



مهاجرت موثر از یک پایگاه داده به پایگاه داده دیگر
با بررسی موردی مهاجرت از پایگاه داده میکروسافت اس کیو ال
سرور^۱ به پایگاه های داده مای اس کیو ال^۲ و اوراکل^۳

استاد: جناب دکتر حجاریان
درس فناوری اطلاعات

تهیه کننده: رضا نقی زاده
دانشجو مدیریت فناوری اطلاعات
دانشگاه علامه طباطبائی

^۱ Microsoft SQL Server

^۲ MYSQL

^۳ ORACLE

۱- چکیده

در این مقاله قصد داریم به بررسی و تشریح مهاجرت^۴ از یک پایگاه داده^۵ به پایگاه داده دیگر بپردازیم و تغییرات و عملیاتی که برای این مهاجرت نیاز است را بررسی نماییم. مهاجرت از یک پایگاه داده به پایگاه داده دیگر همیشه با هزینه های فراوانی چه از لحاظ مادی و چه از ابعاد دیگر همراه بوده است با این وجود همیشه این مهاجرت ها موفقیت آمیز نبوده است. ما در این مقاله سعی داریم تا ضمن بیان نکات مهم در مهاجرت از پایگاه داده تا حد امکان هزینه های ناشی از این مهاجرت را کاهش داده و نتایج این مهاجرت را اثر بخش تر نماییم. و در نهایت ضرورت طراحی پایگاه داده به صورت نرمالیزه شده را بیان می کنیم و اثرات آن را در مهاجرتی بهتر را بررسی می کنیم.

۲- مقدمه

پایگاه داده یکی از بخش های اصلی هر نرم افزاری است که استفاده صحیح از آن می تواند هر نرم افزاری را از بقیه متمایز نماید. با رشد سریع حرکت سازمانها به سمت دیجیتالی کردن اطلاعات حجم های پایگاه های اطلاعاتی افزایش یافته و پایگاههای اطلاعاتی گذشته دیگر پاسخگو نیستند و این مورد مشکلی تحت عنوان مهاجرت از یک پایگاه داده به پایگاه داده دیگر را برای سازمانها پیش آورده است و هزینه زیادی را نیز عموماً به سازمان تحمیل می نماید.

⁴ Migration
⁵ Database

سعی داریم با بررسی بیشتر در مورد مهاجرت از یک پایگاه داده به پایگاه داده، عوامل موثر در انتقالی کم هزینه تر و در نهایت کارآیی بالاتر داشتن را بررسی نماییم. همچنین در نهایت به طور خلاصه درباره دو مدل از مهاجرت ها مطالبی را بیان می کنیم.

۳. چرا مهاجرت؟

شاید بکاربردن واژه مهاجرت به جای تغییر در مقوله پایگاه داده کمی سوال برانگیز باشد؟ دلیل استفاده از این واژه این است که ما در این مقاله سعی داریم این نکته را بیان کنیم که تا جائیکه امکان دارد باید از عوض کردن پایگاه داده پرهیز نمود و با تغییراتی که در همان پایگاه داده می دهیم خواسته های خود را تامین نماییم و در صورتیکه چاره ای نبود به تعویض پایگاه داده روی آوریم. در نتیجه مهاجرت از یک پایگاه داده به پایگاه داده دیگر یعنی تعویض پایگاه داده به این دلیل که چاره ای جز آن نداریم.

۴. ابعاد مهاجرت

مهاجرت از یک پایگاه داده به پایگاه داده دیگر دارای ابعاد مختلفی است که دو بعد اصلی آن شامل بعد مفهومی^۶ مهاجرت و بعد تکنیکی^۷ مهاجرت است. در بعد مفهومی ما به چرایی این مهاجرت می پردازیم و به بررسی این نکته می پردازیم که آیا مهاجرت از این پایگاه داده مناسب است یا خیر؟

⁶ Conceptual aspect
⁷ Technical aspect

اما در بررسی تکنیکی این مهاجرت به محدودیت های تکنیکی پیش رو در مهاجرت از یک پایگاه داده به پایگاه داده دیگر می پردازیم. در این بعد از مهاجرت تلاش خود را بر مهاجرت پایگاه داده با هزینه کمتر بنا می گذاریم. مهاجرت پایگاه داده به صورت تکنیکی با چالش های مختلفی از جمله تغییر در نوع داده ها^۸، توابع داخلی و موارد بسیار دیگر روبرو است که سعی ما بر این است که تا حد امکان این تغییرات را کاهش دهیم.

۱-۴- بعد مفهومی مهاجرت

بعد مفهومی مهاجرت به عنوان یکی از مهمترین ابعاد مهاجرت از یک پایگاه داده به پایگاه داده دیگر و شاید مهمترین بعد آن است در این قسمت در ابتدا ما به نقاط قوت و ضعف پایگاه داده فعلی می پردازیم و به صورت کامل نقاط ضعف و قوت پایگاه داده فعلی را تشریح می کنیم.

در این مرحله از تحلیل باید به این نکته برسیم که چرا باید پایگاه داده را تغییر دهیم؟ و اگر بخواهیم پایگاه داده را حفظ نماییم چه دلایلی دارد؟

هر نوع از پایگاه داده بدون در نظر گرفتن خود محصول از لحاظ وجودی دارای فلسفه ای خاص است که در مهاجرت پایگاه داده باید به آن توجه نمود.

به طور کلی انواع پایگاه داده شامل پایگاه داده های ارتباطی^۹، سلسله مراتبی^{۱۰}، شبکه ای^{۱۱}، شی گرا^{۱۲} و شی گرا رابطه ای^{۱۳} است. به طور مثال در پایگاه داده های رابطه ای

^۸ Data type

^۹ Relational

^{۱۰} Hierarchical

^{۱۱} Network

^{۱۲} Object-Oriented

^{۱۳} Object-Relational

که پرتعدادترین پایگاه داده برای پی سی ها و کامپیوترهای بزرگ و مین فریم ها است. داده ها را در یک مدل دوبعدی ساده قرار می دهد که جداول ممکن است به فایل ها اشاره کنند. [Laudon & Laudon, 2006]

که در این نوع پایگاه داده در کنار سادگی و سرعت عمل در تغییر داده ها و ... دارای معایبی هم است که این معایب به ذات این نوع پایگاه داده ها بر می گردد و فرقی عمده در این که این پایگاه اس کیو ال باشد یا اوراکل ندارد. به طور مثال بازیابی داده ها به صورت مداوم از جدول های مختلف و ترکیب کردن آنها امری زمان گیر و کارایی پایگاه داده را کاهش می دهد. پس این مورد نمی تواند به تنهایی دلیلی بر مهاجرت از یک پایگاه داده اس کیو ال سرور به اوراکل باشد.

بعضا در مهاجرت ها ما مجبور به تغییر در نوع پایگاه داده خود هستیم که این ناشی از انتخاب نادرست پایگاه داده در ابتدای طراحی آن بوده است.

مثلا در سیستم های مدیریت پایگاه داده^{۱۴} متداول برای داده های همگن هستند و داده های از پیش تعریف شده و یا رکوردهای سازماندهی شده در ردیف ها و ستون ها استفاده می شود. و این سیستم ها برای داده های تصویری و مولتی مدیا مناسب نیست. سیستم مدیریت پایگاه داده شی گرا اخیرا دارای طرفداران زیادی شده اند، زیرا آنها می توانند برای کامپوننت های مولتی مدیا مختلف یا اپلت های جاوا که در وب استفاده می شوند استفاده شود. سیستم مدیریت پایگاه داده شی گرا برای داده رکرسیو بسیار مورد استفاده هستند. با این وجود پایگاه داده شی گرا می تواند داده هایی از مدل های مختلف بیشتری نسبت به ارتباطی ها را ذخیره کنند.

¹⁴ DBMS

به طور کلی ما در رویکرد مفهومی در مهاجرت از یک پایگاه داده به پایگاه داده دیگر با توجه به مدل خلاصه ای از پایگاه داده که از دورنمای کسب و کار ناشی شده است به تبیین ضروریات این مهاجرت می پردازیم و اگر در این مرحله به این نتیجه برسیم که این مهاجرت ضروری است به بررسی ابعاد دیگر مهاجرت مبادرت می نماییم.

۲-۴- بعد تکنیکی

در بعد تکنیکی مهاجرت بیشتر تمرکز در حوزه محصولات مختلف است تا نوع پایگاه داده. به همین جهت باید در ابتدا به نقاط قوت و ضعف پایگاه داده فعلی از جنبه های تکنیکی پرداخت و پس از طی این مرحله با توجه به آگاهی از موارد مورد نیاز پایگاه داده جدید را انتخاب نمایید. اما به جهت اینکه این مهاجرت با کمترین هزینه و تغییرات همراه باشد باید تا حد امکان پایگاههایی برای مهاجرت انتخاب نمود که دارای خصوصیات مشترکی باشند.

اولین کاری که برای مهاجرت باید انجام دهید طرح ریزی و برنامه ریزی برای تبدیل و مهاجرت می باشد. در طرح خود بایستی مواردی مانند تغییر در نوع داده , خود داده , نحوه ذخیره شدن داده و ... را مد نظر داشته باشید. همچنین با تغییر پایگاه داده خود باید تغییرات عمده ای در کلاینت^{۱۵} خود مانند تغییر در استفاده از نشانه گر ها^{۱۶}, رویه های ذخیره شده^{۱۷} و داده های داخلی را انجام دهید. شما همچنین باید برای نگهداری و به روز کردن نرم افزارهای کاربردی خود پس از مهاجرت برنامه ریزی کنید.

¹⁵ Client

¹⁶ Cursor

¹⁷ Stored Procedured

در مقوله نوع های داده ای که از اهمیت زیادی در تغییرات ناشی از مهاجرت برخوردار است مهم آن نیست که اسم های نوع داده یکی باشد بلکه باید در نوع داده، طول و گنجایش آن تطابق وجود داشته باشد به این جهت است که در ابتدای امر باید شناسایی کامل از هر دو پایگاه داده فعلی و بعدی انجام گیرد. بعضی از نوع داده ها در پایگاه های داده هیچ ارتباطی به یکدیگر ندارند ولی می توان در آنها با توجه به نوع داده های دیگر انتقال را بوجود آورد.

بعضی اوقات شما نیاز دارید که برای تبدیل داده به نوع های دیگر خود داده را تغییر بدهید. مثلا در یک پایگاه داده از چپ به راست و در یکی از راست به چپ است.

در این موارد گاه ابزاری برای این تبدیلات از یک پایگاه داده به پایگاه داده دیگر است. در غیر این صورت اگر خودتان ابزار تبدیلتان را می نویسید بایستی تمامی این موارد را در نظر داشته باشید. موارد دیگر تغییر در داده شامل تغییرات طرح داده برای نرمال کردن جداول پایگاه داده می باشد تا عمل تبدیل صورت گیرد.

ابزارهای تبدیلی که عموما برای پایگاه داده های مشهور نوشته شده است می تواند آسانترین راه برای مهاجرت از یک پایگاه داده به پایگاه داده دیگر باشد. در مورد توابع داخلی و رویه های ذخیره شده عموما اشتراک کاملی میان پایگاه های داده دیده نمی شود و به همین جهت تقریبا در مهاجرت از پایگاه داده احتیاج به بازنویسی این توابع و رویه ها داریم.

۵. نرمالسازی^{۱۸} پایگاه داده ها

در مهاجرت از یک پایگاه داده به پایگاه داده در صورتی که پایگاه داده فعلی نرمالیزه باشد آنگاه مهاجرت از آن امری ساده تر است برای همین تاکید زیادی بر نرمال سازی پایگاه های داده می شود. همچنین از موارد دیگر که می توان به آن اشاره نمود این است که بسیاری از مواردی که به عنوان ناکارآمدی یک پایگاه داده مطرح می شود ناشی از عدم نرمالسازی مناسب پایگاه داده است.

نرمال سازی یا به تعبیری هنجار سازی فرآیندی است در رابطه با بانک های اطلاعاتی که با دو هدف عمده زیر انجام می شود : [www.persianforum.com]

- کاهش افزونگی اطلاعات ، به این معنی که اطلاعات فقط در یک مکان (جدول) ذخیره و در تمام پایگاه با استفاده از روابط منطقی تعریف شده قابل دسترسی باشد .
- حفظ یکپارچگی اطلاعات ، به این معنی که اعمال تغییرات بر روی اطلاعات (نظیر ایجاد ، بهنگام سازی و حذف) در یک مکان انجام و به دنبال آن آثار تغییرات در تمام بانک مشاهده گردد .

با توجه به اهداف فوق می توان گفت که فرآیند نرمال سازی از ناهنجاری های بوجود آمده به دلیل بروز تغییرات در بانک جلوگیری خواهد نمود . با اعمال فرآیند نرمال

¹⁸ Normalization

سازی ، یک بانک اطلاعاتی کارآ و مطمئن را خواهیم داشت .

فرآیند نرمال سازی ، فرم های متفاوتی دارد که انواع متداول آن به شرح ذیل است :

- فرم اول نرمال سازی^{۱۹}
- فرم دوم نرمال سازی^{۲۰}
- فرم سوم نرمال سازی^{۲۱}
- فرم بویس کد نرمال سازی^{۲۲}
- فرم چهارم نرمال سازی^{۲۳}

۱-۵- فرم اول نرمال

موجودیت و یا جدولی در فرم اول نرمال است که تمامی المان های اطلاعاتی آن یکتا باشند.

به طور خلاصه می توان گفت که هدف از فرم اول نرم سازی حذف گروه های تکرار و آرایه ها از موجودیت یا جدول است . فرآیند فوق ، می بایست بر روی تمامی موجودیت های بانک اطلاعاتی اعمال گردد تا بتوان گفت بانک اطلاعاتی نرمال شده در فرم اول است .

۲-۵- فرم دوم نرمال

¹⁹ 1NF
²⁰ 2NF
²¹ 3NF
²² BCNF
²³ 4NF

گروه های داده ای^{۲۴} که کلید ترکیبی دارند را در نظر بگیرید اگر بخشی از کلید ترکیبی مقدارش به تنهایی بتواند زیر مجموعه ای از عناصر داده ای را منحصر به فرد نماید آن بخش از عناصر داده ای از کلید ترکیبی به یک گروه داده ای جدید منتقل شده و یک را بطه یک به چند بین گروه داده ای جدید و گروه مبدا برقرار می شود.

۳-۵- فرم سوم نرمال

عناصر داده ای که تاکنون کلید نبوده است و مقدارش به تنهایی بتواند زیرمجموعه ای از عناصر داده ای را منحصر به فرد نماید. آن عناصر داده ای و آن بخش به جدول جدیدی منتقل می شود.

موجودیت و یا جدولی در فرم سوم نرمال است که اولاً در فرم دوم نرمال بوده و ثانیاً تمام آیتم های غیر کلید آن وابستگی تابعی به کلید اصلی داشته باشند ، نه به یک آیتم غیر کلید .

۴-۵- فرم بویس کد نرمال

فرم بویس کد دارای مفهوم جامع تری نسبت به فرم دوم و سوم نرمال است . در فرم دوم و سوم نرمال بحث بر سر وابستگی تابعی آیتم های غیر کلیدی به کلید اصلی است . اما در فرم بویس کد ، موجودیتی در فرم بویس کد نرمال است که اولاً در فرم اول نرمال بوده و ثانیاً تمام المان های غیر کلیدی آن کاملاً" وابسته تابعی به یک کلید باشند و نه چیز دیگر . نکته حائز اهمیت در این فرم این است که بحث بر سر وابستگی تابعی با یک کلید است نه فقط کلید اصلی. مفهوم فوق در خصوص موجودیت هائی که

²⁴ Data Groups

دارای چندین کلید هستند^{۲۵} مطرح می شود .

۵-۵- فرم چهارم نرمال

این فرم در خصوص موجودیت هائی است که ارتباط بین المان های آن یک ارتباط چند ارزشه و یا چند به چند باشد . به عنوان مثال ، موجودیت کلاس درس می تواند شامل چندین دانش آموز و چندین معلم باشد. در چنین مواردی ارتباط بین معلم و دانش آموز یک ارتباط چند به چند می باشد . در این حالت با ایجاد یک موجودیت رابط مابین موجودیت های مذکور، مشکل ارتباط چند به چند حل خواهد شد . معمولاً "تمام المان های موجودیت رابط ایجاد شده بخشی از کلید اصلی است .

این نرمالسازی پایگاه های داده که در نهایت منتهی به دیاگرام ارتباط هویت ها و معماری اطلاعات می شود جلوی تغییرات اضافی که بر اثر به هم ریختگی داده ها ایجاد می شود را گرفته و جلوی تغییرات غیر ضروری را در مهاجرت ها می گیرد. نکته کلیدی در اینجا این است که در زمانی که ما دارای پایگاه داده نرمالسازی شده و معماری اطلاعات مناسبی باشیم مسیر حرکت برای تغییرات مشخص است و جلوی هر گونه اضافه کاری را می گیرد.

۶. بررسی موردی مهاجرت از مایکروسافت اس کیوال سرور^{۲۶} به

مای اس کیوال^{۲۷}

²⁵ Key Alternate

²⁶ MicroSoft SQL Server

²⁷ MySQL

یکی از متداول ترین موضوعاتی که مهاجرت کنندگان به مای اس کیو ال مطرح می نمایند مهاجرت و تبدیل داده های پایگاه داده می باشد . این سوال از طرف کاربران میکروسافت اس کیو ال سرور و اکسس مطرح می شود که مایل به مهاجرت به مای اس کیو ال چه در سطح سرور و چه در سطح کلاینت می باشند . توسعه دهندگان معمولاً به دنبال ابزاری هستند که داده هایشان را از کاربران میکروسافت اس کیو ال سرور و اکسس به مای اس کیو ال انتقال دهد .

در ابتدا از این مطلب شروع می کنیم که چه دلایلی وجود دارد که باید به مای اس کیو ال مهاجرت کرد؟ و برعکس چه دلایلی نیز برای عدم مهاجرت وجود دارد ؟ و سپس به توضیح مسائل تکنیکی کار می پردازیم.

۱-۶- چه دلایلی برای مهاجرت به مای اس کیو ال وجود دارد ؟

[<http://dev.mysql.com>]

دلایل زیادی برای مهاجرت به مای اس کیو ال یا حتی اضافه کردن امکان مای اس کیو ال به نرم افزارهای جاری شما وجود دارد که در زیر به چند نمونه اشاره می کنیم.

۱-۶-۱- مای اس کیو ال قابل اجرا روی پلت فرم های مختلف است

یکی از مهم ترین مزیت های^{۲۸} کراس پلت فرم^{۲۸} مای اس کیو ال بودن آن است . به این معنا که قابلیت اجرا بر روی اکثریت پلت فرم های موجود را دارا می باشد . شما می توانید پروژه پایگاه داده خود را بر روی یک ویندوز یا لینوکس خانگی توسعه داده و بر

²⁸ Cross Platform

روی پلت فرم های دیگری مانند، ویندوز سرور^{۲۹} ۲۰۰۳ انواع مختلف لینوکس ، پردازنده های مرکزی آی بی ام ، ماشین های اپل و بدون هیچ گونه تغییری اجرا کنید. این امکان به شما قابلیت انعطاف فراوان در انتخاب سخت افزار و سیستم عامل سرور می دهد . و در انتقال به یک پلت فرم جدید کمترین هزینه و اتلاف وقت را متحمل خواهید شد.

۲-۱-۶- مای اس کیو ال پایگاه داده بسیار سریع

در یک بررسی که توسط زیف دیویس^{۳۰} صورت گرفته مشاهده می شود که مای اس کیو ال تقریباً سریع ترین پایگاه داده در میان پایگاه داده هایی مانند اس کیو ال سرور ۲۰۰۰ ، اوراکل ، آ اس ای^{۳۱} و دی بی^{۳۲} می باشد . مای اس کیو ال هم اکنون توسط شرکت هایی مانند یاهو و سیسکو و ... که خواهان کارایی و استواری بالا هستند استفاده می شود . مای اس کیو ال می تواند برای رسیدن به حداکثر استفاده از سخت افزار به شما کمک کند.

۳-۱-۶- مای اس کیو ال نرم افزاری آزاد است

مای اس کیو ال یک نرم افزار متن باز می باشد . به این معنی که شما ازادید که کد منبع آن را در اختیار داشته و بررسی کنید و آن را برای خود تغییر دهید . با توجه به

²⁹ Windows Server 2003

³⁰ Ziff Davis

³¹ ASE

³² DB2

موافقت نامه جی پی ال^{۳۳} ان شما می توانید نرم افزار تغییر یافته را توزیع کرده و به فروش برسانید به شرطی که نرم افزار خروجی نیز متن باز باشد.

اگر شما نمی خواهید که نرم افزار خود را که بر پایه مای اس کیو ال است متن باز کنید شما ازادید که این کار را انجام دهید به شرطی که ان را به صورت عمومی و خارجی توزیع نکنید.

اگر شما با موافقت نامه جی پی ال این نرم افزار موافق باشید مای اس کیو ال برای شما هزینه ای نخواهد داشت . اما اگر مایلید که نرم افزار خود را به صورت متن بسته توسعه دهید و انرا به صورت متن بسته توزیع کنید بایستی از موافقت نامه های تجاری مای اس کیو ال که باز هم نسبت به نرم افزار های مشابه هزینه کمتری دارد (از \$۲۴۹ شروع می شود) استفاده کرده و از پشتیبانی تجاری بهره مند شوید.

۴-۱-۶- مشتریان متن باز را می خواهند

امروزه تمایل و رضایت مشتریان و استفاده کنندگان از نرم افزار های متن باز به طرز قابل توجهی افزایش داشته و به میزان بسیار زیادی در سرتاسر دنیا رسیده است . دلیلی که برای این موضوع می توان یافت کیفیت و به روز رسانی مناسب نرم افزار و از همه مهمتر " ازادی " در استفاده از نرم افزار است . ازادی به این معنی که شما دیگر اسیر موافقت نامه های تجاری و شرایط دست و پا گیر شرکت های تجاری نخواهید بود . موارد بسیار زیادی وجود دارد که شرکت ها نه به خاطر متن باز بودن مای اس کیو ال

³³ GPL

بلکه به خاطر فشار کاربران آن شرکت در استفاده از نرم افزار متن باز به مای اس کیو ال روی آورده اند.

۲-۶- چه دلایلی برای عدم مهاجرت به مای اس کیو ال وجود دارد ؟

در عین حال که مهاجرت به مای اس کیو ال مزایا و منافع بسیار زیادی را در بردارد در بعضی موارد این مهاجرت مناسب نیست . در زیر به بعضی از این موارد می پردازیم:

۱-۲-۶- کاربردهای ساده تک کاربره و قابل جابه جایی

کاربردهای بسیار زیادی وجود دارند که از اکسس برای نگهداری داده های خود استفاده می کنند . این کاربرد ها توسط فقط یک کاربر مورد استفاده قرار می گیرد و در جایی استفاده می شود که می توان برای انتقال کل پایگاه داده فایل های داده را به راحتی کپی کرد . یعنی کل کار بسیار کم حجم و ابتدایی است.

واقعیت این است که مای اس کیو ال برای این نوع کاربرد مناسب نیست . مای اس کیو ال به صورت یک سرور چند کاربره طراحی شده است و برای موارد دسترسی هم زمان به یک فایل داده و اولویت بندی استفاده از داده ها بسیار مناسب می باشد . این بدین معنی نیست که نمی توان از مای اس کیو ال در این کاربرد استفاده کرد بلکه این کاربرد از سطح کارایی مای اس کیو ال بسیار پایین تر است و مای اس کیو ال توانایی های بسیار بیشتری دارد . اما از طرفی اجرای مای اس کیو ال باعث گرفتن منابع سیستم می شود در حالی که در این حد نیاز نبوده است.

از طرفی مای اس کیو ال یک سرور جاسازی شده^{۳۴} می باشد که برای وارد کردن پایگاه داده به نرم افزار بسیار مناسب است ولی ای پی آی^{۳۵} مشخص خود را دارد که به عنوان عنوان مثال در مورد تکنولوژی هایی مانند آدو^{۳۶} به راحتی قابل مهاجرت نمی باشد.

۶-۲-۲ کاربرد هایی با امکانات ناسازگار

مای اس کیو ال ای بی^{۳۷} در حال اضافه کردن امکانات بسیار زیادی به مای اس کیو ال می باشد . ولی باید همیشه این مطلب را در نظر داشت که امکاناتی در اس کیو ال سرور یا اکسس وجود دارد که در مای اس کیو ال موجود نیست و بالعکس . اگر شما از مای اس کیو ال ۴ استفاده می کنید فقدان امکاناتی مانند جملات آماده شده یا رویه های ذخیره شده را خواهید دید که تاثیر به سزایی در سهولت مهاجرت به کاربرد های مای اس کیو ال دارند . این مطلب به نحوه استفاده شما از این امکانات ناسازگار نیز مربوط می شود.

۶-۲-۳ کاربردهای بزرگ بدون تجرید^{۳۸} در طراحی پایگاه داده

یکی از بزرگترین مواردی که هر توسعه دهنده کاربردهای پایگاه داده باید در نظر بگیرد یا تجرید در طراحی نرم افزار پایگاه داده است . اگر شما از تجرید در طراحی نرم افزار خود بهره برده باشید خواهید دید که مهاجرت از اکسس و اس کیو ال سرور چقدر ساده و سریع خواهد بود.

³⁴ Embedded

³⁵ API

³⁶ ADO

³⁷ MySQL AB

³⁸ Abstraction

اگر نرم افزار شما کوچک است و در آن تجرید به کار برده نشده است نسبتاً با تغییر کد زیادی مواجه نخواهید شد . پیچیدگی در مهاجرت شما و زمانی را که برای آن صرف خواهید کرد با بزرگ شدن کاربرد افزایش می یابد . تمامی این موارد بدین معنی نیست که مهاجرت یک نرم افزار بزرگ به مای اس کیو ال غیر ممکن می باشد بلکه مستلزم صرف وقت و هزینه می باشد .

۳-۶- طرح ریزی برای مهاجرت

اولین کاری که برای مهاجرت به مای اس کیو ال باید انجام دهید طرح ریزی و برنامه ریزی برای تبدیل و مهاجرت می باشد . در طرح خود بایستی مواردی مانند تغییر در نوع داده , خود داده , نحوه ذخیره شدن داده و ... را مد نظر داشته باشید . همچنین با تغییر در سرور خود به مای اس کیو ال تغییرات عمده ای در کلاینت خود بایستی انجام دهید مواردی مانند تغییر در استفاده از نشانگر ها , رویه های ذخیره شده . شما همچنین باید برای نگهداری و به روز کردن نرم افزار خود پس از مهاجرت به مای اس کیو ال برنامه ریزی کنید . و در نهایت بایستی نقاط ضعف و قوت مای اس کیو ال , اس کیو ال سرور , اکسس را مشخص سازید تا در این مهاجرت بیشترین سود را کسب نمایید . در زیر اشاره کوتاهی به این موارد می کنیم :

۱-۳-۶- نوع های داده ای

نوع های داده ای مای اس کیو ال و اس کیو ال سرور تطابق بسیار زیادی با یکدیگر دارند ولی با این وجود تفاوت های کمی نیز وجود دارد . شما بایستی در طرح خود به

بررسی نوع های داده ای پایگاه داده خود و نوع های داده ای مای اس کیو ال بپردازید که در مهاجرت بتوانید از بهترین تطابق نوع های داده ای استفاده کنید.

مواظب یک موضوع باشید : شما بایستی در نوع داده و طول و گنجایش ان تطابق ها را پیدا کنید نه در اسم نوع های داده ای . به عنوان مثال نوع داده ای VARCHAR در مای اس کیو ال می تواند ۲۵۵ کاراکتر نگه دارد در حالی که VARCHAR در مای اس کیو ال سرور دارای طول ۴۰۰۰ کاراکتر می باشد . در این مورد نوع داده ای TEXT از مای اس کیو ال با VARCHAR از مای اس کیو ال سرور تطابق دارد.

بعضی از نوع های داده ای بین مای اس کیو ال سرور و مای اس کیو ال و اکسس هیچ ارتباطی با یکدیگر ندارند . به عنوان مثال نوع داده ای CURRENCY هنوز در مای اس کیو ال وجود ندارد اما تعریف یک ستون با استفاده از DECIMAL همین قابلیت را ایجاد می کند.

پیش فرض مای اس کیو ال سرور یونیکد^{۳۹} می باشد مانند نوع های داده ای nCHAR و nVARCHAR در حالی که مای اس کیو ال به صورت واضح کاراکتر ست را به نوع فیلد ها مربوط نمی کند در عوض به شما این امکان را می دهد که برای یک سری از داده کاراکترست های مختلف از جمله یونیکد را به کار ببرید.

۲-۳-۶- تغییر در داده

بعضی اوقات شما نیاز دارید که برای تبدیل داده به نوع های دیگر خود داده را تغییر بدهید . به عنوان مثال تاریخ در مای اس کیو ال به صورت YYYY-MM-DD ذخیره می

³⁹ UNICODE

شود در حالی که در اس کیو ال سرور به صورت MM-DD-YYYY ذخیره می شود.
در این موارد اگر ابزار تبدیل شما این موارد را متوجه شده و حل نماید کار شما بسیار ساده خواهد بود . در غیر این صورت اگر خودتان ابزار تبدیلتان را می نویسید بایستی تمامی این موارد را در نظر داشته باشید . موارد دیگر تغییر در داده شامل تغییرات طرح داده برای نرمال کردن جداول پایگاه داده می باشد تا عمل تبدیل صورت گیرد .

۳-۳-۶- توابع داخلی

بیشتر توابع داخلی مای اس کیو ال و اس کیو ال سرور یکسان است هر چند ممکن است که در نام گذاری متفاوت باشند . به عنوان مثال تابع ISNULL در مای اس کیو ال به نام IFNULL در مای اس کیو ال وجود دارد . و از سینتکس^{۴۰} یکسانی برخوردار می باشد .

مای اس کیو ال توابع داخلی بسیار بیشتری نسبت به اس کیو ال سرور دارد و از این جهت برای اجرای کوئری های متنوع شما بسیار مناسب است.

۴-۳-۶- نشانگر ها

اغلب نرم افزارهای های تحت ویندوز از نشانگر های دینامیک سمت سرور یا کی ست^{۴۱} برای رسیدن به داده ها از طریق API هایی مانند ADO بهره می برند . درایور Connector/ODBC از نشانگر های مشتق شده از کی ست پشتیبانی نمی کند و پشتیبانی آن از نشانگر های سمت سرور نیز بسیار محدود می باشد .

⁴⁰ Syntax
⁴¹ KEYSET

۵-۳-۶- توابع تعریف شده توسط کاربر

تعریف توابع توسط کاربر در اس کیو ال سرور و مای اس کیو ال یکسان نمی باشد . در اس کیو ال سرور توابع بسیار شبیه رویه های ذخیره شده هستند و شما اجازه دارید تا یکسری از کوئری ها را تحت عنوان یک تابع جدید به کار ببرید . در حالی که این مساله در مای اس کیو ال به صورت کدهای C کامپایل شده قابل پیاده سازی است . هنگامی که کد C شما کامپایل می شود می توانید آنرا وارد سرور کرده و تابع آنرا صدا بزنید.

۶-۳-۶- رویه های ذخیره شده

این امکان به طور آزمایشی در نسخه مای اس کیو ال ۵ پیاده سازی شده و مرتبا در حال تکمیل است . قابل ذکر است که هیچ گارانتی وجود ندارد که در مهاجرت به رویه های ذخیره شده , مای اس کیو ال ها دستخوش تغییر نشوند.

۴-۶- طرح ریزی برای نگه داری

علاوه بر طرح ریزی برای مهاجرت به مای اس کیو ال و تبدیل داده ها به این پایگاه داده شما باید مساله استراتژی نگه داری و ابزار های پایگاه داده خود را طرح ریزی کنید . یکی از مهم ترین مسایل , مساله پشتیبان گیری^{۴۲} مداوم از داده ها می باشد که خوشبختانه ابزار های مشابه و متعددی برای مای اس کیو ال و اس کیو ال سرور وجود دارد.

⁴² Back up

۵-۶- نتیجه گیری

در این بررسی موردی به وضوح مشاهده می نماییم که در ابتدای مهاجرت بعد مفهومی آن مد نظر قرار می گیرد و پس از آن که در مورد اینکه آیا مهاجرت به صورت اصولی در این امر قابل اعمال است به بررسی های بعدی آن می پردازد. در ابتدا باید دلایلی موجه از علل ناکارآمدی پایگاه داده فعلی در مقایسه با پایگاه داده ای که می خواهیم بیابیم و پس از آن به مسائل تکنیکی این مهاجرت بپردازیم در مسائل تکنیکی مهاجرت نیز عوامل مختلفی مطرح هستند که در این بررسی به صورت خلاصه به مهمترین آنها اشاره شده است.

۷. بررسی موردی مهاجرت از میکروسافت اس کیو ال سرور به

اوراکل

به جهت اینکه عموماً مهاجرت از میکروسافت اس کیو ال به اوراکل به عنوان یکی از چالش های سازمان هایی است که ورود داده به پایگاههای داده آنها از مقدار پیش بینی شده بیشتر است. در ابتدا باید به این نکته اشاره نمود که آیا این مهاجرت ضروری است؟ پاسخ به این سوال بستگی به شرایط مختلفی دارد ولی عموماً این مهاجرت غیر ضروری است و ناشی از عدم طراحی مناسب پایگاه داده میکروسافت اس کیو ال سرور است. به این گزارش که شرکت وینتر در سال ۲۰۰۵ [ماهنامه شبکه - آبان ۱۳۸۵ شماره ۷۰] دقت نمایید شاید علت بیان این مسئله روشنتر گردد:

"در فهرست ده پایگاه داده نخست جهان، مکان هشتم متعلق به شرکتی است که از دیتابیس اس کیو ال سرور ساخت مایکروسافت استفاده می کند. این دیتابیس نوزده ترابایتی که روی سیستم عامل ویندوز برپا شده، متعلق به شرکت یونیسیس پیمنت سرویس اند سلوشنز^{۴۳} است که شاخه ای از شرکت بزرگ یونیسیس است. این شرکت که در هلند مستقر است، به تعدادی از بانک های اروپا خدمات بانکداری الکترونیکی ارائه می دهد."

حال نکته این است که چگونه پایگاه داده ای که به خوبی با این حجم خدمات ارائه می دهد نمی تواند پاسخگوی حجم های بسیار کوچکتر در حد چندین گیگا بایت باشد. نتیجه این موضوع این است که مطمئناً اشکال در طراحی مناسب پایگاه داده بوده است. البته این مطلب به هیچ وجه انکارکننده توانایی های فوق العاده اوراکل به خصوص در حجم های بالا نمی شود ولی با توجه به اینکه نسبت کارایی به هزینه مایکروسافت اس کیو ال سرور عموماً نسبت به اوراکل بیشتر است تا جائیکه واقعاً می توان از مایکروسافت اس کیو ال سرور استفاده کرد باید از مهاجرت به اوراکل اجتناب نمود. برای مقایسه بهتر مایکروسافت اس کیو ال سرور و اوراکل می توانید به گزارشی که سایت www.wisdomforce.com ارائه نموده است مراجعه نمود.

۸. نتیجه گیری

مهاجرت از پایگاه داده ای به پایگاه داده دیگر هزینه های نسبتاً زیادی را به سازمان تحمیل می کند بدین جهت باید این مهاجرت با دقت کامل صورت

⁴³ Unisys Payment Services & Solutions

گیرد. برای انجام مهاجرت باید ابتدا به این جمع بندی برسیم که آیا این مهاجرت با توجه به هزینه هایی که در بردارد مثمر ثمر است یا خیر؟ و پس از آن در بعدهای تکنیکی به بررسی پردازیم.

۹. مراجع

1. <http://dev.mysql.com/tech-resources/articles/migrating-from-microsoft.html>

ترجمه و تکمیل : سعید تقوی

2. <http://www.shabakeh-mag.com/Articles/Show.aspx?n=1002697> ماهنامه

شبکه شماره ۷۰

3. Management Information Systems, Laudan & Laudan, 2006, Ninth Edition

4. www.persianforum.com