



e-ISSN: 2319-8753 | p-ISSN: 2320-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 9, Issue 10, October 2020

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.512



9940 572 462



6381 907 438



ijirset@gmail.com



www.ijirset.com

Cost Saving Solutions for Business Intelligence Systems

Madhusudhan Reddy Sureddy ¹, Prathyusha Yallamula ²

Associate Director, Santander Bank, Holmdel, NJ, USA¹

Vice President, Natixis CIB Americas, New York, New York, USA²

ABSTRACT: Business intelligence eco systems with its diverse functionalities of data integration, data quality, master data management, metadata management, reporting and data visualization are very complicated to build and maintain. Every business intelligence functionality follow its own software development life cycle and has to flow through the stages of planning, defining, designing, developing, testing, deployment and maintenance. Maintenance costs are a recurring over head for the enterprises and these costs are very high for business intelligence systems in enterprises due to the usage of multiple tools and technologies. In this paper author provides multiple cost saving solutions using infrastructure optimization, process optimization and process automation techniques to reduce maintenance cost of the business intelligence systems.

KEYWORDS: Business Intelligence, Data warehousing, data, big data, cost saving, infrastructure, process, optimization, disaster recovery, maintenance, operations, ETL

I. INTRODUCTION

Business intelligence (BI) includes leveraging of various software and services to transform data into actionable insights that inform an organization’s strategic and tactical business decisions. Business intelligence eco systems with its diverse functionalities of data integration, data quality, master data management, metadata management, reporting and data visualization are very complicated to build and maintain. Every business intelligence functionality follow its own software development life cycle and has to flow through the stages of planning, defining, designing, developing, testing, deployment and maintenance. Cost of most stages in the life cycle are one time and can be reduced with a proper estimation of the effort involved and choosing cost reduction techniques likes choosing full time resources instead of contractors. Maintenance costs are a recurring over head for the enterprises and these costs are very high for business intelligence systems in enterprises due to the usage of multiple tools and technologies for the core functionalities. Below illustration provides the most common architecture followed by BI core functionality in enterprises.

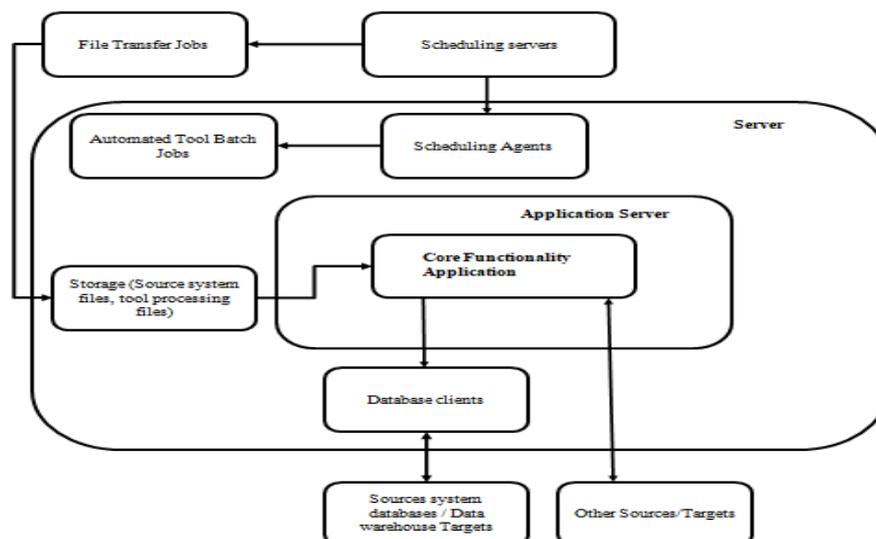


Fig. 1 Standard Business intelligence core functionality application architecture

As observed, Business intelligence application server has multiple layers and runs the below components:

- a) Application server like JBOSS, tomcat, web sphere etc.
- b) Core functionality applications
- c) Database clients to interact with Source system databases and data warehouse targets
- d) Scheduling agents to interact with scheduling servers
- e) Storage drives for source system files, tool processing files
- f) Automated tool batch jobs
- g) Automated file transfer jobs outside of the application servers to deliver source system files

Enterprises have to tread carefully when reducing maintenance costs as the quality of maintenance is directly proportional to the customer satisfaction index. Many business intelligence systems get decommissioned due to poor customer satisfaction as the users are quick to search for alternatives to do business intelligence. In this paper author provides multiple cost saving solutions to reduce maintenance cost of the business intelligence systems.

II. COST SAVING SOLUTIONS

Cost saving must aim at the elimination of wasteful elements in method of doing things. Cost saving solutions proposed in this paper can be classified into three types – **infrastructure optimization, process optimization, process automation**. Infrastructure optimization contains the solutions which can make best use of the infrastructure available in the enterprise and thereby bringing the infrastructure foot print which will reduce the cost. Process optimization involves the modifications to the existing processes which would lower the resource efforts and bring down the cost. Process automation includes all the automation techniques which can eliminate the manual efforts to reduce the cost.

Infrastructure optimization: There are three important solutions which can bring the infrastructure cost down.

- a) Application shared service platforms for enterprises
- b) One source files storage for enterprise
- c) Re-purposing application pre-production servers for disaster recovery

Application shared service platforms for enterprises:

Enterprises have multiple domains like finance, sales, risk, and payments etc. that have business intelligence requirements. Business intelligence systems comprise of the core functionalities - data integration, data quality, master data management, Meta data management, data mart build (using virtualization), reporting, data visualization and scheduling. Enterprises generally tend to utilize vendor product tools rather than building native tools to fulfill the core functionality requirements and each of the functionality is also on boarded to separate servers instead of a single server to get better performance. In many cases in enterprises each domain technology team builds their own platforms causing an operational nightmare. Observation is that none of the application in enterprises teams ever uses the full potential of the platforms and resources always go wasted.

In the proposed method of application shared service platform a high performance single platform for the core functionality is created and all the applications get a chunk of the same platform. Separate divisions are made on the core functionality tool and servers for each application and infrastructure (CPU, RAM) limitations are set for each application. Limitations can be tweaked and further scaling of server can be done based on the consolidated application requirements.

Formulae for the infrastructure and software cost of building independent business intelligence systems is as follows - $\{(Number\ of\ servers\ per\ core\ functionality) * (number\ of\ core\ functionalities) * (number\ of\ domains) * (server\ cost)\} + \{(number\ of\ domains) * (license\ cost\ for\ core\ functionality\ tool)\} + \{(Cost\ of\ operations\ resource)\}$. This formula is based on assumption that license cost and number of servers for all core functionality tools and all environments are same. Cost of infrastructure and software cost can be significantly reduced by going with the approach of shared business intelligence systems where all the domains share the servers for each of the core functionality. Formulae for the Infrastructure and software cost with the proposed approach is $\{(Number\ of\ servers\ per\ core\ functionality) * (number\ of\ core\ functionalities) * (server\ cost)\} + \{(license\ cost\ for\ core\ functionality\ tool)\} + \{(Cost\ of\ operations\ resource) * (percentage\ reduction\ in\ platforms)\}$. As observed in the new formulae there is a drastic reduction in the cost both from infrastructure and resourcing. Below illustration provides the before and after business intelligence systems depicted with two core functionalities (tools) and 4 domains (applications)

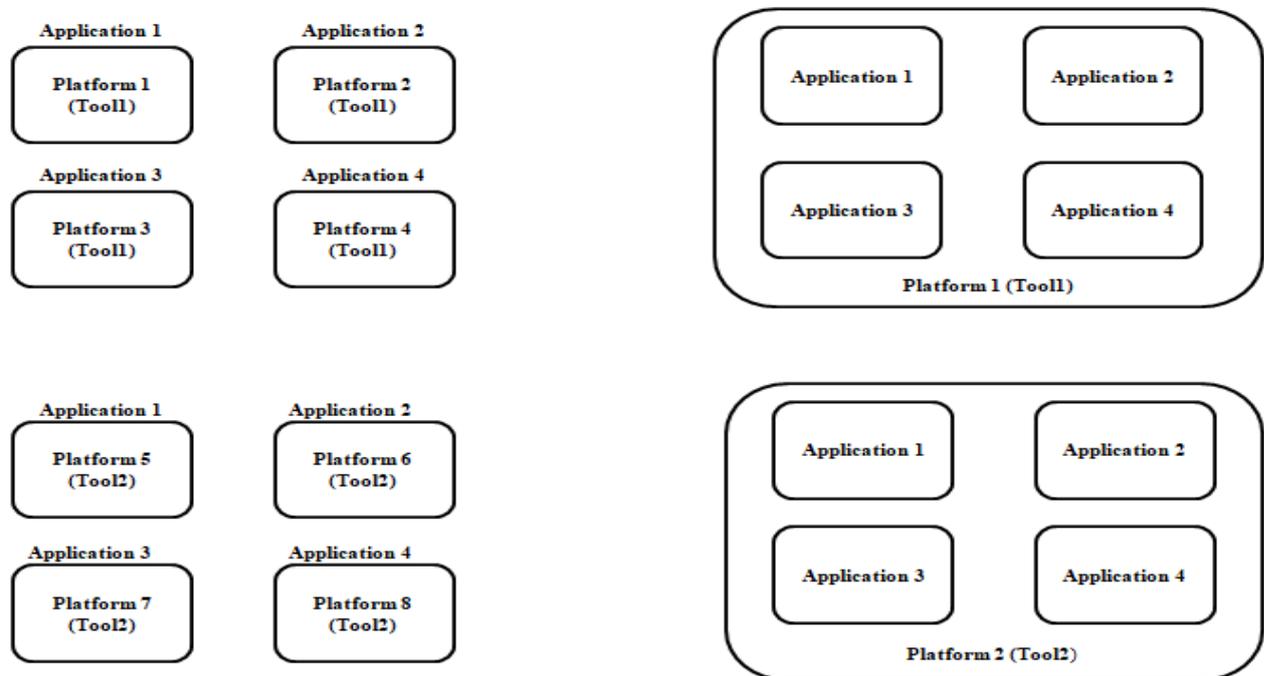


Fig. 2 Before (left) and After (right) application shared service platform architecture

One source files storage for enterprise:

Most critical aspect of any business intelligence application is the source system files. First step of any business intelligence application tool is to process these source system files – data integration tools load these source systems files into structured databases called as data warehouses, data quality tools run quality checks on these source systems files and so forth. In many cases in enterprises each domain technology application utilizes their own source file storage. This has been observed even in enterprises which have application shared service platforms. Considering the source systems available in an enterprise are constant, the same files are used by each domain application causing an infrastructure overhead.

Enterprises should build a “one source files storage” to eliminate this overhead and reduce costs. In the proposed approach all the source systems files are transferred to a high throughput storage space and every application and core functionality tool uses the same common storage for their source file needs. Throughput includes read Input/output operations per second (IOPS), read MB per second, write IOPS, write MB/sec. Proper metadata of the source system files is maintained to ensure no duplicate entry of the files are present. With this approach cost is reduced in the below areas:

- a) Reduced storage space needed.
- b) Reduced network usage as the FTP file transfers between source system servers and business intelligence platforms consume lot of network.
- c) Reduced FTP job creation. This reduces development effort and cost in building these FTP jobs.
- d) Reduced operational effort/cost as less FTP jobs need to be monitored.
- e) In some cases where mainframe source systems are used, heavy reduction of MIPS (millions of instructions per Second). This can provide significant reduction in cost as MIPS in mainframe are very expensive
- f) Reduced effort and cost in the core functionality development as the metadata information of the source files are identified only once.

Below illustration provides the before and after business intelligence systems depicted with two core functionalities (tools) and 3 domains (applications).

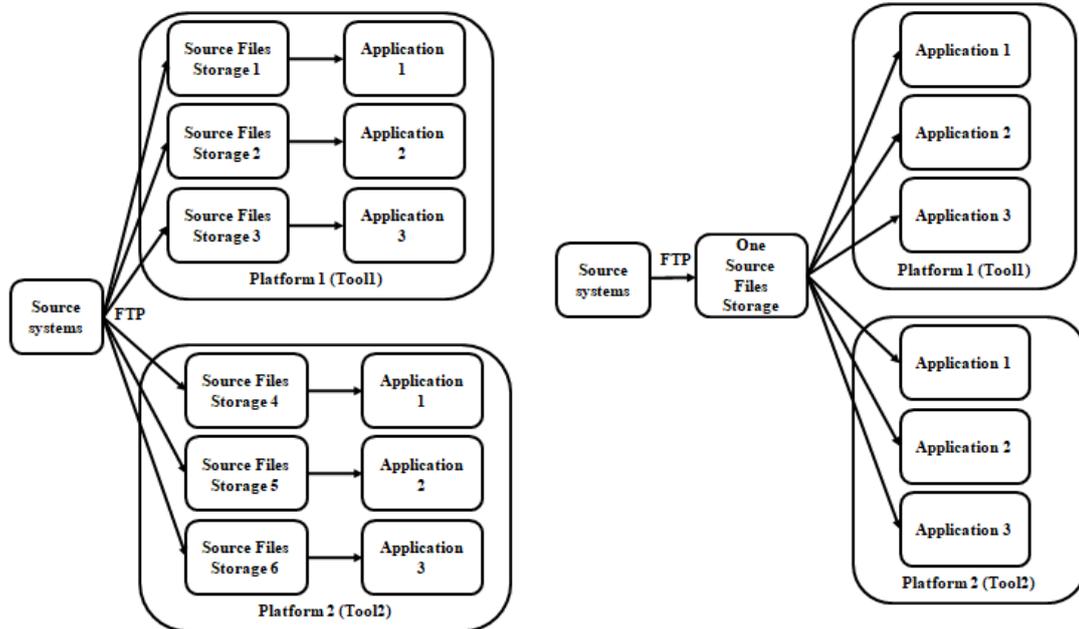


Fig. 3 Before (left) and After (right) one source file storage platform architecture

Re-purposing application pre-production servers for disaster recovery

Enterprises need three environments – Development, UAT and Production to support the Software development life cycle (SDLC) requirements of the Business intelligence systems and a fourth environment called as disaster recovery (DR) for supporting critical business functions during natural and manmade disasters. Disaster Recovery defines the set of policies, tools and procedures to enable the recovery or continuation of vital technology infrastructure and systems. Disaster recovery environments are always built in a different data center and network zone to ensure primary and the DR sites are completely independent. Though the importance of having a DR is not questionable, it’s a known fact that the DR environments in organizations stay unused for years long as the probability of a DR happening is extremely minute. These environments are only used once a year for conducting a yearly DR test to validate if the DR process actually works. DR environments have the exact same infrastructure requirements as the production environments and hence have a considerable infrastructure cost overhead.

In the proposed solution, UAT (pre-production) environments are re-purposed as disaster recovery environments during a disaster and a separate UAT environment is stood up if and when a disaster happens. Below illustration provides the before and after business intelligence systems. Proposed solution only repurposes the core functionality platforms only and keeps intact separate DR servers for the corresponding core functionality platform metadata databases or the data warehouse databases.

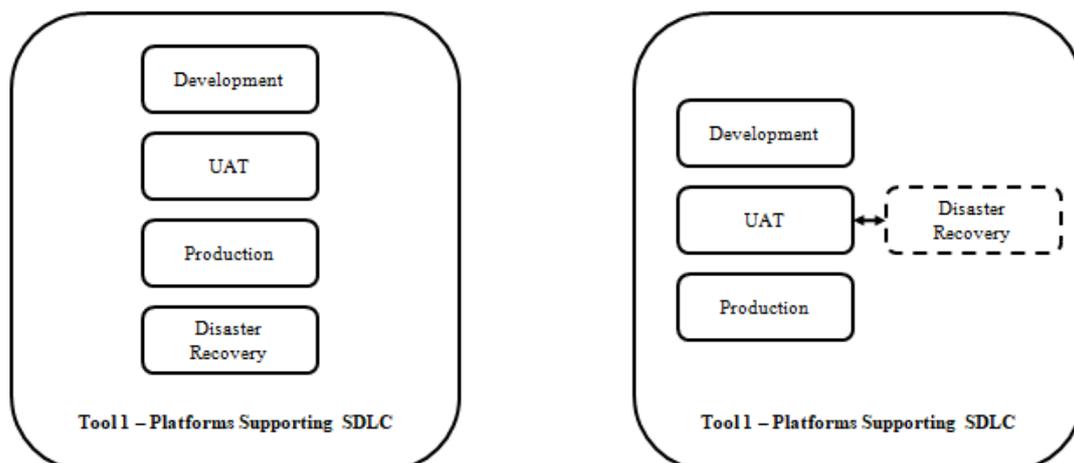


Fig. 4 Before (left) and After (right) re-purposed application pre-production server for Disaster recovery architecture

Business Intelligence system tools follow the below structure in the case of separate DR environment:

- a) Separate core functionality tool DR server is on boarded
- b) Core functionality tool installer code from vendors needs to be installed on the installer file systems on the DR tool server. Production install file system is replicated onto the DR install file systems with exact same name.
- c) Core functionality tools have a metadata database where they store the metadata needed for the tool to function. Production Metadata database is replicated to the DR Metadata using database replication techniques offered by the database systems.
- d) Domain functions (applications) have separate application file systems on the same server as the platform is shared. Production application file systems are replicated onto the DR application systems with exact same name.
- e) Users are also replicated from the Production to the DR server
- f) During DR a manual switch over is done. Metadata database (DR) domain name systems (DNS) and core functionality tool DR server DNS is switched to the one of the production servers

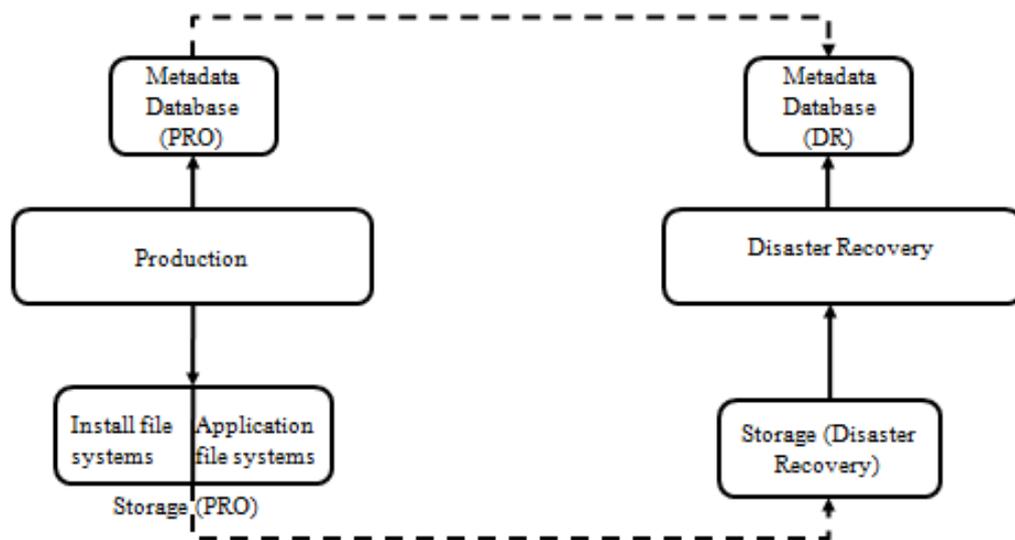


Fig. 5 Separate DR environment architecture

In the proposed solution the structure followed will be as below:

- a) No separate DR server is on boarded.
- b) Production installer file system code is replicated onto a storage file systems on UAT server. Replicated file system name is different on UAT server as UAT file systems use the production file system name.
- c) Production metadata database is replicated to the DR Metadata using database replication techniques offered by the database systems.
- d) Production application file systems are replicated onto the DR application systems. Replicated file system name is different on UAT server as UAT file systems use the production file system name.
- e) UAT and production users are made different. Users are not replicated and whenever a new user is created on the production servers, it needs to be created with exact set up on the UAT servers.
- f) During DR, few additional activities need to be done.
 - 1) Bring down the UAT file systems and rename the production replicated file systems to actual production file system names.
 - 2) Metadata database (DR) domain name systems (DNS) and core functionality tool DR server DNS is switched to the one of the production servers

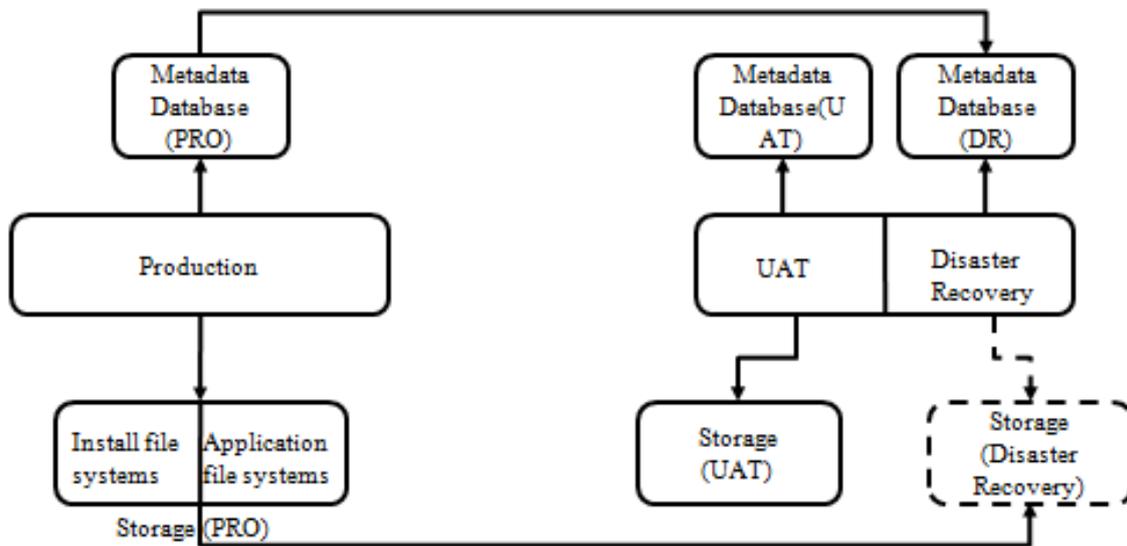


Fig. 6 Re-purposed application pre-production server for Disaster recovery environment architecture

Proposed approach reduces the cost in the below areas:

- a) Reduced DR server infrastructure cost.
- b) Reduced operational effort/cost as DR servers do not need to be monitored

Process Optimization: Process optimization involves modifying of the existing processes into a cost effective ones and still maintaining the same level of performance. There are two main solutions which can achieve this:

- a) Centralized operations team model
- b) Multi geography virtual team Operations model

Centralized operations team model

Operations of the business intelligence platforms have three major parts – Administration, run support, and build support. Administration involves the support needed to keep the business intelligence tools running with no issues. Run support involves the effort needed to run the automated application loads and meeting the Service level agreement (SLA). Build support includes the effort needed to fix the bugs which happen in the automated application loads and the enhancements needed to better the performance of the loads to meet the Service level agreement (SLA). One of the most common support model observed in enterprises is the decentralized support models. As illustrated in the below picture multiple support entities are needed to support the operations of various domains and tools. Formulae for the total cost associated in below fully decentralized support model illustration are as follows – $\{(total\ number\ of\ business\ intelligence\ system\ tools) * (number\ of\ resources\ for\ supporting\ each\ tool) * (cost\ of\ support\ resource) + (total\ number\ of\ business\ intelligence\ system\ tools) * (cost\ of\ manager)\} + \{(total\ number\ of\ business\ intelligence\ system\ tools) * (number\ of\ domains) * (number\ of\ resources\ for\ supporting\ each\ domain\ run\ support) * (cost\ of\ support\ resource) + (total\ number\ of\ business\ intelligence\ system\ tools) * (number\ of\ domains) * (cost\ of\ manager)\} + \{(total\ number\ of\ business\ intelligence\ system\ tools) * (number\ of\ domains) * (number\ of\ resources\ for\ supporting\ each\ domain\ build\ support) * (cost\ of\ support\ resource) + (total\ number\ of\ business\ intelligence\ system\ tools) * (number\ of\ domains) * (cost\ of\ manager)\}$. As observed in the formulae enterprises would spend millions of dollars per year in this approach as a minimum of one resource and one manager would be needed to support each support entity.

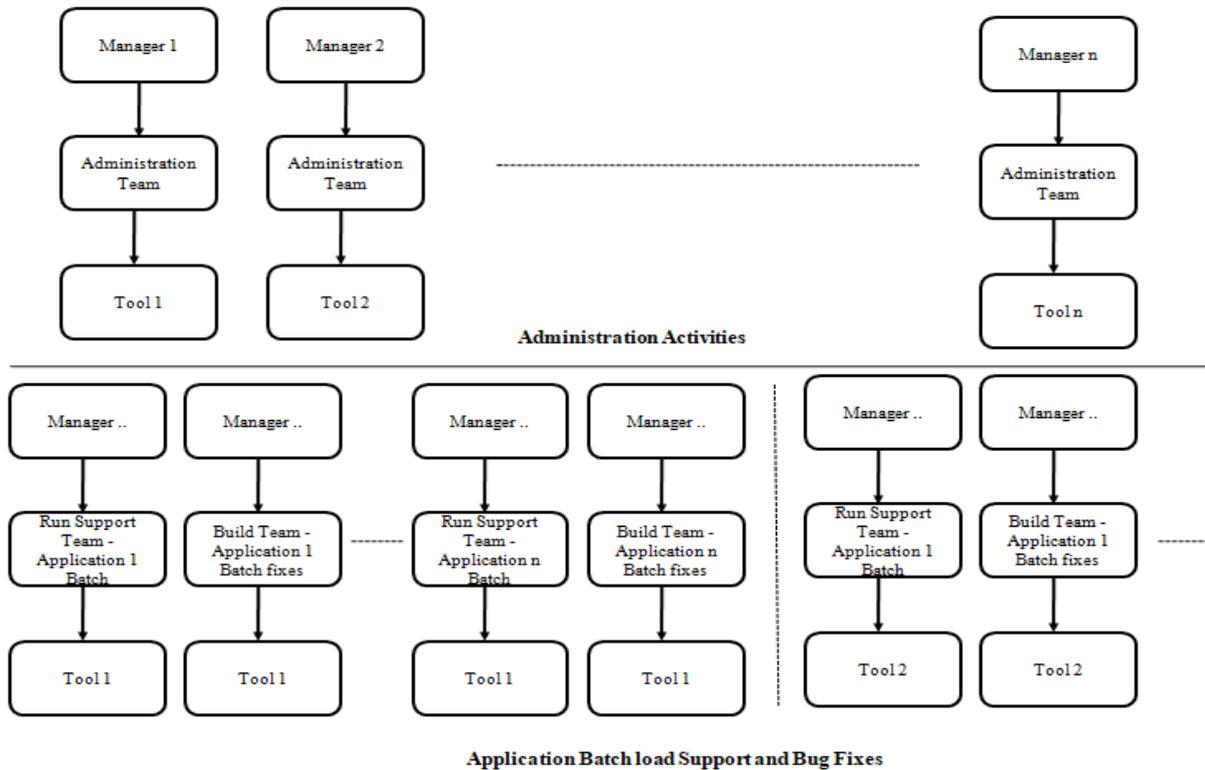


Fig. 7 De-centralized Business Intelligence Operations team model

In the proposed solution entire operations is centralized and there would be only two support entities – Administration and Run Support team and Build team. Two support entities model will ensure the audit requirements of compliance – “developer not having production access” is met and also eliminates the probability of manual errors as developers can never mistakenly run commands in production instead of their needed development, due to access unavailability. Below illustration provides the centralized support model and the formulae for the total cost associated in this model are as follows – $\{(number\ of\ resources\ for\ supporting\ administration\ and\ run\ support) * (cost\ of\ support\ resource) + (cost\ of\ manager)\} + \{(number\ of\ resources\ for\ supporting\ build) * (cost\ of\ build\ resource) + (cost\ of\ manager)\}$. As observed in the formulae there is a significant reduction in the cost. In addition to the explicitly seen cost reduction benefit there are also many additional advantages with this model.

- a) Cross technology development of the resources
- b) Easy to manage shift rotations to provide 24 / 7 support as the team size are definitely 2 or more.

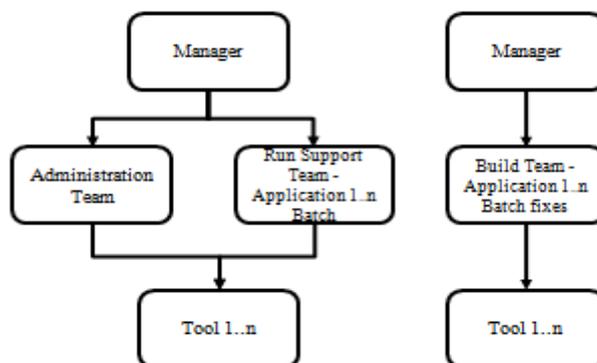


Fig. 8 Centralized Business Intelligence Operations team model

Multi geography virtual team Operations model

One of the major reasons of attrition in business intelligence operational Support teams is the requirement of the team members to be available physically after hours – 6 PM EST to 9 AM EST. As against Online transactional processing systems which need to be available during business hours 9 AM EST to 6 PM EST, most of the business intelligence systems are expected to fully functional during the non business hours 6 PM EST to 9 AM EST. This reverse requirement is due to the fact that business intelligence systems expect the data as of the previous business day by 9 AM EST of the next business day for doing their reporting. To meet this data SLA business intelligence systems run the automated application data and reporting loads overnight. In addition to the non business hours, there are also few tools like reporting, data visualization, reference data management and master management which need to be available during business hours as the users utilize them to do reporting for higher management. Below illustration provides the various operational team activities and the critical and non critical tasks during business and non business hours. As observed in the illustration there is a 24 / 7 physical support requirement for a business intelligence system and there is no escaping of the physically availability of the resource in the night shift. Single geography model has severe impact on both the health and personal life of the resources which eventually causes the resources to either under perform in their work or leave the enterprise.

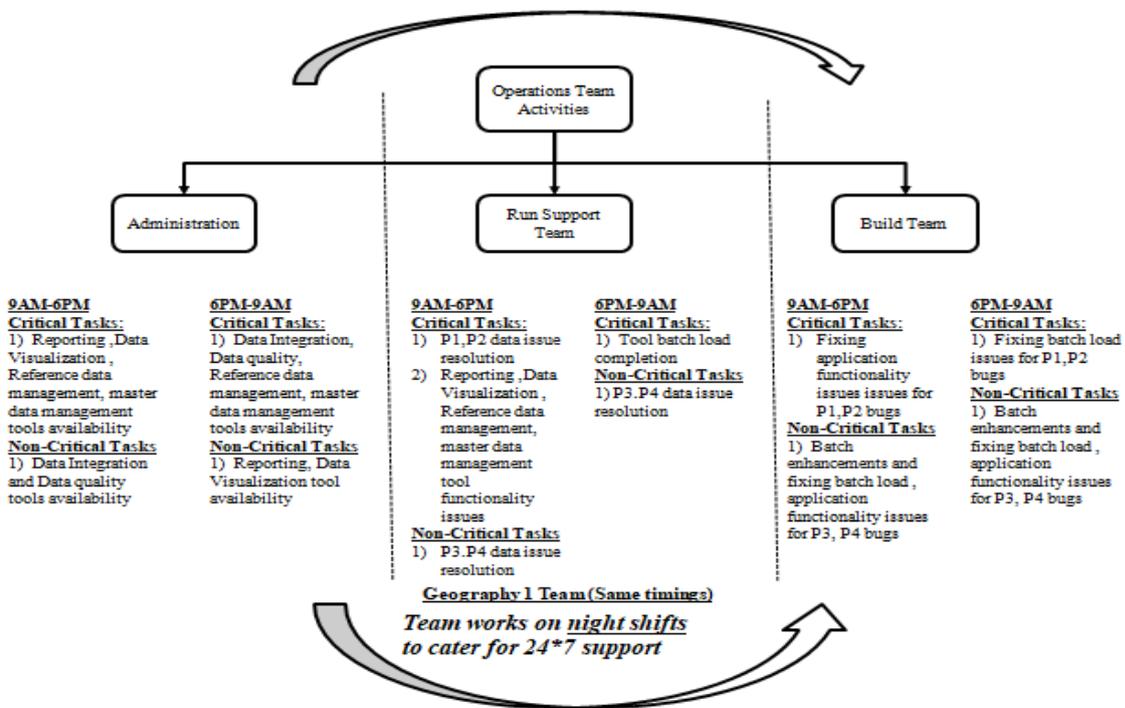


Fig. 9 Single geography Operations team model

In the proposed solution, business intelligence operations team works from two geographies. Geography1 team would work during the same hours as the business counterparts and the Geography2 team would work during the non business hours. Geography2 team would be a virtual team as they would be located in an opposite time zone. Entire tasks would be divided amongst both the geography teams and ensure there is a 24 / 7 support. Below illustration provides the various operations team activities divided across the entire day and amongst both the geographies. As observed with a good hand over and take over during the switch of teams between geographies the same or better level of customer satisfaction can be provided. This novel approach would provide multiple benefits:

- a) Significant increase in team member satisfaction as there will be no night shifts. This also provides an indirect cost benefit with no resource time wastage for knowledge transfers during attrition on boarding and better performance of resources.
- b) Significant reduction in cost can be achieved by wisely choosing the Geography 2 country, were the cost of resource is lower than the business originating country.
- c) Increased customer satisfaction can be achieved by handing over the work wisely between geography 1 and 2 teams. This will ensure the business user issues can be solved faster.

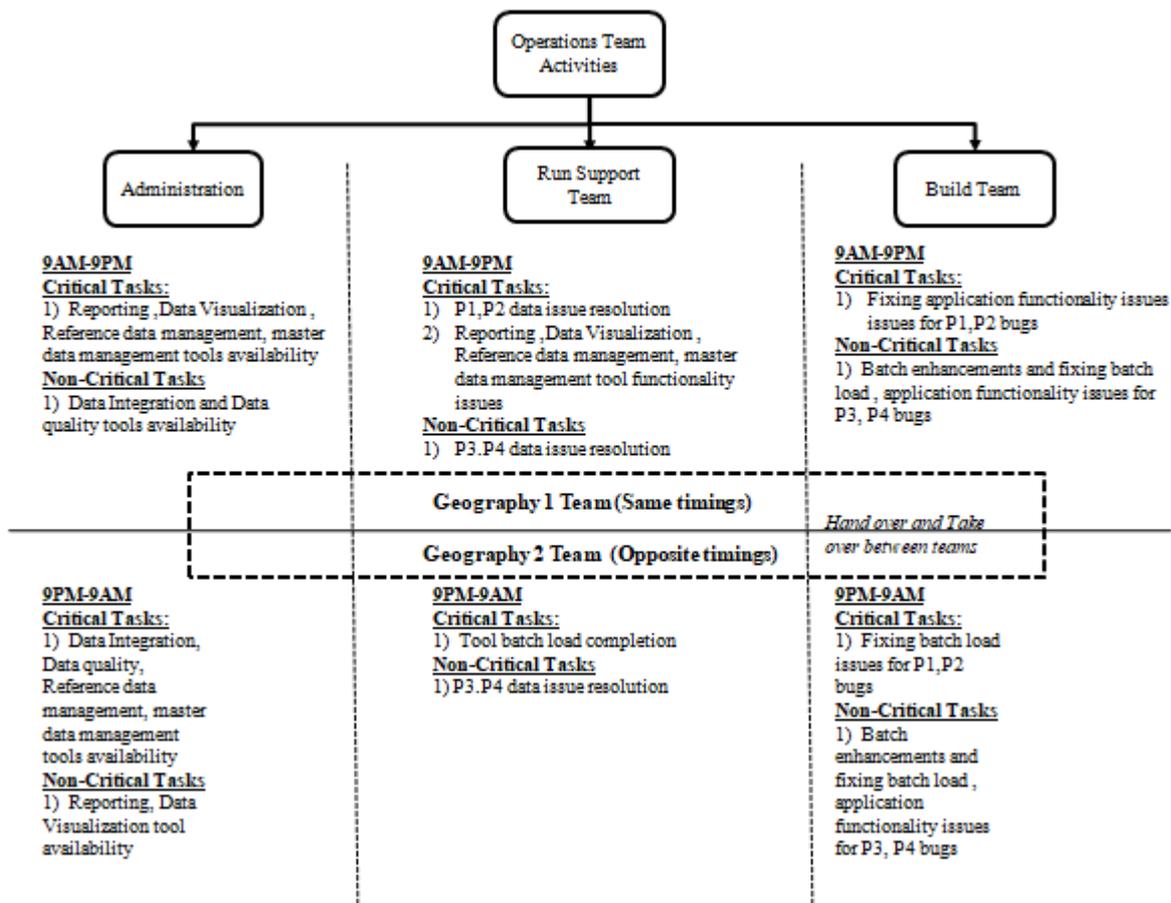


Fig. 10 Multi geography virtual team Operations model

Process Automation:

Process automation involves the automation of various recurring operation processes which require manual handling. There are two main solutions which can achieve this:

- a) Business intelligence tools maintenance automation
- b) Business intelligence tools upgrade automation

Business Intelligence tools maintenance automation

Business intelligence tool eco system requires multiple vendor products tools to handle the build of various core functionalities. Current market leaders in vendor products for Data Integration, Data Quality, Metadata Management, Master data Management functionalities include Informatica, Talend, SAS, ERWIN etc, reporting functionalities include Tableau, Business Objects, OBIEE etc. and Scheduling functionalities include Control M, Autosys, APPWORX etc. Operations teams generally follow a “no questions” asked approach and follow the published guidelines in the run guide. Operations team in enterprises tends to many manual activities which can easily be automated. These manual activities increase the number of resources needed in the operations team.

Proposed solution involves the automation of the recurring administrative maintenance activities needed in the business intelligence tools through scripting language. Vendor product tools generally contain command line interfaces (CLI) which allow the automation of most of the administrative tasks. One of the most common activities done by administrative team which can be automated is the creation of tool users and groups. This can be achieved by integrating the information technology service management (ITSM) systems like service now with the command line interfaces of the tool. ITSM tools provide Rest API calls which can help in checking for tickets created for user creation. Automation script can be built to check for the approved tickets and then call the tool CLI’s with the ticket variables and create the users. Post creation of the users, script can also close the tickets after placing passwords in the ticket. Similar approach can be followed to automate the additional manual tasks in the operations in enterprises.

Business intelligence tools upgrade automation

Business intelligence vendor product tools on a yearly basis release newer version of their products to either handle the bugs of the previous version or keep up to date with the newer trends in technology. Downside of these newer vendor releases is the removal of support of older versions and pushing enterprises to upgrade to the later versions. Enterprises generally initiate a new project whenever vendor product tools need to be upgraded due to the amount of manual effort required and complexity involved, rather than utilizing existing operational resources.

Proposed solution involves the automation of the business intelligence tools upgrade through scripting language. Vendor product tools upgrade core framework always remains constant and the tools always come packaged with a silent install capability for enterprises to automate them. Core framework of any business intelligence tool upgrade involves the below steps:

- a) Backing up of the current installation and metadata
- b) Copying of the newer version compressed software to the servers
- c) Uncompress the newer version software
- d) Configure the configuration file for installation
- e) Run the silent upgrade of the tool
- f) Upgrade metadata of the tool

Scripting of the above solutions can be done in any programming language compatible with the operating system of the enterprise server – Linux or Windows. Preference from the author would be accomplish these in python due to its extensive package availability. Detailed documentation should be published for the automation scripts to allow operations team to handle minor modifications, which might be needed based on the vendor newer version documentation. These automations provide significant cost reduction by eliminating manual tasks to be executed in an upgrade and removing the need of a project for upgrade. Upgrade of tools becomes a part of the routine maintenance activities of the operations team.

III. CONCLUSION

In today's volatile economy enterprises are in a constant search of improving their bottom line. Business intelligence helps enterprises in new market expansion and reducing budget allocated to its projects could have serious impact on profits. There is a need to have cost saving solutions which can reduce the business intelligence system project budget without impacting the quality of products or services rendered. Reduced cost of projects can help enterprises to refocus the budget into newer technology growth. In this paper author provides multiple cost saving solutions using infrastructure optimization, process optimization and process automation techniques to reduce maintenance cost of the business intelligence systems.

REFERENCES

- [1] Madhusudhan Reddy Sureddy, Prathyusha Yallamula (2020), DATA QUALITY ARCHITECTURE FOR DATA WAREHOUSES, International Journal of Research Cultural Society , 4(6), pp 95-100
- [2] Madhusudhan Reddy Sureddy, Prathyusha Yallamula (2020), A Framework for Monitoring Data Warehousing Applications, International Research Journal of Engineering and Technology, 7(6), pp 7023-7029
- [3] Madhusudhan Reddy Sureddy, Prathyusha Yallamula (2020). Approach to help choose right data warehousing tool for an enterprise. International Journal of Advance Research, Ideas and Innovations in Technology, 6(4) www.IJARIT.com.
- [4] Madhusudhan Reddy Sureddy, Prathyusha Yallamula (2020), DEBUGGING METHODOLOGY FOR ISSUES IN DATA WAREHOUSING AND BUSINESS INTELLIGENCE PLATFORMS, Proceedings of IFERP International conference, Luxor, Egypt, August 10-11, 2020
- [5] Madhusudhan Reddy Sureddy, Prathyusha Yallamula (2020). A BIG DATA PLATFORM ARCHITECTURE FOR ENTERPRISE. International Journal for Innovative Research In Multidisciplinary Field, 6(8), pp 197-205
- [6] <https://www.computerworld.com/article/2522806/cost-saving-for-it-infrastructure-management.html>
- [7] https://en.wikipedia.org/wiki/Disaster_recovery
- [8] <https://www.cio.com/article/2439504/business-intelligence-definition-and-solutions.html>



INNO SPACE
SJIF Scientific Journal Impact Factor

**Impact Factor:
7.512**

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details