyrimas atskleidė itin žemą jų internacionalizacijos lygį. Ištirti kompiliatoriai pilnai tenkino tik 10 %, nepakankamai tenkino 58 % ir visai netenkino 32 % reikalavimų. Nustatyta, kad kompiliatorių internacionalizuotumo lygis įtakoja jais kuriamos programinės įrangos internacionalizuotumo lygį. Dėl to kuriant šiais kompiliatoriais programinę įrangą jos internacionalizavimui būtinos papildomos sąnaudos, kurios gali sudaryti 10–20 % programinės įrangos gamybos sąnaudų.

2. Nustatyta, kad didelę dalį programų internacionalizavimo elementų galima perkelti į kompiliatorių internacionalizavimą ir taip padidinti jais

gaminamos programinės įrangos internacionalizuotumo lygį bei taip sumažinti jos internacionalizavimui reikalingas sąnaudas.

3. Sukurtas ir pateiktas siūlomas kompiliatorių internacionalizavimo metodas, kuris yra paremtas susistemintomis, naujausiomis kompiliatorių internacionalizavimo srities žiniomis, sukauptomis teorinės analizės, atliktų kompiliatorių internacionalizuotumo lygio tyrimo ir eksperimentinio kompiliatoriaus internacionalizavimo metu.

4. Eksperimentiškai internacionalizuotas kompiliatorius Free Pascal. Šis kompiliatorius naudojamas mokymui Lietuvos mokyklose, o internacionalizacija suteikia jam pranašumų. Sukurta grafinė internacionalizuota programavimo terpė į kurią integruotas internacionalizuotas Free Pascal kompiliatorius. Ji leidžia geriau išnaudoti kompiliatoriaus internacionalizacijos teikiamus pranašumus ir sėkmingiau taikyti jį mokymui.

Trumpos žinios apie autorių

Rimgaudas Laucius 1995 metais baigė Kavarsko vidurinę mokyklą. 2001 metais Vilniaus pedagoginiame universitete įgijo informatikos magistro laipsnį. Nuo 2002 metų yra Matematikos ir informatikos instituto doktorantas. Jo tyrimų sritis apima programavimo metodologiją, programinės įrangos lokalizavimo ir internacionalizavimo metodologiją. Jis yra kompiuterininkų sąjungos ir Lietuvos matematikų draugijos narys.

Padėka