

Crearea automatizata a procesoarelor adaptabile de nivelul trei în baza formalismului E-T-M de interactiune translatorica a procesoarelor adaptabile

Diana MICUSA, Universitatea de Stat, Republica Moldova

Manfred KUDLEK, Universitatea Hamburg, Germania

Dumitru TODOROI, Zinaida TODOROI, ASE Chisinau, Republica Moldova

E-mail: micusa@mednet.md

Development of E-T-M [Mic-02] formalisms for adaptable third level translation interactions is presented. Are created and alg orithmically proved and demonstrated the possibility to obtain automatically the Adaptability Cube' third level's Adaptable Processors. Using three types of the first level's Adaptable Processors [DTod-03] and the three types of the second level's Adaptable Processors [ZTod-03] are obtained the Adaptable Processors-Compilers of the third level of translation complexity. In the communication are presented the demonstrations of automatically obtaining the third level of Adaptable Processors-Compilers. These results present the logic continuation of the results, presented in the communication [ZTod-03], concerning the process of automatically obtaining the Second levels of translation complexity Adaptable Processors, and the communication [DTod-03] concerning the First level of Adaptable Processors.

Keywords: level, processor, communication, translation.

Introducere

Procesoarele adaptabile sunt repartizate pe trei niveluri de complexitate translatorica. Nivelul unu [DTod-03] îl reprezinta procesoarele adaptabile de tip Timp-Realizare al extinderelor: Pre-procesoarele, Inter-procesoarele si Post-procesoarele. De acelasi nivel unu apartin si procesoarele adaptabile de tip Model-Realizare ale extinderelor: realizarea extinderilor de tip Nivel-Nivel, Nivel-Direct si Nivel-Nivel-Direct. Al treilea set de procesoare adaptabile de nivelul unu îl constituie procesoarele adaptabile de tip Tip-Translatore adaptabile: Compilare adaptabila, Interpretare adaptabila si Compilare-Interpretare adaptabila a extinderilor.

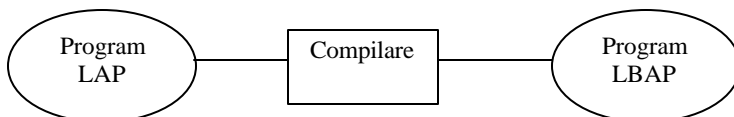
Nivelul doi [ZTod-03] de procesoare adaptabile îl reprezinta procesoarele de tip Timp-Model-Realizare, Tip-Timp-Realizare si Tip-Model-Realizare a extinderelor. Ele formeaza al doilea nivel de complexitate translatorica din cubul adaptabilitatii.

Lucrarea actuala prezinta rezultatele cercetarii si implementarii procesoarelor adaptabile-compilatoare de nivelul trei de complexitate

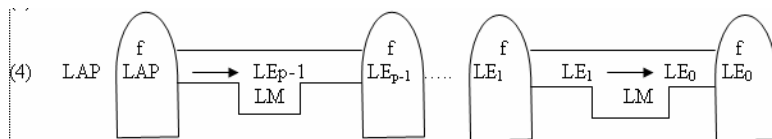
translatorica din cubul adaptabilitatii. Aceste procesoare sunt prezentate în actuala lucrare prin intermediul Pre-procesoarelor-Compilatoare adaptabile, Inter-procesoarele-Compilatoare adaptabile si Post-procesoarele-Compilatoare adaptabile. Teoremele de obtinere automatizata a acestor procesoare adaptabile de nivelul trei de complexitate translatorica sunt demonstrate prin intermediul formalismului E-T-M de interactiuni ale procesoarelor adaptabile.

Preprocesoare-compilatoare adaptabile de nivelul trei

Un procesor adaptabil de nivelul trei de tipul preprocesor-N-N-compilator realizeaza programul f în limbajul adaptabil de programare LAP, utilizând ordinea de procesare a extinderilor, care sunt prelucrate dupa modul de prelucrare nivel-nivel, utilizând tipul de procesare-compilator. Prin urmare, programul sursa la compilare-preprocesare este programul LAP si programul obiect este programul LBAP:

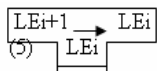


Utilizând formalismul Earley un preprocessor-N-N-compiler este prezentat prin formula (4):

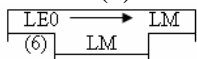


Unde programul f în LAP trece “coborârea” multipla LAP→LEp-1,...,LE2→LE1, LE1→LE0 dupa modul de traducere nivel-nivel, realizând extinderile conform schemei de procesare a lor în procesarea de tip compilator.

Teorema Preprocesor -N-N-compiler: Este suficient de creat un set (5) de extinderi:

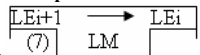


pentru i=0,1,2,...,p-1; p-numarul de niveluri al limbajului adaptabil de programare LAP, si utilizând procesorul de baza (6) dat:

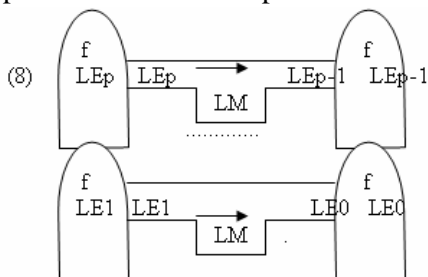


unde LE0-limbajul baza LBAP si LM-limbajul masina, se poate obtine automatizat preprocesorul-N-N-compiler.

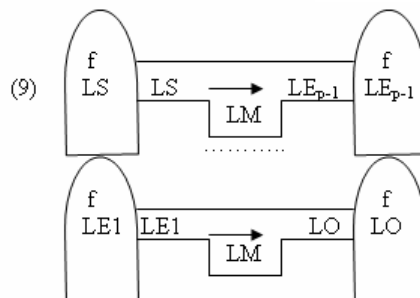
Demonstratie: Utilizam teorema Preprocesor-N-N pentru a primi setul (7) de procesare adaptabile:



pentru i=0,1,2,...,p-1; p-numarul nivelurilor limbajului adaptabil de programare LAP. Utilizam setul (7) de procesoare pentru a obtine formula (8) a preprocesorului-N-N adaptabil:

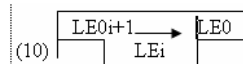


În formula (8) înlocuim LEp (LAP cu p niveluri de extinderi) prin LS si LE0 (LBAP-limbajul de baza adaptabil de programare) prin LO (limbaj obiect). Ca urmare primim formula (9) de functionare a unui preprocesor-N-N-compiler:



Analog pot fi primite automatizat preprocesorul-N-D-compiler si preprocesorul-N-N-D-compiler prin intermediul teoremei Preprocesor-N-D-compiler si respectiv a teoremei Preprocesor-N-N-D-compiler.

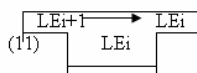
Teorema Preprocesor-N-D-compiler: Preprocesorul-N-D-compiler se poate obtine automatizat din setul (10) de extinderi:



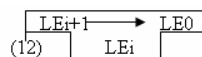
pentru i=0,1,2,...,p-1; p-nivelul LAP, utilizând procesorul de baza (6).

Demonstratie: Se utilizeaza teorema Preprocesor-N-D si, analog ca si în procesul demonstrarii teoremei Preprocesor-N-N-compiler, se obtine preprocesorul-N-D-compiler.

Teorema Preprocesor-N-N-D-compiler: Se poate obtine automatizat preprocesorul-N-N-D-compiler. Este suficient de creat seturile (11) si (12) de extinderi:

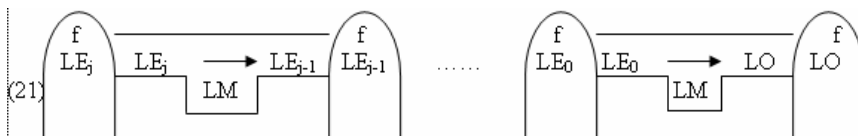


pentru i=j,j+1,...,p-1; p-nivelul LAP si

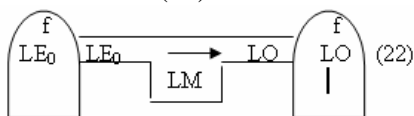


pentru i=0,1,2,...,j-1, utilizând procesorul de baza (6) dat.

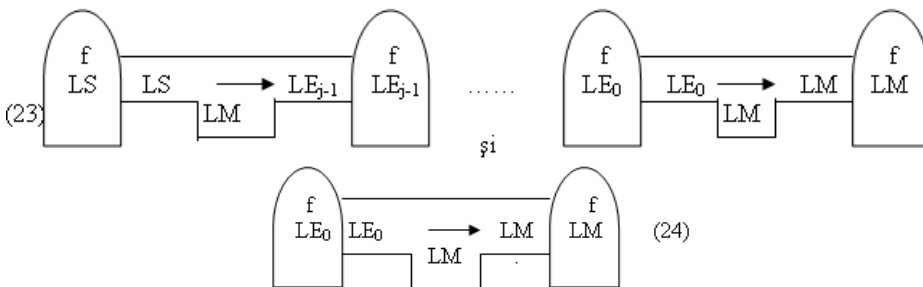
Demonstratie: Se utilizeaza teorema Preprocesor-N-N-D cu schimbul formulelor (2) si (5) prin formulele (11) si (12) pentru a ob-



si, simultan pentru fiecare element de baza conform formulei (22):



obtinem formula complexa pentru interprocesorul-N-N.



În formula (21) înlocuim LEj (LAP cu j niveluri de extinderi) prin LS (limbaj sursa), LO prin LM (limbaj masina) si, în formula (22) LO prin LM pentru a obtine formula complexa (23) si (24) de functionare a unui interprocesor -N-N-compiler:

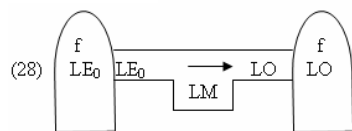
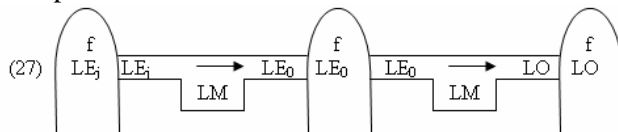
Analog poate fi primit automatizat interprocesorul-N-D-compiler si inter-procesorul-N-N-D-compiler prin intermediul teoremei interprocesor-N-D-com-pilator si a teoremei interprocesor-N-N-D-compiler.

Teorema Interprocesor-N-D-compiler: Fie dat procesorul de baza (19). Este suficient de creat un set (25) de extinderi: $\begin{matrix} LE_{i+1} & \longrightarrow & LE_0 \\ \hline & LE_i & \end{matrix}$ pentru $i=0,1,2,\dots,p-1$; p-nivelul LAP, pentru a obtine automatizat interprocesorul-N-D-compiler.

Demonstratie: Utilizam teorema Interprocesor-N-D pentru a obtine setul (26) de procesoare adaptabile:

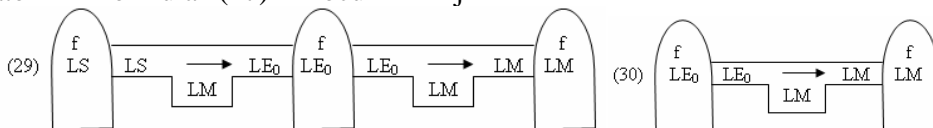


Aceste procesoare, fiind puse în functiune pentru fiecare extindere de nivel j ($j=1,2,\dots,p$) din programul sursa f în LAP conform formulei (27) si, simultan procesorul (19) pentru fiecare element de baza conform formulei (28), obtinem formula interprocesorului-N-D:



Analog cu teorema Interprocesor -N-N-compiler în formula (27) înlocuim LEj

(LAP cu j niveluri de extinderi) prin LS (limbaj sursa), LO prin LM (limbaj masina) si, în formula (28) LO prin LM pentru a obtine formula complexa (29) si (30) de functionare a unui interprocesor-N-D-com-pilator:



Teorema Interprocesor-N-N-D-compiler: Fie dat procesorul (19) si sirurile (31) si (32)

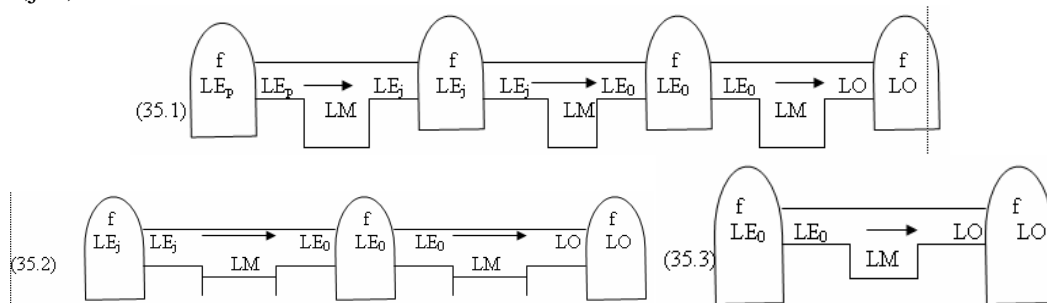
de extinderi: $\begin{matrix} LE_{i+1} & \longrightarrow & LE_j \\ \hline & LE_i & \end{matrix}$ pentru $i=j,j+1,\dots,p-1$; $\begin{matrix} LE_{i+1} & \longrightarrow & LE_0 \\ \hline & LE_i & \end{matrix}$ pentru $i=0,1,2,\dots,j-1$.

Aceste siruri (31) si (32) sunt suficiente pe n-
tru a obtine interprocesorul-N-N-D adaptabil,
prezentat prin intermediul sirului de proce-
soare (33) si (34):

$i=j, j+1, \dots, p-1;$
 $i=0, 1, 2, \dots, j-1,$ care la rândul lor vor fi folosite la

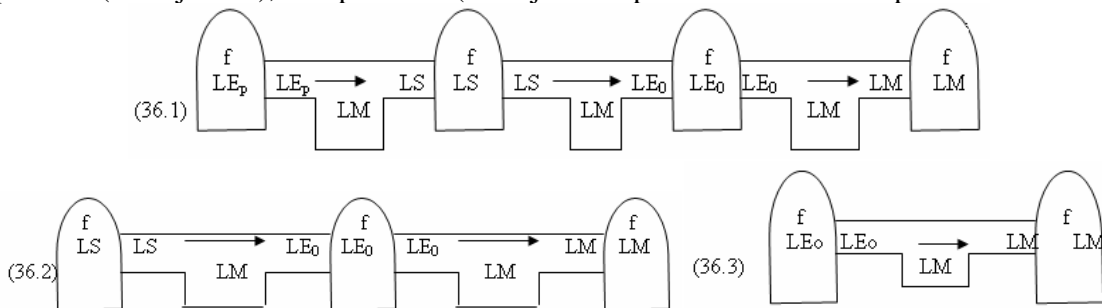
obtinerea automatizata a interpretorului-N-N-
D-compilator.

Demonstratie: Utilizând teorema Interpro-
cesor-N-N-compiler si Teorema Inter-
procesor-N-D-compiler vom obtine formu-
la de functionare a interprocesorului-N-N-D-
compiler:



Analog cu teorema Interprocesor-N-N-
compiler în formula (35.1) si (35.2) înlo-
cuim LE_j (LAP cu j niveluri de extinderi)
prin LS (limbaj sursa), LO prin LM (limbaj

masina) si, în formula (35.3) LO prin LM
pentru a obtine formula complexa (36.1),
(36.2) si (36.3) de functionare a unui
interprocesor-N-N-D-compiler:

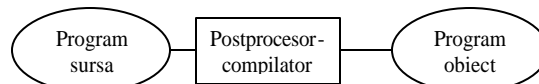


Prin intermediul Teoremei Interprocesor-N-
N-compiler, Teoremei Interprocesor-N-D-
compiler si Teoremei Interprocesor-N-N-
D-compiler sau obtinut automatizat, utili-
zând formalismul E-T-M, al doilea set, com-
pus din urmatoarele trei procesoare de nive-
lul trei al cubului adaptabilitatii (figura 1).

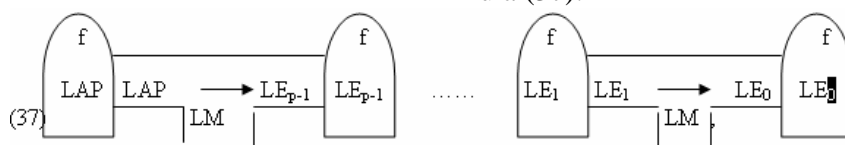
**Procesoare -compilatoare adaptabile de ni-
velul trei cu postprocesare.**

Un procesor adaptabil de nivelul trei de tipul
postprocesor-N-N-compiler

realizeaza programul f scris în limbajul adap-
tabil de programare LAP utilizând ordinea de
postprocesare a extinderilor, care sunt tran-
sferate dupa modul de transfer nivel-nivel, uti-
lizând tipul de procesare-compiler. Pro-
gramul sursa la compilare-postprocesare, prin
urmare, este LAP si programul obiect este
LO:



Utilizând formalismul Earley un postproce-
sor-N-N-compiler este prezentat prin for-
mula (37):



unde programul f în LAP trece “coborârea” multipla $LAP \rightarrow LE_{p-1}, \dots, LE_1 \rightarrow LE_0$ dupa modul de traducere nivel-nivel, realizând extinderile conform schemei de postprocesare a lor în schema de procesare de tip compilator.

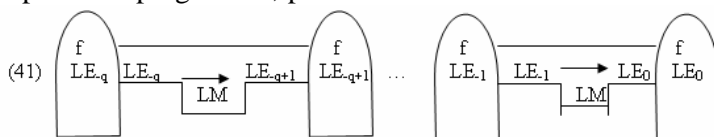
Teorema Postprocesor-N-N-compiler: Fie

$$(38) \begin{array}{c} \boxed{LE_0 \xrightarrow{\quad} LM} \\ \boxed{\quad \xrightarrow{LM} \quad} \end{array}$$

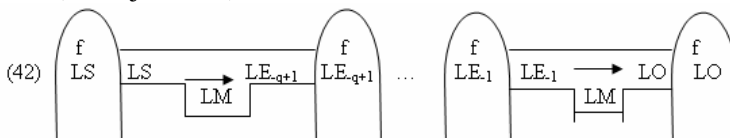
dat procesorul (38). Este suficient de creat un set (39) de extinderi

$$(39) \begin{array}{c} \boxed{LE_{i-1} \xrightarrow{\quad} LE_i} \\ \boxed{\quad \xrightarrow{LE_i} \quad} \end{array}$$

, unde $i=0,-1,-2,\dots,-q$; q reprezinta numarul nivelurilor mai jos de baza a limbajului adaptabil de programare, pen-



Pentru a obtine formula de functionare a unui postprocesor-N-N-compiler, în formula (41) se înlocuieste $LE-q$ (LAP cu extinderi) prin LS (limbaj sursa), LE_0



Teorema Postprocesor-N-D-compiler: Fie dat procesorul (38) si un set (43) de extinderi de niveluri i ($i=-1,-2,\dots,-q$) mai jos de baza:

$$(43) \begin{array}{c} \boxed{LE_{i-1} \xrightarrow{\quad} LE_0} \\ \boxed{\quad \xrightarrow{LE_i} \quad} \end{array}$$

. Atunci este posibil de creat automatizat un set de procesoare

$$(44) \begin{array}{c} \boxed{LE_{i-1} \xrightarrow{\quad} LE_0} \\ \boxed{\quad \xrightarrow{LM} \quad} \end{array}$$

care formeaza postprocesorul-N-D adaptabil necesar, care, la rândul sau, fiind pus în functiune împreuna cu schema compilatorului, formeaza postprocesorul-N-D-compiler.

Demonstrare: Se utilizeaza teorema Postprocesor-N-D si, analogic ca si-n Teorema Postprocesor-N-N-compiler se obtine Postprocesorul-N-D-compiler.

Teorema Postprocesor-N-N-D-compiler: Fie dat procesorul de baza (38), sirurile (45)

$$(45) \begin{array}{c} \boxed{LE_{i-1} \xrightarrow{\quad} LE_i} \\ \boxed{\quad \xrightarrow{LE_i} \quad} \end{array}$$

si (46) de extinderi:

$$(46) \begin{array}{c} \boxed{LE_{k-1} \xrightarrow{\quad} LE_0} \\ \boxed{\quad \xrightarrow{LE_k} \quad} \end{array}$$

$i=-j,-j-1,\dots,-q+1$ pentru

tru a obtine un set de procesoare adaptabile, care formeaza postprocesorul-N-N, care, la rândul sau, va fi folosit la obtinerea automatizata a postprocesorului-N-N-compiler (42).

Demonstratie: Utilizam teorema postprocesor-N-N pentru a obtine setul (40)

$$(40) \begin{array}{c} \boxed{LE_{i-1} \xrightarrow{\quad} LE_i} \\ \boxed{\quad \xrightarrow{LM} \quad} \end{array}$$

de postprocesoare-N-N. Utilizând formalismul E-T-M vom obtine formula de functionare a unui postprocesor-N-N:

(LBAP-limbajul de baza adaptabil de programare) prin LO (limbaj obiect) si se obtine:

$k=-j+1,-j+2,\dots,0$. Poate fi creat automatizat postprocesorul-N-N-D, compus dintr-un set (47) de procesoare de tip N-N si dintr-un set (48) de procesoare de tip N-

$$(47) \begin{array}{c} \boxed{LE_{i-1} \xrightarrow{\quad} LE_i} \\ \boxed{\quad \xrightarrow{LM} \quad} \end{array}$$

D: unde $i=-q+1,\dots,-j$; $j \leq 0$

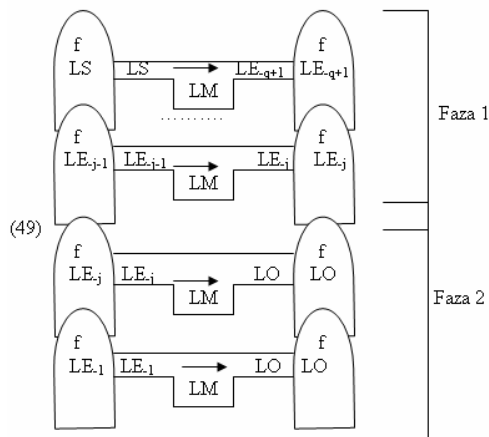
$$(48) \begin{array}{c} \boxed{LE_{k-1} \xrightarrow{\quad} LE_0} \\ \boxed{\quad \xrightarrow{LM} \quad} \end{array}$$

si pentru $k=-j+1,\dots,0$, care, la rândul sau, împreuna cu schema compilatorului va forma postprocesorul-N-N-D-compiler.

Demonstratie: Se utilizeaza Teorema Postprocesor-N-N-compiler si în continuare Teorema Postprocesor-N-D-compiler pentru a obtine postprocesorul-N-N-D-compiler (49).

Prin intermediul Teoremei Postprocesor-N-N-compiler, Teoremei Postprocesor-N-D-compiler si Teoremei Postprocesor-N-N-D-compiler s-au obtinut automatizat, utilizând formalismul Earley, ultimele trei procesoare adaptabile de nivelul trei din primul plan de

procesoare adaptabile al cubului adaptabilitatii (figura 1).



Desenul urmator, care poarta denumirea de *Cubul adaptabilitatii*, prezinta concentrat cele câteva aspecte mentionate pâna acum: tipul de parcurgere, tipul de translare si tipul de procesare. Astfel se obtin 27 de scheme principale posibile de construire a procesoarelor adaptabile de nivelul trei, fiecare cu avantajele si dezavantajele sale.

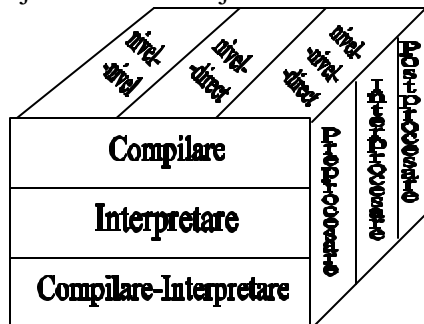


Fig. 1. Cubul adaptabilitatii

Am demonstrat astfel obtinerea automatizata doar a unei parti din procesoarele adaptabile, si anume a celor noua procesoare adaptabile de nivelul unu, a noua procesoare adaptabile de nivelul doi si a noua procesoare de nivelul trei. Aceste procesoare sunt universale si specializate.

În perspectiva se preconizeaza finisarea investigatiilor privind demonstrarea posibilitatilor de creare automatizata a celorlalte procesoare din cubul adaptabilitatii cu aplicarea

demonstrarilor obtinute la organizarea fabricii de productie a procesoarelor adaptabile.

Concluzii si perspective

În lucrarea prezenta au fost demonstrate teoremele de obtinere automatizata a Procesoarelor-Compilatoare adaptabile: Preprocesoare-Compilatoare, Inter-procesoarele-compilatoare si Post-procesoarele-Compilatoare. Demonstratiile au fost efectuate în baza formalismului ET-M de interactiune a procesoarelor adaptabile.

Pre-procesoarele -Compilatoare, Inter-procesoarele-compilatoare si Post-procesoarele-Compilatoare formeaza primul set de procesoare adaptabile de nivelul trei din cubului adaptabilitatii. Urmatoarele procesoare adaptabile de nivelul trei de complexitate translatorica sunt în proces de cercetare si implementare. Acestea sunt Procesoarele-Interpretoare adaptabile si Procesoarele-Compilatoare-Interpretoare adaptabile. Acestora le vor fi consacrate urmatoarele publicatii.

Bibliografie

[DTod-03]. Dumitru Todoroi. The Model Type of Extensions Realization of the Adaptability Cube' First Level of Translation Complexity' Adaptable Processors. // The Sixth International Conference on Economic Informatics IE'03, Romania, Bucharest, May 8-11, 2003, To be published.
 [ZTod-03]. Zinaida Todoroi. The Second Level Adaptable Processors of the Extensible Time-Models Realization. // The Sixth International Conference on Economic Informatics IE'03, Romania, Bucharest, May 8-11, 2003, To be published.
 [Tod-02]. Dumitru Todoroi, Zinaida Todoroi. The Romanian Information Language MULTIMEDIA Dictionaries for European Community. // Proc. of International Conf. "Globalization and University' Economics Education", Vol.2, Iasi, October 24-25, 2002, p. 233-250.