A Survey on Partitioning Techniques for Structured Data

Rabab Mohamed Nabawy
Menoufia Univerisity
Egypt

Heba Elbeh
Menoufia University
Egypt

Hamdi Moussa
Menoufia University
Egypt

ABSTRACT
Cloud data storage attempts to redefine the issues targeted on customer's out-sourced data (data that is not stored or retrieved from the customers own servers) here we noticed that, from a user's point of view, relying upon a single SP for his outsourced data is not very promising. Additionally, providing better privacy as well as ensure data availability and reliability that can be reached by splitting the user's data block into chunks and distributing them among the available SPs in this way less than a threshold number of SPs can take part in successful retrieval of the whole data block.

in a database relations schema are usually decomposed into smaller fragments, but we did not suggest any justification or details for such process.

So our survey will answer some questions in details such as why fragmentation, How can we fragment, How much that we should fragment, how can we test the correctness of fragmentation, how should we allocate, what is the needed information for both fragmentation and allocation.

General Terms
Data and Information Systems.

Keywords
Fragmentation, Survey, Horizontal Fragmentation, Vertical Fragmentation, Hybrid Fragmentation.

1. INTRODUCTION
In this review paper, the main concerned is the partitioning techniques that are used for splitting data that is stored in Database Management System.

With the advancement in technology, data generated from web applications like Facebook, Twitter and etc. become bigger and bigger. It was so difficult for traditional database systems to be managed.in order to increase the data stores performance, partitioning was used, which improves various factors like scalability, efficiency and availability.

Fragmentation reasons:
there are number of reasons to fragment data [1]

First, application views are usually subsets of relations. Therefore, the locality of accesses of applications is defined not on entire relations but on their subsets. For this reason it is only natural to consider subsets of relations as distribution units.
Second, if the applications that have views defined on a given relation reside at different sites,we have two options can be followed, with the entire relation being the unit of distribution

Also such relation is not replicated and will be stored at only one site, or it may be replicated at all or at some of the sites where those applications remain. The former results in an unnecessarily high volume of remote data accesses. The latter, furthermore has unnecessary replication, which causes problems in executing updates and may not be preferred if storage is limited.[1]
Finally, the distribution of a relation into fragments, each being treated as a unit, permits a number of transactions to execute concurrently.

The rest of this paper is organized as follows. Section 2 presents the essential Fragmentation methods that can be used in splitting data as needed. Section 2.1 shows the Horizontal Fragmentation and presents the related techniques that are suggested in such method. A Vertical Fragmentation is covered in Section 2.2 and also shows the different techniques that are used. Section 2.3 presents Hybrid Fragmentation and its portioning techniques. Finally, the conclusion is presented in Section 2.4.

2. FRAGMENTATION METHODS
There are two essential fragmentation strategies: Horizontal and Vertical and there is a possibility of Hybrid Fragments.[2,3]

2.1 Horizontal Fragmentation
Horizontal Fragmentation divides a relation over its tuples [2]. The main objective of the Horizontal fragmentation is to get smaller fragment to maximize the local processing of queries, then the fragments will be stored in the use sites.

The selection predicate is associated to each fragment and defines the property that the grouping of records who compose this specific data fragments are based on. The Horizontal Fragments obtained have the same structure with the global relation from which was extracted; however those fragments will contain different tuples. generally, the Horizontal fragments will be disjoint.
We have two subtypes of Horizontal fragmentation of a global relation:
Primary Horizontal fragmentation and derived horizontal fragmentation.

We added to the PROJ relation a new attribute(LOC) that refers to the place of a project. Fig1 depicts the database instance we will use the PROJ relation of that is partitioned in a Horizontal way into two sub-relations. Subrelation PROJ1 contains information about projects whose budgets are less than $200,000, whereas PROJ2 stores information about projects with larger budgets.

PNO	PNAME	LOC
P1	Instrumentation	Montreal
P2	Database Develop	New York
P3	CAD/CAM	New York
P4	Maintenance	Paris

PNO	BUDGET
P1	150000
P2	135000
P3	250000
P4	310000

Fig 1 : Horizontal Fragmentation [2]

2.1.1 Primary Horizontal Fragmentation

The tuples selection of a relation mainly based upon the properties of the attribute of a specific relation.[2]

This subtype of fragmentation can be obtained by using the selection operator that is used to the global relation and it is illustrated as follow:

Ri=σp(R),where
σ–selection operator
p–fragmentation predicate
R–global relation
Ri- the result of Horizontal Fragmentation.

we can represent The reconstruction of the global relation R can be accomplished by using the reunion of the fragments as follows:

$$\bigcup_{i=1}^{n} R_i$$

2.1.2 Derived Horizontal Fragmentation:

It is based upon Horizontal Fragmentation of a different global relation and it can be accomplished if there exists a binary correspondence with the the table which is going to be fragmented. The resulted fragments can get by gathering the R relation's tuples with Horizontal Fragments of the S table . After the fragmentation , we finish up with Ri fragments.

$$R_i = R \triangleright S_i$$

Where

$\triangleright$ semi join operator
Z maximum number of fragments to be defined on relation R
Si Horizontal fragments on S as following: $\sigma F_i(S)$.

2.1.3 Checking Correctness:

We should now check correctness for the fragmentation algorithms

2.1.3.1 Completeness.

The primary Horizontal Fragmentation completeness is based at most upon the selection predicates that are used provided that the selection predicates are complete; the resulting fragmentation is guaranteed to be complete as well. Since the basis of the fragmentation algorithm is a set of complete and minimal predicates, Pr0, completeness is guaranteed therefore no mistakes.

The completeness of the Derived Horizontal Fragmentation is comparatively more difficult to define. The difficulty is due to the fact that the predicate determining the fragmentation includes two relations. Let us first define the completeness rule officially and then take a look at the following example. Assume R is a member relation for the link with owner relation S, wherever R and S are fragmented as follows: FR={R1; R2; ……. ; Rw } and FS={S1;S2;…… ; Sw } , respectively.

Moreover, let A be the join attribute between R and S. Then for each tuple t of Ri, there should be a tuple t0 of Si So that t[A] =t' [A].

As an example, there should be no ASG tuple which has a project number that is not also included in PROJ. likewise, there should be no EMP tuples with TITLE values wherever the same TITLE value does not exist in PAY as well. Such rule is known as referential integrity and ensures that the tuples of any fragment of the member relation also are in the owner relation.

2.1.3.2 Reconstruction.

The reconstruction of a global relation from its fragments is performed using the union operator for both the primary fragmentation and the Derived Horizontal Fragmentation. Thus, for a relation R with fragmentation FR={R1,R2,……..Rw}

$$R = \bigcup R_i, \forall R_i \ldots\ldots FR$$

2.1.3.3 Disjointness.

It is easier to construct disjointness of fragmentation for primary than for Derived Horizontal Fragmentation. In the former case, disjointness is guaranteed as long as the minterm predicates determining the fragmentation are exclusive.

In the Derived Fragmentation, though, there is a semi-join involved that adds considerable complexity. Disjointness can be guaranteed if the join graph is simple.

On the other hand, it is needful to check actual tuple values. generally, we do not want a tuple of a member relation to join with two or more tuples of the owner relation in case these tuples are in different fragments of the owner.

2.1.4 Horizontal Partitioning Techniques

Researchers have suggested a variety of systems and partitioning techniques in order to provide the performance and . Some of them have been listed here:

Shahidul Islam, August 2010 [4] suggested a new technique of fragmentation by splitting such relation in a Horizontal way according to locality of precedence of its attributes.

Attribute Locality Precedence (ALP) can be defined as the value of importance of an attribute with respect to sites of a distributed database. The database designer constructs the ALP table for each relation of a DDBMS at the time of designing the database with using modified CRUD (Create, Read, Update, and Delete) matrix and cost functions. They produce a block diagram for their system fragmentation of the relations in case of relational database or classes just in case of object-oriented databases, allocation and replication of the fragments in many sites of the distributed system, and local optimization in each site.

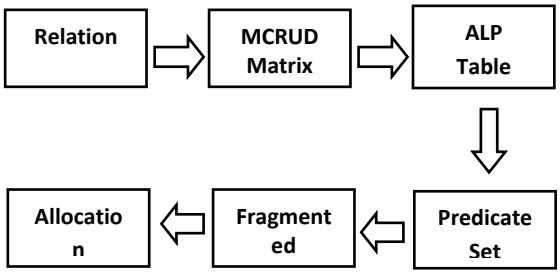

Fig 2 : Block diagram of the system [4]

Mohammed Ibrahim in 2011 [5] proposed customized ISUD (Insert, Select, Update, Delete) technique (5- layer architecture) which is used efficiently as one of the solutions for the database fragmentation in a distributed environment. This customized ISUD application or user interface facilitates to calculate the total cost of an attribute from different sites and also calculate individual cost of an attribute with respect to defined predicate at the nominated site. One of the main objectives of this proposed customized ISUD technique is to show the highest precedence value of the attribute (ALP value) in graphic form, it also motivates the database administrator or end-users to take decisions for fragmenting the relations at the initial stage of the distributed database environment. Thus by observing the graphical statistics of

ALP (Attribute locality precedence) table, we can easily evaluate or measure the performance of the algorithm by having different operational changes of inputs in ISUD frequencies.

Van Nghia Luong, Ha Huy Cuong Nguyen And Van Son Le In 2014 [6] proposed an algorithm that builds the initial equivalence relation that based upon the distance threshold. This threshold is also based on the techniques of knowledge-oriented clustering for both of Horizontal and Vertical fragmentation. In the Algorithm they used the Similarity measures. Experimental results are carrying on the small data set match fragmented results based on the traditional algorithm. During the algorithm the Execution time and data fragmentation significantly reduced while the complexity of their algorithm is stable.

Rizik M. H. Al-Sayyed1, in 2014 [7] proposed a new approach that attains efficiently and effectively the objectives of the data fragmentation, data allocation and network sites clustering. The suggestion plays a role on splitting the data relations into pair-wise disjoint fragments by using Horizontal fragmentation technique and determine whether each fragment should be allocated or not in the network sites, wherever allocation advantage outweighs the cost relying on the high-performance clustering technique. To show the performance of the proposed suggestion, they performed experimental studies on real database application at many networks connectivity. The achieved results proved to achieve minimum total data transaction costs between different sites, it reduces the amount of redundant data to be reached between these sites and improved the overall DDBMS performance.

$$R = R_1 \rhd\lhd R_2 \rhd\lhd .... \rhd\lhd R_m$$

Ms.P.R.Bhuyar on May 2015 [8] offered a Horizontal fragmentation technique of a relation according to the locality of precedence of its attributes. Making suitable fragmentation of the relations and allocation of the fragments is the main research area in distributed databases. Many techniques have been suggested by the researchers using empirical knowledge of data access and query frequencies. But the suitable fragmentation and allocation at the initial stage of a distributed database have not yet been addressed. In this suggestion, they showed that a fragmentation technique to split relations of a distributed database correctly at the initial stage when no data access statistics and query execution frequencies are available. Using such technique, there is no additional complexity added for fragment allocation to the sites of a distributed database like fragmentation is synchronized with allocation. So the performance of a DDBMS can be improved significantly by avoiding frequent remote access and high data transfer among the sites.

2.2 Vertical Fragmentation

Of a relation R consists of grouping attributes together which they are used by some applications [2]. The output of the Vertical Fragments contains groups of various attributes but having the same tuple. For each fragment that is resulted has to contain the primary key so as to be able to rebuild the global relation R. The Vertical Fragmentation has specific operator is a projection. Vertical Fragmentation has two significant ways:

By attributes partitioning or by attributes grouping.

First by attribute partitioning fragmentation is that we have attribute to keep the link between fragments, then the output fragments will be distinct. Furthermore, for attribute grouping fragmentation, the output fragments have to have at least one common attribute. This is affecting the update process which becomes more difficult due to this necessary redundancy. Assume that R is a relation having the attributes {A1, A2,....., An}. The classification of Vertical Fragmentation based on attributes set {Aj, Aj+1,....., Ak} should be written as the following:

$R_i = \pi A_j, A_{j+1}, \ldots\ldots\ldots A_k.(R), 1 \le j \le n, 1 \le k \le n$ **Where**

- π **projection operator;**

$A_j, A_{j+1}, \ldots\ldots\ldots A_k$ **.** - **Fragmentation attributes set;**

R_i - **Output of Vertical fragment**

The reconstruction of the general relation R is attained by joining up the resulted fragments as follows:

An optimal Vertical fragmentation is the one that the application in it can be access only one fragment locally stored. Such apps which need data from two or more Vertical fragments stored in various sites – but they are slower than expected because of requesting many join operation between fragments through many sites.

Figure2 presents the PROJ relation partitioned in a Vertical way into two sub-relations, PROJ1 and PROJ2.

PROJ1 contains the information about project budgets, whereas PROJ2 contains project names and locations. It is important to notice that the primary key to the relation (PNO) is included in both fragments.

The fragmentation method, of course, is nested. If the nesting is of different types, one gets hybrid fragmentation. Even though we do not treat hybrid fragmentation as a primitive fragmentation strategy, many real-life partitioning may be a hybrid.

Emp

ENO	ENAME	Title
E1	J.Doe	Elect.Eng
E2	M.Smith	Syst.Anal.
E3	A.Lee	MechEng
E4	J.Miller	Program.
E5	B.Casey	Syst.Anal.
E6	L.Chu	Elect.Eng
E7	R.Davis	Mech.Eng.
E8	J.Jones	Syst.Anal

ASG

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E8	P3	Manager	40

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

PAY

Title	SAL
Elect.Eng	40000
Syst.Anal.	34000
Mech.Eng.	27000
Programm.	24000

Fig.3 : Vertical Fragmentation [1]

2.2.1 Checking for Correctness

2.2.1.1 Completeness.

Completeness is warranted by the PARTITION algorithm since in case attribute of the general relation is assigned to one of the fragments. As attributes A over which the relation R is defined consists of

$$A = \bigcup R_i$$

$$R = \triangleright\triangleleft_k R_i, \forall R_i \in F_R$$

$$\sigma_n(\Pi_{a1,.....an}(R))$$

2.2.1.2 Disjointness

As indicated previously, the disjointness of fragments is not significant in Vertical Fragmentation as in Horizontal Fragmentation. here there exist two cases:

1. TIDs are used, in case the fragments are disjoint since the TIDs that are replicated in each fragment

2- The key attributes are replicated in each fragment, in case one cannot completeness of Vertical Fragmentation is ensured.

$$R = \triangleright\triangleleft_k R_i, \forall R_i \in F_R$$

2.2.1.3 Re-construction.

The reconstruction of the original general relation is possibly made by the join operation. Therefore, a relation R with Vertical fragmentation

FR = {R1;R2;:::;Rr} and key attribute(s) K,

2.2.2 Vertical Partitioning Techniques

Eltayeb Salih Abuelyaman On January 2008 [9] proposed a scheme for Vertical partitioning of a database at the design cycle. in case a partition is formed, attributes are distributed among various systems or even throughout many geographical locations. This may result in cases where a query may include attributes that are located at various sites. The scheme sets the hit ratio of a partition. Therefore it falls below a predetermined threshold, the partition is changed. Though no proof is provided, experimental data showed that moving an attribute that is loosely coupled to a different subset in a partition makes efficient hit ratio. The simulator was built to test the suggested algorithm. Results of various simulation runs are coordinated with the hypothesis. That is, the suggested algorithm enables a reliable distribution of newly designed database tables through various storage devices based upon a predetermined hit ratio.

The significant advantage of the suggested algorithm is that a database designer doesn't need to wait for experimental data on query frequencies before partitioning a database.

Shahidul Islam In 2010 [10] provided a fragmentation technique which can be applied at the initial stage of database design of distributed database system. They have proposed a single algorithm for both fragmentation and allocation which can be done simultaneously. They have said that this technique can be used for initial fragmentation problem of relational database for any distributed database systems (Horizontal, Vertical or Hybrid Fragmentation).

Van Nghia Luong, In 2015 [11] proposed an algorithm that constructs the initial equivalence relation based upon the distance threshold. Such threshold is based on knowledge-oriented clustering techniques for both Horizontal and Vertical fragmentation. Similarity measures that are used in the algorithms are the measures that are developed from the classical measures. Experimental results performed on the small data set match fragmented results based mainly on the classical algorithm. Execution time and data fragmentation significantly reduced however the complexity of our algorithm in the general case is fixed.

Tejashri S.Nimbalkar, Sep.-2016 [12] used Vertical Fragmentation to propose a novel Integrated Fragmentation clustering allocation approach for increasing care admissions and decrease care difficulties on the suggestion that manages the web service computing that is demanded to promote telemedicine database system performance. Such suggestion focused on large-scale networks involving a large number of sites over the cloud. To perform more intelligent data redistribution, they apply various types of clustering algorithms and introduce search-based techniques. The security concerns that are needed for addressing over data fragments will be taken into consideration for better results.

2.3 Hybrid Fragmentation

The mixed fragment can be attained by getting Vertical Fragmentation of a Horizontal Fragmentation of Such relation R or via Horizontal Fragmentation of a Vertical Fragmentation of such relation R. this fragmentation type can be attained by using the selection and projection operators as following:

$$\sigma_p(\Pi_{a1,.....an}(R)) \text{ or}$$

$$\Pi_{a1.....an}(\sigma_p(R))$$

2.3.1 Hybrid Partitioning Techniques

S. Jagannatha1 , In 2011 [13] presented a UML model, which assists in representing the fragmentation of Distributed Databases. The more precision about the frequency of a transaction gives more visibility for mixed fragmentation. In this suggestion they attempted to present a UML2.0 based on modeling for Distributed Databases Design Fragmentation. It explains the validity of the model by using a case study using the notion of attribute usage matrix, predicate usage matrix. Based upon these it is simple to determine the set of use cases that can be mixed fragmented. They suggested simulating mixed fragmentation in the distributed database design and estimate the performance.

described architecture Fragmentation, allocation. Fragment Allocation method is designed to face the requirements of clustering sites and sets the fragment allocation in a distributed database system, reducing the communication cost between sites, and improving the performance in a heterogeneous network environment system. The clustering method is developed mainly to gather the sites into clusters, which assists in reducing the communication costs between the sites over the allocation process. Fragment Allocation Method is developed to improve system performance by increasing the availability and reliability wherever various copies of the same fragments are allocated.

3. DISCUSSION

Table 1 shows the summary of the classification of fragmentation methods techniques presented by the papers authors , according to this discussion it is clear that papers [4,5,6,7,8] present Horizontal Fragmentation and shows that which one achieves the correctness rules and improves the performance though papers [9,10,11,12] present Vertical Fragmentation and also shows which one achieves all the correctness rules.

4. CONCLUSION

This Paper discussed, in details, the algorithms that anyone can use in order to fragment a relational schema in different ways. Such algorithms have been developed independently and there is no certain design methodology that merges the

Horizontal and Vertical partitioning techniques. If one starts with general relation, there exist algorithms to decompose it horizontally as well as algorithms to decompose it vertically into a set of fragment relations. While, there are no algorithms that can fragment a general relation into a group of fragment relations some of which are decomposed in a Horizontal way and the others is Vertical. It is usually indicated that most real-life fragmentations would be mixed, so, would involve Horizontal and Vertical partitioning of a relation, but such methodology of the research is to complete or achieve this is lacking. Thus, that s a distribution design methodology which involves the Horizontal and Vertical Fragmentation Algorithms and uses them as a part of a general strategy. This methodology should take the general relation together with a set of design criteria and get with a group of fragments some of which are attained by Horizontal and others obtained by Vertical Fragmentation.

One direction of future work is to test the optimization opportunities of fragmentation models. It would also be nice to determine what are the other optimizations are possible for a hybrid of vertical and horizontal fragmentation and how we can determine hybrid fragmentation schemas automatically.

5. REFERENCES

[1] Shahidul Islam Khan "A New Fragmentation and Allocation Technique for Distributed atabase System" ,Database Systems Journal vol. VII, PP 34-51. 2/2016

[2] Anca-Georgiana Fodor ,Ion Lungu, "Implementation Of Fragmentation And Replication Methods In Distributed Systems", Journal of Information Systems & Operations Management, Vol. 10 Issue 2, Pp373-383. 11p. 2016.

[3] Aakanksha Jumle* and Swati Ahirrao," A Survey of Different Data Partitioning Techniques" I J C T A, 10(8), pp. 535-541, 2017

[4] Shahidul Islam Khan and A. S. M. Latiful Hoque, "A New Technique for Database Fragmentation in Distributed Systems", International Journal of Computer Applications (0975 – 8887) Volume 5– No.9, 2010

[5] Mohammed Ibrahim Shareef And Wail Al-Rawi, "The Customized Database Fragmentation Technique in Distributed Database Systems." Computer and Electrical Engineering; University of Jönköping / JTH, Computer and Electrical Engineering, 2011

[6] Van Nghia Loung , Ha Huy Cuong Nguyen and Van Son Le," An improvement on fragmentation in DistributionDatabase Design Based on Knowledge-Oriented Clustering Techniques" , International Journal of Computer Applications (0975 – 8887) Volume 5– No.9, August 2010

[7] Rizik M. H. Al-Sayyed1, Fawaz A. Al Zaghoul1, Dima Suleiman1, Mariam triq1,Ismail Hababeh "A New Approach for Database Fragmentation and Allocation to Improve the Distributed Database Management System Performance. " , Journal of Software Engineering and Applications, 891-905 Published SciRes, October 2014.

[8] Ms.P.R.Bhuyar, Dr.A.D.Gawande and Prof. A.B.Deshmukh "Horizontal Fragmentation Technique in Distributed Database", International Journal of Scientific

and Research Publications, Volume 2, Issue 5 ,May 2012

[9] Eltayeb Salih Abuelyaman," An Optimized Scheme for Vertical Partitioning of a Distributed Database." IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.1, January 2008.

[10] Shahidul Islam Khan and Dr. A. S. M. Latiful Hoque " Efficient Partitioning of Large Databases without Query Statistics " Database Systems Journal vol. VII, 2016

[11] Van Nghia Loung , Ha Huy Cuong Nguyen and Van Son Le In 2015," An improvement on fragmentation in DistributionDatabase Design Based on Knowledge-Oriented Clustering Techniques" , International Journal of Computer Applications (0975 – 8887) Volume 5– No.9, August 2010

[12] Tejashri S. Nimbalkar, Nagaraju Bogiri ,"Integrated Fragmentation Clustering Allocation Approach For Promote WebTelemedicine Database System " , International Journal of Advances in Electronics and Computer Science, Volume-3, Issue-9, Sep.-2016

[13] S. Jagannatha1 , M.Mrunalini1 , T V Suresh Kumar1 , K Rajani Kanth1 1 M S Ramaiah, "Modeling of Mixed Fragmentation in Distributed Database Using UML 2.0 ", International Conference on Computer Engineering and Applications IPCSIT vol.2 IACSIT Press, Singapore 2011.

6. APPENDIX

Table 1 summary of approaches of the Fragmentation methods technique

Approaches	MD	SD	horizontal	Vertical	Hybrid	Performance	correctness rules	source data
Shahidul Islam Khan and A. S. M. Latiful Hoque	√	-	√	-	-	Improved	Disjointness Completeness	Stored
Mohammed Ibrahim Shareef And Wail Al-Rawi	√	-	√	-	-	Improved	Disjointness Completeness Reconstruction	Stored
Van Nghia Luong , Hahhuy Cuong Nguyen and Vanson LE	√	-	√		-	Improved	Disjointness Completeness Reconstruction	Stored
Rizik M. H. Al-Sayyed1, Fawaz A. Al Zaghoul1, Dima Suleiman1, Mariam Itriq1, Ismail Hababeh	√	-			-	Improved	Compleetnes Reconstruction	Stored
Ms.P.R.Bhuyar, Dr.A.D.Gawande and Prof. A.B.Deshmukh		-				Improved	Disjointness Reconstruction	Stored
Eltayeb Salih Abuelyaman		√		√	-	Improved	Disjointness Completeness Reconstruction	Stored
Shahidul Islam Khan and Dr. A. S. M. Latiful Hoque	√	-		√		Improved	Disjointness	Stored
VAN NGHIA LUONG, HA HUY CUONG NGUYEN and VAN SON LE	√	-				improved	Disjointness	Stored
Tejashri S.Nimbalkar	√	-		√	-	improved	Disjointness Reconstruction	Stored
S. Jagannatha1 , M.Mrunalini1 , T V Suresh Kumar1 , K Rajani Kanth1 1 M S Ramaiah		-			√	improved	Disjointness Reconstruction	Stored