**PROJECT PROPOSAL**

DFS Deployment In Windows Failover Cluster

**CT-509 Distributed Systems   
NED UET Fall 2018**

**By: Shaikh Muhammad Siddique  
ROLL NO. IS-103**

**Subject: Distributed System**

**Abstract**

In the present world of Business and commerce, continuity and smoothness of operation is significantly very important.

Business and communication going all over the globe requires a fast, reliable, robust communication and sharing of information with high availability and minimum downtimes i.e failovers or unavailability.

There are number of technologies present to prevent those obstructions hindering business communication and operations among which one of the technology is put under consideration in this very project report.

We are going to demonstrate deployment of DFS on Windows Failover Cluster.

On Windows Server 2008 R2, a Windows Failover cluster can be configured to be a member of a DFSR replication group. This feature can be used to configure highly available replication services. In this demonstrative project let us examine how to configure a Windows Failover cluster as a DFS Replication member server. For a quick recap of the new features in DFS Replication on Windows Server 2008 R2.

The first step is to validate the available hardware that will be used for clustering and to create a Windows Failover Cluster. Then we will cover steps to configure high availability for the DFS Replication service. The third and final part will covers the steps required to add the failover cluster to a replication group.

**Problem Statement**

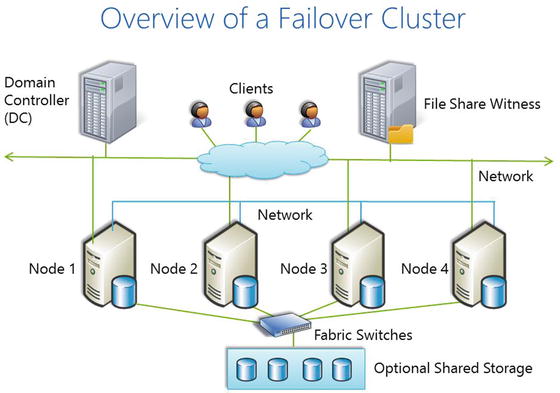
High-availability clusters (also known as HA clusters or fail-over clusters) are groups of [computers](https://en.wikipedia.org/wiki/Computer) that support [server](https://en.wikipedia.org/wiki/Server_(computing)) [applications](https://en.wikipedia.org/wiki/Application_software) that can be reliably utilized with [a minimum amount of down-time](https://en.wikipedia.org/wiki/High_availability). They operate by using [high availability software](https://en.wikipedia.org/wiki/High_availability_software) to harness [redundant](https://en.wikipedia.org/wiki/Redundancy_(engineering)) computers in groups or [clusters](https://en.wikipedia.org/wiki/Computer_cluster) that provide continued service when system components fail. Without clustering, if a server running a particular application crashes, the application will be unavailable until the crashed server is fixed. HA clustering remedies this situation by detecting hardware/software faults, and immediately restarting the application on another system without requiring administrative intervention, a process known as [failover](https://en.wikipedia.org/wiki/Failover).

**Problem Solution**

A failover cluster is a group of servers that work together to maintain [high availability](https://searchdatacenter.techtarget.com/definition/high-availability) of applications and services. If one of the servers, or [nodes](https://searchnetworking.techtarget.com/definition/node), fails, another node in the cluster can take over its workload without any [downtime](https://whatis.techtarget.com/definition/uptime-and-downtime).

Advantages of Windows Failover Clustering

* Clustering reduces the impact of hardware outages
* Clustering can be a form of disaster recovery
* Clustering takes the pain out of software problems
* Clustering makes OS patching less painful



**Project Scope:**

This Project will help us in understanding the overall idea about High Availability or Failover clustering environment and its implementation. This basic components involved in establishing the H.A cluster and its advantages with respect to Enterprise environment perspective.

**Selected Papers:**

*White Paper:* Enterprise Deployment - Failover Clustering with Laserfiche 8

**Web-Links for Reference:**

1. <https://searchwindowsserver.techtarget.com/tip/When-and-when-not-to-use-Windows-server-failover-clustering>
2. <https://en.wikipedia.org/wiki/High-availability_cluster>
3. <https://docs.microsoft.com/en-us/windows-server/failover-clustering/whats-new-in-failover-clustering>