



Universitatea Tehnică a Moldovei

**LIMBAJ DE PROGRAMARE SPECIFIC AGENȚILOR
INTELIGENȚI
SPECIFIC PROGRAMMING LANGUAGE FOR
INTELLIGENT AGENTS**

Student:

Iuzvac Anatolie

Conducător:

**Braga Vasile
lect. univ.**

Chișinău, 2020

REZUMAT

Evoluția complexității continuă a sistemelor software rezultă în dezvoltarea continuă a instrumentelor utilizate de către dezvoltatori pentru dezvoltarea și întreținerea acestor sisteme software. Se poate observa o corelație strânsă între creșterea nivelului complexității problemelor ce apar în dezvoltarea sistemelor software moderne și nivelul de abstractizare a limbajelor de programare utilizat în dezvoltarea sistemelor. Limbajele de programare specific domeniului reprezintă cel mai înalt nivel de abstractizare a limbajelor de programare și astfel minimizează la maxim efortul necesar pentru soluționarea problemelor din domeniul specific astfel creând premisele necesare dezvoltării ulterioare ale acestui domeniu.

Obiectivele principale ale acestui proiect sunt: (1) crearea unui limbaj de programare specific domeniului agenților inteligenți, (2) crearea unui interpretor simplu pentru a verifica posibilitatea integrării limbajului specific nou creat în soluțiile software finale.

Pentru crearea limbajului de programare specific domeniului agenților inteligenți de tip textual a fost aplicată metoda manuală de creare a gramaticii formale și generarea ulterioară a parserului și lexerului conform regulilor specificate în gramatică utilizând instrumentul ANTLR.

Ca rezultat a fost creat un limbaj de programare specific agenților inteligenți ce are ca scop minimalizarea timpului și a efortului pentru creare și integrarea agenților inteligenți în sistemele multi agent. Datorită creării interpretorului a fost demonstrată utilitatea acestuia în produsele software finale.

Analiza teoretică a creării unui limbaj de programare specific sunt descrise în continuare în primul capitol al acestei lucrări, descrierea în detalii a instrumentelor și metodei utilizate pentru crearea limbajului de programare specific sunt regăsite în capitolul doi a lucrării.

Capitolele trei și patru rezumă partea practică ale acestei lucrări. Capitolul trei descrie procesul de creare a limbajului de programare specific agenților inteligenți de la definirea structurii și a sintaxei limbajului specific și ulterior crearea gramaticii limbajului specific agenților inteligenți. Capitolul patru descrie procesul de creare a interpretorului pentru limbajul specific nou creat pentru demonstrarea utilizării ulterioare a limbajului specific în produsele software finale.

Cuvinte-cheie: Programare orientată pe agenți, agenți inteligenți, limbaj de programare specific domeniului.

ABSTRACT

Continuous evolution of the complexity of software systems results in continuous development of used tools by developers for the development and maintenance of respective software systems. Can be observed a tight correlation between increase in complexity level of the problems which occur in the development of modern software systems and the level of abstraction of programming languages used in development of systems. Domain-specific programming languages represents the highest level of programming language abstraction and thus minimizes the effort required for solving the problems from the specific domain, thus creating the necessary premises for the further development of this domain.

The main objectives of this project are: (1) creation of a programming language specific to the domain of intelligent agents, (2) creation of a simple interpreter to verify the integration of the newly created programming language in the final software solutions.

For the creation of programming language specific to the domain of intelligent agents was applied the manual method which implies the creation of a formal grammar and the subsequent generation of the parser and lexer according to the rules specific in the grammar using the ANTLR tool.

As a result, has been created a programming language specific to the domain of intelligent agents that aims to minimize the time and effort required to create and integrate intelligent agents into multi-agent systems. Due to the creation of the interpreter, the usefulness of the domain-specific programming language in the final software products was demonstrated.

The theoretical analysis of the creation of a specific programming languages are described below in the first chapter of this work, the detailed description of the tools and method used to create the specific programming language are found in the second chapter of this work.

Chapters three and four summarize the practical part of this paper. Chapter three describes the process of creating the programming language specific to the domain of intelligent agents from defining the structure and syntax of the specific language and then creating the grammar of the language specific to the domain of intelligent agents. Chapter four describes the process of creating the interpreter for the newly created specific programming language to demonstrate the subsequent use of the specific language in the final software products.

Keywords: Agent-oriented programming, intelligent agents, domain-specific programming language.

CUPRINS

LISTA TABELELOR.....	8
LISTA FIGURILOR.....	9
INTRODUCERE.....	10
1 ANALIZA PROIECTULUI.....	12
1.1 Limbaje de programare orientate pe agent.....	12
1.2 Gramatică, limbaj formal și parsare.....	13
1.3 Limbaj de programare specific domeniului.....	15
1.4 ANTLR.....	17
1.5 Agenți Inteligenți.....	19
1.6 Concluzii.....	21
2 INSTRUMENTUL ȘI METODELE UTILIZATE ÎN DEZVOLTAREA LIMBAJULUI.....	23
2.1 Limbaje de programare orientate pe agent.....	23
2.2 Lexer și parser.....	24
2.3 Gramatică.....	26
2.4 Concluzii.....	29
3 CREAREA LIMBAJULUI SPECIFIC AGENȚILOR INTELIGENȚI.....	31
3.1 Crearea structurii, sintaxei și a logicii limbajului de programare specific.....	31
3.2 Definirea regulilor lexerului în gramatica ANTLR.....	34
3.3 Definirea regulilor parserului în gramatica ANTLR.....	36
3.4 Testarea gramaticii și limbajului de programare specific creat.....	39
3.5 Concluzii.....	42
4 TESTAREA LIMBAJULUI ÎN PRODUSUL SOFTWARE FINAL.....	44
4.1 Setarea proiectului Java utilizând ANTLR.....	44
4.2 Crearea listener-ului pentru parcurgerea arborelui de parsare.....	45
4.3 Crearea interpretorului.....	49
4.4 Testarea interpretorului.....	51
4.5 Concluzii.....	53
CONCLUZII GENERALE.....	55
BIBLIOGRAFIE.....	57

INTRODUCERE

Agenții inteligenți au devenit elemente fundamentale utilizate de zi cu zi, deși mulți dintre utilizatori nu realizează existența acestora. Precum în telefoanele mobile asistenții virtuali precum Siri, Alexa etc, utilizează agenții inteligenți care performă acțiuni în conformitate cu senzorii telefonului mobil și informații accesibile pe internet. Aplicabilitatea agenților inteligenți este enormă de la căutarea informației pe internet în motoarele de căutare, analiza schimbării prețurilor la un produs în diverse magazine până la platforme de zbor automat a avioanelor, sistem inteligent de sprijin pentru aterizarea aeronavelor, etc.

Un agent poate fi considerat un obiect autonom, care poate lua decizii și inițiativa datorită capacității de percepție a mediului de execuție în care este situat. Agenții pot fi utilizați în medii complexe, imprevizibile datorită adaptării dinamice acestora în urma unui proces de învățări.

Se observă o corelație practic liniară între creșterea complexității problemelor necesar de rezolvat și nivelul de abstractizare a limbajelor de programare. Creșterea nivelului de abstractizare a limbajelor de programare asigură posibilitatea apariției sistemelor mai complexe în termen mai scurt. Limbajele de programare pot fi împărțite pe generații după nivelul de abstractizare a acestora. Prima generație de limbaje de programare reprezintă limbaje mașină ce constau în adresarea directă către procesorul sistemului, de exemplu prin cartele perforate. A doua generație de limbaje reprezintă limbajele de asamblare care au abstractizat nivelul comunicării direct cu procesorul și au creat abilitatea de a utiliza simboluri în loc de numere. Următoarea generație a limbajelor de programare, limbajele de nivel înalt sunt limbajele de programare utilizate pe scară largă în momentul de față, ele conțin instrucțiuni mai apropiate limbajului natural omului. A patra și ultima generație de limbaj de programare aparține generatoarelor de aplicații, ce pot genera cod sursă pe baza unui dialog cu programatorul pentru stabilirea anumitor detalii. Aceste limbaje sunt specifice domeniilor respective întrucât sunt necesare cunoștințe în domeniul de aplicație respectiv.

Limbajele de programare pentru domenii specifice reprezintă cel mai înalt nivel de abstractizare întrucât minimizează la maxim cunoștințele necesare de către programator pentru crearea software-ului și efortul necesar pentru crearea software-ului în domeniul respectiv. La bazele dezvoltării rapide a domeniilor specifice se pot observa prezența limbajelor de programare destinate acestor domenii. În management-ul bazelor de date, SQL reprezintă un limbaj de programare destinat domeniului dat ce a rezultat în disponibilitate utilizării acestui limbaj de către specialiști fără obligația de a avea cunoștințele unui dezvoltator. În dezvoltarea paginilor și aplicațiilor web limbajele ce stau la baza precum HTML și CSS de asemenea reprezintă limbaje specifice domeniului, datorită acestor limbaje efortul necesar creării unei pagini web a fost minimalizat întrucât fiecare copil poate în doar câteva săptămâni crea prima sa pagină web și pe zi ce trece apar tot mai complexe aplicații web.

Agenții inteligenți pot fi utilizați în numeroase sisteme complexe pentru a mări eficiența și posibilitățile sistemului, domeniul agenților și programarea orientată pe agent este vitală pentru dezvoltarea continuă a sistemelor complexe utilizate precum internetul obiectelor, aplicațiile software a telefonului, etc. Este vitală crearea unui limbaj specific de programare domeniului agenților pentru a mări potențialul agenților inteligenți și pentru a permite dezvoltarea agenților inteligenți mai complexi în termen scurt, pentru a crea sisteme cu încă și mai multe funcționalități disponibile utilizatorilor finali.

BIBLIOGRAFIE

1. Shoham, Y., Agent-oriented programming. In: Artificial Intelligence. 1993, nr. 60, p.51-92. ISSN 0004-3702.
2. Hindriks, V. K., Programming Rational Agents in GOAL. Multi-Agent Programming: Languages, Platforms and Applications. Boston: Springer US, 2009. p.119-157. ISBN 978-0-387-89299-3.
3. Rodriguez, S., Gaud, N., Galland, S., SARL: a general-purpose agent-oriented programming language. In: International Joint Conferences on Web Intelligence and Intelligent Agent Technologies, 11-14 august 2014. Varşovia: IEEE, 2014. ISBN 978-1-4799-4143-8.
4. Levelt, W.J.M., Formal Grammars in Linguistics and Psycholinguistics. Amsterdam: John Benjamins Publishing Company, 2008, 582p. ISBN 978-902-723-251-9.
5. Fowler, M., Domain-Specific Languages. Pearson: Addison-Wesley Professional, 2010. 640p. ISBN 978-0321712943.
6. Tomassetti, F., The complete guide to (external) Domain Specific Languages [online]. Strumenta, 20 februarie 2017 [citat 2020-09-25]. Disponibil: <https://tomassetti.me/domain-specific-languages>
7. ANTLR, About The ANTLR Parser Generator, ©2014 [citat 2020-10-01]. Disponibil: <https://antlr.org/about.html>
8. Tomasetti, F., The ANTLR Mega Tutorial [online]. Strumenta, 8 martie 2017 [citat 2020-11-03]. Disponibil: <https://tomassetti.me/antlr-mega-tutorial>
9. Leon, F., Agenți inteligenți cu capacități cognitive. Iași: Tehnopress, 2006. 253p. ISBN: 978-973-702-422-0.
10. Parr, T., The Definitve ANTLR4 Reference. Raleigh: The Pragmatic Programmers, 2013. 328p. ISBN 978-1-93435-699-9.