

PRODUKTIVNI MODEL V ŽIVLJENJSKEM CIKLU GRADBENEGA OBJEKTA

Danijel Rebolj
Univerza v Mariboru, Fakulteta za gradbeništvo
Smetanova 17, 2000 Maribor
Rebolj@uni-mb.si

Izveček

Namen prispevka je prikazati probleme in rešitve pri integraciji informacijskih tokov v življenjskem ciklu gradbenega objekta. Čeprav so mnoge posamezne faze življenjskega cikla podprte z bolj ali manj ustrezno informacijsko tehnologijo, pa ti »otoki avtomatizacije« večinoma niso ustrezno povezani. V prispevku je predstavljen že uveljavljeni koncept rešitve, ki temelji na integriranem produktnem modelu, prikazan pa je tudi primer enostavnega produktnega modela ceste in pozitivni učinki njegove uporabe. Opisane so tudi pomanjkljivosti produktnih modelov ter novi koncept »virtualnega produktnega modela«, ki nekatere opisane pomanjkljivosti odpravlja.

Abstract

The intention of the paper is to present problems and solutions in integrating information flows in the life cycle of a building object. Although many life cycle phases have already been supported by more or less suitable information technology, these »islands of automation« are not suitably integrated. The paper describes an already recognized solution concept, based on the integrated product model, and explains it on a simple product model of a road. Positive effects as well as some deficiencies of product models are shown. The article ends with a description of a new concept of »virtual product model«, overcoming some of the stated deficiencies.



UVOD

Gradbeništvo je v kratki zgodovini računalništva v določenih trenutkih izjemno hitro vključevalo nove informacijske tehnologije v rešitve problemov. Gradbeni inženir Konrad Zuse je leta 1938 celo zgradil prvi binarni elektronsko-mehanski računalnik Z1 (Ceruzzi 1981) tudi zato, da bi lahko z njim reševal statične probleme vse kompleksnejših konstrukcij.

Programski jezik FORTRAN je sredi 50-ih let odprl možnost praktične uporabe računalnikov mnogim inženirjem in v gradbeništvu je nemogoče postalo mogoče, predvsem na področju numeričnih analiz, postopoma pa tudi na drugih področjih (Fenves 1996). Tako so posamezni procesi v gradbeništvu postopoma dobivali računalniško podporo, ti t.i. otoki avtomatizacije pa so se vse bolj širili (Hannus 1998). Žal v realnem okolju veliko počasneje, kot v raziskovalno razvojnih centrih, kjer se je za področje uporabe informacijskih tehnologij v gradbeništvu postopoma oblikovalo ime gradbena informatika (Bauinformatik v nemščini in Construction Information Technology v angleščini, kjer so v uporabi še nekateri drugi izrazi).

Razlogov, da se je gradbena informatika v industriji uveljavlja počasneje kot na nekaterih drugih področjih je več: enkratnost izdelkov, razpršenost proizvodnje,

raznolikost in množičnost vključenih podjetij itd. Mnogi avtorji so te posebnosti analizirali in skušali zastaviti smernice za učinkovitejši razvoj gradbene informatike (Björk, 1999).

V realnem okolju so posamezni procesi v gradbeništvu med seboj povezani, saj se vsi nanašajo na skupen izdelek, gradbeni objekt. Vsak proces ima svojo vlogo in položaj v življenjskem ciklu izdelka, procese pa povezujejo informacijski tokovi, ki temeljijo na bolj ali manj strukturiranih podatkih.

Še najpogosteje se podatki prenašajo v klasični obliki načrtov na papirju, kar onemogoča avtomatizacijo prenosa informacij iz ene faze življenjskega cikla v drugo. V široki uporabi je tudi prenos podatkov v digitalnih oblikah, ki temeljijo predvsem na standardih za vektorski opis grafičnih podatkov (npr. DXF), vendar se pri teh prav tako pojavljajo velike težave. Gradbene konstrukcije so namreč zelo kompleksne, prav tako pa tudi podatkovne strukture, s katerimi jih predstavljamo. Standardi, ki se uporabljajo za prenos podatkov, večinoma niso sposobni predstavljanja kompleksnih podatkovnih struktur v celoti, tako da se tudi digitalni zapisi omejujejo na enostavne risbe in sezname, velik del informacij pa se pri tem izgubi.

Sistematično povezovanje računalniških programov, ki podpirajo posamezne procese v življenjskem ciklu

izdelka in so jedro otkov avtomatizacije, je predmet mnogih raziskovalnih in razvojnih projektov v zadnjih dvajsetih letih. Dober pregled nekaterih projektov podajata Brandon in Betts (Brandon 1995).

Glede na način povezovanja programov lahko metode razdelimo v naslednje skupine:

- Povezovanje različnih programskih paketov s pomočjo bolj ali manj inteligentnih podatkovnih vmesnikov. Primer je metoda »software fixing« (Syal 1991), pri kateri se uporabljajo informacijski interpreterji, ki razpoznavajo določene podatkovne strukture in jih prilagajajo ustreznim programom. Te metode ne omogočajo tekočega pretoka informacij, vključevanje novih programov pa je oteženo, ker je potrebno za vsakega izdelati nov podatkovni vmesnik.
- Uporaba posebnega medija za prenos informacij med programi. Znana je metoda »blackboard« (Yau 1991), ki za komunikacijo uporablja skupno »tablo« in preko nje tekoč pretok informacij. Tega omogoča tudi metoda objektnih lupin »Object Shell« (Rebolj 1993), vendar pri obeh metodah ostaja težava z vključevanjem novih programov.
- Koncept integrirane baze podatkov temelji na skupnem viru podatkov, ki ga uporabljajo vsi vključeni programi. Znanih je več projektov, v katerih je bil uporabljen koncept integrirane podatkovne baze: RATAS (Björk 1998), ATLAS (ATLAS 1992), COMBINE (Augenbroe 1993), COMBI (Ammerman 1994), v zadnjem času pa npr. SPACE in OSCON, ki sta služila kot izhodišče za trenutno verjetno najbolj tehnološko razvito integrirano okolje v gradbeni industriji (Faraj 1999). Med zgodnejše, vendar manj znane sisteme sodi tudi Gradbeniški informacijski

sistem z enotno geometrijsko-konstrukcijsko bazo podatkov (Rebolj 1990). Mnogi avtorji so podali pregledne opise naštetih in tudi drugih projektov in sistemov (Amor 1998, Eastman 1998).

Integrirana baza podatkov velja danes kot najučinkovitejša povezava računalniških programov v življenjskem ciklu gradbenega objekta. Vsebina podatkovne baze zajema celovit opis produkta, zato takšnemu podatkovnemu modelu pravimo tudi produktni model.

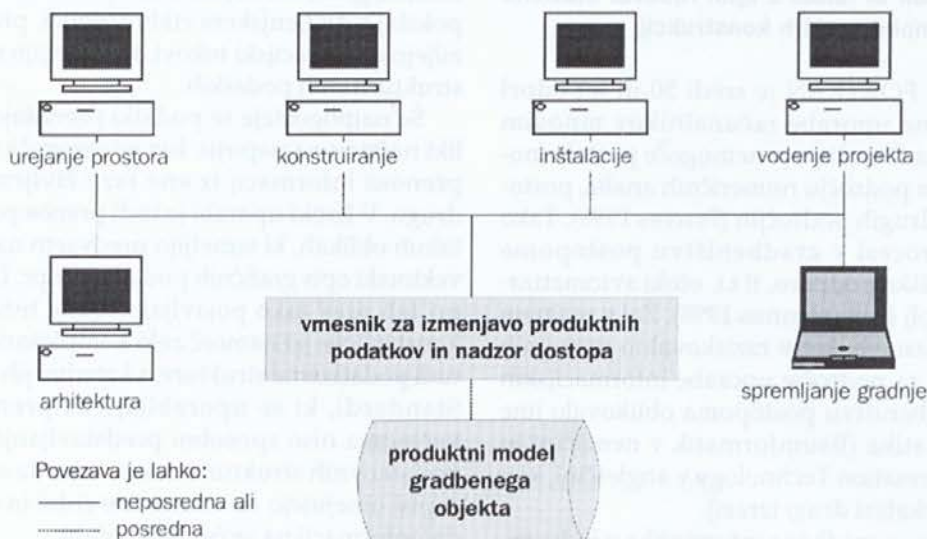
PRODUKTNI IN PROCESNI MODELI V GRADBENIŠTVU

Osnovni namen produktnega modela gradbenega objekta je izboljšati in avtomatizirati prenos podatkov med različnimi aplikacijami, ki so v uporabi v različnih procesih življenjskega cikla objekta (Slika 1).

Koncept integrirane podatkovne baze produkta temelji predvsem na dveh bistvenih predpostavkah:

1. vsi vključeni programi za izmenjavo informacij uporabljajo sintaktično in semantično enak način opisovanja podatkovnih struktur,
2. skupen upravljalni sistem podatkovne baze skrbi za integriteto podatkov in s tem zagotavlja usklajenost informacijskih tokov.

Za zagotavljanje prvega pogoja so informatiki, ki so se ukvarjali z modeliranjem objektov na različnih inženirskih področjih, razvili metode za modeliranje in opisovanje kompleksnih podatkovnih struktur in jih tudi standardizirali. Tako je leta 1994 nastal mednarodni standard za izmenjavo podatkov STEP, STandard for the Exchange of Product model data (ISO 1994), ki omogoča opis poljubnih gradnikov in struktur, ki jih ti



Slika 1. Vloga produktnega modela gradbenega objekta (Center za gradbeno informatiko).

sestavljajo. Standard je bil uspešno uporabljen predvsem pri zelo velikih projektih (Hardwick 1997), vendar še ni doživel široke uporabe, saj ga večina ustreznih programov ne podpira.

Odperto pa je ostalo tudi vprašanje definicije osnovnih gradnikov kompleksnih struktur, ki so ga na posameznih področjih reševali ločeno. Tudi v gradbeništvu, kjer obstajajo posebni aplikacijski protokoli STEP-a za opisovanje elementov gradbenih konstrukcij (npr. AP 225, ki z eksplicitno oblikovno predstavitvijo opisuje gradnike konstrukcij). Ti aplikacijski protokoli se razvijajo počasi in zato ne doživljajo pravega zanimanja. Da bi pospešila standardizacijo osnovnih gradnikov, se je mednarodna skupnost za skladnost delovanja IAI (International Alliance for Interoperability) lotila izdelave svojega standarda IFC, Industry Foundation Classes (Liebich 1999). IFC je zbirka gradnikov za področje gradbeništvu, ki upošteva standard STEP. Po drugi strani je IFC tudi določeno razhajanje od sicer povsem odprtega standarda STEP, saj zapira zbirko gradnikov v meje partnerjev IAI, med katere sodijo predvsem veliki proizvajalci programske opreme.

Seveda je jasno, da morajo biti gradniki do neke ravni enolično definirani, saj sicer ne bi obstajal skupen jezik za izmenjavo informacij. Vprašanje pa je, do katere ravni abstrakcije je poenotenje smiselno. Vprašanje je tudi, ali je takšno mejo sploh mogoče postaviti. Na področju pisav npr. še vedno ni jasno, ali je za zapis informacij primernejša pisava, ki temelji na simbolih (kitajske in japonske pismenke), ali tista, ki temelji na glasovih (črkovne pisave). Katera je manj omejujoča in katera bolj izrazna?

V zadnjem času se je na področju internetne tehnologije pojavil še jezik XML, s katerim je mogoče opisati tudi produktne modele. Ker XML obeta zelo široko uporabo, se postavlja vprašanje, ali ne bi bilo smiselno tudi produktne modele opisovati z njim.

Drugi problem integrirane podatkovne baze je usklajenost podatkov. To je mogoče zagotoviti le, če je znana projektna shema. Projektna shema namreč določa zaporedje izvajanja procesov v življenjskem ciklu objekta, povezave med njimi, odgovornost in pristojnost za določen del podatkov ipd. Takšni shemi pravimo tudi procesni model in ga tudi lahko opišemo z nekaterimi standardnimi tehnikami (npr. EXPRESS, IDEF0 ali UML). Tako je tudi v gradbeni informatiki nesporno dejstvo, da je za uspešno uporabo produktnega modela nujna povezava s procesnim modelom, ki vključuje četrto razsežnost - čas. Integrirane produktno-procesne modele imenujemo tudi 4D modeli (Aalami in Fischer 1998).

Razvoj produktnih in procesnih modelov je v veliki meri usmerjal tudi hkraten razvoj informacijskih tehnologij, predvsem tistih vezanih na internet. To dejstvo ne preseneča, saj velik del težav pri integraciji infor-

macijskih tokov v gradbeništvu izhaja prav iz prostorske razpršenosti procesov. Tehnologija odjemalce-strežnik, ki se je kasneje razvila v 3 plastni model porazdeljene obdelave podatkov, je danes dovolj zrela za praktično uporabo, čeprav je razvoj še vedno v intenzivnem teku. Večina projektov, ki poskuša implementirati koncept porazdeljene uporabe produktnih modelov (npr. Faraj et al. 1999), uporablja tehnologijo CORBA (Common Object Request Broker Architecture), saj je neodvisna od operacijskega okolja računalnikov.

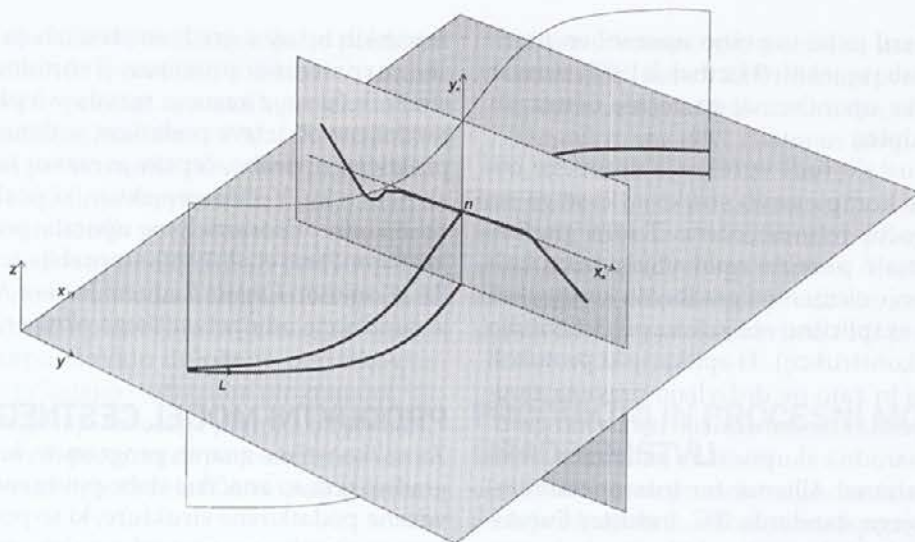
PRODUKTNI MODEL CESTNEGA TELESA

Za veliko večino znanih programov, ki so v uporabi pri gradnji cest, so značilne slabo povezane ali sploh nepovezane podatkovne strukture, ki so povsem podrejene konvencionalnemu postopku načrtovanja cest. Še posebej to velja za nadgradnje različnih univerzalnih CAD-programov, ki vključujejo dodatne specializirane funkcije za načrtovanje cest in so usmerjene predvsem v izdelavo risb - načrtov.

Obstajajo pa tudi programi, ki imajo za osnovo ustrezno celovit model ceste - podatkovne strukture torej, v katerih so med seboj povezani osnovni gradniki kot so os ceste, elementi prečnih profilov in teren. Programov, ki bi vsebovali celovit model ceste, je precej manj in so praviloma zaprti. To pomeni, da svojega modela ne znajo posredovati drugim programom brez znatne izgube informacij, ki je značilna za pretvorbo v široko podprte standardne opise podatkovnih struktur - predvsem geometrijskih (npr. DXF).

Novi standardi na področju izmenjave informacij (STEP, IFC) ponujajo veliko več možnosti za celovitejšo opisovanje modelov gradbenih objektov - tudi cest. Vendar je do njihove učinkovite uporabe dolga pot, na katero je raziskovalna skupina avtorja prispevala stopila v začetku 90-ih let. Velik del tega časa je bil posvečen integraciji računalniško podprtih procesov življenjskih ciklov gradbenih objektov, še posebej cest. Izdelali smo enostaven, odprt produktni model ceste, ki ga je zelo lahko uporabiti. Model cestnega telesa ali MCT, kot smo ga poimenovali, nam služi kot izhodišče za izboljšanje izmenjave podatkov med obstoječimi programi, ki podpirajo različne faze v življenjskem ciklu ceste (Rebolj 1999).

Tudi za tlorisni prikaz osi ceste lahko rečemo, da je model ceste. Vendar tak prikaz predstavlja le izsek iz celotne strukture. O integriranem modelu objekta pa lahko govorimo šele tedaj, ko vključuje vse bistvene komponente objekta in povezave med njimi. Marsikateri CAD-program za ceste vključuje vse potrebne komponente, vendar so povezave med njimi skrite v programu samem in ne v modelu - ali pa jih v računalniku sploh ni in jih ustvari človek v svojem umu. Seveda lahko model ceste definiramo širše ali ožje - odvisno od faze življenjskega cikla ali vidika, s katerega cesto opazujemo. V



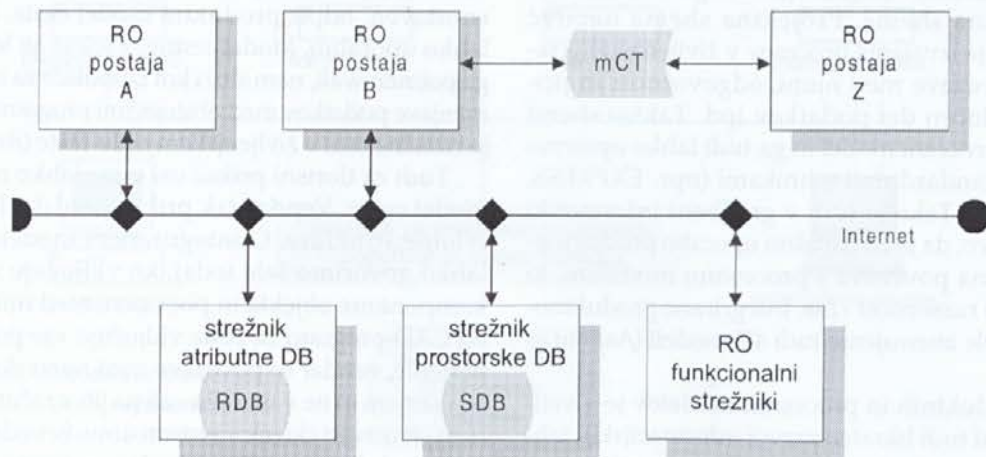
Slika 2. Sinteza projekcij osi in prečnih profilov - osnovnih gradnikov modela ceste (Center za gradbeno informatiko).

fazi priprave zemljišča je lahko bistvena komponenta modela poligon, ki predstavlja obod cestnega telesa, v fazi izgradnje je to tehnologija, vezana na določeno aktivnost v termiskem planu gradnje, za vrednotenje investicije so pomembni ekonomski vplivi, za ugotavljanje vplivov na okolje pa prometni tok.

MCT izhaja iz faze geometrijskega načrtovanja, saj je oblika osnovna lastnost in funkcija ceste. Model je objektno zasnovan in odprt, kar nam omogoča postopno dopolnjevanje z elementi, ki so potrebni v drugih fazah življenjskega cikla. Da bi omogočili največjo možno kompatibilnost z razširjenimi računalniškimi programi, smo ohranili temeljno strukturo modela, ki izhaja iz konvencionalnega postopka načrtovanja cest. Slika 2 še najbolj nazorno prikazuje jedro MCT, v katerega sta med drugim vključeni horizontalna in vertikalna projekcija osi ter opis prečnih profilov.

Zunanja predstavitev MCT, ali kratko mCT (metadatoteka cestnega telesa), je namenjena predvsem izmenjavi podatkov med obstoječimi programi, ki so vključeni v življenjski cikel ceste. Prvi korak v našem pristopu je bila preprosta metadatoteka, za katero je mogoče hitro in enostavno izdelati vmesnike za branje in pisanje ustreznih podatkov v programske pakete vseh vrst in iz njih, ki jih uporabljajo v mnogih različnih birojih in podjetjih, vključenih v cestne projekte. Z vmesnikom mCT so že opremljeni nekateri komercialni programi (npr. program za načrtovanje geometrije ceste Plateia).

Za demonstracijo uporabe MCT in zapolnitev nekaterih avtomatizacijskih lukenj smo izdelali Okolje za podporo življenjskega cikla ceste (ali kratko RO), ki vključuje več funkcionalnih modulov v obliki strežnikov (Slika 3).



Slika 3. Osnovna arhitektura okolja RO (Center za gradbeno informatiko).

Produktni podatki MCT so odjemalcem na voljo v relacijski podatkovni bazi (RDB), okolje pa zajema tudi prostorske podatke, ki so na voljo preko strežnika prostorske podatkovne baze (SDB). Mnogi funkcionalni strežniki namreč vsebujejo funkcije geografskih informacijskih sistemov (GIS).

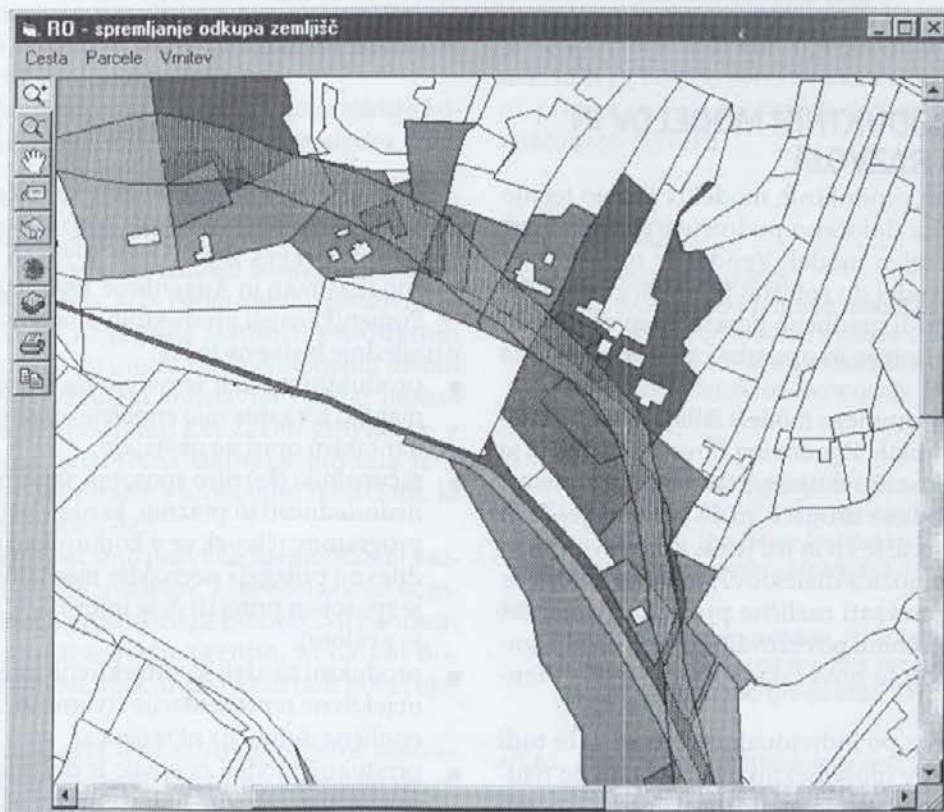
V okolju RO so trenutno na voljo naslednji funkcionalni strežniki, ki vsi uporabljajo del podatkovne strukture, definirane v MCT:

- definicija koridorja (upoštevaje ustrezne geografske podatke poiščemo optimalni geografski koridor za novo cesto),
- spremljanje odkupa zemljišč (generiramo obod cestnega telesa s skrajnimi zunanji točkami prečnih profilov ter z njim prekrijemo parcele v ustrezni geografski temi - Slika 4),
- izračun emisij škodljivih snovi (uporabimo prostorsko predstavitev osi, ki jo generiramo iz projekcij v modelu, dodatno pa moramo navezati ustrezne prometne podatke) in
- hitra 3D vizualizacija (generiramo 3D geometrijski model ter s tem omogočimo hitro vizualno presojo trase; 3D vizualizacijo smo implementirali tudi v obliki samostojnega programa, ki iz mCT izdelava VRML datoteko (virtual reality markup language, Hartman in Wernecke 1996), primerno za objavo na svetovnem spletu - Slika 5).

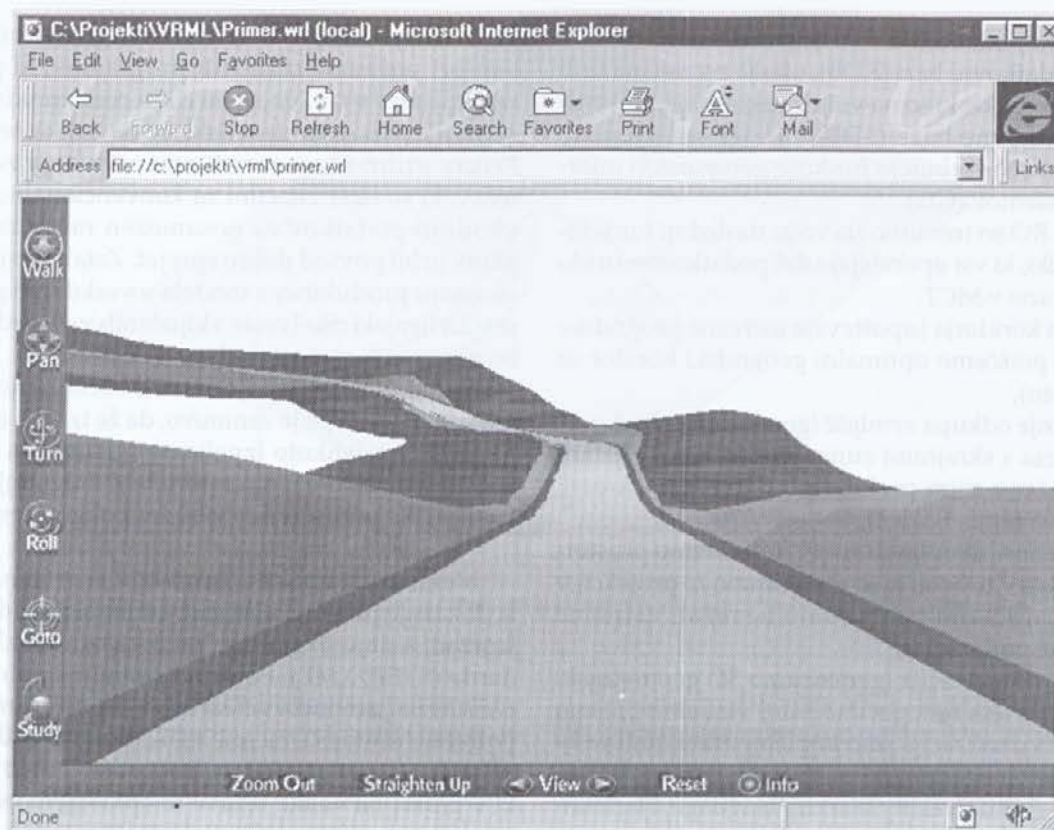
Pri uvajanju okolja RO smo lahko ugotovili, da je problem pri povezavi različnih programov, ki podpirajo različne faze v življenjskem ciklu ceste, predvsem konceptualen in organizacijski in ne toliko tehnološki. Prikaz prihrankov zaradi zmanjševanja zastojev in težav, ki so sicer značilni za konvencionalno pripravo vhodnih podatkov za posamezen računalniški program, je bil povsod dobro sprejet. Zataknilo pa se je pri uvajanju produktnega modela v vsakdanjo prakso, saj je v življenjski cikel ceste vključenih več med seboj slabo povezanih samostojnih podjetij.

Nekaj očitkov smo doživeli zaradi enostavnosti modela. Pri tem pa je zanimivo, da že iz tako enostavnega modela z lahkoto izpeljemo podatke za celo vrsto aplikacij, za katere je sicer potrebno dolgotrajna priprava (npr. 3D predstavitev, obodni poligon ceste, itd.).

Menimo, da visoka učinkovitost enostavnega modela dokazuje pravilno usmeritev, model pa tudi že izboljšujemo, širimo in pri tem preizkušamo različne standarde (STEP, XML). Po drugi strani je implementacija enostavnega modela veliko lažja - upoštevati je namreč potrebno obstoječe računalniške programe. Že pri uvajanju enostavnega modela smo naleteli med uporabniki v praksi na velike težave in sprašujemo se ali ne bi bile te težave v primeru »vse vključujočega«, visoko kompleksnega modela, nepremostljive.



Slika 4. Spremljanje odkupa zemljišč na osnovi prekrivanja parcel z obodom ceste (Center za gradbeno informatiko).



Slika 5. VRML predstavitev modela cestnega telesa MCT(Center za gradbeno informatiko).

OMEJITVE PRODUKTHNIH MODELOV IN NOVE SMERI RAZVOJA

Dosedanji primeri produktnih modelov kažejo težnjo zgraditi enoten, za določeno področje industrije vsevljučujoč univerzalen model. Vendar še nobeden od poizkusov ni privedel do splošno priznanega modela, ki bi ga sprejela tudi gradbena praksa, temveč prej do spoznanja, da definicija in uporaba celovitega modela ni smiselna.

Drug problem enotnega modela izhaja iz potrebe po standardnih gradnikih. Zgodovina človeštva kaže, da je pri komuniciranju edini skupen "standard" raznolikost standardov. Povedano drugače, malo verjetno je, da bi kadarkoli vsi govorili le en in isti jezik. Če pa bi ga, bi se kmalu pojavila množica dialektov. Nekateri avtorji so sicer poskušali povezati različne produktne modele, vendar se je z različnimi povezovalnimi shemami kompleksnost celote le še povečala (Spooner 1997, Pfenigsschmidt 1997)

Želja in potreba po individualnosti pa se kaže tudi v zahtevi subjektov (posameznikov ali družb) po nadzoru nad svojimi podatki, ki so tudi bistveni del »spomina« podjetja (Larson 1998).

Takšne in podobne probleme so že zaznali tudi drugi avtorji ter svoje pomisleke izrazili bodisi med vrsticami (npr. Graves 1998, Amor 1998) ali povsem neposredno (Eastman in Augenbroe 1998, Turk 1999).

Pomanjkljivosti produktnih modelov bi lahko strnili v naslednje bistvene točke:

- produktni modeli temeljijo na jasno definirani semantiki ter zahtevajo enolične opise gradnikov, vendar takšni opisi ne obstajajo,
- računalniki (še) niso sposobni zapolniti semantičnih nedoslednosti in praznin, ki nastopajo pri integraciji programov (človek se v komunikaciji s sočlovekom dnevno prilagaja neenakim mentalnim modelom in je sposoben prirejati dele informacij, ki se ne skladata s celoto),
- produktni modeli so subjektivne interpretacije in ne objektivne reprezentacije stvarnosti, zato učinkovita enolična definicija ni mogoča,
- produktni modeli zajemajo le del podatkov iz procesa gradnje, celovita vključenost v okolje (v prostorskem in socialnem smislu) pa je zanemarjena,

- modeli zaradi svoje kompleksnosti in togih definicij omejujejo kreativno delo,
- prototipni modeli pri uvajanju v prakso doživljajo neuspeh zaradi nezmožnosti upoštevanja bogastva znanja in izkušenj ljudi v realnem okolju,
- produktni modeli so kljub deklarativni odprtosti v realnem okolju togi in težko nadgradljivi,
- v integrirani podatkovni bazi je nadzor subjektov nad lastnimi podatki nezadovoljiv.

Posamezni avtorji (Eastman 1998, Turk 1999) so predlagali tudi nekatere rešitve:

- produktni modeli naj bodo majhni in omejeni na specifična področja; obstoj več modelov na istem področju ni nujno slab (dokazuje le subjektivnost tehničnih modelov),
- izgraditi je potrebno vmesna orodja med aplikacijami in modeli, ki bodo v pomoč človeku pri »navigaciji med otoki avtomatizacije«,
- postopno uvajanje majhnih modelov v industrijo,
- razviti je potrebno bogatejši niz jezikovnih konstruktorov za opisovanje modelov,
- tesneje je potrebno povezati produktni in procesni model,
- preizkusiti je potrebno nove načine povezovanja, ki ne bodo temeljili na integrirani semantiki,
- potrebno je dopustiti koeksistenco strukturiranih informacij in nestrukturiranih podatkov ter njihovo interpretacijo prepustiti človeku,
- programi naj ne omejujejo temveč razširjajo inženirjevo bivanje v svetu (virtualna resničnost, teleprisotnost, multimedia itd.),
- golo izmenjavo informacij je potrebno razširiti s komunikacijsko programsko opremo za podporo sodelovanju.

Na osnovi dobrih in slabih izkušenj pri modeliranju gradbenih produktov smo v raziskovalni skupini Centra za gradbeno informatiko izdelali koncept virtualnega produktnega modela, ki bi lahko ohranil pozitivne in premostil nekatere negativne lastnosti produktnih modelov. Predstavlja ga mreža rahlo sklopljenih delnih modelov, ki jih povezujejo enostavna vendar močna pravila (ta delujejo podobno kot zakon gravitacije v makrokosmosu). Soseščina delnega modela je v logičnem smislu definirana s procesnim modelom, ki določa tudi relacije med deli.

Posebno pozornost smo posvetili harmonizaciji vsebine povezanih delnih modelov, ki predstavljajo komponente virtualnega produktnega modela. Mehanizem temelji na harmonizacijskih agentih, ki delom dopuščajo individualnost, vendar jih hkrati tudi povezujejo v celoto.

Harmonizacijski agenti ne zahtevajo enotne semantike delnih modelov, temveč le enotno definicijo skupnih elementarnih pojmov. Zato lahko delni modeli uporabljajo različne tehnike in standarde za strukturiranje

in reprezentacijo objektov. Agenti pri medsebojni komunikaciji uporabljajo pridobljeno znanje, ki ga shranjujejo v skupnih pojmovnih slovarjih, pri razreševanju problemov pa navezujejo stik tudi s človekom.

Koncept virtualnega produktnega modela ohranja neodvisnost in prilagodljivost delov – obstoječih otokov avtomatizacije (modelov in aplikacij) in enostavnost dela z njimi, hkrati pa ohranja pozitivne lastnosti kompleksnih produktnih modelov. Razlog za takšno prepričanje izhaja iz enostavnosti uporabljenih delov in podobnosti principov virtualnega produktnega modela z naravnimi mehanizmi (osnovnimi zakoni).

ZAKLJUČEK

Koncept produktnega modela je rezultat človekove mentalne aktivnosti in želje po obvladovanju celote do najmanjših možnih podrobnosti. Posebno v gradbeništvu njihov razvoj kaže, da je pri tem človek, kot že nič kolikokrat prej, zanemaril naravne zakone in samega sebe ter se zanašal le na svoje vse bolj kompleksne mentalne konstrukte in jih enačil z objektivno stvarnostjo. V posebnih okoliščinah tako izdelani modeli delujejo, kar so dokazali tudi mnogi raziskovalni in razvojni projekti, ki vztrajno širijo in povezujejo otoke avtomatizacije ter ustvarjajo nove, na novih področjih. Vendar je postalo v zadnjih letih vse bolj očitno, da enoviti kompleksni produktni modeli niso univerzalno uporabni ter da imajo precej pomanjkljivosti. Kar pa ne pomeni, da je neuporaben tudi koncept produktnega modela. Le uskladiti ga bo potrebno z nekaterimi naravnimi zakoni, pri tem pa bolj upoštevati tudi subjektivno vlogo in značilnosti človeka.

REFERENCE

- Aalami F.; Fischer M. 1998:
Joint product and process model elaboration based on construction method models. Zbornik konference CIB W78 The life-cycle of construction IT innovations, The Royal Institute of Technology, Stockholm, Sweden, 1998, 1-11.
- Amor, R.:
A UK survey of integrated project databases. Zbornik konference CIB W78 The life-cycle of construction IT innovations. The Royal Institute of Technology, Stockholm, Sweden, 1998, 67-76.
- Ammerman, E.;
Junge, R.; Katranuschkov, P.; Scherer, R.J.: *Concept of an object-oriented product model for building design. Technische Universität, Dresden, Germany, 1994.*
- ATLAS, Architecture, methodology and tools for computer integrated large scale engineering – ESPRIT project 7280, Technical Annex Part 1, General Project Overview, 1992.

- Augenbroe, G.:
COMBINE, Final Report. Delft University, Delft, The Netherlands, 1993.
- Björk, B.C.:
Basic structure of a proposed building product model. Computer Aided Design. *letnik 21, številka 2*, 1998, 71-78.
- Björk, B. C.:
Information Technology in construction: domain definition and research issues. International Journal of Computer Integrated Design And Construction, *letnik 1, številka 1*, 1999, 3-16.
- Brandon, P.; Betts, M.:
Integrated Construction Information. E & FN Spon, London, 1995.
- Ceruzzi, P. E.:
The Early Computers of Konrad Zuse, 1935 to 1945. Ann. Hist. Comp, *letnik 3, številka 3*, 1981, 241-262.
- Eastman, C.; Augenbroe, F.:
Product modeling strategies for today and the future. Zbornik konference CIB W78 The life-cycle of construction IT innovations. The Royal Institute of Technology, Stockholm, Sweden, 1998, 191-208.
- Faraj, I.; Alshawi, M.; Aouad, G.; Child, T.; Underwood, J.:
Distributed Object Environment: Using International Standards for Data Exchange in the Construction Industry. Computer-Aided Civil and Infrastructure Engineering, Blackwell Publishers, *letnik 14, številka 6*, 1999, 395-405.
- Fenves, S. J.:
The penetration of information technologies into civil and structural engineering design: State-of-the-art and directions towards the future. Information Representation and Delivery in Civil and Structural Engineering Design, Civil-Comp Press. Galashiels, Scotland, 1996, 1-5.
- Graves, G.:
Industry requirements for data standards harmonization. Zbornik konference Global Business Solutions for the new millenium, CD ROM, 1998.
- Hannus, M.:
Islands of Automation in Construction. <http://cic.vtt.fi/hannus/islands.html>, 1998.
- Hardwick, M.; Spooner, D. L.; Rando, T.; Morris, K.C.:
Data protocols for the industrial virtual enterprise. IEEE internet computing, *letnik 1, številka 1*, 1997, 20-29.
- Hartman, J.; Wernecke, J.:
The VRML 2.0 Handbook. Addison-Wesley, Reading, Mass. 1996
- ISO: Industrial automation systems and integration – Product data representation and exchanges, ISO 10303. ISO, Geneva, Switzerland, 1994.
- Larson, M.:
AF integrated digital environment. Zbornik konference Global Business Solutions for the new millenium, CD ROM, 1998.
- Liebich, T.; Wix, J.:
Highlights of the development process of industry foundation classes. Zbornik 8th International conference on Durability of Building Materials and Components, NRC Research Press, Vancouver, Canada, 1999, 2758-2775.
- Pfennigsschmidt, S.; Kolbe, P.; Pahl, P. J.:
Integration von Datenmodellen. Zbornik konference IKM, Weimar, CD-ROM, 1997.
- Rebolj, D.:
Graphic Modelling of Superstructures. Automatika, *letnik 31, številka 1-2*, 1990, 147-156.
- Rebolj, D.:
Computerunterstützter integrierter Straßenentwurf in einer objekt-orientierten Umgebung. Verlag für die Technische Universität Graz, 1993.
- Rebolj, D.:
Integration of computer supported processes in road life cycle. Journal of transportation engineering, ASCE, *letnik 125, številka 1*, 1999, 39-45.
- Spooner, D. L.; Hardwick, M.:
Using views for product data exchange. IEEE Computer Graphics and Applications, *letnik 17, številka 5*, 1997, 58-65.
- Syal, M. G.; Parfitt, M. K.; Willenbrock, J. H.:
Computer integrated design/drafting, cost estimating, and construction scheduling. Housing Research Center Series Report No. 11, The Pennsylvania State University, Dept. of Civil Eng. 1991.
- Turk, Ž.:
Constraints of product modelling approach in building. Zbornik 8th International conference on Durability of Building Materials and Components, NRC Research Press, Vancouver, Canada, 1999, 2776-2787.
- Yau, N. J.; Melin, J. W.; Garrett, J. H.; Kim, S.:
An environment for integrating building design, construction scheduling, and cost estimating. Zbornik ASCE Seventh Conference on Computing in Civil Engineering and Symposium on Databases, Washington, D.C., 1991.

Danijel Rebolj je diplomiral na področju gradbeništva in magistriral na področju računalništva in informatike. Doktorat tehniških znanosti je dosegel na Tehniški univerzi v Gradcu s področja gradbene informatike, s katero se ukvarja na znanstvenem in izobraževalnem področju. Od leta 1995 je na Fakulteti za gradbeništvo predstojnik Centra za gradbeno informatiko, od leta 1999 pa tudi prodekan za izobraževalno dejavnost.