



dr hab. inż.
Bartosz Walter

Pracuje w Instytucie Informatyki Politechniki Poznańskiej oraz Poznańskiego Centrum Superkomputerowo-Sieciowego. Zajmuje się badaniem ewolucji oprogramowania oraz metod utrzymania kodu programów. Z zamiłowania jest popularyzatorem nauki.

bartosz.walter
@cs.put.poznan.pl

ŻYJ 100 LAT, KOMPUTERKU!

Programy komputerowe starzeją się tak samo jak organizmy żywe. Żeby działały bezawaryjnie, wymagają systematycznej konserwacji.

Bartosz Walter

Instytut Informatyki, Politechnika Poznańska
Poznańskie Centrum Superkomputerowo-Sieciowe

Znana frazka Jana Kochanowskiego smutno konstatuje, że wartość zdrowia doceniamy dopiero wówczas, gdy je utracimy. W szerszym ujęciu dotyczy to zresztą nie tylko zdrowia i nie tylko ludzi: przemijanie i naturalna utrata sprawności dotyczy przecież każdego organizmu żywego. Dlatego to właśnie starość jako ten okres, w którym kumulują się skutki wcześniejszych zdarzeń, zaniedbań i przebytych schorzeń, jest tym momentem, gdy przypominamy sobie mądrość płynącą z frazki – często poniewczasie. Czy zatem możemy wziąć ją sobie do serca nieco wcześniej, gdy rada mistrza Jana może jeszcze odnieść pożądany skutek? Oczywiście – wszyscy to w jakiś sposób robimy: w pogoni za wiecznym zdrowiem i młodością staramy się przecież opóźnić nadejście starości, odwiedzając siłownie, ćwicząc, zdrowo jeżdżąc, słowem – prowadząc tzw. zdrowy tryb życia bądź

też stosując rozmaite terapie rehabilitujące. Działania te wprawdzie nie zlikwidują starości, ale mogą pozwolić na utrzymanie sprawności organizmu przez wiele lat. Zatem w pewnym sensie to od zdolności przeciwdziałania starzeniu się zależy nasze zdrowie, a także tak długo, jak mamy siłę (i środki), by te działania podejmować, jesteśmy zdrowi i zdolni do dalszej aktywności.

Co ciekawe, w takim ujęciu „zdrowie” nie dotyczy jedynie człowieka czy organizmów żywych, ale również innych przedmiotów czy wytworów ludzkiej pracy. Na przykład budynki są projektowane i wznoszone w określonym celu (mieszkalnym, użytkowym, dekoracyjnym), jednak często są w ciągu swojego życia wielokrotnie poddawane adaptacjom, rekonstrukcjom czy przebudowom. Budowla taka pozostaje użyteczna, gdy spełnia aktualne oczekiwania lub gdy można ją do nich dostosowywać. Gdy okazuje się to niemożliwe – budynek w pewnym sensie funkcjonalnie umiera, zupełnie jak człowiek.

Choć proces starzenia się może mieć bardzo różne oblicza, to jedno pozostaje niezmiennie: o zdrowiu decyduje zdolność adaptowania się do zmian. W świecie biologii oznacza to odporność na choroby, umiejętność regeneracji zużytych organów oraz aktywne,

wyprzedzające przeciwdziałanie objawom starości. W budownictwie, mechanice i wielu innych dziedzinach – to zapobieganie zużyciu eksploatacyjnemu i utrzymanie zdolności do podjęcia nowych funkcji czy utrzymaniu dotychczasowych parametrów.

Starzejące się bajty

O procesie starzenia się można też mówić w odniesieniu do tak zaskakującej dziedziny jak... budowa programów komputerowych. Na pierwszy rzut oka to stwierdzenie wydaje się nedorzeczne: owszem, starzeje się komputer, psuje drukarka, zużywa monitor, ale jaka starość może spotkać program, czyli ciąg instrukcji do wykonania przez maszynę? Przecież on co najwyżej może stać się przestarzały, niedostosowany do współczesnych wymagań i oczekiwań. A jednak programy również podlegają zmianom i ewolucji. Każdy z nich ma przecież błędy wymagające poprawek, a czasem ich użytkownicy chcą ulepszeń i nowych funkcji.

Zdarza się też, że program przestaje działać po aktualizacji systemu operacyjnego i konieczne jest jego dostosowanie. Na czym więc polega starzenie się oprogramowania? Otóż każda zmiana, niezależnie od swojej przyczyny, ma w sobie pewien potencjał destrukcyjny, wynikający z kumulacji poprawek, niespójności z początkowym zamysłem projektanta czy ryzyka wprowadzenia nowych błędów przy okazji poprawiania poprzednich. To oznacza, że rachunek wystawiany za implementację dowolnej zmiany programu ma dwie kolumny: z jednej strony wynika on z kosztu napisania kodu, przetestowania go i uruchomienia – czyli jest kosztem bezpośrednim, z drugiej jednak każda wprowadzona poprawka czy nowa funkcja jednocześnie utrudnia lub nawet uniemożliwia dalsze zmiany, które pojawiają się w przyszłości. Żeby temu przeciwdziałać, należy wykonać dodatkową pracę polegającą na „wygładzeniu” zmiany, usunięciu „blizn” – z czym wiąże się dodatkowy wysiłek. To trochę jak z rehabilitacją po inwazyjnym zabiegu: wprawdzie w niektórych przypadkach można ją pominąć, ale zwykle dzieje się to kosztem jakości, a w dalszej perspektywie także długości życia. Tak jak trzeba dbać o siebie, trzeba też dbać o utrzymanie budynków i programów komputerowych. Programiści taką niezapłaconą na czas drugą stronę rachunku nazywają długiem technologicznym, ponieważ on nie znika, lecz powoduje, że w przyszłości konieczne będzie spłacanie go w ratach w postaci dodatkowego czasu poświęconego na utrzymanie programu. I jak to z długiem: jest on akceptowalny, a nawet pozwala na różne przyjemności, o ile tylko spłata nie zużywa znacznej części bieżących dochodów. Niestety, po przekroczeniu pewnej granicy łatwo wpaść w spiralę zadłużenia, w której większość pieniędzy i energii idzie na utrzymanie „staroci”, a nie rozwój „nowości”, a z której wydostanie się

jest bardzo trudne. Jeżeli to się nie uda, wówczas program czeka niechybna śmierć, bo nie będzie w stanie przyjmować zmian, a zmiana jest oznaką żywotności.

Badania przesiewowe

Żeby być pewnym stanu swojego zdrowia, można ciągle poddawać się zaawansowanym badaniom, sprawdzając, czy delikatna granica zdrowotnej równowagi nie została przekroczona. Jest to jednak bardzo kosztowna i niezbyt zachęcająca perspektywa. Można również skonstatować, że skoro starość nikogo nie ominie, to należy ją po prostu zaakceptować. Życiowy pragmatyzm nakazuje jednak poszukiwanie trzeciej, pośredniej drogi, dającej korzyści obu skrajnych rozwiązań, a jednocześnie ograniczającej ich koszt. W medycynie taką rolę odgrywają badania przesiewowe, które skupiają się na tanich, stosunkowo łatwych do zauważenia objawach, nawet jeżeli nie są specyficzne i czasem mogą wprowadzić w błąd. Podobnie jest w przypadku programów: zamiast poddawać je ciągłej analizie i przeglądowi, sku-

Podobnie jak w przypadku schorzeń zdrowotnych nie wszystkie dolegliwości programów komputerowych wymagają szybkiej reakcji.

piamy się na pewnych symptomach, o których wiemy, że prowadzą do kłopotów i szybkiego wzrostu długu technologicznego. Znamy obecnie kilkadziesiąt takich symptomów. Dotyczą one np. złożoności modułów, ich nieczytelnej struktury, dużej liczby powtórzeń lub silnej zależności od innych komponentów, które mogą „zarażać” błędami sąsiadów. A skąd właściwie wiemy, że takie konstrukcje są ryzykowne? Stosujemy te same metody co nowoczesna medycyna oparta na dowodach: zbierając dane o skuteczności terapii w odniesieniu do poszczególnych chorób i analizując je statystycznie. W efekcie znajomość losów podobnych programów komputerowych pozwala dość trafnie, choć z wszelkimi statystycznymi zastrzeżeniami, wnioskować o przyszłości.

Podobnie jak w przypadku schorzeń zdrowotnych nie wszystkie dolegliwości programów komputerowych wymagają szybkiej reakcji. W większości przypadków wystarczy monitorowanie ich stanu, by uniknąć przykrych niespodzianek. Dzięki temu przy stosunkowo niewysokich kosztach możliwe jest skuteczne zarządzanie dobrostanem programów, by mogły cieszyć swoich użytkowników przez długie lata. ■

Chcesz wiedzieć więcej?

Król K., *Code decay, software erosion and software entropy*, 2019, homeproject.pl/2019/09/20/code-decay-software-erosion-and-software-entropy

Parnas D.L., *Software aging. In Proceedings of the 16th international conference on Software engineering (ICSE '94)*, 1994.