Develop An Analytical and Design Methodology for Multidimensional Databases for Data Warehouses

Dr. Ihab Aldibaja*

(Received 29 / 11 / 2021. Accepted 3 / 3 / 2022)

□ ABSTRACT □

The complexity of business in companies has increased and the amount of data that these companies deal with has increased, and normal database systems are no longer able to meet the requirements of the decision maker in terms of speed, accuracy and the volume of data that they can deal with, so the idea of data warehouses has emerged, which is one of the most powerful tools Decision support systems DSS, and the urgent need for them has emerged in major companies, as each of these companies manages its own independent databases (marketing, financial...) which include a lot of shared data and the company's need requires finding all the data related to a specific topic, such as customers, for example. By searching in one database instead of searching in different databases Data warehouses are easier for the user in terms of querying and understanding than relational databases that are designed to handle many operations. In this paper, we discuss the development of an analytical and design methodology for building data warehouses, where we define the stages and basic questions to be asked at each stage in order to reach a data warehouse that meets the needs of the organization and allows answering the required queries The results showed that following the methodology presented in the research led to shortening the time needed to build the data warehouse and reducing the time and size of modifications required to reach the final result.

Keywords: Datawarehouse – Software Engineering – Business Intelligence.

* **Assistant Professor – Faculty of Engineering – Informatics Department – Almanara University – Lattakia – Syria. Email: ihab.aldibaja@manara.edu.sy**

تطوير منهجية تحليلية وتصميمية لقواعد البيانات متعددة الأبعاد الخاصة بمستودعات البيانات

د. إيهاب الديباجة*

(تاريخ الإيداع 29 / 11 / 2021. قُبِل للنشر في 3/ 3 / 2022)

□ ملخّص □

لقد ازداد تعقيد الأعمال في المؤسسات، كما ازدادت كمية البيانات التي تتعامل معها هذه المؤسسات، ولم تعد أنظمة قواعد المعطيات العادية قادرةً على تلبية متطلبات متخذ القرار من حيث السرعة والدقة وحجم البيانات الذي تستطيع التعامل معه، لذلك ظهرت فكرة مستودعات البيانات، والتي تعد من أقوى أدوات نظم دعم القرار Decision Support Systems (DSS)، وبرزت الحاجة الملحة إليها في الشركات الكبرى، حيث أن كل قسم من هذه الشركات يدير قواعد بيانات مستقلة خاصة به (تسويق، مالي...)، والتي تتضمن كثيراً من البيانات المشتركة، وتتطلب حاجة الشركات حالياً إيجاد كامل البيانات المتعلقة بموضوع محدد، كالزبائن مثلاً، من خلال البحث في قاعدة واحدة، بدلاً من البحث في القواعد المختلفة.

نبحث في هذه الورقة تطوير منهجية تحليلية وتصميمية لبناء مستودعات البيانات، حيث نحدد المراحل والأسئلة الأساسية الواجب طرحها في كل مرحلة للوصول إلى مستودع بيانات يلبي حاجة المؤسسة، ويسمح بالإجابة على الاستعلامات المطلوبة.

بينت النتائج أن اتباع المنهجية المطروحة في البحث قد أدى إلى اختصار الزمن اللازم لبناء مستودع البيانات وتقليل زمن وحجم التعديلات المطعلالوبة للوصول إلى النتيجة النهائية.

الكلمات المفتاحية: مستودعات البيانات – هندسة برمجيات – ذكاء الأعمال

* **مدرس – كلية الهندسة – قسم المعلوماتية – جامعة المنارة– اللاذقية – سورية.Email: ihab.aldibaja@manara.edu.sy**

مقدمة:

تعدّ مستودعات البيانات بأسلوب بنائها وتمثيل بياناتها أسهل بالنسبة للمستخدم من ناحية الاستعلام والفهم أكثر من قواعد البيانات علائقية التي صممت للتعامل مع العديد من العمليات، كما أن الاستعلامات التي سوف تكون معقدةً جداً في قواعد البيانات التي خضعت لعملية التقييس Normalized Database سوف تصبح أسهل في التصميم والاستخدام في مخازن البيانات التي لا تخضع فيها قاعدة البيانات للعملية السابقة، بل تخضع إلى عملية De-normalize المعاكسة لعملية التقييس، وذلك لتحسين الأداء والفعالية، من خلال تقليل تعقيد بنية قاعدة البيانات.[1]

تختلف طريقة معالجة البيانات بين قواعد البيانات ومخازن البيانات، فالأولى تستعمل المعالجة الآنية On Line Transaction Processing (OLTP) حيث أنه يمكن تعديل، زيادة أو البحث عن البيانات بسرعة، مع الاحتفاظ بتكامل قاعدة البيانات Database Integrity ، أما مستودعات البيانات فتستخدم طريقة تحليل البيانات On Line Analysis Processing (OLAP) من خلال معالجة كميات ضخمة، وتحليلها، وإخراجها بشكل معرفة Knowledge تسمح باستنتاجات وتحديد اتجاهات جديدة بدون أية إمكانية لتعديل البيانات، كما تسمح بطرح أسئلة غير متوقعة تأخذ أحياناً وقتاً طويلاً لإصدار النتائج.[2]

مشكلة البحث

إن منهجيات هندسة البرمجيات التقليدية لبناء وتطوير النظم لا يمكن أن تساعد بشكل فعال في تطوير وبناء مستودعات البيانات نظراً للطبيعة الخاصة لهذه النظم[3]، فهي تتطلب فهم المشكلة على شكل مسألة ذات أبعاد متعددة، وفهم العلاقة بين هذه الأبعاد والبنية الهرمية للأبعاد، وكل هذه المعطيات لا تأخذها المنهجيات التقليدية بعين الاعتبار، وبالتالي، كان لابد من طرح منهجية تدمج بين الدورة التقليدية لحياة البرمجيات وبين المتطلبات الخاصة بمستودعات البيانات.

أهمية البحث وأهدافه:

أهمية البحث

تكمن أهمية البحث في دراسة أثر اتباع المنهجية المطروحة الخاصة بتحليل وتصميم مستودعات البيانات على أداء هذه العمليات، من ناحية تقليل عدد الأخطاء في التحليل والتصميم، والزمن اللازم للوصول إلى النتيجة النهائية، مقارنةً مع اتباع الطرق التقليدية في هندسة البرمجيات، حيث تم هنا مقارنة المنهجية المقترحة مع منهجية الشلال Waterfall.

أهداف البحث

يهدف البحث إلى دمج المفاهيم الخاصة بالنمذجة البعدية لقاعدة البيانات مع المفاهيم التقليدية لتصميم البرمجيات للوصول إلى منهجية متكاملة، تسمح بتصميم مستودع بيانات بأبعاد تغطي المشكلة المدروسة، وبأقل عدد من أخطاء التحليل والتصميم.

طرائق البحث ومواده:

تم استخدام المقارنة الإحصائية لعدد الأخطاء التي تم تصحيحها في المنهجية التقليدية والمنهجية المقترحة، بالإضافة للدراسة الإحصائية للزمن اللازم لتنفيذ المهام وعلاقته مع زيادة عدد المهام المطلوبة، وتم من خلال المقارنة استنتاج التحسين الذي تم الوصول إليه باستخدام المنهجية المقترحة.

مجتمع وعينة البحث

تم تطبيق البحث في الشركة العامة لمرفأ اللاذقية والتي تضم أنظمة معلوماتية تغطي عمل المرفأ، وتم مقارنة تصميم مستودع بيانات بالطريقة التقليدية لنظام الإيرادات، وبالطريقة المقترحة لنظام المانيفست الالكتروني، ومقارنة عدد الأخطاء التي تم تصحيحيها في كلا الحلين، والمقارنة بين زمن إنجاز المهام في كل سيناريو.

1- الإطار النظري للبحث

1-1- مستودعات البيانات:

تُعرّف مستودعات البيانات بأنها أنظمة تستخدم تقنيات جديدة في تخزين كميات كبيرة من المعلومات المتكاملة وغير المتشابهة بهدف استعمالها في اتخاذ القرار، لذلك تسمى أنظمةً تقريريةً في مقابل الأنظمة الإنتاجية التي منها قواعد البيانات، حتى أن الهيكليات Architecture المستخدمة صممت على أساس استخراج المعرفة للمساعدة في اتخاذ القرار.

يتمتع مستودع البيانات بالصفات التالية: متكامل، موجه لموضوع محدد، متعلق بالزمن، غير متطاير.[4]

- متكامل Integrated: مستودع البيانات عبارة عن تجميع مركزي للمعطيات حيث يكامل بين معطيات العمل القادمة من مصادر مختلفة، كما يجب أن تُمثَّل المعطيات في مستودع البيانات بصيغة مشتركة مُتفق عليها.
- موجه لموضوع معين Subject-Oriented: أي انها توفر المعلومات حول موضوع معين بدلا من تخزين البيانات للعمليات الجارية للمؤسسة.
- متعلق بالزمن Time-Variant: يمثل مستودع البيانات تدفق المعطيات عبر الزمن، وعندما يتم تحميل مجموعة جديدة من المعطيات إلى المستودع، تتم عملية حساب جديدة لمستويات التجميع المرتبطة بالزمن.
- غير متطاير Non-Volatile: عندما يتم تحميل المعطيات إلى مستودع البيانات لا يمكن حذفها أبداً، لأنها أصبحت بيانات تاريخية، ويفترض أنها مرت بعمليات تنظيف وتهيئة.

1-2- بعض الوظائف داخل مستودع البيانات[5] :

- Rollup: تلخيص البيانات بشكل متزايد (أسبوعيا، سنويا،...) أو يمكن التعبير عنها في قواعد البيانات متعددة الأبعاد، بأنها الانتقال عبر تسلسل الأبعاد من الأدنى (الأبناء) إلى الأعلى (الآباء).
- Drilldown: بعكس Rollup تتم هنا زيادة مستوى تفصيل البيانات.
- Pivot (Rotation): الانتقال من بعدٍ إلى بعدٍ آخر (تغيير وجهات النظر إلى البيانات).
- Sorting: ترتيب البيانات ضمن مستودع البيانات.
- Slice: أخذ شريحة من مستودع البيانات ويتم ذلك بناء على قيمة مفردة لأحد الأبعاد.
- Dice: اختيار مكعب جزئي من مكعب أكبر.

1-3- تخزين النموذج متعدد الأبعاد *Multidimensional Model Storage*:

تخزين النموذج متعدد الأبعاد يشمل نوعين من الجداول[6] :

- جداول الأبعادDimension Table: وحقوله تسمى سمات البُعد Dimension Attributes.
- جدول الحقيقة Fact Table: ويتضمن مقاييس أداء العمل كما يحتوي على مؤشرات لجداول الأبعاد.

1-4- بنية مستودع البيانات المستخدم في البحث:

تم اعتماد المعايير التالية في بناء مستودع البيانات المستخدم في البحث، وهي المعايير الأبسط والأكثر شيوعاً، وذلك لتقليل دور بنية مستودع البيانات في المقارنة بين الطريقة التقليدية في التحليل والتصميم والطريقة المقترحة:

- نموذج مستودع البيانات: Relational OLAP ROLAP يستطيع المستخدمين أن يروا بياناتهم منظمة في مكعب ولكنها في الحقيقة تكون مخزنةً في قاعدة بيانات علائقية، ومتحكماً بها باستخدام برنامج إدارة قواعد بيانات علائقية RDBMS مثل ORACLE، وتخزن البيانات بنوع من التفصيل، ويكون زمن الاستجابة بطيء نسبياً.[7]
- بنية جداول مستودع البيانات: Star Schema يعدّ من أبسط أشكال تحقيق مستودعات البيانات، ويدعى بهذا الاسم لشبه شكل المخطط بالنجمة. يتكون مركز المخطط من جدول حقيقة واحد أو أكثر، ويكون هناك جدول وحيد لكل بعد، يعيبه الحاجة لمساحة كبيرة للتخزين، ومع ذلك فإن شركة اوراكل تنصح باستخدام هذا النموذج نظراً لأنه يوفر الأداء الأحسن، فهو يحافظ على الاستعلامات بسيطة، ولا يحتاج لأكثر من عملية جداء واحدة بين جدول الحقيقة وجداول الأبعاد.[8]
- يمتلك جدول الحقيقة نوعين من الأعمدة، الأول يحتوي حقائق رقمية تدعى القياسات Measurement والنوع الثاني هو مفاتيح خارجية إلى جداول الأبعاد.
- يمكن أن يحتوي جدول الحقيقة قيماً مجمعة Summarized Values وعندها يدعى بجدول الخلاصة Summery Table، وعادة تكون البيانات مجمعةً على نفس المستوى في جدول الحقيقة.
- المفتاح الرئيسي لجدول الحقيقة مكون من مجموع المفاتيح الخارجية لجداول الأبعاد.[9]
- البيانات الخاصة بجداول الأبعاد تجمع على أصغر مستوى، ثم تتم عملية تجميع للقيم في مستويات أعلى لتصبح أكثر فائدة في عمليات التحليل.

1-5- عملية التحليل والتصميم العامة التي تم اعتمادها في البحث:

تم اعتماد المراحل التالية كخطوات عامة في تحليل وتصميم الوظائف في مستودع البيانات، من أجل مقارنتها مع المنهجية المطورة، وهي تعتبر خطوات مشابهة لمنهجية الشلال التقليدية لتطوير البرمجيات، ولكن مع تخصيص المراحل لتناسب بناء مستودعات البيانات:

1. تحليل احتياجات العمل وتصميم خطة العمل Analyze business needs and design a plan: يجب أن تكون قاعدة البيانات التي ننشئها ترضي الحاجة إلى المعلومات بالنسبة للشركة ولمستخدمي النظام، لذلك يجب تحديد مصدر البيانات والمعلومات التي سيحصل عليها المستخدمين مع أخذ المتطلبات الأمنية بعين الاعتبار.
2. إرساء تصميم قاعدة البيانات Draft a database outline: وتشمل تحديد بنية قاعدة البيانات، المعلومات التي سيتم تخزينها والعلاقات بين عناصر البيانات المختلفة.
3. التحقق من متطلبات النظام Check system requirements: يجب تحديد متطلبات النظام، وتحديد العوامل والعناصر التي تؤثر على أداء وفعالية النظام.
4. تحميل بيانات اختبارية إلى قاعدة البيانات Load test data into the database.
5. تعريف الحقول الحسابية Define calculation fields.

6. تعريف التقارير Define reports: إن عملية وصول المستخدمين للبيانات سوف تكون من خلال التقارير ولذلك يجب تحديد الخطوط العامة للتقارير واختبارها.
7. تحديد المستخدمين الذين سيكون النظام موجها لخدمتهم ويجب أن نأخذ وجهة نظر المستخدمين بعين الاعتبار.
8. يمكن إعادة الخطوات السابقة حتى الوصول إلى التطبيق المقصود.

الشكل (1) الخطوات العامة لبناء مستودع البيانات بالطريقة التقليدية

النتائج والمناقشة:

تم في البحث وضع تفصيل لمنهجية تحليلية وتصميمية لمستودع البيانات تراعي خصوصية مستودع البيانات وتخزين النموذج متعدد الأبعاد، أي أنه تم التركيز على المرحلة 1 والمرحلة 2 (مرحلة التحليل والتصميم) كونهما المراحل الأساسية والجوهرية في بناء مستودع بيانات قائم على قاعدة بيانات متعددة الأبعاد، حيث تلعب مرحلة التحليل والتصميم دوراً أساسياً في إنشاء قاعدة بيانات تلبي الاحتياجات المطلوبة وتسمح بتحقيق أهداف مستخدمي النظام، بالإضافة إلى توفيرها نظام متين قابل للصيانة والتحديث ومواكبة متطلبات العمل.

تركز مرحلة التحليل والتصميم على العمليات التالية:

1. تحليل البيانات التي ستشكل مصدرا للمعلومات في النظام الذي نصممه.
2. تحديد متطلبات مستخدمي النظام.
3. تحديد المتطلبات والفعاليات المتعلقة بأمن النظام والمعلومات في البيئات متعددة المستخدمين.
4. إنشاء نموذج قاعدة البيانات.

يجب توجيه عناية خاصة إلى النقاط التالية، وذلك عند تصميم قاعدة البيانات لأغراض تحليل متطلبات العمل:

- طبيعة وتدفق المعلومات بين الأقسام المختلفة للمؤسسة.
- أنواع التقارير المطلوبة والبيانات المطلوب تضمينها في هذه التقارير والأسلوب الأمثل لتوليدها.

1- مرحلة تحليل مستودع البيانات:

1- تحليل البيانات التي ستشكل مصدراً للمعلومات في النظام الذي نصممه: يجب أن نقوم بعملية فحص للبيانات التي ستخزن في قاعدة البيانات من حيث الأنظمة التي تضم هذه البيانات، وكيفية الحصول عليها، وحاجة هذه البيانات للترقية والصيانة، كما يجب تحديد الدقة Granularity التي ستخزن بها هذه البيانات، فهل سنخزن المعلومات عن أقسام المنتجات أم عن المنتجات بشكل مفصل على سبيل المثال، وهذا يتطلب التعرف على الهيئة الحالية لتخزين البيانات والشكل المتوقع والمطلوب في المستقبل.

كما تشمل هذه المرحلة تحديد أشكال التقارير وأنواع التحليل المطلوب تطبيقه على البيانات وذلك من خلال الإجابة على الأسئلة التالية:

- كيف يريد المستخدمين عرض وتحليل البيانات؟
- ما هو مستوى التفصيل المطلوب للبيانات المخزنة في قاعدة البيانات؟
- هل البيانات الموجودة حاليا تفي بأغراض التحليل المطلوبة؟
- إذا لم يتحقق الشرط السابق، ما هي البيانات الواجب إضافتها لتلبي متطلبات مستخدمي النظام؟

كما يتم في هذه المرحلة تحديد الموقع الحالي الذي تخزن فيه البيانات تمهيداً لدراسة وتحديد كيفية استخلاص هذه البيانات ونقلها إلى النظام الجديد ويمكن طرح الأسئلة التالية في هذه الناحية:

- أين يخزن كل قسم في المؤسسة البيانات الخاصة به؟
- هل البيانات مخزنة بصيغة يمكن استخدامها بشكل مباشر لأغراض التحليل؟
- هل البيانات مخزنة باستخدام مخدم مفرد أم موزعة وهل لها شكل علائقي وما هي نوعية نظام التشغيل الذي يشغل مخدمات قواعد البيانات حيث تتواجد؟
- من يحدث البيانات وما هي الفترات الزمنية الفاصلة بين التحديثات المتتالية؟
- هل من يقوم بعملية التحديث يملك الحق في الوصول إلى البيانات أي تحديد صلاحيات المسؤول عن تحديث البيانات؟
- هل البيانات الموجودة جاهزة للاستخدام فورا؟

2- تحديد متطلبات المستخدمين: يجب أن يكون هناك توصيف كامل لمتطلبات مستخدمي النظام، وأن يشمل هذا التوصيف جميع نواحي متطلباتهم، من تقارير وعمليات تحليل، كما يجب تحديد صلاحيات المستخدمين في الوصول إلى البيانات المصدر.

3- تحديد المتطلبات والفعاليات المتعلقة بأمن النظام والمعلومات في البيئات متعددة المستخدمين: يتم في هذه المرحلة بشكل أساسي إنشاء قائمة بالمستخدمين المتوقعين للنظام والصلاحيات التي يجب أن تمنح لهم، والأسئلة التالية تساعد في هذا المجال:

- من هم المستخدمين وما هي الصلاحيات التي ستمنح لهم؟
- من يملك حق التحكم بالصلاحيات؟
- هل يمكن تجميع المستخدمين بشكل تمنح معه كل مجموعة صلاحيات واحدة؟

4- إنشاء نموذج قاعدة البيانات: بعد تنفيذ الخطوات السابقة نكون مؤهلين لإنشاء تصميم أولي لقاعدة البيانات على الورق، ويجب هنا تحديد الاعتبارات المهمة في العمل، ومن ثم محاولة تحويل هذه الاعتبارات إلى أبعاد في قاعدة البيانات المتعددة الأبعاد التي نقوم بإنشائها.

معظم أعمال التحليل على البيانات تتم من خلال الأبعاد التالية:

- البعد الزمني Time Dimension.
- البعد الجغرافي Geographical Dimension.
- بعد المنتجات Products Dimension.
- البعد الخاص بكمية المنتج Product Quantity Dimension.
- بعد قنوات التوزيع Distribution Channels Dimension.
- الحسابات Calculations.

تساعد التلميحات التالية على تحديد الاعتبارات الأساسية للعمل على إيجاد قاعدة بيانات تلبي الغرض المطلوب منها:

تحديد الغرض من عمليات التحليل: بعد تحديد متطلبات العمل الأساسية، تكون الخطوة التالية هي تحديد كيف ستقوم قاعدة البيانات بتلبية هذه المتطلبات:

- بالنسبة للتحليل عبر الزمن يجب تحديد مدة حياة البيانات، أي عدد السنين التي ستخزن البيانات خلالها بغرض التحليل، وما هو شكل تقسيم البيانات زمنياً، هل هو شهري أم فصلي أو ربع سنوي...
- وبالنسبة للتحليل عبر المناطق الجغرافية، يجب تعريف ماهية المنطقة الجغرافية المقصودة هل هي مدن أم بلدان أم قارات...
- أما إذا كان التحليل عبر بعد المنتجات، فما هي البيانات المطلوب الحصول عليها حول المنتجات وكيف يتم تصنيف المنتجات.
- بالنسبة للحسابات، هل سيتم تخزين الحسابات ضمن حقول بعد تصميم معادلة الحساب، أم سوف تحسب في وقت التنفيذ؟

تحديد الأبعاد وأعضاء هذه الأبعاد: يمكن النظر إلى كل اعتبار من اعتبارات العمل على أنه بعد مستقل، ويمكن تحديد كل منظور للبيانات على أنه بعد منفصل في قاعدة البيانات، فمثلاً يمكن التحليل من خلال لون المنتج أو وزنه أو قياسه أو قناة البيع الخاصة به...

ويمكن النظر إلى الاعتبارات التالية لتحديد تصميم غير نهائي لأبعاد قاعدة البيانات:

- العلاقات بين الأبعاد.
- دراسة مثال عن قاعدة البيانات.
- قائمة تحدد الأبعاد والأعضاء.

بعد تحديد تصميم أبعاد قاعدة البيانات، يجب اختيار العناصر ضمن كل بعد، هذه العناصر تصبح أعضاء في الأبعاد التي تنتمي إليها. مثلاً بعد الزمن يتضمن الفترات الزمنية التي نرغب في تحليل البيانات وفقها، مثل أرباع السنة، وضمن أرباع السنة تكون الأشهر. كل ربع سنة يصبح عضواً ضمن بعد الزمن، تمثل الأرباع والأشهر مستويين لأعضاء بعد الزمن.

تحديد العلاقة بين الأبعاد: إن العلاقة بين عناصر العمل المطلوب تحليله، تترجم إلى علاقة بين الأبعاد التي تمثل هذا العمل، وهو ما يسمح بتحليل سهل للبيانات من وجهات النظر المختلفة، فمثلاً، المحلل المالي يمكن أن يوجه الأسئلة التالية:

- ما هي المبيعات لشهر معين؟ يجب وضع مخطط لمقارنة المبيعات للسنوات الخمس الأخيرة.
- ما هي النسبة المئوية لتزايد الأرباح؟

- ما هو مقدار التطابق بين القيم الحالية والقيم المخطط لها؟

إن الأسئلة السابقة تحلل بيانات قاعدة البيانات من 3 وجهات نظر مختلفة، الزمن والحسابات والمخططات، وإن وجهات النظر الثلاث هذه، تمثل 3 أبعاد مختلفة:

- الزمن: على المحور x ويتضمن الأشهر والمجاميع لأرباع السنة.
- الحسابات: على المحور y يتضمن المبيعات والهوامش.
- قيم الميزانية: على المحور z ويتضمن على سبيل المثال القيم المخططة والقيم الفعلية.

كل خلية في المكعب تتضمن بيانات محددة بالأبعاد الثلاثة للمكعب.

الجدول التالي يضم مثالاً عن بعد الزمن:

الجدول (1) مثال عن بعد الزمن في قاعدة بيانات لمستودع بيانات

Sample Time Dimension		
Dimensions	Members	Child Members
Year	Qtr1	Jan, Feb, Mar
	Qtr2	Apr, May, Jun
	Qtr3	Jul, Aug, Sep
	Qtr4	Oct, Nov, Dec

أما بالنسبة للقائمة التي تحدد الأبعاد وأعضاء هذه الأبعاد، فيمكن الاستعانة بالأسئلة التالية لوضع هذه القائمة:

1. ما هو المرشح لان يكون بعداً؟
2. هل يصف أي من الأبعاد بعداً آخر. في هذه الحالة تكون هذه الأبعاد مرشحةً لأن تكون أبعاد خصائص.
3. هل يريد المستخدمين تضمين مناظيرهم لقاعدة البيانات كأبعاد؟
4. كم عدد مستويات البيانات المطلوبة؟
5. كيف سيتم تنظيم البيانات ضمن الأبعاد؟

2-مرحلة تصميم مستودع البيانات:

بينما لا يزال تصميم الأبعاد على الورق، يجب إعادة النظر في تصميم قاعدة البيانات وفقاً لخطوط عامة، تساعد في زيادة فاعلية التصميم متعدد الأبعاد، وفي حال كنا غير متأكدين من وجوب الإبقاء أو حذف بعد من الأبعاد، فإن هذا يعني أن كميةً أكبر من الدراسة والتحليل أصبحت مطلوبةً لوضع تصور أوضح لقاعدة المعطيات. يجب التطرق الى النقاط التالية عند مراجعة تصميم قاعدة البيانات:

كثافة وتفرق البيانات: يجب أن نحدد ما هي الأبعاد ذات البيانات الكثيفة والأبعاد ذات البيانات المبعثرة.

الأبعاد الأساسية وإبعاد الخواص: البعد الذي يصف بعداً آخر أو يستخدم في بعد آخر، هو بعد خاصة Property Dimension. وتجدر الإشارة إلى أن العضو في البعد الأساسي لا يمكن أن يملك أو يوافق أكثر من عضو خاصية واحد، وإلا يتحول بعد الخاصية إلى بعد أساسي.

يمكن إيضاح ذلك بالمثال التالي: إذا كان لدينا متجر يبيع بضائعه لزبائنه بحيث أن كل زبون تابع لأحد الفروع (أي أننا نفترض أن الزبون لن يشتري إلا من فرع محدد للمتجر)، عندها نعرّف بعد الزبائن كبعد أساسي وبعد "فرع المتجر " كبعد خاصة يمثل خاصية الموقع للزبون:

Customer (Standard dimension)
- Cust A (Attribute: New York)
- Cust B (Attribute: Illinois)
- Cust C (Attribute: California)
- Cust E (Attribute: New York)
- Cust F (Attribute: California)
- Cust M (Attribute: Illinois)
- Cust P (Attribute: Illinois)

Market (Attribute dimension)
- New York
- Illinois
- California

أما في حال تعامل الزبون مع أكثر من فرع لنفس المتجر، فإن كل عضو من أعضاء بعد الزبائن قد يوافق أكثر من عنصر من بعد "فرع المتجر"، وبالتالي يمكن أن نعرف بعد "فرع المتجر كبعد أساسي أيضاً وتكون أعضاء الأبعاد كالتالي:

Customer
- Cust A
- Cust B
- Cust C

Market
- New York
- Illinois
- California

أي يمكن القول إنه يمكن تجميع وتحليل عناصر الأبعاد الأساسية من خلال عناصر أبعاد الخواص (مثلا تحليل مكاسب مبيعات الملابس ذات اللون الأزرق مقابل مكاسب مبيعات الملابس ذات اللون الأحمر).

مراجعة التكرار في عناصر أبعاد البيانات: يجب أن تتم عملية مراجعة للتكرار في عناصر الأبعاد، حيث أن وجود تكرار في عناصر بعد، يعكس الحاجة إلى تقسيم هذا البعد. يمكن إيضاح ذلك بالمثال التالي:

البعد الخاص بالحسابات له البنية التالية:

Accounts
- Budget
  - Profit
    - Margin
      - Sales
      - COGS
    - Expenses
- Actual
  - Profit
    - Margin
      - Sales
      - COGS
    - Expenses

أي أن البعد كهرمية يقسم إلى قسمين، قسم التقديرات، وقسم القيم الفعلية، ولكل قسم حسابات خاصة بالأرباح وهوامش الأرباح والمبيعات والنفقات. يمكن ملاحظة وجود تكرار في الحسابات السابقة لكلا القسمين، وهو ما يعكس وجوب تقسيم هذا البعد، وتصبح لدينا البنية التالية:

Accounts
  Profit
    Margin
      Sales
      COGS
    Expenses
Scenario
  Budget
  Actual

أصبح لدينا بعدين، الأول لأنواع الحسابات، والثاني لطبيعة الحساب. نلاحظ اختفاء التكرار، وقد لا يكون الحل بإنشاء بعدين جديدين وإنما بدمج أعضاء هذه البيانات ضمن أبعاد أخرى.

عناصر الأبعاد المعزولة: تعرّف هذه الحالة كون مجموعة من عناصر الأبعاد لا ترتبط وليس لها علاقة بالأبعاد الأخرى في مكعب، تحل هذه المشكلة بنقل هذه العناصر إلى أبعاد أخرى في مكعب آخر في نفس قاعدة البيانات ذاتها أو في قاعدة بيانات أخرى. على سبيل المثال، في مكعب يمثل العمل المحاسبي لشركة ما ويضم:

- بعد المخططات Scenario ومكون من الأعضاء (Budget, Actual).
- بعد الزمن.Time Dimension.
- بعد أنواع الحسابات Measures Dimension (المبيعات – ...)

في حال أرادت الشركة إدخال الرواتب إلى بعد أنواع الحسابات، نلاحظ أن موضوع الرواتب لا يرتبط بشكل وثيق ببعد الزمن كما انه لا يرتبط نهائيا ببعد المخططات، ولو أردنا إضافة بعد جديد هو بعد الموظفين، فان عناصر بعد الموظفين سوف تكون عناصر معزولة أيضا لأنها لا ترتبط ببعد المخططات مثلاً.

بالتالي، يكون الحل بإنشاء مكعب جديد تضاف هذه العناصر لأبعاده، يمكن أن يكون هذا المكعب ضمن نفس قاعدة البيانات أو في قاعدة بيانات جديدة، ويمكن اقتراح إنشاء قاعدة بيانات خاصة بالموارد البشرية Human Resources (HR) وتكون أعضاء أبعاد هذا المكعب على سبيل المثال:التأمين – الحوافز – الرواتب.

بهذه الخطوات تكون مرحلة تصميم قاعدة البيانات قد انتهت، ويمكن في نهاية هذه المرحلة طرح الأسئلة التالية:

- هل قللنا عدد الأبعاد؟

ومن أجل كل بعد من الأبعاد نسأل:

- هل أضاف هذا البعد فائدة لعملية تحليل البيانات؟
- هل أضاف فائدةً لإنتاج تقارير تلبي متطلبات المستخدم؟
- هل تحققنا من نقاط تقاطع أبعاد المكعبات؟
- هل تمكنا من تجنب التكرار في عناصر أبعاد البيانات؟
- هل تحققنا من عناصر الأبعاد المعزولة؟
- هل قمنا بتقسيم قاعدة البيانات كما يجب؟

بعد مرحلة تحليل وتصميم قاعدة البيانات، يمكن وضع الصيغ النهائية لأغراض قاعدة البيانات، وتفصيل تعريف واستخدامات هذه الأغراض، كما يمكن القول أنه قد تكونت لدينا صورة واضحة عن ماهية الأبعاد وأعضاء هذه الأبعاد وعن قاعدة البيانات بشكل عام، ونصبح قادرين على تحديد الموارد المطلوبة من نظام إدارة قواعد البيانات متعددة الأبعاد لتحقيق التصميم المطلوب، وهي تشمل النقاط التالية بشكل أساسي:

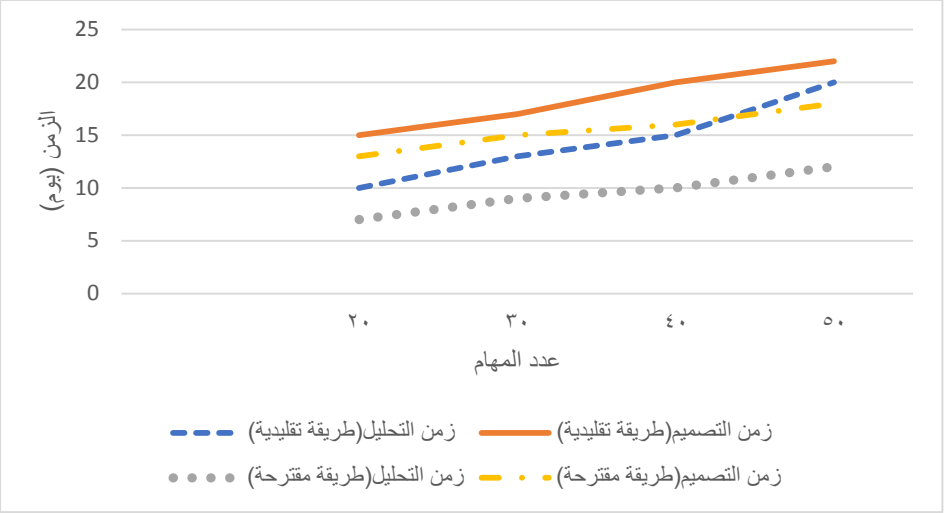
- حجم التخزين المطلوب لاستيعاب قاعدة البيانات حاليا ومستقبلا.
- حجم الذاكرة الذي يوفر الأداء المطلوب للنظام.
- هل سيكون النظام موزعا أم مركزيا؟
- تحديد نظام التشغيل.
- بالنسبة للبنية المتعددة الأبعاد، هل ستخزن بشكل علائقي (ROLAP) أم بشكل مصفوفة متعددة الأبعاد (MOLAP)؟
- تحديد برنامج إدارة قاعدة البيانات.

3-النتائج التجريبية:

تم تجربة كل من الطريقة المقترحة والطريقة التقليدية لبناء نظام مستودع بيانات، حيث تمت الدارسة على مجموعات من المهام المطلوب تنفيذها ضمن المستودع – حيث تعرّف المهمة بأنها وظيفة function تعطي قيمة إحصائية يُطلب من النظام أن يوفرها للمستخدم – مع عدد مهام مختلف لبيان أثر كل منهما على زمن تنفيذ مهام التحليل والتصميم مع ازدياد عدد المهام وبالتالي تعقّد النظام المطلوب، وتم الحصول على القيم التالية بما يخص الزمن اللازم لبناء النظام وتحقيق وظائفه:

الجدول (2) الأرقام التجريبية لأزمنة عملية التحليل والتصميم باستخدام الطريقة المقترحة والطريقة التقليدية

زمن التصميم (طريقة مقترحة) (يوم)	زمن التحليل (طريقة مقترحة) (يوم)	زمن التصميم (طريقة تقليدية) (يوم)	زمن التحليل (طريقة تقليدية) (يوم)	عدد الوظائف
13	7	15	10	20
15	9	17	13	30
16	10	20	15	40
18	12	22	20	50



الشكل (2) مقارنة الأرقام التجريبية لأزمنة عملية التحليل والتصميم باستخدام الطريقة المقترحة والطريقة التقليدية

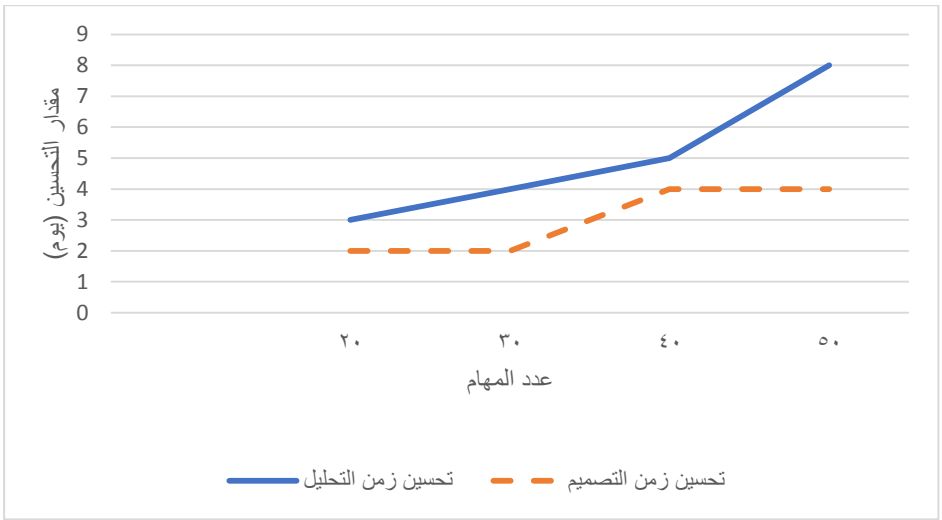
تبين الأرقام في الجدول (2) والممثلة في الشكل (2)، أن الطريقة المقترحة أدت لتقليل الزمن اللازم لإنجاز الوظائف في المستودع، سواء كان ذلك لمرحلة التحليل أو مرحلة التصميم، مقارنة مع الطريقة التقليدية.

يبين الجدول (3) مقدار التحسين في زمن التصميم الذي تم الحصول عليه باستخدام الطريقة المقترحة، وتم تمثيل القيم في الجدول بالشكل (3).

الجدول (3) مقدار التحسين في التحليل والتصميم باستخدام الطريقة التقليدية والطريقة المقترحة

عدد الوظائف	تحسين زمن التحليل(يوم)	تحسين زمن التصميم(يوم)
20	3	2
30	4	2
40	5	4
50	8	4

يبين الشكل 3 مقارنة لتحسين زمن التحليل وزمن التصميم مقارنة مع عدد المهام المطلوب إنجازها.



الشكل (3) مخطط عملية التحسين في التحليل والتصميم باستخدام الطريقة التقليدية والطريقة المقترحة

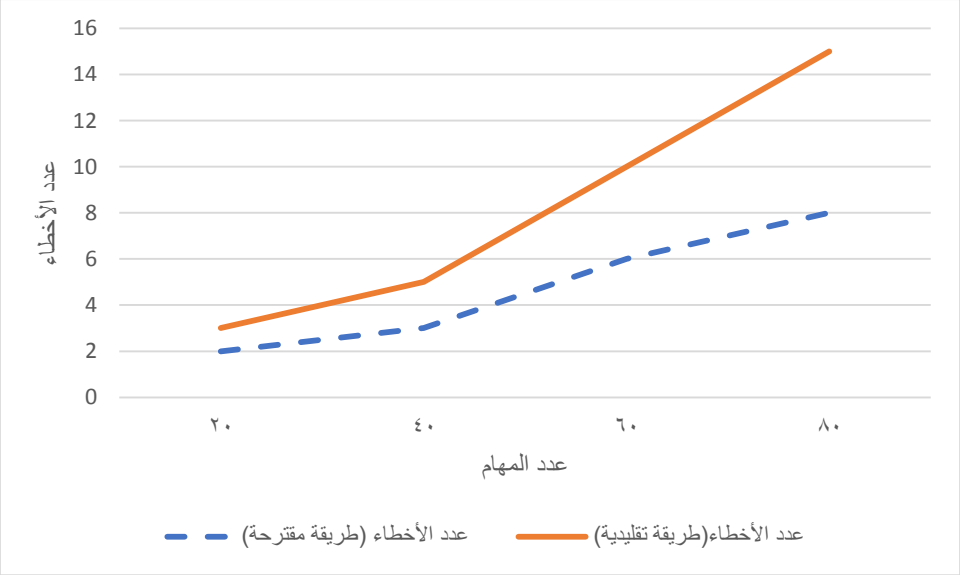
يُستنتج من الشكل (3) أن الطريقة المقترحة كانت أكثر فعالية بشكل خاص مع ازدياد عدد المهام المطلوبة وزيادة تعقيدها.

تم دراسة عدد الأخطاء التي تم تصحيحها في المهام المنجزة مقارنة مع ازدياد عدد المهام المطلوبة، حيث تم تعريف الخطأ بأنه قصور في المهام المطلوبة، والذي تطلب تغيير في النظام في مرحلة التحليل، أي أنه تطلب العودة إلى مرحلة التحليل بعد الوصول إلى مرحلة التنفيذ، وتعتبر هذه العملية مكلفة جداً في تطوير النظم.[10] وبنتيجة الدراسة تم الحصول على القيم الموجودة في الجدول (4).

الجدول (4) عدد الأخطاء في تنفيذ المهام في مستودع البيانات باستخدام الطريقة التقليدية والطريقة المقترحة

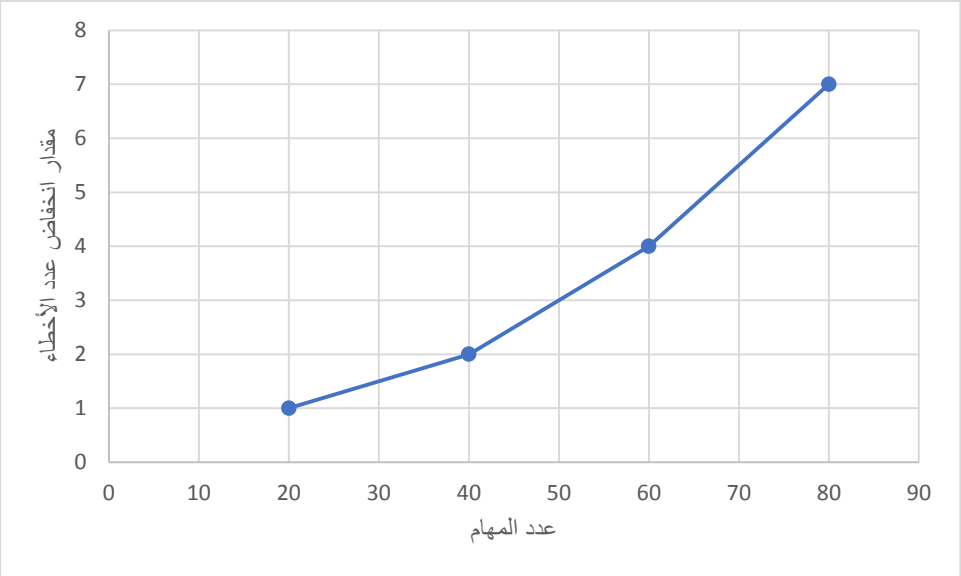
عدد المهام	عدد الأخطاء (طريقة مقترحة)	عدد الأخطاء(طريقة تقليدية)
20	2	3
40	3	5
60	6	10
80	8	15

يبين الشكل (4) مقارنة بيانية لعدد الأخطاء باستخدام كل من الطريقة التقليدية والطريقة المقترحة.



الشكل (4) عدد الأخطاء باستخدام الطريقة التقليدية والطريقة المقترحة

كما تم تمثيل مقدار الانخفاض في عدد الأخطاء (التحسين في أداء دورة حياة النظام) في الشكل (5).



الشكل (5) مقدار التحسين باستخدام الطريقة المقترحة على انخفاض عدد الأخطاء في تنفيذ المهام

يتبين من الشكل (5) أن الطريقة المقترحة أدت إلى تحسين يزداد بشكل أكبر مع ازدياد عدد المهام، وهو ما يساهم في التقليل من كلفة تطوير أنظمة مستودعات بيانات ذات عدد مهام كبيرة مطلوبة وبالتالي الحصول على أداء أفضل في دورة حياة تطوير مستودع البيانات.

الاستنتاجات والتوصيات:

الاستنتاجات

1. إن دورة حياة تطوير النظم التقليدية غير كافية لتطوير نظام مستودعات بيانات بشكل أفضل.
2. يتوجب اعتماد طريقة خاصة بتحليل وتصميم نظم مستودعات البيانات تأخذ بعين الاعتبار خصوصية النموذج متعدد الأبعاد لقاعدة البيانات.
3. الطريقة المقترحة قدمت منهجية لتطوير نظام مستودعات بيانات متعدد الأبعاد وأدت إلى انخفاض في زمن تنفيذ المهام.
4. الطريقة المقترحة أدت إلى انخفاض في عدد الأخطاء في دورة حياة تطوير نظم مستودعات البيانات.

التوصيات

1. مقارنة الطريقة المقترحة مع منهجيات مختلفة في تطوير البرمجيات، حيث اعتمد البحث على المقارنة بين منهجية الشلال والطريقة المقترحة.

References:

[1]Songhe Mu, Qing Zhu, Yue Zhang and Yeteng, Data warehouse dimensional modeling for customer service business, MATEC Web of Conferences 309, 05010 (2020) https://doi.org/10.1051/matecconf/202030905010

[2]Sijie Shen, Rong Chen, Haibo Chen, and Binyu Zang,Retrofitting High Availability Mechanism to Tame Hybrid Transaction/Analytical Processing, 15th USENIX Symposium on Operating Systems Design and Implementation.July 14–16, 2021-978-1-939133-22-9

[3]Matteo GolfarelliStefano, RizziElisa Turricchia,Modern Software Engineering Methodologies Meet Data Warehouse Design: 4WD International Conference on Data Warehousing and Knowledge Discovery DaWaK 2011: Data Warehousing and Knowledge Discovery pp 66-79

[4]Temitope Adeoye Oketunji, Raufu Olalekan Omodara, Design of Data Warehouse and Business Intelligence System A case study of a Retail Industry, Thesis submitted for completion of Master of Science (60 credits) Main field of study: Computer Science Specialization: Informatics, June 2011

[5]Hadj Mahboubi, Marouane Hachicha, Jérôme Darmont. XML Warehousing and OLAP. Encyclopedia of Data Warehousing and Mining, Second Edition, IV, IGI Publishing, pp.2109-2116, 2009. ffhal01447947v2f

[6]Benoit Dageville, Thierry Cruanes, Marcin Zukowski,The Snowflake Elastic Data Warehouse, ACM ISBN 978-1-4503-3531-7/16/06.DOI: http://dx.doi.org/10.1145/2882903.2903741(2016)

[7]M. J. Al Taleb ،Sami Hasan ،Yaqoob Yousif Mahd,On-line Analytical Processing (OLAP) Operation for Outpatient,Iraqi Journal of Science, 2021, Special Issue, pp: 225-231, DOI: 10.24996/ijs.2021.SI.1.32

[8]Jean-Paul Kasprzyk 1, and Guénaël Devillet ,A Data Cube Metamodel for Geographic Analysis Involving Heterogeneous Dimensions,ISPRS Int. J. Geo-Inf. 2021, 10, 87. https://doi.org/10.3390/ijgi10020087 https://www.mdpi.com/journal/ijgi

[9]Yuzhao Yang, Jérôme Darmont, Franck Ravat, Olivier Teste. An Automatic Schema-Instance Approach for Merging Multidimensional Data Warehouses. 25th International Database Engineering & Applications Symposium (IDEAS 2021), Jul 2021, Montreal, Canada. pp.232-241,ff10.1145/3472163.3472268ff. ffhal-03265061

[10] Mubarak Elamin, Elmubarak Daleel, Software Engineering Development and Analysis of Life Cycle Models, International Journal of Computer Applications (0975 – 8887) Volume 133– No.10, January 2016