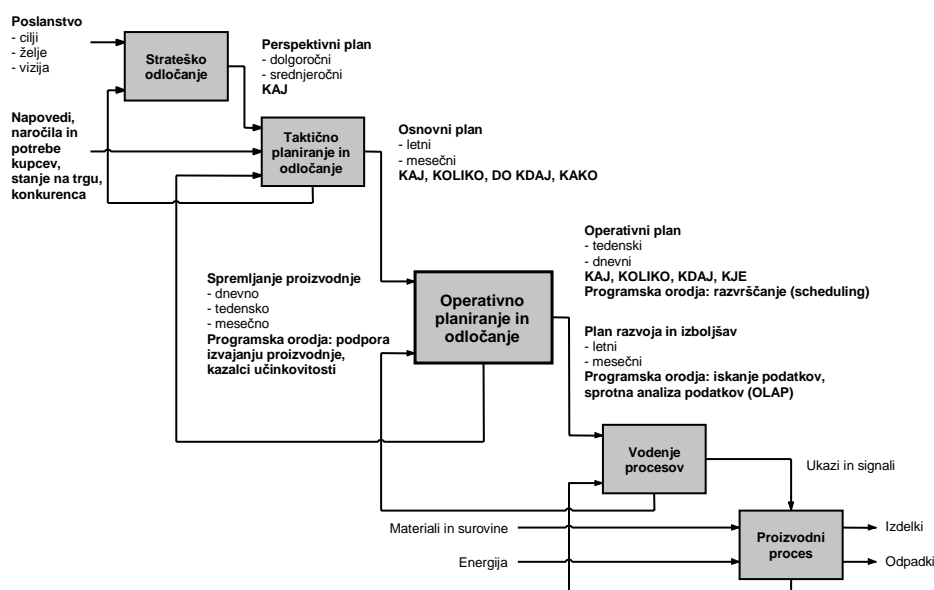


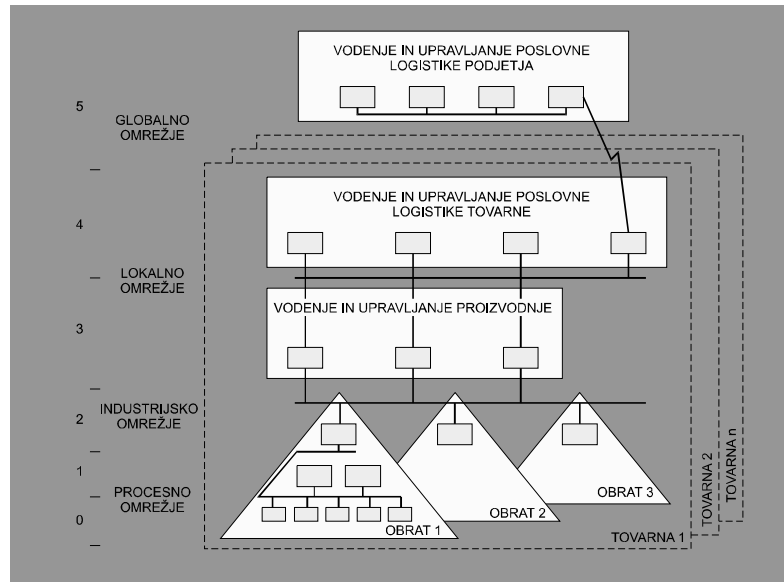
6. Integrirano vodenje proizvodnje

- Computer Integrated Manufacturing (CIM)
 - v ospredju zagotovitev pretoka podatkov med različnimi deli proizvodne organizacije
- Obsega celovito vodenje proizvodne organizacije
 - od najvišjih nivojev - upravljanje podjetja, razvoj izdelkov, planiranje proizvodnje
 - do najnižjih nivojev - regulacija, sekvenčno vodenje
- Bistvena sestavina vodenja - informacija
- Informacije morajo biti
 - dostopne
 - ustrezno strukturirane in obdelane

Integrirano vodenje proizvodnje /2



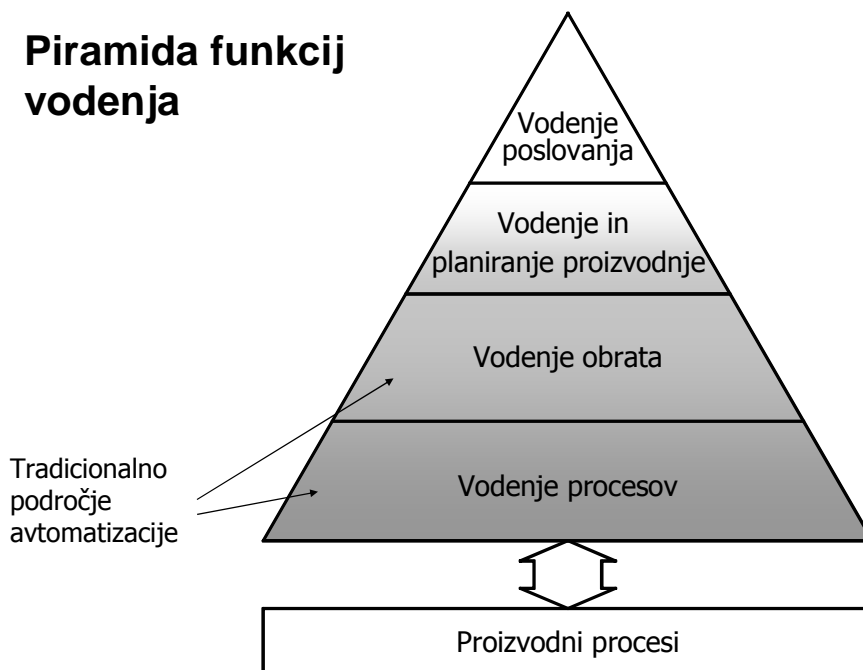
Integrirano vodenje proizvodnje /3



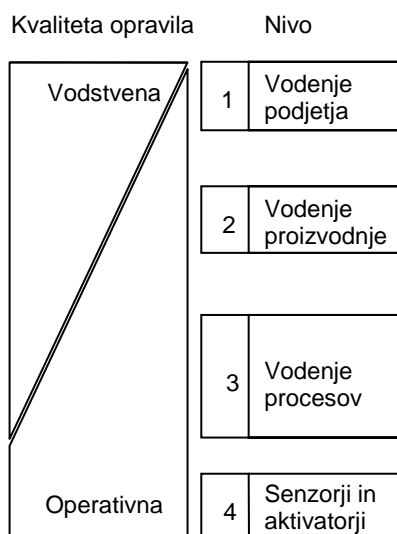
Integrirano vodenje proizvodnje /4

- Uspešna integracija mora reševati tri ključne naloge:
 - zagotavljanje pravih informacij
 - ob pravem času
 - na pravem mestu
- Integracija je trajen proces
 - spreminjajo se notranje razmere v proizvodnji
 - spreminja se (poslovno) okolje
 - potrebno je stalno prilagajanje
- Težavna izvedba
 - podatkovna povezava še ne zagotavlja potrebnega pretoka informacij
 - večji uspeh v kosovni proizvodnji kot v procesni industriji

Piramida funkcij vodenja

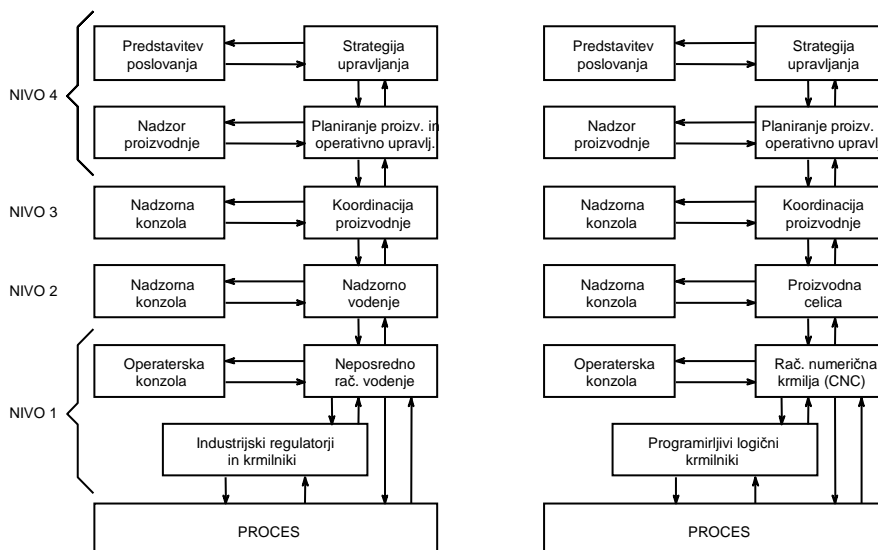


6.1 Večnivojsko vodenje



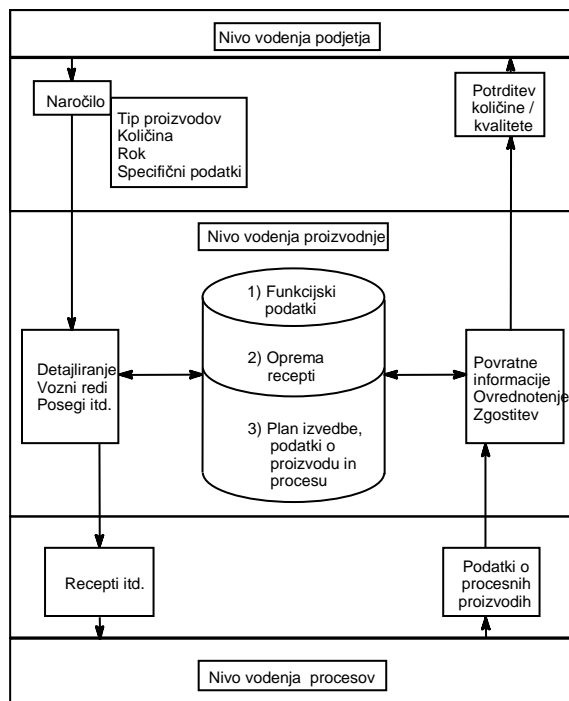
- Nalogo vodenja razdelimo na več hierarhično urejenih nivojev
 - upravljanje oz. vodenje podjetja
 - vodenje proizvodnje
 - vodenje procesov
- Možna še podrobnejša delitev, npr. vodenje procesov obsega
 - krmilno-regulacijski nivo
 - nadzorni nivo

Večnivojsko vodenje /2



Povezava med nivoji

- Preko definiranih vmesnikov
- Vsak nivo
 - uporablja uslužnostne funkcije nižjih nivojev
 - posreduje zgoščene informacije višjim nivojem



Bistvene sestavine

- Lokalna komunikacijska omrežja
 - omogočajo pretok informacij med računalniškimi podsistemi
 - hrbtenica integriranega sistema vodenja
- Podatkovne zbirke
 - dostopnost podatkov
 - organiziranost podatkov
 - prava moč podatkovnih zbirk se pokaže pri obdelavah ogromne količine podatkov - takšen primer pa je ravno vodenje podjetja
- Programska orodja (MES, MRP, MRP II, ERP)

6.2 Uvod v podatkovne zbirke

- angl. *data base* oz. *database*
- Podatkovna osnova (baza)
 - omogoča izvajanje različnih uporabniških programov, ki morajo od nekod črpati podatke za svoje izvajanje
 - opis okolja s pomočjo podatkov
 - temelj, ki omogoča komunikacijo/interakcijo z okoljem
- Podatki so organizirani na način, ki je
 - čim bolj neodvisen od programov, ki jih uporabljajo
 - čim bližje človeškemu razumevanju

6.2.1 Podatek in informacija

- Podatek je na zgoščen način zapisano dejstvo
- Različne definicije
 - poljubna predstavitev s pomočjo simbolov ali analognih veličin, ki ji je pripisan, ali se ji lahko pripiše nek **pomen**
 - predstavitev dejstva, koncepta ali instrukcije na **formaliziran** način, ki je primeren za komunikacijo, interpretacijo ali obdelavo s strani človeka ali stroja
 - dejstva, predstavljena z vrednostmi (številke, znaki, simboli), ki imajo pomen v določenem **kontekstu**

Podatek in informacija /2

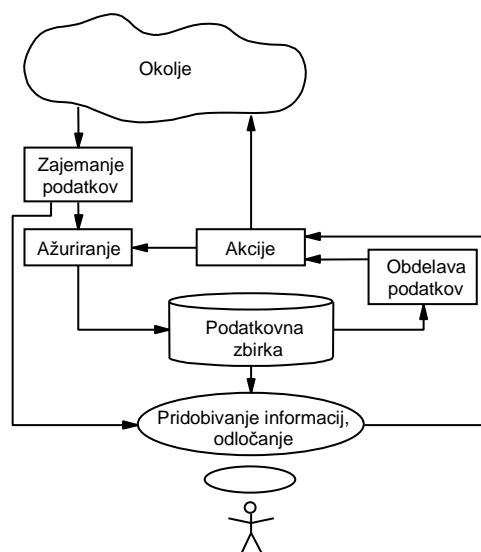
- Informacija
 - pomen, ki ga človek pripiše podatkom s pomočjo znanih konvencij, uporabljenih pri njihovi predstavitvi
 - ovrednoteni podatki v specifični situaciji
 - novo spoznanje, ki ga človek doda svojemu poznavanju sveta
- Zveza med podatki in informacijo
 - podatki niso informacija in ne vsebujejo informacije
 - podatki lahko posredujejo informacijo
 - če je podatkov preveč, da bi jih v ustreznem času interpretirali, ne posredujejo nobene informacije

6.2.2 Sistem za obdelavo podatkov

- Štiri osnovne komponente
 - človek
 - program
 - podatki
 - računalnik
- Pri prvih tovrstnih sistemih v ospredju računalnik, kasneje program, danes podatki
 - podatkovna revolucija
 - strukturiranje, različne predstavitve za različne uporabe
 - shranjevanje podatkov o podatkih - bliže človeku
 - neodvisnost podatkov od strojne in programske opreme
 - središče obdelave je **podatkovna zbirka**

Podatkovna zbirka

- Model (slika) okolja, ki služi kot osnova za sprejemanje odločitev in izvajanje akcij
- Podpira interakcije med človekom (organizacijo) in okoljem



Vrste podatkovnih zbirk

- Podatkovne zbirke imajo različne namene, zato ločimo:
 - referenčne sisteme,
 - sisteme za obdelavo besedil,
 - bibliografske sisteme,
 - specializirane podatkovne zbirke,
 - zbirke znanja in objektno orientirane zbirke.
- zbirka je lahko:
 - centralizirana - en računalnik in en upravitelj,
 - porazdeljena - več lokacij, lokalni upravitelji.

6.2.3 Organizacija podatkovne zbirke

- Podatkovna zbirka v ožjem smislu:
 - podatki, ki jih le-ta vsebuje.
- Podatkovna zbirka v širšem smislu je sistem, ki ga sestavljajo:
 - podatki,
 - uporabniki in uporabniški programi,
 - upravitelj podatkovne zbirke,
 - sistem za upravljanje podatkovne zbirke.
- Podatkovni del zbirke
 - fizična podatkovna zbirka - vrednosti elementov
 - metapodatkovna zbirka - opisi fizičnih podatkov

Organizacija podatkovne zbirke /2

- Obe zbirki sta shranjeni v zunanjem pomnilniku v obliki fizičnih datotek v datotečnem sistemu, ki ga podpira operacijski sistem.
- Metapodatkovna zbirka ima tri vrste opisov fizičnih podatkov (glede na stopnjo abstrakcije):
 - notranja shema – pove, **kako so podatki shranjeni**
 - konceptualna shema - podaja način modeliranja okolja (tipe podatkov, medsebojne relacije, omejitve), pove, **kateri podatki so v zbirki**
 - zunanja shema - podaja način uporabe podatkov in predstavlja **uporabniku lasten pogled**

Organizacija podatkovne zbirke /3

- Fizična podatkovna zbirka ima štiri nivoje:
 - nivo operacijskega sistema: zbirka **fizičnih datotek**,
 - notranji nivo: zbirka fizičnih datotek se prikaže kot zbirka **logičnih zapisov** različnih tipov in njihovih medsebojnih povezav,
 - konceptualni nivo: zbirke logičnih zapisov (in njihovih povezav) se s pomočjo opisov v konceptualni shemi kažejo kot **imena, lastnosti in povezave entitet**,
 - zunanji nivo: konceptualna zbirka se s pomočjo opisov v zunanji shemi prikaže kot **uporabnikov model okolja**.

Organizacija podatkovne zbirke /4

- Entiteta
 - tisti najmanjši del, ki ga lahko ločimo ali želimo ločiti od drugih delov podobe sveta (objekt zanimanja)
- Posamezne vrste shem opišemo s pomočjo **podatkovnega modela**
 - strukturni in opisni mehanizem
 - jezik za opis podatkov
 - sestavni del podatkovnega modela so tudi operatorji, ki tvorijo jezik za manipulacijo s podatki (povpraševalni jezik)

Podatkovni modeli

- Podatkovni model sestavljajo tri komponente:
 - podatkovna struktura,
 - operacije nad podatkovno strukturo,
 - integritetne omejitve, ki so lastne podatkovni strukturi.
- Podatkovne modele delimo v:
 - logične, ki opisujejo zunanjo in konceptualno shemo,
 - fizične, ki opisujejo notranjo shemo.

Podatkovni modeli /2

- Logične podatkovne modele delimo naprej v:
 - površinske, ki temelje na datotečnih zapisih,
 - globinske, ki modelirajo podobo sveta s stališča podatkov, shranjenih v fizični podatkovni zbirki.
- Površinski podatkovni model je lahko:
 - relacijski - temelji na relacijah,
 - mrežni - temelji na grafih,
 - hierarhični - temelji na drevesih,
 - objektno orientirani model (po strukturi podoben mrežnemu modelu).

6.2.4 Relacijski podatkovni model

- Najbolj razširjen v dostopni programski opremi za operiranje s podatkovnimi zbirkami
- Dobre lastnosti
 - sloni na matematičnih strukturah - relacijah
 - ne vsebuje elementov fizičnega shranjevanja podatkov - fizična podatkovna neodvisnost
 - relacije so predstavljive s tabelami - dobro razumljive
- Relacijska podatkovna zbirka - množica tabel
 - čelna vrstica - **relacijska shema** - pojasnjuje pomen posameznih stolpcev
 - podatkovne vrstice - relacija

Osnovni elementi relacijskega podatkovnega modela

- Domena
 - poenostavljeno - zaloga vrednosti stolpca v tabeli
- Relacija
 - poenostavljeno - množica vrstic (zapisov) v tabeli
- Atribut
 - preslikava med vrsticami tabele (objekti relacije) in posamezno domeno - lastnost objekta
- Relacijska shema
 - pripada vsaki relaciji
 - sestavlja jo ime relacije ter seznam atributov in pripadajočih domen: **R(A1: D1, A2: D2, ..., An: Dn)**
 - predstavlja zgradbo tabele

Osnovni elementi relacijskega podatkovnega modela /2

- Ključ relacije
 - elementi, s katerimi je možno identificirati posamezni zapis
 - eden ali več elementov, navedenih v relacijski shemi
 - ključev je lahko več, eden se izbere kot primarni ključ
- Primer:

Zaloga

Koda_izdelka	Naziv	Masa	Kosov
1212987500	kotnik	172	67
2507992505	vijak	135	32
0304988505	vijak	172	63
0105986500	profil	185	67

Domeni: nizi znakov, cela števila
Atributi: Koda_izdelka, Naziv, Masa, Kosov

Relacijska shema:
Zaloga(Koda_izdelka: cel. št.,
Naziv: niz znakov, Masa:
cel. št., Kosov: cel. št.)

Ključ: Koda_izdelka

Povezave med tabelami

- Uporabnik lahko dostopa do posameznih vrstic v tabeli
- Običajno je tabel več in so med seboj povezane preko njihovih relacijskih shem
- V takih primerih je smiselno definirati navidezne tabele
 - zajemajo določeno informacijo iz ene ali več med seboj povezanih tabel
 - uporabniku se kažejo kot običajne tabele
 - v resnici obstaja le preslikava med »primarnimi« tabelami in navideznimi tabelami

Povezave med tabelami /2

- Vrste povezav
 - Ena-proti-ena povezava (angl. *one-to-one relationship*)
 - najbolj enostavna povezava med tabelami
 - vsaki relaciji v eni tabeli pripada natanko ena vrstica v drugi tabeli
 - enostavno »zlepimo« stolpce prve in druge tabele glede na vrednost enega izmed stolpcev
 - isto informacijo bi lahko predstavili z eno večjo tabelo
 - smisel takšne povezave je v pohitritvi dela, če je zbirka manjša, in različnih pravicah posameznih uporabnikov, ki do zbirke dostopajo

Povezave med tabelami /3

- Ena-proti-mnogo povezava (angl. *one-to-many relationship*)
 - najpogostejša povezava med tabelami
 - ena od vrstic v prvi tabeli je povezana z večimi vrsticami v drugi tabeli, medtem ko ima vsaka vrstica v drugi tabeli le eno pripadajočo vrstico v prvi tabeli
 - vrstice so običajno povezane preko primarnega ključa v prvi tabeli in enega izmed stolpcev v drugi tabeli (ki pa seveda ni ključ, saj lahko obstaja več zapisov, ki imajo enako vrednost v tem stolpcu, in se na ta način sklicujejo na en zapis v prvi tabeli).
 - s formalnega stališča to pomeni, da so relacije v prvi tabeli **funkcionalno odvisne** od relacij v drugi tabeli.

Povezave med tabelami /4

- Mnogo-proti-ena povezava (*many-to-one relationship*)
 - je nasprotje povezave ena- proti mnogo
 - če na povezavo gledamo z druge strani in vidimo povezavo ena proti mnogo, pravimo, da je povezava reflektivna. V večini programskih paketov, so podprte le reflektivne preslikave, zato lahko vse povezave opišemo kot ena proti mnogo preslikave.
- Mnogo-proti-mnogo pov. (*many-to-many relationship*)
 - ni je mogoče predstaviti kot enostavne povezave med dvema stolpcema v dveh tabelah
 - vsaki relaciji iz prve tabele pripada več relacij v drugi tabeli in obratno. Problem rešimo z dodatkom nove vmesne tabele, ki je z obema osnovnima tabelama povezana s povezavama mnogo proti ena.

Načrtovanje relacijske podatkovne zbirke

- Bistveno vprašanje: Kako grupirati attribute v ustrezne relacijske sheme?
 - da ne bo težav pri ažuriranju relacij
 - da pri dopolnjevanju podatkovne zbirke z novimi atributi ne bo treba bistveno spreminjati obstoječih relacijskih shem
 - s tem ogrozili obstoječo rabo podatkovne zbirke
 - potrebovali bi nove uporabniške programe (programi so prirejeni obstoječi strukturi zbirke)

Težave pri ažuriranju

- Pokažejo se pri spreminjanju vsebine relacij (tabel), če te niso pravilno zasnovane
- Dodajanje:
 - če se pri dodajanju enega podatka v takšno tabelo zmotimo, lahko postane tabela nekonsistentna - to lahko »pokvari« tudi že obstoječe podatke
- Spreminjanje:
 - pri slabo zasnovani zbirki je isti podatek treba spremeniti večkrat
- Brisanje:
 - z brisanjem ene vstice zberemo več podatkov

6.2.5 Povpraševalni jezik SQL

- Dostop do vsebine tabel - s pomočjo povpraševalnih jezikov
 - lahko temeljijo na relacijski algebri (postopkovni jeziki)
 - ali na relacijskem računu (nepostopkovni jeziki)
- Slednji so bolj razširjeni predvsem po zaslugi SQL
 - angl. *Structured Query Language* - strukturirani povpraševalni jezik
 - standardiziran s strani ANSI in ISO
- SQL je več kot le povpraševalni jezik
 - tudi ukazi za deklariranje, kreiranje in zaščito

Povpraševalni jezik SQL /2

- Osnovna podatkovna struktura - tabela
- Deklaracija tabele
 - enakovredna relacijski shemi
 - obsega:
 - ime tabele,
 - seznam imen podatkovnih elementov skupaj z njihovimi osnovnimi tipi (domenami)
 - integritetne omejitve (ali je lahko nek element v vrstici brez vrednosti, ali morajo biti vrstice med seboj različne ...)
- Navidezne tabele v jeziku SQL imenujemo pogledi (angl. *query*)

Ključne besede SQL

- Po izvoru besede angleškega jezika
- Najpogostejše so besede:
 - ALIAS, ALL, AS, ASC, BETWEEN, BY, DESC, DISTINCT, FROM, GROUP, IN, INNER, INTO, IS, JOIN, LIKE, ON, ORDER, SELECT, WHERE
 - funkcije AVG, MAX, MIN, SUM
 - in logični operatorji AND, OR, NOT
- Obstaja še več drugih ključnih besed v jeziku ANSI SQL, ki je standardiziran
- Proizvajalci programske opreme so dodali v svoje produkte nove besede - obstaja več narečij jezika SQL

Stavek SELECT

- Najpomembnejša beseda v jeziku SQL
- Z njo sestavljamo poizvedbe.
- Sintaktična uporaba besede SELECT:

```
SELECT [ALL|DISTINCT] lista_stolpcev
      FROM imena_tabel
      [WHERE pogoj]
      [ORDER BY ime_stolpca [ASC|DESC]]
```
- Struktura stavka je zelo podobna naravni poizvedbi v angleškem jeziku

Pomen stavka SELECT

- Iz tabel `imena_tabel` izberemo stolpce `lista_stolpcev`, ki bodo stolpci v ciljni navidezni tabeli.
- **ALL** - v ciljni tabeli so vsebovane vse vrstice, ki izpolnjujejo pogoj - tudi, če je več ciljnih vrstic enakih
- **DISTINCT** - vrstice, ki so enake se izločijo
- **pogoj** - omogoča, da izberemo vrstice, ki jih hočemo imeti v ciljni tabeli (npr. `stolpec1>20`); pri sestavljanju pogojev lahko uporabljamo logične operatorje **AND**, **OR**, **NOT**.
- Z **ORDER BY** povemo, kateri stolpec se izbere kot osnova za razvrščanje vrstic v tabeli.
- Elemente v stolpcu lahko razvrščamo od najmanjšega proti največjemu (**ASC**) ali obratno (**DESC**).

Razširjena oblika stavka SELECT

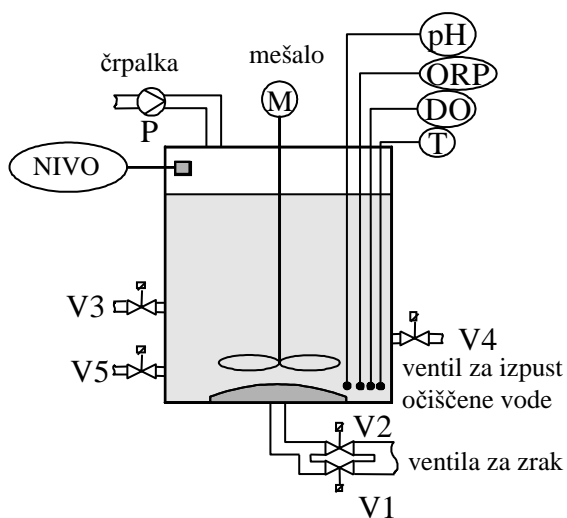
- Omogoča uporabo funkcij nad stolpci v tabeli
- Sintaktična uporaba razširjenega stavka SELECT:

```
SELECT [ALL|DISTINCT]
      funkcija(ime_stolpca) AS privzeto_ime
      [, lista_stolpcev]
FROM imena_tabel
[WHERE pogoj]
GROUP BY pogoj_grupiranja
      [HAVING funkcijski_pogoj]
[ORDER BY ime_stolpca [ASC|DESC]]
```

Pomen razširjenega stavka SELECT

- Iz tabel `imena_tabel` izberemo stolpce `lista_stolpcev`, ki bodo stolpci v ciljni navidezni tabeli, tem lahko dodamo še rezultat funkcije nad skupino zapisov v stolpcu, npr. `AVG()`, `MAX()`, `MIN()`, `SUM()`.
- `AS privzeto_ime` - definira ime navideznega stolpca, v katerem se izpiše rezultat funkcije.
- `GROUP` - definira množico zapisov v stolpcu, nad katero se izvaja funkcija.
- `HAVING` - omogoča, da izberemo vrstice, ki jih hočemo imeti v ciljni tabeli glede na rezultat funkcije (npr. `MAX(stolpec2) > 20`).

Primer: meritve signalov na šaržnem reaktorju



- Šaržno čiščenje odpadne vode
- Čiščenje poteka v več fazah
- Meritve
 - nivo
 - PH
 - redoks potencial
 - raztopljeni kisik
 - temperatura
- Vse meritve se shranjujejo v zbirko

6.2.6 Uporaba in upravljanje podatkovne zbirke

- Uporabniki podatkovne zbirke so lahko:
 - posredni,
 - neposredni (komunicirajo s podatkovno zbirko neposredno preko terminalov)
- Obstaja več vrst neposrednih uporabnikov:
 - upravitelj podatkovne zbirke,
 - sistemski programerji,
 - aplikacijski programerji,
 - uporabniki povpraševalnega jezika,
 - menijsko vodeni uporabniki,
 - parametrični uporabniki (dostopajo do zbirke s pomočjo programov, ki so napisani v višjih programskih jezikih).

Upravljanje podatkovne zbirke

- S tehničnega vidika
 - podatkovna zbirka je zbirka med seboj pomensko povezanih podatkov, ki so shranjeni v računalniškem sistemu, dostop do njih je centraliziran in omogočen s sistemom za upravljanje podatkovne zbirke
- Upravljanje obsega
 - zagotavljanje razpoložljivosti podatkov,
 - nadzor uporabe podatkov, to se pravi skrb za
 - celovitost podatkov – obnavljanje zbirke, nadzor nad sočasnim dostopom, preverjanje vhodnih podatkov
 - uporabo podatkov v skladu z njihovim namenom – up. pravice
 - uporabnost podatkov v prihodnosti – večnivojska zasnova

Upravljanje podatkovne zbirke

- Sistem za upravljanje podatkovne zbirke
 - programska oprema, ki opravlja nalogo posrednika med uporabniki in podatkovno zbirko (MS Access, SQL Server, Oracle ...)
- Funkcije:
 - nadzorne, ki obsegajo zaščito celovitosti in performančni nadzor,
 - dostopne, ki jih lahko naprej delimo na:
 - gradnjo, to je kreiranje shem, kreiranje fizične podatkovne zbirke in reorganizacijo podatkovne zbirke,
 - uporabo, to je zajemanje podatkov iz podatkovne zbirke, ažuriranje podatkovne zbirke in podporo pri razvoju podatkovne zbirke.

Moderni koncepti relacijskih podatkovnih zbirk

- Podatkovni strežnik
 - računalniški sistem, ki je namenjen zgolj upravljanju podatkovne zbirke
- Predvsem se srečujemo z dvema konceptoma
 - koncept odjemalec/strežnik
funkcije ločimo na dve skupini: strežne funkcije in funkcije odjemalca
 - porazdeljena podatkovna zbirka
porazdelitev zbirke na več podatkovnih strežnikov

Koncept odjemalec/strežnik

- Vključuje dva procesa, ki tečeta vsak na svojem ali pa na istem računalniku
 - oba procesa sta podprta s sistemskimi servisi operacijskega sistema in komunikacijske programske opreme
 - odjemalčev proces pošilja strežnikovemu procesu zahteve po določenih storitvah
 - strežnik se odziva s poročili o uspehu
 - tipično nudi svoje usluge večjemu številu odjemalcev.
 - strežnik je lahko en sam, ali pa en primarni (master) strežnik, ki deli naloge sekundarnim (slave) strežnikom.

Koncept porazdeljene podatkovne zbirke

- Zbirka je logično povezana, a fizično porazdeljena po različnih podatkovnih strežnikih.
- Prednosti porazdeljene podatkovne zbirke so:
 - aplikacije so po naravi porazdeljene in je taka arhitektura samoumevna,
 - večja zanesljivost in dostopnost podatkov ob vsakem trenutku,
 - paralelno delovanje procesov, kar povečuje hitrost.
- Glavna slabost porazdeljenih podatkovnih zbirk je njihova kompleksnost.

6.3 Programska oprema na višjih nivojih vodenja

- Narava poslovnih funkcij je zelo drugačna od funkcij vodenja procesov
 - podatki v obliki »transakcij« namesto vzorčenih signalov iz procesa
 - drugačen način obdelave podatkov
 - programska orodja so se razvijala ločeno od programskih orodij za vodenje
- Upravljanje z viri podjetja
 - programska orodja ERP - Enterprise Resource Planning
 - vključujejo tako proizvodne vire kot ostale (vzdrževanje, kadri ipd.)
 - poslovni plan – kaj, koliko in do kdaj mora biti izdelano

Programska oprema na višjih nivojih vodenja /2

- Planiranje proizvodnje oz. upravljanje z viri proizvodnje (MRP II)
 - integrirano planiranje materiala, opreme in ljudi v podjetju z namenom izpolnitve poslovnega plana; integracija zahteva da se iste informacije (napoved prodaje, potrebe po materialu, dejanska naročila) uporabljajo po vsem podjetju; del tega je vodenje proizvodnje
- Planiranje materialnih potreb (MRP)
 - določanje potreb po materialu na podlagi kosovnic, receptov ipd., predvidenega časa dobave ali izdelave, količine naročil, varnostnih zalog z namenom planiranja količine in časa nabave ali proizvodnje; odvisno je od proizvodnega plana

6.3.1 Sistemi za podporo proizvodnje (MES)

- Novejša standardizirana programska orodja, ki delno nadomeščajo koncept CIM
 - sistemi za podporo izvajanja proizvodnje
 - izvedba planiranih aktivnosti
- MES – Manufacturing Execution Systems
 - sistemsko inženirsko programsko orodje za načrtovanje in izvedbo poslovnih procesov in tokov podatkov v tovarni
 - procesne enote in tokovi podatkov so predstavljeni z grafičnimi elementi, katerim lahko priredimo dele specifikacij in s tem dokumentiramo zahteve glede delovanja sistema
 - na nižjem nivoju je možno definirati podrobnosti poslovnega procesa v smislu dogodkov, akcij in prenosov podatkov, ki se navezujejo na prej določene zahteve

MES /2

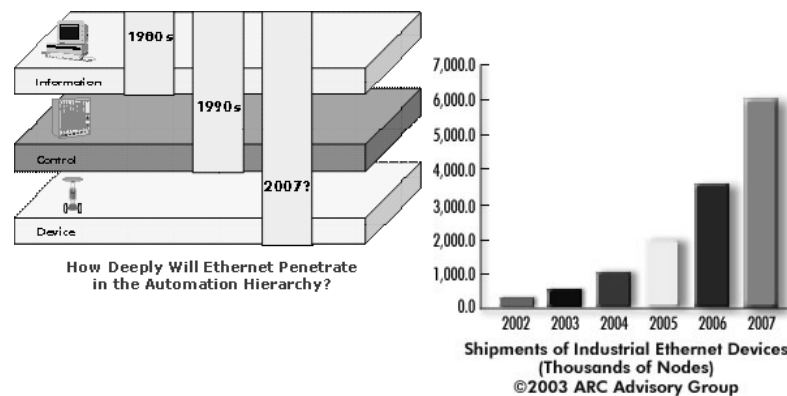
- MES – Manufacturing Execution Systems
 - pravila za prenos določene količine produkta v enoto so definirana kot diagram poteka, ki vključuje interakcije s sistemom vodenja, uporabniki, poslovnimi sistemi (npr. ERP).
 - možno je modelirati alternativne poti, postopke v primeru napak ipd.
 - poslovna logika in logika vodenja sta shranjeni na enem mestu, možno je tudi spremljati delovanje proizvodnega sistema
 - značilno je stalno dograjevanje in spreminjanje funkcij glede na spremembe v proizvodnji

6.4 Smeri razvoja

- Standardizacija
 - funkcijskih blokov (IEC 61499)
 - programskih jezikov (IEC 61131-3)
 - komunikacijskih omrežij (IEC 61158)
- Izvedba
 - vključevanje logičnih funkcij v industrijske regulatorje in vključevanje regulacijskih funkcij v PLK
 - PAC – Programmable Automation Controller
 - razmah področnih vodil (fieldbus)
 - velika ponudba naprav z vgrajenimi komunikacijskimi vmesniki
 - vrstne priključne sponke s komunikacijskim vmesnikom

Komunikacijska omrežja

- prodor omrežij tipa Ethernet



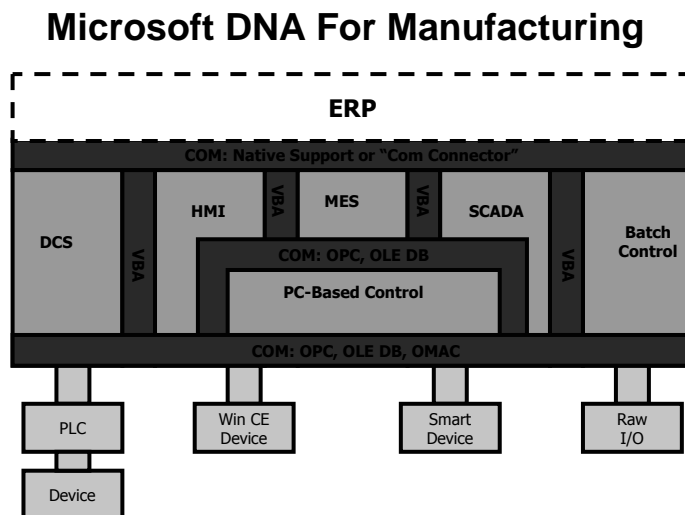
Računalniki in programska oprema

- prodor PC: programirna/konfigurirna naprava, operatorski vmesnik oz. konzola, posebna regulatorska/krmilna enota
 - industrijski računalniki,
 - računalniki za vgradnjo (emdedded systems)
 - PC moduli kot gradnik PLK,
 - PLK CPE moduli za vgradnjo v PC,
 - PC, povezan na področno vodilo- >Soft PLC;
- zaton klasičnih programskih orodij SCADA
 - del nadzornih funkcij prevzemajo PLK in grafični operaterski paneli
 - nadzorni sistemi vključujejo funkcije višjih nivojev vodenja, npr. MES

Načrtovanje

- Pristop k CIM je preveč kompliciran
 - rezultirajoči sistem je preveč kompleksen
 - iskanje novih tehnologij, ki dajo boljše rezultate na krajši rok
- »Agent-based control«
 - sestav avtonomnih inteligentnih programskih modulov, ki znajo medsebojno sodelovati
- »Holonc systems«
 - sestav avtonomnih, samozadostnih proizvodnih enot, ki sodelujejo v doseganju skupnega cilja
 - vsaka enota je hkrati celota in del sistema
 - z gledovanje po živih sistemih

Odprtost in povezljivost programske opreme



6.5 Pojmi in kratice

- ActiveX
 - nadgradnja OLE, ki vključuje tudi možnost povezav preko Interneta
- COM/DCOM (Component Object Model/Distributed COM)
 - tehnologija povezovanja objektov na enem ali več računalnikih v omrežju
- DCS (Distributed Control System)
 - porazdeljen sistem za vodenje procesov
- DDE (Dynamic Data Exchange)
 - standard za izmenjavo podatkov med programi v okolju Windows
- DNA (Distributed interNet Application architecture)
 - predlagana arhitektura podjetja Microsoft za integrirano vodenje proizvodnih procesov

Pojmi in kratice /2

- ERP (Enterprise Resource Planning)
 - upravljanje z viri podjetja, tako proizvodnimi kot ostalimi (vzdrževanje, kadri ipd.)
- HMI (Human-Machine Interface)
 - vmesnik med operaterjem in sistemom vodenja procesa
- MES (Manufacturing Execution System)
 - povezava med programsko opremo za vodenje procesov in poslovnimi sistemi
- MRP (Material Requirements Planning)
 - planiranje materialnih potreb
- MRP II (Manufacturing Resource Planning)
 - upravljanje z viri proizvodnje,
- ODBC (Open DataBase Connectivity)
 - povezava do različnih relacijskih podatkovnih zbirk preko standardiziranega vmesnika

Pojmi in kratice /3

- OLAP (On-Line Analytical Processing)
 - programska orodja za sprotno predstavitev in analizo podatkov
- OLE (Object Linking and Embedding)
 - tehnologija povezovanja in vstavljanja objektov
- OPC (OLE for Process Control)
 - standard razvit v skladu z COM in OLE ter posebej namenjen za področje procesnega vodenja
- PLC (Programmable Logic Controller)
 - programirljivi logični krmilnik
- SCADA (Supervisory Control And Data Acquisition)
 - programska oprema za spremljanje in nadzor
- VBA (Visual Basic for Applications)
 - skriptni programski jezik