

Cadru conceptual privind coeziunea produselor program

Prof.dr. Ion IVAN, ec. Nicolae CONSTANTIN
Catedra de Informatica Economic a, A.S.E. Bucuresti

Software quality is an important topic in software engineering. One of the characteristics of the software quality is the software cohesion. In this paper is defined the software cohesion and its main properties. Indicators for the software cohesion are presented for classical and for object-oriented software. At the end is described a software for cohesion evaluation.

Keywords: software quality, quality characteristics, software cohesion, complexity.

Coeziunea, atribut al calitatii software

Capacitatea de a dezvolta proceduri destinate măsurării atributelor și proceselor specifice producției de software este esențială atunci când se urmărește obținerea unor produse program de înaltă calitate. Metricile software cuantifică nivelul de calitate al produselor software, iar procesele de realizare a acestora determină utilizarea unor tehnici și instrumente de analiză, proiectare, deciziile privind restructurări, alocări și nivelări de resurse presupunând efectuarea de măsurători asupra modulelor, bazelor de date. Sistemul de indicatori utilizat frecvent este caracterizat prin număr redus de variabile exogene, iar în cazul în care complexitatea formei analitice este foarte mare, se lucrează cu date incomplete a căror culegere se încadrează în limite de costuri acceptabile. Concentrarea eforturilor asupra realizării de metrici software este orientată spre identificarea și măsurarea factorilor de influență semnificativi care acționează asupra calitatii software privita sub toate aspectele. În acest context, o parte importantă a eforturilor este concentrată în direcția construirii modelelor de evaluare a coeziunii atât în contextul procedural cât și în cel obiectual. Pentru acest atribut al produselor software este necesară înregistrarea de serii de date pe termen lung și efectuarea de estimări ale coeficienților modelelor care sunt asociați atributului de coeziune. Se studiază distinct stabilitatea modelelor în vederea utilizării curente a lor, respectând cerințele incluse în standardele de calitate pentru program. Tehnologiile orientate obiect au impus modele ale coeziunii care reflectă particulari-

tatile pe care le determină încapsularea, menținerea, și polimorfismul.

Productia industrială de software și piața produselor informatice în contextul societății informaționale pune un accent deosebit pe realizarea de componente având un nivel de calitate deosebit de ridicat, nivel evidențiat prin testare sistematică, urmată de cuantificarea caracteristicilor și atributelor de calitate software. Aceasta a condus la creșterea volumului de muncă pentru culegerea de date și pentru implementarea metricilor software, obținându-se pentru toate produsele valori ale indicatorilor de calitate, așa fel încât să se asigure comparabilitatea produselor și analiza raportului calitate-pret. În același timp, activitatea de cercetare în domeniul calitatii software, a condus la apariția unui număr mare de indicatori, dintre care numai o parte au fost propuși pentru includerea în standardele de calitate, cu caracter orientativ. Dintre acestea, un număr important de modele se referă la atributul de coeziune. Un produs software de calitate înaltă, trebuie să fie înzestrat cu fiabilitate, utilizabilitate, mentenabilitate și portabilitate la niveluri deosebit de ridicate și să aibă, de asemenea, un nivel de coeziune la nivelul componentelor, foarte mare.

În [STEVM74], este definit atributul de coeziune în contextul dezvoltării tehnicilor structurate, ca măsură a gradului în care instrucțiunile care alcatuiesc un modul sunt legate unele de celelalte, conform logicii de derulare a pașilor algoritmului implementat. Într-un modul caracterizat printr-un nivel înalt al coeziunii, toate instrucțiunile sunt strâns legate între ele în vederea realizării

scopului pentru care modulul a fost proiectat. Cu cât modulul are coeziunea mai ridicată, cu atât este mai ușor de dezvoltat, întreținut și refolosit și este mai puțin predispus la erori. Aceeași problemă se pune și în cazul coeziunii la nivelul bazelor de date. Unele rezultate empirice stau la baza conceptului de coeziune pentru sistemele dezvoltate prin tehnici structurate și orientate obiect, [CARDA86], [CARDM85], [BRIMB94].

Coeziunea se referă la legăturile componentelor unui modul și este cunoscută în principal în contextul procedural. În contextul orientat obiect, un concept de coeziune a clasei a apărut ca fiind necesar. Coeziunea este unul dintre cei mai importanți factori de calitate software, alături de mentenabilitate, fiabilitate și reutilizabilitate. Coeziunea modulului este definită ca un atribut de calitate, care caută să masoare rezultatele prelucrărilor care au loc în modul. Un modul de calitate slabă este un obstacol serios pentru calitatea sistemului. Pentru a dezvolta software de calitate foarte ridicată, managerii de proiecte informatice și programatorii trebuie să introducă și să evalueze componentele realizate folosind metrici de coeziune, iar corecțiile care se impun conduc la produse program cu nivelul de calitate global dorit. Realizarea unui produs software de calitate înaltă este un obiectiv pe care dorește să-l realizeze orice echipă care lucrează într-o companie de software certificată, cel puțin ISO 9000. Coeziunea măsoară puterea legăturilor dintre elementele unei clase. Când metodele din interiorul clasei folosesc seturi similare de instanțe, se consideră ca nivelul de coeziune este foarte înalt.

Coeziunea pentru produse program neorientate obiect

Arhitectura Software - Coeziune. Legătura dintre tipul și gradul în care sunt realizate obiectivele definite de către un singur modul software poartă denumirea de *puterea modulului*. Aceasta este una din caracteristicile fundamentale ale oricărei arhitecturi software, descriind nivelul interdependentelor dintre elementele care intră în componența unui modul. Coeziunea măsoară puterea funcțio-

nala relativă a unui modul. A fost definită la începutul anilor 70, pentru a determina cauzele care generează costuri importante în procesul de mentenanță a sistemelor de programe complexe.

S-a ajuns la concluzia că atunci când la nivelul unui modul, o componentă este frecvent modificată, se produce un efect de antrenare multiplă concretizat prin modificarea a numeroase componente din respectivul produs program. Efectul de antrenare multiplă funcționează ca efect cascada și este legat de modul în care programatorii construiesc secvențele de preluare parametri și de returnare a rezultatelor în procedurile ce alcătuiesc module. Cu cât programul apelator preia din sarcinile programului apelat, cu atât mai mult programul apelator preia, de asemenea, cerințe de mentenanță specifice subprogramului apelat. Relațiile dintre componentele înzestrate cu grad înalt de coeziune reduc incidențele atunci când acestea sunt folosite în alte locuri din interiorul sistemului.

Sunt identificate mai multe categorii de modele pentru evaluarea coeziunii, bazate pe influențele diferiților factori care influențează acest atribut (figura 1).

Analizele statistice efectuate pe seturi de module cu niveluri diferite ale coeziunii, în legătura cu comportamentul pe durata utilizării, concretizat prin volum de erori, arată că modulele cu niveluri de coeziune mai ridicate au mai puține erori în raport cu modulele care au coeziune mai scăzută și care în utilizarea curentă înregistrează un număr de erori semnificativ.

Categoriile de coeziune. Când se analizează structura unui produs software, analiștii și programatorii utilizează module, funcții, subrutine, metode sau proceduri. Toate aceste componente care intră în structura produsului software sunt diferite, dar intră în componența aceluiași categorii de structuri. Puterea internă, termen echivalent celui de coeziune, a unei structuri software, impune efectuarea analizei părților componente. Componentele reprezintă una sau mai multe linii de cod care definesc obiectivele modulului. Legăturile dintre instrucțiunile care intră în componența

liniilor de cod determina stabilirea nivelurilor de coeziune pentru fiecare modul în parte.

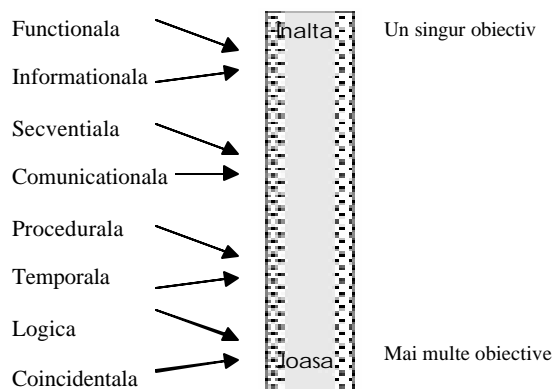


Fig. 1. Nivelurile de coeziune software

În figura 1 se identifica o serie de intervale care stau la baza conceptelor fundamentale ale unei arhitecturi software. Se identifica tehnologii de dezvoltare software și se stabilește efectul generat de fiecare metoda de proiectare a aplicației asupra raportului cost-performanță. Odată cu definirea coeziunii, s-a observat că fiecare dintre tehnicile și metodele de analiză/proiectare/programare/testare, generează atât module cu înalt grad de coeziune, cât și module cu grad scăzut de coeziune, acestea din urmă, fiind purtătoare ale unui volum foarte ridicat de erori. Pentru ameliorarea nivelului costurilor se impune realizarea unui compromis între modulele cu înalt grad de coeziune și celelalte module.

Coeziunea pentru produse software orientate obiect

Scopul obținerii unui nivel deosebit de înalt al coeziunii este realizabil în condițiile dezvoltării de software orientat obiect. Se definesc indicatori de coeziune pentru sistemele software dezvoltate cu tehnologii orientate obiect. Deși coeziunea este un atribut al sistemelor de programe complexe dezvoltate cu tehnologii orientate obiect, au fost construite modele și pentru celelalte tipologii de produse software. Numeroasele modele elaborate impun efectuarea unei analize comparate și definirea unor clasificări a modelelor de măsurare a coeziunii, chiar dacă efortul este deosebit de mare.

Se impune elaborarea unei terminologii și a unui formalism standard pentru exprimarea măsurătorilor în mod unitar, pentru eliminarea oricărei ambiguități. Trebuie stabilit modul în care diferite metrici ale coeziunii sunt legate între ele. Frecvența de utilizare a metricilor existente impune studierea sistematică a eficacității modificărilor din module și din clase, atunci când se urmărește creșterea coeziunii. Numeroși indicatori de evaluare a coeziunii pentru produse program orientate obiect sunt în faza experimentală și necesită noi evaluări pentru cât mai multe produse în vederea fundamentării concluziilor.

Determinarea gradului de coeziune în sistemele software orientate obiect presupune existența unei terminologii și a unui formalism standard pentru:

- facilitarea comparației între indicatorii de coeziune;
- evaluarea și validarea indicatorilor de coeziune definiți;
- sprijinirea definirii de noi metrici de coeziune și selectarea din cele existente pe baza unor obiective particulare ale măsurătorilor.

Există aspecte negative privind modul de dezvoltare a metricilor de coeziune pentru software orientat obiect. Definirea unui indicator de coeziune impune decizii referitoare la numărul factorilor de influență, volumul măsurătorilor pentru asigurarea stabilității coeficienților estimați și asigurarea gradului de acoperire legat de structurile de control incluse în proceduri, module și clase. Acestea

trebuie luate tinând cont de obiectivul indicatorului chiar atunci când se definește un model empiric al coeziunii, bazat pe ipoteze. Multe dintre metricile care au fost propuse în literatura de specialitate sunt parțial fundamentate teoretic, ceea ce îngreunează utilizarea eficientă a indicatorilor.

Nivelul de utilizare a indicatorilor de coeziune existenți și modul în care aceștia sunt folosiți într-un mod complementar pentru a obține o imagine detaliată a coeziunii claselor unui sistem de programe orientate obiect sunt parțial precizate.

În [EDEKS94] se definește un cadru complet cu scopul de a furniza criterii calitative pentru evaluarea coeziunii și se asociază un indicator de putere pentru nivelurile de coeziune precizate.

Metricile de coeziune orientate obiect presupun includerea abordărilor existente într-un cadru teoretic unic, bazat pe un formalism omogen și cuprinzător. Analiza trebuie făcută cu luarea în considerare a tuturor indicatorilor existenți în concordanță cu cadrul teoretic unificator. Acesta este un mecanism cu ajutorul căruia:

- se compară indicatorii de coeziune și utilitatea lor;
- se combină indicatorii existenți care se referă la aceleași concepte, în moduri diferite dar echivalente;
- se obține rigurozitate mai mare și ușurința a deciziei luate cu privire la definirea noilor indicatori și selectarea celor mai reprezentativi.

Această ușurează evaluarea și validarea metricilor de coeziune prin asigurarea ipotezelor specifice, iar acestea sunt asigurate de indicatorii de coeziune care au legătură cu unele atribute externe. De asemenea, acești indicatori trebuie să faciliteze identificarea factorilor care influențează coeziunea dar care au fost neglijati.

Analiza rezultatelor privind analize empirice ale coeziunii.

În [HENRL93], indicatorii prezentați de Chidamber și Kemerer [CHIDK94], au fost testați, inclusiv indicatorul LCOM1. Li și Henry au propus câteva modele de regresie multivariată și folosirea regresiei celor mai

mici pătrate și teste-F pentru estimarea puterii lor de predicție.

În [BRIBM96], indicatorii prezentați de Chidamber și Kemerer [CHIDK94], au fost testați, inclusiv LCOM2. Regresia favorabilă univariată a fost realizată pentru testarea puterii de predicție pentru fiecare indicator izolat. Apoi, câteva modele multivariante au fost construite printr-un proces de selecție pas cu pas și testate. LCOM2 este singurul indicator care nu a fost găsit semnificativ în ambele analize univariate și multivariate.

Briand în [BRIMB94] a testat indicatorul RCI (mai precis, o versiune pentru sistemele bazate pe obiecte, care de asemenea ia în considerare modulele grupate) și câțiva indicatori pentru interdependență. Pentru fiecare sistem testat, regresia favorabilă univariată a fost executată pentru testarea puterii de predicție pentru fiecare indicator izolat. După aceea, un model multivariat a fost construit pentru fiecare sistem printr-un proces de selecție pas cu pas și testat.

În [CHIDK94], indicatorii propuși au fost aplicați asupra a două sisteme. Se prezintă distribuția valorilor indicatorilor și este realizată o interpretare ad-hoc asupra datelor, constând în explicații asupra diferențelor distribuției valorilor măsurărilor realizate între cele două sisteme. Nu se pot trage concluzii despre utilitatea acestor indicatori din această abordare.

Bieman și Kang [BIEMK95] au analizat influența coeziunii măsurate de TCC și LCC pe tratări particulare, vizând mostenirea și instanțierea. Pentru tratarea mostenirii unei clase c , s-a definit numărul de clase derivate din clasa c . Pentru tratarea instanțierii unei clase c , s-a definit numărul de clase din care se fac instanțieri ale clasei c (de exemplu prin agregare, sau dacă un obiect de clasa c este creat într-o metodă a altei clase). Autorii nu au găsit nici o relație între coeziune și tratările particulare vizând instanțierea. Clasele cu un număr mare de descendenți tind să aibă un nivel slab al coeziunii. Aceasta este o contradicție la ceea ce autorii se așteptau să obțină. Ei nu au furnizat nici o explicație pentru această tendință. Utilitatea acestui studiu este limitată, din moment ce schimbările depen-

dente si tratarile particulare sunt indicatori ai atributelor interne. Atributele interne nu reprezinta o calitate externa, vizibila a sistemului. Acestea nu au vreun sens al mostenirii sau o utilitate, numai daca ele sunt luate în relatii cu alte atribute externe.

În concluzie, în [BRIMB96], LCOM2 nu a fost gasit ca fiind un predicator semnificativ al erorii claselor. Din descrierea studiilor din [HENRL93] nu se poate spune daca LCOM1 este un indicator semnificativ pentru numar de linii modificate în cadrul clasei. Singura validare reusita a fost în cazul indicatorului RCI [BRIMB94], ca fiind un predicator al erorii într-un mediu de programare bazat pe obiecte. Analizele empirice realizate sunt resemnificative, trebuind realizate cu mult mai multe.

Definirea atributului extern pentru a acoperi unor probleme limiteaza utilitatea analizei. De exemplu, în [HENRL93] efortul de întreținere a fost definit ca fiind numarul de linii modificate în interiorul clasei. Un avantaj este faptul ca numarul de linii modificate în interiorul clase pot fi calculate daca exista câteva versiuni ale aceluasi sistem, nefiind nevoie de date de efort reale. Numarul de linii modificate nu este un indicator reprezentativ al efortului real necesar întreținerii clasei.

Software pentru evaluarea coeziunii modulelor care alcatuiesc aplicatii informatice

Aplicatia pentru evaluarea coeziunii modulelor are la baza aspectele teoretice si empirice în diferite situatii de analiza. Scopul urmarit este facilitarea analizei de coeziune on-line a aplicatiilor software. Este finalizata într-un site publicat pe Internet si are facilitatile:

- *logarea utilizatorilor*, cu cont si parola pentru securitate si confidentialitate; în cazul unui nou utilizator se procedeaza la înregistrarea acestuia;

- *accesarea datelor personale* dintr-o baza de date SQL; analiza aplicatiilor informatice proprii transmise anterior pe server cu evaluarea unor indicatori de coeziune.

În continuare sunt descrise principalele module din care se compune aplicatia:

- Modulul de **utilizator nou** are rolul de introducere a informatiilor necesare pentru descrierea unui utilizator nou, cuprinzând datele personale, adresa, numere de telefon, adresa de e-mail si optiuni de setare a contului. Pentru fiecare utilizator se va atribui un director de lucru pe server, cu toate drepturile de creare, scriere, modificare, stergere. Codul de identificare a utilizatorului este unic. Fiecare utilizator va avea o parola. Sunt definite proceduri care asigura unicitatea în conditii de siguranta a accesului la resursele aplicatiei pentru fiecare utilizator.

- Modulul de **logare utilizator** verifica autenticitatea celui care încearca sa se conecteze, pe baza parolei Codul de identificare a utilizatorului si parola sunt verificate folosind informatii din baza de date a utilizatorilor. În caz de neconcordanță este afisat un mesaj. În cazul în care utilizatorul este validat se trece la etapa urmatoare, corespunzătoare utilizării contului.

- Modulul **utilizare cont** prezinta mai multe facilitati pentru administrarea de către fiecare utilizator a contului propriu si a directorului de lucru de pe server. Se dispune de mai multe capacitati de lucru asupra directorului, obținându-se vizualizarea subdirectoarelor si fisierelor, eventual stergerea unora dintre fisiere. Facilitatea cea mai importanta corespunde încărcării pe server a fisierelor utilizator. Este realizata de modulul de **copiere pe server**. Utilizatorul are capacitatea de a selecta fisiere sursa, si de a realiza analiza coeziunii cu ajutorul metricilor software adecvate.

- Modulul de **copiere pe server** permite utilizatorului copierea pe server a fisierelor sursa, pentru a fi supuse analizei.

- Modulul de **evaluare si analiza a coeziunii** utilizeaza fisiere sursa la intrare. După traversarea fisierelor sursa, datele obtinute se constituie în intrari ale procedurilor definite pentru indicatorii de coeziune. Se evalueaza nivelurile de coeziune pentru clase, module, programe, se afiseaza valori tabelare si grafice în vederea interpretării de către managerii de calitatea software.

Concluzii

În aceasta prezentare se ofera un cadru pe ntru comparatii, evaluari si definire de indicatori de coeziune în sisteme informatice, cu precadere în cele orientate obiect. Scopul este de a integra si de a dezvolta noi idei privind metricile existente în literatura de specialitate. Se impune ca indicatorii de coeziune sa fie definiti într-un mod amanuntit si operational, iar indicatorii existenti sa fie selectati pe baza unui criteriu explicit. Din prezentarea conceptului de coeziune în sistemele orientate obiect s-au desprins urmatoarele aspecte:

- exista un continut foarte bogat de idei în ceea ce priveste modul de abordare a indicatorilor de coeziune în sistemele orientate obiect;
- multi indicatori nu au la baza modele empirice explicite si, deci, folosirea lor în practica este greu de determinat;
- un numar redus de indicatori au la baza validari empirice puternice; validările empirice sunt puternic influentate de validitatea rezultatelor;
- indicatorii existenti sunt clasificati conform optiunilor disponibile pentru fiecare criteriu al cadrului existent; aceasta permite compararea indicatorilor;
- nu se garanteaza utilizabilitatea indicatorilor construiti numai pe baza caracteristicilor interne ci si considerarea caracteristicilor externe.

Un motiv pentru existenta unor asa de putine analize pentru validari empirice este acela ca pentru indicatori trebuie estimati coeficientii locali sau factorii de influenta sunt partial definiti, ceea ce limiteaza aria de valabilitate a acestora. Pentru validările empirice trebuie definite si masurate unele atribute, iar analiza coeziunii prin nivelul de dificultate necesita eforturi suplimentare. Analiza unui atribut intern nu ridica obstacole, iar aplicarea în practica este numai o problema de culegere de date dintr-un sistem de programe. Exista instrumente disponibile care efectueaza analiza pe text sursa si evalueaza indicatori de coeziune. Masurarea atributelor de calitate externe este dificila întrucât se bazeaza pe seturi suplimentare de informatii privind si compo-

tamentul în utilizarea curenta. Sunt putine sistemele de programe pentru care sunt disponibile informatiile necesare evaluarii indicatorilor, deoarece producatorii trec pe un plan secundar utilizarea de instrumente necesare evaluarilor în timpul ciclului de dezvoltare software.

Se impune dezvoltarea abordarilor practice pentru cuantificarea coeziunii, în special în cazul sistemelor software orientate obiect. De asemenea se impune introducerea în standarde a unor indicatori care sa permita evaluarea coeziunii fara eforturi suplimentare.

Bibliografie

- [ARHIR00] Arhire Romulus, Evaluarea Complexitatii Sistemelor de Programe, Teza de doctorat, ASE, Bucuresti 2000
- [IVANT01] Dr.L. Teodorescu, Dr.Ivan, *Managementul Calitatii Software*, Editura Info-rec, Bucuresti 2001
- [BIEMK95] J.M. Bieman, B.-K. Kang, "Cohesion and Reuse in an Object-Oriented System", in *Proc. ACM Symp., Software Reusability (SSR'94)*, 259-262, 1995.
- [BRIMB93] L. Briand, S. Morasca, V. Basili, "Measuring and Assessing Maintainability at the End of High-Level Design", *IEEE Conference on Software Maintenance*, Montreal, Canada, September 1993
- [BRIMB94] L. Briand, S. Morasca, V. Basili, "Defining and Validating High-Level Design Metrics", *Technical Report, University of Maryland, CS-TR 3301*, 1994.
- [BRIMB96] L. Briand, S. Morasca, V. Basili, "Property-Based Software Engineering Measurement", *IEEE Transactions of Software Engineering* 22 (1), 68-86, 1996.
- [CARDA86] D.N. Card, V.E. Church, W.W. Agresti, "An Empirical Study of Software Design Practices", *IEEE Transactions on Software Engineering* 12 (2), 264-271, 1986.
- [CARDM85] D.N. Card, G.T. Page, F.E. McGarry, "Criteria for Software Modularization", *Proceedings IEEE Eighth International Conference on Software Engineering*, 372-377, 1985.
- [CHIDK91] S.R. Chidamber, C.F. Kemerer, "Towards a Metrics Suite for Object Oriented design", in A. Paepcke, (ed.) *Proc.*

- Conference on Object-Oriented Programming: Systems, Languages and Applications (OOPSLA'91)*, SIGPLAN Notices 26 (11), 197-211, 1991.
- [CHIDK94] S.R. Chidamber, C.F. Kemerer, "A Metrics Suite for Object Oriented Design", *IEEE Transactions on Software Engineering* 20 (6), 476-493, 1994.
- [EDEKS94] J. Eder, G. Kappel, M. Schrefl, "Coupling and Cohesion in Object-Oriented Systems", *Technical Report*, University of Klagenfurt, 1994.
- [FENTO96] N. Fento, S. L. Pfleeger, "Software Metrics: A Rigorous and Practical Approach", *International Thomson Computer Press*, London, 1996.
- [HENRL93] W. Li, S. Henry, "Object-Oriented Metrics that Predict Maintainability", *J. Systems and Software* 23 (2), 111-122, 1993.
- [HOGAN97] Jer Hogan, "An Analysis of OO Software Metrics", Department of Computer Science, University of Warwick, United Kingdom, 2nd May 1997
- [STEV74] W.P. Stevens, G.J. Myers, L.L. Constantine. "Structured design". *IBM Systems Journal*, vol. 2, no. 13. pp. 115-139. 1974.