

RECENZIE

Ion IVAN, Marius POPA, Sergiu CAPIZISU, Lukacs BREDA, Bogdan FLORESCU
 „*Clonarea informatica*”, Editura ASE Bucuresti, 2003, ISBN 973-594-310-7

Într-un domeniu nou, cartea „*Clonarea informatica*” elaborata de autorii Ivan, I., Popa, M., Capizisu, S., Breda L., si Florescu B. de la Academia de Studii Economice realizeaza un studiu riguros al procesului de clonare tratând fenomenul atât din perspectiva tehnica, în masura în care acesta este acceptat din punct de vedere legal si chiar recomandat pentru optimizarea produselor software, cât si din perspectiva constructiva si ofera un sistem de indicatori care determina situatiile în care un produs informatic este clona.

Problema prezinta o importanta deosebita în contextul actual al progresului tehnologiei informationale, a dezvoltarii tehnicilor si metodelor de realizare a aplicatiilor distribuite si de crestere în proportii foarte mari a costurilor impuse de asigurarea protectiei si securitatii tranzactiilor. În opinia autorilor, necesitatea clonelor este impusa de trecerea de la o versiune la alta a produselor program, si influenteaza pozitiv dinamica dezvoltarii software-ului atâta timp cât acestea sunt utilizate de autorii respectivelor produse si sunt mentinute în limitele legale.

În cuprinsul celor aproximativ 200 de pagini, autorii abordeaza gradat problema clonarii informatice clarificând principalele aspecte ale **procesului de clonare**, **modalitatile de obtinere a clonelor**, **aspectele juridice** ale acestora si completeaza lucrarea cu **studii de caz** ce evidentiaza cum sunt utilizate metodele si tehnicile de clonare. Noutatea adusa de autori reiese de la început din titlul cartii.

Capitolul 1, numit „*Procese de clonare*” familiarizeaza cititorii cu notiuni introductive despre programe, module, proceduri, macrodefinitii, respectiv definirea si caracteristicile acestora. Tot aici autorii explica posibilitatea de integrare a componentelor reutilizabile si redundanta din programe. Sunt descrise tehnicile folosite la dezvoltarea de aplicatii cu reutilizarea componentelor: selectarea pachetului de aplicatie, a componentelor reutilizabile, verificarea redundantei la nivel de produs software si în final identificarea componentelor reutilizabile candidate.

Redundanta se refera la realizarea unor copii ale unor aplicatii fiind asimilata backup-ului. Daca în cazul componentelor hardware redundanta nu este acceptata în anumite sisteme, în cazul software-ului este adesea recomandata pentru a îmbunatati calitatea programelor si toleranta la erori. Reutilizarea software-ului are în vedere fructificarea avantajelor pe care le ofera programele realizate anterior în vederea dezvoltarii de produse soft la costuri minime. Dupa prezentarea acestor aspecte autorii își focalizeaza atentia asupra caracteristicilor clonarii diferentiind procesul de copiere a unei entitati (programul per ansamblu, dar si componente organizate ale acestuia fiind considerate entitati) si procesul de clonare. Autorii dau urmatoarele definitii acestor notiuni:

1. „*A copia o entitate E_1 înseamna a realiza unele dintre componente cu sau fara interactiuni, obtinându-se o noua entitate E^1 ce executa sau nu functii din multimea F_1, F_2, \dots, F_n . Copia E^1 difera mai mult sau mai putin semnificativ de entitatea E_1 ”.*
2. „*A clona entitatea E_1 înseamna a dezvolta procese prin care se obtin componente, se genereaza interactiuni si se realizeaza functiile F_1, F_2, \dots, F_n obtinându-se noi entitati E^1, E^2, \dots, E^m , care nu difera cu nimic fata de entitatea de baza E_1 ”.*

În al doilea capitol autorii trateaza problema **ortogonalitatii fisierelor**. Cititorul este familiarizat mai întâi cu notiunile de baza: fisier, director, tipuri de fisiere, operatii la care se preteaza acestea. Pornind de la definitiile dreptei, planului si vectorilor ortogonali autorii elaboreaza o definitie conform careia datele D1 si D2 sunt ortogonale semantic daca continutul informational al acestora, sensul lor, difera într-o maniera categorica si semiotic, daca acestea

au o formalizare matematica total diferita. În continuare autorii definesc un indicator de ortogonalitate I_i , cu valori în intervalul $[0, 1]$, pe baza caruia poate fi verificata originalitatea unei lucrari sau clonarea software-ului facilitând determinarea asemanarilor dintre proceduri.

Pe parcursul urmatoarelor trei capitole se trateaza probleme specifice subiectului abordat cum ar fi: **clonarea software**, **clonarea datelor** si **clonarea interfetelor om-calculator**. Autorii explica clonarea programelor prezentând si tehnici de clonare a acestora însoțite de exemplificari, clarificând aspecte esentiale. De asemenea sunt tratate în mod original si efectele clonarii asupra complexitatii programelor. Cititorilor le sunt oferite detalii despre clonarea datelor si anume: pasii procesului de clonare a bazelor de date, posibilitati de clonare a bazelor de date (clonare integrala, clonarea structurii bazei de date, clonarea partiala a structurii bazei de date cu mentinerea legaturilor între tabelele clonate, clonarea bazelor de date cu adaugarea de noi tabele), masurarea eficientei procesului de clonare, beneficiile aduse de acest proces. Treptat acestia sunt introdusi în problematica clonarii interfetelor om-calculator oferindu-se explicatii despre acest proces, despre tehnicile si metodele avizate în acest tip de clonare, despre importanta acesteia pentru producatorii de software.

Fenomenul de clonare informatica este abordat din perspectiva prevederilor legale în capitolul 7 „**Aspecte juridice ale clonarii informatice**”. Autorii trateaza detaliat acest subiect, cu referinte la legislatia în vigoare, oferind cititorului posibilitatea de a afla modul în care este protejat prin lege dreptul de autor, cum poate beneficia de avantajele pe care le ofera acesta (inclusiv dreptul de transmitere a acestuia catre alta persoana - cesiune) si ce pedepse sunt aplicate în cazul în care îi este încălcat sau el îl încălca. Sinteza realizata de autori este binevenita deoarece destul de putini sunt familiarizati la noi cu aceste prevederi legale în contextul actual în care tehnologia informatica cunoaste o tot mai mare întindere.

În capitolul 8 „**Metrici ale clonarii**” autorii supun atentiei cititorului notiunea de calitate a programelor si folosesc conceptul de metrica din matematica pentru a aprecia aceasta. Ei exemplifica aceste metrici la determinarea ortogonalitatii fisierelor scrise în C++. Tot aici definesc notiunea de amprenta a programului si cum poate fi determinata pe baza acesteia dacă este original sau clona.

Ultimul capitol abordeaza alte aspecte ale clonarii informatice care nu au fost tratate anterior. Se pune în special accentul pe implicatiile si efectele economice ale acestui fenomen. Ei prezinta cititorului câteva instrumente pentru identificarea clonelor si anume: programul *Texte.exe* capabil sa determine o serie de indicatori prezentati în capitolele anterioare pentru analiza gradului de asemanare între texte, programul *Orto.exe* care determina valorile calculate pe fisiere sursa C++ pentru analiza gradului de asemanare între programele sursa scrise în limbajul de programare C++ si programul *Devize.exe* pentru determinarea ortogonalitatii în cadrul unei multimi de devize de cheltuieli toate realizate de Marius Popa, un autor al acestui volum.

Cartea se adreseaza în special producatorilor de software propunându-si sa realizeze o delimitare exacta între productia reala de software si consecintele lipsei controlului asupra clonelor software. Bineînțeles ea nu este destinata doar acestora fiind atractiva pentru tinerii cercetatori în domeniu informaticii, oferindu-le noi directii de cercetare. Bibliografia generoasa studiata si citata, contribuie din plin la reusita acestei carti.

Autorii trateaza cu profesionalism, originalitate si profunzime problema clonarii deschizând portile unui domeniu de cercetare în informatica aflat înca la început în tara noastra dar cu perspective impresionante.