SERVICE ORIENTED ARCHTIECTURE PARADIGM FOR BUSINESS INTELLIGENCE

[1]Akash Ajay Borse, [2]Siddhant Verma, [3]Vipin Wani, [4]Gardas Naresh Kumar

[1]Student, [2]Independent Researcher, [3]Assistant Professor, [4]Project Manager
[1]Department of Computer Science and Engineering,
[1]Sandip University, Nashik, India

Abstract: Adoption of Service Oriented Architecture (SOA) in an organizational environment is having its own potential benefits. Studies has proven that a Service Oriented Architecture can emphatically impact on overall business process and performance enhancement with its powerful working capabilities. SOA is an art of arranging the architectural components in such a way they can communicate with each other over reliable services. It provides a flexible environment for architects and developers to design, develop, deploy and/or modify this service-oriented architecture. This event-driven architecture is also capable of measuring the maturity of business model. SOA in Business Intelligence (BI) domain has credible benefits to enhance the processing and analysis capabilities over huge data, as enormous amounts of data is being generated by every industrial or domestic organization. This proposed work on Service Oriented Architecture for Business Intelligence provides a generic architecture which can be used as base or reference architecture for any organization whose intend is to work on Big Data and Business Intelligence.

IndexTerms **- Service Oriented Architecture, maturity, event-driven, Business Intelligence.**

I. INTRODUCTION

Service Oriented Architecture (SOA) is a trending concept and there has been continuous revolution since a decade. Unlike monolithic architecture, SOA is having its own potential capabilities and benefits; in the sector of Business Management in terms of Business Processes Management, Business Intelligence with Predictive Analytics and many more other application areas. The fundamental conventions of SOA are independent of vendors, products, and technologies. SOA is defined as "An approach of designing a software where the services are served to other in-environment components by application components, via a communication protocol over a network". On the other hand, Business Intelligence (BI) is another area of Data Science which deals with processing of huge amounts of historical or real-time fact based datasets to produce the helpful patterns and insights for ultimate analysis purpose. As enormous amounts of data is being produced in day-to-day processes/actions from domestic & corporate industries and also other internal and external data sources, the industries are looking for faster, robust and yet efficient solutions to gain insights from these huge amounts of produced data with ultimate goal for their business growth. Incorporating Service Oriented Architecture (SOA) designing paradigm for the purpose of Business Intelligence (BI) can produce and serve the better solutions to determine, address and resolve the issues occurred in organizations while implementing and using Business Intelligence solution using traditional approach. In present most of the companies has adopted and are utilizing SOA functionalities for their internal use for the purpose of Business Intelligence and yet many more companies are moving towards leveraging this SOA approach for BI.

This paper in further sections states the aim of research, discussion on existing scenario, related works, proposed work, results and analysis, and in the end conclusion and future scope of study.

II. AIM OF RESEARCH

The aim of this research and proposed work is the enhancement of the performance of overall Business Intelligence (BI) process and designing reliable, sustainable, reusable and efficient web services for carrying out BI operations using Service Oriented Architecture (SOA) paradigm.

A. Objective of Research

To create a generic architecture for Business Intelligence based on reliable services using Service Oriented Architecture paradigm.

B. Scope of Research

1) To study capabilities and potential benefits of using Service Oriented Architecture for domain of Business Intelligence.
2) To study and implement Business Intelligence operations over Service Oriented Architecture Paradigm.

III. DISCUSSION ON EXISTING SCENARIO

Business Intelligence (BI) is an art of applying various set of tools and methodologies over huge sets of seasonal/ non-seasonal, structured or unstructured data with the objective of gaining useful insights for the ultimate goal of business growth. Traditional approach of Business Intelligence follows the stages of Data Collection, ETL Process, Data Analysis, and Data Visualization. Following is a general Business Intelligence (BI) process flow:

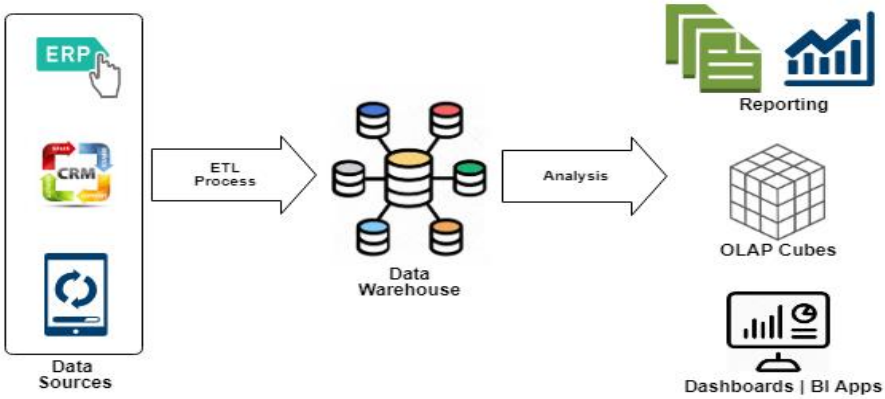


fig. 1: business intelligence process flow[1]

1) *Data Collection*

The first stage of Data Warehousing where data is collected from different internal and/or external data sources like ERP (Enterprise Resource Planning), SAP (Systems Applications and Products), CRM (Customer Relationship Management), and SCM (Supply Chain Management). This data can be in any form like structured data, unstructured data and/or semi-structured data.

2) *ETL Process*

ETL stands for Extract-Transform-Load process, it is enforced over collected data from internal and/or external data sources for dat cleansing, transforming and loading of these data into data warehouse.

3) *Data Warehousing*

A Data Warehouse is a huge archive of collected data from different internal and/or external data sources, which contains huge amounts of historical data. Data Warehousing is a practice of loading/storing of the data into a data warehouse.

4) *Data Analysis*

Data Analysis is the practice of selecting a data cube from the data warehouse and enforcing various algorithms and methods for identifying useful patterns in that data (both structured and/or unstructured) for the purpose of gaining useful insights with ultimate goal of business growth.

5) *Data Visualization*

Data Visualization is a process and an art of illustrating the targeted data or patterns with the help of various interactive and understandable visual forms like shapes, charts, and graphs. And these visualizations are combinely/collectively represented as "Dashboards".

Based on the studies, it is found that there are many limitations with the existing BI systems/ solutions which affects the overall performance of the system in terms of data collection from various heterogeneous data sources, performing OLAP operations over these enormous amounts of data in real time environment, communication overhead among different dependent/ independent components of system, and ultimately producing useful insights from the extracted patterns and facts for analysis purpose.

A. Limitations of Existing Systems

Data collection and Data Warehousing are the primary and essential stages in every other Business Intelligence (BI) process. Based on the tests and analysis, it is observed that the stage of Data Warehousing is the most time consuming process in overall BI phenomenon, and these traditional BI systems are retaining some of the common limitations as compared to present Service Oriented Architectures (SOA) which are as follows:

1) *Piling of Historical Data*

The main objective of Data Warehousing in a BI system is to collect and sustain all collected historical data for the purpose of future operations i.e. to help analysts and experts to deal with the extracted patterns and helpful facts from data for analysis purpose.

2) *Cost*

Implementing and maintaining a BI solution can be quite expensive for even basic operations in terms of training experts and incorporating new functionalities.

3) *Complexity*

While implementing a BI solution it could be troublesome to organize and implement the data, which may lead to have difficulty in accessing and processing of data.

4) *Time-Consuming Implementations*

Even though today's industries are fast enough to rapidly plan and implement all necessary tools and technologies, it takes a quite long period for implementing a BI solution as it involves stage of data warehousing, which can consume more time as data is growing much rapidly in enormous amounts.[1]

IV. RELATED WORK

There are various researches and studies have been over domains of Business Intelligence (BI), Service Oriented Architecture (SOA), their implementation approaches and many more. Some of the studies and implementations from other researchers are described further in this section.

Akash Ajay Borse, Siddhant Verma and others[1] has done a detailed review and survey on Business Intelligence, Challenges in BI, Service Oriented Architectures and Limitations of existing systems. It provides a study on how Service Oriented Architecture (SOA) can improve the processing of data much efficiently and also gives the limitations of the traditional approach.

Arun D. Patil and N. D. Gangadhar [2] states the purpose and impact of using OLAP in advanced BI solutions and provides the prototype for OLAP as a Service. Web Service based API would enable applications to use Service Oriented Architecture (SOA) for Big Data Analytics and also easily be deployed on a Cloud. This paper comprises the designing and prototyping of an OLAP oriented Platform as a Service, termed as OLAP as a Service (OLaaS).

Anurag Shashwat and Deepak Kumar[3] discover the challenge of service identification issues and propose a model which will help in clustering the service repository based on functionality of services. The goal of this paper is to optimize service identification process of SOA lifecycle.

Papazoglou, Michael P., and Willem-Jan Van Den Heuvel[4] provides an overview of methods and techniques used in Service-oriented design and development. This paper explores a service development methodology from the aspect of both service producers and requesters and reviews the range of factors in this methodology which are available to them for consideration.

Klose, Karsten, Ralf Knackstedt, and Daniel Beverungen[5] presents an approach to design and implement the concept of SOA. Also, authors have carried out a survey to identify various basic services and process services in Service Oriented Architecture.

Maia, Italo[6] in a book of Building Web Applications with Flask, states the development of robust web applications with Flask and Python. Flask is considered as a minimalistic solution where we're given the bare minimum work with and choose all other stuff, which provides a granular control over application.

Wu, Rui[7] has introduced a system to manage, process and visualize large data. Authors have used MongoDB to recordfile paths and stored files in a filesystem to manage data. By using this method, we do not need to create different schemas for different files. The system offers users basic data processing methods, such as distribution frequency histograms and also provides the scientific models to users for data processing.

Ramos Somya and other authors[8] has implemented a prototype of reference model for Service-Oriented Business Intelligence (SoBI) using academic and financial data from Satya Wacana Christian University (SWCU), and claims that if SoBI model is applied instead of traditional BI models, the ETL (Extract-Transform-Load) process which is a much more time consuming process especially when data integration can be simplified with the help of web services. Authors have used for their case-study implementation.

Martin Mikuf and other authors[9] has implemented a case-study to setup web portals and create reports. Authors have used BI tools in Visual Studio from Microsoft and Cognos from IBM. They concluded that how switching to new technologies can affect and enhance a company's reporting process. Authors acquired the real manufacturing data & work from company.

M. Miskuf[10] and I. Zolotova[10] has implemented a case-study to generate the regular BI reports for a company based on real acquired manufacturing data of company. Author's main objective was to identify how a company can save their money if they use already owned BI solution for basic reporting. They states that based on used technologies Cognos from IBM seems to be bit more complex solution for this purpose according to its ETL process when collecting data from sensors.

V. PROPOSED WORK

The aim of this research and proposed work is the enhancement of the performance of overall Business Intelligence (BI) process and designing sustainable and efficient web services for carrying out BI operations. A Service Oriented Architectural (SOA) approach can be the best-fit solution for all the issues mentioned in previous chapter of Existing Systems; hence, this proposed model/ system makes use of this approach to design a generic SOA based BI architecture, which will have loosely coupled & independent system components. These independent components can communicate over network with the help of web-based services/ protocols. This proposed model can be most suitable for real-time/ time-critical BI operations like Data Loading and Quick Analysis. Based on the studies, many factors needed to be considered while designing this proposed solution. These factors can be based on performance, OLAP operations, and analysis services. Following is our proposed architecture for Business Intelligence based on Service Oriented Architecture paradigm:

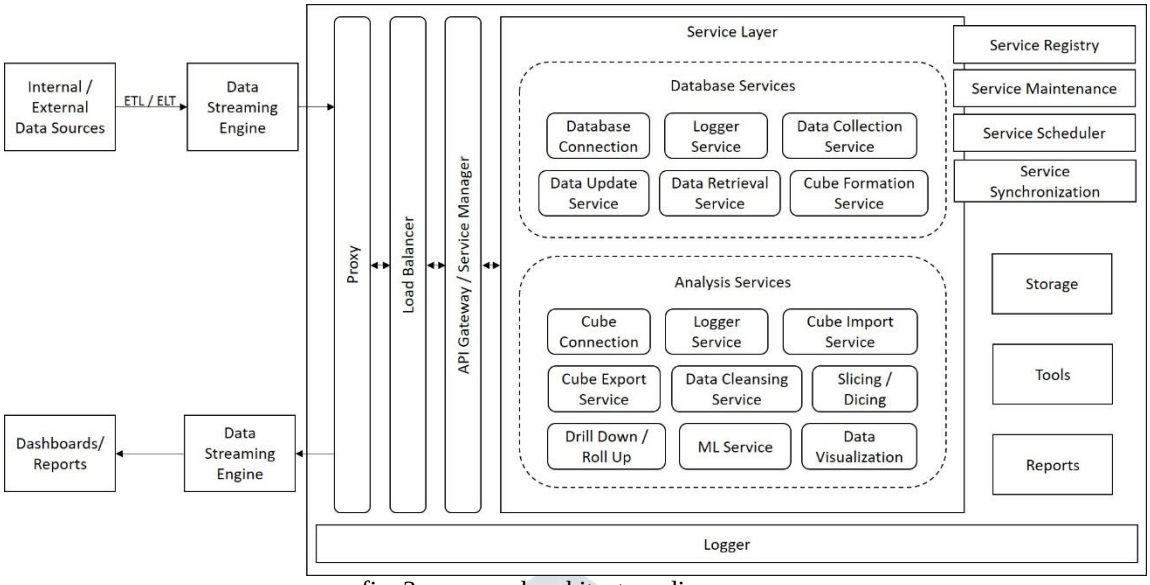


fig. 2: proposed architecture diagram

This architecture consists of various entities, processes, tools and database. It supports cross-platform data handling and processing, real-time logs, various data visualizations using cross-platform tools API's. This architecture is a generic architecture and can be easily adapted/accommodated or integrated and implemented with existing business strategies. It provides agile and faster development of reliable services which can easily be integrated or removed as per business requirement. Following are the elements or components of this proposed architecture:

1) *Internal/External Data Sources:*
Data can be collected from any heterogeneous/distributed internal or external sources such as ERP (Enterprise Resource Planning), SCM (Supply Chain Management), CRM (Customer Relationship Management), Data Marts and so on.

2) *ETL/ELT:*
ELT (Extract, Transform, and Load) is a process of extracting or collecting data from Internal/External Data Sources, Transforming it (Pre-Processing and Cleansing) and Loading it in Data Warehouse. On the other hand ELT is a process of extracting or collecting data from Internal/External Data Sources, Loading it first in the Data Warehouse and then Transforming it as per need. ELT boosts up the Data Warehousing process as data is loaded as it is instead of transforming in prior.

3) *Data Streaming Engine:*
This is responsible for streaming of bulks of datasets in or out to/from system.

4) *Proxy:*
It translates or resolves service calls between two different Client-Service protocols. E.g.: Receiving SOAP calls and translating into RMI calls for interaction between systems.

5) *Load Balancer (Optional):*
It provides a conventional performance gain if a lot of clients are accessing a service concurrently.

6) *API Gateway/Service Manager:*
It is responsible for delivering comprehensive standards-based security, routing, monitoring and management of SOA and Web Services.

7) *Service Layer:*
It consists of number of well-defined web services in service registry or repository. In our architecture we are proposing BI related services which comes under Database operations and Analysis over data.

8) *Service Registry:*
It is a repository or collection of number of web services.

9) *Storage:*
It is consisting of Databases and Memory required for processing on storage device.

10) *Tools:*
It consists of various different tools required for Business Intelligence. E.g.: Tableau.

11) *Reports:*
It consists of static or dynamic/real-time reports, graphs and visualizations generated after analysis of data.

12) Logger:

It logs the internal workings of the module. E.g.: In our architecture Flask uses internal logger which can log tracebacks of 500 errors during debug mode.

13) Dashboards and Reports:

Dashboard is a portal of collective visualizations which effectively describes the insights graphically/visually. These visualizations and reports can be delivered and viewed on various platforms like standalone system or web browsers. E.g.: Tableau Public, Plotly, HighCharts, or static images.

A. Tools and Methodologies

For implementation of prototype for our proposed architecture, various tools and methods have been used which are as follows:

1) Flask – A Python Web Framework

Flask is a popular python web framework which is used for developing web applications. It is based on Werkzeug WSGI Toolkit and Jinja2 Template Engine. We have used this web framework on the top of which we have implemented some BI related web services.

2) MongoDB – NoSQL Database

MongoDB is an open-source document-based database and a leading NoSQL database. We have used this database to show that our proposed system can support not just relational but non-relational databases as well.

3) Flask-PyMongo

Flask-PyMongo is a python library which bridges Flask and MongoDB (PyMongo) and provides some convenience helpers for operating.

4) Matplotlib – Plotting Library in Python

Matplotlib is a plotting library in python which can used in python scripts, web application servers and GUI toolkits. It can generate plots, bar charts, histograms and many more. We have used this library for implementing Data Visualization module in our architecture.

Following are some web-services samples which are implemented over our proposed architecture and it elaborates the deployment of traditional BI operations over web-services:

1. Inserting data into database:

```
@app.route('/star/Insert')
def form_data():
  return render_template('Form.html')
@app.route('/star/Insert', methods=['GET','POST'])
def add_star():
  if request.method == 'POST':
    name = request.form['name']
    distance = request.form['distance']
    if name and distance:
      star = mongo.db.stars
      star_id = star.insert({'name': name, 'distance': distance})
      new_star = star.find_one({'_id': star_id })
            output = {'name' : new_star['name'], 'distance' : new_star['distance']}, "Data Inserted Successfully...!"
    else:
      output = "Data Was Not Inserted...!"
    return jsonify({'result' : output})
```

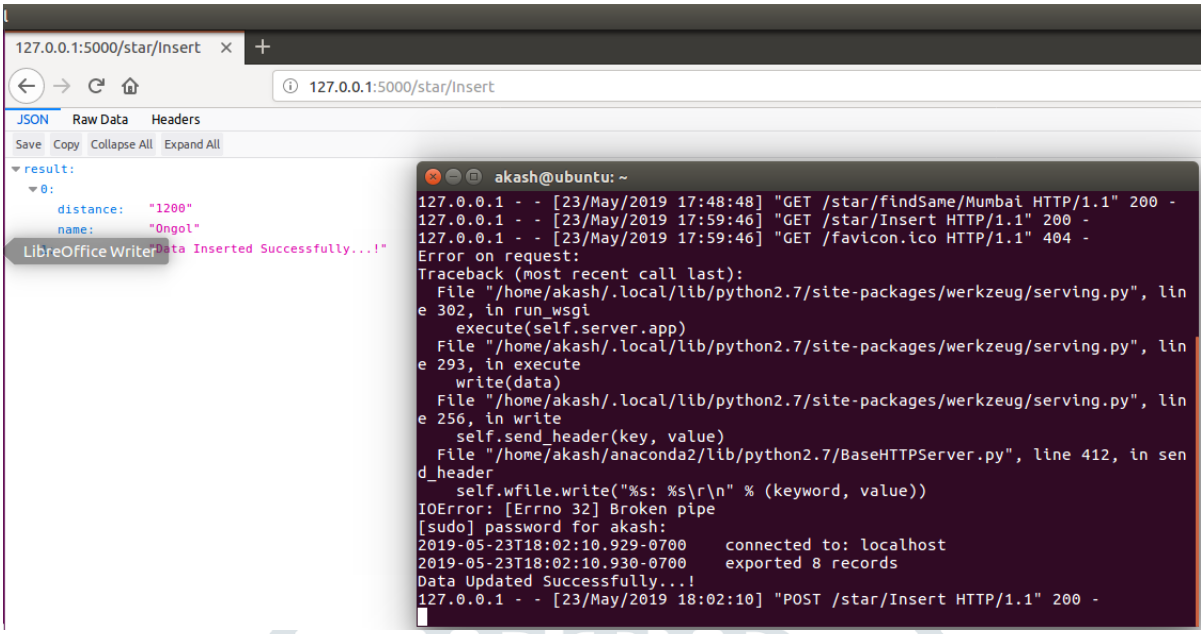


fig. 3: inserting data into database

2. Fetching all data from database:

```
@app.route('/star', methods=['GET'])
def get_all_stars():
  star = mongo.db.stars
  output = []
  for s in star.find():
    output.append({'name' : s['name'], 'distance' : s['distance']})
  return jsonify({'result' : output})
```

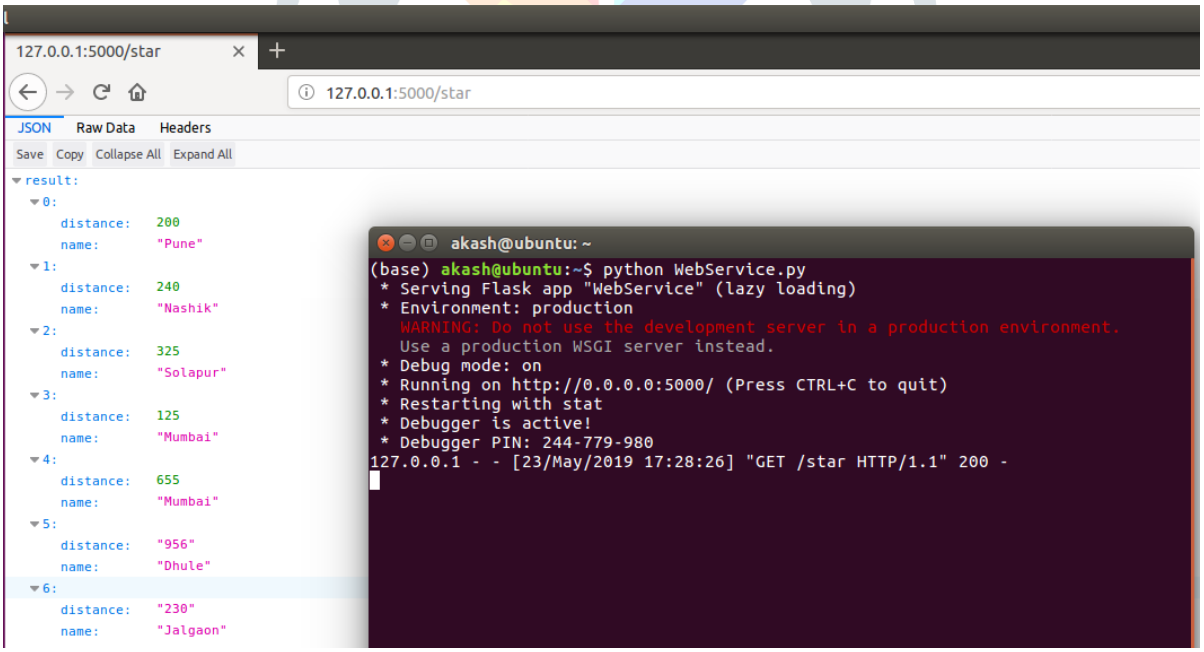
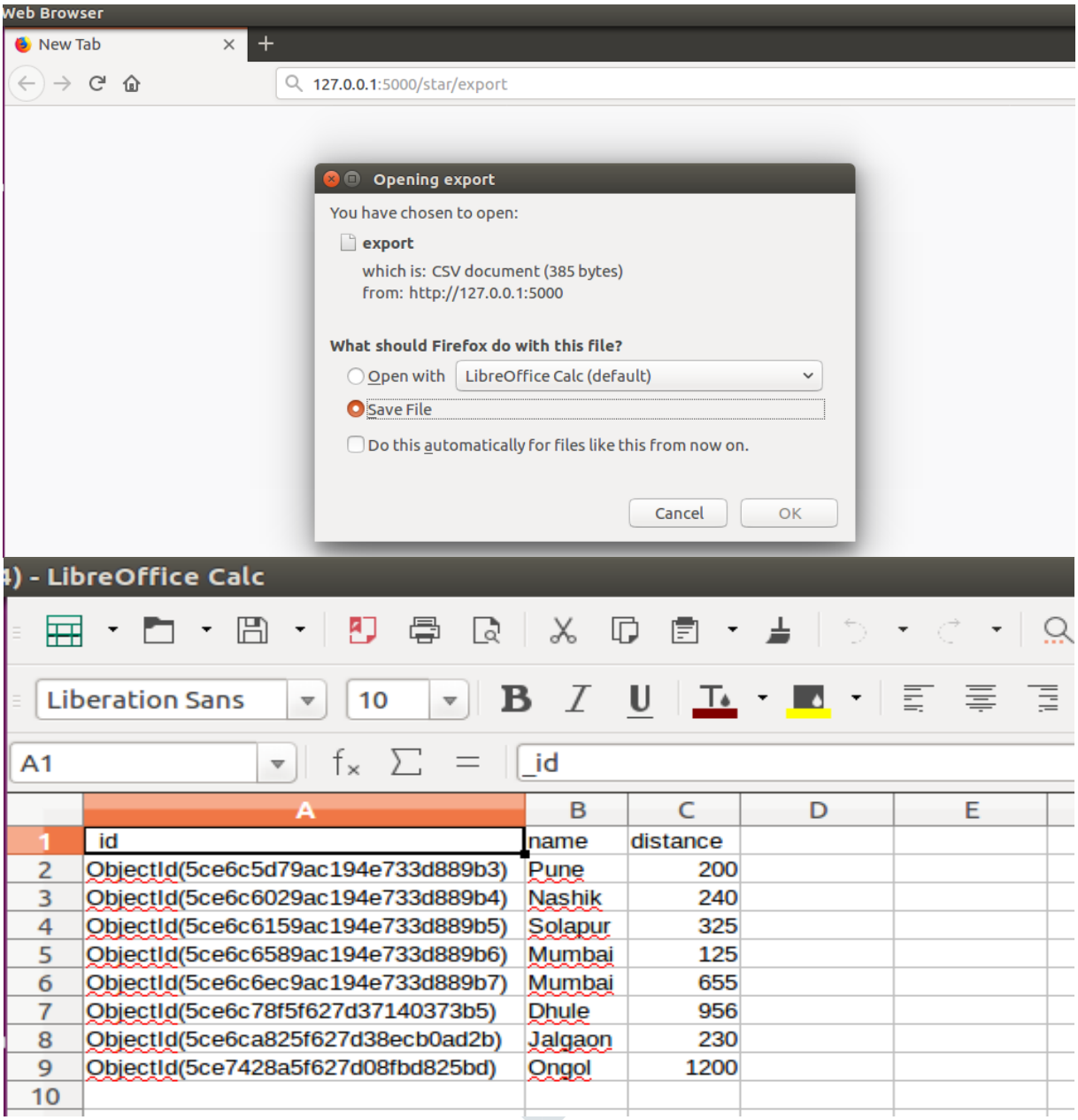


fig. 4: fetching data from database

3. Exporting data:

```
@app.route('/star/export')
def export_csv():
  return send_file('/home/akash/data.csv')
```



	A	B	C	D	E
1	id	name	distance		
2	ObjectId(5ce6c5d79ac194e733d889b3)	Pune	200		
3	ObjectId(5ce6c6029ac194e733d889b4)	Nashik	240		
4	ObjectId(5ce6c6159ac194e733d889b5)	Solapur	325		
5	ObjectId(5ce6c6589ac194e733d889b6)	Mumbai	125		
6	ObjectId(5ce6c6ec9ac194e733d889b7)	Mumbai	655		
7	ObjectId(5ce6c78f5f627d37140373b5)	Dhule	956		
8	ObjectId(5ce6ca825f627d38ecb0ad2b)	Jalgaon	230		
9	ObjectId(5ce7428a5f627d08fbd825bd)	Ongol	1200		
10					

fig. 5: exporting data

4. Static Graph Visualization:

```
@app.route('/star/result', methods = ['GET', 'POST'])
def graph_gen():
  df1=pd.read_csv("/home/akash/data.csv", header = None)
  df1=df1.drop(df1.columns[0], axis = 1)
  df1=df1.dropna()
  df2 = df1.loc[1:,1]
  df3 = df1.loc[:,2]
  df3 = np.asarray(df1.loc[1:,2])
  df3 = df3.astype('float')
  p = pd.DataFrame({'Name': df2, 'Distance':df3})
  p.plot.bar(x='Name', y = 'Distance', rot = 60)
  plt.savefig('/home/akash/static/graph.jpeg')
  image_path = '/static/graph.jpeg'
  return render_template('Graph.html', name = 'Graph Analysis', url='/home/akash/static/graph.jpeg', mimetype =
'image/jpeg')
```

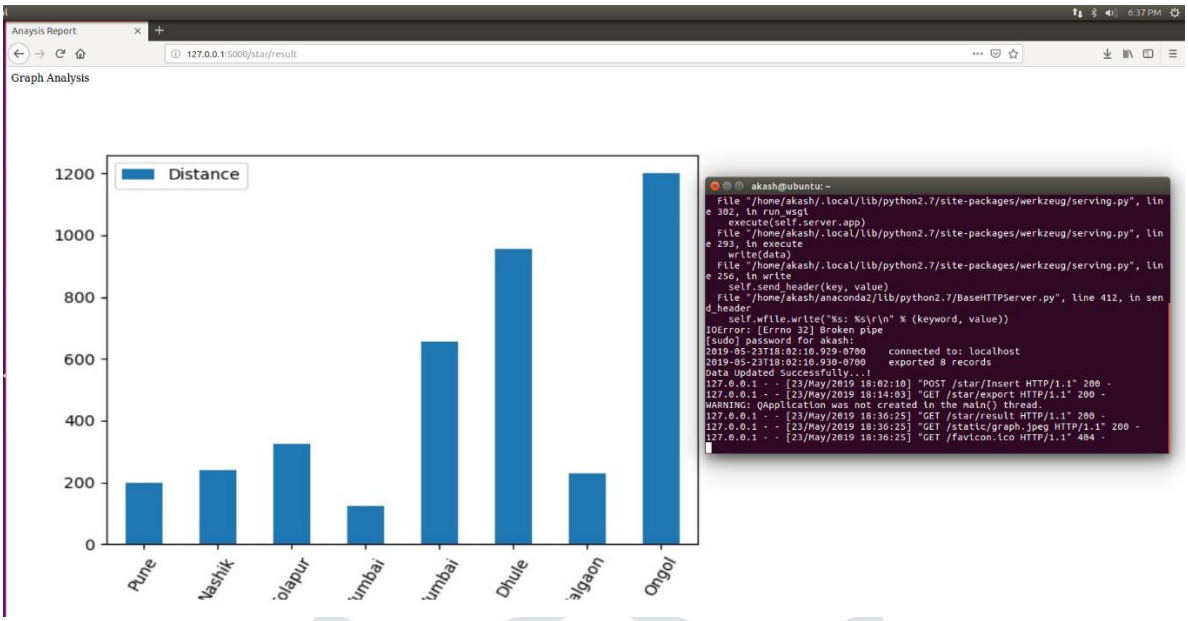


fig. 6: static graph visualization

5. Real-Time Graph Visualization:

```
import pandas as pd
import StringIO
@app.route('/star/dynamic')
def send_csv():
  df = pd.read_csv("/home/akash/data.csv")
  df = df.drop(df.columns[0], axis = 1)
  output = StringIO.StringIO()
  df.to_csv(output, index = False)
  return Response(output.getvalue(), mimetype="text/csv")

@app.route('/star/result_data', methods = ['GET','POST'])
def send_data():
#   csv_file = "/home/akash/data.csv"
  return render_template('LiveGraph.html')
```

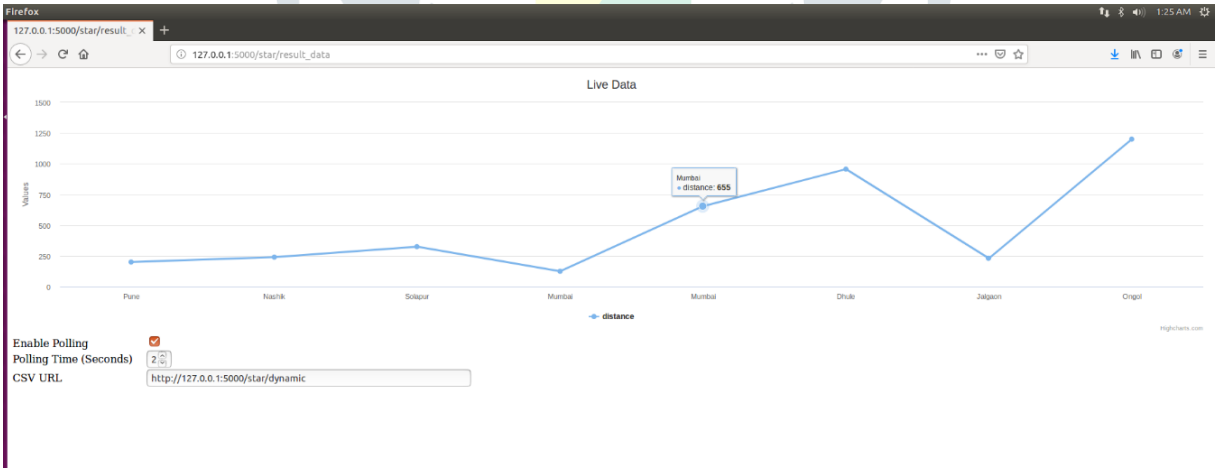


fig. 7: dynamic graph visualization

VI. Results and Analysis

We analyzed an existing traditional approach for Business Intelligence process and found the issues/limitations such as delayed responses from one application to another application, higher cost in terms of installation, deployment and maintenance of BI environment. Targeted users were facing problems when system environment is under maintenance or being updated. Service response also gets slow down, thus causes overall performance degradation. As part of study, we have analyzed the existing solution and have done comparative analysis based on factors like time, cost, performance, availability, and security.

table 1: comparative analysis of existing system and proposed system

PARAMETERS	EXISTING ARCHITECTURE	PROPOSED ARCHITECTURE
Development	Sequential Development approach	Parallel development approach
Agility	low	Very High
Scalability	Low	Very High
Availability	Low. Due to Single Point of Failure, whole system may fail.	Very High. It is Location Independent.
Integration	Complicated to add or remove a component.	Very easy to design, develop, and deploy a new service for an operation.
Reusability	Low	Very high
Platform Independent	No	Yes
Coupling	Tightly coupled	Loosely coupled

VII. Conclusion and Future Scope

From the above discussions, we can conclude that it is possible to implement Business Intelligence operations from stage of data collection to data visualization and from simplex level operation to complex & critical level operation over heterogeneous and huge datasets over Service Oriented Architecture paradigm. There has been a revolutionary change and adaption of various technologies for handling, processing and managing Big Data over a decade. Integrating various such tools and technologies to overcome challenges which occurs in traditional approach results in faster and efficient processing and management of data in business world; this SOA based architecture is proposed and have found assuring benefits resulting in becoming advantageous in several aspects.

Thus, this proposed architecture will play a vital role as a generic architecture in industries for adopting and integrating BI architecture in their environment. Also, it will be a good contribution for other researchers to study, implement and enhance the capabilities of this work.

This proposed work is a proof of concept (POC) for designed proposed architecture and functions efficiently over traditional Business Intelligence (BI) operations. Future scope of this study and work is to deploy this architecture containing BI related web-service over cloud network using Docker which is a container based engine that allows to develop, deploy and run multiple instances of an application requested by various different applications. It will enhance the chances of web-service availability and delivery in large business working environment. Also to design and implement more granular level web-services for small to small operations in an environment.

References

[1] Akash Ajay Borse, Siddhant Verma, Dr. Sasidhar Babu, Gardas Naresh Kumar. "Service Oriented Architecture paradigm for Business Intelligence: A survey." *International Journal of Advance Research, Ideas and Innovations in Technology* 5.1 (2019).

[2] Patil, Arun D., and N. D. Gangadhar. "OLaaS: OLAP as a Service." *2016 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*. IEEE, 2016.

[3] Shashwat, Anurag, and Deepak Kumar. "A service identification model for service oriented architecture." *Computational Intelligence & Communication Technology (CICT), 2017 3rd International Conference on*. IEEE, 2017.

[4] Papazoglou, Michael P., and Willem-Jan Van Den Heuvel. "Service-oriented design and development methodology." *International Journal of Web Engineering and Technology (IJWET)* 2.4 (2006): 412-442.

[5] Klose, Karsten, Ralf Knackstedt, and Daniel Beverungen. "Identification of Services-A Stakeholder-Based Approach to SOA Development and its Application in the Area of Production Planning." *ECIS*. Vol. 7. 2007.

[6] Maia, Italo. *Building Web Applications with Flask*. Packt Publishing Ltd, 2015.

[7] Wu, Rui. *Environment for Large Data Processing and Visualization Using MongoDB*. Diss. 2015.

[8] Somya, R., Manongga, D. and Pakereng, M.A.I., 2018, September. Service-Oriented Business Intelligence (SoBI) for Academic and Financial Data Integration in University. In *2018 International Seminar on Application for Technology of Information and Communication* (pp. 1-5). IEEE.

[9] Miškuf, M., Zolotová, I. and Nemčík, M., 2016, January. Application of business intelligence solutions from microsoft and IBM on manufacturing data. In *Applied Machine Intelligence and Informatics (SAMI), 2016 IEEE 14th International Symposium on* (pp. 41-45). IEEE.

[10] Miškuf, M. and Zolotová, I., 2015, January. Application of business intelligence solutions on manufacturing data. In *Applied Machine Intelligence and Informatics (SAMI), 2015 IEEE 13th International Symposium on* (pp. 193-197). IEEE.