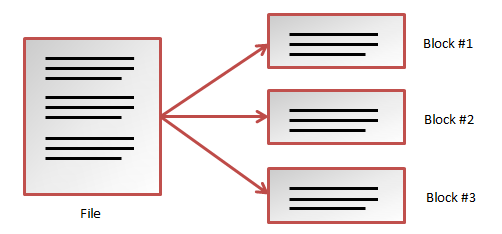
# HDFS (Hadoop Distributed File System)

Son yıllarda toplanan verilerin devasa boyutlara ulaşmasından dolayı verileri tek bir makine yerine birden fazla makineda saklama çözümleri ortaya atılmıştır.(distributed filesystems)

Bu noktada Hadoop projesinde, verileri birden fazla makinede saklayan HDFS kütüphanesi geliştirilmiştir

## Genel Özellikleri

* Petabyte seviyesindeki büyük verileri saklayabilir
* Pahalı bir donanım satın almanıza gerek yoktur. Günlük hayatta kullandığımız bir kaç makine ile Hadoop cluster kurabiliriz
* HDFS içerisine veriler kopyalandıktan sonra birçok kaynak üzerinden aynı anda erişim sağlanabilir
* Verilere hızlı bir erişim sunar(Low-latency data access)
* Veriler küçük dosya blokları halinde saklanır



## Bloklar (Blocks)

HDFS üzerinde veriler bloklar halinde saklanır. Varsayılan değeri ise 128 MB’dir ve kullanıcı tarafından değiştirilebilir

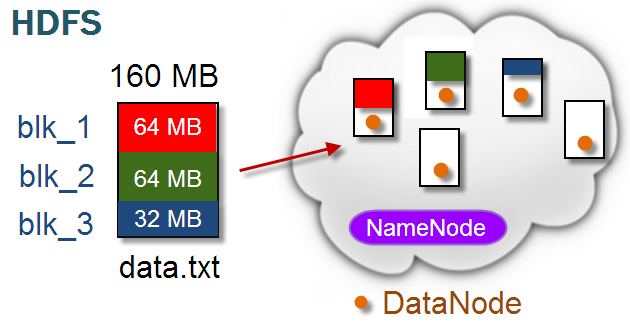
Blok yapısının avantajları şunlardır;

Yüklemek istediğimiz dosya boyutları disk boyutundan büyük olabilir.Böylece verileri bloklara ayırarak farklı diskler üzerinde kopyalanmasını sağlamış oluruz

Disklerden bir tanesinde hata meydana gelirse, o disk üzerindeki bloklar farklı makinelere kopyalanır

Blok boyutu çok küçük olduğu (512 byte gibi) durumda disk üzerinde arama işlemi maliyetli olduğu için Hadoop işlemleri yavaşlar.Bu yüzden blok boyutu ihtiyaca göre 64 MB,128 MB verilebilir

Ayrıca blok boyutu çok büyük olursa(1 gb gibi) birden fazla disk üzerinden okuma yapmak yerine tek bir diskten okuma yapılacağı için yine Hadoop işlemleri yavaşlar



## Namenode ve Datanode

Namenode genel olarak verilerin nerede saklandığı bilgisini tutar. Hadoop sisteminde master olarak düşünebiliriz.Datanode ise verilerin saklandığı makinalardır(slave)

Datanode makineleri belirli periyotlarda hangi blokları sakladıklarını Namenode’a bildirir.

Eğer namenode makinesinde bir hata meydana gelirse Hadoop içerisindeki verilere erişemeyiz.



## HDFS Komutları

### Mkdir : hdfs dfs -mkdir /kaynak

Bu komut ile HDFS dosya sisteminde bir klasör oluşturulabilir

### CopyFromLocal : hdfs dfs -copyFromLocal /kaynak /hdfs-hedef

Bu komut ile local makinedeki dosyalar HDFS dosya sistemine kopyalanabilir

### CopyToLocal : hdfs dfs -copyToLocal /hdfs-kaynak /hedef

Bu komut ile HDFS sistemindeki dosyalar local ortama kopyalanabilir

### Count : hdfs dfs -count /klasör/\*

Bu komut ile hdfs sistemindeki dosya sayısını yazdırabiliriz

### Cat : hdfs dfs -cat /klasör/dosya

Bu komut ile HDFS sistemindeki bir dosyanın içeriğini gösterebiliriz

### Cp : hdfs dfs -cp /kaynak/dosya /hedef

Bu komut ile HDFS sistemindeki bir dosyayı , HDFS içerisinde başka bir klasör altına kopyalayabiliriz

### Rmr : hdfs dfs -rmr /kaynak/\*

Bu komut ile HDFS sisteminden dosyaları silebiliriz

### Mv : hdfs dfs -mv /kaynak/dosya /hedef

Bu komut ile HDFS sistemindeki bir dosyayı , HDFS sistemindeki başka bir klasör altına taşıyabiliriz

### Ls : hdfs dfs -ls /klasör

Bu komut ile HDFS sistemindeki dosyaları listeleyebiliriz

### Tail : hdfs dfs -tail /klasör/dosya

Bu komut ile HDFS sistemindeki bir dosyanın son satırlarını listeleyebiliriz

### Du : hdfs dfs -du /klasör/\*

Bu komut ile HDFS sistemindeki dosyaların boyutlarını listeleyebiliriz

### Chown : hdfs dfs -chown -R owner:group /klasör/\*

Bu komut ile HDFS sistemindeki dosyaların kullanıcı yetkilerini ayarlayabiliriz

### Chmod : hdfs dfs -chmod +x /klasör/\*

Bu komut ile HDFS sistemindeki dosyaların okuma,yazma,çalıştırma yetkilerini değiştirebiliriz

### Setrep : hdfs dfs -setrep 3 -R /klasör/\*

Bu komut ile HDFS sistemindeki dosyaların replication factor değeri değiştirilerbilir.

### Touchz : hdfs dfs -touchz /klasör/dosya

Bu komut ile HDFS sisteminde boş bir dosya oluşturabiliriz