**Om Prakash Dhakal Analysis**

Results were analyzed through Google form, which is a demographic tool to can represent data in various demographic representation(charts, bars etc.).Data were also analyzed through Google spreadsheets after export from Google form. Each question response was represented through pie charts in percentage format.

**Results**

The results based on breakdown of each research question as follows:

**Research question one findings**

The first research question explored the data failure conditions in distributed database system. Different questions were asked to get this result. The first survey question was related with the component which involves to fail data in this system. The component result, which involved to fail data in distributed database indicated that 50% through software,27.3% through network and 22.7% through hardware. Architecture of database result indicated that 54.5% failed data through heterogeneous, 36.4% failed through data through hybrid and 9.1% failed through homogenous architecture. The common data failure type result indicated that 40.9% by software, 31.8% by network and 27.3% by hardware. Network congestion can problem or not?, result indicated that 59.1% said yes, 31.8% said may be and 9.1% said no. The result about operating system role showed that 63.6% can affect to fail data rest of the percentage doesn’t affect. Connection of nodes should care in distributed system to protect data according to the result.

**Research question two findings**

The second question explored the condition of deadlock in distributed system. The survey question related with that about the state of deadlock creation result indicated that 63.6% through network congestion, 18.2% through data slicing and same percentage through data fragmentation. Transaction graph create deadlock or not?, The result indicated that 59.1% through global wait for graph , 27.3% through local wait for graph and 13.6% through recall wait for graph. In distributed deadlock prevention approach, a transaction should acquire all the locks before starting to execute, the result showed that 45.5% strongly agreed, 40.9% agreed and 13.6% neither agreed nor disagreed. The result showed about communication delay to create deadlock, 63.6% response was Maybe, 18.2% response was Yes, 9.1% response was I don’t know. The result of how important resources in deadlock management’s result showed that 90.9% said very important, 9.1% said not very important. A tricky question was asked in this section which was “A system has 3 processes during sharing 4 resources. If each process needs a maximum of 2 units then, deadlock”, result indicated that 54.5% may occur deadlock at that situation, 36.4% may occur deadlock and 9.1% has to occur.

**Research question three findings**

Third research question explored about the how affect the data recovery process in distributed system. The first question related about the strategy of recover data in distributed database system, the result indicated that 59.1% recover strategy includes transaction undo or rollback and 40.9% recovery strategies encompass restoring a pass of the database from archival backup. Second survey question related with this research question was what kind of action can take recovery manager during immediate update mode, the result indicated that 54.5% of transactions which are in before-commit list are redone, 27.3% of transaction which are in active list and failed list are undone and 18.2% had no action taken for transaction in commit or abort lists. The result about the type of effective replication method in recovery process indicated that 68.2% through asynchronous replication and 31.8% through synchronous replication. Check pointing is important or not during data recovery process, in this kind of scenario, the result showed that 50% said important, 31.8% extremely important and 18.2 said not important. The result of redo/undo recovery method using for data recovery process demonstrated that 63.6% said yes, 18.2% said no and same percentage said may be. The purpose of checkpoint, 45.5% to reduce the workload on the log writer process, 22.7% by update data file and control file log sequence number, 18.2% by trigger a checkpoint and 13.6% write a transaction commit entry. And the final survey question was about which backup technique is space efficient, result indicated that 45.5% through incremental backup, 31.8% through differential backup and 22.7% through full backup.