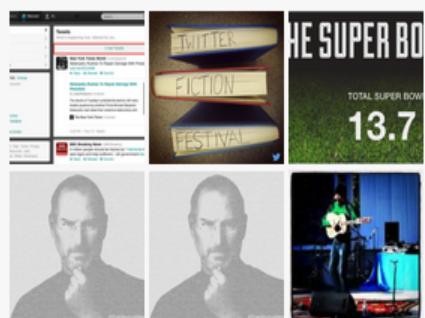


# Gerçek Zamanlı Web Uygulamaları

Özgur Web Günleri 2013

[@emrahayanoglu](#) / STM A.Ş

[Tweets](#)[Following](#)[Followers](#)[Favorites](#)[Lists](#)[Tweet to Twitter](#)**Twitter**

@twitter

Your official source for news, updates and tips from Twitter, Inc.

San Francisco, CA · <http://blog.twitter.com/>1,468  
TWEETS1,306  
FOLLOWING14,325,402  
FOLLOWERS

Follow

**Tweets** All / No replies**Twitter** @twitter

12h

RT @TwitterEng: Bolstering our infrastructure. "As usage patterns change, Twitter can remain resilient."

[engineering.twitter.com/2012/11/bolste...](http://engineering.twitter.com/2012/11/bolste...)

Expand

**Twitter** @twitter

14h

Have you checked Discover lately? A new notification at the top of your stream shows when new Tweets are available.

[pic.twitter.com/dL2NYafx](http://pic.twitter.com/dL2NYafx)[View photo](#)





Dashboards

My Site ▾

My Conversions ▾

Custom Reports

Google Store

www.googlestore.com - http://www.googlestore.com ... ▾



Real-Time (beta)

Overview

Locations

Traffic Sources

Content

Dashboards

Help

Real-Time Reports

Search help center

## Overview

Right now

32

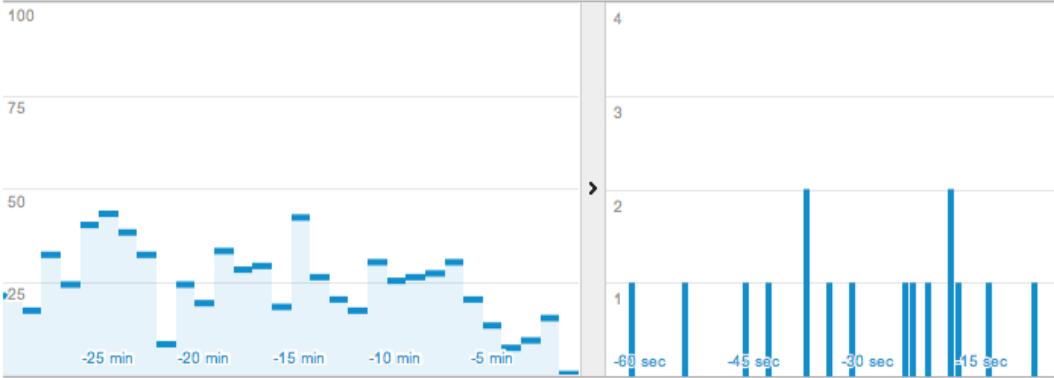
active visitors on site

NEW RETURNING

91%

9%

### Pageviews



### Top Referrals:

	Source	Active Visitors ↓
1.	google.com	10
2.	google.com.br	2
3.	google.ca	1
4.	google.com.ar	1
5.	google.com.tw	1
6.	google.com.vn	1
7.	google.pl	1

### Top Active Pages:

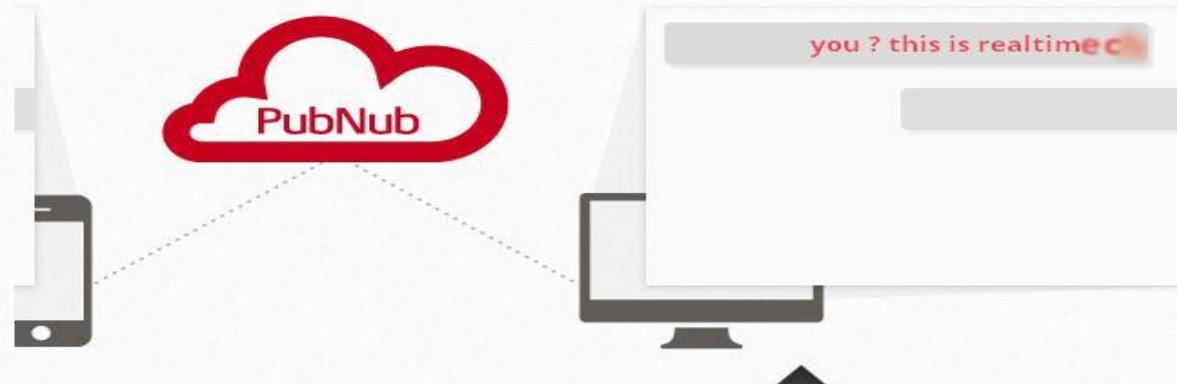
	Active Page	Active Visitors
1.	/	11 34.38%
2.	/googlesearch.aspx?category=doodles	7 21.88%
3.	/shop.axd/Home	3 9.38%
4.	/googlesearch.aspx?category=kids	2 6.25%
5.	/Accessories/Chalk+It+Up+Ceramic+Mug.axd	1 3.13%
6.	/Fun/Eco+Droid+Squishable.axd	1 3.13%
7.	/Specials/	1 3.13%

We're Hiring!

## See the speed of PubNub for yourself.

Enter the textbox below to see just one example of PubNub's real-time action.

Also try it on your mobile device!



## Power your app with PubNub

Hundreds of apps and thousands of developers are using PubNub to deliver "human-perceptive" real-time experiences to millions of users worldwide. PubNub delivers the real-time infrastructure needed to build amazing MMO games, social collaborative solutions, and more.

[GET STARTED](#)[What is PubNub](#)[Demos](#)[Your App](#)[Node.js](#)[Try It Now!](#)

```
// listen for events
PUBNUB.subscribe({
    channel : "hello_world",
    callback : alert
})
```

```
// send events
PUBNUB.publish({
    channel : "hello_world",
    message : "data"
})
```

[GET STARTED](#)

Free Testing & Development

WE ARE  
HIRING!

# Scalable real-time backend

Build apps fast without managing servers

A real-time chat app in 14 lines  
of code. [Try it >](#)

```
var messagesRef = new Firebase('https://chat.firebaseioIO.com/');

// Write messages.
messagesRef.push({name:"Alice",text:"Firebase is awesome!"});

// Read messages.
messagesRef.on("child_added", function(message) {
  renderMessage(message.val());
});
```

[Build a Sample App in 5 minutes](#)

Our Customers:



Build  
Fast



Completely  
Real-time



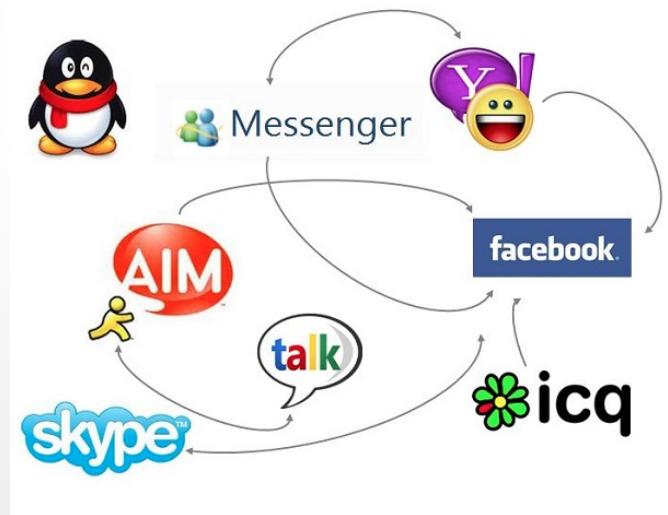
Servers  
Optional

# Ajanda

- Gerçek Zamanlı Uygulama nedir?
- Gerçek Zamanlı Uygulamaların Temel Özellikleri
- Gerçek Zamanlı Uygulamaların Genel Mimarisi
  - WebSocket - EventSource - Ajax Long Polling (Comet) - Flash Socket
  - Non-Blocking I/O vs Blocking I/O
  - Mesaj Kuyruğu Yapıları (Message Queue)
    - XMPP
    - AMQP
    - ØMQ
- Gerekli Açık Kaynak Kodlu Kütüphaneler
  - Java
  - Python
  - Ruby
  - Node.JS
- Sonuç

# Gerçek Zamanlı Uygulama Nedir?

Çok kısa bir zaman aralığında kullanıcılar cevap vermek zorunda olan uygulamalardır.

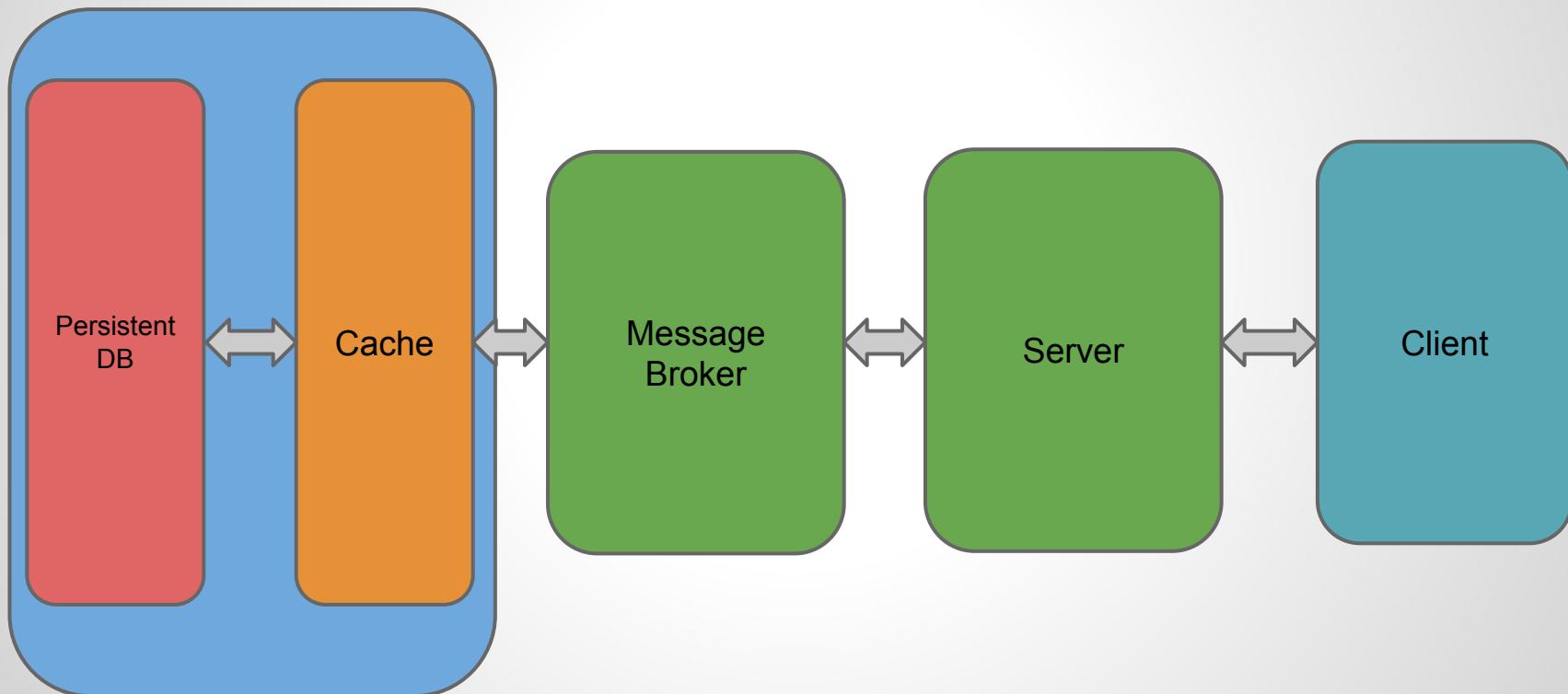


# Gerçek Zamanlı Uygulamaların Temel Özellikleri

Low-Latency



# Gerçek Zamanlı Uygulamaların Genel Mimarisi



# Client - WebSocket

- HTML 5'in en önemli yeniliklerinden
- AJAX'in önemini kazandıktan sonra, HTTP'inin ek yük getirdiği anlaşılırak geliştirildi.
- WebSocket'in desteklemediği Browser'lar için [web-socket-js](#) kütüphanesi kullanılabilir.

```
var connection = new WebSocket('ws://html5rocks.websocket.org/echo');
connection.onopen = function () {
  connection.send('Ping'); // Ping Mesajı Yolla
};
// Hata Mesajları
connection.onerror = function (error) {
  console.log('WebSocket Error ' + error);
};
// Server'dan gelen Mesajlar
connection.onmessage = function (e) {
  console.log('Server: ' + e.data);
};
```

# Client - EventSource

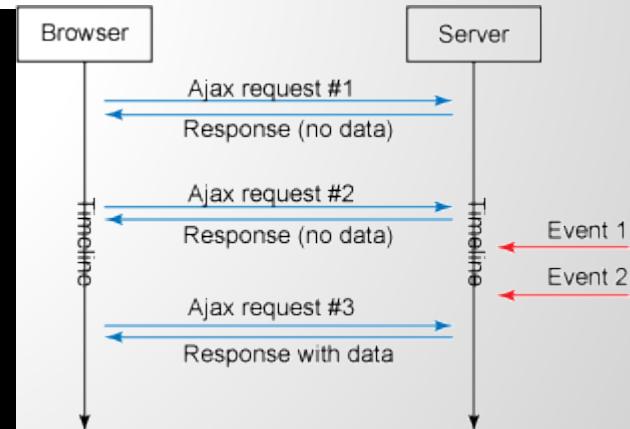
- AJAX Long Polling'de görülen sorunlar üzerine geliştirildi.
- WebSocket'ten farklılık olarak sadece Server tarafından gelen data'lar için kullanılabilir ve HTTP protokolü üzerinde geliştirilmiştir.

```
if (!!window.EventSource) {
  var source = new EventSource('/stream');
} else { // AJAX Long Polling implement edilmeli
}
source.addEventListener('message', function(e) {
  console.log(e.data);
}, false);
source.addEventListener('open', function(e) {
  // Baglanti kuruldu
}, false);
source.addEventListener('error', function(e) {
  if (e.readyState == EventSource.CLOSED) { // Baglanti kapandi
  }
}, false);
```

# Client - Ajax Long Polling (Comet)

- Birbiri ardına açılan AJAX istekleri ile oluşturulur.
- AJAX'ten beri kullanılan gerçek zamanlı istemci tercihidir.

```
function long_polling() {  
    $.getJSON('ajax', function(events) {  
        processEvents(events);  
        long_polling(); // Recursive Ajax Calls  
    });  
}  
  
long_polling();
```



# Client - Flash Socket

- Özellikle Flash uygulamalarının tercih ettiği gerçek zamanlı istemci kütüphanesidir.
- Flash-Javascript bağlantısını kullanarak, Pure Javascript ile bağlantı sağlanabilir



# Non-Blocking I/O vs Blocking I/O



X



# Non-Blocking I/O vs Blocking I/O

- Örnekte de görüldüğü üzere, Non-Blocking I/O ile Low-Latency özelliği sağlanmaktadır.
- Blocking I/O ise Low-Latency özelliğini garanti vermeden, bütün isteklere cevap vermek ister.
- Bu nedenle Gerçek Zamanlı Web Uygulamalarında Server tercihimiz Non-Blocking I/O ile olmalıdır

# Message Queue

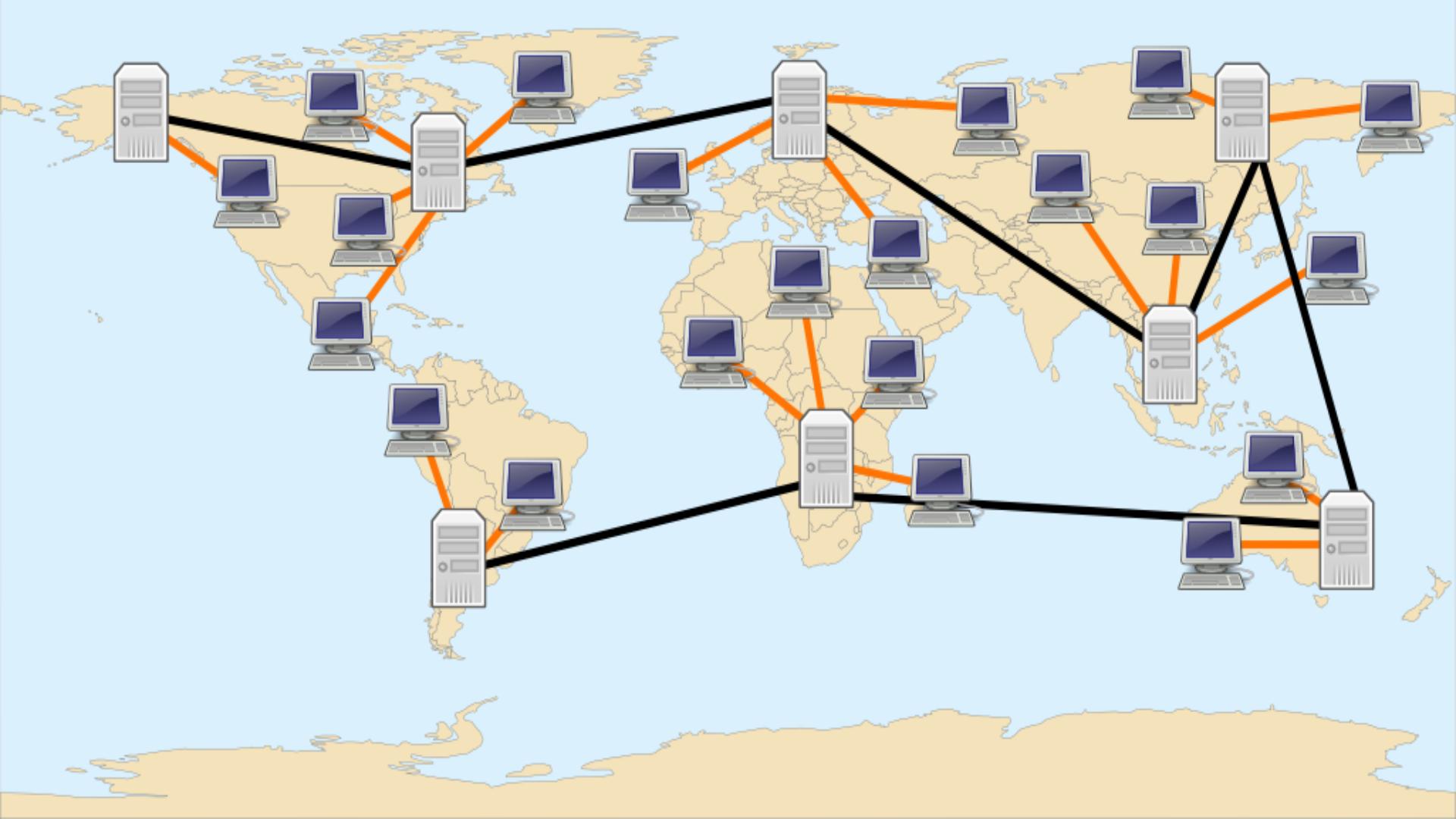
Mesaj Kuyruğu, uygulamalar, process'ler ve thread'ler arasındaki mesajların yönetimini gerçekleştiren yapılardır.

Mesaj Kuyruğunu, şu nedenlerden dolayı kullanmak gereklidir:

- Decoupling (Sistemler arası ayırım)
- Scalability (Ölçeklenebilirlik)
- Mesaj Ulaştırma Garantisi
- Mesaj Teslim Etme Garantisi
- Buffering (Mesaj'ları arabelleğe alma)
- Asynchronous Communication (Asenkron İletişim)
- Message Routing (Mesajların Yönlendirilmesi)

# Message Broker - XMPP

- XMPP(Extendable Messaging and Presence Protocol) XML mesaj tabanlı uygulamalar için haberleşme protokolüdür
- En Önemli Özellikleri; Merkezi Olmayan, Genişletilebilir ve Güvenli olmalıdır.
- Ayrıca Presence (İstemcilerin varlıklarının belirtilmesi) özelliği ile de istemcilerin durumlarının sürekli görülebilmesi amaçlanmıştır.
- VOIP, Video Streaming, File Transfer, Multiplayer Oyun gibi uygulamalarda oldukça kullanışlıdır.
- Gtalk, Facebook Chat, Gabber, Ovi (Nokia) gibi anlık mesajlaşma uygulamalarının temelini oluşturur.
- [Openfire](#), [ejabberd](#), [Prosody](#) gibi açık kaynak kodlu XMPP sunucuları bulunmaktadır.



# Message Queue - AMQP

Genel olarak, mesajlaşma ara katmanı için düzenlenmiş olan, açık bir standarttır. Mesaj Kuyruklarının şu özelliklerini tanımlar:

- Kuyruk Yapısını
- Yönlendirme Detaylarını
- Veri Güvenilirliğini

Protokol, tamamen Açık Kaynaklı olarak geliştirildiğinden, Açık Kaynaklı implementasyonları görebiliriz:

# Message Queue - AMQP



The logo for Qpid features the word 'Qpid' in a large, bold, black sans-serif font. A colorful feather graphic is placed on top of the letter 'Q', with its tip pointing towards the bottom right.

# Message Queue - ØMQ

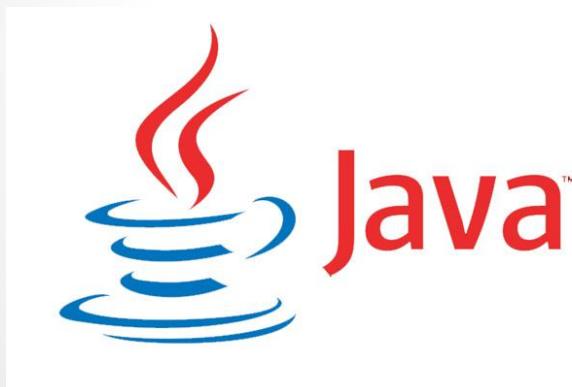
- Aslında bir Socket kütüphanesi olarak görülebilir. Ancak kütüphane üzerinde gerçekleştirilecek işlemler sonucunda Message Queue olarak kullanılabılır.
- AMQP ile karşılaştırıldığında daha istenilen özelliklere sahip olunabilir.
- Mesajların, Process içerisinde, Process'ler arasında ve TCP üzerinde iletilmesini sağlayabilir.
- Mesajları, fanout, pubsub, pipeline ve request-reply şekilleriyle iletебilir.
- Mesajların ve Socket'in kontrolü ZeroMQ sayesinde elinizde olur.
- Bu nedenle, hem AMQP'ye göre hemde XMPP'ye göre daha ölçeklenebilirdir. Ancak kontrolü, diğerlerine nazaran daha zordur.
- Yaklaşık 30 programlama dilinde implementasyonu mevcuttur.



# Gerekli Açık Kaynaklı Kodlu Kütüphaneler

Gerçek Zamanlı Uygulamaların implementasyonu için Açık Kaynaklı projeler olmazsa olmazdır. Sırasıyla diller:

- Java
- Python
- Ruby
- Node.JS



# Java

- **Non-Blocking I/O Servers:** Netty, Apache Mina, Grizzly, v.b
  - Netty Echo Server Kod Örneği
- **XMPP Client:** Smack
- **RabbitMQ (AMQP) Client:** RabbitMQ-Java-Client
- **ØMQ Client and Server:** JeroMQ

# Python

- **Non-Blocking I/O Servers:** Tornado, Twisted, Eventlet, Gevent
- **XMPP Client:** SleekXMPP
  - SleekXMPP Echo Client Kod Örneği
- **AMQP Client:** Pika
- **ØMQ Client and Server:** pyzmq

# Ruby

- **Non-Blocking I/O Servers:** EventMachine, Goliath
- **XMPP Client:** xmpp4r, blather
- **AMQP Client:** amqp
  - Amqp Server Kod Örneği
  - Amqp Client Kod Örneği
- **ØMQ Client and Server:** rbzmq

# Node.JS

- **Non-Blocking I/O Servers:** Node.JS(V8) zaten doğuştan Non-Blocking I/O Yapısındadır. `WebSocket(ws)`
- **XMPP Client:** `node-xmpp`, `simple-xmpp`,
- **AMQP Client:** `node-amqp`
- **ØMQ Client and Server:** `node-zeromq`
  - Request Kod Örneği
  - Reply Kod Örneği

# Sonuç

- Gerçek Zamanlı Web Uygulamaları, artık web dünyasının vazgeçilmezlerindendir.
- WebSocket'in gelişimi Gerçek Zamanlı Web Uygulamalarını'da geliştirmesi ve dönüştürülmesini sağlayacaktır.
- Non-Blocking I/O Server'lar Gerçek Zamanlı Web Uygulamaları için en iyi çözümüdür.
- Mesaj Kuyruk yapıları, geliştirilecek olan uygulamanın özelliklerine göre, ihtiyaca göre tercih edilebilir.

# **Sorular!!!**

# Teşekkürler



@emrahayanoglu