

METODOLOŠKI PRISTUP RAZVOJU PROGRAMSKIH SUSTAVA U PROCESIMA ČUVANJA POLJOPRIVREDNIH PROIZVODA

S. PLIESTIĆ¹, V. GRBAVAC¹, M. ŽAKULA²

¹Agronomski fakultet Sveučilišta u Zagrebu

¹Faculty of Agriculture University of Zagreb

²Euclid d.o.o. Vinkovci

SAŽETAK

Potreba za efikasnošću i visokom kakvoćom izgradnje programskih sustava u procesima čuvanja poljoprivrednih proizvoda, kao i pomagalima nužnim za automatiziranu podršku takvom pristupu, sve više zahtjeva organizirani program uvođenja metodologije razvoja programskih sustava.

Metodologija je prvenstveno razvijena za uklopljene sustave koji rade u realnom vremenu, ali se zbog kompletnosti pristupa podjednako može primjenjivati i na standardne aplikacijske sustave. Bazirana je na strukturnom pristupu Yourdon-ovog tipa kojeg karakterizira kreiranje više slijednih modela sustava, u skladu s faznim pristupom razvoju sustava i principom prikrivanja informacija. Na taj način se sustav postupno izgrađuje i usavršava, kombinirajući informacije prethodnih metoda modeliranja s elementima značajnim za daljnji razvoj sustava.

Ključne riječi: metodologija, automatizacija, informatizacija,

1. UVOD

Činjenica je, da je s praćenjem razvoja djelatnosti vezanih uz izradu programskih paketa ustanovljena i potreba za zaokretom u tehnologiji izrade tih programskih sustava, kako bi se:

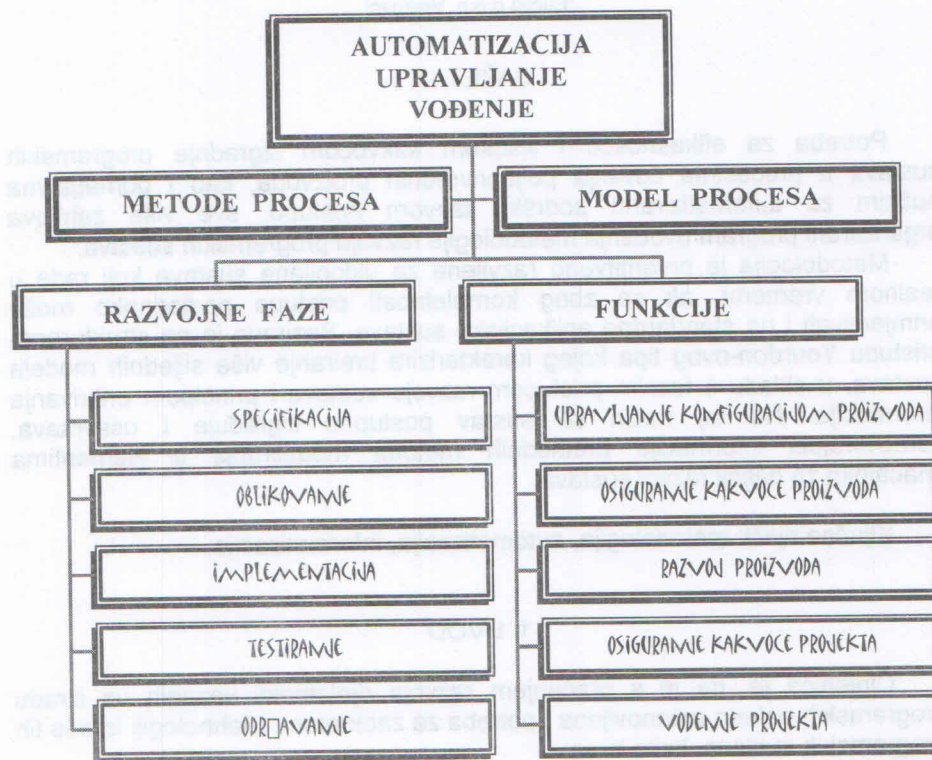
- savladala kompleksnost razvoja,
- olakšalo održavanje,
- smanjilo nastajanje i omogućilo rano otkrivanje pogrešaka,
- povećala učinkovitost,
- postigla željena kakvoća,
- stimulirala kreativnost članova tima.

Stoga, je potreba za kvalitetnom i efikasnom metodologijom izgradnje softverskih sustava, kao i pomagalima nužnim za njenu automatiziranu podršku rezultirala organiziranim programom zadovoljavanja navedenih zahtjeva.

2. PROGRAM INTEGRALNOG PRISTUPA PROGRAMSKOM INŽENJERINQU

Osnovna ideja programa integralnog pristupa programskom okruženju leži u kreiranju okruženja za proizvodnju programskih paketa kao zbroja modela procesa, i metoda vezanih za automatizacije proizvodnje programskih sustava (slika 1).

Slika 1. Struktura metodologije razvoja programskih sustava



2.1. Model procesa u okruženju za proizvodnju programskih paketa

U konceptijskom smislu rečeno, model samog procesa proizvodnje programskih sustava čini temelj okruženja, a njegova osnovna funkcija je opisivanje niza događaja u kreiranju programskog proizvoda. Zapravo, opisom mora biti obuhvaćeno ŠTO se radi, KAKO se to izvršava, KADA se smatra završenim, KAKO se provjerava i u kojoj se fazi radnih postupaka završena komponenta prosljeđuje. U biti, model procesa navedenih sustava mora

podržavati razvojni proces po fazama životnog ciklusa proizvoda, kao i proces vođenja projekta koji se proteže kroz sve faze, a osnovna uloga tog procesa je planiranje, praćenje i kontrola odvijanja projekta. Model procesa mora također biti dovoljno fleksibilan da regulira zahtjeve za izmjenom, kao i specijalne slučajeve koji se ne uklapaju u standardnu shemu proizvodnje. Zapravo, od dobrog modela procesa se traži kako dovoljna općenitost u smislu pokrivanja šireg spektra programskih sustava, tako i egzaktnost, tj. precizan i kompletan način opisivanja komponenti i produkata koji sačinjavaju svaki sustav.

2.2. Metode u okruženju za proizvodnju programskog paketa

Također je činjenica da bi metode u okruženju za proizvodnju programskih paketa, trebale eksplicitnim uputama za obavljanje aktivnosti u izabranom modelu procesa, regulirati cjelokupan proces proizvodnje navedenog sustava.

No, temeljne karakteristike metoda moraju zadovoljiti kriterije u odnosu na:

1. Perspektivu iz koje se promatra problem:

- a. informacijska (odnos između podataka),
- b. funkcijska (tijekovi i transformacije podataka),
- c. dinamička (slijed i sinkronizacija transformacije podataka).

Zapravo, svaki problem na koji se primijeni određena metoda ima perspektivu koja je korisnija od neke druge, ako ta perspektiva nije podržana primjenom metodom rezultat neće biti uspješan.

2. Domenu na koju se primjenjuje, a koja je određena klasom problema i fazom životnog ciklusa.

U biti je klasa problema definirana nekim atributima programskog proizvoda, kao što je npr. postojanjem vremenski kritičnih međuveza, numeričkih proračuna, potrebe za visokom pouzdanošću itd. No, također je moguće pokazati da postoje ortogonalne klase proizvoda u kojima nisu sve metode podjednako efikasne, jer je u biti metoda korisna ako je primjenjiva na odgovarajuću klasu problema.

Drugi čimbenik je određen fazom životnog ciklusa u kojoj se proizvod trenutno nalazi, jer specifičnost informacijskog sadržaja svake od faza diktira primjenu kvalitativno različitih metoda.

3. Zastupljenost principa:

- a. modularnosti,
- b. prikrivanja informacija,
- c. lokalizacije,

- d. konceptualnog integriteta,
 - e. kompletnosti (potpunosti),
 - f. apstrakcije (pretpostavke),
 - g. formalizacije,
 - h. hijerarhijske organizacije komponenti rješenja problema.
4. Sredstvo izražavanja metode:
- i. tekstualno,
 - j. grafički,
 - k. kombinirano.
5. Kakvoću naputaka za provođenje metode.
6. Mogućnost integracije s ostalim metodama u integralnu metodologiju razvoja programskih sustava.

Činjenica je, da metode ne bi trebale samo određivati koje korake poduzeti u obavljanju određenih grupa aktivnosti, već i davati smjernice u poduzimanju tih koraka kao i kriterije provjere korektnosti, kompletnosti i kvalitete poduzetih radnji. Dakle, kombinacija i integracija izabranih metoda definira jedinstvenu metodologiju proizvodnje softverskih sustava koja bi se prema iskustvima u upotrebi i novim dostignućima računarskih znanosti modificirala i modernizirala.

2.3. Automatizacija u okruženju za proizvodnju programske podrške

Automatizacijom kao završnim elementom strukture okruženja za proizvodnju programskog paketa postiže se ubrzanje rada na metodološki uređenim dijelovima procesa proizvodnje i time se u biti, smanjuju troškovi izrade navedenih sustava. Stoga, "ručnim" pristupom izradi imalo složenijeg sustava ne mogu se, zbog ograničenja ljudskog čimbenika, izbjeći nekonzistentnosti u pristupu, neregularno provođenje izmjena i poteškoće u primjeni formaliziranih metoda. U biti, automatizacija povlači mogućnost primjene rigoroznijih metoda, što otvara vrata primjeni automatiziranih provjera i korekcija prema pravilima ugrađenim u formalizirano tijelo metode.

Danas više nije aktualno pitanje: "Automatizacija - da ili ne?", već: "Automatizacija - kako i koliko?".

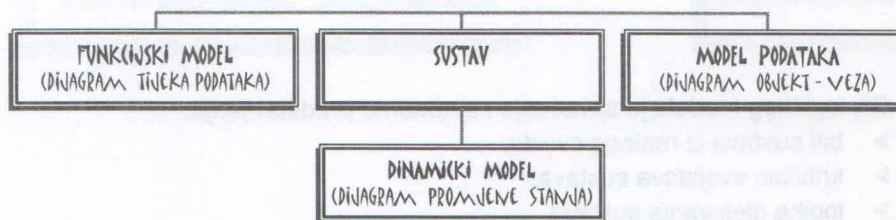
Činjenica je da, postojeća automatizirana pomagala nisu zastupljena u jednakoj mjeri kroz sve faze i grupe aktivnosti životnog ciklusa programskog paketa kao proizvoda, a razlozi leže u različitoj prirodi informacija karakterističnih za pojedinu grupu.

3. REALIZACIJA PROGRAMA

Gotovo je uobičajeno da se kao jedinstven okvir rješavanja problema izabire fazni model procesa proizvodnje programskih paketa unutar kojeg se svaka faza promatra u kontekstu karakterističnih aktivnosti, njihovog slijeda i načina provođenja, načelnog izgleda proizvoda, te općih smjernica rada i načina provjere aktivnosti.

Stoga se najčešće rabi metodologija bazirana na strukturnom pristupu Yourdonovog tipa namijenjenog prvenstveno uklopljenim sustavima koji rade u realnom vremenu. Zapravo, metodologija je sastavljena od niza strukturnih metoda integriranih u cjelinu koja pokriva životni ciklus sustava u rasponu od specifikacije do testiranja te se problem razmatra iz sve tri perspektive (funkcijske, informacijske i dinamičke) kreiranjem i povezivanjem modela transformacija, modela spremljenih podataka i dinamičkog modela (slika 2). Stoga je, zbog tako zastupljene kompletnosti pristupa, metodologija primjenjiva i na standardne aplikacijske sustave.

Slika 2. Modeliranje sustava iz perspektive funkcija, dinamike i informacija



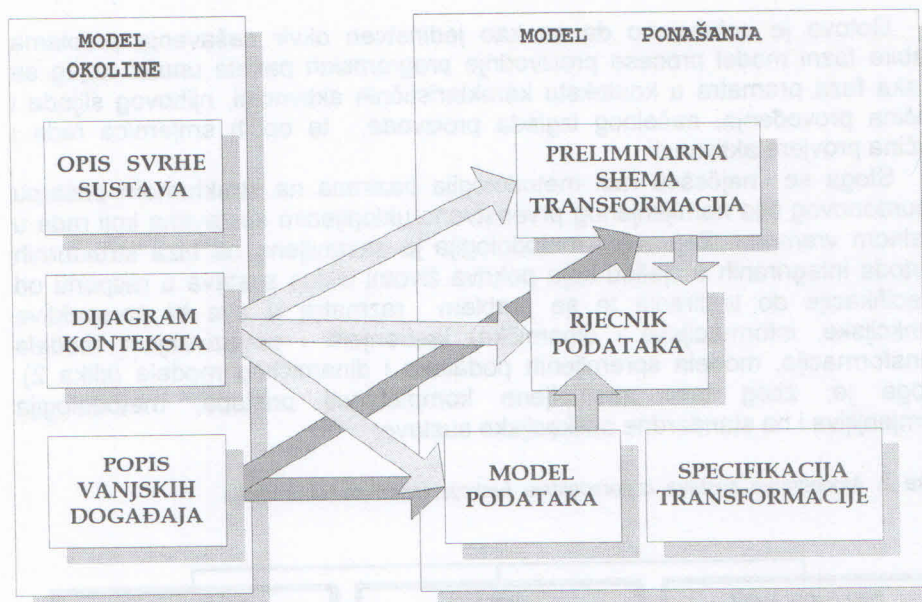
Sam pristup specifikaciji i oblikovanju sustava karakteriziran je izgradnjom sustava postupnim usavršavanjem te kreiranjem više modela, u skladu s principom prikriivanja informacija, konceptualnog integriteta i hijerarhijske organizacije komponenti rješenja problema. Stoga se model sustava realizira kao sustav podmodela:

- okoline sustava,
- ponašanja sustava,
- konfiguracije procesora,
- programskih elemenata,
- organizacije koda,

Ti modeli se međusobno grade i nadopunjavaju, a u biti svaki razrađuje zaseban aspekt sustava na osnovi izgrađenog u prethodnim modelima.

Prva etapa razvoja rezultira logičkim modelom (model okoline + model ponašanja sustava) (slika 3.)

Slika 3. Razvoj modela logike



Cilj logičkog modela je shvaćanje i strukturno predstavljanje:

- biti sustava iz realnog svijeta,
- kritičnih svojstava sustava,
- logike djelovanja sustava
- osnovne strukture sustava,
- tijeka podataka u, iz i kroz sustav
- upravljačke logike sustava,
- potreba sustava iz domene pohranjenih podataka (struktura, svojstva i organizacija podataka),

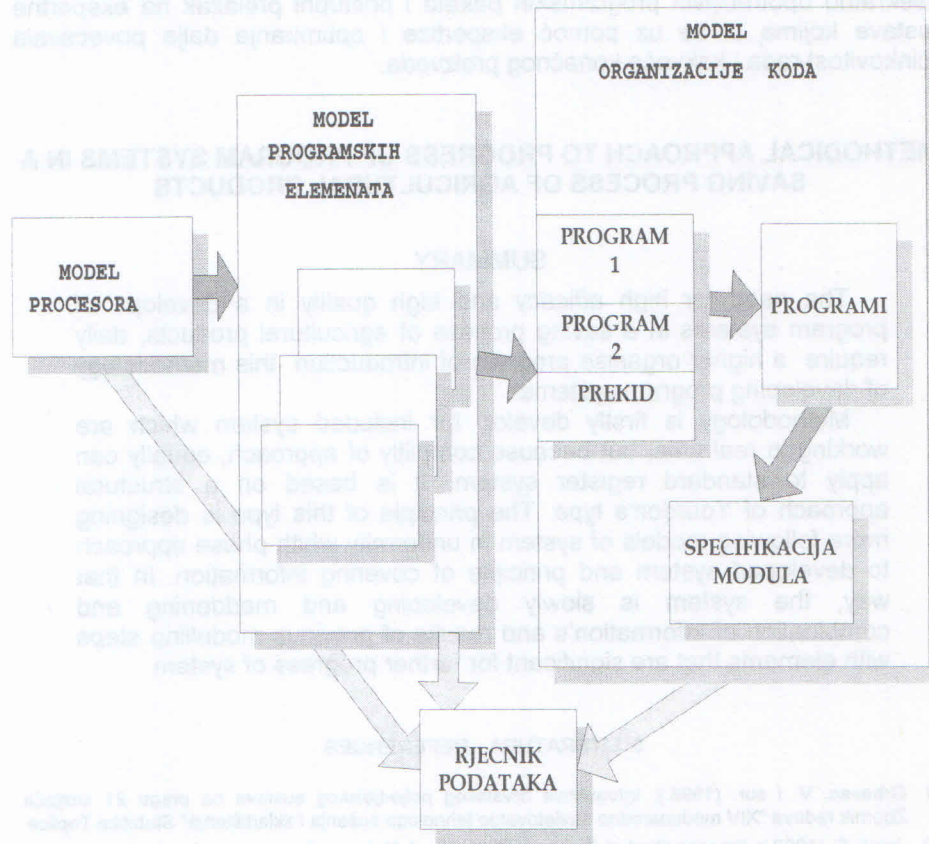
U biti, dodavanjem detalja i implementacijskih ograničenja, unošenjem osobitosti konfiguracije sustava i načina realizacije procesa te razradom algoritamske i sinkronizacijske komponente, dobiva se implementacijski model.

Dakle, izgradnja implementacijskog modela koja čini drugu etapu strukturnog razvoja odvija se u tri temeljna koraka i to u:

- a. preslikavanju logičkog modela na izabranu konfiguraciju procesora uz sačuvanje ponašanja sustava (model procesora),
- b. reorganizaciji modela procesora s ciljem definiranja programskih elemenata koji predstavljaju samostalne jedinice s kojima program sustava u procesoru manipulira (pokreće, zaustavlja, prekida, nastavlja),

c. modeliranju strukture koda na nivou modula unutar svakog programskog elementa (model organizacije koda) (slika 4).

Slika 4. Razvoj modela implementacije



Sredstvo izražavanja zajedničko je za fazu specifikacije i oblikovanja sustava, a sastoji se od kombinacije tekstualnih specifikacija i nekoliko tipova dijagrama. Time se postiže jedinstvenost pristupa kreiranju i dokumentiranju sustava, kao i zajednički jezik analitičara i oblikovatelja sustava.

4. ZAKLJUČAK

Daljnje povećanje učinkovitosti biti će postignuto uvođenjem automatiziranih pomagala čime će analitičari i konstruktori putem radnih stanica,

priključenih na računalo sa centralnom bazom podataka projekta, biti oslobođeni velikog dijela "ručnog"posla. Stoga se planira i uključivanje automatiziranih pomagala iz područja planiranja resursa, procjenjivanja troškova i tijeka projekta, kao i upravljanja konfiguracijom. Dugoročni planovi predviđaju daljnju supstituciju "ručnih" poslova automatiziranim, generiranje višekратно upotrebljivih programskih paketa i postupni prelazak na ekspertne sustave kojima bi se uz pomoć ekspertize i optimiranja dalje povećavala učinkovitost rada i kakvoća konačnog proizvoda.

METHODICAL APPROACH TO PROGRESS OF PROGRAM SYSTEMS IN A SAVING PROCESS OF AGRICULTURAL PRODUCTS

SUMMARY

The need for high efficacy and high quality in a develop of program systems in a saving process of agricultural products, daily require a higher organise program of introduction this methodology of developing program systems.

Methodology is firstly develop for included system which are working in real time, but because complitly of approach, equally can apply to standard register system. It is based on a structural approach of Yourdon's type. The principle of this type is designing more following models of system in uniformity whith phase approach to developed system and principle of covering information. In that way, the system is slowly developing and maddening and combination of information's and results of previous modelling steps with elements that are significant for further progress of system.

5. LITERATURA - REFERENCES

1. Grbavac, V. i sur. (1998.): Infoosnova hrvatskog poljodjelskog sustava na pragu 21 stoljeća. Zbornik radova "XIV međunarodno savjetovanje tehnologa sušenja i skladištenja" Stubičke Toplice
2. Jović, F. (1992.): Process Control Systems. Chapman & Hall, London
3. Jović, F. (1993.): Proces sušenja kao stacionarni nelinearni dinamički sustav. Zbornik radova "IX međunarodno savjetovanje tehnologa sušenja i skladištenja" Stubičke Toplice
4. Mc Ellihiney, R. (1995.): The future of feed technology. Zbornik radova "XI međunarodno savjetovanje tehnologa sušenja i skladištenja" Stubičke Toplice
5. Netushil, A. (1978.): Theory of Automatic Control, MIR Publishers, Moscow
6. Plietić, S. (1993.): Automatsko upravljanje procesom peletiranja. Zbornik radova "IX međunarodno savjetovanje tehnologa sušenja i skladištenja" Stubičke Toplice
7. Rolle, I. (1988.): Verbesserte Wettbewerbsfaehigkeit durch integrierte Datenverarbeitung. Die Muehle + Mischenfuttertechnik, Vol 125. No 35.
8. Sekulić, D. (1995.): Reengineering. Zbornik radova "XI međunarodno savjetovanje tehnologa sušenja i skladištenja" Stubičke Toplice
9. Šurina, T. (1981.): Automatska regulacija, Školska knjiga, Zagreb

10. Vojšta Duda, Ljiljana (1995.): Fuzzy logika kao unapređenje automatizacije. Zbornik radova "XI međunarodno savjetovanje tehnologa sušenja i skladištenja" Stubičke Toplice
11. Frank Greenwood, Office Tehnology Principles of Automatisation, Reston Publishing Co, 1984. [www. barnesandnoble.com](http://www.barnesandnoble.com)
12. Peter J. Barry, Paul N. Ellinger, John A. Hopkins, Financial Management in Agriculture, 1995., 5th Edition www. agaccess.com

Adresa autora – Authors' addresses:
Doc. dr. sc. Stjepan Plietić
Prof. dr. sc. Vitomir Grbavac
Agronomski fakultet Sveučilišta u Zagrebu

Mr. sc. Miroslav Žakula
Euclid d.o.o. Vinkovci

Primljeno – Received:
02. 10. 2001.