

Nove možnosti za premagovanje ovir, ki so značilne za življenjski cikel razvoja sistema, je ponudil prototipni pristop izgradnje računalniških obdelav. V drugih področjih, in še zlasti v materialni proizvodnji, je prototipni pristop pri nastajanju končnega izdelka še dolgo običajna praksa. Pri izdelavi uporabniške programske opreme pa se ta metodologija uveljavlja šele zadnja leta. Razlog za tako pozno vključevanje prototipnega pristopa za razvoj programske opreme je treba iskati predvsem v nezadostni razvitosti informacijske tehnologije za tovrstne pristope še v bližnji preteklosti. Še več: tudi danes je prototipni pristop pionirski pristop, ki se je kljub nekaterim svojim prednostim šele začel uveljavljati.

2. Tehnika prototipnega pristopa

Izdelava računalniškega prototipa (angl. prototyping) je metoda za oblikovanje, predstavitev in izpopolnitev uporabnikovih potreb z izgradnjo delujočega modela končnega sistema - in sicer hitro in v tesnem sodelovanju z bodočim uporabnikom. S postopnimi iterativnimi izboljšavami (sl. sl.2.) izhodnega prototipa kot rezultat medsebojnega, vedno boljšega razumevanja problema, pridemo lahko končno do prototipa, ki v bistvenih potezah zadovoljuje potrebe naročnika. Izdelava prototipov se zelo razlikuje od pristopov izgradnje računalniško podprtih informacijskih sistemov po načelu življenjskega ciklusa razvoja. Filozofija uporabnosti prototipa sloni na človeški lastnosti, da je običajno lažje kritizirati obstoječe kot ustvarjati. Potrebno je poudariti, da je prototip v bistvu poenostavljena interaktivna obdelava, običajno sprogramirana z enim izmed t.i. programskih jezikov četrte generacije (J4G) in da je razvoj prototipa v bistvu heurističen razvoj.

Možnost uporabe prototipnega pristopa v veliki meri sloni na dejstvu, da je večina poslovnih obdelav medsebojno precej podobnih po svoji naravi. Potrebno je oblikovati podatkovne baze, obstoječim podatkom dodajati nove podatke, spreminjati vsebino le-teh, jih presledovati in jih končno brisati, če jih ne rabimo. Veliko je podobnosti tudi med nekaterimi sistemskimi modeli, modeli podatkov, modeli funkcij, v zasnovi obdelav in oblikami poročil. Zato se v svetu vedno bolj uveljavlja pristop "ponovne uporabe uporabniških programov" (angl. reusable software). Dobre programske rešitve je pač treba spraviti v knjižnico programov in jih po potrebi uporabiti v podobni situaciji. Dostikrat so potrebne le neznatne spremembe programov, ki se lahko hitro izvedejo s sodobnimi interaktivnimi urejevalniki (editorji).