

## 6 UPORABA INFORMACIJSKIH ORODIJ

Eden od ključnih pogojev gradnje sodobnih informacijskih sistemov, pri katerih zahtevamo postopnost gradnje s sprotnim preverjanjem ustreznosti aplikativnih rešitev, fleksibilnost in prilagodljivost spremembam informacijskih potreb, je uveljavitev prototipnega pristopa ob uporabi sodobnih informacijskih orodij.

Prototipni pristop je na področju gradnje informacijskih sistemov že dalj časa prisoten, vendar tehnološka raven uporabljenih informacijskih (bolje rečeno programskih) orodij ni omogočala njegove uspešne uveljavitve. Problem je izhajal prvenstveno iz pomanjkanja fleksibilnosti, počasnosti razvoja in neuporabnosti izdelanih prototipov kot delujočih rešitev.

V zadnjem času, s pojavom interaktivnih informacijskih orodij, so dani vsi tehnološki pogoji za polno uveljavitev prototipnega pristopa, ki omogoča sprotno in popolno vključitev uporabnikov v razvoj prototipov in gradnjo delujočih rešitev. Ta informacijska orodja v večini primerov slonijo na programskih jezikih četrte generacije, povezanih z relacijskimi bazami podatkov in katalogi podatkov.

**Jeziki četrte generacije** se razvijajo predvsem s ciljem dviga produktivnosti in uporabniške prijaznosti v primerjavi z uporabo postopkovnih jezikov prejšnje generacije (kot so FORTRAN, COBOL, PL/1...). Največkrat so nepostopkovni, kar pomeni, da z njimi "povemo računalniku" KAJ naj izvede namesto KAKO naj to izvede. Produktivnost razvoja rešitev se tako dvigne v povprečju vsaj za faktor 10. Največkrat zajemajo jezike za vzdrževanje baze podatkov, generatorje zaslonskih slik in poročil, poizvedovalne (query) jezike za pridobivanje ad-hoc podatkov iz baze podatkov. Nekateri so povezani z orodji za podporo odločanju, kot so ekstraktorji podatkov, preglednice, grafična orodja, knjižnice modelov odločanja in ostala matematično-statistična orodja.

Delovanje krmilnih sistemov baz podatkov oz. koncepta **relacijskih baz podatkov** sloni na medsebojno povezanih tabelah podatkov (datotekah), kjer se povezave med podatkovnimi elementi in datotekami dinamično prilagajajo potrebam uporabe. Takšno tehnologijo pa pogojuje uporaba prototipnega pristopa, kjer se opredelitve posameznih predmetov podatkov (entitet) in njihovih medsebojnih povezav često spreminjajo, dodajajo in brišajo, ne da je ob tem nujna ponovna zasnova in programiranje aplikativnih rešitev.