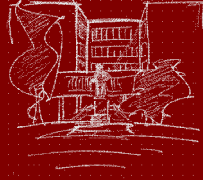


[P273] **Пројектовање база података** 10



Саша Малков
Универзитет у Београду
Математички факултет
2023/2024

[P273] **Пројектовање база података**
Саша Малков




Тема 10
Дистрибуиране базе података

[P273] - Пројектовање база података - Саша Малков - 2023/24 - час 10 1

Дистрибуиране базе података – Основни појмови

Појмови




- **Дистрибуирано извршавање** је када се два посла извршавају у физички раздвојеним адресним просторима
 - раније је коришћено “на физички раздвојеним рачунарима” али данас се сматра да извршавање на различитим рачунарима, а који деле радну меморију, није дистрибуирано извршавање
 - алтернативно “...када два посла могу да комуницирају искључиво посредством рачунарске мреже”

Универзитет у Београду – Математички факултет

[P273] - Пројектовање база података - Саша Малков - 2023/24 - час 10 2

Дистрибуиране базе података – Основни појмови

Појмови



- **ЦБП – Централизована база података** је база података која у целости почива на једном рачунару
- **ЦСУБП – Централизовани систем за управљање базама података** је софтверски систем који омогућава управљање ЦБП-ом

Универзитет у Београду – Математички факултет

[P273] - Пројектовање база података - Саша Малков - 2023/24 - час 10 3



Појмови

- **ДБП - Дистрџбуирана база њодаџака** је скуп више **лоџички међузависних** база података које су дистрибуиране на рачунарској мрежи
 - тј. постоје правила интегритета која се односе на више база података
- **ДСУБП - Дистрџбуирани сисџем за уџрављање базама њодаџака (DDBMS – distributed DBMS)** је софтверски систем који омогућава управљање ДБП тако да је **дистрџбуираносџи џрансаренџна за корисника**



Дистрибуираност и базе

- У контексту разматрања дистрибуираних база података користи се искључиво строга дефиниција дистрибуираности
- суштински је важно да су дистрибуиране компоненте на различитим рачунарима
 - али удаљеност није значајна, могу да буду један до другог или на различитим континентима
- у супротном може да нам промакне већи број веома важних питања
- Претпоставља се да је једини дељени ресурс **мрежа**



Вишепроцесорски системи

- Ако системи деле меморију називају се :
 - **сисџеми са дељеном мемориџом**
 - ако се дели примарна мемориџа
 - **сисџеми са дељеном диском**
 - ако се дели секундарна мемориџа
- Ако деле и примарну и секундарну меморију и У/И уређаје, називају се **сисџеми који деле све**
 - у овакве системе се сврставају и они који деле меморију
- Ако не деле ни примарну ни секундарну меморију ни У/И уређаје, називају се **сисџеми који не деле ништа**



Вишепроцесорски системи (2)

- Вишепроцесорски системи нису довољан предуслов за дистрибуиране базе података
- ако деле меморију (или чак све) онда је то један рачунар у коме се комуникација остварује кроз меморију, а не путем порука кроз мрежу, па нема дистрибуираности
- чак и ако не деле ништа, обично се ради о симетричним системима са истим (или скоро истим) процесорским и меморијским подсистемима, који су контролисани једним истим оперативним системом и чине **један рачунар**



Базе података на мрежи

- Чињеница да се СУБП налази у оквиру мреже није довољна да би се радило о ДСУБП
- Ако је на рачуарској мрежи са већим бројем рачунара база података лоцирана **на само једном чвору**, тада то није ДСУБП
- СУБП је дистрибуиран акко је **лоциран на више различитих чворова** у мрежи



Предмет дистрибуирања

- Кључно питање је **шта** се дистрибуира?
- Може да се дистрибуира практично све:
 - логика обраде
 - појединачне функције
 - подаци
 - управљање обрадом или трансакцијама



Предмет дистрибуирања (2)

- Дистрибуирање логике обраде
 - обрада се дистрибуира међу јединицама за обраду (чворовима)
 - може да буде прихватљива и “блажа” дефиниција ДБП
 - подразумева висок ниво концептуалне хомогености
 - иако хардвер не мора бити исти, софтвер најчешће мора
- Дистрибуирање појединачних функција
 - различите функције се извршавају на различитим чворовима
 - неке (или све) функције могу да се извршавају на више чворова (репликација функција)
 - није неопходан висок ниво хомогености
 - довољно је да подсистеми могу да комуницирају, није неопходна хомогеност ни на нивоу хардвера ни на нивоу софтвера
- ...



Предмет дистрибуирања (3)

- ...
- Дистрибуирање података
 - подаци могу да се налазе на већем броју чворова
 - хоризонтална и вертикална фрагментација
 - репликација података
 - каталог података може да буде централизован, реплициран или дистрибуиран
- Дистрибуирање управљања
 - неки или сви послови управљања могу бити дистрибуирани
 - управљање трансакцијама
 - управљање привилегијама
 - управљање обрадом, тј. организацијом извршавања обраде



Критеријуми за посматрање

- Дистрибуирани рачунарски системи могу да се посматрају и класификују на основу различитих критеријума:
 - степен међусобне спрегнутости чворова
 - структура међусобне повезаности
 - међузависност компоненти
 - синхронизација међу компонентама



Степен спрегнутости

- Степен спрегнутости је мера која описује колико су чврсто међусобно повезани елементи система
 - Може да се опише нумерички, на различите начине
 - релативна мера, нема много смисла при поређењу различитих система
 - Чешће се описује речима, као *ниска* или *висока* спрегнутост
 - ниска спрегнутост елемената означава да они немају дељених подсистема и комуницирају путем рачунарске мреже, док
 - висока спрегнутост елемената означава да они имају неке дељене подсистеме (нпр. меморију, дискове или посвећене комуникационе линије) и да комуницирају на неки ефикаснији начин



Степен спрегнутости (2)

- Пример нумеричког описивања степена спрегнутости
 - $Cpl = AvgDataExchg / AvgLocalProcess$
 - $AvgDataExchg$ = просечна количина података који се размењују између неког чвора система са осталим чворовима у јединици времена
 - $AvgLocalProcess$ = просечна количина посла обављеног на том чвору у јединици времена
 - на пример:
 - ако један чвор при просечном оптерећењу у јединици времена размени са другима у просеку 250MB
 - ...и при томе изврши 550MI (милиона инструкција)
 - ...онда је његова спрегнутост 0,40
 - висока спрегнутост може да означава ниску самосталност чвора у обављању послова



Структура међусобне повезаности

- Разликују се два основна случаја:
 - непосредно повезивање чворова (*P2P – peer to peer*) путем сопствених резервисаних канала везе
 - употреба дељених канала везе за комуникацију међу свим системима
- У пракси су честе комбиноване сложене структуре



Међузависност компоненти

- Међузависност компоненти описује у којој мери компоненте зависе једна од друге при раду
- Пример, према функцијама:
 - висока – ако компонента А не може да заврши посао без помоћи компоненте Б у реалном времену
 - ниска – ако отказ компоненте Б неће значајно ометати рад компоненте А



Синхронизација међу компонентама

- Компоненте могу да комуницирају
 - синхроно и
 - асинхроно
- Ако је комуникација синхрона, међузависност је висока
 - успех обављања посла зависи од успеха комуникације и извршавања одговарајућих послова на другом чвору
- Ако је комуникација асинхрона, међузависност није обавезно ниска



Доприноси ДСУБП

- Основни доприноси ДСУБП су:
 - Транспарентно управљање дистрибуираним и реплицираним подацима
 - Поузданост дистрибуираних трансакција
 - Унапређење перформанси
 - Лакше проширивање система
- (под *доприносима* се мисли на наша очекивања, тј. на оно што ће ДСУБП да нам додатно пружи, у односу на прост скуп више ЦСУБП)



Транспарентно управљање дистрибуираним и реплицираним подацима

- Транспарентност управљања подразумева да се ДСУБП бави раздвајањем семантике високог нивоа од проблема који настају при имплементацији на ниском нивоу
- На пример:
 - нека су у бази података банке похрањени подаци о запосленима у свим експозитурама
 - независно од тога где се физички чувају подаци, да ли су дистрибуирани или не и слично, подацима мора да се приступа на исти универзалан начин (на пример, путем упита на упитном језику)
 - сву сложеност која настаје као последица дистрибуирања решава ДСУБП и сакрива је од корисника

Транспарентно управљање дистрибуираним и реплицираним подацима (2)



- Транспарентност управљања подацима има следеће аспекте:
 - Независност података
 - Мрежна транспарентност
 - Транспарентност репликације
 - Транспарентност фрагментације

Транспарентно управљање дистрибуираним и реплицираним подацима (3)



- Транспарентност управљања подацима има следеће аспекте:

- **Независност података**
 - Логичка независност података
 - Отпорност апликације на промене логичке структуре података
 - Апликације добро подносе додавање нових елемената структуре базе података
 - Физичка независност података
 - Потпуно скривање физичке структуре података од апликације
 - Апликација не сме да трпи никакве последице у случају промене физичке структуре података, макар она укључивала и промене у начину дистрибуирања података
- Мрежна транспарентност
- Транспарентност репликације

Транспарентно управљање дистрибуираним и реплицираним подацима (4)



- Транспарентност управљања подацима има следеће аспекте:
 - Независност података
 - Мрежна транспарентност
 - У централизованим базама података једини ресурс о коме се стара транспарентно јесу подаци
 - Кориснику није потребно да зна како се управља подацима
 - У дистрибуираним системима, мрежна структура је потребно да буде склоњена од очију корисника и транспарентно управљања
 - Кориснику није потребно да зна где су подаци и како чворови комуницирају
 - Назива се и *дистрибутивна транспарентност*
 - Транспарентност репликације
 - Транспарентност фрагментације

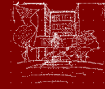
Транспарентно управљање дистрибуираним и реплицираним подацима (5)



- Транспарентност управљања подацима има следеће аспекте:

- Независност података
- Мрежна транспарентност
- Транспарентност репликације
 - Репликација је чување истих података на више локација
 - као и обављање истих послова на различитим чворовима
 - предузима се ради перформанси, расположивости и поузданости
 - (више о томе касније)
 - Корисници не би требало да буду свесни чињенице да се подаци и послови реплицирају
 - било да су реплицирани или не, они се морају користити на исти начин

Транспарентно управљање дистрибуираним и реплицираним подацима (6)



- Транспарентност управљања подацима има следеће аспекте:
 - Независност података
 - Мрежна транспарентност
 - Транспарентност репликације
 - Транспарентност фрагментације
 - Фрагментација је чување различитих делова исте колекције података на више локација
 - Предузима се ради перформанси, расположивости и поузданости
 - (више о томе касније)
 - Корисници не би требало да буду свесни чињенице да су подаци фрагментисани
 - Било да су фрагментисани или не, они морају да се користе на исти начин
 - Обично се упити (тзв. *глобални упити*) преводе у извршавања колекције мањих (тзв. *локалних упита*) и резултати се унирају

Транспарентно управљање дистрибуираним и реплицираним подацима (7)



- Носилац транспарентности може да буде
 - упитни језик
 - сваки упит се аутоматски преводи на одговарајући начин, у зависности од стварне организације и дистрибуираности података
 - оперативни систем
 - оперативни систем може да се стара о повезивању различитих модула без обзира на њихову физичку локацију
 - ДСУБП
 - сва питања структуре и дистрибуције података се разрешавају на нивоу СУБП-а
 - ово је уобичајен приступ

Поузданост дистрибуираних трансакција



- Носилац поузданости у раду СУБП-а је трансакција
- Извођење трансакција у дистрибуираном окружењу доноси нове проблеме
 - Нпр. превазилажење отказивања једног од удаљених уређаја у фази потврђивања трансакције
 - (у даљем току курса бавићемо се и дистрибуираним трансакцијама)

Унапређење перформанси



- Допринос ДБП перформансама обично се огледа у томе што:
 - Фрагментисани подаци могу да се чувају ближе месту употребе
 - Равномерно је расподељено оптерећење на више сервера на различитим локацијама
 - Мање се времена троши на пренос података
 - Иако може да изгледа да брзе мреже умањују значај овог проблема, чињеница је да се подаци дистрибуирано производе и користе, па ће дистрибуирано чување увек бити природна организација
 - Посебно, брза мрежа може да значајно смањи трајање преноса, али не тако добро и почетно кашњење одговора
 - Глобални упити се паралелизују
 - Подаци су дистрибуирани, па се глобални упити деле на мање, који се односе на локалне податке



Лакше проширивање система

- Много је лакше да се повећа капацитет дистрибуираног СУБП него централизованог
 - скоро увек је јевтиније да се иста снага обезбеди помоћу више мањих рачунара него помоћу једног великог
 - наравно да постоји простор за велике рачунаре, али је у великом броју случајева из економског угла оправданије улагање у дистрибуирани систем
 - мање иницијално улагање у хардвер
 - постепено проширивање у складу са реалним потребама
 - *може да представља проблем сложености софтвера*



Отежавајући фактори примене ДСУБП

- Најважнији отежавајући фактори су:
 - Сложеност (пре свега софтвера)
 - Цена
 - Дистрибуирана контрола
 - Безбедност



Отежавајући фактори

- Најважнији отежавајући фактори су:
 - **Сложеност**
 - Дистрибуирани системи су све само не једноставни
 - Дистрибуирани СУБП мора да реши све проблеме које решава и централизован СУБП и још многе друге
 - **Цена**
 - **Дистрибуирана контрола**
 - **Безбедност**



Отежавајући фактори

- Најважнији отежавајући фактори су:
 - **Сложеност**
 - **Цена**
 - Дистрибуирани системи захтевају додатни хардвер (нпр. комуникациони)
 - Софтверски аспекти система су много сложенији и скупљи за развијање
 - На свакој локацији где постоје сервери потребно је и одржавање
 - **Дистрибуирана контрола**
 - **Безбедност**



Отежавајући фактори

- Најважнији отежавајући фактори су:
 - **Сложеност**
 - **Цена**
 - **Дистрибуирана контрола**
 - Отежани су синхронизација и координација компоненти
 - **Безбедност**



Отежавајући фактори

- Најважнији отежавајући фактори су:
 - **Сложеност**
 - **Цена**
 - **Дистрибуирана контрола**
 - **Безбедност**
 - Код централизованих БП старање о безбедности је централизовано
 - Овде је дистрибуирано и укључује додатне аспекте
 - безбедности рачунарских мрежа
 - усклађивања безбедносних аспеката на различитим чворовима
 - повећан број ризичних улазних тачака
 - и друго



Проблеми и теме ДСУБП

- Најважнији нови проблеми који се решавају при употреби ДСУБП су:
 - Пројектовање дистрибуираних база података
 - Дистрибуирано извршавање упита
 - Дистрибуирано управљање метаподацима
 - Дистрибуирана контрола конкурентности
 - Дистрибуирано управљање мртвим петљама
 - Поузданост дистрибуираних СУБП
 - Подршка у оперативним системима
 - Хетерогене базе података



Пројектовање дистрибуираних БП

- Нови проблеми у контексту дистрибуирања:
 - Подаци могу да буду
 - **партиционисани** – (фрагментисани) подељени на више локација
 - **репликацирани** – поновљени на више локација
 - Могуће су и комбинације
 - У случају репликације може се применити
 - **пuna репликација**
 - **делимична репликација**
 - У случају партиционисања
 - одређује се начин фрагментисања података
 - **хоризонтална фрагментација**
 - **вертикална фрагментација**
 - оптимизује се дистрибуција фрагмената
 - Одлучивање често почива на решавању *NP* сложених проблема, па се приступа хеуристички



Дистрибуирано извршавање упита

- Извршавање упита обухвата анализу упита и превођење упита у низ операција
- Оптимизација упита је сложено питање и у централизованим БП, а у случају ДБП то постаје још израженије
- Додатни фактори у одлучивању су
 - цена мрежног транспорта
 - различитост цена извршавања на различитим локацијама
 - различитост расположивих ресурса на различитим локацијама
 - могућности паралелизације



Дистриб. управљање метаподацима

- Метаподаци обухватају све информације које се о структури, дистрибуираности и садржају базе података морају чувати да би њен рад био могућ
- За разлику од централизованих БП, сада:
 - постоје додатне информације о локацијама чворова
 - постоје додатне информације о распоређености података по чворовима
 - фрагментација
 - репликација
 - сваки чвор мора да зна много тога о другим чворовима, али не увек све
 - метаподаци могу бити
 - централизовани
 - реплицирани
 - партиционисани



Дистриб. контрола конкурентности

- Контрола конкурентности подразумева синхронизацију приступања подацима од стране конкурентно извршаваних процеса
- Обухвата
 - старање о интегритету базе података (као и за ЦБП)
 - старање о конзистентности реплика
 - *узајамна конзистентност*
 - сложеније старање о изолованости
 - ако се у трансакцији мења неки податак, онда морају да се изолују све његове реплике
- Неколико приступа
 - *несиметрички приступ*
 - синхронизација пре оствареног приступа
 - *асиметрички приступ*
 - синхронизација после оствареног приступа
- Концепти
 - *закључавање* – почивају на међусобном искључивању
 - *временски њечашци* – почивају на одржавању редоследа извршавања
 - *хибридни механизми*



Дистриб. управљање мртвим петљама

- Потенцијално виша сложеност проблема и конкурентних односа него код ЦБП
- Решава се на сличан начин као и на нивоу оперативних система:
 - превенцијом
 - избегавањем
 - препознавањем и опоравком



Поузданост дистрибуираних СУБП

- Поузданост не долази сама по себи, тј. за сваки од наредних доприноса су потребни неки предуслови
 - Ако један чвор откаже, остали и даље раде
 - Ако један чвор откаже, други могу да преузму његову улогу
 - Када се чвор опорави, аутоматски се освежава његов садржај на основу других чворова



Подршка у оперативним системима

- Оперативних системи не пружају увек одговарајуће услуге сложеним системима, какав ДСУБП извесно јесте
- Проблеми и загушења обично настају по питању
 - управљања радом процесора и расподелом времена
 - управљање процесима
 - управљања меморијом
 - рада са фајловима
 - опоравак
- Често се неки елементи имплементирају *исц̄иог* нивоа оперативног система
 - нпр. непосредан приступ хардверу
 - дискови, меморијски модули,...



Хетерогене базе података

- ДБП често настаје спајањем више постојећих решења
- Тада је обично нарушена
 - хомогеност софтвера – јер различите базе раде под различитим СУБП
 - хомогеност структуре – јер структура различитих база података није усаглашена
- И тада је потребно решавати неке од описаних проблема
 - дистрибуирано извршавање упита
 - управљање метаподацима
 - управљање конкурентношћу
- Овакви системи се често називају *системи са више база њодатака*
 - често се не сврставају у ДБП, тј. нису *праве* ДБП



Модели архитектуре ДСУБП

- При разматрању потенцијалних архитектура од значаја су три основне димензије проблема:
 - аутономија чворова
 - дистрибуираност чворова
 - хетерогеност чворова
- Ове димензије су у највећој мери међусобно ортогоналне



Аутономија чворова

- Аутономија чворова се односи првенствено на управљање, а не на складиштење података
 - описује способност чворова да самостално извршавају операције
- Аутономија зависи од великог броја чинилаца
 - да ли и како чворови размењују информације
 - да ли чворови могу независно да извршавају трансакције
 - да ли један чвор може мењати друге чворове
 - да ли један чвор може самостално да управља локално ограниченим трансакцијама
- Захтеви могу да се формулишу на различите начине
 - размотрићемо аутономију према интензitetу повезивања



Аутономија чворова (2)

- По интензитету повезивања (*Ozsu, Valduriez*, 1999):
 - **Тесна интеграција**
 - једна јединствена слика базе података је доступна свим корисницима свих чворова
 - из угла корисника, база података као да је централизована
 - сваким захтевом корисника управља тачно један од управљача подацима, чак и када његово извршавање изводи више њих
 - управљачи подацима се уобичајено не понашају као да су независни системи, чак иако су обично способни за то
 - **Полуаутономни системи**
 - **Пуна изолованост**



Аутономија чворова (3)

- По интензитету повезивања (*Ozsu, Valduriez*, 1999):
 - **Тесна интеграција**
 - **Полуаутономни системи**
 - чворови система могу да раде и обично раде независно
 - успостављају федерацију ради дељења својих локалних података
 - сваки чвор одређује које ће податке ставити на располагање другим чворовима
 - нису потпуно аутономни зато што могу да размењују податке
 - **Пуна изолованост**



Аутономија чворова (4)

- По интензитету повезивања (*Ozsu, Valduriez*, 1999):
 - **Тесна интеграција**
 - **Полуаутономни системи**
 - **Пуна изолованост**
 - сваки чвор је самосталан СУБП који нити је свестан постојања других СУБП нити комуницира са њима
 - из угла чвора, свеједно је да ли му је неки захтев упутио клијент (апликација) или други чвор
 - извршавање глобалних трансакција је посебно сложено јер не постоји пуна глобална контрола

Дистрибуираност чворова

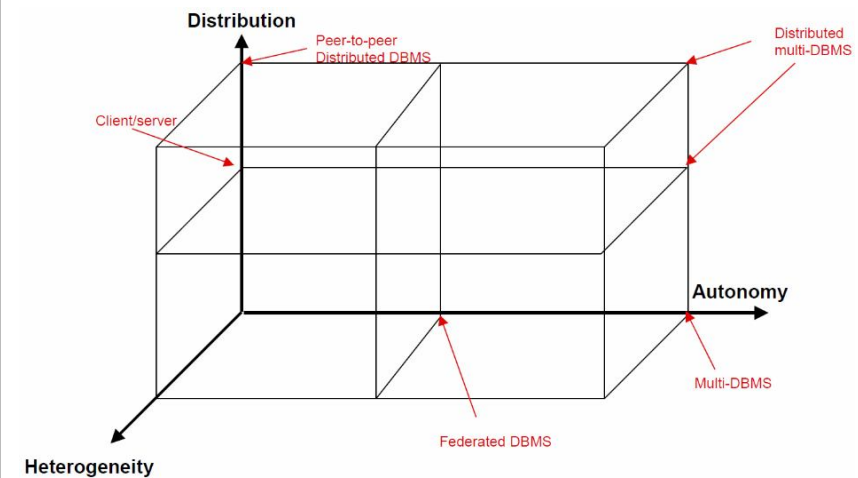
- Дистрибуираност се односи на податке
 - разматра се физичка дистрибуираност података
 - корисник све податке види као јединствену колекцију
- Постоје три основна нивоа дистрибуирања
 - Централизованани подаци (нема дистрибуирања)
 - Клијент-сервер
 - (у овом контексту клијент може да буде и други чвор)
 - Равноправни чворови

Хетерогеност

- Хетерогеност има више аспеката и односи се
 - на хардвер
 - на оперативне системе
 - на мрежне протоколе
 - на софтвер СУБП
- Посебно значајни аспекти су
 - модели података
 - упитни језици
 - протоколи управљања трансакцијама

Основне алтернативе архитектура

- За сваку димензију уводимо значајне тачке:
 - Аутономија:
 - 0 – тесна интеграција
 - 1 – полуаутономни системи
 - 2 – пуна изолованост
 - Дистрибуираност:
 - 0 – нема дистрибуираности
 - 1 – клијент/сервер системи
 - 2 – равноправни чворови
 - Хетерогеност:
 - 0 – хомогени системи
 - 1 – хетерогени системи (делимично или потпуно хетерогени)



Литература за ову тему



- Ozsü, Valduries – **Principles of Distributed Database Systems** (2.ed) (1999)