دانشگاه آزاد اسلامی واحد آذرشهر

جزوه استاد:

امیر مصطفی زاده کوچه

زمستان ۹۰

فصل اول

نسل اول سیستم ها:

عدم سیستم واسط (برنامه) باید تمام مشکلات و مسائل کار مستقیم با محیط فیزیکی را حل می کرد.

از معایب عمده این سیستم ها:

۱. *وابستگی کامل به Applications[۱] به سخت افزار*
۲. *افزایش حجم برنامه ها*
۳. *برنامه نویس به جای اینکه حواص خود را کاملاً معطوف به Application بکند مجبور بود به مسائل دیگری مثل نحوه تعاول برنامه کاربردی با سخت افزار هم توجه داشته باشد.*

نسل دوم سیستم ها:

ارائه سیستم فایل FS[۲] : این سیستم ها بسیاری از مسائل مربوط به لایه فیزیکی را حل نموده و لیکن استقلال از محیط فیزیکی صد در صد تامین نشده بود، چرا که هنوز هم کاربران با مفاهیم فایل سیستم مثل ساختار های شاخص دار در تماس بودند.

نسل سوم سیستم ها :

ارائه سیستم FDMS[۳] : این سیستم ها مثل FS ها بودند ولی یکسری امکانات بیشتری را در اختیار کاربران قرار می دادند یعنی یکسری توابع نرم افزاری به FS اضافه شده بود و امکانات بیشتری را محیا می کرد.

نسل چهارم سیستم ها:

ارائه DBMS[۴] [سیستم مدیریت پایگاه داده] :با ارائه این سیستم یک محیط کاملاً انتزاعی و سطح بالا برای کاربران ایجاد شد . DBMS همواره مبتنی بر ساختار داده ای[۵] مشخص است.

---

[۱] برنامه ها
[۲] File system : یکی از لایه های سیستم عامل می باشد.
[۳] File Data Management System
[۴] Data Base Management System
[۵] Data Structer

نسل پنجم سیستم ها:

KBMS[1] : این سیستم بر اساس مئیریت سیستم پایگاه دانش می باشد.

نسل ششم سیستم ها:

OODBMS[2] : این سیستم بر شی گرایی است که بیشتر در باره عکس ها کار می کند می باشد.

پایگاه داده[3] چیست؟

پایگاه داده مجم.عه ای داده های ذخیره شده، تعریف شده به صورت صوری اقلاً در سطح خارجی علاوه بر تعاریف در سطح درونی تر مبتنی بر Data structure [ساختار داده] حداقل فاقد افزونگی که مورد استفاده یک یا چند کاربر به صورت همزمان و اشتراکی قرار می گیرد و تحت مدیریت یک سیستم متمرکز عمل می کند.

مشی یا اسلوب تولید یک سیستم : در حالت کلی برای تولید یک سیستم دو مشی وجود دارد :

۱. *مشی کلاسیک (غیر بانکی ، سنتی ، ناپایگاهی)*
۲. *مشی بانکی (پایگاهی)*

در مشی کلاسیک FS به عنوان واسط عمل می کند ولی در مشی بانکی DBMS به عنوان واسط عمل می کند.

در مشی کلاسیک سیستم توسط افراد مختلف و بدون هماهنگی ایجاد می شود گاه ممکن است حتی در یک سیستم نیز زیر سیستم ها مجزا و مستقل از هم باشند در نتیجه تعدادی سیستم جزیره ای و بدون ارتباط ایجاد می شود .

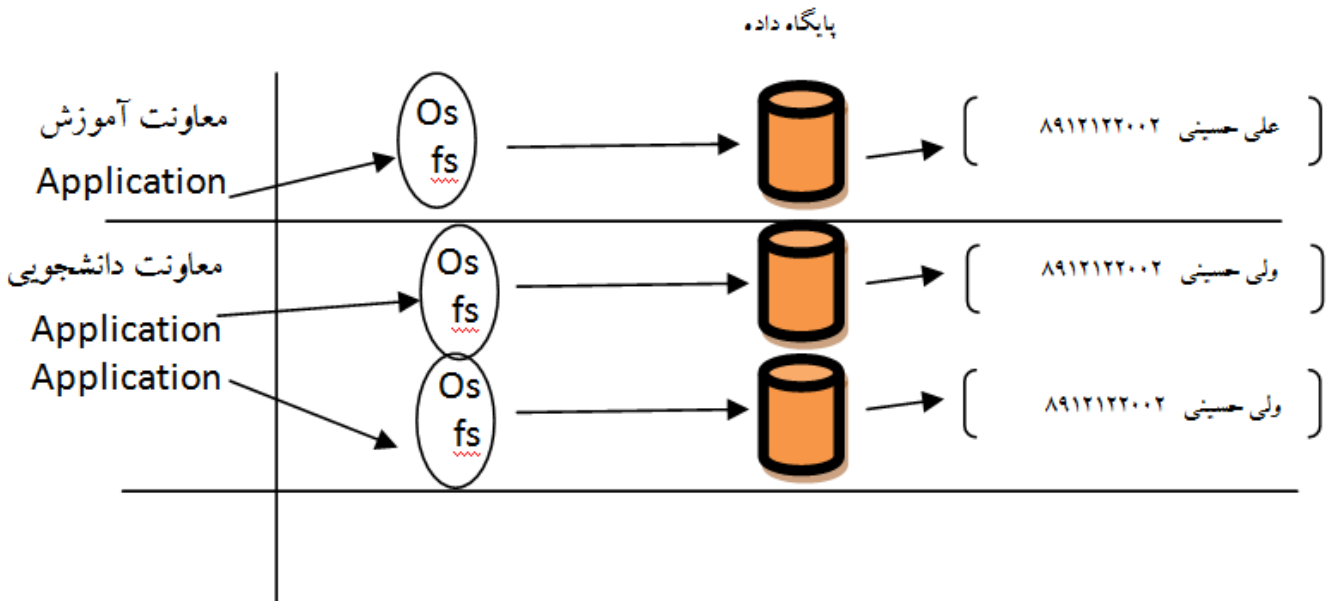
---

[1] Knowledge Base Management System

[2] Object Oriented Data Base Management System

[3] Data Base

یک مثال برای مشی کلاسیک:

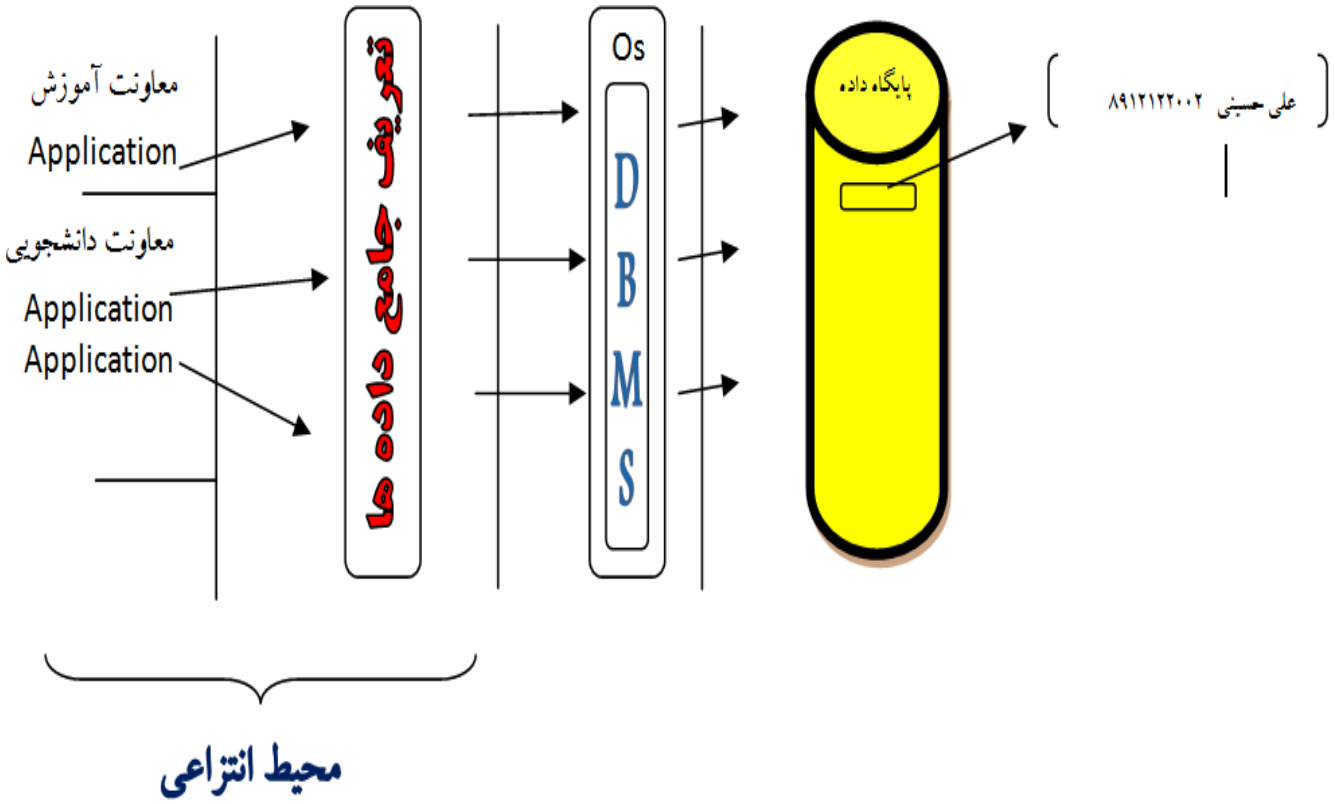


توجه : در شکل زیر مشاهده می کنید که نام علی حسینی و در قسمت دیگر با نام ولی و با شماره دانشجویی یکسان ثبت شده است که این اشتباه هنگام ثبت نام رخ داده است ولی در هر قسمت این دانشجو با نام دیگر شناخته می شود.

ویژگی های مشی کلاسیک:

۱. *سیستم های متعددی وجود دارد*
۲. *زبان برنامه نویسی معمولاً یک زبان سطح بالا ست و می تواند بین زیر سیستم های مختلف متفاوت باشد.*
۳. *افزونگی اطلاعات داریم*
۴. *امکان رخ داد ناسازگاری زیاد است.(افزونگی و ناسازگاری در تعاول با یکدیگرند)*
۵. *مصرف بیش از حد حافظه ذخیره سازی وجود دارد.(اقلاً ناشی از افزونگی است)*
۶. *عدم امکان اعمال استاندارد ها در محور های مختلف(عدم امکان اعمال مجموعه ای واحد از استانداردها در مراحل مختلف مطالعه تخلیل – تعیین مشخصات فنی – طراحی – پیاده سازی – تولید و نگه داری – گسترش و بهره برداری)*
۷. *سیستم واسط Fs است با تمام محدودیت هایی که دارد همچنین سیستم ها جزیره ای بوده و موضوعی تحت عنوان سیستم متمرکز وجود ندارد.*
۸. *عدم اشتراک داده ها یا اشتراک داده در حیّ بسیار ضعیف*

مشی بانکی یا پایگاهی:

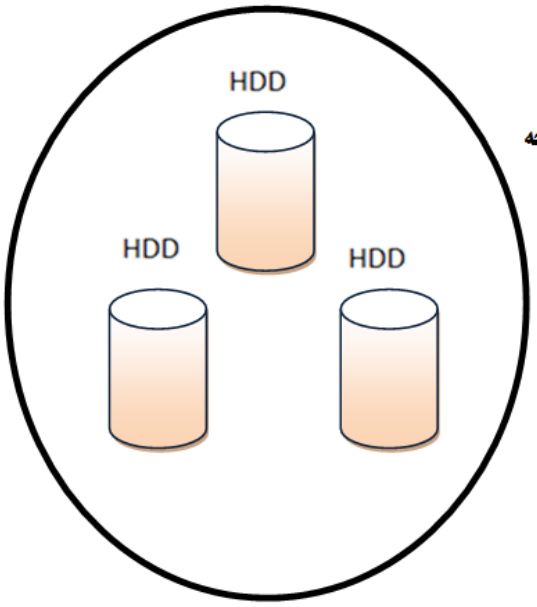


محیط انتزاعی : یعنی هر کاربر تصور می کند که پایگاه خاص خود ذا می بیند .یا به عبارت دیگر می توان گفت: کثرت و تنوع دید نسبت به داده های ذخیره شده است.

۱. *نیازهای اطلاعاتی و پردازشی همه قسمت ها ی مورد نظر مدیریت کل دانشگاه توسط یک گروه مورد مطالعه بررسی و تحلیل با هدف ایجاد یک سیستم یک پارچه قرار گیرد.*
۲. *داده هاب جمع آوری شده به شرحی که در ادامه خواهیم دید مدل سازی معنایی می شوند و در ادامه طراحی پایگاه داده در سطوح لازم انجام گرفته و این طراحی بط .ری که خواهیم دید در سطوح مختلفی بر اساس یک ساختار داده ای [Data structure] به منظور تعمین سطوح انتزاعی پایگاه داده انجام می شود بخ بیان دیگر سطح منطقی DS مشخص دارد و این DS است که تعیین کننده انتزاع است.*

۳. *مجموعه ای از برنامه ها های ایجاد پایگاه داده و کنترل پایگاه داده طراحی و تولید می شوند.*

۴. *محیط واحد و مجت‌مع و مشترک ذخیره ساری بین کاربران مختلف ایجاد می شود محیط واحد مجتمع و مشترک ذخیره سازی لزوماً به این معنا نیست که چنین محیطی از نظر فیزیکی و محل جغرافیایی واحد و یک پارچه باشد بلکه محیط پایگاه داده ها از لحاظ منطقی یک پارچگی دارد به عبارت دیگر پایگاه داد ه می تواند در عین واحد مجتمع و مشترک بودن در سطح منطقی از نظر فیزیکی چند پارچه و توزیع شده باشد.*



۵. *مجموعه ای از برنامه های بهره برداری از پایگاه داده طراحی و تولید می شود و پس از انجام تست های لازم و حصول اطمینان از عملکرد صحیح سیستم بهره برداری واقعی آغاز می گردد.*

عناصر تشکیل دهنده سیستم پایگاه داده:

مثل هر سیستم بازیابی اطلاعات سیستم پایگاه داده هم ۴ جزء اصلی زیر را داراست:

۱. سخت افزار
   - ❖ رسانه های ذخیره سازی
   - ❖ ماشین یا پردازنده
   - ❖ شبکه

۲. نرم افزار
   - ❖ DBMS
   - ❖ برنامه هایی که در محیط DBMS اجرا می شوند مثل ابزارهایی که در طراحی [۱] یا توسعه [۲] و غیره بکار می رود.[CAD\CAM][۱]

---

[۱] Design

[۲] Development

❖ نرم افزارهایی که شبکه در صورتی که بنا باشد پایگاه داده در محیط شبکه قرار می گیرد.

۳. کاربر

۴. خود داده

رسانه های ذخیره سازی : معمولا از دیسک هایی با ظرفیت بالا و سرعت دستیابی زیاد حدالامکان دیسک با بازوی ثابت استفاده می شود معمولا از دیسک ها برای ذخیره سازی داده های فعال استفاده می شود .

نوار هم معمولا برای تهیه نسخه های پشتیبان یا Backup یا آرشیو کردن داده ها استفاده می شود در این حالت معمولا داده ها غیر فعال هستند همچنین ممکن است از نوار برای عملیات رویداد نگاری و ثبت رخداد استفاده می کنیم . اصطلاحاً journaling event منظور از این ثبت هراتفاقی است که در داخل فایلی به نام log file نوشته می شود .

دلایل تهیه Backup[۲] : داده های پایگاه داده ممکن است در معرض حوادث مختلف قرار گیرد و از بین برود بنابراین لازم است همواره از داده های سیستم نسخه پشتیبان تهیه کرد .

بحث ترمیم (Recovery) : چنانچه بر اثر حادثه ای داده ها از بین رفت با استفاده از ترمیم پایگاه داده را به حالت قبل از رخداد failure[۳] طبیعتاً نسخه های پشتیبان می توانند در این عمل مورد استفاده قرار گیرند اما ترمیم می تواند طوری باشد که از نسخه های پشتیبان استفاده نکند .

Backup گیری باعث سطح ایمنی و قابلیت اطمینان خواهد شد.

Reliability[۴] & safety[۵]

نحوه ترمیم یا **Recovery** در DBMS ها یکی از پارامترهای مهم مقایسه DBMS ها است .

ماشین یا پردازنده : از ماشین های مختلف برای ایجاد و اجرای پایگاه داده می توان استفاده کرد . از ان جمله می توان به ماشین های داده سرور مثلاً main frame ها اشاره کرد .

---

[۱] Computer Aided Design \ Computer Aided Manipulation

[۲] پشتیبان

[۳] خطا

[۴] اطمینان

[۵] سلامت

با توجه به اهمیت تکنولوژی پایگاه داده ماشین های خاص پایگاه داده تولید و ارائه شده اند که به این ماشین ها Data Base machine (DBM) گفته می شود از ویژگی های DBM می توان به موارد زیر اشاره کرد .

۱ -تعداد زیادی به ثبات (Register) دارد .

۲ -حافظه های زیادی دارند .
- ۱- main memory
- ۲- cash
- ۳- حافظه های جانبی

۳ -معمولاً از کلنولوژی RAID برای ذخیره سازی استفاده می شود .

۴ -مکانیزم های آدرس دهی قوی دارند .

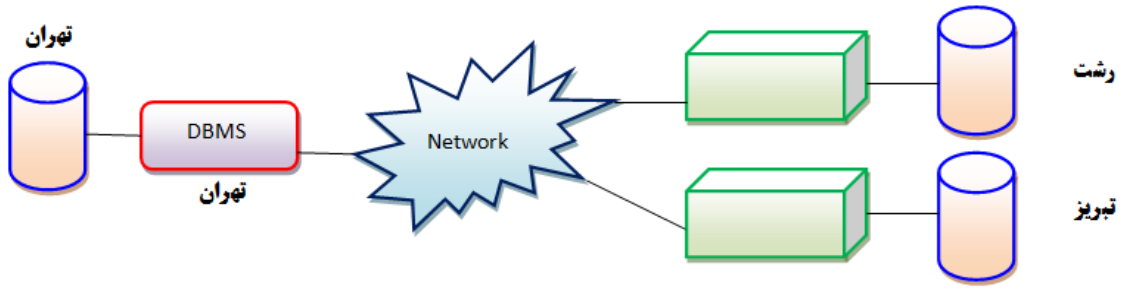
۵ -IO قوی دارد.

شبکه و ارتباطات

یک ماشین یا پردازنده و هزاران ترمینال شبکه تشکیل نمی دهند بلکه لازمه تشکی ل شبکه حداقل در ماشین مستقل است.

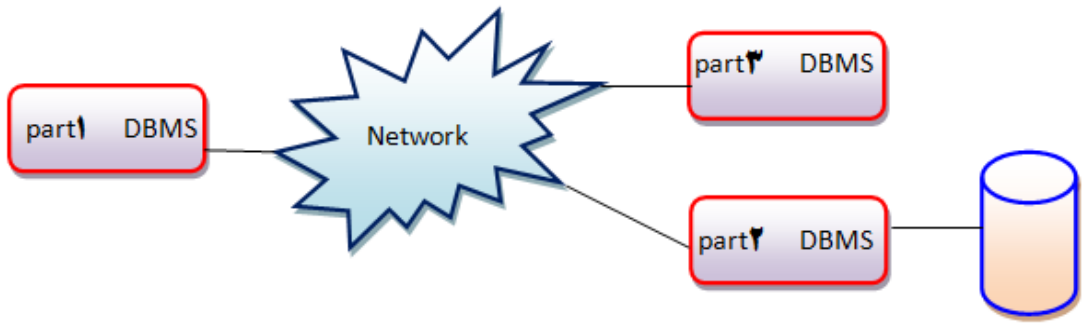
سیستم پایگاه داده توزیع شده ضرورتاً نیازمند شبکه است و مشتمل بر ۳ نوع زیر است .

۱ -سیستم (DBMS) متمرکز ها داده توفیع شده :



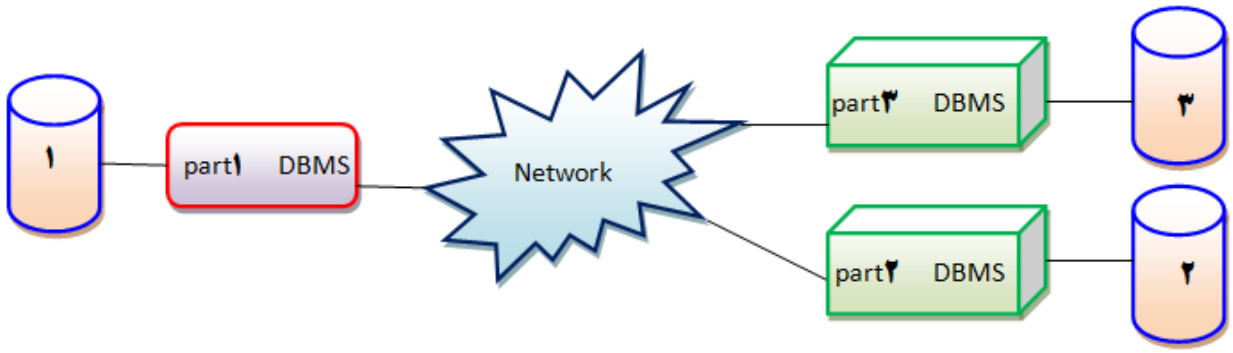
توجه : نکته مهمی که در این جا حائز اهمیت است بحث شفافیت مکانی یا شفافیت داده ای یا اصطلاحاً Data Transparency بدین معنی است که مهم نیست که داده در کجا قرار داشته باشد فقط قابل دسترس بودن آن مهم است به بیان دیگر کاربر هیچ اطلاعی از محل ذخیره سازی داده ندارد و تنها می تواند با داده مورد نظر خود کار کند .

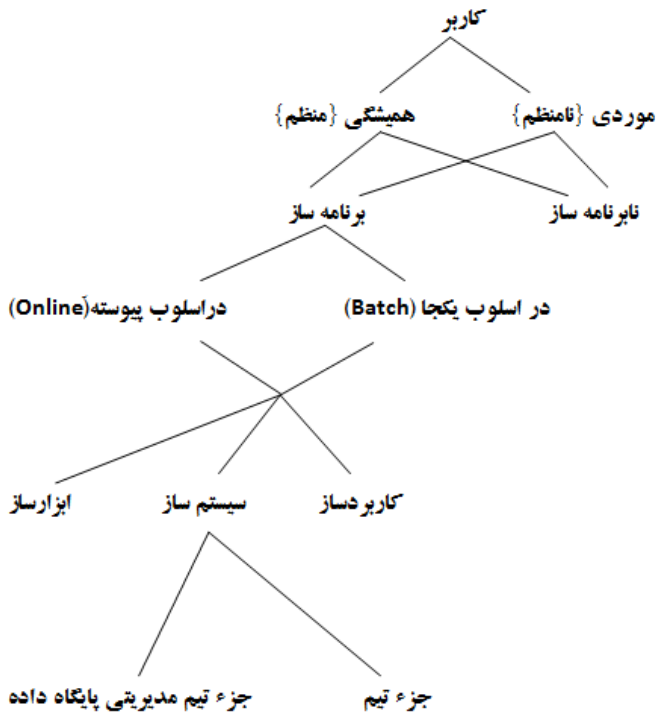
۲ -سیستم توزیع شده داده متمرکز :



در این روش هر ماشین قسمتی از DBMS را در خود دارد . در واقع این روش برای ماشین های با قابلیت پردازش پایین مناسب است .

۳ -سیستم وداده هر دو تضیع شده :





کاربران سیستم پایگاه داده تمامی افرادی هستند که از اولین روز تولید سیستم پایگاه داده با سیستم تعامل دارد .

نرم افزار : اصلی ترین نرم افزار در سیستم پایگاه داده DBMS است DBMS در واقع یک نرم افزار پیچیده است که به کاربران اجازه می دهد پایگاه خود را تعریف کنند و یا پایگاه خود را ایجاد کنند و یا پایگاه خود کار کنند و بر روی پایگاه خود کنترل داشته باشند .

داده : مجموعه اطلاعات درمورد موجودیت ها بعلاوه چگونگی ار تباط بین انها تشکیل داده های پایگاه داده را می دهند.

ساختار سیستم پایگاه داده :

User Inter face[1]		
DMBS		
Concurred control	Script	**Recovery**

کاربرانی که بیشتر به انها سروکار خواهیم داشت :

۱- DBA : Duta base Administrator
۲- SP : System programmer
۳- End-user

اهم وظایف DBA :

۱ -تعیین شمای فیزیکی
۲ -تعیین شمای منطقی
۳ -تعیین قونین جامعیت
۴ -تعیین قوانین امنیتی
۵ -پاسخ گویی به نیازهای روزمره کاربران
۶ -ایجاد تعادل بین نیازهای متضاد

[1] واسط کاربر

مدیریت تراکنش (Transaction management):

تعریف تراکنش : هر برنامه ای که در محیط بانک اطلاعاتی اجرا شود تراکنش نامیده می شود به بیان دقیق تر تراکنش مجموعه ای فرامینی است که روی داده های بانک اطلاعاتی اعمال شده و آنها را از یک وضعیت پایدار به یک وضعیت پایدار دیگر منتقل می کند . کنترل های لازمی می بایست بر روی یک تراکنش صورت گیرد عبارتنداز:

A	C	I	D
Atomicity	Consistency	Isolation	Durability
غیر قابل تجزیه	صحت	جداسازی(عایق سازی)	**پایداری یا ماندگاری**

Atomicity: این خاصیت به خاصیت یا همه یا هیچ معروف است بدین معنی که یا همه دستورات یک تراکنش با موفقیت تا انتها اجرا می شوند و یا هیچ یک از دستورات اجرا نمی شوند . مثال : انتقال ۱۰۰۰ دلار پول از یک حساب به حساب دیگر .

Beyin Transaction

Read (X , Acnt ۱)

Sub (x , ۱۰۰۰)

Write (x , Acnt ۱)

Read (Y , Acnt ۲)

ADD (Y , ۱۰۰۰)

Write (Y , Acnt ۲)

End Transaction

Consistency : یک تراکنش به تنهایی نمی تواند سیستم را به یک وضعیت ناپایدار و غلط برساند به عبارت دیگر نمی تواند قوانین امنیتی و جامعیتی را زیر پا بگذارد.

Isolation: دو تراکنش همزمان (همروند) باید طوری اجرا شوند که روی همدیگر اثر مخرب نگذارد.

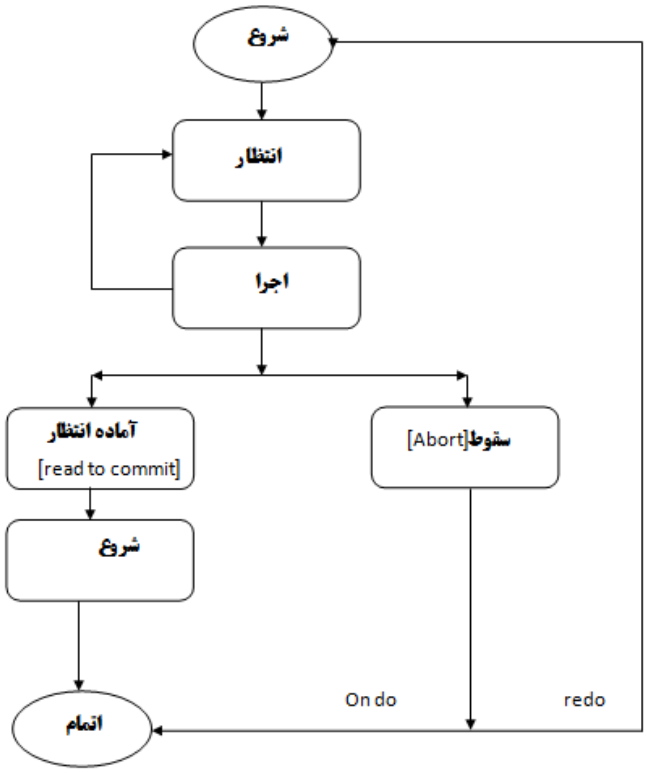
Begin Transaction ۱ *Read (x,p)* *Inc (x)* *Write (x,p)* *End Transaction* ۱	*Begin Transaction* ۲ *Read (y,p)* *Inc (y)* *Write (y,p)* *End Transaction* ۲
یک واحد اضافه می کند	یک واحد کم می کند

Durability: نتیجه اجرای یک تراکنش که با موفقیت به اتمام رسیده است نباید تحت هیچ شرایطی به طور اتفاقی از بین برود به عبارت دیگر داده های پایگاه داده ، داده های مانایی باشند . (ماندگار)

سرنوشت یک تراکنش :
۱- با موفقیت به طور کامل تا انتها اجرا می شود و به اتمام می رسد . انجام = Commit
۲- به هر دلیلی در یکی از مراحل کار متوقف می شود . سقوط = Abort

حالت هایی که باعث Abort یک تراکنش می شوند می توانند وجود اشکال – رفتن برق اشکالات نرم افزاری و یا همزمان تراکنش ها باشد .

فلوچارت مراحل اجرای یک تراکنش :



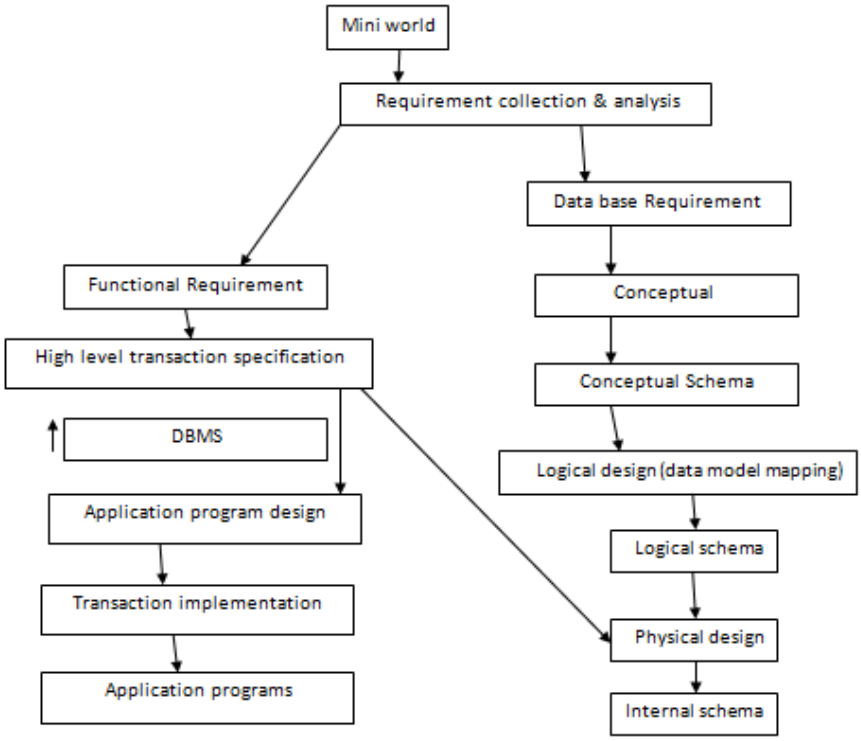
فصل دوم :

مدل سازی معنایی داده ها (ER) :

سال ها این سوال ذهن پژوهشگران را به خود معطوف داشته بود که بانک اطلاعات را چگونه می توان به شکل داده بیان کرد . به عبارت دیگر پژوهشگران به دنبال روشی بودند که بانک اطلاعاتی خو د را در قالب آن طراحی کنند و آنگاه بتوانند آن ها را در هر مدل دلخواهی پیاده سازی نمایند . در ۱۹۷۶ یک دانشجوی دوره دکترای انیستتو تکنولوژی ماساچوست (MIT) به نام چن هدلی برای طراحی بانک اطلاعاتی پیشنهاد کرد که این مدل مقبولیت عام یافت . او مدل خود را (ER) یا Entity Relationship[1].

نکته مهم : داده های ذخیره شدنی در پایگاه داده ها ابتدا می باست در بالاترین سطح انتزاعی مدل سازی معنایی شوند یعنی مدلی از محیط عملیاتی یا فرد جهان واقع با توجه به معنای داده ها به کمک مفاهیم مستقلی از ملاحظات مربوط به نمایش منطقی و جنبه های مربوط به نمایش فیزیکی داده ها ارائه شود . به مدل سازی گاه طراحی ادراکی یا مصنوعی گفته می شود .

نمودار زیر بیانگر فازهای اصلی طراحی پایگاه داده است .



[1] ارتباط موجودیت

مفاهیم بنیادی روش Entity Relation : ER

در روش ER سه مفهوم مبنایی وجود دارد و معنای داده های هر محیطی به کمک همین سه مفهوم نمایش داده می شود.

اولین مفهوم ۱- Entity موجودیت (نوع موجودیت)

نوع موجودیت ---------- نمونه نوع موجودیت

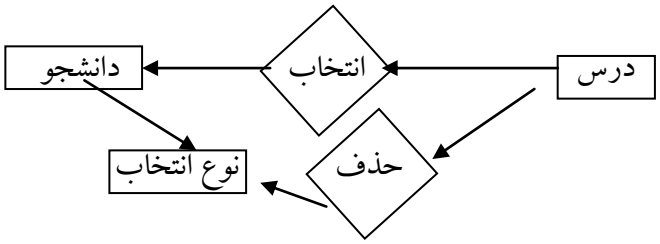
دانشجو ---------- نرگس توکلی

۲- Attribute صفت

دانشجو :نام ، نام خانوادگی ، شماره ، نام پدر

درس ---------- نام درس – تعداد واحد

۳- Relation Ship ارتباط یا رابطه (نوع ارتباط)



نوع موجودیت :

نوع موجودیت مبحث اصلی در ER می باشد که معادل چیز یا پدیده و یا به طور کلی هر آن چه که می خواهیم در مورد آن اطلاع و شناخت داشته باشیم مثلا در خردجهان واقع دانشکده انواع موجودیت ها عبارتند از : دانشجو – درس – استاد – کارمند دانشگاه و ...

نکته : هر نوع موجودیت نمونه ها و یا مصادیقی در خرد جهان واقع دارند نمونه های هر نوع موجودیت از طریق صفاتشان از هم دیگر متمایز می شوند .

نوع موجودیت درس نمونه موجودیت پایگاه – ریاضی – ذخیره

نکته : تشخیص دقیق انواع موجودیت ها در یک محیط کاریست سهل و ممتنع و تنها می توان به طور شهودی درباره انواع موجودیت ها سخن گفت و آنها را تشخیص داد اما می توان سه ضابطه زیر را در تشخیص یک نوع موجودیت در نظر داشت :

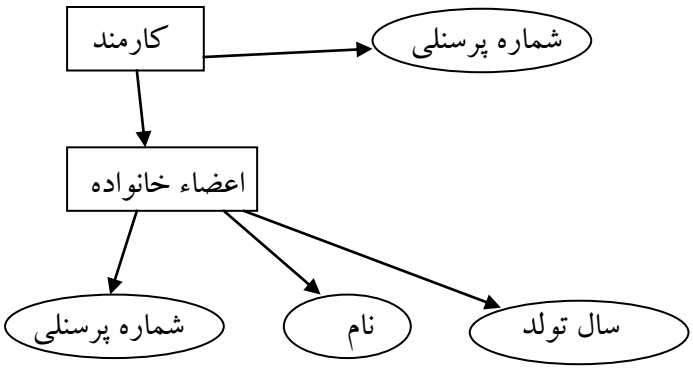
۱ -*یک نوع موجودیت از یک محیط معمولا بیش از یک نمونه متمایز از یکدیگر دارد .*

۲ -*یک نوع موجودیت معمولا بیش از یک صفت دارد .*

۳ -*معمولا حالت فاعلیت با مفعمولی دارد .*

نکته : یک نوع موجودیت ممکن است قوی (مستقل) و یا ضعیف (وابسته) باشد . موجودیت قوی موجودیتی است که مستقل از هر نوع موجودیت دیگر و به خودی خود در یک محیط مشخص مطرح باشد . مثل دانشجو . موجودیت ضعیف موجودیت ی است که وجودش در سیستم وابسته به یک نوع موجودیت دیگر است به عبارت دیگر اگر موجودیت قوی از سیستم حذف حذف شود این موجودیت نیز خود به خود از سیستم حذف خواهد شد .

نکته : در کتب دیگر چنین بیان شده است که نوع موجودیت ضعیف از خود نشانه ندارد اما گرچه موجودیت ضعیف نشانه ندارد اما یک صفت ممیزه دارد که به آن کلید جزئی گفته می شود به عبارت دیگر نشانه نوع موجودیت ضعیف ترکیبی است از نشانه نوع موجودیت قوی + صفت یا صفات ممیزه نوع موجودیت ضعیف آن .



صفت یا خصیصه :

ویژگی های ..... در موجودیت با تعداد صفت مشخص می شود هر صفت یک نام و یک نوع معنای مشخص دارد . مثال : نوع موجودیت درس : شماره درس – عنوان درس – تعداد واحد درس – نوع درس (پایه - .....)

ماهیت درس (نظری – عملی) سطح درس (کاردانی – کارشناسی – دکترا) فهرست محتوای درس (سیلابی) و ...

نکته : هر صفت از یک مجموعه از مقادیر مجاز و معتبر مقدار می گیرد که به این مجموعه از مقادیر اصطلاحا یا دامنه یا میدان مقادیر آن صفت می گویند .Doman

نکته : برای یک موجودیت فقط صفاتی از آن نگهداری می شود که در سیستم مورد نیاز است بنابراین ممکن است در یک سیستم برای یک نوع موجودیت برخی از صفات در نظر گرفته نشود .

انواع صفات:

۱ -ساده یا مرکب : صفت ساده صفتی است که مقدار آن از لحاظ معنایی اتومیک یا تجزیه شدنی باشد به این معنا که اگر مقدار آن را به اجزا تجزیه کنیم مقادیر جزئی حاصل فاقد مهنا باشد مثل نام دانشجو ولی صفت مرکب صفتی است که از چندین صفت ساده تشکیل شده باشد یعنی در صورت تجزیه اجزای بدست آمده دارای معنا باشد مثل آدرس

۲ -تک مقداری و یا چند مقدار : صفت تک مقداری صفتی است که برای یک نمونه از ی ک موجودیت فقط و یا حداکثر یک مقدار دامنه مقادیر می گیرد صفتی که بتواند بیش از یک مقدار را از دامنه مقادیر بگیرد صفت چند مقداری می گویند. مانند : شماره تلفن

۳ -صفت شناسه و یا ناشناسه: صفت شناسه موجودیت که گاه به آن کلیئ نیز گفته می شود صفتی است که دو وژگی زیر را داشته باشد.

الف) یکتایی معنا داشته باشد یعنی در هیچ دو نمونه از یک نوع موجودیت مقدارش یکسان نباشد که صفت شناسه عامل جدا کننده دو نمونه از یک موجودیت است.

ب) حداقل امکان طول مقادیرش کوتاه باشد.

نکت: هر موجودیت اگر ضعیف نباشد حداقل یک صفت شناسه دارد.

۴ -صفت هیچ مقدار پذیر و هیچ مقدار ناپذیر

هیچ مقدار یعنی ناشناخته یا مقدار غیر قابل اعمال یا مقدار تعریف نشده که در عمل بیشتر برخورد خواهیم کرد چرا که همیشه ممکن است مقدار یک صفت برای برخی از نمونه های یک نوع موجودیت ناشناخته یا ناموجود باشد.

نکته صفت شناسه موجودیت نمی تواند هیچ مقدار پذیر باشد زیرا عامل نیز نمونه های یک موجودیت خود نمی تواند ناشناخته باشد.

۵ -صفت ذخیره شده (واقعی) یا صفت مشتق (مجازی یا محاسبه شده)

صفت ذخیره شده سفتی است که مقادیر آن عیناً در پایگاه داده ذخیره شده باشد مثال : نام دانشجو – نمده درس

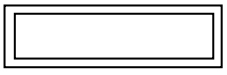
صفت مشتق صفتی است که مقادیرش در پایگاه داده ذخیره شده باشد بلکه حاصل یک عملیات بر روی داده های ذخیره شده باشد.

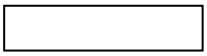
ارتباط : ارتباط رابطه ای است که بین موجودیت ها به عبارت دیگر تعامل بین ۲ یا چند موجودیت (و پایین یک نوع موجودیت و خودش) را ارتباط گویند .

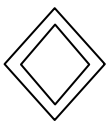
در واقع نوع ارتباط عملی است که بین انواع موجودیت ها جاری بوده هست و یا خواهد بود.

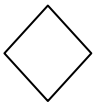
مثال : دانشجو درس را انتخاب کرد .

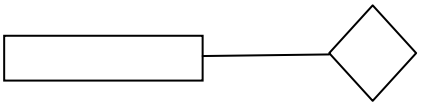
نمادهای رسم نمودار های ER:

نوع موجودیت ضعیف 

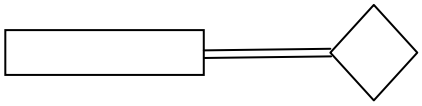
نوع موجودیت 

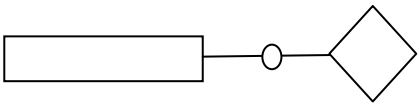
نوع ارتباط با موجودیت ضعیف 

نوع ارتباط 

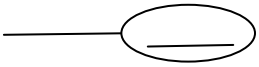
مشارکت نوع موجودیت در نوع ارتباط 

مشارکت الزامی

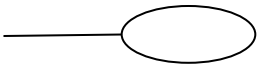




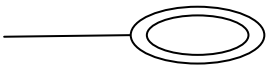
صفت شناسه اول



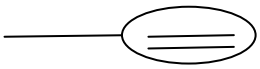
صفت



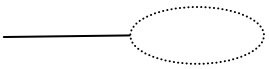
صفت چند مقداری



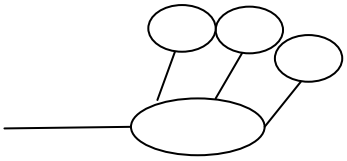
صفت شناسه دوم



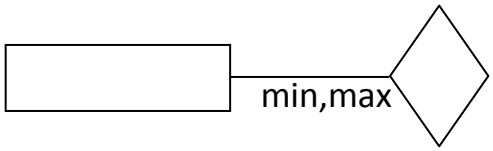
صفت مشتق



صفت مرکب

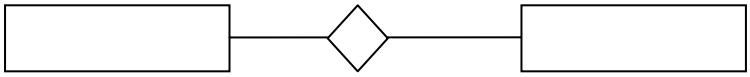


چندی ارتباط

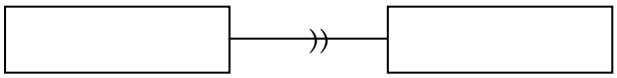


شناسه مرکب

چندی ارتباط نوع R

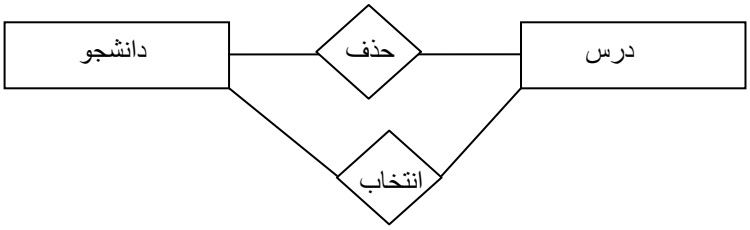


ارتباط گونه است از



خصوصیات نوع ارتباط

۱ -هر ارتباط یک نام دارد معمولا کلمه یا عبارت فعلی است که گاهی هم با یک عبارت مختصر که حاصل ترکیب حرف اول موجودیت ها است بیان می شود .



۲ -هر ارتباط یک معنای مشخص دارد و این معنا با معنای هر ارتباط دیگری متفاوت است معنای ارتباط انتخاب کردن با معنای ارتباط حذف کردن متفاوت است .

۳ -هر ارتباط نمونه هایی دارد .

مثال : انتخاب درس توسط دانشجو : نوع ارتباط

حسن حسین پایگاه داده را انتخاب کرد : نمونه ارتباط

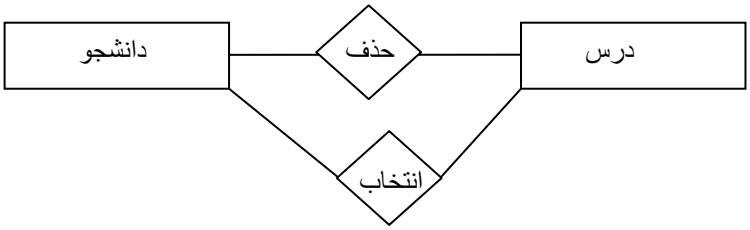
۴ -وضع مشارکت در ارتباط :

نکته : انواع موجودیت هایی که بین آنها یک ارتباط برقرار است شرکت کنندگان یا Portici pant در آن ارتباط نام دارد .

نکته : مشارکت یک نوع موجودیت در یک ارتباط ممکن است الزامی یا غیر الزامی باشد .

مشارکت یک نوع موجودیت در یک ارتباط را الزامی گویند اگر تمام نمونه های آن نوع موجودیت در آن نوع ارتباط شرکت کنند در غیر این صورت مشارکت غیرالزامی است .

مثال: مشارکت دانشجو در ارتباط انتخاب الزامی است اما مشارکت دانشجو در ارتباط حذف کردن الزامی نیست .

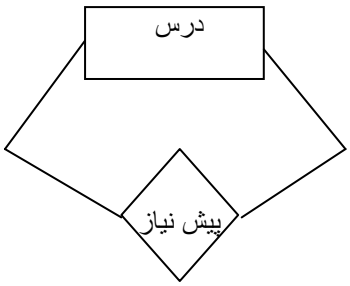


نکته : هر نوع موجودیت شرکت کننده در یک نوع ارتباط یک نقش (Role) مشخص را ایفا می کند مثلاً نقش موجودیت دانشجو در ارتباط انتخاب ما انتخاب کردن و نقش موجودیت درس انتخاب شدن است .

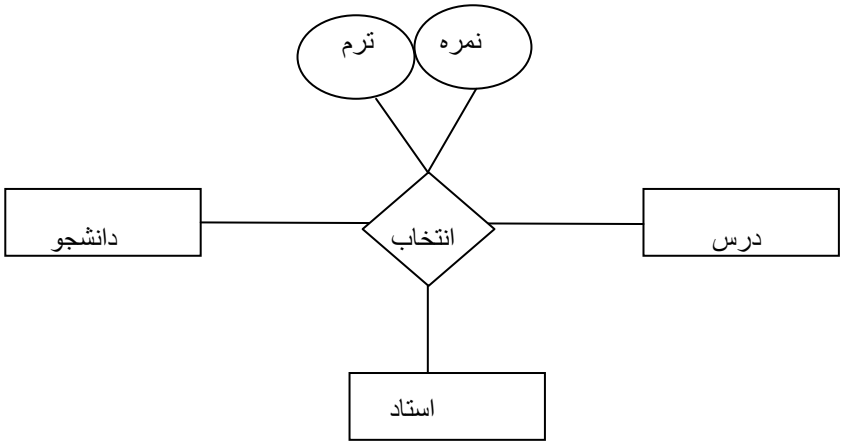
۵ -درجه نوع ارتباط Degree : تعداد نوع موجودیت های شرکت کننده در یک ارتباط را درجه ارتباط گویند و داریم:

Degree		تعداد نوع شرکت کنندگان
یگانی	unary	۱
	Binary	۲
چندگانی	n-ary	۳

مثال برای یگانی:



مثال برای ۳ گانی:



نکته : هر نوع موجودیت شرکت کننده در یک نوع ارتباط یک نقش Role مشخص را ایفا می کند .

مثلا می توان گفت دانشجو با شماره ۸۳۴۵۱۸۳ درس با کد ۱۱۱ را با اسناد ۲۳ در ترم اول سالتحصیلی ۸۷-۸۶ انتخاب کرده و نمره ۱۵ گرفته است .

نکته : نوع ارتباط نیز می تواند دارای صفت باشد .

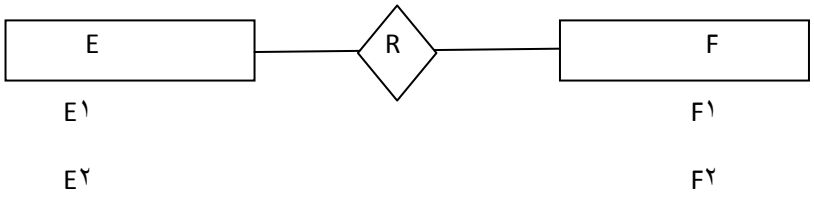
نکته : چندی یا ماهیت نوع ارتباط (کاردینالیتی ارتباط) : بین دو نوع موجودیت عبارت است از چگونگی تنازل بین دو مجموعه نمونه های آن دو نوع موجودیت و در ۳ نوع است .

تنازل یک به یک ۱:۱

تنازل یک به چند ۱ : N

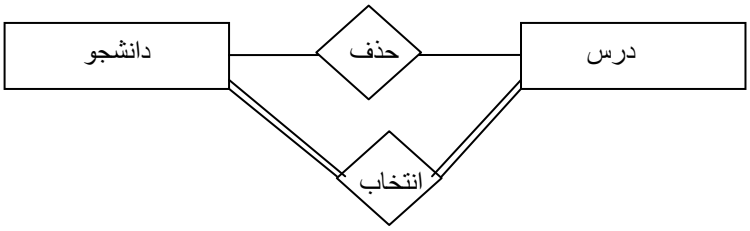
تنازل چند به چند V : M

اگر دو نوع موجودیت E و F را در نظر بگیریم در ارتباط یک به یک E به F یک نمونه از E حداکثر با یک نمونه از F و یک نمونه از F حداکثر با یک نمونه از E در ارتباط است .

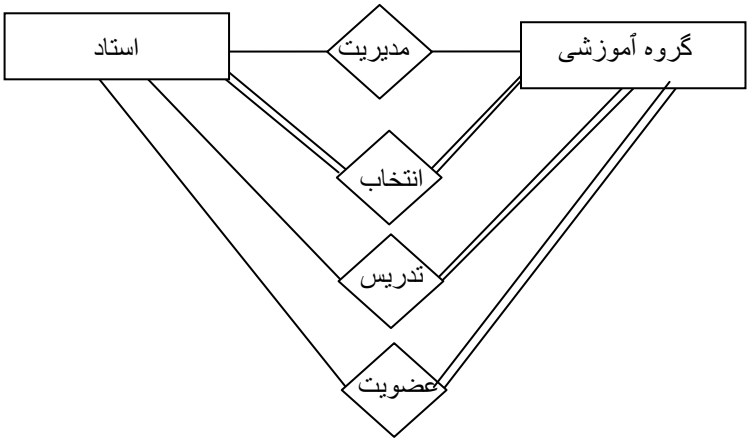


در ارتباط یک به چند (از سوی E به F) یک نمونه از E با تعدادی از نمونه های F در ارتباط است ولی یک نمونه از F حداکثر حداکثر با یک نمونه از E ارتباط دارد .

در ارتباط چند به چند یک نمونه از E با تعدادی از نمونه F ارتباط دارد و یک نمونه از F با تعدادی از نمونه های E ارتباط دارد .



با فرض اینکه یک دانشجو فقط یک بار می تواند حذف کند.

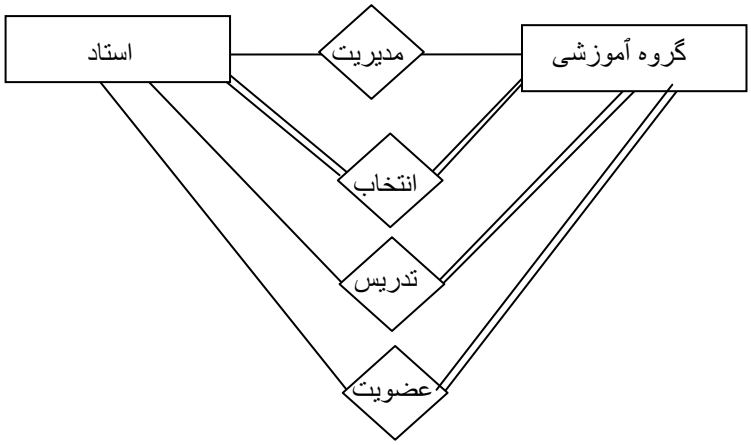


نکته : برای نمایش چندی ارتباط در نمودار ER روش دیگری هم وجود دارد در این روش به هر مشارکت یک نوع موجودیت E در ارتباط R یک جفت عدد صحیح به صورت (min , max) منتخب می شود که در آن

۰ <= min <= max

Max >= ۱

معنا این است که در هر لحظه هر نمونه e از نوع موجودیت E باید حداقل در min و حداکثر در max نمونه از ارتباط R شرکت داشته باشد اگر min=۰ باشد یعنی مشارکت غیر الزامی است و اگر min>۰ باشد یعنی مشارکت الزامی است .



نکته : هرکجا < ۱ باشد آنجا الزامی است .

نکته مهم : چندی نوع ارتباط و درجه نوع ارتباط – وضع مشارکت در یک نوع ارتباط از جمله محدودیت های معنایی در محیط هستند و به طوریکه خواهیم دید و چگونگی طراحی منطقی پایگاه داده تاثیر دارد.

مراحل مدل سازی معنایی داده ها با روش ER به قرار زیر می باشند .

۱ -مطالعه – تحلیل و شناخت محیط

۲ -برآورد خواسته ها و نیازهای اطلاعاتی و پردازشی همه کاربران و تشخیص محدودیت های معنایی و قواعد فعالیت های محیط

۳ -فهرست کردن نیازها و خواسته های کاربران

۴ -بازشناسی انواع موجودیت های مطرح در محیط و تامین وضع هر نوع موجودیت

۵ -تعیین مجموعه صفات هر نوع موجودیت و جنبه های هر صفت

۶ -بازشناسی انواع ارتباطات بین انواع موجودیت ها و تشخیص الزامی یا غیرالزامی بودن مشارکت در آنها و تشخیص هر ارتباط .

۷ -رسم نمودار ER به صورت واضح و خانا

۸ -فهرست کردن پرسش هایی که پاسخ آنها از نمودار ER به دست می آید (این پرسشها بیانگر نیازهای کاربران است) .

۹ -وارسی مدل های انجام شده تا اطمینان حاصل شود که مدل سازی پاسخگوی تمام نیازهای کاربران است .

مثال: نمودار ER برای یک کمپانی طراحی کنید که نیازهای آن به صورت زیر است .

۱ -نیازها در سطح اول معطوف کارمندان شرکت – بخش ها و پروژه ها است .

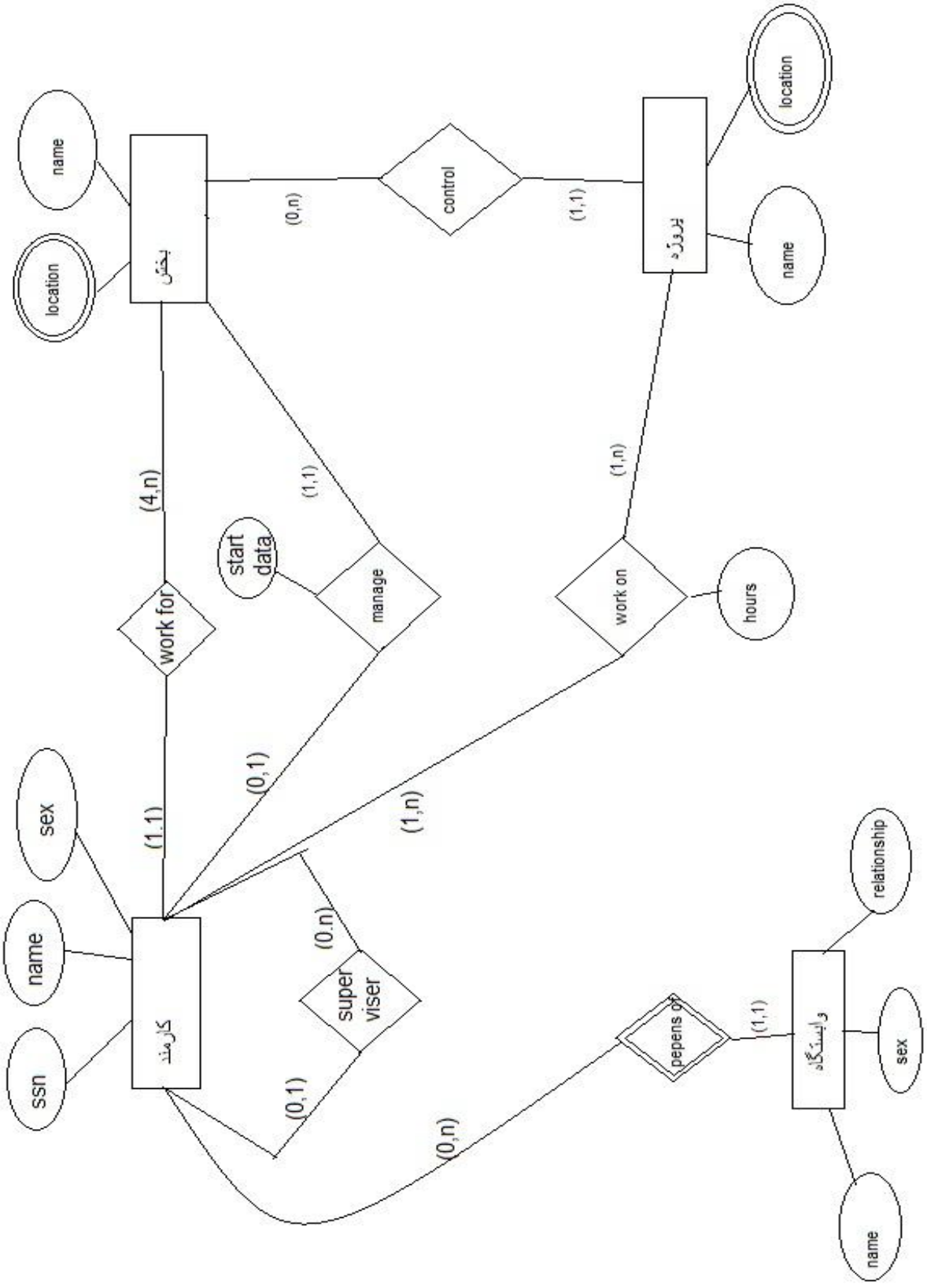
۲ -کمپانی شامل تعدادی بخش اس ت که هر بخش نام انحصاری – شماره انحصاری محل بخش و یک کارمند مشخص آن را مدیریت می کند و حداقل کارمندی که برای تشکیل بخش نیاز است ۴ کارمند می باشد.

۳ -هر بخش تعدادی پروژه مدیریت می کند که هر پروژه صفات چون نام – شماره و محل دارد .

۴ -هر کارمند به یک بخش منصوب می شود ولی ممکن است روی چند پروژه کار کند .

۵ -تعداد ساعات کارکرد هر کارمند روی هر پروژه همچون مدیر هر کارمند را می بایست در سیستم مشخص کند .

۶ -متخصصین یا وابستگان هر کارمند را برای امور بیمه نیاز داریم .


name
location
بخش
(0,n)
control
(1,1)
پروژه
location
name
(4,n)
work for
(1,1)
start data
manage
(1,n)
work on
hours
sex
name
ssn
کارمند
(1,1)
(0,1)
(1,n)
(0,n)
super viser
(0,1)
(0,n)
pepens on
(1,1)
وابستگان
relationship
sex
name

فصل سوم

آشنایی با ساختار داده ای

همانطور که در مدل سازی معنایی داده ها نیاز به امکانی برای نمایش واقعیات داریم برای طراحی منطقی پایگاه داده ها نیز امکانات خاصی لازم است بی تردید این امکان بایستی ساخت انتزاعی باشد به عبارت دیگر می بایست یک مدل داده ای دارای ساختار داده ای باشد همین ساختار داده ای (Data structure) = DR

از یک مدل داده ای است که تعیین کننده محیط انتزاعی پایگاه داده و نیز چارچوب طراحی منطقی آن می باشد به بیان دیگر طراحی منطقی یک پایگاه داده با استفاده از مفاهیم اساسی یک مدل داده ای و در چارچوب ساختار داده ای آن مدل انجام می شود .

مدل داده ای = ساختار داده ای

انواع ساختارداده ای { رابطه ای ، سلسله مراتبی ، شبکه ای }

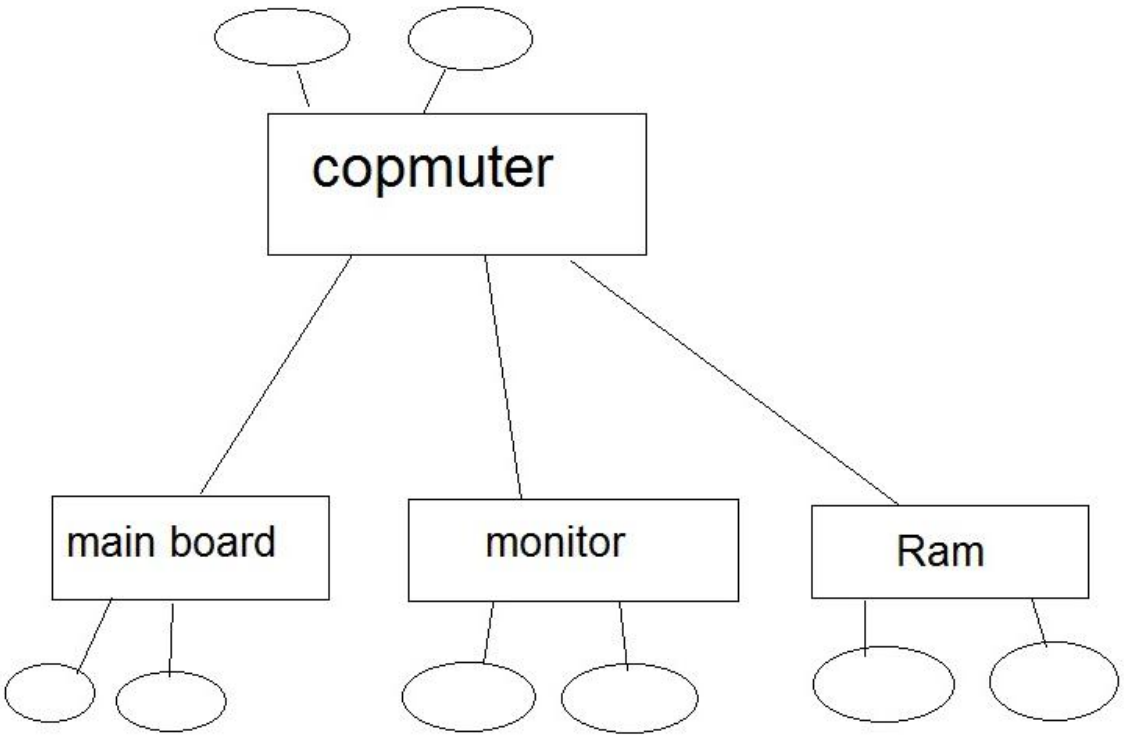
توجه : بر اساس این ساختار داده ای بویژه ساختار داده ای DBMS ها طراحی و تولید شده اند.

ساختار داده ای EER[1]

مفاهیمی که در مدل سازی شی گرا وجود دارد مثل تجزیه شی گرا به اشیاء جزء رسیدن به شی کل تخصیص – تعمیم – وراثب – تجمع را نمی توان با روش ER بدلیل کمبودهایی که در آن وجود دارد پیاده سازی نمود لذا روش EER توسط پژوهشگران مطرح شد.

[1] Extended ER

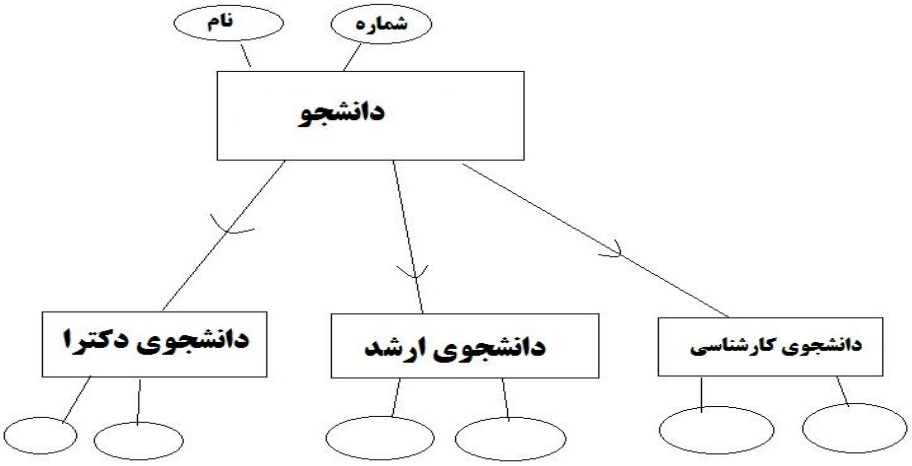
تجزیه و ترکیب



بحث ترمیم (گونه ای است از...)

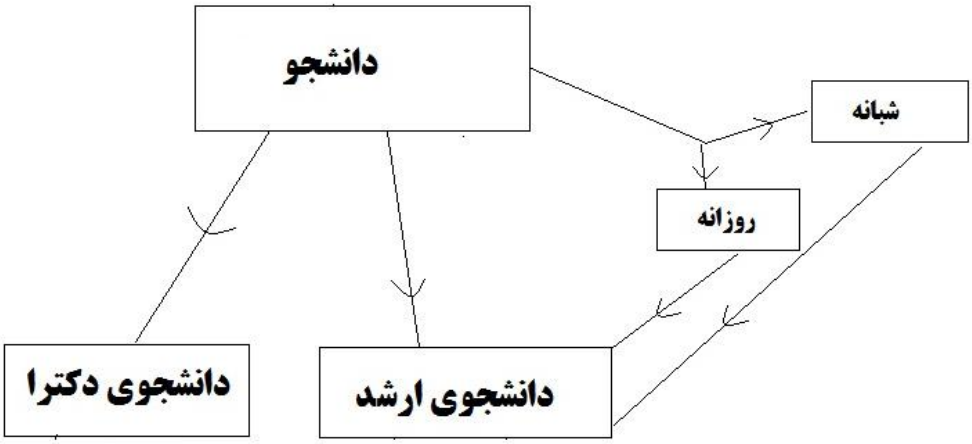
تخصیص عبارت است از مشخص کردن گونه ای خاص از یک شی مثلاً موجود زنده سه گونه انسان – حیوان – نبات دارد

نکته: هر زیر نوع صفات خاص خود را دارد ولی صفات زیر نوع خود را نیز به ارث می برد



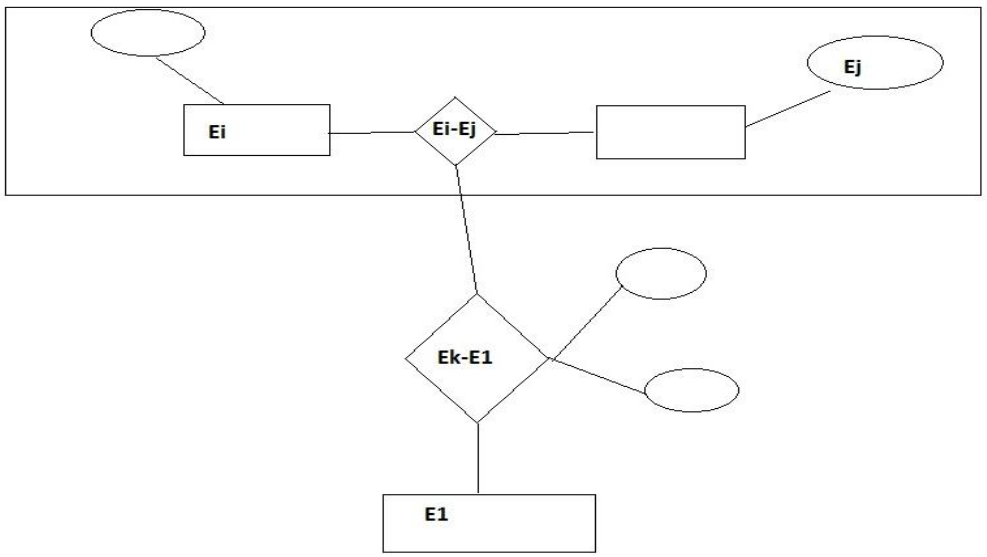
نکته : تعمیم عکس عمل تخصیص است به این معنا که با داشتن زیر نوع های خاص و صفات مشترک بودن بین آنها می توان زبر نوع و صفات آن را استخراج نمود.

نکته : با توجه به اینکه یک زبر نوع موجودیت می تواند زبر نوع موجودیت دیگری باشد بدین ترتیب می توان مفهوم وراثت چند گانه را نیز در روش ER مطرح کرد.

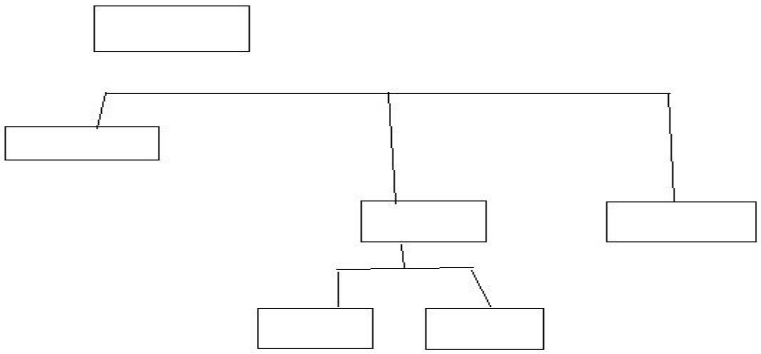


تجمع

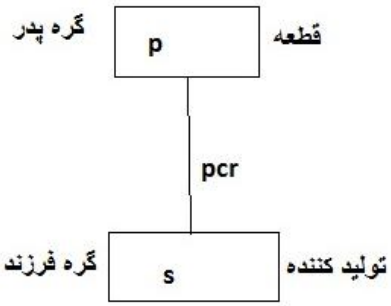
تجمع عبارت است از ساختن یک موجودیت جدید با در نظر گرفتن دو یا بیش از ۲ نوع موجودیت که خود با هم ارتباط دارند بصورت یک نوع موجودیت واحد بطور شهودی می توان گفت زمانی از تجمع استفاده می کنیم که بخواهیم ارتباطی را بین ارتباط بیان کنیم.



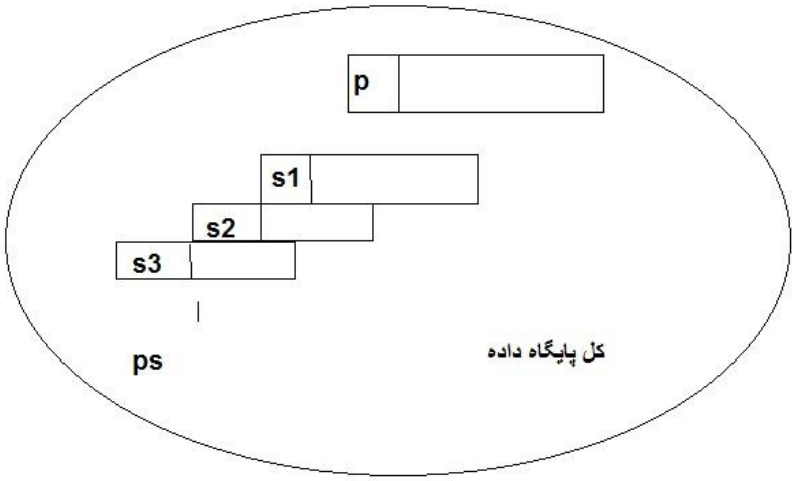
بررسی مختصری راجع به DS سلسله مراتبی



ارتباط پدر فرزندی [PCR[1]]



اساساً DS سلسله مراتبی برای مدلسازی داده ای ۱:n بکار می رفت و هر سلسله مراتب تعدادی نمونه دارد که مثلاً اگر سلسله مراتب PS است آنگاه نمونه ها عبارتند از:



[1] Parent child relationship

مثال : Query[1] :

چاپ کن لیست تولید کننده قطعه P۷

۱ – برو به سلسله مراتب PS

۲ – بیا نمونه ای از نمونه سلسله مراتب با ریشه p=p۷

۳ – چاپ کن فرزندانش را

مثال Query :

تولید کننده s۲ چه قطعاتی را تولید می کند؟

یک راه حل این است که از سلسله مراتب sp استفاده کنیم اما وجود سلسله مراتب sp برای پاسخ گویی به این پرسجو لازم نیست در سورتی که ساختار sp داشتیم روال پاسخ گویی به گونه زیر است:

۱ – برو به س سله مراتب ps

۲ – بیا به سلسله مراتب با ریشه s= s۲

۳ – چاپ کن فرزندانش را

در صورت عدم وجود ساختار ps

۱ – برو به سلسله مرراتب ps

۲ –به ازای تمام سلسله مراتب ها عمل زیر را انجام بده

۳ – فرزندان هر نمونه را بررسی کن اگر فرزندان s= s۲ دارد چاپ کن p# پدرش را

[1] پرسجو

شمای ادراکی(تعریف منطقی):

فرض کنید می خواهیم سلسله مراتب PS را تعریف کنیم:

نام سلسله مراتب ps

گره ریشه نمونه P

رکورد p# : p ۶ کاراکتر

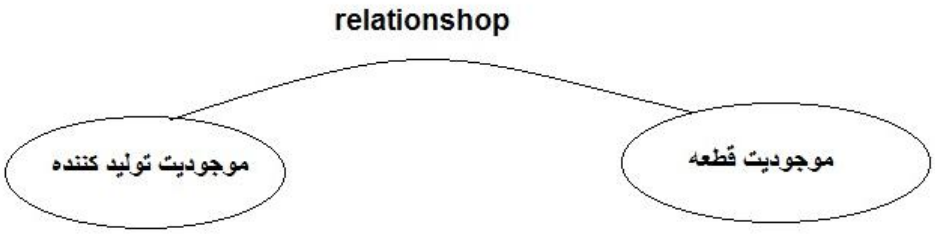
رکورد p# : name ۵۰ کاراکتر

رکورد p# : sname ۶۰ کاراکتر

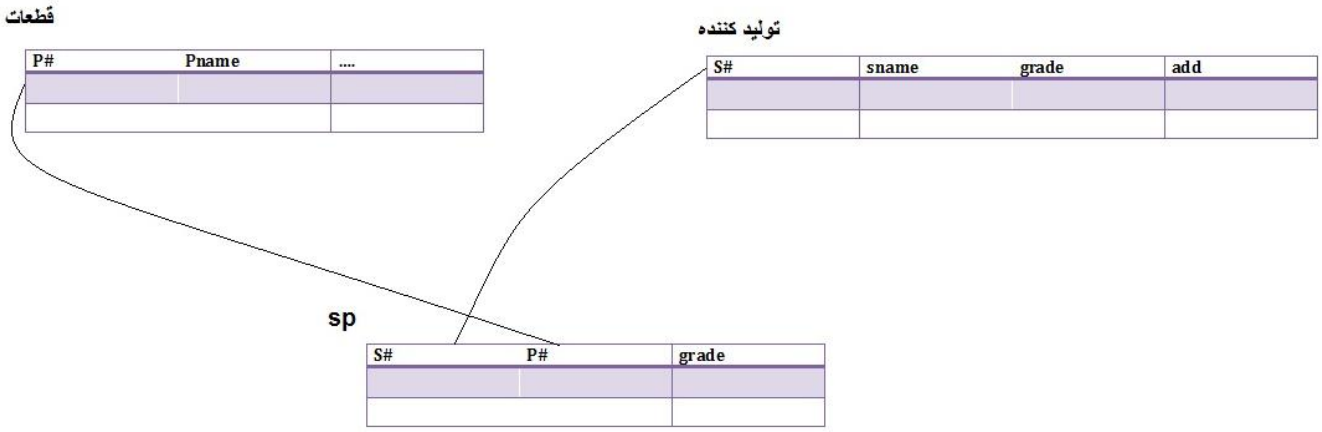
ساختار داده ای رابطه ای

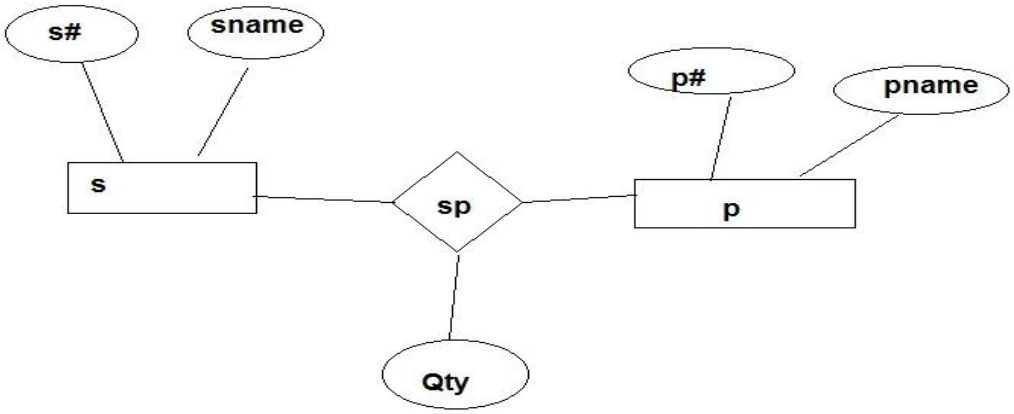
رابطه مفهومی ریاضی که از دید کاربر رابط نمایش جدولی دارد و پایگاه داده رابطه ای مجموعه ای از تعدادی جدول یعنی عنصر ساختاری اساسی در مدل رابطه ای جدول است . سه عنصر اصلی در این ساختار داده ای عبارت است از جدول – سطر جدول و ستون جدول . به عبارت دیگر مفاهیم اساسی بانک اطلاعاتی در تبدیل به آن در مدل رابطه ای مطابق جدول زیر است .

مفاهیم اساسی بانک اطلاعاتی	معادل آن در موارد رابطه ای	نحوه نمایش
موجودیت	رابطه Relation	جدول table
نمونه موجودیت	Topple	سطر جدول Row
صفت	Attribute	ستون جدول Column
ارتباط	–	توسط کلید (اصلی – خارجی)



همان طور که گفته شد ps رابطه ای نمایش جدولی دارد بنابراین موجودیت در نمودار ER بر یک یا چند جدول تبدیل می کند.



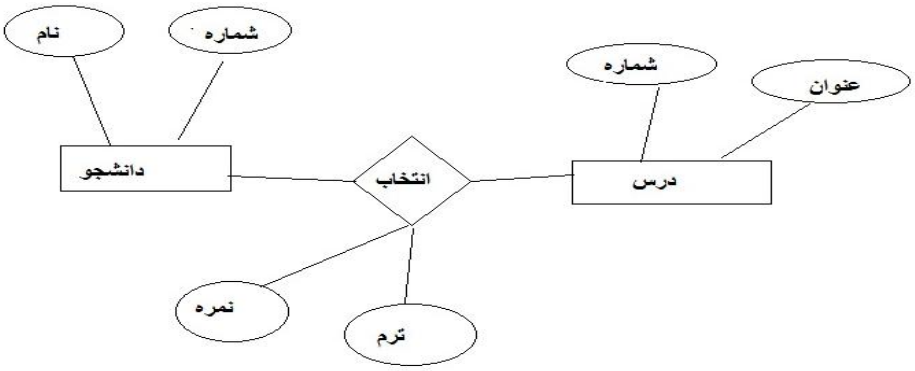


اساساً مدل رابطه ای برای مذلینگ رابطه ۱ به n است و برای رابطه های n به n یک جدول ثالث ایجاد می شود همان طور که در مثال فوق ذکر شد هو نوع موجودیت مستقل و با مشارکت غیر الزامی با یک نوع جدول نمایش داده شد هر سطر از ج دول s یک نمونه متمایز از موجودیت تولید کنندگان و هر سطر از جدول p یک نمونه متمایز از موجودیت قطعات است هر نوع ارتباط چند به چند دو نوع موجودیت مستقل و با مشارکت غیر الزامی در ارتباط را با یک نوع جدول نمایش می دهیم بنابراین در مثالفوق نیاز به جدول سوم بنام sp بود.

و هر سطر از این جدول بیانگر نمونه ای از ارتباط بین تولید کننده از قطعه می بود همچنین شایان ذکر است برای پیاده سازی جداول نشان داده شده است.

نکته: رابطه اطلاعات داخل جداول است یعنی مجموعه ای از سطرها مثلاً صفت pname است و جاهد مطلق مقدار صفت pname در آن نوع نمونه موجودیت می باشد .

مقدماتی از طراحی پایگاه داده جدولی



نکته: هر نوع موجودیت مستقل و با مشارکت غیر الزامی در یک نوع ارتباط را با یک نوع جدول نمایش می دهیم.

STUDENT

STID	STNAME	STDEG	STMJR	STDEID
۸۶۰۷۶۷۱۵۳	امیر	کاردانی	کامپیوتر	۲۵۰۱

COURSE

COIP	COTITLE	CREDIT	COTYPE	CODEID
۸۳۱	پایگاه	۳	تخصصی	۱۰

نکته : هر سطر از جدول student یک نمونه متمایز از نوع موجودیت دانشجو و هر سطر از جدول course یک نمونه متمایز از نوع موجودیت درس می باشد .

نوع موجودیت= دانشجو نوع موجودیت = درس

نمونه موجودیت = امیر نمونه موجودیت = پایگاه

نکته : هر نوع ارتباط چند به چند بین دو نوع موجودیت مستقل با مشارکت غیر الزامی در ارتباط را با یک نوع جدول نمایش می دهیم بنابراین در مثال ذکر شده نیاز به جدول سوم داریم که هریک ا ز سطر این جدول نشان دهنده یک نمونه از یک نوع ارتباط انتخاب است .

stcot

Slid	Coid	Term	Year	Grade
۸۶۰۷۶۷۱۵۳	۸۳۱	۱	۸۸	۲۰
۸۶۰۸۱۸۰۰۰	۴۱۷	۲	۸۷	۱۸
۸۵۰۶۱۸۰۰۰	۸۳۱	۱	۸۸	۱٤

نکته : **شِمای پایگاه جدولی :**

برای نوشتن شِما هریک از جداول را باید به سیستم معرفی کنیم (تعریف شِمای ادراکی) به عبارت دیگر نام جدول – مشخصات هر ستون و ... را می بایستی برای سیستم تعریف کنیم روشن است که یک سری احکام و دستورات تعریف جدول نیاز داریم ما در این درس از امکانات زبانی استفاده می کنیم که موسوم به زبان SQL که زبان استاندارد سیستم های رابطه می باشد .

SQL = structured Query Language

Create Table Student

(stid char ***** Not null

Stname char (۱۲)

Stmjr char (۲۰)

Stdeid char (۴) ,)

Primary key siid

Crieate table vourse

(cold char (۶) notnull

Cotitle char (۱۰)

Creadit small int

Cotype char (۱)

Codzed char (۴) ,)

Primary key coid

Create tabe stcot

(stid char (۸) not null ,

Coid char (۶) not null ,

Tr char (۱) not null ,

Year char (۵) not null ,

Grade cecimal (۲,۳) ,)

Primary key (stid , coid , tr , year) , کلید داخلی

Foreign key stid references student (stid) , کلید تاجی

Foreign key coid references coerce (coid) , کلید تاجی

مجاری پایگاه داده :

بعد از سال ۱۹۸۰ توسط موسسه ANSII معماری برای پایگاه داده ها ارائه شد رعایت این معماری در طراحی پایگاه داده ها الزامی است این معماری به معماری ۳ سطحی و پایگاه ۴ سطحی مشهور است که سطوح این معماری عبارتنداز :

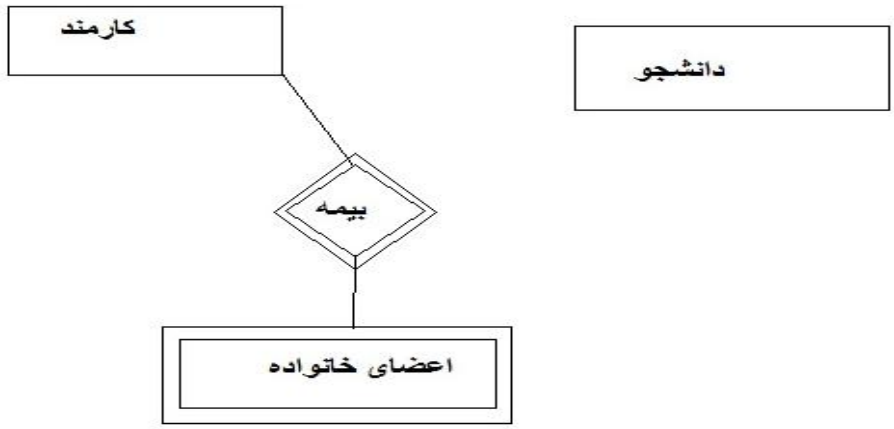
۱ - سطح خارجی (دید خارجی) external view

۲ - سطح ادراکی (دید ادراکی) con coptud view

۳ - سطح داخلی internal view

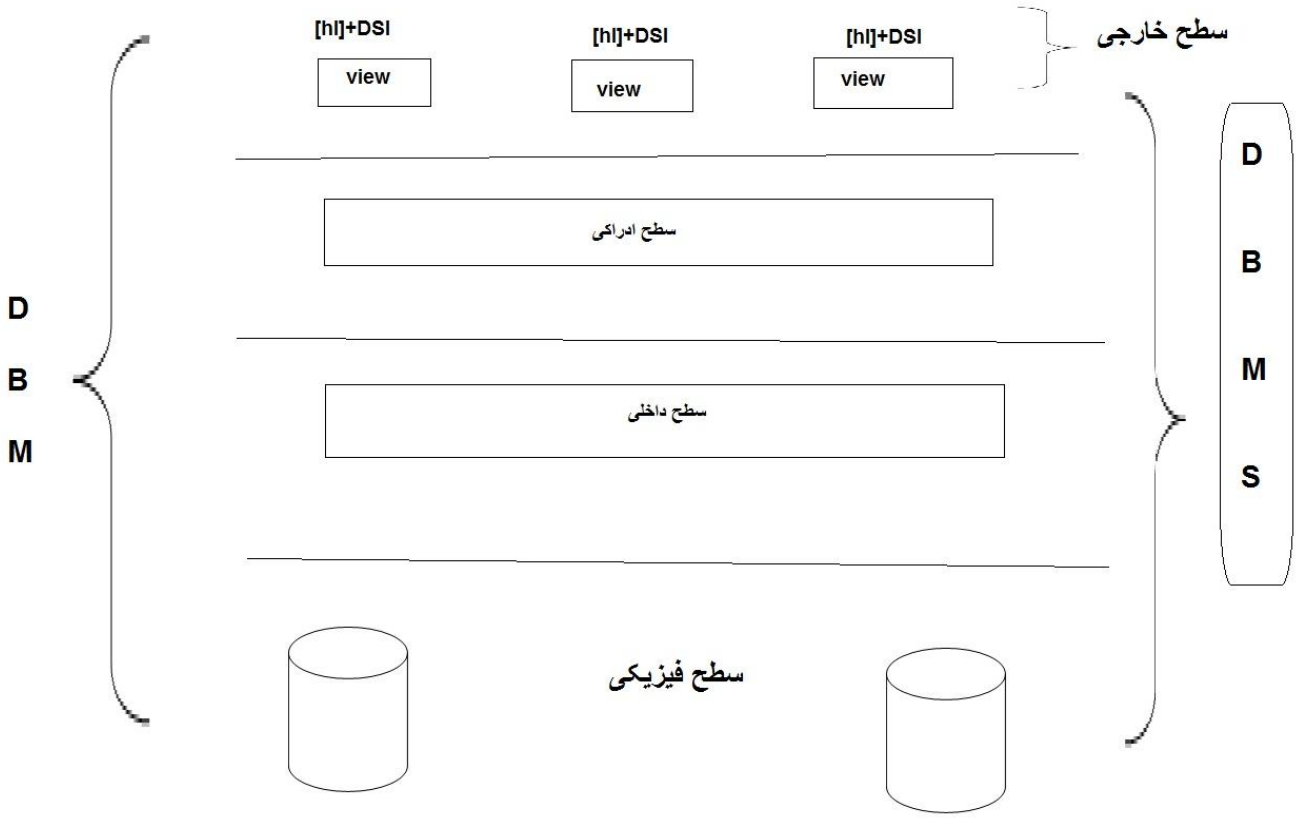
۴ - سطح فیزیکی physical view

فرق موجودیت قوی با ضعیف



برای بیمه کردن اعضای خانواده اگر کارمند از بین برود اعضای خانواده دیگر مفهومی ندارد پس موجودیت ضعیف است .

معماری پایگاه داده



اجزای معماری پایگاه داده :

۱ - Host language : HL

۲ - Data sub language : DSL

۳ - سطح خارجی

۴ - سطح ادراکی

۵ - سطح داخلی

۶ - سطح فیزیکی

۷ - DBMS

۸ - Mapping

۹ - DBA

سطح ادراکی : دید طراح است از کل داده های ذخیره شده در پایگاه داده (کل داده های محیط عملیاتی) دیدی جامع است یعنی دربرگیرنده تمام نیازهای کاربران محیط می باشد . سطح ادراکی باید تعریف شود به تعریف سطح ادراکی اصطلاحاً شِمای سطح ادراکی یا conceptual schema می گویند . (create ها که قبلا شده ایجاد جدول) سطح ادراکی در کادر مفاهیم یک Ds (Data structure) تعریف و ارائه می شود .

DS ← رابطه ← جدول
سطر
ستون

DS تامین کننده انتزاع است. شِمای ادراکی می بایست همانند یک برنامه و در غالب مجموعه ای از احکام تعریف و پس از کامپایل در جایی به نام کاتالوگ ذخیره می شود .

سطح خارجی : دید کاربر خاص است از کل داده های ذخیره شده مبتنی بر یک DS مشخص (معمولا DS سطح ادراکی) یعنی ممکن است در برخی موارد خاص DS سطح خارجی با DS سطح ادراکی متفاوت باشد .

سطح خارجی می بایست تعریف شود که به تعریف آن شِمای خارجی گفته می شود شِمای خارجی نیز مجموعه ای است از احکام که پس از کامپایل که در کاتالوگ سیستم ذخیره می شود .

Sc	Avv ayr	Na me
S_1	۱۲	Sn_1
S_2	۱۸	Sn_2
S_3	۱۹	Sn_3

Code	Town
S_1	C_1
S_2	C_2
S_3	C_3

$s \neq$	Sname	City	Addr	****	Avy
S_1	Sn_1	C_1	Y_1	۱۰	۱۲
S_2	Sn_2	C_2	Y_2	۱۳	۱۸
S_3	Sn_3	C_3	Y_3	۱۰	۱۹

تعریف شِمای خارجی :

U_1V_1 : ستون های code و town تعریف شده بر روی جدول student ستونهای (S # و City)

نام دید: U_1V_1

ستونهای average و name

تعریف شده بر روی جدول : student ستونهای (s# و avg وsname)

نکته : یک دید می تواند بین چندین کاربر مشترک باشد .

نکته : یک کاربر می تواند چندین دید یا view داشته باشد .

نکته : یک view لزوما روی یک جدول تعریف نمی شود بلکه ممکن است یک view ستونهایش را از چندین جدول بگیرد یعنی اینکه روی چند جدول تعریف شود .

نکته : یک view ممکن است روی یک view دیگر تعریف شود یعنی view می تواند روی view سطح ادراکی ها جدول سطح ادراکی یا view خارجی تعریف شود سطح ادراکی می تواند view داشته باشد که خود این view یا روی view سطح ادراکی و یا روی جدول سطح ادراکی تعریف می شود .

سطح داخلی : همان محسط منطقی فایلینگ است و در این سطح نمی دانیم فایل ها در کجا و چگونه ذخیره شده اند تنها می دانیم که مثلا داده های جدول student شاخص داریم چرا که خود این سطح تقاضای ایجاد چنین فایلی را داده است (مبحث access metoth مطالعه شود) .

سطح فیزیکی : محیط ذخیره شده داده ها و شامل رسانه های ذخیره سازی مانند Teip (نوار) و دیسک .

HL : (haste language) : هر زبان برنامه نویسی متفاوت در تکنولوژی پایگاه داده ها می تواند به عنوان زبان میزبان مطرح شود از زبان میزبان می توان برای تولید سیستم های نرم افزاری روی پایگاه استفاده شود برای اینکه یک زبان برنامه نویس به عنوان haste language مطرح شود می بایست هم زبان وهم CBMS امکاناتی را فراهم نمایند همچنین برای تولید یک نرم افزار پایگاه داده ای (نرم افزاری که با یک پایگاه داده کار کند) لزوما نیازی به استفاده از HL نداریم و این بستگی به وضعیت DSL دارد .

از جمله HL ها می توان به CH و C و C# و VB و VBNET و ASP و ... نام برد .

Data SUB language : DSL

DSL مجموعه از احکام که امکان می دهد :

۱ - کاربر پایگاه خود را تعریف کند و این بخش از احکام DSL فقط برای تعریف هستند .

DDL: Data Defection language

۲ - کارگاه با پایگاه خود کار کند .

DNL: Data manipulation language

۳ - کاربر بر روی داده های خود کنترل داشته باشد .

DCL: Data control language

اینکه کاربر تا چه حدی پایگاه داده اش را بتواند کنترل کند به نوعی کاربر و جایگاهش بستگی دارد.

نکته خیلی مهم :

ویژگی های DSL: DSL در کادر مفاهیم یک DS است .

Ds رابطه ای { ۱ – جدول ۲ – سطر ۳ – ستون }

۱ – مثلا اگر DS رابطه است DSL می بایست در کادر مفاهیم رابطه باشد یعنی DML مربوط به ان احکامی مثل سطریاب – ستون یاب و ... داشته باشد یا مثلا در این صورت DDL مربوط به ان احکام مثل تعریف جدول داشته باشد .

۲ – DSL می بایست اکمال ساختاری داشته باشد یعنی اینکه در کادر مفاهیم DS هر نمونه ای که از نظر تئوریک تعریف باشد در عمل نیز DSL آن را پشتیبانی نماید.

۳ – اصل و مدت احکام رعایت شود یعنی برای یک عمل منطقا واحد یک دستور یا قدرت عملیاتی مناسب بهتر از چند دستور مختلف است .

۴ – برای امکان تعداد دستورات کم باشد لزوما افزایش تعداد دستورات برای یک DSL ، حُسن محسوب نمی باشد .

۵ – احکام DSL ساده و فراگیر آنها ساده باشد .

۶ – DSL اکمال برنامه سازی داشته باشد یعنی اینکه ساختارهای متعارف زبانهای برنامه نویسی را داشته باشد.

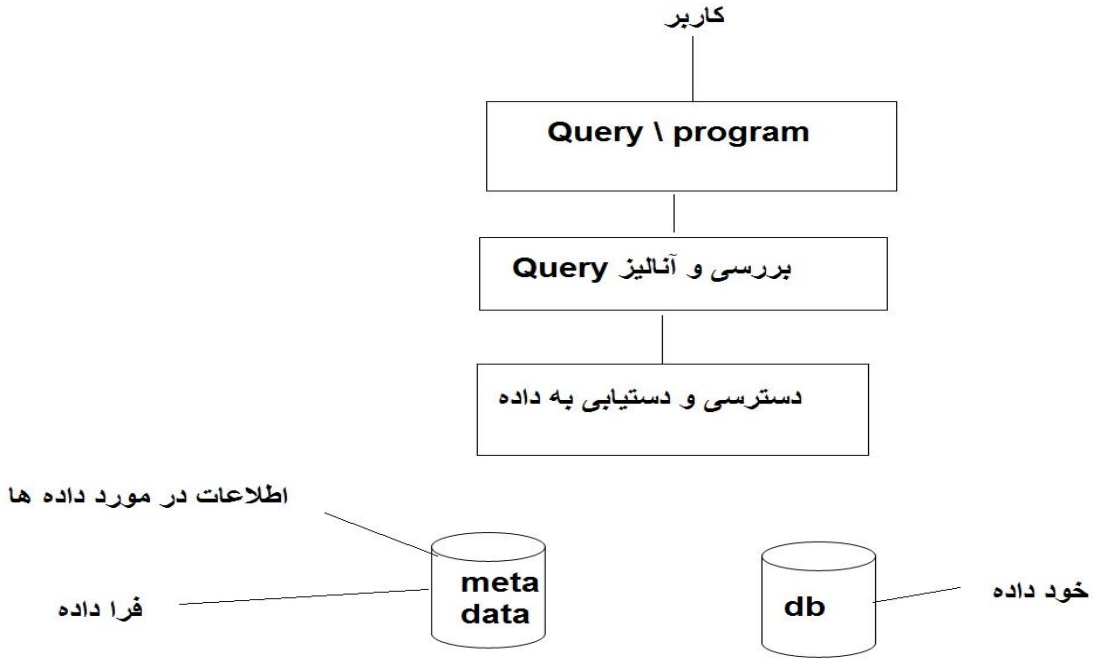
DBA : مدیر پایگاه دادهData Base administrator

در یک سیستم کوچک DBA یک فرد است و در سیستم های واقعی یک تیم است .

ترکیب تیم :

۱. DBA : فردی است که کاملا بر DBMS تسلط دارد فرد کامپیوتری که اگر اراده کند کل پایگاه را از بین می برد .
۲. DA : یک فرد سازمانی است که بر داده های سازمان تسلط دارد .
۳. مسئول تیم های برنامه سازی : مدیریت اجرایی دارد و عملیات اشکال زدایی و رفع مشکلات برنامه ساز به عهده این فرد می باشد .
۴. مسئول مستند سازی
۵. مسئول تماس با کاربران محیط جهت تضمین خدمات .

نحوه درخواست کاربر



معمولاً meta data در data dictunery کاتالوگ سیستم ذخیره می شود.

از جمله مواردی که قبلا نیز اشاره شد در کاتالوگ سیستم ذخیره می شود می توان به شِمای ER ادراکی اشاره کرد .

نحوه اجرای درخواست کاربر :

۱. بررسی صحت Query کاربر
۲. بررسی سطح خارجی کاربر
۳. بررسی سطوح ادراکی : در این مرحله می خواهیم بین سطح خارجی و ادراکی mapping انجام دهیم و می خواهیم مکانیزم های mapping در سه سطح را اعمال کنیم .

بررسی سطح داخلی

اجرای درخواست کاربر بر روی سطح فیزیکی { خواندن / نوشتن

خواندن : دید کاربر بر عینیت درآمده است .

عینیت { مستقیم :
غیر مستقیم :

نوشتن : درج – حذف – به هنگام سازی insert- Delete – update

تعریف استدلال داده ای : Data independency

مصونیت دیدها و برنامه های کاربردی در سطح خارجی در قبال تغییرات در سطوح زیرین را استقلال داده ای گویند .

اگر تغییرات در سطح ادراکی باشد استقلال داده ای منطقی و در صورتی که تغییرات در سطح داخلی صورت گیرد استقلال داده ای گویند .

تعریف استقلال داده ای فیزیکی :

مصونیت دیدها و برنامه های کاربردی در قبال تغییرات در سطوح فیزیکی داخلی مثلا نوع رکورد فیلد / بلوک و ... را استقلال داده ای فیزیکی می نامند .

همواره در یک DBMS استقلال داده فیزیکی تامین می کند چرا که کاربر در یک محیط انتزاعی عملی می کند همچنین سطح ادراکی سطح واسط بین سطح خارجی و فیزیکی است که اثرات احتمالی را در خود هضم می کند .

استقلال داده ای منطقی :

مصونیت دیدها و برنامه های کاربردی در قبال تغییرات در سطح ادراکی را استقلال داده ای منطقی می نامند توجه به این نقطه ضروری است که استقلال داده ای منطقی حساسیت بالایی دارد چرا که سطح ادراکی به طور مستقیم و بی واسطه با سطح خارجی در ارتباط است لذا در توسعه سطح ادراکی و همچنین سازمان دهی مجدد آن می بایست توجه زیادی به تامین استقلال داده ای منطقی صورت گیرد .

مدل رابطه ای :

در سال ۱۹۷۰ توسط ***** ابداع و معرفی شد در واقع کاد در جست و جوی یک ساختار داده ای بود که دارای انتزاع قوی ای باشد تا محیط انتزاعی قوی تر و از آنجا استقلال داده ای تامین شود می دانیم که ساختارهای ریاضی

اساسا انتزاعی هستند و کاد هم به سراغ ریاضیات رفت ساختو داده ای در مدل رابطه ای یک مفهوم ریاضی است به نام رابطه یک رابطه مجموعه است از عناصر به شکل زیر :

D۱ D۲ D۳ ... Dn

<di۱ , di۲ , di۳ , ... , din>

Di۱ € D۱

Di۲ € D۲

Di۲ € D۳

...

Din € Dn

تعریف کاد از رابطه :

فرض وجود n مجموعه sn و....s۲ و s۱ نلزوماً متمایز رابطه R تعریف شده روی n مجموعه ، مجموعه ای است از n تایی هایی که از این پس تاپل می نامیم به نحوی که جزء اول هر n تایی از $s_۱$ و جزء دوم از $s_۲$ و جزء n ازsn قرار بگیرد به بیان دقیق تر r عبارت است از زیر مجموعه ای از ضرب کارتزین و در این صورت می گوییم رابطه R از درجهN می باشد یا فرض وجود N میدان $D_۱$ و $D_۲$ و .... D_N نلزوماً متمایز رابطه R از ۲ قسمت تشکیل شده است .

۱. سر آیه یا عنوان یا heading : مجموعه ای نامدار است از n صفت به صورت Ai:Di که در ان هر Ai نام صفت است و Di نام میدانی است که آن صفت از ان میدان قرار می گیرد .
۲. بدنه / پیکر / Body : مجموعه ای است از n تاپل T که زیرمجموعه ای از ضرب کارتزین($D_۱$ و $D_۲$ و .... D_N) می باشد.

مجموعه عنوان / heading : یک مجموعه نامدار است و ماهیت و ذات رابطه را مشخص می کند .

مجموعه body مجموعه ای است از تاپل ها که دائما در حال تغییر می باشد بسط یک رابطه که همان کل بدنه یک رابطه درباره زمانی مشخص است همواره در حال تغییر می باشد .

نکته ۱ : تعداد صفات خاصه (تعداد ستونها) درجه رابطه نامیده می شود .

نکته ۲ : تعداد عناصر مجموعه body یا همان تعداد تاپل ها که دائما در حال تغییر است کاردینالیتی رابطه نامیده می شود .

نکته ۳ : روش نمایش رابطه جدول است .

نکته ۴ : رابطه فاقدنظم است بدین معنی که هم در مجموعه heading و هم در مجموعه body ترتیب سطرها و ستونها اهمیتی ندارد .

نکته ۵ : در رابطه عنصر تکراری وجود ندارد یعنی در مجموعه heading دو صفت خاصه هم نام وجود ندارد و در مجموعه body نیز دو سطر یکسان وجود ندارد .

نکته ۶ : مقادیر صفت خاصه اتمیک تا تجزیه ناپذیر می باشد بنابراین همواره دارای رابطه های نرمال خواهد بود یعنی رابطه هایی که تمام صفات خاصه آن دارای مقادیر اتمیک هستند تا به عبارت دیگر تمام صفات خاصه آن روی میدان های اتمیک تعریف شده است بنابراین رابطه نرمال یک رابطه مسطح است .

S#	PQ	
	P#	QTY
S۱	P۱	۱۰
	P۲	۱۵
	P۹	۱۷
S۲	P۱۴	۱٤۰
	P۳	۱۸
	P۲	۹
S۳	P۹	۲۰
	P۳	۵۰

S#	P#	QTY
S۱	P۱	۱۰
S۱	P۲	۱۵
S۱	P۹	۱۷
S۲	P۱۴	۱٤۰
S۲	P۳	۱۸
S۲	P۲	۹
S۳	P۹	۲۰
S۳	P۳	۵۰
S۳		

انواع کلید در مدل رابطه ای :

۱. ابر کلید (super key)
۲. کلید کاندیدا (candidate key)
۳. کلید اصلی (primary key)
۴. کلید فرعی (ثانویه) جایگزین (Alter nerrt key)
۵. کلید خارجی (for gin key)

تعریف خاصیت کلید :یعنی منحصر به فرد بودن ، یعنی مقدار آن صفت یا صفات در کل آن جدول و در تمامی روابط قابل تعریف در آن شی تکرار نداشته باشد . به عبارت بهتر یعنی این که هیچ دو سطری نتوان پیدا کرد که مقدار آن دو سطح در ان یکسان باشد .

ابر کلید : صفت یا مجموعه ای از صفات که دارای خاصیت کلیدی باشد را ابر کلید می گویند .

کلید کاندیدا : ابر کلید کمیته را کلید کاندیدا گویند یعنی صفت با مجموعه ای از صفات که دارای خاصیت کلیدی بود و کمیته باشند .

-منظور از کمیته این است که هیچ جزئی از ان نتوان پیدا کرد که با حذف آن خاصیت کلیدی برقرار باشد .

مثال از جدول قبل:

S#,sname *

S# ✓

Sname,city# *

کلید اصلی : یعنی یکی از کلیدهای کاندیدا است که توسط مدیر بانک اطلاعاتی به عنوان کلید اصلی انتخاب می شود .

کلیدفرعی : کلیدهای کاندیدا به جز کلید اصلی کلیدهای فرعی انتخاب می شوند .

کلید خارجی : صفت یا مجموعه صفات در یک یک جدول که در جدول دیگری کلید اصل باشد .

نکته: موارد ابرکلید و کلید خارجی دارای استشاء هستند کلید خارجی از نظر کلید بودن استثناء است یعنی اینکه کلید خارجی ممکن است در جدول که در آن قرار دارد خاصیت کلیدی باشد در حالی که بقیه کلید حتماً باید خاصیت کلیدی را داشته باشند.

همچنین ابر کلید از نظر کمینه بودن استثناء است ، یعنی ابر کلید ممکن است دارای خاصیت کمینه نباشد در صورتی که سایر کلیدها حتماً دارای خاصیت کمینه هستند.

نکته کلید خارجی الزاماً خاصیت کلیدی ندارد امام ممکن است در مواردی بسیار استثناء امکان دارد کلید خارجی ، کلید اصلی نیز باشد. (در همان جدول) بنابراین اگر کلید خارجی در جدولی کلید اصلی نیز باشد می بایست به طراحی جدول شک کرد.

چرا که اگر کلید خارجی کلید اصلی نیز باشد به بیان دیگر ارتباط یک به یک بین دو جدول است همانطور که گفتیم یک به یک در بانک اطلاعاتی مجاز نیست اگر ارتباط یک به یک باشد.