

## Apache Cassandra Nedir?

- Apache Cassandra en basit tanımıyla bir **NoSQL veritabanıdır**.
- İlişkisel veritabanlarına göre daha esnek bir yapıya / veri modeline sahiptir. Özellikle ilişkisel olmayan verilerin depolanması için uygundur.
- Apache Cassandra'nın çoğu avantajını sağlayan özelliği ise **dağıtık** yapıda olmasıdır. Bu yapıda veriler bir **cluster** içerisinde belirlenen kriterlere göre farklı bileşenlerde depolanmaktadır.

## Cassandra'nın Avantajları

- Cassandra aynı özellikte birden fazla node'dan oluşan bir cluster üzerinde veriyi dağıtık (decentralised) bir yapıda depolamaktadır. Bu yapıda, cluster içerisinde bir Master/Slave kurgusu bulunmamaktadır. **Böylece bir veya daha fazla node üzerinde bir hata meydana geldiğinde dahi veriye ulaşamadığımız bir zaman aralığından veya veri kaybından korunmuş oluyoruz.** "Single Point of Failure" dediğimiz tüm sistemin tek bir noktada meydana gelen bir hatadan dolayı çalışamaz hale gelmesi engellenmektedir.
- Master/Slave kurgusu olmadığı için eklenen her node yine diğerleriyle aynı yetkileri ve özellikleri taşımaktadır. Bir başka deyişle yeni eklenen node'lar da mevcut node'lar gibi okuma/yazma yetkilerine sahip olmaktadır. **Böylece yeni eklenen bir node hem veri depolama kapasitesini arttırmakta hem de daha yüksek okuma performansı elde edilmesini sağlamaktadır.** Bu noktada, donanım izin verdiği sürece ölçeklendirme yapmak mümkün hale gelmektedir. Özellikle büyük boyutta veriler üzerinde çalışılan senaryolarda fazla sayıda node ile çözümleme yapmak mantıklı hale gelmektedir.
- Veriler tablolar halinde tutulmaya devam etse dahi bu tablolar sütunlara ayrılarak tutulmaktadır. Bu algoritma/yöntem tek node üzerinde dahi Cassandra'nın **okuma/yazma sürelerinin çok daha hızlı olmasını sağlamaktadır.**
- Cassandra'yı benzerlerinden **ayıran en önemli özelliklerinden biri ise kendi sorgu (query) dili olan Cassandra Query Language (CQL)**'dir. Benzer ürünlerde veriye ulaşmak için JSON formatında sorgular yazmak gerekebilirken, Cassandra'da tek satırlık sorgularla veriye ulaşmak mümkündür. Buna ek olarak, farklı NoSql veritabanlarında sorguların dönüşünde hash table tarzı veri modellemeleri kullanılırken Cassandra direkt olarak sorguda aranan veriyi dönebilmektedir.

## Cassandra'nın Dezavantajları

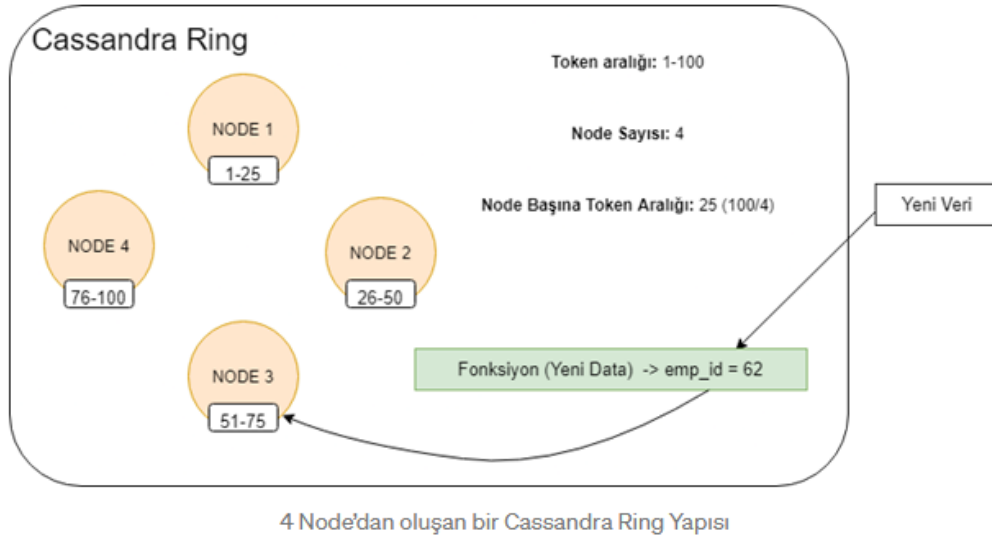
- Cassandra'nın bir **NoSQL veritabanı olması bu dezavantajların** çoğunun temel sebebi olarak gösterilebilir.
- Cassandra, ilişkisel veri tabanlarının aksine **foreign key yapısına sahip değildir.** Basitçe tanımlamak gerekirse bir objenin veya verinin bir diğer tablodaki primary key'ini gösteren foreign key yapısı Cassandra'da bulunmamaktadır. Bu kararın gerekçesi olarak foreign key yapısının sebep olabileceği potansiyel performans kaybı gösterilmektedir.

## Mimari Yapı ve Çalışma Prensipleri

İlk olarak Cassandra'nın 1 numaralı özelliği olan dağıtık yapısından bahsedelim. Cassandra günümüzde birçok araçta da gördüğümüz, farklı donanımlar/cihazlar üzerinde konumlandırılmış birçok node'dan meydana gelen cluster yapısı ile çalışmaktadır. Node sayısının yüksek miktarda olduğu durumlarda ise Cluster içerisinde daha küçük ve yerel Data Center gruplandırmalarından söz etmek mümkündür.

Cluster (Kume)

Cluster yapısını inceleyelim:



Yeni bir veri geldiğinde, Cassandra içerisindeki bir hashing algoritması ile yeni veri için benzersiz bir hash (token)(emp\_id) hesaplanır ve bu hash'in içerisinde bulunduğu token range hangi node üzerinde ise yeni veri bu node üzerinde saklanır.