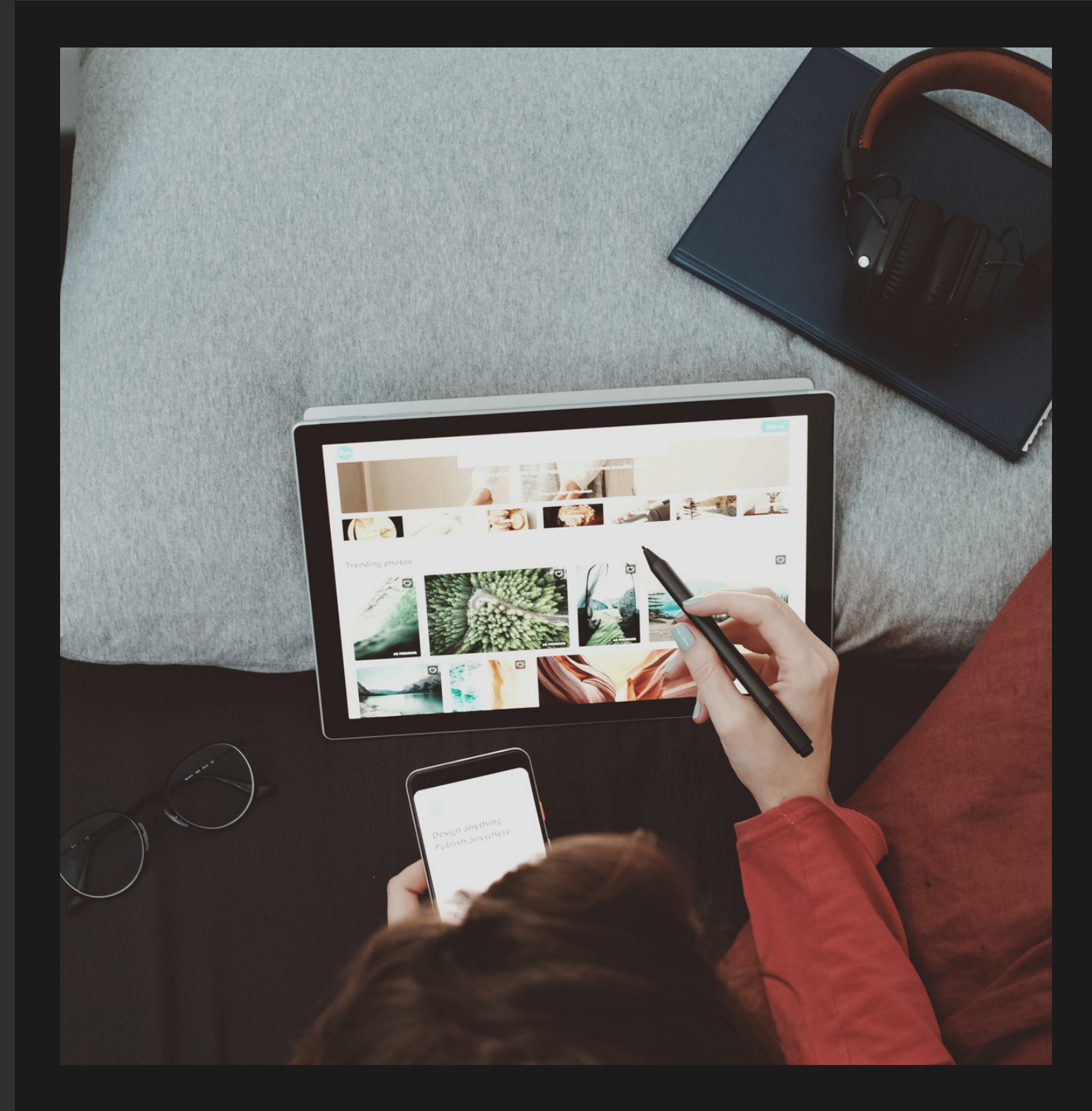


02220224003-BUĞRAHAN KARACA

NoSQL Veri Tabanı

1. Giriş
2. Bilişim Sistemleri ve Yönetimi
3. Veri Tabanı Yönetimi
4. Veri Tabanı Tasarımı
5. NoSQL Sistemi
6. Veri Tabanları Performans Karşılaştırması



Giriş

- Bilgisayar ve iletişim teknolojilerindeki hızlı gelişim, organizasyonları yeni çözümler arayışına yönlendirmiştir.
- Bu süreçte, veri tabanı kullanımı da önemli bir role sahiptir ve ilişkisel olmayan veri tabanı yönetim sistemleri daha fazla tercih edilmiştir.
- İlişkisel olmayan veri tabanı sistemleri birçok açıdan tercih sebebi olmuştur.

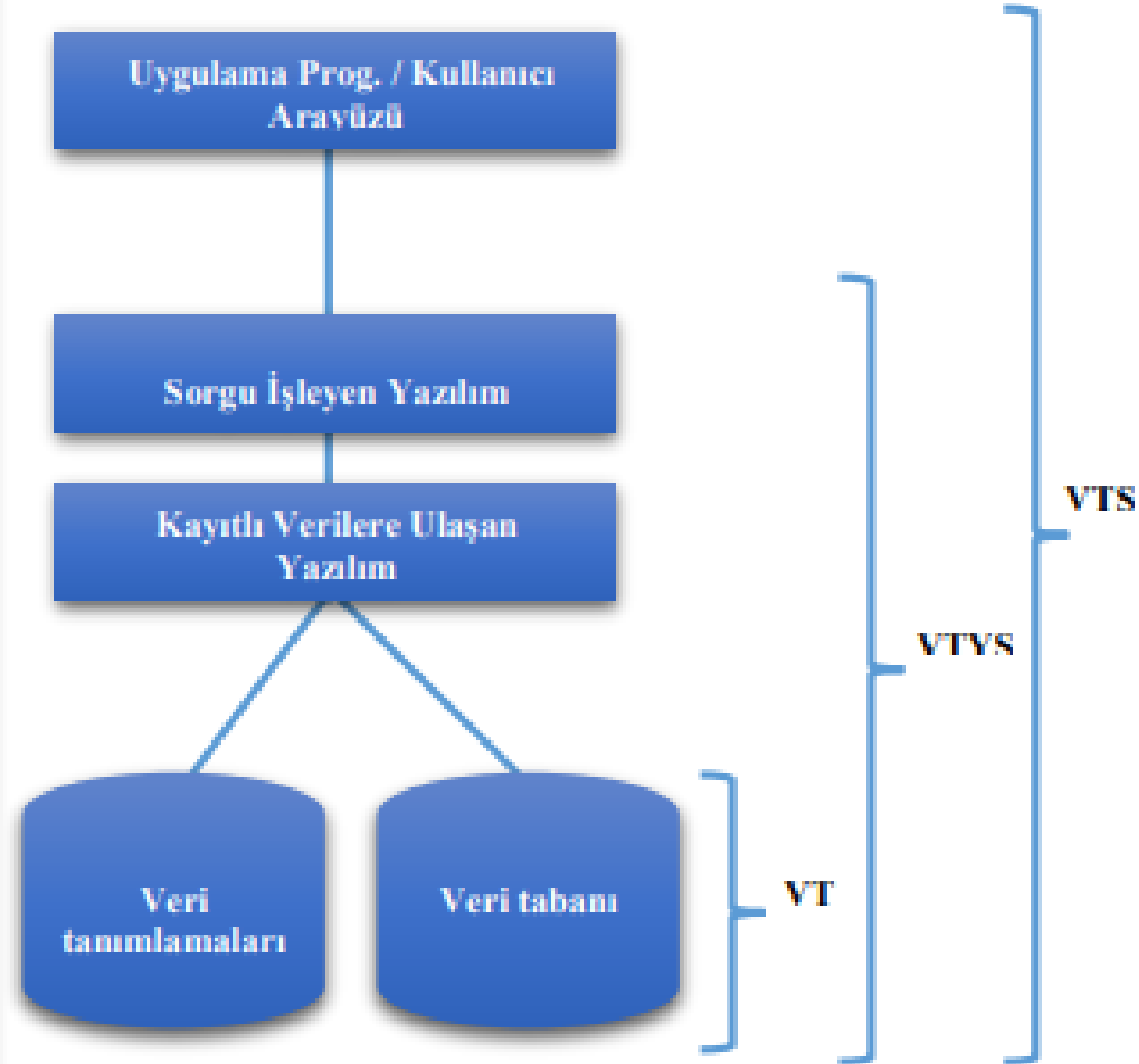
Bilişim Sistemleri ve Yönetim



- Bilişim sistemi, organizasyonlarda bilgiyi toplamak, düzenlemek, işlemek ve saklamak amacıyla kullanılır.
- Bu sistemlerde 3 temel etkinlik vardır. (Girdi-İşlem-Çıktı)
- Girdi, ham veri toplar. İşlem, veriyi anlamlı hale getirir. Çıktı ise bilgiyi aktarma kısmıdır.

Veri Tabanı Yönetimi

- Veri tabanı, belirli bir amaç için düzenlenmiş veri koleksiyonunu temsil eder ve ilişkili verilerin mantıksal ve fiziksel tanımlarını içerir.



Veri tabanı modellerini sekiz
kategoriye ayırabiliriz

Düz Veri Modeli

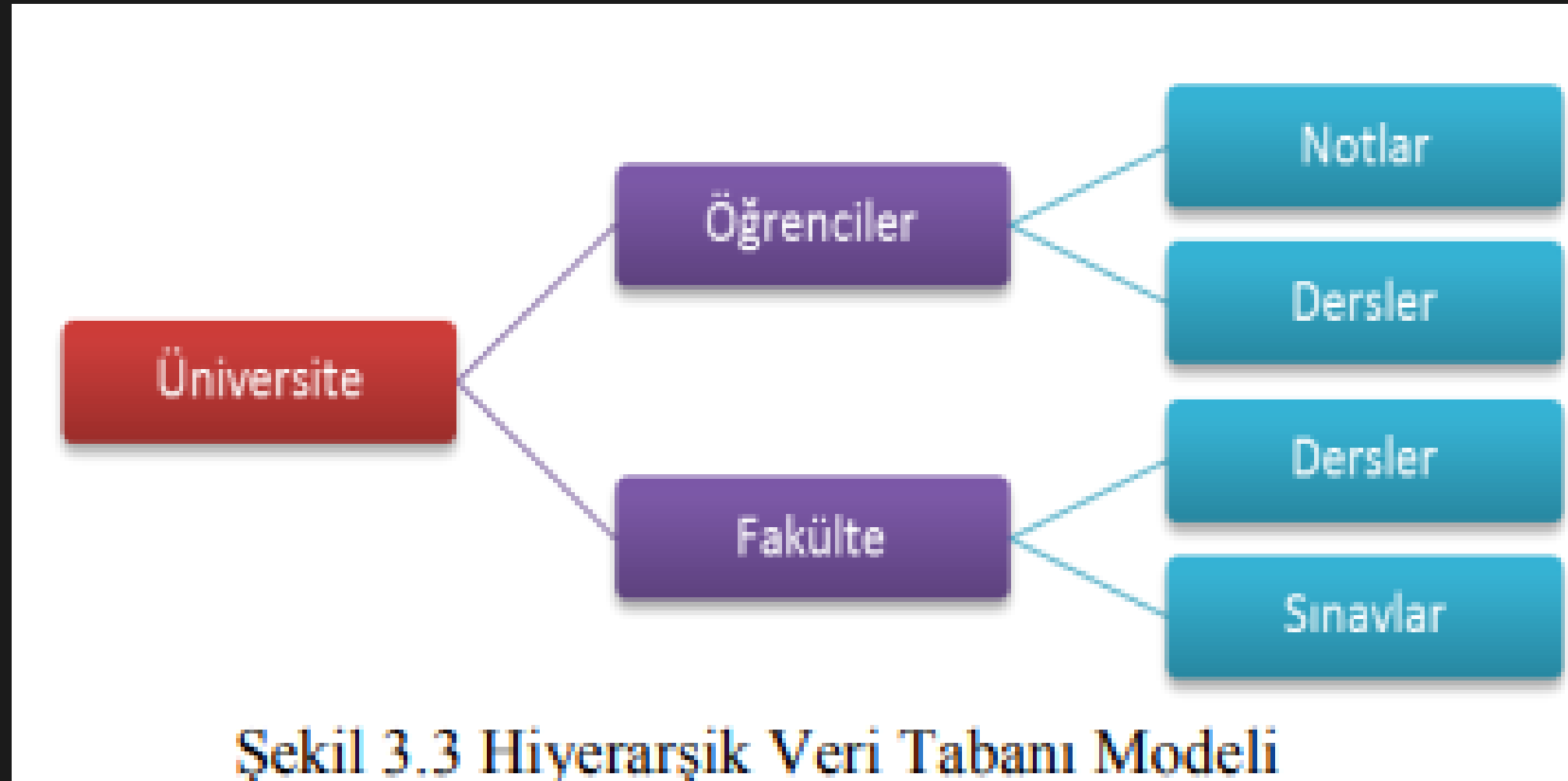
- Sütunlarda benzer özelliklere sahip veriler bulunurken, satırlarda ise veri grupları yer alır.
- Alışlagelmiş basit bir tablo görünümüdür.

	Ad Soyad	Kullanıcı Adı	Parola
Kayıt 1	Murat ERGİN	Mergin	kjVdb125
Kayıt 2	Ayşe YILMAZ	Ayılmaz	Bks46db7
Kayıt 3	Can TÜRK	Cturk	fhG8dbt9

Şekil 3.2 Düz veri modeli örneği

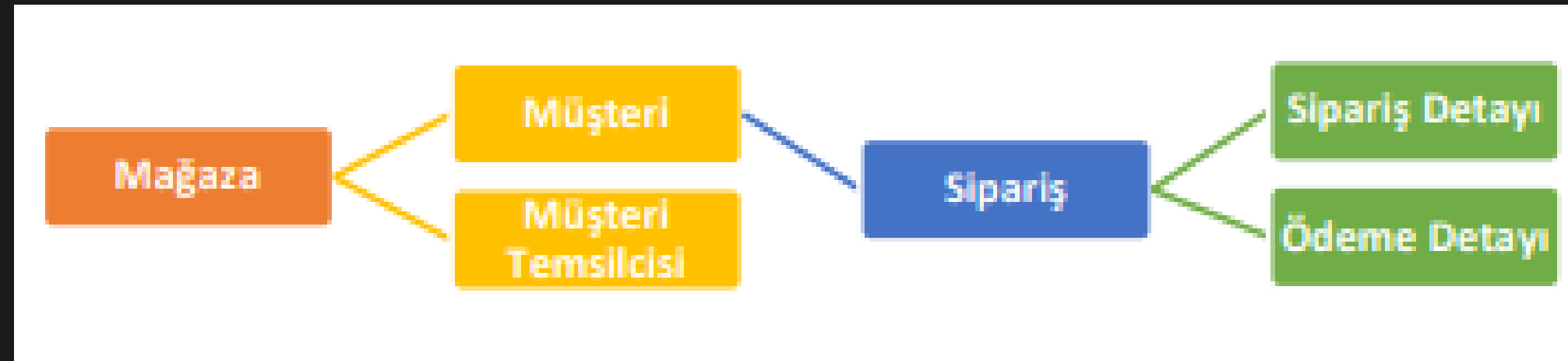
Hiyerarşik Veri Modeli

- 1960'larda ortaya çıkan model, adını yapısından almıştır.
- Ağaç Mimarisiyle aynı yapıya sahiptir.



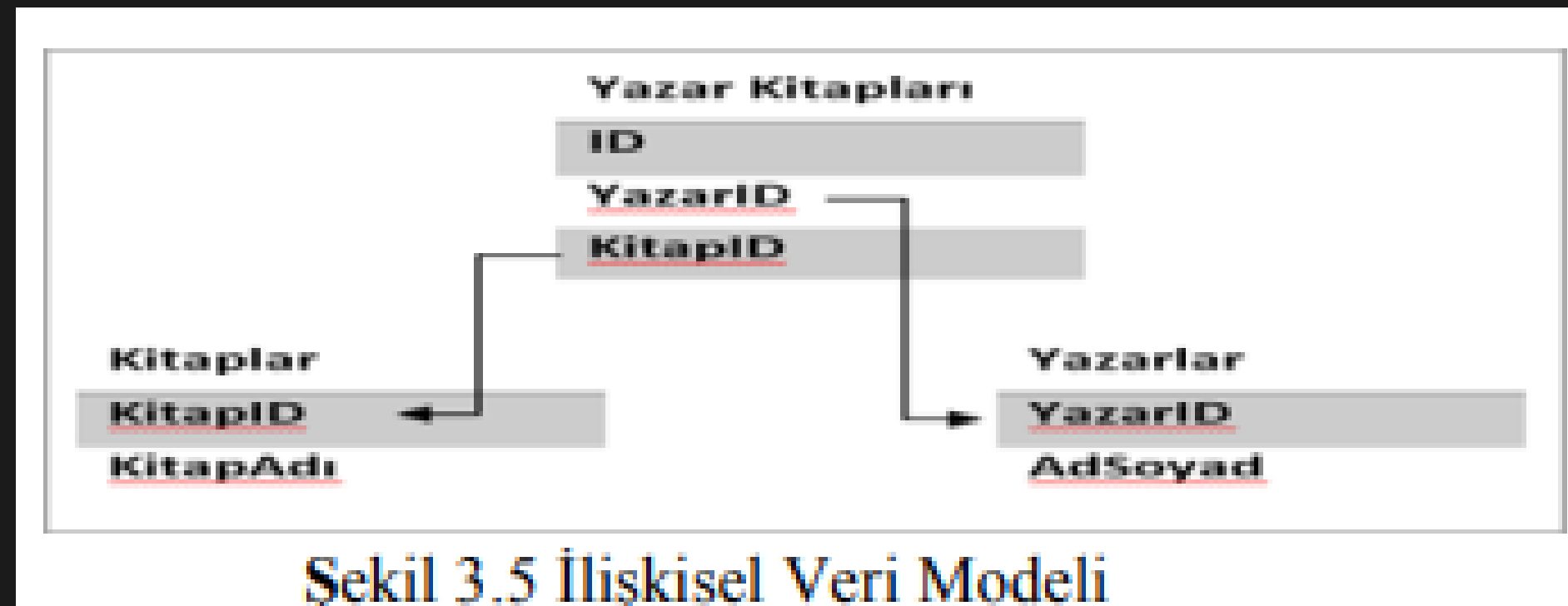
Ağ Veri Modeli

- 1970'lerde ortaya çıkan model, hiyerarşik modelin gelişmiş halidir.
- Veri tekrarını azalttığı için kolayca kabul görmüştür.



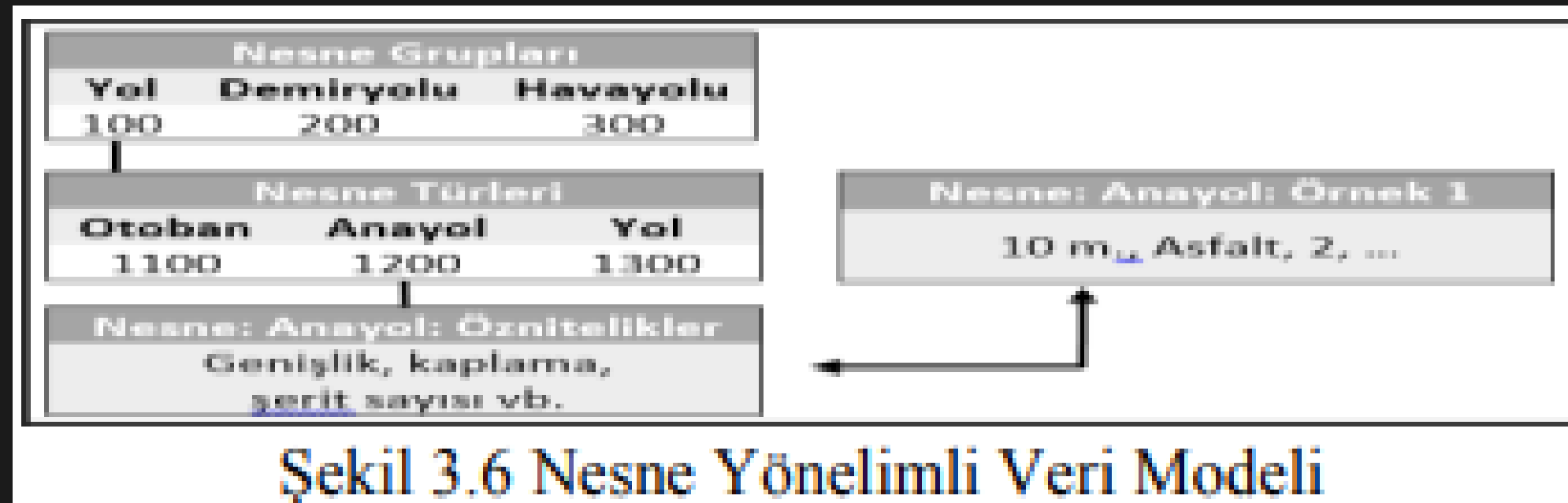
İlişkisel Veri Modeli

- Hiyerarşik ve Ağ veri modellerinin artan gereksinimleri karşılayamamasıyla ortaya çıkan bir modeldir.
- İlişkiler, satır ve sütunlardan oluşan iki boyutlu tablolarla tanımlanır.
- Her tablo genellikle veri tabanında bir dosyaya karşılık gelir



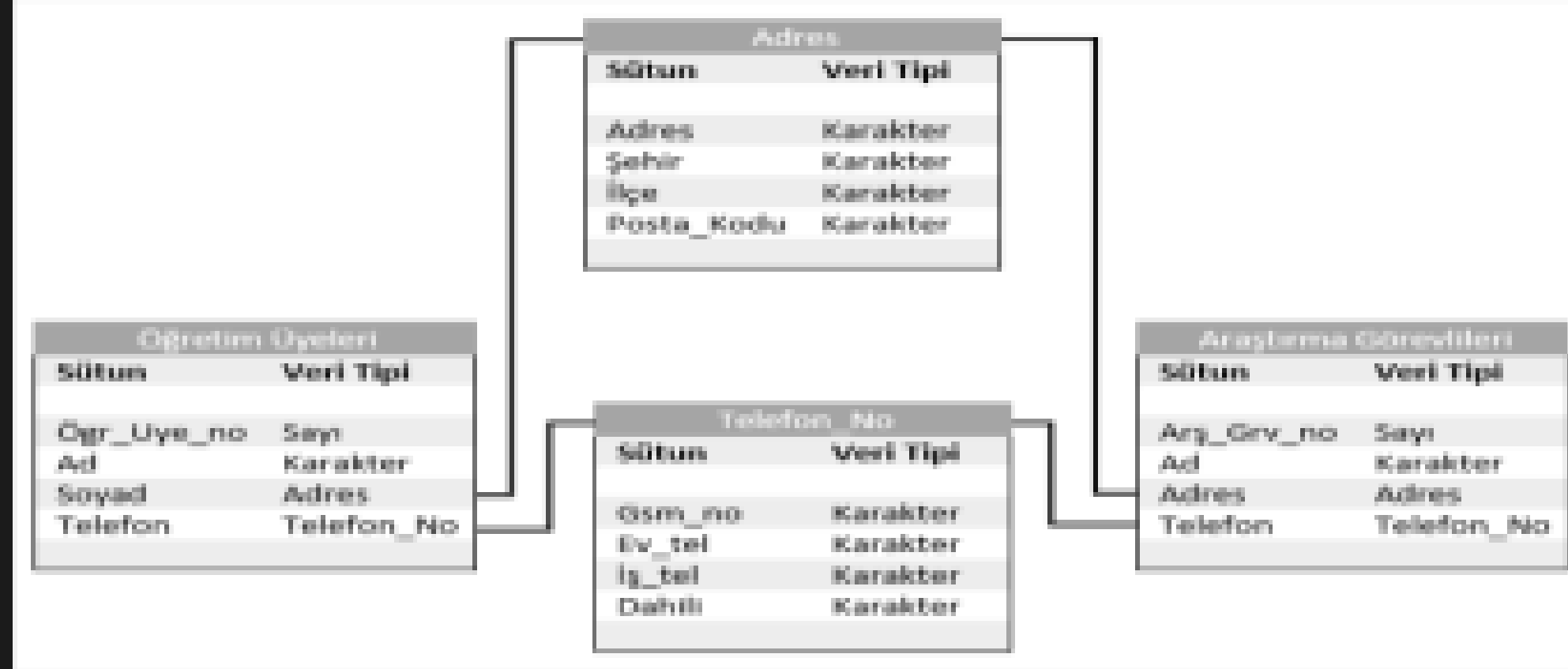
Nesne Yönelimli Veri Modeli

- Nesne yönelimli programlamaya dayalıdır.



Nesne İlişkisel Veri Modeli

- İlişkisel Veri Modelinin nesne yönelimli uygulanmasıdır.



Şekil 3.7 Nesne İlişkisel Veri Modeli [11]

Çoklu Ortam Veri Modeli

- Nesne ilişkisel veri tabanları ile benzerdir.
- Büyük nesneleri işlemek ve kullanıcıdan gizlemek için özellikleri vardır.
- Tıbbi görüntü kayıtları saklamadan dolayı tıp bilgi sisteminde kullanılmaktadır.

Dağıtık Veri Modeli

- En az iki bilgisayar ve üstünde depolanan bilginin bir ağla dağıtımıyla kullanılır.
- Kullanıcı buna rağmen tek bir veri tabanıyla uğraşır.

Veri Tabanı Tasarımı

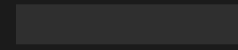
- Veri Tabanı Tasarımı için gerekli aşamalar görselde belirtildiği şekildedir.
- Kullanıcı gereksinimleri belirlenip kavramsal, mantıksal ve fiziksel düzeylerde şemalar oluşturulur.



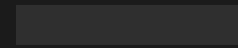
Şekil 4.1 Veri Tabanı Tasarım Aşamaları

İlişkisel Olan Ve Olmayan Veri Tabanı Sistemleri

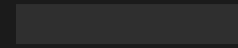
İlişkisel Veri Tabanı



Günümüzde kullanılan en yaygın sistemlerdendir.

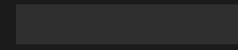


Satır ve sütundan oluşan tablolardan oluşur.

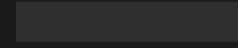


En az iki tablo gerekir.

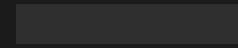
İlişkisel Olmayan Veri Tabanı



İlişkisel olmayan veri tabanları
yatay ölçeklenir.

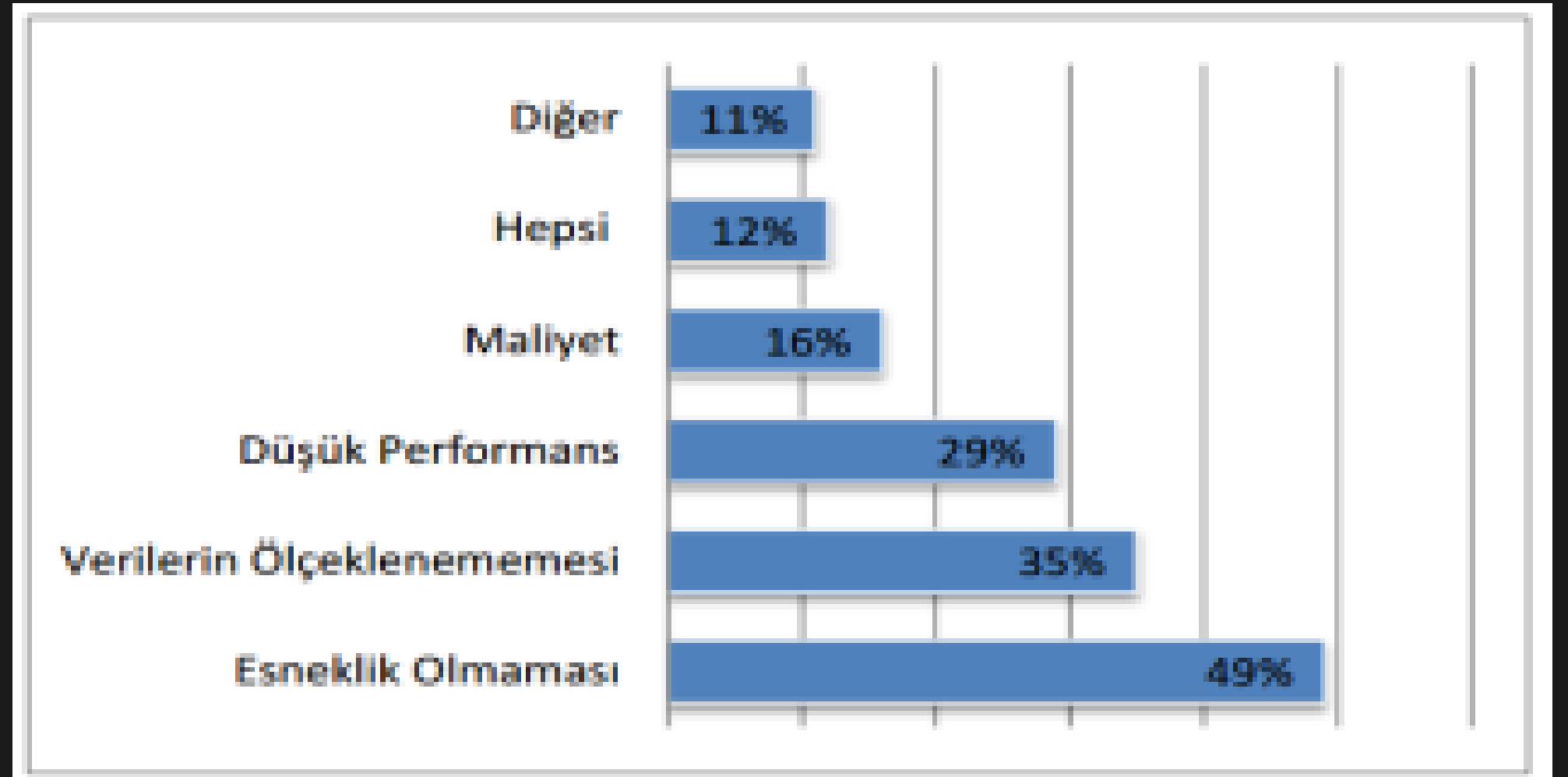


Ölçek sorununa en iyi çözümdür.



En az iki tablo gerekir.

Neden NoSQL Gerekli ?



Lider NoSQL ürünlerinin teknik karşılaştırması

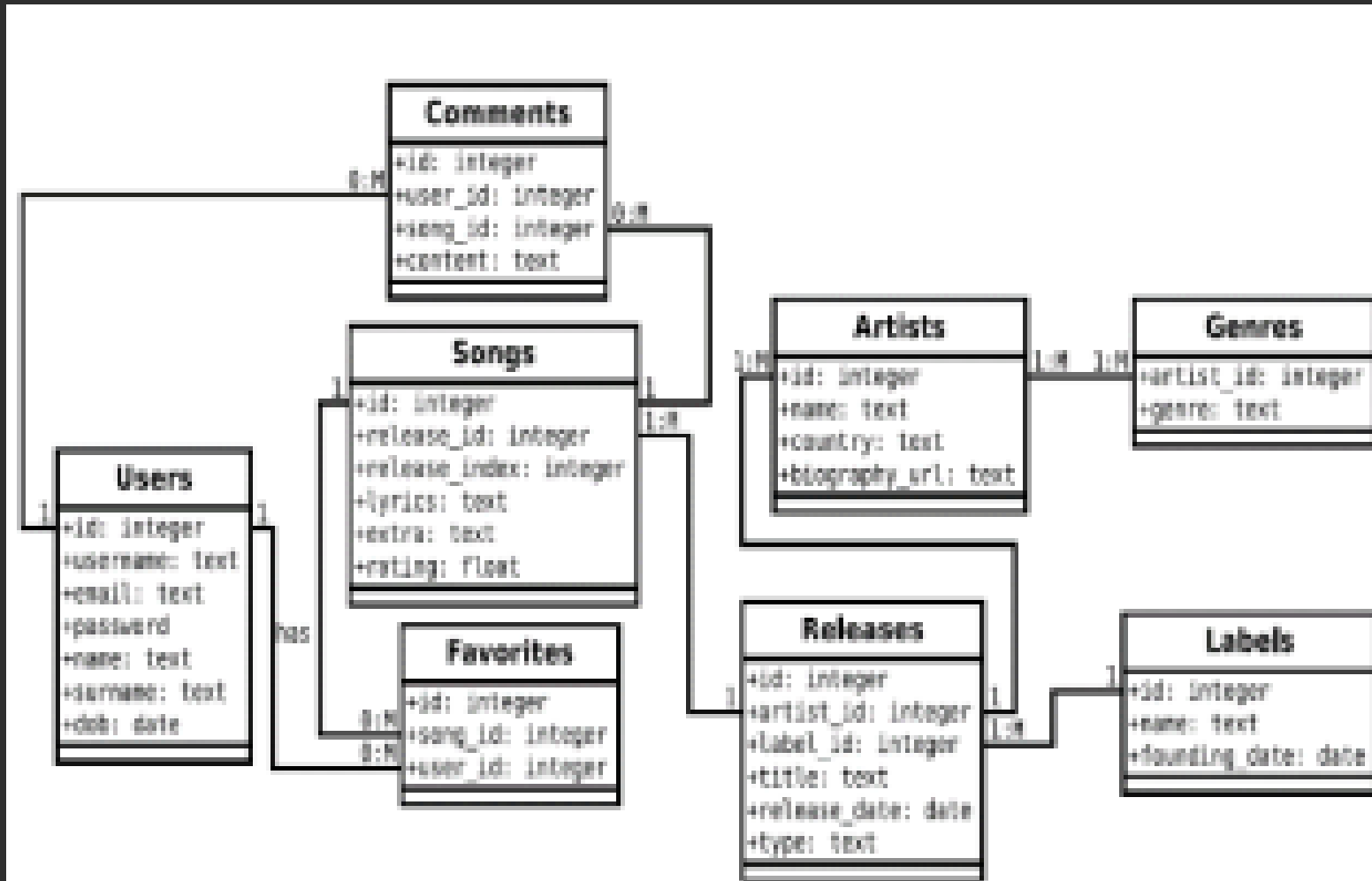
Depolama	Protokol	Model	Lisans	Dil	
Memory mapped b-trees	BSON	Document	AGPL	C++	MongoDB
COW-BTree	HTTP/REST	Document	Apache	Erlang	CouchDB
Pluggable: InnoDB, LevelDB, Bitcask	HTTP/REST or TCP/Protocol	Key/Value	Apache	Erlang	Riak
In memory, snapshot to disk	bufs TCP	Key/Value	BSD	C++	Redis
Pluggable: BSV, MySQL, in-		Key/Value	Apache	Java	Voldemort
Memtable/SStable	TCP/Thrift	Wide Column	Apache	Java	Cassandra
HDFS	HTTP/REST or	Wide Column	Apache	Java	HBase

Veri Tabanı Mimarilerinin Performans Karşılaştırması

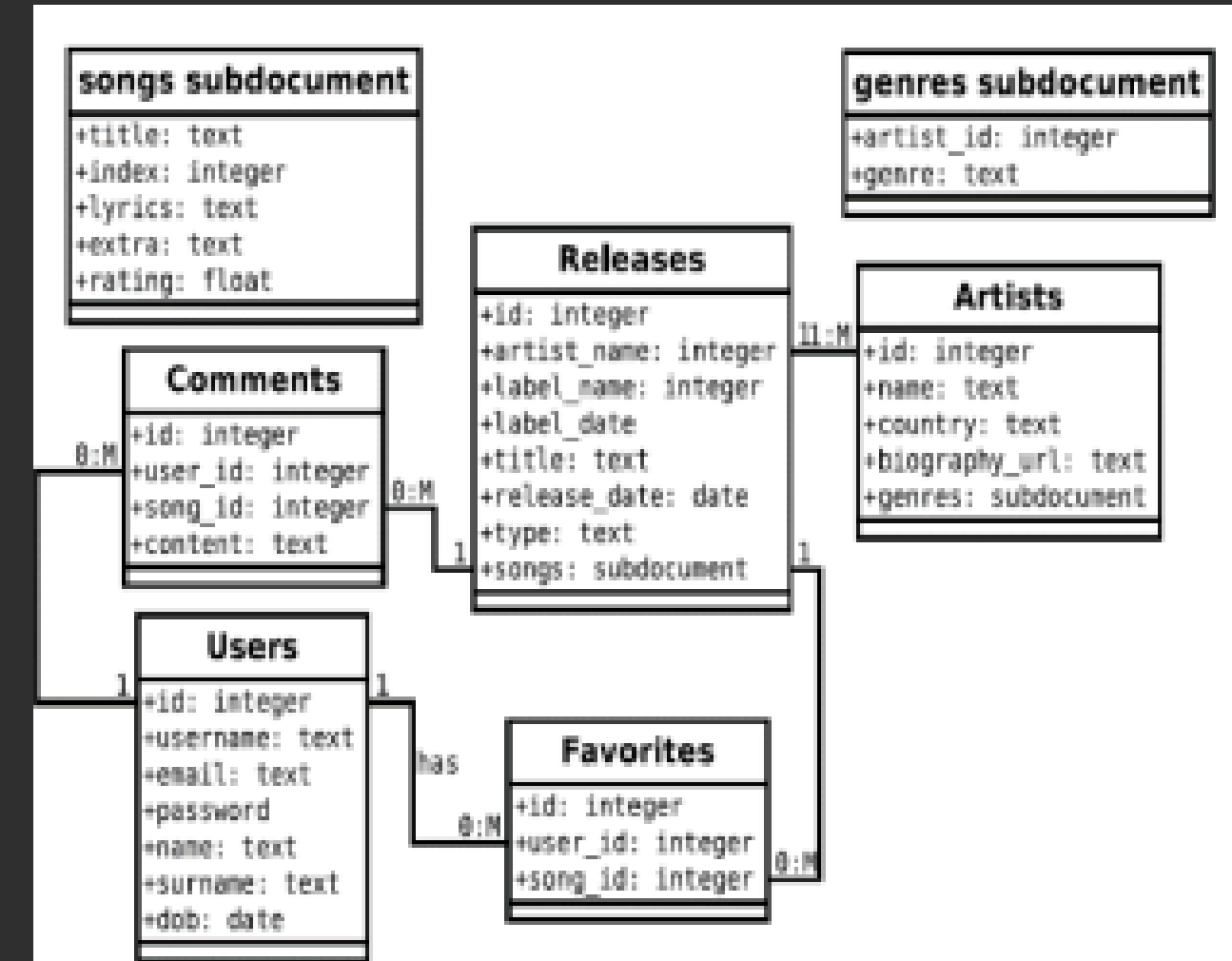
- Günümüzde ilişkisel olarak MySQL, ilişkisel olmayan ise MongoDB en çok kullanılan sistemlerdir.

Veri Tabanı Şeması

Kullanıcılara şarkılar önermek için tasarlanan algoritmaların modelidir.



Şekil 6.1 MySQL veri tabanı şeması



Şekil 6.2 MongoDB şeması

Veri Tabanı Sorguları

Üç sorgu yapılmıştır.

```
SELECT * FROM Users WHERE username = 'username '
```

Basit

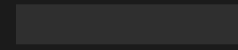
```
SELECT ' Favourites. song_id ' AS fSID , ' Favourites. user_id ' AS fUID  
FROM Favourites AS b INNER JOIN Favourites AS a  
ON b. user_id = a. user_id  
WHERE a. song_id = 123456 AND a. user_id != 987654
```

Karmaşık

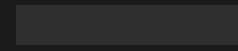
```
SELECT ' Songs. release_id ' AS sId , ' Releases. id ' AS rId  
FROM Songs INNER JOIN Releases  
ON Songs. release_id = Releases. id  
WHERE artist_id IN  
SELECT ' Genres. artist_id ' AS gAID  
FROM Genres AS c  
INNER JOIN Artists AS d  
ON c.artist_id = d.id WHERE d.name = ' artist_name '
```

Detaylı ve Karmaşık

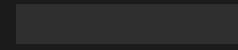
Ölçüm Sonuçları için 3 yöntem kullanılmıştır.



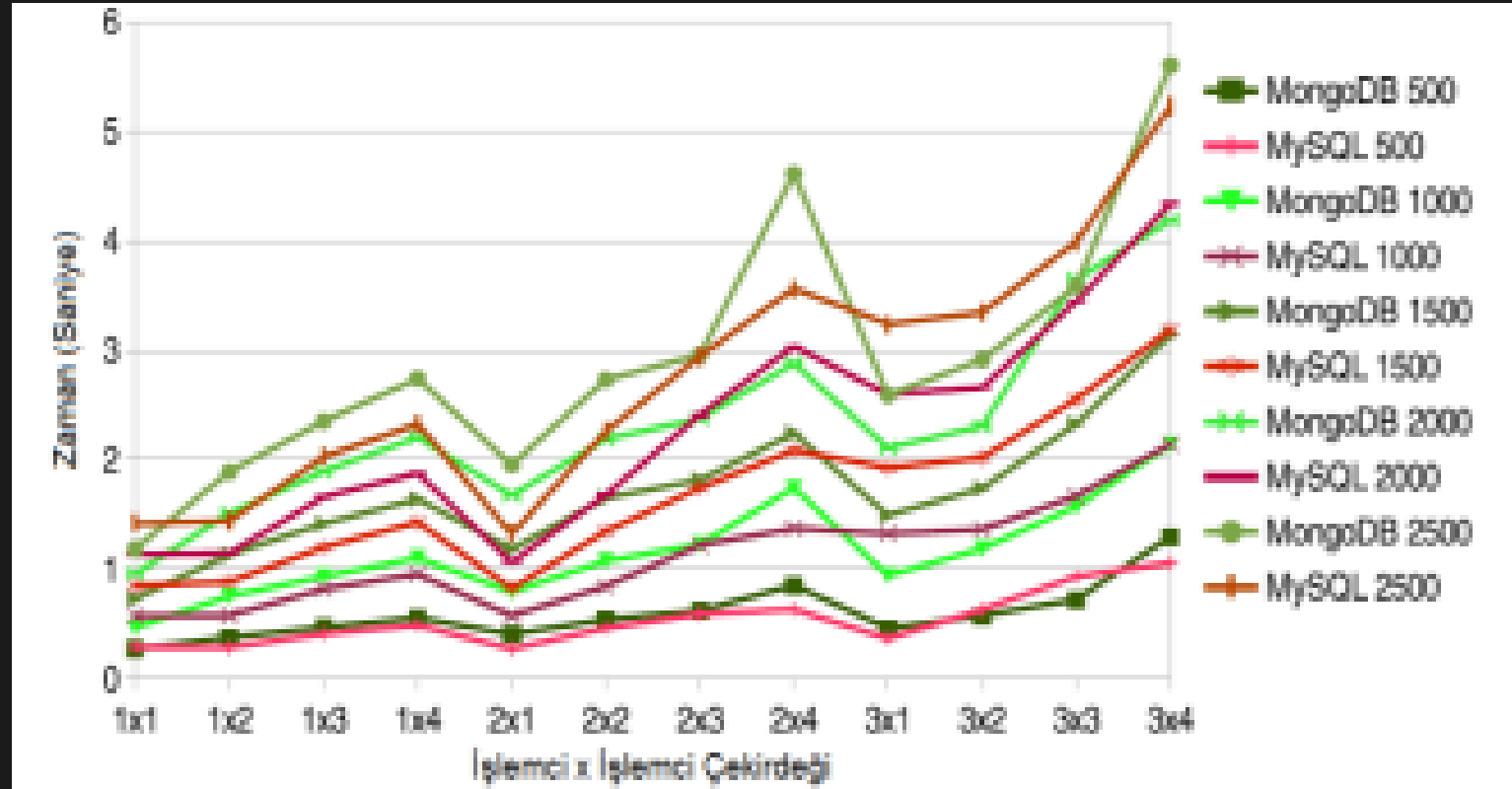
Clock fonksiyonu ile CPU
üzerindeki geçen zamana göre
ölçüm



Gettimeofday() fonksiyonu ile
hassas ölçüm



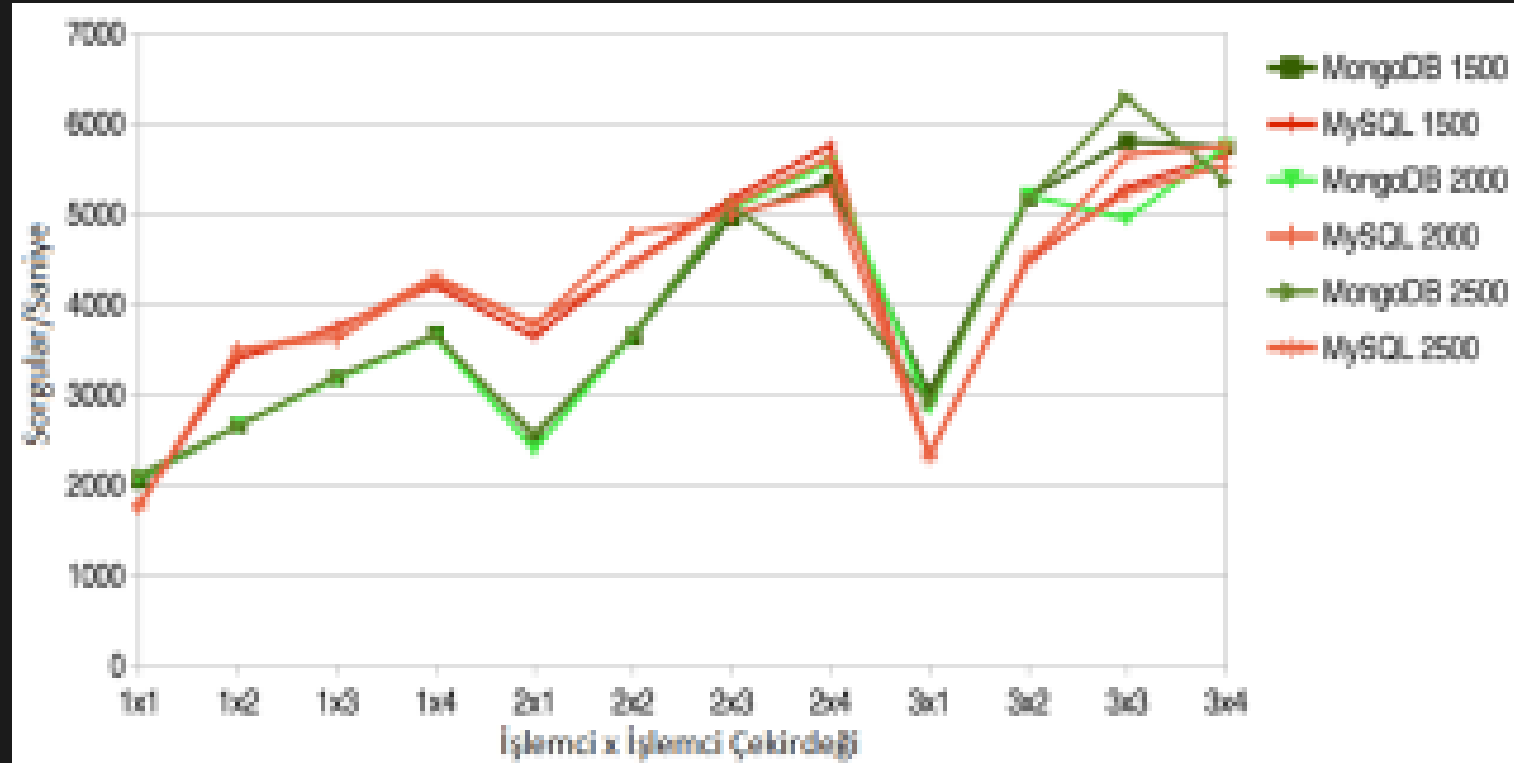
Yavaş sorgu kaydı ile ölçüm



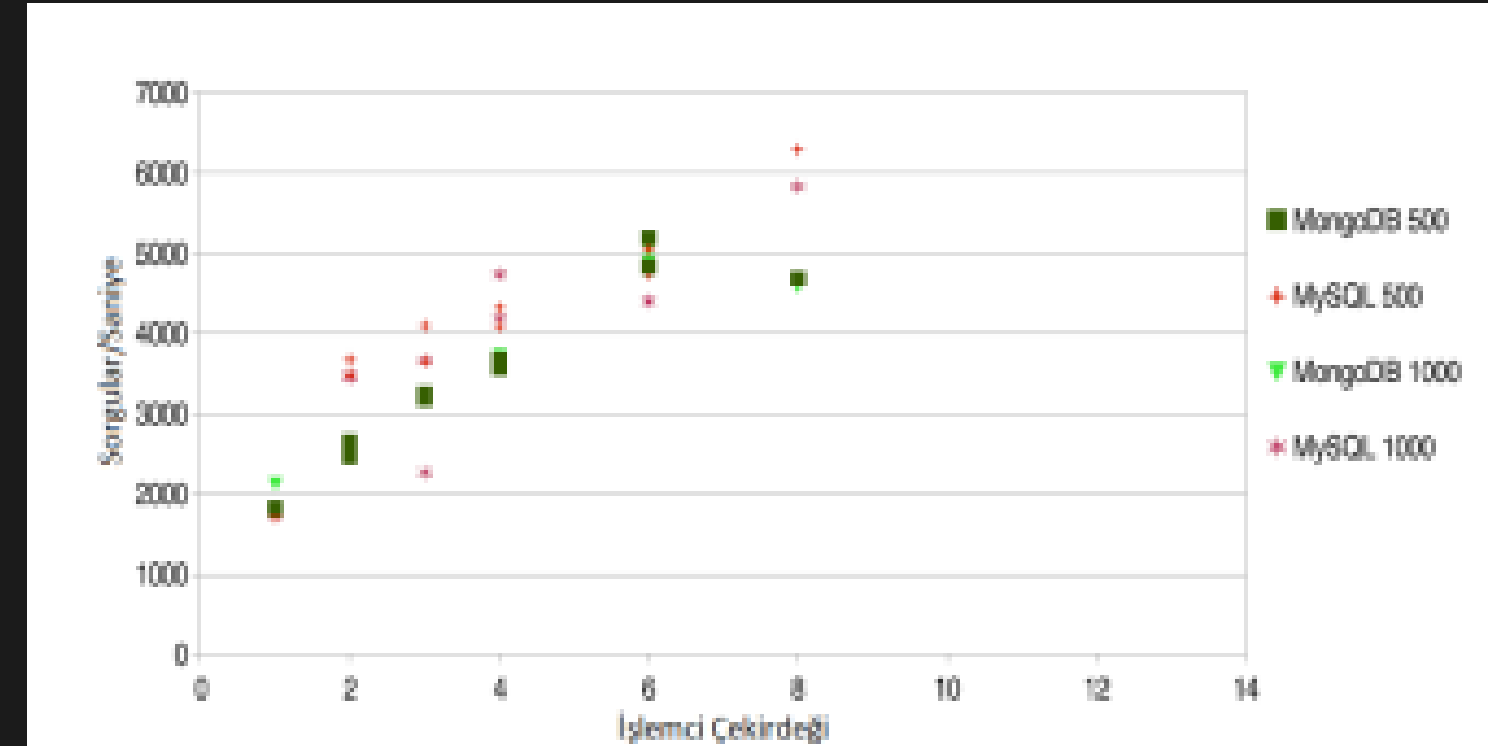
Şekil 6.3 Sorgu 1- Analiz işlemi



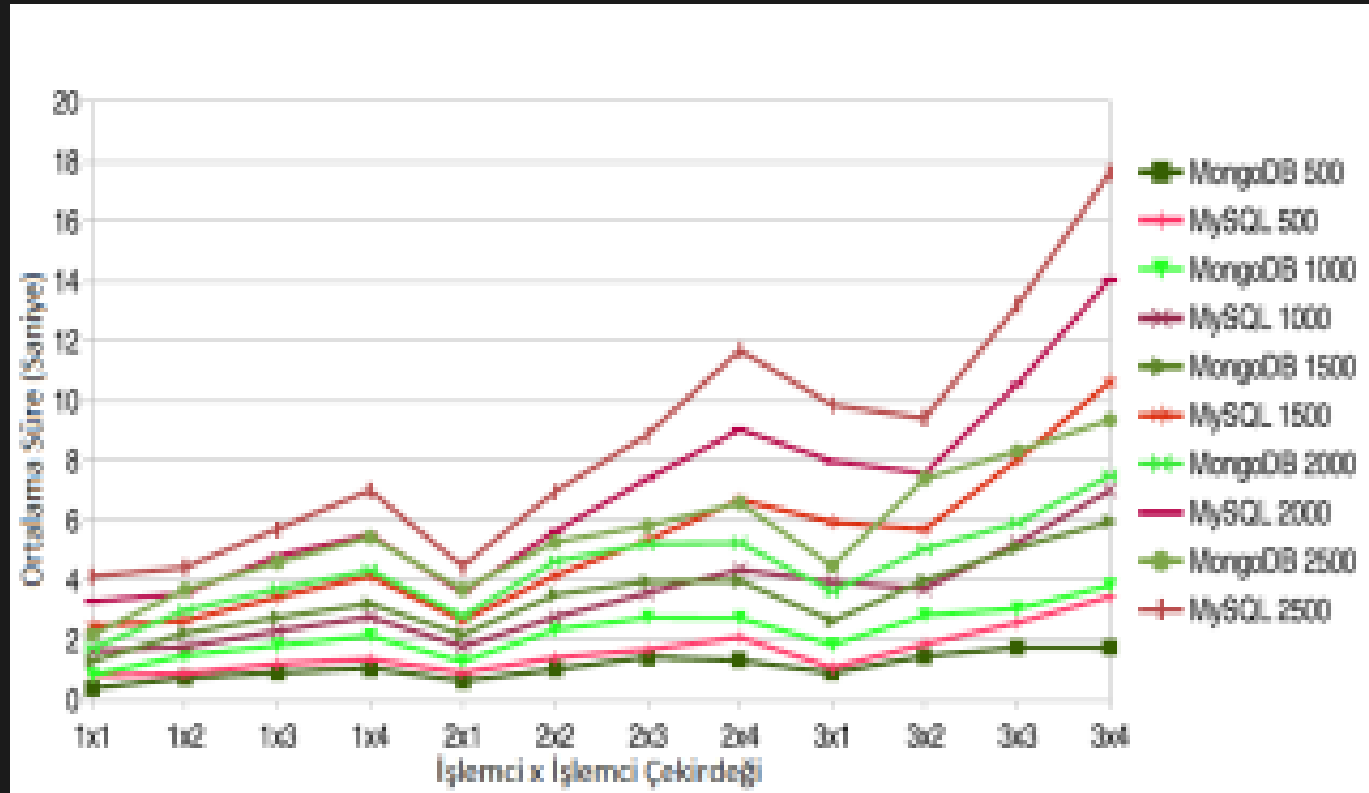
Şekil 6.4 Sorgu 1 - Sorgu/saniye analiz işlemi



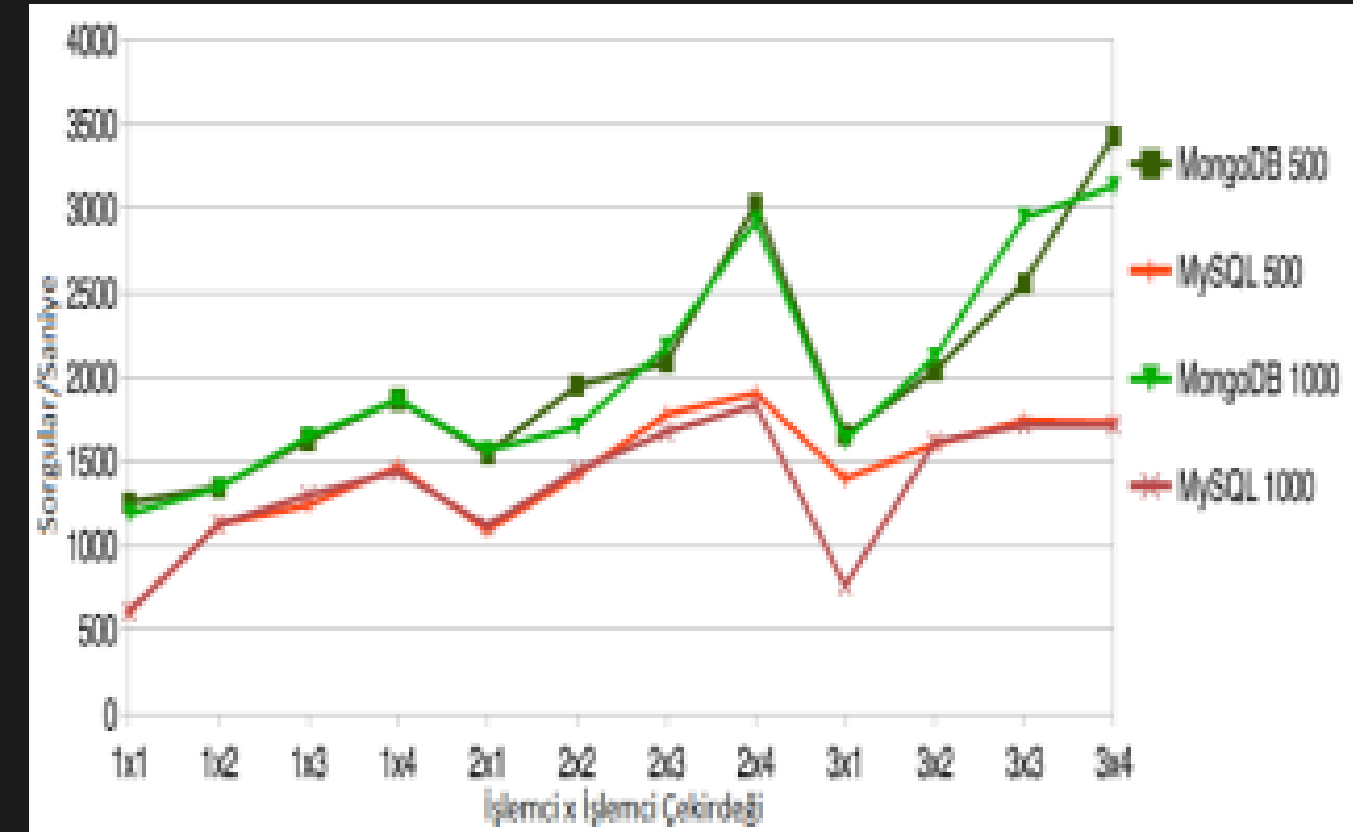
Şekil 6.5 Sorgu 1-Çok sayıdaki sorgu miktarı analiz işlemi



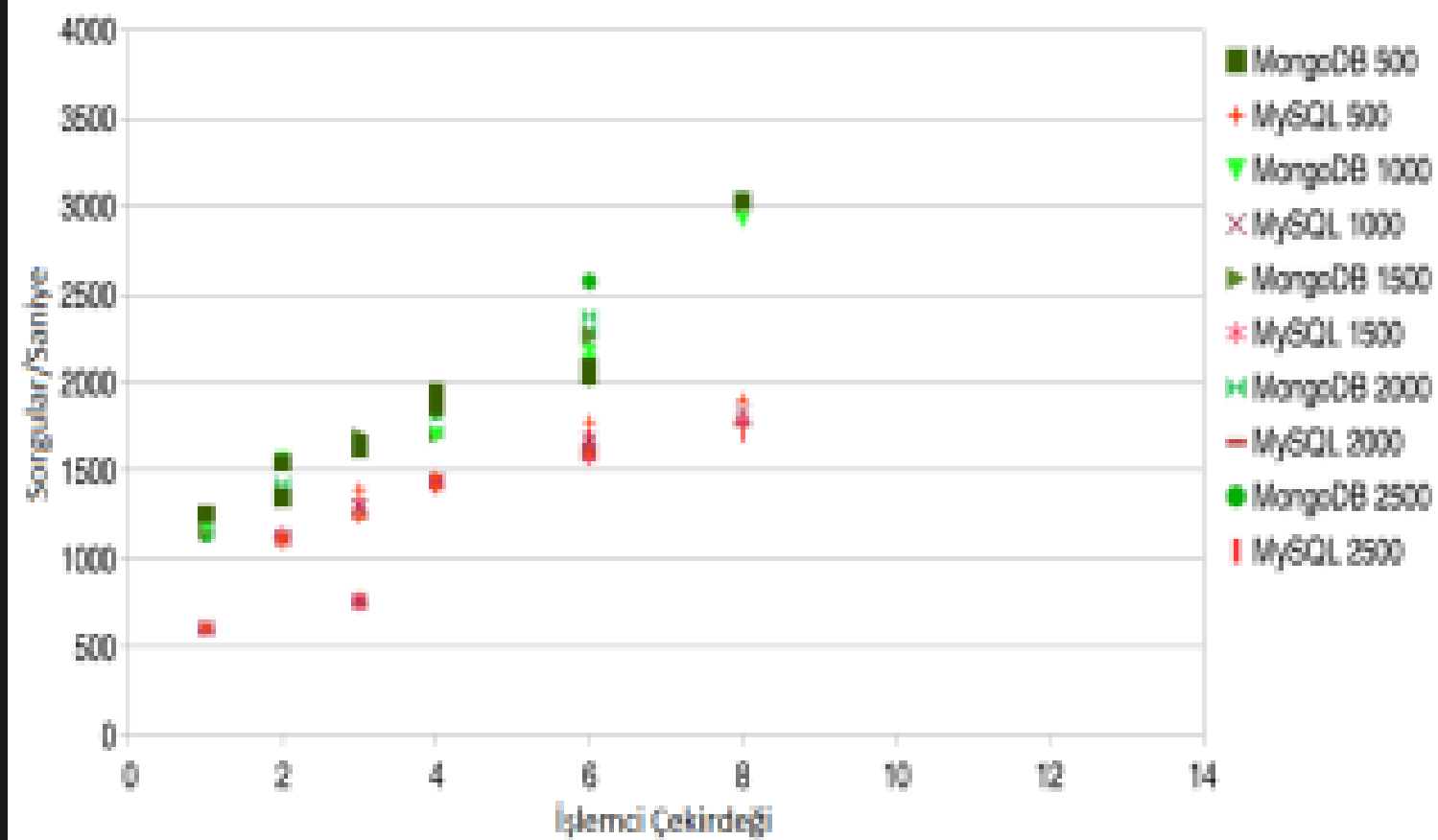
Şekil 6.6 Sorgu 1-Sorgular/Saniye ile işlemci çekirdeği miktarı için analiz işlemi



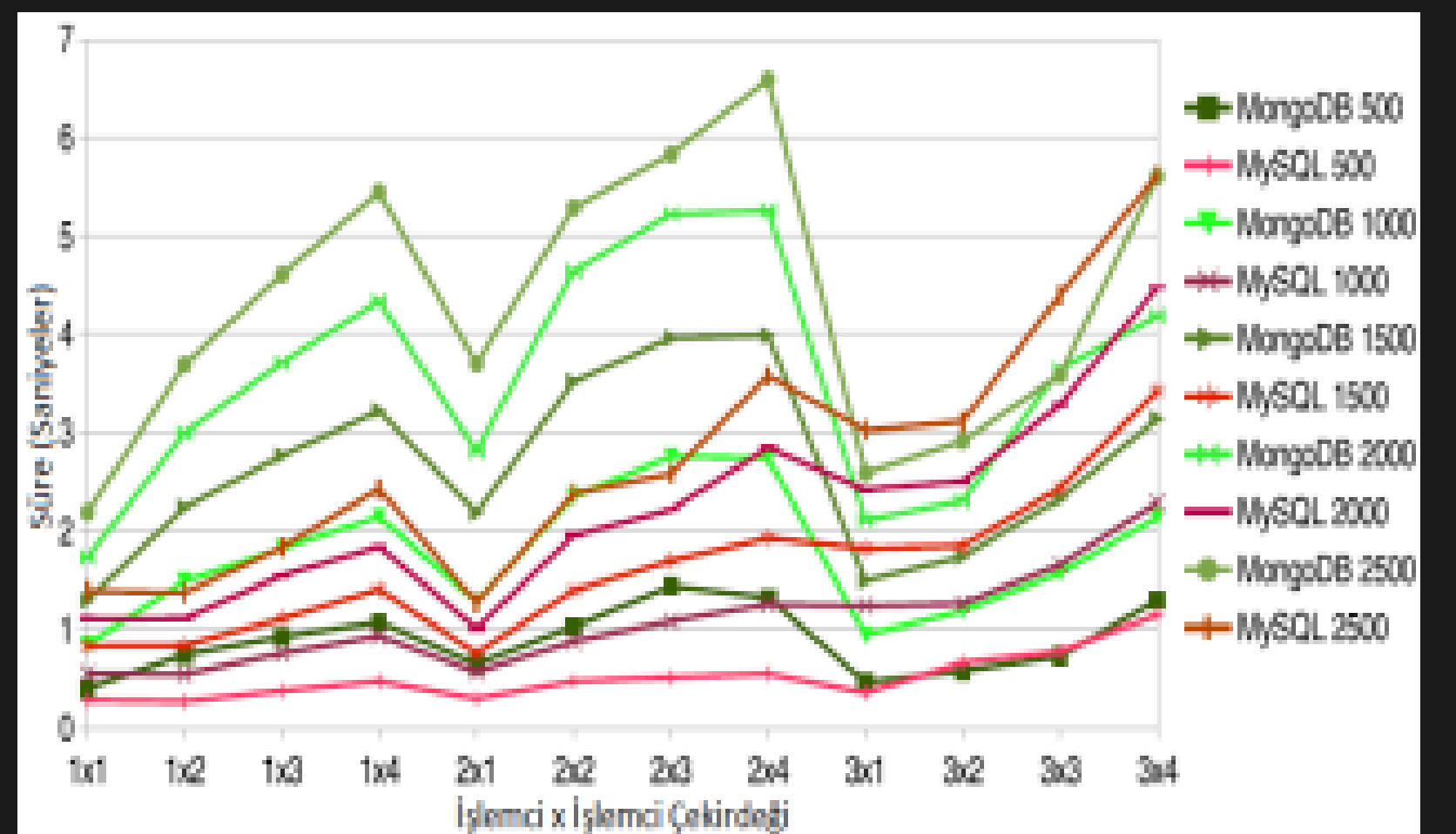
Şekil 6.7 Sorgu 2 - INNER JOIN ile karmaşık sorgu analizi işlemi



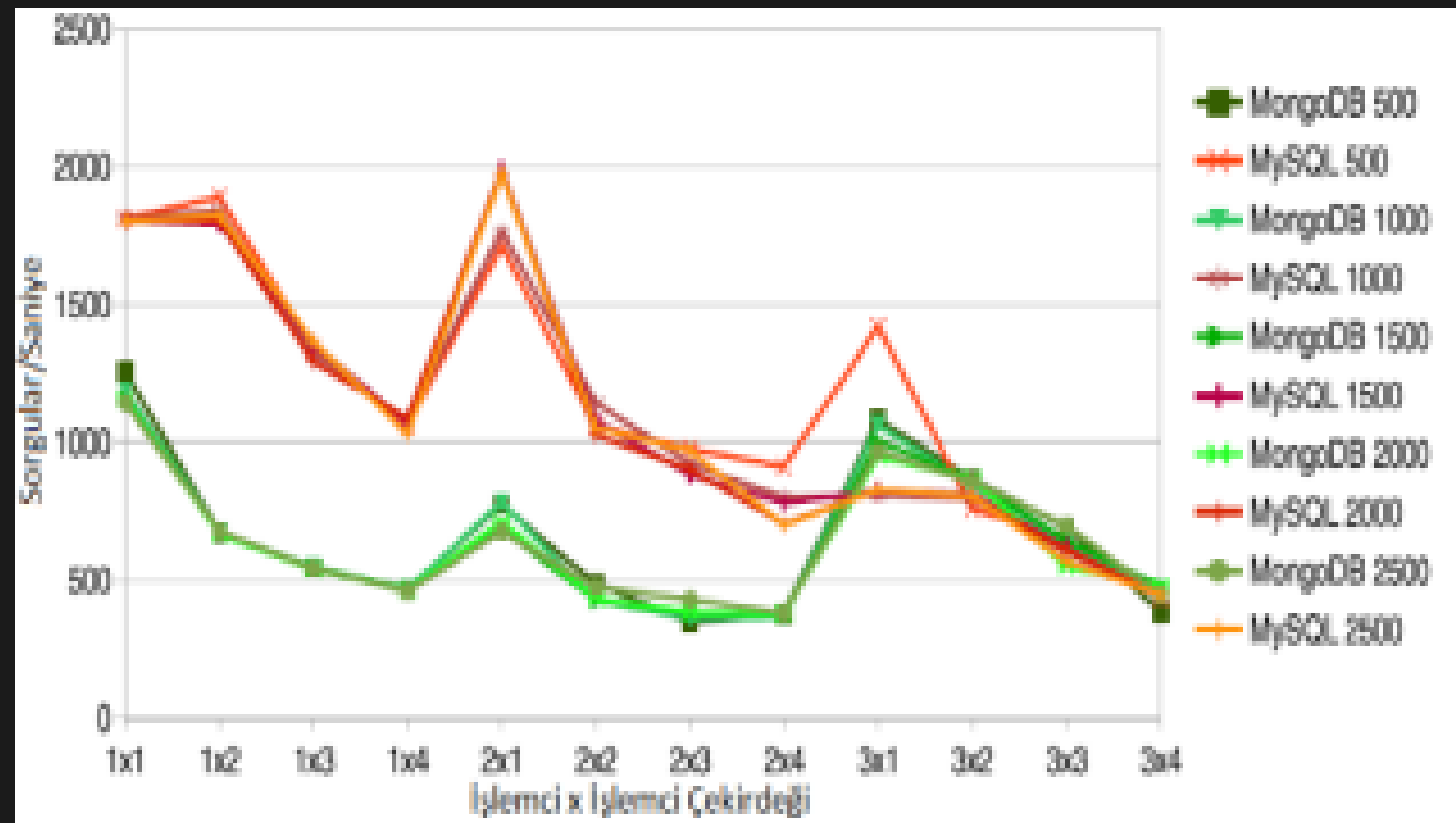
Şekil 6.8 Sorgu 2- INNER JOIN ile 500 ve 1000 veri için sorgu analizi işlemi



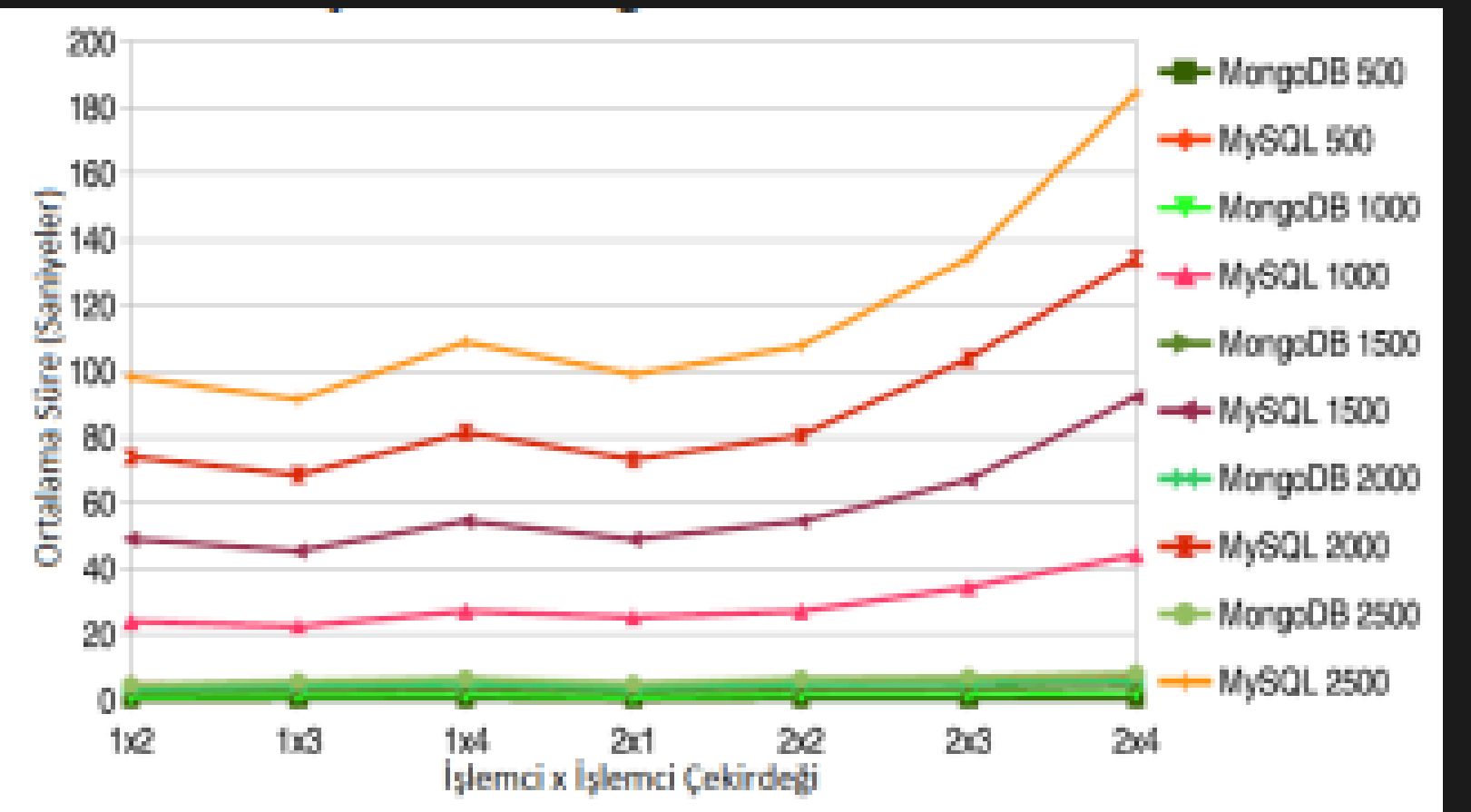
Şekil 6.9 Sorgu 2- INNER JOIN ile işlemci çekirdeği miktarı üzerinde analiz işlemi



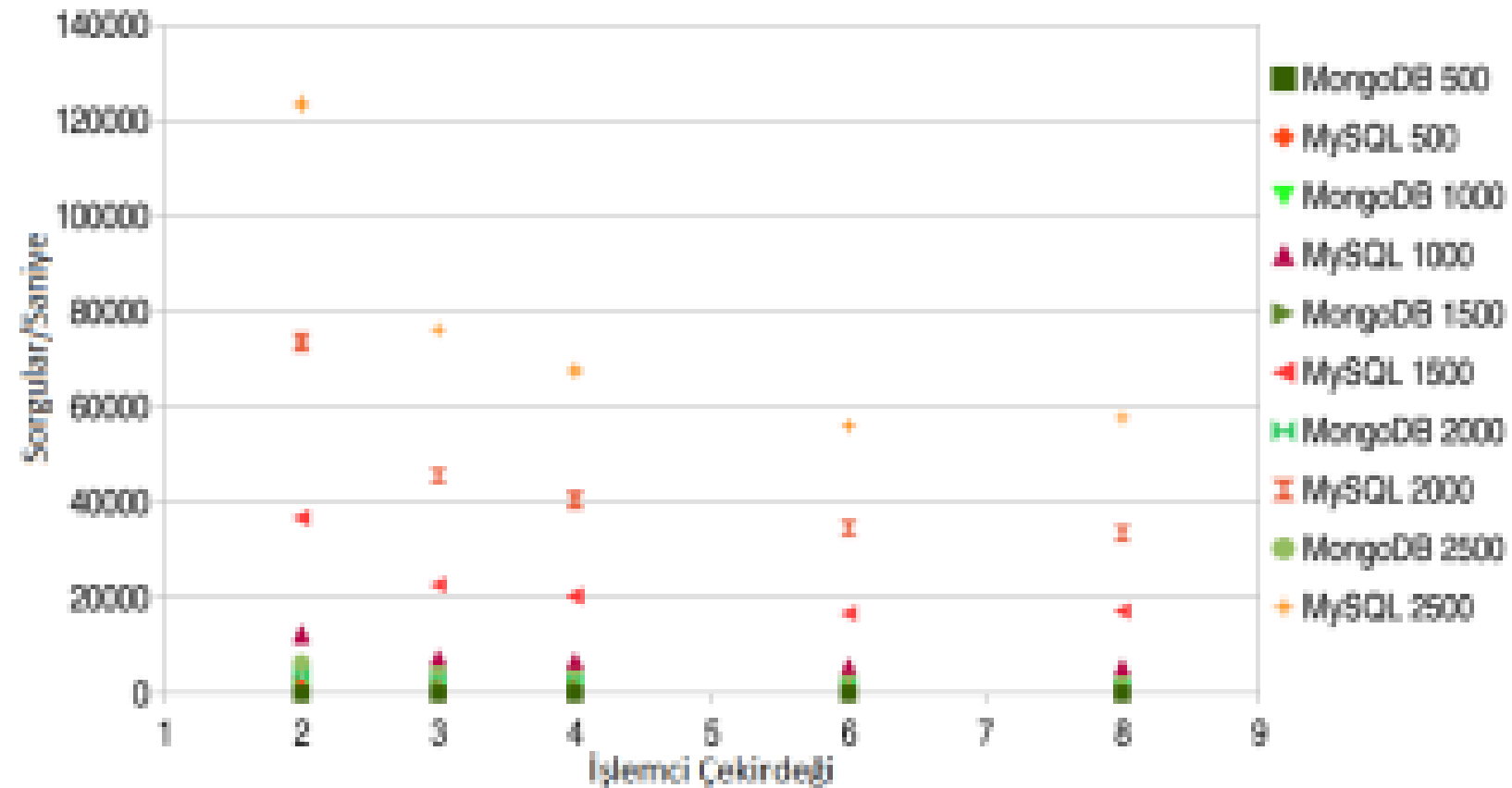
Şekil 6.10 Sorgu 3 – Detaylı karmaşık sorgu süre analizi



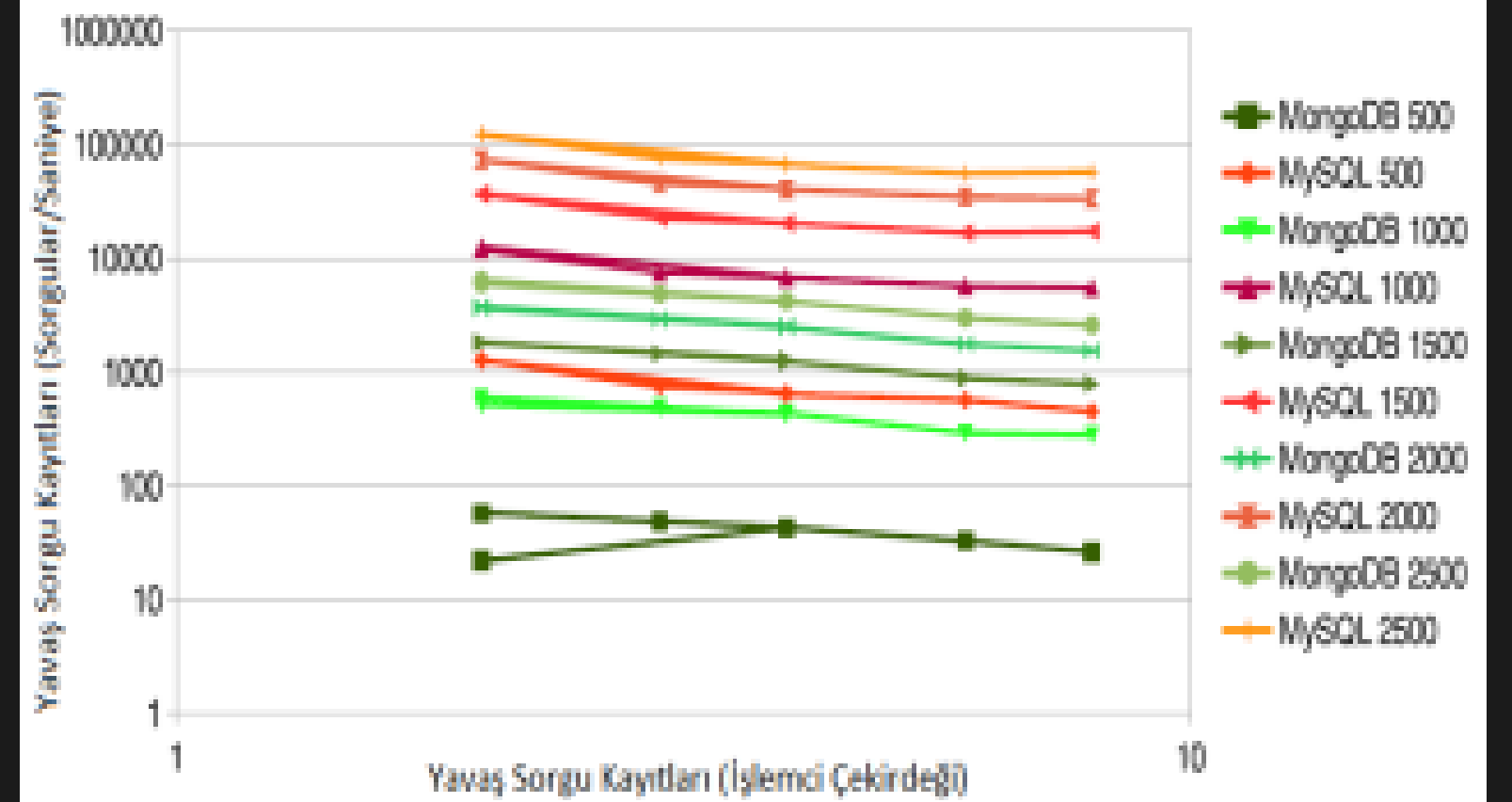
Şekil 6.11 Sorgu 3- Detaylı ve karmaşık sorgu ile Sorgular/saniye analiz işlemi



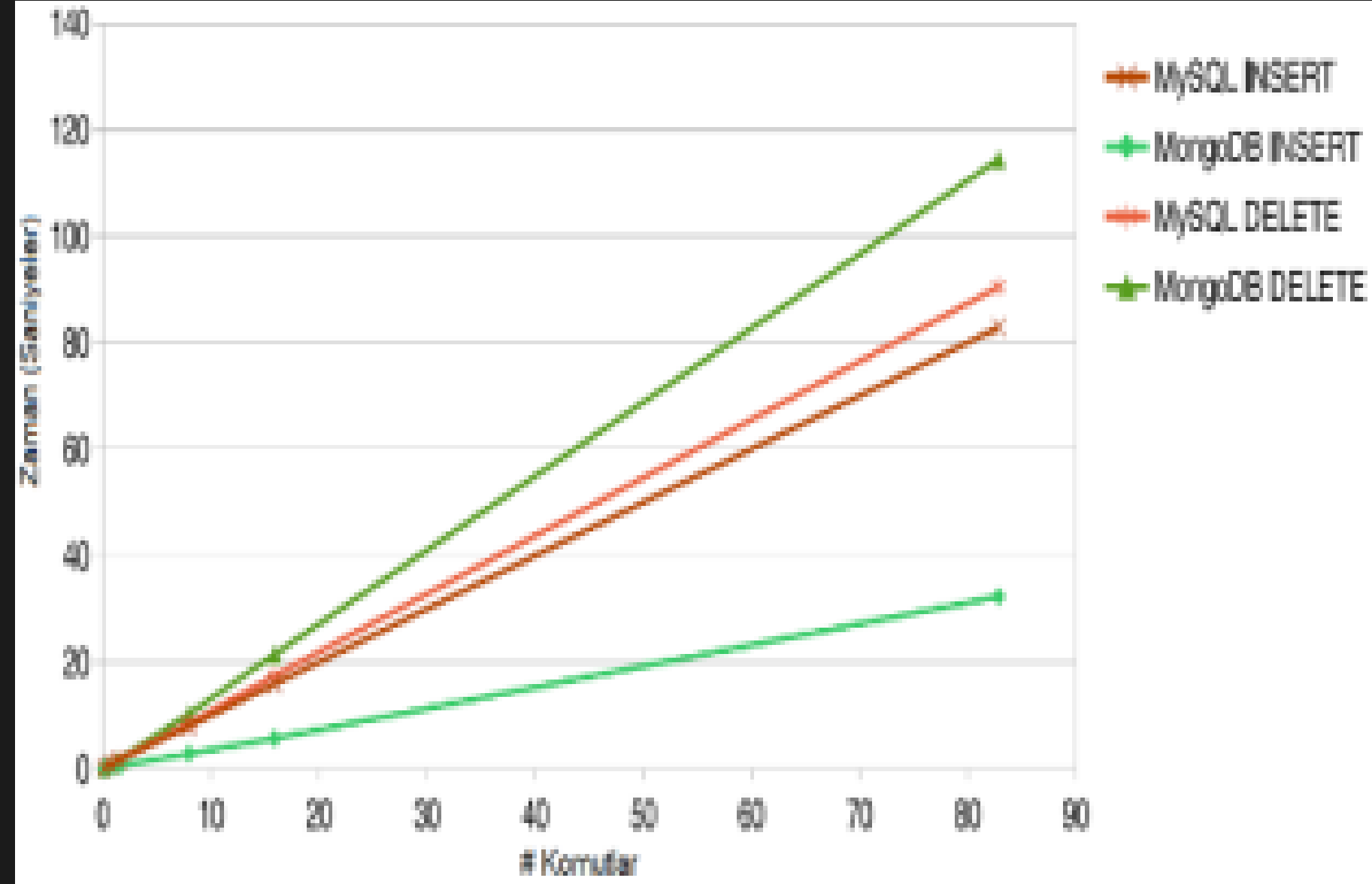
Şekil 6.12 Sorgu 3 – Detaylı ve karmaşık sorgu kodu ile ortalama süre analiz işlemi



Şekil 6.13 Sorgu 3 - Detaylı ve karmaşık sorgu ile işlemci çekirdeği üzerinde analiz işlemi



Şekil 6.14 Sorgu 3- Detaylı ve karmaşık sorgu ile ölçeklendirilmiş analiz işlemi



Şekil 6.15 INSERT ve DELETE işlemleri