Design and Application of Data Centric Systems in Large Scale Computing and Complex Analytics

CIS – 661, Advanced Computer Architecture Term Paper

Sachin Ramesh
SUID:384471510
MS, Computer Science, Fall 2016
Department of Electrical Engineering and Computer Science
Syracuse University

Table of Contents

1. Abstract:

The paper discusses about the need of Data Centric Systems (DCS) to handle computations and analytics that are evolving in complexity and dependency day by day. It provides insight about the various designs and applications that have been proposed and being employed that uses data centric approach in solving as the solution. It explains how data explosion could use data centric systems as an architectural solution for high performance computing and High Performance Analytics. It speaks about the Life cycle models of data centric systems and how the DCS architecture is transformed for performance optimization of big data systems. It gives the idea about the transition to DCS architecture and its application in the domains such as cloud computing, data mining, HPC Analytics, Interactive visualization of Large Data sets, etc. It also discusses the result from DCS application analysis. It mentions about the design patterns that can be integrated with data centric systems. It also mentions current challenges in adaptation of this architecture. It concludes how data would be a significant natural resource in near future and DCS approach would be an optimized solution for tackling the computational and Analytical problems. By careful administration and maintenance, these visionary technology trends optimizes CPU performance, memory, storage, and power, which is the major challenge considered while designing any architectural solution.

2. Introduction and Background:

2.1 What is Data Centric Design?

As the name suggests, Data Centric Systems (DCS) are the computer systems that have design architecture in which Databases play a major role, in other words these DCS have architectures which have combined functionalities specific to data/databases i.e, application behavior is encapsulated by data. The combined functionality may include many application specific designs such as[24], a relational database management system with customized memory, data structures and operating system, Systems using stored procedures that run on database servers, Usage of shared database systems as communication medium between parallel processes in distributed computing etc. This design leads to a set of bound activities defining how data is built, manipulated and how it is accessed. The main application of data centric systems will lie in following domains [10]:

- ✓ Business analytics/intelligence
- ✓ Social analytics

- ✓ Financial Analytics
- ✓ Life Sciences
- ✓ Technical Computing (Engineering Design, Optimization, Prototyping analysis.
- ✓ Oil and Gas (Imaging and Interpretation)
- ✓ Climate and Environment (Prediction)

2.2 What is the necessity of Data Centric Systems/Architecture?

Seemingly in most of the data available for computing in large real time environments is unstructured. With the traditional storage, computational and analytics infrastructures, unstructured data is scattered in essentially different kind of applications and platforms causing increased costs and lower efficiency. Employing a data-centric approach to manage data is a cost effective optimal strategy for the business. Cluster of systems forming application, information and storage layers are integrated to form one Single system centered with Data. Data Centric systems provides cost effectiveness, data intensive integrity, better computational capability and security, and helps to understand and calibrate the values from meta data by translating the usable data from the available bulk of data, with time efficiency and providing greater opportunities for the organization to move towards higher revenue. This approach of administering raw data reduces the investments on infrastructures and the design provides better business solutions.

Below mentioned are few of the crucial factors that motivated for the data centric approach.

2.2.1 Data Explosion:

[2]1 trillion connected objects and devices on the planet are generating data by 2015. 2.5 billion gigabytes of data generated every day. Data is increasing rapidly than expected. It is growing significantly faster than the Moore's law. Evidently, online data indexed by Google is estimated to have increased from 5 exabytes (one exabyte = 1 million trillion bytes) in 2002 to 280 exabytes in 2009 which is 56-fold increase in seven years. This data growth is not limited to the Internet alone, but is consistent across all markets. In the enterprise space, the size of the largest data warehouse has been increasing at a cumulative annual growth rate of 173 percent—again, significantly more than Moore's law. This calls for the development of architecture specific to data handling.

2.2.2 Data Security:

Growing data becomes vulnerable. Without proper care it would result in financial loss of an organization. It is very important to protect the data and information to adhere compliance and regulatory requirements, which increases productivity and expectations. Data if neglected can be misused by attackers, which on manipulation may result in the loss of integrity. Data Centric Architecture would provide following advantages in data security. [27]

- Protects Sensitive data
- Enables analytics on protected data
- Protects data in motion and rest in the use
- Neutralizes the value of data to cyber attackers
- Delivers regulatory compliance and risk reductions
- Compliance requirements can be easily implemented using stored procedures

2.2.3 Data Computation and Analytics

Data centric technology will help to leverage computation and analytics by integrating already existing software, tools and application around data.It helps the data handling personal to improve the efficiency of knowledge gathering, information processing, interpreting and representing. Data centric Computation reduces incurred cost in tradition computations. Computation and analytics constrained with time can be easily managed with this architecture. As all the systems will be clustered together around data operations will be easier with minimal latency as the movement of data will be easier.

[10] Applications such as Human heart/brain simulation, High resolution models for chemical activity simulation in chemical industries; accurate weather prediction , Machine learning, Cognitive computing etc. would require computation process to be more than 100 times than the existing technologies, this can be achieved by moving the architecture to be data centered with lower energy consumption and better cost efficiency.

2.2.4 Data Synchronization:

With the huge growth in complexity of manageable data, Global Data synchronization is made easy with DCS, as it syncs master data with the all relevant systems through its efficient distributed computing. The goal is not only to synchronize data across systems but to improve data quality and to deliver as a service accurate, consistent data to transactional and operational systems.

3. Design and application:

There is no fixed design or single application approach for data centric systems. Data Centric systems are designed based on specific requirements targeting explicit applications. This property of DCS gives the flexibility for computer scientists to make use of the approach in various domains depending on the underlying infrastructures and involved environments. This section aims to explore few of the System designs and applications that make use of data-centric architecture based on necessity and demands.

3.1 Data centric systems Life cycles:

[14]The major function on data includes creation, publishing, exportation, importation, usage, transformation storage and re-usage by variety of users/organizations/applications for different purposes. These functions together form a life cycle of the data. Understanding the life cycle of the data helps us to understand nature of the data to use the data efficiently in different platforms. Integration, operation and application of data are made easy by understanding the life cycles.

Following are the list of few of the basic life cycle models of data centric domain.

3.1.1 Life cycles for multimedia

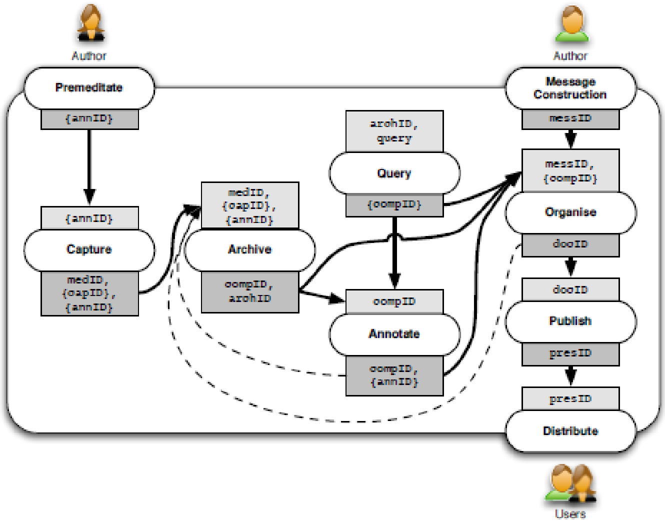


Fig. 1. Processes in media production, from Hardman

The above canonical model describes the process obtaining metadata and its integration and processing. Following are the nine processes involved that explains the integration process:

(i) Premeditate: Process that actually involved with the creation of multimedia message.

(ii) Capture : Physical acquisition of the involved media

(iii) Archive : storing and indexing a media asset indicating its details

(iv) Annotate : adding arbitrary information to a particular media asset

(v) Query : retrieving a media asset from an archive

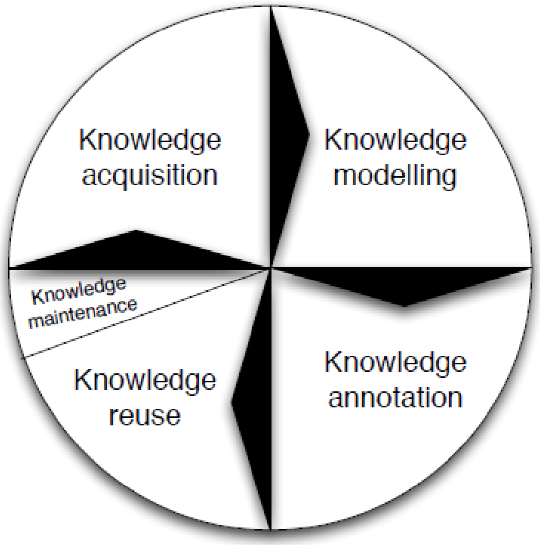
(vi) Message construction: Explaining the meaning of the data that is currently being processed

(vii) Organize : composing a set of media assets to a larger document,

(viii) Publish : converting a document into a format ready for external use

(ix) Distribute : making a media document available to external users

3.1.2 Life cycles in e-learning



Metadata lifecycles for learning objects, from Millard et al

The goal of the life cycle is to acquire required data translate it to understandable structure and to provide the analysis result for users who are seeking to acquire the knowledge of any specific domain. This life cycle has 4 major processes explained below:

(i) Knowledge Acquisition:

This involves fetching sufficient information on the topic by with the help of experts. At the end of this stage raw data will be available which contains random information about the required topic

(ii) Knowledge Modeling:

This process involves in transforming the gathered data into formal structural data and encodes the knowledge about the required domain with reasoning rules supporting further processing of required knowledge

(iii) Knowledge Annotation:

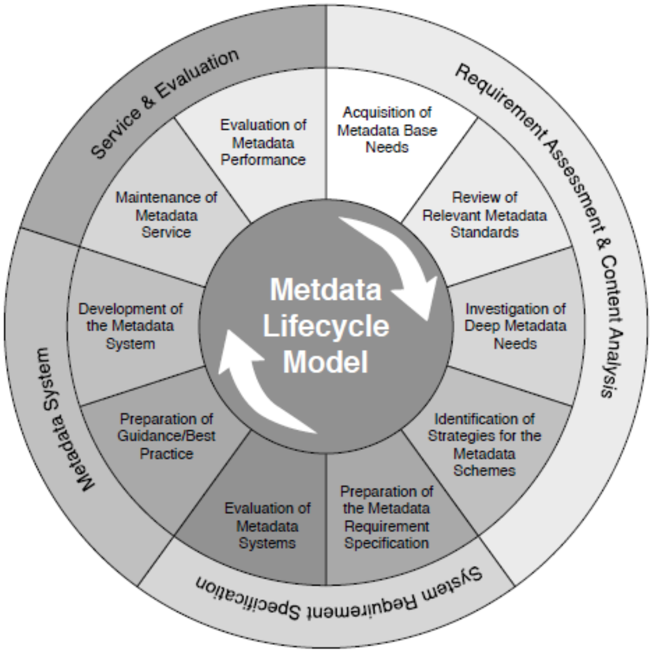
This process indexes the output of knowledge modeling process filtering the result in specific groups.

(iv) Knowledge Reuse:

The grouped knowledge in annotation phase is saved and it will be used as a reference when particular query is made.

knowledge maintenance step in between reuse and acquisition simply means that the processed knowledge on specific domain is maintained in case if required in the future.

3.1.3 Lifecycles in digital libraries


Service & Evaluation
Requirement Assessment & Content Analysis
System Requirement Specification
Metadata System
Acquisition of Metadata Base Needs
Review of Relevant Metadata Standards
Investigation of Deep Metadata Needs
Identification of Strategies for the Metadata Schemes
Preparation of the Metadata Requirement Specification
Evaluation of Metadata Systems
Preparation of Guidance/Best Practice
Development of the Metadata System
Maintenance of Metadata Service
Evaluation of Metadata Performance
Metdata Lifecycle Model

Metadata lifecycle for digital libraries, from Chen et al

This Metadata life cycle for digital library describes the process of acquiring, analyzing and representing digital artifacts in four major categories, these categories are further divided into steps and processes as shown in the above figure.

(i) Requirement Assessment and Content Analysis
This category involves the processes for requirement gathering and analysis of the task at hand. It involves scheduling of functions and scopes for knowledge gathering translate them into formal structures and review the candidate metadata standards.

(ii) System Requirement Specification
This phase produces the detailed specification document with clear indication of metadata requirements and evaluation of the requirements against existing metadata softwares and tools, finding the possibility of developing the system from the beginning.

(iii) Metadata System and Service:
This stage creates the guidelines for applying the specifications prepared in the previous steps. It also involves development of the system that was suggested in the previous step.
Service steps involves setting up of environment for smooth data workflow.

(iv) Evaluation:
It provides feedback that backtracks different possibilities, searching for optimal and more effective methods possibility.

3.1.4 Life cycle in data bases

Databases belong to technology domain rather than application domain.

The life cycle can be generalized in this case. Evidently, Databases are the domains in which data lifecycle has higher importance.

The lifecycle is simple and is termed as CRUD operations. CRUD is an acronym for create, read, update and delete. These are the four fundamental and atomic operations on any databases.

These operations are merely individual low level processes than a phase, in a sequential life cycle. These basic processes on data can be easily mapped to any of the life cycles because data has to be created first to operate on it and subsequently read, update and delete operations happens while modeling.

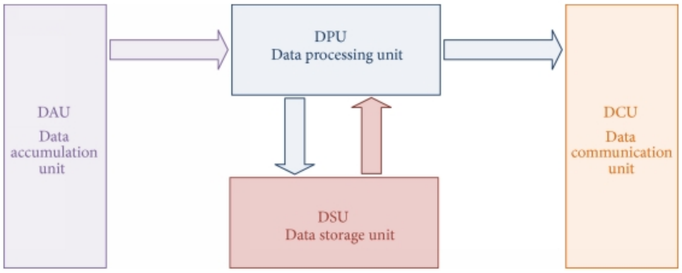
3.2 Data-Centric Knowledge Discovery Strategy for a Safety-Critical Sensor Application

[18]This particular application imparts how Data Centric approach can be used in sensor networks for high level computations.

Background:

In any indoor safety critical application, critical activities will be performed by sensors and actuators (which are clustered together) in a given time constraints. Since the application is very critical there will be large amount of data collected for processing and conclusion. Even though cluster based data collection approach has proved to be more efficient, the authors have posed a problem that the data collected in the data-centric sensor networks(DSN) will have 3 -60% chances of being unstructured and ambiguous, due to this reason they have also proposed a data-centric knowledge discovery strategy (DKDS) which would help in actual interpretation of most of the usable data along with the unstructured, ambiguous data thus producing maximum efficiency in a data-centric sensor networks.

Below figure demonstrates how a typical DSN alignment would appear to be;



Motivation:

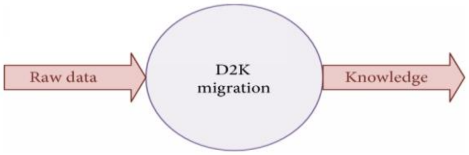
The main motivation behind developing DKDS is the risk factor involved in the DSN. In the clustered sensors and actuators, the actuators need to have prior knowledge to process and interpret the raw data supplied by the sensor, which otherwise may lead to improper interpretation and may impose severe risk on society. For example: Pressure sensor in aircrafts is considered to be the safety critical sensor application, misinterpretation of air pressure may lead in faulty operation risking the lives of the passengers.

Strategy:

It is very important to choose appropriate data collection strategies for the maximum efficiency of DKDS. Evidently, the collected data is the mainstay for knowledge discovery.

The proposed Cluster-level data-to-knowledge migration strategy (D2K) collects the data from the cluster heads and it sends the interpreted knowledge to the actuators, considering the overall time constraints. The DKDS would impart the appropriate knowledge database that would store the events and actions performed on those events for future reference.

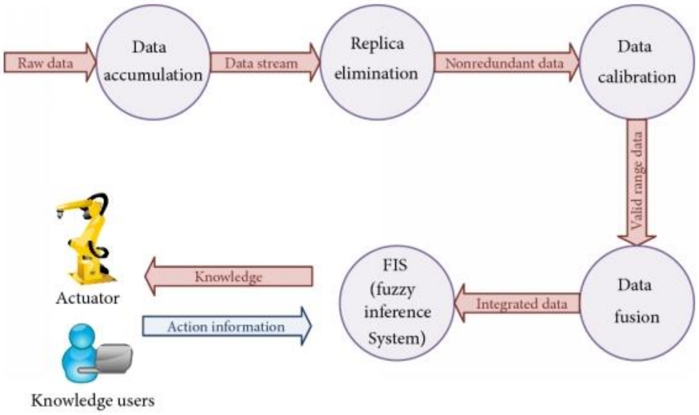
Below figure describes the black-box process of D2K.



D2K migration label-0 architecture

It implies the process of accepting raw data as the input from the sensors and gathering knowledge as the desired output. This can also be termed as the context model for the proposed strategy.

Below figure describes the five main functions that are performed by the processing unit to obtain the data , gather the intended knowledge validate it and send to the actuators and other knowledge users.



D2K migration label-1 architecture

Step 1. The data accumulator collects raw sensed data from the sensors and passes the raw data to the replica eliminator.

Step 2. The replica eliminator removes the redundant data and immediately passes them to the data calibrator.

Step 3. The data calibrator only filters the useful data based on a valid range and passes them to the data fuser to produce integrated data.

Step 4. The data fuser produces the integrated database using the fusion operation and passes the integrated database to the fuzzy controller.

Step 5. The fuzzy controller infers the knowledge from the integrated database using a FIS (fuzzy inference system) and ensures that the inferred knowledge can be well perceived by the actuators to perform the intelligent actions. The configuration of a typical fuzzy controller depends on the application requirement of the deploying environment.

This application of Data-Centric Knowledge Discovery strategy has been verified for its usefulness in real time application. In conclusion, the proposed work of the authors is tested in real time environment for correctness and its evident that knowledge cropped form the strategy is efficiently used by the actuators to decrease the risk factors in DSN.-

3.3 Data – Centric Supercomputing of Big Data using IBM computing technology

Motivation for the design:

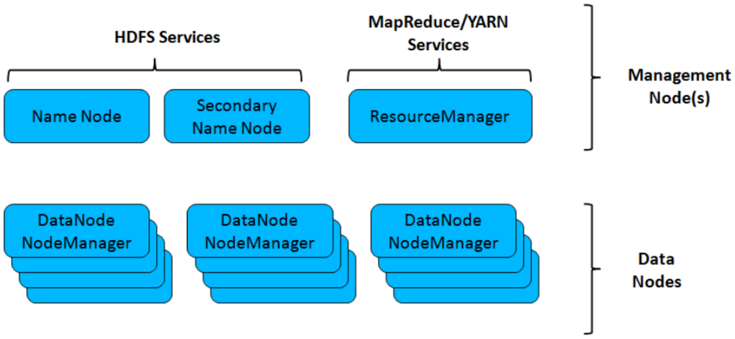
- Data Explosion
- Accelerating discovery and innovation
- Necessity to move unstructured raw data towards making decisions using full contextual analytics
- Evolving requirements for high performance analytics and computing
- Optimizations for power efficiency
- Limitations of system and chip design due to fast growing data which calls for the innovation of new system architecture approach where data has to be transported to central processing unit.

- Requirements for a flexible computer architecture which has the capacity to address the workloads of the growing data
- Current systems are not efficient in analyzing the raw data

Data Centric Design Strategy as a Solution

[19] Hadoop Distributed file systems(HDFS) and MapReduce engine are the two clusters that supports computing of Large data sets. The proposed architecture contains these systems by default using IBM open platform with Hadoop and Spark. Along with this features IBM spectrum scale software and IBM symphony software supports extended HDFS File system and MapReduce

The below shown diagram indicates the basic component model of the system:



This solution has four server roles:

1) System management node

Primary Monitoring and provisioning node for the Cluster, the system management node is pre-installed with IBM Spectrum Cluster Foundation software. This node can also have addon softwares required for further setup.

[19]The default software list is mentioned below

a. Red Hat or Linux operating system
b. RHEL operating system repository including all device drivers
c. IBM Spectrum Cluster Foundation cluster management software
d. IBM Spectrum Cluster Foundation web interface software

2) Hadoop management node

[19]These nodes encompass daemons that are related to managing the cluster and coordinating the distributed environment

The software that is to be installed on Hadoop management nodes includes:

a. Operating system (OS): Red Hat® Enterprise Linux® (RHEL) Server

b. Hadoop and Spark runtime environment: IBM Open Platform (IOP) for Hadoop and Spark, IBM Platform™ Symphony and IBM Spectrum Scale

After the cluster is deployed, these nodes provide the following functions:

i. A MapReduce engine for workload management, provided by IOP or IBM Platform™ Symphony.

ii. A web interface that enables users to access the cluster and run their applications, which are provided by IOP or IBM Platform™ Symphony

3) Hadoop data node

[19]These nodes include daemons that are functioned to storing data and finishing the task within the distributed environment .Data nodes run the HDFS DataNode service and the YARN NodeManager. They are configured for data intensive Hadoop applications. Data nodes have some local disks for the OS, the application, and runtime libraries. They have many more local disks for the HDFS or IBM Spectrum Scale

4) Spark worker node

Like Data Node except for the hardware and software configuration Spark worker nodes runs HDFS and the YARN NodeManager. They are configured for memory-intensive Spark applications. Spark worker nodes have some local disks for the OS, the application, and runtime libraries. They have many more local disks for the HDFS or IBM Spectrum Scale. Optionally, additional local SSDs can be used as cache spaces for Spark tasks MapReduce is the distributed-computing and high-throughput data-access framework through which Hadoop understands jobs and assigns work to servers within the Hadoop cluster. In this solution, MapReduce services are provided by IBM Open Platform with Hadoop and Spark, by default. IBM Platform Symphony provides a fully-compatible MapReduce API. However, it also enhances MapReduce through lower latency communication and more sophisticated resource management than standard MapReduce. MapReduce YARN has two associated daemons,

Daemon Description

i. ResourceManager (RM): Runs on a Hadoop management node and is responsible for submitting, tracking, and managing MapReduce jobs.

ii. NodeManager (NM) : Acts on stored data and runs the necessary computations and completes the task in MapReduce, and typically runs on all nodes.

[19]Network connections

There are three distinct networks included in the architecture of this solution, each serving a specific role:

(1) Service network: This network is connected to the cluster on each IBM Power Systems server and switches management ports. Usually provides the server management software to administer hardware in an Ethernet connection.

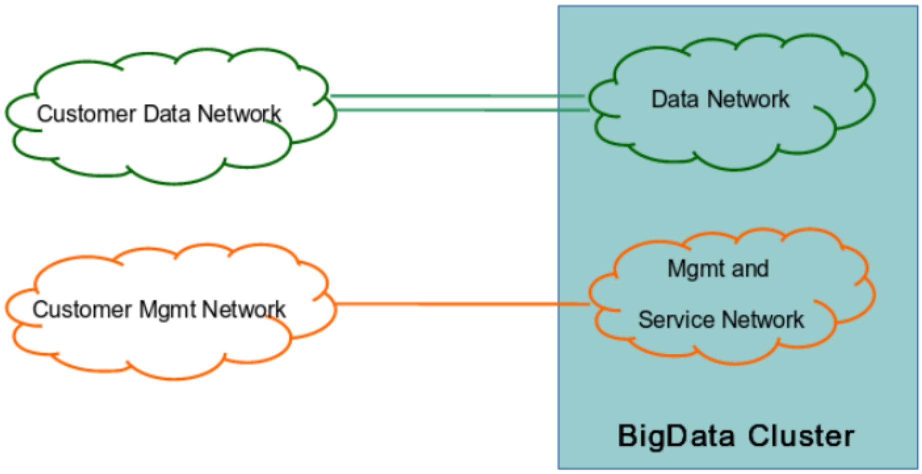
(2) Management network: This network is used to support operating system in deploying the software components and applications and its administration.

t. It uses a 1-Gigabit

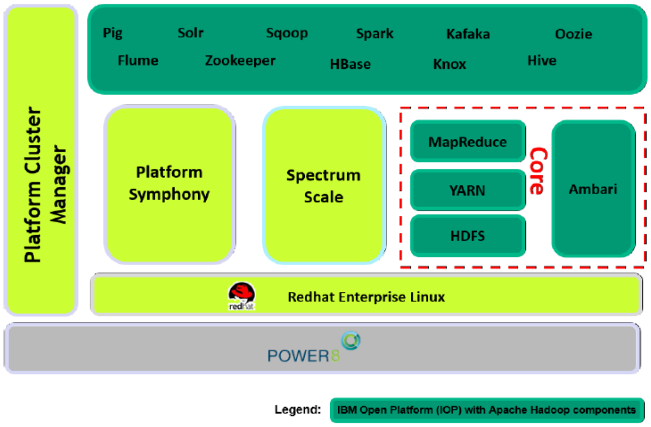
Ethernet interconnect.

(3) Data network: This high-performance network is used for accessing data in the cluster file system, communicating between analytics applications, and moving data into and out of the cluster. The data network uses 10-Gigabit Ethernet high-speed interconnect.

Below figure indicates the network been used in the solution



Software integration:



Software stack for the system

In this software stack:

• **IBM Spectrum Cluster Foundation (SCF)** provides hardware management and monitoring functions and a web console.

• **IBM Open Platform (IOP) for Hadoop and Spark** provides standard Hadoop and Spark services and its management console. Components of IOP are shown in dark green boxes in figure 3.

• **IBM Platform Symphony** provides the MapReduce engine functions and job-related web Console.

• **IBM Spectrum Scale** provides the HDFS functions for the solution.

This solution includes IBM power

This solution from IBM provides the futuristic solution for Big Data computation. Further pre-defined configuration and recommended configuration details can be obtained by contacting IBM for the type of application that the system must be implemented.

3.4 Nanostore Architecture for Data Centric Approach

[2]The paradigm shift from Process/Message/Application/Architecture/ Centric systems to Data Centric Systems, has led to the development of various architectural operations. These developments in the system design and Workload change certainly give a reason to rethink about the architectural styles that may further push us to develop more sophisticated architectural styles. Once such creative thinking has directed us about shifting from Microprocessors to Nanostores. The Author Clearly explains the approach in context with Data Centric systems.

The term Nanostores has been devised to represent the change from microprocessor to Nanotechnology and the emphasis given for data rather than traditional compute.

Motivation:

The convergence of Increasing and changing workloads, development of large scale distributed systems to handle the increasing workloads, changes in I/O behavior, prominence of nonvolatile memories in contrast to scaling for DRAM and innovations in hardware and software stacks provides an opportunity to think about this futuristic architectural style called as nanastores. Nanostores gives the favorable situation to leverage the convergence of these factors.

Design:

The main constituents of Nanostore are combination processors with nonvolatile memory to eliminate wastage occurring in storage hierarchy. This means that the data is stored in a single non-volatile memory structure that eliminates traditional disk and DRAM layers.

Physically multiple nanostore chips are organized to form microblades which are plugged into blade server boards.

Below figure shows how combination of processors and nonvolatile memory on the single chip are connected to one another to form a large cluster to handle data centric workloads.

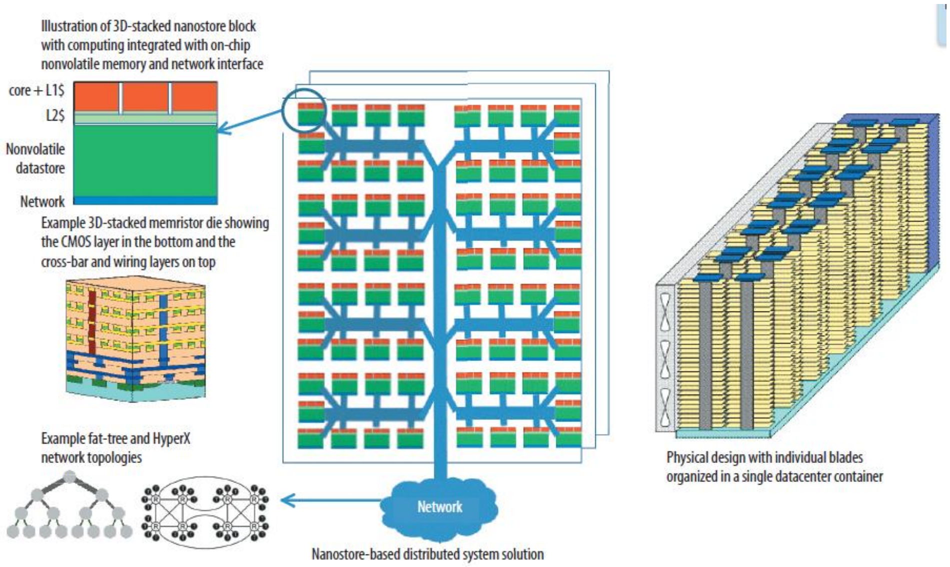
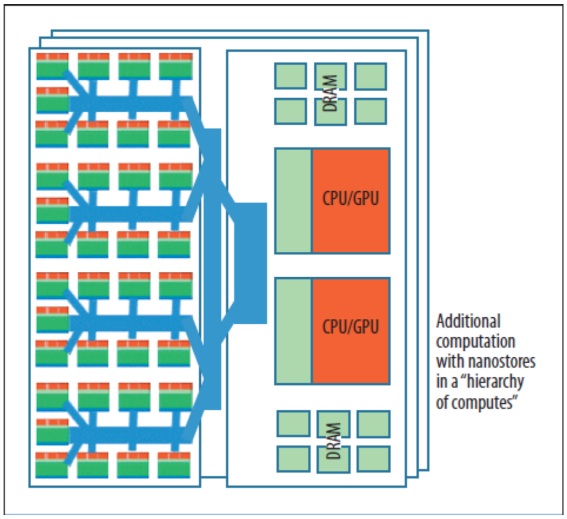


Illustration of 3D-stacked nanostore block with computing integrated with on-chip nonvolatile memory and network interface
core + L1$
L2$
Nonvolatile datastore
Network
Example 3D-stacked memristor die showing the CMOS layer in the bottom and the cross-bar and wiring layers on top
Example fat-tree and HyperX network topologies
Network
Nanostore-based distributed system solution
Physical design with individual blades organized in a single datacenter container

Power efficiency and heat dissipation limits the stacking of multiple processors, which results in low computational capacity.

To Increase the efficiency, additional elements can be added to create more determined computational hierarchy. This helps nanostores to bare the data workloads. The additional elements can be decided based on many factors such as application, infrastructure, desired outputs, available inputs, costs, storage capacity, network ability etc.

The sample demonstration on how additional elements can be included with the 3D stacked nanostores is shown below.


DRAM
CPU/GPU
CPU/GPU
DRAM
Additional computation with nanostores in a "hierarchy of computes"

The capacity of computation of these nanostores on data centric workloads can be determined by calculating the amount of raw compute processing that can be applied on one unit of data, on global as well as local levels.

Challenges:

Computation with the proposed design will cause bottlenecks at hardware, software and network levels as the complexity and sophistication of the data workload increases. It also creates issue in scalability as the additional elements added for high level computation increases. The proposed architecture calls for the adaptation of flash memory which limits the design by cost. Lastly, the design choices to increase the endurance of the architecture becomes difficult as computational load changes.

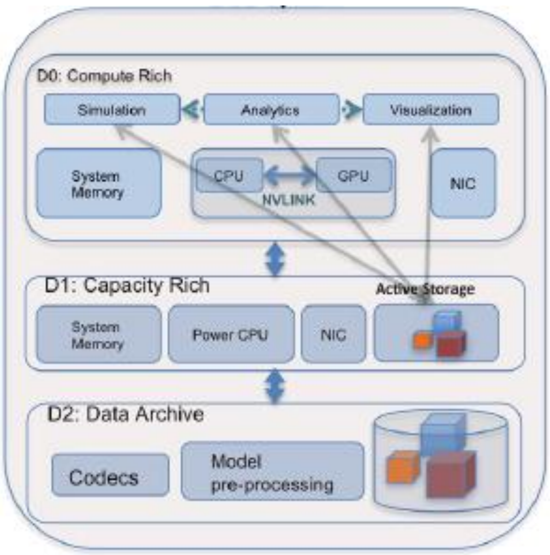
3.5 The interactive visualization of Big data using data centric approach

[17]Limitations on current strategies which motivated for Data Centric Approach:

- Performance challenge mainly due to limitations in movement of data
- Challenges in data security due to scattering of data among various systems
- Unable to provide complete visualization result to remote users
- Loose coupling of visualization with simulation and analytics
- Latency in data workflow

Design:

Below figure shows the Data Centric system architecture for Visualization of Big Data



As shown in the above figure there are separate simulation and visualization cluster of systems which avoid bottleneck of data flow and reduce the total latency of analytics through multi-application workflow. Also, due to active storage on shared cluster globalization of data becomes easy, which also give extended functionality of data protection and accessibility.

OpenGL on IBM Power systems connected to GPU by NVLINK gives the best of interactive visual analysis. This architecture has been prototyped in following applications:

- ✓ Big Data Visualization
- ✓ Isoperimetric Surfacing
- ✓ Ray casting
- ✓ Real time OCT Tree visualization

Data Centric approach in visualization has made the analytics easier and interactive by providing real life imaging, scalable rendering of large volumes of data, easy visual manipulation also supporting remote client visualization.

3.6 Data Centric Approach for automated data mining

[4]Data mining has always proved its worth as one of the most sophisticated conceptualization. The authors have proposed a design which makes the data mining methodology much more accessible for data community specifically for database and Business Intelligence circle.

Complexity of the methodologies for model definition, preparation, selection, and evaluation required for data mining has given the authors motivation to look for alternative design strategy, as these existing methodologies are unknown to the targeted group. The main goal is to design a system for data mining applications which can be understood and applied by targeted group of people.

Design:

The proposed design is mainly based on two factors.

Data centric focusing

Process automation

[4]The approach is demonstrated using two Data mining application:

- ✓ The Oracle Database 10g Release, 2 Predictive Analytics package [5] (OPA) and the
- ✓ Oracle Spreadsheet Add-In for Predictive Analytics (SPA)

The first application targets the database users and the second application targets the Business intelligence users.

The design is proposed around goal oriented high level tasks.

The Goal oriented tasks taken into consideration for the design are:

[4]EXPLAIN – Following are the functions that are expected form this task

- ✓ Use a high-level API.
- ✓ Embed data preparation.
- ✓ Do not persist other objects besides the results.
- ✓ Return a figure of merit (explanatory power) and rank for each attribute

[4]PREDICT – Following are the functionality expected from this task

- ✓ Automatically determine problem type (classification or regression).
- ✓ Embed data preparation.
- ✓ Avoid trivial results for classification problems
- ✓ Provide a measure of generalization performance that allows comparison across different datasets and falls in the [0,1] range.
- ✓ Provide predictions for all the records in the input dataset

These goal oriented tasks should be implemented using Data Centric application programming interfaces such that those API's should define some set of operations that can be performed on the target dataset and provide desired results. This makes data mining as easy as any other form of querying on data.

Process Automation:

Automation here refers the operation on data by the system with minimal user intervention.

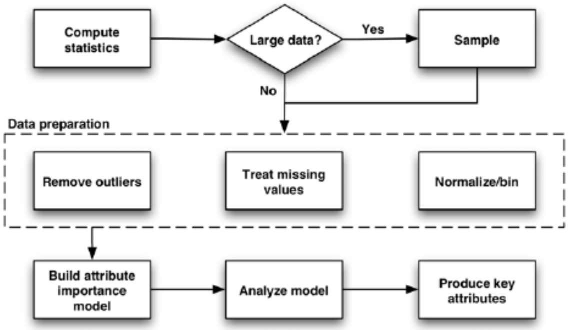
This can be achieved by developing complex API's for data operations.

[4]The process automation is defined by following set of steps.

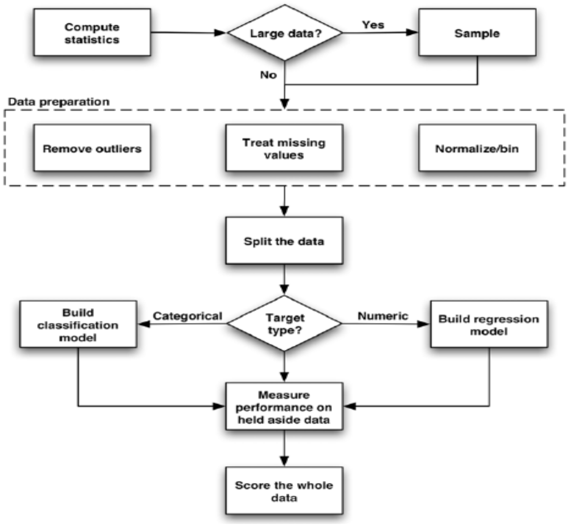
1. Computation of statistics
2. Sampling
3. Attribute data type identification
4. Attribute selection
5. Algorithm selection
6. Data transformation
7. Model selection and quality assessment
8. Output generation

Following are the models designed for EXPLAIN and PREDICT functionalities which when combined would satisfy the above steps in Data Mining automation.

EXPLAIN:



PREDICT



This approach has been prototyped and tested. It has been proved to be effective approach on challenging data levels. This satisfies performance and ease of usability factors which were desired outcome of this invention.

3.7 Data centric approach in cloud computing:

[3]Due to changing technology, many issues such as ownership of data, lack of transparency in the flow of the data, data access in cloud etc. had to be addressed. Data centric detective approach is adopted to overcome disadvantages of traditional preventive approaches. The approach is addressed on a framework known as Trust Cloud

Motivation:

- ✓ Paradigm shift from “System as assets” to “Information as assets”.

  The growing need of cloud computing has made the whole technology to be available as a service. This forces to deal with the issues such as risk of hardware and software integrity, Security and maintenance of the infrastructure. Data being the real asset of the organization, efficient handling of data becomes typical trait. Since the users only own the data not the infrastructure itself when it comes to cloud, a question arises about the privacy concerns and security relevance.

Below incidents conveys that the existing technology of data protection has flaws:

3 Google's Lost Email Accounts

4 EMC/ RSA Security Breach

5 User Data Stolen from Sony PlayStation Network

6 UK National Healthcare System (NHS) Hacked

7 Amazon EC2 Outages

This calls for a better approach for better results, A detective approach! that's superior to current preventive approach. This detective approach not only prevents the incident from reoccurring but also find the policy/ system breaches beforehand so that the incidents won't occur in the first place. This detective approach is based on data centric logging.

Approach:

Data centric logging means that the logging is done with respect to data/ information which are the key assets of cloud computing. Data centric logging should satisfy the following requirements.

- Tracking files:

  This means that the proposed detective approach should let us know the travelling history of the file, that is tracing and recording the exact file life cycle.

- Tracking data:

  This feature tracks the history of the data, which includes the origin, collection/recreation, evolution and the use of the data. This was data owners and even data provider can be protected by maintaining the provenance data. Data centric logging also enables the support of consistency of the data, recovery, backup etc.

- Tracking information:

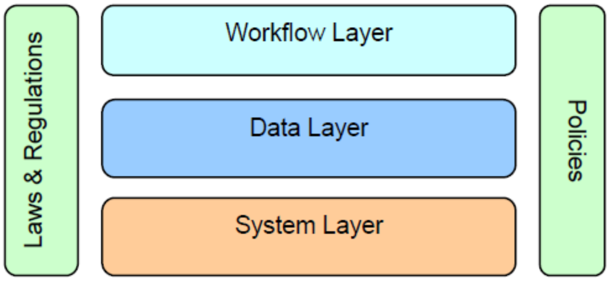
  Data being the raw content, information is extracted from this data. Ensuring the correctness and the appropriate amount of information extraction is the typical trait here.

- Tracking information and data flows:

  This manages the accountability of the services/business functions and their providers within the cloud. Auditing the access/workflow of the data ensures that approval routes, role management and decision making flows have kept their integrity.

  Trust cloud framework.

  Below diagram shows the Data centric abstraction layer in the framework.



- System layer – addresses tracking of files across the cloud.
- Data layer – addresses tracking of change of data and Information across the cloud.
- Workflow layer – addresses data and information flow in the cloud.
- Law and regulations layer – addresses data-centric logging requirements mandated by external laws and regulations.
- Policies layer – addresses data-centric audit requirements mandated by internal governance and audit requirements.

Data centric logging not only increases the transparency of the information in the cloud but also increases the accountability of the information in service provider's perspective. This approach not only gives the advantage of the detective methods but also provides the protective traits such as end to end encryption. The data can be tracked traced and the integrity can be maintained.

4. Challenges and Limitations of Data Centric Systems:

Even though data centric systems eliminate the challenges in traditional architectural styles such as application centric architectures, message centric architectures etc. It has some limitations for its implementation methodologies. Below mentioned are the few of the major challenges of Data Centric systems.

a) Data Motion is expensive which forces performance tradeoff for cost.

b) Fault detection will be difficult on system failure as the issue must be narrowed down among several clustered systems

c) Scalability will be a challenge due to complex integration and high performance costs.

d) The system should interpret the data to the maximum efficiency before it analyses it as it cannot rely on hardware

5. Future Trends:

Data centric systems design approach can be extended to different real world applications. With the necessary addons this approach can dominate various fields of data providing optimal and effective solutions. Below listed are those areas in which the approach can be used effectively with enhancements:

1) [1]Data Centric systems can be provided with improved performance factors by adopting distributed scheduler. This can overcome the weakness of centralized scheduling in traditional data centered systems.
2) [15]HPC Analytics applications which make use of access patterns can be supported with Data restructuring and data centric scheduling techniques[21] in mapreduce.This technique can help in reorganization of data before it's been fed into analytical tools such as MAP Reduce, when they are migrated to data intensive environment.
3) [20]Based on application SCM and NAND flash SSD drives can be combined together to form a hybrid for data centric computation. This can reduce the data movement inside storage, reducing latency and contributing for system efficiency and performance.
4) [5]Phase change memory can be employed for Data Centric Systems computing to to exploit storage structure for maximum efficiency
5) [6]Data centric approach can be extended to embedded computing with distributed technology, for better quality of service
6) [9]Data centric approach can be used to estimate the quality of role mining
7) A data centric middleware can be developed for as an independent software infrastructure to Monitor, control the Analytics and computational process.

6. Conclusion

Data Centric approach is the new future of data science. With growing data, the data world is full of raw unstructured data, which requires a dedicated infrastructure to extract the real information from available metadata and translate it into meaningful and usable data. Data Centric approach provides such infrastructure with efficient performance and smooth data workflow compared to existing approaches. The design and application section speaks about five of the many prominent designs and applications of data centric system domain. With minimal limitations this technology will soon be wide spread covering most of the real world applications. Its ease of adaptiveness, lower power consumption, high performance factors, lower data integration costs and flexible structure makes computing and analytics of data more efficient than ever, making this technology the future of Supercomputing Era.

7. References

[1] S. Goel; H. Sharda; D. Taniar," Distributed Scheduler for High Performance Data-Centric Systems",IEEE TENCON 2003. Conference on Convergent Technologies for the Asia-Pacific Region, Pages: 1157 - 1161 Vol.3, DOI: 10.1109/TENCON.2003.1273429

[2] Parthasarathy Ranganathan," Microprocessors to Nanostores: Rethinking Data-Centric Systems",IEEE Year: 2011, Volume: 44, Issue: 1, Pages: 39 - 48, DOI: 10.1109/MC.2011.18

[3] Ryan K. L. Ko, Markus Kirchberg, Bu Sung Lee," From System-centric to Data-centric Logging – Accountability, Trust & Security in Cloud Computing",IEEE Defense Science Research Conference and Expo (DSR), 2011, Pages: 1 - 4, DOI: 10.1109/DSR.2011.6026885

[4] M.M. Campos, P.J. Stengard, B.L. Milenova," Data-centric automate d data mining ",IEEE Machine Learning and Applications, 2005. Proceedings. Fourth International Conference on 15-17 Dec. 2005

[5] Jing Li,"Enabling phase-change memory for data-centric computing: Technology, circuit and system", 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Pages: 21 - 24, DOI: 10.1109/ISCAS.2015.7168560

[6] Hector Perez and J. Javier Gutierrez," Enabling Data-Centric Distribution Technology for Partitioned Embedded Systems", IEEE Transactions on Parallel and Distributed Systems 2016, Pages: 3186 - 3198, DOI: 10.1109/TPDS.2016.2531695

[7] Hong-Mei Chen; Rick Kazman; Serge Haziyev "Agile Big Data Analytics for Web-Based Systems: An Architecture-Centric Approach" , IEEE Transactions on Big Data 2016, Pages: 234 - 248, DOI: 10.1109/TBDATA.2016.2564982

[8] Pierre Bourhis; Daniel Deutch; Yuval Moskovitch,"Analyzing data-centric applications: Why, what-if, and how-to", 2016 IEEE 32nd International Conference on Data Engineering (ICDE), Pages: 779 790, DOI: 10.1109/ICDE.2016.7498289

[9] Lijun Dong; Kui Wu; Guoming Tang,"A Data-Centric Approach to Quality Estimation of Role Mining Results", IEEE Transactions on Information Forensics and Security 2016, Pages: 2678 - 2692, DOI: 10.1109/TIFS.2016.2594137

[10] Dr. Tilak Agerwala, Dr. Michael Perrone," Data Centric Systems The Next Paradigm in Computing", IBM Research

[11] Wafa Najjar; Houda Jouani; Rim Bouhouch; Salem Hasnaoui,"A Cluster-Based Data-Centric Model for Network-Aware Task Scheduling in Distributed Systems", IEEE Latin America Transactions 2012,Pages: 1149 - 1149, DOI: 10.1109/TLA.2012.6142451

[12] Carlo Batini, Monica Scannapieca," Data Quality Concepts, Methodologies and Techniques",July 2006

[13] Yanpei Chen," Workload-Driven Design and Evaluation of Large-Scale Data-Centric Systems", Technical Report No. UCB/EECS-2012-73, May 9, 2012

[14] Knud Möller," Lifecycle models of data-centric systems and Domains", 2012 – IOS Press

[15] Saba Sehrish; Grant Mackey; Pengju Shang; Jun Wang; John Bent," Supporting HPC Analytics Applications with Access Patterns Using Data Restructuring and Data-Centric Scheduling Techniques in MapReduce ",IEEE Transactions on Parallel and Distributed Systems ,2013, Pages: 158 - 169, DOI: 10.1109/TPDS.2012.88

[16] Rajive Joshi, "Data-Centric Architecture for Space Systems", Real-Time Innovations, 3rd Annual Workshop on Flight Software, Nov 5, 2009 , http://www.rti.com

[17] Bruce D'Amora," Data Centric Interactive Visualization of Very Large Data", IBM T.J. Watson Research/Data Centric Systems

[18] Nilamadhab Mishra, Hsien-Tsung Chang, and Chung-Chih Lin," Data-Centric Knowledge Discovery Strategy for a Safety-Critical Sensor Application", International Journal of Antennas and Propagation, Volume 2014 (2014), Article ID 172186, 11 pages

[19] “IBM Data Engine for Hadoop and Spark – Power Systems Edition”, An IBM System Reference Guide, August 22, 2016. https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?subtype=ca&infotype=an&supplier=897&letternum=ENUS116-011#availx

[20] Shun Okamoto; Chao Sun; Shogo Hachiya; Tomoaki Yamada; Yusuke Saito; Tomoko Ogura Iwasaki; Ken Takeuchi,”Application Driven SCM and NAND Flash Hybrid SSD Design for Data-Centric Computing System”, 2015 IEEE International Memory Workshop (IMW), Pages: 1 - 4, DOI: 10.1109/IMW.2015.7150277

[21] Zujie Ren, Xiaohong Zhang and Weisong Shi, “Resource Scheduling in Data-Centric Systems”.

[22] http://flagshipsg.com/it-empowers-with-data-centric-design/

[23] http://www.informationweek.com/government/big-data-analytics/ibm-doe-working-toward-data-centric-supercomputing/d/d-id/1321651

[24] https://en.wikipedia.org/wiki/Database-centric_architecture

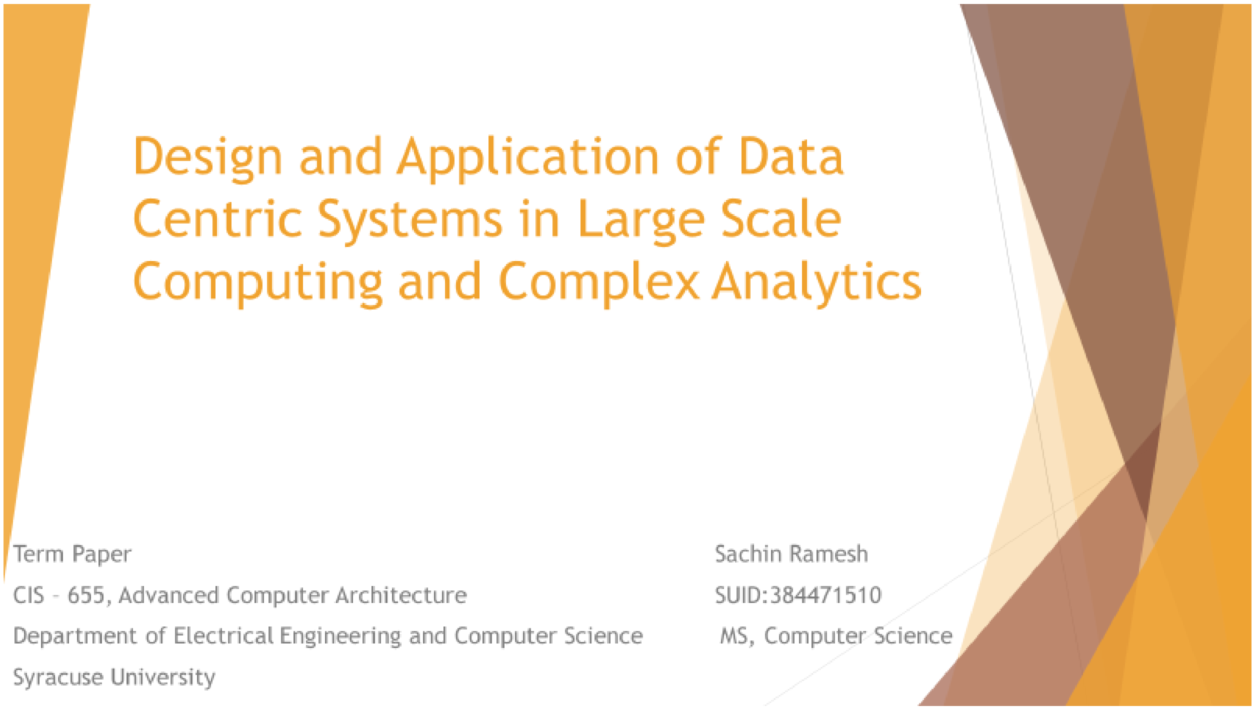
[25] http://data-informed.com/take-data-centric-approach-managing-unstructured-data/

[26] http://www.tarmin.com/company/tarmin-updates/data-defined-storage-blog/415-a-data-centric-approach-to-managing-data-in-the-cloud

[27] https://www.voltage.com/big-data-2/five-reasons-use-data-centric-security-secure-hadoop-deployment

Appendix:

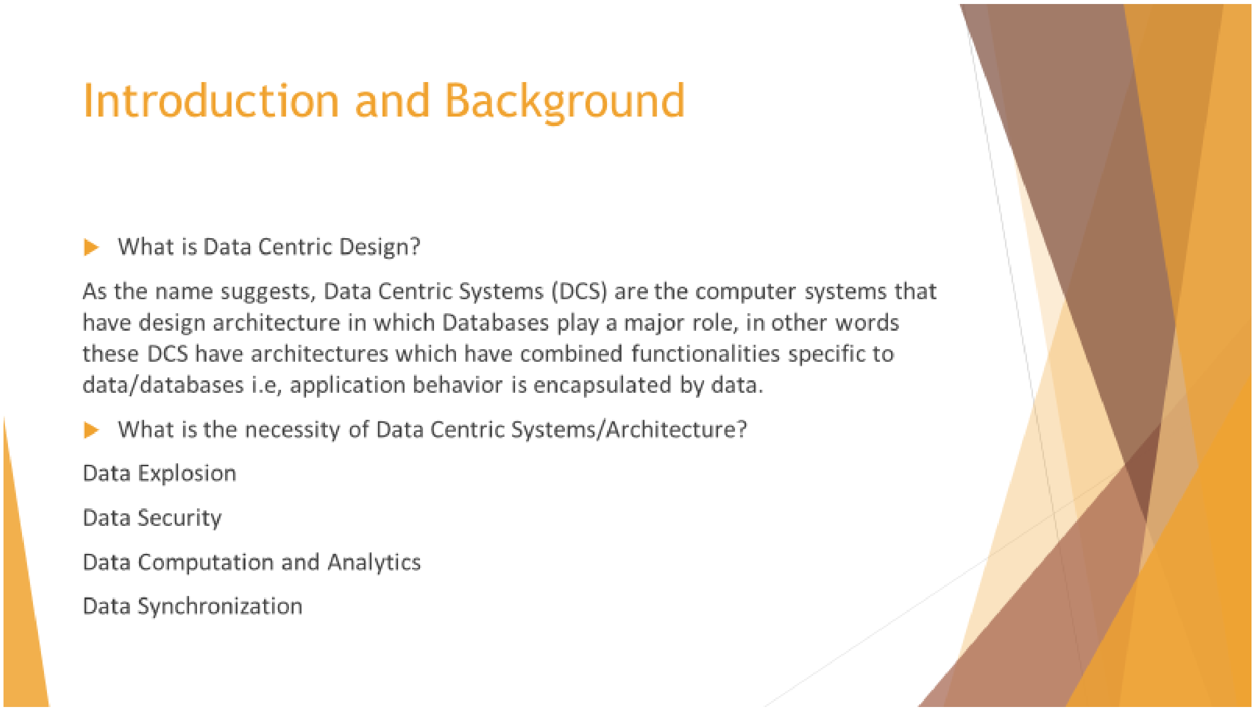
Term paper
Summary.pptx



Design and Application of Data Centric Systems in Large Scale Computing and Complex Analytics

Term Paper
CIS - 655, Advanced Computer Architecture
Department of Electrical Engineering and Computer Science
Syracuse University

Sachin Ramesh
SUID:384471510
MS, Computer Science



Introduction and Background

- What is Data Centric Design?

As the name suggests, Data Centric Systems (DCS) are the computer systems that have design architecture in which Databases play a major role, in other words these DCS have architectures which have combined functionalities specific to data/databases i.e, application behavior is encapsulated by data.

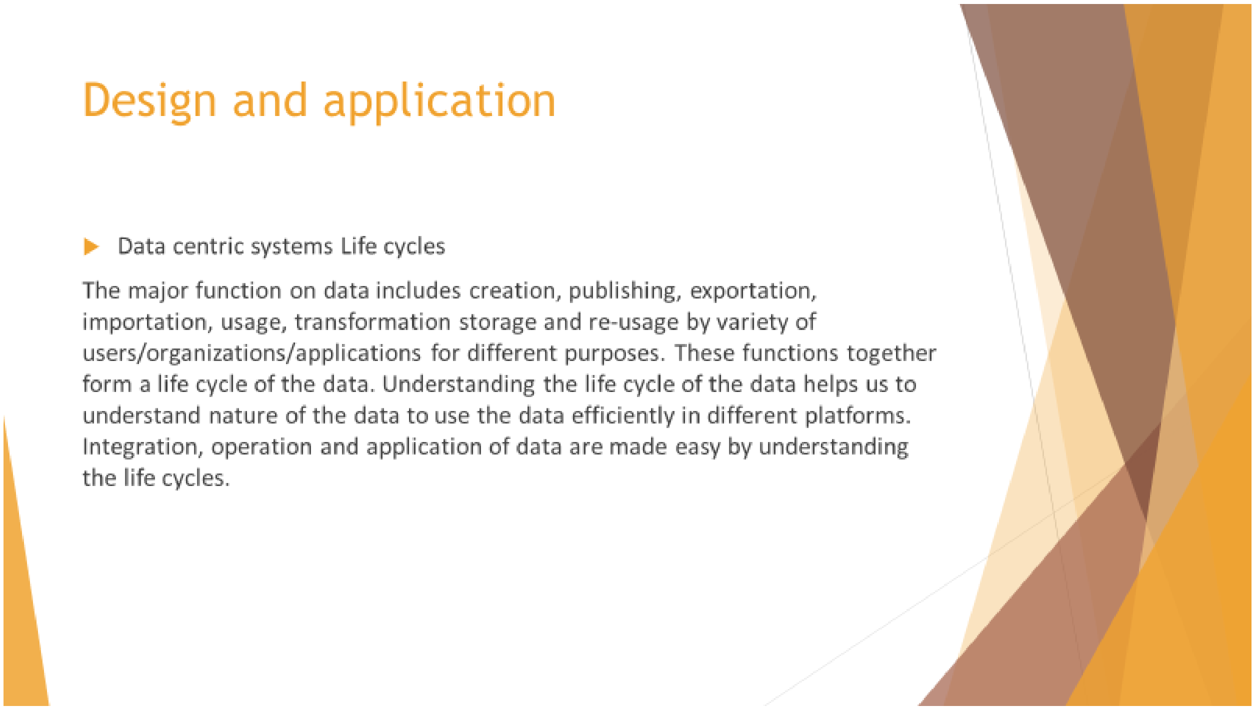
- What is the necessity of Data Centric Systems/Architecture?

Data Explosion

Data Security

Data Computation and Analytics

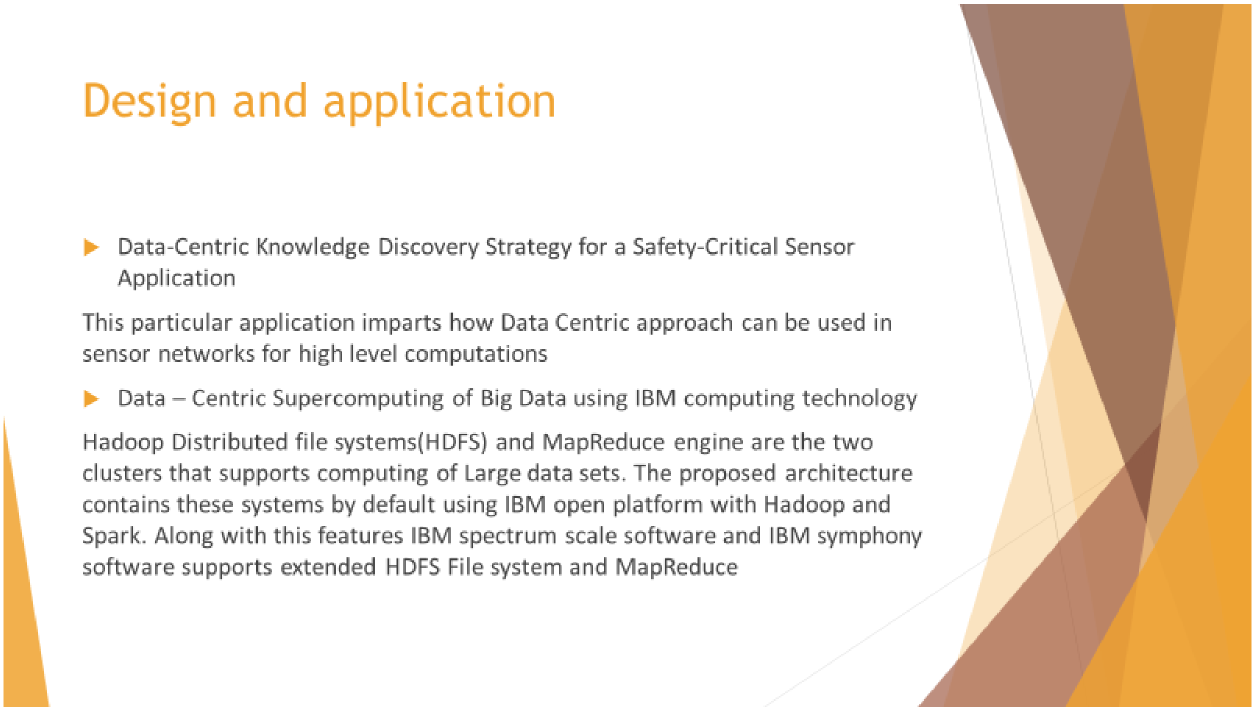
Data Synchronization



Design and application

- Data centric systems Life cycles

The major function on data includes creation, publishing, exportation, importation, usage, transformation storage and re-usage by variety of users/organizations/applications for different purposes. These functions together form a life cycle of the data. Understanding the life cycle of the data helps us to understand nature of the data to use the data efficiently in different platforms. Integration, operation and application of data are made easy by understanding the life cycles.



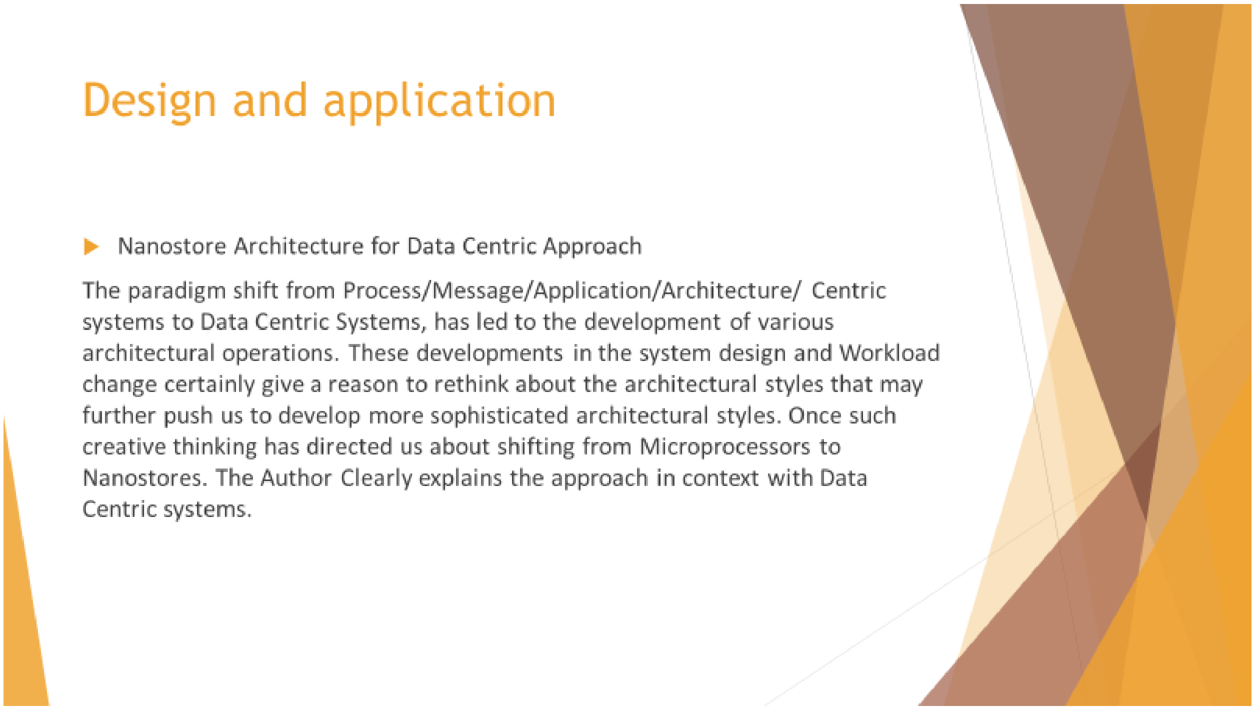
Design and application

- Data-Centric Knowledge Discovery Strategy for a Safety-Critical Sensor Application

This particular application imparts how Data Centric approach can be used in sensor networks for high level computations

- Data – Centric Supercomputing of Big Data using IBM computing technology

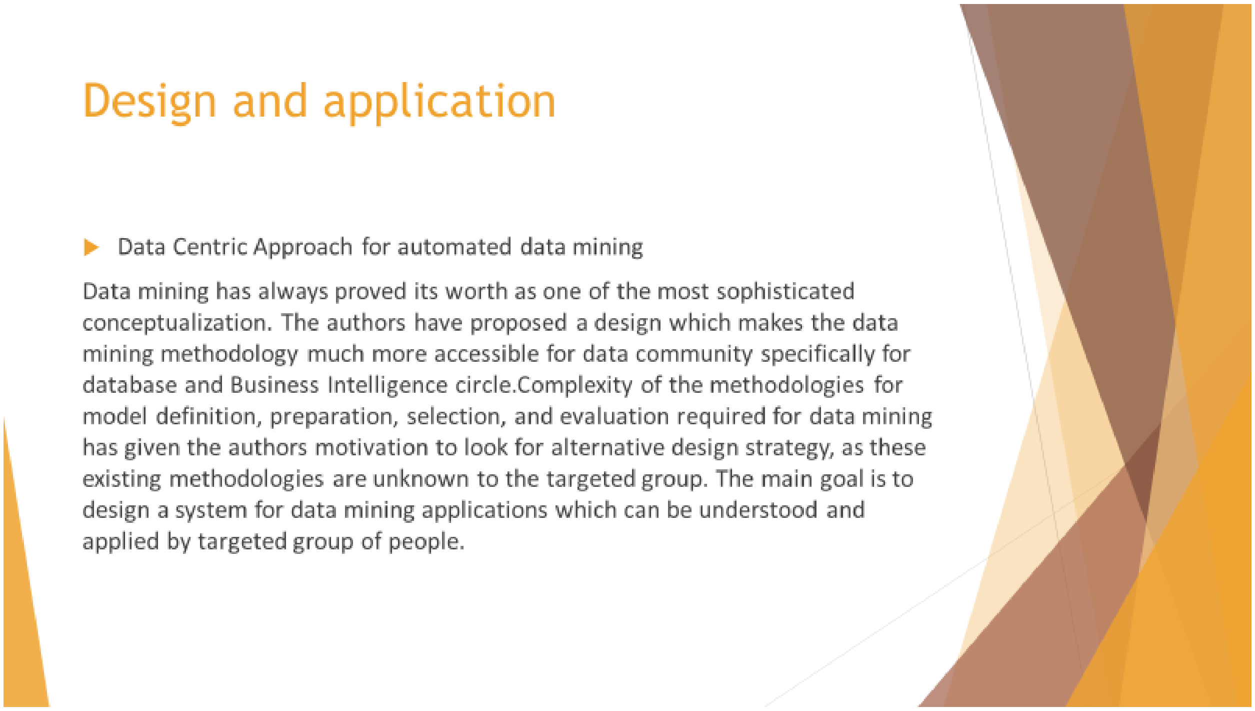
Hadoop Distributed file systems(HDFS) and MapReduce engine are the two clusters that supports computing of Large data sets. The proposed architecture contains these systems by default using IBM open platform with Hadoop and Spark. Along with this features IBM spectrum scale software and IBM symphony software supports extended HDFS File system and MapReduce



Design and application

- Nanostore Architecture for Data Centric Approach

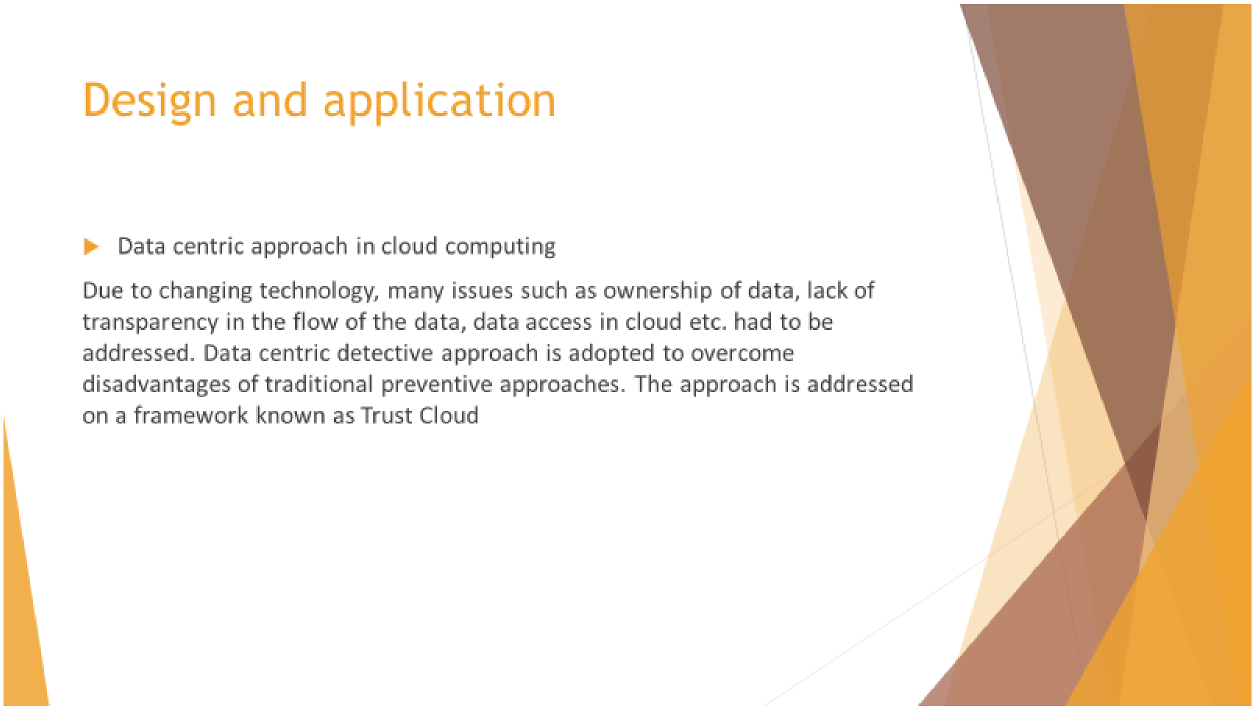
The paradigm shift from Process/Message/Application/Architecture/ Centric systems to Data Centric Systems, has led to the development of various architectural operations. These developments in the system design and Workload change certainly give a reason to rethink about the architectural styles that may further push us to develop more sophisticated architectural styles. Once such creative thinking has directed us about shifting from Microprocessors to Nanostores. The Author Clearly explains the approach in context with Data Centric systems.



Design and application

- Data Centric Approach for automated data mining

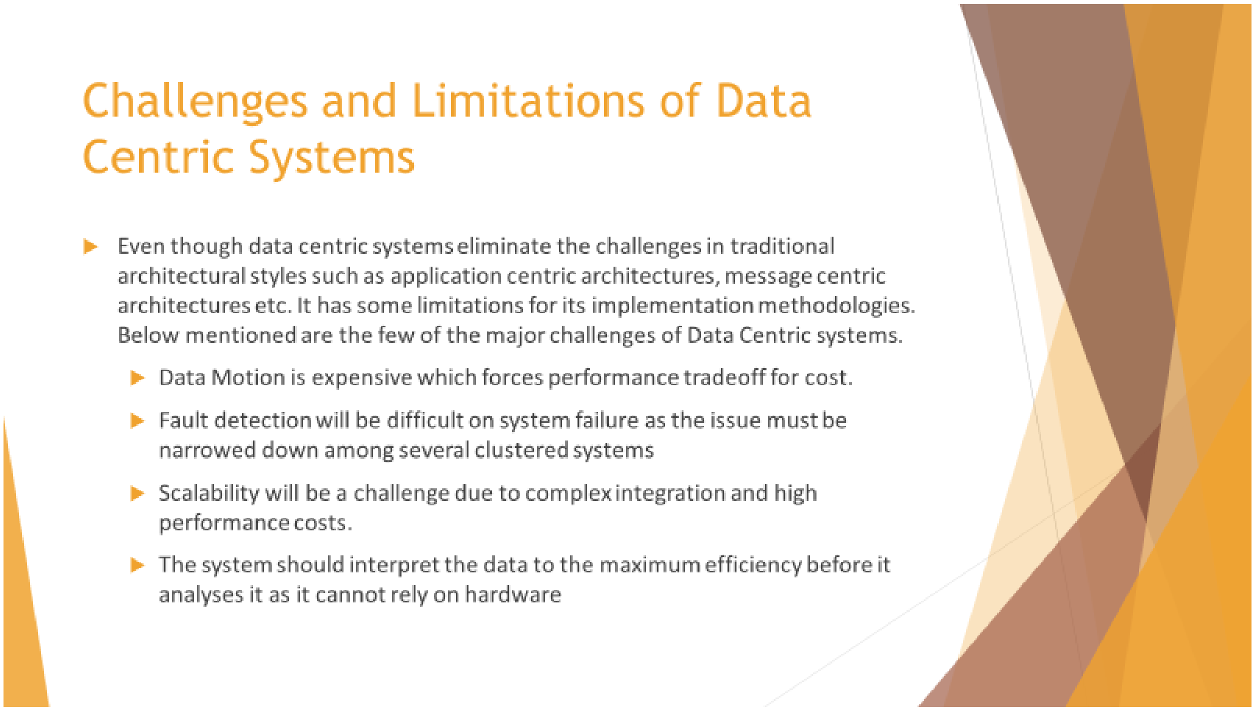
Data mining has always proved its worth as one of the most sophisticated conceptualization. The authors have proposed a design which makes the data mining methodology much more accessible for data community specifically for database and Business Intelligence circle.Complexity of the methodologies for model definition, preparation, selection, and evaluation required for data mining has given the authors motivation to look for alternative design strategy, as these existing methodologies are unknown to the targeted group. The main goal is to design a system for data mining applications which can be understood and applied by targeted group of people.



Design and application

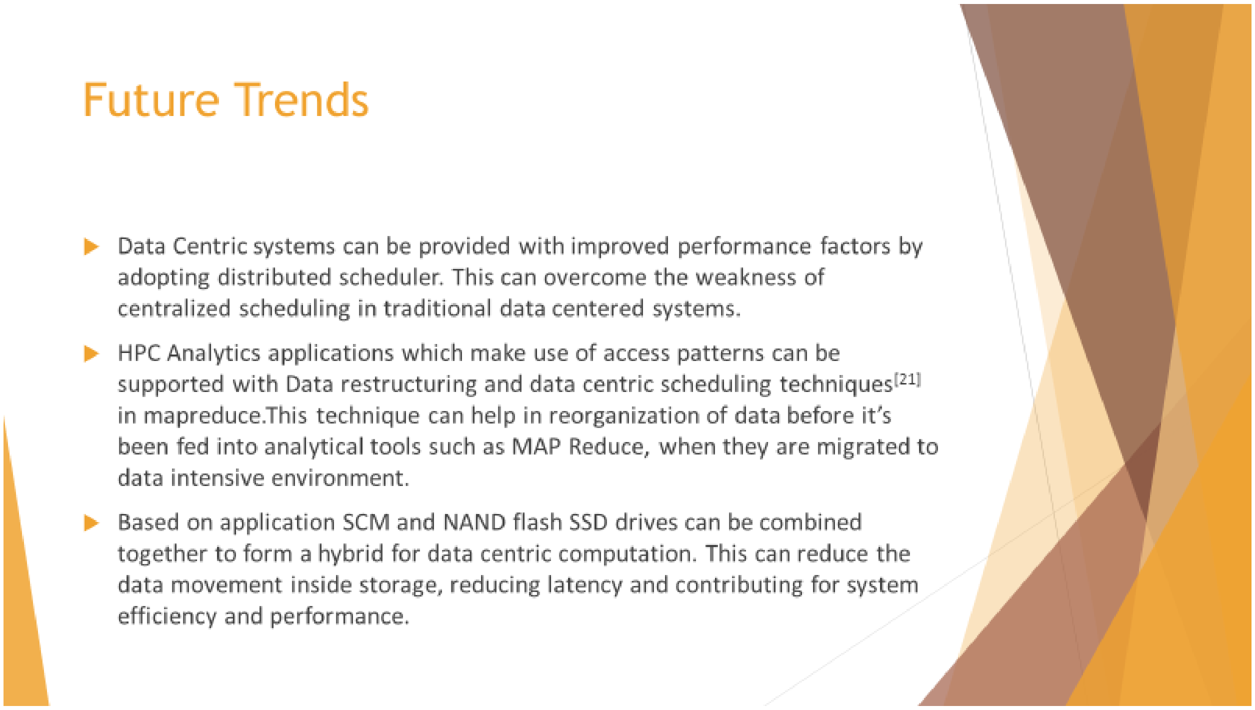
- Data centric approach in cloud computing

Due to changing technology, many issues such as ownership of data, lack of transparency in the flow of the data, data access in cloud etc. had to be addressed. Data centric detective approach is adopted to overcome disadvantages of traditional preventive approaches. The approach is addressed on a framework known as Trust Cloud



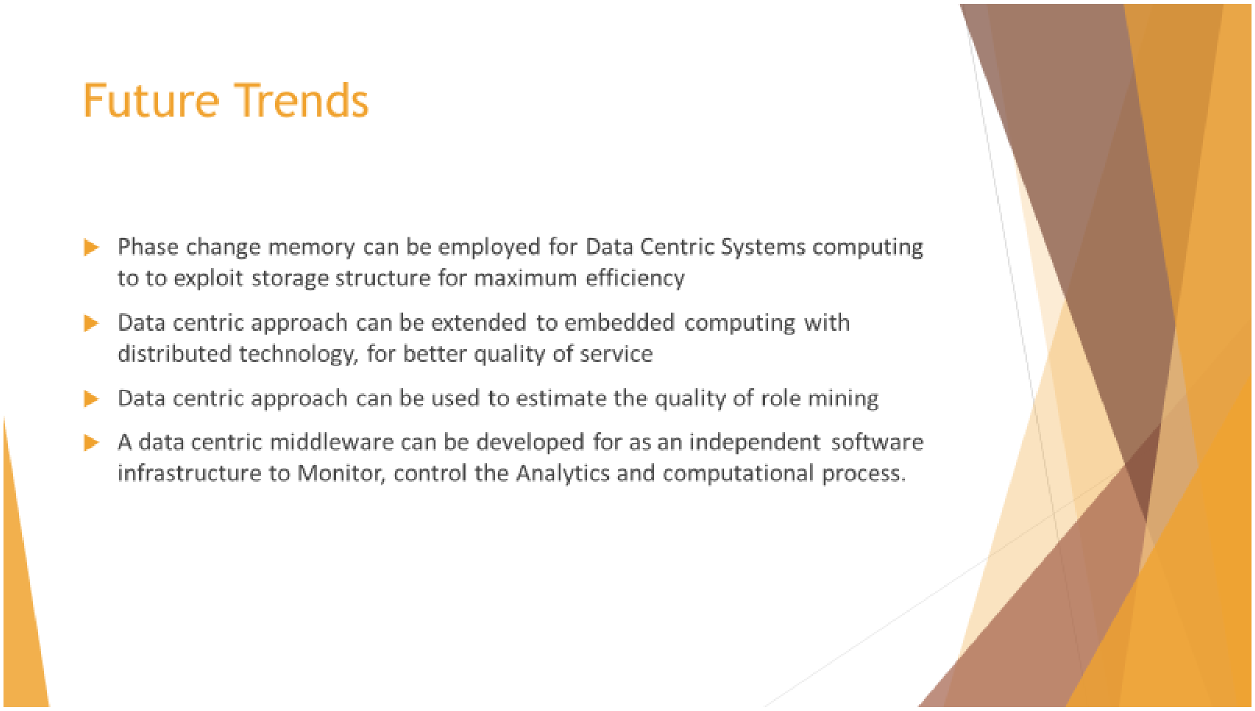
Challenges and Limitations of Data Centric Systems

- Even though data centric systems eliminate the challenges in traditional architectural styles such as application centric architectures, message centric architectures etc. It has some limitations for its implementation methodologies. Below mentioned are the few of the major challenges of Data Centric systems.
  - Data Motion is expensive which forces performance tradeoff for cost.
  - Fault detection will be difficult on system failure as the issue must be narrowed down among several clustered systems
  - Scalability will be a challenge due to complex integration and high performance costs.
  - The system should interpret the data to the maximum efficiency before it analyses it as it cannot rely on hardware



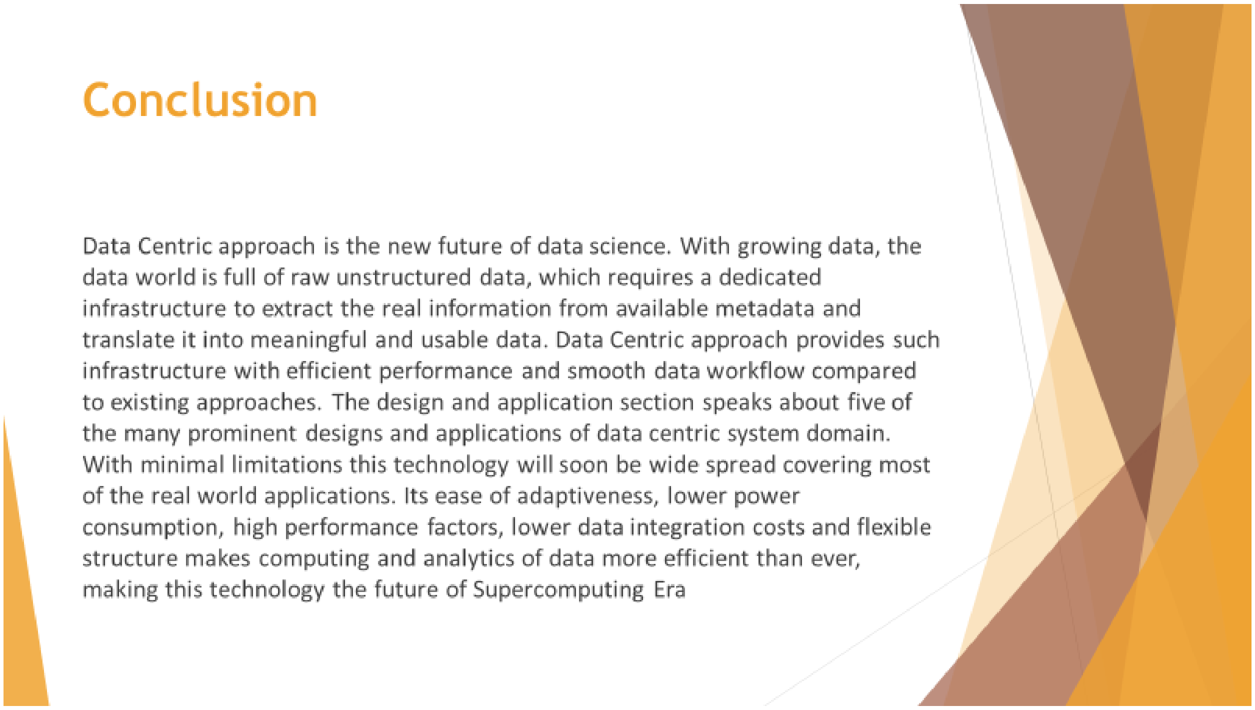
Future Trends

- Data Centric systems can be provided with improved performance factors by adopting distributed scheduler. This can overcome the weakness of centralized scheduling in traditional data centered systems.
- HPC Analytics applications which make use of access patterns can be supported with Data restructuring and data centric scheduling techniques[21] in mapreduce.This technique can help in reorganization of data before it's been fed into analytical tools such as MAP Reduce, when they are migrated to data intensive environment.
- Based on application SCM and NAND flash SSD drives can be combined together to form a hybrid for data centric computation. This can reduce the data movement inside storage, reducing latency and contributing for system efficiency and performance.



Future Trends

- Phase change memory can be employed for Data Centric Systems computing to to exploit storage structure for maximum efficiency
- Data centric approach can be extended to embedded computing with distributed technology, for better quality of service
- Data centric approach can be used to estimate the quality of role mining
- A data centric middleware can be developed for as an independent software infrastructure to Monitor, control the Analytics and computational process.



Conclusion

Data Centric approach is the new future of data science. With growing data, the data world is full of raw unstructured data, which requires a dedicated infrastructure to extract the real information from available metadata and translate it into meaningful and usable data. Data Centric approach provides such infrastructure with efficient performance and smooth data workflow compared to existing approaches. The design and application section speaks about five of the many prominent designs and applications of data centric system domain. With minimal limitations this technology will soon be wide spread covering most of the real world applications. Its ease of adaptiveness, lower power consumption, high performance factors, lower data integration costs and flexible structure makes computing and analytics of data more efficient than ever, making this technology the future of Supercomputing Era