

The Orthogonality of the Object-Oriented Software Applications

Prof.dr. Ion IVAN, prep. Marius POPA
Catedra de Informatică Economică, A.S.E. București

The paper aims to present some aspects regarding the object-oriented orthogonality metrics. There are presented the basic elements of the object-oriented development environment, the object-oriented metrics of the software products and software cohesion. Also, there are provided some orthogonality metrics in order to evaluate the similarity degree of this kind of applications.

Keywords: *orthogonality, metrics, object-oriented software.*

Ortogonalitatea produselor program Analiza și proiectarea orientate-obiect a produselor program sunt elemente foarte utilizate în dezvoltarea de software. Această modalitate de realizare de software aduce un plus de calitate în ceea ce privește mentenanța și modificarea aplicațiilor informatice dezvoltate.

Metodologia orientată-obiect de dezvoltare de produse program presupune o abordare diferită de metodele tradiționale a descompunerii funcționale și fluxurilor de date a sistemului analizat. Tradițional, sistemul este analizat prin considerarea separată a datelor de comportament. Analiza și proiectarea orientate-obiect combină cele două elemente, fiecare clasă de date și operațiile aferente fiind grupate într-o singură entitate.

Concepte utilizate în analiza și proiectarea orientate-obiect:

- *clasă de obiecte* – o mulțime de obiecte care împart o structură și un comportament comune; este un șablon pe baza căruia se instanțiază obiecte;
- *moștenire* – o relație între clase prin care o clasă preia atributele și metodele definite în una sau mai multe clase;
- *instanțiere* – proces de creare a unui obiect și inițializarea sa cu date specifice;
- *atribut* – variabilă definită în cadrul unei clase de obiecte; mulțimea atributelor reflectă starea unui obiect;
- *metodă* – operează asupra datelor ca răspuns a mesajelor primite, fiind parte a declarației clasei de obiecte; mulțimea metodelor precizează comportamentul unui obiect;
- *obiect* – o instanțiere a unei clase capabilă

să salveze o stare și care oferă o serie de operații care examinează starea și/sau o modifică.

Ortogonalitatea surprinde diferențele dintre două elemente. Datele sunt ortogonale dacă ele sunt complet diferite. Se definește un indicator de ortogonalitate care, în caz de ortogonalitate ia valoarea 1, iar în caz de identitate ia valoarea 0. Astfel, se remarcă existența unui spectru al gradelor de ortogonalitate dintre două elemente, cu valori în intervalul [0,1].

Forma pe care o iau indicatorii de ortogonalitate este strâns legată de tipul datelor comparate, de complexitatea acestora, precum și de specificul analizei. În lucrarea de față se va considera ortogonalitatea la nivelul aplicațiilor orientate-obiect.

Cuvintele reprezintă unitatea constructivă a programelor sursă, fiind identificate două categorii: cuvinte rezervate ale limbajului; cuvinte utilizator. Orice analiză trebuie să înceapă de la elementele de construcție a programelor sursă.

Metrici de ortogonalitate orientate-obiect

Analiza ortogonalității produselor program orientate-obiect se efectuează pe baza sistemului de metrici orientate-obiect prin intermediul cărora sunt cuantificate elementele specifice programării orientate-obiect. Într-o aplicație informatică orientată-obiect metricele tradiționale sunt aplicate pentru metodele care conțin operațiile clasei de obiecte. În categoria metricilor tradiționale se includ:

- complexitatea ciclomatică – utilizată în evaluarea aplicației unui algoritm;

- numărul de linii – include linii de cod, expresii și comentarii; prezintă importanță în înțelegerea, reutilizarea și mentenanța codului aplicației;

- ponderea liniilor de comentariu – utilizată pentru evaluarea caracteristicilor de înțelegere, reutilizare și mentenanță a aplicației.

Există un număr foarte mare de metrici specifice abordării orientate-obiect a aplicațiilor informatice prin intermediul cărora se cuantifică elemente caracteristice claselor de obiecte, moștenirii, metodelor, obiectelor, polimorfismului. În categoria metricilor orientate-obiect se includ:

- metode per clasă de obiecte – se exprimă prin numărul de metode implementate în cadrul unei clase de obiecte sau ca sumă a complexităților metodelor; este o metodă de predicție a timpului și efortului necesar pentru dezvoltarea și mentenanța clasei de obiecte;

- răspunsul la mesaje – numărul metodelor care pot fi invocate ca răspuns la un mesaj transmis de un obiect sau o altă metodă din clasă; o valoare mare a acestui indicator determină un efort sporit în testarea și depanarea clasei;

- coeziunea – reflectă măsura în care metodele dintr-o clasă sunt legate una de cealaltă în vederea asigurării unui comportament corespunzător al obiectului;

- asocierea – reflectă intensitatea asocierii stabilită de conexiune de la o entitate la alta;

se exprimă prin numărul de clase asociate cu clasa de obiecte pentru care se cuantifică acest indicator;

- înălțimea arborelui de moștenire – se determină ca maximum dintre lungimile drumurilor de la noduri la rădăcină în cadrul arborelui de moștenire și se exprimă ca număr de noduri traversate de drumul de lungime maximă; este utilă în evaluarea caracteristicilor de reutilizare, înțelegere și testare a codului sursă;

- numărul descendenților – reprezintă numărul subclaselor imediate unei anumite clase din ierarhia de moștenire; permite evaluarea testabilității și proiectării produsului program.

Pentru aplicația *Liste_simple* dezvoltată în limbajul de programare Java, valorile metricilor prezentate mai sus sunt date în tabelul 1. Aplicația *Liste_simple* implementează operații de bază cu care se operează în cazul structurii de date lista simplă înălțuită:

- inserarea unui element în listă;

- verificarea existenței unui element în listă prin testarea valorilor informațiilor utile din noduri;

- afișarea pe ecran a valorilor din noduri în ordinea inserării lor;

- afișarea pe ecran a informațiilor utile din noduri în ordinea inversă a inserării lor în listă;

- determinare lungime listă.

Tabelul 1. Metrici orientate-obiect pentru aplicația *Liste_simple*

Clasa de obiecte	Nr. de metode	Răspunsuri la mesaje	Nr. clase asociate	Nr. de descendenți
Nod	6	0	0	0
ListaSimpla	9	11	2	1
ListaOrdonata	11	12	1	0
Test1	1	29	1	0
Test2	1	12	1	0

Codurile sursă a două produse software se compară raportând valorile metricilor orientate-obiect. De asemenea, anterior efectuării acestor calcule este necesar să asigure comparabilitatea celor două programe din punctul de vedere a limbajelor de programare utilizate în implementarea lor.

Pentru două texte sursă P_i și P_j , forma generală a indicatorului de ortogonalitate pentru

metrica m este: $O_{i,j} = \frac{\min(I_i^m, I_j^m)}{\max(I_i^m, I_j^m)}$, unde: I_i^m -

valoarea metricii m pentru programul sursă P_i ; I_j^m - valoarea metricii m pentru programul sursă P_j .

Pentru două programe sursă P_i și P_j se construiește matricea M_O care conține valorile indicatorilor de ortogonalitate pentru cele n metrici luate în considerare în analiza orto-

gonalității. Matricea M_O este simetrică în raport cu diagonala principală și are următoarea formă:

$$M_O = \begin{bmatrix} O_{1,1} & O_{1,2} & O_{1,3} & \dots & O_{1,n-1} & O_{1,n} \\ O_{2,1} & O_{2,2} & O_{2,3} & \dots & O_{2,n-1} & O_{2,n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ O_{n-1,1} & O_{n-1,2} & O_{n-1,3} & \dots & O_{n-1,n-1} & O_{n-1,n} \\ O_{n,1} & O_{n,2} & O_{n,3} & \dots & O_{n,n-1} & O_{n,n} \end{bmatrix}$$

Pentru o mulțime de programe sursă scrise într-un limbaj de programare orientat-obiect se determină un indicator agregat de ortogonalitate pe baza valorilor din matricea M_{OM} :

$$M_{OM} = \begin{bmatrix} OS_{1,1} & OS_{1,2} & OS_{1,3} & \dots & OS_{1,ns-1} & OS_{1,ns} \\ OS_{2,1} & OS_{2,2} & OS_{2,3} & \dots & OS_{2,ns-1} & OS_{2,ns} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ OS_{ns-1,1} & OS_{ns-1,2} & OS_{ns-1,3} & \dots & OS_{ns-1,ns-1} & OS_{ns-1,ns} \\ OS_{ns,1} & OS_{ns,2} & OS_{ns,3} & \dots & OS_{ns,ns-1} & OS_{ns,ns} \end{bmatrix}$$

unde: ns – numărul de texte sursă supuse analizei comparative; $OS_{i,j}$ – indicator agregat al ortogonalității două programe sursă scrise într-un limbaj de programare orientat-obiect, cu următoarea expresie analitică:

$$OS_{i,j} = \frac{n(n-1)}{2} \sqrt{\prod_{\substack{i=1,n \\ j=i+1,n}} O_{i,j}}$$

Pe baza valorilor cuprinse în matricea M_{OM} se desprind concluzii cu privire la modul de dezvoltare a produselor program orientate-obiect, precum și caracteristici de calitate a acestora.

Coeziunea software

Coeziunea reprezintă măsura în care elementele unei colectivități sunt legate unele de celelalte. Cu cât coeziunea are un nivel mai ridicat, cu atât este mai ușor de dezvoltat, întreținut și refolosit modulul respectiv, fiind mai puțin predispus la erori.

În contextul programării procedurale, coeziunea modulului este o caracteristică de calitate software. Problema care se pune este de a conferi unui modul un nivel ridicat de calitate care să ofere performanțe ridicate pentru sistemul proiectat. Pentru asigurarea unui nivel ridicat a produselor program au fost introduse metrice de coeziune, realizarea unui

software de calitate înaltă constituind obiectivul care se dorește a fi atins.

În contextul orientat obiect, coeziunea urmărește măsurarea intensității legăturilor dintre elementele unei clase. Pentru o anumită clasă de obiecte un nivel ridicat de coeziune se asigură atunci când metodele definite folosesc seturi similare de instanțe a variabilelor. Conform analizelor realizate pe module cu diferite nivele de coeziune, modulele cu nivele de coeziune mai puternice au mai puține defecte raportate decât modulele cu nivele de coeziune mai slabe.

Cercetările în domeniu au condus la identificarea a opt categorii generale ale coeziunii [NICU03]: funcțională, informațională, secvențială, comunicațională, procedurală, temporală, logică și coincidentală.

Structura unui program este evidențiată în termeni de module, funcții, subrutine, metode sau proceduri. Stabilirea coeziunii acestor structuri impune analiza părților componente. Aceste elemente reprezintă una sau mai multe linii de cod care dau obiectivele modului. Legăturile dintre aceste componente determină nivelele de coeziune ale modulelor. În [IVAN03] sunt definiți indicatori de evaluare a coeziunii pentru sistemele software orientate obiect.

Coeziunea clasei de obiecte, pornind de la atributele definite în aceasta, se determină conform expresiei analitice a indicatorului C_a

$$[IVAN03b]: C_a = \frac{\sum_{i=1}^n na_i}{n \cdot na_c}$$

unde: na_i – numărul de atribute referite în cadrul metodei i ; n – numărul de metode definite în cadrul clasei; na_c – numărul de atribute ale clasei de obiecte.

Conform acestui indicator, coeziunea este maximă atunci când toate metodele clasei utilizează toate atributele definite.

În cazul aplicației *Liste_simple* valorile acestui indicator sunt date în tabelul următor:

Tabelul 2. Valorile indicatorului de coeziune în aplicația *Liste simple*

	Nod	ListaSimpla	ListaOrdonata	Test1	Test2
C_a	0,66	0,88	0,81	0	0

Gradul de conectivitate al metodelor clasei se determină conform indicatorului LCC, având următoarea formă analitică [IVAN03a]: $LCC = \frac{NIC}{NP}$, unde: NIC – numărul de conexiuni, directe sau indirecte, stabilite între metodele clasei de obiecte; NP – numărul maxim de perechi de metode publice ale clasei

de obiecte, care se determină conform relației: $NP = \frac{N \cdot (N-1)}{2}$, unde N reprezintă numărul de metode publice ale clasei de obiecte. Pentru aplicația *Liste_simple* valorile indicatorului LCC sunt date în tabelul:

Tabelul 3. Valoarea indicatorului de conectivitate a metodelor claselor pentru aplicație *Liste simple*

	Nod	ListaSimpla	ListaOrdonata	Test1	Test2
LCC	0	0,05	0,07	0	0

Ponderea perechilor de metode publice ale clasei de obiecte, care folosesc aceleași atribute, în numărul maxim de perechi de metode este: $TCC = \frac{NDC}{NP}$, unde: NDC – numărul

de conexiuni directe între metodele clasei de obiecte; NP – numărul maxim de perechi de metode publice ale clasei de obiecte.

În cazul aplicației *Liste_simple* valorile indicatorului TCC sunt identice cu cele ale indicatorului LCC deoarece există numai legături directe între metodele claselor de obiecte construite în cadrul aplicației. În practică, utilizarea indicatorilor prezentați în paragrafele anterioare este dificilă deoarece nu au la bază modele empirice explicite.

În ceea ce privește ortogonalitatea coeziunii, se aplică aceleași etape prezentate în capitolul anterior.

Concluzii

Determinarea ortogonalității aplicațiilor informatice orientate-obiect oferă informații cu privire la procesul de dezvoltare a acestora: reutilizare de componente în realizarea de noi aplicații sau versiuni ale unor produse existente pe piață, evaluarea nivelului de calitate a noului software etc. Aplicarea metricilor și efectuarea analizei de ortogonalitate permit asigurarea și evaluarea unor caracteristici de calitate în raport cu celelalte produse de pe piață privind: eficiența implementării proiectului produsului program; complexitatea; lizibilitatea/utilitatea programului sursă; reutilizabilitatea/aplicabilitatea aplicației; testabilitatea/mentenanța aplicației. Viitoarele direcții de cercetare vizează stabilirea categoriilor de ortogonalitate pentru produse

orientate-obiect, având în vedere elementele caracteristice analizei și proiectării de software orientat-obiect.

Bibliografie

- [CHID94] Shyam CHIDAMBER, Chris KEMERER – A Metrics Suite for Object-Oriented Design, IEEE Transactions on Software Engineering, June, 1994, pg. 476 – 492
- [IVAN04a] Ion IVAN, Marius POPA, Roland DRĂGOI – *Validarea ortogonalității firmelor și emblemelor agenților economici*, revista Studii și Cercetări de Calcul Economic și Cibernetică Economică, 2004, în curs de apariție
- [IVAN04b] Ion IVAN, Marius POPA, Cristian TOMA, Iulian RĂDULESCU – *The Aggregation of the Data Orthogonality Metrics*, Proceedings of „The 35th International Scientific Symposium of METRA”, vol. 1, București, 27 – 28 mai 2004, pg. 590 – 595
- [IVAN03a] Ion IVAN, Constantin NICULAE, Paul POCATILU – *Metrici pentru evaluarea coeziunii software orientat obiect*, revista Studii și Cercetări de Calcul Economic și Cibernetică Economică, București, 2003
- [IVAN03b] Ion IVAN, Zsolt MARK, Paul POCATILU, Marius POPA, Doru UNGUREANU – *Cohesion – Software Quality Characteristic*, The 30th Session of Scientific Presentations „Modern Technologies in the XXI Century”, Academia Tehnică Militară, 6 – 7 noiembrie 2003, în format electronic pe suport CD, ISBN 973-640-012-3
- [NICU03] Constantin NICULAE – *Coeziunea – caracteristică de calitate software*, Lucrare de licență, ASE, București, 2003