

**طراحی پایگاه داده منطقی با مدل های مشخص ارتباط نهادی از منظر هستی شناختی**

**چکیده**

استدلال بر این است که مدل­های ارتباطی نهاد روشن از نقطه نظر هستی­شناسی در مقایسه با مدل­های نامشخص هستی­شناسی دامنه­های جهان واقعی را دقیق­تر مدلسازی می­کند. با اینحال، تحول این مدل­ها به مدل ارتباطی در سطح منطقی تا بدین جا به اندازه­ای مطالعه نشده است تا قوانین تحول جدید را تدوین کند. این مقاله مجموعه­ای از قوانین تحول را ارائه می­کند تا مدل­های هستی­شناسی مفهومی روشن را به مدل­های منطقی تبدیل کند. در نهایت، مقایسه­ای از دو مدل منطقی را انجام می­دهیم که از مدل­های مشخص و نامشخص هستی­شناسی با استفاده از یک معیار کیفیت ترکیبی از منابع موجود توسعه می­یابد. نتایج اولیه از این تحقیق در حال انجام حاکی از آن است که کیفیت مدل منطقی توسعه یافته از مدل مفهومی مشخص هستی­شناسی نسبت به همتای نامشخص هستی­شناسی خودش برتری دارد.

**1. مقدمه**

مدلسازی مفهومی فعالیتی است که در طول مراحل ابتدائی کار توسعه سیستم­های اطلاعاتی انجام می­شود، که در آن یک نمودار گرافیکی تولید می­شود که در حال ارائه پدیده کلمه واقعی از یک دامنه کاربرد است. محققان توانائی بیان متدولوژی مدلسازی مفهومی خوب شناخته شده، مدل ارتباطی موجودیت (ER) [4]، را با استفاده از هستی­شناسی [7] بهبود می­بخشند که ساختار و رفتار جهان واقعی را تشریح می­کند. این مدل­ها نمودارهای ارتباطی نهاد مشخص هستی­شناسی (OC-ERD) نام دارد [8].

به منظور استفاده از مزیت OC-ERDها، چنین مدل­هایی را می­بایست بصورت مناسبی به طرح منطقی پایگاه داده (RDS) [9، 10] بدون از دست دادن شماتیک­های OC-ERD تبدیل نمود. مجموعه کنونی از قوانین توسعه یافته برای تبدیل ERDهای عمومی برای OC-ERDها کاملا قابل اجرا نیست. بر این اساس، این مقاله برخی از نتایج ابتدائی از یک مطالعه تحقیقاتی در حال انجام را ارائه می­کند تا قوانین تحول جدید و روش ارزیابی کیفیت مدل منطقی حاصله از این قوانین توسعه یابد.

بقیه مقاله به شرح زیر است. بر این اساس، بخش 2 یک ERD نامشخص هستی­شناسانه (OUC-ERD) از یک سناریوی جهان واقعی مخصوص و تحول آن را همراه با موضوعات تحول نشان می­دهد. بخش 3 نسخه مشخص هستی­شناسانه از OUC-ERD، OC-ERD، و موضوعات تحول آن با استفاده از الگوریتم موجود را ارائه می­کند. همچنین این بخش یک الگوریتم جدید ارائه شده و تحول با استفاده از آن را ارائه می­کند. در بخش 4، یک معیار کیفیت برای ارزیابی کیفیت دو نوع از RDSهای منتج از هر دو رویکرد ارائه می­شود و کیفیت مقایسه نشان داده می­شود. در نهایت، نتایج مقدماتی و لثر آتی از این مطالعه تحقیقاتی در حال انجام در بخش 5 ارائه می­شود.

**2. موضوعات طراحی پایگاه داده منطقی با OUC-ERDها**

اکنون یک OUC-ERD و موضوعات تبدیل آن به مدل ارتباطی با استفاده از ER موجود به الگوریتم تحول ارتباطی را ارائه میکنیم.

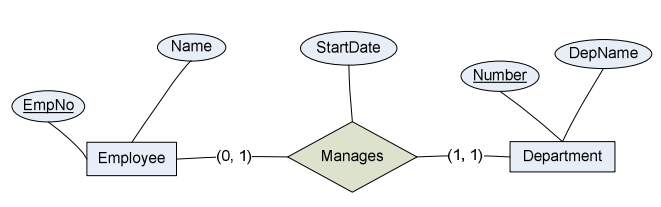
شکل 1 یک OUC-ERD است که یک شرکت را در دنیای واقعی ارائه میکند. نمودار شامل یک دودویی 1:1 (یک به یک) و «مدیران» نوع ارتباط بهینه بین دو نهاد از نوع «کارکنان» و «دپارتمان» است. نسبت کاردینالیتی (0، 1) بین نوع نهاد کارکنان و نوع رابطه دو معنا دارد یعنی یک کارمند ممکن نیست یک بخش دپارتمان را مدیریت کند یا کارمندی که دپارتمان را مدیریت می­کند تنها یک دپارتمان را می­تواند مدیریت کند. مشخصه «StartDate» نشان­دهنده­ی مورخه­ای است که یک کارمند شروع به اداره یک دپارتمان مخصوص می­کند.

اعتقاد بر این است که روابط اختیاری مانند رابطه فوق از جانب کاربران معمول حوزه به سختی درک شود. هستی­شناسی مانع استفاده از انواع روابط اختیاری است و استفاده از انواع روابط اجباری با ساب­تایپینگ را توصیه می­کند [8]. بدین ترتیب، نمودار نشان داده شده در شکل 1 یک نمودار ER نامشخص هستی­­شناسانه (OUC-ERD) است.

الگوریتم تحول موجود ارائه شده توسط المصری و ناواته [9] در زیر داده شده است.

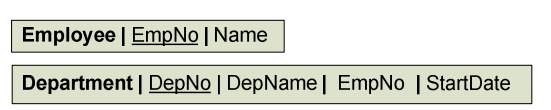
برای هر نوع موجودیت منظم E در طرح ER ، یک رابطه L ایجاد کنید که شامل تمام خصوصیات E باشد. یکی از ویژگیهای اصلی E را به عنوان کلید اصلی L انتخاب کنید.

برای هر نوع رابطه باینری 1: 1 در طرح ER، روابط S و T شناسایی می­شود که مربوط به انواع موجودیت مشارکت کننده در R است. یکی از روابط S را انتخاب کنید که شامل یک کلید خارجی در کلید اصلی S از T است. تمامی ویژگی­های ساده R را بعنوان ویژگی­های S به حساب آورید.



شکل 1. ERD نامشخص هستی­شناسانه (OUC-ERD)

شکل 2 RDS به دست آمده توسط تبدیل OIC-ERD با استفاده از الگوریتم فوق را نشان می­دهد.



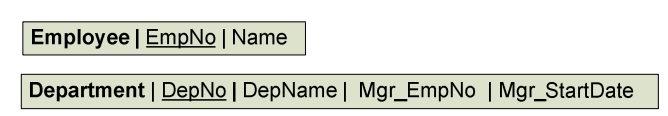
شکل 2. RDS به دست آمده با تبدیل OUC-ERD در شکل 1 با استفاده از ERD موجود برای الگوریتم تبدیل مدل منطقی.

از آنجا که RDS در شکل 2 توسط قوانین تحول دقیق زیر به دست می­آید، پس آن را تحول درون الگوریتمی می­نامیم.

با اینحال، برخی ابهامات غالب بر RDS وجود دارد. دلایل برای داشتن کلید ابتدائی (PK)، EmpNo کارکنان بعنوان یک کلید خارجی (FK) در رابطه دپارتمان به خاطر موارد مزبور است: در ابتدا به منظور ارائه ارتباط «مدیران» OUC-ERD، و سپس به منظور اشاره بر کارکنانی است که قرار است دپارتمان را اداره کنند. با اینحال، بطور کلی، این ممکن است برای شخصی غیر از طراح RDS به منظور درک این دلایل سخت باشد.

بطور مشابه، معنای ورود مشخصه «StartDate» در رابطه دپارتمان درک نمی­شود. کلمه «مدیران» در ERD در طول تحول حذف می­شود و در نتیجه این در RDS شامل نمی­شود. این احتمالا باید به ERD رجوع کند و گاهی اوقات حتی به سناریوی دامنه برای یافتن دلیل اشاره دارد. راه­حل اسفاده شده در عمل فعلی به منظور حل این ابهامات تغییر نام ویژگی­ها با استفاده از پیشوندهای مناسب است. بر این اساس، EmpNo و StartDate در دپارتمان به ترتیب به Mgr\_EmpNo و Mgr\_StartDate تغییر می­یابد. پیشوند «Mgr» کلمه «مدیر» را نشان می­دهد. RDS اصلاح شده در شکل 3 داده شده است.

با اینحال، معتقدیم که این تنظیمات خارج از قوانین است. هر یک از قوانین تحول یا تبدیل ویژگی­های نامگذاری مجدد، روش نامگذاری مجدد و چگونگی تصمیم­گیری درباره پیشوندها برای نامگذاری مجدد را ارائه نمی­کنند. از آنجا که این تنظیمات بیرون از الگوریتم تبدیل قانونی ایجاد می­شوند، پس آن را تنظیمات الگوریتم اضافه می­نامیم. این فرایند اضافه الگوریتم از تخصص دامنه طراح، بینش، داوری، و نظر شخصی استفاده می­کند.



شکل 3. تنظیمات الگوریتم اضافی ایجاد شده برای RDS تبدیل شده در شکل 1. ویژگی­های «EmpNo» و «StartDate» به «Mgr\_EmpNo» و «Mgr\_StartDate» نامگذاری مجدد می­شود. پیشوند اختصاری «Mgr» حاکی از کلمه «مدیر» است.

تنظمیمات اضافه الگوریتم چندین موضوع را به شرح زیر ایجاد می­کند.

a. نامگذاری دوباره مشخصات خارج از الگوریتم،

b. چگونه یک کلمه خاص به منظور استفاده تصمیم¬گیری می¬شود تا پیشوندی را برای نامگذاری مجدد ایجاد کند، برای مثال «مدیر» در این مورد،

c. چگونه پیشوند، برای مثال «Mgr» ایجاد می¬شود.

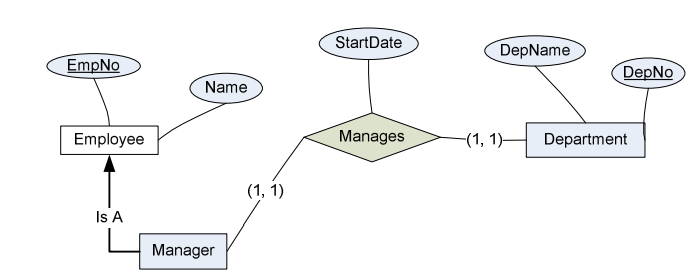
حتی پس از تنظیمات الگوریتم اضافی، به نظر میرسد برخی از سناریوها هنوز هم مبهم هستند. برای مثال، OUC-ERD نشان می­دهد که یک کارمند تنها یک دپارتمان را اداره می­کند یا اصلا دپارتمانی را مدیریت نمی­کند، و این باید در RDS در شکل 3 منعکس شود. با اینحال، این مشخص نیست که آیا کارکنان یک مدیر ذکر شده توسط ویژگی «Mgr\_EmpNo» مجددا نامگذاری شده می­توانند تنها یک یا چند دپارتمان را اداره کنند یا خیر، به این دلیل که همان Mgr\_EmpNo در چند تائی­های مختلف تکرار می­شوند. این نشان می­دهد که برخی از اطلاعات معنای در طول مدت تبدیل از دست رفته است و حتی با فرایند تنظیم الگوریتم اضافی نیز مجددا ایجاد نمی­شود.

فرض کنید یک طراحی با دامنه آشنا نیست، دامنه­ای همچون OUC-ERD که به منظور طراحی یک RDS داده شده است. احتمالا، طراح تبدیل درون الگوریتم را انجام دهد که با RDS نشان داده شده در شکل 2 یافته می­شود. با اینحال، وی گاهی اوقات ممکن است کل طراحی را بهم بریزد زمانی که ورود می­کند تا تنظیمات الگوریتم اضافی را با توجه به نبود دانش دامنه انجام دهد.

**3. OC-ERD و تبدیل آن**

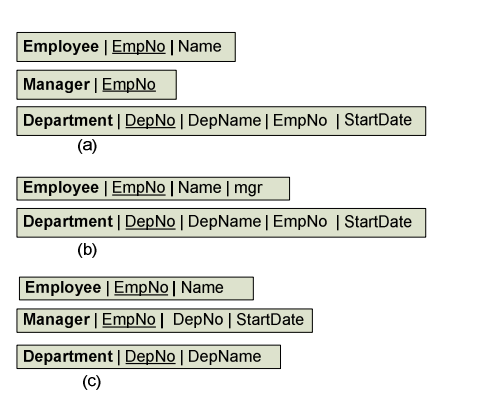
در OC-ERD نشان داده شده در شکل 4، رابطه بهینه با یک رابطه اجباری بین دپارتمان و یک یک زیر نوع جدیدا خلق شده به نام «مدیر» کارکنان حل می­شود؛ کارمند مبدل به یک زیر نوع می­شود.

اکنون نسبت­های کاردینالیتی بسیار روشن و آشکار است، و نشان­دهنده­ی این موضوع هستند که هر «مدیر» بطور قطعی یک «دپارتمان» را «مدیریت می­کند» و یک و تنها یک «دپارتمان» در هر زمان معین را «مدیریت می­شود».



شکل 4. ERD مشخص هستی­شناسانه (OC-ERD)

تبدیل OC-ERD با استفاده از قوانین موجود سه RDS بهینه مجزا را ارائه می­کند همانطور که در شکل 5 داده شده است. انواع مختلف ابهامات در هر RDS ارائه شده است، برای مثال، در شکل 5(الف)، افزونگی­های اطلاعات، عدم توانائی به منظور شناسایی ارتباط زیر نوع سوپر نوع و نهاد یا موجودیت دقیقی که در رابطه FK مشارکت می­کند، و فقدان توانائی به منظور شناسایی هدف ارتباط FK و غیره. همچنین شکل 5(ب) و (ج) حاوی برخی از ابهامات است.



شکل 5. تبدیل OC-ERD در شکل 4 با استفاده از قوانین موجود، (الف) 1– RDS بهینه، (ب) RDS-2 بهینه، mgr یک مشخصه نوع بولی است که حاکی از این است که چندتائی متعلق به آن است، (ج) RDS-3 بهینه

مثال ساده فوق نشان می­دهد که قوانین تبدیل موجود برای هر دو OUC-ERDها و OC-ERDها کارائی مناسبی ندارد. در نتیجه، این امر امکانپذیر نیست که یک RDS غیر مستقیم، سرراست و کاملا قابل اعتماد به دست آید که تمامی اطلاعات ارائه شده در هر یک از RDSها را ارائه کند. در نتیجه، این موضوع ضروری است که مجموعه جدید یا تغییر یافته­ای از قوانین به منظور تطبیق با OC-ERDها و همچنین OUC-ERDها توسعه یابد.

مشاهده می­شود که OC-ERDها پشتیبانی نسبتا بیشتری را برای جبران یا رفع ابهامات در فرایند تبدیل ارائه می­کنند. برای مثال، موضوع حل پیشوند «Mgr» برای نام­گذاری مجدد ویژگی­ها. در مورد OUC-ERD، این کاملا مبتنی بر نظر و تخصص طراح است. با اینحال، در مورد OC-ERD، خود ERD کلمه «مدیر» را بعنوان یک زیر نوع در حال مشارکت در ارتباط مربوط به «مدیران» اشاره می­کند، به طوری که کلمه را می­توان به منظور فرم­دهی یا شکل­دهی پسوند اختصاری کرد یا کلمه کامل را می­توان بعنوان پسوند به کار برد اگر که مطلوب باشد. وضعیت موجود این انگیزه را به ما می­دهد تا یک رویکرد منطقی را برای فرایند نامگذاری مجدد معرفی کنیم که OUC-ERD را به RDS نهایی پیوند می­دهد.

الگوریتم جدید ارائه شده که OC-ERDها را تحت پوشش قرار می­دهد دارای جنبه­های زیر است.

1. یک مجموعه نامگذاری برای مشخصات، روابط و انواع ناهد در OUC-ERD.

2. یک مجموعه جدید از قوانین تحول برای تبدیل OC-ERDها به مدل ارتباطی.

3. قوانین برای نامگذاری مجدد اسامی ویژگی¬ها در طرح ارتباطی پاسگاه داده.

الگوریتم ارائه شده که دارای چندین مرحله است و همچنین در فرمی جدید یا اتخاذ شده است [9] در زیر ارائه می¬شود.

**مرحله 1: نقشه­برداری از انواع نهاد منظم**

1. برای هر نهاد منظم نوع E در ERID، یک رابطه L را ایجاد کنید که دربرگیرنده¬ی تمامی ویژگی¬های E باشد.

2. یکی از ویژگی¬های کلیدی E را بعنوان کلید اصلی L یعنی PK(L) انتخاب کنید.

3. PK را بعنوان نخستین مشخصه رابطه جدید L در حال تداوم توسط ویژگی¬های باقیمانده¬ی E نهاد قراردهی کنید

**مرحله 2: نقشه­برداری بخش­های زیرنوع/ سوپر نوع**

هر بخش زیر نوع/ سوپر نوع با تعداد m از زیر نوع­های  و همچنین یک سوپر نوع C را به طرح­های رابطه از طریق یکی از گزینه­های زیر تبدیل کنید.:

گزینه 1:

1. ایجاد یک رابطه L برای C با ویژگی¬های و کلید اصلی، .

2. ایجاد یک رابطه مجزای برای هر زیر کلاس ، با ویژگی¬های و .

گزینه 2:

1. ایجاد یک رابطه مستقل L با مشخصات و . هر ، ، یک ویژگی نوع بولی است که نشان می¬دهد آیا چندتائی متعلق به زیر کلاس است یا نیست.

2. در L، ویژگی¬های یک زیر نوع مخصوص باید مشتمل بر مشخصه نوع بولی مرتبط آن باشد.

**مرحله 3: نامگذاری روابط تبدیل شده**

مورد 1: نام سوپر نوع C در حال حاضر در نام زیر نوع S بعنوان آخرین کلمه آن به شرح زیر به حساب آورده می­شود

* که در آن X یک کلمه جدید مورد استفاده برای شکل­دهی بخشی از نام زیر نوع است.

در این مورد، خط زیرین یا زیرین خط بایستی در بین X و C برای شکل­دهی X\_C قرار گیرد، به طوری که .

مورد 2: زمتنی که نام سوپر نوع در نام زیر نوع به حساب نمیآید، کل نام زیر نوع بایستی به نام سوپر نوع توسط یک خط زیرین متصل شود تا S\_C را شکل دهد، که در آن S نام زیر نوع است و C نام سوپر نوع است.

**مرحله 4: نامگذاری یک ویژگی نوع بولی استفاده شده برای تبدیل یک زیر نوع**

نام مشخصه نوع بولی، به اصطلاح t، بایستی با پیوند نام زیر نوع، به اصطلاح S، توسط یک خط تیره همراه با ردیف متن «Flg» ایجاد شود که کلمه «Flag» را نشان می­دهد، به طوری که t = S-Flg.

**مرحله 5: نقشه­برداری یک رابطه دودویی نوع R در بین یک نوع نهاد منظم و یک زیر نوع از یک سوپر نوع**

فرض کنید که یک زیر نوع S از یک سوپر نوع C دارای یک نوع ارتباط دودویی 1:1 با یک نهاد منظم نوع E باشد.

مورد 1: زیر نوع به یک رابطه جداگانه تبدیل می­شود.

به منظور تبدیل رابطه نوع R روابط مجزای  و  را شناسایی کنید که مربوط به نهاد نوع S (زیر نوع) و E (منظم) است که در R مشارکت دارند.

گزینه 1

1. بعنوان یک کلید بیرونی در کلید اولیه ، شامل می¬شود و آن بعنوان مجددا نامگذاری می¬شود.

2. یک پسوند اضافه میشود که بعنوان ظاهر می¬شود. حرف (U) نشان میدهد که کلید خارجی منحصربفرد است.

3. هر ویژگی ساده، به اصطلاح A، از رابطه نوع R در به حساب می¬آید که پیرو کلید خارجی مشتمل است و آن بعنوان S\_A نامگذاری مجدد می¬شود.

4. اگر زیر نوع S در طرح ER رابطه مجزای ویژگی¬های خود آن را در بر نگیرد، که زیر نوع S را نشان می¬دهد بایستی در این گزینه نادیده گرفته شود.

گزینه 2

1. بعنوان یک کلید خارجی در کلید اصلی به حساب میآید، همراه با یک پسوند U بعنوان ظاهر می¬شود. کلمه «U» اشاه بر این دارد که کلید بیرونی محصربفرد است.

2. تمامی ویژگی¬های رابطه نوع R در L به تبعیت از کلید بیرونی شامل می¬شود.

مورد 2: زیر نوع با استفاده از یک ویژگی نوع بولی به یک رابطه مستقل همراه با سوپر نوع آن تبدل می­شود.

برای هر رابطه نوع R، رابطه واحد L شناسایی می­شود که مربوط به بخش S/C زیر نوع/ سوپر نوع و رابطه  از نهاد منظم نوع E است که در آن S و E در حال مشارکت در R هستند.

1. شامل کلید خارجی در کلید اصلی از L، است و آن بعنوان مجددا نامگذاری می-شود.

2. یک پسوند (U) اضافه می¬شود که بعنوان ظاهر می¬شود. کلمه «U» نشان می¬دهد که کلید بیرونی منحصربفرد است.

3. شامل تمامی ویژگی¬های ساده R بعنوان ویژگی¬های است که به تبعیت از کلید بیرونی مشتمل می-باشد.

4. شامل هر مشخصه ساده، به اصطلاح A، از رابطه نوع R در است به تبعین از این کلید بیرونی که مشتمل است و بعنوان S\_A نامگذاری مجدد می¬شود.

5. اگر زیر نوع S در طرح ER شامل مشخصات خود آن نباشد پس مشخصه نوع بولی شامل در رابطه L می¬بایست برای این گزینه حذف شود.

**مرحله 6: نقشه­برداری یک رابطه دودویی 1:1 نوع R بین دو زیر نوع مختلف از سوپر نوع­های متفاوت**

فرض کنید که A و B دو زیر نوع متفاوت از دو سوپر نوع مختلف از یک طرح ER باشند و فرض کنید که R یک نوع رابطه دودویی 1:1 است که در بین A و B موجود است.

1. تبدیل یکی از زیر نوع¬ها، به اصطلاح A، به یک رابطه مجزای .

2. با فرض بر این که یک رابطه جداگانه مربوط به زیر نوع B است، و بعنوان یک کلید اصلی از کلید بیرونی ، شامل می¬شود، و آن بعنوان نامگذاری مجدد می¬گردد. این کلید خارجی بایستی به تبعیت از آخرین ویژگی موجود از شامل شود.

3. شامل تمامی ویژگی¬های ساده R در به دنبال کلید بیرونی است.

4. اگر زیر نوع B حاوی ویژگی¬های متعلق به خودش باشد، پس این باید همراه با سوپر نوع خودش با استفاده از یکی از عملیات مناسب زیر تبدیل شود.

(1) دو رابطه جداگانه برای هر زیر نوع B و سوپر نوع آن (مرحله 2 گزینه 1 فوق)

(2) یک رابطه مستقل برای هر دو زیر نوع B و سوپر نوع آن (مرحله 2 گزینه 2 فوق).

5. اگر زیر نوع B حاوی ویژگی¬های متعلق به خودش نباشد پس سوپر نوع آن بایستی تبدیل شود.

**مرحله 7: نقشه­برداری یک نوع ارتباط 1:N دودویی بین دو زیر نوع متفاوت از سوپر نوع­های مختلف**

فرض کنید که A و B دو زیر نوع مختلف از دو سوپر نوع متفاوت از طرح ER هستند که در آن R یک نوع رابطه 1:N دودویی موجود در بین دو زیر نوع است و همچنین فرض کنید که A در جانب 1 و B در جانب N از رابطه است، به طوری که یک نمونه از A با بسیاری از نمونه­های B در ارتباط است.

1. تبدیل زیر نوع B که در جانب N با یک رابطه مجزای L¬B¬¬ است.

2. فرض کنید که LA¬¬¬¬ رابطه مربوط به زیر نوع A باشد، بعنوان کلید خارجی کلید اصلی LA¬¬¬¬ را به حساب آورید و آن را بعنوان A\_PK(LA¬¬¬¬) مجددا نامگذاری کنید. این کلید خارجی می¬بایست شامل آخرین مشخصه موجود در L¬B¬¬ باشد.

3. در L¬B¬¬ تمامی ویژگی¬های ساده رابطه نوع R را به حساب آورید که به تبعیت از کلید خارجی است.

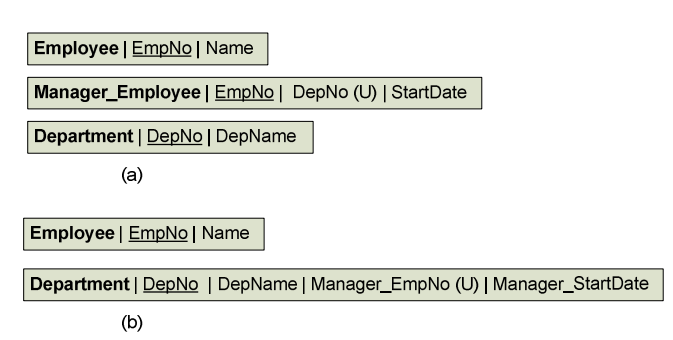
4. اگر زیر نوع A حاوی ویژگی¬های متعلق به خودش باشد، پس A و زیر نوع آن می¬بایست با استفاده از هر یک از روش¬های بهینه زیر تبدیل شود.

الف) دو رابطه جداگانه برای هر زیر نوع A و زیر نوع آن (مرحله 2 گزینه 1 فوق)

ب) یک رابطه مستقل برای هر دو زیر نوع A و زیر نوع آن (مرحله 2 گزینه 2 فوق)

5. اگر زیر نوع A شامل ویژگی­های خودش نباشد پس سوپر نوع می­بایست تنها به یک رابطه مجزا تبدیل شود.

شکل 6 (الف) و (ب) نتایج تبدیل OC-ERD داده شده در شکل 4 با استفاده از الگوریتم جدید ارائه شده در بالا را نشان می­دهد. بر این اساس، RDS بهینه (ب) نتیجه­ای از مورد 1 گزینه 1.1 است؛ RDS بهینه (الف) نتیجه­ای از مورد 1 گزینه 1.2 است. نتیجه مورد 2 مشابه با RDS (ب) است. در نتیجه تنها دو RDS بهینه تولید می­شود هر چند که سه روش بهینه مجزا برای تبدیل OC-ERD وجود دارد.



شکل 6. تبدیل OC-ERD در شکل 4 با استفاده از الگوریتم جدید، (الف) OC-RDS-1 مطلوب (ب) OC-RDS-2 مطلوب

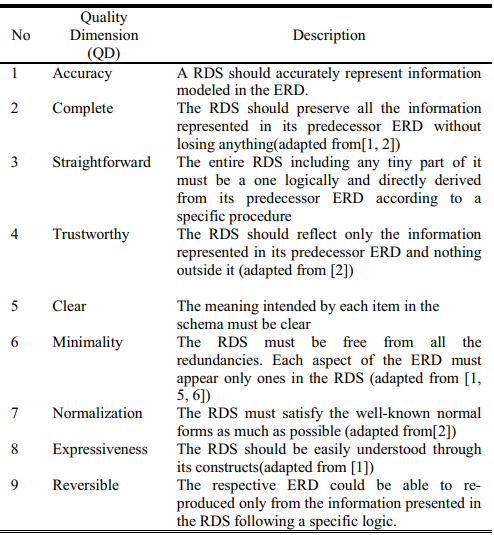
حتی در این تحول، برخی از ویژگی­های نیاز به نامگذاری مجدد دارند، برای مثال، EmpNo به Manager\_EmpNo نامگذاری مجدد می­شود. تفاوت بین این فرایند نامگذاری مجدد و فرایند نامگذاری مجدد به دلیل OUC-ERD است که در این مورد این امر توسط خود فرایند تحول رسیدگی می­شود بدون این که از بینش طراح گذر کند. این رابطه نظام­مند قبلا بعنوان یک قانون کلی ایجاد شده است و در الگوریتم جدید در مرحله 5، مورد 1، گزینه 1.1 زیر مرحله 2 و 3 داده شده است. مشخصه FK دوباره نامگذاری شده یک پسوند Uرا داده است که حاکی از این است که کلید بیرونی منحصربفرد در حال پیشگیری از هر نوع تکرار غیر معمول از مقادیر FK در چندتائی­های مختلف است همانطور که این مورد در شکل 3 رخ داده است. بر این اساس، تمامی ابهامات تجربه شده در تبدیل OUC-ERD توسط یک قانون بخصوص حل میشود.

**4.کیفیت طرح پایگاه داده طراحی شده با استفاده از الگوریتم جدید**

ارزیابی کیفیت RDSهای منتج از OC-ERDها از طریق الگوریتم تبدیل جدید حائز اهمیت است. بخصوص این شایان توجه و اهمیت است که، کیفیت نسبی RDSها از طریق مدل نامشخص هستی­شناسانه تولید می­شود و مدل­های مشخص هستی­شناسانه نیاز به ارزیابی دارند. پس تنها موفقیت الگوریتم جدید را می­تواند ادعا نمود.

چند مطالعه قبلی [1، 2، 5] ابعاد کیفیت و طرح­های اندازه­گیری کیفیت را برای طرح­های پایگاه داده و داده­ها ارائه کرده­اند.

جدول 1 مجموعه­ای از ابعاد کیفیت ترکیب شده از منابع موجود را نشان می­دهد. بر این باوریم که ارزیابی کیفیت ابتدائی را می­توان با استفاده از این ابعاد انجام داد.



همچون تست کیفیت ابتدائی RDS در شکل 3 را همراه با دو RDS بهینه در شکل 6 با استفاده از جدول 1 فوق مقایسه می­کنیم. مشاهده می­کنیم که تمامی روابطدر RDSهای شکل 6 در سومین فرم معمول هستند. در عین حال، همه روابط در شکل 3 در فرم­های اول و دوم معمول هستند. همچنین، می­توان نتیجه­گیری کرد که طرح­های پایگاه داده منطقی از ERDهای مشخص هستی­شناسانه تولید می­شوند که در تبعیت از الگوریتم جدید هستند که کیفیت بالاتر را با توجه به سطح عادی­سازی در مقایسه با همایان نامشخص هستی­شناسانه خودش نشان می­دهد.

**5. بحث و کار آتی**

با استفاده از یک مثال ساده از زندگی واقعی، نشان داده­ایم که نمودارهای ER نامشخص هستی­شناسانه مد­ل­های ارتباطی کیفیت بالا را در مقایسه با همتای خودش یعنی مدل نامشخص هستی­شناسانه تولید می­کند. این امر با اصلاح قوانین تبدیل موجود حاصل می­شود. با اینحال، به منظور دستیابی به نتایج مطلوب، این الگوریتم تبدیل باید از بهبود بیشتری برخوردار شود. همچون مطالعه اولیه تنها یک بخش از معیار کیفیت برای اریابی کیفیت مدل­های منطقی تولید شده استفاده می­شود.

اثر آتی از این تحقیق در حال انجام شامل موارد زیر است:

• توسعه قوانین جدید به منظور پوشش تمامی ساختارهای هستی¬شناسی

• اصلاح قوانین موجود به منظور غلبه بر معایب موجود با توجه به نمودارهای هستی¬شناختی نامشخص

• توسعه ارزیابی کیفیت به منظور غلبه بر سایر معیارهای کیفیت

• بهبود معیار کیفیت در این اثناء

انتظار بر این است که این مطالعه بر محبوبیت و کاربرد مدل­های ER مفهومی مشخص هستی­شناختی در صنعت تاثیرگذار باشد.

**REFERENCES**

[1] C. Batini, S. Ceri, and S. B. Navathe, Conceptual database design: an Entity-relationship approach: Benjamin-Cummings Publishing Co., Inc. Redwood City, CA, USA ©1992, 1992.

[2] R. Y. Wang, M. P. Reddy, and H. B. Kon, "Toward quality data: An attribute-based approach," Decision Support Systems, vol. 13, pp. 349-372, 1995.

[3] R. Weber, "Research review paper conceptual modeling and ontology: posibilities and pitfalls," Journal of Database Management, vol. 14, pp. 1-20, 2003.

[4] P. P.-S. Chen, "The entity-relationship model: toward a unified view of data," ACM Trans. Database Syst., vol. 1, pp. 9-36, 1976.

[5] C. Fahrner and G. Vossen, "A survey of database design transformations based on the Entity-Relationship model," Data & Knowledge Engineering, vol. 15, pp. 213-250, 1995.

[6] Y. Wand and R. Weber, "On the ontological expressiveness of information systems analysis and design grammars," Information Systems Journal, vol. 3, pp. 217-237, 1993.

[7] M. Bunge, Treatise on basic phylosophy: Vol3: ontology I: The furniture of the world.: D. Reidel Publishing Co., Inc., New York, NY., 1977.

[8] Y. Wand, V. C. Storey, and R. Weber, "An ontological analysis of the relationship constuct in conceptual modeling," ACM Transactions on Database Systems, vol. 24, pp. 494-528, December 1999.

[9] R. Elmasri and S. B. Navathe, "Fundamentals of Database Systems," in Fundamentals of Database Systems, 5th Edition ed New York: Addison Wesley, 2007, pp. 223-239

[10] E. F. Codd, "A relational model of data for large shared data banks," Commun. ACM, vol. 13, pp. 377-387, 1970.