

Utilizari ale metodelor structural-semiotice în realizarea sistemelor informatice (partea a II-a)

Ec. Robert KOMARTIN
Oracle România

The discrepancy between the efficiency of the application development phases against the analysis and design phases in almost all methods and methodologies currently in use it is obvious, especially due to a fundamentally different level of formalization of the two areas: while the programming languages area has a strong theoretical foundation for few decades already (formal grammars, compilers theory, etc.), the theoretical foundation of the analysis – and also of the design – are still at a ‘folklore and craftsmanship’ level.

Moreover, due to this discrepancy, which generates huge differences in operational performance, the focus of the ‘practicians’ moved towards these phases/moments in the development process. Consequently – and correlated with the lack of adequate mathematical support – the appearance of a ‘de-mathematization’ of the information systems development methods was created – the modern system analysis and IS project management courses are rather belonging to the social sciences field (sociology, ethnology), rather than the field originating them – computer science.

That’s why we consider that what is actually necessary to improve the performances of the information systems development processes is not ‘de-mathematization’ but rather ‘re-mathematizing’ them in a proper manner. One possible way – presented below – in order to obtain significant increase in methodological quality is a structuralist and semiotic approach of the information systems development processes. The proposed approach is based on semiotic algebra, as defined by the group led by prof. Joseph Goguen at the University of San Diego.

Keywords: information systems, structuralism, semiotics, semiotic algebra, semiotic morphisms, semiotic projections, best practice and representation support systems.

2.3 Elemente de calculul reprezentarilor

Morfismele semiotice pot fi compuse; acest fapt este important deoarece este foarte obisnuita compunerea reprezentarilor (de exemplu în procesele de proiectare iterativa, unde detaliile sunt adaugate în stadii succesive). Compunerea morfismelor semiotice satisface câteva ecuatii simple, care demonstreaza ca sistemele de semne, împreuna cu morfismele semiotice formeaza ceea ce se numeste o *categorie* (în sensul teoriei categoriilor, mai sus amintite). Pentru mai multe detalii, a se consulta lucrarile [GOGU91] si [GOGU99].

Definitie Fiind date morfismele $M: S_1 \otimes S_2$ si $M': S_2 \otimes S_3$ atunci **compozitia** lor, notata $M; M': S_1 \otimes S_3$ este formata prin compunerea functiilor componente ale fiecarui morfism, luând în considerare caracterul partial al acestora.

Se demonstreaza usor ca si $M; M'$ este un morfism semiotic.

Observatie: Notatia $;$ este folosita în locul celei traditionale \circ pentru a sugera ca ordinea de efectuare (parcurs) a celor doua morfisme este inversa decât ordinea de compunere uzuala ($M; M'$ înseamna *mai întâi* M *si* *mai apoi* M').

Daca este dat $M': S_3 \otimes S_4$ atunci $(M; M'); M'' = M; (M'; M'')$ sau, cu alte cuvinte, compunerea morfismelor semiotice este **asociativa**.

Pentru orice sistem de semne S , exista un morfism semiotic **identitate** $1_S: S \otimes S$, care este definit ca având ca si componente functii identitate pe fiecare multime din S . Pentru orice $M: S_1 \otimes S_2$ sunt satisfacute urmatoarele ecuatii:

$$1_S; M = M \text{ si } M; 1_S = M.$$

Compunerea si identitatea morfismelor semiotice ne permit sa dam urmatoarea definitie:

Definitie: Un morfism $M: S_1 \otimes S_2$ pentru care exista un morfism **invers** $M': S_2 \otimes S_1$ (adica $M; M' = 1_S$ si $M'; M = 1_S$) se numeste **izomorfism** si vom nota $M \gg M'$. Sistemele de

semne izomorfe definesc exact aceleasi structuri (mai precis teorii) si au exact aceleasi reprezentari.

Din proprietatile de asociativitate si de element neutru mai sus specificate rezulta imediat ca inversul – daca exista – este unic si ca relatia de **izomorfie** a sistemelor de semne este reflexiva, simetrica si tranzitiva:

$$S \gg S$$

$$S_1 \gg S_2 \text{ implica } S_2 \gg S_1$$

$$S_1 \gg S_2 \text{ si } S_2 \gg S_3 \text{ implica } S_1 \gg S_3$$

Mai mult, daca notam inversa lui M cu M^{-1} atunci urmatoarele ecuatii se demonstreaza si ele cu usurinta:

$$(1_S)^{-1} = 1_S$$

$$(M^{-1})^{-1} = M$$

$$(M; M')^{-1} = M'^{-1}; M^{-1}$$

Compozitia morfismelor semiotice poate fi aplicata (si, în mod empiric a si fost aplicata) în descompunerea (factorizarea) problemelor de proiectare; astfel – în mod traditional proiectarea a fost descompusa în proiectarea logica – ce are de-a face cu aspectele conceptuale/abstracte ale proiectarii si proiectarea fizica – ce sse relateaza mult mai strâns cu caracteristicile fizice ale perifericelor utilizate, etc.

2.4 Calitatea morfismelor semiotice

Scopul fazelor de analiza si proiectare a sistemelor este de a produce reprezentari ale realitatii (problemei) si respectiv ale solutiei, însa nu a fost foarte clar cum se poate determina în mod obiectiv calitatea acestor reprezentari.

Pentru a determina ce face ca o structura sa fie mai buna decât o alta, întreaga structura a sistemelor de semne trebuie luata în considerare. În acest sens, structura conservata de morfismele semiotice furnizeaza un mod consistent de a compara calitatea acestora. Trebuie sa avem în vedere ca deoarece nu este necesar ca un morfism semiotic sa fie total, unele din semnele sistemului sursa pot sa nu aiba reprezentari în sistemul destinatie.

Peirce [PEIR65] a introdus o modalitate de clasificare a semnelor devenita clasica, în icon, index si simbol – ce furnizeaza o prima aproximare a calitatii reprezentarilor. Având o semnificatie precisa, diferita de utilizarea lor curenta, cele trei semnifica:

■ Un semn este **icon** atunci când se refera la obiectul pe care îl denota numai în virtutea

unor caracteristici proprii – ca de exemplu o dunga trasata cu creta ce reprezinta o linie geometrica.

■ Un semn x este **index** pentru obiectul y daca x si y sunt conectate în mod regulat, în sensul ca întotdeauna unde apare x , exista si y aflat într-o relatie spatio-temporala - specificabila într-o maniera mai mult sau mai putin exacta - cu x .

■ Un **simbol** este un semn constituit ca atare numai – sau în principal datorita faptului ca – este utilizat sau înteles ca atare.

Astfel, o reprezentare iconica conserva unele proprietati importante ale sistemelor semnificate; pentru un morfism semiotic, acestea pot aparea ca axiome si/sau atribute (functii luând valori în sorturile de date). O reprezentare indexata participa într-o situatie (teorie) mai ampla, în cadrul careia putem deduce legatura dintre semnele semnificat si semnificant. Pentru un simbol, o asemenea relatie „fundamentala” nu exista.

■ Din punctul de vedere al inteligibilitatii, iconurile sunt mai bune decât indicii, care sunt mai buni decât simbolurile.

■ Din punctul de vedere al operationalitatii, situatia se prezinta invers, cele mai adecvate fiind simbolurile, urmate de indici si în cele din urma de iconuri.

Ca exemplu, sa consideram un sistem de numeratie iconic, în care 2 se reprezinta ca II, iar 4 ca IIII. Prin urmare, $2 + 4 = 6$ se va scrie ca II + IIII = IIIIII. Sistemul este usor de înteles (si probabil preferat de un scolar din clasele mici) însa folosirea sa pentru numere mari este practic imposibila, problema ce nu se pune în cazul sistemului de numeratie arab.

Problemele ridicate de clasificarea lui Peirce sunt multiple, si ofera numai un ghid orientativ de clasificare a reprezentarilor. Astfel, cele trei clase se suprapun în practica si prin urmare unele semne sunt dificil de clasificat. Foarte adesea, este necesar sa luam în considerare contextul unui semn (altfel spus, modul sau de utilizare în practica) si bineînteles relatia cu alte semne din cadrul aceluasi sistem.

Urmatoarea definitie da un cadru mai precis de comparare a calitatii reprezentarilor:

Definitie: Fiind dat un morfism semiotic $M : S_1 \otimes S_2$ atunci:

(1) **M conserva nivelul** daca si numai daca relatia de ordine partiala pe nivele este conservata de M , în sensul în care daca un sort s este de nivel mai scazut decât sortul s' în S_1 , atunci $M(s)$ are un nivel mai scazut (sau egal) decât $M(s')$ în S_2

(2) **M conserva prioritatea** daca si numai daca $c < c'$ în S_1 , atunci $M(c) < M(c')$ în S_2

(3) **M conserva axioma** daca si numai daca pentru fiecare axioma a a lui S_1 , translatia sa $M(a)$ la S_2 este o consecinta logica a axiomelor din S_2

(4) Fiind dat si $M' : S_1 \otimes S_2$, atunci M' este (cel puțin) **la fel de definit ca si M** , si scriem $M \hat{=} M'$, daca si numai daca pentru fiecare constructor c al lui S_1 , $M'(c)$ este definit acolo unde este definit $M(c)$.

(5) Fiind dat si $M' : S_1 \otimes S_2$, atunci M' **conserva toate axiomele conservate de M** , si scriem $M \hat{=} M'$, daca si numai daca pentru fiecare constructor c al lui S_1 , $M'(c)$ este definit acolo unde este definit $M(c)$.

(6) Fiind dat si $M' : S_1 \otimes S_2$, atunci M' este (cel puțin) **la fel de inclusiv ca si M** daca si numai daca $M(x) = x$ implica $M'(x) = x$ pentru fiecare semn x din S_1 .

(7) Fiind dat si $M' : S_1 \otimes S_2$, atunci M' **conserva** (cel puțin) **la fel de mult continut ca si M** , scris $M \ll M'$, daca si numai daca M' este la fel de definit ca si M si M' conserva fiecare selector conservat si de M , unde spunem despre un morfism $M : S_1 \otimes S_2$ **conserva un selector** f_1 din S_1 daca si numai daca exista un selector f_2 din S_2 astfel încât pentru fiecare semn x din S_1 unde M este definit, atunci $f_2(M(x)) = f_1(x)$, unde:

(8) Un **selector** pentru un sistem de semne S este o functie $f : s \rightarrow d$, unde s este un sort de semn si d este un sort de date al lui S , astfel încât exista axiome A' astfel încât adaugarea lui f si a lui A' la S este consistenta si defineste o valoare unica $f(x)$ pentru fiecare semn x al sortului s . Spre exemplu, fiecare parametru al unui constructor are un selector corespondent pentru a-i extrage valoarea.

Intuitiv, semnificatia lui (7) este ca are loc o conservare a continutului daca exista un mod oarecare de a regasi fiecare valoare de data din sistemul sursa pe baza imaginii sale în sistemul de semne tinta.

Se poate întâmpla ca nici M si nici M' sa nu conserve strict mai mult decât celalalt; spre exemplu, M poate conserva constructorii, în timp ce M' conserva continutul. De asemenea, fiecare din aceste relatii de ordine este o relatie partiala.

Rezultate experimentale (în domeniul lingvisticii) au demonstrat ca conservarea nivelelor dintr-un sistem de semne este mai importanta decât conservarea priorităților, care la rândul ei este mai importanta decât conservarea continutului. De asemenea, exista o puternica tendinta de conservare a nivelelor ridicate pe socoteala celor scazute, atunci când trebuie sa se renunte la unele elemente de structura. Aceste observatii sunt consistente cu principiul general conform caruia structura are prioritate asupra continutului sau, cu alte cuvinte, *forma este mai importanta decât continutul*.

2.5 Amestecuri. Suport de reprezentare si practici acceptate

Primele definitii si reprezentari de amestecuri („blends”) apartin lui Fauconnier si Turner, care au studiat amestecurile spatiilor conceptuale, pentru a obtine noi spatii ce combina partile spatiilor de intrare.

Definitie: Se numeste **amestec** B al sistemelor de semne I_1 si I_2 (numite **intrari**) peste G (numit **generic**) si se reprezinta ca în figura de mai jos, folosind morfismele semiotice $G \rightarrow I_1$ si $G \rightarrow I_2$ date, un sistem de semne pentru care exista morfismele $I_1 \rightarrow B$, $I_2 \rightarrow B$ si $G \rightarrow B$ numite toate **injectii**, astfel încât diagrama sa fie **slab comutativa** (atât compozitia $G \rightarrow I_1 \rightarrow B$ cât si $G \rightarrow I_2 \rightarrow B$ sunt **slab egale** cu morfismul $G \rightarrow B$, în sensul ca fiecare semn din G este translatat prin ele catre acelasi semn, daca ambele morfisme sunt definite pentru semnul considerat).

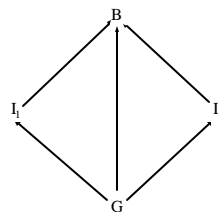


Fig.1. Diagrama de amestec

Recomandam consultarea lucrării [GOGU99] pentru mai multe detalii privind modalități alternative de reprezentare ale amestecurilor și proprietățile ce deriva din aceste modalități.

O aplicație directă a amestecurilor o constituie studiul modalităților de reprezentare (al diagramelor) din diversele metodologii de realizare a sistemelor informatice, în special în faza de analiză a sistemelor. Diagramele pot fi văzute ca fiind amestecuri ale descrierilor textuale ale problemelor și ale regulilor de compunere ale diagramelor studiate. Se observă că de fapt, din punctul de vedere al relevanței conținutului informațional amestecul preia întru totul conținutul reprezentării inițiale, în format textual, în timp ce sistemul de semne al regulilor de compunere are numai un rol de suport pentru acest conținut. Vom numi astfel de sisteme de semne **sisteme suport de reprezentare**, fără a da în acest moment o definiție mai riguroasă a conceptului introdus.

Invers, în cazul pachetelor software asistăm adesea la situația în care rezultatul amestecului dintre specificațiile de analiză și modele/procesele deja implementate (cablate) în sistem conduc la un amestec cu un conținut informațional mai bogat decât al sistemului inițial. Vom numi astfel de sisteme **sisteme de practici acceptate** (din expresia engl. „best practices”).

2.6 Proiecții semiotice

În mod intuitiv, orice morfism semiotic, prin caracterul său parțial (vezi definiția de la pagina 36) este o **proiecție semiotică** a sistemului de semne inițial după o direcție dată. În acest sens, vezi și diagrama următoare, din [OPRE99], precum și [FORR79] pentru un comentariu similar: “Sistemele cu reacție (conexiune inversă) sunt inefabile. Structura lor și implicațiile dinamice pe care le generează sunt greu de înțeles și memorat. Sunt necesare foarte multe unghiuri de vedere. Din fiecare unghi se poate vedea ceva care este omis într-o altă poziție.”

Importanța acestei observații constă în faptul că este oferită posibilitatea (cel puțin teoretică, an acest stadiu!) de a studia modalitățile de reprezentare diagramatică ca seturi de proiecții ale unui sistem de semne și de a analiza ortogonalitatea (cu alte cuvinte, posibilitatea de a

reconstrui sistemul de semne pornind de la proiecțiile sale) diagramelor unei metodologii date.

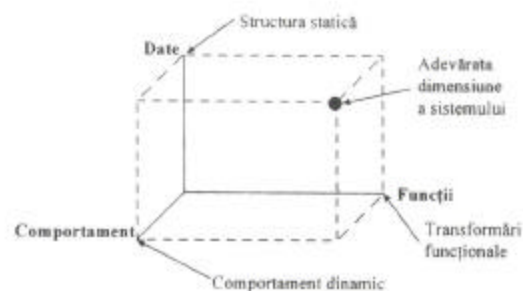


Fig2. Dimensiuni ale sistemelor informatice

3. Concluzii

Pe parcursul prezentei lucrări a fost propusă o nouă modalitate de abordare a studiului metodelor și metodologiilor de realizare a sistemelor, și anume studiul acestor metodologii prin *algebra semiotică*, metoda de inspirație structuralist-semiotică elaborată de școala de la Universitatea de la San Diego condusă de prof. Joseph Goguen. Au fost introduse o serie de concepte noi, utile pentru acest tip de studiu, între care amintim noțiunile de *proiecție semiotică*, *sistem suport de reprezentare* și *sistem practică acceptată*.

Ariile cele mai susceptibile de dezvoltări viitoare în legătură cu elemente introduse sunt:

- Identificarea de modalități de asociere concretă a elementelor metodologiilor de realizare cu elementele definitorii ale sistemelor de semne și ale morfismelor semiotice, pentru a permite extragerea unor concluzii practice cu privire la calitatea metodelor utilizate și a unor modalități sistematice de îmbunătățire a acestora
- Definirea unor algoritmi de calcul al reprezentărilor, care eventual împreună cu alte metode și tehnici (gramatici dinamice, LSA) să permită dezvoltarea de metode (semi)automate de analiză și proiectare a sistemelor informatice integrate.

Bibliografie

[BOUL94] - P. Boullier, *Dynamic grammars & semantic analysis*, Rapport de recherche No. 2322, Institut National De Recherche en Informatique et en Automatique, 1994

- [FORR79] - J.W. Forrester, *Principiile sistemelor. Teorie si autoinstruire programata*, Ed. Tehnica, Bucuresti, 1979
- [GOGU91] - J. Goguen - *A categorical manifesto*, Mathematical Structures in Computer Science, 1(1):49-67, March 1991
- [GOGU99] - J. Goguen - *An Introduction to Algebraic Semiotics, with Applications to User Interface Design*, Computation for Metaphors, Analogy and Agents, Lecture Notes in Artificial Intelligence, Volume 1562, pages 242-291, Springer, 1999
- [LAND98] - T.K. Landauer, P.W. Foltz, D. Laham, An Introduction to Latent Semantic Analysis, in Discourse Processes, 25, pg. 259-284, 1998
- [LUNG94] - I. Lungu, Gh. Sabau, T. Surcel, C. Bodea, *Sisteme informatice pentru conducere*, Ed. SIAJ, Bucuresti, 1994
- [OPRE99] - D. Oprea, *Analiza si proiectarea sistemelor informationale economice*, Ed. Polirom, Iasi, 1999
- [PEIR65] - C. S. Peirce. *Collected Papers of Charles Saunders Peirce*, Harvard, 1965
- [SERB87] - L. D. Serbanati, *Limbaje de programare si compilatoare*, Editura Academiei Republicii Socialiste România, Bucuresti, 1987