

Software Measurement Process

Iulian RĂDULESCU, Intrasoft International S.A. iulian.radulescu@gmail.com

Liviu MIJACHE, Intrarom S.A. liviu@softwave.ro

Marius DUMITRACHE, Iptana S.A. mariu@softwave.ro

Software measurement process is becoming a key component for the development life cycle. It is very important to have constant information about the products that are developed, about how the processes are conducted and planning is followed. It helps in early identification of the problems, in offering support for managerial decisions and mitigating risks. The results can be also used in future projects, in this way, a company can create and maintain a database of software metrics and measures.

Keywords: *software measurement process, software metrics, statistical indicator, data collection.*

Fundamente

Un proces de măsurare software reprezintă o metodă sistematică de măsurare, analiză și ajustare a procesului de dezvoltare folosind date obiective. Cu o astfel de abordare sistematică, datele sunt colectate pe baza unor criterii bine stabilite, analizate din punct de vedere al caracteristicilor produsului dar și ale proceselor, și utilizate pentru a evalua progresul, calitatea și performanța de-a lungul ciclului de dezvoltare. Motivele principale pentru care procesele, produsele și resursele software sunt măsurate și analizate sunt:

- *pentru a caracteriza;* astfel, se înțeleg mai bine procesele, produsele, resursele și mediile de dezvoltare și de producție, dar se stabilesc și baze de comparație pentru analize viitoare.

- *pentru a evalua;* evaluarea ajută la determinarea stării proiectului, prin raportare la planuri. Astfel, sunt semnalate abaterile de la plan, și există posibilitatea de a adopta măsuri pentru controlarea acestor abateri. De asemenea, evaluarea ajută la analiza îndeplinirii obiectivelor de calitate și a impactului îmbunătățirii tehnologiilor asupra produselor și proceselor.

- *pentru a previziona;* măsurarea în scopul previzionării implică înțelegerea relațiilor dintre procese și produse, modelarea acestor relații, astfel încât valorile observate pentru anumite atribute să fie utilizate pentru a previziona altele. Aceste măsurători pot fi de

asemenea bază pentru extrapolarea tendințurilor; astfel, estimările pentru costuri, timp și calitate sunt actualizate pe baza datelor curente.

- *pentru a îmbunătăți.* Datele sunt culese în scopul de a identifica blocajele, cauzele problemelor ce apar, ineficiența, precum și alte oportunități de a îmbunătăți calitatea produsului și performanța procesului. Măsurătorile ajută și la planificarea și urmărirea eforturilor de îmbunătățire; măsurarea performanței curente furnizează o bază pentru a determina dacă acțiunile de îmbunătățire au rezultate sau nu, precum și dacă există efecte secundare ca urmare a acestora.

În structura oricărui proces de măsurare trebuie să se regăsească definirea clară a elementelor din procesul de dezvoltare ce sunt supuse măsurării, precum și a măsurătorilor software, adică a datelor elementare ce sunt colectate; procesarea datelor culese, evaluarea de indicatori, construirea de grafice și tabele statistice; analizarea indicatorilor pentru a extrage informațiile despre elementele supuse analizei; și în final utilizarea rezultatelor pentru a implementa îmbunătățiri sau pentru a identifica noi elemente ce necesită o analiză mai detaliată.

Elementele componente ale unui proces de măsurare

Conform [PARK96], măsurarea este un proces prin care numere sau simboluri sunt asig-

nate unor atribute ale unor entități din lumea reală, astfel încât să caracterizeze acele atribute prin reguli clar definite. Astfel, procesul de măsurare necesită:

- entități
- atribute, care sunt caracteristici ale entităților
- reguli (sau scale), pentru atribuirea de valori atributelor.

Există mai mult tipuri de entități pentru care se pot colecta date, cum sunt: produse, procese, resurse, artefacte, activități, agenți, organizații, medii, constrângeri. Entitățile pot fi la

rândul lor, colecții de alte entități. De exemplu, un proces software poate conține alte subprocesse precum și diverse fluxuri de execuție, fiecare producând, transformând și transmițând rezultate. Atributele sunt caracteristici sau proprietăți ale acestor entități. Arta măsurării stă în identificarea acelor atribute pentru care să se culeagă date și care oferă o imagine utilă asupra entităților analizate.

În tabelul 1 sunt exemplificări de entități, împreună cu atributele urmărite pentru fiecare în parte și măsurători propuse pentru aceste atribute.

Tabelul 1. Entități, atribute și măsurători

Tip Entitate	Entitate	Atribut	Măsurători
<i>Resurse</i>	personalul	dimensiunea echipei	număr oameni asignați
		experiența	- experiență în domeniu (ani) - experiență în programare (ani)
	instrumente CASE	tipul	nume tip instrument
		este utilizat?	Da/nu (clasificare binară)
	timpul	data de start / data de final	Date calendaristice
<i>Produs</i>	sistemul	dimensiune	- număr module - număr funcții punct - număr linii cod sursă
		densitatea defectelor	- număr defecte pe KSLOC - defecte pe funcție punct
	modulul	lungime	- număr linii cod sursă
		procent de utilizare	raport între liniile de cod nemodificate și numărul total de linii de cod
	unitatea de cod	număr de fluxuri de execuție liniar independente	complexitate McCabe
	document	lungime	număr de pagini
	defect	tip	nume de tipuri
		origine	nume activitate care l-a introdus
		severitate	set de clase de severitate
		efort pentru depanare	om-ore
	vârsta	timpul scurs (în zile), de la raportarea defectului	
<i>Proces</i>	dezvoltare	timpul scurs	- zile calendaristice - zile lucrătoare
		jaloane	date calendaristice
		efort de dezvoltare	om-ore, zile sau luni
		performanță	număr de teste trecute raportat la numărul total de teste
	testare	volum	număr de teste planificate
		progres	- număr de teste executate - număr teste trecute cu succes
	mentenanță	cost	- EURO pe an - Om-ore pe modificare

Înainte ca datele să fie colectate și utilizate, este important de luat în considerare scala de măsurare pentru informația culeasă. În funcție de aceasta, există diverse prelucrări sau

transformări ce se pot realiza asupra datelor. Sunt patru scale de măsurare acceptate: scala nominală, scala ordinală, scala de interval și scala de raport.

Într-o **scală nominală**, valorile pentru atribute sunt nume sau etichete. Ordinea valorilor pe scală nu are importanță. Singurele relații permise între valori sunt identitatea ($=$) și non-identitatea (\neq). De exemplu, în tabelul precedent, este menționat atributul „*tip instrument CASE*”, pentru care, ca posibile valori sunt tipurile existente de instrumente. Măsurătorile nominale sunt utilizate de regulă pentru a clasifica entitățile, astfel pentru ca ele să poată fi sortate înainte de numărarea aparițiilor unei anumite valori sau înainte de agregarea acestora. De exemplu, se dorește aflarea numărului de instrucțiuni executabile dintr-un program. Procesul de atribuire a unei instrucțiuni unei clase cum este „*instrucțiuni executabile*”, spre deosebire de clasa „*instrucțiuni de declarare*” sau clasa „*comentariu*”, reprezintă o măsurare realizată pe baza unei scale nominale. Când se analizează măsurimile nominale, indicatorii statistici folosiți cel mai des sunt modurile și frecvențele de apariție.

O **scală ordinală** permite ca rezultatele măsurate să fie plasate într-o ordine crescătoare sau descrescătoare. Cu toate acestea, distanța între pozițiile pe scală nu are importanță. Datelor li se atribuie numere de ordine, $1, 2, \dots, n$, numite *ranguri*. De exemplu, nivelul de experiență al unui programator poate fi măsurat ca *scăzut, mediu și ridicat*. Pentru ca un astfel de indicator să fie o metrică obiectivă, este necesar să existe criterii clare prin care se face încadrarea într-una sau în alta dintre categorii, astfel încât diverși observatori să asocieze întotdeauna aceeași valoare pentru un anumit programator.

O **scală de interval** aduce în plus conceptul de distanță. Valorile de pe o scală de interval, nu numai că pot fi ordonate, asemeni celor de pe scala ordinală, dar diferențele dintre ele au o semnificație. Pe scala de interval are sens definiția *distanței* dintre valorile înregistrate. Punctul zero ale acestei scale și unitatea de măsură se aleg în mod arbitrar. Datorită caracterului relativ al localizării originii pe această scală, nu are sens *suma* a două valori, precum și raportul acestora. În schimb, au sens, *diferența* dintre două valori și *suma* sau

raportul diferențelor. Metrica de complexitate a lui McCabe poate fi interpretată ca având o scală de interval. Diferențele între valori au semnificație; nu există zero absolut iar raportul dintre valori nu are vreo relevanță. Din punct de vedere statistic au sens media aritmetică, pentru studiul tendinței centrale, precum și deviația standard pentru analiza dispersiei.

O **scală de rapoarte** aduce în plus față de o scală de interval originea (o valoare zero aleasă nearbitrar și care are semnificație). Cu o origine reală, operațiile de înmulțire și împărțire au semnificație, ca de altfel toate operațiile matematice aplicate numerelor reale. Scalele de rapoarte rămân scale de rapoarte și după înmulțirea cu o constantă. De exemplu costul de dezvoltare, costul de integrare, timpul dintre două căderi ale sistemului, sunt mărimi pentru care se folosesc scale de rapoarte. Din punct de vedere statistic, au sens mediile (aritmetică, geometrică, armonică), dispersiile, deviațiile standard, precum și coeficientul de corelație.

Metrici software

O metrică reprezintă o valoare numerică calculată pe baza unei formule aplicată pe o colecție de date. În termeni statistici, o metrică este un indicator. Astfel, ea comportă numeroase forme analitice, specifice acestora. Poate fie un indicator primar sau derivat; indicatorii derivați se obțin prin comparări, abstracționări, generalizări și sintetizări, operații aplicate mărimilor absolute sau relative ale indicatorilor primari. Compararea a doi sau mai mulți indicatori se realizează fie sub forma unei diferențe, fie sub forma unui raport.

Pentru ca rezultatul unei metrici să fie utilizabil în analiza produsului sau procesului software, este necesar ca metrica să aibă anumite proprietăți.

În primul rând, metrica trebuie să fie *robustă*. Calculul unei metrici este un proces repetabil iar rezultatul este nesenzitiv la schimbări minore în mediul de lucru, asupra instrumentelor utilizate sau legate de observator. Măsurarea este precisă iar procesul de colectare a datelor necesare măsurării este obiectiv. Pe de altă parte, variații ale variabilelor inde-

pendente trebuie să determine variații ale valorilor metricii. Aceasta exprimă *caracterul sensibil* ale metricii la modificările factorilor care o influențează.

Metricile software, ca și indicatori statistici, trebuie să fie necompensatorii, adică la variații ale nivelurilor variabilelor independente nu trebuie să se obțină niveluri identice ale metricii. Această necesitate constituie un factor de influență important al analizelor, deoarece stă la baza ipotezei unicității situațiilor incluse în studiu. Pentru a asigura reprezentativitatea și semnificația rezultatelor trebuie evitate cazurile în care se obțin aceleași valori ale metricii pentru niveluri diferite ale variabilelor de intrare.

De asemenea, metrica trebuie să sugereze o normă, o scală și limite. Scala este foarte importantă în momentul în care se realizează comparații, în procesul de analiză a rezultatelor; limitele ajută în procesul de comparare. De exemplu, valoarea 0 (zero) este o limită inferioară pentru numărul de erori ale aplicației.

Un indicator este catastrofic dacă există valori particulare care fac imposibilă obținerea unei valori pentru metrică. Caracterul catastrofic al unui indicator nu implică utilizarea unui alt indicator care să fie lipsit de acest atribut, ci atrage atenția asupra situațiilor particulare. Acestea sunt definite de valori ale variabilelor ce sunt măsurate în condiții specifice. Utilizarea unor astfel de indicatori trebuie precedată de definirea și analiza clară a problemei. Astfel de situații apar în special pentru indicatori exprimați sub formă de raport, radical sau logaritm, și sunt legate de nerespectarea condițiilor matematice pentru realizarea calculelor: anularea numitorului, valori negative sau egale cu 0 pentru argumentul funcției logaritmice, radical din valori negative, etc.

În fine, dar nu în ultimul rând, o metrică trebuie să fie relevantă. Metrica este legată de produs sau de proces, de aceea trebuie să existe un motiv pentru a culege date în vederea evaluării acestei metrici.

De cele mai multe ori, o singură metrică este insuficientă pentru a măsura caracteristicile

unui design sau pentru a îndeplini obiectivele proiectului. Astfel, o colecție sau o suită de metrici este necesară pentru a obține o analiză mult mai consistentă a proiectului software. O astfel de suită trebuie să fie, la rândul ei, consistentă. Aceasta înseamnă că, de exemplu, dacă o valoare mai mică reprezintă o valoare bună pentru o metrică, același principiu trebuie să fie aplicat și celorlalte metrici din suită. O astfel de condiție ajută la analiza rezultatelor.

În plus, procesul de colectare a datelor, de unde rezultă colecțiile de date pe care metricile sunt calculate, trebuie planificat și condus cu atenție. Procesul de colectare fără un obiectiv definit, produce rezultate nerelevante.

Concluzii

Implementarea unui proces de măsurare în cadrul proiectelor software ale unei organizații nu trebuie să fie o decizie la nivel de proiect. Ea trebuie să facă parte din strategia acesteia de a crește în calitate și performanță. Nu este suficientă o simplă culegere a datelor. Întregul proces trebuie inițiat și condus după proceduri clare, prin care să se asigure obiectivitatea datelor, relevanța rezultatelor și implicit utilitatea deciziilor rezultate.

Bibliografie

[PARK96] Robert E. Park, Wolhart B. Goethert, William A. Florac - *Goal-Driven Software Measurement – A Guidebook*, CMU/SEI-96-HB-002, August 1996.

[MITR95] Constantin MITRUȚ, Alexandru ISAIC-MANIU, Vergil VIONEAGU – *Statistica pentru managementul afacerilor*, Editura Economică, București, 1999.

[IVAN04] Ion IVAN, Cătălin BOJA – *Metode statistice în analiza software*, Editura ASE, București, 2004.

[IVAN97] Ion IVAN, Mihai POPESCU, Panagiotis SINIOROS, Felix SIMION – *Metrici software*, Editura INFOREC, București, 1997.