جامعة السودان للعلوم والتكنولوجيا

كلية علوم الحاسوب وتقانة المعلومات

تخزين ومعالجة البيانات الضخمة لأنظمة تخطيط موارد المؤسسة

Storing and Processing Big Data for Enterprise Resource Planning Systems

بحث مقدم كأحد متطلبات الحصول على درجة الشرف في هندسة البرمجيات\ حاسوب و نظم المعلومات

أكتوبر 2017

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

جامعة السودان للعلوم والتكنولوجيا

كلية علوم الحاسوب وتقانة المعلومات

قسم هندسة البرمجيات \ حاسوب ونظم المعلومات

تخزين ومعالجة البيانات الضخمة لأنظمة تخطيط موارد المؤسسة

Storing and Processing Big Data for Enterprise Resource Planning Systems

أكتوبر 2017

إعداد الطالبات:

- **إيمان سعيد أحمد خليفة**
- **ساره حسب الرسول محمد احمد**
- **علا يوسف محمد اسماعيل**
- **علياء شمس الدين البشير بله**

بحث مقدم كأحد متطلبات الحصول على بكالوريوس الشرف في هندسة البرمجيات \ حاسوب و نظم المعلومات

إشراف:د. هشام عبدالله منصور

التاريخ:22 /2017/10 **توقيع المشرف:.....**

الآيه

قال تعالى:

(مَّا يَفْتَحِ اللَّهُ لِلنَّاسِ مِن رَّحْمَةٍ فَلَا مُمْسِكَ لَهَا ۖ وَمَا يُمْسِكْ فَلَا مُرْسِلَ لَهُ مِن بَعْدِهِ ۚ وَهُوَ الْعَزِيزُ الْحَكِيمُ)

صدق الله العظيم

سورة فاطر آيه (2)

الحمـــــــــــد

الحمد لله قاضي الحاجات فاطرَ الأرضِ و السماوات، و الصلاة والسلام على من بلغ الرسالة وأدى الأمانة ونصح الأمة إلى نبي الرحمة ونور العالمين سيدنا محمد ﷺ

اللهم لك الحمد و لك الشكر، كما ينبغي لجلال وجهك وعظيم سلطانك، الحمد لله الذي سخر لنا ما ينفعنا، و وفقنا لإكمال هذا العمل ونسأل الله العلي القدير أن يجعله حجةً لنا لا علينا .

اللهم صل وسلم على سيدنا محمد وعلى آل محمد ، كما صليت على إبراهيم وعلى آل إبراهيم ، وبارك على محمد وعلى آل محمد ، كما باركت على إبراهيم وعلى آل إبراهيم ، في العالمين إنك حميد مجيد.

الإهداء

إلى من حملو همنا منذ الميلاد

إلى من علمونا أن ثمرات النجاح لا تلغط من الثرى

إلى كل من رسم على وجهنا البسمات

وأزاح عن قلوبنا الحسرات

إلى من لم نكن لولاهم :

أَمهاتنا الفاضلات: فائزة، سلمى ، نعمات و الى روح الوالدة بثينة.

آبائنا الأفاضل وإلى روح الوالد يوسف محمد.

معلمينا ومعلماتنا

أصدقائنا وصديقاتنا جميعا بلا إستثناء.

زملائنا وزميلاتنا الذين قضينا معهم أربعة أعوام تشاركنا فيها الهم و الفرح، لهم منا كل التحية.

ها نحن الآن نسطر أحرفنا الأخيرة في هذه المسيرة، وها نحن الآن نضع بين أيديكم هذا البحث الذي أتى نتاج جهد متواصل، راجين أن تعم به الفائدة المرجوة.

الشكر و العرفان

الحمد والشكر أولاً وأخيراً لله سبحانه وتعالى على نعمه التي لا تعد و لاتحصى

ثم الشكر أجزله لمشرفنا الموقر د.هشام عبدالله منصور، عملاق قواعد البيانات على توجيهاته و إرشاداته لنا في كل حين، حتى تم إنجاز هذا العمل.

والشكر أيضاً لمجتمعات البرمجة في منصات العمل التقني في كل من :

Stack Overflow, gitHub

للإجابة على إستفساراتنا و أسئلتنا المتكررة.

وصديقنا المقرب محرك البحث Google .

المستخلص

يشهدُ العصرُ الحالي نهضةً وتزايداً مستمراً في كمية البيانات المنتجة يومياً من مختلف الأجهزة أو الأنظمة مثل مواقع التواصل الإجتماعي، أوالمستخدمة في مجال إنترنت الأشياء(IoT) و تلك الناتجة عن الأعمال التجارية في المؤسسات مثل أنظمة تخطيط موارد المؤسسة، مما أدى لظهور عصر البيانات الضخمة.

أنظمة تخطيط موارد المؤسسة تعد أحد أهم الأنظمة التي تعتمد عليها أغلب الشركات، المؤسسات، وغيرها قد تعمل في مؤسسات كبيرة أوقد تعمل لعشرات السنين فإن حجم البيانات فيها يصل للحجم الضخم، مما يستوجب الحاجة لإدارتها بطريقة مناسبة، قواعد البيانات العلائقية المستخدمة في هذه الأنظمة لا تمكن من إدارة الحجم الضخم من البيانات بكفاءة، هذا البحث يعتبر جزء من مجالات علم إدارة البيانات حيث يقوم على إستخدام أنظمة التخزين (NoSQL) من أجل بناء مستودع لأرشفة البيانات الضخمة الموجودة في أنظمة تخطيط موارد المؤسسة، وذلك من أجل التخزين و المعالجة المتوازية، أو التخزين والتعامل المباشر مع بيانات وحدات نظام (Odoo) .

من أهم مخرجات البحث أن أرشفة البيانات الضخمة في أنظمة التخزين (NoSQL) هو خيار إقتصادي يقلل الجهد و الزمن المبذولين في بناء المستودعات بالطرق التقليدية، و أن التخزين المباشر لبيانات وحدات نظام (Odoo) في أنظمة (NoSQL) يوفر خصائص الإتاحية والتوسع و سرعة تنفيذ العمليات على البيانات.

Abstract

The number of data produced daily by different devices or systems such as social media,the technologies used in the internet of things and the business-based, such as Enterprise Resource Planning (ERP) systems is increasing lead to the era known as big data century.

Enterprise Resource Planning (ERP) systems are one of the most important systems on which most companies, organizations, these systems may operate in large organizations or operate for decades; Relational databases used in these systems can not manage the large volume of data efficiently. This research is one of the fields of data management science, which is based on the use of storage systems (NoSQL) to build a data warehouse for archiving big data in ERP systems, for storage and parallel processing, or direct handling with Open source data (Odoo).

One of the most important outputs of the research is that the archiving of big data in storage systems (NoSQL) is an economical option that reduces the effort and time spent in constructing warehouses by traditional methods. The direct storage of the Odoo system data in NoSQL systems provides availability, scalablity and speed of data operations.

فهرس المصطلحات

المصطلح	إختصار المصطلح	معنى المصطلح
Enterprise Resource Planning	ERP	تخطيط موارد المؤسسة
Hadoop	_	هي تقنية من تقنيات تخزين البيانات الكبيرة
Map Reduce	_	هو نموذج برمجي لمعالجة البيانات الكبيرة بالطريقة المتوازية
Relational Database Management System	RDBMs	أنظمة إدارة قواعد البيانات العلائقية
Object Relational Mapper	ORM	مخطط الكائنات العلائقي
Not Only SQL	NoSQL	نظام قواعد البيانات الغير علائقي
(Create,Read,Update,Delete)	CRUD	العمليات الأساسية في قواعد البيانات
Structure Query Language	SQL	لغة الإستعلامات المستخدمة في قواعد البيانات العلائقية
Cassandra Query Language	CQL	هي لغة الإستعلامات الخاصة بقاعدة Cassandraالبيانات غير العلائقية
Hive Query Language	HQL	هي لغة الإستعلامات الخاصة بHive
Hadoop Distributed File System	HDFS	نظام الملفات الموزع في نظام Hadoop
Application Programming Interface	API	واجهة برمجة التطبيقات
Internet of Thing	IoT	إنترنت الأشياء

فهرس الجداول

رقم الباب . رقم الشكل . رقم الجدول	موضوع الجدول	رقم الصفحة
(1.3. 2)	مقارنه بين قواعد البيانات العلائقيه وغير العلائقيه	20
(1.4.2)	الفرق بين الدراسات السابقة	29
(1.3)	نماذج بيانات Cassandra و ال RDBMS	32
(1.5)	توزيع البيانات على العقد	48
(2.5)	مواصفات العقد	52

فهرس الأشكال

رقم الباب . رقم الشكل	موضوع الشكل	رقم الصفحة
(1.1.2)	العلاقة بين البيانات و المعلومات	7
(1.3.2)	نظرية ال CAP	19
(2.3.2)	معمارية نظام الملفات الموزع (HDFS)	21
(3.3 .2)	طريقة نموذج البرمجة بإستخدام Map Reduce	22
(1.4.2)	المعمارية للبحث التجريبي لمقارنة اداء أنظمة تخزين البيانات NoSQL و RDBMS في نظام ال ERP (Odoo)	25
(2.4.2)	نتيجة المدخلات علي نظام توزيع المياة الأوروبي باستخدام Hadoop	26
(3.4.2)	إيجاد اكبر قيمة بين Hadoop وMySQL في نظام توزيع المياة الأوروبي	27
(4.4.2)	البنية التحتية لمنصة الأعمال الجاهزة	28
(1.3)	معمارية Cassandra	32
(2.3)	عملية الكتابة في Cassandra	33
(3.3)	عملية القراءة في Cassandra	33
(4.3)	طريقة عمل Sqoop	35
(1.1.4)	مكونات Hadoop التي تم إختيارها لتحقيق غرض الأرشفة	41
(2.1.4)	معمارية التعامل المباشر بين Odoo و Cassandra	42
(1.2.4)	معمارية النظام	44

فهرس الشاشات

رقم الباب . رقم الشاشة	وظيفة الشاشة	رقم الصفحة
(1.5)	أمر إستيراد جميع الجداول من postgreSQL الى Hadoop	46
(2.5)	أمر إستيراد جداول معينة من postgreSQL الى Hadoop	46
(3.5)	إستيراد البيانات المدخلة حديثاً	47
(4.5)	توزيع البيانات بإستخدام MapReduce	47
(5.5)	نموذج من بيانات جدول الhr_department	47
(6.5)	توزيع بيانات جدول داخل HDFS	48
(7.5)	ملف (hdfs-site.xml)	48
(8.5)	طريقة تعريف جدول في Hive لإستقبال البيانات القادمة من HDFS	49
(9.5)	طريقة التعامل مع البيانات داخل Hive	49
(10.5)	ملف وحدة Odoo المطورة	50
(11 .5)	عملية الإتصال بالعنقود وإنشاء قاعدة بيانات فيه	51
(12.5)	عملية الإدخال في قاعدة Cassandra	51
(13.5)	توضح عملية الإسترجاع من قاعدة Cassandra	51
(14.5)	عملية التعديل في بيانات قاعدة Cassandra	51

(15.5)	عملية الحذف من بيانات قاعدة Cassandra	51
(16.5)	تشغيل خادم قاعدة بيانات Cassandra	53
(17.5)	تشغيل خادم ال Odoo	53
(18.5)	تشغيل خادم Hadoop	53
(19.5)	نموذج من دالة توليد البيانات في جدول hr_department	54
(20.5)	لوحة تحكم عنقود Hadoop	54

فهرس المحتويات

الآيه ..... ب

الحمـــــــــــد ..... ج

الإهداء ..... د

الشكر و العرفان ..... ه

المستخلص ..... و

فهرس المصطلحات ..... ح

فهرس الجداول ..... ط

فهرس الأشكال ..... ي

فهرس الشاشات ..... ك

الباب الأول (الإطار العام للبحث)

2.1 المقدمة : 1

2.1مشكلة البحث: 1

3.1 أهمية البحث: 2

4.1 أهداف البحث:- 2

5.1 حدود البحث: 2

6.1 منهجية البحث: 2

7.1 فروض البحث: 3

8.1 هيكلية البحث 3

الباب الثاني (الخلفية النظرية للبحث) .....

المبحث الأول(إدارة البيانات والبيانات الضخمة) .....

1.2 المقدمة: 4

1.1.2 البيانات (Data): 4

1.1.1.2 أنواع البيانات: 4

2.1.2 المعلومات (Information): 4

3.1.2 إدارة البيانات (Data Management): 5

1.3.1.2 الإتجاهات الجديدة في التعامل و إدارة البيانات: 5

4.1.2 إدارة المعلومات (Information Management): 5

5.1.2 قاعدة البيانات (Database): 5

6.1.2 قواعد البيانات الموزعة (Distributed Databases): 5

7.1.2 مستودعات البيانات (Data Warehouse): 5

8.1.2 أدوات نقل، إستخراج، تحميل البيانات (ETL tools): 6

9.1.2 معالجة البيانات(Data Processing): 6
1.9.1.2 المعالجة المتوازية (Parallel Processing): 6
10.1.2 البيانات الضخمة (Big Data): 6
1.10.1.2 خصائص البيانات الضخمة: 6
11.1.2 الأنظمة مفتوحة المصدر (Open source software): 7
1.11.1.5 مزايا الأنظمه مفتوحة المصدر: 7
2.11.1.5معايير البرمجيات مفتوحة المصدر: 7
12.1.5 رخصة أباتشي (Apache license): 8
المبحث الثاني(نظام تخطيط موارد المؤسسة) .....
2.2 المقدمة: 13
1.2.2 نظام تخطيط موارد المؤسسة 9
2.2.2 أنواع أنظمة تخطيط موارد المؤسسة بناءً على مكان تخزين البيانات وكيفية إدارتها: 9
3.2.2 المكونات الرئيسيه في أنظمة تخطيط موارد المؤسسة: 9
1.4.2.2مميزات نظام (Odoo) 10
5.2.2مخطط الكائنات العلائقي (ORM) 15
1.5.2.2فوائد مخطط الكائنات العلائقي (ORM): 15
2.6.2.2عيوب مخطط الكائنات العلائقي (ORM): 11
7.2.2 أداء النظام (System performance) 11
1.7.2.2 أداء نظام Odoo(Odoo performance) 16
المبحث الثالث(أنظمة إدارة قواعد البيانات غير العلائقية (NoSQL)) .....
3.2 مقدمة : 13
1.3.2 قواعد البيانات غير العلائقية (NoSQL DB) : 13
2.3.2 أقسام قواعد البيانات غير العلائقية (NoSQL): 13
4.3.2 بيئة نظام Hadoop (Hadoop Eco system): 14
1.4.3.2 مكونات نظام Hadoop 20
المبحث الرابع(الدراسات السابقة) .....
4.2مقدمة: 15
1.4.2 البحث التجريبي لمقارنة أداء أنظمة تخزين البيانات NoSQL و RDBMS في نظام ال ERP (Odoo) [1] 15
2.4.2 تطبيق NoSQL باستخدام Mongo DB في تطوير نظام ERP لإدارة الموارد البشريه.[18] 16
3.4.2 تطبيق نظام إدارة الموارد البشرية لإستخدام mongodb 16
4.4.2 تطبيق Hadoop لتخزين ومعالجة البيانات الضخمة المجمعة من نظام توزيع المياة الأوربي 17
4.4.2 منصة الأعمال الجاهزه 18
الباب الثالث (التقنيات و الأدوات المستخدمة).....
1.3 المقدمة:.... 31

2.3إطار عمل Odoo: 31
3.3 python.....19
4.3 PostgreSQL: 19
5.3 قاعدة بيانات Cassandra: 31
6.3 python driver-Cassandra: 20
7.3 Hadoop : 20
8.3 Zookeeper: 20
9.3 Oozie:..... 20
10.3 Hive:..... 21
11.3 Sqoop:. 21
12.3 HTML و CSS و Java Script و Bootstrap 22
13.3 Django framework:..... 22

الباب الأول

الإطار العام للبحث

2.1 المقدمة :

تحتاج البيانات الضخمة إلى طرق إدارة أكثر كفاءة من الطرق التقليدية التي تُعامَل بها غيرها من البيانات، حيث أنها تُكَوِن مجاميع بيانات متزايدة الحجم و التعقيد مما يجعل تحليلها و تبادلها و أرشفتها أحد أهم التحديات في هذا العصر، ولذلك ظهرت عدد من التقنيات وبيئات العمل التي تتيح التعامل مع البيانات الضخمة وتوفر طرق للتخزين والمعالجة تمكن من الإستفادة منها في إتخاذ القرارات بناءً على بيانات ضخمة متراكمة.

أنظمة إدارة قواعد البيانات العلائقية (RDBMs) كانت ومازالت تستخدم كوسيط التخزين الأول للأنظمة المطورة، ولكنها لاتتمتع بكل المميزات التي تمكنها من إدارة الحجم الضخم من البيانات، حيث أن طريقة التخزين التي تقدمها أغلب هذه النظم تقوم على مفهوم التخزين المركزي مما يزيد العبء على النظام ويؤدي الى زيادة زمن التأخير عند المعالجة، إضافة إلى الحاجة لوجود مخطط (scheme) معرف مسبقاً، و غيرها من الخصائص التي تعيق التعامل مع الأنظمة ذات البيانات الضخمة، وعليه نجد أن نظام تخطيط موارد المؤسسة في تطور مستمر، و مع إنتشار إستخدامه نجد أنه نظام مستقر قد يعمل لعشرات السنين أو في مؤسسات كبيرة مما يعني زيادة حجم البيانات فية بصورة مستمرة، مما يجعل من المناسب جداً أن يعمل هذا النظام بالتوازي مع التقنيات المخصصة للتعامل مع الأنظمة ذات البيانات الضخمه، وهذا البحث يهدف إلى أرشفة و تخزين بيانات نظام تخطيط موارد المؤسسة مفتوح المصدر (Odoo) في قواعد البيانات غير العلائقيه، والتي ستمكن من التخزين الموزع و المعالجة المتوازية لبيانات النظام الضخمة، و تقليل التكلفة و الزمن المبذولين لبناء مستودعات بيانات خاصة أو إجراء عمليات أرشفه متكررة نتيجة زيادة حجم بيانات النظام، و تحقيق ترابط نظام Odoo مع أحد أنظمة (NoSQL) من أجل التفاعل المباشر مع البيانات، مما يوفر خاصيتي الإتاحية و سرعة تنفيذ العمليات على البيانات، وتلك هي أهم مميزات أنظمة التخزين غير العلائقي (NoSQL) التي تعتمد على التخزين الموزع لنسخ البيانات و المعالجة المتوازية لها.

2.1 مشكلة البحث:

أنظمة تخطيط موارد المؤسسة كغيرها من الأنظمة التي تعتمد على قواعد البيانات العلائقية تواجه عدد من المشاكل والصعوبات التي تحول دون تطورها والقدرة على إدارة بياناتها المتضخمة، ولذلك فإن أهم مشاكل البحث تتمثل في النقاط الآتية :

1. بناء مستودعات بيانات (Data warehouse) تحتاج الى وقت وجهد وتكلفه، إضافه الى أنها لاتمكن من أخذ نظرة شاملة على البيانات الجديدة و المؤرشفه قديماً فيها ككل [1] .
2. صعوبة معالجة و إدارة الحجم الضخم من البيانات بالطرق التقليدية، وبالتالي تأخر تنفيذ العمليات المعقدة.

3.1 أهمية البحث:

أنظمة تخطيط موارد المؤسسة التقليديه تؤدي عملاً ممتازاً من حيث أداء أعمالها اليومية، ولكنها أقل جودة في إدارة ومعالجة البيانات الضخمة وذلك بسبب إعتمادها على قواعد البيانات العلائقية.

وهنا تكمن أهمية البحث في أنه يسعى لتحقيق الترابط بين وسيط تخزين يتناسب مع الحجم الضخم للبيانات من أجل عمليات الأرشفة أو التعامل المباشر معها من قبل نظام (Odoo)، من أجل توفير إتاحية و أداء عالي في الوصول لبيانات النظام، والتي يمكن إستخدامها في عمليات التحليل و المساعدة في إتخاذ القرارات مستقبلاً.

4.1 أهداف البحث:-

هذا البحث يحقق الأهداف التالية :

1- بناء مستودع لأرشفة بيانات أنظمة تخطيط موارد المؤسسة في وسائط تخزين مخصصة للتعامل مع البيانات الضخمة.
2- معالجة البيانات بإستخدام نماذج تتوافق مع البيانات الضخمة.
3- تمكن نظام Odoo من التواصل مباشرة مع وسيط التخزين NoSQL للإستفادة من مميزات الإتاحية والتوسع و الأداء العالي للنظام.

5.1 حدود البحث:

هذا البحث يعتمد على إستخدام نظام (Odoo) كدراسة حالة، حيث يتم إستيراد البيانات من قاعدة (PostgreSQL) و إجراء عمليات التخزين لأغراض الأرشفة في نظام الملفات الموزع (HDFS) الذي يوفره Hadoop ومعالجتها عن طريق نموذج البرمجة (MapReduce) بإستخدام (Hive)، كما يتيح التخزين والتعامل المباشر عن طريق وحدة (Odoo) جديدة تم تطويرها للتواصل مع قاعدة التخزين غير العلائقي (Cassandra) .

6.1 منهجية البحث:

إعتمد البحث على عمل إستطلاعات ومقابلات مع عدد من مطوري (Odoo) والشركات العاملة في المجال مثل مركز النيل للأبحاث، و البحث عن دراسات مشابهة ودراسات تجريبية لتحليل المشاكل التي إعتمد عليها البحث، ثم تصميم معمارية من مكونات مفتوحة المصدر، ومن ثم بناء وتطبيق المعمارية و أخيراً تقييم معمارية البحث المقترحة.

7.1 فروض البحث:

1- بناء مستودع بيانات يمكن إستخدامة في المستقبل لإتخاذ القرارات بأقل تكلفة.

2- المقدرة على تخزين بيانات كبيرة بإتاحية و أداء عالي مهما بلغ حجمها.

8.1 هيكلية البحث

يتكون البحث من ستة أبواب بالإضافه إلى الاجزاء التمهيديه والختاميه على النحو الآتي:

الباب الأول يتناول هذا الباب الإطار العام للبحث ويتمثل في : مقدمة البحث ومشاكله وأهدافه وحدوده، أما الباب الثاني يتكون من اربعة مباحث ، المبحث الأول يتناول البيانات و إدارة البيانات لأن هذا البحث هو جزء من علم إدارة البيانات و البيانات الضخمة، ويتناول الأنظمة مفتوحة المصدر، المبحث الثاني ويتناول أنظمة تخطيط موارد المؤسسة، المبحث الثالث يتناول بيئة عمل Hadoop و قواعد البيانات اللاعلائقية، المبحث الرابع يتناول الدراسات السابقة، و الباب الثالث يتناول التقنيات والأدوات المستخدمه، أما الباب الرابع فيتكون هذا الباب من مبحثين ، المبحث الأول تحليل النظام، المبحث الثاني تصميم النظام، الباب الخامس يتناول تطبيق و إختبارات النظام، وأخيراً الباب السادس يتناول النتائج والتوصيات و الخاتمة.

الباب الثاني

الخلفية النظرية

مدخل :

هذا الباب سيتناول الخلفية النظرية و المفاهيم العامة المتعلقة بالبحث، وكذلك سيعرض الدراسات السابقة و ذات العلاقة و التي تمثل احد اهم دعائم البحث مع عرض أوجه الاختلاف والتشابه بينها و بين ما سيقدمه البحث من عمل ، وقد قَسم هذا الباب إلى أربعة مباحث .

حيث يتناول **المبحث الأول** منه **:** يتناول إدارة البيانات والبيانات الضخمة إضافة لعدد من المفاهيم العامة المتعلقة بالمعالجة و التخزين ومستودعات البيانات**..**

و يتناول **المبحث الثاني** منه : التعرف بشكل عام على نظام تخطيط موارد المؤسسة مفتوح المصدر (Odoo).

وأيضاً يتناول **المبحث الثالث** : التعرف على قواعد البيانات غير العلائقية أو ما يسمى أنظمة (NoSQL)، و مقارنتها مع أنظمة إدارة قواعد البيانات العلائقية، و التعريف بماهية نظام (Hadoop) وبيئة عمله.

وأخيراً يتناول **المبحث الرابع** : الدراسات السابقة وعلاقتها بهذا البحث و مدى تقدمها ، و مبررات قيامها، و أوجه الاختلاف و التشابه بينها وبين ما يتناوله البحث .

المبحث الأول

إدارة البيانات و البيانات الضخمة

(Data Management & Big Data)

1.2 المقدمة:

تهتم أغلب المؤسسات والشركات بإدارة مواردها مثل (الموارد البشرية، المالية، المعدات، المواد الخام وغيرها) ولكنها لاتحسن الإستفادة من أهم الموارد وهي البيانات والإستثمار في الإستفادة منها بعد تحولها الى معلومات، والتي ستساعد الإدارة في إتخاذ القرارات الهامة إعتماداً على معلومات متراكمة ناتجة عن التعاملات اليومية داخل المؤسسة، وكلما إزداد حجم البيانات زادت كفاءة إتخاذ القرارات.

1.1.2 البيانات (Data):

وهي الشكل الخام لأي محتوى يتم إنتاجه، ويمكن معالجته وتفسيره وتبادله.[2]

1.1.1.2 أنواع البيانات:

1- **البيانات المهيكلة** (Structured Data) : وهي البيانات التي يمكن تنظيمها في جداول ذات أنواع بيانات(Data Types) معرفّة، مثل البيانات التي يتم تخزينها في قواعد البيانات العلائقية.

2- **البيانات شبة المهيكلة** (Semi-structured Data): مثل ملفات صيغ ال Xml, json

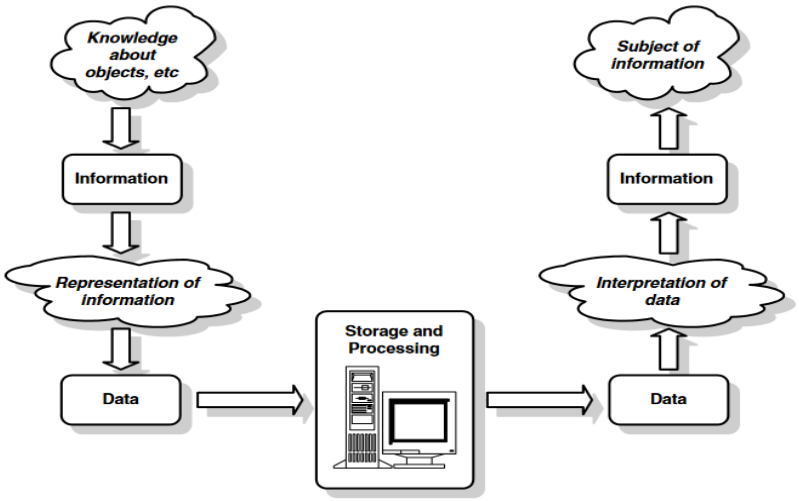
3- **البيانات غير المهيكلة**(Un-Structured Data) : مثل بيانات الصوت و الفيديو.[3]

2.1.1.2 مصادر البيانات (Data sources)

هي أماكن تولد أو تواجد البيانات إما أن تكون قواعد بيانات تشغيلية، أو بيانات تاريخية مؤرشفة في وسيط تخزين، مثل الصور و الوسائط المتعددة، أو بيانات مولدة من أجهزة إستشعار أو مراقبة.[3]

2.1.2 المعلومات (Information):

وهي المخرج من البيانات الخام بعد المعالجة، مثل الحقائق و الأحداث، والتي تأخذ معنى معين في صياغ معين.



شكل (1.1.2) العلاقة بين البيانات و المعلومات[2]

3.1.2 إدارة البيانات (Data Management):

وهي عملية تجمع عدد كبير من النشاطات المتعلقة بالتعامل مع البيانات مثل جودة البيانات، الوصول للبيانات، ملكية البيانات، توزيع البيانات، معالجة البيانات، تخزين البيانات، حيث توفر مبادئ وطرق التعامل معها وفق استراتيجيات معينة.[2]

1.3.1.2 الإتجاهات الجديدة في التعامل و إدارة البيانات:

1- إستخدام الحزم البرمجية.
2- توزيع البيانات و قواعد البيانات.
3- مستودعات البيانات.
4- تنقيب البيانات.
5- أدوات نقل البيانات.
6- قواعد بيانات الوسائط المتعددة.
7- تقنيات البيانات الضخمة.
8- تكنولوجيا الويب.[2]

4.1.2 إدارة المعلومات (Information Management):

وهي عملية إدارة المعلومات كواحدة من موارد المؤسسة تتضمن تخطيط، تنظيم، توجية و التحكم بالمعلومات.[2]

5.1.2 قاعدة البيانات (Database):

هي أحد طرق تخزين السجلات داخل أنظمة الحواسيب، وهي عبارة عن مجموعة من ملفات البيانات الموجودة تحت تحكم نظام إدارة قواعد البيانات(DBMS).[3]

6.1.2 قواعد البيانات الموزعة (Distributed Databases):

هي مجموعات متعددة من قواعد البيانات المترابطة منطقياً مع بعضها، والموزعة على شبكة الحواسيب من أجل القيام بوظيفة معينة أو العمل معاً كنظام معين، دون شعور المستخدم بتوزيعها[3].

7.1.2 مستودعات البيانات (Data Warehouse):

هي نوع خاص من قواعد البيانات تحتوي على عدد كبير من البيانات التاريخية التي شُكِلت من عدد من المصادر أوقواعد البيانات الموجودة، وذلك لدعم إتخاذ القرارات، ويتميز هذا النوع من قواعد البيانات بتطابق بنيته الداخلية مع ما يحتاجه المستخدم من مؤشرات ومحاور التحليل وهي مستودعات خاصة لإجراء عمليات تحليل عميق على البيانات.[3]

1.7.1.2 إختلاف مستودعات البيانات عن قواعد البيانات التشغيلية

تختلف مستودعات البيانات عن قواعد البيانات التشغيلية للأنظمة في طريقة تصميم و غرض كل منهما، من حيث التصميم فقد صممت قواعد البيانات لتخزين البيانات التشغيلية، بينما صممت مستودعات البيانات بغرض تخزين و تحليل و جمع البيانات من عدة مصادر ثم إعادة تنظيمها بطريقة تتيح التعامل معها والمساعدة في إتخاذ القرارات وتخزين بيانات ضخمة على المدى الطويل.

أما في الغرض الذي صممتا من أجله، فقواعد البيانات التشغيلية غرضها الإستخدام اليومي و التعامل مع العمليات المدخلة، أما مستودعات البيانات ليست للإستعلام اليومي و إنما للإستعلام والتحليل على المدى الطويل لإستخلاص تقارير معقدة تساعد في إتخاذ القرارات، ولذلك فهي مخصصة للقراءة منها فقط.[3]

8.1.2 أدوات نقل، إستخراج، تحميل البيانات (ETL tools):

هي مكون أساسي من مكونات مستودعات البيانات، وقد صممت هذه الأدوات لتسهيل عملية ادارة البيانات وتقليل الجهد المستهلك في إستيراد البيانات ونقلها من مكان لآخر، لذلك فإن عملية إختيار الأداة التي تقوم بهذه المهمة هي عملية مهمة ومعقدة لجلب البيانات من عدة مصادر أو قواعد بيانات الى مستودع البيانات[4].

9.1.2 معالجة البيانات(Data Processing):

هي عملية إنتاج معلومات ذات معنى عن طريق جمع عناصر البيانات مع بعضها، ومن ثم إجراء عمليات عليها كوحدة واحدة، من أجل إستخلاص المعلومات التي نحتاجها منها.[2]

1.9.1.2 المعالجة المتوازية (Parallel Processing):

هي أحد أنواع معالجة البيانات، حيث أن أكثر من تعليمة يمكن معالجتها وتنفيذها في نفس الوقت، وذلك بتقسيم المشكلة إلى أجزاء أصغر ومعالجتها آنياً عن طريق المعالجات المتعددة، مما يوفر سرعة في المعالجة على حجم كبير من البيانات و التعليمات.[2]

10.1.2 البيانات الضخمة (Big Data):

هو أحد الحقول والمجالات التي نشأت مع التطور في مجالات التكنولوجيا والتعامل معها ، ومانتج عنها من تولد حجم ضخم من البيانات ، ولا يقتصر فقط هذا المجال على حجم البيانات وإنما يحيط أيضاً بعدة أوجه منها تنوع أشكال البيانات المتولدة و سرعة تولدها من المصدر، مما يقتضي عندها إختلاف طرق المعالجة والتخزين .[3]

1.10.1.2 خصائص البيانات الضخمة:

حتى توصف بيانات معينة بأنها بيانات ضخمة، يجب أن تتوفر فيها ثلاثة خصائص أساسية:

1- **الحجم** (Volume) : وهو مايتم توليده من البيانات، وهنا يجب أن تفوق التيرابايت.

2- **التنوع**(Varity) : وهو تنوع أنواع البيانات المتولدة بين مهيكلة وشبة مهيكلة وغير المهيكلة.

3- **السرعة**(Velocity) : وهي سرعة تولد البيانات من المصدر.[3]

2.10.1.2 مصادر البيانات الضخمة:

1- الحساسات و أجهزة الإرصاد المستخدمة في مجال إنترنت الأشياء.

2-قواعد بيانات أنظمة تخطيط موارد المؤسسة.

3-وسائط التواصل الإجتماعي.[3]

11.1.2 الأنظمة مفتوحة المصدر (Open source software):

هي برامج طورت واتيحت حرية التعديل للمستخدم بناءً على متطلباته و احتياجاته الخاصة وفق اتفاقيات رخصة المشروع، مما يتيح للمطور توزيع نسخته المعدلة، وإمكانية التعديل عليها من قبل مجتمعات المطورين الأخرى، مما يعني تواجد مجتمع كبير من مطوري البرامج مفتوحة المصدر .[3]

1.11.1.5 مزايا الأنظمه مفتوحة المصدر:

1- **تكلفة اقل (Low cost)**:

حيث لا يحتاج المستخدم لشراء رخصة البرنامج او دفع رسوم على استخدام النظام.

2- **سرعة التطوير و الإبتكار (Innovative)**:

مما يسمح بتطوير افكار جديدة مبتكرة دون بذل الكثير من الجهد والوقت.

3- **جودة أعلى (High quality)**:

حيث يُمّكن من تفادي الأخطاء التي سبق إكتشافها .[13]

2.11.1.5 معايير البرمجيات مفتوحة المصدر:

يجب أن تتوفر في البرمجيات مفتوحة المصدر عدد من المعايير وهي :

1- الوصول الى الشيفره المصدرية.

2- إذن التعديل .

3- المساهمة العلمية.

4- إعادة نشر وتوزيع التعديلات.[13]

12.1.5 رخصة أباتشي (Apache license):

مؤسسة البرمجيات (Apache) تقدم الدعم لمجتمع البرمجيات مفتوحة المصدر، وتقدم هذه المؤسسة الرخص للمشاريع مفتوحة المصدر، فهي تضمن إنتاج مشاريع عالية الجودة، وبذلك كانت الرائدة في مجال المشاريع والرخص مفتوحة المصدر، حيث يقصد بـ "الرخصة" شروط وبنود الإستخدام وإعادة الإنتاج والتوزيع، فهذه الرخصة تحمي المطورين من خسارة الحق، وضياع الشفرة المصدرية وتحفظ له حقوق الملكية.[13]

المبحث الثاني

نظام تخطيط موارد المؤسسة

(Odoo)

2.2 المقدمة:

هذا المبحث يقوم بالتعريف بنظام تخطيط موارد المؤسسة (Odoo) وأداء النظام والتحسينات التي أُجريت عليه.

1.2.2 نظام تخطيط موارد المؤسسة

هو نظام متكامل يضم مجموعة من الأنظمة التي تعمل مع بعضها لإتمام العمليات الخاصة بالشركة أو المؤسسة من أجل تخطيط الموارد وربطها بشكل موحد ومتكامل مما يغنيها عن إستعمال أنظمة مختلفة من عدة شركات حيث أنة يوفر واجهة موحدة لجميع التطبيقات والأنشطة، ويستخدم قاعدة بيانات موحدة لمتابعة كافة أنشطة المؤسسة، وتصنف أنظمة تخطيط موارد المؤسسة على إعتبارات هي:

1- معمارية النظام (ثنائي الطبقات، ثلاثي الطبقات، متعدد الطبقات(2-tier , 3-tier ,..,N-tier)
2- السوق المستهدف.
3- مكان تخزين البيانات وكيفية إدارتها. [5]

2.2.2 أنواع أنظمة تخطيط موارد المؤسسة بناءً على مكان تخزين البيانات وكيفية إدارتها:

1- **On-Premise ERP**:

هذا النوع يعني أن النظام قد تم تنصيبه محلياً في البنية التحتية للمؤسسة، وهو مناسب مع المؤسسات ذات الحجم الكبيرحيث أنه ذو تكلفة مرتفعة ويتطلب الكثير من الوقت لتشغيله.

2- **Cloud ERP**:

هذا النوع يعتمد على إدارة النظام و البيانات من قبل مستضيف في ال (cloud) ، وهو مناسب مع المؤسسات الصغيرة و المتوسطة.

3- **Hybrid ERP**:

وهو يجمع بين النوعين الأول والثاني، حيث أن الهدف الأساسي منه هو إبقاء بعض الوحدات الـ (Modules) متاحة الوصول عبر ال Cloud، و الأخرى تبقى كـ on-premises .[1]

3.2.2 المكونات الرئيسيه في أنظمة تخطيط موارد المؤسسة:

- **خادم قاعدة البيانات**

والذي يحوي كل قواعد البيانات التي تحتوي على جميع بيانات ومكونات وإعدادات النظام .

- **خادم التطبيق**

وهو يحوي كل العمليات المنطقية للمؤسسة ويتأكد من أن النظام يعمل بشكل مثالي.

- **خادم الويب**

وهو الذي يقوم بربط المستخدم بالنظام عن طريق متصفح الويب. [5]

4.2.2 نظام تخطيط موارد المؤسسة مفتوح المصدر(Odoo)

هو أحد أنظمة تخطيط موارد المؤسسة مفتوحة المصدر، وقد طور ليكون بديلاً عن أنظمة (SAP و Oracle ERP)فهو ليس بمفهوم (Framework) و لابمفهوم (IDE) فهو نظام تخطيط موارد مؤسسة متكامل، حيث يتميز بإحتوائه على عدد كبير من النظم الفرعية المتكاملة والتي تمتد من 400 نظام إلى 4000 نظام فرعي تم تطويره عن طريق الوكلاء في حوالي 120 دولة حول العالم[6]

1.4.2.2 مميزات نظام (Odoo)

1- لا يوجد مقابل مادي لرخصة البرنامج ولا للتجديد السنوي لأنه متكامل مفتوح المصدر تم إصداره طبقاَ للوائح الرخصة الدوليه للبرامج مفتوحة المصدر، مما يوفر التكلفة المنخفضة والخروج عن نطاق الاحتكار نتيجة إنعدام تكلفة التراخيص وتجديد التراخيص حيث تنحصر تكلفته فقط في التركيب والتطوير.

2- برنامج واحد يمكن الاعتماد عليه مع إمكانية تنزيل أي نظام فرعي عند الحاجة اليه بضغطة زر واحدة .

3- يتميز البرنامج بتصميم رائع ومتميز وسهل للمستخدم النهائي مع مراعاه أحدث الاساليب الحديثة للتعامل من خلال البوابات الإلكترونية وكذلك الأجهزة اللوحية المحمولة.

4- نسبة لأنه مفتوح المصدر للجميع يسهل على المبرمجين قراءة ومتابعة الشيفرات ممايزيد من جودته والذي بدوره ينعكس على سرعة تطويره.

5- المرونه فهو كغيره من البرمجيات مفتوحة المصدر يسهل تشكيله وتخصيصه وفقاً لطبيعة العمل الخاصة بالمؤسسة.

6- الاحتفاظ بمعلومات متكاملة عن جميع المعاملات اليومية .

7- الربط الكامل ما بين الإدارات الفرعية من خلال نظام واحد تم بناؤه بأحدث الأساليب التقنية كبرنامج ويب (Odoo ERP) .

8- توفير إمكانية التعامل مع النظام من خلال الإنترنت حيث أنه نظام (Web based Application)

9- توفير لوحات التحكم و مؤشرات الأداء للإدارة العليا لكي تمكنهم من المعرفة اللحظية لحالة المديونيات و التحصيل و غيره من مؤشرات الأداء .[6]

5.2.2 مخطط الكائنات العلائقي (ORM)

هو طبقة وسيطة في كثير من الأنظمة تتيح تواصلة مع قاعدة بياناته، وفي نظام Odoo تسمح هذه الطبقة للنظام بالتواصل مع قاعدة البيانات العلائقية PostgreSQL لتنفيذ العمليات التي تعرف بالـ (CRUD)، على وجه العموم فإن وظيفة الـ (ORM) هو إعطاء النظام مرونة ليستطيع التواصل مع وسائط التخزين المختلفة، بالرغم من ذلك فإن الـ (ORM) الحالي في نظام (Odoo) يدعم فقط التواصل مع نظام بيانات الـ (PostgreSQL) . [1]

1.5.2.2 فوائد مخطط الكائنات العلائقي (ORM):

1 عدم الحاجة للتعامل مع إستعلامات (SQL) لحفظ وإسترجاع البيانات.
2 بساطة التكوين.
3 واجهة برمجة تطبيقات(API) قياسية وموحدة لجميع الاعمال.
4 سرعة تطوير التطبيقات.
5 الدعم المتزامن.
6 الدعم الممتاز لأداء التطبيقات.
7 تحسين أداء البرامج عن طريق جلب البيانات إلى الذاكرة.
8 سهولة التعلم والإستخدام.[7]

2.6.2.2 عيوب مخطط الكائنات العلائقي (ORM):

1 ابطأ قليلا مقارنة بالـ (JDBC).
2 بطء سرعة تنفيذ العمليات في حالة التحديثات الكبيرة[7]

7.2.2 أداء النظام (System performance)

أداء النظام هو حجم العمل الذي يمكن إنجازه في النظام خلال فترة زمنية معينة، والتأثيرات التي تحدث للنظام جراء عمله، وعموماً فإن أداء أي نظام يتكون من **جانبين هما**:

الجانب الأول :- سرعة عمل النظام أثناء أداء مهمه معينة، مثل إستخراج تقرير أو إجراء عملية تحليل، أو هو عدد العمليات التي يقوم النظام بإنجازها خلال فترة زمنية معينة.

الجانب الثاني:- هو جانب التأثيرات التي تحدث على النظام جراء عمله، وذلك يشمل (إتاحية النظام (System Availability) ، التوسع(Scalability)، زمن الإستجابه(Response Time)، سرعة المعالجة (Speed of Processing)، سعة التخزين(Storage Capacity)، الطاقة الإنتاجية (Throughput)، إستهلاك الطاقه، التأثيرات البيئيه).[8]

1.7.2.2 أداء نظام Odoo(Odoo performance)

تقوم شركة (Odoo) بتحسينات مستمرة في أداء النظام في الإصدارات الحديثة نسبة لطلبات المستخدمين المستمرة حول أداء النظام، وقد كانت أحد حلول Odoo في الإصدار التاسع هو إستخدام (API) جديدة أجريت عليها تحسينات من حيث كمية المعاملات المرسلة وغيرها، وعليه تم الإستغناء عن الـ (API) القديمة، و بشكل عام يتم تحسن الأداء بتوفير معالجات متعدده (multi-processor) واجهزة بمواصفات عالية، خادم التطبيق المركزي هو أحد مشاكل النظام من حيث عدم القدرة على التوسع، إضافة إلى أن تخزين المرفقات (Attachment) من أهم الأسباب التي ساهمت في تأخر أداء النظام، وقد تم إجراء تجربة حل هذه المشكلة في الإصدار الثامن من (Odoo) بتخزين هذه المرفقات في نظام الملفات (File system) وعدم تخزينها في قاعدة البيانات، الى أن هذا الحل يعتبر مقيد من حيث تحديد حجم معين للمرفق لايمكن تجاوزه وإلا فلن تتم هذه العملية . [9]

المبحث الثالث

أنظمة إدارة قواعد البيانات غير العلائقية (NoSQL)

و

بيئة نظام Hadoop

3.2 مقدمة :

في هذا المبحث سيتم التعرف على الجيل الجديد من أنظمة إدارة قواعد البيانات، والتي طُورت لتتوافق مع متطلبات الجيل الجديد، وهي أنظمة عرفت بـ (NoSQL)، و سيتم أيضاً التعرف على أحد أهم بيئات العمل التي تم إستخدامها في هذا البحث وهي بيئة عمل نظام (Hadoop).

1.3.2 قواعد البيانات غير العلائقية (NoSQL DB) :

واجهت عديد من الشركات الكبرى مشكلة التطبيقات التي تكتظ بكمية ومدى واسع من البيانات والمستخدمين مثل شركات خدمات الرسائل كـ(Yahoo، Gmail)، و شركات تطبيقات وسائط التواصل الاجتماعي كـ(Facebook و tweeter)، والتي عادة ما يفوق عدد مستخدميها المليوني مستخدم، عدا أنّ كل مستخدم لدية عدد كبير من الرسائل و البيانات، لذلك فإنّ هذه الشركات وجدت نفسها في حوجه ماسة لنظام تخزين يسمح بإدارة والتعامل مع هذه الكمية الضخمة من البيانات و المستخدمين، و حيث أنّ خيار استخدام نظم إدارة قواعد البيانات العلائقية لا يعتبر حلاً مناسباً نسبة لأنّها تعتبر مقيدة لمثل هذه الأنواع من الأنظمة، مما يجعل عملية بناء مثل هذا النوع منها في غاية الصعوبة، و حتى نُظم الإدارة التي تعرف بالـ (Object – relational model) ليست مناسبة فهي أيضاً تحتاج إلى مخطط (Schema) معرفة مسبقاً مما يزيد من تعقيد بناء النظام، لذلك قامت عدد من الشركات التي تواجه مثل هذه المشكلة بتطوير أنظمة قواعد بيانات تتناسب مع احتياجاتها وأنواع بياناتها المتولدة، وطورت عدد من الحلول والتي أصبحت تعرف بما يسمى (NoSQL Systems) و يشير مسمى هذا النوع من الأنظمة ل(Not Only SQL) أي أنّه ليس من الضروري أن تكون طريقة التعامل مع البيانات عن طريق استعلامات (SQL) فقط، و إنما يتوفر لأي نوع منها عدد من الاستعلامات الخاصة بها بالإضافة لدعمها أيضاً استعلامات (SQL) ، وهي أيضاً انظمه قواعد بيانات تتيح حرية اكثر في تصميم قواعد البيانات حيث لا يشترط أن تحتوي قاعدة البيانات على مخطط (schema) معرفة مسبقاً، مما جعلها تتناسب مع احتياجات التطبيقات التي لا يمكنها الخضوع لقيود قواعد البيانات العلائقية، وقد طورت هذه النظم من قواعد البيانات لتدعم المعالجة المتوازية (Parallel processing) للبيانات ذات الحجم الضخم. [3]

2.3.2 أقسام قواعد البيانات غير العلائقية (NoSQL):

تصنف هذه الأنظمة إلى أربعة أقسام حسب أنماط نماذج البيانات المستخدم و هي:-

1. **Key-value NoSQL System**:

هذه الأنظمة مبنيه على نماذج بيانات مبسطة ، تسهل الوصول إلى البيانات المطلوبة، حيث يستخدم المفتاح (Key) للوصول لقيمة البيانات (Value)، ومن أمثلتها(Dynamo DB وCassandra)

2. **Columnar NoSQL System**:

هذه الأنظمة تقوم بتقسيم جداول قاعدة البيانات إلى أعمدة يمكن ضمها فيما يسمى بعائلة الأعمدة

(Column family)، وهي تتيح أيضاً الوصول إلى إصدارات البيانات السابقة مثل (HBase).

3. **Document NoSQL System**:

هذه الأنظمة تقوم بتخزين البيانات في شكل مستندي (Document) باستخدام صيغ بيانات معروفه مثل (JSON)، ويتم الوصول إلى المستند عن طريق (id) أو باستخدام (index) آخر، ومن أمثلتها (Mongo DB).

4. **Graph NoSQL System**:

يتم فيها تمثيل البيانات في شكل (graph)، والذي يتكون من عقد وحواف (edges, nodes) ومن أمثلتها (Neo4j)[3].

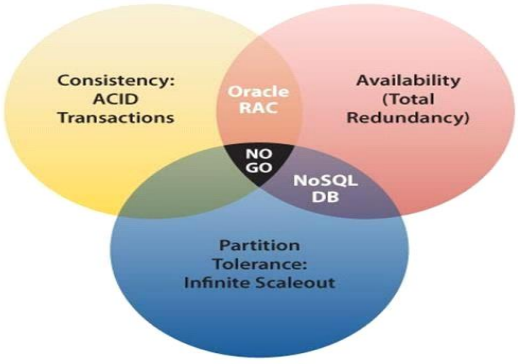
5.3.2 نظرية الـ CAP (CAP Theorem)

هي نظرية خاصة بالأنظمة الموزعة تقوم على أنه في أي قاعدة بيانات موزعة يمكن أن تحقق خاصيتين على الأكثر من أصل ثلاث، وهذه الخصائص الثلاث هي :

1- الإتساق (**Consistency**) : وتعني توافق كل النسخ الموزعة على ال Nodes والناتجة عن عملية ال Replication ، - وعلية فإن انظمه SQL التقليدية تضمن الاتساق عن طريق خاصية ACID وهي تعني الخصائص التالية **A**tomicity, **C**onsistency, **I**solation, **D**urability ، ولذلك فهي أنظمه عالية الاتساق، أما أنظمة NoSQL فهي ضعيفة الإتساق وهي تعتمد على خاصية BASE وهي الخصائص التالية (**B**asically **A**vailable **S**oft-state **E**ventual consistency)

2- الإتاحية (**Availability**) : أي طلب لعملية قراءة أو كتابة للبيانات إما أن تتم بنجاح وإما إن تعطي رسالة توضح أن العملية لا يمكن أن تتم .

3- إمكانية التقسيم (**Partition tolerance**): وتعني أن النظام يمكنه مواصلة العمل حتى عند حدوث عطل في شبكة التواصل بين ال Nodes .[1][2]



شكل رقم (1.3.2) CAP Theorem[1]

جدول رقم (1.3.2) مقارنه بين قواعد البيانات العلائقيه وغير العلائقيه.[11]

قواعد البيانات غير العلائقيه NoSQL	قواعد البيانات العلائقيه RDBMS
لا تعتمد فقط على إستعلاماتSQL	تعتمد على إستعلاماتSQL
لا تتطلب وجود مخطط(Scheme) معرف مسبقاً	تتطلب وجود Scheme ثابتة،معرفة مسبقاً
نماذج البيانات فيها تعتمد على خصائص (BASE)	نماذج البيانات فيها تعتمد على خصائص (ACID)
تعمل بكفاءة أكبر في وجود البيانات الكبيرة	يقل الأداء فيها مع زيادة حجم البيانات
تعتمد على التوسع الأفقي (Horizontal)	تعتمد على التوسع الرأسي (vertical)

4.3.2 بيئة نظام Hadoop (Hadoop Eco system):

هي مجموعة من المشاريع التي تقع تحت مظلة البنية التحتية للحوسبة الموزعة، وهو أحد بيئات العمل التي نشأت بالتوازي مع مفاهيم البيانات الكبيره (Big data) وأحد أهم التكنولوجيا المستخدمة في هذا الحقل، نظام (Hadoop) هو نظام مفتوح المصدر مرخص تحت رخصة أباتشي ، حيث قدم هذا النظام عدد من المفاهيم والمكونات التي أثبتت كفاءتها في معالجة و تخزين البيانات الكبيرة عن طريق التخزين الموزع (Distributed systems) و المعالجة المتوازية للبيانات(parallel processing) ، وتتكامل معه عدد من المشاريع و الأنظمة لإكمال مهام عديدة.

و قد ظهرت هذه البيئه في الوقت الذي فشلت فية التقنيات القديمة التي لم تعُد تستطيع التعامل مع البيانات الكبيرة المتولدة في هذا العصر و أصبح Hadoop مُستخدم من قبل عدد من الشركات الكبرى كـــ أُوراكل و مايكروسوفت و قوقل وفيسبوك وتويتر و يوتيوب... الخ.

حتى أن الخدمات السحابية التي تقدمها الشركات كأمازون أصبحت تقدم التعامل مع نظام (Hadoop) كخدمة، وقد نشأت وطورت عدد من المشاريع مفتوحة المصدر، وضُمِنت تحت بيئة النظام مفتوح المصدر.[3]

1.4.3.2 مكونات نظام Hadoop

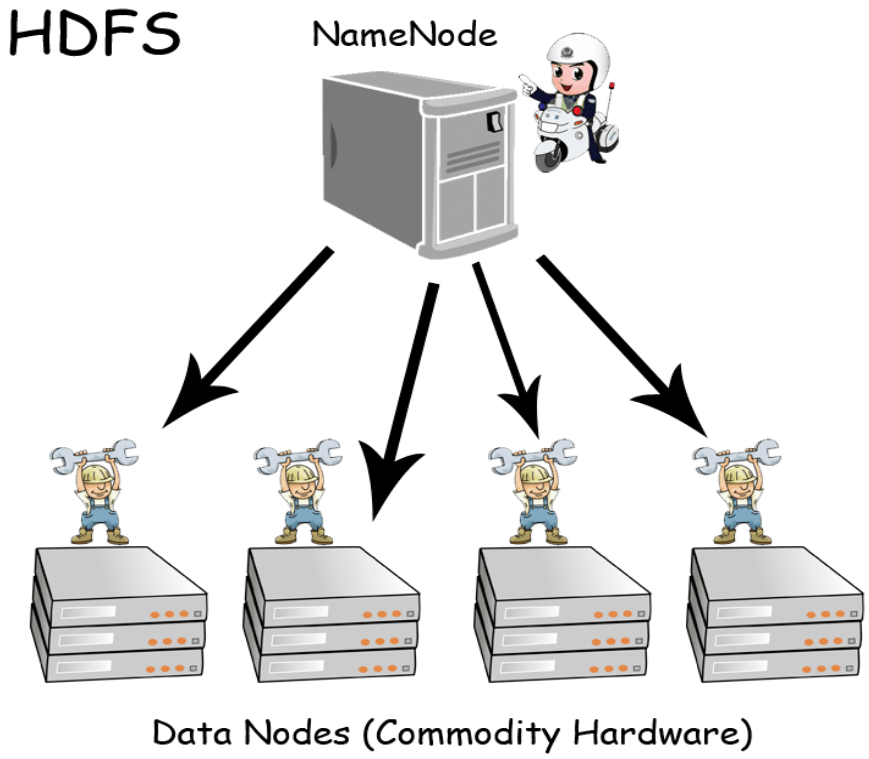
يعتمد نظام Hadoop على مكونين أساسيين هما :

1- نظام الملفات الموزع **HDFS (Hadoop Distributed File System)**:

وهو المكان الرئيسي لتخزين وإدارة البيانات في نظام Hadoop صممت لتوافق أنظمة الملفات الموزعة(Distributed file system)، ويتميز بقابليتة للتوسع و توفير إتاحية و أداء عالي للبيانات حيث يتم تقسيم البيانات على أكثر من عقدة (node) .

- **معمارية نظام الملفات الموزع HDFS:**

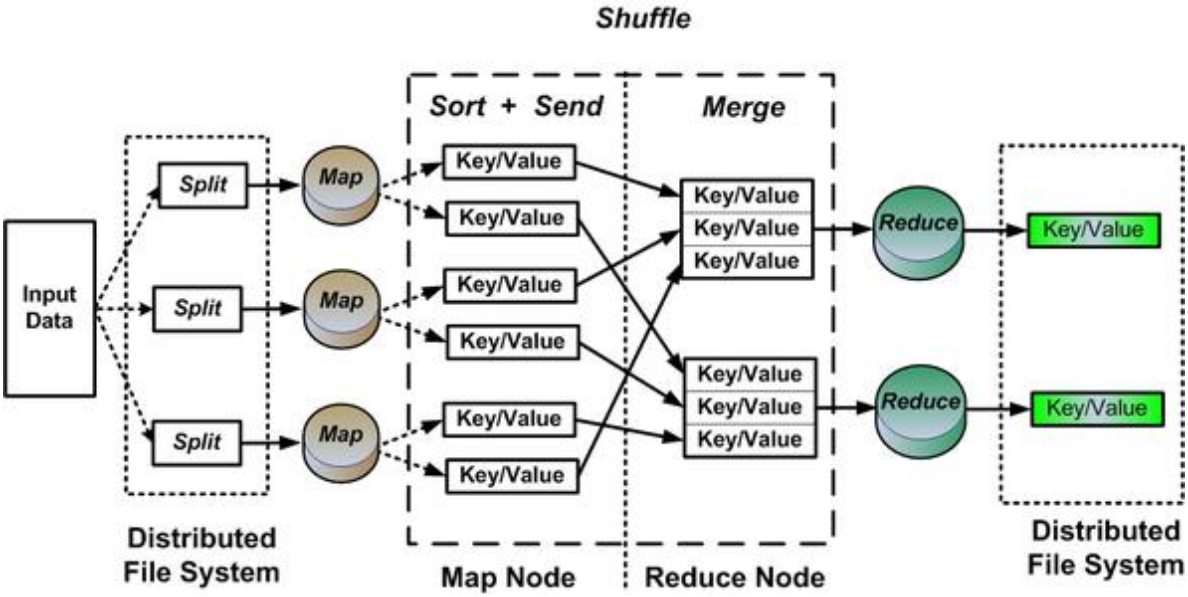
تتكون من عدد من العقد،أحدها هي العقدة الرئيسية التي تسمى القائد(Master) وتحوي وصف البيانات وتوزيعها داخل العقد الخوادم، ولذلك فهي تسمى بال(NameNode)، أما العقد التابعة للعقدة القائد فتحوي البيانات الحقيقية الموزعة عليها، لذلك تسمى بال(Data Nodes)، ويعتمد هذا النظام على معمارية (Master/Salve)[12] .



شكل رقم (2.3.2) معمارية نظام الملفات الموزع (HDFS)

2- نموذج البرمجة Map Reduce

وهو نموذج لمعالجة البيانات الكبيرة الموجودة في ال (HDFS) بطريقة المعالجة المتوازية (parallel processing) وذلك بكتابة وظائف (Map and Reduce) والتي يمكن كتابتها بلغات برمجة مختلفة مثل جافا، بايثون، روبي ،...الخ، ويعتمد على مفهوم تقسيم البيانات الكبيرة إلى أجزاء أصغر و من ثم معالجتها حيث أنه يعتمد على المبدأ " المشكلة الكبيرة يمكن حلها بتقسيمها إلى مشاكل اصغر" و يتم إستقبال البيانات في شكل قيمة و مفتاح (key ,value).[3]



ويوضح الشكل(3.3.2) أعلاة طريقة نموذج البرمجة بإستخدام Map Reduce ، وتقسيم البيانات في مراحل Map , Shuffle , Reduce ومن ثم تخزين نتيجة المعالجة في نظام الملفات الموزع.

المبحث الرابع

الدراسات السابقة

4.2 مقدمة:

في هذا المبحث سيتم عرض الدراسات السابقة وذات العلاقه بالبحث ومميزات وعيوب ومبررات قيام كل دراسة إن وجدت.

1.4.2 البحث التجريبي لمقارنة أداء أنظمة تخزين البيانات NoSQL و RDBMS في نظام ال ERP (Odoo) [1]

An Experimental Performance Comparison of NoSQL and RDBMS Data Storage Systems in the ERP System Odoo.

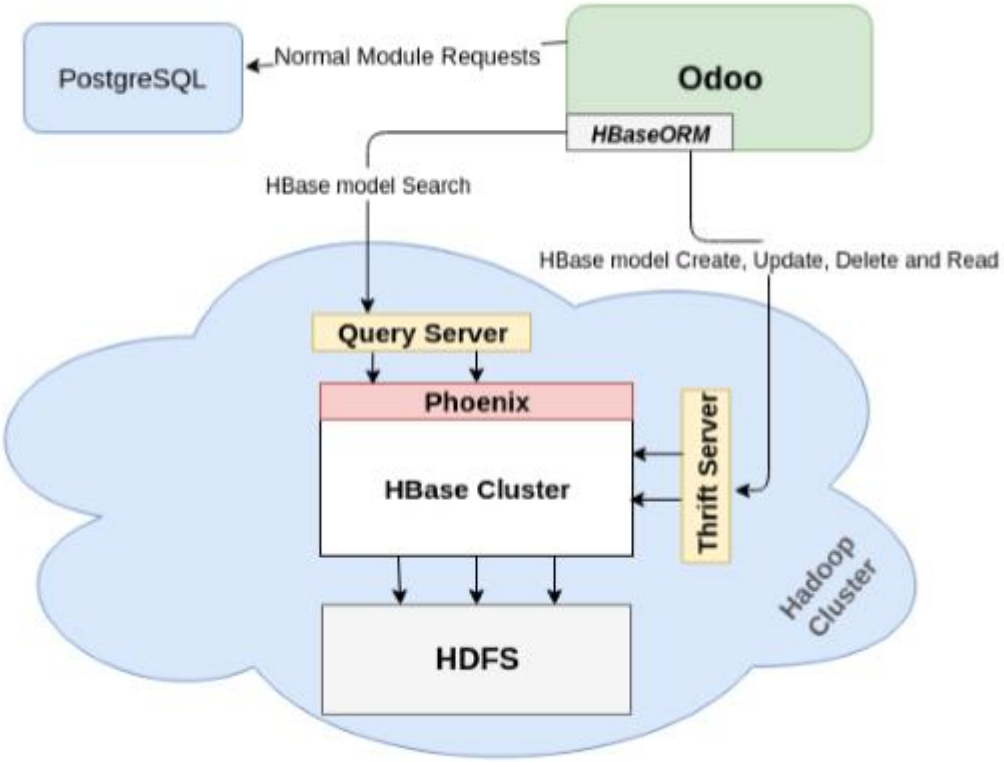
أُجري هذا البحث التجريبي إعتماداً على تقرير شركة InitOS الألمانية العاملة في مجال ال Odoo ، كنتيجة لتتبعها مشكلة تناقص الأداء التي ظهرت في أحد أنظمه عملائها مما إستوجب على الشركة البحث و تعقب سبب المشكلة، و الخروج بالتقرير الذي أوضح : أن تناقص الأداء كان بسبب تضخم حجم البيانات في كل من وحدتي الـ :

Attachment module و **Mail message Module** والتي تتعامل معها باقي الوحدات في النظام ، فهي أسرع الوحدات تضخماً بالبيانات، و بالتالي فإن أداء النظام ككل يصبح ضعيفاً.

يعتمد هذا البحث على مقارنة أداء نظام Odoo بإستخدام ال NoSQL مع الـ RDBMS، وذلك بنقل تخزين وحدتي ال Attachment & mail Message في نظام تخزين آخر من وسائل التخزين التي تعرف بـ NoSQL ، وقد تم إختيار بيئات عمل من Hadoop كبيئة تخزين لهذه الوحدتين، أما بقية الوحدات الأخرى فتخزن في قاعدة بيانات PostgreSQL المعتادة.

خرج البحث بأنه عندما كان حجم البيانات في وحدتي ال Attachment & Mail Message يوصف بالضخم، أي أكبر من 1 TB فإن أداء النظام بالمعمارية الجديدة أظهر تحسناً واضحاً، بينما حين كان حجم البيانات اقل من ذلك فإن أداء النظام بالمعمارية الحالية يكون أفضل، وعلية فإن البحث أوضح أن إستخدام Hadoop كنظام تخزين ثانوي ل Odoo سيكون خياراً موفقاً.

و الشكل التالي يوضح المعماريه التي اعتمد عليها البحث:



الشكل رقم (1.4.2) يوضح المعمارية للبحث التجريبي لمقارنة اداء أنظمة تخزين البيانات NoSQL و RDBMS في نظام ال ERP (Odoo)

2.4.2 تطبيق NoSQL باستخدام Mongo DB في تطوير نظام ERP لإدارة الموارد البشريه.[18]

The Application of NoSQL Mongo DB in Developing EPR System for Managing Human Resource

قامت هذه الدراسة بتطبيق نظام تخطيط موارد المؤسسة وتحديداً نظام إدارة الموظفين وإستخدام نوع من (NoSQL DB) هو (Mongo DB) كوسيط تخزين، ويعمل كمستودع لبيانات النظام من أجل الإستفادة منها بإجراء عمليات التحليل والمساعدة في إتخاذ قرارات مناسبة و متابعة الموظفين مما يسمح بتحسين الأداء و تقليل التكلفة في المؤسسة .

وقد كان من أهم المميزات التي دعت الى استخدام (mongo DB)هي خاصية التوسع الأفقي التي توفرها قواعد البيانات غير العلائقية، و إمكانية توافقها مع مصادر بيانات أُخرى بسهولة.

مما يسمح بإستخدام التقنيات التي ظهرت في هذا العصر مثل تقنيات إنترنت الأشياء(IoT)، حيث سيتيح إمكانية متابعة و تواصل الموظفين مع النظام .

وتهدف الدراسة إلى توضيح أحد الطرق التي يمكن فيها إستخدام قواعد البيانات غير العلائقية مع نظام تخطيط موارد المؤسسة، وهي هنا إستخدامها كمستودع للبيانات، حيث أن قاعدة Mongo DB تقوم بتخزين الملفات الكبيرة بمعاونة نظام يسمى ب GridFS، مما يسمح بإدارة الملفات و تخزينها في قاعدة البيانات بشكل أكثر كفاءة.

و خرجت الدراسة بأن إستخدام قواعد البيانات غير العلائقية كمستودعات بيانات سيمثل خياراً جيداً لإمكانية تحليل البيانات و توسع حجمها فيما بعد داخلها، وخاصة مع الأنظمة التي تستخدم في العمل اليومي كأنظمة تخطيط موارد المؤسسة.

3.4.2تطبيق Hadoop في تخزين ومعالجة البيانات الكبيره المجمعه من نظام توزيع المياة الأوروبي.[19]

Application of HADOOP to Store and Process Big Data Gathered from an Urban Water Distribution System

نظم توزيع المياه (WDS) أصبحت تعتمد على نظم المعلومات و تحتاج في الآونة الأخيرة الى تخزين كمية كبيرة من البيانات في قواعد البيانات المناسبة وحل مشكلة جمع ورصد بيانات المياه في الوقت الحقيقي، ويتمثل التحدي المعاصر في إستخراج هذه البيانات للحصول على نتائج موثوقة تدعم الأنشطة التنفيذية في إدارة المياه بالبلديات ، هذه الدراسة إعتمدت في تخزين البيانات علي(NoSQL database) مع تقديم مقارنة بين MySQL و Hadoop. وقد تم استخدام Map Reduce بإعتبارها الاكثر شيوعا لضمان المعاجة المتوازية.

إن الهدف الأساسي من هذه الدراسة هو مقارنة وتقييم النهج التقليدي لنظام إدارة قواعد البيانات العلائقية ولأغراض المقارنة أُجريت عدة تجارب بإستخدام (Apache HADOOP) و (MySql) حيث تم تثبيت خوادم قواعد البيانات "كما هي" دون تعديلات أو تحسينات في الأداء في كلتا الحالتين ثم إدخال البيانات المختلفة على كل منها وقد كان شكل مدخلات النظام في كل من MySQL و Hadoop كلاتي:

Table 1. The results of computational experimens.

	MAX		AVG		Group MAX		Group AVG	
No of data	MySQL	HADOOP	MySQL	HADOOP	MySQL	HADOOP	MySQL	HADOOP
800 000	-	5	-	4	-	6	-	4
1 600 000	-	6	-	5	1	7	1	5
3 200 000	-	9	1	7	2	9	2	10
6 400 000	3	12	2	12	4	15	4	14
12 800 000	6	20	5	22	8	25	9	26
25 600 000	10	38	10	37	16	55	16	47

الشكل رقم (2.4.2) يوضح المدخلات علي نظام توزيع المياة الأوروبي

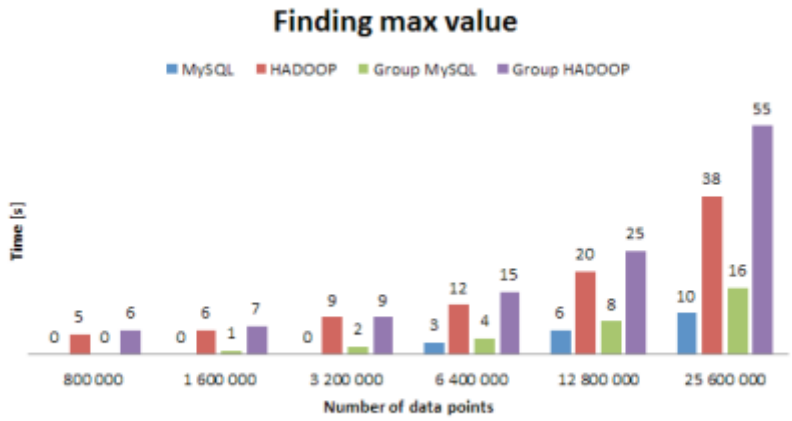


Fig. 2. The results for finding the maximal value

الشكل رقم(3.4.2) يوضح إيجاد أكبر قيمة بين Hadoop وMySQL في نظام توزيع المياة الأوروبي

وخرجت الدراسة بالنتائج التالية:

- أن MySQL فاقت Hadoop في زمن التنفيذ في وجود عدد قليل من نقاط البيانات
- أن Hadoop يمتلك القدرة على العمل أفضل في البيئة الموزعة.
- أن Hadoop له القدرة على النسخ التلقائي للبيانات والامان والموثوقية مما جعلها الحل الامثل لمعالجة وتخزين البيانات.

4.4.2 منصة الأعمال الجاهزه[21]

Enterprise-Ready platform

هي منصة قامت ببناءها شركتا (Intel) و (SAP) حيث تعمل على كل من هذه الشركتين كميات من ال petabytes تجمع بين البيانات المهيكله وغير المهيكله.

الهدف الأساسي من هذه الدراسة هو إستخدام نهج يبسط البنية التحتية ويزيل التعقيدات المادية والتشغيلية لنقل كميات كبيرة من البيانات من النظم التشغيلية إلى النظم التحليلية، والحاجه إلى حل فعال من حيث التكلفة يُمَكِن من توسيع نطاقها لإلتقاط هذه البيانات وتخزينها وتحليلها ويجب أن تكون قادرة على دمج البيانات الكبيرة في بيئة التحليلات في الوقت الحقيقي.

الحل الذي تم الاعتماد عليه هو دمج Hadoop مع SAP HANA _قاعدة البيانات التي تم بناءها ليعتمد عليها SAP ERP_ ، والنتيجة هي منصة تحليلات في الوقت الحقيقي مصممة لكفاءة استيعاب وتخزين ودمج وتحليل جميع بيانات المؤسسة .

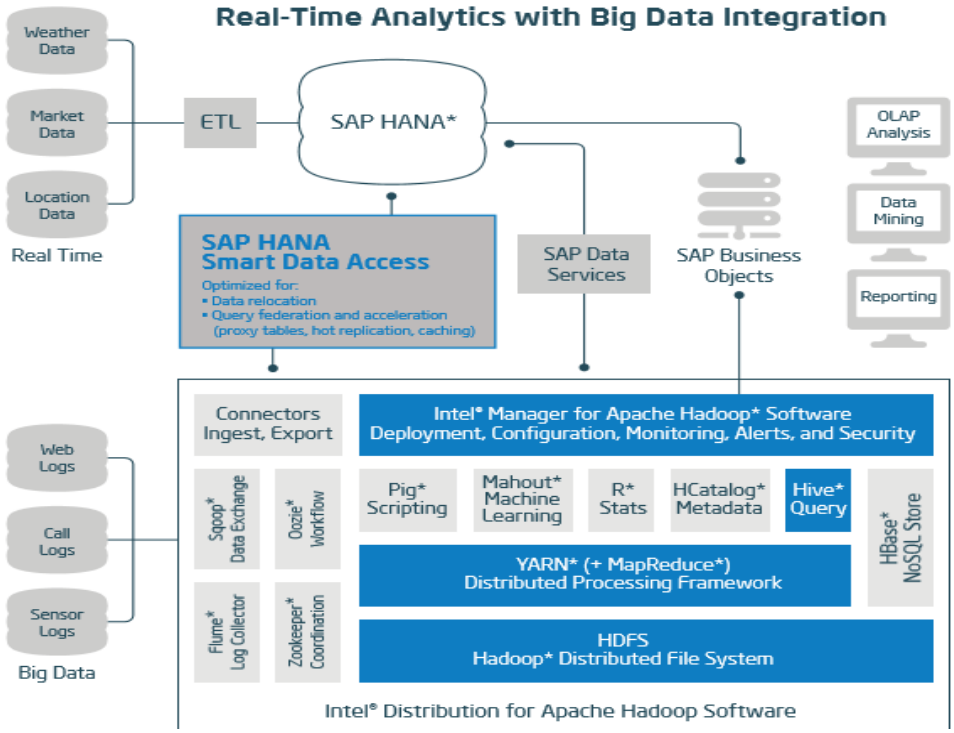
و تقدم المنصة الخصائص التاليه:

1- التحليلات في الوقت الحقيقي

مع تخزين فعال من حيث التكلفة التي يمكن أن تمتد إلى Exabyte من البيانات.

2- تكامل البيانات و الاستعلام

لذلك يمكن أن تكون هنالك تحليلات متقدمة تطبق على جميع البيانات باستخدام أدوات SAP ونماذج البرمجة القائمة على SQL.



شكل رقم (5.4.2) البنية التحتية لمنصة الأعمال الجاهزة

جدول (1.4.2) أوجه التشابه و الإختلاف بين البحث والدراسات السابقة

رقم الدراسة	أوجه التشابه	أوجه الإختلاف
1	– إستخدام نظام Odoo. – إستخدام قواعد البيانات غير العلائقية و Hadoop	- هذه الدراسة هي دراسة تجريبية لمقارنة الأداء، و تم الإعتماد على مخرجات هذا البحث التجريبي ونتائجه في البدء لحل مشكلة البحث، بنقل جميع وحدات النظام بعد وصولها الى قاعدة البيانات. - نقل وحدتين فقط من وحدات نظام Odoo.
2	– إستخدام قواعد البيانات غير العلائقية في نظام ال ERP، كمستودع للبيانات.	– النظام المطور هو نظام إدارة الموظفين. – إختيار قاعدة البيانات غير العلائقية MongoDB.
3	– الإعتماد في تخزين البيانات على NoSQL DB – إستخدام نموذج المعالجة المتوازية MapReduce	– هدف الدراسة هو المقارنة بين MySQL و Hadoop
4	- نقل البيانات من النظم التشغيلية إلى النظم التحليلية، من أجل الإستفادة منها لاحقاً	- دمج Hadoop مع قاعدة بيانات SAP HANA

5.4.2 خلاصة الباب:

في هذا الباب تم عرض المفاهيم والنظريات العامة المتعلقه بالبحث، وعرض الدراسات السابقة وذات العلاقة بالبحث و التي تم أجراءها كبحوث تجريبية أو كأنظمة كحلول تجارية، بهدف حل مشكلة معينة أما توسع النظام أو أداء النظام ككل، وتم عرض أوجه التشابه و الإختلاف بينها وبين هذا البحث

الباب الثالث

التقنيات و الأدوات المستخدمة

1.3 المقدمة:

في هذا الباب سيتم تناول التقنيات و بيئات العمل المستخدمة لتنفيذ الحل الذي يقدمه البحث.

2.3 إطار عمل Odoo:

أو كما يطلق عليه سابقاً نظام إدارة موارد المؤسسه الموزع مفتوح المصدر(Open ERP)، وهو نظام متكامل لإدارة وتخطيط موارد المؤسسه يتوفر كنسخة مجانية جاهزة الإستخدام، ويتكون من عدد من الوحدات والتي تؤدي وظائف معينة و تتكامل فيما بينها لإدارة المؤسسة ككل، وعليه فإن تطوير وحدات مخصصه بمتطلبات الشركة أو المؤسسة أمر غاية في السهولة مما يقلل من المخاطر المحتمله من التطوير وقد روعي في تصميمه أفضل الخبرات في المجال.[15]

3.3 :Python

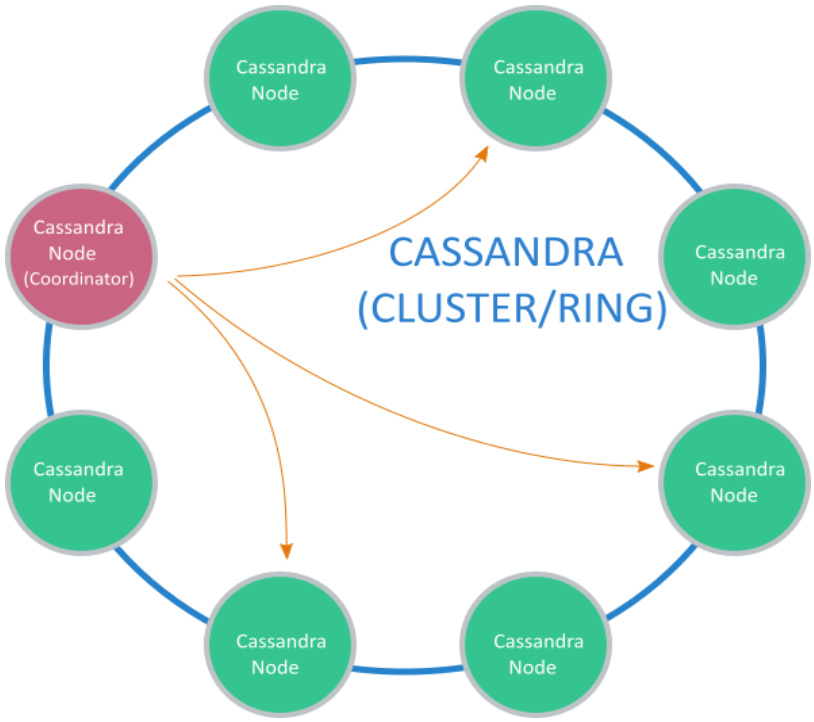
بايثون لغة برمجة من اللغات ذات المستوى العالي، تتميز ببساطة كتابتها وقراءتها و تستخدم أسلوب البرمجة الكائنية (OOP) ، وهي مفتوحة المصدر، وقابلة للتطوير. وهي تعتبرلغة تفسيرية، متعددة الأغراض تستخدم بشكل واسع في العديد من المجالات، كبناء البرامج المستقلة بإستخدام الواجهات الرسومية المعروفة، وفي عمل برامج الويب، بالإضافة إلى استخدامها كلغة برمجة نصية للتحكم في أداء بعض من أشهر البرامج المعروفة، أو في بناء برامج ملحقة لها، وبشكل عام يمكن إستخدام بايثون لبرمجة البرامج البسيطة للمبتدئين ولإنجاز المشاريع الضخمة كأي لغة برمجية أخرى في نفس الوقت. [16]

4.3 :PostgreSQL

هي قاعدة بيانات علائقية مفتوحة المصدر، ترتبط مع نظام Odoo عن طريق ال ORM ، وتعتمد عليها اغلب انظمه الأودو المطورة حالياً ، قمنا باستخدامها في حالة ان النظام الذي يريد الارتباط ب Cassandra يعمل كنظام مستقر مما يعني وجود بيانات في قاعدة بياناته، و بالتالي يتوجب جلب البيانات الى بيئة تخزين Hadoopمن قاعدة PostgreSQL.[3]

5.3 قاعدة بيانات Cassandra:

هي نوع من أنواع قواعد البيانات اللاعلائقية(NoSQL) من نوع (key, value systems) توفر قاعدة بيانات لامركزية موزعة طورت في البداية عن طريق Facebook والآن هي تحت مظلة أباتشي مفتوحة المصدر، يتم إستخدامها من قبل شركات كبرى منها Facebook و Tweeter، تحتوي (Cassandra)على مجموعة من العقد المتماثلة لإعتمادها على معمارية ال (Peer-to-peer) ، ولا تقوم على مفهوم العقدة الرئيسية كما في معمارية (Master/slave) ، وتتواصل العقد مع بعضها البعض عن طريق بروتوكول يسمى (Gossip Protocol).[10]



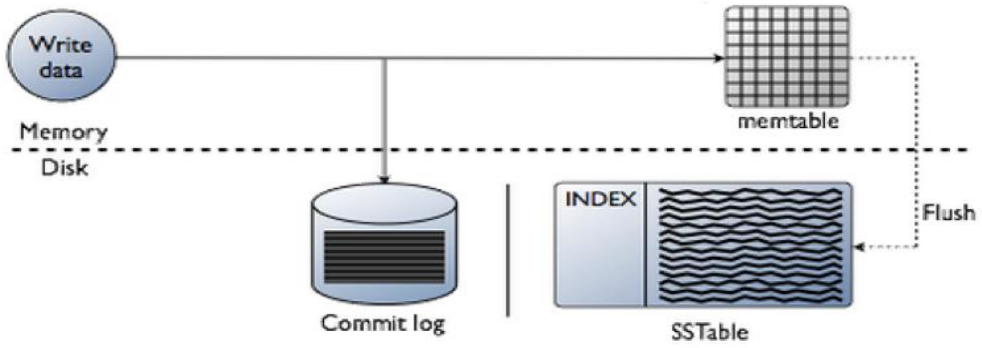
شكل (1.3) معمارية Cassandra

جدول رقم (1.3) نماذج بيانات Cassandra و ال RDBMS [10]

RDBMS	Cassandra
خادم قاعدة البيانات (Database server)	عنقود (Cluster)
قاعدة بياناتDatabase	keySpace
جدولTable	Column family
لغة الإستعلام المهيكلهSQL	CQL
سطرعمودRow columns	Row, columns
الإتحادالربطJoin Union	لايوجد
التقسيماتPartition	Partition

1.5.3 عملية الكتابة في Cassandra :

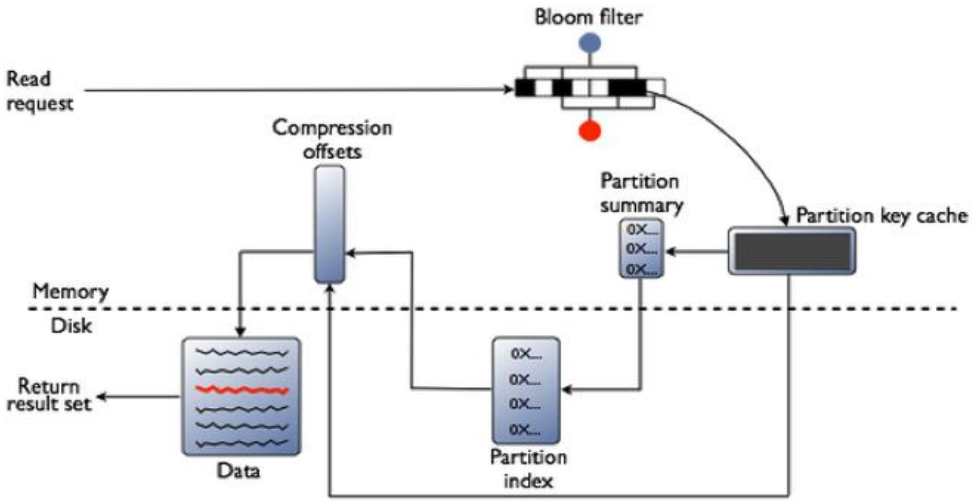
تقوم بالكتابة أولاً على القرص الصلب ثم تتم الكتابة بعدها على ذاكرة رئيسية تسمى ب(Memtable) وعندما يزيد حجم البيانات في ال(Memtable) تتم كتابة البيانات على قرص غير قابل للتعديل يسمى ب(SStable) وتتم كل هذه العمليات في وقت واحد مما يعطي (Cassandra) القدرة العالية على الكتابة ، بأقل زمن تأخير .



شكل رقم (2.3) عملية الكتابة في Cassandra

2.5.3 عملية القراءة:

أما قراءة البيانات من (Cassandra) فتتضمن العديد من العمليات حيث يتم طلب القراءة من بنية في الذاكرة تسمى (bloom filter) والتي تقوم بإختبار ال(SStable) إن كان يحتوي على بيانات أم لا بشكل سريع.



شكل رقم (3.3)عملية القراءة في Cassandra

3.5.3 مميزات قاعدة بيانات Cassandra:

1- ذات سعه تخزين كبيرة ، تتناسب مع البيانات الكبيرة.

2- سهولة توزيع البيانات على العقد .

3- سرعة عملية الكتابة، مقارنة بعملية القراءة .

4- تعتمد على معمارية الـ (Peer-to-peer) مما يميزها بعدم وجود نقطة فشل وحيدة للنظام.

5- توفر خاصية الـ (Replication) تكرار البيانات مما يزيد من إتاحية وسرعة إسترجاع البيانات .

6- المرونة العالية في التوسع و التوزيع : حيث أنها تتميز بالقابلية العالية لإضافة المزيد من الأجهزة والمزيد من البيانات.

7- إستيعابها لجميع انواع البيانات بما فيها : البيانات المهيكله وشبه المهيكلة والغير مهيكلة ويمكن أن تستوعب التغييرات على هياكل البيانات حسب الحاجة.[10]

6.3 python driver-Cassandra:

هو عبارة عن (ORM) مفتوح المصدر بلغة بايثون يتيح التعامل مع قاعدة بيانات (Cassandra)، و يدعم العديد من العمليات التي تتم في (Cassandra)، ولكنه مازال لايدعم كل الخصائص والعمليات التي يمكن القيام بها، وتوفر شركة (DataStax) مخطط علائقي خاص بـ(Cassandra) كخيار تجاري غير مجاني، لذلك في هذا البحث تم إستخدام هذا المخطط مفتوح المصدر.[10]

7.3 Hadoop :

وقد تم إستخدامة في هذا البحث لتخزين البيانات في نظام الملفات الموزع (HDFS)، ومن أهم المميزات التي يتميز بها هي دعمه للمعالجة المتوازية(Parrallel processing)، إمكانية إدارة و التعامل مع كميات كبيرة من البيانات، إمكانية تخزين أنواع مختلفة من البيانات فيه (مهيكله، شبه مهيكله، غير مهيكله).[3]

8.3 Zookeeper:

هو عبارة عن خدمه(service) مستخدمة بواسطة مجموعه من العقد(nodes) لتنظيم وصيانة البيانات المتبادلة بينها وذلك بإستخدام تقنيات متزامنة، و يستخدم في بيئة عمل (Hadoop)لإدارة الموارد في النظام ، و يتميز ببساطة تنظيم وتوزيع العمليات، المزامنة في عمليات المعالجة، التسلسل في التنفيذ، و دعمه لخاصية التحقق.[17]

9.3 Oozie:

عباره عن نظام جدولة يقوم بتنفيذ وإدارة وظائف "Hadoop" كما يسمح بتنفيذ عدد من الوظائف المعقدة بأوامر متتاليه لتحقيق مهمات كبيرة، ومن أهم مميزاته تكامله مع"Hadoop" ودعمه لوظائف أنظمة مثل Hive, Pig, Sqoop ، قيامه بالتحقق من إكتمال المهام في النظام وتتابع العمليات.[18]

:Hive 10.3

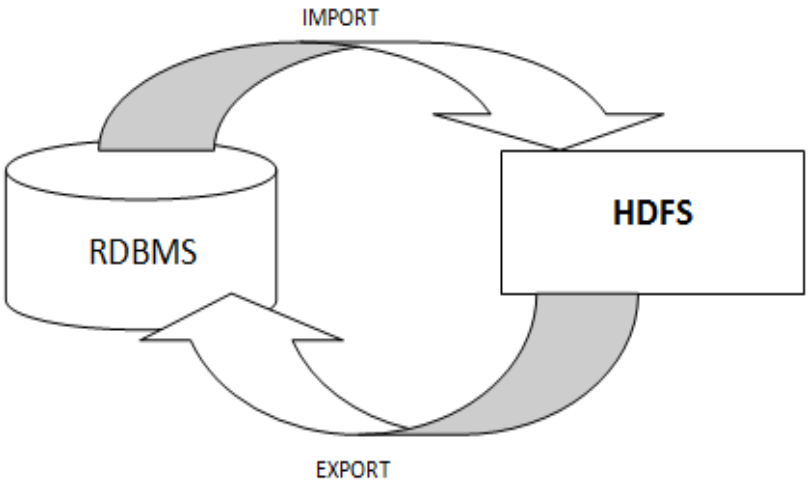
هو مستودع بيانات للإدارة والإستعلام عن البيانات المنظمة المبنية على نظام الملفات الموزع(HDFS) الخاص ب (Hadoop)، يقوم بإستخدام الـ (Map Reduce) لتنفيذ عمليات المعالجة، كما أنه قابل للتوسعة في مستودعات البيانات الأخرى، و يتيح للمطورين إمكانية التعامل مع البيانات بإستخدام صيغ الـ (SQL) ولكن بإستخدام لغة الإستعلامات التي توفرها والتي تسمى (HQL)، والتي تمكن من إجراء عمليات معالجه معقده على البيانات وتدعم عمليات الـ (Join) عكس اغلب قواعد البيانات اللاعلائقيه التي لاتدعم هذه العمليات، تم إستخدامه في هذا البحث لإجراء عمليات المعالجة بإستخدام MapReduce على البيانات المتواجدة في نظام الملفات الموزع(HDFS).[1]

:Sqoop 11.3

هي أداة صممت لنقل البيانات بين خادم قاعدة البيانات العلائقية و بيئة نظام Hadoop ، وتعمل على جلب وتصدير البيانات بين النظامين، وقد ضُمت هذه الأداة الى رخصة أباتشي مفتوحة المصدر في العام 2013 .

مميزاته :

1- سهولة إستخدامها في نقل البيانات من وإلى قواعد البيانات العلائقية و Hadoop
2- توفر إمكانية النقل التدريجي للبيانات الكبيرة.
3- الدعم الكبير من قبل المطورين للأداة.
4- إمكانية نقل كل أو بعض جداول قاعدة البيانات.[1]



شكل رقم (4.3) يوضح عمل Sqoop

كما تم إستخدام عدد من تقنيات الويب لعرض نتائج البحث في واجهات، وهذه التقنيات هي:

12.3 HTML و CSS و Java Script و Bootstrap

13.3 Django framework:

هو إطار عمل خاص بتطوير تطبيقات الويب بلغة بايثون، ونظراً لخصائصة المتعددة يعتبر من أبرز إطارات العمل الخاصه بتطبيقات الويب، طور بواسطة فريق مصممين في نهاية العام 2003 .

مميزاته :

1- يدعم ال"ORM" والإتصال بقواعد بيانات مختلفة منها ال NoSQL DB.

2- يدعم مجموعة من تقنيات الويب الحديثة مثل " Ajax, RSS,...الخ"

3- يقدم واجهات جيدة ومتكاملة لمدراء التطبيقات.

4- يمتلك بيئة عمل متطورة. [20]

14.3 خلاصة الباب:

في الباب السابق تم عرض الأدوات و التقنيات التي تم إستخدامها في البحث بعرض أهم مميزاتها ومكان استخدامها لتطبيق الحل المقترح في البحث .

الباب الرابع
النتائج و التوصيات

1.6 مقدمة :

يتناول هذا الباب نتائج البحث وعرض ماتوصلت اليه الدراسة، وعرض التوصيات التي يوصي بها البحث في مواصلة العمل عليه مستقبلاً.

2.6 النتائج:

في هذا البحث تم تصميم وتطبيق مستودع للبيانات لأنظمة تخطيط موارد المؤسسة وذلك لأرشفتها، كما تم بناء وحدة فى نظام Odoo للتخزين والمعالجة المباشرة للبيانات الضخمة، حيث تم تحقيق النتائج التالية:

1- تصميم وتطبيق مستودع للبيانات لأنظمة تخطيط موارد المؤسسة وذلك لأرشفتها
2- تطبيق المعالجة المتوازية على البيانات بإستخدام نموذج البرمجة MapReduce.
3- بناء وحدة فى نظام Odoo للتخزين والمعالجة المباشرة للبيانات الضخمة و تمَكُن من التفاعل المباشر مع قاعدة بيانات Cassandra، مع تحقيق إتاحية و سرعة معالجة أعلى.

3.6 التوصيات:

وبناءً على ماسبق فقد خرج البحث بالتوصيات التالية:

1- الإستفادة من مستودع البيانات في تطبيق عمليات ذكاء الأعمال وتنقيب البيانات لمؤسسات المختلفه.

2- الأخذ بعين الإعتبار خاصية الإتساق(consistency) للبيانات في بعض الحالات.

4.6 الخاتمه

تم بحمد الله ما أردنا عمله من معالجة وتخزين البيانات من أجل أغراض الأرشفة أو التخزين والتعامل المستمر مع بيئة التخزين الجديدة.، هذا ونسأل الله العلي القدير أن ينفع بعملنا هذا كل طالب علم، وأن يجعله في ميزان حسناتنا، هذا فإن أصبنا فهي نعمة من الله عز وجل و إن أخطأنا فمن أنفسنا و الشيطان

المصادر و المراجع

[1] Fawaz Enaya, “An Experimental Performance Comparison of NoSQL and RDBMS Data Storage Systems in the ERP System Odoo”, 2016 University of Magdeburg, pages 1,7,46.

[2] K.Gordon, "*Principles of data management: Facilitating information sharing"*, 2007, ISBN 978-1-902505-84-8, The British Computer Society,page 4.

[3] R.Elmasri,S.Navathe," Fundemental of database system "*Seventh Edition,*2007,2011,2016, ISBN-13: 978-0-13-397077-7, Library of Congress Cataloging-in-Publication Data on File.

[4] N. Mali1, S.Bojewar ,"A Survey of ETL Tools" , 2015,"International Journal of Computer Techniques – Volume 2 Issue 5".

[5] M.Sumner,"*Enterprise Resource Planning*, First Edition", ISBN 10: 1-292-03980-9 ,2014, Pearson New International Edition.

[6] https://www.tutorialspoint.com/apache_oozie/apache_oozie_tutorial.pdf "4:28:AMالساعةp. 2017\7\6 .اليوم الخميس

[7] https://www.odoo.com/forum/help-1/question/tutorial-that-explains-interaction-with-the-openerp-orm-18168 11:30AM . 2017\7\10 :السبت.

[8] Sabine Sonnentag, Michael Frese “Performance Concepts and Performance Theory,” *Psychol. Manag. Individ. Perform.*, pp. 1–25, 2005.

[9] *http://www.roseindia.net/jpa/jdbc-vs-orm.shtml* 1:10 PM. 2017\10\14 السبت

[10] Eben Hewitt,"Cassandra: The Definitive Guide",2011, ISBN: 978-1-449-39041-9, O’Reilly Media, Inc.

[11] R.Sharma, Y.kashyap, "A STUDY OF NOSQL DATABASES AND WORKING OVERVIEWS ",2016, International Journal of Recent Trends In Engineering And Research, ISSN: 2455-1457.

[12] Garry Turkington,"Hadoop Beginner's Guide",2013, ISBN 978-1-84951-7-300, Packt Publishing.

[13] A.Singh,R.K.Bansal,"Open Source Software vs Proprietary Software", International Journal of Computer Applications (0975 – 8887) , Volume 114 – No. 18, March 2015

[14] B. Radulović, D. Radosav, and M. Malić, “The Application of NoSQL MongoDB in Developing the EPR System for Managing Human Resources,” vol. 3, no. 1, pp. 181–185, 2016.

[15] http://odooarabia.org/about/odoo/ 2:32 PM 2017\7\21 الأربعاء

[16] https://www.python.org/about , 7:12 PM 19/8/2017 السبت

[17] http://www.tutoriaspoint.com/zookeeper_tutorial.pdf 2017/4/15: السبت

4:32 PM.

[18] Radulović , B.Radosav ,"The Application of NoSQL MongoDB in Developing the EPR System for Managing Human Resource ", Int'l Journal of Computing, Communications & Instrumentation Engg. (IJCCIE) Vol. 3, Issue 1 (2016), ISSN 2349-1469 ISSN 2349-1477.

[19] T.Jacha, Ewa Magiera, "Application of HADOOP to Store and Process Big Data Gathered from an Urban Water Distribution System" ,13th Computer Control for Water Industry Conference, CCWI 2015.

[20] https://media.readthedocs.org/pdf/django/latest/django.pdf
4:28 AM 2017\6\4 : الأحد

[21] https://pdfs.semanticscholar.org/79d7/e5ef1752c611cb0aff107ee900216c34adfb.pdf
2017\8\16:السبت, 12:30 PM